

Redhat LINUX

mcsevic

Mục lục chính

Bài 1: Giới thiệu, cài đặt Redhat, và tổng quan các lệnh thông thường	3
Bài 2: Cài đặt và Cấu hình DHCP Server.....	21
Bài 3: Cài đặt và Cấu hình DNS Server	25
Bài 4: Cài đặt và Cấu hình Samba Server.....	31
Bài 5: Cài đặt và Cấu hình Squid Server.....	45
Bài 6: Cài đặt và Cấu hình Apache Server.....	51
Bài 7: Cài đặt và Cấu hình Sendmail	73
Bài 8: Cài đặt và Cấu hình RAS.....	79
Các phần mềm công cụ hỗ trợ (third party)	
Bài 10: Webmin	
Bài 11: Secure CRT.....	
Bài 12: VNC.....	
Phụ lục	
Một số website hữu ích.....	

CHƯƠNG TRÌNH ĐÀO TẠO REDHAT LINUX CHO MCSA,MCSE

1

Giới thiệu LINUX

Linux là miễn phí (free). Đối với chúng ta hôm nay không quan trọng vì ngay WindowsNT server cũng “free”. Nhưng trong tương lai, khi chúng ta muốn hòa nhập vào thế giới, khi chúng ta muốn có một thu nhập chính đáng cho người lập trình, hiện tượng sao chép trộm phần mềm cần phải chấm dứt. Khi đó, free là một thông số rất quan trọng để chọn Linux.

Linux rất ổn định. Trái với suy nghĩ “truyền thống” là “của rẻ là của ôi”, Linux từ những phiên bản đầu tiên cách đây 5-6 năm đã rất ổn định. Ngay cả server Linux của những mạng lớn (hàng trăm máy trạm) cũng hoạt động rất ổn định.

Linux đầy đủ. Tất cả những gì bạn thấy ở IBM, SCO, Sun ... đều có ở Linux. C compiler, perl interpreter, shell, TCP/IP, proxy, firewall, tài liệu hướng dẫn ... đều rất đầy đủ và có chất lượng. Hệ thống các chương trình tiện ích cũng rất đầy đủ.

Linux là HDH hoàn toàn 32-bit. Như các Unix khác, ngay từ đầu, Linux đã là một HDH 32 bits.

Linux rất mềm dẻo trong cấu hình. Linux cho người sử dụng cấu hình rất linh động, ví dụ như độ phân dải màn hình Xwindow tùy ý, dễ dàng sửa đổi ngay cả kernel ...

Linux chạy trên nhiều máy khác nhau từ PC 386, 486 tự lắp cho đến SUN Sparc.

Linux được trợ giúp. Ngày nay, với các server Linux sử dụng dữ liệu quan trọng, người sử dụng hoàn toàn có thể tìm được sự trợ giúp cho Linux từ các công ty lớn. IBM đã chính thức chào bán IBM server chạy trên Linux. Tài liệu giới thiệu Linux ngày càng nhiều, không thua kém bất cứ một HDH nào khác.

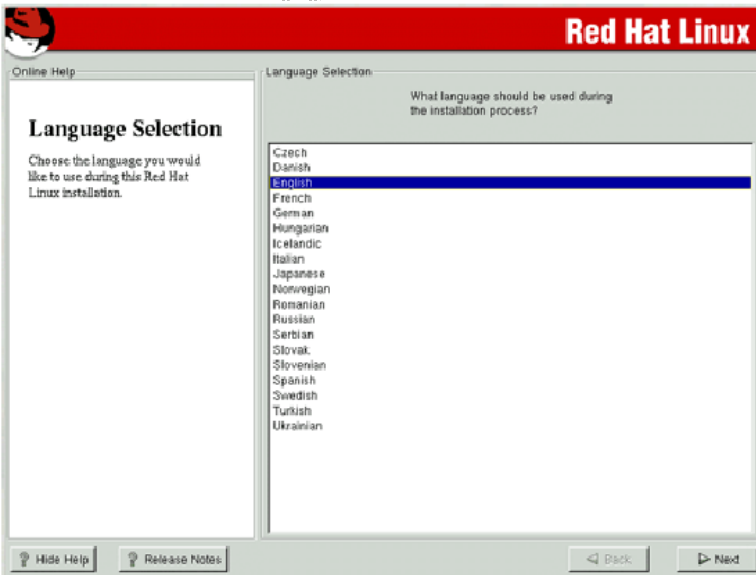
Cài đặt Redhat 7.3

Thiết bị cần thiết :

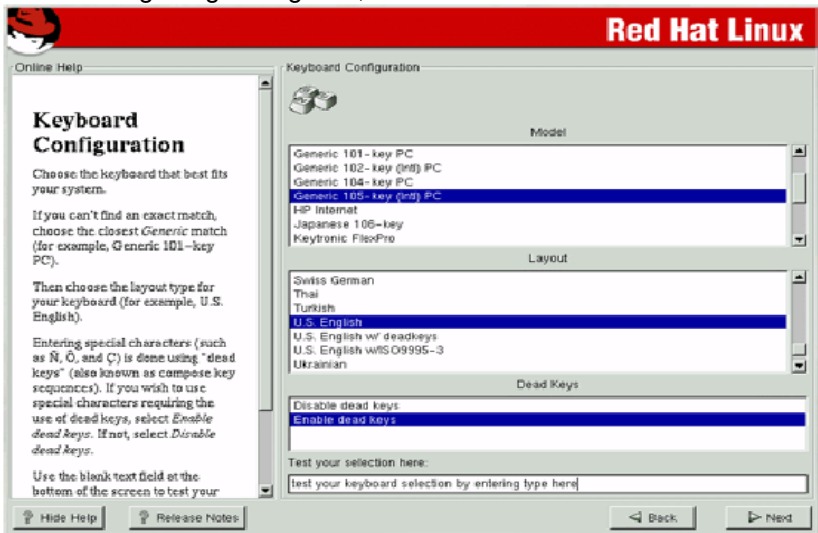
- a. Server : yêu cầu cấu hình :
 - + CPU : Intel PII 400 hoặc cao hơn.
 - + RAM : 128MB hoặc cao hơn.
 - + HDD : 10GB hoặc cao hơn.
 - + NIC card : 100 Mbps
- + External Modem
- +
- b. Switch có tốc độ 100 Mbps.
- c. Cable và các thiết bị cần thiết khác.
- d. Các máy client có cấu hình :
 - + CPU từ 486 trở lên
 - + RAM : 64 MB
 - + HDD : 4.3GB hoặc cao hơn.
 - + NIC card : 100 Mbps.
 - +

Cài đặt và cấu hình :

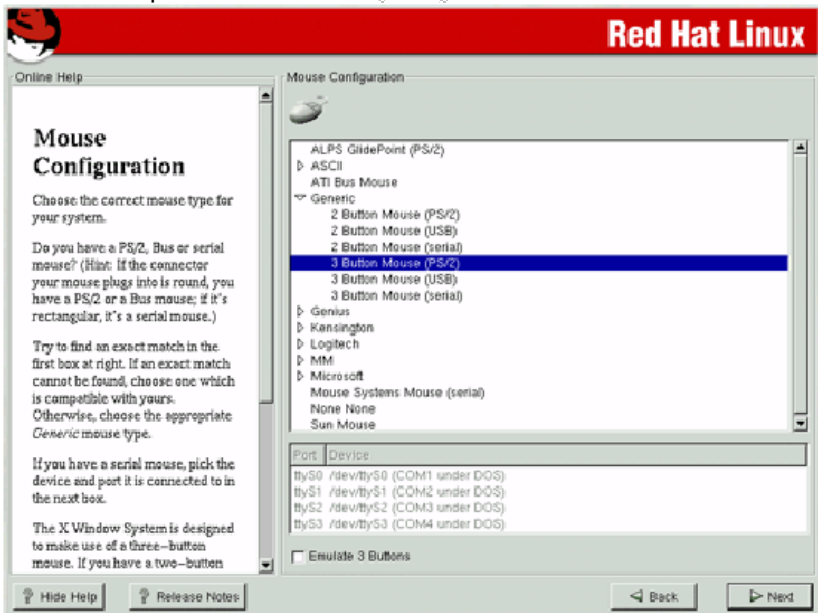
Bước đầu tiên, chúng ta cài đặt và khởi động từ CDROM. Đưa đĩa Red Hat 7.3 thứ 1 vào, tại dấu nhắc "boot :", nhấn ENTER. Khi vào được bên trong, bỏ qua bước kiểm tra CDROM, màn hình tiếp theo sẽ xuất hiện như sau :



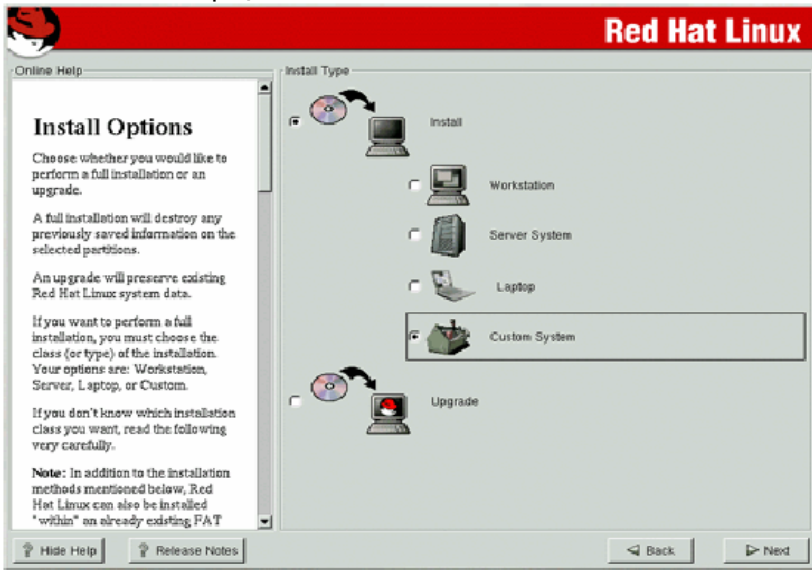
Default là ngôn ngữ tiếng Anh, Click Next.



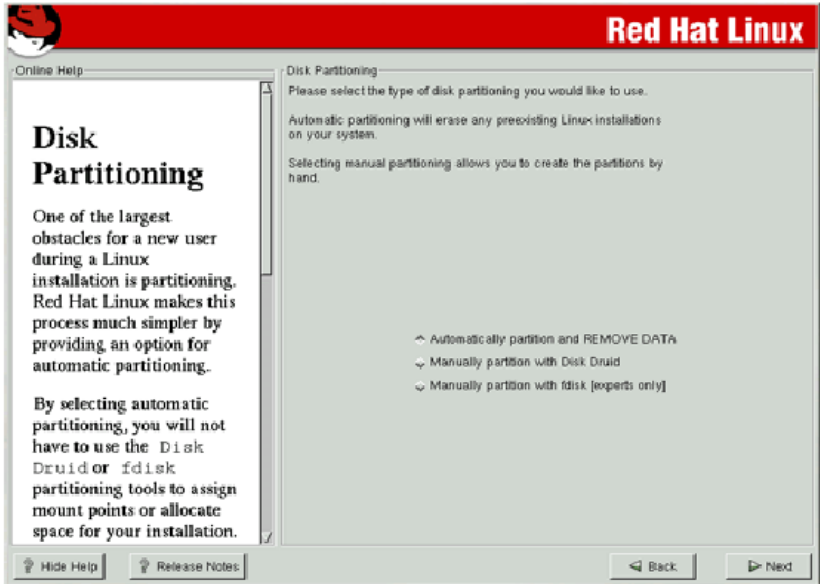
Hãy lựa chọn thiết bị phần cứng cho thích hợp, sau đó Click Next, màn hình tiếp theo sẽ như sau



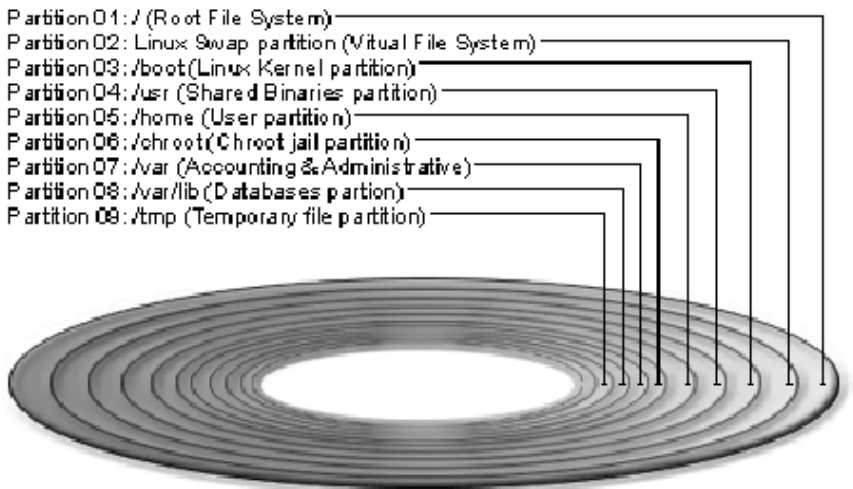
Click NEXT để tiếp tục.



Hãy lựa chọn loại hình muốn sử dụng, ở đây chúng ta sẽ chọn là “Custom System” để thêm một số package cần thiết. Click NEXT.



Ở đây có 2 lựa chọn cho chúng ta là mục chọn thứ 1, hệ điều hành sẽ xóa hết DATA trên máy và tự động chọn phân vùng để cài đặt. Còn mục chọn thứ 2 là do chính ta sẽ chỉ định phân vùng nào muốn cài và cài với dung lượng là bao nhiêu. Chúng ta có sơ đồ :



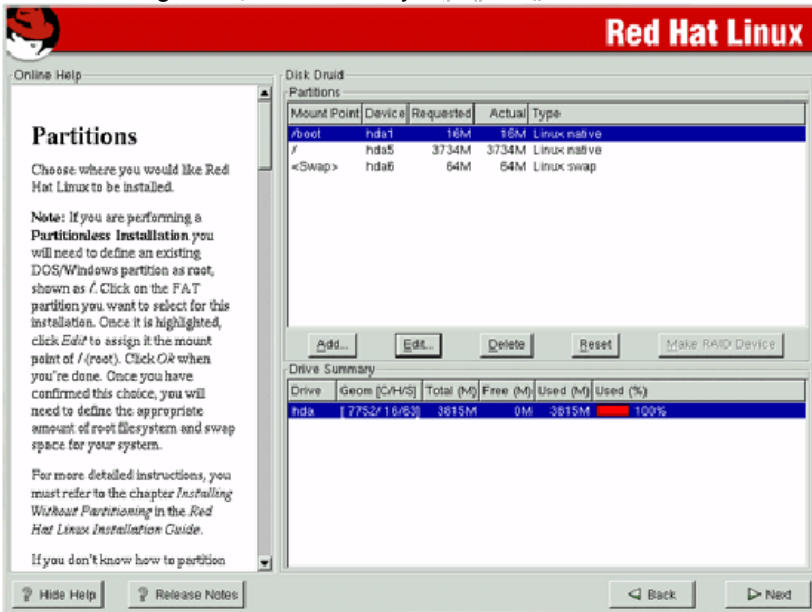
Ví dụ chúng ta sẽ tạo các phân vùng như sau :

```
/boot          5    MB
<Swap>        512  MB
/              256  MB
/usr          512  MB
/home        5700 MB

/var          256  MB
/tmp         329  MB
```

Lưu ý là với / boot là nơi chứa toàn bộ những ảnh của kernel, partition Swap sẽ là nơi làm bộ nhớ ảo của Linux, phân vùng Swap sẽ có dung lượng gấp đôi số dung lượng RAM hiện có. Phân vùng /home sẽ là nơi chứa dung lượng của mỗi user, nghĩa là mỗi user sẽ có dung lượng là 100MB, /home sẽ bằng số user x 100.

Khi đó sẽ có giao diện như thế này :



Server của chúng ta sẽ có các phân vùng sau :

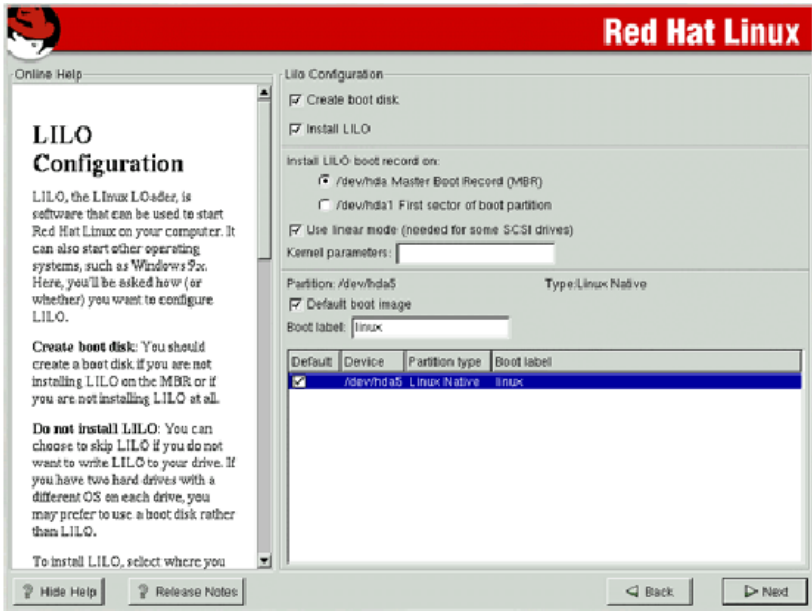
- + /boot : 100MB
- + swap : RAM x 2
- + /home : tùy theo số user
- + /var : 3 GB
- + /opt : 512 MB
- + / : sử dụng toàn bộ dung lượng còn lại.

Ngoài ra chúng ta cũng có thể sẽ có một số phân vùng khác tùy theo từng package sẽ sử dụng chúng làm nơi lưu trữ như thế nào. (ví dụ : /cache, /chroot,.....).

