124

HELP COMMANDS

1. whatis <command> =One line information about command

whatis is a database stored under /var/cache/man/whatis

rm -rf /var/cache/man/whatis ((Delete whatis file))

makewhatis ((Updates whatis db manually)

2. command --help =one page information, explaining options of command

3. man <command> =Manual pages of command

/text =search text

n =next search

N =previous search

q =quit

Sections of man page --> Each command is grouped into diff sections

1. User command

2. System calls

3. Library calls

4. File format

5. Special file

6. Games

7. Missleanous

8. Administrative commands

p. Programmers commands

man page is zipped file stored under /usr/share/man/manx -->where x is num

man -a <command> =Shows all manual page other than default

man -k <keyword> =Shows commands related to keyword

man -w <command> =Path of command

4. info <page> =Detailed info about command,designed like webpage each

*** represents a url**

tab =Takes you to * //Give enter to expand the * and u for undo

s =Search text

q =undo

/ This is the root directory. This is where the whole tree starts.

/bin This directory contains executable programs.

/boot Contains static files for the boot loader. This directory only holds the files which are needed during the boot process.

/dev Special or device files, which refer to physical devices.

/etc Contains configuration files

/home users home directories are stored here.

/lib This directory should hold those shared libraries like drivers

/mnt and /media This directory contains mount points for temporarily mounted filesystems

/opt This directory contain third party tools.

/proc This is a mount point for the proc filesystem, which provides information about running processes and the kernel.

/root This directory is usually the home directory for the root user (optional).

/sbin Like /bin, this directory holds commands needed to boot the system, but which are usually not executed by normal users.

/tmp This directory contains temporary files which may be deleted with no notice, such as by a regular job or at system boot up.

/usr This directory is usually mounted from a separate partition. contains documents .

/var This directory contains files which may change in size, such as spool and log files.

FEATURES OF FILE SYSTEM.

1.Linux file system is inverted root tree like structure

2.Linux file system is casesensitive.

3.Hidden files start with . (dot) extension.

4.dot (.) refers to current dir

5.double dot (.) refers to immediate parent dir

Commands

date =Shows current date and time

date <mmddHHMMYYYY> =Set current date and time

date +%a =Shows week

date +b =Shows month

date +Z =Shows timezone

date +T =Shows time only

cal =Show current cal

cal -3 =Show current next and previous cal

cal <YYYY> =Show particular years calendar

CREATE FILE

cat > <file> =Create file and ctrl c save file

cat >> <file> =Append exisiting file

cat -n <file> =Number file

rm <file> =Remove file by prompting

rm -f <file> =Remove file without prompt

touch <file> =Creates empty file

touch file file2 file3 file4 =Create 4 empty files

CREATE DIR

mkdir <Dir> =Create dir

mkdir -p <Dir1/dir2/dir3> =Create parent dir

rmdir <dir> =remove empty dir

rm -rf <dir> =remove dir which has contents

LIST

ls =List contents of dir

ls -l =Long listing

ls -a =List hidden file

ls -l <file>=Long list individual file

ls -ld <dir>=Long list individual dir

ls -i <file>=Shows inode num of file

ls -h <file>=Shows human readable size of file

CHANGE DIR

cd <dir> =Change to specified dir

cd =Change to home dir

cd .. =Change to immediate parent dir

cp <file1> <file2> =Copy file to file

cp <file1> <Dir> =Copy file to dir

cp -r <dir1> <Dir2> =Copy dir1 to dir2

mv <file1> <file2> =Move/rename file, once moved source file dont exist.

USER ADMINISTRATION

useradd <user> =Add a user

passwd <user> =Add password to a user

When user is added it updates 4 files automatically.

1. /etc/passwd

2. /etc/shadow

3. /etc/group

4. /etc/gshadow

	uid	gid
root	0	0
systemuser	1-499	1-499
normaluser	500 onwards	500 onwards

uid is unique num given to user, gid is unique num given to group.

Whnever u add a user, automatically group is created in his name.

FIELDS OF ls -l

1.Permission

2. Link count //link count for file is 1 link count for dir is 2

link count of dir increases as u add dir inside it.

3.owner of file

4.group-owner of file

5.size of file

6.time of creation

7.file name

	READ	WRITE	EXECUTE
file	cat	cat > file	sh file
dir	ls	mkdir,cat,touch	cd

CHANGE PERMISSION

u=owner r=read + -->Add

g=group w=write - -->Del

o=others x=execute = -->Assign

Default permission for file/dir when root creates

FILE -|rw-|r--|r-- //where - is file , d is dir

DIR d|rwx|r-x|r-x

Default permission for file/dir when normal user creates

FILE -|rw-|rw-|r--

DIR d|rwx|rwx|r-x

1.Symbolic method.

chmod u+x,g+x,o+w <file> -->add permission

chmod u-x,g-x,o-w <file> -->take back permission

chmod ugo=rw <file> --> assign permission

2. Numeric Method

1=Execute

2=write

3=write + execute

4=read

5=read + execute

6=read + write

7=full

chmod 777 <file/dir>

chmod 444 <file/dir>

chmod 664 <file/dir>

BASH SHELL

/bin/sh =Bourne shell

/bin/bash =Bourne again shell

/sbin/nologin =Nologin shell,usually given for system users

/bin/tcsh =Turbo c shell

/bin/csh =c shell

/bin/ksh =korn shell

<shell> and enter -->takes you to that particular shell

ctrl d -->Logs out from shell

echo $0 =Shows current shell name

echo $SHLVL =Current shell level

1. /bin/bash =Default shell of linux

FEATURES OF BASH SHELL

1. command line completion

2. command line expansion

3. command line history

4. command line shortcuts

5. File globbing and wild card character

1. command line completion

Completes filename and command using tab.

