

جهت خرید فایل word به سایت www.kandooon.com مراجعه کنید
یا با شماره های ۰۹۳۶۶۰۲۷۴۱۷ و ۰۹۳۶۶۴۰۶۸۵۷ و ۰۶۶۴۱۲۶۰-۰۵۱۱ تماس حاصل نمایید

موضوع :

علم کامپیوتر

شبکه های کامپیوتری

چکیده:

امروزه علم کامپیوتر به عنوان یکی از حساس ترین و پراهمیت ترین علوم روز دنیا در شاخه های بسیار گوناگون و متفاوتی در حال پیشرفت سریع و همه جانبه است. یکی از مهمترین شاخه ها علم شبکه است. با توجه به اهمیت شایان ذکر انتقال اطلاعات در دنیای امروز شبکه های کامپیوتری بیش از پیش مورد توجه قرار گرفته اند و امروزه در سازمانها، ادارات و مراکز بزرگ و کوچک استفاده از شبکه های کامپیوتری امری مرسوم و معمول است. لذا موضوع و تمرکز اصلی این دوره کار آموزی روی شبکه های کامپیوتری بود. من در این دوره مطالعه خود روی شبکه های را با محوریت دو بخش متفاوت انجام دادم. اولین بخش مربوط به کلیات شبکه و اطلاعات مقدماتی نصب و راه اندازی شبکه ها بود که شامل بخشهای تئوریک و علمی فراوان می باشد. از جمله مهمترین این بخشها عبارت بودند از:

- پترولکل های ارتباطی
- مدل ارتباطی سیستم باز OSI و بررسی لایه های مختلف شبکه که شامل تجزیه و تحلیل هر لایه وظایف آن در معماری لایه ای می باشد.
- مدل های مختلف فیزیکی شبکه از قبیل مدل Peer to Peer , Sener base model و

Client – sener model

- بررسی اجزای ارتباطی شبکه ها مانند کارت شبکه (NIC)، کابل های ارتباطی مانند Twisted Pair و Coaxial و نحوه اتصالات آنها و قطعات سخت افزاری
- مهم مانند تکرار کننده ها (Repeaters)، پل ها (bridges)، مسیر یابها

(Routers) دروازه ها (Gateways)، سوئیچ ها (Switch es) و اینکه هر کدام از این

قطعات چه وظایفی دارند و در کدامیک از لایه های شبکه مورد استفاده قرار می گیرند.

در بخش دوم به جنبه های مدیریتی شبکه ها و نحوه تنظیمات و اداره آنها پرداختم. در این

قسمت که بیشتر جنبه نرم افزاری داشت با استفاده از سیستم عامل توانمند Linux مدیریت

بخشهای مختلف یک شبکه را مطالعه کردم. برای این امر ابتدا لازم بود آشنایی های اولیه را

با سیستم عامل Linux کسب کنیم و سپس مطالعه خود روی این سیستم عامل را بیشتر با

جهت گیری در زمینه کاربرد آن در شبکه ها ادامه دهم. لذا در ابتدا با طریقه نصب و راه

اندازی Linux آشنا شدم و سپس با تعدادی دستورهایی اولیه مفهوم و مفاهیم کاربردی

ضروری آن آشنا گشتم. در قسمت بعد به مطالعه کاربرد Linux در شبکه های کامپیوتری

پرداختم. البته مطالعه و کار در این بخش بسیار وسیع است و من تنها کار خود را محدود به

کار با تعدادی از Sener های مهم Linux کردم.

این Sener ها عبارت بودند از : DNS Sener , Samba Sener , HTTP Sener , Squid

Sener , DHCP Sener که در هر مورد ابتدا به کلیات و وظایف هر یک پرداختم. سپس

به معرفی دستورات لازم برای کار با هر یک پرداختم و در نهایت تنظیمات هر کی را مطالعه

کردم. در هر یک از این مورد سعی کردم بیشتر به جنبه های علمی قضایا نگاه کنم و از

پرداختن به جزئیات تئوریک صرف حتی المقدور پرهیز کنم. لذا در جاهایی که احساس

کردم توضیحات و جزئیات بیشتر صرفا جنبه تئوریک یا استفاده های بسیار خاص دارد به

معرفی منابع و سایتهای اینترنتی مربوطه اکتفا کردم ولی سعی کردم نمونه هایی از فایل های

مهمی که در تنظیمات این سرورها نقش مهمی دارند در این گزارش بیاورم.

شرح واحد صنعتی :

شرکت گیلدس کامپیوتر یک شرکت قدیمی و با سابقه در بازار کامپیوتر است که علاوه بر جنبه های تجاری سعی کرده به جنبه های علمی و تئوریک نیز بها دهد. فعالیت عمده این شرکت در دو بخش سخت افزار و شبکه صورت می گیرد.

در بخش سخت افزار، اسمبل کردن قطعات کامپیوتری و تعمیر قطعات صورت می گیرد. البته این کار معمولاً در حجم های بزرگ صورت می گیرد زیرا مشتریان طرف قرار داد این شرکت با برخورداری از امکانات و مهارت های کانی در بخش شبکه، با شرکت در مناقصه های بزرگ، پروژه های مختلفی را به عهده می گیرد. این پروژه ها معمولاً شامل نصب در راه اندازی شبکه های محلی (LAN) می باشد که شامل تهیه قطعات سخت افزاری و نصب کامل و دقیق شبکه و سپس انتخاب Sener های مناسب با توجه به نیازهای شبکه و تنظیمات آنها و در نهایت تنظیمات Client های شبکه می باشد.