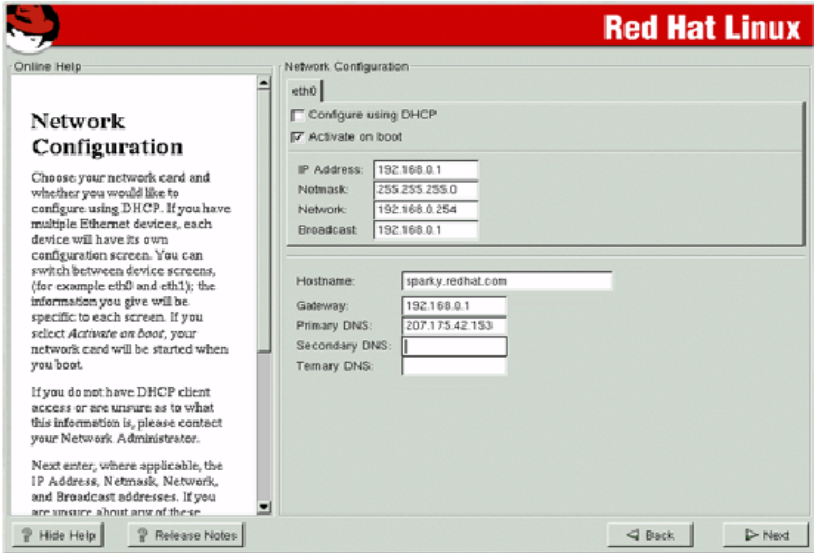
Sau khi tạo xong phân vùng, màn hình tiếp theo sẽ là :



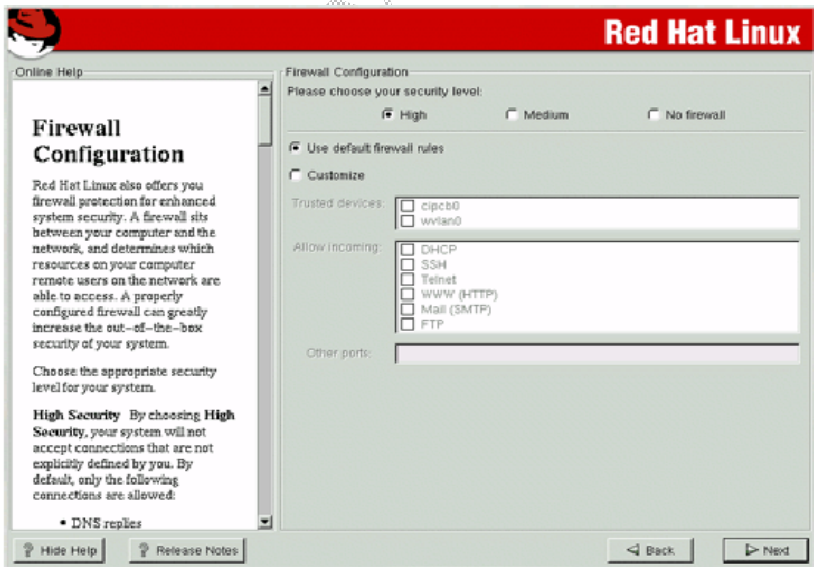
Hệ điều hành sẽ đánh dấu những phân vùng cần format, click NEXT.



Click NEXT và hệ điều hành sẽ dùng LILO làm phần mềm boot default Linux.

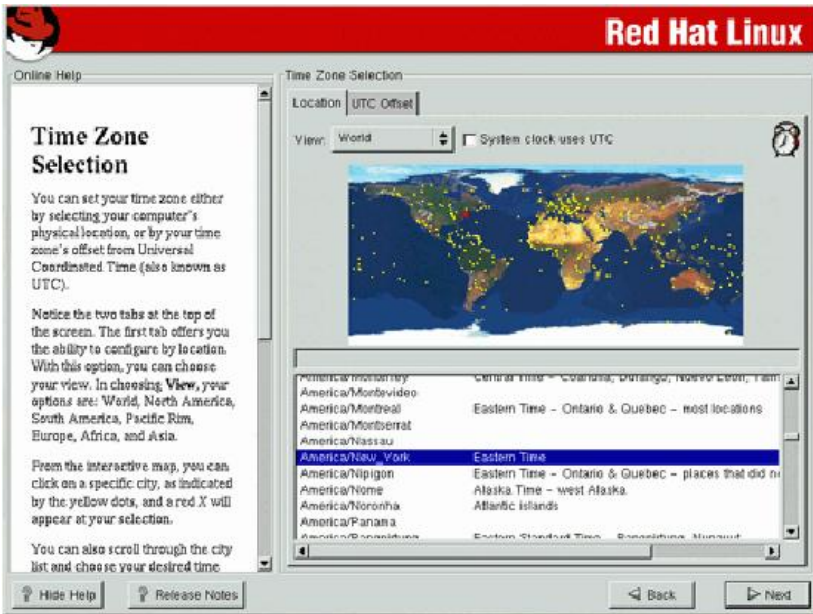


Tiếp theo là chúng ta sẽ quy định những thông số về mạng theo như hình trên. Click NEXT.



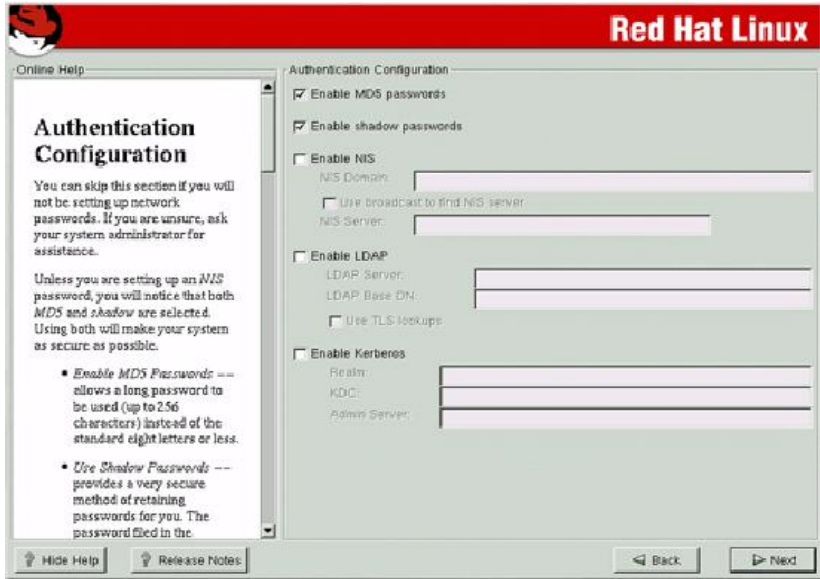
Tới đây, chúng ta sẽ quy định chế độ bảo mật cho hệ thống, theo tùy chọn là “No Firewall”, ở chức năng này, chúng ta có thể thay đổi dễ dàng khi vào trong giao diện của Linux. Click NEXT.

— Đây là tùy chọn để chúng ta quy định những ngôn ngữ mà hệ thống cung cấp. Click NEXT.



Chọn múi giờ khu vực nơi cài đặt. Click NEXT.

Ở đây, chúng ta sẽ quy định mật khẩu của user root(Administrator), và việc tạo một số account khác. Click NEXT.



Cuối cùng là nơi quy định chế độ bảo mật sử dụng cơ chế mã hoá MD5 và Shadow passwords, mặc định là default. Click NEXT.



Tùy theo mục đích sử dụng mà chúng ta sẽ cài đặt những package cần thiết. Click NEXT và hệ điều hành sẽ bắt đầu cài đặt.



Sau khi cài đặt xong, hệ điều hành sẽ bảo chúng ta khởi động lại máy. Xem như quá trình cài đặt đã hoàn tất.

Các lệnh cơ bản và các file cấu hình:

init

Linux cho phép khởi động từ 6 chế độ khác nhau:

Chế độ (Run level)	Trường hợp sử dụng (Common Usages)
0	Tắt máy – shutdown
1	chế độ 1 người dùng (single user)
2	Đa người dùng – không hỗ trợ Network
3	Đa người dùng – có hỗ trợ Network
4	Chưa sử dụng
5	Đa người dùng, network, Graphic
6	Restart

Cú pháp:

```
# init [run level]
```

Ví dụ:

để vào chế độ Graphic của Redhat : # init 5

để vào chế độ textmode đa người dùng: # init 3

- Trong quá trình khởi động, mặc định init sẽ chạy tập tin **/etc/inittab** ở chế độ 3. Bạn có thể chỉnh sửa chế độ khởi động trong tập tin này. Trong mỗi thư mục **/etc/rc.d/rcX.d** sẽ lưu trữ các files, mỗi file này là 1 script mà nếu tên file bắt đầu bằng chữ K có nghĩa là file này không được kích hoạt, nếu bắt đầu bằng chữ S thì file này sẽ được kích hoạt trong chế độ đó.

Shutdown

Cú pháp: # shutdown -h <minute:seconds>

Ví dụ: # shutdown -h 22:00 (tới 10:00pm sẽ tiến hành shutdown máy)

```
# shutdown -h now (shutdown ngay lập tức)
```

```
# shutdown -r now ( restart ngay lập tức )
```

```
# shutdown -h+10 (tiến hành shutdown sau 10 phút nữa )
```

vi

Cú pháp: # vi <tên file>

Ví dụ: # vi /etc/issue

Sau khi vào chương trình soạn thảo của **vi** thì có 2 chế độ cần quan tâm:

Chế độ soạn thảo và chế độ nhập lệnh. Mặc định khi vào Vi là bạn hiện ở chế độ nhập lệnh, nếu muốn vào chế độ soạn thảo thì nhấn phím “i” hoặc “a”. Sau khi ở chế độ soạn thảo mà muốn thoát ra chế độ nhập lệnh thì nhấn “ESC”. Các chức năng của Vi bạn cần quan tâm:

- “i” dùng để bắt đầu ở chế độ soạn thảo văn bản
- Ở chế độ nhập lệnh có các chức năng chính sau
 - + “:w” dùng để lưu đoạn văn bản vừa được thay đổi
 - + “:q” dùng để thoát khỏi Vi
 - + “dd” dùng để xóa 1 dòng tại dấu nhắc con trỏ hiện thời
 - + “/string” dùng để tìm kiếm 1 chuỗi trong đoạn văn bản
 - + “u” undo
 - + Ctrl + F cuộn xuống 1 trang màn hình Vi
 - + Ctrl + B cuộn lên 1 trang màn hình Vi

cat: lệnh dùng xem nội dung của 1 tập tin

Cú pháp: # cat <tên_file_cần_hiển_thị>

Ví dụ: # cat myfile.txt

cat /tmp/temp.text

Logging in and out of a Linux System

Tại dấu nhắc hệ thống phải nhập tên bạn và Password để đăng nhập vào hệ thống Linux.

Dấu nhắc: “#” nghĩa là đang là account root

“\$” nghĩa là đang là account thường

Linux là một hệ điều hành đa người dùng, cho phép nhiều người cùng lúc sử dụng hệ thống bằng nhiều đường khác nhau:

Có 2 loại console để người dùng có thể vào hệ thống Linux: Telnet (vc) và Linux Terminal (tty). Mỗi loại có 11 đường: vc/1 à vc/11 và tty1 à tty11. Để bảo mật hơn, thì nên giới hạn chỉ còn 2 đường cho mỗi loại mà thôi. Cách thức để giới hạn như sau:

Dùng **vi** để modify file **/etc/securetty** như sau:

```
vc/1
vc/2
#vc/3
#vc/4
..
..
#vc/10
#vc/11
tty1
tty2
#tty3
#tty4
..
..
#tty10
#tty11
```

man : hiển thị thông tin chi tiết về công dụng cách dùng các lệnh khác

Cú pháp: # man < tên_lệnh_khác >

Ví dụ: # man ls

man man

(Để kết thúc lệnh man hãy gõ kí tự "q")

ls: liệt kê danh sách tập tin và thư mục hiện thời

Cú pháp: # ls <tham số>

Ví dụ: # ls -la (sẽ hiển thị toàn bộ dạng sách kể cả file ẩn)

cd: thay đổi thư mục hiện thời

Cú pháp: # cd <đường dẫn>

Ví dụ: # cd /root

pwd : hiển thị thư mục hiện thời **Cú pháp: # pwd**

cp: copy tập tin và thư mục

Cú pháp: # cp <tham số> [source] [destination]

Ví dụ: # cp -R /tmp/ /etc/ (copy toàn bộ thư mục tmp sang /etc/)

cp /etc/shadow /tmp/ (copy tập tin shadow sang thư mục /tmp)

mv: di chuyển tập tin và thư mục. Lệnh này sẽ di chuyển hay đổi tên file từ nơi này đến nơi khác

Cú pháp: # mv <file_hoặc_thư_mục_nguồn> <file_hoặc_thư_mục_đích>

mkdir: tạo thư mục

Cú pháp: # mkdir [tên thư mục]

Ví dụ: # mkdir jupiter (tạo thư mục Jupiter)

mkdir -p /etc/1/2/3 (tạo 1 loạt thư mục phả hệ)

rmdir: dung để xóa 1 thư mục

Cú pháp: # rmdir <thư_mục_muốn_xóa>

Ví dụ: # rmdir /tmp

rm: xoá tập tin và thư mục

Cú pháp: # rm <tham số> [tên]

Ví dụ: # rm -rf sẽ xoá sạch nội dung bên trong 1 thư mục => lệnh này rất nguy hiểm, bạn cần kiểm tra lại trước khi xóa

rm -f /etc/khangves.txt

exit và logout: trong text mode, Linux cung cấp cho bạn 6 desktop (tty1...tty6) để làm việc. Để di chuyển qua lại giữa các desktop bằng cách nhấn tổ hợp phím Alt-F1, Alt-F2...,Alt-F6. HAI lệnh này dung để thoát khỏi phiên làm việc desktop trở về màn hình login

Cú pháp: # exit

logout

chown : lệnh này dùng để thay đổi chủ sở hữu của 1 tập tin hay thư mục, gán cho tập tin thư mục thuộc về quyền sở hữu của 1 user nào đó

Cú pháp: # chown username[groupname]

<tên_file_hoặc_thư_mục>

chown .groupname <tên_file_hoặc_thư_mục>

Ví dụ: # chown user1.user /tmp

chown .user /tmp

Nếu chown quyền cho **username** thì file/thư mục sẽ được đặt là thuộc quyền sở hữu của **username** đó. Nếu chown quyền cho **group** thì file/thư mục sẽ thuộc về **group** đó. Hai phần này độc lập với nhau, thay đổi quyền sở hữu **user** sẽ không làm thay đổi quyền sở hữu **group** và ngược lại.

chmod : lệnh này dùng thay đổi thuộc tính của file và thư mục. Có tất cả 3 thuộc tính read, write, execute được áp đặt lên 3 nhóm Owner, Group, Other.

Quyền	Giá trị
r (read)	4
w (write)	2
x (execute)	1

Cú pháp: # chmod thuộc_tính_dạng_số

<tên_file_hoặc_thư_mục>

Ví dụ: # chmod 755 /tmp

Thư mục /tmp owner sẽ có quyền (r,w,x = 4 + 2 + 1=7), group sẽ có quyền (r,x = 4 + 1 = 5), other (r,x = 4 + 1 = 5)

Ta tạo 2 account (u1, u2), 2 account này thuộc nhóm User. Khi login bằng u1 tạo tập tin test.txt, mặc định test.txt sẽ có quyền 700 (chỉ có u1 có toàn quyền trên test.txt). Nếu muốn u2 đọc được test.txt ta phải gán quyền 740 (group User sẽ có quyền đọc tập tin test.txt, u2 thuộc group User sẽ có quyền đọc trên test.txt). Nếu 1 account khác không thuộc group user muốn đọc tập tin test.txt ta phải gán quyền 744 (group other sẽ có quyền đọc)

useradd : dùng để thêm 1 account vào hệ thống

Cú pháp: # useradd <username>

Ví dụ: # useradd usertest

Sau khi tạo một account mới bạn phải đặt password cho account bằng lệnh "passwd"

userdel : xóa 1 account ra khỏi hệ thống

Cú pháp: # userdel <username>

Ví dụ: # userdel usertest

passwd: thay đổi mật mã của 1 account

Cú pháp: # passwd <username>

Ví dụ: # passwd usertest

chkconfig : kiểm tra và bật/tắt các dịch vụ trong Linux

Cú pháp: # chkconfig --<tham số> <tên dịch vụ> <on/off>

Ví dụ: # chkconfig --list (liệt kê danh sách các dịch vụ đang tồn tại)

chkconfig --level 345 kudzu on (bật dịch vụ kudzu ở chế độ 345)

* Khi sử dụng lệnh này sẽ không tác dụng ngay lập tức mà chỉ tác dụng khi bạn khởi động lại máy tương ứng với từng Level

ntsysv: giống chkconfig nhưng ở giao diện GUI

mount : dùng để ánh xạ ổ đĩa vào thư mục bất kì

DỊCH VỤ CUNG CẤP ĐỊA CHỈ IP ĐỘNG (DHCP Server)

2

1. Khái niệm:

Khi quản trị một hệ thống mạng, thường ta phải cung cấp một địa chỉ IP cho mỗi máy tính khác nhau để các máy này có thể liên lạc được với nhau. Với mô hình mạng tương đối nhỏ (khoảng 10 đến 20 máy), việc cung cấp IP cho mỗi máy tính trong mạng thì tương đối dễ dàng cho một quản trị viên, anh ta chỉ việc sử dụng vài thao tác quen thuộc trong việc gán các địa chỉ IP. Nhưng nếu đối với một mô hình mạng lớn (từ 20 máy trở lên) thì việc cung cấp IP như thế là thật sự mệt mỏi và khó khăn rồi, thỉnh thoảng nếu có vấn đề di chuyển thường xuyên giữa những máy tính với nhau thì đây là một công việc khá phức tạp và phí sức.

Chính vì những lý do như thế mà ngày nay, hầu hết trên tất cả các hệ điều hành đều cung cấp cho chúng ta một dịch vụ để giải quyết vấn đề cần thiết trên, đó là dịch vụ cung cấp địa chỉ IP động **DHCP** (*Dynamic Host Configuration Protocol*).

Không những cung cấp được IP mà dịch vụ trên còn đưa ra cho chúng ta nhiều tính năng để cung cấp những yếu tố khác cho các máy client, ví dụ như cung cấp địa chỉ của máy tính dùng để giải quyết tên miền DNS, địa chỉ của một Gateway router, địa chỉ máy WINS .v.v...