2. command line expansion

a> Arithmetic expansion //echo <text> =Prints text

a=43 //echo $a =Prints value of a variable

b=534

echo $[$a + $b]

b> Quote expansion =Executes command

echo `date` //backquotes below escape key

echo `ls -l`

c> Brace expansion

mkdir redhat{1,2,3,4,5}

rmdir redhat{1,2,3,4,5}

3. Command line history

history =Shows history of commands

!<char> =Executes command starting with character

!<num> =Executes command starting with number

4. Command line shortcuts

ctrl a =Beginning of line

ctrl e =End of line

ctrl c =Gets back command prompt

ctrl d =Log out

ctrl l =Clear screen

ctrl z =Stop process //yes <text> =Print text infinite times

ctrl u =cut/copy line

ctrl y =Paste whatis cut/copied

ctrl w =Delete text after cursor

ctrl k =Delete text before cursor

5. File globbing and wild card character

ls a* =List files starting with a //you can list/copy/remove

ls *a =List files ending with a

ls *a* =Files just having letter a

ls ? =List single lettered file

ls ?? =Double lettered file

SHELL SCRIPT --> Used to automate jobs.

1. To Create file

vim file1.sh //Opens/create file

i //Allows you to insert contents

echo "Calender for month is `cal` "

echo "Number of users logged in `who` "

echo "List of files in my home dir `ls /root` "

:wq //Save and quit file

2. TO Execute file (3 methods to execute file)

sh <file>

./ <file> //For using this method give chmod 744 to file

source <file>

1. Input redirection (<)

cat < file1 -->Open a file

tr 'a' 'A' < file2 -->Translate letter a with A in file2

2. Ouptput redirection (>) //output can be redirected to file or terminal

cal > file3 -->redirect output to file

cal > /dev/tty3 -->redirect output third terminal

3. Error redirection (2>)

cal 222222 2> file4 -->Put error to file 4

tr 'a' 'A' file2 2> file5 -->Put error to file5

pipe (|) =Used to combine two related commands, output of first command is taken as input to second command

cal 2009 |less

cat /etc/passwd | cat -n

ps -au |grep tty2

Tee = Creates new file and stores previous commands output

cat a|tee b|sort|tee c|cat -n|tee d

1. head <command> =Shows first ten lines of file by default

head -n <command> =Shows first n num of lines //where n is num

2.tail <command> =Shows last ten lines of file by default

tail -n <command> =Shows last n num of lines //where n is num

3. <file> |less =Opens file from first page

n =next

N=Previous

gg=Beginning page of file

G=End of file

q=quit

4. cut -d: -f1 <file> //cut and display first filed of file,where d->delimeter f->field

cut -d: -f1,3 <file> //First and 3 field of file

cut -d: -f1-3 <file> //First to 3 field of file

5. sed 's/abc/xyz/g' <file> //Edit word abc with word xyz in file

s-> substitute g->globally

sed -i 's/abc/xyz/g' <file> //Edit permanently in file

sed '1p' <file> //Print first line twice

sed '1d' <file> //Delete first line

sed -e 's/abc/xyz/g' -e 's/123/456/g' <file> //Edit 2 words at a time

6. grep keyword <file>=Search keyword and display lines having keyword

grep -v keyword <file>=Search keyword and display lines NOT having keyword

grep -c keyword <file>=Count num of words having keyword

grep -l keyword <file1> <file2> =Search keyword in 2 files and display file tht contain keyword.

grep -L keyword <file1> <file2> =File tht does-not contain keyword.

7. sort <file> =Sort file

sort -r <file> =Sort file in reverse order

8. uniq -u <file> =Print uniq words in file

uniq -d <file> =Print repeated words in file

uniq -c <file> =Print num of lines having word repeated

vi <file> =Open/create file

1. iNSERT MODE

i=inserts, and you can type text

2. COMMAND MODE

:set nu =Set number

:set nonu =UnSet number

u =undo

ctrl r =redo

yy =Copy a line

yn =Copy n number of line //where n is num

dd =Delete a line

cc =Cut a line

:%s/abc/xyz/g //substitute abc with xyz globally

:1,4w <file> //Copy first four lines to a file

:4w <file> //Copy fourth line to a file

3. Execution mode

:wq =Write and quit

:q! =Quit without saving

:wq! =write and quit by force

VIM EDITOR =Advanced editor

1.vimtutor ->Has tutorial about vim

1.Open multiple files

a> vim -o <file1> <file2> ->open file horizontally

b> vim -O <file1> <file2> ->open file vertically

2.Select the text

a> vim <file1>

v ->

Nic cards are indentified by eth, 1st nic card is eth0,

2nd nic card is eth1

1. ifconfig -->shows you ipaddress +loopback and vmware ip (if any)

2. ifconfig eth0 -->shows you ipaddress

3. ifdown eth0 -->Disable your network temporarily

4. ifup eth0 -->Enable your network

->To Set ipaddress

1. system-config-network -->opens interface for entering ip/subnetmask

2. system-config-network-tui -->opens interface (tui is needed whn ur in gui)