اینجانب دوره کارآموزی خود را زیر نظر آقای مهندس بابادی نیا مدیر فنی شرکت انجام دادم.

ایشان به عنوان مدیر فنی هم مسئولیت بخش سخت افزار شرکت یا بخش فنی آن را دارا میباشند.

و هم در پروژه های بخش IT و شبکه با سایر اعضای شرکت همکاری می کنند و نقش مؤثری را ایفا می کنند.

- دستورهای Mount , Umount :

سیستم عامل Linux فایلها و شاخه های خود را به صورت یک درختی کلی به هم پیوسته سازماندهی می کند.

این درخت از شاخه Root شروع می شود تا به شاخه های سیستم و کاربرد و ... برسد. در Linux شاخه های مختلف با اساسی خاص برای منظوره های مشخصی در نظر گرفته می شوند. به عنوان مثال :

ساختار فایلی با این شاخه آغاز می شود (root نامیده می شود) /

برای نگهداری فایل های Kernel و Module هایی که هنگام boot شدن loud می شوند.

/ Boot

/ Home شاخه های کاربر را در بر می گیرد.

/ Dev فایل های واسط برای device هایی مثل پرینتر را نگهداری می کند.

برای نگهداری شاخه هایی که به طور موقت mount می شوند مانند Floppy, CD ,

/ etc ROM

/ tmp برای نگهداری فایل های موقت (temporary)

برای استفاده از فایل ها و منابعی که در مکانهایی خارج از درخت بهم پیوسته تعریف شده

برای Linux می باشند باید آنها را mount کرد. این منابع ممکن است Floppy disk یا

CD-ROM یا مثلاً روی یک partihon غیر از Linux روی Hard disk واقع باشد.

برای استفاده از چنین منابعی می توانیم آنها را به دو صورت mount کنیم که یکی موقتی و

دیگری دائمی است که به ذکر هر کدام می پردازیم:

۱) روش اول این است که با استفاده از دستور mount و تعیین یک مکان برای mount کردن منبع مورد نظر، آنرا به طور mount کنیم. فرصت کلی این دستور به شکل زیر است:

```
$ mount device mountpoint
```

فایلهای مربوط به device مورد نظر ساخته می شوند. مثلا Fdo به اولین Floppy drive متصل به سیستم اشاره می کند. در Linux و partition های هارد دیسک با دو حرف «hd» آغاز می شوند که بعد از آن حرفی که بعنوان label برای partitan در نظر گرفته می شود می آید.

مثلا hdaz نامی است که Linux برای دومین partitan از اولین Hard disk متصل به سیستم در نظر می گیرد. در اکثر موارد می توان با کمک گرفتن از دستور man به همراه یک با چند حرف اول device مورد نظر، اطلاعات و جزئیات آن را استخراج کرد. مثلا با دستور man sd می توان صفحات راهنمای SCSI device را مشاهده نمود.

```
$ mount / dev / mydir
```

مثلا
این دستور Floppy disk موجود در اولین FDD را در شاخه / mydir mount می کند.
شاخه ای که می خواهیم چیزی را در آن mount کنیم باید خالی باشد. اگر در این شاخه فایلی از قبل mount شده باشد پیغامی مبنی بر اینکه فایل دیگری قبلا در این شاخه mount شده، دریافت می کنید. برای جایگزینی فایلی که قبلا mount شده با یک فایل جدید، باید ابتدا آن فایل را un mount کنید. مثلا باید ابتدا floppy disk را un mount کنید تا

بتوان یک دیسک دیگر را در drive قرارداد و mount کرد. فرمت کلی این دستور مشابه

دستور mount می باشد:

\$ umount device or mount point

\$ umount / dev/ fdo

مثال
/ ۱

این دستور فلاپی دیسک mount شده در شاخه / mydir را umount می کند.

همچنین به طور مشابه برای unmount کردن فلاپی می توان در دستور بالا به

جای device ، mount point را ذکر کرد. نتیجه یکسان خواهد بود:

\$ umount / mydir

نکته بسیار مهم: یک محدودیت بزرگ درباره دستور umount وجود دارد که ممکن است

برای کاربر مشکل ساز شود. هیچگاه یک شاخه یا یک فایل سیستم را که در حال حاضر در

آن می باشید یا در حال استفاده از آن هستید unmount کنید. اگر سعی در انجام چنین کار

می کنید پیغام خطائی از طرف سیستم می آید:

The file system is busy !

برای مثال، فرض کنید که یک CD- ROM را در شاخه / mut / cdrom , mount کرده

اید و سپس به همان شاخه /mnt / cdrom بروید. اگر در این حالت تصمیم به عرض

کردن CD- ROM بگیریید ابتدا باید CD-ROM فعلی را با کمک دستور umount ، ار

محل فعل umount کنید ولی این دستور عمل مورد نظر شما را انجام نمی دهد زیرا در

حال حاضر همان در شاخه ای هستند که می خواهید آن را umount کنید. برای انجام این

کار باید ابتدا شاخه مورد نظر را ترک کنید و سپس اقدام به unmount کردن CD-ROM نماید.

```
$ mount / dev / hdc / mut / cdrom
```

```
$ cd / mut / cdrom
```

```
$ umount / mut / cdeom
```

```
umount : / dev / hdd : device is busy
```

```
$ cd / root
```

```
$ umount / mnt / cdrom
```

دستور mount ، option های مختلفی دارد که می توانید با مراجعه به مآخذ از جزئیات آنها مطلع شوید.

