راه اندازی سرور با

FreeBSD®

سری مقالات ارایه شده در سایت

www.Mabedini.ir

نویسنده
محمد عابدینی

فهرست مطالب

مقدمه

سالها قبل بعد از شروع فعالیت در سایت خودم به آدرس Mabedini.ir تلاش کردم که در ابتدا سری از مقالاتی به عنوان BSD به زبان ساده را در سایت ارایه کنم برای آموزش مقدماتی BSD این دوره بعد از تکمیل شدن در قالب یک فایل در کانال رسمی تلگرام سایت ارایه شد، بعد از اتمام دوره BSD به زبان ساده دوره ای در سایت ایجاد کردم با عنوان FreeBSD برای مهندسین شبکه که به راه اندازی کردن سرویسها و سرورهای شبکه با زبانی ساده و کاربردی پرداختم و تلاش من در ارایه این مطالب دقت در پیاده سازی کامل کدها و دستورات و عملکرد درست سرویسها بود، بعد از اتمام این سری از مقالات به صورت تصویری هم در کانال آپارات رسمی سایت هم دوره تصویری آنرا در بخشهای مختلف ارایه کردم، حال در این زمان و با تغییر ساختار سایت سری مقالات ارایه شده به ترتیب و پشت سر هم ارایه نشده است و نیاز به ارایه کتابی در این زمینه با استفاده از مقالات ارایه شده در سایت را شروع کردم ، باشد که کتابی مفید و کاربردی در دست داشته باشید و با استفاده از آن بتوانید بخشی از نیاز های شبکه ای خود را در قالب سیستم عاملی امن و پر قدرت راه اندازی کنید.

راه اندازی اولین سرور در قالب FreeBSD سرور ftp:

ftp چیست؟

روتکل ftp یا همان file transfer protocol یک پروتکل انتقال فایل در شبکه است که به نوع سیستم عامل متصل شده به سرور وابسته نبوده سیستم عامل های مبتنی بر ویندوز و لینوکسی و یونیکسی هم می توانند از آن استفاده کنند. راه اندازی آن ساده است و در حالت client/server این سرویس ارایه می شود. اولین بار این پروتکل در سال 1971 توسط Abhay Bhushan در RFC 114 ارایه شد، در سال 1980 یکی از بخش های پروتکل TCP/IP شد و در September 1998 قابلیت استفاده از ورژن 6 آدرس IP به آن اضافه شد.

پروتکل ftp یا همان file transfer protocol یک پروتکل انتقال فایل در شبکه است که به نوع سیستم عامل متصل شده به سرور وابسته نبوده سیستم عامل های مبتنی بر ویندوز و لینوکسی و یونیکسی هم می توانند از آن استفاده کنند. راه اندازی آن ساده است و در حالت client/server این سرویس ارایه می شود. اولین بار این پروتکل در سال 1971 توسط Abhay Bhushan در RFC 114 ارایه شد، در سال 1980 یکی از بخش های پروتکل TCP/IP شد و در September 1998 قابلیت استفاده از ورژن 6 آدرس IP به آن اضافه شد.

برای برقرار ارتباط با سرور کلاینتها هم می توانند از نام کاربری و رمز عبور استفاده کنند و هم می توانند در قابل Anonymous به سرور شما متصل شوندو حالت Anonymous باید در سمت سرور شما فعال شده باشد و یکی از راههای اشتراک گذاری فایل در سطح شبکه های بزرگ است که شما قصد ندارید به هر فردی نام کاربری و رمز عبور دهید.

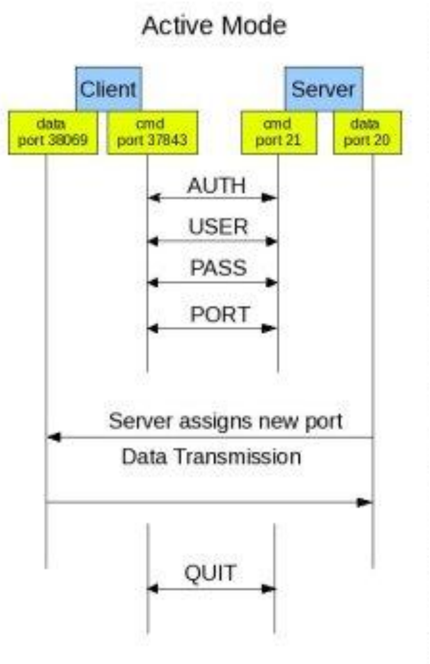
یکی از نقاط ضعف این سرویس کدگذاری نشدن رمزعبور و نام کاربری در زمان اتصال به سرور است، این بخش به صورت متن ساده منتقل شده و هر فردی که به شبکه شما دسترسی داشته باشد به راحتی با قابلیت اسنیف می تواند به رمزعبور و نام کاربری شما دسترسی پیدا کند. برای افزایش امینت در این سرور فابلیت ftps به آن اضافه شده است.

این سرور در دو حالت active و passive به ارایه سرویس می پردازد در ادامه با این دو حالت آشنا می شود.

بررسی حالت Active:

در این روش برقرار ارتباط، سیستم کلاینت با یک پورت تصادفی با عدد بالای 1023 به پورت سرور در شماره 21 متصل می شود، در مرحله بعد کلاینت در پورت تصادفی قبلی بعلاوه یک عدد (n+1) به برقرار ارتباط توسط سرور می نشیند و این پورت را برای سرور ftp ارسال می کند.

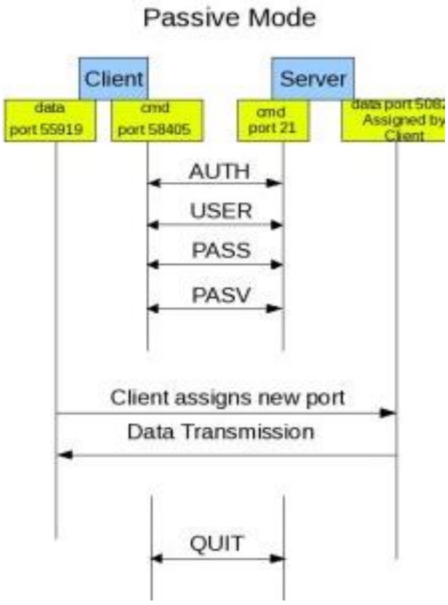
نکته:
پورتهای ارتباطی در سمت سرور برای اجرای فرمانها همیشه پورت 21 است و سرور برای انتقال فایل همیشه از پورت 20 استفاده می کند.



بررسی حالت Active درftp

حالت passive

در بعضی از موارد کلاینت در پشت فایروال و nat قرار دارد و سرور ftp نمی تواند به صورت مستقیم به پورت ارسالی از سمت سرور متصل شود برای حل این مطلب حالتی دیگر در برقرار ارتباط به نام passive ایجاد شده است. برای برقرار کردن این نوع ارتباط کلاینت با فرمان pasv به سرور می گوید که قصد برقرار ارتباط در حالت passive را دارد. در این روش قبل از برقرار ارتباط کلاینت دو پورت تصادفی بزرگتر از عدد 1023 به صورت تصادفی با حالت n و n+1 انتخاب می کند و به شماره پورت 21 سرور متصل می شود. بعد از تعیین کردن حالت passive با استفاده از فرمان pasv بین کلاینت و سرور این بار سرور یک پورت تصادفی بالاتر از 1023 باز کرده و به کلاینت اعلام می کند که برای دریافت فایل خود به این شماره پورت با استفاه از پورت تصادفی n+1 خود متصل شود و به دریافت فایل اقدام کند. در این حالت برقرار کنند ارتباط برای دریافت فایل خود کلاینت است، این حالت را در شکل زیر مشاهده می کنید:



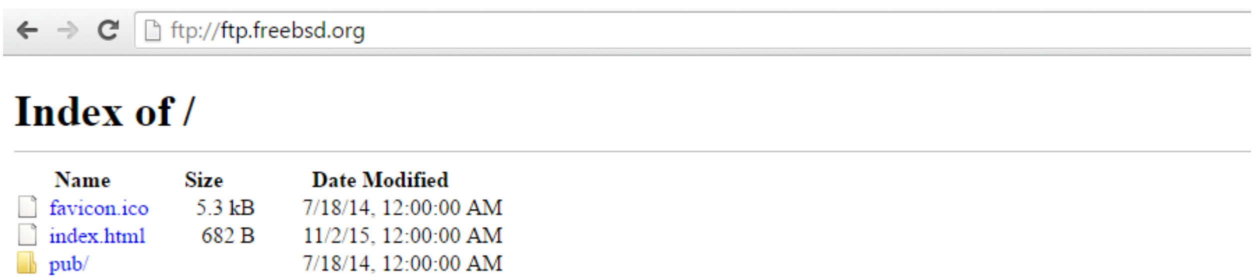
بررسی حالت passive در ftp

روشهای برقرار ارتباط با سرورftp :

شما به دو طریق می توانید با یک سرور ftp ارتباط برقرار کنید، هم از طریق خط فرمان و هم از طریق یک مرورگر وب. در ادامه این دو روش را برای شما توضیح خواهم داد.

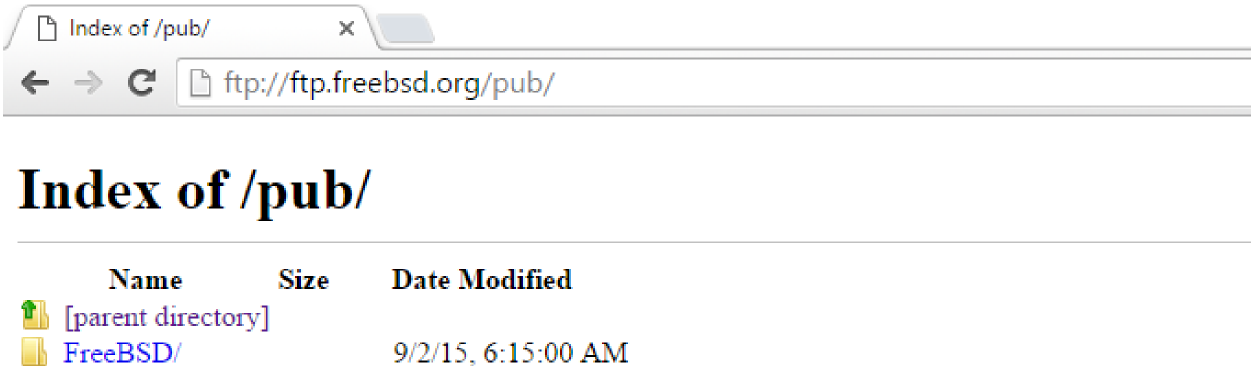
روش استفاده از یک مرورگر وب

اگر شما از DNS سرور برای برقراری ارتباط با سرور های خود استفاده می کنید می توانید از نام بجای آدرس IP برای متصل شدن به سرور استفاده کنید، شما همچنین می توانید از آدرس ip هم به صورت مستقیم برای برقرار کردن ارتباط استفاده کنید. در این روش ما قصد داریم که به سرور ftp مربوطه به سایت freebsd.org به آدرس ftp.freebsd.orgمتصل شویم. در این سرور حالت Anonymous فعال بوده و شما نیاز به نام کاربری و رمز عبور ندارید. یک مرورگر وب را باز کنید و آدرس ftp.freebsd.orgرا در قسمت ادرس وارد کنید به صورت زیر تا به سرور متصل شویدبرای اطمینان حاصل کردن از اینکه شما با استفاه از پروتکل ftp به سرور متصل می شوید می توایند خط //:ftp را به قبل از آدرس خود اضافه کنید تا مرورگر شما از ftp استفاده کند:



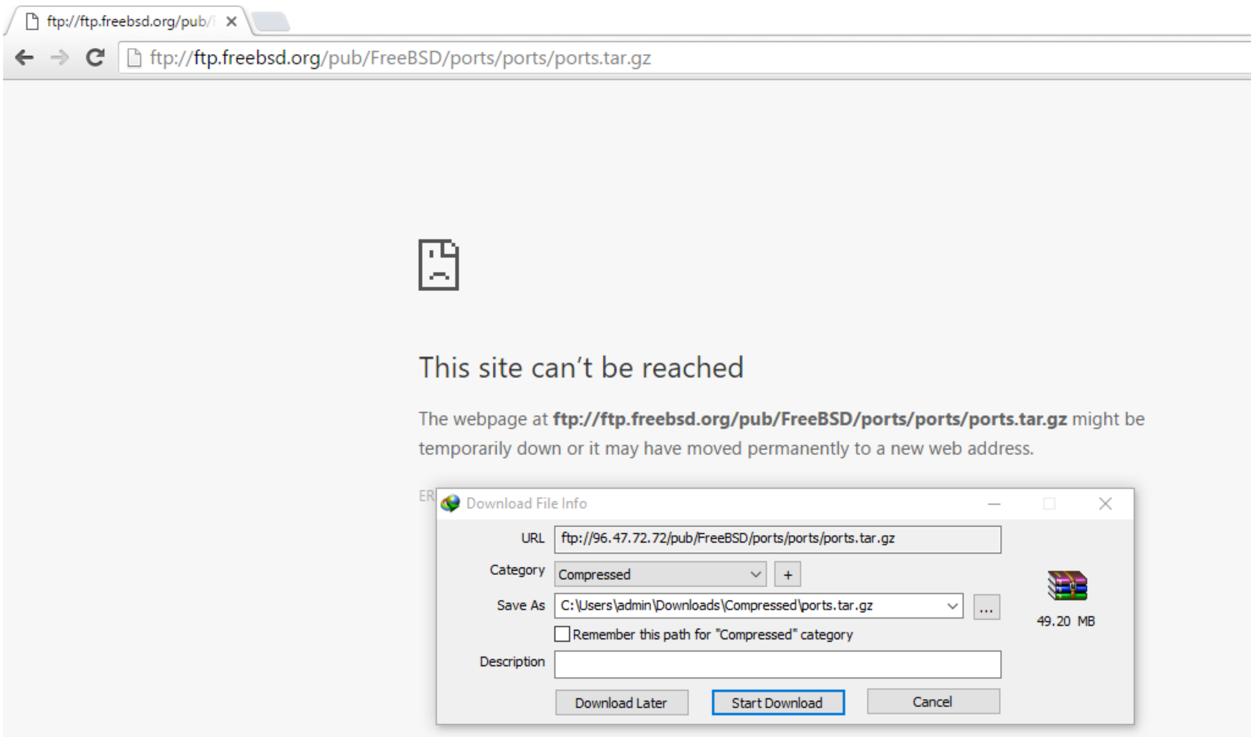
برقراری اتباط با ftp از طریق وب

در این بخش شما فایلها و پوشهای موجود برروی سرور را مشاهده می کنید پوشه های به / ختم می شود و با کلیک کردن برروی آنها می توانید به آنها وارد شوید در شکل زیر وارد پوشه pub شده اینم:



گشت وگذار در ftp با وب

همانطوری که مشاهده می کنید این بخش خود شامل پوشه دیگری به نام FreeBSD است، در بخش ادرس شما آدرس کامل این بخش یعنی ftp://ftp.freebsd.org/pub را مشاهده می کنید. برای رفتن به شاخه قبل کافیست که برروی parent directory کلیک کنید. برای دانلود کردن یک فایل کافیست که برروی ان کلیک کنید تا نرم افزارهای دانلود باز شده و عمل دانلود را انجام دهند، این عمل در شکل زیر نمایش داده شده است:



دانلود کردن از صفحه وب

ما قصد دریافت فایل ports.tar.gz را از آدرس ftp://ftp.freebsd.org/pub/FreeBSD/ports/ports/ports.tar.gz را داشته ایم شما می توانید از این ادرس هم به صورت مستقیم استفاده کنیم.

روش استفاده از خط فرمان

در این روش برقرار ارتباط شما نیاز به یک خط فرمان دارید باز هم مستقل از سیستم عامل و هرجایی که شما فرمان FTP را داشته باشید می توانید به سرور متصل شوید. کافیست که در خط فرمان ftp را با یک نام و یا یک ادرس ip اجرا کنید، برای برقراری حالت Anonymous کافیست که از سویچ a– قبل از نام سرور استفاده کنید، این فرمان در شکل زیر مشاهده می کنید:

```
Terminal
File  Edit  View  Terminal  Tabs  Help
root@FreeBSD:/home # ftp -a ftp.freebsd.org
Trying 96.47.72.72:21 ...
Connected to ftp.geo.freebsd.org.
220 This is ftp0.nyi.freebsd.org - hosted at NYI.net.
331 Please specify the password.
230-
230-This is ftp0.nyi.FreeBSD.org, graciously hosted by
230-New York Internet - NYI.net
230-
230-FreeBSD files can be found in the /pub/FreeBSD directory.
230-
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp>
```

ارتباط با ftp از طریق خط فرمان

همانطوری که مشاهده می کنید شما به سرور متصل شده اید و سیستم عامل سرور UNIX است و برای دریافت فایل باید از حالت binary استفاده کنید.اگر حالت Anonymous فعال نباشد شما باید قبل از ورود به سیستم از نام کاربری و رمزعبور استفاده کنید، در شکل زیر شما روش اتصال به سرور موجود برروی همان سیستم عامل به آدرس 127.0.0.1 را مشاهده می کنید.در این روش ابتدا فرمان ftp را اجرا کنید تا وارد خط فرمان ftp> شوید و بعد از اجرا کردن فرمان op شما می توانید آدرس سرور را در مقابل to وارد کنید، بعد از آن شما می توانید نام کاربری و رمز عبور را به صورت زیر وارد کنید، این بخش در شکل زیر نمایش داده شده است :

```
File  Edit  View  Terminal  Tabs  Help
root@FreeBSD:/home # ftp
ftp> op
(to) 127.0.0.1
Connected to 127.0.0.1.
220 FreeBSD FTP server (Version 6.00LS) ready.
Name (127.0.0.1:root): admin
331 Password required for admin.
Password:
230 User admin logged in.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> ls
229 Entering Extended Passive Mode (|||63679|)
150 Opening ASCII mode data connection for '/bin/ls'.
total 32
-rw-r--r--   1 admin  admin   1066 Jul 31 20:37 .cshrc
-rw-r--r--   1 admin  admin    252 Jul 31 20:37 .login
-rw-r--r--   1 admin  admin    163 Jul 31 20:37 .login_conf
-rw-------   1 admin  admin    379 Jul 31 20:37 .mail_aliases
-rw-r--r--   1 admin  admin    336 Jul 31 20:37 .mailrc
-rw-r--r--   1 admin  admin    817 Jul 31 20:37 .profile
-rw-------   1 admin  admin    281 Jul 31 20:37 .rhosts
-rw-r--r--   1 admin  admin    978 Jul 31 20:37 .shrc
226 Transfer complete.
```

ارتباط با خط فرمان

این برنامه دارای فرمانهای مختلفی است که در شکل زیر لیست کامل انرا مشاهده می کنید برای مشاهده این فرمانها کافیست که ؟ را تایپ و بعد Enter کنید تا لیست به صورت نمایش داده شده در شکل زیر برای شما نمایش داده شود:

```
Terminal
File  Edit  View  Terminal  Tabs  Help
Commands may be abbreviated.  Commands are:

!               epsv6           mget            preserve        sendport
$               exit            mkdir           progress        set
account         features        mls             prompt          site
append          fget            mlsd            proxy           size
ascii           form            mlst            put             sndbuf
bell            ftp             mode            pwd             status
binary          gate            modtime         quit            struct
bye             get             more            quote           sunique
case            glob            mput            rate            system
cd              hash            mreget          rcvbuf          tenex
cdup            help            msend           recv            throttle
chmod           idle            newer           reget           trace
close           image           nlist           remopts         type
cr              lcd             nmap            rename          umask
debug           less            ntrans          reset           unset
delete          lpage           open            restart         usage
dir             lpwd            page            rhelp           user
disconnect      ls              passive         rmdir           verbose
edit            macdef          pdir            rstatus         xferbuf
epsv            mdelete         pls             runique         ?
epsv4           mdir            pmlsd           send
ftp>
```

کلیه فرمان های موجود در FTP Client

فرمانهایی که استفاده زیادی در این بخش دارد فرمان های cd و ls است، در مقاله BSD به زبان ساده روش استفاده از این فرمان ها توضیح داده شده است کاربرد این دو فرمان برای نمایش فایلها و شاخه و رفتن به شاخه بعدی است، برای مشاهده شاخه ای که در آن قرار دارید می توانید از فرمان pwd استفاده کنید.

نکته :

فرمان ! یا همان علامت تعجب به شما این امکان را می دهد که در زمانی که به سرور وصل هستید به خط فرمان بازگشته و فرمانهای خود را اجرا کنید بدون اینکه ارتباط شما با سرور قطع شود، این فرمان در زمان دانلود فایلهایی که زمان بر هستند مفید است.

نمایش توضیحات هر فرمان

شما به راحتی می توانید توضیحات کمک کننده هر فرمان را با استفاده از فرمان help مشاهده کنید کافیست بعد از help کافیست که نام فرمان مورد نظر خود را وارد کنید در شکل زیر شما چند help از فرمانهای را مشاهده کنید:

```
Terminal
File Edit View Terminal Tabs Help
ftp> help ls
ls              list contents of remote path
ftp> help pwd
pwd             print working directory on remote machine
ftp> help get
get             receive file
```

نمایش توضیحات هر فرمان

دریافت و ارسال فایل

برای دریافت کردن فایل از سمت سرور باید از فرمان get استفاده کنید، برای دریافت یا باید در همان شاخه فایل مورد نظر باشید و یا آدرس آنرا بعد از فرمان get وارد کنید، فایل مورد نظر بعد از دانلود شدن در شاخه ای که شما قبل از فرمان ftp در قرار داشته اید ذخیره می شود مگر از فرمان lcd استفاده کنید و مسیر جدید را تعیین کنید در شکل زیر همان فایل ports.tar.gz که در بخش قبلی از طریق مرورگر دانلود کردید را در این بخش هم با استفاده از خط فرمان دانلود می کنید به صورت نمایش در شکل زیر:

```
Terminal
File Edit View Terminal Tabs Help
250 Directory successfully changed.
ftp> ls
229 Entering Extended Passive Mode (|||51720|)
150 Here comes the directory listing.
drwxr-xr-x    2 ftp      ftp             4 Jul 08  2014 distfiles
drwxr-xr-x    2 ftp      ftp             4 Jul 08  2014 local-distfiles
drwxr-xr-x    2 ftp      ftp             4 Oct 28  2012 ports
226 Directory send OK.
ftp> cd ports
250 Directory successfully changed.
ftp> ls
229 Entering Extended Passive Mode (|||64311|)
150 Here comes the directory listing.
-rw-r--r--    1 ftp      ftp           993 Jul 16  2013 README.TXT
lrwxr-xr-x    1 ftp      ftp            47 Oct 28  2012 ports.tar.gz -> ../../de
velopment/tarballs/ports_current.tar.gz
226 Directory send OK.
ftp> pwd
Remote directory: /pub/FreeBSD/ports/ports
ftp> get ports.tar.gz
local: ports.tar.gz remote: ports.tar.gz
229 Entering Extended Passive Mode (|||51965|)
150 Opening BINARY mode data connection for ports.tar.gz (51591521 bytes).
  2% |                                   |  1193 KiB   91.78 KiB/s    08:55 ETA
```

دانلود کردن فایل

شما در خط اخر از این شکل مقدار زمان مورد نیاز برای دریافت، سرو و حجم فایل برای شما نمایش داده می شود. بعد از پایان دانلود شما می توانید از فرمان quit برای خروج از ftp و قطع کردن ارتباط با سرور استفاده کنید و سرور پیغامی برای شما در قالب goodbye نمایش می دهد.

```
ftp> pwd
Remote directory: /pub/FreeBSD/ports/ports
ftp> get ports.tar.gz
local: ports.tar.gz remote: ports.tar.gz
229 Entering Extended Passive Mode (|||51965|)
150 Opening BINARY mode data connection for ports.tar.gz (51591521 bytes).
100% |***********************************| 50382 KiB  109.71 KiB/s    00:00 ETA
226 Transfer complete.
51591521 bytes received in 07:39 (109.66 KiB/s)
ftp> quit
221 Goodbye.
root@FreeBSD:/home #
```

نمایش وضعیت دانلود

همانطوری که مشاهده کردید فایل به صورت کامل دریافت شده و با فرمان quit شما ارتباط خود را با سرور قطع کرده اید.

برای ارسال فایل در ابتدا شما باید به شاخه ای که قصد اپلود کردن فایل را دارید دسترسی نوشتن داشته باشید و بعد از برقرار ارتباط با سرور و رفتن به شاخه مورد نظ از فرمان put استفاده کنید بعد از آن باید نام فایل با آدرس ذخیره شده برروی هارد خود را وارد کنید. در شکل زیر به سرور 127.0.0.1 با کاربر admin متصل شده این و فایل ports.tar.gz را با put به سمت سرور ارسال می کنیم:

```
Terminal
File Edit View Terminal Tabs Help
root@FreeBSD:/home # ftp
ftp> op
(to) 127.0.0.1
Connected to 127.0.0.1.
220 FreeBSD FTP server (Version 6.00LS) ready.
Name (127.0.0.1:root): admin
331 Password required for admin.
Password:
230 User admin logged in.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> put ports.tar.gz
local: ports.tar.gz remote: ports.tar.gz
229 Entering Extended Passive Mode (|||52680|)
150 Opening BINARY mode data connection for 'ports.tar.gz'.
100% |***********************************| 50382 KiB   45.62 MiB/s    00:00 ETA
226 Transfer complete.
51591521 bytes sent in 00:01 (45.48 MiB/s)
ftp> quit
221 Goodbye.
root@FreeBSD:/home #
```

برقراری اتباط با سرور FTP

راه اندازی کردن ftp در FreeBSD:

برای راه اندازی کردن سرور ftp شما نیاز به نصب هیچگونه برنامه اضافی ندارید و فقط کافیست که این برنامه را از طریق فرمان های rc و یا فرمان service راه اندازی کنید. در ادامه با روش های پیکربندی به سرویس آشنا می شوید.

فایل /etc/ftpusers:

یکی از مهمترین بخش های پیکربندی این سیستم تعیین کردن نوه دسترسی کاربران محلی به سرور ftp است، اگر قصد دارید که در حالت Anonymous از این سرویس استفاده کنید نیاز به تنظیمات این بخش ندارید. در هر صورت در سیستم عامل FreeBSD یک سری کابران سیستم وجود دارند که در زمان نصب به سیستم اضافه می شوند و برای راه اندازی هر سرویس سیستمی ایجاد شده اند. برای جلوگیری از ورود این چنین کاربران در FreeBSD فایلی به نام ftpusers ایجاد شده است که نام هم کاربری که در آن قرار داشته باشد نمی تواند از طریق ftp به سیستم شما متصل شود. به این نکته توجه کنید که کاربرانی را که شما در مراحل بعد از نصب ایجاد می کنید در این فایل وجود ندارد و برای اضافه کردن کاربران در این فایل کافیست که نام کاربر را در خطی جداگانه در این فایل اضافه کنید.

محدود کردن دسترسی به فایلها با /etc/ftpchroot:

در بعضی از موارد نیاز می شود که دسترسی کاربر وارد شده از طریق ftp را برروی یک شاخه خاص محدود کنید، این امر به این خاطر است که زمانی که کاربر محلی از طریق سرویس ftp به سیستم وارد می شود می تواند در همه شاخه ها و پوشه ها گشت و گذار کند و از زمان ls برای مشاهده لیست فایلها و غیر استفاده کند. برای محدود کردن این دسترسی فایل دیگری باید ایجاد کنید که به صورت پیش فرض در سیستم وجود ندارد و باید ایجاد شود. نام این فایل ftpchroot است که در شاخه /etc باید ایجاد شود. این فایل توسط برنامه Ftpd در زمان شروع یک ارتباط ftp خوانده می شود البته بعد از زمانی که کاربر در سیستم با نام کاربری و رمز عبور درست وارد شد.

هر خط در این فایل به یک کاربر و یا گروه خاصی مربوط می شود، اگر نام کاربر و یا گروهی که کاربر به آن تعلق دارد وارد سیستم شما و با خطی در این فایل منطبق شود شاخه root کاربر به شاخه ای که در مقابل نام کاربر در این فایل نوشته شده است بر کاربر اعمال می شود.

ترتیب در خواندن این فایل از بالا به پایین است و اولین خطی که با نام کاربری و یا گروه شما منطبق می شود دسترسی آن اعمال می شود. هر بخش هم در این فایل با یک space از هم جدا می شود. اولین فیلد در این فایل نام کاربری و یا گروه مورد نظر شماست. اگر قبل از نام گروه علامت @ قرار گرفته باشد یعنی همه کاربرانی که در آن گروه عضو هستند. اگر هم خطی با فقط @ شروع شده باشد یعنی همه کاربران. بخش دوم که باید space از بخش اول جدا می شود شاخه است که کاربر و یا گروه کاربران به آن محدود خواهند شد. اگر هم هیچ مسیری در این بخش وارد نکرده باشید کاربر فقط به شاخه log in خود محدود می شود.در شکل زیر یک مثال از این فایل را مشاهده می کنید:

```
Terminal
File  Edit  View  Terminal  Tabs  Help
^[ (escape) menu  ^y search prompt  ^k delete line   ^p prev li   ^g prev page
^o ascii code     ^x search         ^l undelete line ^n next li   ^v next page
^u end of file    ^a begin of line  ^w delete word   ^b back 1 char
^t top of text    ^e end of line    ^r restore word  ^f forward 1 char
^c command        ^d delete char    ^j undelete char ^z next word
=====line 6 col 21 lines from top 9 ==========================================
#admin chroot to /var/ftp
admin /var/ftp

#all user in user1 groups chroot to /var/ftp/user
@user1 /var/ftp/user

#every user chroot to public_ftp

@ /var/ftp/public_ftp
```

نمایش فایل ftpchroot

در خط اول کاربر admin به شاخه /var/ftp محدود می شود، در خط دوم همه کاربران عضو در گروه user1 به شاخه /var/ftp/user محدود می شوند و در خط پایانی همه کابران به شاخه /var/ftp/public_ftp محدود خواهند شد.

فعال کردن حالت Anonymous:

برای راه اندازی کردن سرور ftp در حالت Anonymous شما باید یک کاربر به نام ftp ایجاد کنید و این کاربر بتواند در سیستم وارد شود و نام آن در فایل ftpusers نباشد و در زمان وارد شدن از طریق نام کاربری ftp یا Anonymous اقدام کنید و یا از سویچ –a استفاده کنید(برای مطالعه بیشتر در مورد راه های اتصال به سرور Ftp مقاله ftp چیست را مطالعه کنید) در صورت نیاز به رمز عبور یک آدرس میل وارد کنید و برنامه chroot به صورت خودکار بعد از وارد شدن شما به سیستم مسیر شاخه home کاربر را برای شما chroot می کند.

پیغام های ورود به سیستم:

در زمان وارد شده به سرور Ftp از طریق خط فرمان برای شما دو نوع پیغام خوش آمد گویی نمایش داده می شود، اولین پیغام در برقرار ارتباط با سرور برای شما نمایش داده می شود که این پیغام ها از یک فایل متنی به نام ftpwelcome خوانده شده و نمایش داده می شود که در شاخه /etc قرار گرفته است. شما می توانید هر متنی را دوست دارید نوشته و این پیغام به صورت عمومی منتشر می شود.

بعد از وارد شدن کاربر به سیستم و به اصطلاح log in کردن صحیح با نام کاربری و رمز عبور معتبر به سیستم پیغام دیگری برای کاربر قابل نمایش است که این پیغام هم از فایل متنی به نام ftpmotd خوانده شده و نمایش داده می شود که در شاخه /etc قرار دارد شما می توانید هر پیغامی را که مایل هستید کاربر بعد از وارد شدن به سیستم مشاهده کند را در این بخش وارد کنید.

راه اندازی سرور FTP:

برای راه اندازی دایمی این سرور از طریق فایل /etc/rc.conf اقدام نمایید و خط زیر را در این فایل اضافه کنید:

```
ftpd_enable="YES"
```

بعد از اضافه کردن این بخش از دو طریق می توانید راه اندازی سرور را انجام دهید روش اول با استفاده از فرمان service به صورت زیر:

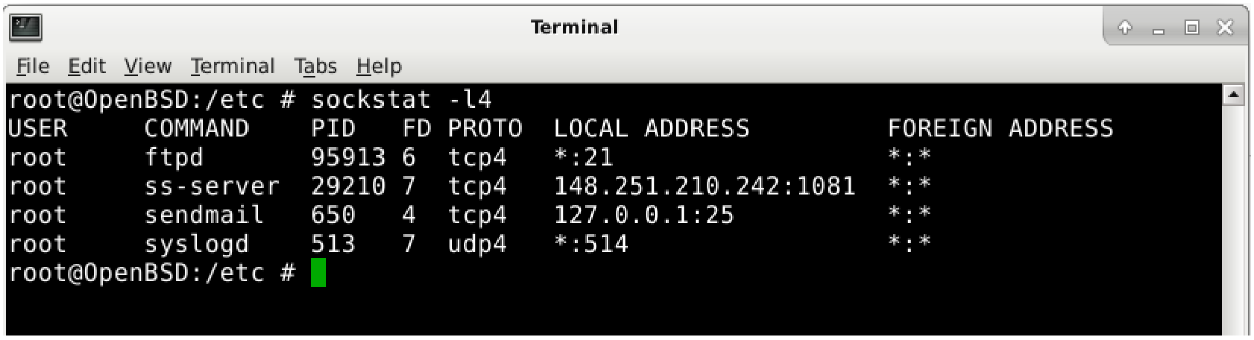
```
#service ftpd start
```

و یا فرمان های rc به صورت زیر:

```
#/etc/rc.d/ftpd start
```

چک کردن فعال بودن سرویس

یکی از راه های چک کردن سرویس باز بودن پورت باز سیستم است در FreeBSD فرمانی است به نام sockestat که وضعیت پورت های باز را نمایش می دهد در شکل زیر خروجی این فرمان نمایش داده شده است:



نمایش وضعیت سرویسهای شبکه

در خروجی فرمان بالا مشاهده می کنید که برروی پورت 21 برنامه ftpd اماده اراه سرویس است.

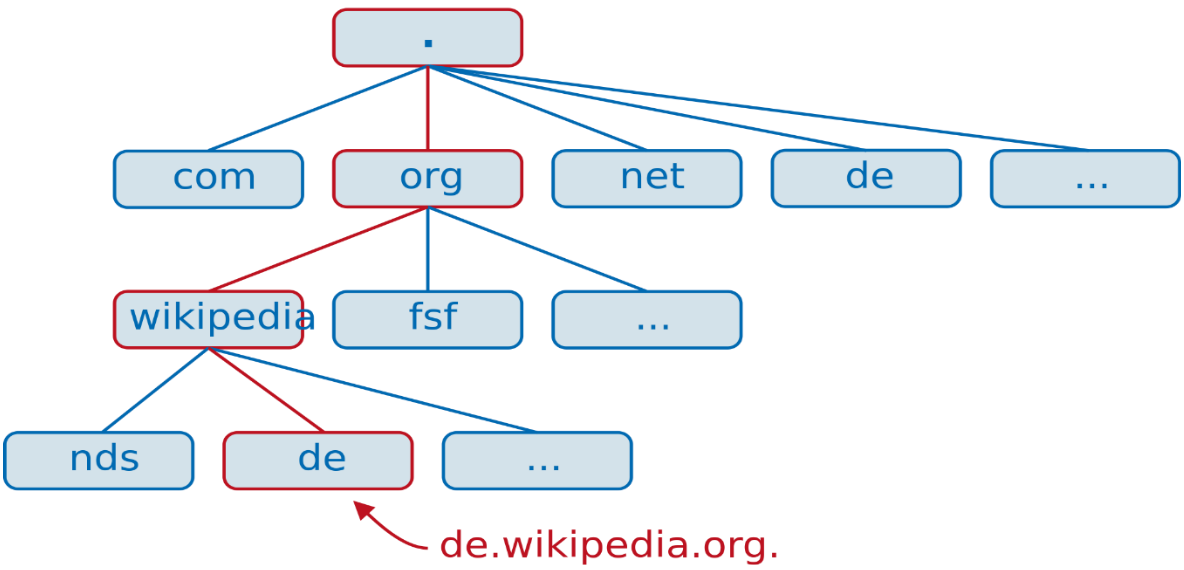
راه اندازی کردن سرور DNS در FreeBSD

DNS چیست؟

ذهن انسان توانایی به خاطر سپاری اسم ها را بهتر از اعداد را دارد، در دنیای شبکه کامپیوترها از آدرس های IP برای برقراری ارتباط استفاده می کنند. در این میان راهی برای تبدیل کردن این نام به آدرس IP وجود دارد آن هم استفاده سرور DNS در شبکه است . این سرور وظیفه تبدیل کردن نام به آدرس IP و برعکس را برعهده دارد و یکی از سرورهای مورد نیاز برای دسترسی به شبکه جهانی اینرتنت است. حجم زیادی از ترافیک اینترنت به پرس و جو کردن بین کاربران و سرورهای DNS برای برقرار کردن ارتباط با سیستم های مورد نظر می شود. برای کاهش این ترافیک یک مدلی از DNS ایجاد شده است به نام cache DNS که وظیفه انها ذخیره کردن اطاعاتی است که کاربران از ان پرس و جو می کنند. در ادامه با انواع DNS سرورها آشنا می شوید.

روش پرس و جو

پرس و جو کردن از سرورهای DNS به صورت سلسله مراتبی انجام می شود و هر سروری نیازی نیست که اطلاعات کاملی از همه سیستم ها و نام ها داشته باشد فقط باید نام سرورهای دیگری را داشته باشد که اون سرور ها نام سرور بعدی را دارد. در شکل زیر سلسه مراتبی بودن تقسیم بندی نام ها در اینترنت را برای شما نمایش داده شده است :



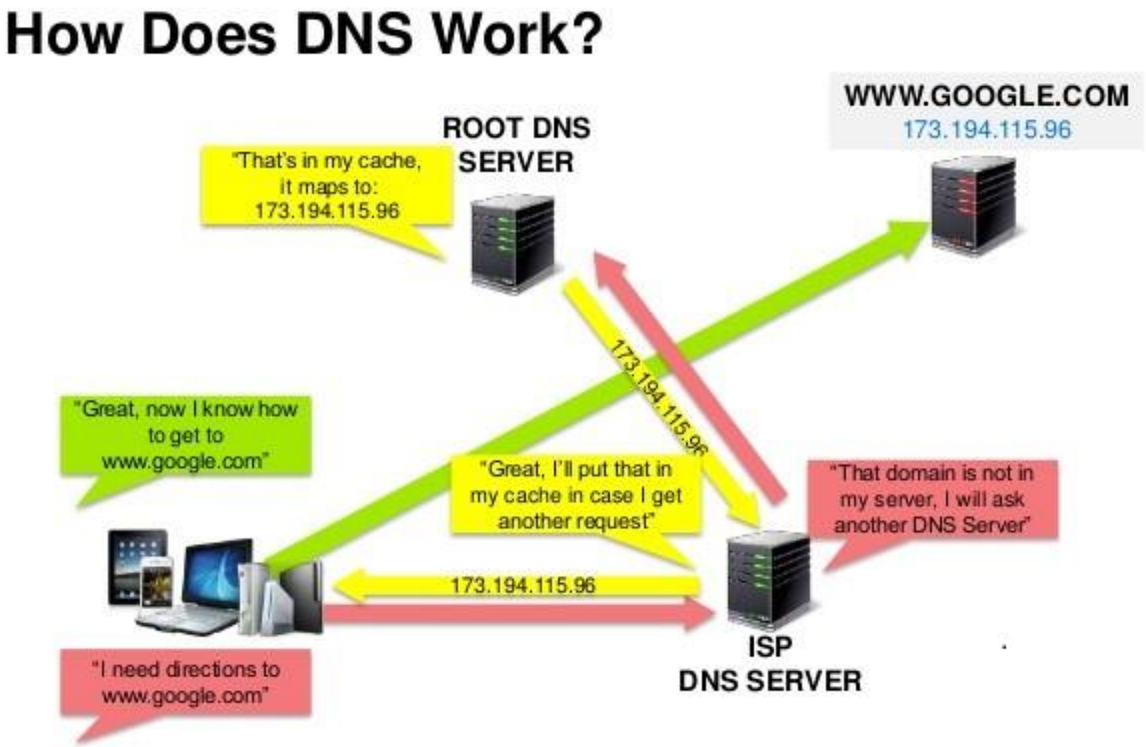
مدل جستجو کردن دامین ها در اینترنت

همانطوری که مشاهده می کنید یک سرور root به نام . یا همان نقطه در اینترنت وجود دارد که تعداد انها در حال حاضر 13 عدد است که آدرس IP انها ثابت است، دلیل ثابت بودن آدرس IP آنها این است که این سرور ها حالت راهنما برای کاربران را دارند، root server ها از سرورهای لایه بعدی یعنی سرورهای DNS مربوطه به حوزه های com net و سرورهای DNS مربوط به هر کشور مطلع هستند و زمانی که درخواستی را دریافت می کنند به تناسب این بخش ادرس سرور بعدی را در اختیار client قرار می دهند و سرورهای level بعدی فقط بخش دوم نام یعنی آدرس ip نام های اصلی هر بخش را در دارند، در

مرحله بعدی این DNS سرور های حوزه های هر نام هستند که ip هر سرور خود را در اختیار client قرار می دهند برای مثال DNS سرور حوزه Wikipedia از ip سرور بخش de این نام حوزه خبر دارد و آنرا می تواند در اختیار client قرار دهد. به این صورت سلسله مراتبی می توانید به هر سرور موجود در اینترنت دسترسی داشته باشید.

سایتی به نام http://www.root-servers.org وجود دارد که شما می توانید لیست پخش شده از DNS سرورها را مشاهده کنید.

در شکل زیر روش جستجو برای باز کردن سایت www.google.com برای شما به تصویر کشده شده است، به این نام در اصطلاح FQDN می گویند که .com به این معنا است که این سایت یک سایت تجاریست، google نام حوزه و یا همان اسم سایت است و در بخش آخر هم www. قرار دارد که بدین معنی است که شما با سرور www کار دارید. سرور های دیگری هم در این نام حوزه قرار دارند مثل mail.google.com و غیره.



روش جستجو کردن یک نام

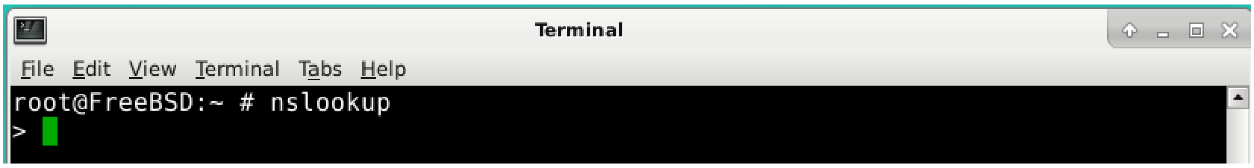
در شکل بالا مشاهده می کنید که ابتدا client از سرور DNS شرکت سرویس دهنده خود با همان ISP در مورد آدرس IP سوال می پرسد و این DNS سرور هست که سایر مراحل را طی می کند تا آدرس را تهیه کرده و به Client می دهد تا بتواند ارتباط خود را برقرار کند.

برنامه های جستجوی نام

دو برنامه برای سوال و جواب کردن با سرور های DNS وجود دارد، برنامه اول به نام nslookup است که برنامه ای قدیمیست و در همه جا وجود دارد. برنامه بعدی dig است که خروجی بیشتری برای شما نمایش می دهد و در سیستم عامل های که از BIND استفاده می کنند وجود دارد و اطلاعات بیشتری برای شما نمایش می دهد و در FreeBSD هم وجود دارد، اگر هم قصد دارید که از این برنامه استفاده کنید و برروی سیستم شما نصب نیست به سایت http://dig-nslookup.nmonitoring.com مراجعه کنید.

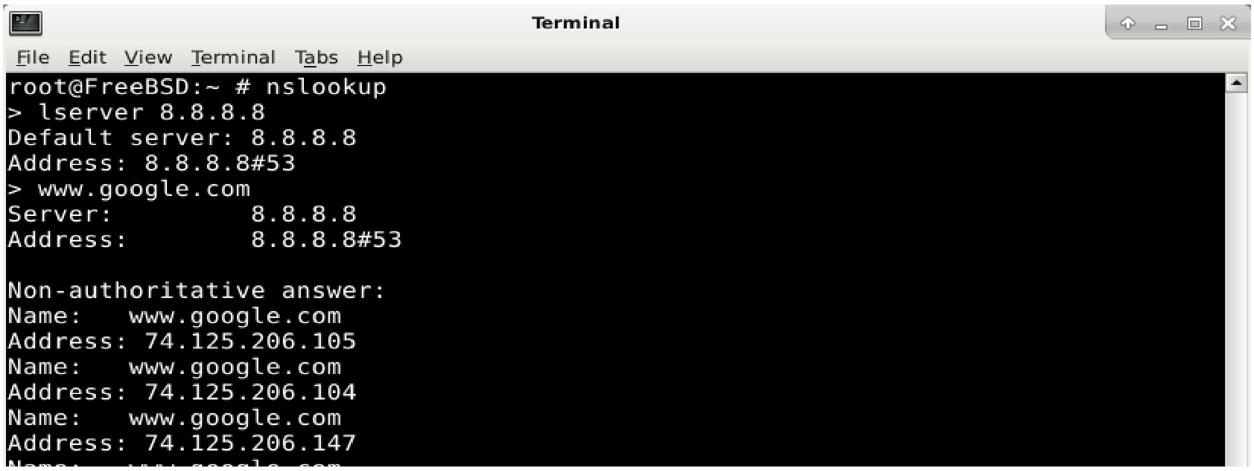
برنامه nslookup

این برنامه در حال حاضر در FreeBSD ورژن 10 به بعد دیگر وجود ندارد باید برنامه bind-tools را از شاخه برنامه ها به مسیر /usr/ports/dns/bind-tools نصب کنید.(برای آموزش روش نصب برنامه در FreeBSD به مقاله آن به نام مقدمه ای بر نصب برنامه در FreeBSD مراجعه کنید) . شما با استفاده از این فرمان می توانید به سرور های DNS مختلف متصل شوید و به پرس و جو آدرس IP نام حوزه خود بپردازید، اگر سروری که به آن متصل هستید قابلیت تبدیل IP به نام را داشته باشید شما می توانید از این فرمان استفاده کنید. بعد از اجرا این فرمان شما وارد خط فرمان این برنامه می شوید مثل شکل زیر:



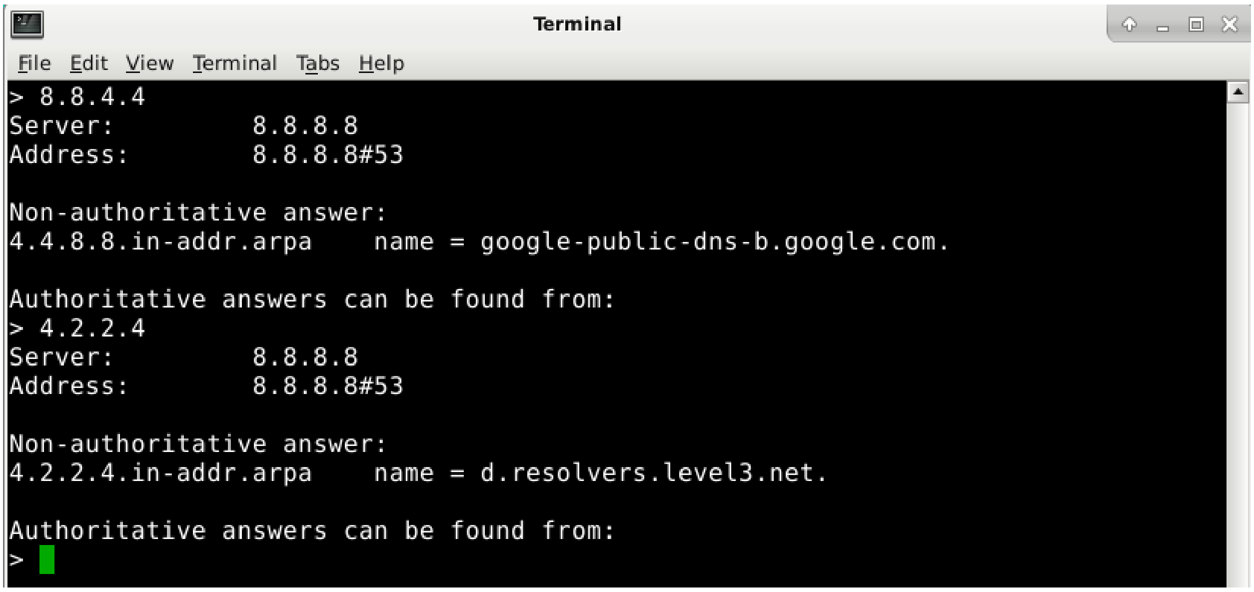
فرمان nslookup

علامت < بدین معناست که شما وارد برنامه شدید و می توانید فرمان های این بخش را اجرا کنید، برای تغییر دادن سرور پیش فرض از فرمان lserver و بعد آدرس IP سرور استفاده کنید، برای جستجو هم کافیست که نامی که مورد نظر است را وارد کنید . این دو امر در شکل زیر نمایش داده شده است:



جستجوی کردن با فرمان nslookup

برای جستجو بر اساس IP کافیست که آدرس IP مورد نظر را وارد کنید این بخش در شکل زیر نمایش داده شده است:



تغییر دادن آدرس سرور DNS پیش فرض

برنامه dig

در FreeBSD و سیستم عامل هایی که از BIND استفاده می کنند فرمانی وجود دارد به نام dig که عمل جستجو را انجام می دهند، این فرمان کل اطلاعات یک zone را برای شما نمایش می دهد. بعد اضافه شدن unbound و حذف شدن dind از FreeBSD این برنامه هم از FreeBSD حذف شده و نیاز به نصب bind-tools دارید. استفاده کردن از این برنامه بسیار ساده است و شما می توانید به راحتی بعد از فرمان dig نام حوزه خود را وارد کنید تا اطلاعات کاملی را مشاهده کنید به صورت زیر:

```
root@FreeBSD:~ # dig www.mabedini.com

; <<>> DiG 9.10.3-P4 <<>> www.mabedini.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 19500
;; flags: qr rd ra; QUERY: 1, ANSWER: 2, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; MBZ: 0005 , udp: 8192
;; QUESTION SECTION:
;www.mabedini.com.              IN      A

;; ANSWER SECTION:
www.mabedini.com.       5       IN      CNAME   mabedini.com.
mabedini.com.           5       IN      A       5.135.219.2

;; Query time: 165 msec
;; SERVER: 192.168.233.2#53(192.168.233.2)
;; WHEN: Mon Aug 01 00:41:52 IRDT 2016
;; MSG SIZE  rcvd: 75

root@FreeBSD:~ #
```

خروجی فرمان dig در جستجو کردن

برای مثال شما اطلاعات mabedini.com را در شکل بالا مشاهده می کنید. برای تغییر دادن سرور خود کافیست که بعد از فرمان dig از @ و بعد نام سرور استفاده کنید به صورت زیر :

```
root@FreeBSD:~ # dig @4.2.2.4 mabedini.ir

; <<>> DiG 9.10.3-P4 <<>> @4.2.2.4 mabedini.ir
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 26270
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 8192
;; QUESTION SECTION:
;mabedini.ir.                   IN      A

;; ANSWER SECTION:
mabedini.ir.            14391   IN      A       5.135.219.2

;; Query time: 149 msec
;; SERVER: 4.2.2.4#53(4.2.2.4)
;; WHEN: Mon Aug 01 00:45:38 IRDT 2016
;; MSG SIZE  rcvd: 56

root@FreeBSD:~ #
```

تغییر دادن آدرس سرور DNS پیش فرض در dig

برنامه نمایش اطلاعات از همه سرورهای موجود در یک حوزه ای کلمه ANY در بعد از نام حوزه استفاده کنید برای مثال در شکل زیر خروجی این فرمان را مشاهده می کنید:

```
Terminal
File Edit View Terminal Tabs Help

; <<>> DiG 9.10.3-P4 <<>> @4.2.2.4 mabedini.ir ANY
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 32787
;; flags: qr rd ra; QUERY: 1, ANSWER: 5, AUTHORITY: 0, ADDITIONAL: 2

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 8192
;; QUESTION SECTION:
;mabedini.ir.                   IN      ANY

;; ANSWER SECTION:
mabedini.ir.            14400   IN      SOA     ns231.mihanwebhost.com. mihanspa
.gmail.com. 2016081303 14400 130 1209600 14400
mabedini.ir.            14400   IN      NS      ns232.mihanwebhost.com.
mabedini.ir.            14400   IN      NS      ns231.mihanwebhost.com.
mabedini.ir.            14187   IN      A       5.135.219.2
mabedini.ir.            14400   IN      MX      0 mail.mabedini.ir.

;; ADDITIONAL SECTION:
mail.mabedini.ir.       14400   IN      A       5.9.123.24
```

نمایش همه اطلاعات DNS یک سایت با استفاده از dig

نکته:
در FreeBSD ورژن 10 به بعد از فرمان driil استفاده می شود که روش استفاده از ان مثل فرمان dig است.

معرفی انواع سرورهای موجود DNS:

در بخش بعدی با انواع سرور های DNS و روشهای نصب و راه اندازی آنها آشنا میشود قبل از ان با انواع آنها به صورت مختصر آشنا می شوید، یک سری ازDNS سرورهای فقط در حالت Cache کار می کنند و وظیفه انها ذخیره کردن اطلاعات پرس و جوی کاربران است و zone از خود ندارند.

نوع دیگری هم از DNS سرورها قابلیت cache نداشته و فقط در مورد حوزه هایی که در zone خود وجود دارند اطلاعات را در اختیار client ها قرار می دهند، به این نوع از Authoritative است.

در بعضی از موارد یک سرور می تواند هر دو عمل بالا را با هم انجام دهند.

نصب DNS unbound در FreeBSD

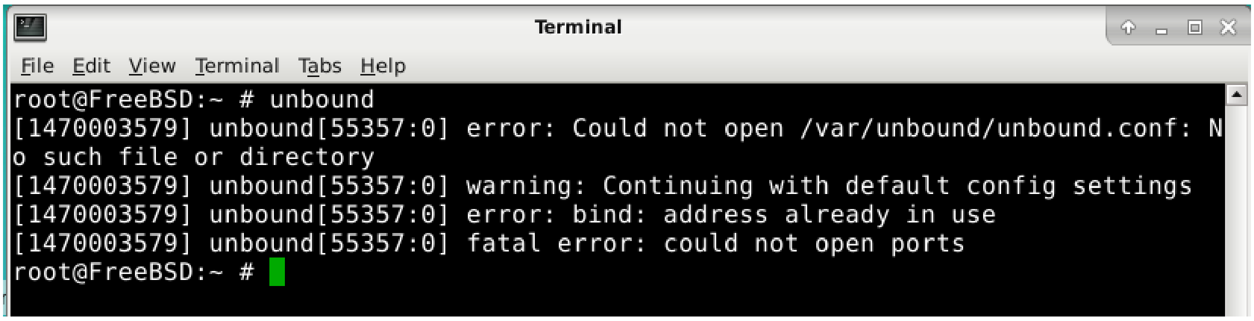
از ورژن 10 به بعد در FreeBSD DNS سرور BIND از سیستم عامل حذف شد و بجای آن unbound به این سیستم اضافه شد. ان سرور فقط برای استفاده از Chaching استفاده می شود و برای استفاده از حالت aouthoritive باید از سیستم ports برنامه مورد نظر خود را نصب کنید. برنامه unbound به صورت پیش فرض از ورژن 10.1 در FreeBSD نصب شده است و به صورت پیش فرض فقط برای کاربر سیستم محلی یا همان localhost عمل کش کردن را انجام می دهد، برای دسترسی به قابلیت های بیشتر از آن unbound را از سیستم ports به آدرس /usr/ports/unbound برنامه را نصب کنید و یا از فرمان زیر از طریق بسته های این کار را انجام دهید:

```
#pkg install unbound
```

در صورتی که شما نیاز به تنظیمات بیشتری داشته باشید بهتر است که این برنامه را از طریق سیستم ports نصب کنید.

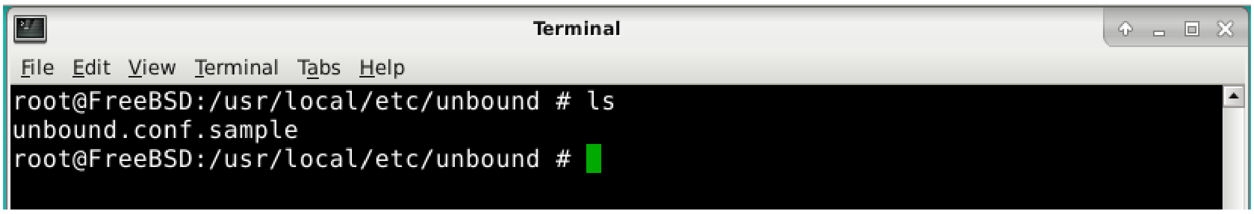
راه اندازی Unbound:

بعد از نصب برنامه unbound را با استفاده از این فرمان راه اندازی کنید تا از نصب شدن برنامه برروی سیستم خود مطمین شوید، خروجی این فرمان را در شکل زیر مشاهده می کنید:



خروجی فرمان unbound بعداز نصب کردن

همانطوری که مشاهده می کنید برای راه اندازی شما باید فایل unbounde.conf را در شاخه /var/unbound قرار دهید. اگر برنامه unbound را نصب کرده باشید نمونه ای از این فایل پیکربندی در زیر شاخه قرار می گیرد:



فایل پیش فرض سرور unbound

نام این فایل unbound.conf.sample است که باید انرا در شاخه مورد نظر کپی کنید. بعد از کپی کردن این فایل می توانید آنرا با استفاده از فرمان unbound-checkconf آنرا چک کنید و از عدم وجود خطا در این فایل مطمین شوید، اگر خطایی در این فایل وجود نداشته باشد خروجی به صورت زیر مشاهده می کنید:

```
Terminal
File  Edit  View  Terminal  Tabs  Help
root@FreeBSD:/usr/local/etc/unbound # unbound-checkconf
unbound-checkconf: no errors in /var/unbound/unbound.conf
root@FreeBSD:/usr/local/etc/unbound #
```

چک کردن پیکربندی سرور Unbound

فرمان unbound-control

برای مدیریت کردن این برنامه فرمانی وجود دارد به نام unbound-control که شما می توانید از طریق ان به مدیریت کردن برنامه unbound بپردازید در ابتدا برای راه اندازی آن باید فرمان unbound-control-setup برای تولید کردن کلید های مربوط به اتصال را ایجاد کنید، خروجی این فرمان در شکل زیر نمایش داده شده است:

```
Terminal
File  Edit  View  Terminal  Tabs  Help
root@FreeBSD:/usr/local/etc/unbound # unbound-control-setup
setup in directory /usr/local/etc/unbound
generating unbound_server.key
Generating RSA private key, 3072 bit long modulus
.............................................................................++
.++
e is 65537 (0x10001)
generating unbound_control.key
Generating RSA private key, 3072 bit long modulus
................................................................................
................................................................................
...................++
.......................++
e is 65537 (0x10001)
create unbound_server.pem (self signed certificate)
create unbound_control.pem (signed client certificate)
Signature ok
subject=/CN=unbound-control
Getting CA Private Key
Setup success. Certificates created. Enable in unbound.conf file to use
root@FreeBSD:/usr/local/etc/unbound #
```

خروجی فرمان unbound-control-setup

بعد از اتمام این بخش حال شما نیاز دارید که در فایل پیکربندی unbound.conf خط زیر را اضافه کنید:

```
# enable remote-control
remote-control:
        control-enable: yes
```

در مرحله بعد شما باید کلید هایی را که در زمان راه اندازی فرمان unbound-control-setup ایجاد کرده اید را در شاخه /var/unbound کپی کنید، شما باید 4 کلید را در شاخه ذکر شده کپی کنید و در اخر هم برای اطمینان از صحت این امر فرمان unbound-checkconf را اجرا کنید، این اعمال در شکل زیر نمایش داده شده است:

```
root@FreeBSD:~ # cd /usr/local/etc/unbound/
root@FreeBSD:/usr/local/etc/unbound # ls
unbound.conf.sample     unbound_control.pem     unbound_server.pem
unbound_control.key     unbound_server.key
root@FreeBSD:/usr/local/etc/unbound # cp unbound_server.key /var/unbound/
root@FreeBSD:/usr/local/etc/unbound # cp unbound_server.pem /var/unbound/
root@FreeBSD:/usr/local/etc/unbound # cp unbound_control.key /var/unbound/
root@FreeBSD:/usr/local/etc/unbound # cp unbound_control.pem  /var/unbound/
root@FreeBSD:/usr/local/etc/unbound # unbound-c
unbound-checkconf     unbound-control       unbound-control-setup
root@FreeBSD:/usr/local/etc/unbound # unbound-checkconf
unbound-checkconf: no errors in /var/unbound/unbound.conf
root@FreeBSD:/usr/local/etc/unbound #
```

انجام تنظیمات unbound

بعد از اجرا این بخش شما می توانید از فرمان unbound-control استفاده کنید. برای مثال با سویچ status از این فرمان شما خروجی زیر را مشاهده می کنید:

```
root@FreeBSD:~ # unbound-control status
version: 1.5.7
verbosity: 1
threads: 1
modules: 2 [ validator iterator ]
uptime: 164 seconds
options: control(ssl)
unbound (pid 55723) is running...
root@FreeBSD:~ #
```

نمایش وضعیت سرور با استفاده از فرمان unbound-control

در این فرمان شما وضعیت سرور unbound را مشاهده می کنید.