-> Ip can be set manually or thru dhcp server

dynamic ip --> assigned thru dhcp server

static ip --> assigned manually

->After setting ip address

service network restart -->refresh new ip address

->config file for ipaddress

cat /etc/sysconfig/network-scripts/ifcfg-eth0 //observe entries

->To set hostname to machine

vim /etc/sysconfig/network //Opens the file

HOSTNAME=server1.example.com //Add this line into the file

-> hostname server1.example.com //Define once hostname on commandline also

->service network restart //For hostname to take effect

TO CONFIGURE PRINTER

1. **system-config-printer & OR**
2. **system --->Administration--->printer**

lpr <file> --> Give job to default printer

lpr -P <printername> <file> --> Give job to Specified printer

lpq --> List print queue of default printer

lpq -a --> List print queue of all printers

lpstat <printer> -->show status of specified printer

lpstat -->show status of all printer

lprm <jobnum> --> remove a job from printer

Anything under execution is called process

ps =Shows process of current terminal

ps -a =Shows process of all terminal

ps -au =Shows process of all terminal + user who started process

ps -x =All background running process

ps -au |grep tty3 -->show process of third terminal

ps -au |grep suma -->show process started by user

ps -au |grep yes -->show process of any terminal where yes command running

STATES OF PROCESS

R --> Running

S --> Interupptable sleep

T --> Stopped jobs

D --> Uninterruptable sleep

Z --> Zombie state

+ -->running in foreground

- -->running in background

KILL/STOP PROCESS

kill -l --->Show all signals to kill

kill -9 <pid> --->Kill abruptly

kill -15 <pid> --->Kill with warning message

kill -19 <pid> --->Stop a process

kill -18 <pid> --->Start a process

TO SCHEDULE JOB AT PARTICULAR TIME

date //Check for system date,lets assume its 10.12

at 10:15 / you want to execute command at 10.15

cal > /dev/tty3 //want to execute cal in terminal 3

ctrl d //save

at -l //List scheduled jobs

atrm //Remove all scheduled jobs

TO SCHEDULE A JOB REPETEADLY

1. * -->min

2. * -->hour

3. * -->day

4. * -->month

5. * -->week

crontab -e //Opens a editor to schedule a job

crontab -l --> list all cronjobs

crontab -r --> Remove all cronjobs

crontab -eu <user1> --> Set crontab to user1, as root

EXAMPLES OF CRON JOB

1. Execute hello @ 10.15

15 10 * * * echo "hello"

2. Execute date every 15 min in terminal 3

/15 * * * * date > /dev/tty3

3. Execute cal every 5th and 10th of EVERY month @ 10.15

15 10 5,10 * * cal > /dev/tty2

4. Execute cal between 5th to 10th of 1 and 3rd 01.20 p.m

20 13 5-10 1,3 * cal > /dev/tty2

SCHEDULE PRIORITY OF JOB

Nice command helps to set priority,priority ranges for -20 to +19

nice -n -20 yes 123 --> yes command gets highest prority

nice -n +10 firefox google.com --> browser Gets lowest prority

ONLINE STATUS

top -->shows online status of all running process

f ->customise fields of top

k ->kill process

d ->set delay

r ->renice priority of any job

Executing unrelated commands together/grouping commands

who;date -->execute both date and who

(who;date) > file1 -->redirect o/p of both command to file1

EXIT STATUS

exit status ranges from 0-255

Anythng other than 0 is set to be wrong commands

echo $? -->command to show exit status

a=3233 //local variable only for current shell

export a //Makes variable a globally available for subsequent shells

Inbuilt variables are defined in set|less and env|less

Login scripts -->scripts tht executes when person logs in

1./etc/profile

2./etc/profile.d/lang.sh

3./root/.bashrc

4./root/.bash_profile

5./etc/bashrc

NON-Login scripts -->scripts tht executes when you do su or normaluser

1./etc/profile

2./etc/profile.d/lang.sh

5./etc/bashrc

 # su - <user1> //Root switching as user1 --no authentication needed

$ su - root //Normal user switchin as root -- authentication needed

-->find <placetofind> <find-based-on> <parameter>

find /etc -name passwd

find /etc -size 10K OR +10K -10K

find /etc -type f OR d

find /dev -user suma

find /home -uid 505

Types of file

- ---> represents file

d ---> represents dir

s ---> represents socket file

l ---> represents linked file

p ---> represents piped file

c ---> represents character special file

b ---> represents block special file

stat <file> -->shows file status of whn read/written/permissionchanged

-->when you read a file Access time gets updated

-->when you write to file Modify time gets updated

-->when you change permission to file Change time gets updated

NETWORK CLIENTS

Browsing net in text

1. elinks <url> =Open a url

2. wget <downloads url> =Download url and store it under file in ur pwd

REMOTE LOGIN

1. ssh <remoteip> =Remotely login to a machine as root by default

2. ssh <user@remoteip> =Remotely login as normal user

COPY FILE FROM MACHINE TO MACHINE OVER NETWORK

Methods to copy file over network

1.scp -> secure copy

2.ftp -> file transfer protocol

3.lftp -> line file transfer protocol

4.gftp -> graphical file transfer protocol

5.telnet

1.scp

copy file from local to remote

a. scp <file1> <remoteip:/filepath> -->copy FROM local TO remote.

scp abc 192.168.0.1:/123 -->copy abc file to 192.168.0.1 's /123 dir.

copy file from remote to local

b. scp <remoteip:/path> /filepath --> copy FROM remote TO local

scp 192.168.0.2:/var/abc /root --> copy file /var/abc from 192.168.0.2 to ur machines /root dir

copy DIR from remote-local or local-remote use -r option

scp -r <remoteip:/path> /filepath --> copy FROM remote TO local

2. ftp

a. Anonymous login --> Allows users to login without a/c info about remote machine, but you get acess to only files of pub dir

allows only get(download) but not put(upload)

ftp <remoteip>

user anonymous

pass ..... //blank passwd

ftp> pwd -->shows pwd as / ,but ur placed under pub dir only.

ftp> ls -->shows remote machines file/dir list

ftp> ls -->shows remote machines file/dir list

ftp> !ls -->shows local machines file/dir list

ftp> get <file> -->Get file FROM remote machine

ftp> bye -->come out of ftp prompt

b. Non-Anonymous login --> Allows users to login with a/c info about remote machine,ftp allows login only as normal user.