(۲) mount دائمی

در صورت استفاده از دستور mount ، حتی اگر دستور umount را اجرا نکنید پس از reboot شدن سیستم، منابع و فایل های mount شده خود به خود umount می شوند.

ولی گاهی اوقات نیاز است به اینکه یک فایل یا یک device برای مدتی یا برای همیشه در محلی mount شود. برای انجام این کار می توانیم از فایل fstab استفاده کنیم.

ورودی هایی که در این فایل قرار دارند هر بار هنگام boot

mount می شوند و دیگر نیازی به mount کردن آنها به طور کلی نمی باشد. این فایل در

شاخه /etc / وسیع است. یک ورودی (entry) در فایل fstab دارای فیلدهای متعددی

است که به وسیله یک Space یا tab از یکدیگر جدا شده اند. فرمت کلی این ورودیها بدین شکل می باشد.

< device > < mountpoint > < filesystemtype > < options > < dump >

< fsck >

اولین فیلد مربوط به نام فایل سیستم مورد نظر است. دویمین فیلد شاخه ای که می خواهیم فایل سیستم در آنجا mount شود (به فایل سیستم Linux متصل شود) را مشخص می کند. سومین فیلد مربوط به تایپ یا نوع فایل سیستم است. تایپ یک partiton استاندارد در Linux ، ext3 می باشد.

فایل سیستم برای windaus های 95 و 98 و Millennium ، vfat می باشد و برای Windows های NT ، 2000 ، XP این تایپ ntfs می باشد. تایپ auto سعی می کند به طور خودکار نوع فایل سیستم مورد نظر را تشخیص دهد.

مثال ، با باز کردن فایل /etc / fstab و دارد کردن در entey ذیل ، فلاپی دیسک و cdrom همیشه به طور خودکار هنگام boot شدن سیستم ، mount می شوند:

```
/ dev / hdc / mnt / cdrom iso 9660 ro. Noauto 00
```

```
/ dev / fdo / mnt / floppy / auto defaults / noauto 00
```

یا به عنوان مثال دیگر اگر بخواهیم اولین patiton از اولین Hard disk را به طور خودکار mount کنیم چنین entey را در فایل fshib وارد می کنیم:

```
/ dev / had / mnt / windows vfat default 3 00
```

و label این partiton در شاخه /mnt ، windows خواهند بود .

سیستم پرونده (File System)

Linux نیز همانند همه سیستمهای عامل به طرز چشمگیری بر اطلاعات ذخیره شده در پرونده ها تکیه می کند:

اطلاعات کاربران مختلف، پرونده های اجرایی مورد نیاز کاربران، پرونده های داده ای مربوط به آنها، کتابخانه های مورد نیاز برای برنامه نویسی ، اطلاعات مربوط به تنظیمهای سخت افزاری و امکانات موجود در سیستم، کد اجرایی خود سیستم عامل و بسیاری اطلاعات دیگر، همگی به صورت پرونده ذخیره می شوند. بنابراین با توجه به اهمیت و حساسیت اطلاعات فوق الذکر لازم است که این پرونده ها تحت یک نظام قوی و قابل اطمینان مدیریت و نگهداری شوند. در سیستمهای عامل ، انجام این وظایف بر عهده سیستم پرونده است. مثلاً در DOS سیستم پرونده FAT و در سیستم عامل Windows NT ، سیستم پرونده NTFS برای این کار طراحی شده اند. گونه هایی از Unix که قبل از BSD نگارش ۴/۲ ایجاد شده اند، هر یک سیستم پرونده مربوط به خود را داشتند. یکی از ویژگیهای جالب توجه سیستم عامل Linux در نگارشهای System V Release 4 به بعد این است که سیستم پرونده آن انواع سیستمهای پرونده موجود را می شناسد و قادر است اطلاعات موجود در آنها (یعنی پرونده هایشان را) بخواند. سیستمهای پرونده پر استفاده در Unix عبارتند از: Extended File System 2 و System V File System صرف نظر از نوع ، سیستم پرونده باید اطلاعاتی را که سیستم عامل برای شناسایی کامل یک پرونده نیاز دارد مهیا کنند. در ادامه به شرح این اطلاعات کمی پردازیم:

نوع پرونده: در Linux انواع مختلفی از پرونده ها وجود دارد. به هر نوع یک حرف کوچک انگلیسی متناظر اختصاص داده شده است. انواع مختلف پرونده به همراه حروف متناظرشان در شکل فهرست شده است:

نوع پرونده	حرف متناظر
Ordinary file	-
Directory	d
fifo	p
Character device	c
Block device	b
Link to another file	l

- نوع اول اغلب شامل پرونده های داده ای می شود. مثلا پرونده های برنامه های C از نوع اول هستند.

- نوع دوم نیز که برای کاربران DOS و Windows، نوع پرونده شناخته شده ای است برای دسته بندی مجموعه های پرونده استفاده می شود.