مختصری در مورد فایل unbound.conf

این فایل در قالب متغییر و مقدار طراحی شده است و مثل تمام فایلهای پیکربندی خطوطی که با # شروع می شود کامنت بوده و تاثیر در تغییرات سرور نخواهند داشت و برای چک کردن این فایل از فرمان unbound-checkconf استفاده کنید. این فایل به صورت کامل دارای توضیحات خوب برای اعمال تغییرات است. در ادامه با چند بخش مهم از تغییرات این فایل آشنا می شوید. این فایل دارای چند بخش اصلی است به نام های server python remote-control sub-zone forwarder-zone که بخش به بخش آنها مشخص شده اند و بعد از نام این بخشها به : ختم شده است و شما می توانید تنظیمات خود را اعمال کنید.

بخش اصلی server

سرور unbound برای افزایش امنیت در قالب chroot اجرا می شود و با ناک کاربری خاصی به نام unbound پردازش اصلی خود را اره اندازی می کند تا در صورت هک شدن برنامه به سیستم شما هکر به کلیه شاخه های سیستم شما دسترسی نداشته باشد و نتواند به سطح دسترسی root برسد، در بخش server از فایل unbound.conf برای تغییر دادن این بخشها متغییرهای در نظر گرفته شده است.

بخش username

در بخش :username شما می توانید نام کاربری که با استفاده از سرور unbound راه اندازی شود را مشخص کنید که به صورت پیش فرض این نام unbound است.

شاخه chroot

در بخش :chroot شما می توانید مسیر نام شاخه chroot را مشخص کنید. این آدرس نباید به / ختم شود و در صورتی که شما قصد نداشته باشید که از این قابلیت استفاده کنید مقدار آنرا "" قرار دهید.

مسیر log file

شما می توانید در بخش :logfile مسیر را که قصد داریدlog های برنامه در آن ذخیره شود را تعیین کنید.

مسیر pidfile

هر برنامه ای که در FreeBSD راه اندازی می شود که pid برای خود از سیستم دریافت می کند و این عدد برای هسته سیستم عامل برای برقرای ارتباط با سرور مورد استفاده قرار می گیرد شما می توایند در این بخش مسیر این فایل را تعیین کنید. این فایل شامل عددی است که همان pid تخصیص داده شده به ان می باشد.

بخش directory

شما در این بخش می توانید شاخه کاری برنامه را مشخص کنید که مقدار پیش فرض آن /usr/local/etc/unbound است . شما در این بخش می توانید این مسیر را تغییر دهید.

بخش hide-version

شما می توانید در این بخش ورژن برنامه خود را مخفی کنید که این بخش دارای دو متغییر yes و no است.

بخش تنظیم آدرس Ip ورژن های 4 و 6:

شما می توانید ورژن ip مورد استفاده در unbound را مشخص کنید، برای این کار باید در مقابل دو بخش do-ip4 مقدار yesرا قرار دهید تا از ورژن 4 استفاده کند و در صورتی که شما قصد دارید ورژن 6 را هم فعال کنید باید در مقابل do-ip6 مقدار yes را قرار دهید.

بخشdo-not-query-localhost

شما می توانید در این بخش تنظیم کنید که به درخواست های local host پاسخ داده شود و یا داده نشود با تعیین کردن دو بخش yes و no ، البته مقدار پیش فرض این بخش yes است و سرور به تمام درخواست های ارسال شده از سمت local hostپاسخ می دهد.

بخش interface

شما می توانید در این بخش مشخص کنید که برنامه unbound برروی کدام کارت های شبکه شما فعال باشد و به کدام sub netشما پاسخ درخواست ها را بدهد.

بخش اصلیRemote Control

شما در این بخش می توانید تنظیمات مربوط به پیکربندی از راه دور را تنظیم کنید و با استفاده از فرمان-unbound controlگزارشات و اعمال مدیریت را اعمال کنید در بخش بالا از این مقاله با پیکربندی ساده در این بخش آشنا شدید. در ادامه با زیر شاخه های دیگری از این بخش آشنا می شوید.

بخش control-enable

برای فعال کردن قابلیت کنترل از راه دور شما در ابتدا باید متغییر این بخش را تنظیم کنید که در دو حالت yes و no قرار دارد، البته در همین مقاله روش فعال سازی این بخش به صورت کامل توضیح داده شده است.

بخش control-interface

در این بخش شما می توانید آدرس IP که قصد دارید فقط از طریق ان به درخواست های فرمان unbound-control در شبکه گوش داده شود را مشخص کنید.

بخش control-port

این برنامه به صورت پیش فرض در برروی پورت 8953 در شبکه به فرمان های ارسال شده گوش می دهد و در این بخش شما می تواینذ این پورت مدیریت را تغییر دهید، برای اعمال شدن تغییر پورت شما باید برنامه را restart کنید.

بخشهایی هم در این قسمت وجود دارد که شما می توانید با استفاده از آنها مسیر فایلهای کلید ها را مشخص کنید.

بخش اصلی Forward Zone Options

برای فعال کردن حالت caching در unbound شما باید لیست از سرورهایی را که برنامه unbound باید درخواست ها را به سمت انها ارسال کند را مشخص کنید. برنامه unbound این قابلیت را دارد که برای هم نام حوزه مورد نظر شما لیستی از DNSسرورها را تعیین کنید و برای کش کردن درخواست های ارسال شده به سمت نام مورد نظر شما از آنها استفاده کند. برنامه unbound به صورت تصادفی از لیست تعیین شده استفاده می کند تا همه درخواست ها به سمت یک سرور ارسال نشود.

نکته :

برای اینکه همه درخواستهای دریافت شده از سمت Client ها به سمت سرور های تعیین شده ارسال شوند شما باید نام "." در بخش name مشخص کنید.

بخشname

در این بخش شما مشخص می کنید که برای یک نام حوزه خاص از که سرورهای برای پرس و جو استفاده شود. شما می توانید برای مثال مشخص کنید که برای نام google از سرورهایی که در بخش forward-addr برای پرس و جو استفاده کند. این دو بخش با هم برای شما حالت کش را فعال می کند و با قرار دادن نام "." و بعد لیست forward-addr ها می توانید حالت کش را فعال کنید.

بخش forward-addr

در این بخش ادرس ip لیست forwarder تعیین و مشخص می شود و هر خط شامل یک آدرس IP می باشد.

در زیر یک مثال از فایل پیکربندی برای حالت کش کردن برای کاربران شبکه LAN برای شما ارایه شده است:

```
## Simple recursive caching DNS
## unbound.conf
#
server:
        interface: 0.0.0.0
        access-control: 10.0.0.0/16 allow
        access-control: 127.0.0.0/8 allow
        access-control: 192.168.0.0/16 allow
forward-zone:
      name: "."
      forward-addr: 8.8.4.4        # Google
      forward-addr: 8.8.8.8        # Google
      forward-addr: 37.235.1.174   # FreeDNS
      forward-addr: 37.235.1.177   # FreeDNS
      forward-addr: 50.116.23.211  # OpenNIC
      forward-addr: 64.6.64.6      # Verisign
      forward-addr: 64.6.65.6      # Verisign
      forward-addr: 74.82.42.42    # Hurricane Electric
      forward-addr: 84.200.69.80   # DNS Watch
```

```
forward-addr: 84.200.70.40   # DNS Watch
forward-addr: 91.239.100.100 # censurfridns.dk
forward-addr: 109.69.8.51    # puntCAT
forward-addr: 208.67.222.220 # OpenDNS
forward-addr: 208.67.222.222 # OpenDNS
forward-addr: 216.146.35.35  # Dyn Public
forward-addr: 216.146.36.36  # Dyn Public
```

بعد از بخش access-control شما می توانید تعیین کنید که کدام رنج از آدرس های ip می تواند به سمت این سرور درخواست ارسال کنند.

مدیریت کردن استفاده از حافظه سیستم:

یکی از بخش های مهم پیکربندی سرور unbound مدیریت کردن حافظه است، در خطوط زیر شما تنظیمات بخش server را مشاهده می کنید که از طرف تیم ارایه کنند unbound تهیه شده است:

```
# example settings that reduce memory usage
    server:
        num-threads: 1
        outgoing-num-tcp: 1 # this limits TCP service, uses less buffers.
        incoming-num-tcp: 1
        outgoing-range: 60  # uses less memory, but less performance.
        msg-buffer-size: 8192   # note this limits service, 'no huge stuff'.
        msg-cache-size: 100k
        msg-cache-slabs: 1
        rrset-cache-size: 100k
        rrset-cache-slabs: 1
        infra-cache-numhosts: 200
        infra-cache-slabs: 1
        key-cache-size: 100k
        key-cache-slabs: 1
        neg-cache-size: 10k
        num-queries-per-thread: 30
        target-fetch-policy: "2 1 0 0 0 0"
```

```
harden-large-queries: "yes"
harden-short-bufsize: "yes"
```

راه اندازی در زمان boot سیستم

برای راه اندازی خودکار در زمان boot شدن سیستم خطوطی را که در شکل زیر نمایش داده شده است را در به فایل rc.conf اصافه کنید:


Terminal
File Edit View Terminal Tabs Help

```
root@FreeBSD:~ # /usr/local/etc/rc.d/unbound rcvar
# unbound
#
unbound_enable="NO"
#   (default: "")

root@FreeBSD:~ #
```

راه اندازی کردن در زمان Boot با استفاده از rc.conf

شما به راحتی می توانید از فرمان های rc موجود در شاخه /usr/local/etc/rc.d نسبت به start stop و restart کردن برنامه unbound بپردازید.

فایل zone چیست؟

قبل از راه اندازی DNS سرور در حالت authoritative شما باید روش ایجاد کردن فایل zone را فرا گیرید. فایل zone یکی از مهمترین بخش های DNS سرور است. این فایل یک فایل متنی است و شما می توانید با ویرایشگر متنی آنرا ایجاد کنید، فقط باید از قواعد آن پیروی کنید و نوع بخشهایی که در ان قابل اضافه شدن و یا اجباری هست را بیاموزید.

انواع رکوردهای موجود در zone:

چند رکورد خاص در zone فایل وجود دارد که در ادامه با آنها آشنا می شوید.

معروفترین و اولین رکورد A است که برای مشخص کردن کالاینت های آدرس IP ورژن 4 مورد استفاده قرار می گریند.

رکورد AAA برای نمایش کلاینتهای آدرس IP ورژن 6 مورد استفاده قرار میگیرند.

رکورد CNAME برای تعیین کردن یک نام مستعار استفاده می شود.

رکورد MX برای تعیین کردن Mail exchange دامنه استفاده می شود که همان سرور میل است.

از رکورد NS برای مشخص کردن Name Server ها یک دامین استفاده می شود.

رکورد اصلی SOA:

یکی از بخشهای مهم فایل zone رکورد SOA است که باید در همه فایلهای zone دامین ها وجود داشته باشد. این رکورد شامل اطلاعات authoritative مثل نام DNS سرور اصلی دامین، ادرس میل مدیری دامین، شماره سریال دامین و آخرین زمان تغییرات در فایل zone را مشخص می کند. این ها در یک خط نوشته می شود به صورت زیر:

```
example.com.  IN  SOA   ns.mabedini.com. admin@example.com. ( 2016120710 1d 2h 4w 1h )
```

شروع یک فایل zone با بخشی شروع می شود به نام @TTL که در حقیقت time to live است که شما برای مدت زمان Cache شدن اطلاعات دامین خود تعیین می کنید، در بسیاری از موارد سرورهای DNS متغییر های موجود در یک دامنه برای برای کاهش تافیک شبکه cache می کنند که بعد از اتمام مدت زمان تعیین شده در این بخش دوباره درخواست انتقال اطلاعات را به سمت سرور ارسال می کنند، این زمان بسته به نوع zone شما دارد.

نکته: خطوطی که با علامت ; شروع می شوند توضیحات است و هیچ تاثیری ندارند.

بعد از این خط شما باید رکورد اصلی SOA را تعریف کنید به صورت زیر:

```
abedini.com. IN     SOA    ns1.mabedini.com.  dnsmaster@abedini.com. (
                                  2016091401  ; Serial
                                  3H          ; refresh after 3 hours
```

```
                                1H            ; retry after 1 hour
                                1W            ; expire after 1 week
                                1D)           ; minimum TTL of 1 day
```

همانطوری که مشاهده می کنید در ابتدا باید نام دامین را مشخص کنید و بعد از SOA باید نام DNS سرور اصلی این دامین را مشخث کنید و بعد از آن هم باید آدرس میل مدیر دامین را بنویسید.

هر دامین باید یک شماره سریال داشته باشد، سرور های دیگری DNS برای بروزرسانی دیتابیس خود ابتدا رکورد SOA را از سرور درخواست می کند و در صورتی که شماره سریال دریافتی با شماره سریالی که در پایگاه داده خود دارند یکسان باشد بدین معناست که نیازی به دریافت تمام فایل zone ندارند و در صورتی که این سریال تغییر کرده باشد از سرور درخواست ارسال کل فایل zone را می کند برای بروزرسانی دیتابیس خودش. بعد از شماره سریال شما باید زمان های cache شدن دامین را مشخص کنید که به نوع سرور شما بستگی دارد .

در بخش بعدی شما حتما باید نام دو یا چند DNS سرور را برای دامین خود مشخص کنید به صورت زیر:

```
; Name Server
        IN      NS      ns1.mabedini.com.          ; VeriSign verteilt (anycast)
        IN      NS      ns2.mabedini.com.          ; ns.nasa.gov, Mountain View,
```

رکورد برای تعریف کردن این بخش NS است و شما باید دقت کنید که نام ها به . یا همان نقطه ختم می شوند.

در بخش بعدی باید سرور میل خود را مشخص کنید برای این کار باید به صورت زیر عمل کنید :

```
; Mail Exchanger
        IN      MX      50 mx1.mail.com. ; Your Mail Server
```

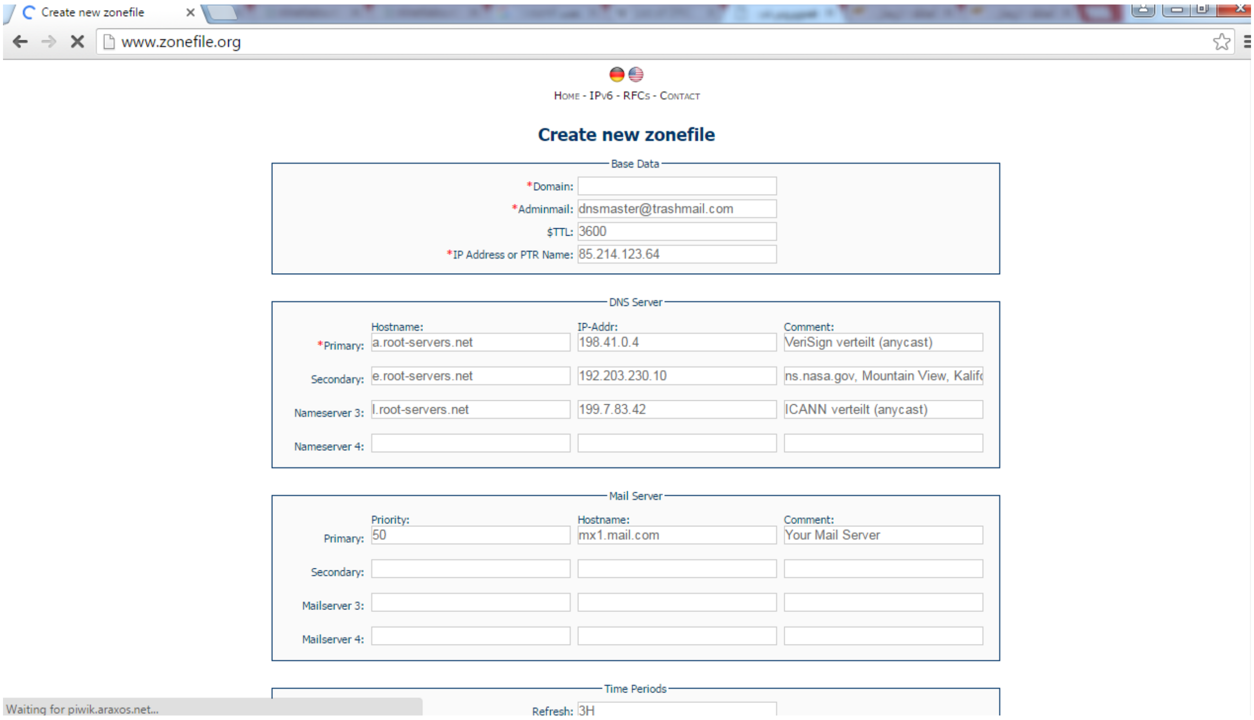
رکورد این بخش MX است.

اگر شما قصد داشته باشید که هاست های www و بدون www یا همان وب سرور برای دامین خود مشخص کنید باید به صورت زیر این بخش ها را تعریف کنید:

```
abedini.com.            IN A            85.214.123.64
www                     IN CNAME        85.214.123.64
```

بعد از نام abedini.com حتما باید نقطه اضافه کنید و نوع ان یک کلاینت با آدرس نوع 4 است و در خط بعدی این client در حقیقت یک نام مستعاری می شود از www این دامنه.

برای ایجاد کردن به صورت خودکار فایل zone دامنه خود می توانید به سایت http://www.zonefile.org مراجعه کنید و با پر کردن فیلدهای ان یک فایل کامل برای دامین خود ایجاد کنید، صفحه اول این سایت برای شما نمایش داده شده است:



سایت ایجاد خودکار فایل zone

راه اندازی NSD در FreeBSD

بعد از نصب کردن unbound که فقط قابلیت Forwarder کردن درخواست های کاربران را دارد و از آن برای cache کردن درخواست ها استفاده می شود نیاز به یک DNS سرور از نوع aouthoritive است که در این بخش سرور امن NSD را مورد بررسی قرار می دهیم. ساختار فایل پیکربندی و Remote control آن هم شبیه به Unbound است. این سرور فقط به درخواست های Zone خود جواب می دهد. در ادامه با نصب و پیکربندی و راه اندازی آن بیشتر آشنا می شوید.

نصب NSD

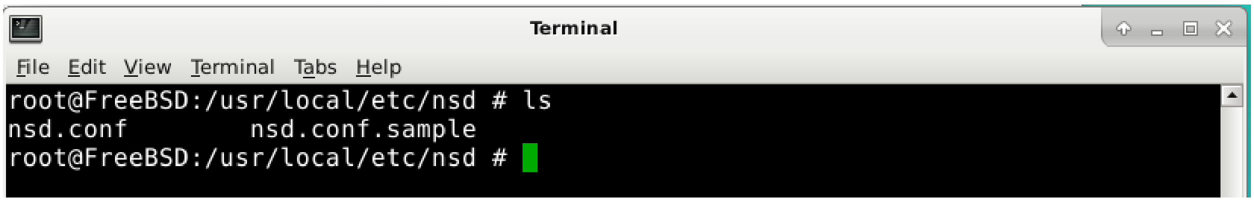
برای نصب شما هم می توانید از طریق سیستم ports اقدام کنید هم از طریق سیستم نصب بسته های باینری، در این بخش از روش نصب بسته های باینری استفاده می کنیم:

```
root@FreeBSD:~ # pkg install nsd
Updating FreeBSD repository catalogue...
FreeBSD repository is up-to-date.
All repositories are up-to-date.
The following 1 package(s) will be affected (of 0 checked):

New packages to be INSTALLED:
        nsd: 4.1.12

Number of packages to be installed: 1

The process will require 2 MiB more space.
337 KiB to be downloaded.

Proceed with this action? [y/N]: y
Fetching nsd-4.1.12.txz: 100%   337 KiB  20.3kB/s    00:17
Checking integrity... done (0 conflicting)
[1/1] Installing nsd-4.1.12...
===> Creating groups.
Creating group 'nsd' with gid '216'.
===> Creating users
Creating user 'nsd' with uid '216'.
[1/1] Extracting nsd-4.1.12: 100%
Message from nsd-4.1.12:
```

```
Message from nsd-4.1.12:
*************************************************************************
*
*   To run nsd from startup, add nsd_enable="YES" to your etc/rc.conf
*
*   Starting with nsd version 4 the old nsdc control program has been
*   replaced by nsd-control. This requires some manual setup with
*   nsd-control-setup and editing of the config files.
*
*   nsd-control is incompatible with nsdc so when that is used in scripts,
*   these should be adapted
*
*************************************************************************
root@FreeBSD:~ #
```

مراحب نصب کردن NSD در FreeBSD

بعد از نصب برنامه فایلهای این برنامه در زیر شاخه /usr/local/etc/nsd ذخیره می شود:



فایل پیکربندی NSD

در این شاخه دو فایل وجود دارد که فایلهای پیکربندی است و شما باید تمام تغییرات مورد نظر خود را در آن اعمال کنید در بخش بعدی شما با بخشهای مهم فایل nsd.conf آشنا می شوید.

پیکربندی سرور با فایلnsd.conf

برنامه NSD تمام تنظیمات خود را از فایل پیکربندی nsd.conf می گیرد. این فایل دارای بخشهای مهمی است که بخش اول آن بخش Server است که در آن شما می توانید تنظیمات اصلی سرور را تغییر دهید که بخش های مهم آن شامل لیست های زیر می شود:

بخش ip-address

در این بخش ما می توانید آدرس IP ورژن 4 و 6 که سرور برروی آن به درخواستها پاسخ دهد را مشخص کنید. با استفاه از ports@شما می توانید.

بخش ip-transparent

این بخش به NSD اجازه می ده که با آدرسهای IP غیر محلی هم کار کند. این بخش مفید است در زمانی که NSD به یک آدرس IP گوش کند که به کارت شبکه اضافه نشده است و به محض اضافه شدن آن آدرس IP به کارت شبکه فورا NSD می تواند برروی آن آدرس پاسخگو باشد. پیش فرض این بخش No است و این بخش شامل No و Yesاست.

بخش debug-mode

این بخش دو حالت yes و no را دارد و می تواند قابلیت دیباگ کردن را فعال و یا غیرفعال کند. این حالت هیچ بار اضافی به پردازش شما نمی دهد. به صورت پیش فرش این حالت خاموش است.

بخش do-ip4

این بخش شامل دو حالت yes و no است و اگر yes باشد یعنی از ورژن 4 ادرس ip استفاده می کند.

بخش do-ip6

این بخش شامل دو حالت yes و no است و اگر yes باشد یعنی از ورژن 6 ادرس ip استفاده می کند.

بخش database

در مقابل این بخش شما باید مسیر فایل دیتابیس مربوط به برنامه NSD را مشخص کنید که مسیر پیش فرض آن /var/db/nsd/nsd.db است. این فایل کامپایل شده اطلاعات zone هاست که برای افزایش سرعت سرور استفاده می شود.

بخش zonelistfile

در مقابل این فایل شما باید مسیر فایل zone.list را مشخص کنید، این فایل استفاده می شود برای zone هایی که به صورت dynamic اطلاعات خود را تغییر می دهند. این فایل با استفاده از فرمان add یا delet zone تغییر می کند.

بخش logfile

در این بخش شما باید مسیر فایلی که لاگ برنامه در آن نوشته می شود را مشخص کنید .

بخش server-count

در مقابل این بخش شما باید عددی را مشخص کنید که تعداد سرورهایی را که NSD می تواند را ه اندازی کند را مشخص می کند که تعداد پیش فرض آن 1 است.

بخش tcp-count

در مقابل این بخش شما بیشترین تعداد برقراری هم زمان ارتباط TCP با یک سرور را مشخص می کنید.

بخش pidfile

شما در این بخش نام فایلی را که pid برنامه NSD در آن قرار می گیرد را مشخص می کنید.

بخش port

شما در این بخش شماره پورتی را که NSD از طریق آن در شبکه سرویس می دهد را مشخص می کند که پیش فرض آن 53 است.

بخش chroot

در این بخش شما باید مسیر شاخه را مشخص کنید که برنامه NSD را به صورت chroot اجرا می کند را مشخص کنید.

بخش username

برای کاهش خطرات حاکی از هک شدن برنامه و محدود کردن سطح دسترسی شما برنامه NSD را باید با نام کاربری راه اندازی کنید که دسترسی ورود به سیستم را نداشته باشد و با ایجاد خطر عملا به سیستم شما آسیبی وارد نشود. هر برنامه در BSD باید از طریق یک نام کاربری اجرا شود تا هسته بتواند آنرا مدیریت کند.

بخش zonesdir

در مقابل این بخش شما باید مسیری که فایلهای zone در آن ذخیره شده است را مشخص کنید که مسیر پیش فرض آن /etc/nsdاست.

بخش hide-version

این بخش شامل yes و no است که پیش فرض ان no است.

بخش zonefiles-check

این بخش شامل دو گزینه yes و no است، این بخش باعث می شود که در صورتی که NSD راه اندازی مجدد شده است در زمان راه اندازی فایلهای zone را چک کند. پیش فرض این بخش yes تعریف شده است.

مدیریت NSD با nsd-control

برای مدیریت کردن سرور NSD برنامه ای وجود دارد به نام nsd-control که به صورت پیش فرض غیرفعال است و در زمان فعال شدن هم فقط به همان سیستمی که NSD برروی آن نصب است پاسخگو است. با این فرمان شما می توانید سرور را stop و start کنید، فایل پیکربندی را دوباره بارگذاری کنید، فایل zone اضافه و پاک کنید و از همه مهم تر گزارش کاملی از سرور خود رامشاهده کنید.

برای استفاده از فرمان شما باید در ابتد فرمان nsd-control-setup را راه اندازی کنید تا کلید و certification های مورد نیاز برای برقرار tls برروی TCPایجاد شود، بعد از آن باید در فایل nsd.conf بخش remote control را فعال کنید و بعد کلید های و certification های مورد نیاز را در بخش مشخص در nsd.conf کپی کنید در ادامه با مراحل راه اندازی این سرویس آشنا می شوید. در بخش اول فرمان nsd-control-setup را راه اندازی کنید

برای مشاهده کردن مراحل نصب و راه اندازی به دلیل استفاده از فایل gif به بخش مقالات سایت مراجعه کنید.

اضافه کردن Zone در NSD

بعد از انجام مراحل نصب و راه اندازی nsd dns server حال زمان اضافه کردن zone به این سرور شده است. برای این کار باید به فایل اصلی nsd.conf مراجعه کنید و بخشهایی را در آن فایل اضافه کنید، یکی از بخش های مهم داشتن یک فال zoneدرست است که در مقاله فایل zone چیست با آن آشنا می شوید.

در فایل nsd.conf بخشی برای اضافه کردن zone در نظر گرفته شده است که با نام :zone شروع می شود و شما باید بخش های مورد نظر را در آن اضافه کنید. هر zone باید با این بخش شروع شود و در بخش بعدی باید نام zone را مشخص کنید، نام را باید بعد از :nameاضافه کنید و در مرحله بعد هم باید مسیر فایل zone را مشخص کنید برای اینکار باید از بخش :zonefile استفاده کنید و مسیر را در مقابل آن اضافه کنید، در شکل زیر یک مثال ساده از اضافه کردن zpne را مشاهده می کنید:

```
Terminal
File  Edit  View  Terminal  Tabs  Help
^[ (escape) menu  ^y search prompt  ^k delete line   ^p prev li   ^g prev page
^o ascii code     ^x search         ^l undelete line ^n next li   ^v next page
^u end of file    ^a begin of line  ^w delete word   ^b back 1 char
^t top of text    ^e end of line    ^r restore word  ^f forward 1 char
^c command        ^d delete char    ^j undelete char ^z next word
=====line 309 col 0 lines from top 309 ========================================

        # RRLconfig
        # Response Rate Limiting, whitelist types
        # rrl-whitelist: nxdomain
        # rrl-whitelist: error
        # rrl-whitelist: referral
        # rrl-whitelist: any
        # rrl-whitelist: rrsig
        # rrl-whitelist: wildcard
        # rrl-whitelist: nodata
        # rrl-whitelist: dnskey
        # rrl-whitelist: positive
        # rrl-whitelist: all
        # RRLend
zone:
        name: abedini.com
        zonefile: /usr/local/etc/nsd/zone/abedini.com
```

اضافه کردن فایل Zone

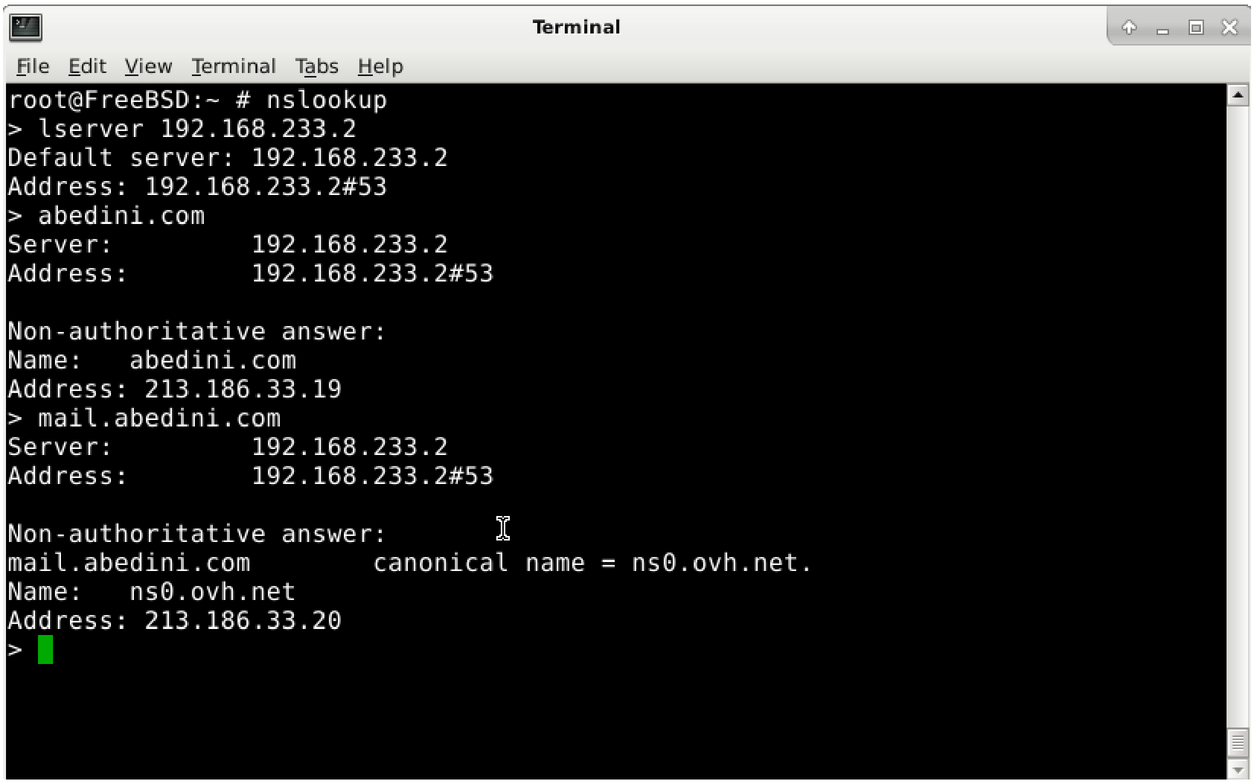
حال با استفاده از فرمان nsd-control فایل nsd.conf را دوباره بارگذاری کنید و یا سرور را restart کنید به صورت زیر:

```
Terminal
File  Edit  View  Terminal  Tabs  Help
root@FreeBSD:/usr/local/etc/nsd/zone # nsd-control reload
ok
root@FreeBSD:/usr/local/etc/nsd/zone #
```

بارگذاری مجدد تنظیمات NSD

در مقاله مدیریت NSD با nsd-control روش ایجاد و استفاده کردن از این فرمان کامل شرح داده شده است.

در مرحله بعدی با استفاده از فرمان nsllokup از صحت کارایی سرور مطمین شوید به شرح زیر:



چک کردن تنظیمات

با استفاده از فرمان dig هم می توانید به صورت زیر اطلاعات را مشاهده کنید:

```
root@FreeBSD:~ # dig abedini.com any @192.168.233.2 | less

; <<>> DiG 9.10.3-P4 <<>> abedini.com any @192.168.233.2
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 63864
;; flags: qr rd ra; QUERY: 1, ANSWER: 7, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; MBZ: 0005 , udp: 512
;; QUESTION SECTION:
;abedini.com.                   IN      ANY

;; ANSWER SECTION:
abedini.com.            5       IN      SOA     dns.ovh.net. tech.ovh.net. 20160
90702 86400 3600 3600000 86400
abedini.com.            5       IN      NS      ns.ovh.net.
abedini.com.            5       IN      NS      dns.ovh.net.
abedini.com.            5       IN      MX      1 mx0.ovh.net.
abedini.com.            5       IN      MX      100 mxb.ovh.net.
abedini.com.            5       IN      A       213.186.33.19
abedini.com.            5       IN      TXT     "v=spf1 include:mx.ovh.com ~all"

;; Query time: 550 msec
```

استفاده کردن از فرمان dig

سرور NSD در دو حالت master و secondary

در بسیاری از روشها استفاده از سرور DNS به دلیل اهمت زیاد در همیشه در دسترس بودن این سرویس برای شبکه DNS سرورها در حالت master و secondary دارند که در صورتی که سرور اصل نتواند پاسخگوی درخواست ها باشد سرور دیگری با همان zone ها بتواند به سرویس دهی بپدازد. در zone های که پویا هستند اطلاعات zone در حال تغییر می باشد این امر بسیار حیاتی است که دو یا چند سرور در شبکه شما وجود داشته باشد. در مورد zone های که پویایی زیادی ندارند شما می توانید دو سرور جداگانه ایجاد کنید، DNSسرور NSD هم می تواند هر دو نقش master و secondary را ایفا کند، برای اینکار باید در زمان تعریف zone این نوع را مشخص کنیددر مثال زیر در بخش اول سرور master را تعریف کرده و در بخش دوم هم باید در سرور secondary دامین خود را به صورت زیر تعریف کنید:

سرور Master:

```
zone:
        # this server is master, 10.0.2.1 is the secondary.
        name: masterzone.com
        zonefile: /etc/nsd/masterzone.com.zone
        notify: 10.0.2.1 NOKEY
        provide-xfr: 10.0.2.1 NOKEY
```

سرور Secondary:

```
zone:
        # this server is secondary, 10.0.2.2 is master.
        name: secondzone.com
        zonefile: /etc/nsd/secondzone.com.zone
        allow-notify: 10.0.2.2 NOKEY
        request-xfr: 10.0.2.2 NOKEY
```

راه اندازی DHCP سرور در FreeBSD

سرور DHCP یا همان dynamic Host Configurations protocol به کابران متصل شده به شبکه شما این اجازه می دهد که به صورت خودکار تنظیمات خاص شبکه شما را دریافت و از منابع موجود در شبکه استفاده کند. این سرور در دو بخش Client و server ارایه می شود، در بخش client در سیستم عامل FreeBSD از برنامه dhclient موجود در OpenBSD استفاده می کند و برای دریافت اطلاعات از سرور DHCP از این برنامه استفاده می کند.

در بخش سروری در FreeBSD این برنامه به صورت پیش فرض نصب نشده است و تعداد زیادی برنامه سروری با قابلیت های خاصی در بخش سیستم ports وجود داردکه هر کدام قابل نصب و راه اندازی است. در ادامه یکی از برنامه های سروری را با هم نصب و پیکربندی می کنیم.

نصب و پیکربندی DHCPسرور

در این بخش برای نصب و پیکربندی DHCP از سروری که توسط Internet System Consortium ایجاد و ارایه شده است استفاده می کنید که در بخش سیستم ports از زیر شاخه net/isc-dhcp43-server برای نصب و راه اندازی استفاده می کنیم. این برنامه را هم شما می توانید هم از طریق سیستم pkg هم نصب کنید

بعد از انجام دادن مراحل نصب به صورت خودکار فایل پیکربندی پیش فرضی به نام dhcp.conf.example در زیر شاخه /usr/local/etcایجاد شده است که با استفاده از فرمان cp به صورت زیر نام این فایل را به dhcp.conf تغییر دهید:

```
#cp /usr/local/etc/dhcp.conf.example /usr/local/etc/dhcp.conf
```

فایل پیکربندیdhcp.conf

در این فایل دو بخش برای اعمال تنظیمات وجود دارد، در بخش اول تنظیمات دامین سیستم، DNS که کاربر اطلاعات نام سایت ها را از ان پرس و جو کند و مدت زمان در اختیار داشتن آدرس و sub net mask کلاینت ها را در آن تنظیم می کنید این بخشها را در زیر مشاهده می کنید:

```
option domain-name "mabedini.ir";
option domain-name-servers 192.168.1.100;
option subnet-mask 255.255.255.0;

default-lease-time 600;
max-lease-time 72400;
```

نکته:

تمام خطوط در فایل dhcp.conf به علامت ; ختم می شود، به استفاده از این علامت توجه کنید.

بخش بعدی شما باید اطلاعات آدرس ip که قصد دارید در اختیار کاربران شبکه خود قرار دهید را مشخص کنید در این بخش شما با استفاده از این فایل به صورت زیر این عمل را انجام دهید:

```
subnet 192.168.1.0 netmask 255.255.255.0 {
  range 192.168.1.50 192.168.1.150;
  option routers 192.168.1.1, 192.168.1.2;
}
```

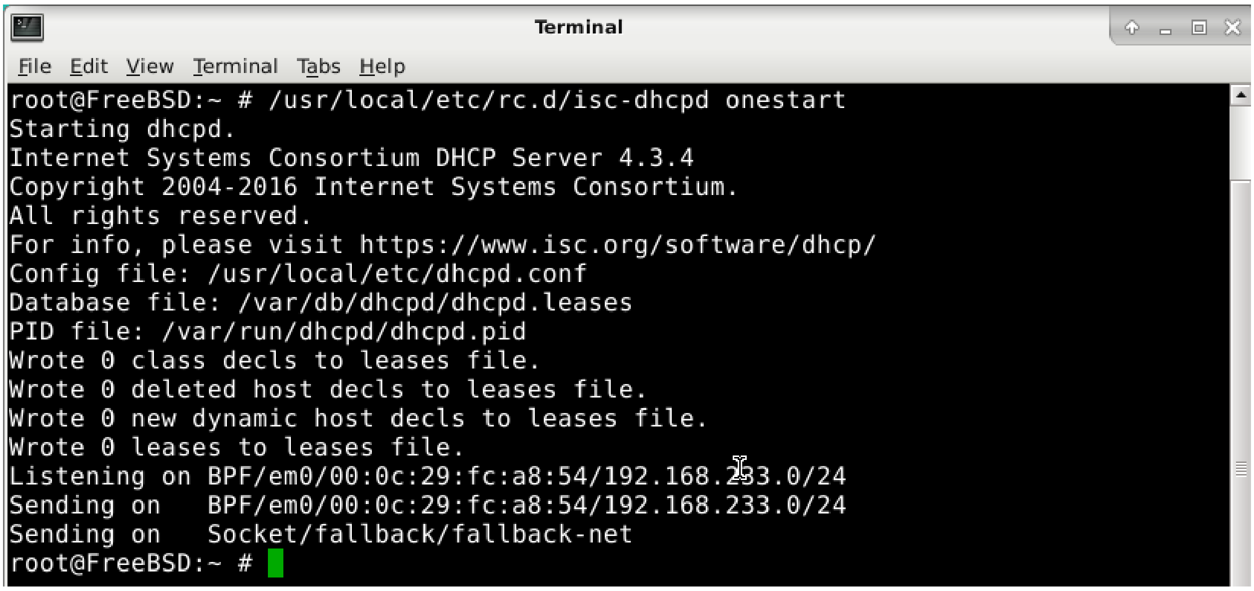
همانطوری که مشاهده می کنید در ابتدا بخش شبکه ای که قصد دارید از ان استفاده کنید را مشخص کنید و در بخش دوم subnetmaskهر کاربر را مشخص کنید. در ادامه باید یک رنج از آدرس که قصد اجاره دادن آنرا دارید را مشخص کنید. در ادامه default gateway کلاینت ها را مشخص کنید. در این بخش ما می توانید چند default route مشخص کنید و برای جدا کردن آنها از علامت , استفاده کنید.

راه اندازی سرویس dhcp

از آنجایی که سرور DHCP را از طریق نصب بسته به سیستم عامل FreeBSD اضافه کرده اید این برنامه فایل های پیکربندی و راه اندازی را در شاخه /usr/local ذخیره می کند و برای راه اندازی آن هم باید از شاخه dc.d واقع در آدرس /usr/local/etcاستفاده کنید، برای راه اندازی به صورت خودکار هم باید از فایل /etc/rc.conf استفاده کنید. بعد اضافه کردن خط زیر به فایل rc.conf حال شما می توانید از فرمانهای rc.d استفاده کنید به صورت زیر:

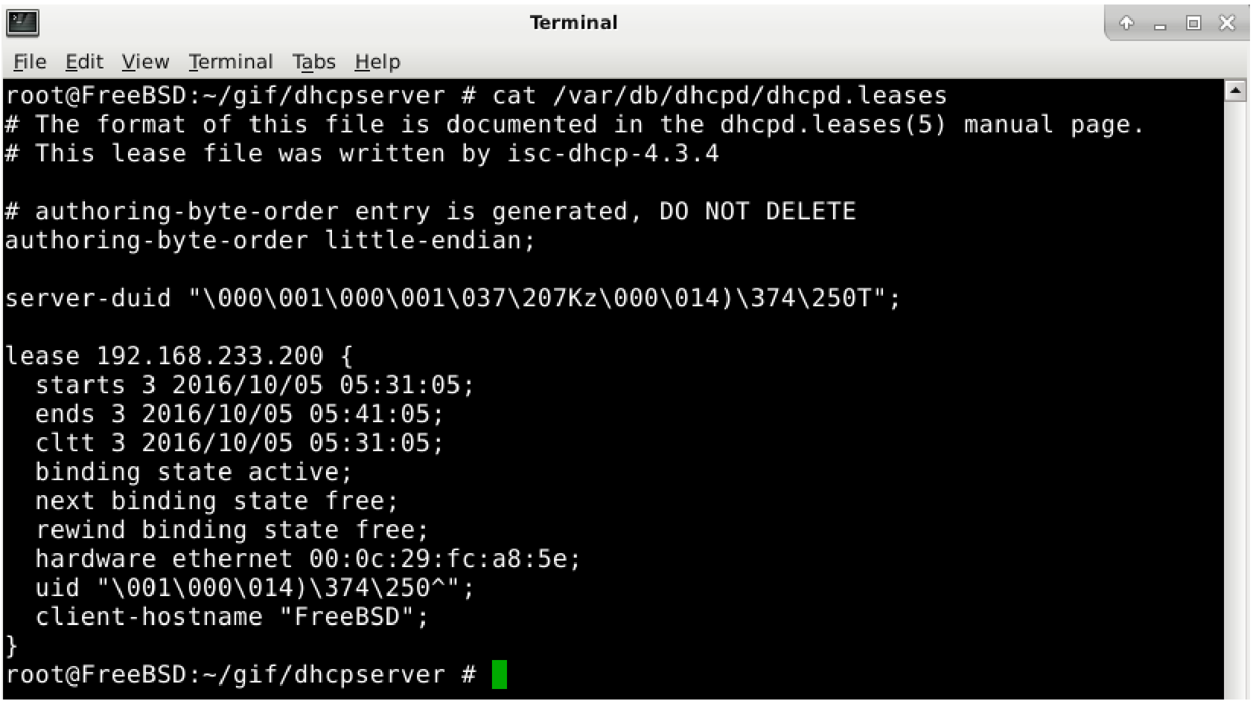
```
dhcpd_enable="YES"
dhcpd_ifaces="em0”
```

اگر هم قصد راه اندازی یکبار این سیستم را دارید از سیستم onestart استفاده کنید به صورت زیر:


```
Terminal
File Edit View Terminal Tabs Help
root@FreeBSD:~ # /usr/local/etc/rc.d/isc-dhcpd onestart
Starting dhcpd.
Internet Systems Consortium DHCP Server 4.3.4
Copyright 2004-2016 Internet Systems Consortium.
All rights reserved.
For info, please visit https://www.isc.org/software/dhcp/
Config file: /usr/local/etc/dhcpd.conf
Database file: /var/db/dhcpd/dhcpd.leases
PID file: /var/run/dhcpd/dhcpd.pid
Wrote 0 class decls to leases file.
Wrote 0 deleted host decls to leases file.
Wrote 0 new dynamic host decls to leases file.
Wrote 0 leases to leases file.
Listening on BPF/em0/00:0c:29:fc:a8:54/192.168.233.0/24
Sending on   BPF/em0/00:0c:29:fc:a8:54/192.168.233.0/24
Sending on   Socket/fallback/fallback-net
root@FreeBSD:~ #
```

نمایش حالت راه اندازی OneStart

بعد از اجرا کردن این فرمان و دریافت آدرسهای IP از سمت سرور فایلی در زیر شاخه /var/db/dhcpd وجود دارد به نام dhcp.leases که اطلاعات آدرس های ارایه شده به سیستم ها را نمایش می دهد محتوای این فایل را در شکل زیر مشاهده می کنید:



```
root@FreeBSD:~/gif/dhcpserver # cat /var/db/dhcpd/dhcpd.leases
# The format of this file is documented in the dhcpd.leases(5) manual page.
# This lease file was written by isc-dhcp-4.3.4

# authoring-byte-order entry is generated, DO NOT DELETE
authoring-byte-order little-endian;

server-duid "\000\001\000\001\037\207Kz\000\014)\374\250T";

lease 192.168.233.200 {
  starts 3 2016/10/05 05:31:05;
  ends 3 2016/10/05 05:41:05;
  cltt 3 2016/10/05 05:31:05;
  binding state active;
  next binding state free;
  rewind binding state free;
  hardware ethernet 00:0c:29:fc:a8:5e;
  uid "\001\000\014)\374\250^";
  client-hostname "FreeBSD";
}
root@FreeBSD:~/gif/dhcpserver #
```

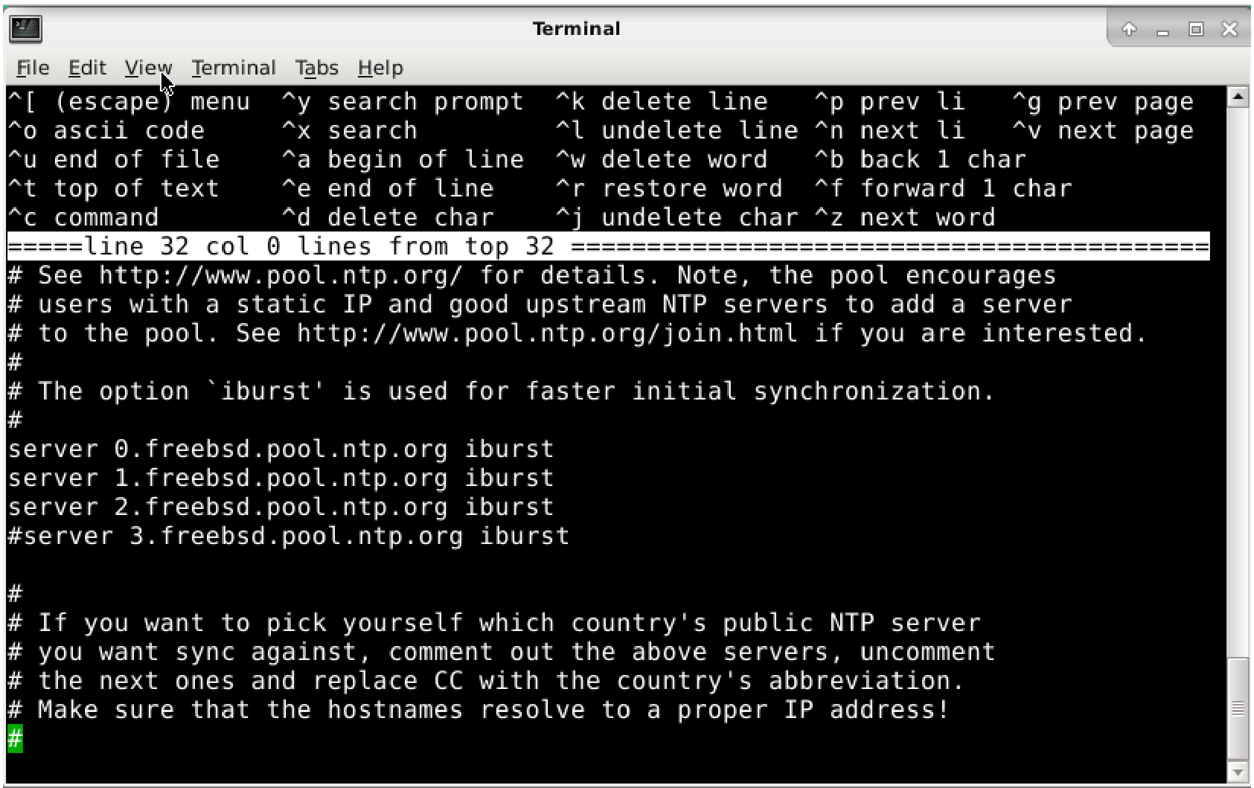
نمایش وضعیت آدرسهای اختصاص داده شده

راه اندازی time سرور در FreeBSD

شاید برای اولین بار باشد که با مفهوم time serverآشنا شده باشید اگر از سیستم عامل ویندوزی به سمت FreeBSD مهاجرت کرده باشید چنین سروری را در سیستم خود ندارد ولی در FreeBSD یکی از مهترین سرور های شبکه است، اهمیت زمان در شبکه از آنجا مشخص می شود که شما در شبکه خود در بین چندین سرور قصد دارد که واقعه ای که در زمان خاصی اتفاق افتاده باشد را پیگیری کنید، حال اگر زمان سرورهای شما یکی نباشد شما به چه صورتی می توانید آنها چک کنید؟ در این بین سرور را نیاز دارید که زمان را برای شما از سرور های اصلی در انترنت پرس و جو کرده و در اختیار سرور ها و سیستم های موجود در شبکه قرار بدهند. در FreeBSD شما از سرویس ntime می توانید برای انجام دادن این عمل استفاده کنید. این پروتکل شبکه هم دو سویه است هم قابلیت سروری دارد هم قابلیت کلاینتی، در ادامه اول با بخش کلاینتی اشنا می شود و در بخش دوم هم قابلیت سروری این سرویس را مورد بررسی قرار می دهیم.

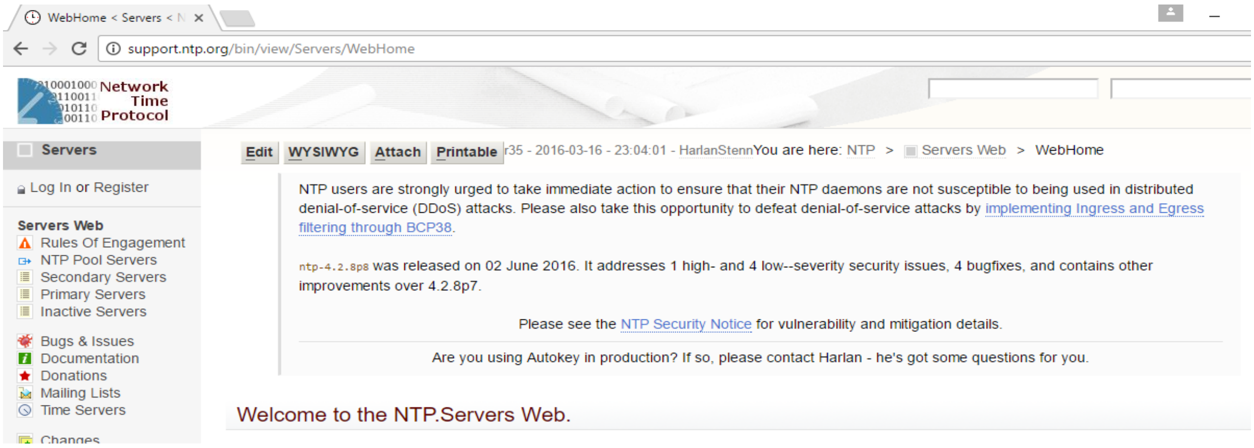
بروز رسانی کردن زمان سیستم

به صورت پیش فرض برنامه ای به نام ntpd برای تنظیم کردن ساعت سیستم شما با time سرور های موجود، برای فعال کردن این سرویس در زمان راه اندازی در فایل /etc/rc.conf باید خط ntpd_enable="YES" را وارد کنید. این برنامه به صورت خودکار فایل /etc/ntp.conf را برای پیدا کردن سرور ها جستجو می کند. در این فایل بخشهای مختلفی وجود دارد که برای جستجو از سرورهای time در این فایل باید بخش server را ویرایش و مشاهده کنید در شکل زیر این بخش برای شما نمایش داده شده است:



نمایش فایل ntp.conf

اگر در بخش مقابل نام سرور عبارت prefer را اضافه کنید این سرور به عنوان اولین سرور مورد بررسی قرار می گیرد. سعی کنید در این بخش سروری را انتخاب کنید تا قدرت بالایی را داشته باشد. برای دریافت لیستی از سرورهای time موجود از سایت http://www.ntp.org استفاده کنید و به بخش Public Time Server Lists وارد شوید تا صفحه ای به صورت زیر برای شما باز شود:



سایت ntp.org

حال به تناسب منطقه خود لیست سرور های خود را پیدا کنید برای مثال برای آسیا لیستی به صورت زیر مشاهده می کنید:

Asia — asia.pool.ntp.org

To use this pool zone, add the following to your ntp.conf file:

```
server 0.asia.pool.ntp.org
server 1.asia.pool.ntp.org
server 2.asia.pool.ntp.org
server 3.asia.pool.ntp.org
```

لیست سرورهای زمان بخش آسیا

برنامه ntpdate

برای تنظیم کردن سریع زمان سیستم خود از فرمان ntpdate استفاده کنید این فرمان در دو حالت استفاده می شود. در مرحله اول اگر در مقابل این فرمان سروری را مشاهده نکرده باشید از فایل ntp.conf برای انتخاب کردن سرور استفاده می کند و در بخش دوم شما می توانید هر سروری را که دوست دارید در مقابل این فرمان وارد کنید تا از طریق آن زمان را جستجو کند به صورت زیر:

```
Terminal
File  Edit  View  Terminal  Tabs  Help
root@FreeBSD:~ # date
Thu Oct 13 17:21:09 IRST 2016
root@FreeBSD:~ # ntpdate 0.asia.pool.ntp.org
13 Oct 17:36:28 ntpdate[943]: step time server 203.158.118.2 offset 890.546146 s
ec
root@FreeBSD:~ # date
Thu Oct 13 17:36:31 IRST 2016
root@FreeBSD:~ #
```

تنظیم زمان درست سیستم با استفاده از فرمان ntpdate

همانطوری که مشاهده می کنید در اجرای فرمان date اول زمان سیستم 17:21 دقیقه است که بعد از راه اندازی ntpdate و سرور آسیا به نام 0 aisa.pool.ntp.org.زمان سیستم در فرمان دوم date به زمان مناسب و درست سیستم تغییر کرده است.

راه اندازی سرور ntpبا OpenNTPd

برای اینکه بخش سروری باید از برنامه ای به نام OpenNTPd که در مجموعه پروژه های OpenBSD ایجاد شده است نصب و راه اندازی کنید، این برنامه در همه ساختارها به ویژه FreeBSD قابل نصب و اجراست برای نصب باید از طریق سیستم portsدر FreeBSD اقدام کنید و در مرحله اول وارد شاخه /usr/ports/net/openntpd وارد شوید و با دو فرمان makeو make install این برنامه را نصب کنید در انتها پیغام هایی به صورت زیر مشاهده می کنید:

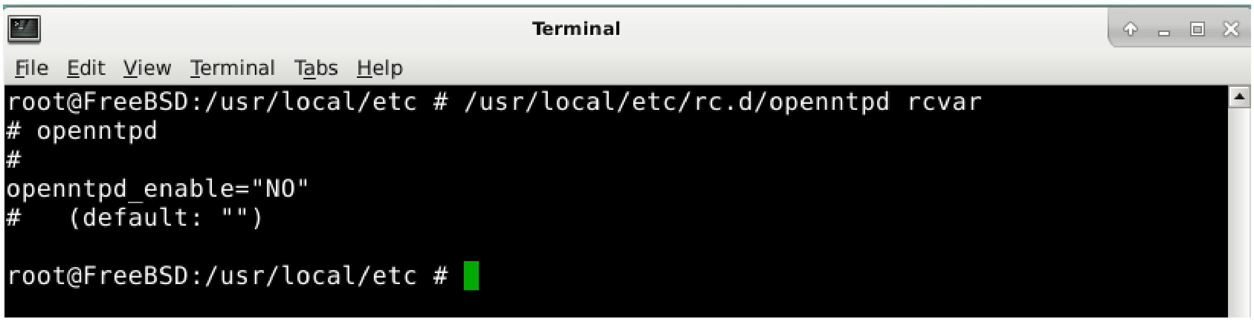
```
Terminal
File  Edit  View  Terminal  Tabs  Help
===>   Registering installation for openntpd-5.7p4_1,2
Installing openntpd-5.7p4_1,2...
===> Creating users and/or groups.
Creating group '_ntp' with gid '123'.
Creating user '_ntp' with uid '123'.
===> SECURITY REPORT:
      This port has installed the following files which may act as network
      servers and may therefore pose a remote security risk to the system.
/usr/local/sbin/ntpctl
/usr/local/sbin/ntpd

      This port has installed the following startup scripts which may cause
      these network services to be started at boot time.
/usr/local/etc/rc.d/openntpd

      If there are vulnerabilities in these programs there may be a security
      risk to the system. FreeBSD makes no guarantee about the security of
      ports included in the Ports Collection. Please type 'make deinstall'
      to deinstall the port if this is a concern.

      For more information, and contact details about the security
      status of this software, see the following webpage:
http://www.openntpd.org/
root@FreeBSD:/usr/ports/net/openntpd #
```

نمایش بخشهای نصب OpenNTPD در FreeBSD

این برنامه در زیر شاخه /usr/local نصب شده و فایل ntpd.conf در زیر شاخه /usr/local/etc قرار می گیرد و برای راه اندازی از طریق سیستم rc.d از فرمان /usr/local/etc/rc.d/openntpd استفاده کنید در فرمان هم به سیستم به نام های ntpctl و ntpd به سیستم اضافه می شود که شما می توانید از طریق آن openntpd را مدیریت کنید در ادامه با روش استفاده از openntpd آشنا می شوید.

راه اندازی از طریق rc.conf

برای راه اندازی openntpd از طریق سیستم rc.d شما باید خط زیر را در فایل rc.conf اضافه کنید تا بتوانید با استفاده از فرمان ای rc.d این سرویس را مدیریت کنید، این بخش را در شکل زیر مشاهده می کنید:



تنظیمات راه اندازی سرور NTP در فایل rc.conf

فایل ntpd.conf

فایل پیکربندی این سرور در زیر شاخه /usr/local/etc به نام ntpd.conf قرار دارد. در این فایل خطوطی که با # شروع می شود توضیحات است و این فایل به صورت متغییر و مقدار است.در بخش اول در این فایل باید آدرس ip که سرور باید برروی آن به درخواستها پاسخ دهد را مشخص کنید، این بخش با کلید listen on شروع می شود و در خط زیر شما نمونه از این پیکربندی را مشاهده می کنید:

```
Listen on *
Or
Listen on 127.0.0.1
```

در بخش اول یعنی برروی هر آدرس ip به ارایه سرویس بپرداز و در بخش دوم فقط برروی کارت شبکه localhost به درخواست ها پاسخ می دهد این بدان معناست که فقط به درخواست هایی که از سمت همین سیستم عامل ارسال می شود پاسخ دهد.