ftp <remoteip>

user u1 //allows only to login as normal user

pass redhat

ftp> pwd -->shows pwd as / ,but ur placed under pub dir only.

ftp> ls -->shows remote machines file/dir list

ftp> ls -->shows remote machines file/dir list

ftp> !ls -->shows local machines file/dir list

ftp> get <file> -->Get file FROM remote machine

MONITORING LOGINS

last|less --->Shows all correct login

last -b|less --->Shows all wrong login

who --->Shows users logged in in all terminals

whoami --->Name of user whose logged in

w --->Shows remote login if any

UMASK CONCEPT

Original values for file was 666 and dir was 777 before umask came into existence so all used to get full permission for file/dir so to avoid

ROOT	FILE	DIR	
Originalvalue	666	777	//So now if root creates it gets
umask--value	022	022	// 644 for files and 744 for dir.
Default--value	644	744	

NormalUser	FILE	DIR	
Originalvalue	666	777	//So now if normal user creates it gets
umask--value	002	002	// 664 for files and 774 for dir.
Default--value	664	774	

Umask -> Responsilbe for setting default permissions to files/dir we create

umask of root is 0022

umask of normal user is 0002

SUID and SGID and STICKYBIT

1.SUID ->set userid (4)

suid is special permission(4) set along with existing permissions, which allows normal user to execute command with privilege of root.

ex: chmod 4770 /bin/cat

Login in two diff terminals and try below commands

TTY1 (As root) TTY2 (as u1)

1. whereis cat 2. cat /etc/gshadow (Not allowed)

3. ll /bin/cat 5. cat /etc/gshadow (Allowed)

4. chmod 4755 /bin/cat 8. cat /etc/gshadow (Not allowed)

6. chmod 755 /bin/cat

7. ll /bin/cat

2.STICKYBIT ->STICKYBIT (1)

stickybit is special permission(1) which stops non-owner of file from deleting the file from a dir for which stickybit is set.

ex: chmod 1770 /var

TTY 1 (AS Root) TTY2 (As u2)

1. cd /tmp 5. cd /tmp

2. ll -d /tmp 6. cat a (Allowed)

3. rm -rf /tmp/* 7. rm -rf a (Not allowed)

4. touch a b 12. rm -rf a (Allowed

8. chmod 777 /tmp 13. rm -rf b (Not allowd

9. ll -d /tmp

10. chmod 1777 /tmp

11. ll -d /tmp

LINUX-FILE-SYSTEM

Inode num is unique num given to file, we identify files by filename

but harddisk identify files by sequential uniq num called inode

lets observer what happnes to inode whne we copy/remove/move file

cp file1 file2 -->copy file1 file2

ls -li file1 file2 -->inode nums r diff,so file1 file2 occupies seperate space in hard-disk

mv file1 file2 -->move file1 file2

ls -li file1 file2 -->inode num of file1 is given to file2 file1 no more exist

rm file2 -->remove file

ls -li file2 -->inode of file2 gets vacant,next time you creat file inode of file2 is given to that.

LINKING FILE

when you copy a source to destination, contents of both are same, later you

update source or destination, other file wotn get updated, so linkign file

helps to update the file even if contents are added to file after linking.

1. Hard link

ln file1 file2 //hard link file1 file2

ll -i file1 file2 //observe inode/filesize

a. Inode nums are same

b. Size of file is same

c. If source is deleted data can be accessed from destination

d. Only files can be linked dir cant

2. Soft link

ln -s file1 file2 //soft link file1 file2

ll -i file1 file2 //observe inode/filesize

a. Inode nums are diff

b. Size of file is diff

c. If source is deleted data is lost

d. Even dir can be linked.

3. df -h //Shows current systems partion, and used and free space

mount //Shows List of partions and which dir it is mounted

fdisk -l // Also shows system partitions

du -h <file/dir> =Shows size occupied by the file/dir

4. eject //opens cd tray put cd then

eject -t //Close cd tray, thn to acces cd contents

To access contents of cd

mount /dev/cdrom /media //mounts contents of cd to /media dir

cd /media //Change dir to /media

cp OR ls //list contents or copy contents of cd

cd [and press enter] //Takes you out of /media dir

umount /media //Unmount /media dir before removing cd

To access contents of pendrive or any external device

fdisk -l //At end shows u device identity

OUTPUT OF fdisk -l looks like this

[root@server1 ~]# fdisk -l

Disk /dev/hda: 80.0 GB, 80026361856 bytes

255 heads, 63 sectors/track, 9729 cylinders

Units = cylinders of 16065 * 512 = 8225280 bytes

Device	Boot	Start	End	Blocks	Id	System
/dev/hda1	*	1	2	16033+	83	Linux
/dev/hda2		3	1177	9438187+	83	Linux
/dev/hda3		1178	1488	2498107+	83	Linux
/dev/hda4		1489	9729	66195832+	5	Extended

/dev/hda5 1489 1814 2618563+ 83 Linux

/dev/hda6 1815 1945 1052226 83 Linux

/dev/hda7 1946 2199 2040223+ 82 Linux swap / Solaris

/dev/hda8 2200 2215 128488+ 83 Linux

Note: sector size is 4096 (not 512)