Fifo نوعی پرونده است که برای برقراری ارتباط بین فرایندها استفاده می شود. یکی از راههای تبادل اطلاعات بین فرایندها در Linux، استفاده از fifo است. ویژگی این کانال ارتباطی این است که پس از قطع ارتباط بین فرایندها از بین نمی رود و پایدار باقی می ماند. قبل از توضیح دادن نوعی های c و b لازم است مطالبی راجع به نحوه استفاده Linux از سخت افزار بدانید:

سیستم عامل Linux با هر سخت افزاری به صورت یک پرونده برخورد می کند. مثلاً ارسال یک بلوک داده به چاپگر معادل نوشتن آن بلوک داده و پرونده متناظر با چاپگر است. به این پرونده، دستگاه (device file) گفته می شود. عملیات ورودی و خروجی در سخت افزارهای مختلف به دو صورت انجام می شود:

ارسال و دریافت بایت به بایت داده ها

ارسال و دریافت بلوکی داده ها (در یک انتقال بیش از یک بایت منتقل شود): متناظر با این تقسیم بندی، پرونده های دستگاه نیز به دو دسته تقسیم می شوند:

character device : مانند درگاه سری (Sereal Port).

Block device : مانند دیسک سخت.

پرونده های نوع اول را با حرف c و نوع دوم را با b نمایش می دهند. بعداً درباره این پرونده می تواند آن را به مالکیت کاربران دیگر در آورد. مدیر سیستم و کاربران ایجاد کننده پرونده ، نمونه هایی از مالکان پرونده اند.

گروه پرونده : در سیستم عامل Linux کاربران به گروههایی تقسیم می شوند. برای یک پرونده علاوه بر مالک آن شماره گروهی از کاربران که می توانند به آن پرونده دسترسی داشته باشند، نگهداری می شود.

اجازه های دسترسی (Access P ermissions) : نحوه دسترسی افراد به پرونده ها را مشخص می کند. سه دسته از افرادی که می توانند به یک پرونده دسترسی داشته باشند، عبارتند از :

۱- مالک پرونده

۲- افراد درون گروه مربوط به آن پرونده

۳- بقیه کاربران

برای هر کدام از افراد فوق سه نوع اجازه دسترسی مطرح می شود:

۱- خواندن (Read)

۲- نوشتن (Write)

۳- اجرا کردن (eXecute)

- طریقه نصب نرم افزار در سیستم عامل Linux:

نصب نرم افزار یک کامپیوتر (administrative function) است که به وسیله کاربر root

انجام می شود.

اگر هنگام نصب Linux تمام package های موجود را انتخاب نکرده باشید، تنها تعدادی

از application و utility های فراوان موجود برای کاربران Linux روی سیستم شما

نصب شده است. بسیاری از نسخه های Linux مانند SUSE ، Mandrake ، Calaera ،

Rad Hat از ابزاری به نام Rad Hat Package Manager یا RPM برای سازماندهی نرم

افزارهای Linux و تبدیل آنها به package هاییکه به طور خودکار قابل نصب یا حذف

کردن باشند استفاده می کنند. Debian از ابزار مدیریتی package خاص خود به نام

Advanced Package Tool یا ATP استفاده می کند. با این ابزار می توان براحتی یک

package را روی سیستم نصب (install) کرد یا پس از نصب آنرا حذف (remove)

نمود. نرم افزارهایی که روی CD-ROM شما موجودند با تمام گستردگی شان تنها بخش

کوچکی از نرم افزارهای موجود برای کاربران Linux را تشکیل می دهند.

شما می توانید نرم افزارهای اضافی از سایتهای نرم افزاری مانند سایت Soura forge به آدرس اینترنتی [Sourceforge . net](http://Sourceforge.net) , download کنید.

- نصب نرم افزار به وسیله RPM :

- یک برنامه کاربردی در Linux اغلب شامل تعدادی فایل‌های مختلف است که باید در شاخه های متفاوتی نصب شوند. خود برنامه اغلب در شاخه ای به نام `/usr / bin` قرار می گیرد در حالیکه Online manual files در شاخه دیگری قرار می گیرند و فایل‌های کتابخانه ای (library files) در شاخه ای دیگر. به علاوه عملیات نصب ممکن است نیاز به تغییرات بعضی فایل‌های Configy داشته باشد. Pakagre های نرم افزاری RPM تمام این کارها را برای شما انجام می دهند. همچنین ، اگر در آینده تصمیم گرفتید که مثلا به یک application به خصوص دیگر نیازی ندارید می توانید آن را Uninstall نمایید. در یک محیط desktop مانند KDE یا Gnome ، می توانید با استفاده از ابزار مدیریتی GUI ، Package های RPM را نصب یا حذف نمایید. بعضی نسخه ها ابزار مدیریتی خاص خود را دارند. Mandrake Software Manager Mandrake که تا حد زیادی شبیه به Kpackaye رفتار م یکنند استفاده می نماید. Caldera این خدمات را از طریق COAS فراهم می کند.

همچنین می توان Package های RPM را به کار بردن دو ابزار مستقل مدیریت نمود که اولی Gnome RPM و اولی Kpackage Utility می باشد. GnomeRPM برای کار روی هر محیط Gnome desktop طراحی شده است و Kpackage بخشی از KDE desktop می باشد.

با این ابزارها می توان محل یک package را روی سیستم تشخیص داد. Package هایی را که می خواهید نصب کنید انتخاب نمایید. هر دو این ابزار یک لیست از package های نصب شده روی سیستم و نشان می دهند.