در مرحله بعد شما باید سرور هایی که از آنها زمان را سوال کند را مشخص کنید در این بخش شما هم می توانید از نام آن سرور و هم آدرس IP آن استفاده کنید، این بخش با server شروع می شود به صورت زیر:

```
Server 10.0.2.2
Or
Server pool.ntp.org
```

حال سرور خود را راه اندازی کنید به صورت زیر:

```
Terminal
File  Edit  View  Terminal  Tabs  Help
root@FreeBSD:~ # /usr/local/etc/rc.d/openntpd start
Starting openntpd.
root@FreeBSD:~ # sockstat -l4
USER     COMMAND     PID    FD PROTO  LOCAL ADDRESS          FOREIGN ADDRESS
_ntp     ntpd        4193   7  udp4   192.168.85.128:123     *:*
_ntp     ntpd        4193   10 udp4   127.0.0.1:123          *:*
root     sendmail    799    3  tcp4   127.0.0.1:25           *:*
root     sshd        796    4  tcp4   *:22                   *:*
root     syslogd     576    7  udp4   *:514                  *:*
root@FreeBSD:~ #
```

راه اندازی کردن سرور و نمایش وضعیت پورتهای باز

برای نمایش اطلاعات بیشتر و گزارش بهتر است که از flags های s- و v- در بخش فایل rc.conf به صورت زیر اضافه کنید:

```
# openntpd
#
openntpd_enable="YES"
openntpd_flags="-s -v"
#   (default: "")

root@FreeBSD:/var/run #
```

تنظیمات بیشتر در فایل rc.conf

برای نمایش وضعیت سرور باید از فرمان ntpctl استفاده کنید در شکل زیر شما peers های که سرور شما از آنها زمان را دریافت می کنند مشاهده می کنید به صورت زیر:

```
Terminal
File  Edit  View  Terminal  Tabs  Help
root@FreeBSD:/var/run # ntpctl -s peers
peer
   wt tl st  next  poll          offset       delay      jitter
129.250.35.251 0.asia.pool.ntp.org
    1 10  2    9s   33s       503.772ms   668.605ms   662.320ms
5.160.24.41 from pool pool.ntp.org
    1  2  3   35s  300s           ---- peer not valid ----
46.209.15.97 from pool pool.ntp.org
    1 10  3   18s   30s       590.247ms   674.154ms   753.122ms
46.209.14.1 from pool pool.ntp.org
    1 10  3    2s   30s       526.728ms   609.907ms   681.053ms
194.225.50.25 from pool pool.ntp.org
    1 10  2   22s   32s       591.769ms   675.372ms   756.601ms
root@FreeBSD:/var/run #
```

نمایش peers در خروجی فرمان ntpctl

حال در بخش پایانی برای دریافت اطلاعات از طریق سرور خود با استفاده از فرمان ntpdate به صورت زیر فرمان را با استفاده از آدرس ipسرور خود اجرا کنید:

```
root@FreeBSD:/var/run # ntpdate -dv 192.168.85.128
14 Oct 00:43:19 ntpdate[4425]: ntpdate 4.2.8p8-a (1)
transmit(192.168.85.128)
receive(192.168.85.128)
transmit(192.168.85.128)
receive(192.168.85.128)
transmit(192.168.85.128)
receive(192.168.85.128)
transmit(192.168.85.128)
receive(192.168.85.128)
192.168.85.128: Server dropped: Leap not in sync
server 192.168.85.128, port 123
stratum 3, precision -23, leap 11, trust 000
refid [192.168.85.128], delay 0.02567, dispersion 0.00136
transmitted 4, in filter 4
reference time:    dbaa770a.8fc9cfff  Fri, Oct 14 2016  0:41:38.561
originate timestamp: dbaa7776.58a937ff  Fri, Oct 14 2016  0:43:26.346
transmit timestamp:  dbaa7775.64502f4f  Fri, Oct 14 2016  0:43:25.391
filter delay:  0.02567  0.02579  0.02583  0.02583
         0.00000  0.00000  0.00000  0.00000
filter offset: 0.957379 0.956383 0.955363 0.954366
         0.000000 0.000000 0.000000 0.000000
delay 0.02567, dispersion 0.00136
offset 0.957379
```

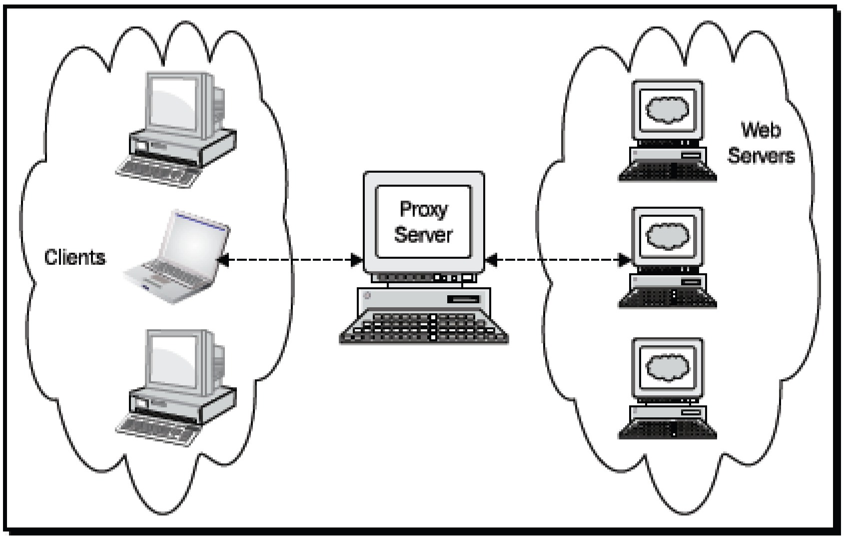
نمایش مراحل تنظیم کردن زمان

Squidچیست؟

Proxyچیست؟

سرور Proxy سیستمی است که در بین درخواست وب یک کلاینت و سرور مورد نظر وب مورد درخواست قرار می گیرد و به ذخیره کردن مطالب و درخواستها می پردازد. سرور های Proxy بدون تغییر دادن درخواست ها بین کلاینت و سرور عمل می کنند و در حقیقت در میان این ارتباط قرار می گیرند و خود ار به جای کلاینت برای سرور و سرور برای کلاینت معرفی می کنند. نوع از proxy سرور به نام transparent وجود دارد که کلاینت از ظهور آن در شبکه اطلاعی ندارد و تصور می کند که خود اقدام به برقرار ارتباط می پردازد.

در حالت پیشرفته از proxy سرور ، این سرور درخواست های ارسال شده را بر اساس یک سری قواعد فیلتر کرده و به درخواستهایی که با قواعد آن مطابقت دارد اجازه خارج شدن از شبکه را می دهد. این قواعد می تواند بر اساس آدرس IP پروتکلهای شبکه و نوع مطالب درخواست شده از طرف کلاینت باشد.



نمایش مفهوم پراکسی سرور

همانطوری که در شکل بالا مشاهده می کنید این سرور در بین Client و Web Servers قرار می گیرد و کلاینتها نمی توانند به صورت مستقیم با سرورهای وب ارتباط برقرار کنند. در برخی از انوع proxy ها این سرور در درخواستهای ارسال شده هم دخل و تصرف می کند و گزارشات ارسال شده از سمت کلاینتها را هم ذخیره می کند.

در اصل این سرور ها برای مدیریت پهنای باند، افزایش سرعت بازکردن صفحه های وب درخواستی از سمت کلاینت، مدیریت دسترسی، مشاهده کردن گزارش ترافیک های مصرف شده از سمت کلاینتها و مدیریت ترافیک ارسال شده سمت وب سرورهای مختلف، استفاده از چند ارتباط اینترنت به صورت همزمان است.

یکی از proxy server ها معروف و قدیمی Squid است که در ادامه با آن آشنا می شوید.

دریافت Squid

سرور Squid اولین بار در سال 1996 ارایه شد و تا این لحظه با ورژن ها و قابلیت های مختلفی به صورت رایگان و Open Sourceدر سابت http://www.squid-cache.org برای دانلود کردن در دسترس است.این برنامه در قالبهای مختلف قابل دانلود است هم در قالب کد برنامه و هم در قالب برنامه های باینری و از پیش نصب شده برای همه سیستم عاملها مثل FreeBSDو Linux شما به راحتی می توایند در FreeBSD هم از بسته های باینری برای نصب استفاده کنید. ولی خود پروژه به شما پیشنهاد می کند که از کد مبدا Squid را نصب کنید.

قبل از شروع به نصب باید در مورد ورژنهایی که برای Squid وجود دارد توضیحاتی داده شود، برای دریافت آخرین اطلاعات در مورد ورژن های موجود از لینک . http://www.squid-cache.org/Versions/ این بخش در شکل زیر نمایش داده شده است:

Source Code Packages

These allow you to customize your Squid installation when you compile it.

After downloading,refer to Compiling Squid for assistance with compiling the source code.

Stable Versions:

Current versions suitable for production use.

Version	Latest Release	
3.5	09 Oct 2016	3.5.22
langpack	17 Aug 2016	ROLLING

Beta Versions:

Meant for testing the next release before it is ready for production use.

Suitable for making upgrade plans and similar activities.

Version	Latest Release	
4.0	09 Oct 2016	4.0.15

Development Versions:

Meant for Squid users who are already familiar with Squid. You should expect to find numerous bugs and problems. We do not recommend running a development release on your production cache. If you have any problems with a development release please write to our *squid-bugs@squid-cache.org* or *squid-dev@squid-cache.org* lists. DO NOT write to squid-users with code-related problems.

Version	Development Start Date	Development Plans
4	09 Oct 2014	Squid RoadMap

If you are a developer, or want to closely track the source code, repository and rolling release services are outlined in the Developer Resources.

Old Versions:

Provided for archival purposes only. Not intended for general use in new installations.

Version	First STABLE release Date	Latest Release	Latest Release Date
3.4	09 Dec 2013	3.4.14	01 Aug 2015
3.3	09 Feb 2013	3.3.14	01 May 2015
3.2	14 Aug 2012	3.2.14	01 May 2015

نمایش ورژنهای موجود squid

چهار ورژن وجود داردStable ، Beta،   Development Versionو .old Versions برای نصب در محیط های حساس از ورژن Satable استفاده کنید و در بعضی از موارد هم ورژنهای قدیمی توصیه می شود. بعد از انتخاب کردن ورژن مورد نظر و کلیک کردن برروی آن وارد صفحه ای می شوید که می توانید ورژن که نیاز دارید دانلود کنید به صورت زیر:

Squid version 3.4

Release	Date	diff	Download
Latest 3.4 series release			
squid-3.4.14	01 Aug 2015	diff (sig)	tar.gz (sig) / tar.bz2 (sig) / tar.xz (sig)
See langpack for latest Language Package			
Daily auto-generated release. This is the most recent bug-fixed update to the formal release. see Change details for the fixes included in this bundle.			
squid-3.4.14-20160509-r13240	10 May 2016		tar.gz / tar.bz2
squid-3.4.14-20160508-r13239	09 May 2016		tar.gz / tar.bz2
squid-3.4.14-20160506-r13238	07 May 2016		tar.gz / tar.bz2
squid-3.4.14-20160502-r13236	03 May 2016		tar.gz / tar.bz2
squid-3.4.14-20160420-r13235	21 Apr 2016		tar.gz / tar.bz2
squid-3.4.14-20160331-r13232	01 Apr 2016		tar.gz / tar.bz2
squid-3.4.14-20160221-r13231	22 Feb 2016		tar.gz / tar.bz2
squid-3.4.14-20160212-r13230	13 Feb 2016		tar.gz / tar.bz2
squid-3.4.14-20151210-r13229	11 Dec 2015		tar.gz / tar.bz2
squid-3.4.14-20150901-r13228	02 Sep 2015		tar.gz / tar.bz2
squid-3.4.14-20150801-r13227	02 Aug 2015		tar.gz / tar.bz2
Squid 3.4 BZR	Source rsync		Launchpad Mirror

اخرین ورژنهای موجود از برنامه Squid

حال ورژن مورد نظر خود را دانلود کنید تا وارد مرحله نصب شوید. در بخش بعدی با نصب squid ادامه می دهیم.

نصب Squid

برای نصب از طریق کد مبدا شما نیاز به کامپایلر دارید که Squid به زبان C/C++ نوشته شده است در نتیجه شما نیاز به این کامپایلر دارید، سیستم عامل FreeBSD شامل کامپایلر C/C++ است در نتیجه شما برای نصب مشکلی نخواهید داشت.

در قدم اول شما باید بعد از دانلود برنامه آنرا در شاخه ای که فضای لازم را دارید کپی کنید و با استفاده از برنامه tar آنرا از حالت فشرده خارج کنید به صورت زیر این فرمان را راه اندازی کنید:

```
#tar -xvzf squid-3.1.10.tar.gz
```

بعد وارد شاخه squid-3.1.10 شده و در این شاخه شما برای نصب باید از سه فرمان make /configure و make installرا راه اندازی کنید. قبل از اینکه فرمام Configure را اجرا کنید باید در مورد این فرمان توضیحاتی داده شود، این فرمان تنظیمات مورد نیاز شما را از طریق سویچهایی که به آن می دهید اعمال می کند. برای نمایش همه سویچ های موجود از فرمان زیر استفاده کنید که در شاخه کد برنامه اجرا شود:

```
# ./configure --help | less
```

چند سویچ مهم و پر استفاده برای شما در ادامه شرح داده خواهد شد. به این نکته توجه کنید که هر سویچ با -- شروع می شود.

سویچ prefix :

با این سویچ شما می توانید مسیر نصب را مشخص کنید، در بسیاری از موارد شاید شما قصد داشته باشید که چند ورژن از Squidرا هم زمان برروی سیستم خود نصب و یکی را انتخاب کنید برای این کار بهتر است که مسیر های نصب پیش فرض را تغییر دهید و مسیر های جدید ایجاد و انتخاب کنید، این سویچ برای این کار بسیار مفید است. در ادامه یک مثال از فرمان ./configureرا مشاهده می کنید:

```
./configure --prefix=/opt/squid/3.1.10/
```

فرمان Configure علاوه بر توانایی در انتخاب مسیر نصب شما می توانید سایر مسیرها مثل فایلهای اجرا و غیره را با استفاده از سویچ های—bindir, --sbindir-- تغییر دهید.

نکته:

سایر بخشهایی که شما در ادامه فرمان Configure با انها سرو کار دارید برای فعال کردن و غیرفعال کردن یک قابلیت در Squidاست که کافیست برای اینکار از دو سویچ enable-FEATUR برای فعال کردن قابلیت و-disable FEATUREبرای غیرفعال کردن یک قابلیت است که در مثال زیر آنرا مشاهده می کنید:

```
./configure --enable-FEATURE # FEATURE will be enabled
./configure --disable-FEATURE # FEATURE will be disabled
```

سویچ enable-gnuregex:

با این سویچ قابلیت Regular expression را در Squid فعال می کنید که از آن برای ایجاد کردن لیست کنترل کننده استفاده می کند. اگر از سیستم عاملهای بروز شده یونیکیسی استفاده می کنید نیازی به فعال کردن این بخش نیست چون در سیستم عامل شما این قابلیت فعال شده است.

سویچ enable-storeio:

در زمانی که از قابلیت Cache در Squid استفاده می کنید عملکرد آن به سرعت خواندن و نوشتن برروی دیسک بسیار وابسته است و شما می توانید در این بخش تکنولوژی های استفاده شده در سیستم عامل خود را معرفی کنید به صورت زیر:

```
./configure --enable-storeio=ufs,aufs,coss,diskd,null
```

سویچ :enable-removal-policies

برای حذف کردن و آزاد کردن فضای Cache در سرور squid می توانید از سیاست های ایجاد شده در آن استفاده کنید با استفاده از این بخش سرور Squid در مورد مطالب ذخیره شده برای حذف آنها تصمیم می گرید در زیر دو سیاست در سرور squidرا با استفاده از این فرمان می توانید راه اندازی کنید:

```
./configure --enable-removal-policies=heap,lru
```

سویچ :enable-referer-log

با استفاده از این سویچ شما می توانید تمام هدر های درخواست های ارسال شده از سمت کاربران را ثبت کنید.

سویچ :disable-wccp

در Squid قابلیتی وجود دارد به نام Cisco's Web Cache Communication Protoco که به صورت پیش فرض فعال است و در صورتی که نیاز به آن دارید آنرا فعال کنید و با استفاده از این سویچ می توانید آنرا غیرفعال کنید.

سویچ : disable-snmp

در Squid شما می توانید با استفاده از پروتکل SNMP وضعیت سرور خود را مانیتور کنید که به صورت پیش فرض این بخش فعال بود و بعد از اضافه کردن ان در فایل پیکربندی Squid می توانید از آن استفاده کنید اکر هم به این پروتکل نیازی ندارید می توانید با این سویچ آنرا غیرفعال کنید. البته این قابلیت از ورژن 3 به بعد به صورت پیش فرض فعال شده است.

سویچ :enable-cachemgr-hostname

در Squid بخش به نان cachemgr.cgi وجود دارد که شما می توانید با استفاده از یک رابط صفحه وب به مدیریت squid بپردازید. شما در این بخش یک نام برای برقرار ارتباط با سرور می توانید به جای آدرس IP و یا localhost استفاده کنید. در زیر روش تنظیم کردن آنرا مشاهده میکنید:

```
./configure --enable-cachemgr-hostname=squidproxy.example.com
```

سویچ :enable-arp-acl

شما با استفاده از این سویچ می توانید قابلیت ACL را مبتنی بر MAC را در سرور خود راه اندازی کنید، این قابلیت به صورت پیش فرض غیرفعال است و شما باید آنرا فعال کنید.

سویچ :enable-default-err-language

در زمانی که squid خطای در برقراری ارتباط پیدا می کند یک صفحه خطا برای کاربر درخواست کنند صفحه ارسال می کند که در زبانهای مختلف وجود دارد شما می توانید زبان این صفحه خطا را با استفاده از این سویچ مشخص کنید به صورت زیر:

```
./configure --enable-default-err-language=Spanish
```

قابلیت transparentدر :Squid

شما برای استفاده از یک سرور Proxy باید در مرورگر وب خود تنظیماتی را اعمال کنید ولی در بسیار از موارد سرویس دهنده های اینترنت یا همان ISP ها قصد مدیریت پهنای باید مصرفی از سمت کاربران خود را دارد و شاید مدیران شبکه هم دوست دارند گزارشات استفاده هر کاربر را مشاهده کنند پس برای انجام دادن این کار باید تمام ترافیک های وب شبکه خود را به سمت سرور Squid منتقل کنند برای انجام دادن این کار شما نیاز به یک فایروال دارید که در FreeBSD فایروالهای ipfw و pf وجود دارد و شما می توانید از هر کدام از این دو فایروال استفاده کرده و ترافیک را به سمت سرور Squid منتقل کنید و باید سرور Squid هم از این قابلیت پشتیبانی کند برای فعال کردن این قابلیت از سویچ های زیر استفاده کنید:

:enable-ipfw-transparent

این سویچ برای استفاده از فایروال IPFW در نظر گرفته شده است.

:enable-pf-transparent

این سویچ برای استفاده از فایروال IPFW در نظر گرفته شده است.

در بخشی مناسب با روش کار با حالت transparent آشنا خواهید شد.

سویچ :with-logdir

شما می توانید برای راحتی در مشاهده Log های برنامه برای آن مسیری با استفاده از این سویچ ایجاد کنید در FreeBSD تمام لاگها در زیر شاخه /var/log ذخیره می شود و برای مدیریت جداگانه Log بهتر است که مسیر را خود تعیین کنید. در زیر شما یک مثال از این بخش را مشاهده می کنید:

```
./configure --with-logdir=/var/log/squid/
```

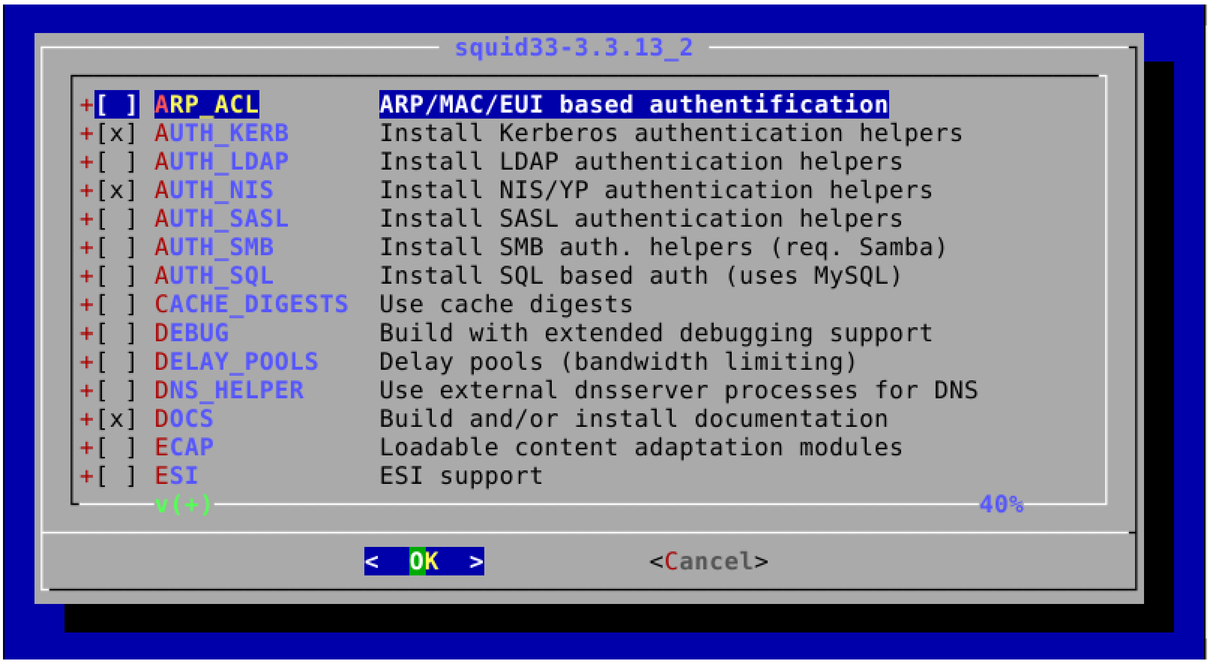
سویچ : with-pidfile

در FreeBSD در زیر شاخه /var/run هر برنامه ای که جدید به سیستم اضافه می شود PID خود را در فایلی به نام خودش ذخیره می کند که سایر پردازشها بتوانند از طریق این PID با برنامه در ارتباط باشند شما می توانید با استفاده از این سویچ مسیر این فایل را تغییر دهید.

نصب Squid با استفاده از کد برنامه:

حال بعد از انتخاب کردن بخشهای مورد نیاز خود در فرمان /configure. این فرمان را اجرا کنید و در صورتی که با خطایی مواجه نشدید با دسترسی عادی فرمان Make را اجرا کنید و در صورتی که این فرمان هم به درستی به پایان رسید با سطح دسترسی کاربر root فرمان make install را اجرا کنید تا Squid نصب شود.

بخشهای فرمان Configure را می توانید با استفاده از سیستم ports در FreeBSD بدون نیاز به تایپ کردن انتخاب کنید و بسته نصبی خاص خود را ایجاد کنید. در شکل زیر آنرا مشاهده می کنید:



نمایش تنظیمات در زمان نصب کردن Squid با استفاده از ساختار ports در FreeBSD

راه اندازی Squid و برنام های sqtop وsquidclient

بعد از اتمام مراحل نصب شما با راه اندازی سریع و دو برنامه کاربردی برای تست کردن و نمایش وضعیت Squid می پردازیم.
بعد از نصب در FreeBSD شما برای دسترسی داشتن به فایل اصلی پیکربندی آن باید به شاخه /usr/local/etc/squid مراجعه کنید. البته فایل اجرایی squid در شاخه /usr/local/sbin قرار گرفته و صفحه های توضیحات man آن هم در زیر شاخه /usr/local/man قرار گرفته است. شما با استفاده از فرمان Whereis به صورت نمایش داده شده در شکل زیر می توانید محل هر برنامه را مشخص کنید:

```
Terminal
File  Edit  View  Terminal  Tabs  Help
root@FreeBSD:/usr/local/etc/squid # whereis squid
squid: /usr/local/sbin/squid /usr/local/man/man8/squid.8.gz /usr/ports/www/squid
root@FreeBSD:/usr/local/etc/squid # whereis squidclient
squidclient: /usr/local/sbin/squidclient /usr/local/man/man1/squidclient.1.gz
root@FreeBSD:/usr/local/etc/squid #
```

نمایش محل فایل های اصلی برنامه squid

```
cache_dir ufs /var/cache/ 500 16 256

acl my_pc src 192.168.2.21 # Replace with your IP address

http_access allow my_pc
```

سرور Squid فایلهای کش شده خود را برروی هارد دیسک ذخیره می کند قبل از هر اقدامی باید مسیر این شاخه را مشخص کرده و در مرحله بعدی آنرا ایجاد کنید خط اول این کار را انجام می دهد.

در خط دوم شما با استفاده از acl می توانید آدرس IP سیستم خود را مشخص کنید که با استفاه از آن قصد استفاده کردن از Squidرا دارید، این بخش به تنظیمات شبکه شما بستگی دارد. در این خط شما لیستی ایجاد کرده اید به نام my_pc که فقط یک آدرس در آن اضافه کرده اید

در خط سوم شما با استفاده از تگ http_access به لیست بالا اجازه استفاده می دهید، این سه خط را در بالای فایل squid.confقرار داده تا در مرحله بعدی شاخه Cache را ایجاد کنید

```
Terminal
File  Edit  View  Terminal  Tabs  Help
^[ (escape) menu  ^y search prompt  ^k delete line   ^p prev li   ^g prev page
^o ascii code     ^x search         ^l undelete line ^n next li   ^v next page
^u end of file    ^a begin of line  ^w delete word   ^b back 1 char
^t top of text    ^e end of line    ^r restore word  ^f forward 1 char
^c command        ^d delete char    ^j undelete char ^z next word
=====line 8 col 0 lines from top 8 ==========================================
cache_dir ufs /var/spool/squid 500 16 256
acl my_pc src 192.168.2.21 # Replace with your IP address
http_access allow my_pc
#
```

ویرایش کردن فایل squid.conf

در مرحله بعدی شما باید شاخه اصلی را که در فایل پیکربندی مشخص کرده اید را ایجاد کنید و صاحب این شاخه را کاربر squidقرار دهید به صورت زیر:

```
# mkdir /var/spool/squid


# chown squid:squid /var/spool/squid
```

حال در مرحله بعد فرمان squid را با سویچ z به صورت زیر اجرا کنید تا خروجی ایجاد فایل را مشاهده کنید به صورت شکل زیر:

```
Terminal
File  Edit  View  Terminal  Tabs  Help
2016/10/14 02:21:22 kid1| Set Current Directory to /var/squid/cache
2016/10/14 02:21:22 kid1| Creating missing swap directories
2016/10/14 02:21:22 kid1| /var/spool/squid exists
2016/10/14 02:21:22 kid1| Making directories in /var/spool/squid/00
2016/10/14 02:21:23 kid1| Making directories in /var/spool/squid/01
2016/10/14 02:21:23 kid1| Making directories in /var/spool/squid/02
2016/10/14 02:21:23 kid1| Making directories in /var/spool/squid/03
2016/10/14 02:21:23 kid1| Making directories in /var/spool/squid/04
2016/10/14 02:21:23 kid1| Making directories in /var/spool/squid/05
2016/10/14 02:21:23 kid1| Making directories in /var/spool/squid/06
2016/10/14 02:21:23 kid1| Making directories in /var/spool/squid/07
2016/10/14 02:21:23 kid1| Making directories in /var/spool/squid/08
2016/10/14 02:21:23 kid1| Making directories in /var/spool/squid/09
2016/10/14 02:21:23 kid1| Making directories in /var/spool/squid/0A
2016/10/14 02:21:23 kid1| Making directories in /var/spool/squid/0B
2016/10/14 02:21:23 kid1| Making directories in /var/spool/squid/0C
2016/10/14 02:21:23 kid1| Making directories in /var/spool/squid/0D
2016/10/14 02:21:23 kid1| Making directories in /var/spool/squid/0E
2016/10/14 02:21:23 kid1| Making directories in /var/spool/squid/0F

root@FreeBSD:/var # cd /var/spool/squid/
root@FreeBSD:/var/spool/squid # ls
00      02      04      06      08      0A      0C      0E
01      03      05      07      09      0B      0D      0F
root@FreeBSD:/var/spool/squid #
```

ساختار شاخه کش در Squid

راه اندازیsquid

بعد از انجام مراحل بالا اگر سرور شما دارای آدرس IP معتبر و تنظیمات DNS درست باشد با استفاده از فرمان زیر راه اندازی می شود:

```
#/usr/local/etc/rc.d/squid onestart
```

در صورتی که تنظیمات درست نباشد شما با پیغامهایی به صورت زیر مواجه می شوید:

```
Terminal
File Edit View Terminal Tabs Help
root@FreeBSD:/var/spool/squid # /usr/local/etc/rc.d/squid onestart
Starting squid.
2016/10/14 02:33:13| WARNING: Could not determine this machines public hostname.
 Please configure one or set 'visible_hostname'.
2016/10/14 02:33:14| WARNING: Could not determine this machines public hostname.
 Please configure one or set 'visible_hostname'.
2016/10/14 02:33:16| WARNING: Could not determine this machines public hostname.
 Please configure one or set 'visible_hostname'.
```

راه اندازی کردن سرور squid با استفاده از ساختار rc.d

برای رفع این مشکل خط زیر را در فایل squid.conf اضافه کنید:

```
visible_hostname squid.example.com
```

حال فرمان را دوباره راه اندازی کنید تا خروجی به صورت زیر مشاهده کنید:

```
Terminal
File Edit View Terminal Tabs Help
root@FreeBSD:/var/spool/squid # /usr/local/etc/rc.d/squid onestart
Starting squid.
root@FreeBSD:/var/spool/squid # sockstat -l4
USER     COMMAND    PID   FD PROTO  LOCAL ADDRESS         FOREIGN ADDRESS
squid    squid      6668  6  udp4 6 *:58486               *:*
squid    squid      6668  8  udp4   *:58908               *:*
squid    squid      6668  16 tcp4 6 *:3128                *:*
_ntp     ntpd       4255  7  udp4   192.168.85.128:123    *:*
_ntp     ntpd       4255  10 udp4   127.0.0.1:123         *:*
root     sendmail   799   3  tcp4   127.0.0.1:25          *:*
root     sshd       796   4  tcp4   *:22                  *:*
root     syslogd    576   7  udp4   *:514                 *:*
root@FreeBSD:/var/spool/squid #
```

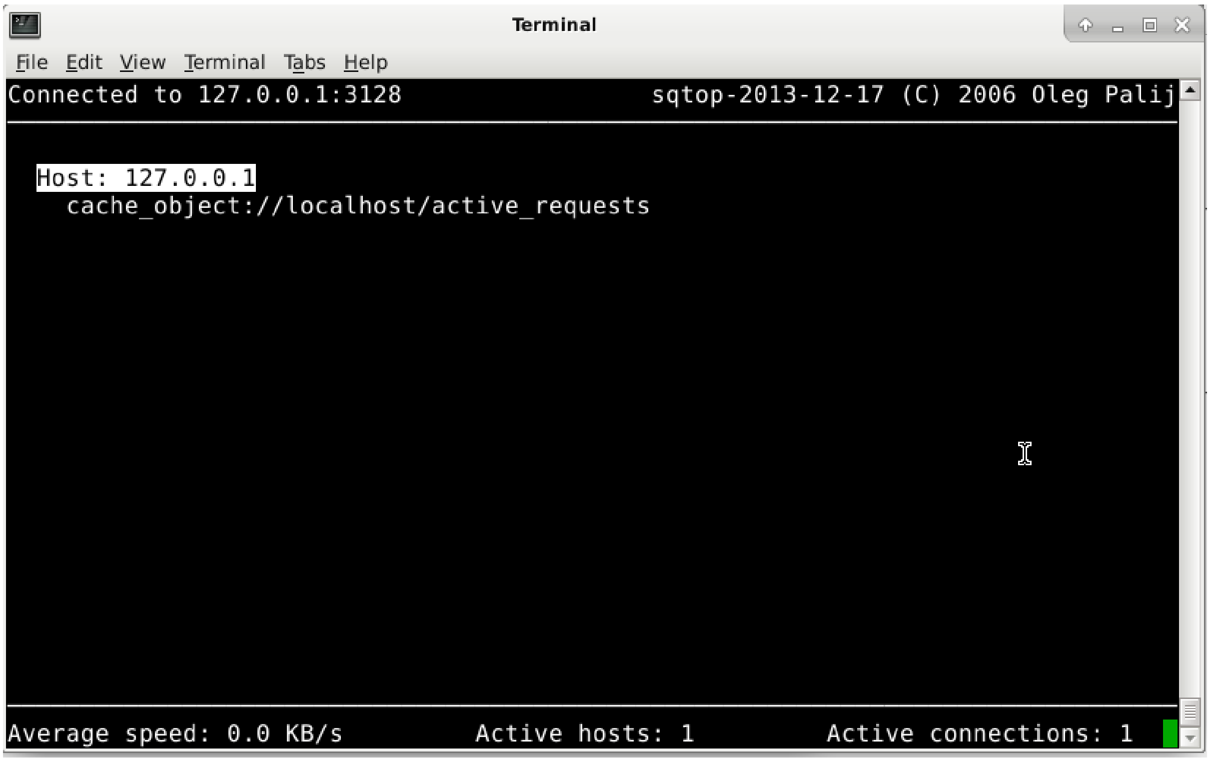
نمایش وضعیت پورتهای باز سرور Squid

برنامه :squidtop

برنامه ای برای مشاهده وضعیت عملکرد لحظه ای در squid وجود دارد به نام sqtop که شما می توانید از طریق ports و به صورت زیر این بسته را نصب کنید ب صورت زیر:

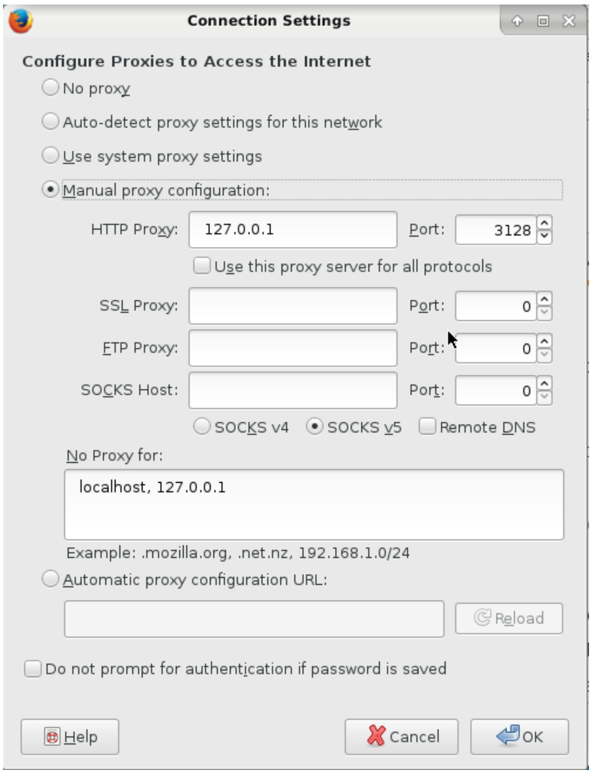
```
#/usr/ports/net/sqtop
#make
#make install
```

در مرحله بعد فرمان را Sqtop را اجرا کنید تا برنامه برای شما به صورت زیر اجرا شود:

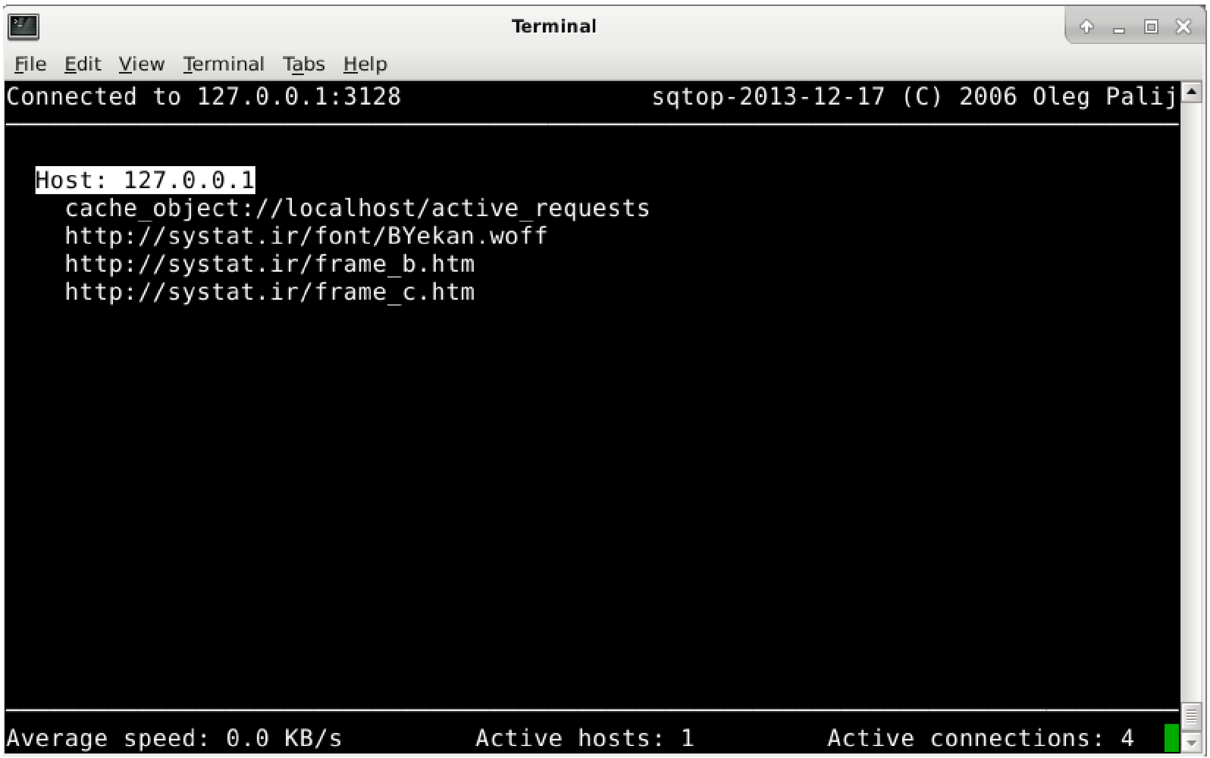


خروجی فرمان sqtop

اگر سرور squid راه اندازی شده باشد در خط اول از این فرمان درسمت را وضعیت connected را مشاهده می کنید، بعد از تنظیم کردن بخش proxy از مرورگر وب خود به صورت زیر صفحه های باز شده را در این برنامه مشاهده می کنید:



حال بعد از باز کردن یک صفحه از طریق مرورگر خود خروجی برنامه Sqtop به صورت زیر تغییر خواهد کرد:



نمایش تغییرات sqtop

برنامه squidclient

در زمان نصب برنامه squid برنامه دیگری هم برای چک کردن سرور به صورت خودکار وجود دارد به نام squidclien که کافیست بعد از این فرمان نام سایت مورد نظر را تایپ کنید تا در قالب یک فایل html خروجی را برای شما نمایش دهد در صورتی که هیچ پیغامی برای شما نمایش داده نشود این بدان معنی است که سرور شما به درستی پیکربندی نشده است.

سرور NFS در FreeBSD

سیستم عامل FreeBSD از پروتکل NFS برای اشتراک گذاری فایل در شبکه در بین سیستم عامل های مختلف استفاده می کند، در زمانهای قدیم برای اشتراک فایل بین WINDOWS و FreeBBSD سرور دیگری به نام samba ایجاد شد و هنوز هم این سرور وجود دارد، ولی در حال حاضر سیستم عامل windows هم از nfs پشتیبانی می کند، در این مقاله شما با اشتراک گذاری فایل بین windows و FreeBSD با استفاده از NFS آشنا می شوید.

راه اندازی NFSدر FreeBSD

سرور NFS به صورت پیش فرض در FreeBSD نصب شده و شما فقط کافیست که آنرا فعال کنید. برای اینکار باید سرویسهای زیر را فعال کنید:

برنامه **nfsd**که برای پاسخ دادن به درخواستهای سرویس گیرنده ها راه اندازی می شود.

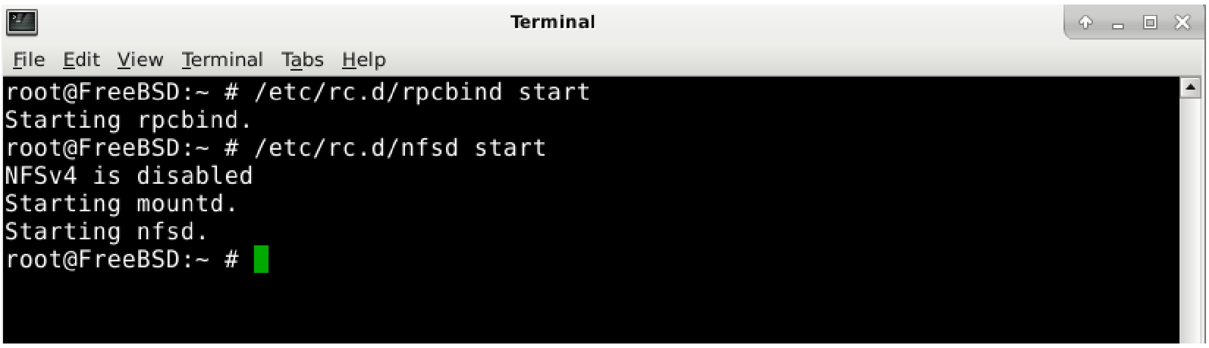
برنامه **mountd**این سرویس برای انجام دادن درخواستهای رسیده شده به nfsd استفاده می شود.

برنامه **rpcbind**این برنامه لازم است تا سرویس گیرنده ها بتواند پورت سرور NFS را برای اتصال پیدا کنند.

در قدم اول شما باید از طریق فایل معروف rc.conf این سرویسها را بری راه اندازی تعریف کنید به صورت زیر:

```
rpcbind_enable="YES"
nfs_server_enable="YES"
mountd_enable=”YES”
```

حال شما با استفاده از فرمان service یا فرمان های rc.d این سه سرویس را راه اندازی کنید به صورت زیر:



راه اندازی کردن سرویس nfsd با rc.d در FreeBSD

همانطوری که مشاهده می کنید با راه اندازی سرویس nfsd به صورت خودکار سرویس mountd هم راه اندازی می شود.

برای اینکه از راه اندازی سرویس ها برروی شبکه مطمن باشید فرمان sockstat را به صورت زیر اجرا کنید:

```
root@FreeBSD:~ # sockstat -l4
USER     COMMAND    PID   FD PROTO  LOCAL ADDRESS         FOREIGN ADDRESS
root     nfsd       1705  5  tcp4   *:2049                *:*
root     mountd     1703  8  udp4   *:761                 *:*
root     mountd     1703  9  tcp4   *:761                 *:*
root     rpcbind    1675  9  udp4   *:111                 *:*
root     rpcbind    1675  10 udp4   *:835                 *:*
root     rpcbind    1675  11 tcp4   *:111                 *:*
root     nfsuserd   1145  3  udp4   *:930                 *:*
root     nfsuserd   1144  3  udp4   *:930                 *:*
root     nfsuserd   1143  3  udp4   *:930                 *:*
root     nfsuserd   1142  3  udp4   *:930                 *:*
root     nfsuserd   1141  3  udp4   *:930                 *:*
root     sendmail   868   3  tcp4   127.0.0.1:25          *:*
```

نمایش وضعیت پورت های باز شبکه

به اشتراک گذاشتن شاخه:

برای به اشتراک گذاشتن شاخه شما در ابتدا باید فایلی به نام exports را در زیر شاخه /etc ایجاد کنید به صورت زیر:

```
root@FreeBSD:~ # touch /etc/exports
root@FreeBSD:~ #
```

ایجاد کردن فایل exports

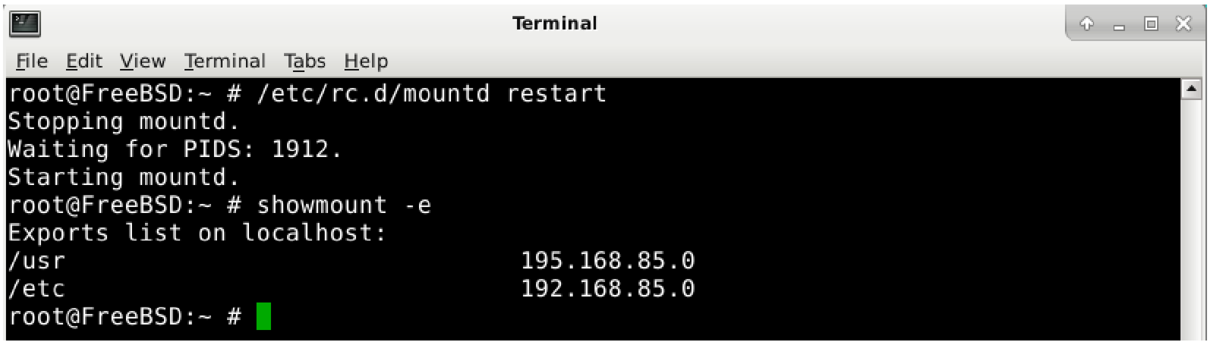
شاخه های به اشتراک گذاشته شده را در این فایل قرار می دهید در بخش اول شما باید نام شاخه را مشخص کنید و در قسمت دوم نوع دسترسی را مشخص کنید و در بخش آخر هم آدرس ip یا اسم سیستم هایی که قصد استفاده از این شاخه را دارند را مشخص کنید، در صورت مشخص نکردن بخش آخر منظور همه یا همان everyone می باشد در شکل زیر نمونه ای از این فایل را مشاهده می کنید:

```
root@FreeBSD:~ # cat /etc/exports
/usr     -alldirs -network 195.168.85.0/24
/etc     -ro      -network 192.168.85.0/24
root@FreeBSD:~ #
```

شاخه های به اشتراک گذاشته شده در فایل exports

در این فایل ما دو شاخه به نام های /usr و /etc را share کردیم. شاخه اول را با تمام زیر شاخه هایش را برای شبکه 192.168.85.0 به اشتراک گذاشتیم و شاخه دوم /etc را برای همان شبکه فقط در حالت خوانده شدن به اشتراک گذاشتیم.

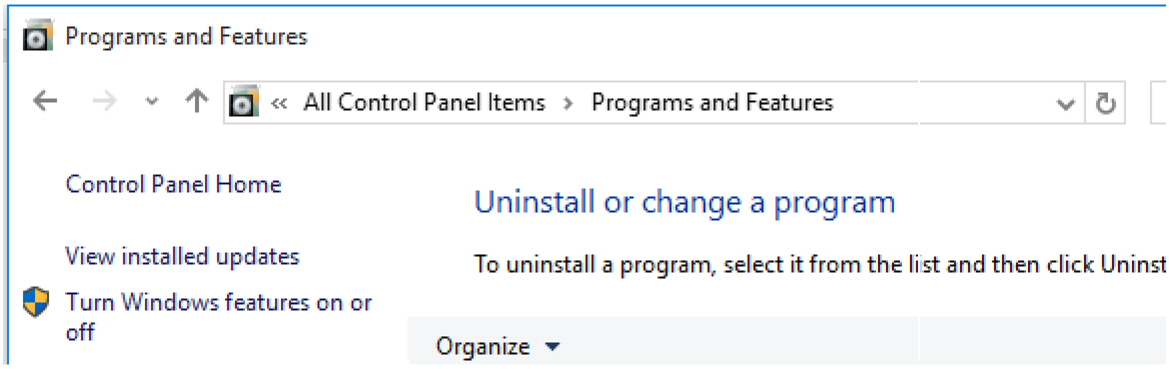
بعد از اعمال این تغییرات سرویس mountd را به صورت زیر راه اندازی مجدد کنید و از فرمان showmounts به صورت زیر برای نمایش شاخه های به اشتراک گذاشته استفاده کنید:



نمایش وضعیت شاخه های اشتراک گذاشته شده

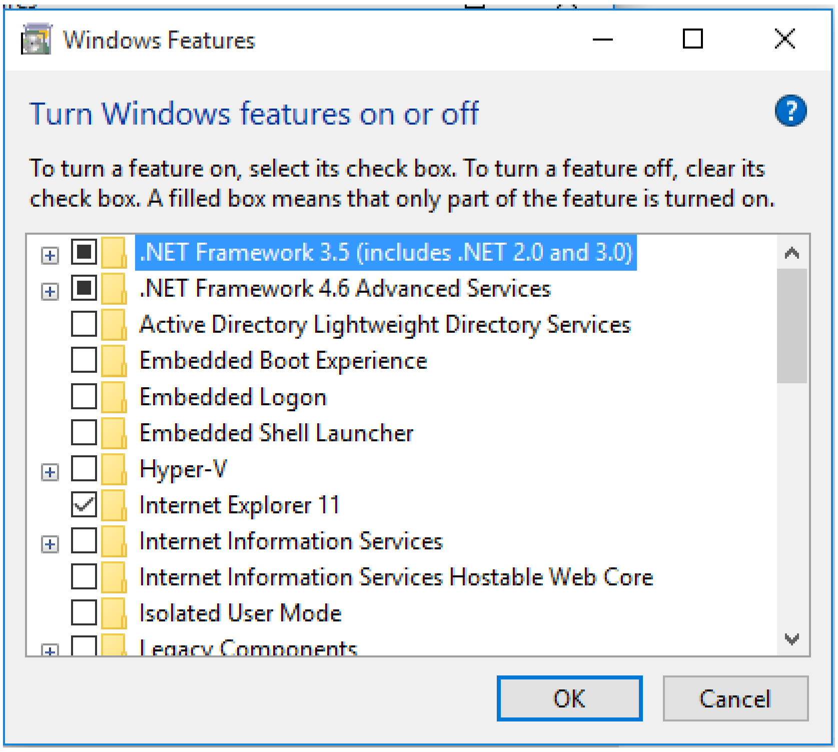
تنظیم کردن ونیدوز 10 برای استفاده از شاخه ها:

به صورت پیش فرض NFS client در سیستم عامل ویندوز راه اندازی نشده است و شما باید انرا فعال کنید برای این امر باید ابتدا به control panle رفته و در بخش Programs and Features وارد شده و در سمت چپ این بخش گزینه turn windows feature on or offرا باز کنید:



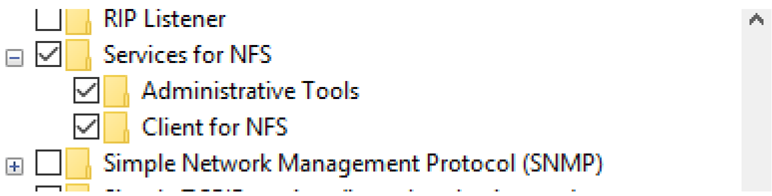
بخش اضافه کردن قابلیت به ویندوز

بعد از باز کردن شما با منوی به صورت زیر مواجه می شوید:



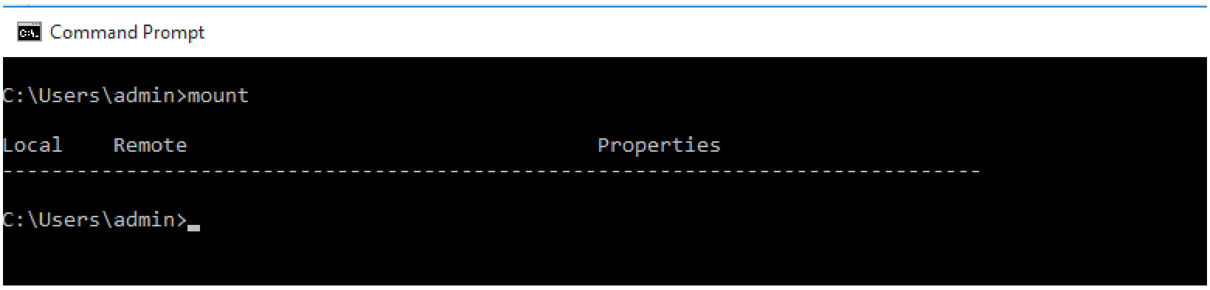
منوی اضافه کردن قابلیت به ویندوز

بخش service for nfs را باز کرده و هر دو تیک این بخش را فعال کنید:



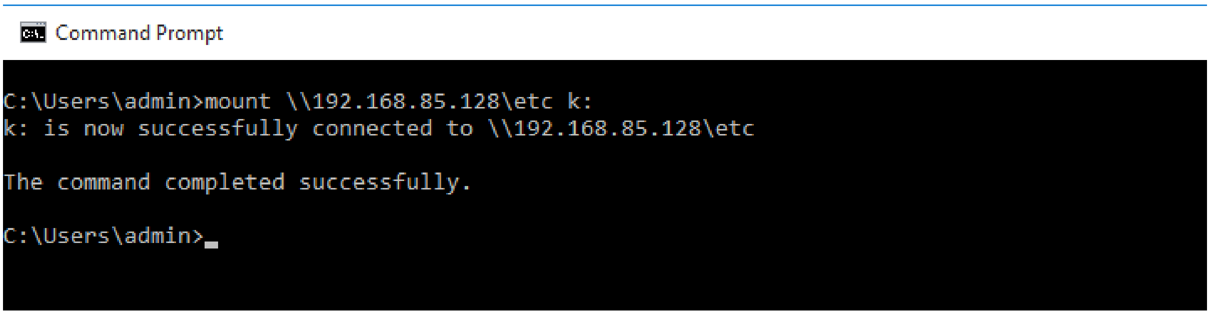
اضافه کردن NSF در بخش کلاینت

بعد از ok کردن این قابلیت به سیستم شما اضافه شده و شما می توانید از فرمان mount در windows استفاده کنید اجرا این فرمان را در cmd مشاهده می کنید:



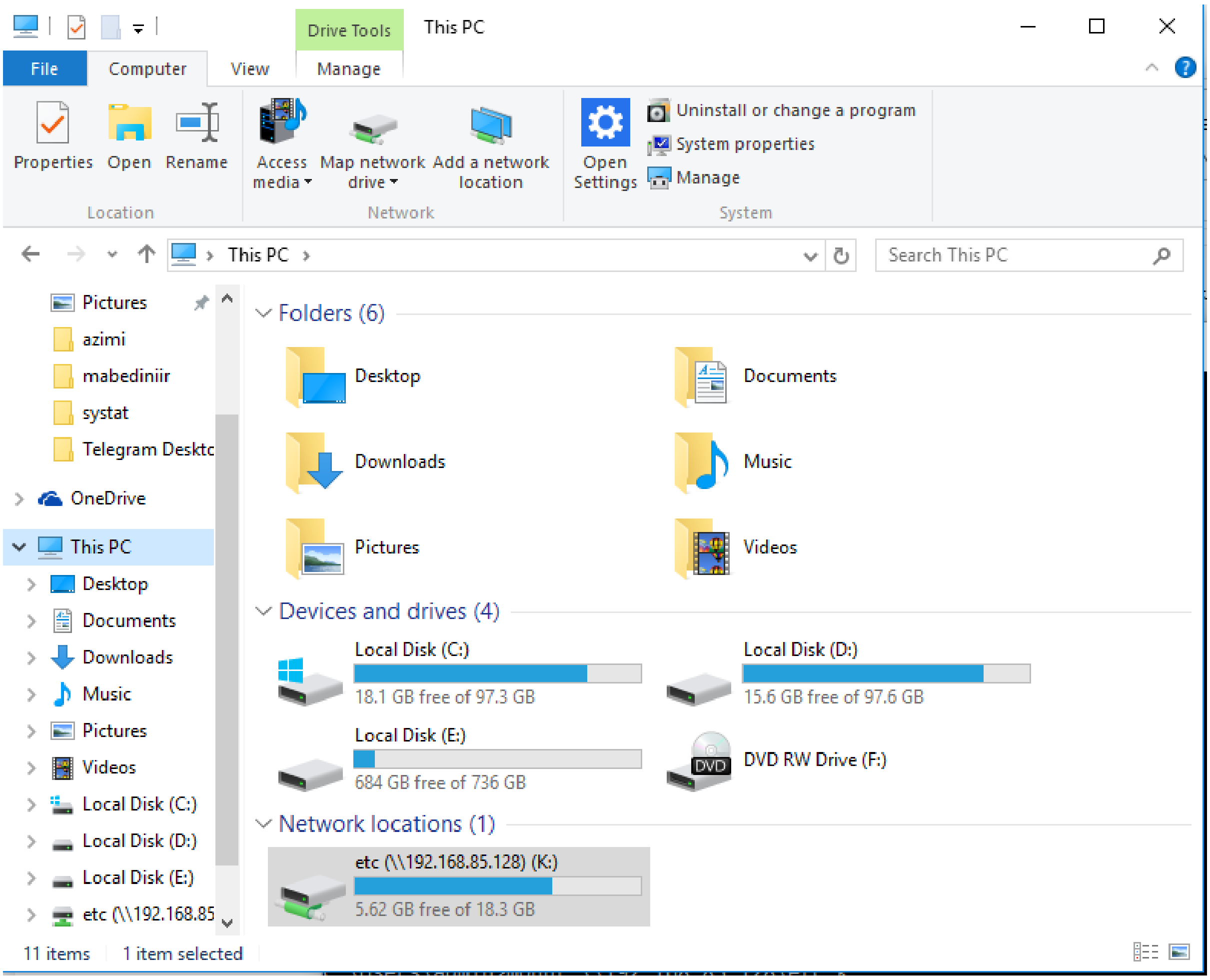
اجرا فرمان mount در ویندوز

برای mount کردن شاخه های مورد نظر کافیست که فرمان زیر را در cmd به صورت زیر اجرا کنید:



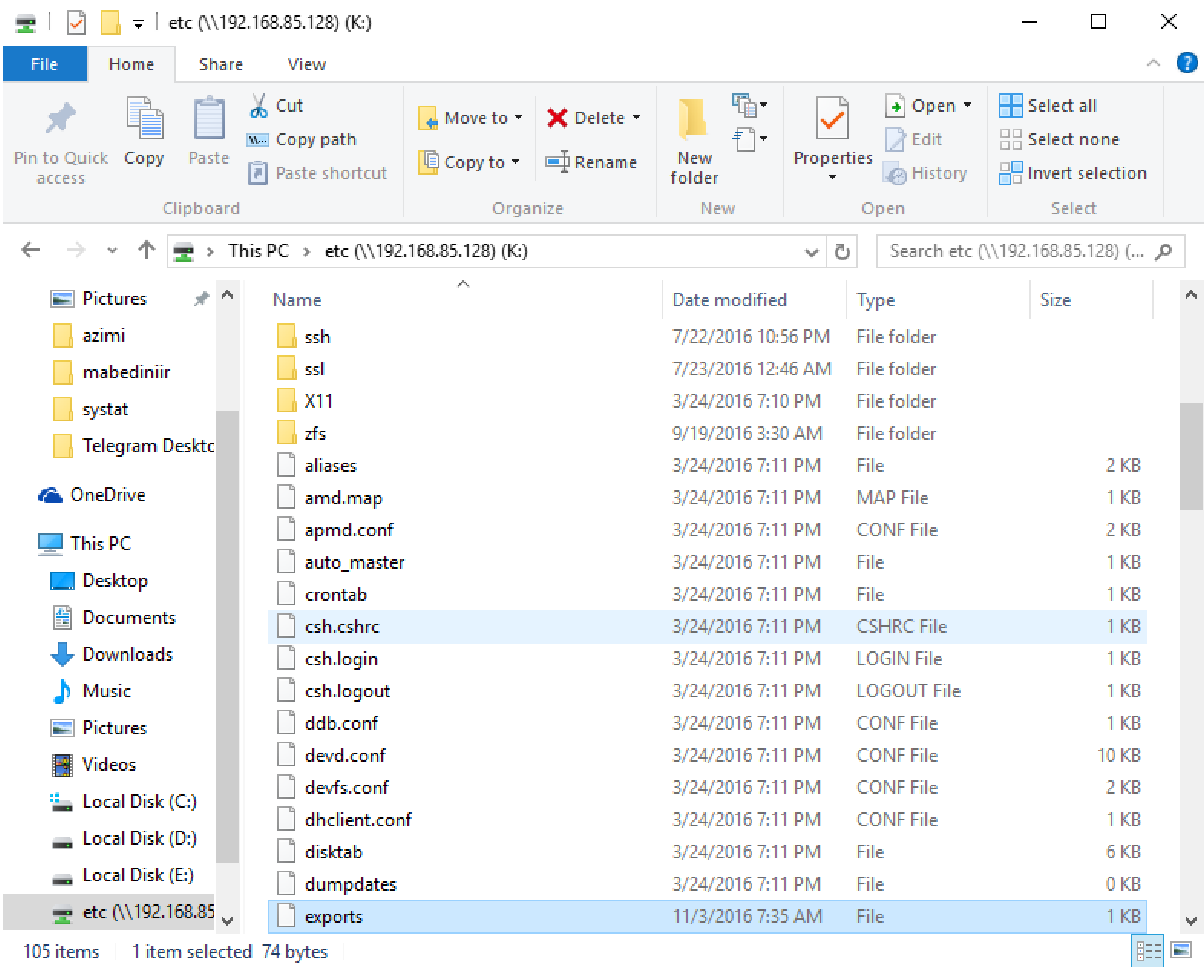
Mount کردن NFS در ویندوز

حال در بخش This PC این شاخه اضافه می شود:



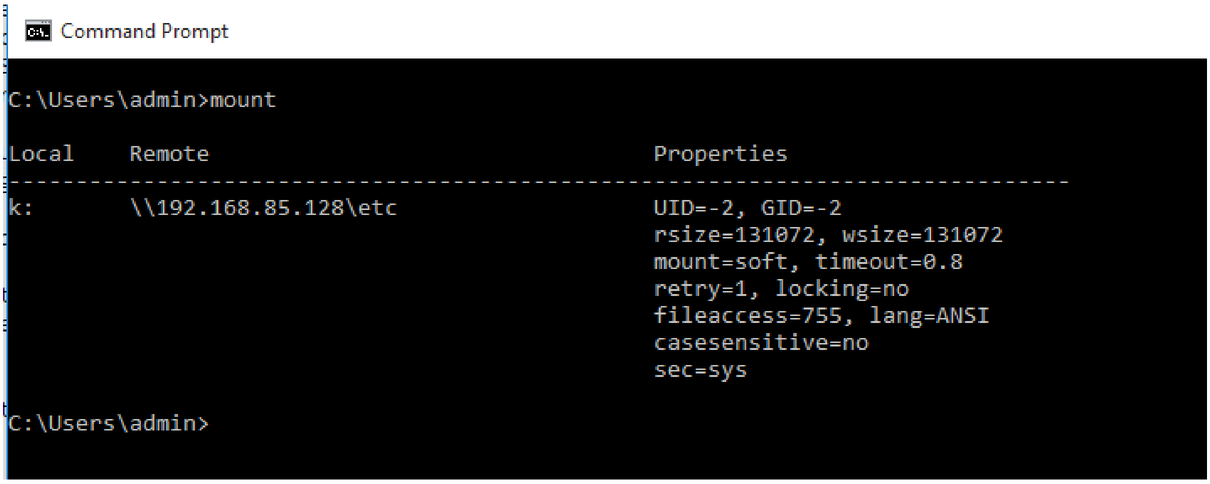
نمایش شاخه mount شده در My computer

شما می توانید به راحتی وارد این شاخه شوید:



وارد شدن از طریق ویندوز به شاخه های share شده

خروجی فرمان mount را در شکل زیر مشاهده می کنید:



خروجی فرمان mount در windows

برای umount کردن شاخه های اضافه شده کافیست که فرمان umount را با سویچ a– به صورت شکل زیر اجرا کنید:

Command Prompt

```
C:\Users\admin>umount -a
You have these active NFS connections:

k:       \\192.168.85.128\etc
Continuing will cancel the connections.

Do you want to continue this operation? (Y/N) [N]:y

Disconnecting           k:       \\192.168.85.128\etc
There are open files and/or incomplete directory searches pending on the connection.

Do you want to continue this operation? (Y/N) [N]:y

The command completed successfully.

C:\Users\admin>
```

اجرا کردن فرمان umount در ویندوز

این فرمان در ابتدا شاخه های mount شده را نمایش می دهد و شما با پاسخ Y به سوالات آن می توانید شاخه ها را unmounts کنید.