Disk /dev/sda: 79.8 GB, 79824777216 bytes

26 heads, 50 sectors/track, 14991 cylinders

Units = cylinders of 1300 * 4096 = 5324800 bytes

Device Boot Start End Blocks Id System

/dev/sda1 1 14992 77953628 b W95 FAT32

------>> Here /dev/hda1 to /dev/hda8 are system partitions

and /dev/sda1 is external device identity

SO TO ACCESS CONTENTS OF EXTERNAL DEVICE

mount /dev/sda1 /mnt //Here identity of usb is sda1,and mounting to /mnt

cd /mnt //Change dir to /mnt

cp OR ls //list contents or copy contents of usb

cd [and press enter] //Takes you out of /mnt dir

umount /mnt //Unmount /mnt dir before removing usb

TO TAKE BACKUP USING TAR COMMAND

tar -cvf dump file1 file2 file3 //Take backup of file1,2,3 and store under dump dir

tar -tvf dump //List all files backed up under dump dir

tar -xvf dump //Restore all contents of dump to pwd

where c->create v->verbose f->file t->list x->extract

TO COMPRESS FILE (2 utilities)

1. gzip file //compress file

gunzip file.gz //once file compressed .gz extension added so file.gz

2. bzip2 file //compress file

bunzip2 file.bz2 //once file compressed .bz2 extension added so file.gz

Before compressing and after compressing observer file size by giving ls -lh and observe file-size

INSTALLATION

Mode of installation

1. enter --> Takes you to gui mode of installation

2. linux text --> Takes you to text mode of installation

Types of installation

1. Kickstart --> Unattentded installation

2. network installation -->use cd to boot then read remain from server

For Client to install a package it need 2 refer to a file to find which is server and read packages from there, so create a client.repo file in /etc/yum.repos.d directory.

vim /etc/yum.repos.d/client.repo

[Server]

name=rhel5

baseurl=ftp://192.168.0.154/pub/RHEL5.1/Server

enable=1

gpgcheck=0

1. yum clean all //changes to client.repo file updated

2. yum install <pkg> //Install a package

3. yum remove <pkg> //Remove a package

4. yum list <pkg> //Status of package

5. yum list all //List all packages installed in your machine

6. yum list available //List all packages available in server

Package are:-

1. Server //Server related packages

2. Cluster //Replication related packages

3. ClusterStorage //Storage related packages

4. Virtualization //vmware of redhat related packages.

134

Steps of boot process

1. BIOS ->Do POST ->power on self test and Check boot priority,

2. MBR ->I stage boot loader,partition table, o/s signature

3. /etc/grub/grub.conf ->root (hd0,0)

kernel /vmlinuz..... ro root=LABEL=/

initrd /initrd.....img

4. /etc/inittab ->/etc/rc.d/rc.sysinit ->Set hostname

->Set clock

->Enable selinux,quota

->check filesystem

->Enable root in rw mode

5. /etc/inittab ->runlevel ->/etc/rc.d/rcx.d (where x is num)

6. /etc/rc.d/rc ->whenever runlevel changes

7. /etc/rc.d/rc.local ->last script executed in boot process

GRUB FEATURES:

1. Grand unified boot loader

2. Grub indentifies upto 15 file systems

3. Grub has a pre-os enviornment use commands to load o/s

4. Grub supports LBA(Logical block addressing)

GRUB COMMANDS

1. root (hd0,0) ->mount the boot partition

2.find /etc/fstab ->find the partition which contains label of /

3.cat (hd0,4)/etc/fstab -> TO check label of root

OR

4.find /etc/fstab ->find the partition which contains label of /

5. root (hd0,4) ->Mount the route partiton

6. cat / + press tab ->shows contents of mounted partition

7. cat /etc/fstab ->open the file to see file contents

8. kernel /vmlinuz..... ro root=LABEL=/ ->Loads the kernel

9. initrd /initrd...img ->provides initital ram disk

10. boot -> Helps to boot the o/s

<u>TO EXTRACT THE INITRD</u>

1. mkdir initrd

2. cp /boot/initrd-2.6.18-53.el5.img initrd/

3. cd initrd/

4. ls

5. file initrd-2.6.18-53.el5.img

6. mv initrd-2.6.18-53.el5.img initrd.gz

7. ls

8. gunzip initrd.gz

9. ls

10.file initrd

11.cpio -ivd < initrd

<u>TO SET GRUB PASSWORD</u>

1.grub-md5-crypt ->generate the password

2.vim /etc/grub.conf ->open file (write below line in this file)

password --md5 <generated password> (write below hidden menu)

3.reboot

4.Try pressing e or c in grub prompt it wont let you unless you type p and provide password

TO EXTRACT SPLASH IMAGE

1. mkdir splash

2. cp /boot/grub/splash.xpm.gz splash

3. cd splash/

4. ls

5. gunzip splash.xpm.gz

6. ls

7. firefox splash.xpm

TO ADD MULTIPLE TEXT TERMINAL

1. vim /etc/inittab

2. 50:2345:respawn:/sbin/mingetty tty50 -->creates 50th terminal

3. init q -->make changes to inittab

4. chvt 50 -->change to 50th terminal

Note: To allow root to login also add entries to /etc/securetty.

TO MAKE SERVICE PERMANENT

1. chkconfig --list |grep network

2. chkconfig --list |grep sendmail

3. chkconfig --list |grep cups

4. chkconfig --list |grep vsftpd

5. chkconfig vsftpd on

6. chkconfig --list |grep vsftpd

7. chkconfig --levels 35 vsftpd on

chkconfig --levels 35 <service> <on/off>

TO SET A REPOSITORY

A> SERVER SETUP

1. mount 192.168.0.154:/var/ftp/pub/RHEL5.1 /media

2. cd /media/Server

3. rpm -ivh createrepo.....

4. rpm -ivh vsftpd.....

5. cd ..