اگر package مورد نظر برای نصب یا update کردن package دیگری داشته باشد ممکن است هنگام نصب آن مشکلاتی رخ دهد. این مشکلی است که اغلب وقتی یک application نیاز به نسخه updateion نیاز به نسخه update شده یک کتابخانه مشترک دارد، اتفاق می افتد. در این صورت شما از مشکل مطلع می شوید و برای ادامه از شما سؤال می شود. شما می توانید کار را در آن زمان Cancel کنید و یا با پید کردن جای package مورد نیاز به نصب آن اقدام کنید.

اگر به محیط desktop دسترسی ندارید یا ترجیح می دهید که از واسط Command line استفاده کنید، می توانید با استفاده از دستور package rpm مورد نظر را مدیریت را نصب کنید.

فرم کلی `$ rpm options rpm - package - name`

دسته rpm :

`$ rpm - ql mozilla lmore`

مثال

- ql فایل های موجود در application را لیست می کند.

`$ rpm - ivh xv- 3010a -23.i386. rpm`

-I برای نصب package های جدید استفاده می شود. برای update کردن از U -

استفاده می شود.

V - فایل‌های را که حال نصب شدن هستند نمایش می دهد و h - یک سمبل متحرک است که نشان می دهد rpm هنوز مشغول اجرا است. استفاده از vh - در در حین کاربرد دستور rpm کاملاً مرسوم می باشد.

\$ rpm - Urh mozilla - l. 0rcJ - D . I 386. Rpm.

\$ mount / dev / mydir

در اینجا کافی است علامت ≠ را حذف کنید و سپس نام وب سرور خود را بجای new.host.name تایپ کنید.

در Red Hat و بعضی نسخه های دیگر، یک entry استاندارد برای نصب web server با استفاده از /var/www بعنوان شاخه web site شما در نظر گرفته شده است. شما می توانید web site خود را بفرم دلخواهتان تغییر دهید. این کار بوسیله انجام تغییرات مناسب در راهنماهای مربوط می باشد. راهنمای Dowment Root ، شاخه در نظر گرفته شده برای Web page های شما را تعیین می کند. همچنین Server Root مشخص می کند تنظیمات web server های شما را تعیین می کند. همچنین Sever Root مشخص می کند که تنظیمات web server ، خطاها و فایل‌های log در کجا نگهداری شوند.

Do wment Root /var/www/htm/

Sever Root /etc/httpd

راهنمای MaxClient ، ماکزیمم تعداد کاربرانی که بطور همزمان می توانند به سایت شما وصل شوند را تعیین می کند.

MaxClient 150

بدیهی است شما می توانید این عدد را مطابق دلخواهتان تغییر دهید.

- کنترل دسترسی: (Access Control)

با کمک راهنماهای کنترل دسترسی مانند allow و deny می توانید دسترسی به کاربران را

به web site تان کنترل کنید. بکاربردن allow به همراه لیستی از whostname ، اجازه

دسترسی را تنها به آن hostname های مشخص شده محدود می کند. با بکاربردن deny به

همراه لیستی از hostname ها ، اجازه دسترسی از آنها سلب می شود. آرگومان all اجازه

دسترسی یا عدم دسترسی را به تمام hostname ها تعمیم می دهد. راهنمای order تعیین می

کند را همماهای کنترل دسترسی با چه ترتیبی باید اجرا شوند. سایر راهنماها مانند require

باعث ایجاد authentication می شوند که کاربران را مجبور به log in کردن می نماید.

راهنماهای اجازه دسترسی ممکن است بطور سراسری برای کنترل دسترسی به کل سایت

استفاده شوند یا دریک راهنمای شاخه خاص قرار گیرند و کنترل دسترسی را درمورد آن

شاخه انجام دهند. دراین مثال اجازه دسترسی به تمام کاربران داده می شود:

order allow, deny

allow from all

با web site شما ارتباط برقرار می کند. برای مثال ، یک browser ممکن است اطلاعات

موردنظر را ازیک صفحه بگیرد (GET) یا اطلاعات را از طریق آن صفحه بفرستد (post).

شما می توانید چنین دسترسی هایی را با استفاده از راهنمای `<Limit>` محدود کنید.
`<Limit>` زوج می شود تا یک block محدود را که در آن می توانید اجازه دسترسی ها را کنترل کنید تعریف نماید. این راهنماها تنها در مورد متدهای دسترسی معین شده کاربرد دارند. می توانید چنین بلاکهای Limit را در یک Directory بکار ببرید تا اجازه دسترسی ها را تنها برای آن شاخه کنترل کنید. در این مثال بلاک directory در فایل `/etc/config/httpd.conf` متدهای دسترسی را برای شاخه home سایت شما کنترل می کند.

≠ This should be changed to whatever you set Document Root to.

```
<Directory /var/www/html>
```

```
Options Indexes Follow SymLinks
```

```
AllowOverride All
```

```
</ Limit GET>
```

```
Order allow. Deny
```

```
Allow from all
```

```
</Limit>
```

```
</Directory>
```

مثالی دیگر

```
<Directory /var/www/html/mypics>
```

```
Order Deny. Allow
```

Deny from [www. Myvids. Com](http://www.Myvids.Com)

در این مثال اجازه دسترسی به شاخه mypics را از تقاضاهایی که از www.myvids.com

می آیند سلب می کند.