راه اندازی web سرور در FreeBSD

به صورت پیش فرض در FreeBSD وب سروری نصب نیست و شما می توانید هر وب سروری را که دوست دارید نصب کنید، در سیستم عامل OpenBSD وب سرور apache به صورت پیش فرض نصب است. در FreeBSD هم ما قصد داریم از این وب سرور استفاده کنیم و در ادامه با نصب و راه اندازی آن بیشتر آشنا شوید.

در این مقاله ما روژن 2 x.از این وب سرور را مورد بررسی قرار می دهیم.

نصب : apache

شما به دو روش می توانید به نصب این سرور بپردازید در بخش اول استفاده از بسته از پیش کامپایل شده یا همان باینری و بخش دوم اگر قصد دارید که تنظیمات خاص خود را اضافه کنید استفاده از سیستم پورت در FreeBSD البته شما می توانید به صورت دستی هم به نصب بپردازید که در این بخش مورد بحث ما نیست.

برای نصب کردن از طریق بسته های باینری باید از فرمان pkg install به صورت زیر استفاده کنید:

برای نصب کردن از طریق سیستم پورت کافیست که وارد شاخه زیر شده:

```
#/usr/local/ports/www/apache24
```

وفرمان های make را در ابتدا و فرمان make install را با دسترسی کاربر root اجرا کنید. در میان مراحل make هم شما می توانید تنظیمات دلخواه خود را به بسته اضافه کنید. در انتها هم می توانید با استفاده از make clean تمام بسته های اضافه و فایلهای سورس را پاک کنید.

فایل پیکربندی:

بعد از نصب کردن apache این برنامه فایل های خود را در زیر شاخه /usr/local/etc در شاخه ای به نام pache24 قرار می دهد در این شاخه فایل اصلی پیکربندی این برنامه به نام httpd.conf قرار دارد که تمام تنظیمات از این بخش شروع می شود، این فایل هم مثل سایر فایلهای پیکربندی دارای بخش ها مختلف است و خطوط که با # شروع می شود هم توضیحات بوده و هیچ تاثیری در برنامه ندارند. در این بخش با بخشهای مهم این فایل پیکربندی آشنا می شوید.

بخش "ServerRoot "/usr/local

این بخش محل قرار گرفتن فایلهای برنامه apache زیر شاخه های مربوطه است، برای مثال برنامه های اجرایی در زیر شاخه های bin و sbin از /usr/local قرار می گیرند و همه برنامه های دیگر مثل فایل پیکربندی در زیر شاخه etc/apache24 از زیر شاخه /usr/local قرار میگیرند. شما به راحتی در این مسیر می توانید فایلهای مورد نظر خود را پیدا کنید. این مسیر هم در فایل پیکربندی استفاده شده و برای آدرس دهی بخشهای دیگر هم از این آدرس استفاده می کنند.

بخش ServerAdmin you@example.com

اگر برای سرویس شما مشکلی ایجاد شده باشد از طریق این آدرس ایمل یک میل برای شما ارسال می شود، شما مب توانید این بخش را با آدرس مدیر سیستم تغییر دهید.

بخشListen

در این بخش شما می توانید پورت پیشفرض برنامه که 80 است را تغییر دهید، حتی اگر سرور شما دارای چند آدرس IP باشد شما در این بخش می توانید تنظیم کنید که برروی هر آدرس بروی کدام پورت به ارایه کردن سرویس بپردازد.

بخش"DocumentRoot "/usr/local/www/apache2x/data

این بخش بسیار بخش مهمی است در سرور apache که در حقیقت محلی است که فایلهای وب سایت شما در آن ذخیره می شود . شما بعد از طراحی سایت خود کافیست که ان را در این محل کپی کنید و فایلهی هم به نام index.html داشته باشید که صفحه اول سایت شما باشد بعد از راه اندازی سرور شما برنامه apache این فایل را از این مسیر برای شما بارگذاری می کند.

راه اندازی :apache

شما از چند طریق می توانید برای اینکه بتوانید از طریق سیستم rc.conf سرور apache را راه اندازی کنید و در زمان راه اندازی خودکار سیستم عامل این سرویس هم راه اندازی شود باید ابتدا در فایل معروف rc.conf در زیر شاخه /etc خط زیر را اضافه کنید:

```
apache24_enable="YES"
```

حال شما می توانید به راحتی با استفاده از فرمان های rc.d به صوت زیر این سرور را راه اندازی کنید، غیرفعال کنید و یا راه اندازی مجدد کنید:

```
#/us/local/etc/rc.d/apache24 start
```

در FreeBSD فرمانی هم به نام service اضافه شده است که شما بتوانید با استفاده از آن سرویس های مورد نیاز خود را مدیریت کنید به صورت زیر می توایند:

```
# service apache24 start
```

خب بعد از اینکه این فرمان بدون خطا راه اندازی شده برای چک کردن این سرویس کافیست که در مرورگر سیستم خود localhostرا وارد کنید تا پیغام It is working را مشاهده می کنید.

برای چک کردن فایل پیکربندی کافیست که فرمان service را به صورت زیر اجرا کنید:

```
# service apache24 configtest
```

قابلیت : Virtual Hosting

شما با استفاده از این قابلیت می توانید چندین سایت را برروی یک سرور apache راه اندازی کند. هاست های مجازی می تواند بر اساس آدرس IP باشد یا بر اساس نام . در روش استفاده از IP برای هر سایت باید یک آدرس IP داشته باشید، در روش استفاده از نام به جای آدرس IP سرور به هدر درخواست ارسال شده از سمت Client نگاه می کند و سایت مورد نظر را در اختیار کاربر قرار می دهد در این روش شما می توانید از یک ادرس IP یکسان برای سایت های مختلف استفاده کنید.

برای پیکربندی این بخش شما باید برای هر سرور مجازی از یک بلوک به نام virtual host به صورت زیر ایجاد کنید:

```
<VirtualHost *>
    ServerName www.systat.ir
    DocumentRoot /www/ systat.ir
</VirtualHost>
```

برای هر هاست مجازی باید یک نام در بخش server Name انتخاب کنید و باید یک مسیر برای شاخه DocumentRoot انتخاب کنید.

نصب و راه اندازیSSH

یکی از دلایلی که من به سیستم عامل های خط فرمانی علاقمند هستم قابلیت انجام دادن همه کارها با استفاده از خط فرمان است، شما می توانید از هر مکانی با هر سرعت اینترنتی به سرور خود در هر جای دنیا متصل شوید و کارهای خود را انجام دهید. در زمان های قدیم برای برقرار ارتباط از سرویس telnet استفاده می شود که این سرویس امن نبوده و در مقابل شنود در شبکه بی دفاع است، در مقابل این سرویس، سرویس دیگری به نام ssh ایجاد و طراحی شد ارتباطی امن در بستر نا امن ایجاد می کند و شما به صورت امن می توانید فرمانهای خود را راه اندازی کنید و نگران سرقت شدن اطلاعات خود نباشید. حتی شما در برقرار ارتباط با سرور خود می توانید از کلید های امنیتی به جای رمز عبور استفاده کنید.

در ادامه این مقاله شما با روش فعال کردن سرور ssh در FreeBSD آشنا می شوید و بعد به سراغ روشهای برقرار ارتباط با سرور از طریق فرمان ssh آشنا می شوید. و در مرحله بعدی از کلید به عنوان رمزعبور برای برقرار ارتباط استفاده می کنید و در آخر روش tunnel زدن و bypass کردن firewall را شرح خواهیم داد.

فعال کردن سرور ssh

این سرور به صورت پیش فرض در سیستم عامل FreeBSD نصب شده و شما نیازی به نصب کردن آن ندارید و فقط باید آنرا فعال کنید برای فعال کردن این سرویس مثل همه سرویسهای دیگر شما می توانید هم از فرمان Service استفاده کنید و هم از سیستم rc.d برای این کار استفاده کنید. در ادامه دو روش مورد بررسی قرار می گیرد.

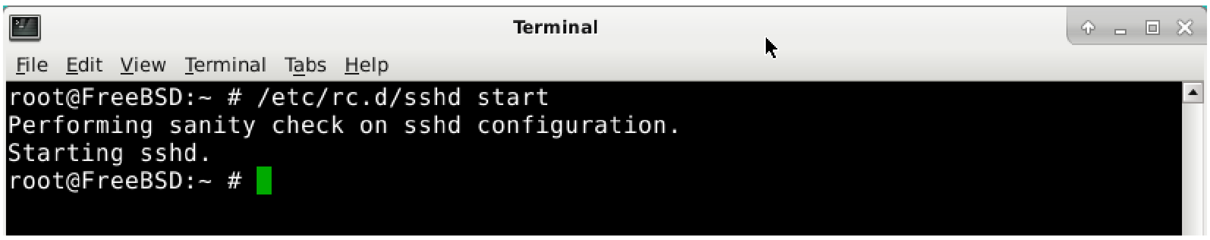
برای فعال کردن از طریق فرمان service کافیست فرمان زیر را اجرا کنید:

```
#service sshd start
```

برار راه اندازی دایمی سرویس ssh و راه اندازی خودکار در زمان راه اندازی سیستم کافیست که خط زیر را در فابل rc.conf در زیر شاخه /etc اضافه کنید.

```
sshd_enbale="YES"
```

حال بعد از اضافه کردن می توانید با استفاده از فرمانهای rc.d به صورت زیر این سرویس را راه اندازی کنید:



راه اندازی کردن ssh

برای اطمینان پیدا کردن از اینکه این سرویس راه اندازی شده و برروی شبکه در حال ارایه دادن سرویس است فرمان sockestatرا به صورت زیر اجرا کنید:

```
Terminal
File  Edit  View  Terminal  Tabs  Help
root@FreeBSD:~ # sockstat -l4
USER     COMMAND    PID   FD PROTO  LOCAL ADDRESS         FOREIGN ADDRESS
root     sshd       2239  4  tcp4   *:22                  *:*
root     ftpd       1448  6  tcp4   *:21                  *:*
root     mountd     1228  7  udp4   *:727                 *:*
root     mountd     1228  8  tcp4   *:727                 *:*
root     sendmail   867   3  tcp4   127.0.0.1:25          *:*
_ntp     ntpd       817   7  udp4   192.168.85.128:123    *:*
_ntp     ntpd       817   10 udp4   127.0.0.1:123         *:*
root     nfsd       749   5  tcp4   *:2049                *:*
root     rpcbind    617   9  udp4   *:111                 *:*
root     rpcbind    617   10 udp4   *:869                 *:*
root     rpcbind    617   11 tcp4   *:111                 *:*
root     syslogd    613   7  udp4   *:514                 *:*
?        ?          ?     ?  udp4   *:2049                *:*
root@FreeBSD:~ #
```

نمایش وضعیت پورتهای باز

در زمان راه اندازی اولین بار سرور ssh این سرور به صورت خودکار کلید های برقرار ارتباط را به صورت شماتیک برای سیستم شما ایجاد می کنید. با استفاده از فرمان rc.d شما می توانید بعد از راه اندازی هم این عمل را به صورت زیر انجام دهید:

```
Terminal
File  Edit  View  Terminal  Tabs  Help
root@FreeBSD:/etc/ssh # /etc/rc.d/sshd keygen
Generating RSA1 host key.
2048 SHA256:Dl1NO6AjYpyXXN7RHG12HfP1hRGsNq0CDnK6g5mn0ck root@FreeBSD (RSA1)
Generating RSA host key.
2048 SHA256:j6Tf03j3MBkDsWsMB9jwj5qEwQaq0/f1+RJI3h3Uz34 root@FreeBSD (RSA)
Generating DSA host key.
1024 SHA256:90YAprQoNGZg95Np6sRrRPtAX/DUCghDqgVDGIStCRA root@FreeBSD (DSA)
Generating ECDSA host key.
256 SHA256:kCYalbGAfcVAks0TTIUIahRVGGWmAouDdEJlqJ1jX30 root@FreeBSD (ECDSA)
Generating ED25519 host key.
256 SHA256:H5Wwomd9fae7+Lj0H7YFTTm6S03MZMyt9erub+uhqNk root@FreeBSD (ED25519)
root@FreeBSD:/etc/ssh # ls
moduli                          ssh_host_ed25519_key.pub
ssh_config                      ssh_host_key
ssh_host_dsa_key                ssh_host_key.pub
ssh_host_dsa_key.pub            ssh_host_rsa_key
ssh_host_ecdsa_key              ssh_host_rsa_key.pub
ssh_host_ecdsa_key.pub          sshd_config
ssh_host_ed25519_key
root@FreeBSD:/etc/ssh #
```

ایجاد کردن کلید

پیکربندی :sshd

بعد از راه اندازی سرویس ssh این فرمان sshd است که همه کارها را انجام می دهد. با این فرمان می توانید وضعیت سرور خود را در آن واحد تغییر دهید و یا تنظیمات اصلی سرور خود را از فایل پیکربندی این سرور واقع در مسیر /etc/ssh/sshd_config می خواند. در این بخش با فرمان sshd و فایل پیکربندی sshd_conf آشنا می شوید.

مسیر اصلی این فرمان در زیر شاخه usr/sbin/ قرار دارد این بدان معناست که فقط کاربر root می تواند به آن دسترسی داشته باشد. در ادامه با چند سویچ از این فرمان آشنا می شوید:

سویچ 4- با استفاده از این سویچ سرور ssh فقط از ipv4 استفاده می کند.

سویچ 6 -با استفاده از این سویچ سرور ssh فقط از ipv6 استفاده می کند.

سویچ c- در بخش استفاده از کلید برای اتصال به سرور با فایلی آشنا می شوید که محل ذخیره کردن کلید هاست، شما با استفاده از این سویچ می توانید محل پیش فرض این فایل را تغییر دهید.

سوسچ D- این سویچ برای مشاهده وضعیت سرور استفاده می شود.

سویچ - E بعد از این سویچ شما نام فایل گزارش گیری از سرور را انتخاب می کنید، این فایل جدا از گزارشگیری خود سیستم است.

سوسچ p- بعد از این سویچ شما می توانید شماره پورت دلخواه خود را تعیین کنید و سرور را برروی هر پورت دلخواه راه اندازی کنید.

سویچ t- این سویچ سرور شما را در حالت تست کردن قرار می دهد برای بروزرسانی کردن سرور قدیمی به سرور جدید استفاده می شود.

فایل پیکربندی :sshd_config

فایل پیکربندی دارای بخشهای مختلفی است در ادامه با بخشهایی آشنا می شوید که برای راه اندازی اولیه با آن بیشتر سروکار داریم. این فایل هم مثل همه فایلهای پیکربندی در FreeBSD است و خطوط که با # شروع می شوند و خطوط خالی کامنت است. این بخشها هم به حروف کوچک و بزرگ حساس هستن.

بخش AllowUsers

در مقابل این بخش ما باید لیستی از کاربرانی را که قصد دارید به آنها اجازه استفاده از سرویس ssh را داشته باشند قرار دهید. هر نام در این لیست باید با نام کاربری که قصد استفاده داشته باشد یکسان بوده و با فاصله از هم جدا شوند. به دلیل اینکه به صورت Local این سرویس راه اندازی شده و از RAS سرور برای تایید هویت استفاده نمی شود پس باید کاربر مورد نظر برروی سیستم اضافه شود.

بخشAllowGroups

این بخش هم مثل بخش قبلی است و شما می توانید به جای نام کاربران گروهی را تعیین کنید و هر کاربری را که نیاز به دسترسی به این سرویس را دارد را به آن گروه اضافه کنید و در این بخش از نام گروه به جای نام تک تک افراد استفاده کنید.

بخش AllowTcpForwarding

شما با استفاده از این بخش می توانید بعد از اتصال به سرور اگر این قابلیت فعال بود از سرور خود به عنوان یک پراکسی استفاده کنید. در بخش دور زدن فایروال به آن بیشتر می پردازیم. این بخش خود داری متغییرهای yes و all است که این قابلیت را فعال می کند، noکه این قابلیت را غیرفعال کرده و اگر هم قصد دارید که کاربر local از این قابلیت استفاده کند.

بخش Banner

این بخش مخشص کننده فایلی است که قبل از بخش ورود کاربر به سیستم برای کاربر نمایش داده می شود، این بخش می تواند شامل پیغامهای خوش آمد گویی هم باشد. این بخش به صورت پیش فرض غیرفعال است.

بخش ChrootDirectory

شما با استفاده از این بخش می توانید بعد از وارد شدن کاربر به سیستم مسیری که باید به آن دسترسی داشته باشد را مشخص کنید. شاخه مورد نظر شما باید شامل فایلهای لازم و شاخه های مورد نظر که برای ورود یک کاربر لازم است را داشته باشد مثل یک Shell به نام sh و همچنین شاخه /dev/ باید با نود های null zero stdin stdout stderr و tty باشد.

بخش DenyGroups

شما در این بخش می توانید نام گروه هایی که قصد دارید به آنها اجازه استفاده از SSH را ندهید را وارد کنید و باید نام هر گروه با یک فاصله از هم جدا شود.

بخشDenyUsers

در این بخش ما می توانید نام کاربرای را که قصد دارید به آنها اجازه استفاده از SSH را ندهید را وارد کنید و باید نام هر گروه با یک فاصله از هم جدا شود.

بخش Port

شما در این بخش می توانید شماره port پیش فرض سرور SSH که 22 است را تغییر دهید و یا برروی تعداد بیش از یک portسرور را راه اندازی کنید .

بخش ListenAddress

شما در این بخش می توانید آدرس IP ها و port های ماشین محلی خود را که قصد دارید فقط برروی آنها سرویس ارایه شود را مشخص کنید.

بخش MaxAuthTries

در این بخش شما می توانید تعداد تلاش های نا موفق برای ورود به سیستم را که هر ارتباط می تواند داشته باشد را مشخص کنید. بعد از این تعداد ارتباط از سمت سرور قطع خواهد شد.

بخش PermitRootLogin

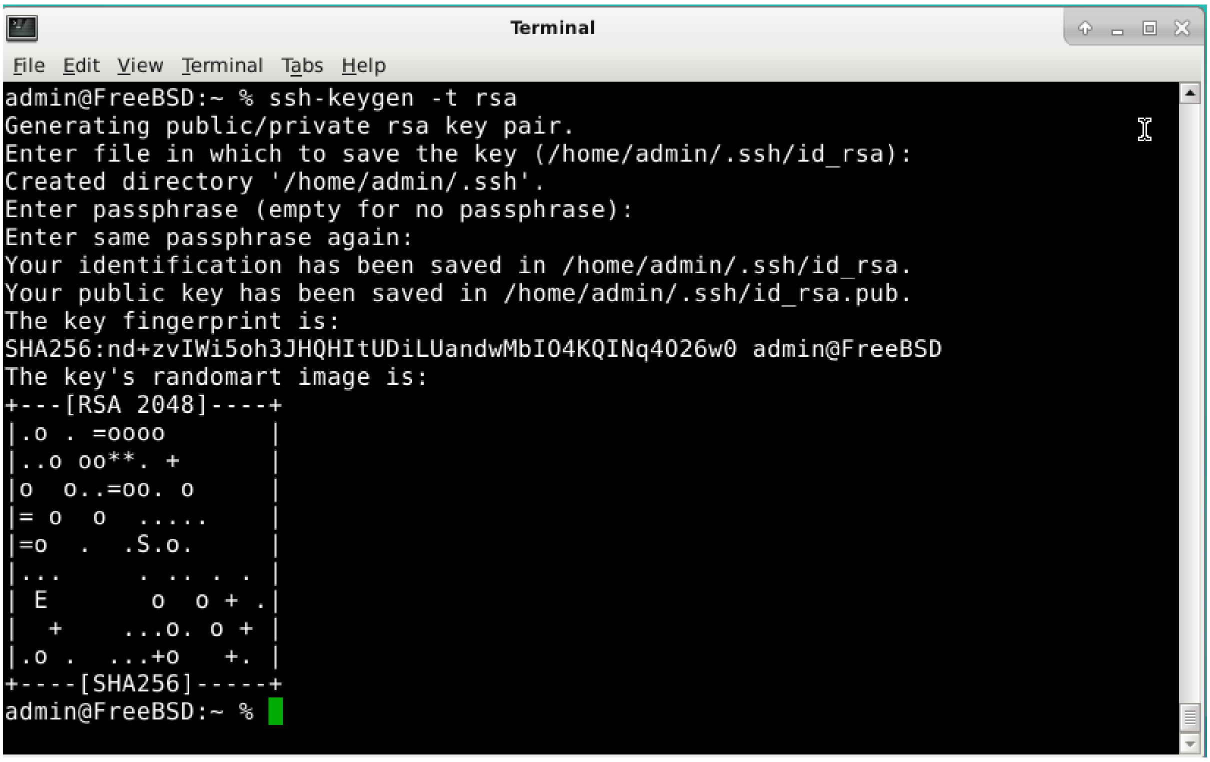
این بخشی است که در آن می توانید به تو خاص دسترسی کاربر Root را برای ورود به سیستم از طریق ssh را مشخص کنید. این بخش به صورت پیش فرض به کاربر root اجازه ورود نمی دهد. برای اینکه قصد داشته باشید که به کاربر root این اجازه را دهید بهتر است آن را YES قرار دهید. این بخش شامل متغییر های دیگری مثل prohibit-password without-passwordهم هست.

اتصال به سرور SSH از طریق کلید

برای بالا بردن سطح امنیتی سرور های SSH شما می توانید از کلید برای برقرار اتباط با سرور استفاده کنید بجای اینکه از نام کاربری و رمز عبور استفاده کنید، شما می توانید کلید خود را برروی سرور upload کنید و بعد بدون تایپ نام کاربری و رمز عبور به سرور متصل شوید. در حالت های دیگری شما می توانید سطح دسترسی همه کابرانی را که با رمزعبور و نام کاربری وارد سیستم می شوند را بسته و فقط استفاده از کلید را اجبار کنید. شما حتی در محیط های ویندوزی هم می توانید این کلید ها را ایجاد و از برنامه putty برای برقرار ارتباط با کلید هم استفاده کنید که در این مطلب شما روش استفاده از این برنامه را آموزش خواهید دید .

ایجاد کردن کلید در BSD

در مرحله اول هر فردی که قصد برقرار ارتباط را دارد باید برروی سیستم محلی خود کلید های RSA را ایجاد کند، برنامه ای به نام ssh-keygen به همراه برنامه ssh برروی سیستم شما نصب می شود که توان ایجاد کردن کلید های عمومی و شخصی به اصطلاح key public وprivate key را دارد کافیست که با کاربر خود این کلید را ایجاد کنید، به این نکته توجه کنید که کلید های خود را با یک رمزی که به خاطر دارید و سخت است از نظر امنیتی حمایت کنید این بخش passphrase نام دارد که در زمان ایجاد کردن یک کلید از شما پرسیده می شود. در شکل زیر خروجی حاصل از اجرای برنامه ssh – keygen را مشاهده می کنید:



```
admin@FreeBSD:~ % ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/home/admin/.ssh/id_rsa):
Created directory '/home/admin/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/admin/.ssh/id_rsa.
Your public key has been saved in /home/admin/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:nd+zvIWi5oh3JHQHItUDiLUandwMbIO4KQINq4026w0 admin@FreeBSD
The key's randomart image is:
+---[RSA 2048]----+
|.o . =oooo       |
|..o oo**. +      |
|o  o..=oo. o     |
|= o  o  .....    |
|=o   .  .S.o.    |
|...      . .. . .|
| E        o  o + .|
|  +     ...o. o + |
|.o .   ...+o    +. |
+----[SHA256]-----+
admin@FreeBSD:~ %
```

ایجاد کردن کلید

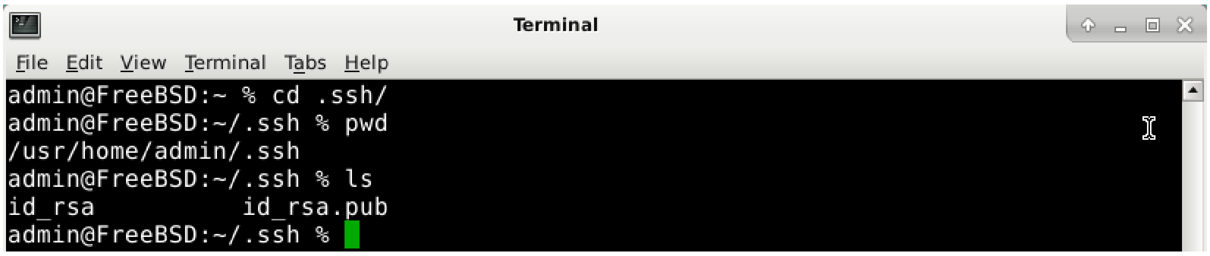
برای ایجاد کردن کلید از نوع RSA شما باید بعد از فرمان برنامه یا همان ssh-keygen از سویچ t– و مقدار RSA استفاده کنید. در بخش زیر شما باید رمز حمایت کنند کلید خود را در دو بار پشت سره هم وارد کنید:

```
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
```

اگر از passphrase استفاده کنید هر زمانی که قصد استفاده از کلید را داشته باشید باید این رمز را وارد کنید، اگر از-ssh agentو ssh-add استفاده کنید فقط یکبار کافیست که این کار را انجام دهید.

بعد از اجرا کردن این فرمان در شاخه Home کاربری که فرمان را با آن اجرا کرده اید شاخه ای ایجاد می شود به نام .ssh که یک شاخه مخفی شده است و در داخل آن کلید های public و private ذخیره می شود، نام کلید public در این شاخه .id_rsa.pubو نام کلید private در این شاخه id_rsa است.

در شکل زیر فایلهای مود در شاخه .sshd را مشاهده می کنید.

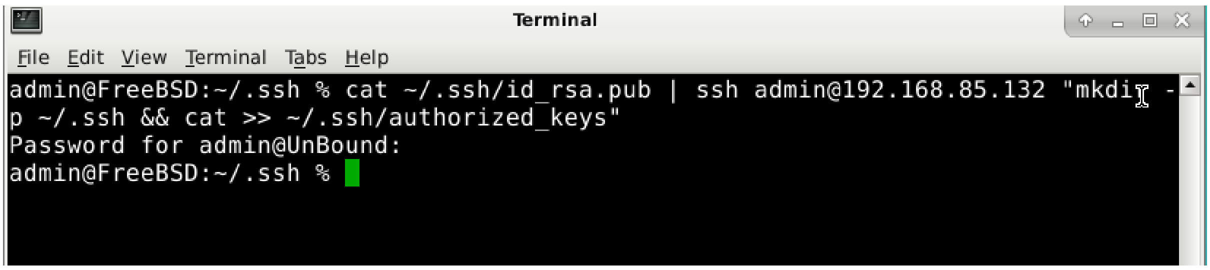


نمایش کلید های ایجاد شده

شما باید از کلید private خود محافظ کنید و کلید public را برای برقرار ارتباط با شما و یا برقرار کردن ارتباط با سرور ssh برروی سرور قرار دهید و یا در صورتی که دوست داشته باشید که دیگرا مطالب خود را با استفاده از این روش رمزنگاری برای شما ارسال کنند کافیست که کلید عمومی یا همان public را در اختیار آنها قرار دهید و یا حتی روی سایت خود به اشتراک گذارید. کلید private در این بخش نقش مهم و اساسی دارد و هر شخصی که به این کلید دسترسی پیدا کند و تواند به همه داده های رمز شده ای که به سمت شما ارسال می شد به صورت غیر رمز شده دسترسی پیدا کنید.

در مرحله بعدی در سمت سرور فایلی وجود دارد به نام authorized_keys در زیر شاخه نام کاربری که قصد دارید با نام آن وارد سیستم شوید که در شاخه .sshd قرار دارد. برای اینکار شما می توانید از فرمان ssh به صورت زیر استفاده کنید:

```
#cat ~/.ssh/id_rsa.pub | ssh admin@192.168.85.132"mkdir -p ~/.ssh && cat >> ~/.ssh/authoriz
ed_keys
```



همانطوری که مشاهده کردید به دلیل اینکه من برروی کلید خود رمزی را قرار داده بودم قبل از استفاده کردن از این کلید از من رمز عبور را درخواست کرد.

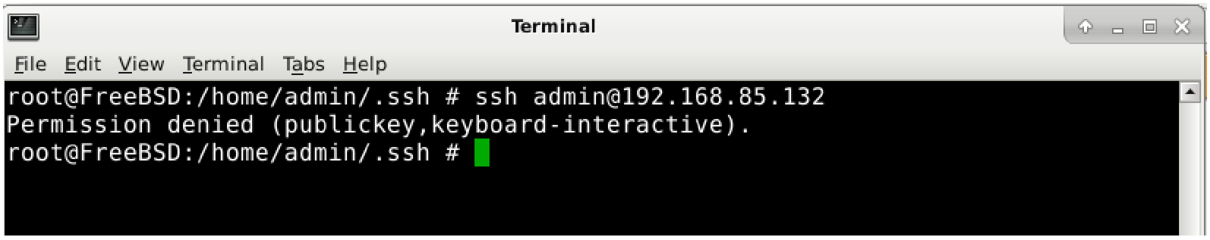
حال اگر قصد داشته باشید که همه کاربران از کلید استفاده کنند باید خط زیر را در فایل sshd.conf علامت # را از ابتدای خط زیر برداشته.

```
PasswordAuthentication no
```

در مرحله بعد هم استفاده کردن از pam را با استفاده از خط زیر و تبدیل کردن به no این قابلیت را غیر فعال کنید

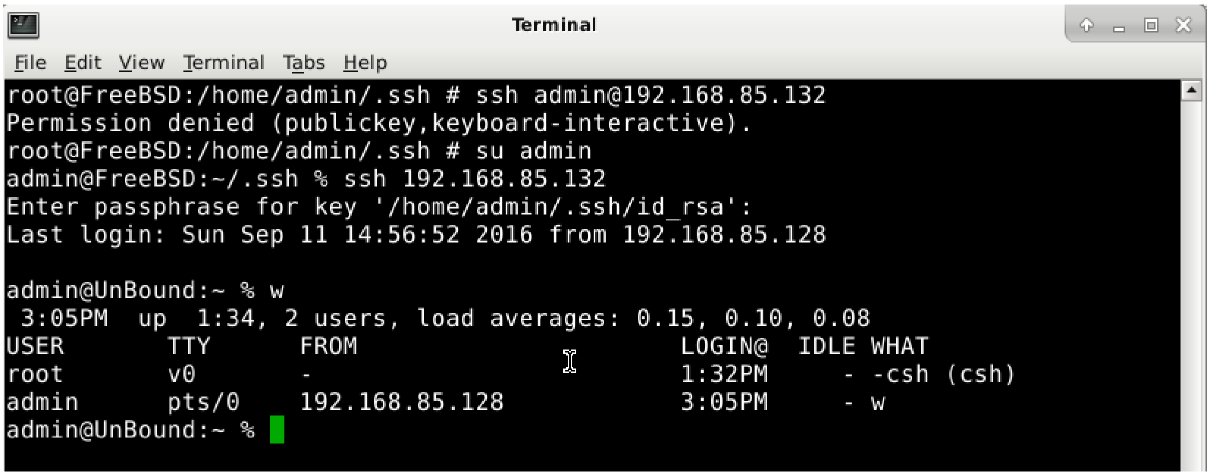
```
UsePAM no
```

حال شما اگر قصد داشته باشید به این سرور متصل شوید با خطای زیر مواجه می شوید:



مسدود کردن دسترسی از طریق رمزعبور

حال با استفاده از su وارد بخش کاربر admin می شوید و دوباره تلاش می کنید:



اتصال از طریق کلید

همانطوری که ملاحظه کردید می توانید با کلید به سرور متصل شوید.

استفاده از برنامه ssh-agent

برنامه ssh-agent در حقیقت یک راه انداز یا همان launch سایر برنامه هاست و شما می توانید با این برنامه محیط های Shellو یا windows ها دیگری را هم باز کنید. برای اینکار کافیست که از نام یک برنامه shell مثل csh بعد از این برنامه استفاده کنید تا برنامه مورد نظر شما باز شود و در مرحله بعدی کلید خود را یکبار با فرمان ssh-add کلید خود را در حافظه ramبارگذاری کنید تا دیگر نیازی به وارد کردن هر بار رمز کلید خود نداشته باشید، این بخش در شکل زیر نمایش داده شده است:

```
Terminal
File  Edit  View  Terminal  Tabs  Help
admin@FreeBSD:~/.ssh % ssh-agent csh
admin@FreeBSD:~/.ssh % ssh-add
Enter passphrase for /home/admin/.ssh/id_rsa:
Identity added: /home/admin/.ssh/id_rsa (/home/admin/.ssh/id_rsa)
admin@FreeBSD:~/.ssh % ssh 192.168.85.132
Last login: Sun Sep 11 15:05:26 2016 from 192.168.85.128

admin@UnBound:~ % w
 3:09PM  up  1:38, 2 users, load averages: 0.14, 0.12, 0.08
USER       TTY      FROM                          LOGIN@  IDLE WHAT
root       v0       -                             1:32PM     4 -csh (csh)
admin      pts/0    192.168.85.128                3:09PM     - w
admin@UnBound:~ % exit
logout
Connection to 192.168.85.132 closed.
admin@FreeBSD:~/.ssh % ssh 192.168.85.132
Last login: Sun Sep 11 15:09:50 2016 from 192.168.85.128

admin@UnBound:~ % exit
logout
Connection to 192.168.85.132 closed.
admin@FreeBSD:~/.ssh %
```

نمایش استفاده از ssh-agent

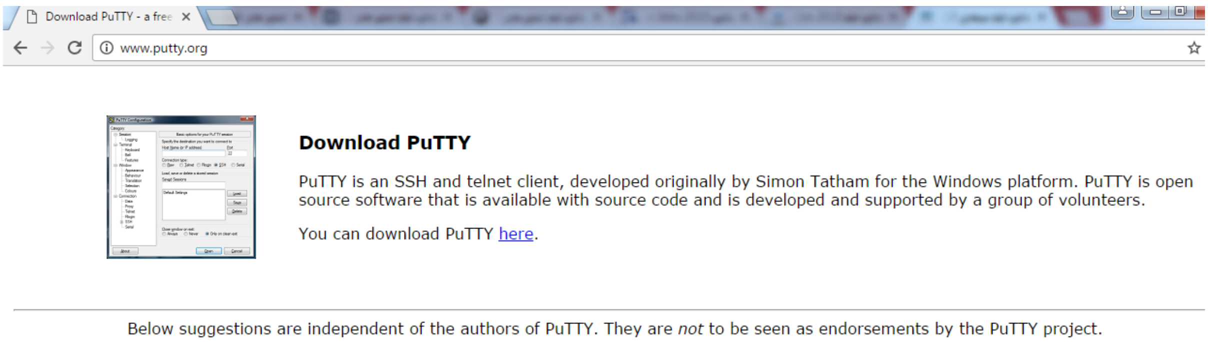
همانطوری که مشاهده می کنید برای هر بار برقرار ارتباط دیگر نیازی به وارد کردن رمز تنظیم شده برروی کلید خود ندارید.

روش اتصال به سرور SSH با putty و کلید عمومی

اگر شما از ویندوز برای انجام کارهای روزمره خود استفاده می کنید فرمان ssh وجود ندارد، نگران نباشید برنامه های برای برقرار اتصال با سرور ssh در ویندوز است که معروفترین آنها برنامه putty است که شما می توانید از آن به صورت رایگان استفاده کنید، این برنامه علاوه بر پروتکل ssh و telnet قابلیت تولید کلیده ای public و private را هم برای اتصال با کلید با سرور ssh را دارند. در این مقاله در بخش اول با برقرار ارتباط و در بخش دوم با ساختن کلید و استفاده از کلید بیشتر اشنا می شوید.

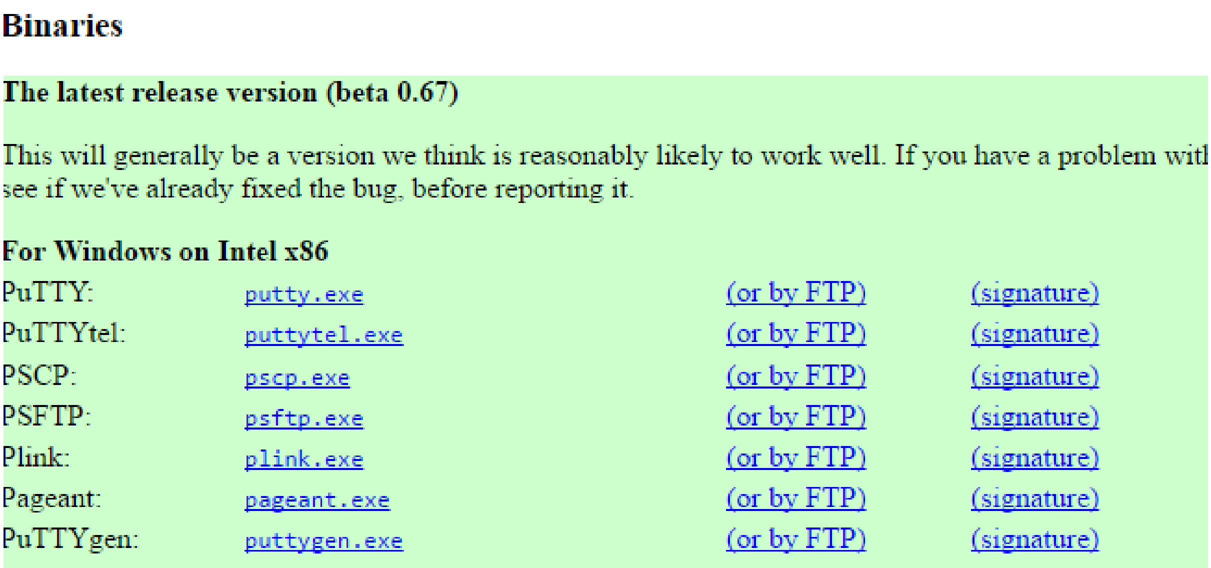
دریافت برنامه putty و اتصال به سرور با استفاده از نام کاربری و رمز عبور:

برنامه putty را می توانید از سایت http://www.putty.org دانلود کنید صفحه اصلی این سایت به شکل زیر است:



سایت putty

برای دانلود بر رروی here کلیک کنید تا وارد بخش دیگری از سایت شوید در بخش Binariesa شوید این بخش را در شکل زیر مشاهده می کنید:



Binaries

The latest release version (beta 0.67)

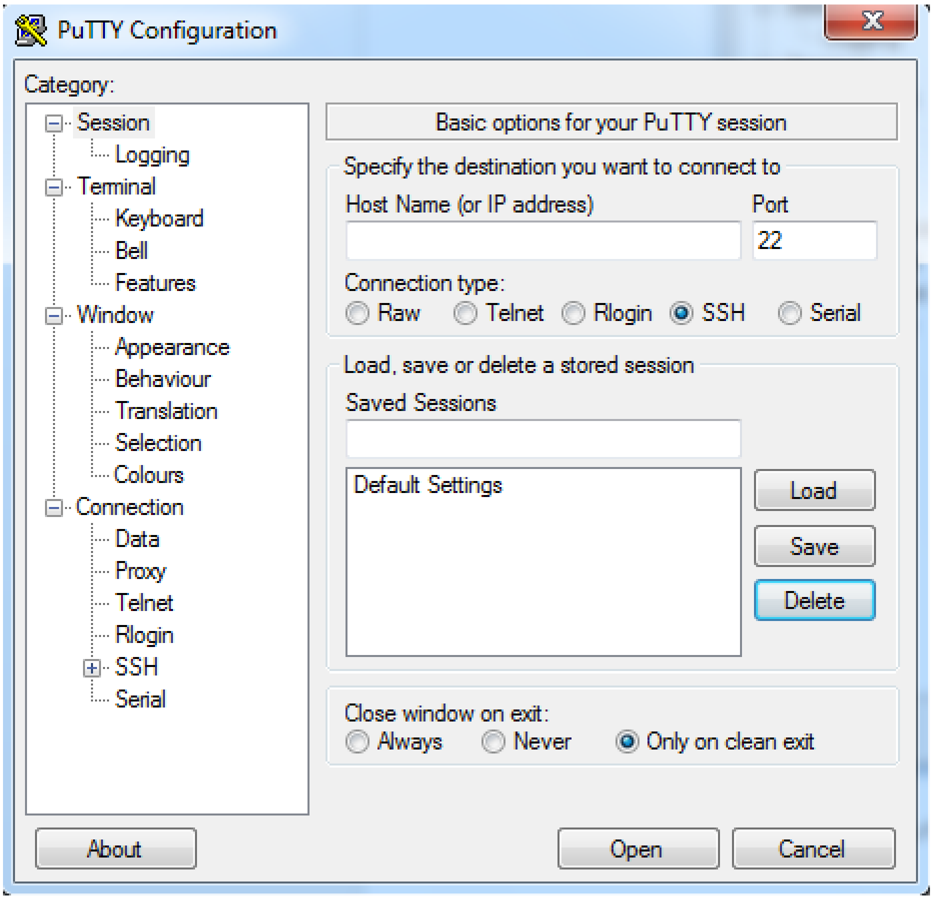
This will generally be a version we think is reasonably likely to work well. If you have a problem witl see if we've already fixed the bug, before reporting it.

For Windows on Intel x86

PuTTY:	putty.exe	(or by FTP)	(signature)
PuTTYtel:	puttytel.exe	(or by FTP)	(signature)
PSCP:	pscp.exe	(or by FTP)	(signature)
PSFTP:	psftp.exe	(or by FTP)	(signature)
Plink:	plink.exe	(or by FTP)	(signature)
Pageant:	pageant.exe	(or by FTP)	(signature)
PuTTYgen:	puttygen.exe	(or by FTP)	(signature)

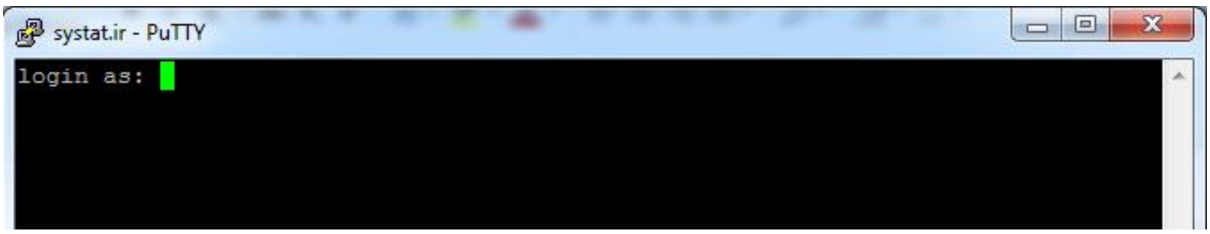
لیست فایلهای دانلودی

در قدم اول برای یک ارتباط ساده با سرور شما نیاز برنامه خود putty به نام putty.exe دارید که برای دانلود برروی نام ان کلیک کنید، حجم این برنامه 518 کیلو بایت بوده. بعد از دانلود کردن برنام را اجرا کنید تا محیطی به صورت زیر برای شما باز شود:



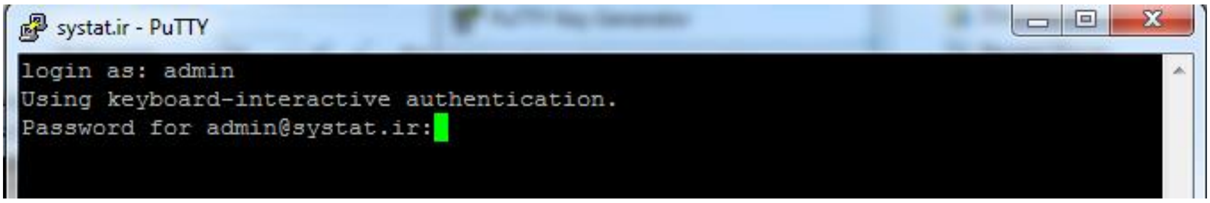
محیط برنامه putty

برای اتصال به سرور مورد نظر خود در بخش Host Name یا نام سرور خود را وارد کنید و یا آدرس IP را وارد کنید، در صورتی هم که سرور شما برروی شماره پورت خاصی فعال است می توانید در بخش Port شماره مورد نظر سرور خود را وارد کنید، در قسمت زیر هم می توانید نوع ارتباط خود را با سرور مشخص کنید که هم از telnet پشتیبانی می کند هم از ssh و شما می توانید از این برنامه برای برقرار ارتباط با پورت سریال دستگاه ها استفاده کنید. بعد از وارد کردن مشخصات مورد نظر شما برروی گزینه Open کلیک کنید تا به سرور متصل شوید به صورت شکل زیر:



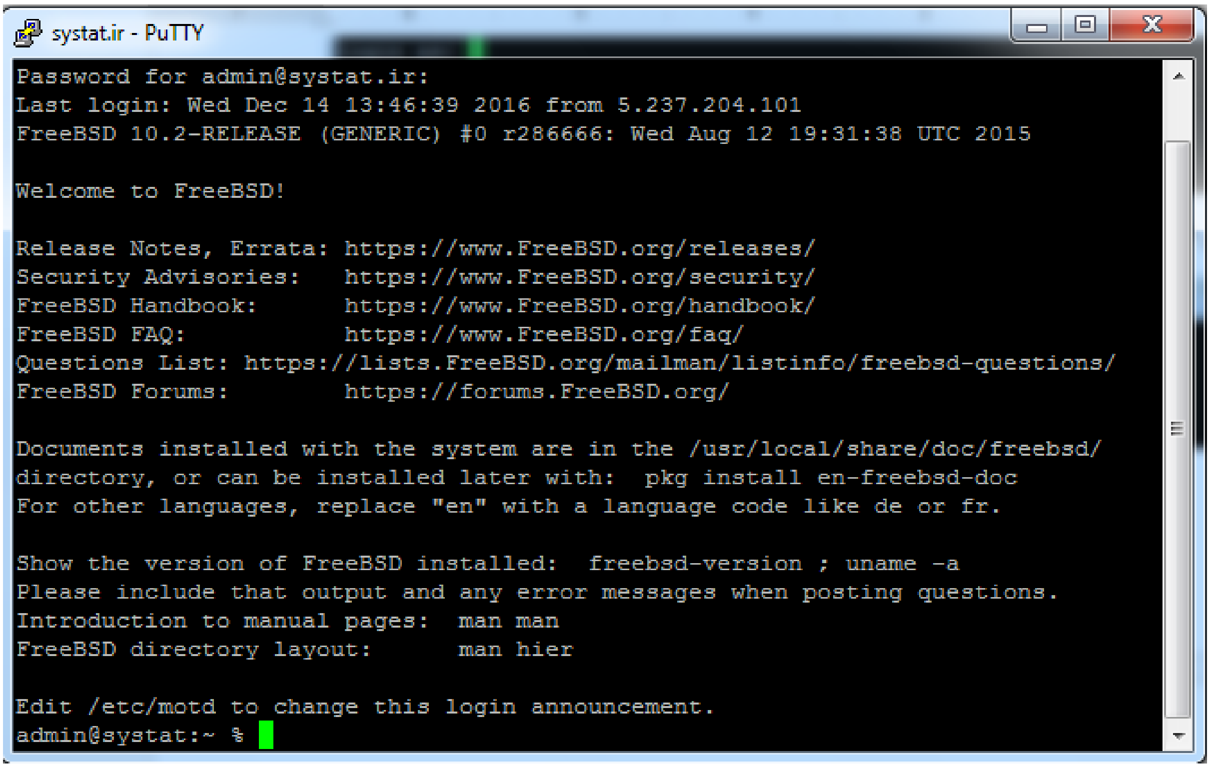
وارد کردن نام کاربری در برنامه putty برای وارد شدن به سیستم

بعد از وارد کردن نام کاربری و Enter کردن شما باید رمز عبور را وارد کنید به صورت شکل زیر:



بخش وارد کردن رمزعبور

بعد از وارد کردن رمز صحیح شما به صورت زیر به سرور متصل می شوید:

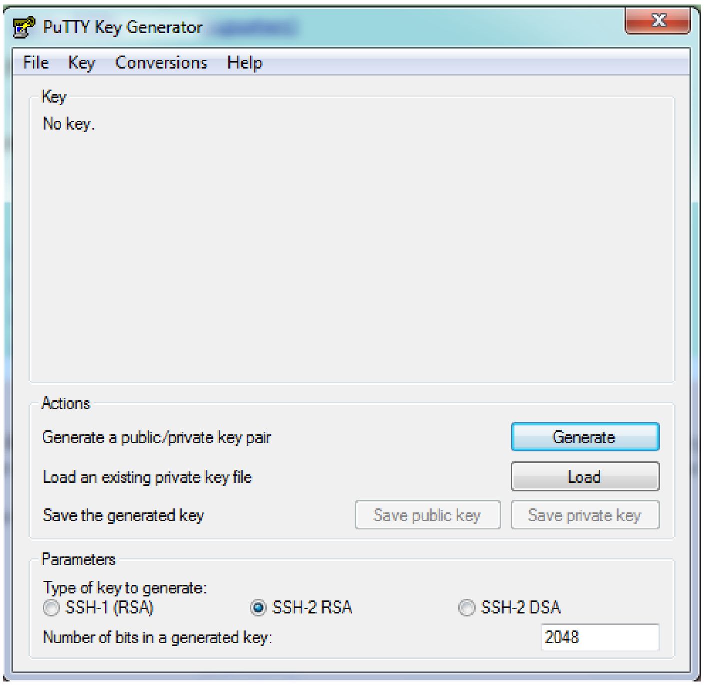


وارد شدن به سیستم FreeBSD با استفاده از ssh و برنامه putty در ویندوز

حال شما می توانید به راحتی فرمان های مورد نظر خود را وارد کنید. برای خارج شدن از این برنامه کافیست که از exit در شل استفاده کنید.

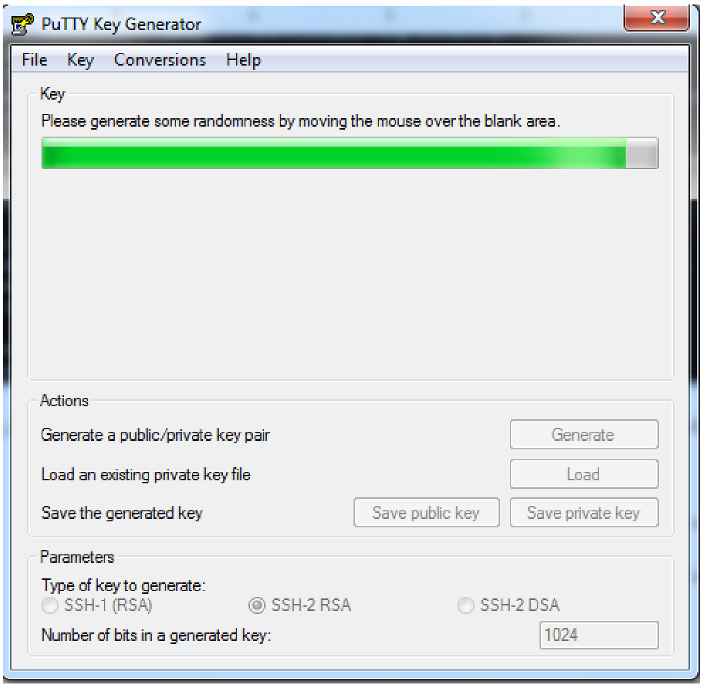
ایجاد کردن کلید RSA باPutty

پروژه putty این امکان را به شما می دهد که کلید های public و private را برای برقرار ارتباط با کلید به سرور SSH را ایجاد کنید. برای این کار شما به برنامه ای به نام puttygen.exe نیاز دارید، این برنامه را از سایت http://www.putty.org دانلود کنید و برنامه را اجرا کنید تا وارد محیط برنامه به صورت زیر شوید حجم این برنامه 138 کیلو بایت است:



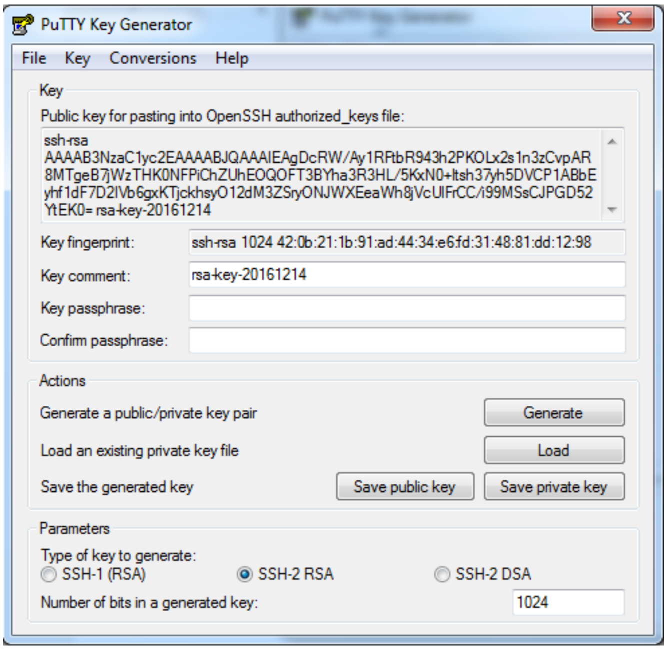
برنامه putty key gene

شما می توانید با استفاده از این برنامه کلید های RSA برای SSH1 و SSH2 ایجاد کنید و حتی کلید های ایجاد شده را در این برنام برای نمایش بارگذاری کنید، حتی شما می توانید طول کلید را هم انتخاب کنید، بعد از انتخاب کردن گزینه های مورد نظر خود برروی کلید Generate کلیک کنید تا کلید برای شما ایجاد شود به صورت نمایش داده شده در شکل زیر:



برنامه در حال تولید کردن کلید

بعد از اتمام این مرحله برنامه پیغام هایی به زیر برای شما نمایش می دهد:

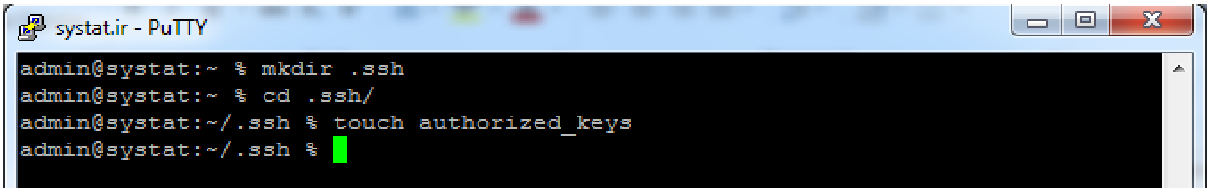


اتمام تولید کلید توسط برنامه Putty Key Generator

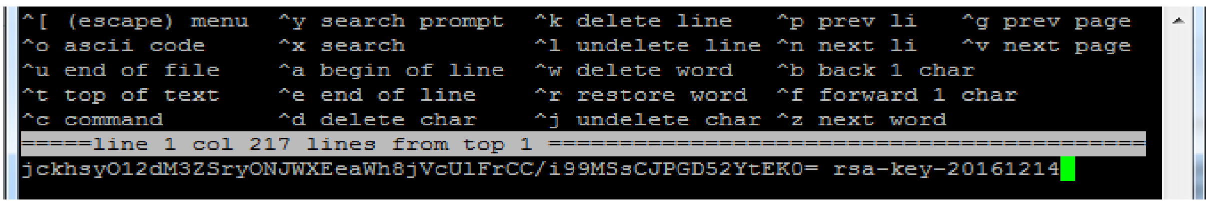
در کادر .. Public key for pasting شما باید این متن را در فایل authorized_keys موجود در شاخه .ssh کاربر مورد نظر خود کپی کنید. همچنین شما می توایند برای کلید خود رمز هم در بخش key passphrase تعریف می کنید.

برای ذخیره کردن کلیدهای public و private در بخش Save the generated key می توانید این دو کلید را save کنید. کلید private را در محلی مناسب قرار دهید و کلید public را می توانید در هرجایی که دوست دارید منتشر کنید، حتی برروی سایت خود.

حال یه سرور خود با putty وصل شوید و با کپی و paste کردن کلید public خود را به سرور منتقل کنید، به صورت زیر:

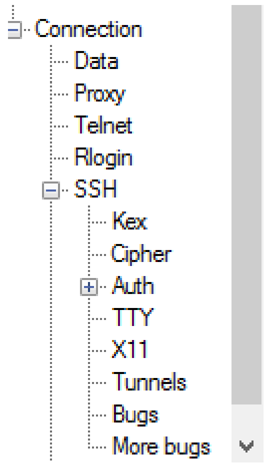


ایجاد کردن فایل مورد نظر برای انتقال کلید عمومی



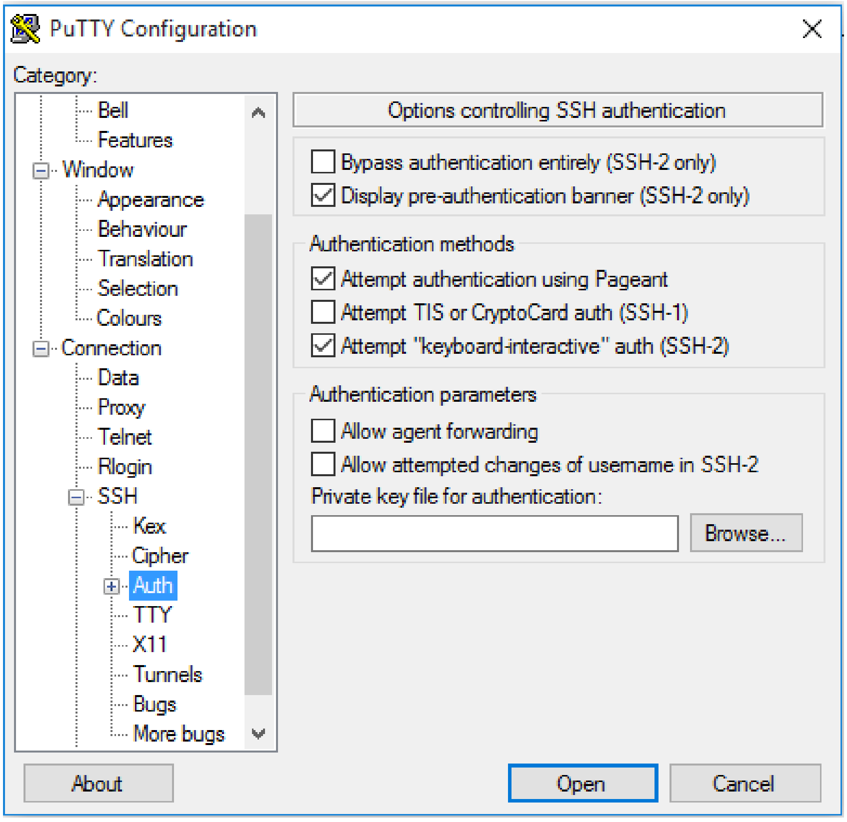
مراحل کپی کردن کلید عمومی

حال در حال باید با استفاده از برنامه putty کلید private خود را بارگذاری کنید، برای این کار وارد برنامه putty شوید و در بخش Connections وارد بخش ssh شوید و برروی بخش AUTH کلید کنید، این بخش رو در شکل زیر مشاهده می کنید:



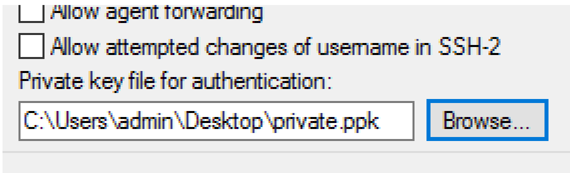
منوی دسترسی به AUTH در putty

صفحه ای به صورت زیر برای شما نمایش داده می شود:



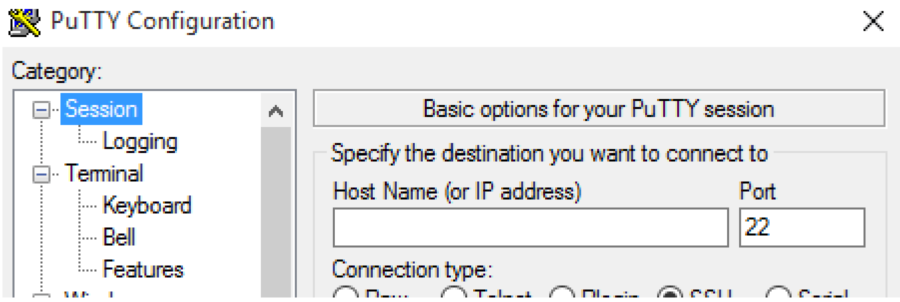
بخش AUTH در Putty

در بخش Private key file for authentications برروی گزینه Browse رفته کلیک کنید و فایل کلید شخصی خود را وارد کنید مثل شکل زیر:



وارد کردن کلید به برنامه putty

حال به بخش sessions رفته تا صحفه اولیه برنامه برای شما باز شود تا مشخصات سرور را وارد کنید، مثل شکل زیر:



بازگشت به بخشsessions

در صفحه login نام کاربر را وارد کنید که فایل کلید عمومی را در شاخه ssh. آن کپی کردین و دیگر نیازی به استفاده از رمز عبور نیست و با استفاده از کلید وارد سیستم شده اید.