Thành phần của một DHCP server bao gồm bốn mục chính sau :

Thành phần	Chức năng
Options	Dùng để cung cấp các yếu tố cho phía client như địa chỉ IP, địa chỉ subnet mask, địa chỉ Gateway, địa chỉ DNS .v.v...
Scope	Một đoạn địa chỉ được quy định trước trên DHCP server mà chúng ta sẽ dùng để gán cho các máy client.
Reservation	Là những đoạn địa chỉ dùng để “để dành” trong một scope mà chúng ta đã quy định ở trên.
Lease	Thời gian “cho thuê” địa chỉ IP đối với mỗi client.

2. Cài đặt:

Để sử dụng được dịch vụ DHCP này, bạn phải cài đặt vào hệ thống thông thường bằng gói dịch vụ có sẵn trên đĩa CD có phần đuôi mở rộng là *.rpm*, ngoài ra chúng ta có thể cài đặt package ở dạng *source code* và tải gói này về từ trang web của GNU. Quá trình cài đặt bao gồm những bước sau đây :

- Ở dạng phần đuôi mở rộng là *.rpm*, ta chạy lệnh:
`rpm -ivh dhcp-*.rpm`
- Ở dạng source code, ta biên dịch như sau :
`tar -xzf dhcp-*.tar.gz`
`cd dhcp-*`
`./configure`
`make`
`make install`

- Sau khi hoàn tất xong quá trình cài đặt, kế tiếp chúng ta sẽ cấu hình để dịch vụ này có thể hoạt động theo ý muốn của chúng ta bằng cách tạo và sửa đổi file ***/etc/dhcpd.conf***. Tập tin này sẽ có những nội dung sau :

```
deny client-updates;
ddns-update-style interim;

subnet 192.168.0.0 netmask 255.255.255.0 {
    range dynamic-bootp 192.168.0.190
    192.168.0.240;

    option routers 192.168.0.10;
    option subnet-mask 255.255.255.0;

    option nis-domain "mydomain.com";
    option domain-name "mydomain.com";
    option domain-name-servers 192.168.0.20;
    option netbios-name-servers 192.168.0.100;
    option ntp-servers 192.168.0.25;
    option smtp-server 192.168.0.35;
```

```
default-lease-time 360000;  
max-lease-time 259200;  
}
```

Client-definitions

```
host big-daddy {  
    hardware ethernet 00:a0:d9:cb:94:8a;  
    fixed-address 192.168.0.18;  
}
```

- Các dòng trên có ý nghĩa như sau :
 - Hai dòng đầu tiên sẽ không cho phép DHCP Server cập nhật động DNS.
 - Dòng kế tiếp là đoạn địa chỉ mà bạn cần cung cấp cho hệ thống các máy con của bạn, bao gồm địa chỉ NET IDs và một đoạn địa chỉ. (Như ở trên Server sẽ cấp cho phía máy con một đoạn địa chỉ chạy từ 192.168.0.190 đến 192.168.0.240)
 - Option routers cung cấp cổng gateway mặc định.
 - Option subnet-mask Subnet mask mặc định cho phía client.
 - Option nis-domain cung cấp tên NIS Domain Server
 - Option domain-name cung cấp tên domain mặc định nếu sử dụng FQDN
 - Option domain-name-servers cung cấp name-servers cho mạng của bạn.
 - Option netbios-name-servers cung cấp địa chỉ mặc định của WINS-server
 - Option ntp-servers cung cấp địa chỉ timeserver.
 - Option smtp-server cung cấp địa chỉ smtp-server (**duy nhất chỉ 1 server**)
 - Dòng cuối cùng là nếu bạn dự định cấp một địa chỉ cố định cho một máy nào đó thì bạn

phải khai báo địa chỉ MAC của máy đó và IP tương ứng

- Và trước khi khởi động DHCP Server lên thì bạn phải tạo một tập tin cuối cùng dùng để xem xét việc cấp phát các địa chỉ IP cho phía client:

touch /etc/dhcpd.lease

- Để bật tắt dịch vụ DHCP thì bạn chỉ chạy hai script tương ứng như sau:

/etc/init.d/dhcpd start

/etc/init.d/dhcpd stop

mcsevietnam

3

DNS Server: Bind

Đây là dịch vụ cơ bản đầu tiên và quan trọng nhất của Internet. DNS là quan trọng vì nếu DNS hoạt động sai hoặc không hoạt động, toàn bộ phần mạng Internet liên quan sẽ bị tê liệt hoàn toàn. Hiểu rõ DNS rất quan trọng với quản trị viên máy chủ có kết nối Internet. Nó cho phép quản trị viên tìm ra nhanh chóng các nguyên nhân của các trục trặc trên mạng.

DNS nói một cách đơn giản là dịch vụ cho phép ánh xạ, chuyển đổi tên của một hệ thống nối Internet ra địa chỉ IP của nó. Nguyên nhân của sự tồn tại DNS là do con người có thói quen đặt tên cho các trang thiết bị mà các trang thiết bị thì lại chỉ có thể dùng số để liên lạc với nhau. Vào những thời kỳ đầu tiên của Internet, người ta lập bảng về mối liên hệ giữa tên và địa chỉ IP và cài đặt trên một máy tính để tất cả cùng tham khảo. Nhưng với sự phát triển quá nhanh của Internet, bảng này phát triển nhanh chóng và không một máy nào có thể hoàn thành nổi nhiệm vụ tuy đơn giản nhưng lại rất quan trọng này. Hơn nữa, mỗi thay đổi dù ở đâu cũng phải thông qua server trung tâm. Điều này trở nên không thể chấp nhận được vì luôn có thay đổi trên Internet. Một giải pháp được cộng đồng Internet chấp nhận là chia toàn bộ không gian các địa chỉ IP và tên ra thành các nhóm logic nhỏ hơn. Mỗi nhóm có quyền tổ chức thông tin của các máy của mình.

Như vậy bước đầu tiên, một máy nối vào Internet, không phụ thuộc vào việc nó có chạy hay không DNS server, phải được cấu hình resolver, tức là chỉ ra cách thức hành động khi có yêu cầu phân giải địa chỉ. Resolver được cấu hình qua tập tin **/etc/host.conf** :

```
[root@pasteur tnminh]# more /etc/host.conf
order hosts,bind
multi on
```

- Dòng thứ nhất của **/etc/host.conf** cho biết khi có yêu cầu phân giải tên, resolver sẽ xem xét đầu tiên tập tin **/etc/hosts** sau đó đến sử dụng DNS server (bind).

- Dòng thứ hai cho phép một host có nhiều địa chỉ IP trong tập tin `/etc/hosts`.

Tập tin `/etc/hosts` chính là tiền thân của dịch vụ DNS. Hiện nay, `/etc/hosts` chỉ còn thường lưu các địa chỉ của mạng nội bộ hay dùng tới nhất đối với một máy. Khi yêu cầu phân giải vượt qua khả năng trả lời của `/etc/hosts` từ khóa **bind** chỉ ra cần phải sử dụng dịch vụ DNS. BIND là viết tắt của Berkeley Internet Name Domain và một triển khai rộng rãi nhất của dịch vụ DNS hiện nay.

Khi đó, resolver cần thông tin tiếp theo về DNS server. Thông tin này lưu trữ trong tập tin `/etc/resolv.conf`. Tập tin này kiểm tra cách resolver sử dụng DNS để phân giải địa chỉ. Nó quyết định DNS server cụ thể cần phải truy vấn và cách bổ sung phần domain cho phần tên của máy. Ví dụ một tập tin `/etc/resolv.conf`

```
[root@linuxsrv root]# more /etc/resolv.conf
search mcsevietnam.com
nameserver 192.168.2.10
[root@linuxsrv root]#
```

Dòng đầu tiên cho phép resolver không chỉ phân giải tên như chương trình client yêu cầu, mà trong trường hợp phân giải không thành công, tiếp tục thử phân giải tên với phần domain tiếp nối sau. Ví dụ bạn muốn tìm địa chỉ máy *khangves*. Nếu quá trình phân giải *khangves* không thành công, resolver sẽ thử phân giải *khangves.mcsevietnam.com*. Dòng tiếp theo là địa chỉ của name server cần phải truy vấn. Nhớ rằng địa chỉ của name server là số IP chứ không phải là tên, vì nếu ngược lại, ai sẽ là người phân giải tên cho máy làm nhiệm vụ phân giải tên?

Bây giờ chúng ta sẽ chuyển qua xem xét đến cấu hình của bản thân name server. Chương trình server của DNS name server là một chương trình daemon named (đọc là nê-m-ê). Named thường được khởi động ngay từ đầu cùng với khởi động của hệ thống. Thường thì named được chạy thông qua một script trong `/etc/rc.d/rc3.d/named`. Trong quá trình khởi động named đọc các tập tin dữ liệu rồi chờ các yêu cầu phân giải qua cổng xác định trong tập tin `/etc/service` (thông thường là cổng 53). Named dùng đầu tiên là giao thức UDP để phân giải tên, nếu phân giải bằng UDP không có kết quả, named sẽ dùng TCP sau đó.

Tập tin đầu tiên được named tham chiếu là /etc/named.conf. Nội dung tập tin này của Linux Redhat 7.3 được cài mặc định là :

```
options {
    directory "/var/named";
};
zone "." {
    type hint;
    file "root.hints";
};
zone "0.0.127.in-addr.arpa" {
    type master;
    file "pz/127.0.0";
};
```

Mở đầu là từ khóa **options** cho phép nhập các tùy chọn (options) toàn cục. *directory "/var/named";* cho biết là các tập tin sau đây sẽ là tương đối đối với thư mục này.

Ta có thể bổ sung thêm trong phần *options* dòng lệnh :
forwarders {205.15.2.10 ; 193.214.2.12};

Khi đó, DNS server của chúng ta sẽ tham chiếu các name server **205.15.2.10; 193.214.2.12** mỗi khi nó không tìm thấy câu trả lời trong dữ liệu mà nó có. Sau phần tham số toàn cục options, ta thấy các khối zone "tên_zone" { type master (hoặc slave hoặc hint); file "tên_tập_tin"; }; liên tiếp nhau.

Đối với mỗi domain, chúng ta cần 2 tập tin dữ liệu. Tập tin thứ nhất lưu trữ các dữ liệu liên quan đến phân giải "xuôi" từ name sang IP và tập tin thứ hai để phân giải "ngược" từ IP ra name. Trừ miền "." có tính chất giúp đỡ là có tập tin cache đặc biệt

; There might be opening comments here if you already have this file.

; If not don't worry.

```
;
. 6D IN NS G.ROOT-SERVERS.NET.
. 6D IN NS J.ROOT-SERVERS.NET.
. 6D IN NS K.ROOT-SERVERS.NET.
. 6D IN NS L.ROOT-SERVERS.NET.
. 6D IN NS M.ROOT-SERVERS.NET.
```

```
. 6D IN NS A.ROOT-SERVERS.NET.  
. 6D IN NS H.ROOT-SERVERS.NET.  
. 6D IN NS B.ROOT-SERVERS.NET.  
. 6D IN NS C.ROOT-SERVERS.NET.  
. 6D IN NS D.ROOT-SERVERS.NET.  
. 6D IN NS E.ROOT-SERVERS.NET.  
. 6D IN NS I.ROOT-SERVERS.NET.  
. 6D IN NS F.ROOT-SERVERS.NET.  
G.ROOT-SERVERS.NET. 5w6d16h IN A 192.112.36.4  
J.ROOT-SERVERS.NET. 5w6d16h IN A 198.41.0.10  
K.ROOT-SERVERS.NET. 5w6d16h IN A 193.0.14.129  
L.ROOT-SERVERS.NET. 5w6d16h IN A 198.32.64.12  
M.ROOT-SERVERS.NET. 5w6d16h IN A 202.12.27.33  
A.ROOT-SERVERS.NET. 5w6d16h IN A 198.41.0.4  
H.ROOT-SERVERS.NET. 5w6d16h IN A 128.63.2.53  
B.ROOT-SERVERS.NET. 5w6d16h IN A 128.9.0.107  
C.ROOT-SERVERS.NET. 5w6d16h IN A 192.33.4.12  
D.ROOT-SERVERS.NET. 5w6d16h IN A 128.8.10.90  
E.ROOT-SERVERS.NET. 5w6d16h IN A 192.203.230.10  
I.ROOT-SERVERS.NET. 5w6d16h IN A 192.36.148.17  
F.ROOT-SERVERS.NET. 5w6d16h IN A 192.5.5.241
```

Đây thực chất là địa chỉ IP của các name server gốc (root) của Internet.

Ví dụ như đối với miền **mcsevietnam.com** ta cần có :

```
zone "mcsevietnam.com" {  
    type master;  
    file "db.mcsevietnam.com";  
};  
zone "1.16.172.in-addr.arpa" {  
    type master;  
    file "db.172.16.1";
```

Chú ý các viết cú pháp *1.16.172.in-addr.arpa* cho tên của miền phân giải ngược IP ra name.

Sau đây ta sẽ xem xét đến cấu trúc tập tin

```
/var/named/db.mcsevietnam.com
```

```
@ IN SOA mcsevietnam.com. root.mcsevietnam.com. (  
199609206 ; serial, todays date + todays serial #  
8H ; refresh, seconds
```

```
2H ; retry, seconds
1W ; expire, seconds
1D ) ; minimum, seconds
NS mcsevietnam.com.
MX 10 mcsevietnam.com. ; Primary Mail Exchanger
TXT "MCSEVIETNAM Corporation"
localhost A 127.0.0.1
mcsevietnam.com. A 172.16.1.1
linuxsrv A 172.16.1.1
www A 172.16.1.1
ftp CNAME mcsevietnam.com.
mail CNAME mcsevietnam.com.
news CNAME mcsevietnam.com.
```

Ký tự "@" đầu tiên thay cho miền *mcsevietnam.com*; IN là Internet ; SOA là Start Of Authority; tiếp nối bởi tên miền và địa chỉ người chịu trách nhiệm. Chú ý là trong địa chỉ email của người chịu trách nhiệm, dấu @ quen thuộc được thay bằng dấu chấm ".". Sau các tên miền có dấu chấm "." ở cuối. Trong tất cả các tập tin dữ liệu của DNS, những tên không kết thúc bởi dấu chấm sẽ được DNS server thêm vào bởi tên miền tương ứng của tập tin đó. Ví dụ đây là tập tin ứng với miền *mcsevietnam.com*, **khangves** sẽ được bổ sung thêm thành *khangves.mcsevietnam.com*.

Sau phần ngoặc đơn với 5 số miêu tả số serie và các thông số thời gian của thông tin, bắt đầu các dòng (record) dữ liệu. **Khoảng trắng** ở đầu dòng tương đương với tên miền (như dấu @), **NS** ám chỉ record dạng nameserver. **MX** là mail exchange, dùng để chỉ ra máy chịu trách nhiệm nhận thư điện tử cho domain này. Số 10 là mức độ ưu tiên cho mail server này. Độ ưu tiên sẽ càng cao nếu số càng nhỏ. **A** là viết tắt của Address, sẽ tiếp theo bởi một địa chỉ IP. CNAME là canonical name. Với CNAME ta có thể gán cho máy biệt danh tùy ý tiện cho việc sử dụng. Các dòng bắt đầu bởi ; là các chú thích.

Ví dụ tập tin dùng cho phân giải ngược **/var/named/db.172.16.1**
@ IN SOA mcsevietnam.com. root.mcsevietnam.com. (
199609206 ; Serial
28800 ; Refresh
7200 ; Retry
604800 ; Expire

```
86400) ; Minimum TTL
NS mcsevietnam.com.
;
; Servers
;
1 PTR simbahcm.mcsevietnam.com.
2 PTR trantungbtre.mcsevietnam.com.
3 PTR hungden.mcsevietnam.com.
;
```

Cấu trúc tập tin `/var/named/db.172.16.1` có phần đầu giống hệt như tập tin phân giải xuôi. Chỉ có từ khóa **PTR** = *Pointer* là khác.

Việc cấu hình các dữ liệu của name server cần rất thận trọng vì nhiều khi lỗi của nó rất khó tìm. Mỗi khi chúng ta thay đổi dữ liệu, cần phải khởi động lại named bằng các sử dụng `kill -9 named_PID` để dừng named rồi khởi động lại bằng cách nhập dòng lệnh named. Tập tin `/var/log/messages` có thể giúp đỡ nhiều để tìm ra lỗi nếu named không hoạt động theo ý chúng ta muốn. Để thử hoạt động của quá trình phân giải tên, Linux có lệnh **nslookup** với nhiều tính năng rất mạnh. Xem manpage của nslookup để biết cách sử dụng.



SAMBA

I. Giới thiệu

1. **Khái niệm:**

Ngày nay nhu cầu chia sẻ tài nguyên trong mạng nội bộ là không thể thiếu. Chia sẻ đĩa, chia sẻ thư mục, máy in dùng chung trong mạng nội bộ. Trong bài này hướng dẫn nối mạng Linux với Windows sử dụng giao thức *Server Message Block (SMB)*, hay còn gọi là *Session Message Block* để giao tiếp và chia sẻ tập tin, máy in lẫn nhau. Sử dụng chương trình Samba để đáp ứng nhu cầu trên. Biểu tượng Linux PC xuất hiện trong Windows Network Neighborhood.