6. cp -vrf /media/* /var/ftp/pub/

7. cd /var/ftp/pub

8. createrepo -v Server

9. createrepo -v Cluster

10. createrepo -v ClusterStorage

11. createrepo -v VT

B> vim /etc/yum.repos.d/server.repo

[Server]

name=RHEL 5 Server

baseurl=file:///var/ftp/pub/Server

enable=1

gpgcheck=0

C> yum clean all

CLIENT SETUP

[Server]

name=RHEL 5 Server

baseurl=ftp://192.168.0.154/pub/RHEL5.1/Server

enable=1

gpgcheck=0

YELLOWDOG UPDATE MANAGER

yum install <pkg> -->installs package

yum remove <pkg> -->remove package

yum list   <pkg> -->status of package

yum clean   all   -->Flush the cache

yum list   all   -->List all packages in server

yum list   available -->List all packages installed

RPM COMMANDS

Zenity -2.16.0   -2.el5 .i386   .rpm

syntax of rpm --> basename-versionnum-release.architecture.rpm

1.TO INSTALL

rpm -ivh <pkgname> -->Install a package

rpm -Uvh <pkgname> ->upgrade package even if older version dont exist

rpm -Fvh <pkgname> -->upgrade a packg ONLY if older version exist

rpm -ivh --force coreutils-5.97-12.1.el5.i386.rpm ((--force is used when the

package is not corrupted but files or commands produced by that package is

corrupted))

2.TO DELETE

rpm -e <basename> -->To erase a package

3.TO QUERY installed packages

rpm -q <basename> -->status of package

rpm -qi <basename> -->Query for info about package

rpm -qd <basename> ->List of documents created by package

rpm -qc <basename> ->List of configuration files created by package

rpm -ql <basename> -->List of files created by package

rpm -qf <command> -->show package which produced this command.

4. TO QUERY UN-Installed packages

use -p along with any options + Give full package num instead

of base name.

GROUP REPOSITORY

1.mount /dev/cdrom /media

2.cd /media

3.cp Server/repodata/comps-rhel5-server-core.xml /var/ftp/pub/Server/ repodata/

4.cp VT/repodata/comps-rhel5-vt.xml /var/ftp/pub/VT/repodata/

5.cp Cluster/repodata/comps-rhel5-cluster.xml /var/ftp/pub/Cluster/ repodata/

6.cp ClusterStorage/repodata/comps-rhel5-cluster-st.xml /var/ftp/pub/ClusterStorage/repodata/

7.cd

8.createrepo -g /media/Server/repodata/comps-rhel5-server-core.xml /var/ftp/pub/Server

9.createrepo -g /media/Cluster/repodata/comps-rhel5-cluster.xml /var/ftp/pub/Cluster

10.createrepo -g /media/ClusterStorage/repodata/comps-rhel5-cluster-st.xml /var/ftp/pub/ClusterStorage/

11.createrepo -g /media/VT/repodata/comps-rhel5-vt.xml /var/ftp/pub/VT/

12.vim /etc/yum.repos.d/server.repo //Add entries to the file

YUM GROUP RELATED COMMANDS

13.yum clean all

14.yum grouplist all

15.yum groupremove GNOME Desktop Environment

16.yum grouplist all

17.yum groupinstall Virtualization

KERNEL SERVICES

1.lsmod |less -->list all currently loaded modules

2. lsmod |grep <module>

modprobe <module> -->load a module

EX:. lsmod |grep usb_storage

modprobe usb_storage

3. lsmod |grep usb_storage

modprobe -r <module> -->Unload a module

EX: lsmod |grep usb_storage

modprobe -r usb_storage

3.modinfo <module> -->info about a module

All modules are stored under /lib/module/<kernelversion>/kernel

Inbuilt documents about kernel can by obtained by installing

a package yum install kernel-doc.

4.vim /etc/modprobe.conf -->config file for aliasing a module

SYNTAX: alias <newname> <oldname> -->write this inside file

EXAMPLE: alias pendrive usb_storage

modprobe <aliasname> -->load a module using alias name

modprobe -r <aliasname> -->Unload a module using alias name

5.Booleans related to kernel configurations are stored here.

ls /proc/sys/net/ipv4

6.vim /etc/sysctl.conf //To set kernel booleans

net.ipv4.ip_forward = 1 //1 enable, 0 disable, this boolean is

used for enabling ip forwarding

7.sysctl -p -->Make changes permanent

8.Dynamic creation of /dev directory

rm -rf /dev/* -->if /dev contents are removed

start_udev -->command dynamicaly creates contents of /dev

9. Seven types of files

1.(-) file -->touch/cat file

2.(d) dir --> mkdir dir

3.(l) link --->ln -s file1 file2

4.(c) characterspecial -->use mknod command

5.(b) Blockspecial -->use mknod command

6.(s) Socket file -->write program to create socket

7.(p) Pipe file --> and pipe file

10. To create blockspecial/characterspecial file

mknod <nameoffile> <typeoffile> <majornum> <minornum>

ex: mknod /dev/abc b 8 15

11./proc dir and /dev dir does not occupy any space in harddisk they just created in ram once system is booted.

SYSTEM SERVICES

GUI of redhat is maintained by XORG

All config files related to GUI is available in /etc/X11

1. Gui in runlevel 3 --> /etc/X11/xinit/xinitrc

2. Gui in runlevel 5 --> /etc/X11/prefdm

vim /etc/X11/xorg.conf -->config files to set display resolution

system-config-display -->tool to change display settings

TO ADD MULTIPLE GUI'S

Syntax: startx -- :n ->where n is number.

EX: startx -- :5 ->create 5th gui.