برنامه syslog در FreeBSD

در این بخش با مباحث مربوط به فایلهای Log در FreeBSD آشنا می شوید، یکی از بحثهای مهم در سیستم عاملها توان آن سیستم آن در ثبت واقع است. یکی از وظایف هر مدیر سیستم آشنا با انواع و اقسام فایل های Log است. در این مدل از فایل ها اطلاعات سخت افزاری، نرم افزاری و خطا های برنامه ها ذخیره می شود، این اطلاعات برای امنیت سیستم شما بسیار مفید است. بیشتر برنامه ها و سرویس ها می توانند فایل Log تولید کرده و در محلی که شما معین می کنید آنها را ذخیره کنند، حتی می توانید حجم فایلها را مدیریت کنید و به اندازی مورد مظر شما رسید به اصطلاح rotate شده و به تعدادی که شما معین کرده اید ذخیره می شوند.

اکثریت این فایلها در زیر شاخه ای به نام var/log/ ذخیره می شود در شکل زیر شما فایلهای این زیر شاخه را مشاهده می کنید:

```
Terminal
File  Edit  View  Terminal  Tabs  Help
root@FreeBSD:/var/log # ls
ConsoleKit          dmesg.yesterday       sendmail.st.1
Xorg.0.log          lpd-errs              sendmail.st.2
Xorg.0.log.old      maillog               sendmail.st.3
aculog              maillog.0.bz2         sendmail.st.4
atop                maillog.1.bz2         sendmail.st.5
auth.log            messages              sendmail.st.6
bsdinstall_log      messages.0.bz2        setuid.today
cron                messages.1.bz2        setuid.yesterday
cups                messages.2.bz2        squid
debug.log           mount.today           userlog
debug.log.0.bz2     mount.yesterday       utx.lastlogin
debug.log.1.bz2     pf.today              utx.log
debug.log.2.bz2     ppp.log               utx.log.0
debug.log.3.bz2     security              xferlog
devd.log            sendmail.st
dmesg.today         sendmail.st.0
root@FreeBSD:/var/log #
```

فایلهای موجود در شاخه Log

در سیستم عامل FreeBSD از برنامه ای به نام syslogd برای مدیریت کردن Log ها استفاده می کند. به صورت پیش فرض این برنامه در زمان راه اندازی سیستم راه اندازی می شود و شما می توانید با استفاده از فایل rc.conf نقشهای مورد نظر خود را با استفاده از flags ها با خط syslog_flags در فایل rc.conf مدیریت کنید.

پیکربندی syslog در FreeBSD

برنامه syslog یک فایل پیکربندی در مسیر etc/ به نام syslog.conf دارد. در ادامه با این فایل بیشتر آشنا می شوید. در شکل زیر شما نمونه ای از این فایل را مشاهده می کنید:

```
#
#       Spaces ARE valid field separators in this file. However,
#       other *nix-like systems still insist on using tabs as field
#       separators. If you are sharing this file between systems, you
#       may want to use only tabs as field separators here.
#       Consult the syslog.conf(5) manpage.
*.err;kern.warning;auth.notice;mail.crit                /dev/console
*.notice;authpriv.none;kern.debug;lpr.info;mail.crit;news.err   /var/log/message
security.*                                      /var/log/security
auth.info;authpriv.info                         /var/log/auth.log
mail.info                                       /var/log/maillog
lpr.info                                        /var/log/lpd-errs
ftp.info                                        /var/log/xferlog
cron.*                                          /var/log/cron
!-devd
*.=debug                                        /var/log/debug.log
*.emerg                                         *
```

فایل برنامه syslog

همانطوری که مشاهده می کنید این فایل دارای دو بخش اصلی است، بخش اول نام برنامه ها و یا قابلیت های سیستم است که قصد دارید از آنها گزارش بگیرید که خودش به دو بخش level و facility تقسیم می شود که در ادامه با آن بیشتر آشنا می شوید و بخش دوم مسیری است که فایل گزارش مربوطه در آن ذخیره می شود.

بخش facility نام برنامه ها و دایمون هایی است که سیستم از آنها گزارش می گیرد مثل uth, authpriv, console, cron, daemon ftp, kern, lpr, mail, mark, news, ntp, security, syslog, user, uucp و local0تاlocal7 که برای برنامه هایی است که شما قصد دارید به این لیست اضافه کنید.

سطح یا همون Level میزان اطلاعات و سطح حساس دریافت اطلاعات را از برنامه مشخص می کند، که تربیت در آن مهم است که از سمت چپ به راست در لیست زیر از درجه اهمیت آن کاسته می شود:

emerg, crit, alert, err, warning, notice, info و debug.

در برخی از موارد مشاهده کرده اید که در زمان استفاده از سیستم پیغام های خطایی برای شما در صفحه اصلی نمایش داده می شود این پیغام ها به دلیل تنظیمات خط اول شما نمایش داده می شود. این خط شامل خط زیر است:

.err;kern.;auth.notice;authpriv.none;mail.crit   /dev/console

بسیار مشاهده شده است که در زمان بارگذاری ماژول در هسته شما پیغام های هسته را مشاهده می کنید این بخاطر بخش kern.*است در بخش دوم این خط مسیر ذخیره سازی فایل نمایش داده شده است که مسیر آن /dev/console است.

در زیر شاخه /var/log فایل جامعی است از گزارشات به نام messeges که اطلاعات کاملتری از سیستم شما در آن ذخیره میشود این فایل با استفاده از تنظیم کردن خط زیر ایجاد شده:

*.notice;authpriv.none;kern.debug;lpr.info;mail.crit;news.err   /var/log/messages

اگر شما به دنبال گزارشات امیتی مثل از سیستم خود هستید فایل security را چک کنید.

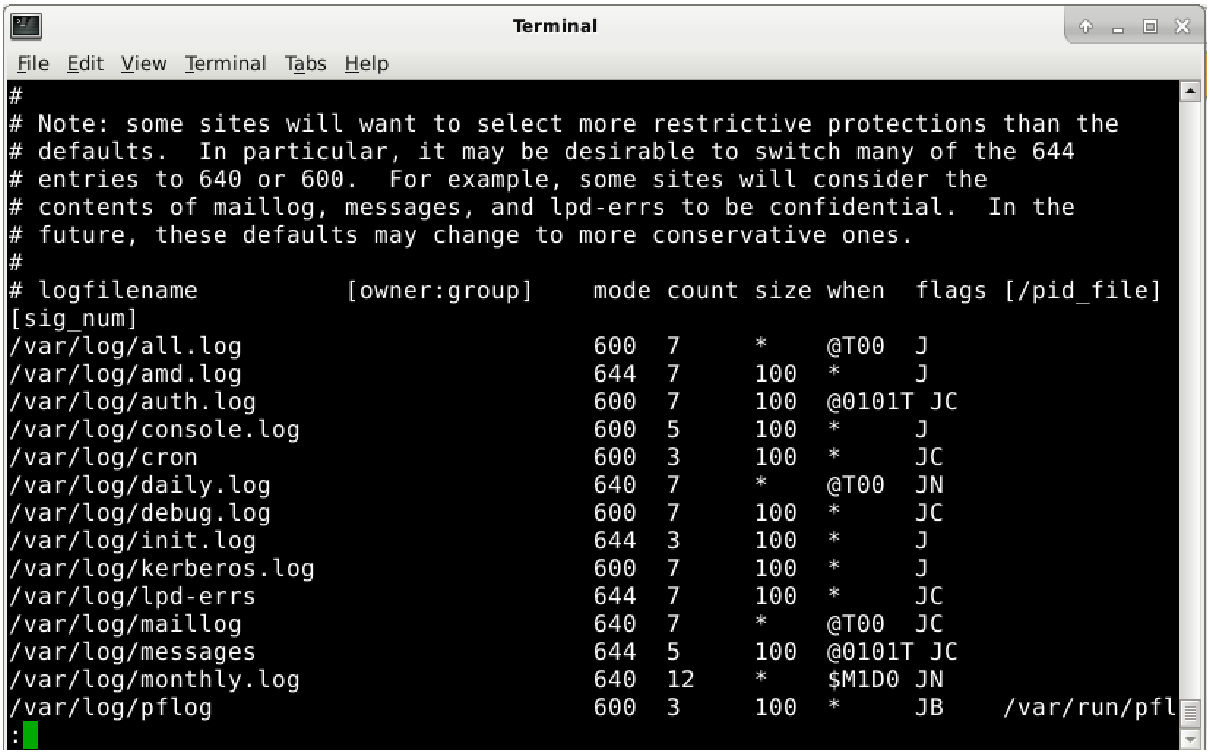
همانطوری که در خطوط بالا مشاهده کردید برای نوشتن level و facility می توانید از قواعد جدا کننده با ; استفاده کنید و اگر منظور شما همه چیز باشد از * استفاده کنید.

برنامه newsyslog در FreeBSD

فایلهای Log در بعضی از موارد به سرعت رشد کرده و بزرگ می شوند، این امر می تونه از فضای دیسک شما کاسته و کارایی سیستم شما را به خطر بندازد. در FreeBSD برنامه ای وجود دارد به نام Newsyslof که هدف اصلی آن مدیریت کردن فایل های log است. این برنامه دارای قابلیتهایی است که می تواند به صورت خودکار فایلها را بر حسب زمان یا حجم به فایل های کوچکتر تقسیم کند و یا حتی آنها را فشرده کند تا در میزان استفاده از فضا هم صرفه جویی شود. هر برنامه ای مثل syslog می تواند فایل Log ایجاد کند، این برنامه Newsyslog است که از طریق برنامه Cron راه اندازی می شود و اعمال خواسته شده را انجام می دهد. به صورت پیش فرض این برنامه هر یک ساعت یکبار اجرا می شود که البته این مقدار در فایل پیرکبندی این برنامه قابل تغییر است.

فایل پیکربندی newsyslog.conf

این برنامه شامل یک فایل پیکربندی است به نام newsyslog.conf در مسیر /etc ، این فایل شامل خطوطی است که هر خط معرف یک فایل log است که در ابتدای این خط نوشته می شود و در ادامه اگر شما قصد داشته باشید که دسترسی خاصی گروه یا کاربری بدهید می توانید در بخش بعدی آنرا ذکر کنید. در ادامه نوع سطح دسترسی را به فایلهایی که ایجاد شده است را می توانید مشخص کنید(برای اطلاعات بیشتر در مورد سطح دسترسی به مقاله آن مراجعه کنید). بخش از این فایل در شکل زیر نمایش داده شده:



```
Terminal
File Edit View Terminal Tabs Help
#
# Note: some sites will want to select more restrictive protections than the
# defaults.  In particular, it may be desirable to switch many of the 644
# entries to 640 or 600.  For example, some sites will consider the
# contents of maillog, messages, and lpd-errs to be confidential.  In the
# future, these defaults may change to more conservative ones.
#
# logfilename          [owner:group]    mode count size when  flags [/pid_file]
[sig_num]
/var/log/all.log                        600  7     *    @T00  J
/var/log/amd.log                        644  7     100  *     J
/var/log/auth.log                       600  7     100  @0101T JC
/var/log/console.log                    600  5     100  *     J
/var/log/cron                           600  3     100  *     JC
/var/log/daily.log                      640  7     *    @T00  JN
/var/log/debug.log                      600  7     100  *     JC
/var/log/init.log                       644  3     100  *     J
/var/log/kerberos.log                   600  7     100  *     J
/var/log/lpd-errs                       644  7     100  *     JC
/var/log/maillog                        640  7     *    @T00  JC
/var/log/messages                       644  5     100  @0101T JC
/var/log/monthly.log                    640  12    *    $M1D0 JN
/var/log/pflog                          600  3     100  *     JB    /var/run/pfl
:
```

بخش از فایل newsyslog

بخش count

شما می توانید تعداد فایلهای ایجاد شده را مدیریت کنید این کار را در بخش count مشخص کنید، برای مثال شما قصد دارید که 7 عدد از این فایلها را ذخیره کنید و بعد از اینکه فایل 88تم ایجاد شد فایل اولی به صورت خودکار پاک خواهد شد.

بخش size بخش when

شما می توانید به دو صورت برای ایجاد کردن فایلها محدودیت اعمال کنید، محدودیت بر اساس میزان حجم فایل و مدت زمانی خاص، برای هر فایل باید یکی از این روشها را انتخاب کنید، اگر هم دو محدودیت را اعمال کنید هر کدام که زودتر اتفاق بیفتد آن محمدویت اعمال می شود، برای مثال شما می خواهید که فایل شما در زمان خاصی ایجاد شود و فایل قبلی ذخیره شود و همچنین گفته اید که به 100 مگابایت برسد هم این عمل را انجام دهد، هر کدام که زودتر اتفاق رخ دهد آن عمل اعمال می شود. شما در بخش size میزان حجم فایل را بر حسب کیلوبایت مشخص می کنید، اگر در این بخش علامت * باشد بدین معناست که این بخش شما محدودیتی نیست.

در بخش زمان شما به دو روش می توانید زمان مورد نظر خود را وارد کنید

روش اول استفاده از استاندارد ISO 8601

این فرمت در بخش when با علامت @ شروع می شود. فرمت کلی این ساختار زمانی به صورت زیر است:

[[[[[*cc*]*yy*]*mm*]*dd*][T[*hh*[*mm*[*ss*]]]]]

بخش اول یا همان CC مربوط به قرن می شود که می تواند عددی بین 00 تا 99 در آن وارد کنید این بخش در بسیاری از موارد نوشته نمی شود.

بخش دوم یا همان yy است که در این بخش باید سال مورد نظر خود را وارد کنید

بخش mm که مربوط میشه به عدد ماه مورد نظر شما و بخش dd هم مربوطه میشه به روزی که مورد نظر شماست.

نکته : اگر شما بخش بالا یا همان بخش Date رو وارد کنید سیستم به صورت خودکار از زمان جاری شما استفاده می کند.

بخش دوم از این فرمت زمانی بعد از حرف T قرار میگیرد که این بخش خودش به 3 بخش دو عددی تقسیم میشه.

بخش hh مربوط میشود به ساعت مورد نظر شما.

بخش mm مربوط می شود به دقیق مورد نظر شما.

بخش ss هم مربوط می شود به ثانیه مورد نظر شما.

فرمت بالا در فایل newsyslog با علامت @ شروع می شود. و تنظیم کردن آن راحت نیست.

روش اول استفاده از حالت:Day Week Month Time

شما در این بخش می توانید به راحتی با استفاده از سه حرف D W M زمان های مورد نظر خود را بنویسید. قالب اصلی این فرمت به صورت زیر است:

[D*hh*], [W*w*[D*hh*]], and [M*dd*[D*hh*]],

در این فرمت زمان اختیار نیمه شب در نظر گرفته شده است. در زیر رنج هر بخش توضیح داده شده است:

در بخش HH زمان را مشخص می کند و عددی بین 0 تا 23 می تواند در این بخش مورد استفاده قرار گیرد.

در بخش w شامل روزهای هفته است که اعداد آن بین 0 تا 6 است و اولین روز هفته یکشنبه یا Sundayاست.

بخش dd روز های ماه است که عددی است بین 1 تا 31 و اگر می خواهید اولین روز هر ماه باش از حرف L و برای آخرین روز ماه از l استفاده کنید.

در زیر شما با چند مثال از این قالب های زمانی آشنا می شوید:

@T00	فایلها هر شب ساعت 12 rotate می شود.	**$D0**
@T23	فایلها هر روز ساعت 23 rotate می شود .	**$D23**
	فایلها هر هفته در روز یکشنبه در ساعت 23 rotate می شود.	**$W0D23**
	فایلها هر هفته در روز جمعه و در ساعت 16 rotate می شود.	**$W5D16**
@01T00	فایلها در اولین رو هر ماه و در ساعت 23 rotate می شود.	**$M1D0**
@05T06	فایلها در روز 5 هر ماه ساعت 6 rotate می شود.	**$M5D6**

بخش :Flags

شما می توانید در این بخش یکی یا چند Flag را در این بخش برای فایل خود اضافه کنید که هر Flag کار خاصی انجام می دهد و شما در ادامه با معروف ترین آنها آشنا می شوید:

Bاین Flag فایل مورد نظر شما را در قالب باینری ذخیره می کند.

Cاستفاده از این Flag باعث می شود که اگر فایل گزارش شما وجود نداشت برای شما ایجاد کند.

Jاین Flag باعث می شود که برنامه newsyslog با استفاده از BZIP فایلهای شما را فشرده کند و در مصرف فضا صرفه جویی کند.

Xاین Flag باعث می شود که برنامه newsyslog با استفاده از XZ فایلهای شما را فشرده کند و در مصرف فضا صرفه جویی کند.

Zاین Flag باعث می شود که برنامه newsyslog با استفاده از Gzip فایلهای شما را فشرده کند و در مصرف فضا صرفه جویی کند.

سرور Log در FreeBSD

خواندن فایلهای Log هر سیستم به صورت جداگانه کاری بسیار وقت گیر است، در FreeBSD قابلیتی وجود دارد به نام log serverکه از دو برنامه syslog و newsyslog استفاده می کند و این امکان را برای شما مهیا می کند که از یک سیستم مرکزی تمام گزاشات مورد نظر خودت رو بخوانید و از قابلیت های دو برنامه ذکر شده به صورت کامل استفاده کنید.

قبل از شروع به کار کردن و پیکربندی سرور و کاینت در این سیستم شما باید به دو نکته زیر توجه کنید:

1. برنامه syslog برای دریافت بسته های اطلاعاتی خود در شبکه از شماره پورت 514 در حالت UDP استفاده می کند و شما قبل از راه اندازی و پیاده سازی این سیستم از باز بودن و در دسترس بودن پورت مورد نظر در سمت سرور خود مطین شوید و چک کنید که فایروال سیستم شما دسترسی به این پورت را محدود نکرده باشد.

2. برنامه syslog به سرور DNS وابسته است و برای ارسال و دیافت بسته های شبکه از سرویس DNS استفاده می کند اگر در شبکه خود DNS دارید آنرا با نام های مورد نظر خود به درستی پیکربندی کنید یا برروی دو یا هر چند سیستم که در این طراحی قرار میگیرد در فایل /etc/hots تنظیمات خود را انجام دهید.

پیکربندی سرور Log

این بخش شامل دو قسمت است که در بخش اول شما کاینت را مشخص می کنید و در بخش دوم سرور log را برروی شبکه تنظیم می کنید.

بخش اول اضافه کردن کلاینت در سرور:

برای شروع باید فایل /etc/syslog.conf را ویرایش کنید، در این فایل در یک خط خالی شما باید نام کلاینیت را که قصد دارید گزارشات Log را از ان دریافت کنید با + شروع کنید و نام را ذکر کنید، در بخش بعدی هم باید سطح log را مشخص کنید و مسیر ذخیره شدن فایل را تعیین کنید:

```
+client.mabedini.com
*.*     /var/log/logclient.log
```

راه اندازی سرور:

حال بعد از اضافه کردن کلاینتها نوبت به پیکربندی سرور است، برای انجام دادن این عمل شما باید مقادیری را به فایل /etc/rc.confبه صورت زیر اضافه کنید:

```
syslogd_enable="YES"
```

```
syslogd_flags="-a logclient.example.com -v -v"
```

در خط اول راه اندازی خودکار در زمان boot را فعال می کنید و در خط دوم syslogd را با سویچ a راه اندازی میکند و نام کلاینت را در این بخش مشخص می کنید و v– v– اطلاعات کامل تری را برای شما نمایش می دهد.

نکته :

شما در مقابل سویچ a می توانید مقادیر دیگر را هم ذکر کنید به شرح زیر:

شما می توانید از آدرس IP استفاده کنید و بعد از آدرس IP و Subnetmask بعد از : می توانید شماره پورت UDP سرور خود را هم مشخص کنید.

شما می توانید از کاراکتر * هم استفاده کنید و نام چندین کلاینت را در یک خط در بخشی که مشترک هستند ار مشخص کنید به صورت mabedini.com.* این خط به این معناست که بخش * هر نامی که شما قرار دهید می تواند باشد.

در بخش بعدی شما باید فایل log مورد نظر و ذکر شده در فایل syslog.conf را ایجاد کنید برای این کار از فرمان زیر استفاده کنید:

```
# touch /var/log/logclient.log
```

حال زمان راه اندزی مجدد سرویس syslogd با استفاه از فرمان service است و در بخش بعدی هم برای اطمینان از راه اندازی درست سرویس از فرمان pgrep استفاده کنید، در ادامه فرمان های مورد نیاز این بخش را مشاهده می کنید:

```
# service syslogd restart
# pgrep syslog
```

پیکربندی :Client

سیستم کالاینت وظیفه ارسال کردن فایلهای گزارش به سمت سرور را دارد و یک نسخه هم به صورت local ذخیره می کند. در بخش کلانیت هم شما نیاز دارید که خطوط زیر را در فایل /etc/rc.conf اضافه کنید:

```
syslogd_enable="YES"
syslogd_flags="-s -v -v"
```

خط اول راه اندازی سرویس را در زمان boot سیستم بر عهده دارد و در خط اون هم سویچ s باعث فعال شدن ارسال گزاشات سمت سرور به صورت می شود، در قدم بعدی شما باید در فایل /etc/syslog.conf خطی را اضافه کنید برای ارسال کردن گزارشات به سمت سرور در بخش زیر این خط را مشاهده می کنید که همه سطح های گزارشات را به سمت سرور ارسال کنید که نام سرور با @ شروع می شود.

```
*.*             @logserv.mabedini.com
```

حال سرویس syslog را به صورت زیر دوباره راه اندازی کنید:

```
# service syslogd restart
```

برای چک کردن و ایجاد کردن یک گزارش تستی از فرمان logger استفاده کنید این برنامه پیغام شما را در فایل /var/log/messagesذخیره می کند و روش اجرا این فرمان را د شکل زیر مشاهده می کنید:

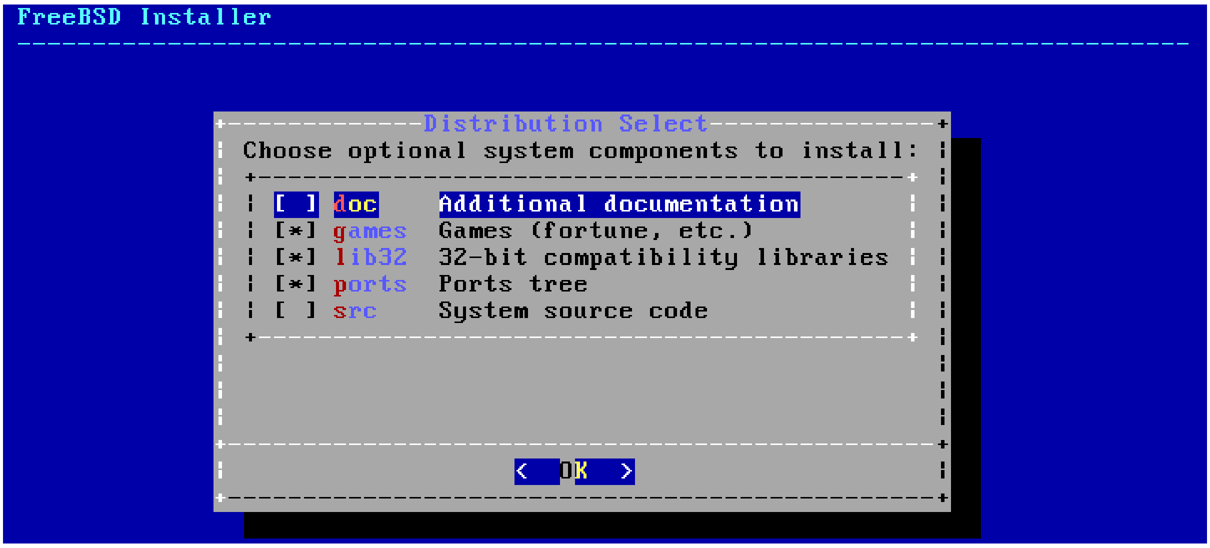
```
# logger "Test message from client"
```

پیکربندی هسته در FreeBSD

با گسترش روز افزون استفاده از ماژول ها و افزایش سرعت سیستم و پشتیبانی از همه سخت افزار های موجود در هسته FreeBSDکمتر کاربران این سیستم عامل نیازی به کامپایل کردن هسته را دارند چونکه این کار بسیار زمانبر هم هست. ولی در موارد بعضی از قابلیت های تخصیص فقط در هسته کامپال شده و شما نیاز به دانش کامپایل کردن دارید. در این مقاله راه های نصب سورس FreeBSD و بعد کامپال کردن هسته آموزش داده می شود.

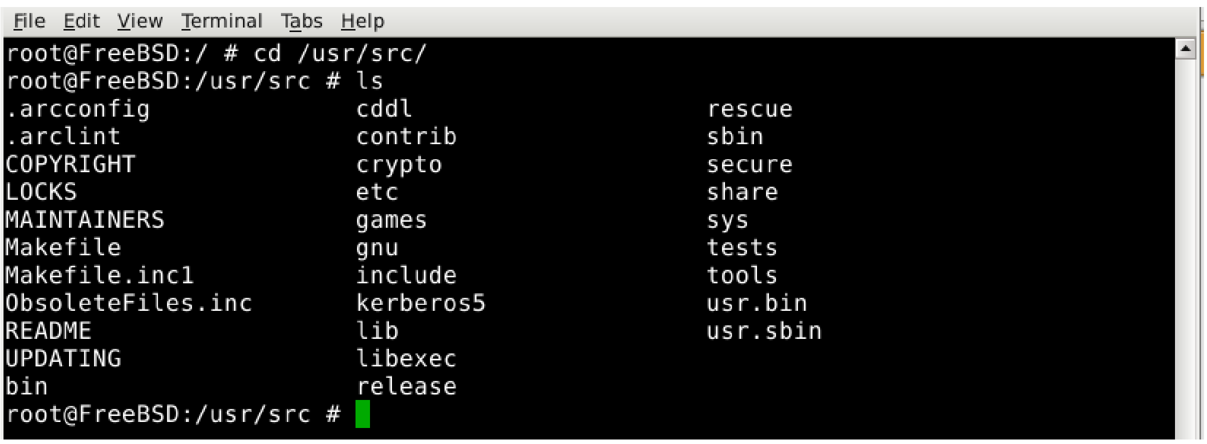
نصب سورس کد FreeBSD

شما به دو روش می توانید این کار را انجام دهید، راه اول در زمان نصب سیستم عامل است و راه دوم به صورت دستی این کار را انجام دهید. در زمان نصب در بخش Distributions Select شما با گزینه های نمایش داده در شکل زیر مواجه می شود:



منوی نصب کردن سورس FreeBSD در زمان نصب

اگر شما بهش SRC را انتخاب کنید سورس سیستم عامل برای شما به صورت خودکار نصب می شود، شاخه این بخش در /usr/srcاست اگر وارد این شاخه شده اید و زی شاخه هایی در آن قرار داشت به این معناست که سورس نصب شده است، در شکل زیر خروجی فرمان ls را در این زیر شاخه مشاهده می کنید:



این شاخه دارای زیر شاخه های زیادی است که یکی از معروفترین آنها شاخه sys است که در زیر شاخه آن فایلهای مربوط به ساختاره های سخت افزاری به نام های amd64 i386 و ... قرار دارد، به تناسب نوع سیستم عاملی که نصب کرده اید مثل amd64که به FreeBSD 64 بیتی است وارد آن شاخه شوید ، فایلهای موجود در این شاخه را مشاهده می کنید:

```
Terminal
File  Edit  View  Terminal  Tabs  Help
root@FreeBSD:/usr/src/sys # cd amd64/
root@FreeBSD:/usr/src/sys/amd64 # ls
Makefile        compile         include         pci
acpica          conf            linux           vmm
amd64           ia32            linux32
root@FreeBSD:/usr/src/sys/amd64 # cd conf/
root@FreeBSD:/usr/src/sys/amd64/conf # ls
DEFAULTS        GENERIC         GENERIC.hints   Makefile        NOTES
root@FreeBSD:/usr/src/sys/amd64/conf #
```

شاخه اصلی محل هسته

در این زیر شاخه شاخه است به نام conf که که فایل متنی اصلی در آن قرار دارد به نام GENERIC که یک فایل متنی است و شما با ویرایشگر متنی می توانید آنرا باز کنید و بخشهایی که می خواهید به هسته اضافه و یا از هسته کم کرد را در این فایل می توانید آنرا اضافه و کم کنید، چند خط اولیه این فایل را در شکل زیر مشاهده می کنید:

```
Terminal
File  Edit  View  Terminal  Tabs  Help
#
# $FreeBSD: releng/10.3/sys/amd64/conf/GENERIC 286132 2015-07-31 15:25:07Z gjb $

cpu             HAMMER
ident           GENERIC

makeoptions     DEBUG=-g                # Build kernel with gdb(1) debug symbols
makeoptions     WITH_CTF=1              # Run ctfconvert(1) for DTrace support

options         SCHED_ULE               # ULE scheduler
options         PREEMPTION              # Enable kernel thread preemption
options         INET                    # InterNETworking
options         INET6                   # IPv6 communications protocols
options         TCP_OFFLOAD             # TCP offload
options         SCTP                    # Stream Control Transmission Protocol
options         FFS                     # Berkeley Fast Filesystem
options         SOFTUPDATES             # Enable FFS soft updates support
options         UFS_ACL                 # Support for access control lists
options         UFS_DIRHASH             # Improve performance on big directories
options         UFS_GJOURNAL            # Enable gjournal-based UFS journaling
options         QUOTA                   # Enable disk quotas for UFS
options         MD_ROOT                 # MD is a potential root device
options         NFSCL                   # New Network Filesystem Client
 ESC
```

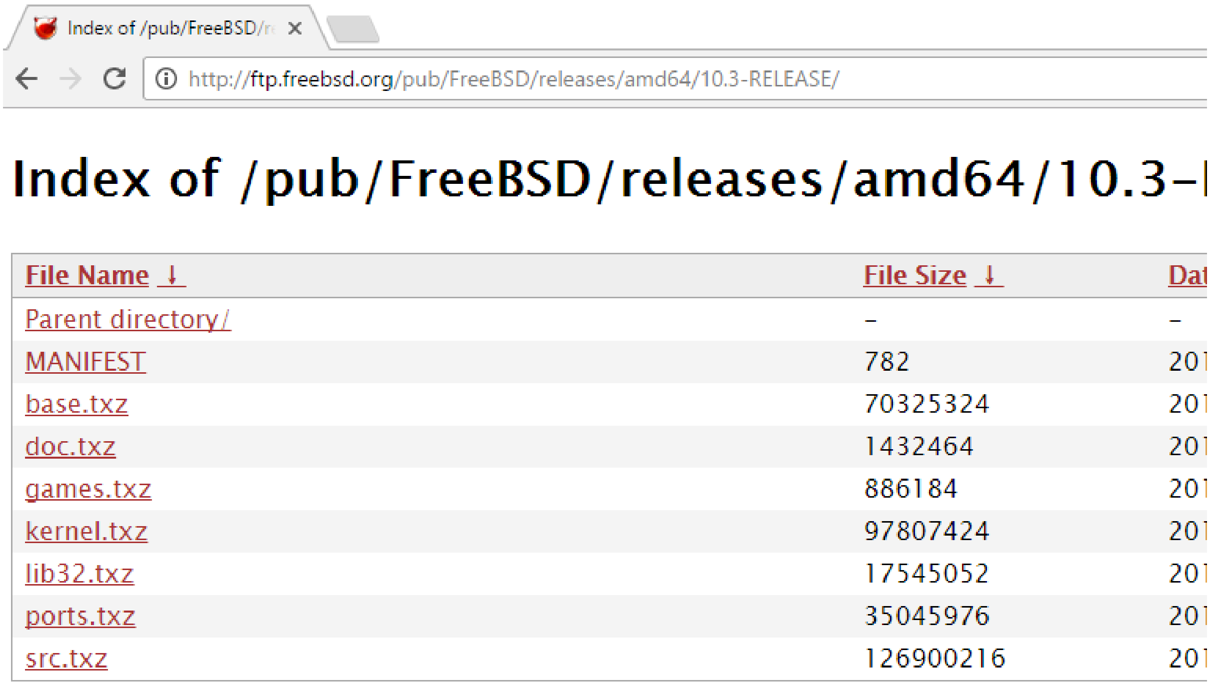
فایل اصلی هسته

قواعد این فایل بسیار ساده است ، شما می توانید در خط نمایش داده شده در بالا شکل ساختار هسته ای را که این فایل ایجاد می کند را مشاهده کنید، هر خط شامل یک قابلیت است، برای اضافه کردن هر قابلیت از کلمه کلیدی options استفاده کنید

و در روبروی آن نام قابلیت را اضافه کنید، برای مثال برای اضافه کردن NFS Client به هسته باید کلمه NSFCL را بعد از optionsاضافه کنید.

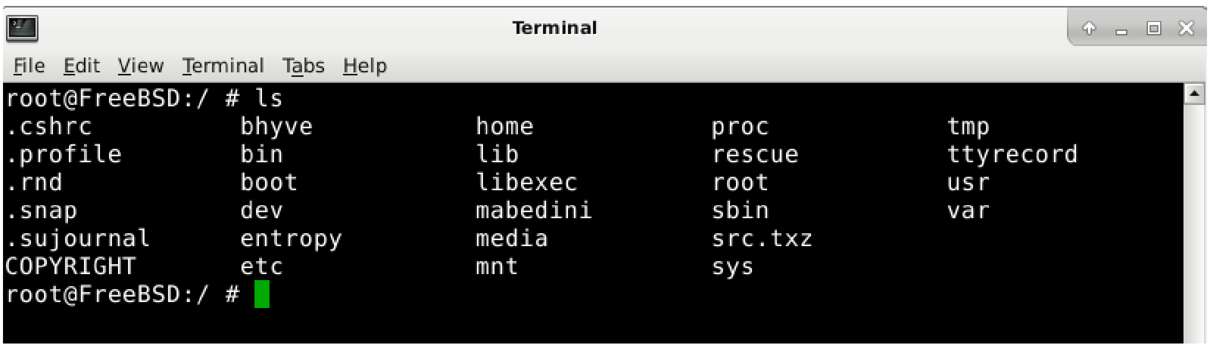
نصب کردن فایل سورس به صورت دستی:

اگر در زمان نصب src را نصب نکردید باید از این راه برای نصب استفاده کنید و ابتدا فایل src.txz را دانلود کنید. ابتدا باید وارد ftp سرور سایت freebsd.org شوید، آدرس http://ftp.freebsd.org/pub/FreeBSD/releasesیکی از آدرسهای این بخش است، حال نوع ساختار را انتخاب کنید و بخش بعدی ورژن سیستم عاملی را که نصب کرده اید انتخاب کنید صفحه ای بصورت زیر برای شما باز می شود:



دانلود کردن سورس از طریق سایت

حال می توانید فایل src.txz را دانلود کنید و در شاخه FreeBSD / خود قرار دهید،

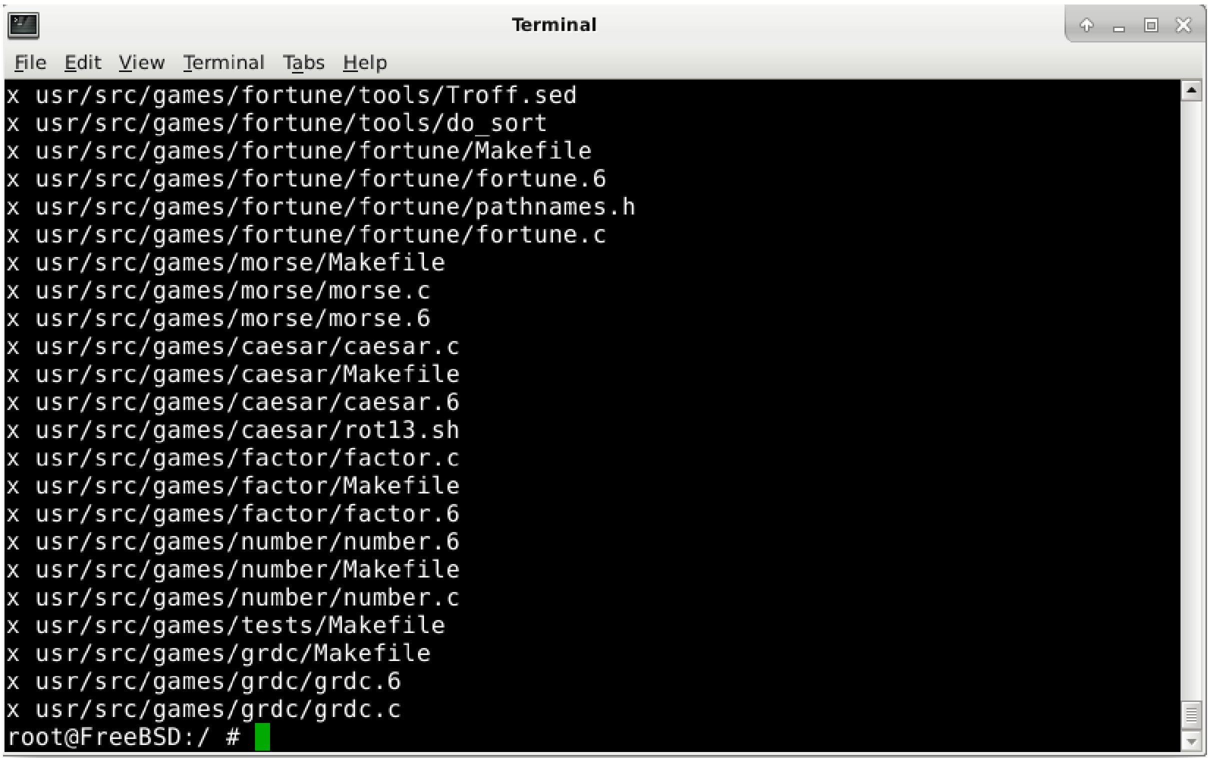


نمایش فایلهای دانلود شده

حال با استفاده از فرمان tar و سویچ های zxvf این فایل را از حالت فشرده خارج کنید تا در زیر شاخه /usr/src شاخه ها و فایلهای سورس برای شما ایجاد شود.

حتما فایل سورس را از شاخه / extract کنید.

خروجی و پایان این فرمان برای شما در شکل زیر نمایش داده شده است:



Extract کردن فایلهای هسته

نصب هسته جدید:

این کار با دو فرمان انجام شده و بعد شما برای راه اندازی سیستم خود با هسته جدید نیازی به راه اندازی مجدد سیستم دارید، نصب هسته در FreeBSD تنها عملی است که نیازی به restart کردن کل سیستم دارد. در ادامه مراحل این کار ذکر می شود:

شما هم می توانید برروی فایل اصلی هسته خود تغییرات را اعمال کنید و هم پیشنهاد می شود که از فایل GENERIC یک کپی گرفته و تغییرات را در آن اعمال کنید، در فرمان های زیر ابتدا وارد شاخه مورد نظر شوید و در بخش دوم با فرمان cp یکی کپی از فایل اصلی ایجاد می کنید:

```
# cd /usr/src/sys/amd64/conf
# cp GENERIC MYKERNEL
```

بعد از اعمال تغییرات حال شما به سر شاخه usr/src/ رفته و اولین فرمان را به صورت زیر اجرا کنید:

```
# make buildkernel KERNCONF=MYKERNEL
```

به این نکته توجه کنید که KERNCONF=MYKERNEL در حقیقت نام همان فایلی است که بخش قبلی از آن کپی گرفته اید، اگر هم شما برروی فایل اصلی تغییرات را ایجاد کرده اید نیازی به این بخش در اجرا فرمان ندارید.

بعد از اتمام این فرمان اگر شما با خطایی مواجه نشوید بخش کامپال کردن هسته جدید تمام شده و شما حال نیاز به نصب کردن آن دارید، برای انجام این کار کافیست که فرمان زیر را اجرا کنید:

```
# make installkernel KERNCONF=MYKERNEL
```

اگر این بخش هم بدون خطا به اتمام برسد هسته جدید شما نصب شده و فقط شما نیاز دارید که سیستم را restart کنید تا FreeBSDبا هسته جدید شما راه اندازی شود.

فایروال IPFW در FreeBSD

در سیستم عامل FreeBSD سه نوع فایروال با امکانات خواص برای هر کدام وجود دارد که یکی از بهترین و قدیمیترین آنها فایروال ipfw است که از ورژنهای 4 و 6 آدرس ip پشتیبانی می کند و دارای قابلیتهای گزارشگیری، Natکردن، مدیریت کردن ترافیک با استفاده از dummynet ، قابلیت bridge و در نهایت با استفاه از قابلیت ipstealth به شما ان امکان را می دهد که فایروال خود را با تغییر ندادن ttl از دید کاربران شبکه ای که از فرمان tracert استفاده می کنند مخفی کنید.

علاوه بر این فایروال دو فایروال دیگری هم به نام های pf و ipfilter که هر کدام در بخشهای مورد نظر توضیح داده خواهد شد ولی در این بخش هدف ما آموزش در مورد استفاه از ipfw است.

راه اندازی IPFW

به دو شیوه شما می توانید ipfw را راه اندازی کنید، روش اول با اعمال تغییرات در هسته FreeBSD و اضافه کردن آن به هسته. روش دوم بارگذاری ماژول مربوطه در هسته. روش اول زمان بر بوده و شما نیاز است که مقاله پیکربندی هسته FreeBSD را مطالعه کنید.

برای راه اندازی کردن IPFW در هسته باید خطوط زیر را به فایل GENERIC اضافه کنید و هسته را کمپایل کردن و با هسته جدید سیستم را راه اندازی کنید:

```
options    IPFIREWALL                          # enables IPFW
options    IPFIREWALL_VERBOSE                  # enables logging for rules with log keyword
options    IPFIREWALL_VERBOSE_LIMIT=5          # limits number of logged packets per-entry
options    IPDIVERT                            # enables NAT
options    IPFIREWALL_DEFAULT_TO_ACCEPT       # sets default policy to pass what is not
                                               explicitly denied
```

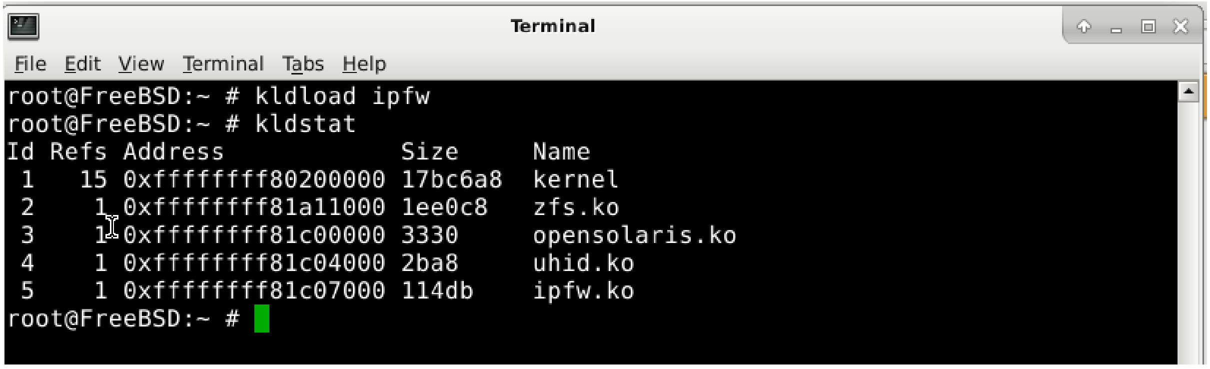
خط **IPFIREWALL**باعث فعال شدن ipfw می شود، خط **IPFIREWALL_VERBOSE**قابلیت log گیری از وضعیت رول های ipfw را در هسته فعال می کند شما می توانید با استفاده از خط **IPFIREWALL_VERBOSE_LIMIT=5**برای گزارشکری محدودیت اعمال کنید تا سیستم شما در اثر فایلهای زیاد log دچار از دسترس خارج شدن نشود.

در زمانی که شما به سیستم راه دور متصل هستید و فقط ارتباط ssh به سرور خود دارید و قصد راه اندازی ipfw را دارید بهتر است به این نکته مهم توجه کنید، این فایروال به صورت پیش فرض دارای رولی است که همه ترافیک ها را مسدود می کند مگر در حالتی که شما در زمان کامپایل کردن در هسته خط **IPFIREWALL_DEFAULT_TO_ACCEPT**را در فایل هسته خود قرار دهید و بعد عمل کامپایل کردن را انجام دهید.

برای فعال سازی قابلیت NAT و استفاده کردن از این قابلیت باید خط **IPDIVERT**را در هسته قرار دهید.

روش راه اندازی کردن از طریق ماژول:

شما با استفاده از فرمان kldload می توانید ماژول ipfw را در هسته سیستم خود بارگذاری کنید و نیازی به راه اندازی سیستم خود ندارید، روش استفاده از این فرمان در شکل زیر نمایش داده شده است:




```
root@FreeBSD:~ # kldload ipfw
root@FreeBSD:~ # kldstat
Id Refs Address            Size     Name
 1   15 0xffffffff80200000 17bc6a8  kernel
 2    1 0xffffffff81a11000 1ee0c8   zfs.ko
 3    1 0xffffffff81c00000 3330     opensolaris.ko
 4    1 0xffffffff81c04000 2ba8     uhid.ko
 5    1 0xffffffff81c07000 114db    ipfw.ko
root@FreeBSD:~ #
```

نمایش وضعیت load شدن ماژولها در هسته

در فرمان اول با استفاده از kldload ipfw ماژول مربوطه را بارگذاری کرده اید و با استفاده از فرمان kldstat شما وضعیت ماژول های بارگذاری شده در سیستم خود را مشاهده می کنید.

روش راه اندازی از طریق سیستم rc.conf

یکی دیگر از روشهای راه اندازی ipfw استفاده از فایل rc.conf و فرمان های rc.d است. برای راه اندازی ابتدایی در زمان bootسیستم خط زیر را در فایل rc.conf اضافه کنید:

```
firewall_enable="YES"
```

حال باید نوع فایروال خود را مشخص کنید، در فایل rc.conf نوع فایروال را با اضافه کردن خط زیر مشخص کنید:

```
firewall_type="open"
```

این بخش دارای حالت های زیر است:

- حالت : Open که فایروال هیچ اقدامی انجام نمی دهد و همه ترافیک ها را از خود عبور می دهد، این بخش برای زمانی مناسب است که از راه دور به سیستم متصل می شوید و قصد دارید که در زمان مناسب رولهای خود را اضافه کنید.
- حالت : Client در این حالت تمام ترافیک های ارسال شده به سمت سیستمی که ipfw برروی آن راه اندازی شده مسدود می شود.
- حالت: Simple در این حالت کل شبکه شما مورد نظارت و حمایت فایروال قرار می گیرد، این حالت برای NAT و یا Gateway شبکه مفید است.
- حالت: Closed تمام ترافیک های ایجاد شده برروی همه کارتهای شبکه بجز کارت شبکه loopback مسدود می شود.

- حالت Workstations: سیستمی که ipfw برروی آن راه اندازی شده است در حالت StateFull مورد حمایت قرار می گیرد.
- نام فایل حاوی رول: شما می توانید رولهای مورد نظر خود را در فایلی متنی با قواعد خاص این برنامه بنویسید و آنرا برروی دیسک خود ذخیره کنید و مسیر آنرا در این بخش ذکر کنید تا در زمان راه اندازی این رولهای شما باید که اعمال شود البته روش دیگری هم وجود دارد.

راه دیگری برای بارگذاری کردن رولهای انتخابی خود استفاده از firewall_script در فایل rc.conf است که شما می توانید در قالب یک فایل اسکریپتی رولهای خود را نوشته و بعد از این بخش نام فایل خود را ذکر کنید به صورت زیر این بخش فعال می شود:

```
firewall_script="/etc/ipfw.rules"
```

در حقیقت فایل etc/ipfw.rules فایل مورد نظر است.

برای فعال کردن حالت گزارشگیری باید خط زیر را در فایل rc.confاضافه کنید:

```
firewall_logging="YES"
```

برای محدود کردن تعداد لاگ های هر ارتباط باید فرمان زیر را اجرا کنید:

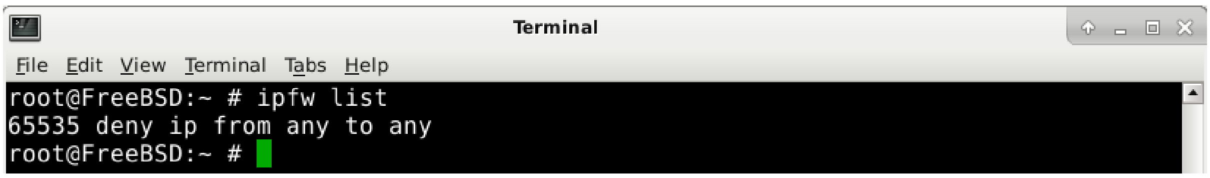
```
#sysctl net.inet.ip.fw.verbose_limit=5
```

یا خط net.inet.ip.fw.verbose_limit=5 را در خط فایل /etc/sysctl.conf اضافه کنید.

برای راه اندازی ipfw از فرمان service به صورت زیر استفاده کنید:

```
# service ipfw start
```

برای کنترل کردن ipfw فرمانی در سیستم عامل FreeBSD به همین نام وجود دارد که در بخش بعدی با آن بیشتر آشنا می شوید، برای تست کردن از صحت راه اندازی شدن ipfw شما می توانید از این فرمان با سویچ list استفاده کنید تا تمام رولها برای شما نمایش داده شود، خروجی این فرمان در شکل زیر برای شما نمایش داده شده است :



نمایش وضعیت رولها در IPFW

قواعد رول نویسی در IPFW

زمانی که یک بسته به فایروال ipfw بسته با هر رولی که توسط مدیر سیستم نوشته شده باشد از بالا به پایین چک می شود و اولین رولی که با مشخصات بسته دریافتی یکی باشد آن رول برروی بسته اعمال می شود مثلا اگر اجازه رد شدن از فایروال را نداشته باشد بسته درباقتی رد می شود به این قابلیت خاصیت First match Wins می گویند. بسته هایی که مشخصات آنها در هیچ یک از رولها نباشد به صورت خودکار با رول شماره 65535 که آخرین رول و رول پیش فرض است تعیین تکلیف می شوند، به صورت پیش فرض این رول به کلی همه بسته ها را deny می کند مگر شما این رول پیش فرض را تغییر داده باشید که در مقاله قبلی روش تغییر دادن آنرا توضیح داده شده.

نوشتن رول در ipfw دارای قواعدی است که در ادامه با آن آشنا میشوید، بعضی از این بخشها اجباری بوده و حتما باید در رول باشند و در بعضی از بخشها هم اجباری وجود ندارد و فقط برای شما کار خاصی را که توضیح داده می شود انجام می دهد.

شما می توانید رول های مورد نظر خود را در قالب یک فایل اسکریپت بنویسید تا هر زمانی که لازم بود آنرا اعمال، ویرایش کنید و در زمان راه اندازی مجدد سیستم تغییرات شما پاک نشود، رولهای شما در ipfw در لحظه اعمال می شود و تا زمان راه اندازی سیستم معتبر خواهد بود پس به این نکته توجه کنید، برای نوشتن این فایل دو قاعده کلی وجود دارد که قاعده اول در همه فایلهای پیکربندی وجود دارد و آن استفاده از # در ابتدای خطوطی است که توضیحات در آن قرار دارد و هیچ تغییری برای شما اعمال نمی کند، قاعده جامع دوم هم این است که خطوطی که خالی هستند هم تغییرات را به همراه نخواهد داشت و بی تاثیر هستند.

در زیر یک نمونه کلی و جامع از یک رول در ipfw را مشاهده می کنید:

```
CMD RULE_NUMBER set SET_NUMBER ACTION log LOG_AMOUNT PROTO from SRC SRC_PORT to DS
T DST_PORT OPTIONS
```

بخش cmd

در حقیقت یک بخش اجباری است و باید در ابتدای هر رول وجود داشته باشد و با ipfw add شروع می شود.

بخش RULE_NUMBER

هر رول که نوشته می شود باید دارای یک عدد باشد که بین 1 تا 65534 است. این عدد در حقیقت ترتیب عددی رول ها بوده و باید به ترتیب آن دقت کنید به خاطر قاعده . First match Wins چند رول هم می توانند شامل یک عدد باشد در این حالت آن رولی که زودتر نوشته شده باشد رول اول است.

بخش set SET_NUMBER

این بخش اجباری نبوده و شما می توانید با استفاده از آن برای یک سری از رول های خود یک ست با عددی بین 0 تا 31 در نظر بگیرد. شما به راحتی می توانید این ست ها را فعال و غیرفعال کنید. این کار سرعت تغییرات شما را افزایش می دهد، اگر شما در بخش SET_NUMBER عددی وارد کنید به صورت خودکار عدد 0 اعمال می شود.

بخش ACTION

در این بخش عملی که برروی بسته دریافتی اعمال می شود را باید تعیین کنید. این بخش شامل دو کلید واژه اصل است allowکه با accept و pass یکی است و به معنی این است که بسته اجازه رد شدن دارد Deny. که با prop مترادف است و هر بسته ای که با آن مچ باشد اجازه عبور نخواهد داشت.

بخش log LOG_AMOUNT

در این بخش قسمت log برای هر رول را فعال می کنید که اطلاعات به syslog ارسال شده و با facility name خاص SECURITYذخیره می شود. گزارشگیری زمانی رخ می دهد که شما بخش تعداد log را مشخص کنید، اگر این بخش را مشخص نکرده باشید به صورت خودکار از net.inet.ip.fw.verbose_limit این مقدار را می گیرد. برای دوباره تنظیم کردن مجدد لاگ از فرمان ipfw reselog استفاده کنید.

بخش PROTO

این بخش جزو بخشهای اجباری بوده و نوع پروتکلی را که قصد اعمال محدودیت یا باز کردن آنرا دارید را مشخص می کنید برای دریافت لیست کاملی از پروتکلها به فایل /etc/protocols مراجعه کنید.

بخش from SRC

کلمه frome یک کلمه ای است که باید در این بخش قبل از مشخص کردن مبداء مورد نظر ذکر کنید، در بخش بعدی از این متن شما باید مبداء را مشخص کنید، چند کلمه رایج در این بخش وجود دارد به نام any یعنی هر آدرسی یا me یعنی همین سیستم شما که برروی آن فایروال را نصب کرده اید برای تعریف کردن آدرس ورژن 6 از me6 استفاده کنید. در این بخش شما می توانید آدس ip را هم به صورت مستقیم در این بخش وارد کنید، این حالت زمانی اتفاق می افتد که سیستم شما Gatewayدر شبکه است و می خواهید ارتباطات کامپیوترهای شبکه خود را محدود کنید، روش نوشتن آدرس به دو حالت در زمان نوشتن subnet mask حالت اول CIDR است مثل 24/192.168.1.100 و حالت دوم هم نوشتن عددی ان به صورت 192.168.1.100 255.255.255.0.

بخش SRC_PORT

در این بخش شما به صورت اختیاری می توانید شماره پورت سرویس مبداء بسته را مشخص کنید برای دریافت تمام پورت های مربوطه به هر سرویسی به فایل /ert/services مراجعه کنید.

بخش to DST

این بخش هم برای مصقد بسته تعیین کننده است که توضیحاتی مثل بخش from SRC است و هم می توانید از me و any استفاده کنید و یا آدرس ip را به روش های ذکر شده مشخص کنید.

بخش DST_PORT

در این بخش شما به صورت اختیاری می توانید شماره پورت سرویس مقصد بسته را مشخص کنید برای دریافت تمام پورت های مربوطه به هر سرویسی به فایل /ert/services مراجعه کنید.

بخش OPTIONS

در این بخش شما می توانید هر گزینه های keep-state و ... در این بخش قرار دهید ولی پرکاربردترین این بخشها دو گزینه in و out است که مسیر بسته ها را مشخص می کند و شما با استفاده از via هم می توانید کارت شبکه را مشخص کنید.

نمونه ای از ولها:

```
Ipfw add 10 allow all from any to any via lo0
```

در رول بالا شما به همه ترافیک هایی که از کارت شبکه loopback رد می شود امکان تردد می دهد.

```
Ipfw 110 allow tcp from any to any 21 in
Ipfw 120 allow tcp from any to any 21 out
```

در دو رول بالا شما کلا استفاده از پورت 21 که سرویس ftp است را باز می کنید.

```
Ipfw add 00310 deny icmp from any to any in via
```

در رول بالا همه ترافیک های مربوطه به سرویس ping یا همان پروتکل icmp را مسدود می کنید.

فرمان ipfw

فایروال ipfw دارای یک رابط فرمان است به نام ipfw که علاوه بر استفاده در اضافه کردن رول برای بخش های مدیریتی هم مورد استفاده قرار می گیرد و سویچهای مفیدی دارد که وضعیت رول های سرور شما را نمایش می دهد و یا حتی یک رول خاص و یا تمام رولها را پاک می کند، در ادامه با روش استفاده از این فرمان در قابل دریافت گزارش بیشتر آشنا می شوید.

سویچlist

اگر شما بعد از فرمان ipfw از این سویچ استفاده کنید همه رولهای راه اندازی شده شما را نمایش می دهد.

سویچ –t list

این سویچ علاوه بر خروجی فرمان بالا time stamp هر رول را نیز برای شما نمایش میدهد.

سویچ a list–

این سویچ عددی را در کنار هر رول نمایش می دهد که شامل شمارش بسته هایی است که با این رول هماهنگ بوده ، به اصطلاح شمارنده ای است از هر رول.

سویچ zero

با استفاده از این سویچ تمام شمارنده های همه رولها صفر می شود.

پاک کردن یک رول خواص:

برای پاک کردن یک رول خواص کافیست که از فرمان ipfw و سویچ delete و شماره رول استفاده کنید(در این زمینه در سایت و مقاله اصلی چند فایل gif تصویری وجود دارد)

tablesدر IPFW

یکی از امکانات IPFW این است که می توانید با استفاده از جدول برروی هم زمان چند آدرس ip یک عمل خاص را انجام دهید، در قدم اول شما باید یک جدول را ایجاد کنید برای این کار باید از فرمان ipfw و table استفاده کنید، بعد از آن باید نام جدول را ذکر کنید و با کلمه Add و آدرس ip جدول مورد نظر را ایجاد کنید، به عنوان مثال جدول شماره 1 را با اضافه کردن آدرس 192.168.1.1 بصورت زیر اضافه کنید:

```
#ipfw table 1 add 192.168.1.1
```

برای مشاهده کردن لیست همه جداول از فرمان زیر استفاده کنید:

```
#ipfw table all list
```

برای مشاهده کردن لیست های موجود در یک جدول فقط کافیست در فرمان بالا بجای all از نام جدول اضافه کنید.

برای پاک کردن یک جدول خاص فرمان زیر را اجرا کنید:

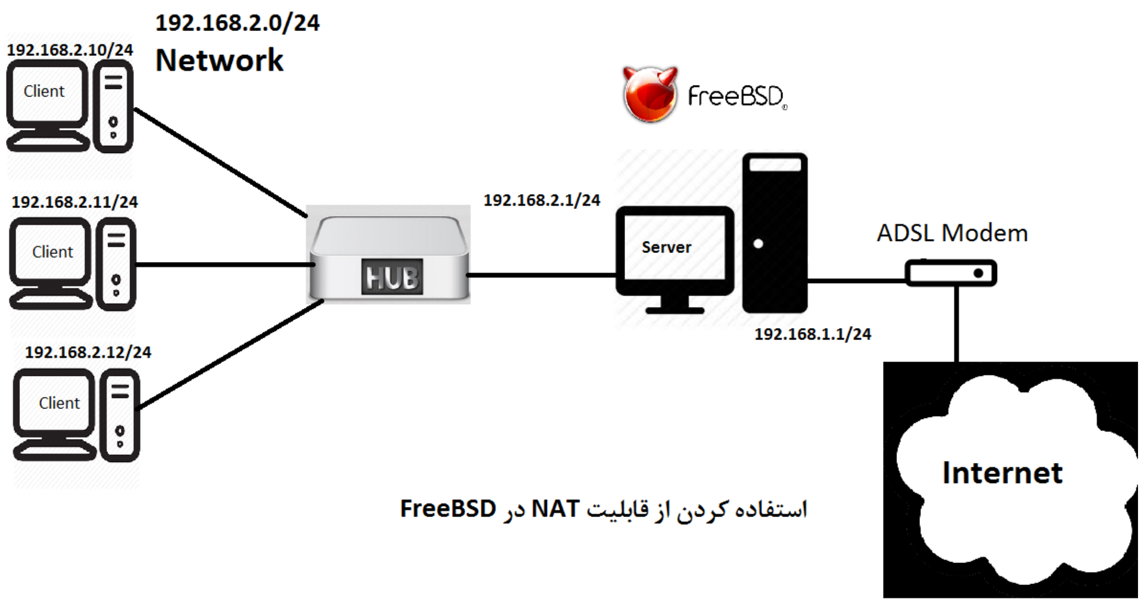
```
#ipfw table 1 flush
```

Natکردن ترافیک با IPFW در FreeBSD

یکی از راه حل های به اشتراک گذاشتن اینترنت به شبک های محل استفاده کردن از سرویس NAT است، البته این سرویس کارکردهای دیگری هم دارد که در بخشهای بعدی در مورد آنها بحث می شود. این سرویس برای راه اندازی شدن به فایروال ipfw وابسته است و شما باید ابتدا این فایروال را هم فعال کنید. برروی سیستم که شما قصد دارید این سرویس را راه اندازی کنید باید دو کارت شبکه داشته باشید، یک کارت شبکه که به شبکه محلی شما که قصد دارید اینترنت را در اختیار آنها قرار دهید متصل باشد و کارت شبکه دوم هم باید به اینترنت متصل باشد. طراحی آدرس IPشبکه ها انتخابی است.