2. **Samba:** giao thức *Server Message Block (SMB)*, hay còn gọi là *Session Message Block*

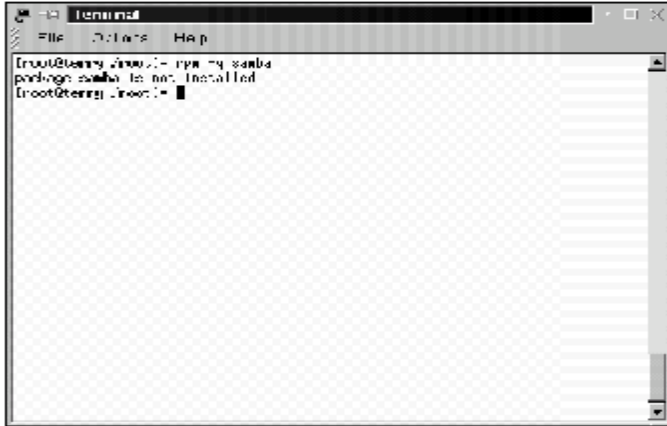
Giao thức SMB được dùng để chia sẻ đĩa và máy in cho Microsoft Windows 3.11, NT và 95/98. Sử dụng công cụ Samba trên Linux có thể chia sẻ tài nguyên của Linux cho Windows. Bốn điều cơ bản Samba có thể làm:

- Chia sẻ đĩa Linux cho Windows
- Chia sẻ SMB với máy Linux
- Chia sẻ máy in trên Linux cho Windows
- Chia sẻ máy in trên Windows cho Linux

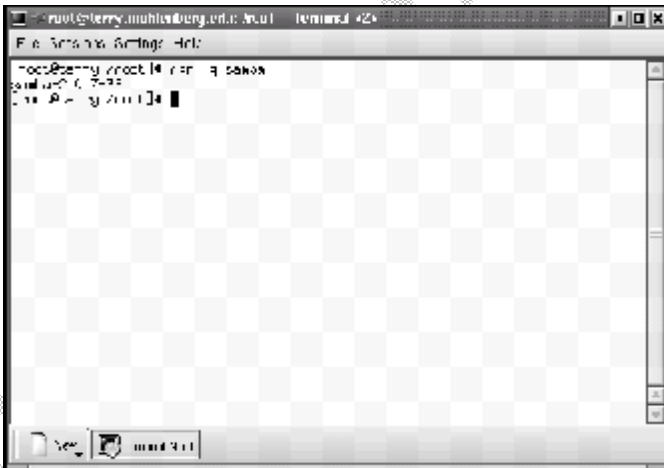
II. Cài đặt

1. **Cài đặt và cấu hình Samba**

- Kiểm tra xem Samba đã cài chưa
rpm -qi samba
- + Nếu chưa cài thì màn hình terminal sẽ trả về



+ Nếu đã cài màn hình terminal sẽ trả về



Thư mục cài Samba

Directory	Miêu tả
<i>/usr/local/samba</i>	Thư mục chính
<i>/usr/local/samba/bin</i>	Binaries
<i>/usr/local/samba/lib</i>	<i>smb.conf</i> , <i>lmhosts</i> , configuration files, etc.

Thư mục cài Samba	
Directory	Miêu tả
<code>/usr/local/samba/man</code>	Tài liệu hướng dẫn Samba
<code>/usr/local/samba/private</code>	File password đã mã hóa
<code>/usr/local/samba/swat</code>	Files SWAT
<code>/usr/local/samba/var</code>	Samba log files, lock files, browse list info, shared memory files, process ID files

- Nếu chưa cài Samba bạn có thể vào website www.samba.org theo hướng dẫn của trang web để tải tập tin RPM. Để cài đặt dùng lệnh
rpm -i samba
- Từ Version 2.0 trở đi Samba kèm theo tên ích Swat (công cụ quản trị Samba qua giao diện Web) , công cụ này cho phép cấu hình Samba một cách dễ dàng. Swat cho phép bạn dùng trình duyệt web thay đổi trực tiếp lên tập tin cấu hình chính của Samba `/etc/smb.conf`
- File cấu hình chính Samba `/etc/samba/smb.conf`

```
# Samba config file created using SWAT
# from localhost (127.0.0.1)
# Date: 2000/05/25 10:29:40
# Global parameters
[global]
    workgroup = ONE
    netbios name = TERRY
    server string = Samba Server
    security = SHARE
    log file = /var/log/samba/log
    max log size = 50
    socket options = TCP_NODELAY
    SO_RCVBUF=8192 SO_SNDBUF=8192
    wins support = Yes
    hosts allow = 192.168.1.
    hosts deny = all
```

```
[homes]
    comment = Home Directories
    read only = No

[printers]
    comment = All Printers
    path = /var/spool/samba
    guest ok = Yes
    print ok = Yes
    browseable = Yes

[test]
    path = /tmp/sambatest
    valid users = test
    read only = no
    guest ok = no
    browseable = yes
```

[global]

[Global] là phần đầu tiên của smb.conf, mỗi phần trong smb.conf gồm lựa chọn và giá trị định dạng: option = values. Bạn có hàng trăm lựa chọn và giá trị định dạng khác nhau. Dưới đây là những định dạng chung nhất

- **Workgroup = HUNG** tên của workgroup xuất hiện trong network properties trên máy windows
- **Netbios name = Linux** là tên mà Samba server sẽ được biết bởi máy windows
- **Server string = Samba Server** là tên của Samba server
- **Security = SHARE** mức độ quyền trên Server, các mức độ khác: User, Default, Domain, Server. Sử dụng Share sẽ dễ dàng tạo chia sẻ cho anonymous, không cần chứng thực.
- **Log_file = /var/log/samba/log** thư mục chứa tập tin log
- **max log size = 50** dung lượng tối đa của tập tin log tính bằng KB
- **socket options = TCP_NODELAY SO_RCVBUF=8192 SO_SNDBUF=8192** tối ưu hóa server
- **wins support = Yes** samba server đóng vai trò là Wins Server

- **hosts allow = 192.168.1.** chỉ cho phép yêu cầu từ network này
- **hosts deny = all** không nhận yêu cầu từ tất cả các host

[Homes]

Lựa chọn này cho phép người dung nhanh chóng truy nhập vào thư mục home của họ

- **comment = Home Directories** ghi chú
- **read only = No** người dung có toàn quyền trong thư mục home của họ

[printers]

Thiết lập lựa chọn máy in

- **Path = /var/spool/samba** thư mục của máy in
- **Guest ok = Yes** cho phép guest truy cập vào máy in
- **Print ok = Yes** cho phép người dùng sử dụng máy in
- **Browseable = Yes** biểu tượng máy in sẽ xuất hiện trong browse list

[test]

Cấu hình chia sẻ thư mục test trên Linux

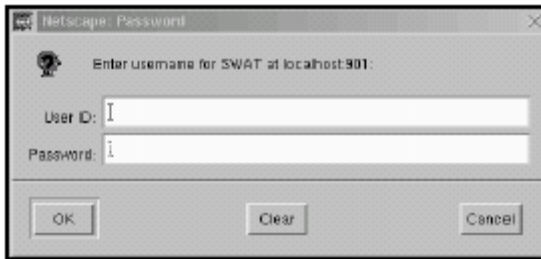
- **Path = /tmp/sambatest** đường dẫn thư mục chia sẻ
- **Valid users = test** chỉ định người dùng sử dụng thư mục này
- **Read only = No** cho phép quyền ghi trên thư mục
- **Guset ok = No** không cho guest quyền truy nhập
- **read only = No** người dung có toàn quyền trong thư mục home của họ
- **Browseable = Yes** thư mục share sẽ xuất hiện trong browse list

2. Sử dụng Swat:

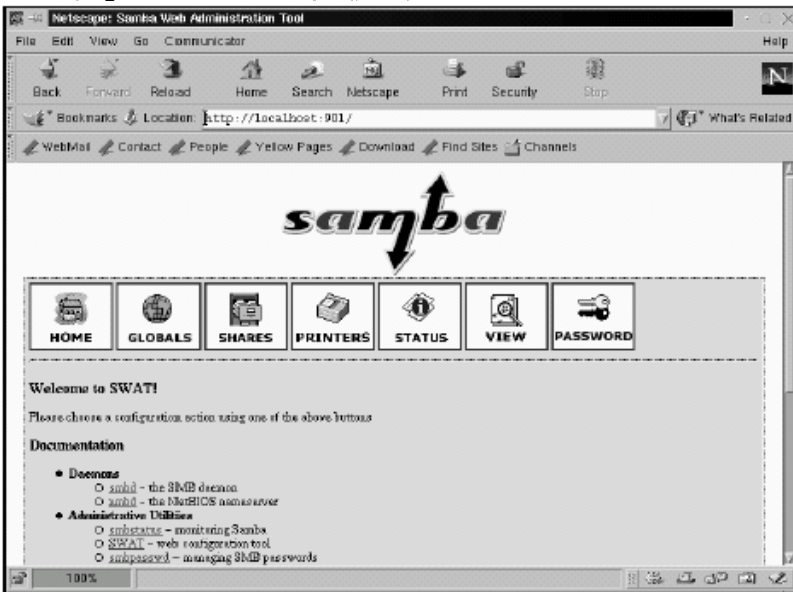
Trước khi có thể sử dụng Swat cần thay đổi 2 tập tin để bật tiện ích này lên

+ Thêm vào /etc/services

- Swat 901/tcp
- + Thêm vào /etc/inetd.conf
Swat stream tcp nowait.400
root /usr/sbin/swat swat
- + khởi động lại Inetd
killall -HUP inetd
- Sử dụng trình duyệt web để chạy Swat <http://localhost:901>
.Hộp thoại yêu cầu nhập User ID và mật khẩu xuất hiện, đăng nhập với quyền root:



- Đầu tiên bạn phải cấu hình [globals] bằng cách bấm vào biểu tượng GLOBALS



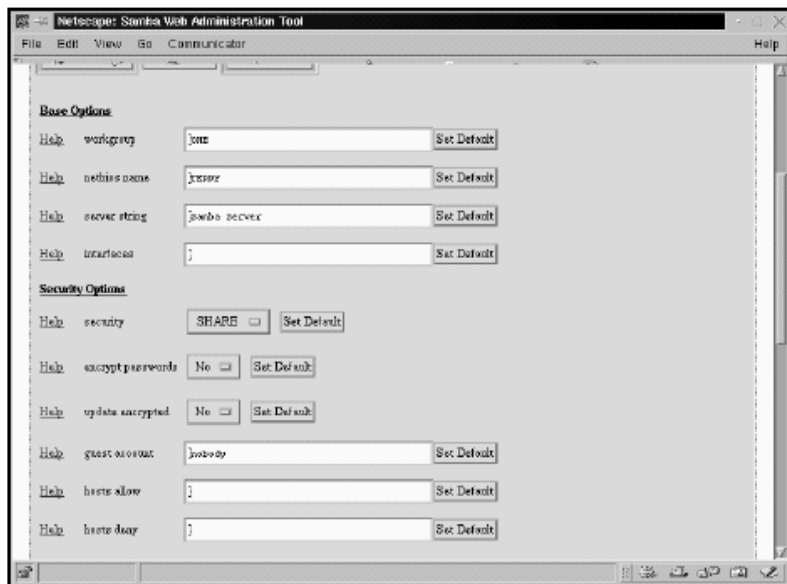
Những biến Global xuất hiện. Giá trị này là giá trị file smb.conf



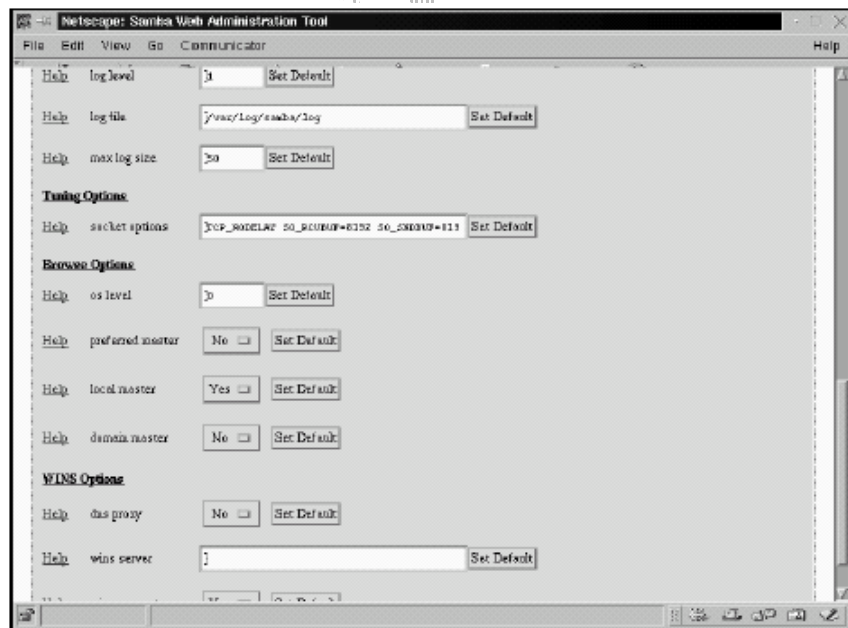
Trang Global Variables cho chúng ta để cấu hình [Globals] trong file smb.conf

Trang Global Variables chia thành 6 lựa chọn

- Base Options
- Security Options
- Logging Options
- Tuning Options
- Browse Options
- WINS Options



Base e Security Options



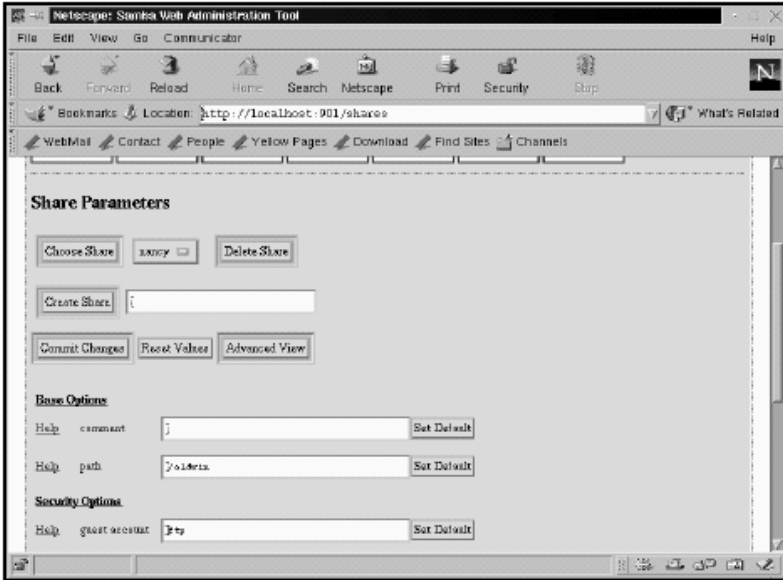
Log, tuning, browse, và WINS options
Sau khi điền vào những giá trị cần thiết, bấm vào Commit Changes để lưu thay đổi

- Tiếp theo chọn biểu tượng SHARES để mở trang Share Parameters



Trang Share Parameters

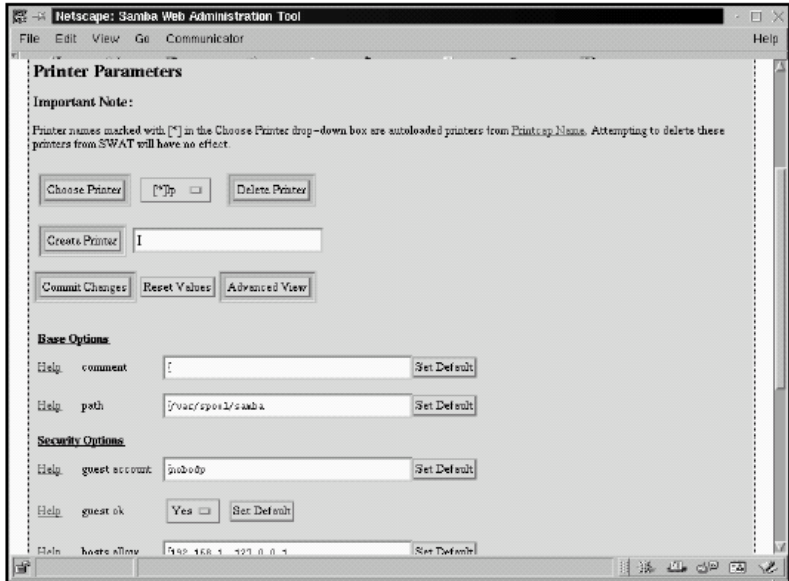
Để tạo chia sẻ điền vào tên share và nhấn nút Create Share



Điền vào những thông tin cấu hình để Windows có thể truy cập vào Samba server

Sau khi hoàn tất nhấn Commit Changes để lưu vào file smb.conf

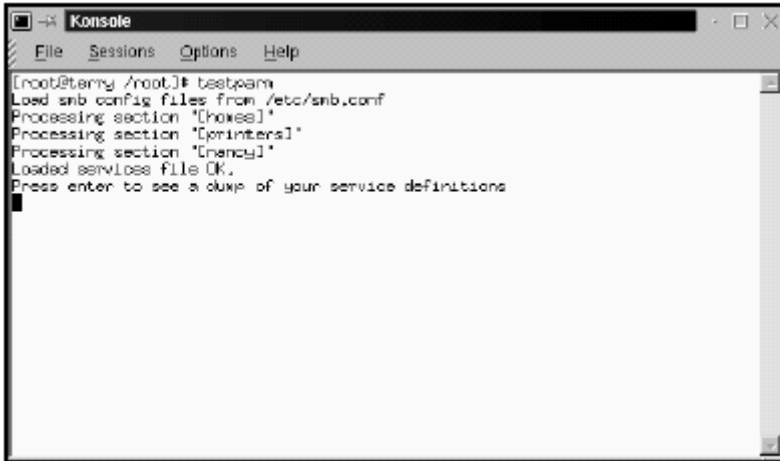
- Tiếp theo chia sẻ máy in cho máy Windows sử dụng. Chọn biểu tượng PRINTERS



Hiển thị tên máy in mà bạn đã chọn

Để tạo mới chọn Create Printer, nếu bạn đã có sẵn máy in bạn có thể chọn từ menu Drop-down. Chú ý nếu bạn đã cài sẵn máy in trong RedHat, nó sẽ được sử dụng như máy in mặc định trong samba và không thể xóa. Nhấn vào Commit Changes để lưu lại vào smb.conf

- Sau khi đã hoàn tất sử dụng tiện ích testparm để kiểm tra lại. Từ màn hình dòng lệnh gõ vào: testparm