* برای اطلاعات بیشتر از جزئیات یا هرگونه تنظیمات احتمالی دیگر می توانید به مراجع یا

سایتهای زیر رجوع کنید.

www.apache.org – httpd.apache.org

DNS Servers

Domain Name Service یا DNS یک سرویس اینترنتی است که اسمی حوزه (domain

name) را به IP های متناظرشان تبدیل می کند. همانطور که می دانید تمام کامپیوترهایی که

به اینترنت متصل می باشند با استفاده از پروتکل اینترنت (IP) ، آدرس دهی می شوند.

آدرس دهی به شیوه IP ممکن است به روش جدیدتر IP6 (ویرایش 6 Internet Protocol)

یا فرم قدیمی تر IP4 انجام شود. ولی از آنجا که هنوز اکثر سیستم ها از روش آدرس دهی

IP4 حمایت می کنند مثلهایی که در این جا استفاده می کنیم از همین شیوه آدرس دهی

تبعیت می کنند. این شیوه آدرس دهی از این قرار است که به کامپیوتر یک سوی اعداد

در غالب چهاربخش (Segment) متفاوت نسبت داده می شود. بطور مثال 192.168.0.2 .

در اینجا جزئیات تسالین شیوه آدرس دهی و معنای این Segment ها و کلاسهای مختلف IP

بحث مورد نظر؟؟؟. بحث مهم اینکه در هر صورت کاربرد این اعداد بعنوان آدرس در عمل

بسیار سخت و حتی غیرممکن می باشد و احتمال بروز خطا نیز بالا میرود. برای حل این

مشکل متناظر با هر یک از این IP ها یک نام در نظر گرفته می شود که به آن fully

qualified domain name می گویند. بطور مثال www.linux.org دارای یک IP های متناظرشان به یکدیگر است. می توان برای هر کامپیوتر یک فایل در نظر گرفت که این عمل متناظر بوسیله آن انجام شود. در Linux , Unix این فایل `/etc/hosts` می باشد. در اینجا می توان آدرس IP و نام یک کامپیوتر را وارد کرد. ولی این متد این عیب را دارد که هر کامپیوتر روی شبکه نیاز به یک لیست کامل از تمامی کامپیوترهای دیگر متصل به شبکه دارد و تازه این لیست باید مرتباً Update نیز شود. هر چند این متد برای شبکه های کوچک قابل اجرا است ولی کاملاً بدیهی است که برای اینترنت به هیچ وجه عملی نمی باشد. به این دلیل سرویس نام حوزه یا Domah Name Service برای ترجمه اسامی کامپیوترهای متصل به اینترنت به آدرسهای IP متناظرشان بوجود آمد. این عمل بوسیله بهم وصل کردن severهایی که سرویس نام حوزه را ارائه می دهند (که به آنها DNS Server یا name server می گویند) انجام می شود. این severها لیستی از نامهای کامپیوترها و IPهای متناظرشان را نگه داری می کنند. هرگاه یک نام بعنوان آدرس تایپ می شود، ابتدا این نام به DNS Server مربوطه ارجاع داده می شود و IP متناظر آن استخراج می شود و سپس درخواست درحقیقت به آن آدرس IP فرستاده می شود. مجموعه این عملیات را resolve کردن آدرس IP گویند. برای مثال وقتی ما درخواست برقراری ارتباط با سایت www.Linux.org را می کنیم ابتدا آدرس IP این سایت resolve می شود و سپس درخواست ما به آدرس 192.182.196.56 که آدرس IP متناظر این نام است ارسال می شود.

:BIND

نرم افزار DNS Server ای که در حال حاضر در سیستمهای Linux استفاده می شود BIND
یا Berkeley Internet Name Domain می باشد. این نرم افزار از یک دیمن name
server، تعدادی تنظیمات نمونه و کتابخانه هایی که در عملیات resolve کردن استفاده می
شوند تشکیل شده است. نام BIND name sever daemon, named می باشد. برای
اینکه سیستم شما بتواند بعنوان یک name server عمل کند کافی است این named
daemon را اجرا کنید. این daemon به تقاضاها برای resolution یک نام (پیدا کردن
آدرس IP متناظر) گوش می دهد و آدرس IP متناظر صحیح را برمی گرداند. مشابه آنچه
در مورد httpd داشتیم برای start, stop, ... در اینجا نیز می توانیم از دستور servuce
استفاده می کنیم:

مثال: Service named restart

این دستور daemon مربوط به DNS Sever یا همان named را restart می کند.

- تنظیمات و پیکربندی DNS با استفاده از فایل named.conf:

در این قسمت باید ابتدا با مفهوم Zone آشنا شویم. Zone به قسمتی از شبکه که name
server ??? آنرا به عهده دارد گفته می شود. یک Zone مانند domain نیست زیرا در یک

domain بزرگ ممکن است Zone های مختلفی وجود داشته باشند که هر کدام name
server خود را دارند. همچنین ممکن است یک name server به چندین Zone سیروس

دهد. در چنین مواردی هر Zone، فایل مربوط به خود را دارد. Zone file ها اطلاعات

مربوط به منابعی که تناظر بین IP ها و اسامی را دارند در خود نگه می دارند. ورودی های

Zone در فایل named.conf تعریف می شوند. در این فایل ورودی های Zone برای forward DNS server, slave, master قرار می گیرند. تایپهای معمول Zone عبارتند از:

- Master Zone: فایل اصلی یا primary zone file برای یک شبکه است. این فایل شامل نقشه ای از domain name ها به آدرسهای IP متناظر برای تمام host های شبکه است.
- Slave Zone: این فایل درحقیقت یک ارجاع به DNS Server های روی شبکه است. شبکه شما می تواند یک Master DNS داشته باشد و تعدادی Slave DNS Server برای کمک به کاهش بار روی شبکه. یک Slave DNS Server بطور خود کار تنظیمات خود را از master DNS Server کپی می کند که شامل تمام Zone file ها می باشد.
- Forward Zone: این فایل شمال لیست اسامی name server های خارج شبکه است که در صورتیکه name server شبکه نتوانست آدرسی را پیدا کند به آن server ارجاع دهید، forward کند. IN-ADDR. ARPA Zone: برای عملیات reverse resolution یا پیدا کردن IP بطور معکوس است. در این حالت یک IP داریم که می خواهیم domain name متناظر آنرا پیدا کنیم. چنین عملی برسیم تا فایل های این Zone انجام می شود. هر فایل master zone معمولاً یک فایل IN-ADDR. ARPA Zone متناظر دارد که عملیات reverse resolution را انجام دهید. به ازای هر entry در فایل master zone یک reverse mapping entry در این فایل وجود دارد با این تفاوت که segment های این ورودی برعکس هستند. مثلاً به ازای یک ورودی IP، 192.168.0.4 در فایل master zone، در این فایل ورودی 4.0168.192 وجود دارد.
- Hint Zone: مشخص می کند که root name server کیست.

فایل named.conf :

همانطور که گفته شد تنظیمات لازم برای named.conf در این فایل قرار می گیرند. فرمت

کلی این فایل بدین شکل است:

[in نام domain “ zone

type می تواند Slave. Master. Hint. Forward باشد type

ز “ نام فایل “ file

{

options {

این zone های name server ز “ نام شاخه “ directory

شاخه های

فایل های cache فایل های

نام DNS Server هایی که query ها در صورت { } forwarders {

لزوم به آنها Forward می شوند.

} ز

در اینجا برای فهم تریک نمونه از فایل تنظیم شده named.conf را می آوریم.

11 A Simple BIND configuration

logging {

z { null } category cname {

z }

```
options {  
    directory "/var/named" ;
```

```
};
```

```
zone "." {
```

```
    type hint ;
```

```
    file "named.ca" ;
```

```
};
```

```
zone "my.trek.com" {
```

```
    type master ;
```

```
    file "my.trek.com" ;
```

```
};
```

```
zone "1.168.192.in-addr.arpa" {
```

```
    type master ;
```

```
    file "142.168.o" ;
```

```
zone "0.0.127.in-addr.arpa" {
```

```
    type master ;
```

```
    file "named.local" ;
```

```
};
```

قدم بعدی ایجاد ZONE FILE دارد named.conf های مربوطه می باشد.

- منابع رکورد (Resource Records) :

name sever اطلاعات مربوط به domain name را برای هر host روی شبکه در منابع رکوردی که داخل فایل های zone و reverse mapping هستند نگهداری می کند. برای هر کامپیوتر نیاز به یک رکورد در فایل zone می باشد.

هر رکورد معمولاً یک خط اشغال می کند. فرمت استاندارد یک رکورد بدین شکل است:

name [< tt1>] [< class>] < type> < rdata> [< comment>]

name نام رکورد می باشد. Tt1 یا time to live زمانی است که رکورد بوسیله . cache sender می شود.

Class ، کلاس رکورد را تعیین می کند. کلاسی که در اثر رکوردها بکار می رود IN برای اینترنت است.

Type ، تایپ رکورد را مشخص می کند. این تایپ می تواند برای یک host ، NS برای

NAME SEVER مسول یک Zone ، mx برای یک mail sener ، txt برای رشته های

تستی که معمولاً حاوی اطلاعاتی درباره یک host هستند و چندین تایپ دیگر باشد. در

اینجا به ذکر مثالی می پردازیم که در آن نام rabbit . mytrek. Com باشد، کلاس

کلاس in یا اینترنت است و تایپ از نوع یک host (A) می باشد و data همان آدرس IP است.

Rabbit. Mytrek. Com IN A 192. 168.O.2

: Start of Authority (SOA)

بر فایل Zone و معکوس آن همیشه با رکورد خاصی به نام SOA شروع می شوند. این رکورد معین می کند که رکوردهای بعدی برای این domain می باشند. همچنین این رکورد شامل اطلاعاتی درباره name server حوزه جاری است که باید به name server های دیگر داده شود. فرمت این رکورد تقریباً مشابه فرمت رکوردهای دیگر است بجز سمت deota که قدری متفاوت است.

Name { ttl } class soa origin person- in charge [

Serial number

Refresh

Retry

Expire

Minimum

، مثال (@ IN soa turtle . mytiel . com hostmaster. Turtle. My trik . com

1997022700 ; serial

28800 ; Refresh

14400 ; Retry

3600000 ; Expire

86400 ; Minimum

علامت @ معمولاً بعنوان اسم بکار می رود و مانند یک macro عمل می کند که نام حوزه را به آن تعمیم می دهد. Person- in charge همان آدرس e-mail شخصی است که name server را مدیریت می کند. (در این آدرس از " به جای @ استفاده می شود).
اولین شماره، serial no است. هنگامی که رکوردها را تغییر می دهید یا رکوردی اضافه می کنید این serial number را تغییر می دهد. / Refresh تعیین می کند که اطلاعات SOA پس از چه مدتی refresh (تازه) شود.