طراحی شبکه برای nat

برای پیاده سازی سناروی شبکه nat شما باید قبل از شروع آدرس IP شبکه خود را مشخص کنید،شکل زیر را مشاهده کنید:



طراحی شماتیک از شبکه ای با قابلیت Nat

شبکه بالا دارای دو بخش است بخش اول شبکه محلی است که رنج آدرس های آن 192.168.2.0 است و سه عدد PC در آن قرار دارد، تمام این سیستم ها باید آدرس 192.168.2.1 را به عنوان Default Gateway تنظیم کنند. سرور FreeBSD دارای دو کارت شبکه است که یکی از آدرس های آن 192.168.2.1 است و آدرس دوم 192.168.1.1 است که به مودم ADSLوصل است که خود شبکه اینترنت متصل است. شبکه بالا یک طراحی ساده از یک شبکه است. از هاب برای متصل کردن سیستم ها در این شبکه استفاده شده است. در ادامه با راه اندازی کردن این مدل آشنا می شوید.

فعال کردن IPDIVERT و IPFIRWALL

شما به دو روش می توانید قابلیت های ذکر شده را در FreeBSD فعال کنید روش اول خطوط زیر را در هسته فعال کنید و هسته دوباره بارگذاری کنید، برای انجام دادن این روش مقاله پیکربندی هسته در FreeBSD را مطالعه کنید:

```
options  IPFIREWALL
options  IPDIVERT
```

روش بعدی بارگذاری کردن ماژول ipdivert در هسته با استفاده کردن از kldload به این نکته توجه کنید که ماژول ipfw هم به صورت خودکار در هسته بارگذاری می شود و رول پیش فرض آن بستن همه ترافیک هاست. در شکل زیر این عمل نمایش داده شده است:

```
Terminal
File  Edit  View  Terminal  Tabs  Help
root@FreeBSD:~ # kldstat
Id Refs Address             Size     Name
 1   13 0xffffffff80200000 17bc6a8   kernel
 2    1 0xffffffff81a11000 1ee0c8    zfs.ko
 3    1 0xffffffff81c00000 3330      opensolaris.ko
 4    1 0xffffffff81c04000 2ba8      uhid.ko
root@FreeBSD:~ # kldload ipdivert
root@FreeBSD:~ # kldstat
Id Refs Address             Size     Name
 1   17 0xffffffff80200000 17bc6a8   kernel
 2    1 0xffffffff81a11000 1ee0c8    zfs.ko
 3    1 0xffffffff81c00000 3330      opensolaris.ko
 4    1 0xffffffff81c04000 2ba8      uhid.ko
 6    1 0xffffffff81c07000 4cb6      ipdivert.ko
 7    1 0xffffffff81c0c000 114db     ipfw.ko
root@FreeBSD:~ # ipfw list
65535 deny ip from any to any
root@FreeBSD:~ #
```

خلاصه ای از راه اندازی IPFW در FreeBSD

در مرحله بعدی شما باید قابلیت Gate way را هم فعال کنید این کار به دو صورت انجام می شود در روش اول با استفاده از rc.conf که باید در این روش خط زیر را در این فایل قرار دهید:

```
# gateway_enable=YES
```

حال در زمان راه اندازی سیستم شما قابلیت Gate Way برای شما فعال می شود در حالت دوم هم شما می توانید با استفاده از فرمان sysctl این قابلیت را فعال کنید:

```
# sysctl net.inet.ip.forwarding=1
```

در مرحله بعدی قابلیت natd را فعال کنید . در این بخش هم می توانید از سیستم rc.conf استفاده کنید و یا فرمان natd را به صورت مستقیم اجرا کنید، برای فعال سازی rc.conf خطوط زیر را در این فایل اضافه کنید:

```
natd_enable="YES"         # enables NAT
natd_interface="em0"      # specify interface name of NIC attached to Internet
```

در خط اول سرویس natd را فعال کرده اید و در خط بعدی باید نام کارت شبکه ای را که به شبکه دیگر که اینترنت دارد را مشخص کنید که در این بخش em0 است و در هر سیستم می تواند متفاوت باشد. شما می توانید با استفاده از natd_flags سویچ های لازم برای این سرویس را مشخص کنید. حال با استفاده از فرمان زیر سرویس natd را فعال کنید:

```
#/etc/rc.d/natd start
```

در روش استفاده از خط فرمان شما از فرمان natd در شل می توانید با استفاده از سویچ های مختلف این سرویس را راه اندازی کنید، برای مثال برای فعال کردن قابلیت ارایه گزارش از log– یا l– استفاده کنید و بعد از فعال شدن قابلیت logفایلی به نام /var/log/alias.log ایجاد می شود وlog ها در آن ذخیره می شود، برای مشخص کردن کارت شبکه ای که با اینترنت متصل است از سویچ interface– یا n– استفاده کنید و در مقابل آن نام کارت شبکه را واردی کنید در زیر این فرمان نمایش داده شده است:

```
#natd -log -interface em0
```

در بخش آخر هم باید در فایروال ipfw دو فرمان زیر را وارد کنید:

```
#ipfw add divert natd all from any to any via em0
#ipfw add 500 allow all from any to any
```

تنظیمان بخش Client

در بخش کلاینتی اگر سیستم عامل شما FreeBSD باشد بعد از تنظیم کردن آدرس ip باید با استفاده از فرمان route به صورت زیر آدرس سرور FreeBSD خود را به صورت default gateway معرفی کنید:

```
#route add default 192.168.2.1
```

سویچ های فرمان natd

برنامه natd برای nat کردن ترافیک های شبکه از سوکت divert استفاده می کند و به صورت daemon در پس زمینه سیستم راه اندازی می شود و عمل nat کردن را انجام می دهد، در این بین این فرمان دارای سویچ ها و قابلیتهای زیادی است که برخی از آنها را برای شما ذکر می کنم.

بخش log

برای فعال کردن قابلیت log در natd باید از سویچ های log– یا سویچ l– استفاده کنید، گزارشات در فایلی به نام alias.log در زیر شاخه /var/log ذخیره می شود.

بخش مسدود کردن ترافیک های ورودی غیر ضروری:

برای اینکه سرویس natd به ترافیک هایی را که در جدول ترجمه خود وجود ندارند اجازه ورود ندهد از سویچ –deny_incoming و یا d– استفاده کنید. این قابلیت را همیشه فعال کنید

بخش use_sockets

این بخش در حالتهایی که شما از سرویس هایی که سوکتهای اضافی برای دریافت اطلاعات خود نیاز دارند ایجاد می کند مثل فرمان .ftp برای به درستی کار کردن این نوع از برنام ها از سویچ use_sockets - یا s– استفاده کنید.

سویچ verbose

این سویچ برنامه natd را راه اندازی نمی کند بلکه برنامه را در حالتی قرار می دهد که بتوانید اطلاعات بیشتری از وضعیت سرویس خود را در کنسول شل مشاهده کنید برای دیباگ کردن برنامه ها بسیار مفید است و با استفاده از سویچ verbose – یا v–فعال می شود.

قابلیت port redirection و address redirection در Natd

یکی از قابلتهای که کمتر از در nat مورد بررسی قرار گرفته redirection در port و آدرس است. کاربرانی که در شبکه محلی در پشت سرویس nat قرار دارند به هیچ عنوان از طریق شبکه بیرونی قابل دسترس نبود مگر در سرویس natd قابلیت redirectionفعال و پیکربندی شده باشد. در این بخش با روش های راه اندازی این قابلیت در natd بیشتر آشنا می شوید.

فایل پیکربندی :natd قبل از شروع با سویچی از برنامه natd آشنا می شوید به نامconfig ، این سویچ که به اختصار هم می توانید آنرا با f– مشخص کنید باید با نام فایلی همراه باشد که فایل پیکربندی است برای .natd این فایل در قابل یک فایل متنی است و هر خط باعث فعال شدن یک قابلیت می شود و خطوط خالی تاثیری در رفتار برنامه ندارد و خطی که با # شروع شود حکم توضیحات را دارد، در ادامه با روش های ایجاد کردن این فایل بیشتر آشنا می شوید، برای مثال برای فعال کردن قابلیت log در natd خطی به صورت زیر در فایل مورد نظر بنویسید log yes :این بخش در حقیت با سویچ log– برابر است. یکی از قابلیت های این فایل دارا بودن بخشهای متفاوت است که هر بخش به صورت جداگانه خورد بررسی توسط برنامه natd قرار می گیرد و می توانید یک یا چند پردازش با قابلیت های مختلفی ایجاد کند به این بخشها به اصطلاح instance گفته شده که اولین و پیش فرض آن default نامیده می شود، برای بخش بندی کردن فایل پیکربندی با instance های مختلف باید از کلمه instance و بعد نام مورد نظر خود استفاده کنید به صورت زیر instance instance_name :شما به راحتی می توانید این فایل را ایجاد کنید و صورت زیر فایل را به برنامه natd معرفی کنید #natd –config /etc/natd.conf : or #natd –f /etc/natd.confنامه فایل natd.conf یک نام انتخابی است.

قابلیت :port redirection شما با استفاده از این قابلیت می توانید درخواستهایی که از به سمت پورت خاصی از سمت شبکه بیرونی شما می اید را به سمت کلاینتی که در شبکه محلی شماست منتقل کند دستور کلی این بخش به صورت زیر است:

```
-redirect_port proto targetIP:targetPOR
```

در زیر مثالی از این قابلیت را مشاهده می کنید

-redirect_port tcp 192.168.0.3:80 80

با استفاده از این فرمان ترافیک هایی که به سمت پورت 80 از سمت سیستم های خارج از LAN ارسال می شوید به کمک برنامه natd ترجمه شده و به سمت کلاینت با آدرس 192.168.0.3 ارسال می شود. شما به 3 روش می توانید redirection را فعال کنید،

روش اول با استفاده از فرمانnatd

```
#natd -redirect_port tcp 192.168.0.3:80 80
```

روش دوم در فایل rc.conf

که در زیر شاخه etc/ قرار دارد با استفاده کردن از گزینه natd_flags به صورت زیر این قابلیت را فعال کنید و دوباره سرویس natdرا restart کنید :

```
natd_flags="-redirect_port tcp 192.168.0.3:80 80
```

حال برای راه اندازی natd از فرمان زیر استفاده کنید

```
#/etc/rc.d/natd restart
```

و یا خط redirect_port tcp 192.168.0.3:80 80 را در فایل config در instance پیش فرض قرار دهید و با سویچ f–فرمان natd را اجرا کنید.

قابلیت address redirection

این قابلیت زمانی مورد استفاده قرار میگرد که بیش از یک آدرس ip در بخش nat سرور وجود داشته باشد و برای هر کلاینت در شبکه محلی بتوان یک آدرس ip بیرونی اختصاص داد. با استفاده از این قابلیث سرویس nat می تواند ترافیک ها را در بین کلاینتها و درخواست های بیرونی مدیریت کند و با اصطلاح این نوع از nat به static NAT معروف است. حالت کلی این قابلیت به صورت زیر است

```
-redirect_address localIP publicIP
```

در بخش localIP شما آدرس های کلاینتهای شبکه محلی خود را نوشته و در بخش publicIP آدرس شبکه خارجی نوشته تا این دو آدرس متناظر هم قرار گیرند. در زیر مثالی از این نوع nat را مشاهده می کنید:

```
-redirect_address 192.168.0.2 217.218.100.10 -redirect_address 192.168.0.3 217.218
.100.11
```

شما مثل بخش قبلی می توانید از سه روش فعال سازی این قابلیت استفاده کنید، روش فرمانnatd ، فایل rc.conf و فایل پیکربندی.

مخفی کردن NAT از فرمان traceroute

در هسته FreeBSD قابلیتی وجود داردبه نام IPSTEALTH که با استفاده از آن شما می توانید سرور NAT شبکه خود را از دیده فرمان traceroute مخفی کنید این قابلیت فقط در هسته فعال می شود و برای راه اندازی آن نیاز به کامپال کردن هسته دارید. این قابلیت به این صورت کار می کند که TTL بسته های دریافتی را بدون تغییر رد کرده، برای فعال کردن خط زیر را در فایل پیکربندی هسته اضافه کنید

```
options IPSTEALTH # Enable stealth forwarding
```

برای فعال کردن این قابلیت از فرمان sysctl به صورت زیر استفاده کنید

```
# sysctl net.inet.ip.stealth=1
```

در بسیاری از موارد شاید شما دوستان با خطای زیر در انجام دادن فرمان های زیر مواجه شوید:

```
natd: instance default: aliasing address not given
```

این خطا در زمانی روخ می دهد که سرویس natd شما در حالت سرویس دهی قرار دارد و برا اضافه کردن قابلیت جدید امادگی ندارد، برای حل این مشکل بهتر است که شما سرویس خود را غیرفعال کنید و تمام تنظیمات خود را در فایل پیکربندی قرار دهید و از طریق سویچ f- سرویس natd را دوباره راه اندازی کنید یا در یک فرمان تمام قابلیتهای مورد نظر خود را به سرویس اضافه کنید.

راه اندازی Zebra در FreeBSD

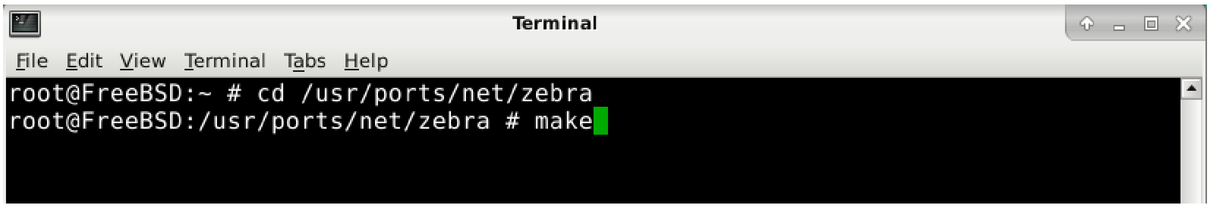
در FreeBSD برنامه ای وجود دارد به نام Zebra که با استفاده از آن شما می توانید سیستم خود را به یک router قوی با پروتکلهای مسیریابی زیر تغییر کاربری دهید، پروتکل هایی که zebra از آن حمایت می کند شامل لیست زیر است:

- پروتکل BGP
- پروتکل OSPF
- پروتکل RIP

علاوه بر این پروتکل های مسیر یابی از آدرس IP ورژن 4 و 6 پشتیبانی کرده و شما می توانید از SNMP برای دریافت گزارشات استفاده کنید. این برنامه از پورت کنسول VTY استفاده می کند، در ادامه با روش نصب و پیکربندی و استفاده از این سرور قوی آشنا می شوید.

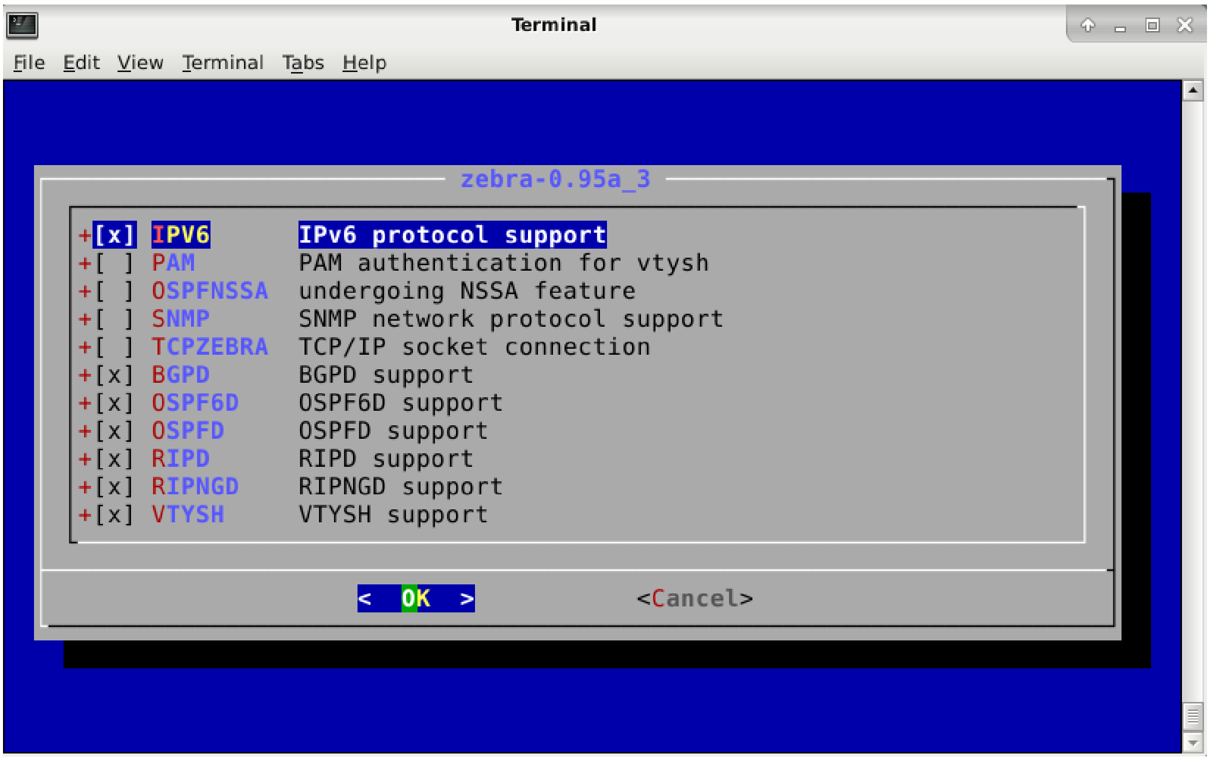
نصب Zebra در FreeBSD:

شما هم می توانید از طریق سیستم بسته های باینری و هم از طریق سیستم پورت برنامه zebra را نصب کنید، بهترین حالت نصب استفاده از سیستم پورت در FreeBSD است به این علت که شما می توانید گزینه های انتخابی و خاصیت های که از این گزینه ها به آن نیاز دارید را در برنامه فعال کنید در قدم اول وارد شاخه /usr/ports/net/zebra شوید و فرمان make را اجرا کنید این فرمان در شکل زیر نمایش داده شده است:



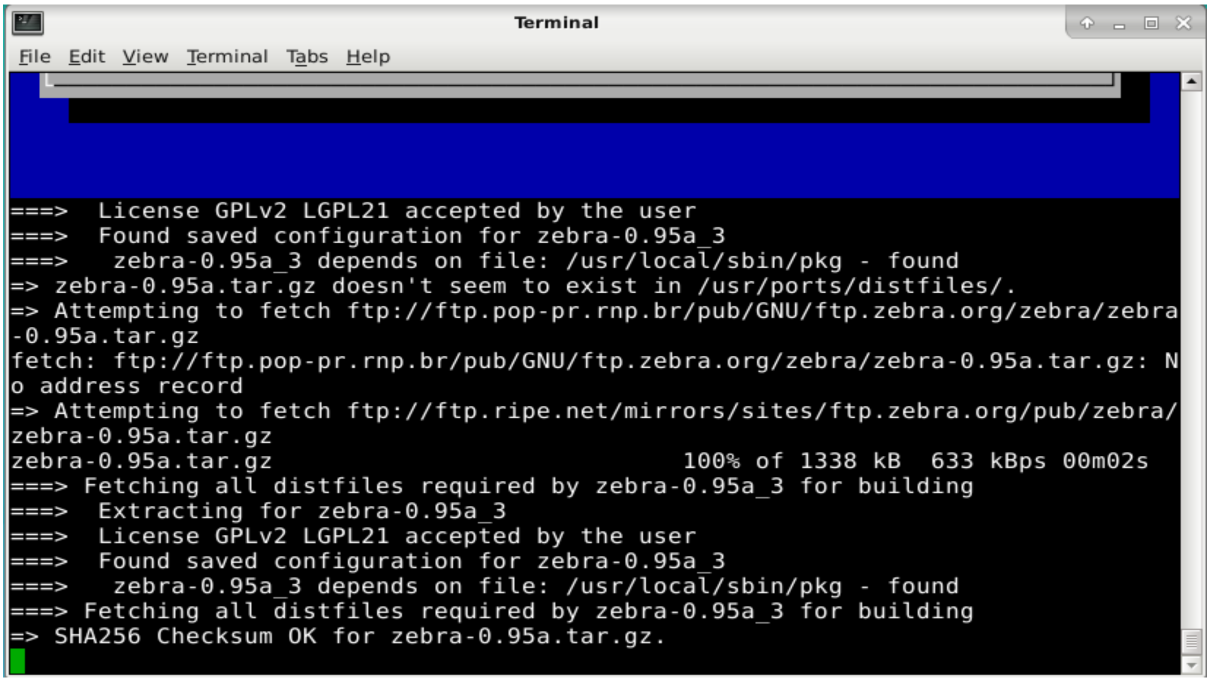
شروع به نصب zebra

در مرحله بعدی شما با صفحه ای مواجه می شوید که در آن گزینه های مورد نیاز را می توانید فعال و یا غیر فعال کنید، برای مثال شما می توانید از SNMP استفاده کنید به شرطی که در برنامه آن را اضافه کرده باشید، برنامه پیش فرض این قابلیت را ندارد:



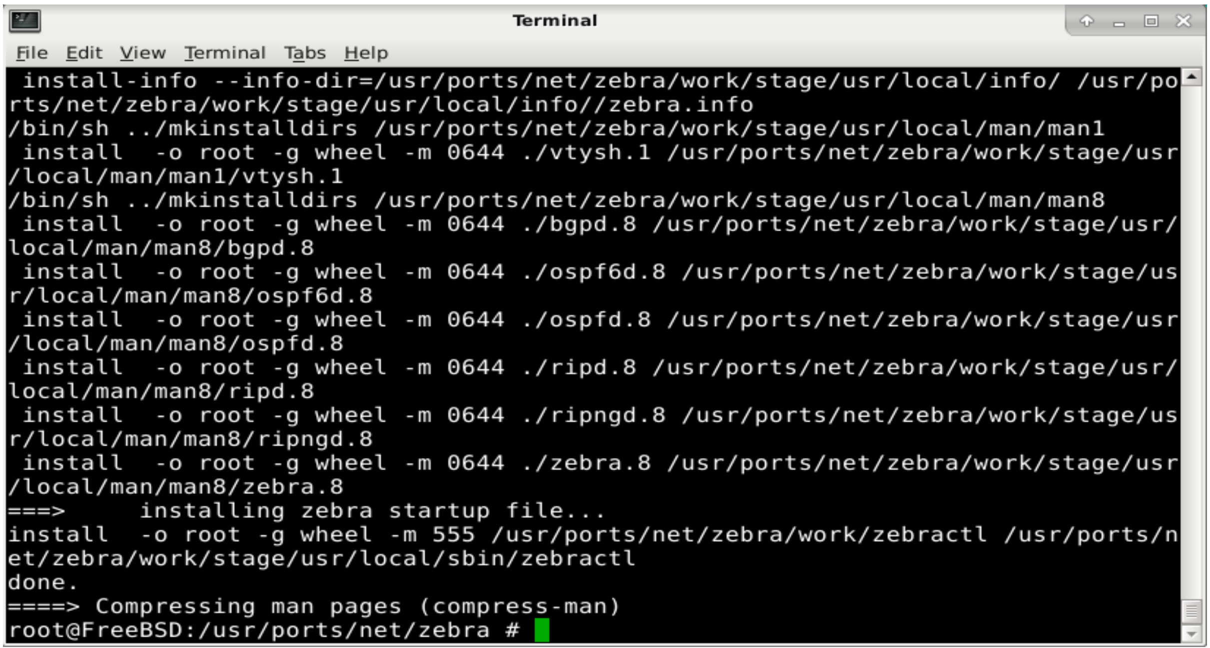
بخش تنظیمات zebra در زمان نصب

بعد از انتخاب کردن گزینه های مورد نیاز سیستم پورت به صورت خودکار اقدام به دانلود کردن برنامه و پیش نیاز های نصبی zebraکرده و در خروجی مراحل را برای شما نمایش می دهد، بخش ابتدایی از این خروجی برای شما نمایش داده شده است:



نمایش مراحل نصب zebra

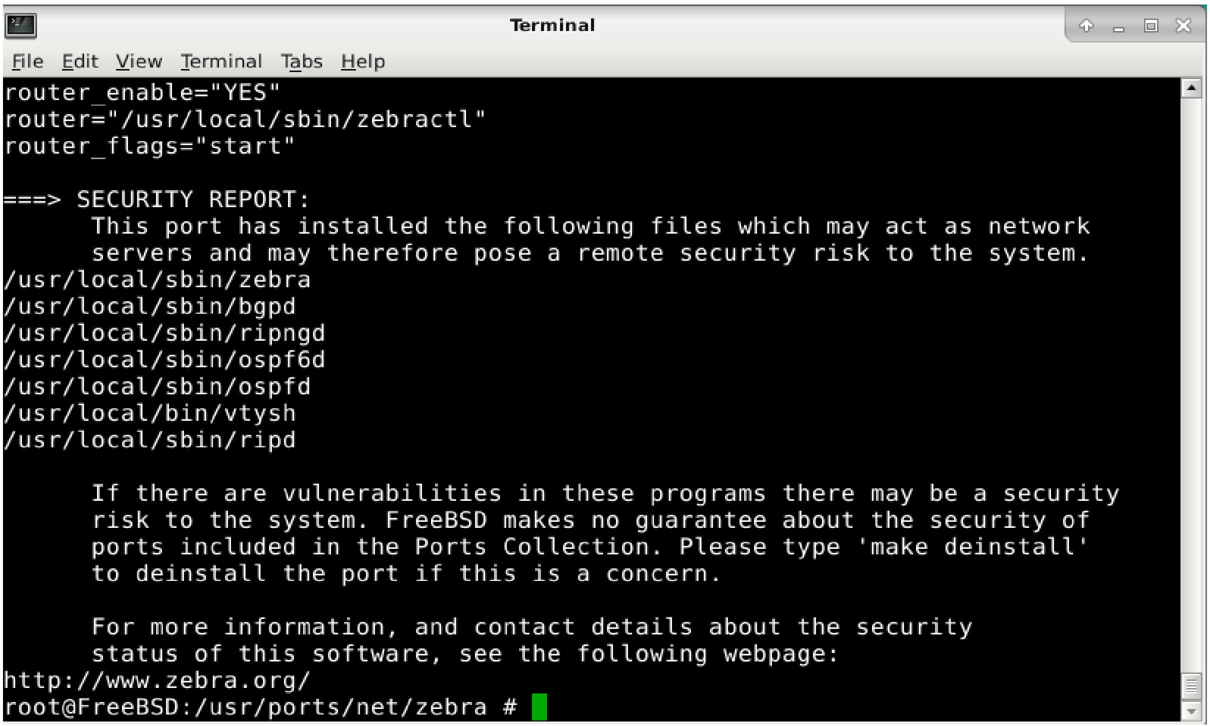
بعد از اتمام فرمان make اگر این برنامه با خطایی مواجه نشود و به درستی اجرا شود شما به خط فرمان باز می گردید و خروجی به صورت شکل زیر را مشاهده می کنید:



```
 install-info --info-dir=/usr/ports/net/zebra/work/stage/usr/local/info/ /usr/po
rts/net/zebra/work/stage/usr/local/info//zebra.info
/bin/sh ../mkinstalldirs /usr/ports/net/zebra/work/stage/usr/local/man/man1
 install  -o root -g wheel -m 0644 ./vtysh.1 /usr/ports/net/zebra/work/stage/usr
/local/man/man1/vtysh.1
/bin/sh ../mkinstalldirs /usr/ports/net/zebra/work/stage/usr/local/man/man8
 install  -o root -g wheel -m 0644 ./bgpd.8 /usr/ports/net/zebra/work/stage/usr/
local/man/man8/bgpd.8
 install  -o root -g wheel -m 0644 ./ospf6d.8 /usr/ports/net/zebra/work/stage/us
r/local/man/man8/ospf6d.8
 install  -o root -g wheel -m 0644 ./ospfd.8 /usr/ports/net/zebra/work/stage/usr
/local/man/man8/ospfd.8
 install  -o root -g wheel -m 0644 ./ripd.8 /usr/ports/net/zebra/work/stage/usr/
local/man/man8/ripd.8
 install  -o root -g wheel -m 0644 ./ripngd.8 /usr/ports/net/zebra/work/stage/us
r/local/man/man8/ripngd.8
 install  -o root -g wheel -m 0644 ./zebra.8 /usr/ports/net/zebra/work/stage/usr
/local/man/man8/zebra.8
===>     installing zebra startup file...
install  -o root -g wheel -m 555 /usr/ports/net/zebra/work/zebractl /usr/ports/n
et/zebra/work/stage/usr/local/sbin/zebractl
done.
====> Compressing man pages (compress-man)
root@FreeBSD:/usr/ports/net/zebra #
```

نمایش مراحل نصب zebra

حال در مرحله بعدی فرمان make install را اجرا کنید:



```
router_enable="YES"
router="/usr/local/sbin/zebractl"
router_flags="start"

===> SECURITY REPORT:
      This port has installed the following files which may act as network
      servers and may therefore pose a remote security risk to the system.
/usr/local/sbin/zebra
/usr/local/sbin/bgpd
/usr/local/sbin/ripngd
/usr/local/sbin/ospf6d
/usr/local/sbin/ospfd
/usr/local/bin/vtysh
/usr/local/sbin/ripd

      If there are vulnerabilities in these programs there may be a security
      risk to the system. FreeBSD makes no guarantee about the security of
      ports included in the Ports Collection. Please type 'make deinstall'
      to deinstall the port if this is a concern.

      For more information, and contact details about the security
      status of this software, see the following webpage:
http://www.zebra.org/
root@FreeBSD:/usr/ports/net/zebra #
```

نمایش مراحل نصب zebra

در این بخش اطلاعات جالبی برای شما نمایش داده شده است، برای مثال برنامه هایی که نصب شده است را برای شما نمایش می دهد.

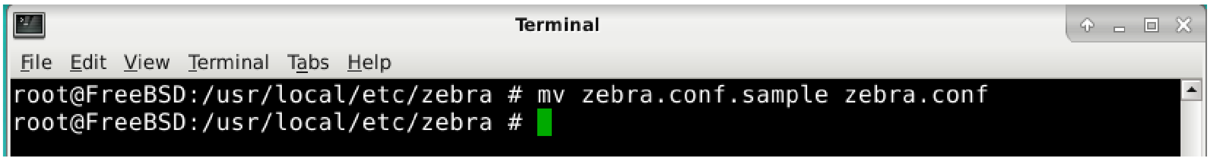
شاخه فایل های سرور Zebra

بعد از اتمام نصب در زیر شاخه /usr/local/etc/zebra فایل هایی قرار دارد که در شکل زیر لیست آنها را مشاهده می کنید:

```
Terminal
File Edit View Terminal Tabs Help
root@FreeBSD:~ # cd /usr/local/etc/zebra/
root@FreeBSD:/usr/local/etc/zebra # ls
bgpd.conf.sample        ospfd.conf.sample       vtysh.conf.sample
bgpd.conf.sample2       ripd.conf.sample        zebra.conf.sample
ospf6d.conf.sample      ripngd.conf.sample
root@FreeBSD:/usr/local/etc/zebra #
```

شاخه فایلهای پیکربندی zebra

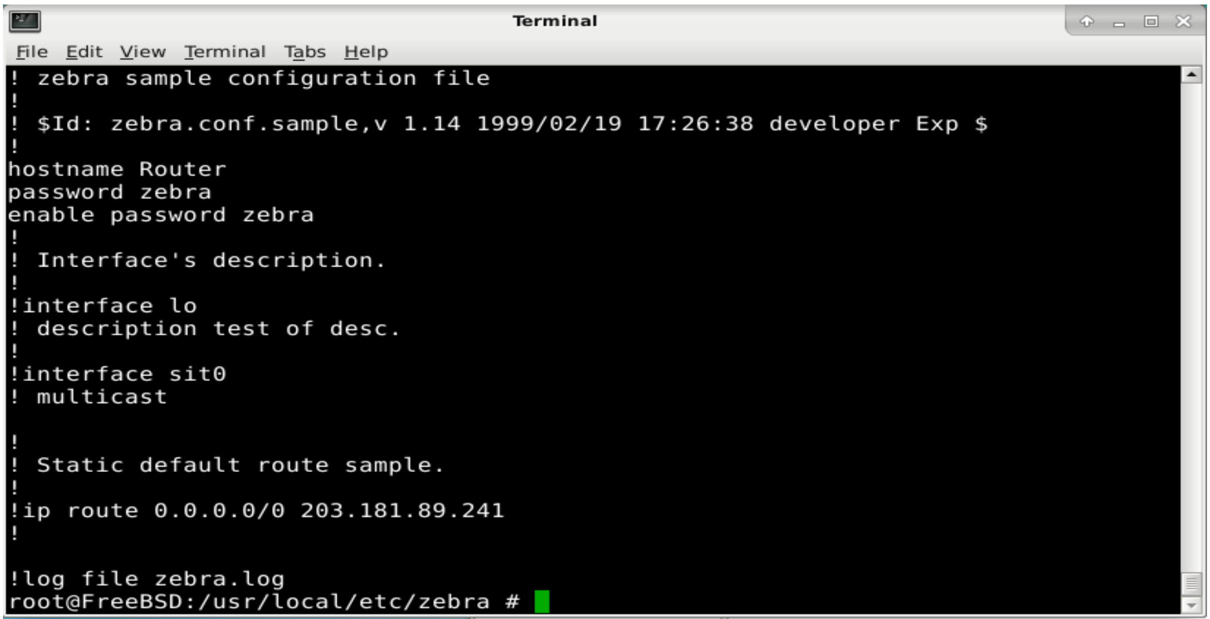
در قدم اول شما باید فایل اصلی پیکربندی به نام zebra.conf را ایجاد کنید، نام این فایل به صورت پیش فرض zebra.conf.sampleاست.



```
Terminal
File Edit View Terminal Tabs Help
root@FreeBSD:/usr/local/etc/zebra # mv zebra.conf.sample zebra.conf
root@FreeBSD:/usr/local/etc/zebra #
```

Backup گرفتن از فایل پیکربندی پیش فرض

فایل پیش فرض این برنامه شامل خطوط زیر است:



```
Terminal
File Edit View Terminal Tabs Help
! zebra sample configuration file
!
! $Id: zebra.conf.sample,v 1.14 1999/02/19 17:26:38 developer Exp $
!
hostname Router
password zebra
enable password zebra
!
! Interface's description.
!
!interface lo
! description test of desc.
!
!interface sit0
! multicast

!
! Static default route sample.
!
!ip route 0.0.0.0/0 203.181.89.241
!

!log file zebra.log
root@FreeBSD:/usr/local/etc/zebra #
```

نمایش محتوای فایل پیش فرض برنامه zebra

فایل پیکربندی این برنامه به صورت یک پیکربندی روتر است، برای مثال Hostname برنامه Router در نظر گرفته شده است، دو password مهم این برنامه zebra است که شما می توانید از طریق این فایل آنرا تغییر دهید.

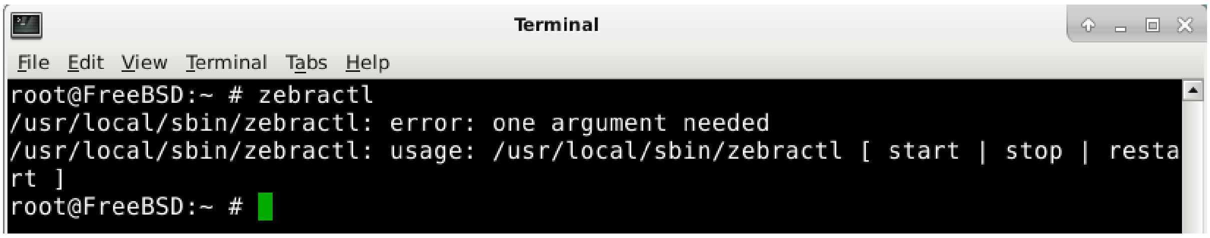
راه اندازی Zebra:

برنامه Zebra از طریق سیستم rc.conf قابل راه اندازیست، برا راه اندازی آن باید در فایل /etc/rc.conf خطوط زیر را وارد کنید:

```
defaultroute=”NO”
router_enable=”YES”
router=”/usr/local/sbin/zebractl”
router_flags=”Start”
```

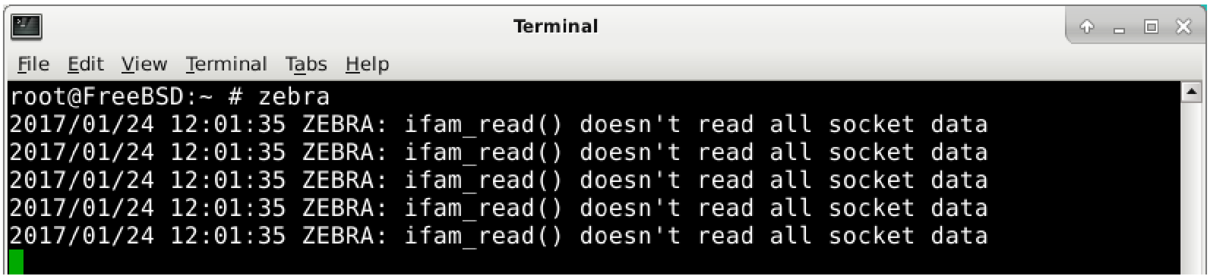
برنامه zebra با دو فرمان راه اندازی می شود، فرمان اول zebra و فرمان دوم .zebractl برای راه اندازی کردن با استفاده از فرمان zebra به سویچ های این برنامه دقت کنید، برای راه اندازی کردن در حالت daemon از سویچ d و برای اینکه فایل پیکربندی غیر از فایل پیش فرض را قرار دهید از سویچ f استفاده کنید.

خروجی فرمان zebractl را هم در شکل زیر مشاهده می کنید:



خروجی فرمان zebractl

این فرمان همان کار فرمانهای rc.d را انجام می دهد، در ادامه فرمان zebraرا بدون سویچ اجرا کنید تا برنامه راه اندازی شود، خروجی این فرمان به صورت زیر است که با چند خطای هشداری برای فعال کردن سیستم log مواجه می شود:



خطاهای هشداری در راه اندازی zebra

همانطوری که مشاهده می کنید برنامه راه اندازی شده و خط فرمان برای شما باز نشده است برای اینکه خط فرمان برای شما باز شود از سویچ d استفاده کنید و در بخش بعدی با استفاده از فرمان sockstat وضعیت پورتی را که برنامه برای انجام اقدامات مدیریتی برروی آن به گوش کردن از شبکه می پردازد را مشاهده کنید:

```
root@FreeBSD:~ # zebra -d
2017/01/24 12:04:03 ZEBRA: ifam_read() doesn't read all socket data
2017/01/24 12:04:03 ZEBRA: ifam_read() doesn't read all socket data
2017/01/24 12:04:03 ZEBRA: ifam_read() doesn't read all socket data
2017/01/24 12:04:03 ZEBRA: ifam_read() doesn't read all socket data
2017/01/24 12:04:03 ZEBRA: ifam_read() doesn't read all socket data
root@FreeBSD:~ # sockstat -l4
USER     COMMAND    PID   FD PROTO  LOCAL ADDRESS         FOREIGN ADDRESS
root     zebra      8209  8  tcp4   *:2601                *:*
root     ftpd       1442  6  tcp4   *:21                  *:*
root     sendmail   652   3  tcp4   127.0.0.1:25          *:*
root     sshd       649   4  tcp4   *:22                  *:*
root     syslogd    513   7  udp4   *:514                 *:*
```

تست کردن راه اندازی شدن zebra

برای برقرار ارتباط با Zebra کافیست که به پورت شماره telnet 2601 کنید و بعد از وارد کردن رمز عبوری که در فایل Zebra.conf مشخص کرده اید وارد کنسول مدیریت این برنامه بشوید:

```
root@FreeBSD:~ # telnet 127.0.0.1 2601
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.

Hello, this is zebra (version 0.95a).
Copyright 1996-2004 Kunihiro Ishiguro.


User Access Verification

Password:
Router> ena
Router> enable
Password:
Router#
```

وارد شدن به zebra با استفاده از telnet

شما به راحتی مثل یک روتر سیسکو می توانید فرمان ها را اجرا کنید و خروجی را مشاهده کنید، برای مثال برای مشاهده کردن وضعیت جاری از فرمان show running-config به صورت زیر استفاده کنید:

```
Router# show running-config

Current configuration:
!
hostname Router
password zebra
enable password root
!
interface em0
 ip address 192.168.1.1/24
 ipv6 nd suppress-ra
!
interface lo0
!
!
line vty
!
end
Router#
```

فرمان show running-config در zebra

شما به راحتی مثل یک روتر می توانید وارد بخش پیکربندی شده و برروی کارت شبکه خود آدرس IP اضافه کنید:

```
Router# configure terminal
Router(config)# interface em0
Router(config-if)# ip address 192.168.2.2/24
Router(config-if)# exi
Router(config)# exit
```

تنظیم کردن آدرس IP در کارت شبکه با استفاده از zebra

روش اتصال با vtysh

در زیر شاخه فایلهای پیکربندی موجود در شاخه /usr/local/etc/zebra فایلی وجود دارد به نام vtysh.conf.sample که شما باید نام آنرا به vtysh.conf تغییر دهید، این فایل توسط فرمان vtysh خوانده شده و شما را وارد بخش مدیریت Zebra می کند در زیر فایل پیکربندی این فرمان را مشاهده می کنید:

```
^[ (escape) menu  ^y search prompt  ^k delete line   ^p prev li   ^g prev page
^o ascii code     ^x search         ^l undelete line ^n next li   ^v next page
^u end of file    ^a begin of line  ^w delete word   ^b back 1 char
^t top of text    ^e end of line    ^r restore word  ^f forward 1 char
^c command        ^d delete char    ^j undelete char ^z next word
=====line 1 col 0 lines from top 1 ==========================================
!
! vtysh sample configuratin file
!
!username kunihiro nopassword
```

استفاده از vtysh در zebra

سرویس NIS یا Network Information System درFreeBSD

سرویس NIS برای مدیریت کردن مرکز منابع برای سیستم عاملهای شبه یونیک مثل سولاریس و خانواده BSD طراحی وایجاد شده است. در زمان قدیمی این سرویس به نام Yellow Pages معروف بود ولی به دلیل مشکلات ایجاد شده برای برند آن نام به NIS تغییر پیدا کرده است ولی باز هم فرمانهای این برنامه با yp شروع می شوند و در بسیاری از موارد به yellow pages معروفت است.

NISاز سیستم مشتری/ سروری مبتنی بر پروتکل RPC استفاده می کنید که به همین دلیل می توانید فایلهای پیکربندی خود را در بین سیستمهایی که در یک دامنه NIS قرار دارند به اشتراک بگذارید. همین قابلیت به مدیران سیستم این اجاره را می دهد که به صورت مرکزی بتوانند تغییراتی در سیستم های مورد نظارت خود به صورت مرکزی انجام دهند. در ادامه این مقاله با این سرویس آشنا می شوید.

در سیستم عامل FreeBSD ار ورژن NIS 2 استفاده می شود.

اصطلاحات استفاده شده در NIS

بخشNIS domain name

سرور و کلاینت اطلاعات خود را در یک دامنه اسمی به اشتراک گذاشته می شود. این نام انتخابی است.

سرویسrpcbind

این سرویس باعث فعاال شدنRPC می شود که باعث برقراری ارتباط بین سرور و کلاینت می شود.

سرویسypbind

این برنامه هسته اصلی ارتباطات در محیط NIS است. اگر این سرویس برروی سیستم کلاینت فعال نشود نمی تواند به سرور متصل شود.

سرویسypserv

این برنامه برای راه اندازی سرور NIS استفاده می شود. در راه اندازی NIS شما می توانید علاوه بر سرور اصلی سروی جایگزین هم داشته باشید تا در صورتی که این سرویس غیرفعال شد درخواستها به سمت سرور جایگزین ارسال شود.

برنامه:rpc.yppasswdd

این برنامه فقط در سرور اصلی NIS نصب و راه اندازی می شود و به کلاینتهای شبکه NIS این اجازه می دهد که بتوانند رمز عبور مربوط به NIS خودشون را تغییر بدهند. اگر این سرویس راه اندازی نشده باشد کلاینت ها توان تغییر دادن رمزعبور خود را نخواهند داشت.

نقش های موجود در NIS

در شبکه های مبتنی بر NIS سه نقش وجود دارد که در ادامه این سه نقش را توضیح می دهیم.

نقش Master

این سرور نقش اصلی را در NIS ایفا می کند و تمام اطلاعات و فایلهای پیکربندی شبکه شما برروی این سیستم ذخیره شده است، فایلهای اصلی در این بخش فایلهای password فایل groups و سایر فایلهای پیکربندی در این سرور ذخیره می شود.

نقش slave

این سرور در حقیقت یک کپی از سرور اصلی است که در صورت از دسترس خارج شدن سرور اصلی از شبکه این سرور پاسخگویی در شبکه داشته باشد. از این سرور هم می توان به عنوان تقسیم کننده درخواستها و ترافیک بین سرور اصلی هم باشد.

نقش Client

سایر سیستم هایی که در شبکه NIS وجود دارد و از سرور اصلی سرویس می گیرد سیستم های Client هستند.

انتخاب کردن نام NIS Domain

از این نام برای برقراری ارتباط درون شبکه ای استفاده می شود. در بعضی از موارد برخی از سازمان ها این نام را با نام DNS خود یکی می کنند که برای رفع کردن ایرادات شبکه ای مشکل ایجاد می کند. پس در اننتخاب این نام دقت کنید. در انتخاب کردن سروری که قصد دارید NIS را برروی آن نصب کنید به این نکته دقت کنید که سروری را انتخاب کنید که همیشه در شبکه در دسترس بوده و خاموش و روشن نشود.

نصب کردن NIS Master Server

قبل از راه اندازی کردن فرمان های مربوط به سرور NIS در ابتدا در فایل r.conf خط زیر را وارد کنید و همه تنظیمات شبکه خود را دوباره راه اندازی کنید، این سرویس به نام دامنه NIS حساس بوده و در صورت موجود نبودن خطا ایجاد می کند:

```
nisdomainname="Domain name"
```

برای راه اندازی کردن مجدد تمام تنظیمات شبکه خود فرمان زیر را وارد کنید:

```
# /etc/netstart
```

تنظیمات DNS خود را هم چک کنید.

تمام فایل های اصلی در این سرور ذخیره می شود، دیتابیسی که اطلاعات در آن ذخیره می شود NIS Mape نام دارد که در زیر شاخه /var/yp ذخیره می شود البته در شاخه به نام همان NIS domain. این حالت زمانی خوب است که شما بتوانید چندین نام NIS برروی یک سرور داشته باشید. پردازش اصلی این سرور ypserv است که به همه درخواستها پاسخ می دهد.در حقیت این پردازش یک مترجم بین درخواستهای کلاینت هاو دیتابیس است.

شما در FreeBSD نیازی نصب NIS ندارید چون به صورت پیش فرض در خود سیستم نصب شده است و شما فقط نیاز دارید که آنرا از طریق فایل rc.conf سرور NIS را فعال کنید برای فعال کردن این سرور خطوط زیر را در فایل /etc/rc.conf اضافه کنید:

```
nisdomainname="abedini"
nis_server_enable="YES"
nis_yppasswdd_enable="YES"
```

در خط اول شما یک نام برای دامنه اسمی خود انتخاب کرده اید به نامabedini ، با استفاده از گزینه خط دوم در زمان راه اندازی سیستم سرویس سرور NIS را فعال می کنید.و در خط سوم سرویسی که مخصوص سرور است را راه اندازی می کند.

بعد از اعمال تنظیمات شما می توانید با فرمان زیر سرور را راه اندازی کنید:

```
# service ypserv start
```

در صورت بروز خطا مبنی بر وجود شاخه در زیر شاخه /var/yp شاخه مورد نظر مطابق با نام NIS Domain را ایجاد کنید.

ایجاد کردن NIS Maps

قبل از راه اندازی این بخش باید ypserve را راه اندازی کنید.

این دیتابیس از فایلهای پیکربندی که در زیر شاخه /etc قرار دارد ایجاد می شود، به استثنای فایل /etc/master.passwd. برای جلوگیری از انتشار رمزهای عبور در بین تمام سرور ها به صورت دستی این فایل را زیر شاخه /var/yp کپی کنید و در مرحله بعدی تمام نام کاربری هایی که در سیستم NIS به آنها نیازی ندارید را جذف کنید، از فرمان های زیر برای کپی کردن فایل استفاده کنید:

```
# cp /etc/master.passwd /var/yp/master.passwd
# cd /var/yp
# vi master.passwd
```

سطح دسترسی 600 را برروی این فایل اعمال کنید. برای ایجاد کردن دیتابیش در FreeBSD فرمانی وجود دارد به نام ypinit که شما با استفاده از آن می توانید سرور خود را ایجاد کنید، برای ایجاد کردن نقش master از سویچ m– استفاده کنید و بعد از آن نام انتخابی خود را وارد کنید، خروجی فرمان زیر به صورت زیر است:

```
FreeBSD# ypinit -m abedini
Server Type: MASTER Domain: abedini
Creating an YP server will require that you answer a few questions.
Questions will all be asked at the beginning of the procedure.
Do you want this procedure to quit on non-fatal errors? [y/n: n] n
Ok, please remember to go back and redo manually whatever fails.
If not, something might not work.
At this point, we have to construct a list of this domains YP servers.
rod.darktech.org is already known as master server.
Please continue to add any slave servers, one per line. When you are
done with the list, type a <control D>.
master server   :  pc
next host to add:  ^D
The current list of NIS servers looks like this:
ellington
coltrane
Is this correct?  [y/n: y] y

[..output from map generation..]

NIS Map update completed.
FreeBSD has been setup as an YP master server without any errors.
```

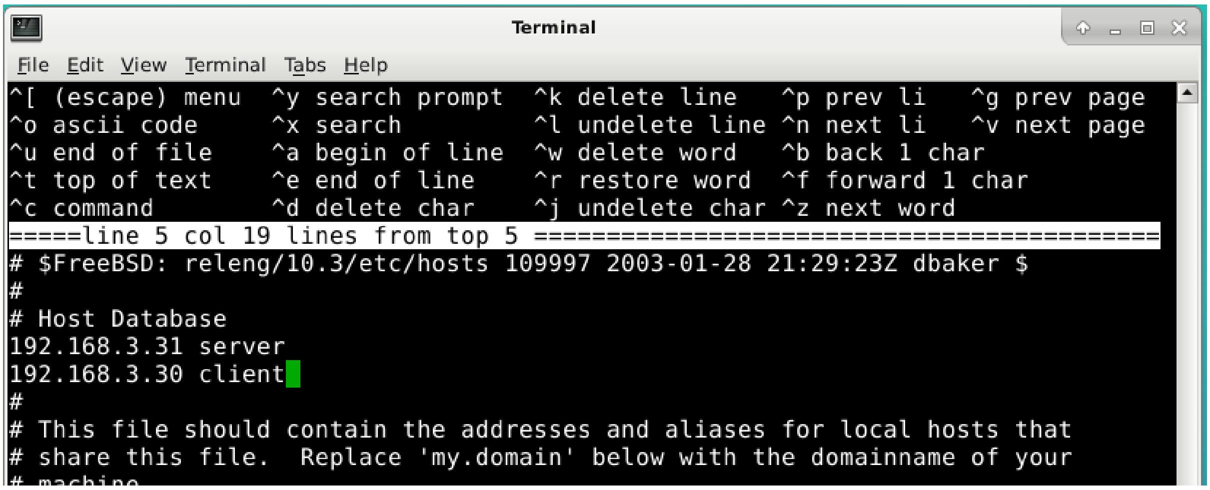
ر زمان نوشتن نام slave server باید برای خارج شدن از آن بخش از کلید های control و کلید D استفاده کنید،بعد از اتمام این بخش فایلی ایجاد می شود در زیر شاخه yp به نام ypserver که شما می توانید بعد از اتمام این بخش به آن اضافه کنی.

بعد از اتمام این مراحب خط انتهای خروجی به شما می گویند که سرور شما بدون خطا ایجاد شده است.

پیکربندی FreeBSD Client در NSI

در این بخش با روش های پیکربندی سیستم های client برای برقرار ارتباط با سرور های NIS آموزش داده می شود، در بخش قبلی سرور NIS پیکربندی شده است و در این بخش ما قصد داریم که مدیریت کردن کاربران را با استفاده از NIS مرکزی کنیم، قبل از شروع به کار کردن نکات زیر را رعایت کنید:

قبل از شروع در فایل host واقع شده در زیر شاخه /etc آدرس ip و نام هایی را که از آنها در شبکه خود استفاده می کنید را وارد کنید، فرض کنید که شما دو سیستم در شبکه خود دارید که یکی حال سروری دارد و یکی کلاینت است. به اختصار یکی را server و دیگری را client نام گذاری کنید و در بخش hostname در rc.conf هم این نام ها را تعیین کنید، در شکل زیر یک نمونه از تنظیمات فایل hots را مشاهده می کنید:



تعریف کردن نام ها در فایل host

همانطوری که در شکل بالا مشاهده می کنید برای client آدرس 192.168.3.30 و برای سرور 192.168.3.31 تنظیم شده است. این بخشها اختیاری بوده و به طراحی شبکه شما بستگی دارد، به این نکته توجه کنید که این تنظیمات برروی سیستم های دیگر هم اعمال شود و یا از سرویس DNS در شبکه استفاده کنید.

نکته مهم:

قبل از راه اندازی کردن فرمان های مربوط به سرور NIS در ابتدا در فایل r.conf خط زیر را وارد کنید و همه تنظیمات شبکه خود را دوباره راه اندازی کنید، این سرویس به نام دامنه NIS حساس بوده و در صورت موجود نبودن خطا ایجاد می کند:

```
nisdomainname="Domain name"
```

برای راه اندازی کردن مجدد تمام تنظیمات شبکه خود فرمان زیر را وارد کنید:

```
# /etc/netstart
```

در پایان خروجی این فرمان نام دامنه NIS برای شما نمایش داده می شود و برای چک کردن مجدد تنظیمات کارت شبکه خود را با ifconfig چک کرده و از ping با نام های server و client هم استفاده کنید تا از صحت ip شبکه خود مطمئن شوید .

برای راه اندازی بخش کلاینتی در NIS کافیست که خطوط زیر را در فایل /etc/rc.conf قرار دهید:

```
nis_client_enable="YES"
nis_client_flags="-S abedini,server"
```

خط اول باعث راه اندازی ypbind یا همان بخش کلاینتی NIS می شود و در خط دوم شما بعد از سویچ حرف اس بزرگ نامه دامنه و بعد نام سرور را مشخص می کنید، خط دوم اجباری نبوده ولی برای افزایش کارایی بسیار مفید است.

در بخش سرور فایل رمزعبور یا master.passwd را در شاخه /var/yp کپی کرده و برای استفاده کردن از آن اماده است، در بخش کلاینت هم شما باید در انتهای فایل /etc/master.passwd خط زیر را اضافه کنید:

```
+:::::::::
```

برای ویرایش کردن این فایل /etc/master.passwd از ویرایشگر vipw استفاده کنید که مثل ویرایشگر vi است با این تفاوت که بعد از اعمال تغییرات فایل password شما را به روز می کند.

برای اینکه این سیستم شما بتواند از NIS برای دریافت اطلاعات کاربری استفاده کند علاوه بر خط بالا در فایل /etc/group هم خط زیر را اضافه کنید:

```
+:*::
```

این دو خط باعث می شود که سیستم شما برای کاربرانی که اطلاعات آنها در فایلهای مورد نظر نباشد از سرور NIS استفاده کند، شما می توانید تمام کاربران خود را در سیستم کلاینت حذف کرده و فقط کاربری را برای انجام دان امور مدیریت محلی سیستم خود نگه دارید که در گروه wheel باشد تا بتواند دسترسی root در صورت نیاز داشته باشد.

حال سرویس NIS در بخش کلاینت را به صورت زیر اجرا کنید:

```
# service ypbind start
```

محدود کردن کاربر حتی اگر در دیتابیس NIS وجود داشته باشد:

در برخی از موارد شما قصد دارید که ورود شدن به سیستم و استفاده از سیستم های خاصی را برای یک سری از کابران محدود کنید، برای این کار در فایل محلی master.passwd قبل از خط + قبل از نام کاربر مورد نظر علامت – قرار داده و به تعداد صحیح بخش های مربوط به فایل از علامت : استفاه کنید به صورت زیر:

```
nobody:*:65534:65534::0:0:Unprivileged user:/nonexistent:/sbin/nologin
-admin:::::::::
```

```
+:::::::::
```

نکته:

تا زمانی که کاربر در فایل master.passwd در شاخه /var/yp اضافه نشده باشد اجازه وارد شدن به سیستم را نخواهد داشت، این بخش را قبل از هر اقدامی تست کرده و حتما برای کاربر رمزعبوری هم با passwd در نظر بگیرید؛ کاربر بدون رمز هم نمی تواند وارد شود.

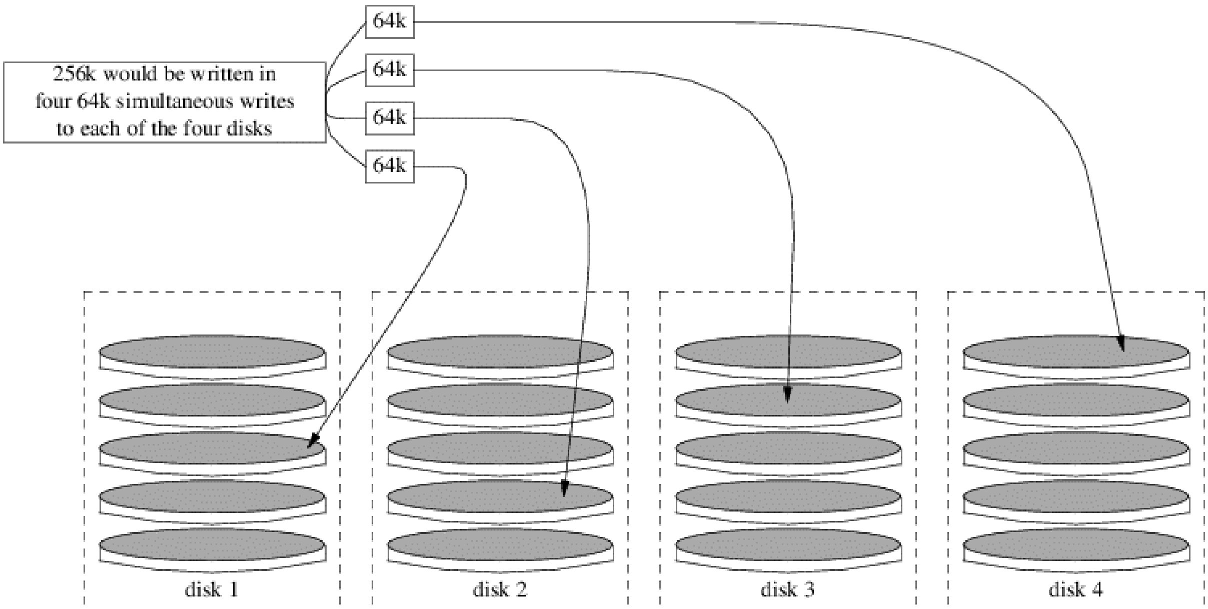
RAID0 در FreeBSD

یکی از بخشهای هم در مدیریت سیستم هایی که اطلاعات مهم شما برروی آن ذخیره شده است استفاده کردن از مکانیزم ها و رشهایی است که شما بتوانید در زمانی که اطلاعات شما به هر دلیل از بین رفت، فایلهای خود را بازگردانی کنید، برای این منظور دو روش وجود دارد، روش اول استفاده کردن از سیستم ها و روش های backup گیری است که در این روش در ساعات خواصی شما فایلهای مهم سیستم خود را در محلی دیگر و برروی سیستمی دیگر ذخیره می کنید، در این روش هم شما می توانید از فرمانهای ساده برای انجام دادن این کار مثل cp و sftp برای جابجابی و منتقل کردن فایلها برروی شبکه از آن استفاده کنید و یا از برنامه ای خودکار و حرفه ای Bacula برای انجام دادن این کار استفاده کنید.

این روش دارای یک عیب بزرگ است که اگر هاردی که شما فایلهای backup خود را برروی آن قرار می دهید دچار مشکل شود شما همه اطلاعات خود را از دست داده اید، برای جلو گیری کردن از این مشکل ف سیستمی طراحی شده است به نام RAIDکه هدف اصلی آن تقسیم کردن فایلهای سیستم شما در بین چندین هارد است که در صورتی که یکی از هارد های شما خراب شد به سرعت شما بتوانید از اطلاعات خود استفاده کنید، البته این سیستم هم دارای عیب شکسته شدن RAID است که در صورتی که RIAD از بین برود سیستم بازگردانی فایلهای شما دچار مشکل خواهد شد، البته از RAID ها برای بالا بردن سرعت خواندن و نوشتن سیستم هم استفاده می شود که انواع مختلفی دارد که برای مثل raid0 قابلیت بازگردانی فایلهای سیستم شما را نداشته و فقط برای افزایش سرعت به کار می رود.