Configuring vnc server (virtual network console)

Server Config

1. yum list vncserver

2. vncserver -->creates passwd and creates .vnc/xstartup file

3. vncpasswd -->change passwd

4. vim /root/.vnc/xstartup

uncomment first 2 lines of above file

5. vim /etc/sysconfig/vncservers

uncomment last two lines of file + set

root:20 ->where root is user and 20 is session id

6. service vncserver restart

Client Config

1. yum install vnc -y

2. vncviewer //Opens a rectangular box

3. type serverip:sessionid / and you get connectd to server gui

SSH -->secured shell remote login

ssh <remoteip> -->connect to text terminal of remote machine

ssh <user>@<remoteip> ->connect as u1 txt termial of remote machine

ssh -X <remoteip> -->connect to GUI of of remote machine

ssh <remoteip> <command> -->execute command from remote machine

Crontab -->used to schedule jobs

1. cat /etc/crontab -->main config file of crontab

cat /etc/cron.weekly -->system scheduled weekly job

cat /etc/cron.hourly -->system scheduled hourly job

cat /etc/cron.monthly -->system scheduled montly job

cat /etc/cron.deny -->adding user to this file denies user from scheduling job

2. User scheduled cron jobs are stored here.

ls /var/spool/cron/root

ls /var/spool/cron/student.

cat /etc/cron.allow file doens not exist, if you add user to both /etc/cron.deny and /etc/cron.allow file.

cron.allow file takes precedence.

<u>USER ADMINISTRATION</u>

useradd <user> -->adds a user and updates below files

passwd <user> -->add password for a user

/etc/passwd

/etc/shadow

/etc/group

/etc/gshadow

	uid	gid
root	**0**	**0**
systemuser	**1-499**	**1-499**
normaluser	**500+**	**500+**

FILEDS OF /ETC/PASSWD

1. User

2. password pointer

3. uid

4. gid

5. comments

6. home dir

7. shell

To edit fields of user, or to create user with option

useradd -u <uid> <user> ->Add user with specified uid

useradd -g <gid> <user> ->changes users primary group

useradd -G <gid> <user> ->changes users secondary group

useradd -c <comments> <user> ->Add comments to user

useradd -d <homedir> <user> ->Add specified home dir

useradd -s <shell> <user> ->Add specified shell to user

chsh <user> -->change shell of user

finger <user> -->shows shell of user

id <user> -->shows users uid,gid info

usermod -option <user> -->modify existing info of user

userdel <user> -->Delete user, but leave home dir

userdel -r <user> -->Delete user and home dir both

when we add user it copies hidden file from /etc/skel dir.

and password aging policy are read from /etc/login.defs

2. FIELDS OF /ETC/SHADOW

1. user

2. password

3. num of days since jan 1970 passwd is changed

4. min num of days to wait for user to change his password

5. max num of days after which user is forced to change password

6. Once password is above to expire start giving warnign message

7. If pasword is expired, also disable a/c aftr this many days

chage -l <user> --->show password aging policy of user

chage <user> --->Change password aging policy of user

3. FIELDS OF /ETC/GROUP

1. group

2. group password

3. gid

4. members of group

4. FIELDS OF /ETC/GSHADOW

1. group

2. group password

3. admin of group

4. members of group

groupadd <grp> --> Add a group

groupdel <grp> --> Delete a group

gpasswd <grp> --> Set password for a group

AS ROOT

gpasswd -M <u1,u2> <grp1>-->Add u1,u2 as members to group grp1

gpasswd -A <u1> <grp1>   -->Make u1 as Admin to group grp1

gpasswd -d <u2> <grp1>   -->Delete u2 from group grp1

AS ADMIN (Login as admin and try below commands)

gpasswd -a <u3> <grp1>   -->Add u3 to group grp1

gpasswd -d <u3> <grp1>   -->Delete u3 from group grp1

newgrp <grp> -->when you know grouppassword of other group

you can change your group temporarily

ACL -ACCESS CONTROL LIST

getfacl <file> ->shows acl of file

setfacl -m u:u1:rw <file> ->set acl for user

setfacl -m g:g1:r <file> ->set acl for othergroup

setfacl -m o::-- <file> ->set acl for other users leftout

setfacl -x u:u1 <file> ->Delete acl set for user u1

setfacl -x g:g1 <file> ->Delete acl set for group g1

setfacl -b <file> ->Delete all acls been set.

setfacl -Rm u:u1:rw <dir> ->Acls set for dir is inherited by

files under that dir

VISUDO

VISUDO->Give privilege of root user to a induvidual user without giving him root password.

Permissions can be given to user

Permissions can be given to existing group

Permissions can be given to virutal group

To give to user

copy line 76 and edit tht copied line

ex: t1 ALL=SERVICES

To give to existing group

copy line 83 and edit tht copied line

ex: %india ALL=SERVICES

To give to Virtual group

copy line 20 and edit tht copied line

+ copy line 76 and edit tht copied line

ex: User_Alias ABC = i2, t2 //line 20 exp +

ABC ALL=SERVICES

Once visudo is set, when user who got privilege of visudo tries to execute the command, he has to include sudo + full path of command

ex: sudo /sbin/service network restart

LOG MESSAGES OF VISUDO

1. tailf /var/log/secure

SGID

Group owner of the directory should be inherited by the file under that group.