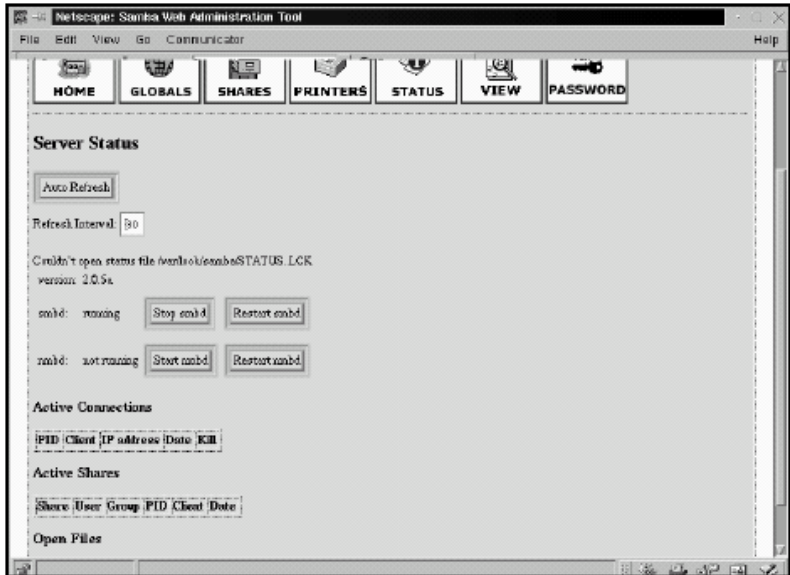


```
[root@terry ~root]# testparm
Load smb config files from /etc/smb.conf
Processing section "[homes]"
Processing section "[printers]"
Processing section "[nancy]"
Loaded services file OK.
Press enter to see a dump of your service definitions
```

tiện ích testparm kiểm tra lỗi tập tin smb.conf

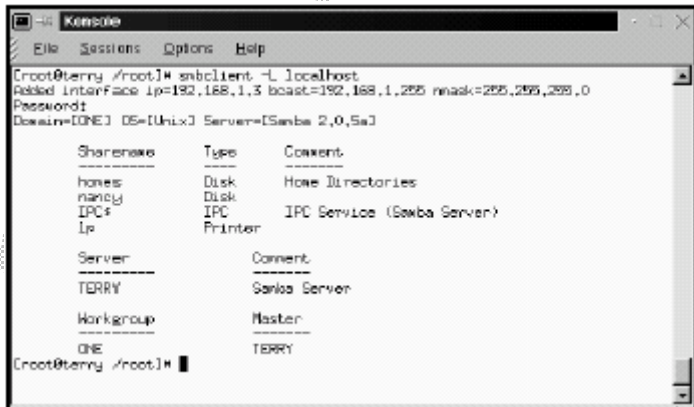
- Sau khi thay đổi file smb.conf, bạn phải khởi động lại samba. Khởi động Samba bằng dòng lệnh: **/usr/sbin/samba start** hoặc **/etc/init.d/samba start**. Để khởi động Samba bằng Swat chọn biểu tượng STATUS. 2 dịch vụ smbld và nmbd phải được khởi động.

mcse



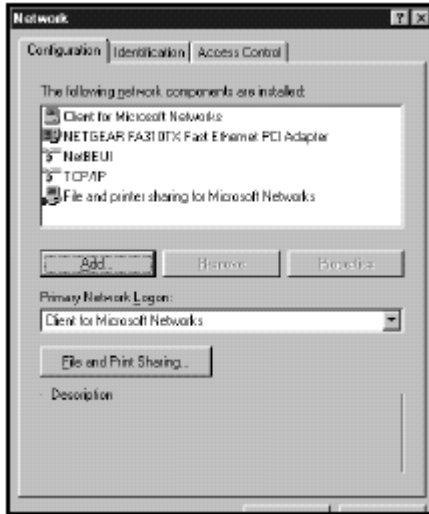
Trang Server Status cho biết hiện trạng của samba server

- Sau khi Samba khởi động, dùng lệnh smbclient trên localhost để thấy thông tin cấu hình samba: **smbclient -L localhost**



3. Cấu hình Samba Client

Trên máy Windows Client phải được cài “Client for Microsoft Network” và “File and printer sharing for Microsoft Networks”



Hộp thoại Network Properties

4. Kiểm tra Samba server

Bạn hãy kiểm tra lại mọi thứ bạn đã làm và chắc chắn rằng sẽ không có sai sót. Trên máy Windows -> Network Neighborhood .Trong cửa sổ Network Neighborhood bạn có thể thấy được danh sách máy Windows, những thư mục chia sẻ, bạn cũng sẽ thấy Linux Server. Trên máy Linux bạn cũng có thể truy cập vào thư mục Windows bằng lệnh smbclient: **smbclient //tên máy tính/tên thư mục**

III. Kết luận

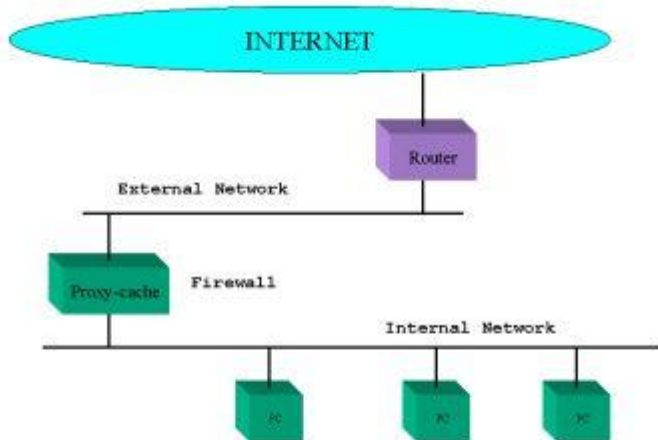
Vậy là bạn có thể cấu hình Samba server để kết nối máy tính dùng HĐH Linux với Windows thông qua giao thức SMB (Server Message Block). Bạn cũng có thể dùng NFS (Network File System) để chia sẻ file trong mạng nội bộ, nhưng sẽ dễ cấu hình hơn nếu dùng Samba.

DỊCH VỤ ỦY QUYỀN (Proxy)

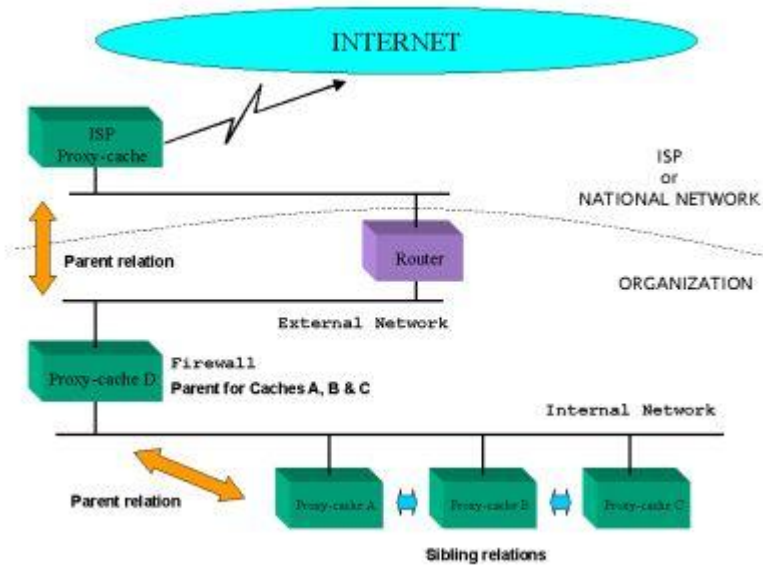
5

1/. Khái niệm:

- **Proxy** cho phép bạn chỉ cần một máy hoặc một nhóm nhỏ các máy để trợ giúp cho việc truy cập Internet cho tất cả các máy của bạn. Sử dụng proxy có hai lợi thế quan trọng, thứ nhất là bạn chỉ cần ít, hay một địa chỉ IP chính thức mà lại có thể cho nhiều máy cùng được truy cập Internet, thứ hai là nếu một trang Web đã được lấy về, nó sẽ được lưu trên đĩa của máy proxy và khi có một yêu cầu khác lấy đúng trang web đó, proxy không cần phải ra Internet lấy dữ liệu nữa mà lấy thẳng từ trong đĩa cứng của mình đã được lưu lại lần trước đó và như vậy sẽ tiết kiệm được đường kết nối ra Internet thường rất mắc tiền và bận bịu. Mô hình sử dụng Proxy có thể được minh họa như sau :



- Nhưng với cấp độ lớn hơn, chúng ta có thể sử dụng mô hình cache nhiều lớp bao gồm những nhóm máy Proxy tương tự như sau :



2/. Linux SQUID Proxy Server:

- **Squid** là một proxy server, khả năng của squid là tiết kiệm băng thông(bandwidth), cải tiến việc bảo mật, tăng tốc độ truy cập web cho người sử dụng và trở thành một trong những proxy phổ biến được nhiều người biết đến. Hiện nay, trên thị trường có rất nhiều chương trình proxy-server nhưng chúng lại có hai nhược điểm, thứ nhất là phải trả tiền để sử dụng, thứ hai là hầu hết không hỗ trợ **ICP** (ICP được sử dụng để cập nhật những thay đổi về nội dung của những URL sẵn có trong cache – là nơi lưu trữ những trang web mà bạn đã từng đi qua). Squid là sự lựa chọn tốt nhất cho một proxy-cache server, squid đáp ứng hai yêu cầu của chúng ta là sử dụng miễn phí và có thể sử dụng đặc trưng ICP.
- Squid đưa ra kỹ thuật lưu trữ ở cấp độ cao của các web client, đồng thời hỗ trợ các dịch vụ thông thường như FTP, Gopher và HTTP. Squid lưu trữ thông tin mới nhất của các dịch vụ trên trong RAM, quản lý một cơ sở dữ liệu lớn của các thông tin trên đĩa, có một kỹ thuật điều khiển truy cập phức tạp, hỗ trợ giao thức SSL cho các kết nối bảo mật thông qua proxy. Hơn nữa, squid có thể liên kết

với các cache của các proxy server khác trong việc sắp xếp lưu trữ các trang web một cách hợp lý.

- Sau đây chúng ta sẽ thực hiện cách thức cài đặt một Proxy server như thế nào.

3./ Cài đặt:

- Đầu tiên chúng ta nên có một số khái niệm về đòi hỏi phần cứng của một proxy server:

*** Tốc độ truy cập đĩa cứng : rất quan trọng vì squid thường xuyên phải đọc và ghi dữ liệu trên ổ cứng. Một ổ đĩa SCSI với tốc độ truyền dữ liệu lớn là một ứng cử viên tốt cho nhiệm vụ này.

*** Dung lượng đĩa dành cho cache phụ thuộc vào kích cỡ của mạng mà Squid phục vụ. Từ 1 đến 2 Gb cho một mạng trung bình khoảng 100 máy. Tuy nhiên đây chỉ là một con số cố tính chất ví dụ vì nhu cầu truy cập Internet mới là yếu tố quyết định sự cần thiết độ lớn của đĩa cứng.

*** RAM : rất quan trọng, ít RAM thì Squid sẽ chậm hơn một cách rõ ràng.

*** CPU : không cần mạnh lắm, khoảng 133 MHz là cũng có thể chạy tốt với tải là 7 requests/second.

- Cài đặt Squid với RedHat Linux rất đơn giản. Squid sẽ được cài nếu bạn chọn nó trong quá trình cài đặt ngay từ đầu. Hoặc nếu bạn đã cài Linux không Squid, bạn có thể cài sau qua tiện ích *rpm* với lệnh :

```
rpm -i tên_gói_Squid
```

Khi đó squid sẽ được cài và bạn có thể bước qua phần cấu hình squid.

- Các thư mục mặc định của squid:

```
/usr/sbin  
/etc/squid  
/var/log/squid
```

- **Cài đặt từ source :**

+ Ta có file source của squid là squid-version.tar.gz, ta thực hiện các bước lệnh sau:

```
tar -xvf squid-version.tar.gz  
cd squid-version  
./configure  
make  
make install
```

Sau khi ta thực hiện các lệnh trên, coi như ta đã cài đặt xong squid.

3./ Cấu hình Squid:

- Sau khi cài đặt xong squid, ta phải cấu hình squid để phù hợp với từng yêu cầu riêng. Ta cấu hình một số tham số trong file /etc/squid/squid.conf như sau:

** *http_port*: mặc định là 3128.

** *icp_port*: mặc định là 3130.

** *cache_dir*: khai báo kích thước thư mục cache cho squid, mặc định là: *cache_dir /var/spool/squid/cache 100 16 256*

Giá trị 100 tức là dùng 100MB để làm cache, nếu dung lượng đĩa cứng lớn, ta có thể tăng thêm tùy thuộc vào kích thước đĩa. Như vậy squid sẽ lưu cache trong thư mục /var/spool/squid/cache với kích thước cache là 100MB.

** *Access Control List* và *Access Control Operators*: ta có thể dùng hai chức năng trên để ngăn chặn và giới hạn việc truy xuất dựa vào destination domain, IP address của máy hoặc mạng. Mặc định squid sẽ từ chối phục vụ tất cả, vì vậy ta phải cấu hình lại tham số này. Để được vậy, ta cấu hình thêm cho thích hợp với yêu cầu bằng hai tham số là : *acl* và *http_access*.

Ví dụ: Ta chỉ cho phép mạng 172.16.1.0/24 được dùng proxy server bằng từ khoá **src** trong *acl*.

```
acl MyNetwork src 172.16.1.0/255.255.255.0
```

```
http_access allow MyNetwork
```

```
http_access deny all
```

+ Ta cũng có thể cấm các máy truy xuất đến những site không được phép bằng từ khoá **dstdomain** trong *acl*, ví dụ:

```
acl BadDomain dstdomain yahoo.com
```

```
http_access deny BadDomain
```

```
http_access deny all
```

+ Nếu danh sách cấm truy xuất đến các site dài quá, ta có thể lưu vào 1 file text, trong file đó là danh sách các địa chủ như sau:

```
acl BadDomain dstdomain "/etc/squid/danh sach cam"
```

```
http_access deny BadDomain
```

+ Theo cấu hình trên thì file /etc/squid/danh sach cam là file văn bản lưu các địa chỉ không được phép truy xuất được ghi lần lượt theo từng dòng.

+ Ta có thể có nhiều *acl*, ứng với mỗi **acl** phải có một **http_access** như sau:

```
acl MyNetwork src 172.16.1.0/255.255.255.0
```

```
acl BadDomain dstdomain yahoo.com
```

```
http_access deny BadDomain
```

```
http_access allow MyNetwork
```


http_access deny all

+ Như vậy cấu hình trên cho ta thấy proxy cấm các máy truy xuất đến site www.yahoo.com và chỉ có mạng 172.16.1.0/24 là được phép dùng proxy. “**http_access deny all**”: cấm tất cả ngoại trừ những acl đã được khai báo.

- Nếu proxy không thể kết nối trực tiếp với Internet vì không có địa chỉ IP thực hoặc proxy nằm sau một Firewall thì ta phải cho proxy query đến một proxy khác có thể dùng Internet bằng tham số sau :

**cache_peer khangves.linuxsrv.mcsevn.com
parent 8080 8082**

+ Cấu hình trên cho chúng ta thấy proxy sẽ query lên proxy “cha” là *khangves.linuxsrv.mcsevn.com* với tham số **parent** thông qua *http_port* là 8080 và *icp_port* là 8082.

- Ngoài ra trong cùng một mạng nếu có nhiều proxy server thì ta có thể cho các proxy server này query lẫn nhau như sau:

**cache_peer proxy2.linuxsrv.mcsevn.com sibling
8080 8082**

**cache_peer proxy3.linuxsrv.mcsevn.com sibling
8080 8082**

sibling dùng cho các proxy ngang hàng với nhau.

4./ Khởi động Squid:

- Sau khi đã cài đặt và cấu hình lại squid, ta phải tạo cache trước khi chạy squid bằng lệnh:

squid -z

- Nếu trong quá trình tạo cache bị lỗi, ta chú ý đến các quyền trong thư mục cache được khai báo trong tham số *cache_dir*. Có thể thư mục đó không được phép ghi. Nếu có ta phải thay đổi bằng:

**chown squid: squid /var/spool/squid
chmod 770 /var/spool/squid**

- Sau khi tạo xong thư mục cache, ta khởi động và dừng squid bằng script như sau:

**/etc/init.d/squid star
/etc/init.d/squid stop**

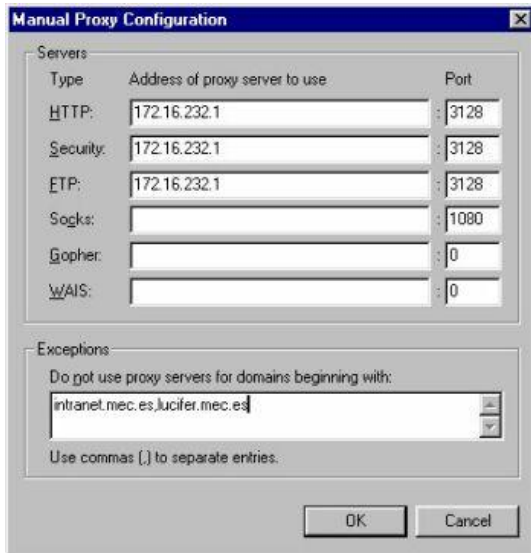
- Sau khi squid đã khởi động, muốn theo dõi và quản lý việc truy cập của các client hay những gì squid đang hoạt động cache như thế nào, ta thường xuyên xem xét những file sau đây:

*** **cache_log**: bao gồm những cảnh báo và thông tin trạng thái của cache

*** **store_log**: bao gồm những cơ sở dữ liệu về những thông tin gì mới được cập nhật trong cache và những gì đã hết hạn

*** **access_log**: chứa tất cả những thông tin về việc truy cập của client, bao gồm địa chỉ nguồn, đích đến, thời gian.....

- Về phần Server đã cài đặt xong, còn về phía client, bạn phải hiệu chỉnh lại cấu hình địa chỉ của Server và port proxy của Server, ví dụ như hình sau:



Chúc bạn thành công.

APACHE Web Server

I. Giới thiệu

A. Quá trình phát triển

- Apache web Server đi vào thế giới Server từ giữa những năm 90. Một nhà lập trình đã nhận định: “Apache như là 1 viên đá quý của chương trình mã nguồn mở, chi phí cho nó thì hầu như không có, hoạt động tốt hơn những đối thủ cạnh tranh khác, do đó nó được sử dụng ngày càng rộng rãi hơn những Web Servers thương mại khác”.
- Apache thường đi kèm với bản phân phối cùng Linux hoặc tải từ trang www.apache.org (nó đảm bảo cho bạn luôn có phiên bản mới nhất)











Index of /dist/httpd

Make sure you're downloading from a nearby mirror site!