در ادامه این بخش و در مقالات پیش رو شما با روش های راه اندازی RIAD هایی نرم افزاری موجود در FeeBSD آشنا می شود، همانطوری که می دانید RAID بر دو نوع کلی سخت افزاری و نرم افزاری تقسیم می شود و در سیستم عامل FreeBSD شما می توانید از RIAD نرم افزاری به استفاده کردن از GEOM framework استفاده کنید و سیستم خود را به صورتی مدیریت کنید که در صورت بروز مشکل بتوانید اطلاعات سیستم خود را بازیابی کنید.

در این مقاله شما با riad0 آشنا می شوید که در آن شما چندین دیسک را به یک دیسک تبدیل می کنید ، با این روش هم شما فضای دیسک خود را افزایش داده و هم سرعت در خواندن و نوشتن را به تناسب سرعت هر دیسک افزایش می دهید، به این نوع از raid به اصطلاح Striping می گویند که در شکل زیر یک طرح کلی و مجازی از آنرا مشاهده می کنید:



معرفی حالتهای riad

همانطوری که مشاهده می کنید در این نوع از raid نوشتن اطلاعات در بین چهار دیسک موجود سیستم شما تقسیم می شود و سرعت نوشتن شما چهار برابر قبل می شود. اندازی دیسک ها شما در این بخش باید یکسان باشد و I/O سیستم شما به صورت موازی در بین دیسک ها تقسیم می شود. همانطوری که قبلا اشاره شده این نوع از raid قابلیت redundancy ندارد و اگر اطلاعات شما مهم است از سایر روشهای backup گیری استفاده کنید چون زمانی که یک دیسک شما در این سیستم از دسترس خارج شود اطلاعات ذخیره شده روی آن از بین می رود، این سیستم برای سرور های کشی که اطلاعات متغییری را تولید می کنند و این اطلاعات دایمی نیستند بسیار مفید است چون سرعت نوشتن اطلاعات که یکی از باگهای راه اندازی این نوع از سرور ها است را می تواند افزایش دهد.

مراحل راه اندازی RAID0در FreeBSD

در اولین قدم شما برای راه اندازی کردن این نوع از raid برروی سیستم خود نیاز به بارگذاری کردن ماژول geom_stripe.ko در هسته سیستم عامل خود دارید، برای انجام دادن این عمل از فرمان kldload به صورت زیر استفاده کنید:

```
# kldload geom_stripe
```

در بخش بعدی شما باید یک مسیر یا شاخه برروی سیستم عامل خود ایجاد کنید تا RADi0 را به اصطلاح در آن شاخه mountکنید، البته در FreeBSD شاخه ای به نام /mnt برای این منظور در نظر گرفته شده است.

در مرحله بعدی شما باید دیسک های را که برای این منظور در نظر گرفته اید را به سیستم خود اضافه کنید و با استفاده از فرمان gstripe که در ادامه روش اجرا کردن آنرا مشاهده می کنید دیسک های خود را به یک دیسک واحد تبدیل کنید.

دیسک های اضافه شده در زیر شاخه /dev و بسته به نوع آنها در FreeBSD ناگذاری می شود،

در مثال زیر فرض بر این است که دو دیسک برای ایجاد کردن RAID0 به سیستم شما اضافه شده که این دیسک ها قبلا هیچ پارتیشنی برروی خود ندارند و نام آنها/dev/ad2 و/dev/ad3 هستند، و به صورت انتخابی و اختیاری نام st0 را برای RAIDخود در نظر بگیرید، فرمان های اجرا شده را در بخش زیر مشاهده می کنید:

```
# gstripe label -v st0 /dev/ad2 /dev/ad3
Metadata value stored on /dev/ad2.
Metadata value stored on /dev/ad3.
Done.
```

همانطوری که مشاهده کرده اید این دو دیسک Striping شده. بعد از اجر شدن این فرمان در زیر شاخه /dev/ شاخه ای به نام stripe ایجاد شده که Device به نام st0 که در حقیقت دو دیسک شما در قابل این یک دیسک است در آن شاخه ایجاد شده است،

در قدم بعدی شما باید label استاندارد موجود برروی سیستم FreeBSD را که به اصطلاح partition table است را برروی دیسک st0 ایجاد کنید، برای انجام دادن این بخش از فرمان bsdlabel به صورت زیر استفاده کنید:

```
# bsdlabel -wB /dev/stripe/st0
```

بعد از اجرا شدن این فرمان partition table مخصوص FreeBSD برروی دیوایس st0 که خود از دو هارد RAID شده از نوع Striping است ایجاد می شود، در مرحله بعدی شما نیاز دارید که فایل سیستمی برروی دیسک مورد نظر خود ایجاد کنید، برای این منظور باید از فرمان newfs به صورت زیر استفاده کنید و این فرمان را اجرا کنید تا فضای دیسک شما برای نوشتن و خواندن آماده شود:

```
# newfs -U /dev/stripe/st0a
```

بعد اجرا این فرمان به درستی و مشاهده کردن خروجی فرمان ، دیسک شما آماده به کار است و شما می توانید به راحتی و با استفاده کردن از فرمان mount به صورت زیر از این فضای دیسک برروی mount Point که در نظر گرفته اید استفاده کنید:

```
# mount /dev/stripe/st0a /mnt
```

همانطوری که در فرمان بالا مشاهده می کنید شما می توانیدبه RIAD ایجاد شده را از طریق شاخه /mnt دسترسی پیدا کنید.

راه اندازی خودکار RIAD0در زمان راه اندازی :FreeBSD

یکی از کارهایی که شما باید بعد از اتمام این کار برای دایمی شدن تنظیمات خود آنجام دهید در این بخش توضیح داده می شود، به هر دلیل شاید سیستم شما دوباره راه اندازی شود، پس شما باید تنظیمات خود را در FreeBSD ذخیره کنید تا سرویس

هایی که از این شاخه استفاده می کنند در زمان راه اندازی مجدد سیستم دجار مشکل نشوند. این بخش به دو مرحله کلی تقسیم میشود.

در مرحله اول شما باید ماژول geom_stripe.ko که مورد نیازی این نوع از RAID است را در زمان راه اندازی سیستم در هسته بارگذاری کنید، شما یا می توانید آن قابلیت را به صورت دایمی در هسته FreeBSD ایجاد کنید و یا به صورت زیر با استفاده از فایل loader.conf آنرا در زمان راه اندازی به هسته اضافه کنید، برای انجام دادن این کار فرمان زیر را اجرا کنید:

```
# echo 'geom_stripe_load="YES"' >> /boot/loader.conf
```

با اجرا کردن این فرمان خط ”geom_stripe_load="YES در فایل loader.conf قرار میگیرد، این فایل فایلی است که در زمان boot شدن سیستم توسط FreeBSD خوانده می شود و ماژول ها را قبل از بارگذاری شدن کامل هسته در هسته بارگذاری می کنید، این عملی است هوشمندانه به این دلیل که یکی از وظیفه های اصلی هسته مدیرت کردن دیسکها هستند.

در FreeBSD فایلی قرار دارد که در زمان راه اندازی شدن سیستم و قبل از راه اندازی کامل خوانده شده و دیسک های سیستم شما را در زیر شاخه ها مورد نظر به اصطلاح mount می کند تا سایر فرمان های راه انداز مثل سیستم rc بتواند به فایلها و فرمانهای راه اندازی سرویس ها دسترسی پیدا کنند، نام این فایل fstab است و در زیر شاخه etc/ قرار دارد و شما باید فرمان زیر را اجرا کنید تا خط مربوط به راه اندازی RIAD0 در آن ایجاد شود:

```
# mkdir /stripe
# echo "/dev/stripe/st0a /stripe ufs rw 2 2" \
>> /etc/fstab
```

در خط اول این فرمان شاخه ای انتخابی به نام stripe ایجاد شده تا به صورت خودکار هارد های raid0 بر زیر شاخه آن قرار گیرند.

راه اندازی RAID1 در FreeBSD

در بخش قبلی شما با RAID0 و مراحل راه اندازی، محاسن و معایب آن آشنا شده اید، در این بخش با نوعی دیگر از سیستم RAIDبه نام RAID1 یا mirroring آشنا می شوید. در این نوع از RAID اطلاعات برروی بیش از یک دیسک و به صورت کامل نوشته می شود و هر دیسک شامل اطلاعات کاملی از داده های شماست به این دلیل است که این نوع از radi را حالت full backupهم می نامند و د صورتی که یکی از دیسک ها شما در RAID1 از دسترس خارج شود در سرویس دهی شما اختلالی ایجاد نمی شود و مدیر سیستم می تواند هارد خراب را با هارد جدید تعویض کند و دوباره اطلاعات را در حالت RAID1ذخیره کند. در این نوع از RAID1سرعت نوشتن کند بود این علت که همزمان اطلاعات باید در بین چند دیسک کپی شود و لی سرعت خواندن بالاست چون سیستم عامل شما می تواند اطلاعات را از چند دیسک mirrorشده مطالعه کند.

دو حالت در این بخش توضیح داده می شود، یکی ایجاد کردن RAID1 در زمانی که شما قصد دارید دو دیسک جدید را با هم mirror کنید و در حالت شما mirror را از دیسکی می گیرید که اطلاعات برروی آن ذخیره شده است، این حالت هم زمانی مفید است که شما یکی از دیسک های خود را در RAID1 از دست داده اید و قصد دارید دیسک جدید به سیستم اضافه کنید.

پاک کردن Metadata دیسک:

قبل از شروع شما باید metadata ها موجود برروی هر دیسک را پاک کنید، این داده ها در پایان فضا هر دیسک ذخیره می شود و قبل از اینکه شما قصد راه اندازی RAID1 را داشته باشید باید پاک شود. در این مرحله شما با دو نوع metadataسرو کار دارید، metadataای که در نوع پارتیشن بندی GPT ایجاد می شود و metadata که از سیستم RAID1 یا همان mirroringبرروی دیسک شما ایجاد شده است.

پاک کردن در حالت GPT :

برای پاک کردن metadata بروری دو دیسک اصلی و پشتیبان در RAID1 که از GPTاستفاده می کنند شما از فرمان gpartمی توانید استفاده کنید برای این منظور فرمان زیر را اجرا کنید:

```
# gpart destroy -F ada8
```

خارج کردن دیسک از mirror و پاک کردن اطلاعات:

برای اینکه شما بتوانید یک دیسک را از سیستم RAID1 خارج کنید و اطلاعات metadataآنرا پاک کنید در قدم اول باید فرمان زیر را برای خارج کردن دیسک استفاده کنید:

```
# gmirror remove gm4 ada8
```

حال برای پاک کردن از فرمان زیر استفاده کنید:

```
# gmirror clear ada8
```

ایجاد کرد RAID1 برای دو دیسک جدید:

در این بخش فرض بر این است که در سیستم شما FreeBSD برروی دیسک ada0 نصب شده و شما دو دیسک جدید به نام های ada1 و ada2 دارید که قصد دارید این دو دیسک را با هم mirror کنید.

در قدم اول شما باید ماژول مربوط به mirroring را در هسته بارگذاری کنید، برای اعمال این تغییرات در زمان راه اندازی سیستم از فرمان kldload به صورت زیر استفاده کنید:

```
# gmirror load
```

در مرحله بعدی شما باید با استفاده از فرمان زیر دو دیسک مورد نظر خود را mirror کنید:

```
# gmirror label -v gm0 /dev/ada1 /dev/ada2
```

تا این مرحله حالت mirror ایجاد شده و بعد از ایجاد شده آن در زیر شاخه dev زیر شاخه دیگری به نام mirror ایجاد می شود که دیوایس ها در زیر شاخه آن قرار می گیرد.

هدف بعدی ما در این بخش این است که کل هاردی که FreeBSD برروی آن نصب است را به حالت mirror ایجاد شده منتقل کنیم. برای انجام دادن این انتقال در مرحله اول باید طرح پارتیشن بندی مورد نظر را با استفاده از gpartبرروی mirrorایجاد کنید برای مثال شما به پارتیشن های / و swap و /var و /tmp نیاز دارید، در طرح کلی شما می توانید فقط دو پارتیشن / و swap را ایجاد می کنیم:

```
# gpart create -s MBR mirror/gm0
# gpart add -t freebsd -a 4k mirror/gm0
# gpart create -s BSD mirror/gm0s1
# gpart add -t freebsd-ufs  -a 4k -s 2g mirror/gm0s1
# gpart add -t freebsd-swap -a 4k -s 4g mirror/gm0s1
# gpart add -t freebsd-ufs  -a 4k -s 2g mirror/gm0s1
# gpart add -t freebsd-ufs  -a 4k -s 1g mirror/gm0s1
# gpart add -t freebsd-ufs  -a 4k       mirror/gm0s1
```

حال باید هارد mirror شده را bootable کنید، برای این منظور باید از فرمان gpart به صورت زیر استفاده کنید و بعد هم بید پارتیشن active را تعیین کنید:

```
# gpart bootcode -b /boot/mbr mirror/gm0
# gpart set -a active -i 1 mirror/gm0
# gpart bootcode -b /boot/boot mirror/gm0s1
```

در بخش بعدی باید با استفاده از فرمان newfs پارتیشن های ایجاد شده را format کنید،این عمل را باید برای هر پارتیشن به صورت زیر اجرا کنید:

```
# newfs -U /dev/mirror/gm0s1a
# newfs -U /dev/mirror/gm0s1d
# newfs -U /dev/mirror/gm0s1e
# newfs -U /dev/mirror/gm0s1f
```

منتقل کردن اطلاعات از دیسک قدیمی به mirror

برای این انتقال می توانید از دو فرمان dump و restore استفاده کنید، در هر بخش باید پارتیشن مورد نظر را در زیر شاخه /mnt به اصطلاح mount کنید و از دو فرمان dump و restore استفاده کنید، این دو فرمان پایه گرفتن backup در FreeBSD است، در ادامه شما با فرمان های منتقل کننده آشنا می شوید:

```
# mount /dev/mirror/gm0s1a /mnt
# dump -C16 -b64 -0aL -f - / | (cd /mnt && restore -rf -)
# mount /dev/mirror/gm0s1d /mnt/var
# mount /dev/mirror/gm0s1e /mnt/tmp
# mount /dev/mirror/gm0s1f /mnt/usr
# dump -C16 -b64 -0aL -f - /var | (cd /mnt/var && restore -rf -)
# dump -C16 -b64 -0aL -f - /tmp | (cd /mnt/tmp && restore -rf -)
# dump -C16 -b64 -0aL -f - /usr | (cd /mnt/usr && restore -rf -)
```

انتقال کامل در فایل fstab:

برای منتقل کردن کامل ابتدا باید ماژول mirroring را در هسته FreeBSD در زمان راه اندازی و با استفاده از فایل loader.confانجام دهید برای انجام دادن این کار باید در فایل /mnt/boot/loader.conf خط زیر را اضافه کنید، به این نکته توجه کنید که باید پارتیشن mirrorشده را در زیر شاخه mnt که فعلا موقتی است mountکنید:

```
geom_mirror_load="YES"
```

حال باید برای بخش پایانی این قسمت فایل fstab را که در مسیر /mnt/etc/fstab قرار گرفته است را به صورت زیر ویرایش کنید تا در زمان راه اندازی سیستم به صورت خودکار پارتیشن های شما mount شود:

```
# Device          Mountpoint       FStype   Options  Dump     Pass#
/dev/mirror/gm0s1a        /                 ufs      rw       1        1
/dev/mirror/gm0s1b        none              swap     sw       0        0
/dev/mirror/gm0s1d        /var              ufs      rw       2        2
/dev/mirror/gm0s1e        /tmp              ufs      rw       2        2
/dev/mirror/gm0s1f        /usr              ufs      rw       2        2
```

راه اندازی RIAD03 در FreeBSD

اخرین نوعی از RIAD رو که شما می توانید در FreeBSD راه اندازی کنید RIAD03 است. در این مدل چند دیسک به صورت یک دیسک در می آید و یک دیسک به صورت parity عمل کرده و همه اطلاعات بین تمام دیسکها تقسیم شده تا در صورتی که یک دیسک از دسترس خارج شود اطلاعات موجود بتواند دیسک از دسترس خارج شده را دوباره بازسازی کرده. در این مدل شما به حداقل سه دیسک با سایزها و توان های یکسان نیاز دارید. تعداد دیسک های بیشتر به صورت 5، 9و 17 عدد است.

در سیستم عامل FreeBSD از graid3(8) برای پیاده سازی RIAD03 استفاده می شود و در قدم اول شما باید ماژول مورد نظر را در هسته FreeBSD بارگذاری کنید این عمل با استفاده از فرمان زیر قابل اجراست:

```
# graid3 load
```

و یا شما می توانید از فرمان kldload به صورت زیر استفاده کنید:

```
# kldload geom_raid3
```

در قدم بعدی شما باید شاخه مورد نظری را که قصد دارید از طریق آن به RIAD03 دسترسی داشته باشید را ایجاد کنید برای انجام این کار از فرمان mkdir به صورت زیر استفاده کنید:

```
# mkdir /multimedia
```

در مرحله بعد شما باید با استفاده از فرمان graid3 به صورت زیر 3 دیسک مورد نظر خود را به یک دیسک تبدیل کنید ، این فرمان در زیر بیان شده است:

```
# graid3 label -v gr0 /dev/ada1 /dev/ada2 /dev/ada3
Metadata value stored on /dev/ada1.
Metadata value stored on /dev/ada2.
Metadata value stored on /dev/ada3.
Done.
```

بعد از اجرا فرمان بالا یک دیسک به نام gr0 در زیر شاخه /dev/raid3 ایجاد شده است که شما می توانید با استفاده از فرمان gpart برروی آن پارتیشن ایجاد کرده و با استفاده از فرمن newfs فایل سیستمی برروی آن ایجاد کنید برای این بخش فرمانین مورد استفاده در زیر نمایش داده شده است:

```
# gpart create -s GPT /dev/raid3/gr0
# gpart add -t freebsd-ufs /dev/raid3/gr0
# newfs -j /dev/raid3/gr0p1
```

برای استفاده کردن از این پارتیشن باید آنرا mount کنید، به صورت زیر:

```
# mount /dev/raid3/gr0p1 /multimedia/
```

حال این پارتیشن در زیرشاخه ای که ایجاد کرده اید قابل دسترسی است.

برای mount شدن خودکار این پارتیشن در زمان راه اندازی شدن سیستم باید ماژول مورد نیاز را در زمان راه اندازی در هسته بارگذاری کنید و خط زیر را در فایل /boot/loader.conf قرار دهید:

```
geom_raid3_load="YES"
```

در گام بعدی باید فایل fstab موجود در زیر شاخه /etc قرار دارد را به صورت زیر ویرایش کنید تا در زمان راه اندازی سیستم به صورت خودکار این سیستم mount شود:

```
/dev/raid3/gr0p1 /multimedia      ufs      rw       2        2
```

راه اندازی کردن PXE در FreeBSD

تکنولوژی Preboot eXecution Environment (PXE) این اجازه را به شما می دهد تا بتوانید بدون داشتن هارد دیسک محلی سیستم عامل خود را راه اندازی کنید. برای مثال شما می توانید سیستم عامل FreeBSD را بدون داشتن هارد دیسک localاز طریق NFS راه اندازی کنید. سیستم PXE به ساختار سخت افزاری سیستم شما مربوط می شود و از طریق BIOS فعال می شود.

برای راه اندازی کردن سیستم از طریق PXE شما به سرورهای DHCP TFTP و سرور NFS نیاز دارید.در مرحله اول که سیستم شما راه اندازی می شود سیستم شما از طریق DHCP یک آدرس IP دریافت می کند و در قسمت بعدی سیستم عامل از طریق tftpبارگذاری می شود و در بخش پایانی تمام فایل سیستم شما از طریق NFS در اختیار سیستم شما قرار می گیرد. پس برای راه اندازی کردن این سیستم شما به سه سرور ذکر شده در شبکه خود نیاز دارید.

در این مقاله سعی می شود که روش پیاده سازی PXE در سیستم عامل FreeBSD را برای شما بیان کنم. این مقاله خود در دو بخش ارایه می شود.

راه اندازی کردن TFTPو NFS

در اولین قدم با روش پیکربندی و آماده سازی محیط PXE در FreeBSD آشنا می شوید. در اولین قدم شما باید دو سرویسی که در خود سیستم عامل FreeBSD وجود دارد به نام های TFTP و NFS را پیکربندی کنید، در بخش بعدی هم با پیکربندی کردن DHCP آشنا می شوید.

در ابتدای کار شما باید یک شاخه ای که کاربران PXE از فایلهای داخل آن استفاده می کنند را ایجاد کنید. برای مثال شاخه ای به آدرس /b/tftpboot/FreeBSD/install ایجاد کنید. ایجاد کردن این شاخه بسیار مهم است به این دلیل که در پیکربندی سرور از آن استفاده می کنید. این شاخه هم در فایل /etc/inted.conf و هم در فایل /usr/local/etc/dhcp.confمورد استفاده قرار می گیرد.

در اولین بخش باید شاخه ای که ایجاد کردید را به صورت زیر در nfs به اصطلاح mount کنید به صورت اجرا کردن فرمان زیر:

```
# export NFSROOTDIR=/b/tftpboot/FreeBSD/install
# mkdir -p ${NFSROOTDIR}
```

این شاخه باید دارای فایل های نصبی FreeBSD باشد که در ادامه این بخش روش ایجاد کردن آن را مشاهده می کنید.

در قدم بعدی شما باید سرور NFS را در بخش سرور خود فعال کنید به صورت زیر این کار با ویرایش کردن و اضافه کردن خط زیر در فایل /etc/rc.conf انجام می شود:

```
nfs_server_enable="YES"
```

در قدم بعدی باید شاخه ایجاد شده را به اشتراک گذارید، برای اشتراک گذاشتن فایل ها و پوشه ها در FreeBSD شما باید از فایل /etc/exports استفاده کنید، برای انجام این عمل خط زیر را در داخل فایل مذکور اضافه کنید:

```
/b -ro -alldirs
```

حال با استفاده از فرمان service سرور nfs خود را راه اندازی کنید:

```
# service nfsd start
```

راه اندازی کردن:tftp

برای راه اندازی کردن سرور tftp شما باید از سیستم قدیمی inted استفاده کنید، این سرور در گذشته برای راه اندازی کردن سرویس های مثل ftp و حتی ssh هم مورد استفاده قرار می گرفته است و در حال حاضر هم در سیستم وجود دارد. فایل اصلی آن /etc/inted.conf است و برای فعال سازی آن کافیست که خط زیر را در rc.conf به صورت زیر اضافه کنید:

```
inetd_enable="YES"
```

حال در بخش بعدی باید قسمت مربوط به راه اندازی سرور tftp را در فایل /etc/inted.conf به صورت زیر ویرایش کنید:

```
tftp dgram udp wait root /usr/libexec/tftpd tftpd -l -s /b/tftpboot
```

این فایل شامل خطوطی است که هر خط مربوط به یک سرور است، برای راه اندازی tftp شما کافیست که مسیر مورد نظر خود را در بخش اخر اضافه کنید، سرور tftpبه صورت پیش فرض udp است و در برخی از موارد برخی از ورژنهای PXE نیاز به برقرار کردن ارتباط از نوع TCP با سرور tftp دارند که باید شما خطی را فعال کنید که بخش tcp ان باشد.

نکته:

ابتدای هر خط علامت # یا همان comment قرار دارد که شما کافیست که فقط این علامت را از ابتدای خط که سرویس مورد نظر شماست پاک کنید و سرویس inted را راه اندازی کنید.

برای راه اندازی کردن سرویس inted که خودش باعث راه اندازی tftp می شود کافیست که فرمان زیر را در FreeBSD خود وارد کنید:

```
# service inetd start
```

برای ایجاد کردن فایلهای نصبی در شاخه ایجاد شده شما باید هسته سیستم عامل خود را rebuild کنید که این امر با استفاده کردن از اجرای فرمان زیر امکان پذیر است، به این نکته توجه کنید که شما باید سورس سیستم عامل خود را در زمان نصب FreeBSDنصب کرده باشید و یا به مقاله پیکربندی هسته در FreeBSD برای دریافت اطاعات بیشتر مراجعه کنید:

```
# cd /usr/src
# make buildworld
```

```
# make buildkernel
```

حال شاخه نصبی خود را که با استفاده از nfs برروی شبکه share کرده اید را با استفاده از فرمان زیر ایجاد کنید:

```
# make installworld DESTDIR=${NFSROOTDIR}
# make installkernel DESTDIR=${NFSROOTDIR}
# make distribution DESTDIR=${NFSROOTDIR}
```

مراحل اجرا شدن فرمان های بالا طولانی بوده و به سرعت سیستم شما بستگی دارد، هر بخش باید بدون خطا به پایان برسد.

برای چک کردن از صحت راه اندازی سرور tftp و فایلهایی که شما آنها را ایجاد کرده اید باید فرمان زیر را اجرا کنید و یک فایل را به صورت زیر دانلود کنید:

```
# tftp localhost
tftp> get FreeBSD/install/boot/pxeboot
Received 264951 bytes in 0.1 seconds
```

حال باید فایل fstab مربوط به کلاینتها را ویرایش کنید، به این نکته توجه کنید که در قدم اول وارد شاخه ای شوید که فایلهای نصبی را در آن قرار داده اید و در مرحله باید فایلی که در زیر شاخه /etc به نام fstab قرار دارد را ویرایش کنید، فایل اصلی سیستم خود را ویرایش کنید، در این مثال شاخه مورد نظر /b/tftpboot/FreeBSD/install می باشد در نتیجه فایل fstabدر آدرس زیر قرار دارد:

```
/b/tftpboot/FreeBSD/install/etc/fstab
```

این فایل را به صورت زیر ویرایش کنید:

```
# Device                          Mountpoint    FSType    Options   Dump Pass
server:/b/tftpboot/FreeBSD/install       /          nfs        ro          0    0
```

در قسمت server در خط بالا شما آدرس ip سرور nfs خودتون را وارد کنید.

هر کلانیت به رمزعبور کاربر root برای استفاده کردن از PXE نیاز دارد، این کاربر با کاربر root موجود در سیستم شما متفاوت است و باید در قسمت فایلهای راه اندازی که در شاخه /b/tftpboot/FreeBSD/install است رمزعبور کاربر root را تغییر دهید. برای این کار باید از فرمان chroot برای تغییر دادن شاخه ریشه خط فرمان خود به شاخه /b/tftpboot/FreeBSD/install استفاده کنید و بعد فرمان chpass را اجرا کنید، به صورت زیر می توانید از فرمان chroot استفاده کنید:

```
# chroot /b/tftpboot/FreeBSD/install/
# passwd
```

تا مراحل قبلی در قسمت راه اندازی سروری PXE با راه اندازی NFS وTFTP سرور و ایجاد کردن شاخه مورد نظر برای راه اندازی و بوت شدن سیستم ها در شبکه توضیحاتی ارایه شده است، در قسمت پایانی با راه اندازی کردن DHCP در خدمت شما دوستان هستم.

راه اندازی DHCPبرای PXEسرور:

سرور DHCP مثل دو سرور دیگر این مثال در FreeBSD به صورت پیش فرض نصب نبوده و شما در مرحله اول باید آنرا نصب کنید، شما به دو صورت می توانید این سرور را نصب کنید، برای به منظور به مقاله مقدمه ای بر نصب برنامه در FreeBSDمراجعه کنید، در این بخش از سیستم ports برای نصب استفاده کنید برای این منظور فرمانهای زیر را در اجرا کنید:

```
#cd /usr/ports/net/isc-dhcp43-server
#make
#make install
```

در صورتی که شما با خطایی در مراحل نصب مواجه نشده باشید بسته مورد نظر شما به درستی نصب شده است.

فایل پیکربندی سرور DHCP در مسیر /usr/local/etc قرار دارد البته نام این فایل متفاوت است و باید با استفاده از فرمان زیر نام فایل مورد نظر را به dhcp.conf تغییر دهید:

```
#cp /usr/local/etc/dhcp.conf.example /usr/local/etc/dhcp.conf
```

در این فایل باید بخش netx-server را به صورت زیر ویرایش کنید و بخش های filename و root-path را هم به صورت زیر تغییر دهید:

```
subnet 192.168.0.0 netmask 255.255.255.0 {
   range 192.168.0.2 192.168.0.3 ;
   option subnet-mask 255.255.255.0 ;
   option routers 192.168.0.1 ;
   option broadcast-address 192.168.0.255 ;
   option domain-name-servers 192.168.35.35, 192.168.35.36 ;
   option domain-name "example.com";

   # IP address of TFTP server
   next-server 192.168.0.1 ;

   # path of boot loader obtained via tftp
     filename "FreeBSD/install/boot/pxeboot" ;
```

```
  # pxeboot boot loader will try to NFS mount this directory for root FS
  option root-path "192.168.0.1:/b/tftpboot/FreeBSD/install/" ;

}
```

در بخش netx-server باید آدرس ip سرور TFPT تنظیم شده در بخش قبل را مشخص کنید.

در بخش filename باید مسیر راه انداز pxeboot را مشخص کنید، به این نکته توجه داشته باشید که /b/tftpboot از این مسیر حذف شده است.

بخش دیگری را هم شما باید پیکربندی کنید به نام root-path که شاخه root سرور NFS را مشخص می کند.

در این بخش سرور dhcp شما آماده راه اندازی است و شما باید در بخش اول در فایل rc.conf خط زیر را وارد کنید:

```
dhcpd_enable="YES"
```

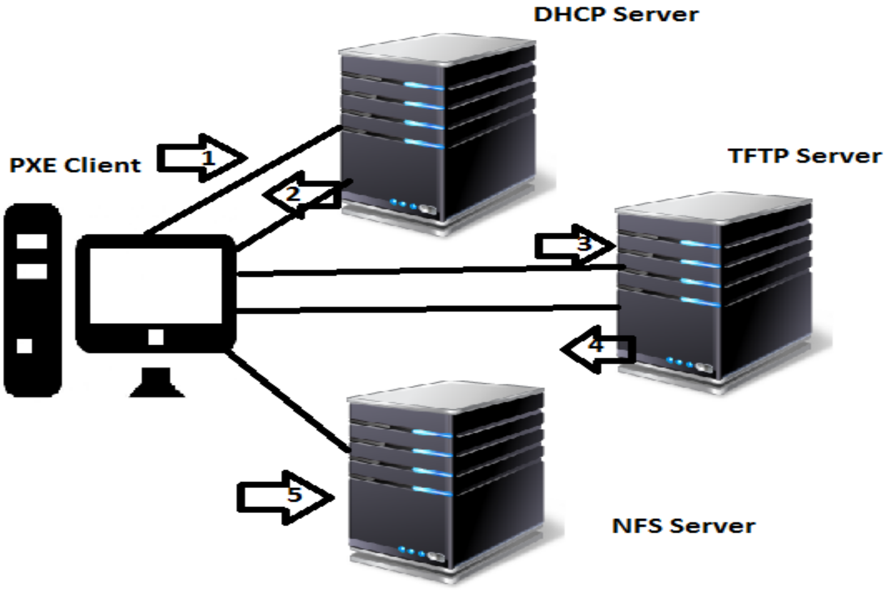
در بخش بعدی باید با استفاده از فرمان Service به صورت زیر سرور را راه اندازی کنید:

```
# service isc-dhcpd start
```

در پایان این بخش قسمت سرور تنظیم شده و آماده کار است حال باید شما در بخش client سیستم های مورد نظر خود را به صورتی تنظیم کنید که از طریق شبکه راه اندازی شود، هم باید کارت شبکه شما از این قابلیت پیشتیبانی کند و هم باید در Biosسیستم شما قابلیت راه اندازی از طریق شبکه را داشته باشد.

مراحل راه اندازی سیستم از طریق PXE:

در شکل زیر شما با مراحل راه اندازی شدن به تصویر کشیده شده است:



مراحل راه اندازی

در مرحله اول Client یک بسته با حالت پخش فراگیر به مدل DHCPDISCOVER برروی شبکه ارسال می کند تا سرور DHCpرا پیدا کردن تا آدرس TFTPسرور NFS و root-path را از سرور DHCP دریافت می کند.

سرور DHCP در مرحله دوم جواب درخواست سیستم Client را با مشخات تنظیم شده را ارسال می کند.

در مرحله سوم سیستم Client در خواستی به سرور TFTP ارسال می کند filename را دریافت کند.

در مرحله چهارم درخواست از سمت سرور TFTP ارسال می شود.

در این قسمت سیستم Client به وسیله سیستم pxeboot هسته FreeBSD را راه اندازی می کند و با استفاده از-root pathبه سرور NFS متصل می شود تا پارتیشن موجود را mount کند تا بتواند به کار خود ادامه دهد.

File Flag و kernel secure level در FreeBSD

یکی دیگر از امکانات پیشرفته سیستم عامل های مبتنی بر BSD استفاده کردن از یک سطح دسترسی است که جدای دو سطح دسترسی معمولی و ACL است. این سطح دسترسی به File Flag معروف است و شما می توانید علاوه بر دو روش ذکر شده سطح دسترسی مثل append-only یا undelete را برای افزایش دادن امنیت سیستم خود به فابلها اضافه کنید. در زیر شما لیست file flag ها موجود در FreeBSD را مشاهده می کنید:

Archived با این flag فایل حتما باید آرشیو شود.

Opaque این flag برروی شاخه اعمال می شود و در حالتی که شما شاخه ای را در زیر شاخه دیگر mount می کنید باعث مخفی شدن شاخه می شود. به این نوع از mount کردن به اصطلاح unionfs می گویند.

Nodump فایلی که این flag را داشته باشد به هیچ عنوان backup گرفته نمی شود.

Sappend هر فایلی که شامل این flag باشد فقط می توانید به آخر آن اضافه کنید و به هیچ عنوان قابل تغییر محتوای قبلی نیست.

Schange فایلی که شامل این flag باشد به هیچ عنوانی برای هیچ کاربری قابل تغییر دادن نیست.

Sunlink لینک کردن یک فایل یکی از قابلیت های موجود در BSD است شما می توانید بعد از این کار flag مذکور را به فایل اضافه کنید تا کسی نتواند آنرا از حالت link خارج کند.

Uappend فایلی که شامل این flag باشد فقط توسط صاحب آن فایل قابل اضافه کردن است و دیگران به آن دسترسی ندارند.

Uchange فایلی که شامل این flag باشد فقط توسط صاحب آن فایل قابل تغییر دادن است و دیگران به آن دسترسی ندارند.

Uunlink فایلی که شامل این flag باشد فقط توسط صاحب آن فایل قابل تغییر دادن link و دیگران به آن دسترسی ندارند.

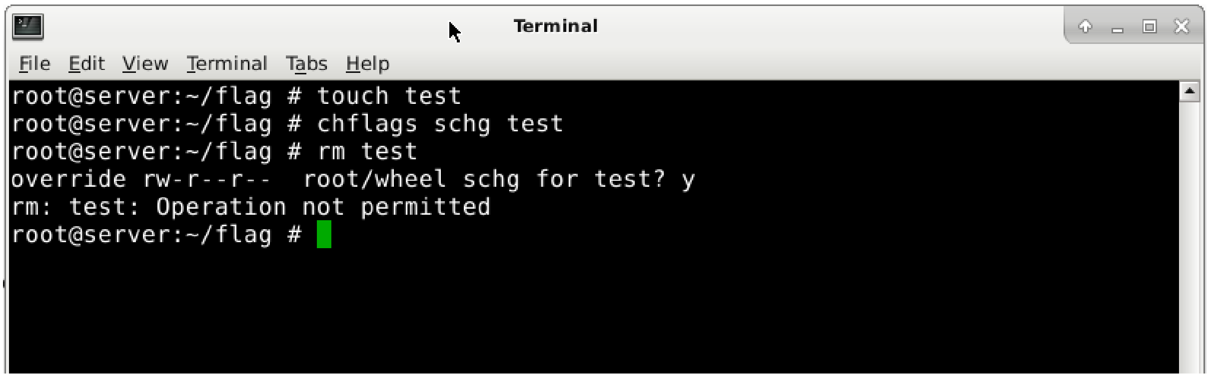
در لیست بالا آن دسته از flag هایی که با حرف s شروع می شود فقط توسط کاربر root قابل تغییر است.

پیشنهاد ما در استفاده کردن از flag ها به این صورت است:

- باید سعی کنید که برای فایلهای log که بخش مهمی در کشف کردن حملات و تغییرات در سیستم شما هستند از sappendاستفاده کنید.
- سعی کنید که فایلهای مهم پیکربندی سیستم را در حالتی قرار دهید که قابل تغییر نباشد. حتی در بخش بعدی شما یاد خواهید گرفت که به چه صورت دسترسی کاربر root را هم تغییر دهید تا در صورتی که سیستم شما به خطر افتاد و هکری با کاربر root به سیستم شما وارد شد نتواند فایلهای پیکربندی اصلی را که امنیت سیستم شما را حفظ می کند تغییر دهد.

اعمال کردن file flag در FreeBSD

برای اعمال کردن file flag در FreeBSD شما باید از یک فرمان خاص به نام chflags استفاده کنید، در شکل زیر یک مثال از این بخش را مشاهده می کنید:



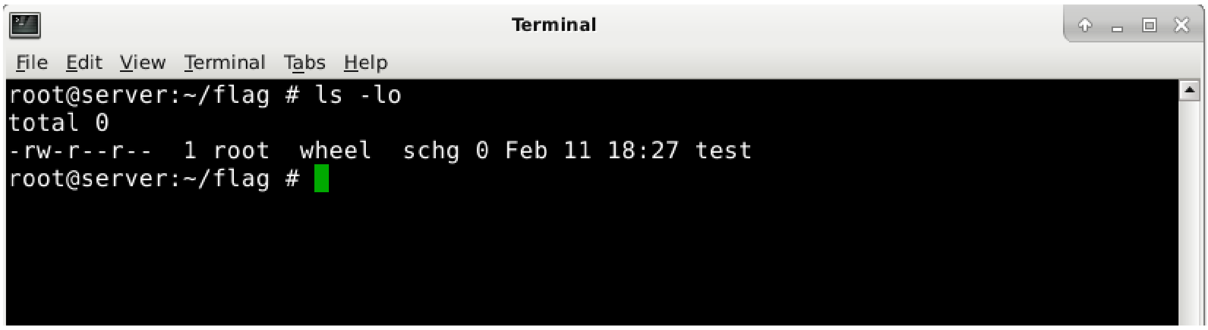
مثالی از استفاده از فرمان schg

در خط اول این بخش با استفاده از فرمان touch یک فایل ایجاد شده است.

در خط دوم با استفاده از فرمان chflags به فایل ایجاد شده حالت بدون تغییر را اضافه کردیم.

همانطور که مشاهده می کنید دسترسی در shell با کاربر root است و در خط بعدی قصد داریم که فایل را با فرمان rm با سطح دسترسی کاربر root حذف کنیم که این عمل امکانپذیر نیست.

با استفاده از فرمان بالا و flag مشاهده کردید که می توانید سطح دسترسی جدید به فایلها اضافه کنید، برای مشاهده کردن file flagشما باید از فرمان ls به سویچ های lo– استفاده کنید که در شکل زیر خروجی این فرمان را در شاخه قبلی که شما فایل در این ایجاد کرده اید را مشاهده می کنید:



نمایش وضعیت flag

همانطوری که مشاهده می کنید در فیلد بعد از مشخص شدن گروه صاحب فایل شما file flag اعمال شده به نام schg را مشاهده می کنید.

securelevel در FreeBSD

هدف اصلی از بیان file flag در FreeBSDبیان کردن قابلیت امنیتی اضافه شده به سیستم عامل های مبتنی بر BSD است به نام security profiles یا همانsecurelevel ..

هسته سیستم عامل های مبتنی بر BSD به صورتی طراحی شده اند که می توانند حتی دسترسی کاربر root را هم محدود کنند تا سطح امنیتی سیستم شما را افزایش دهند. برای این منظور پنج سطح مختلف از securelevel در FreeBSD در نظر گرفته شده است که هر کدام در ادامه توضیح داده شده است:

سطح -1:

در این سطح هیچ بخش امنیتی برروی سیستم شما اعمال نمی شود و به صورت پیش فرض هم از این سطح استفاده می شود که هیچ تغییر امنیتی در سطر هسته برروری سیستم شما فعال نیست.

سطح 0:

در این سطح اگر کاربری file flag های غیر فایل تغییری به فایلها اضافه کرده باشد می توان در این سطح آنها را خاموش کند. این نوع از flag ها فقط در سطح 0 اعمال می شود و شما اگر سطح 1- را فعال کنید این flag ها غیر فعال می شوند. در این سطح هم همه دستگاه ها قابل خواندن و نوشتن هستند با توجه به سطح دتسرسی مشخص شده.

سطح 1:

در این سطح اگر کاربری file flag های غیر فایل تغییری به فایلها اضافه کرده باشد **<u>نمی توان</u>** در این سطح آنها را خاموش کند.

فایل سیتم های /dev/mem, /dev/kmem و dev/io اگر در ساختار سخت افزاری شما وجود داشته باشد، در این سطح از امنیت غیر قابل mountکردن است.

یکی از قابلهایی که از کاربر در این سطح گرفته می شود این است که دیگر نمی توان در هسته FreeBSD ماژولی با استفاده kldloadبارگذاری کرد و یا ماژول های بارگذاری شده را با استفاده از فرمان kldunload از حالت بارگذاری خارج کرد.

سطح2:

این سطح علاوه براینکه قابلیتهای سطح 1 را شامل می شود قابلتهای دیگری به صورت زیر در هسته سیستم عامل روش می شود:

در این سطح دیسک ها برای خواندن و نوشتن نمی توانند mount شوند،

فرمان newfs حتی اگر سیستم شما در حالت multi-user محدود شده و دیگر دیسک جدید را شما نمی تواند فرمت کنید و دیسک های قبلی را که در سیستم شما هست غیر قابل پاک شدن با استفاده از فرمان newfs است.

این سطح از securelevel برروی زمان در هسته اعمال می شود و شما نمی توانید زمان هسته را دستکاری کنید.

سطح 3:

این سطح از securelevel به سطح امنیتی شبکه ای معروف است که سطح پایانی در بالا بردن سطح امنیتی سیستم شما که علاوه بر تمام قابلتهای سطح قبلی خود برروی تنظیمات فایروایل سیستم شما تاثیر می گذارد و فارغ از هر فایروالی که شما استفاده می کنید رولهای آن دیگر قابل تغییر نخواهد بود.

securelevel با استفاده از فرمان init قبل از راه اندازی شدن سیستم در حالت multi-user اجرا و اعمال می شود.

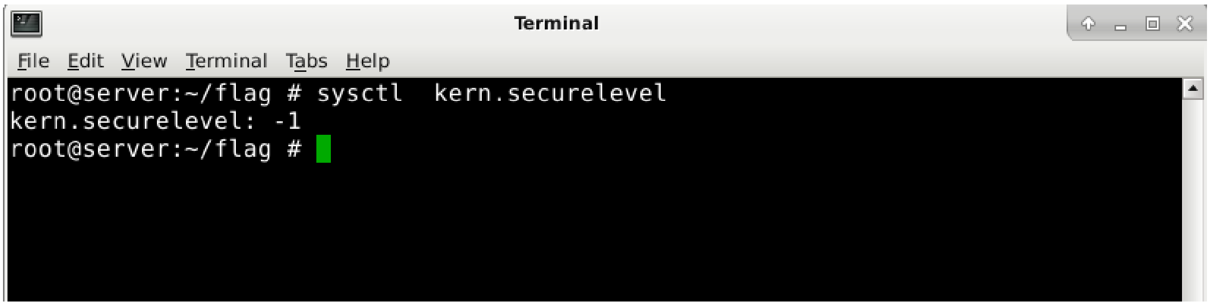
به این نکته توجه کنید که کاربر root یا همان super-user فقط می تواند سطح را افزایش دهد.

کاهش دادن securelevel در زمان راه اندازی شدن سیستم حتی توسط کاربر root هم امکان پذیر نیست.

اگر تغییرات securelevel را بصورت دایمی در سیستم خود اعمال نکرده باشید، که در بخش بعدی روش آن توضیح داده می شود شما با راه اندازی مجدد سیستم می توانید آنرا پایین آورید، ولی اگر به صورت خودکار و در زمان راه اندازی توسط فایل rc.conf تغییر دادن securelevel را اعمال کرده باشید فقط باید در حالت single user mod وارد شوید و خط مربوطه را از فایل rc.conf پاک کنید.

نمایش و تغییر دادن Live سطح securelevel

برای نمایش دادن سطح خاضر سیستم شما می توانید از فرمان sysctl به صورت اجرا شده در شکل زیر استفاده کنید:



نمایش سطح هسته فعلی سیستم

برای تغییر دادن این سطح کافیست از فرمان زیر استفاده کنید:

```
# sysctl kern.securelevel=2
```

با استفاده از فرمان بالا شما securelevel را به سطح 2 افزایش داده اید، حال در شکل پایین مشاهده می کنید که حتی کاربر root هم نمی تواند سطح securelevel را به سطح پایین تر تغییر دهد:

```
root@server:~/flag # sysctl kern.securelevel=2
kern.securelevel: -1 -> 2
root@server:~/flag # sysctl kern.securelevel=1
kern.securelevel: 2
sysctl: kern.securelevel=1: Operation not permitted
root@server:~/flag #
```

نمایش عدم تغییر در امنیت هسته توسط کاربر root

تغییر دادن دایمی securelevel

برای انجام دادن این کار کافیست که از فایل rc.conf استفاده کنید و خطوط زیر را در این فایل اضافه کنید:

```
kern_securelevel_enable="YES"

kern_securelevel="2"
```

با استفاده از این دو خط سطح securelevel به سطح 2 تغییر پیدا می کند.

اشتراک گذاشتن دسترسی root با کاربر sudo و su

در پروژه BSD بسیار تاکید شده است که از کاربر root برای انجام دادن فعالیت های مدیریتی در سیستم استفاده نکنید، در راستای همین هدف و تفکر به شما اجازه ورود کاربر root را از طریق ssh به سیستم را نمی دهد مگر اینکه شما آنرا تنظیم کنید. برای کمک کردن به شما دو راه را در اختیار شما قرار داده است در روش اول استفاده کردن از فرمان su است که یک روش قدیمی است و در روش دوم استفاده کردن از فرمان sudo است. در ادامه با دو روش آشنا می شوید.

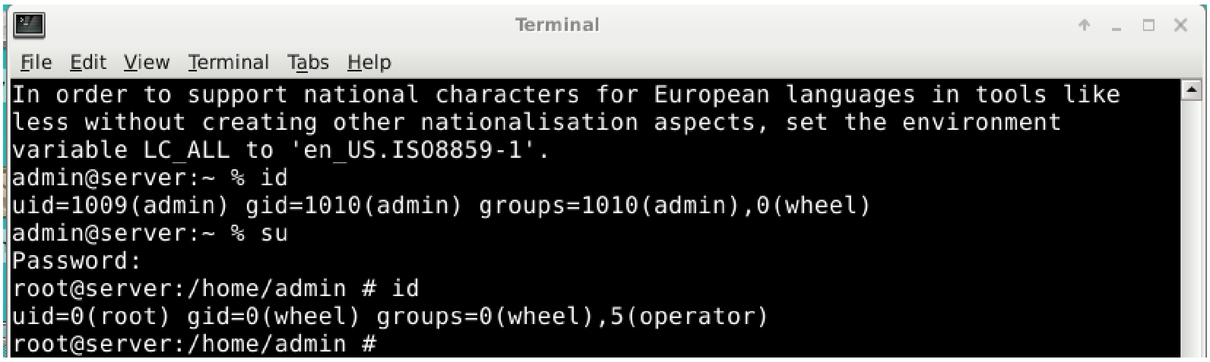
فرمان su

در FreeBSD و سایر سیستم عامل های مبتنی بر BSD گروه کاربری به صورت پیش فرض وجود دارد به نامwheel ، این گروه یک گروه مهم و کاربردی است. هر کاربری که عضو این گروه باشد می تواند با فرمان su سطح دسترسی خود را به کاربر rootارتقاع دهد. برای این کار کاربری که اجازه دارد از این فرمان استفاده کنید باید رمز عبور کاربر root را در اختیار داشته باشد.

برای اضافه کردن کاربر مورد نظر شما به گروه wheel باید از فرمان pw به صورت زیر استفاده کنید:

```
# pw usermod admin -G wheel
```

با استفاده از فرمان بالا گروه admin به گروه wheel اضافه می شود. حال این کاربر می تواند از فرمان su استفاده کند و در صورتی که رمز کاربر root را داشته باید سطح دسترسی خود را به کاربر root برساند، این عمل در شکل زیر نمایش داده شده است:



نمایش وضعیت یک کاربر

فرمان id برای نمایش داده وضعیت کاربر جاری شماست، در زمانی که شما کاربر Admin هستید خروجی فرمان id علاوه بر user IDکاربر شما نمایش می دهد که این کاربر علاوه بر گروه admin عضو گروه wheel هم هست، در نتیجه این کاربر می تواند از فرمان su استفاده کند، بعد از اجرا این فرمان شما وارد خطی می شوید که باید رمز عبور کاربر root را وارد کنید و بعد از وارد کردن صحیح رمزعبور شما وارد سطح دسترسی root می شود و همانطوری که مشاهده می کنید خروجی فرمان idمتفاوت با دفعه قبلی می باشد.

نکات فرمان :su

در اجرا کردن فرمان su باید این نکته را بدانید که کاربر root می تواند به سطح دسترسی هر کاربری بدون داشتن رمزعبور آن کاربر برسد کافیست که بعد از فرمان su نام کاربر مورد نظر را وارد کند.

در زمانی که شما از فرمان su استفاده می کند متغییر های شما در Shell تغییر نمی کند برای اعمال این تغییر باید از سویچ افرمان استفاه کنید تا به صورت کامل مراحل ورود به سیستم برای شما شبیه سازی شود، این بخش در فرمان زیر نمایش داده شده است.

```
Terminal
File Edit View Terminal Tabs Help
root@server:~ # su admin
admin@server:/root %
```

دسترسی کاربر root برای وارد شدن به هر کاربری با su

در شکل بالا su بدون سویچ اجرا شده است، اما در شکل زیر فرمان su با سویچ l اجرا شده است.

```
Terminal
File Edit View Terminal Tabs Help
root@server:~ # su -l admin
Can't remember if you've installed a certain port or not? Try "pkg info
-x port_name".
admin@server:~ %
```

استفاده از سویچ l در Su

یکی دیگر از قابلیت های فرمان su این است که به کاربر root این اجازه را می دهد که login class مختلف را تست کند، برای اینکار کافیست که از سویچ c– و نام آن login classاستفاده کنید.

فرمان :sudo

روش استفاده از فرمان بالا یک ایراد کلی دارد و آن این است که کاربر به صورت کامل به سطح دسترسی کاربر root ارتقاع داده می شود و رمز عبور کاربر root هم در دسترس کاربری قرار می گرد که از فرمان su استفاده می کند، راه دیگری در سیستم عامل های مبتنی بر BSD در نظر گرفته شده است به نام sudo که در بخش بعدی با آن آشنا می شوید.

در بعضی از موارد شما به صورت تیمی کار مدیریت سیستم خود را انجام می دهد، برای این کار دیگر نمی شود از روش su استفاده کنید، افراد گروه شما می تواند دارای دسترسی کاربر معمولی باشند با رمزعبور خود کار مورد نظر را انجام دهد و شما هم به عنوان یک مدیر سیستم می تواند تمرکز بیشتر و بهتری برروی دسترسی های سیستم خود داشته باشید. در این بخش شما با این فرمان و روش استفاده از آن آشنا می شوید.

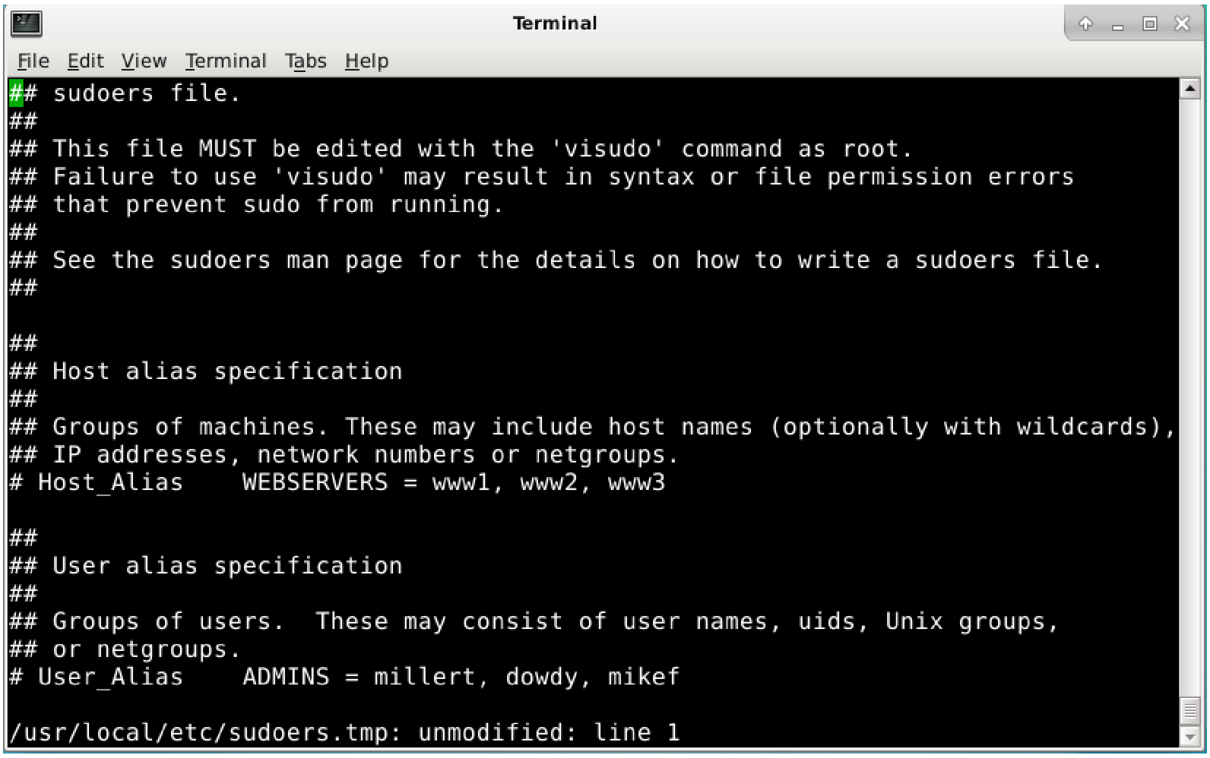
برای استفاده کردن از sudo ابتدا باید آنرا نصب کنید هم از طریق نصب بسته های باینری این کار را کنید هم از طریق سیستم پورت، اگر از سیستم پورت استفاده می کنید باید وارد شاخه زیر شوید به صورت زیر:

```
# cd /usr/ports/security/sudo/
#make install clean
```

برای نصب کردن از سیستم بسته از فرمان زیر استفاده کنید:

```
# pkg install sudo
```

به دو روش بالا برنامه sudo نصب می شود. بعد از نصب برای ویرایش کردن فایل اصلی برنامه sudo باید از برنامه visudo استفاده کنید، بعد از اجرا کردن این برنامه فایل پیکربندی برنامه به صورت زیر باز می شود:



```
## sudoers file.
##
## This file MUST be edited with the 'visudo' command as root.
## Failure to use 'visudo' may result in syntax or file permission errors
## that prevent sudo from running.
##
## See the sudoers man page for the details on how to write a sudoers file.
##

##
## Host alias specification
##
## Groups of machines. These may include host names (optionally with wildcards),
## IP addresses, network numbers or netgroups.
# Host_Alias     WEBSERVERS = www1, www2, www3

##
## User alias specification
##
## Groups of users.  These may consist of user names, uids, Unix groups,
## or netgroups.
# User_Alias     ADMINS = millert, dowdy, mikef

/usr/local/etc/sudoers.tmp: unmodified: line 1
```

نمایش فایل پیکربندی sudo

مانطوری که مشاهده می کنید این فرمان فایل پیکربندی را با استفاده از ویرایشگر vi باز می کند.

نکته

برای راحتی در استفاده از این برنامه به شما دوستان پشنهاد می کند که فیلم آموزشی روش استفاده از ویرایشگر vi را مشاهده کنید

روش استفاده کردن از این فایل پیکربندی بسیار ساده و راحت است، خطوطی که با # شروع می شوند توضیحات هستند و با مطالعه آنها می توانید به راحتی از این فایل استفاده کنید، در ادامه با روش اضافه کردن سطح دسترسی خاص خود را در این فایل آشنا می شوید.

برای شروع ایجاد کردن یک سطح دسترسی کافیست که در ابتدا نام کاربر مورد نظر که قصد دارید به ان اختیار اجرای فرمان را دهید وارد کنید و در سطح بعدی باید دسترسی آنرا بیان کنید به صورت زیر . ALL=(ALL) در بخش بعدی باید فرمانی را که قصد اجرا آنرا دارید بیان کنید خط زیر یک مثالی است که شما به کاربر test اجازه انجام کلیه امور ftpd را به آن می دهید:

```
test   ALL=(ALL)       /usr/sbin/service ftpd *
```

برای اجرا کردن این فرمان توسط کاربر کافیست که در ابتدا و شروع فرمان از suod استفاده شود توسط کاربر test به صورت زیر:

```
% sudo /usr/sbin/service ftpd  start
```

مانیتور کردن وضعیت سیستم با iostat

یکی از بخشهای مهم در مدیریت کردن سرور های خود مانیتور کردن وضعیت I/O سیستم شماست که به شما میزان بار موجود برروی سیستم شما را معین می کند، در سیستم عامل FreeBSD چندین ابزار برای مشاهده کردن وضعیت سیستم وجود دارد به نام های که به بخش stat ختم می شوند مثل iostat , systat,gstat,vmstat,nfstat و غیره. در این بخش توضیحات کوتاهی در مورد هر یک از فرمان ها برای شما ارایه می شود.

فرمان iostat:

این فرمان وضعیت i/o دیسک های سیستم شما را در لحظه نمایش می دهد و قابلیت هایی دارد که با آن آشنا می شوید، اگر این فرمان را یکبار اجرا کنید برای شما یک خروجی پیشفرض در یک خط نمایش می دهد، این خروجی به صورت زیر است:

```
Terminal
File  Edit  View  Terminal  Tabs  Help
root@server:~ # iostat
       tty             da0              da1              cd0            cpu
 tin  tout  KB/t tps  MB/s   KB/t tps  MB/s   KB/t tps  MB/s  us ni sy in id
   1    35 35.18   1  0.03   6.28   0  0.00   1.69   0  0.00   0  0  0  0 99
root@server:~ #
```

خروجی فرمان iostat

این فرمان برای نمایش وضعیت i/o دستگاهایی که به سیستم شما متصل است استفاده می شود و به صورت پیش فرض 5 دستگاه اول سیستم شما را نمایش می دهد مگر اینکه شما دستگاه خاصی را در نظر داشته باشید، طول خروجی این فرمان هم به صورت پیش فرض 80 ستون است. در بخش های بعدی با خروجی این فرمان آشنا می شوید.

هر ستونی که نام دستگاه بر روی آن قرار دارد آخرین ستونی است که اطلاعات آن دستگاه در آن بخش نمایش داده شده است برای مثال بخش tty شامل دو ستون است که tty برروی آخرین ستون اطلاعات این بخش قرار دارد، در ادامه با خروجی این فرمان بیشتر آشنا می شوید،

بخش اول این فرمان به نمایش وضعیت tty یا همان ترمینال سیستم شما می پردازد که تنها بخشی است که در دو ستون نمایش داده می شود و مقادیری که نمایش می دهد در دو قسمت tin و tout است:

بخش tin: تعداد کاراکترهای خوانده شده از ترمینال را نمایش می دهد.

بخش tout: تعداد کاراکترهای نوشته شده در ترمینال را نمایش می دهد.

در بخش بعدی که نام دستگاه هایی که i/o هستند را نمایش می دهد اگر از فرمان iostat به صورت پیش فرض استفاده کنید این خروجی در سه بخش به صورت زیر تقسیم بندی می شود:

KB/tبه معنی kilobytes per transfer

tps به معنی transfers per second

MB/s به معنی megabytes per second

این خروجی می تواند حالت های مختلفی به خود بگیر مثلا اگر از سویچ I حرف i بزرگ استفاده کنید) خروجی این فرمان اطلاعات زیر را نمایش می دهد:

KB/tبه معنی kilobytes per transfer

xfrsبه معنی total number of transfers

MBبه معنی total number of megabytes transferred

وضعیت نمایش CPU سیستم شما هم در بخش آخر از این فرمان نمایش داده می شود که با بخش های دیگر متفاوت بوده و خروجی آن شامل اطلاعات زیر می شوید:

usبه معنی % of cpu time in user mode

niبه معنی % of cpu time in user mode running niced processes

syبه معنی % of cpu time in system mode

inبه معنی % of cpu time in interrupt mode

idبه معنی % of cpu time in idle mode

مدیریت کردن خروجی فرمان :iostat

شما با استفاده کردن از سویچ های مختاف می توانید حالت ها خروجی این برنام را متفاوت کنید در ادامه با سویچ های مورد نیاز دز این فرمان آشنا می شوید:

سویچ : c-

به صورت پیش فرض زمانی که شما فرمان iostat را اجرا می کنید فقط یک خط خروجی برای شما نمایش می دهد، برای اینکه به تعداد دلخواه خود بتوانید خروجی را مشاهده کنید بعد از سویچ c تعداد وارد کنید برای مثال در شکل زیر مشاهده می کنید که 5 بار این خروجی برای شما نمایش داده شود:

```
Terminal
File  Edit  View  Terminal  Tabs  Help
root@server:~ # iostat -C x
       tty            cpu
 tin  tout us ni sy in id
   1    31  0  0  0  0 99
root@server:~ #
```

نمایش خروجی C– در فرمان iostat

سویچ d-

برای حذف کردن دو بخش tty و cpu و نمایش فقط وضعیت دستگاه های متصل به سیستم شما باید از سویچ d استفاده کنید به صورت زیر:

```
Terminal
File  Edit  View  Terminal  Tabs  Help
root@server:~ # iostat -d
             da0               da1               cd0             pass0
  KB/t tps  MB/s   KB/t tps  MB/s   KB/t tps  MB/s   KB/t tps  MB/s
 12.80   3  0.03   6.11   0  0.00   1.69   0  0.00   0.00   0  0.00
root@server:~ #
```

نمایش خروجی d– در فرمان iostat

سویچ : n-

برای محدود کردن تعداد 5 عدد دستگاه نمایش داده شده به تعداد دلخواه شما از سویچ n و بعد تعداد مورد نظر خود استفاده کنید.