Retey تعداد دفعات تلاش برای برقراری ارتباط با server است. Expine طول زمانی است که یک name server ثانویه اطلاعات مربوط به یک zone را بدون update کردن آن نگهداری می کند.

Minimum حداقل زمانی است که یک رکورد در یک zone باقی می ماند.
برای اطلاعات بیشتر از نحوه تنظیمات و چگونگی کار با name server می توانید به سایتهایی مانند www.Dns.Netldnsrd مراجعه کنید یا DNS HOW – TO را مطالعه نمایید.

ولی برای تکمیل شدن بین قسمت در پایان به ذکر یک نمونه فایل ZONE می پردازیم که در آن رکوردهایی از انواع مختلف و در حوزه های مختلف وجود دارند.

DHCP SERVER

Dynamic Host configuration protocol یا DHCP اطلاعات مربوط به سیستمهای متصل به شبکه های Tcclip را فراهم می آورد. ماشینهای روی شبکه بعنوان Client های لین server تنظیمات شبکه ای خود را از این server می گیرند. در حقیقت نتشل لین server،

اختصاص آدرس IP به ماشینهایی است که تصد اتصال به شبکه را دارند. Dhcp لین آدرس را از میان یک pool از آدرس که برایش تعیین شده است به /// ماشین اختصاص می دهد. DHCP از روشهای مختلفی برای اختصاص IP به Client ها حمایت می کند. این روشها عبارتند از اتوماتیک ، دینامیک یا پویا ودستی، در اختصاص IP به شیوه اتوماتیک یک IP دائمی به یک HOST اختصاص می دهد. در روش دستی، بدر شبکه یک IP را به Host تقاضا کننده اختصاص می دهد.

در اختصاص IP به شیوه dynamic (پویا) ، DHCP SERVER هرگاه یک HOST به IP نیاز داشته باشد یک IP به آن اختصاص می دهد. در اختصاص IP به شیوه پویا، DHCP، IP ها را از یک POOL انتخاب می کند و هرگاه که یک HOST کارش تمام شد آن IP را آزاد می نماید.

در اینجا مشکلی که پیش می آید اینستکه DNS SERVER به ازای هر IP شخص یک نام تعریف شده دارد در حالی که DHCP در اختصاص به شیوه پویا، هر بار یک IP متفان به هر HOST متقاضی اختصاص می دهد که ممکن است با IP ونامهای تعریف شده برای DNS Server متفان باشد. برای حل این مشکل DHCP SERVER، هر بار که یک IP به HOST ای اختصاص می دهد بطور خودکار DNS SENRE را UPDATE می کند.

- مراحل نصب: اجرا و تنظیمات DHCP:

اگر DHCP را هنگام نصب Linux، نصب نکرده اید، می توانید از طریق منوی start در

تحت AAD/ Remove Application. System settings را انتخاب کنید:

system settings > Add/ Remove Application > Network servers > dhcp

dncp برای اجرا از فایل dncp . conf استفاده می کند. در صورت عدم وجود این فایل، باید خودتان آنرا در شاخه etc / ایجاد کرده و edit نمایید.

طریقه دیگر راه اندازی DHCP مشابه متنهای قبل است که می توان از دستور SERVICE استفاده کرد.

Senice dncpd start | stop | restart | status
تنظیمات فایل dncp . Conf :

در این فایل باید IP یک کامپیوتر (آدرس IP کامپیوتری که DHCP روی آن اجرا می شود) را بعنوان سیریاب یا router بدهیم. همچنین subnet ای می خواهیم در آن با client ها کار کنیم را باید مشخص کنیم. و دیگر اینکه باید محدوده آرمهای IP که می خواهیم DHCP از بین آنها IP انتخاب کرده و به Client ها اختصاص دهد را تعیین و تعریف کنیم.

Dhcp. Conf

Ddns. Update – style ad – hoc ز

Option routers 134. 166. 5.1 ز آدرس SERVERIP مربوط به

Subnet 134.166. 5.5 net mask 255. 255. 255. 0 {

Range 134. 166. 0.5 134.168.0. 128 ز

Default – lease – time 21600 ز

Max – lease – fime 43200 ز

}

dhcp از pool تعیین شده یعنی از 134. 168. 0. 5 تا 134. 168. 0. 128 ، یک IP انتخاب

می کند و به CLIENT می دهد.

اگر خواستیم به یک Client مشخص دینامیک IP خاص داده شود می توانیم تنظیمات لازم را بصورت زیر در فایل Dhcp. Conf انجام دهیم:

```
host winxp {  
hardware ethernet 00: a0: d2 : 04: ac: d4 ;  
fixed – address          134- 188. 0. 100;  
  
}
```

در اینجا client با نام winxp همیشه IP ثابت 134. 168.0.100 را از DHCP دریافت می کند .

توجه به این نکته ضروری است که برای این شیوه اختصاص IP ، دانستن MAC برای

Client مورد نظر ضروری است. مثلاً در این جا آدرس MAC مربوط به 00: A0: d2:

04:ae: dp: winxp می باشد.

با دستور arp- a می توانیم MAC تمام Client هایی که در کلاس سرور ما هستند را ببینیم.

- در پایان ذکر این نکته لازم است که می توان آمار client های وصل شده را در فایلی بنام

var/ lib/ dhcp/ dhcpd. Leases / مشاهده کرد.