If you're having trouble accessing these files, there's probably a closer mirror to you.
[Go here to find it.](#)

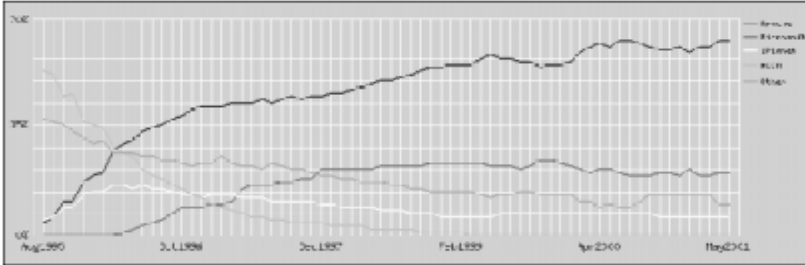
Apache 1.3.20 is now available.

Apache 2.0.16 Beta is now available for testing.

Name	Last modified	Size	Description
 Parent Directory	18-Jun-2001 17:22	-	
 Announcement.html	21-May-2001 19:11	7k	Apache 1.3 Release note
 Announcement.txt	20-May-2001 05:07	6k	Apache 1.3 Release note
 Announcement2.html	04-Apr-2001 13:05	4k	Apache 2.0 Release note
 Announcement2.txt	04-Apr-2001 13:05	4k	Apache 2.0 Release note
 Apache-1.3-docs.pdf.gz	03-Apr-2000 13:03	1.4M	Apache 1.3.12 documenta>
 CHANGES_1.3	20-May-2001 05:08	367k	List of changes in 1.3
 CHANGES_2.0	04-Apr-2001 13:05	415k	List of changes in 2.0
 KEYS	28-Mar-2001 11:57	43k	Developer PGP keys
 apache_1.3.19.tar.Z	28-Feb-2001 06:11	2.9M	1.3.19 compressed sour>
 apache_1.3.19.tar.Z.asc	28-Feb-2001 06:11	1k	PGP signature
 apache_1.3.19.tar.Z.md5	28-Feb-2001 06:11	1k	MD5 hash

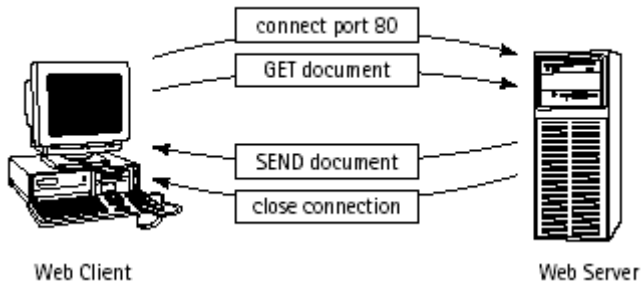
Trang www.apache.org/dist/httpd

- Apache thống trị thị trường web Server từ rất sớm. Thông tin tham khảo tại Netcraft (www.netcraft.com), Ziff-Davis (www.zdnet.com), Apache Week (www.apacheweek.com), và Apache Today (www.apachetoday.com)



B. Tiến trình giải quyết yêu cầu và đặc điểm Apache

- Web Server là sự kết hợp giữa phần cứng và phần mềm phục vụ cho những tài liệu HTTP khi client yêu cầu. Một web Server cơ bản là một máy tính với hệ điều hành Linux, một file hệ thống đầy đủ khả năng hỗ trợ tốt cho ứng dụng Web Server, và một kết nối mạng (đó là đặt trung cho Internet hoặc tổ chức intranet). Khi làm việc với Web Server cần có sự cân nhắc về các loại người dùng đảm bảo hệ thống chạy thực sự hiệu quả, như là:
 - Mục đích Web Server
 - Tiến trình request/response cho Client
- + Mục đích của Web Server có thể thay đổi. Từ đơn giản như mạng server nội bộ, đến phức tạp như e-commerce server. Nó rất quan trọng để xác định mục đích Server trước khi xây dựng và đưa vào hoạt động
- + Tiến trình request/response bắt đầu từ việc Client yêu cầu, thường là từ trình duyệt Web, và sự trả lời từ Server, trả về thông tin cho Client
- Tiến trình hoạt động Web Server



+ Client sử dụng trình duyệt Web kết nối đến Server và đưa ra yêu cầu. Yêu cầu này sử dụng giao thức HTTP mà người dùng muốn Server cung cấp, và nói cho Server biết phiên bản nào HTTP dùng để trả lời. Web Server lắng nghe những yêu cầu trên mạng. Khi một yêu cầu được gửi đến, Web Server phân tích thành 3 phần:

- Cách thức sử dụng là GET, POST, hay HEAD. Phương pháp GET yêu cầu Uniform Resource Indicator (URI - sự chỉ định tài nguyên đồng nhất) hoặc tài liệu từ Web server. Phương pháp POST gửi dữ liệu điều khiển chỉ định bởi URI. Phương pháp HEAD chỉ yêu cầu headers từ Web server.
- Tài nguyên đang được yêu cầu: Web Server đổi URI, xác định đối tượng yêu cầu thành đường dẫn vật lý trên hệ thống file của Web server
- Phiên bản HTTP

+ Web Server tiếp tục quy trình giải quyết yêu cầu bằng việc dùng child processes (tiến trình con) để hoàn thành yêu cầu, và gửi trả lời lại cho người dùng. Trong khoảng thời gian đó Web Server sẽ kiểm tra quyền hạn của Client. Trước khi hoàn tất yêu cầu, Web Server sẽ xác định loại MIME của đối tượng được yêu cầu và sắp đặt lại aliases

+ Yêu cầu Client đã được thực hiện. Trình duyệt Web sẽ cập nhập thông tin. Ví dụ một trang HTML, một file, một thông báo lỗi sẽ xuất hiện. Khi kết thúc yêu cầu Web server sẽ cập nhập lại file log và ngắt kết nối đến Client.

- Đặc điểm Web Server:

Là phần mềm mã nguồn mở và hoàn toàn miễn phí. Hỗ trợ trên những hệ điều hành khác nhau như: Linux, UNIX, Windows (95, 98, NT, and 2000), OS/2, Solaris, FreeBSD, OpenBSD, và HP/UX.

Apache là một Modular, dễ lựa chọn và có thể tích hợp với sản phẩm khác như là IBM Websphere

II. Cài đặt và cấu hình

A. Xây dựng và cài đặt Apache Web Server

- Khi bạn tải phiên bản Apache. Có 2 cách để cài : từ source code hoặc từ tập tin nhị phân (RPM). Bạn có thể sử dụng cả 2 cách để cài đặt.
- Cài đặt từ RPM
 - + Tải file RPM từ trang <http://www.rpmfind.net>
 - + Login với quyền Root và gõ lệnh:
`rpm -ivh apache-1.3.xx-y.i386.rpm`
 - + Nếu muốn nâng cấp bạn phải stop Apache và gõ lệnh
`rpm -Uvh apache-1.3.xx-y.i386.rpm`
- Cài đặt từ Source: việc cài đặt từ nguồn không dễ như cài từ RPM. Có những đòi hỏi khác nhau đối với những hệ điều hành khác nhau
 - + Tải file .tgz hay tar.gz từ trang <http://www.apache.org/dist/httpd/> vào thư mục /usr/local/src
 - + Từ thư mục /usr/local/src giải nén file apache_1.3.24.tar.gz gõ lệnh
`tar zxvf apache_1.3.24.tar.gz`
 - + Apache source đã giải nén nằm trong thư mục /usr/local/src/apache_1.3.24
 - + Tạo User và Group mặc định cho Apache
`groupadd www (tạo group www)`
`useradd -g www www`

Chú ý: Sau khi tạo user www, dùng lệnh passwd với tham số -l để khoá user www. Điều này sẽ đảm bảo tính bảo mật cao vì sau này chỉ sử dụng root để cấu hình.

 - + Sử dụng *configure* Script gõ lệnh:
`#!/configure --prefix=/usr/local/apache --server-uid=www --server-gid=www --htdocdir=/opt/web/html --cgidir=/opt/web/cgi-bin --enable-module=most --enable-shared=max`

- Tham số `--server-uid=www` chỉ định Apache server sẽ chạy với user `www`. User `www` phải được tạo trước
- Tham số `--server-gid=www` chỉ định Apache server sẽ chạy với nhóm `www`.
- Tham số `--htdocsdirectory` chỉ định Web site files mặc định sẽ đặt trong thư mục `/opt/web/html`.
- Tham số `--cgidirectory=/opt/web/cgi-bin` chỉ định thư mục mặc định cài CGI `/opt/web/cgi-bin`.

+ Gõ lệnh `make`
+ Gõ lệnh `make install`

+ Hướng dẫn cài đặt tham khảo từ

<http://www.php.net/manual/en/install.unix.php>

```
1. gunzip apache_1.3.x.tar.gz
2. tar xvf apache_1.3.x.tar
3. gunzip php-x.x.x.tar.gz
4. tar xvf php-x.x.x.tar
5. cd apache_1.3.x
6. ./configure --prefix=/www
7. cd ../php-x.x.x
8. ./configure --with-mysql --with-apache=../apache_1.3.x --
enable-track-vars
9. make
10. make install
11. cd ../apache_1.3.x
12. ./configure --activate-module=src/modules/php4/libphp4.a
13. make
14. make install
15. cd ../php-x.x.x
16. cp php.ini-dist /usr/local/lib/php.ini
17. Edit your httpd.conf or srm.conf file and add:
    AddType application/x-httpd-php .php
18. Use your normal procedure for restarting the Apache
server. (You must stop and restart the server, not just cause
the server to reload by use a HUP or USR1 signal.)
```

- Kiểm tra chạy thử Apache <http://127.0.0.1>



Figure 15-2: Apache's test page

B. Cấu hình Apache

1. Cấu hình Apache tổng quát

- Trong mục này chúng ta sẽ nghiên cứu bản hướng dẫn bằng hướng dẫn cấu hình Apache tổng quát. Giá trị của bản dưới đây là giá trị mặc định.

Tham số	Miêu tả
ServerType standalone	Điều khiển Apache chạy như là standalone process hay chạy ở inetd
ServerRoot /etc/httpd	Định nghĩa thư mục gốc Apache chứa tập tin cấu hình và tập tin log
PidFile /var/run/httpd.pid	Quy định tập tin chứa PID (Process ID) của tiến trình Master Server
Timeout 300	Thời gian tối đa tính bằng giây mà Apache chờ để gửi và nhận packet.
KeepAlive On	Cho phép nhiều requests trong cùng kết nối. Tăng tốc

MaxKeepAliveRequests 100	phân phát tài liệu HTML Đặt số lượng request cho phép cho mỗi connection
KeepAliveTimeout 15	Khoản thời gian trôi qua giữa những yêu cầu từ cùng 1 Client trên cùng kết nối khi KeepAlive ở chế độ On
MinSpareServers 5	Thời gian rảnh tối thiểu cho Child servers
MaxSpareServers 20	Thời gian rảnh tối đa cho Child servers (do master server sinh ra)
StartServers 8	Số lượng Child server được tạo khi Apache được khởi động
MaxClients 150	Số lượng kết nối cùng một lúc mà Child server hỗ trợ
MaxRequestsPerChild 100	Số lượng Requests tối đa của mỗi Child Server trước khi đạt đến giới hạn
Listen [ipaddress:]80	Xác định sự kết hợp giữa địa chỉ IP và Port mà Apache cho phép kết nối, nhiều port có thể được sử dụng
LoadModule modname filename	Đường dẫn module hoặc tập tin thư viện trên server và thêm vào danh sách modules đang hoạt động Modname
ClearModuleList	Xóa list của module đang hoạt động, nó sẽ được xây dựng lại khi dùng lệnh AddModule
AddModule module.c	Kích hoạt những built-in nhưng không active module module.c

- Khi chỉ đường dẫn tập tin log file trong cấu hình, mặc định sẽ được gán đường dẫn /etc/httpd. Ví dụ: tập tin log được khai báo /logs/mylog.log thì đường dẫn sẽ là /etc/httpd/logs/mylog.log.
- Khai báo KeepAlive On sẽ cải thiện hoạt động của Server, làm tăng sự kết nối thành công giữa Client và Server. Thông số

MinSpareServers và MaxSpareServers cho phép Apache tự điều chỉnh, thêm vào và xóa đi các tiến trình khi tài nguyên hệ thống thay đổi đột ngột. Khi có nhiều hơn số MaxClient kết nối, mỗi yêu cầu sẽ được đưa vào hàng chờ (first-in-first-out vào trước ra trước), những dịch vụ sẽ nhận theo thứ tự những kết hiện thời và đóng lại, thông số này có lợi cho những WebSite có lượng truy cập lớn.

- Đối với nhiều Sites giá trị tập tin cấu hình mặc định ở trên không cần thay đổi. Và cũng không cần thay đổi thứ tự nạp Module và kích hoạt Module bằng tham số LoadModule và AddModule cho đến khi nào bạn biết bạn đang làm gì. Một vài Module lệ thuộc vào các Module khác để hoạt động. Apache sẽ không Start lên khi Module nạp không đúng
- Hình bên dưới là tập tin cấu hình mặc định của Apache không có tham số AddModule, AddModule và ClearModuleList.

```
ServerType standalone
ServerRoot "/etc/httpd"
LockFile /var/lock/httpd.lock
PidFile /var/run/httpd.pid
ScoreBoardFile /var/run/httpd.scoreboard
Timeout 300
KeepAlive On
MaxKeepAliveRequests 100
KeepAliveTimeout 15
MinSpareServers 5
MaxSpareServers 20
StartServers 8
MaxClients 150
MaxRequestsPerChild 100
Listen 80
```

- Các thông số trên không cần thay đổi nhiều, giá trị này được đưa ra bởi Apache Group.
2. Cấu hình mặc định (không chứa Virtual hosts)
- Trước đây, nói đến Default Server hay Primary Server là nói đến Web Server trả lời tất cả yêu cầu HTTP không dùng đến Virtual Hosts hay Virtual Servers. Virtual Hosts hay Virtual Servers là

Web Server chạy trên 1 máy giống như Default Server nhưng nó là Main Server có nhiều host name hoặc IP. Cấu hình Default Server có thể dùng cấu hình Virtual Servers.

- Bảng hướng dẫn cấu hình Default Server

Tham số	Miêu tả
Port 80	Cổng dùng cho kết nối đến Server
User [#]apache	Chỉ định UID có quyền thực thi Apache
Group [#]apache	Chỉ định GID có quyền thực thi Apache
ServerAdmin root@localhost	Địa chỉ mail sẽ được gửi đến Client khi có lỗi
ServerName	Tên Server như là <u>www.mydomain.com</u> , khác tên host trên server
DocumentRoot "/var/www/html"	Đường dẫn thư mục mặc định chứa trang web
UserDir public_html	Đường dẫn thư mục con trong thư mục Home của User dùng chứa trang web
DirectoryIndex filename	Chỉ định một hay nhiều tên file Index khi mà yêu cầu không thể xác định file
AccessFileName .htaccess	Quy định quyền truy cập tập tin trong thư mục hay trong thư mục con, khi tập tin này được chỉ định bởi AccessFile
UseCanonicalName On	Apache tự tham chiếu đến URL. Nếu On sử dụng tên Server và Port, nếu off sử dụng tên Host và Port cung cấp cho Client
TypesConfig /etc/mime.types	Quy định tên tập tin theo chuẩn MIME, phần mở rộng được phép trên Server
DefaultType text/plain	Chuẩn mặc định của kiểu MIME khi có yêu cầu được gửi đến.
HostnameLookups Off	Quy định việc Apache dùng DNS lookup khi kết nối đến
ErrorLog /var/log/httpd/error_log	Xác định đường dẫn file error log
LogLevel warn	Xác định thông tin chi tiết Apache ghi vào tập tin error log
LogFormat formatstr	Định dạng kiểu formatstr, Apache ghi lại log vào Access log

CustomLog /var/log/httpd/access_log combined	Xác định tên của tập tin access log và định dạng tập tin log.
ServerSignature On	Hiển thị tên Server và phiên bản vào cuối trang khi có thông báo lỗi, liệt kê tập tin trong FTP,...
Alias urlpath dirpath	Liên kết đường dẫn thư mục liên quan đến DocumentRoot, đến thư mục tập tin hệ thống, nằm ngoài hệ thống tập tin server
ScriptAlias urlpath dirpath	Hoạt động giống Alias và cũng dùng chỉ đường dẫn chức script CGI
IndexOptions FancyIndexing	Xác định đặc điểm hoạt động thư mục Apache indexing.
AddIconByEncoding mimeencoding	Đặt biểu tượng xuất hiện bên cạnh tập tin dạng mimeencoding, sử với FancyIndexing
AddIconByType icon mimetype	Đặt biểu tượng xuất hiện bên cạnh tập tin dạng mimetype, sử với FancyIndexing
AddIcon icon name	Đặt biểu tượng bên cạnh những tập tin có phần mở rộng name
DefaultIcon /icons/unknown.gif	Đặt những biểu tượng mặc định với những tập tin MIME hoặc không xác định loại nào
AddDescription str file	Gán kiểu String cho phần miêu tả với 1 hay nhiều tập tin file, dung với FancyIndexing
AddEncoding mimeencoding name	Gán kiểu mã hoá MIME bởi mimeencoding cho tập tin có phần mở rộng name
AddType mimetype name	Thêm mimetype cho tập tin có phần mở rộng name vào danh sách MIME type