سویچ : t-

شما می توانید خروجی سیستم خود را به تناسب نوع دستگاه ها طبقه بندی کنید، برای مثال فقط برای نمایش داده دستگاه هایی که از نوع SCSI هستند بعد از t– باید نوع SCSIرا تایپ کنید در زیر لیستی از نوع هایی این فرمان را مشاهده می کنید:

da	Direct Access devices
sa	Sequential Access devices
printer	Printers
proc	Processor devices

worm Write Once Read Multiple devices

cd CD devices

scanner Scanner devices

optical Optical Memory devices

changer Medium Changer devices

comm Communication devices

array Storage Array devices

floppy Floppy devices

از نظر نوع رابط می توانید از کلمات زیر استفاده کنید:

IDE Integrated Drive Electronics devices

SCSI Small Computer System Interface devices

other Any other device interface

در شکل زیر روش استفاده از این فرمان را مشاهده می کنید برای حذف کردن cpu و tty از سویچ d– استفاده کنید:

```
Terminal
File  Edit  View  Terminal  Tabs  Help
root@server:~ # iostat -d -t IDE
             cd0
  KB/t tps  MB/s
  1.69   0  0.00
root@server:~ # iostat -d -t SCSI
             da0              da1
  KB/t tps  MB/s    KB/t tps  MB/s
 12.83   3  0.03    6.11   0  0.00
root@server:~ #
```

خروجی های خاص فرمان iostat

همانطوی که در خروجی فرمان بالا مشاهده می کنید با استفاده از سویچ t می توانید خروجی isostat را بر اساس نوع طبقه بندی کنید.

سویچ -w

برای تعیین کردن زمان مکث در بین هر خط از خروجی باید از سویچ w− و مقدار مکث به ثانیه تعیین کنید ، این سویچ باید با سویچ cبرای نمایش تعداد خروجی ها استفاده شود.

سویچ -x

برای نمایش هرچه بهتر وضعیت دیسک ها در حالت گسترده از سویچ x استفاده کنید، در صورتی که تعداد دیسک های سیستم شما زیاد باشد و همه در خروجی فرمان قرار نگیرند بهترین گزینه استفاده کرده از این مدل خروجی است که هر دیسک در یک خط افقی قرار می گیرد، نمایش این خروجی را در شکل زیر مشاهده می کنید:

Terminal

File Edit View Terminal Tabs Help

```
root@server:~ # iostat -x
                        extended device statistics
device     r/s   w/s    kr/s    kw/s qlen svc_t  %b
da0        2.1   0.4    18.4    13.1    0   3.8   0
da1        0.0   0.0     0.1     0.0    0   1.0   0
cd0        0.0   0.0     0.0     0.0    0   0.4   0
pass0      0.0   0.0     0.0     0.0    0   0.0   0
pass1      0.0   0.0     0.0     0.0    0   0.0   0
pass2      0.0   0.0     0.0     0.0    0   0.0   0
root@server:~ #
```

سویچ x در فرمان iostat

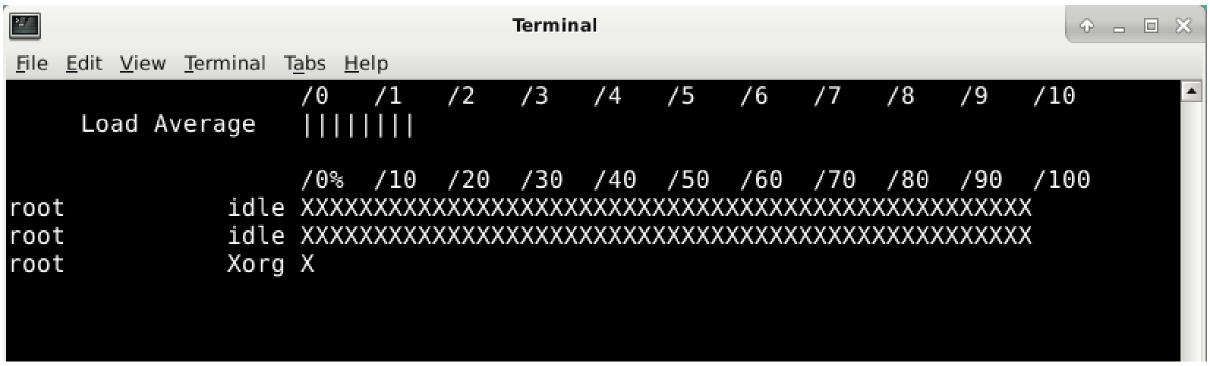
فرمان systat در FreeBSD

در سیستم عامل FreeBSD چندین فرمان برای نمایش وضعیت سیستم وجود دارد که فرمان پایه ای آن iostat بود که در مقاله قبلی با آن آشنا شدید، فرمان دیگری را در این بخش برای شما معرفی می کنم که به صورت کامل تمام وضعیت ها و فرمان های stat را پوشش می دهد، اسم این فرمان systat است که در این مقاله روش استفاده کردن از این فرمان را آموزش خواهید دید.

این مقاله شامل بخشهای زیر است:

- بخش pigs برای نمایش میزان مصرف پردازش هر کاربر.
- بخش ifstat نمایش وضعیت کارت های شبکه سرور شما.
- بخش icmp برای نمایش وضعیت icmp سرور شما.
- بخش ip نمایش وضعیت پروتکل ip سرور شما.
- بخش TCP برای نمایش وضعیتTCP سرور شما.
- بخش iostat برای نمایش وضعیتIO سرور شما.
- بخش swap برای نمایش وضعیت swap در سرور شما.
- بخش netstat برای نمایش وضعیت پورتهای باز و ارتباطات باز سرور شما.

زمانی که این فرمان را اجرا می کنید شما وارد صفحه اصلی این برنامه می شوید که به دو بخش تقسیم شده است، خروجی این فرمان را در شکل زیر مشاهده می کنید:



خروجی فرمان systat بدون هیچ فرمان اضافه ای

این فرمان از تلکنولوژی ncurses برای طبقه بندی کردن صفحه نمایش استفاده می کند، در بخش بالایی این خروجی load سیستم به صورت bar برای شما نمایش داده می شود و در بخش پایینی این فرمان شما می توانید اطلاعات انتخابی مورد نظر خودتون را برای نمایش انتخاب کنید.

برای وارد شدن به بخش اطلاعاتی که این فرمان می تواند برای شما نمایش دهد کافیست که از علامت : استفاده کنید و بعد کلمه help را وارد کنید و به محض اجرا help با استفاده از کلید Enterدر قسمت پایینی این فرمان خطی به صورت زیر برای شما نمایش داده می شود:

```
pigs swap iostat vmstat netstat icmp ip icmp6 ip6 sctp tcp ifstat
```

بخشهای قابل فعال در systat

12بخش قابل نمایش برای شما در این فرمان وجود دارد که شما می توانید به تناسب نیاز خود هر کدام را اجرا کنید و از اطلاعات خروجی آن استفاده کنید، در ادامه با بخش های اجرای مختلف این فرمان آشنا می شوید.

قبل از شروع:

قبل از شروع کار با این فرمان باید با گزینه های موجود در این فرمان آشنا شوید. برای وارد شدن به محیط فرمان این برنامه باید از کلید : استفاده کنید، بعد از اجرا : در زیر این صفحه شما علامت : را مشاده می کنید که فرمان های بعدی را در این بخش باید وارد کنید، بعد از اجرا فرمان شما وارد برنامه می شوید، برای خارج شدن از این برنامه کافیست که ابتدا وارد بخش فرمان شوید و بعد حرف q را وارد کنید و بعد Enter کنید.

این برنامه به صورت پیش فرض هر 5 ثانیه اطلاعات نمایش را برای شما refresh می کند، برای تغییر دادن این مقدار کافیست که در حالت فرمان وارد شوید و بعد عددی را وارد کنید که به ثانیه است و Refresh کردن پیش فرض را تغییر می دهد. بعد از وارد کردن عدد مورد نظر در زیر صفحه برنامه خط زیر را مشاهده می کنید:

```
Showing pigs, refresh every 1 seconds.
```

تغییر دادن زمان Refresh

این خط به شما اعلام می کند که بخش pigs هر 1 ثانیه refresh می شود.

برای متوقف کردن refresh کافیست که در بخش فرمان stop را تایپ کنید و برای راه اندازی مجدد آن کافیست که از فرمان startو عددی به ثانیه برای refresh کردن وارد کنید.

برای نمایش کردن load average سیستم خود هم کافیست که فرمان load را وارد کنید تا به صورت شکل زیر شما load averageسیستم خود را در بخش فرمان مشاهده کنید:

```
0.3  0.6  0.6
```

نمایش load average

برای نمایش همه فرمان ها در این بخش باید از فرمان help استفاده کنید بعد از وارد شدن به بخش فرمان گزینه help را وارد کنید تا اطلاعات به صورت زیر برای شما نمایش داده شود:

```
pigs swap iostat vmstat netstat icmp ip icmp6 ip6 sctp tcp ifstat
```

بخشهای قابل فعال در systat

برای اجرا کردن هر کدام از این بخش ها که برای شما سیستم مانیتوریگ جداگانه ای باز می کند هم می توانید به صورت مستقیم آنرا وارد کنید و یا بعد از - انرا تایپ کنید.

بخش pigs

این بخش به صورت پیش فرض در زمان راه اندازی فرمان اجرا می شود، هدف اصلی این برنامه نمایش دادن پردازش هایی است که در حافظه فعال قرار دارد ، این بخش فرمان های idle را نمایش نمی دهد.

بخش ifstat

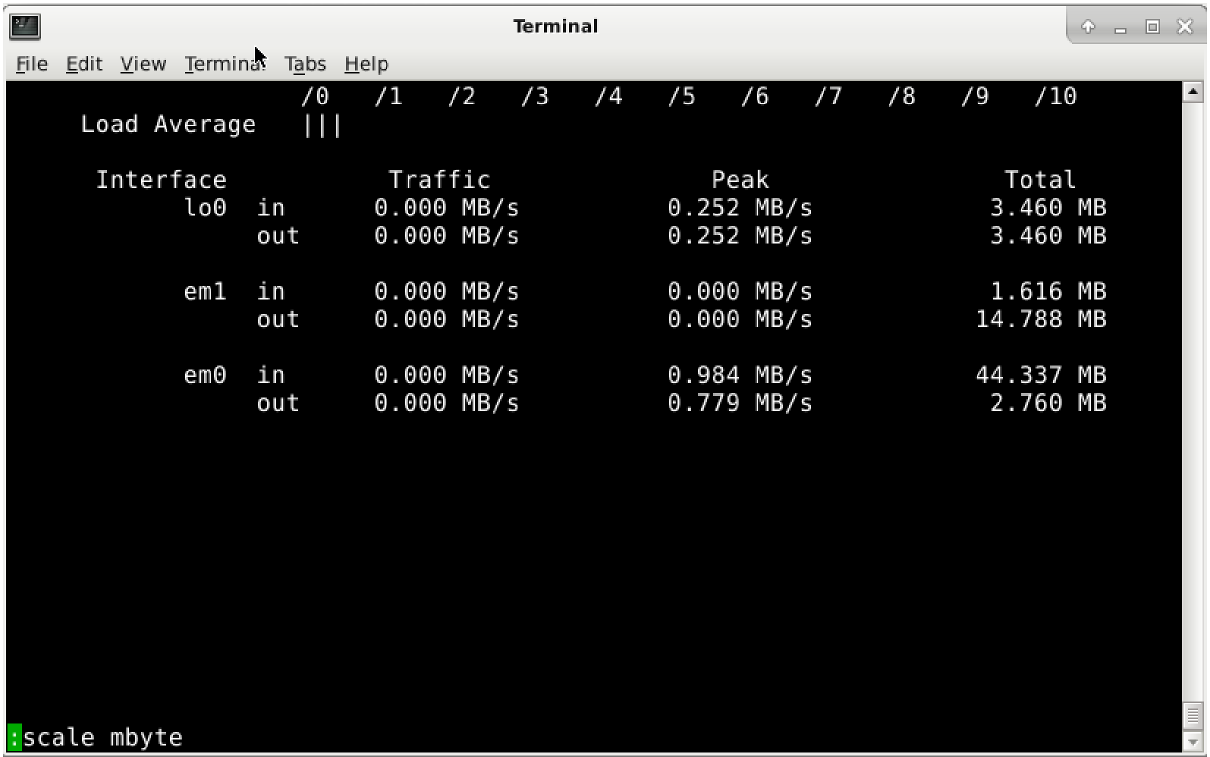
در بسیاری از موارد شما نیاز دارید که وضعیت send و receive کارت های شبکه سیستم خود را مشاهده کنید، برای اجرا کردن و نمایش خروجی این بخش کافیست که بعد از : گزینه ifstat را وارد کنید و بعد Enter کرده تا خروجی فرمان در بخش دوم به صورت زیر برای شما نمایش داده شود:

```
Terminal
File  Edit  View  Terminal  Tabs  Help
                      /0   /1   /2   /3   /4   /5   /6   /7   /8   /9   /10
     Load Average    |

      Interface           Traffic               Peak                Total
            lo0  in      0.000 KB/s          0.000 KB/s           3.444 MB
                 out     0.000 KB/s          0.000 KB/s           3.444 MB

            em1  in      0.000 KB/s          0.000 KB/s           1.615 MB
                 out     0.000 KB/s          0.000 KB/s          14.788 MB

            em0  in      0.000 KB/s          0.000 KB/s          43.907 MB
                 out     0.000 KB/s          0.000 KB/s           2.449 MB



Showing ifstat, refresh every 5 seconds.
```

نمایش ifstat در systat

همانطوری که مشاهده می کنید در این سیستم 3 کارت شبکه وجود دارد به نام های lo0 و em0 وem1 ، در بخش های بعد از نام کارت های شبکه میزان ترافیک خارج شده و پیک هر کارت شبکه را مشاهده می کنید و در بخش total شما با مجموع بسته های ارسال شده و دریافت شده را نمایش می دهد.

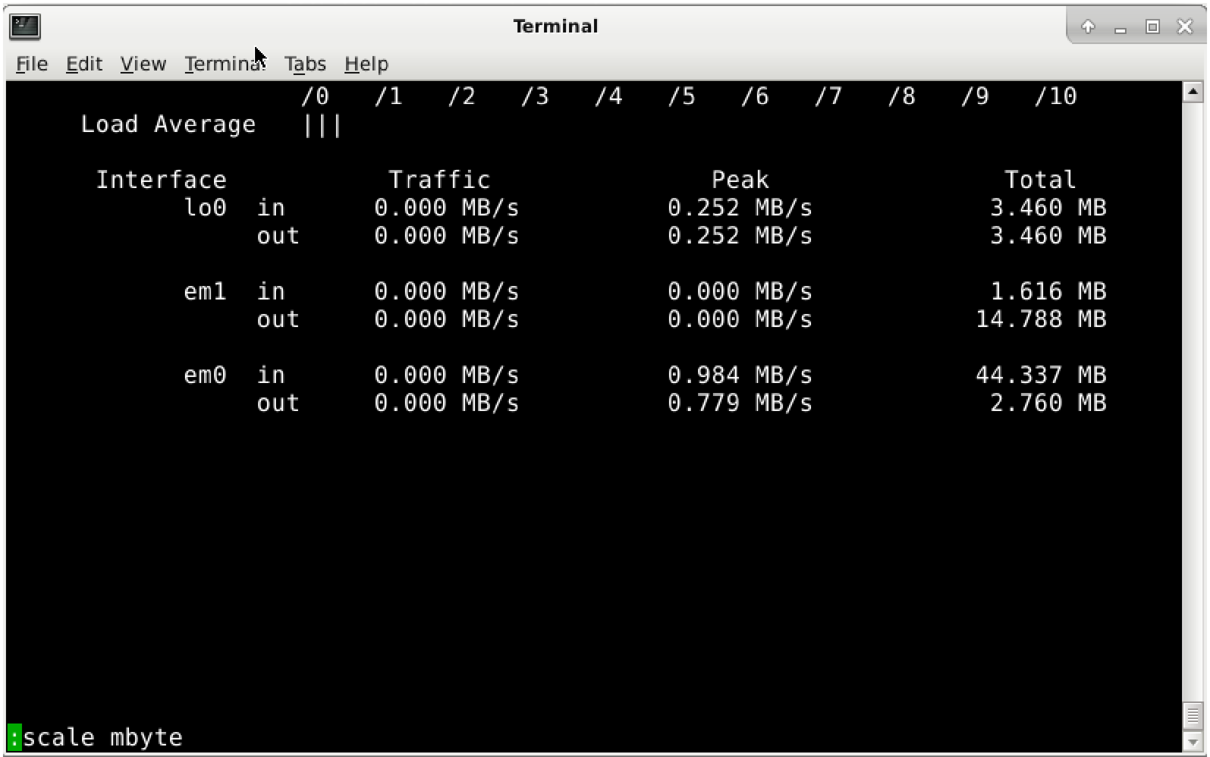
هر برنامه ای در این بخش شامل زیر فرمان هایی است که رفتار برنامه اجرا شده را تغییر می دهد، برای بخش ifstat زیر فرمانی وجود دارد به نام scale که به شما این امکان را می دهد که بتوانید مقادیر نمایش داده شده را در قالبهای kbit kbyte mbit وgbit .. تغییر دهید. در شکل زیر بعد از اجرا کردن scale mbyte مقدار از کیلو بایت به مگابایت تغییر می کند:



تغییر دادن مقیاس در ifstat

بخش :icmp

برای مشاهده کردن اطلاعات پروتکل icmp در این بخش فرمان icmp را در بخش فرمان وارد کنید تا قسمت زیر این فرمان به صورت زیر تغییر می کند:



وارد شدن برای تغییر نمایش وضعیت icmp

این بخش همانطوری که مشاهده می کنید به دو بخش در دو ستون تقسیم می شود که در قسمت سمت راست بسته هایی که وارد می شود را نمایش می دهد و در بخش سمت راست بسته هایی که خارج می شوند را در بخش های مختلف نمایش می دهد.

زیر مجموعه این فرمان خود در دو حالت mode و reset است،

در بخش mode

شما می توانید به چهار حالت زیر وضعیت نمایش را تغییر می دهد:

حالت rate

این بخش که پیش فرض این فرمان است نرخ تغییر هر بسته را در واحد زمان که به صورت پیش فرض یک ثانیه است نمایش می دهد.

حالت delta

این حالت هم مثل حالت بالاست.

حالت since

شما در بخش بعدی یاد می گیرد که حالت reset را اعمال کنید که شمارنده های این بخش صفر شود، برای نمایش هر بسته که وارد را خارج می شود می توانید از این حالت استفاده کنید.

حالت absolute

این حالت همه اطلاعات را نمایش می دهد که تفاوت این دو حالت rate و absolute را در شکل زیر مشاهده می کنید، شکل زیر حالت absolute است:

```
                     /0   /1   /2   /3   /4   /5   /6   /7   /8   /9   /10
     Load Average   |||

          ICMP Input                         ICMP Output
     4711 total messages                1568 total messages
        0 with bad code                  129 errors generated
        0 with bad length                  0 suppressed - original too short
        0 with bad checksum                0 suppressed - original was ICMP
        0 with insufficient data        1439 responses sent
                                           0 suppressed - multicast echo
                                           0 suppressed - multicast tstamp

          Input Histogram                    Output Histogram
     3191 echo response                 1439 echo response
     1439 echo request                     0 echo request
       81 destination unreachable        129 destination unreachable
        0 redirect                         0 redirect
        0 time-to-live exceeded            0 time-to-live exceeded
        0 parameter problem                0 parameter problem
        0 router advertisement             0 router solicitation


:mode absolute
```

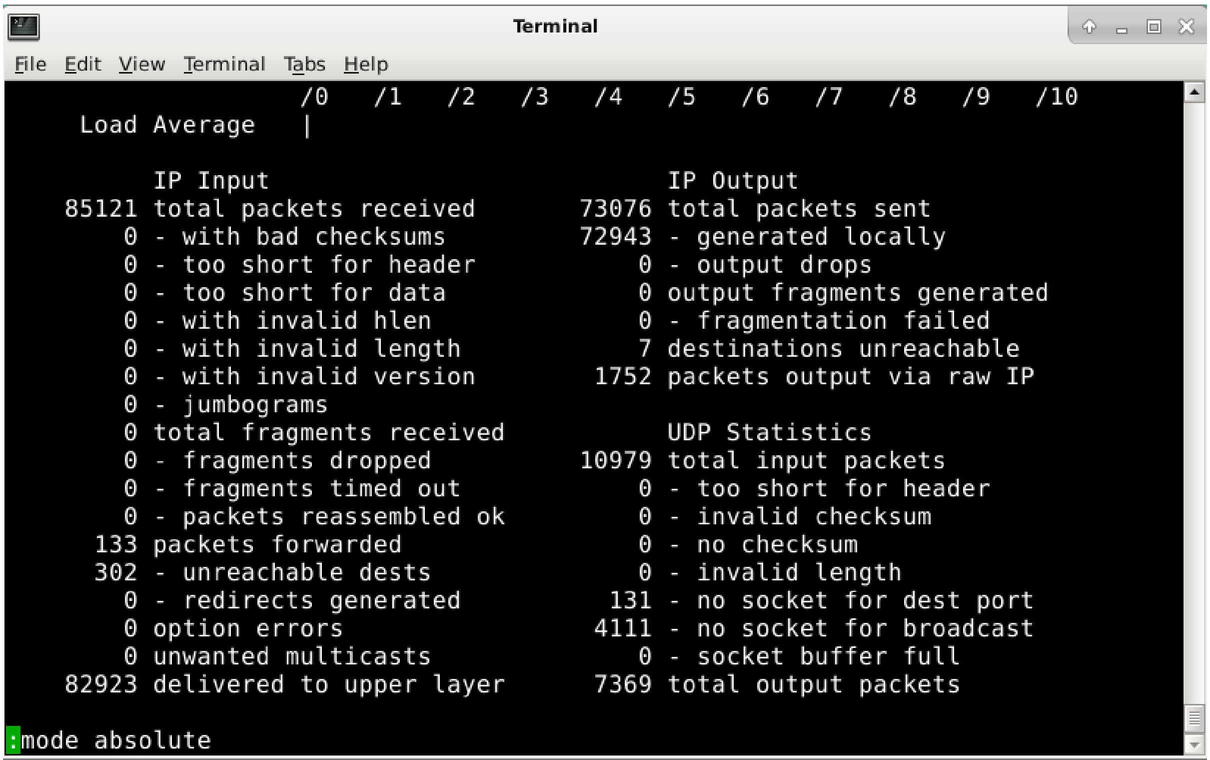
حالت absolute

حالت reset هم همه این اعداد را صفر می کند.

برای نمایش دادن وضعیت icmp در ورژن 6 کافیست که از فرمان icmp6 به جای icmp استفاده کنید و همه بخشها و حالت های بالا هم در آن دو مشترک هستند.

حالت ip و ip6

برای نمایش وضعیت بسته های ip برروی سیستم شما از فرمان های ip و ip6 استفاده کنید که در شکل زیر حالت خروجی این فرمان را مشاهده می کنید:



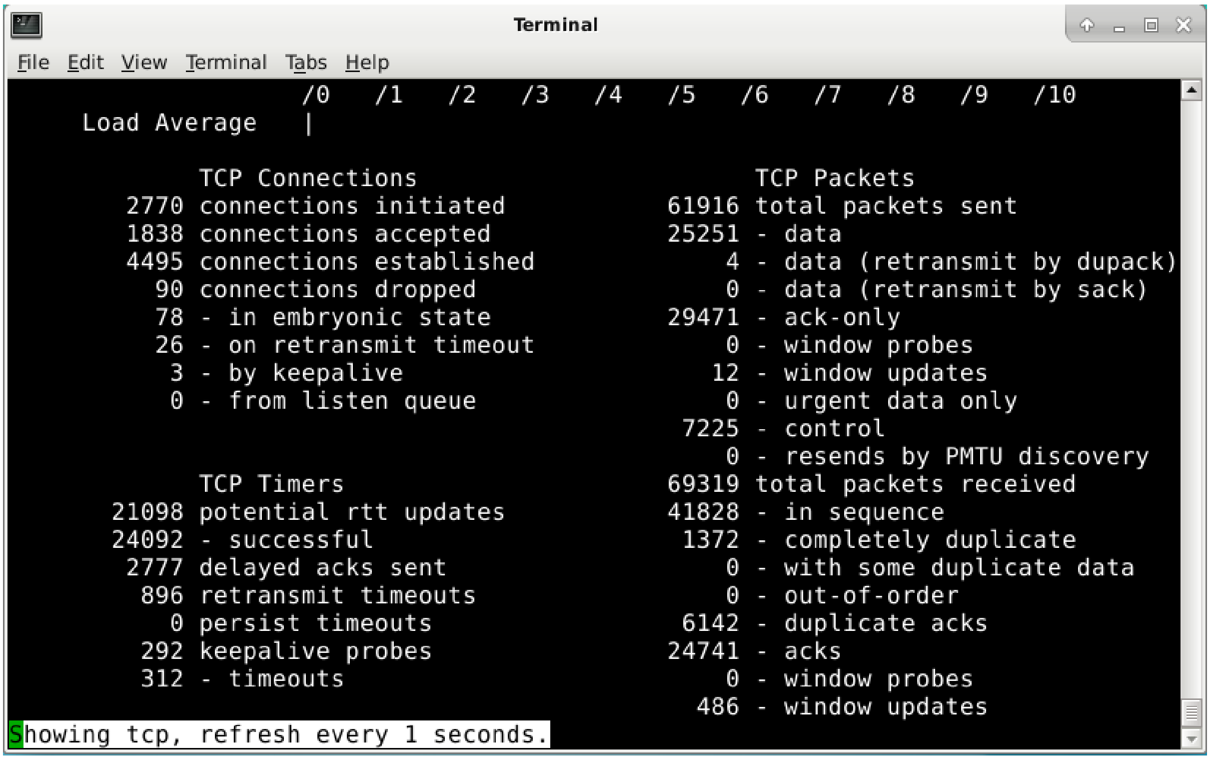
```
Terminal
File  Edit  View  Terminal  Tabs  Help
                    /0   /1   /2   /3   /4   /5   /6   /7   /8   /9   /10
     Load Average   |

          IP Input                                  IP Output
    85121 total packets received           73076 total packets sent
        0 - with bad checksums             72943 - generated locally
        0 - too short for header               0 - output drops
        0 - too short for data                 0 output fragments generated
        0 - with invalid hlen                  0 - fragmentation failed
        0 - with invalid length                7 destinations unreachable
        0 - with invalid version            1752 packets output via raw IP
        0 - jumbograms
        0 total fragments received                  UDP Statistics
        0 - fragments dropped              10979 total input packets
        0 - fragments timed out                0 - too short for header
        0 - packets reassembled ok             0 - invalid checksum
      133 packets forwarded                    0 - no checksum
      302 - unreachable dests                  0 - invalid length
        0 - redirects generated              131 - no socket for dest port
        0 option errors                     4111 - no socket for broadcast
        0 unwanted multicasts                  0 - socket buffer full
    82923 delivered to upper layer          7369 total output packets

:mode absolute
```

نمایش وضعیت ip

این فرمان هم مثل بخش icmp که توضیح داده شده هم شامل بخش mode است که خود شامل 4 زیر گروه rate delta sinceو absolute است و هم شما می توانید اطلاعات را به rest به مقادیر 0 ببرید.

بخش :tcp

برای مشاهده کردن بسته های tcp منتقل شده برروی سیستم، باید وارد بخش tcp در بخش فرمان برنامه systat شوید. برای این کار کافست که از فرمان tcp استفاده کنید تا اطلاعات به صورت شکل زیر برای شما نمایش داده شود:



نمایش وضعیت tcp در systat

این فرمان هم مثل بخش icmp که توضیح داده شده هم شامل بخش mode است که خود شامل 4 زیر گروه rate delta وsince absolute است و هم شما می توانید اطلاعات را به rest به مقادیر 0 ببرید.

بخش iostat

در مقاله قبلی در سایت www.mabedini.ir شما با فرمان isostat آشنا شده اید، در این بخش هم شما می توانید اطلاعات فرمان isotat را در قالبی قابل خواندن به صورت زیر مشاهده کنید:

```
                    /0   /1   /2   /3   /4   /5   /6   /7   /8   /9   /10
     Load Average   |

          /0%  /10  /20  /30  /40  /50  /60  /70  /80  /90  /100
cpu  user|
     nice|
   system|
interrupt|
     idle|XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

          /0%  /10  /20  /30  /40  /50  /60  /70  /80  /90  /100
da0   MB/s
       tps|
da1   MB/s
       tps|
cd0   MB/s
       tps|
pass0 MB/s
       tps|
pass1 MB/s
       tps|

Showing iostat, refresh every 1 seconds.
```

نمایش وضعیت iostat در systat

بخش:swap

برای مشاهده کردن وضعیت swap سیستم خود از فرمان swap در بخش فرمان این برنامه استفاده کنید تا خروجی این فرمان را به صورت زیر مشاهده کنید:

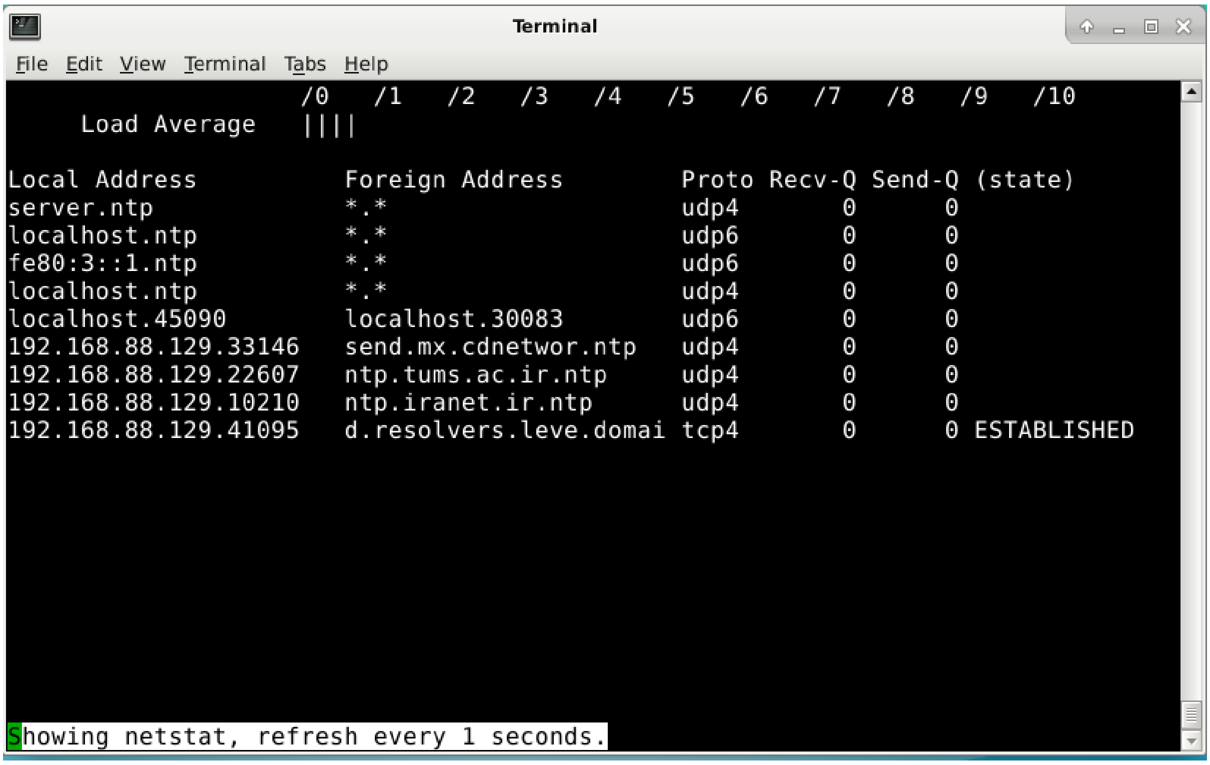
```
                    /0   /1   /2   /3   /4   /5   /6   /7   /8   /9   /10
     Load Average   |

Disk 1K-blocks Used /0%  /10  /20  /30  /40  /50  /60  /70  /80  /90  /100
da0p3   1048412    0 X

Showing swap, refresh every 1 seconds.
```

نمایش وضعیت Swap در systat

این بخش به صورت بار نمایش داده می شود که البته به این دلیلی که این حافظه برروی سیستم من استفاده نمی شود وضعیت در بخش 0٪ قرار داد.

بخش nestat

شاید برای شما اتفاق افتاده باشد که خروجی فرمان netstat را دوست نداشته باشید و نتوانید اطلاعات مفیدی را از آن بدست آورید، در ضمن خروجی این فرمان nestat به صورت داینامیک تغییر نمی کند و شما نمی توانید در لحظه وضعیت ارتباط های شبکه ای خود را مشاهده کنید، برای رها شدن از این مشکل در برنامه systat شما بخشی دارید که این مشل را برای شما مرتفع کرده است به نام netstat که با اجرا کردن این فرمان رفتار برنامه systat به صورت زیر تغییر می کند:



نمایش وضعیت netstat در iostat

به صورت پیش فرض فقط این برنامه ارتباطات فعال را نمایش می دهد و حالت waiting را نمایش نمی دهد، در نمایش پیش فرض این بخش شما اطلاعات را در قالب host:prts مشاهده می کنید. شما می توانید با استفاده کردن از بخشهای زیر مجموعه فرمان systat به تغییر رفتار و حالت های برنامه بپردازید.

فرمان all

برای نمایش همه وضعیت ارتباطی سیستم خود از این فرمان استفاده کنید تا بخش های waiting هم نمایش داده شود.

فرمان number

برای اینکه روش نمایش را به صورت عددی تغییر دهیداز این فرمان استفاده کنید تا به جای نام سرور ها از آدرس ip و شماره های ports ها برای نمایش استفاده شود.

فرمان names

برای برگشت به حالت تبدیل آدرس ip به نام از این فرمان استفاده کنید، اجرای این فرمان کمی زمانبر است.

فرمان porot

شما با استفاده از این فرمان می توانید در مقابل porto نام پروتکل هایی tcp udp و all را تایپ کنید تا فقط پروتکل هایی مورد نظر شما فیلتر شده و فقط آنها نمایش داده شود.

فرمان show

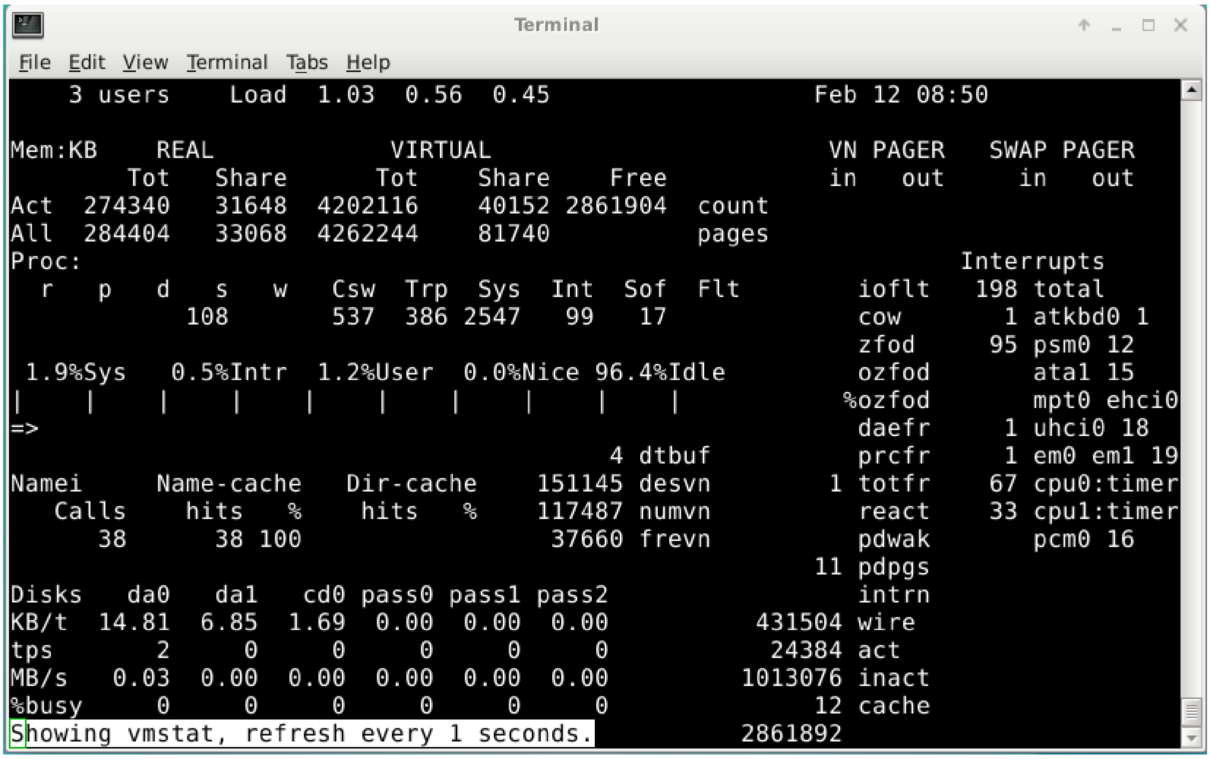
شما می توانید فقط رفتار یک هاست و یک پورت خاص را با استفاده از این فرمان فیلتر کنید، در مقابل ابن فرمان ابتدا ports بعد hosts مورد نظر را وارد کنید،

فرمان reset

برای غیرفعال کردن فیلترهای اعمال شده از فرمان reset استفاده کنید.

بخش vmstat

این بخش برای نمایش اطلاعات حافظه سیستم شما به صورت کامل است که در شکل زیر خروجی این بخش را مشاهده می کنید:



بخش vmstat در systat

کار با inetd در FreeBSD

در FreeBSD برای راه اندازی کردن سرویس ها برنامه ای وجود دارد به نام inetd که در به آن Super server هم گفته می شود، به این دلیل به inetd به اصطلاح super server گفته می شود چون شما با راه اندازی آن می توانید چندین سرور را در قالب یک سرور راه اندازی کنید. زمانی که یک درخواست به این سرور می رسد، این برنامه تعیین می کند که برای درخواست باید کدام سرور را راه اندازی کند. این برنامه مثل یک وکیل بین درخواست و سرور قرار میگیرد و بعد از دریافت درخواست برای برقراری ارتباط بین سرور و کلاینت یک سوکت ایجاد می کند. در حقیقت inetd به تکثیر کردن سایر daemonsها می پردازد، اما از چند پروتکل ساده در درون خود این برنامه پیاده سازی شده است مثل day echo time time.

بخشای این مقاله عبارتند از:

- فایل پیکربندی برنامه.inetd
- بخش های مختلف فایل پیکربندی.

فایل پیکربندی برنامه inetd

برای کار کردن با این برنامه شما باید فایل پیکربندی این برنامه را ویرایش کنید مسیر این فایل در زیر شاخه /etc و نام فایل آن inetd.conf است. هر خط در این فایل برای راه اندازی کردن یک سرویس است، خطوطی کا با علامت # شروع می شوند اجرا نمی شود و برای اجرا کردن هر برنامه کافیست که علامت # ابتدای خط مورد نظر حذف کنید و سرویس inetd را دوباره راه اندازی کنید. به دو روش می توانید سرور inetd را راه اندازی کنید، برای هر دو روش باید در داخل فایل rc.conf خط زیر را برای راه اندازی خودکار سیستم اضافه کنید:

```
inetd_enable="YES"
```

حال می توانید با استفاده از فرمان Service سرور inted را راه اندازی کنید برای این کار کافیست که فرمان زیر را اجرا کنید:

```
# service inetd start
```

شما از طریق فرمان های موجود در زیر شاخه /etc/rc.d هم می توانید inetd را به صورت زیر اجرا کنید:

```
# /etc/rc.d/inetd start
```

اگر قصد دارید که برای یکبار فقط این سرویس را راه اندازی کنید می توانید از فرمان زیر به اصطلاح onestart را انجام دهید و در فایل rc.conf خطی اضافه کنید:

```
# /etc/rc.d/inetd onestart
```

در بسیاری از موارد اتفاق می افتد که شما تغییری در فایل پیکربندی inted اعمال کرده اید برای فقط بارگذاری فایل پیکربندی و قطع نشدن ارتباط قبلی فقط کافیست که از reload به صورت زیر استفاده کنید:

```
# service inetd reload
```

در خط زیر یک مثال از فایل پیکربندی برای راه اندازی کردن سرور ftp را مشاهده می کنید:

```
ftp     stream  tcp     nowait  root    /usr/libexec/ftpd       ftpd -l
```

این خط شامل اطلاعاتی است که در ادامه بیشتر با آنها آشنا می شوید، کل اطلاعات این بخش به صورت زیر است:

```
service-name
socket-type
protocol
{wait|nowait}[/max-child[/max-connections-per-ip-per-minute[/max-child-per-ip]]]
user[:group][/login-class]
server-program
server-program-arguments
```

بخش service-name

بخش اول از خط مورد نظر شما با نام سرویس آغاز می شود این نام ها باید مطابق با فایل /etc/services باشد، که از این فایل شماره پورت مورد نظر جهت ارایه دادن سرویس مشخص می شود. اگر شما از سرویسی استفاده می کنید که در این فایل وجود ندارد در بخش اول باید آنرا در این فایل اضافه کنید.

بخش socket-type

این بخش شامل گزینه های stream , dgram, raw, seqpacket که برای ارتباطهای tcp باید از stream و برای ارتباط از نوع udp باید از dgram استفاده کنید.

بخش protocol

در این بخش شما برای تنظیم کردن نوع ارتباط tcp و udp در دو حالت ورژن 4 و 6 باید از کلید های زیر استفاده کنید:

کلید	**نما استفاده شده**
TCP IPv4	tcp or tcp4
UDP IPv4	udp or udp4
TCP IPv6	tcp6
UDP IPv6	udp6
Both TCP IPv4 and IPv6	tcp46

کلید	نما استفاده شده
Both UDP IPv4 and IPv6	udp46

در یک خط هم شما می توانید دو ورژن 4 و 6 را با هم مشخص کنید.

بخش :wait | nowait

این دو گزینه چگونگی مدیریت کردن سوکت توسط برنامه را مشخص می کند، اتباطات از نوع udp باید از حالت wait استفاده کنند و ارتباط tcp که multi-threaded هستند باید از حالت nowait استفاده کنند، این امر به این دلیل است که حالت waitبرای یک پردازش چندین سوکت را باز نمی کند و حالت nowait می تواند هم زمان از چندین سوکت استفاده کند .

بخش /max-child

این بخش دقیقا به از حالت nowait استفاده شده و با / از آن جدا می شود و تعداد child daemons که inetd اجازه دارد برای هر پردازش باز کند را مشخص می کند، برای محدود کردن 10 زیر پردازش باید از عدد /10 استفاده کنید و /0 به معنی بی نهایت است.

بخش /max-connections-per-ip-per-minute

این بخش تعداد ارتباطات برقرار شده از یک آدرس ip خاص در مدت یک دقیقه را مشخص می کند، اگر تعداد درخواست ها بیشتر از مقدار مورد نظر باشد اتصال برقرار نشده تا مدت زمان محدودیت به اتمام برسد. این امر باعث کاهش مصرف منابع سیستم می شود و در مواردی هم از حمله ddos جلوگیری می کند.

بخش user

در این بخش می توانید سطح دسترسی کاربری را که می خواهید سرور شما با آن سطح دسترسی کاربر راه اندازی شود را مشخص کنید. این بخش در صورتی که سرور شما هک شود مفید بود و هکر در صورت shell گرفتن از سرور با سطح دسترسی کاربر معمولی به سیستم وارد می شود. معروف ترین این بخش ها deamon و nobody است.

بخش Server-program

در این بخش شما مسیر کامل برنامه ای که به عنوان سرور باید توسط inetd راه اندازی شود را مشخص کنید.

بخش Server-program-arguments

بعضی از سور ها دارای flags های خاصی هستند که باعث ایجاد تغییرات در نوع و رفتار سرور مورد نظر شما می شود، با استفاده از این بخش شما می توانید تنظیمات خاص خود را اعمال کنید.

فایروال از نوع host-based firewall در FreeBSD

در FreeBSD علاوه بر فایروال های معروف مثل IPFW و PF نوع دیگری از مدیریت کردن درخواست های ارسالی سمت سرویس ها مثل ftp ssh و غیره وجود دارد به نام host-based firewall که این نوع از فایروال برروی سیستم محلی راه اندازی می شود و به درخواست هایی که سمت سرور ارسال می شود نظارت می کند، در FreeBSD این نوع از فایروال از تکنولوژی TCP wrappers استفاده می کند که خود این برنامه از طریق inetd راه اندازی می شود. قبل از راه اندازی این سرویس مقاله مربوط به کار با inetd در :FreeBSD را مطالعه کنید. در بخش اول از این مقاله با فرمان inetd آشنا شوید که باعث راه اندازی tcp wrapper می شود و در بخش دوم با TCP wrappers آشنا می شوید و در قسمت آخر با فایل hostd_accsessآشنا می شوید.

بخش ها:

- فرمان.inetd
- آشنایی باTCP wrappers
- فایلhostd_accsess

فرمان inetd

به صورت پیش فرض برنامه inetd با سویچ های پیش فرض wW –C 60– اجرا می شود که این سویچ ها باعث اجرا شدن TCP wrappersمی شود. این سویچ باعث راه اندازی TCP wrappers برای همه سرویس های راه اندازی شده توسط inetdمی شود و به صورت پیش فرض از ارتباط بیش از 60 درخواست از سمت یک آدرس IP در یک دقیقه جلوگیری می کند. برای تغییر دادن باید به رفتار inetd خود یک سری flags اضافه کنید که این امر از طریق inetd_flags در فایل /etc/rc.confممکن می شود .

سویچ c maximum–

به صورت پیش فرض تعداد فراخوانی های همزمان از یک سرویس را تعیین می کند که محدود نشده و بی نهایت است. این بخش ممکن است با استفاده از max-child باطلل شود.

سویچ C rate–

این بخش مشخص می کند که یک سرویس در واحد زمان از یک آدرس IP خاص چندبار فراخوانی شود. این بخش با استفاده از سویچ max-connections-per-ip-per-minute می تواند باطل شود.

سویچ R rate–

این بخش بیشترین تعدادی که یک سرویس می تواند در واحد زمانی طلب شود را تعیین می کنید که مقدار پیش فرض آن 256 است و مقدار 0 به معنای بی نهایت است.

سویچ s maximum–

این بخش تعداد بیشینه که یک سرور در واحد زمانی می تواند از یک آدرس IP خاص در خواست داشته باشد را تعیین می کند و این بخش می تواند با مقدار max-child-per-ip در فایل inetd.conf قابل تغییر است.

کار با TCP Wrapper

برنامه TCP Wrapper یک فایروال تحت سیستم محلی است که با استفاده از برنامه inetd راه اندازی می شود. با استفاده از این برنامه قابلیت گزارش گیری به برنامه اضافه می شود،برنامه های اجرا شده تحت این سیستم می توانند پیغام مناسب به سمت کاربر بازگردانی کند و می تواند دسترسی به سرور را محدود کند. برای دریافت اطلاعات بیشتر به tcpd مراجعه کنید.

از این برنامه نمی توانید به عنوان فایروال استفاده کنید و فایروال را حذف کنید، بلکه باید از آن در کنای سایر سیستم های مدیریت ارتباطی و فایروال استفاده کنید.

راه اندازی ابتدایی:

همانطوری که در بخش قبلی از این مقاله به آن اشاره شده است، برای راه اندازی کردن TCP Wrapper باید از طریق inetd اقدام کنید. در نتیجه برای را ه اندازی باید خط زیر را در فایل /etc/rc.conf اضافه کنید:

```
inetd_enable="YES"
inetd_flags="-Ww"
```

حال در بخش بعدی باید فایل به نام /etc/hosts.allow را ویرایش کنید که در ادامه این فایل است که مدیریت دسترسی ها را انجام می دهد.

نکته: برخلاف سایر پیاده سازی های TCP Wrapper که از فایل hosts.deny استفاده می کند. این برنامه در FreeBSD از فایل hosts.allowاستفاده می کند.

ابتدایی ترین بخش مدیریت در TCP Wrapper این است که دسترسی به یک سرویس را محدود و یا اجازه دهد. این مدل از سطح دسترسی بر مبنای فایل hosts.allow است. به صورت پیش فرض در FreeBSD همه سرویس ها allow هستند.

در سطح مقدماتی از پیکربندی ، این فایل شامل خطوطی به صورت زیر است:

daemon : address : action,

بخش daemon نام سروری است که باید برنامه inetd راه اندازی شود که مقادیر آن در فایل inetd قرار دارد. در بخش دوم نام یا آدرس IP سیستمی است که شما قصد دارید دسترسی آنرا مورد بررسی قرار دهید. این بخش هم می تواند IPV4 باشد همIPV6 ، باید آدرس IPV6 را داخل براکت قرار دهید. در بخش action دو حال وجود دارد allow و.deny

برنامه TCP Wrapper از قاعده first rule match برای جستجو کردن در فایل پیکربندی خود استفاده می کند و اول خطی که با شرایط مطابقت داشته باشد را می پذیرد و دیگر به جستجو نمی پردازد در نتیجه به قواعدی که در این فایل قرار می دهید دقت کنید.

در خط زیر یک مثالی را مشاهده می کنید که به سرویس pop3 که از طریق inetd راه اندازی می شود در خط زیر اجازه داده می شود که همه کامپیوتر های موجود در شبکه به آن دسترسی داشته باشند، این برنامه qpopper است:

```
# This line is required for POP3 connections:
qpopper : ALL : allow
```

در مرحله بعدی برای اعمال شدن تغییرات باید برنامه inted را دوباره راه اندازی کنید به صورت زیر:

```
# service inetd restart
```

مقدمه بر شبکه بی سیم در FreeBSD

بیشتر شبکه های بیسیم مبتنی بر استاندارد 802.11 است. این استاندارد در دو حالت infrastructure و مد ad-hoc networkراه اندازی می شود که در حالت infrastructure شبکه مبتنی بر یک Access point راه اندازی می شود و همه سیستم ها بسته ها اطلاعات خود را به سمت AP ارسال کرده و این AP است که به مدیریت کردن شبکه می پردازد. در حالت ad-hoc network هیچ AP برای مدیریت کردن وجود ندارد، این دو حالت از شبکه در سیستم عامل FreeBSD قابل ارایه و پیاده سازی است FreeBSD. از استاندارد های 8020.11 802.11b, and 802.11g , aپشتیبانی می کند.

مراحل قبل از راه اندازی کردن:

این بخش خود به چند زیر مجموعه تقسیم می شود، در قسمت اول شما باید هسته FreeBSD را پیکربندی کنید، در قدم اول شما باید درایور کارت شبکه خود را در هسته FreeBSD بارگذاری کنید، یکی از کارت های وایرلس متداول در بازار کارت شبکه Atheros است که درایور آن ath نام دارد و به صورت زیر باید آنرا در هسته باز گذاری کنید، برای این کار باید قبل از بارگذاری هسته آنرا در هسته فعال کنید به همین دلیل باید از loader.conf استفاده کنید ، این فایل در شاخه boot قرار دارد و با استفاده از اضافه کردن خط زیر درایور در هسته بارگذاری می شود:

```
if_ath_load="YES"
```

بخش اطلاعات بیشتر در مورد انواع کارتهای شبکه در FreeBSD از آنها حمایت می شود و درایور ها آن موجود است به آدرس زیر مراجعه کنید:

https://www.freebsd.org/releases/10.2R/hardware.html#wlan

در قدم بعدی باید قابلیت رمزنگاری را هم به هسته FreeBSD اضافه کنید، برای این کار باید ماژولهایی را که این قابلیت را به هسته اضافه می کنند را در هسته بارگذاری کنید، این بخش هم باید در فایل loader.conf اضافه شود. سه ماژول مورد نظر ما به اسم های زیر هستند:

- wlan_ccmp
- wlan_wep
- wlan_tkip

ماژول های wlan_tkip(4) wlan_ccmp(4), زمانی مورد استفاده قرار می گیرند که شبکه شما از WPA یا 802.11iاستفاده می شود، به هر حال برای بارگذاری این ماژول ها در هسته از خطوط زیر استفاده کنید، این خطوط را به فایل loader.conf اضافه کنید:

```
wlan_wep_load="YES"
wlan_ccmp_load="YES"
wlan_tkip_load="YES"
```

ماژول wlan_wep(4) برای WEP cryptographic در هسته استفاده می شود.

ماژول wlan_tkip(4) از دو حالت TKIP و Michael cryptographic که برای دستگاه هایی که از استاندارد 802.11 حمایت می کند راه اندازی می کند.

ماژول wlan_ccmp(4) باعث اضافه شدن رمزنگاری AES-CCMP برای دستگاه هایی که از استاندارد 802.11 حمایت می کند راه اندازی می کند.

```
device wlan              # 802.11 support
device wlan_wep          # 802.11 WEP support
device wlan_ccmp         # 802.11 CCMP support
device wlan_tkip         # 802.11 TKIP support
device wlan_amrr         # AMRR transmit rate control algorithm
device ath               # Atheros pci/cardbus NIC's
device ath_hal           # pci/cardbus chip support
options AH_SUPPORT_AR5416 # enable AR5416 tx/rx descriptors
device ath_rate_sample   # SampleRate tx rate control for ath
```

بعد از راه اندازی مجدد سیستم شما باید در پیغام های هسته سیستم خود در dmesg قرار داده می شود خطوط زیر را مشاهده کنید که به کارت شبکه سیستم شما مربوط می باشد:

```
ath0: <Atheros 5212> mem 0x88000000-0x8800ffff irq 11 at device 0.0 on cardbus1
ath0: [ITHREAD]
ath0: AR2413 mac 7.9 RF2413 phy 4.5
```

محدود کردن فضای مصرفی دیسک با Disk quota

برای محدود کردن میزان مصرف فضای دیسک از قابلیت Disk quota در FreeBSD استفاده می شود که این محدودیت می تواند به یک کاربر خاص یا گروهی خاص که کاربران در آن قرار دارند اعمال شود.

همانطوری که می دانید در FreeBSD دو فایل سیستم وجود دارد، یکی UFS و دیگری ZFS که در این بخش شما با اعمال محدودیت در فایل سیستم UFS آشنا می شوید.

بخشهای این مقاله:

- فعال سازی. disk quota
- تنظیم کردن یک محدودیت با disk quota
- چک کردن محدودیت اعمال شده.

فعال سازی Disk quota

در بخش اول از راه اندازی شما باید چک کنید که آیا هسته سیستم عامل FreeBSD شما آیا قابلیت راه اندازی disk quota را دارد ؟ برای این کار از فرمان sysctl به صورت زیر استفاده کنید:

```
% sysctl kern.features.ufs_quota
kern.features.ufs_quota: 1
```

گر مقداری که در خروجی این فرمان برگشت داده می شود عدد 1 باشد بدین معناست که این قابلیت در هسته سیستم عامل شما فعال شده است و شما می توانید آنرا راه اندازی کنید، در غیر این صورت شما باید هسته FreeBSD خود را به خط زیر دوباره کامپایل کنید، این خط را در زیر مشاهده می کنید:

```
options QUOTA
```

برای دریافت اطلاعات بیشتر در زمینه راه اندازی و کامپایل کردن یک هسته جدید به مقاله زیر مراجعه کنید.

پیکربندی هسته در FreeBSD

حال بعد از فعال کردن این بخش در هسته نوبت به فعال سازی آن با استفاده از فایل rc.conf می رسد که شما باید در این بخش خط زیر را در این فایل اضافه کنید:

سیستم quota بعد از راه اندازی شده با استفاده از برنامه quotacheck هر بار به چک کردن وضعیت دیتاسیت فایل سیستم شما می پردازد که این امر زمان گیر است برای غیرفعال کردن این بخش شما باید خط زیر را در فایل Rc.conf قرار دهید:

```
check_quotas="NO"
```

در بخش پایانی از این بخش باید به تناسب نیاز خود در مورد فعال سازی اقدام کنید دو حالت فعال سازی وجود دارد ، حالت اول per-user است که با استفاده از آن می توانید برای هر کاربر محدودیت اعمال کنید و حالت دوم group quotas که برای اعمال محدودیت برای یک گروه خاص است و کاربرانی که در آن گروه قرار دارند را محدود می کند، شما می توانید به تناسب نیاز خود یکی یا هر دوی این تغییرات را فعال کنید، راه فعال کردن در فایل fstab است که در زیر شاخه /etc قرار دارد و شما می توانید برای هر فایل سیستمی که نیاز دارید این قابلیت را اعمال کنید.

نکته: اگر شما قصد دارید که برای مثال شاخه home کاربران را محدود سازی کنید بهتر است که در زمان نصب شاخه /home را یک فایل سیستم جداگانه در نظر بگیرید.

فعال سازی :per-user

برای فعال کردن این بخش کافیست که در پایان هر فایل سیستم یا خطی که در فایل fstab قرار دارد کلمه کلیدی userquota را اضافه کنید به صورت زیر:

```
/dev/da1s2g   /home    ufs rw,userquota 1 2
```

فعال سازی :group quotas

برای فعال کردن این بخش کافیست که در پایان هر فایل سیستم یا خطی که در فایل fstab قرار دارد کلمه کلیدی groupquota را اضافه کنید به صورت زیر:

```
/dev/da1s2g    /home    ufs rw,userquota,groupquota 1 2
```

بعد از راه اندازی کردن این بخش و راه اندازی شدن سیستم دو فایل به نام های quota.user و quota.group در شاخه اصلی فایل سیستمی که شما در آن quota را فعال کرده اید ایجاد می شود. با راه اندازی سیستم و راه اندازی rc این عمل اتفاق می افتد.

برای مدیریت کردن بخش راه اندازی در سیستم شما به فرمان های quotaoff(8) ,quotaon(8) quotacheck(8) نیاز دارید که برای دریافت اطلاعات بیشتر به man هر کدام از این صفحات مراجعه کنید.

تنظیم کردن یک محدودیت با disk quota

برای چک کردن از وضعیت راه اندازی quota از فرمان زیر استفاده کنید:

```
# quota -v
```

اگر خروجی این فرمان شامل خطوطی باشد که در مورد اطلاعات خلاصه وضعیت quota باشد بدین معناست که تنظیمات راه اندازی شما به درستی اعمال شده.

بخش های مختلفی برای اعمال محدودیت وجود دارد برای اعمال برروی کاربران و یا گروه ها که هم می تواند مبتنی بر مقدار فضای مصرفی از هارد باشد به نام block quota و هم می تواند بر اساس تعداد فایل های هر کاربر باشد که به آن inod quotaگویند. اگر شما همزمان هر دوی این محدودیت ها را اعمال کرده باشید هر کدام که زودتر اتفاق افتاد آن محدویت اعمال می شود. این دو حالت در دو دسته hard و soft هم طبقه بندی می شود.

Hard limit

اگر در این حالت محدودیت اعمال کرده باشید اجازه پیشروی بیشتر به کاربر داده نمی شود، برای مثال اگر در فایل سیستمی شما فضای یک کاربر را 800 کیلوبایت محدود کرده اید و کاربر 790 کیلوبایت از فضا را مصرف کرده باشد در صورت ایجاد کردن یک فایل با میزان 11 کیلوبایت ایجاد کردن آن fail می شود.

Soft limit

این نوع از محدودیت با بازه زمانی خاصی کار می کند که به صورت پیش فرض یک هفته است و اگر کاربر در این مدت زمانی به فضای قبلی خود باز نگشت soft limit به حالت hard limit تبدیل می شود و کاربر دیگر اجازه ایجاد کردن فایل های بیشتر را ندارد تا زمانی که به مقدار فضای اعمال شده برگردد و دوباره حالت به soft limit بازگردانی می شود.

فرمان edquota

برای ویرایش کردن وضعیت محدودیت هر کاربر از این فرمان استفاده می شود. این فرمان شما با استفاده از ویرایشگر متنی کار می کند و مقدار پیش فرض را از متغیر EDITOR میگیرد و به صورت پیش فرض این مقدار vi است درنتیجه اطلاعات با ویرایشگر vi باز می شود. برای اعمال کردن محدودیت به یک کاربر خاص باید نام آن کاربر را بعد از سویچ u وارد کنید به صورت زیر:

```
# edquota -u abedini
Quotas for user abedini:
/usr: kbytes in use: 65, limits (soft = 50, hard = 75)
        inodes in use: 7, limits (soft = 50, hard = 60)
/usr/var: kbytes in use: 0, limits (soft = 50, hard = 75)
        inodes in use: 0, limits (soft = 50, hard = 60)
```

هر فایل سیستمی که شما در نظر دارید شامل یک خط برای اعمال محدودیت می باشد که می توانید با استفاده از ویرایشگر خود مقدار را تغییر دهید.