Dưới đây là bản tham khảo cấu hình Default Server

```
Port 80
User apache
Group apache
ServerAdmin root@localhost
DocumentRoot "/var/www/html"
<Directory />
Options FollowSymLinks
AllowOverride None
</Directory>
<Directory "/var/www/html">
Options Indexes Includes FollowSymLinks
AllowOverride None
Order allow,deny
Allow from all
</Directory>
UserDir public_html
DirectoryIndex index.html index.htm index.shtml index.php index.php4 index.php3
index.cgi
AccessFileName .htaccess
<Files ~ "\.ht">
Order allow,deny
Deny from all
</Files>
UseCanonicalName On
TypesConfig /etc/mime.types
DefaultType text/plain
<IfModule mod_mime_magic.c>
MIMEMagicFile conf/magic
</IfModule>
HostnameLookups Off
ErrorLog /var/log/httpd/error_log
LogLevel warn
LogFormat "%h %l %u %t \"%r\" %>s %b \"%{Referer}i\" \"%{User-Agent}i\"" combine
LogFormat "%h %l %u %t \"%r\" %>s %b" common
LogFormat "%{Referer}i -> %U" referer
LogFormat "%{User-agent}i" agent
CustomLog /var/log/httpd/access_log combined
ServerSignature On
Alias /icons/ "/var/www/icons/"
<Directory "/var/www/icons">
Options Indexes MultiViews
AllowOverride None
Order allow,deny
Allow from all
```

```
</Directory>
ScriptAlias /cgi-bin/ "/var/www/cgi-bin/"
<Directory "/var/www/cgi-bin">
AllowOverride None
Options ExecCGI
Order allow,deny
Allow from all
</Directory>
IndexOptions FancyIndexing
AddIconByEncoding (CMP,/icons/compressed.gif) x-compress x-gzip
AddIconByType (TXT,/icons/text.gif) text/*
AddIconByType (IMG,/icons/image2.gif) image/*
AddIconByType (SND,/icons/sound2.gif) audio/*
AddIconByType (VID,/icons/movie.gif) video/*
AddIcon /icons/binary.gif .bin .exe
AddIcon /icons/binhex.gif .hqx
AddIcon /icons/tar.gif .tar
AddIcon /icons/world2.gif .wrl .wrl.gz .vrml .vrm .iv
AddIcon /icons/compressed.gif .Z .z .tgz .gz .zip
AddIcon /icons/a.gif .ps .ai .eps
AddIcon /icons/layout.gif .html .shtml .htm .pdf
AddIcon /icons/text.gif .txt
AddIcon /icons/c.gif .c
AddIcon /icons/p.gif .pl .py
AddIcon /icons/f.gif .for
AddIcon /icons/dvi.gif .dvi
AddIcon /icons/uuencoded.gif .uu
AddIcon /icons/script.gif .conf .sh .shar .csh .ksh .tcl
AddIcon /icons/tex.gif .tex
AddIcon /icons/bomb.gif core
AddIcon /icons/back.gif ..
AddIcon /icons/hand.right.gif README
AddIcon /icons/folder.gif ^DIRECTORY^
AddIcon /icons/blank.gif ^BLANKICON^
DefaultIcon /icons/unknown.gif
ReadmeName README.html
HeaderName HEADER.html
IndexIgnore .??* *~ *# HEADER* README* RCS CVS *,v *,t
AddEncoding x-compress Z
AddEncoding x-gzip gz tgz
AddLanguage en .en
AddLanguage fr .fr
AddLanguage de .de
AddLanguage da .da
AddLanguage el .el
AddLanguage it .it
```

```
LanguagePriority en fr de
<IfModule mod_php4.c>
AddType application/x-httpd-php .php4 .php3 .html .php
AddType application/x-httpd-php-source .phps
</IfModule>
<IfModule mod_php3.c>
AddType application/x-httpd-php3 .php3
AddType application/x-httpd-php3-source .phps
</IfModule>
<IfModule mod_php.c>
AddType application/x-httpd-php .html
</IfModule>
AddType application/x-tar .tgz
AddType text/html .shtml
AddHandler server-parsed .shtml
AddHandler imap-file map
BrowserMatch "Mozilla/2" nokeepalive
BrowserMatch "MSIE 4.0b2;" nokeepalive downgrade-1.0 force-response-1.0
BrowserMatch "RealPlayer 4.0" force-response-1.0
BrowserMatch "Java/1.0" force-response-1.0
BrowserMatch "JDK/1.0" force-response-1.0
<IfModule mod_perl.c>
Alias /perl/ /var/www/perl/
<Location /perl>
SetHandler perl-script
PerlHandler Apache::Registry
Options +ExecCGI
</Location>
</IfModule>
Alias /doc/ /usr/share/doc/
<Location /doc>
order deny,allow
deny from all
allow from localhost .localdomain
Options Indexes FollowSymLinks
</Location>
```

- Tham số User và Group cho biết apache là chủ của Child Server. Điều này sẽ an toàn hơn vì nó không cần những quyền như root
- Tham số ServerName chỉ rõ những tên được trả về cho Client. Ví dụ: nếu tên Server DNS là webbeast.mydomain.com bạn có thể đặt tên Server là www.mydomain.com , lúc này Server sẽ trả lời yêu cầu gửi đến www.mydomain.com

- DocumentRoot /var/www/html xác định thư mục chứa trang Web của Server. Ví dụ: tên server www.mydomain.com khi Client truy cập trang www.mydomain.com/index.html , server sẽ trả về tập tin /var/www/html/index.html cho Client
- Mỗi thẻ <Directory> </Directory> cấu hình quyền thư mục hay thư mục con. Thẻ đầu tiên sẽ đặt quyền cho tất cả thư mục:

```
<Directory />  
    Options FollowSymLinks  
    AllowOverride None  
</Directory>
```

+ Thẻ này sẽ tác dụng lên /var/www/html, /var/www/icons và /var/www/cgi-bin

+ Dưới đây là những Options áp dụng lên thư mục là:

- All : chấp nhận tất cả Option trừ MultiViews. All là giá trị mặc định
- ExecCGI : cho phép thực thi CGI
- FollowSymLinks : cho phép Link symbolic trong thư mục
- Includes : cho phép SSI (server-side includes)
- IncludesNOEXEC : cho phép SSI như không cho phép lệnh #exec và #include cho CGI scripts
- Indexes : cho Server trả về danh sách list thư mục và tập tin nếu không có index.html
- MultiViews : cho phép tìm kiếm MultiViews. Nếu Server nhận được yêu cầu cho những tài nguyên không tồn tại ví dụ /doc/resource, sau đó Server sẽ scans những thư mục tên resources.* , nếu có sẽ lựa chọn phù hợp và trả về cho Client
- None : tắt hết những option cho thư mục và thư mục con
- SymLinksIfOwnerMatch : chỉ cho Server đường dẫn đại diện của những tập tin và thư mục của UID

+ Ở phần trên chỉ có một Option cho tất cả thư mục từ / (root) là FollowSymLinks. Từ đó trở về sao tất cả những sự khác biệt với Option / (root) sẽ có tác dụng trên thư mục bạn qui định

```
<Directory "/var/www/html">  
    Options Indexes Includes FollowSymLinks  
    AllowOverride None  
    Order allow.deny  
    Allow from all  
</Directory>
```


+ Mục AllowOverride nói cho Server biết có quyền truy cập những tập tin được qui định bởi AccessFileName (AccessFileName .htaccess trong trường hợp trên). Nếu chọn None Server sẽ lờ đi tập tin access file. Nếu chọn All tập tin AccessFileName .htaccess sẽ có hiệu lực

+ Order điều khiển thứ tự áp đặc quyền hạn cho những tài nguyên. Order có các giá trị:

- Order Deny.Allow : Xét quyền Deny trước Allow sau, mặc định cho phép truy cập. Client không bị Deny và được Allow thì được truy cập
- Order Allow.Deny : Xét quyền Allow trước Deny sau và mặc định là Deny. Client nếu không Allow hoặc bị Deny thì không được truy cập
- Order Mutual-failure : chỉ những Client có trong danh sách Allow và không có trong danh sách Deny thì được truy cập

- Cài đặt quyền hạn cho file và thư mục rất quan trọng cho Apache chạy ổn định và bảo mật. Root sẽ làm chủ file httpd.conf và thư mục bin. Những User (web) sẽ làm chủ thư mục log v.v..

3. Cấu hình Virtual Hosts

- Bản hướng dẫn sau đây dùng cho cấu hình Virtual Server

Tham số	Miêu tả
<Virtual Host ipaddr[:port]> directives </VirtualHost>	Xác định địa chỉ IP của Virtual host. Directives là những tham số cấu hình default server
NameVirtualHost ipaddr[:port]	Địa chỉ IP của Virtual Host
ServerName fqdn	Tên đầy đủ của Server VirtualHost
ServerAlias altname	Cho phép Virtual Host trả lời những host names khác.

- Cấu hình chuẩn Virtual Server :

```
...  
Port 80  
ServerName webbeast.domain.com  
NameVirtualHost 192.168.0.1  
<VirtualHost 192.168.0.1>
```

```
DocumentRoot /var/www/thisdomain
ServerName www.domain.com
</VirtualHost>
<VirtualHost 192.168.0.1>
DocumentRoot /var/www/thatdomain
ServerName www.that.domain.com
</VirtualHost>
```

- Trong ví dụ trên www.domain.com và www.that.domain.com là những aliases (CNAME records) cho địa chỉ 192.168.0.1, có nghĩa 2 tên miền trên và webbeast.domain.com đều trỏ đến 192.168.0.1. NameVirtualHost được qui định là địa chỉ 192.168.0.1, tên ServerName là tên máy tính chạy Server webbeast.domain.com. Yêu cầu gửi đến www.that.doamain.com sẽ được đáp ứng từ /var/www/thatdomain, nhưng yêu cầu gửi đến www.domain.com sẽ được đáp ứng từ /var/www/thisdomain.

III. Cài đặt cấu hình kết hợp bảo mật Apache hỗ trợ PHP

A. Chiến lược bảo mật cơ bản khi sử dụng Apache

- Việc bảo mật Apache gắn liền với việc bảo mật PHP, một ngôn ngữ lập trình sử dụng để tạo ra những trang web động tương tác với người dùng và những dữ liệu của người dùng được lưu trong cơ sở dữ liệu tại local. Việc bảo mật sẽ mang lại :
 - PHP được cấu hình tối ưu trong một cơ chế bảo mật tốt
 - Đoạn mã PHP sẽ thực thi trong môi trường chrooted
 - Apache Server sẽ từ chối những yêu cầu (Get và Post) chứa những thẻ HTML (có thể bị tấn công bằng phương pháp Cross-Site-Scripting) hoặc những kí tự (') hay (") (chống lại sự tấn công bằng phương pháp SQL Injection)
 - Không có lời cảnh báo PHP hoặc những thông báo lỗi

B. Cài đặt cấu hình Apache

- Cần đảm bảo MYSQL đã được cài trên Server và được đặt trong thư mục "/usr/local/mysql" sẽ tích hợp SQL với PHP. Trước tiên chúng ta cần tải về phiên bản mã nguồn mới nhất của Apache, PHP, và những modules mod_security (www.modsecurity.org) những module này được dùng bảo vệ CSS và tấn công bằng SQL injection. Tiến hành giải nén chương

trình vào thư mục HOME, và mod_security được chép vào apache_1.3.27/src/modules/extra/

```
gzip -dc apache_1.3.27.tar.gz | tar xvf -
gzip -dc php-4.3.2.tar.gz | tar xvf -
gzip -dc mod_security_1.5.tar.gz | tar xvf -
cp mod_security_1.5/apache1/mod_security.c
apache_1.3.27/src/modules/extra/
```

- Trước khi biên dịch chương trình chúng ta cần quyết định 3 phương pháp mà PHP sẽ được cài như :
 - Một Web Server với module tĩnh
 - Một Web Server với module động
 - Thể hiện như CGI

Các cách trên có ưu điểm và nhược điểm riêng. Biên dịch PHP như một module tĩnh sẽ cải thiện hoạt động Web Server nhưng khi nâng cấp phiên bản mới PHP thì cần biên dịch lại hoàn toàn. Lựa chọn thứ 2 biên dịch như module động khi nâng cấp không cần biên dịch lại nhưng hoạt động web server sẽ giảm 5%. Phương pháp thứ 3 cài PHP như là CGI đó là sự liên kết cơ chế suEXEC của Apache, nó hoàn toàn là một giải pháp tốt, nhưng nếu không cài đặt đúng sẽ trở thành lỗi bảo mật nghiêm trọng. Lựa chọn tốt nhất cho việc bảo mật và tối ưu là dùng phương pháp thứ nhất. Phần hướng dẫn sau đây làm theo phương pháp thứ nhất

- Quá trình cài đặt Apache có PHP giống với cài quá trình cài Apache ở phần trên nhưng có thêm 2 modules là mod_php và mod_security. Cần tạo User và Group "apache" trước khi biên dịch Apache. Làm theo các bước sau:

```
cd apache_1.3.27
./configure
+ Biên dịch module PHP
cd ../php-4.3.2
./configure --with-mysql=/usr/local/mysql --with-
apache=../apache_1.3.27 --enable-safe-mode
make
su
make install
```

```
+ Chuyển vào thư mục Apache và tiếp tục cài đặt
cd ../apache_1.3.27
./configure --prefix=/usr/local/apache --disable-module=all
--server-uid=apache --server-gid=apache --enable-
module=access --enable-module=log_config --enable-
module=dir --enable-module=mime --enable-module=auth
--activate-module=src/modules/extra/mod_security --
enable-module=security --activate-
module=src/modules/php4/libphp4.a
make
su
make install
chown -R root:sys /usr/local/apache
```

“./configure” ở trên chỉ những module cần thiết cho bảo mật được cài

+ Bước tiếp theo chuyển vào thư mục PHP và chép tập tin cấu hình PHP mặc định

```
cd ../php-4.3.2
mkdir /usr/local/lib
chmod 755 /usr/local/lib
cp php.ini-recommended /usr/local/lib/php.ini
chown root:sys /usr/local/lib/php.ini
chmod 644 /usr/local/lib/php.ini
```

+ Cấu hình /usr/local/apache/conf/httpd.conf điều khiển PHP

Script

```
AddType application/x-httpd-php .php
```

+ Lúc này chúng ta có thể tiến hành chạy thử và kiểm tra PHP có thể giao tiếp với MySQL chưa, Tạo tập tin “test.php” trong thư mục /var/www/html

```
<html><body>
<?php
$link = mysql_connect("localhost", "user_name",
"password")
or die;
print "Everything works OK!";
mysql_close($link);
```

```
?>
</body></html>
```

Nếu không chạy cần kiểm tra lại quá trình cài đặt Apache và MySQL

C. Bảo mật Apache

1. Chrooting Server

- Bước đầu tiên của bảo mật là tạo môi trường chrooted cho Apache với module PHP. Để tạo môi trường “Chrooting Server” xem tài liệu hướng dẫn đi kèm. Phần thêm vào sau đây dùng cho PHP. Trước khi chạy Apache lần đầu trong môi trường chrooted cần chép thêm vào những thư viện sau:

```
cp /usr/local/mysql/lib/mysql/libmysqlclient.so.12
/chroot/httpd/usr/lib/
cp /usr/lib/libm.so.2 /chroot/httpd/usr/lib/
cp /usr/lib/libz.so.2 /chroot/httpd/usr/lib/
```

- Chép tập tin cấu hình PHP mặc định

```
umask 022
mkdir -p /chroot/httpd/usr/local/lib
cp /usr/local/lib/php.ini /chroot/httpd/usr/local/lib/
```

- Tạo thư mục “/chroot/httpd/tmp” .Thư mục này phải của Root và được chmod 1777. Sau khi tạo môi trường mới chúng ta có thể kiểm tra nếu Apache hoạt động tốt:

```
chroot /chroot/httpd /usr/local/apache/bin/httpd
```

- Trước khi cấu hình PHP cần phải cẩn thận kiểm tra lại sự giao tiếp giữa PHP và MySQL. Bởi vì sự giao tiếp giữa PHP và MySQL trên local bằng việc sử dụng socket “/tmp/mysql.sock”. Sau khi dùng PHP trong môi trường chrooted cần tạo hard link đến môi trường chrooted

```
ln /tmp/mysql.sock /chroot/httpd/tmp/
```

2. Cấu hình PHP

- Để Apache tích hợp PHP cần thêm vào tập tin httpd.conf

```
AddModule mod_php4.c
AddType application/x-httpd-php .php
AddType application/x-httpd-php .inc
AddType application/x-httpd-php .class
```

Có thể thêm phần mở rộng khác như html, dhtml tùy thuộc vào Server

- Những thay đổi quan trọng cải thiện bảo mật của PHP cho tập tin “/chroot/httpd/usr/local/lib/php.ini”

Tham số	Miêu tả
Safe_mode= on	PHP Script chỉ có thể truy cập những tập tin khi mà Owner những tập tin này là Owner của PHP Script. Đó là kĩ thuật bảo mật quan trọng cho PHP. Ngăn chặn hiệu quả việc truy cập file hệ thống (/etc/passwd)
Safe_mode_gid=off	Khi safe_mode on và safe_mode_gid off, PHP script có thể truy cập tập tin cùng Owner và cùng group
Open_basedir = directory[:...]	Khi open_basedir được thiết lập PHP chỉ có thể truy cập những tập tin trong thư mục được chỉ định (và thư mục con)
safe_mode_exec_dir = directory[:...]	Khi safe_mode on, hàm system(), exec() và những hoạt động thực thi chương trình sẽ bị từ chối ,nếu không đặt vào thư mục chỉ định
expose_php = Off	Tắt “expose_php” PHP sẽ từ chối cho HTTP Headers gửi đến Client khi trả lời yêu cầu
register_globals = Off	Khi register_global on tất cả các biến EGPCS (Environment, Get, Post, và Server) tự động đăng kí như biến global, nó có thể tạo ra lỗi bảo mật nghiêm trọng. Nên tắt register_global (Mặc định từ phiên bản 4.2.0 tham số này off)
display_errors = Off	Nếu display_errors tắt, PHP error và lời cảnh báo sẽ không xuất hiện. Bởi vì lời

	cảnh báo này thường để lộ những thông tin đường dẫn, câu truy vấn SQL v.v..
log_errors = On	Khi log_errors được bật, tất cả lỗi và cảnh báo được ghi nhận vào tập tin được khai báo trong error_log. Nếu không khai báo error_log những thông tin này sẽ được Apache server ghi nhận lại
Error_log = filename	Chỉ định tên tập tin được dùng ghi lại cảnh báo và lỗi (User và group Apache có quyền ghi)

3. Chống lại cách tấn công CSS và SQL Injection

- Bước cuối cùng cho việc bảo mật là thực hiện việc logging GET và POST, đồng thời thực hiện việc bảo vệ CSS và SQL injection. Chúng ta sẽ sử dụng module mod_security. Thêm vào tập tin httpd.conf

```
AddModule mod_security.c
```

- Bật chế độ logging GET và POST, thêm vào tập tin httpd.conf

```
<IfModule mod_security.c>
    AddHandler application/x-httpd-php .php

    SecAuditEngine On
    SecAuditLog logs/audit_log
    SecFilterScanPOST On
    SecFilterEngine On
</IfModule>
```

- Những lệnh trên sẽ bật chức năng Audit Engine có nhiệm vụ logging lại những yêu cầu, và bộ lọc POST Engine log lại yêu cầu POST. Để bảo vệ ứng dụng Web chống lại CSS cần thêm vào trước "</IfModule>"

```
SecFilterDefaultAction "deny,log,status:500"
SecFilter "<(.|\n)+>"
```

- Dòng đầu tiên Server sẽ trả về thông báo "Internal Server Error" khi một yêu cầu tìm kiếm cụm từ trong biến SecFilter được gửi

đến. Dòng thứ 2 thiết lập cho bộ lọc tìm những thẻ HTML trong yêu cầu GET và POST

- Một trong những kí hiệu điển hình của việc tấn công SQL Injection là dấu (') hoặc (") trong yêu cầu GET hay POST. Bằng việc từ chối những yêu cầu chứa những kí tự trên chúng ta sẽ làm cho việc tấn công SQL Injection trở nên khó hơn, thêm vào tập tin httpd.conf

```
SecFilter ""  
SecFilter "\\"
```

- Việc lọc những kí tự <, >, ' , " giúp chống lại việc tấn công CSS và SQL Injection, nhưng những ứng dụng PHP sẽ hoạt động không tốt, vì người dùng không thể sử dụng những kí tự này trong những forms HTML. Để giải quyết vấn đề này ngôn ngữ JavaScript có thể được dùng cho phía Client, nó có thể thay thế những kí tự trên với những thẻ như < > " ...

IV. Những liên kết tham khảo

- Apache HTTP Server Project: <http://httpd.apache.org/>
- Sample httpd.conf: www.securityfocus.com/data/tools/httpd.conf
- Sample apache.sh: www.securityfocus.com/data/tools/apache.sh
- Securing Apache: Step-by-Step: www.securityfocus.com/infocus/1694
- Sample httpd.conf with PHP support: www.securityfocus.com/unix/linux/images/maj_httpd.conf
- PHP: www.php.net
- mod_security: www.modsecurity.org

DỊCH VỤ THƯ ĐIỆN TỬ (Sendmail)



1. Khái niệm:

Thư điện tử, Electronic mail, Email, là dịch vụ có thể nói là quan trọng nhất đối với người sử dụng Internet. Do tính phổ cập của email, việc cấu hình tốt Mail server, tạo điều kiện cho người sử dụng có thể trao đổi Email là công việc đầu tiên và quan trọng nhất của người quản trị. Một cấu hình sai email có thể dẫn đến tình trạng không gửi hoặc nhận được thư, hoặc tệ hơn là mất thư mà không có phản hồi. Hoạt động của dịch vụ mail gắn rất chặt chẽ với cấu hình của DNS.

Chúng ta thử hình dung quá trình gửi mail để hiểu về cơ chế hoạt động của hệ thống Email.

Đầu tiên, bạn phải có một chương trình cho phép bạn soạn thảo mail. Có rất nhiều chương trình thực hiện nhiệm vụ này : Internet Explorer, Eudora, Netscape cho Windows; eml, netscape, mail cho Unix ... Các chương trình đầu tiên cho phép bạn đánh địa chỉ Email của người nhận. Địa chỉ đó ngày nay có dạng receptient name@domain name.top domain , ví dụ như vqthang@mcsevietnam.com. Sau đó bạn soạn thảo nội dung thư và gửi đi bằng một lệnh hay một nhấp chuột. Khi đó, chương trình mail client sẽ theo cấu hình mà bạn đã làm, tìm một SMTP server, outgoing server. SMTP là viết tắt của Simple Mail Transfer Protocol và server sử dụng giao thức SMTP được gọi là SMTP server. Người ta còn thường quen dùng là mail server. Khi bạn khai báo SMTP server bạn thường dùng tên và như vậy bạn phải sử dụng DNS server mà máy bạn phải khai báo từ trước để nhờ phân giải và tìm địa chỉ IP tương ứng. Sau khi tìm ra địa chỉ IP của SMTP server, chương trình mail của bạn sẽ thực hiện một kết nối TCP/IP với SMTP server vào cổng 25, là cổng quy định cho SMTP server. Hai tiến trình mail client và mail server sẽ trao đổi thông tin với nhau thông qua SMTP protocol. Nếu mọi việc thông suốt, email của bạn sẽ được chấp nhận lưu trữ trên SMTP server và chương trình mail client của bạn kết thúc phiên làm việc.

Công việc tiếp theo là SMTP server của bạn tìm cách gửi mail của bạn tới người nhận. Để làm việc này, SMTP server của bạn thực hiện 2 thao tác :

- + Tìm mail server của người nhận của email của bạn
- + Gửi email của bạn đến mail server của người nhận trong email của bạn.

Thao tác đầu tiên hoàn toàn dựa vào DNS servers. Cụ thể là SMTP của bạn sẽ đóng vai trò một DNS client để hỏi DNS server của miền của bạn xem “ai là mail server của miền *mcsevietnam.com* ?” Quá trình tra hỏi này đưa đến việc tìm ra một record có dạng *mcsevietnam.com. IN MX 10 mailserver.mcsevietnam.com.* nằm trong CSDL của một DNS server nào đó, thường là DNS server của miền *mcsevietnam.com*.

Nếu quá trình này không thành công, thư của bạn sẽ không gửi đi được và bạn sẽ nhận được một thông báo trả lời rằng email của bạn không được vì “host unknown”. Nếu ngược lại, SMTP của bạn sẽ mở một kết nối TCP/IP đến *mailserver.mcsevietnam.com* vào cổng 25 để gửi email của bạn. Lúc này SMTP của bạn đóng vai trò một mail client. Giao thức SMTP lại được sử dụng để chuyển thư trong khâu này.

Nếu mọi thứ thành công, email của bạn sẽ được lưu trữ trên *mailserver.mcsevietnam.com* và người nhận *vqthang* sẽ phải kết nối với *mailserver.mcsevietnam.com* để lấy thư về máy của mình và đọc thư.

Trên đây là miêu tả một quá trình gửi mail điển hình trên Internet. Trên thực tế, quá trình này có thể phức tạp và thay đổi khá nhiều phụ thuộc vào cấu hình của từng mạng. Đó chính là yếu tố làm phức tạp hóa rất nhiều hệ thống Email và khó khăn đối với công tác quản trị dịch vụ Email. Các bạn cũng nhận thấy có ít nhất 4 máy tính tham gia vào quá trình chuyển mail, nhiều lần DNS server tham gia vào và nếu hệ thống DNS server không chạy hoàn hảo, chúng ta không thể gửi Email được.

2. Dịch vụ mail server trên Linux (Sendmail):

Có nhiều chương trình SMTP server, nhưng Sendmail có lẽ là chương trình SMTP server nổi tiếng nhất trên Unix từ lâu nay bởi tính năng mạnh và cũng bởi tính phức tạp của nó. Chương trình Sendmail được viết bởi Eric Allman khi ông là một sinh viên của University of California at Berkeley vào năm 1979. RedHat Linux có hai chương trình mail server là *smail* và *sendmail*. Nhìn chung *smail* thích hợp cho một mạng đơn giản, còn *sendmail* thì có thể dùng cho

cả hai. Trong khuôn khổ bài viết này, chúng ta sẽ nghiên cứu chương trình sendmail.

Chương trình sendmail có thể được gọi lên bộ nhớ bởi hai cách. Cách thứ nhất là sendmail được gọi lên bởi chương trình mail client, ví dụ như chương trình cùng tên mail. Khi đó sendmail sẽ mở một kết nối để gửi mail đi. Đây là cấu hình sendmail nếu máy của bạn không phải là SMTP server. Cách thứ hai là sendmail được hoạt động theo kiểu daemon, tức là thường trú trên bộ nhớ. Khi đó, daemon sendmail “nghe” sau cổng 25 các kết nối đến. Mỗi khi có kết nối đến cổng 25, sendmail daemon sinh ra một tiến trình sendmail con để tiếp nhận kết nối này, còn bản thân mình thì tiếp tục chờ đợi các kết nối khác. Với lệnh *netstat -n* ta có thể hiển thị các kết nối đang trong thực hiện. Sendmail sử dụng các tập tin cấu hình và thư mục như sau:

- Đầu tiên, Sendmail sử dụng tập tin cấu hình */etc/sendmail.cf* mỗi khi được gọi lên bộ nhớ. Tập tin này rất thích hợp cho các công tác của sendmail nhưng cực kỳ khó hiểu đối với người đọc. Ví dụ như đoạn sau đây của *sendmail.cf*:

```
R$- $@ $1 @$ {HUB} user -> user@hub
```

```
R$- @$w $@ $1 @$ {HUB} user@local -> user@hub
```

- Đây là một nhược điểm đồng thời là một ưu điểm của sendmail vì nó cho phép cấu hình sendmail cực kỳ uyển chuyển và thỏa mãn các yêu cầu dù éo le nhất của một mail server. Nếu bạn chưa một lần phải “vỡ đầu” bởi những ký tự ả rập này thì bạn chưa phải là quản trị viên thực thụ.

- Trong tập tin *sendmail.cf* có một số trường quan trọng là :

```
# Alias for this host
```

```
Cwkhanges.mcsevietnam.com. vqthang.itvn.com.
```

```
Cwlocalhost linuxsrv.mcsevietnam.com.
```

- Dòng thứ 2 xác định rằng tất cả các email với địa chỉ user@khangves.mcsevietnam.com, user@vqthang.itvn.com là thuộc về máy mà chương trình sendmail đang chạy, cần phải đưa về cho chương trình chuyển mail trên máy local và phải thử xem *user* là có tồn tại trên máy này không. Tất cả những mail với phần domain ngoài **Cw** đều được coi là cho miền ngoài và phải chuyển đi qua mạng bằng sendmail.

```
# Smart host
```

```
Dssrv.mcsevietnam.com
```

```
# Use this mailer to reach the Smart host
```

```
DNsmtp
```

- Dòng thứ 2 của ví dụ trên chỉ ra rằng với tất cả các mail không local, chỉ cần chuyển đến trạm mail trung chuyển (mail relay) và tên của mail relay là chuỗi ký tự nằm sau DS.

- Để thử xem sendmail có phân giải địa chỉ và chuyển thư đúng theo ý định của mình hay không, bạn có thể dùng lệnh **sendmail -bt** hoặc **mail -v địa_chỉ** [root@linuxsrv root]\$ /usr/sbin/sendmail -bt ADDRESS TEST MODE (ruleset 3 NOT automatically invoked)
Enter <ruleset> <address>

```
> 3,0 a@khangves.mcsevietnam.com
rewrite: ruleset 3 input: a @khangves.mcsevietnam.com
rewrite: ruleset 96 input: a < @khangves.mcsevietnam.com >
rewrite: ruleset 96 returns: a < @khangves.mcsevietnam.com. >
rewrite: ruleset 3 returns: a < @khangves.mcsevietnam.com. >
rewrite: ruleset 0 input: a < @khangves.mcsevietnam.com. >
rewrite: ruleset 196 input: a < @khangves.mcsevietnam.com. >
rewrite: ruleset 196 returns: a < @khangves.mcsevietnam.com. >
rewrite: ruleset 98 input: a < @khangves.mcsevietnam.com. >
rewrite: ruleset 98 returns: a < @khangves.mcsevietnam.com. >
rewrite: ruleset 195 input: a < @khangves.mcsevietnam.com. >
rewrite: ruleset 195 returns: $# local $: a
rewrite: ruleset 0 returns: $# local $: a
>
> 3,0 a@yahoo.com
rewrite: ruleset 3 input: a @yahoo . com
rewrite: ruleset 96 input: a < @yahoo . com >
rewrite: ruleset 96 returns: a < @yahoo . com . >
rewrite: ruleset 3 returns: a < @yahoo . com . >
rewrite: ruleset 0 input: a < @yahoo . com . >
rewrite: ruleset 196 input: a < @yahoo . com . >
rewrite: ruleset 196 returns: a < @yahoo . com . >
rewrite: ruleset 98 input: a < @yahoo . com . >
rewrite: ruleset 98 returns: a < @yahoo . com . >
rewrite: ruleset 195 input: a < @yahoo . com . >
rewrite: ruleset 90 input: < yahoo . com > a < @yahoo . com . >
rewrite: ruleset 90 input: yahoo . < com > a < @yahoo . com . >
rewrite: ruleset 90 returns: a < @yahoo . com . >
rewrite: ruleset 90 returns: a < @yahoo . com . >
rewrite: ruleset 95 input: < srv.mcsevietnam.com > a < @yahoo .
com. >
rewrite: ruleset 95 returns: $# smtp $@ srv.mcsevietnam.com $: a <
@
```

yahoo . com . >

rewrite: ruleset 195 returns: \$# smtp \$@ srv.mcsevietnam.com \$: a <
@

yahoo . com . >

rewrite: ruleset 0 returns: \$# smtp \$@ srv.mcsevietnam.com \$: a <
@

yahoo . com . >

3. Cài đặt và cấu hình Sendmail:

+ Cài đặt :

- Để cài đặt Sendmail từ package rpm, ta dùng lệnh :
rpm -ivh sendmail-x.xx.x-xx.rpm

+ Cấu hình :

- Để cấu hình Sendmail, ta cấu hình trong file
/etc/mail/sendmail.cf. Trong file này, ta chỉ cấu hình một số các tham số thường dùng như sau :

Các tham số cần điều chỉnh	Giải thích
Cwlocalhost mcsevietnam.com	Cấu hình cho sendmail nhận mail cho miền linuxsrv.mcsevietnam.com.
# "Smart" relay host (may be null) Dslinuxsrv.mcsevietnam.com	Các mail sẽ được chuyển lên máy linuxsrv.mcsevietnam.com để gửi đi.
# maximum number of recipients per SMTP envelope O MaxRecipientsPerMessage=50	Giới hạn số người nhận trong một lá mail.
# maximum message size O MaxMessageSize=3000000	Giới hạn kích thước tối đa của một lá mail, tính theo đơn vị bytes

- Ngoài ra phải cấu hình cho sendmail nhận relay mail cho miền đã khai báo(linuxsrv.mcsevietnam.com) trong file */etc/mail/sendmail.cf* ta thêm tên miền đó (linuxsrv.mcsevietnam.com) vào trong file */etc/mail/access* như sau :

Linuxsrv.mcsevietnam.com RELAY

- Dòng khai báo này cho phép các client gửi được mail thông qua mail server sendmail này, mặt khác mail server này có thể nhận mail cho miền linuxsrv.mcsevietnam.com gọi là cơ chế chống relay :

ngoài miền này, sendmail không nhận chuyển mail cho bất cứ miền nào.

- Sau khi thêm dòng này vào, ta phải chuyển file dạng text sang dạng chuẩn của sendmail có thể đọc được bằng lệnh sau :

```
Cd /etc/mail/  
Makemap hash access<access
```

- Khi đã thực hiện xong các bước trên, ta có thể khởi động sendmail bằng lệnh :

```
/etc/init.d/sendmail stop  
/etc/init.d/sendmail start
```

mcsevietnam

PPP SERVER



Cài đặt PPP Server trên Linux (sử dụng kernel 2.2.x hay 2.4.x).
PPP server tương tự RAS server bên windows. Cho phép client truy cập từ xa vào server thông qua modem. Server sử dụng modem External gắn vào COM1 hoặc COM2

1. Đảm bảo các package sau đây đã được cài đặt:

```
mgetty-1.1.28-9.i386.rpm  
mgetty-sendfax-1.1.28-9.i386.rpm (netpbm)  
mgetty-viewfax-1.1.28-9.i386.rpm  
mgetty-voice-1.1.28-9.i386.rpm  
mingetty-1.00-3.i386.rpm  
ppp-2.4.1-7.i386.rpm
```

2. Mở tập tin /etc/inittab tìm đến

```
# Run gettys in standard runlevels  
1:2345:respawn:/sbin/mingetty tty1  
2:2345:respawn:/sbin/mingetty tty2  
3:2345:respawn:/sbin/mingetty tty3  
4:2345:respawn:/sbin/mingetty tty4  
5:2345:respawn:/sbin/mingetty tty5  
6:2345:respawn:/sbin/mingetty tty6
```

Thêm vào :

```
7:2345:respawn:/sbin/mgetty ttyS0 -n 1 ( Nếu  
modem gắn vào COM1)  
8:2345:respawn:/sbin/mgetty ttyS1 -n 1 ( Nếu modem  
gắn vào COM2)
```

số lần Ring được xác định bằng tham số -n 1 (ring 1 lần)

3. Kích hoạt lại inittab
kill -HUP 1

Sau khi kích hoạt lại inittab đèn tín hiệu AA và TR (AA: Auto Answer, TR: Terminal Ready) bật sáng. Lúc này nếu quay số vào PPP Server sẽ thấy tín hiệu trả lời

4. Mở tập tin /etc/mgetty+sendfax/login.config thêm vào:

```
/AutoPPP/ - a_ppp /usr/sbin/pppd auth -chap +pap login  
debug
```

5. Mở tập tin /etc/ppp/pap-secrets và thêm vào "*" cho server, "*" cho secret và "*" cho địa chỉ IP, bạn cũng có thể chỉ định tên, password và IP cho việc chứng thực PAP
6. Mở tập tin **/etc/ppp/options** thêm vào những options:

```
lock  
-detach  
modem  
crtstcs  
proxyarp  
asynmap 0
```

- **lock**: Tạo tập tin lock giành riêng quyền truy xuất những thiết bị đặt biệt
- **-detach**: nói cho pppd không phân mảnh thành những tiến trình nền khác, cho đến khi thiết bị serial được chỉ định
- **modem**: người dùng sẽ phải đợi tín hiệu từ modem để có thể được xác nhận khi thiết bị serial được mở, nếu không được chỉ định trước
- **crtstcs**: sử dụng phần cứng điều khiển flow
- **proxyarp**: chỉ định cho client xuất hiện trên mạng Lan ngang hàng
- **asynmap 0**: thiết lập pppd không cài và sử dụng escape control sequences

7. Tạo tập tin /etc/ppp/options.ttyname chỉ định IP cho client và server trên mỗi cổng tty

```
192.168.0.1:192.168.0.100  
#serverIP:clientIP
```

Dùng windows tạo kết nối vào PPP Server với Username và Password của user trên hệ điều hành Linux