1.groupadd admin //Add a group

2.mkdir /share //Create dir under /

3.ll -d /share //See permission of dir created

4.chgrp admin /share //Change groupowner of dir

5.ll -d /share //See permission of dir created

6.touch /share/a //Create file under dir

7.ll -d /share //Permission of file dir

8.ll /share/a //permission of file

9.chmod 2770 /share //Add special permission 2 to dir

10.ll -d /share //permission of dir

11.touch /share/b //create file under dir

12.ll /share/b // permission of dir

DAC =Discretionary access control

MAC =Mandatory access control

DAC ->chmod,acl,sudo,visudo

MAC ->selinux

3 states of selinux

1. ENABLED = DAC + MAC both are implemented

2. PERMISSIVE = DAC + warning messages of MAC

3. DISABLED = ONLY DAC

TO show status of selinux

1. cat /etc/sysconfig/selinux

2. sestatus

3. getenforce

TO Change status of selinux from or to permissive/enabled

1. setenforce 0 ->set to permissive from enforcing

2. setenforce 1 ->set to enforcing from permissive

TO Change status of selinux from or to disabled/enabled

1. vim /etc/sysconfig/selinux OR

2. system-config-selinux

Once u change from disable-enable or enable-disable you haf

to reboot for changes

To enable/disable booleans for services

1. getsebool -a |grep <service> ->show status of service

2. setsebool -P <service> <on/off> ->Change status of service

To check status of selinux on files

1. ls -Z <file> ->show sestatus for files

To change context of selinux on files

1. chcon -t <policy> <file> OR

2. chcon -R --reference <srcfile> <dstfile>

To restore context/policy to originals

1. restorecon <srcfile>

Log Messages of selinux are stored in

1. tailf /var/log/audit/audit.log ->>text mode

2. sealert -b /var/log/audit/audit.log ->>gui mode

CREATE A PARTITION

1.fdisk -l -->identify harddisk identity

2.fdisk /dev/sda -->open to create partiton

m display help

n add a new partition ->first give enter, then +500M OR 1G

d delete a partition

l list known partition types

p print the partition table

q quit without saving changes

t change a partition's system id

w write table to disk and exit

3.partprobe ->update kernel about newly created partition

4. mkfs /dev/sda8 ->format partition using default(ext2)

5. mkdir /songs -->create a dir to mount the partition

6. mount /dev/sda8 /songs -->mount partiton on dir (temporary)

7. cd /songs --->once mounted you can get into dir

8. touch aaa bbbb ccc ddd --->create files under tht dir

9. vim /etc/fstab (without label)

/dev/sda8 /songs ext2 defaults 1 2

10. mount -a --> To check syntax of fstab

11. e2label /dev/sda8 --->check label of sda8 partition

12. e2label /dev/sda8 songs --->set label of sda8 partition

13 vim /etc/fstab (entries with label)

LABEL=/songs /songs ext2 defaults 1 2

14. mount -a

COMMANDS SHOWING INFO ABT HARDDISK

1.mount ->shows dir where partitions are mounted + filesystype

2.df -h ->shows use of partition

3.fdisk -l ->shows cylinders,id,partition

4.blkid ->shows filesystem type and some more info

5.dumpe2fs /dev/sdaX ->shows detailed info abt a partition

where X is a partition number

TO CONVERT EXT2 TO EXT3

1. blkid /dev/sda8

2.tune2fs -j /dev/sda8 -->add journal to ext2

3. blkid /dev/sda8

TO CONVERT EXT3 TO EXT2

1. umount /dev/sda8

2. tune2fs -O ^has_journal /dev/sda8 -->remove journal

3. mount /dev/sda8 /songs/

4. blkid /dev/sda8

EXT3 =ext2 + journal

journal =When file system is checked using e2fsck program, instead of checking entire file system checks only for latest files/dir been added aftr the last e2fsck check.

TO CREATE SWAP

a> Using Partiton

1.fdisk /dev/sda -->create a partition and label as 82

2.partprobe -->update kernel about newly created partition

3.mkswap /dev/sda9 -->create swap on /dev/sda9

4.free -m -->size of swap

5.swapon /dev/sda9 -->enable swap

6.free -m --->size of swap

7.swapoff /dev/sda9 -->disable swap

8.free -m --->check size of swap

b> Using a Big file created by dd command

1. dd if=/dev/zero of=/var/swapfile bs=1 count=1G

if ->input file of->output file bs->blocksize

2. df -h /var

3. free -m

4. mkswap /var/swapfile

5. swapon /var/swapfile

6. free -m

NFS CLIENT

showmount -e 192.168.0.38 -->shows dir shared by remote/server

1. TEMPORARY MOUNTING

mount 192.168.0.38:/songs /media

cd /media/

ls

mount

cd

umount /media

2. PERMANENT MOUNTING

->vim /etc/fstab

192.168.0.38:/songs /media nfs defaults 1 2

->mount -a

->mount

3. AUTOMOUNTING

a> DIRECT MAPPING

1. vim /etc/auto.master

/media /etc/auto.media

2. vim /etc/auto.media

abc -fstype=nfs 192.168.0.37:/share (Put this line in above file

3. service autofs restart

4. cd /misc

5. ls Cant see the dir

6. cd abc when u directly get into tht dir abc gets created

7. ls can see contents of /share dir

8. mount can see /misc/abc is mounted

If ur no more using the dir After 600 sec it automatically umounts.

b> INDIRECT MAPPING

1. vim /etc/auto.master

/- /etc/auto.direct (put ths line in above file)

2. vim /etc/auto.direct

/456 -fstype=nfs 192.168.0.37:/share (put this line in above file)

3. service autofs restart

4. ls /

5. ls /456

6. mount

QUOTAS

To set up quota

1.vim /etc/fstab

LABEL=/home /home ext3 defaults,usrquota 1 2

2.mount -a

3.mount

4.mount -o remount /home

5.mount

6.ls /home

7.quotacheck -cufmg /home

8.ls /home

9.quotaon /home

a> To Set quota for user limiting his block size (BLOCKS)

Set 80 as soft (min), and 120 as hard (max)

blocks=56

soft =blocks + soft(min) ex: 56 + 80

hard =blocks + hard(min) ex: 56 + 120

1.edquota -u suma (edit the blocks of file as given below)

Filesystem blocks soft hard inodes soft hard

/dev/sda7 56 136 176 7 10 12

2.su - suma

$> dd if=/dev/zero of=/home/suma/xyz bs=1024 count=80 //Warn

$> dd if=/dev/zero of=/home/suma/xyz bs=1024 count=120 //Stop

$> ctrl d

3.repquota -t /home

b> To Set quota for user limiting his num of files (INODES)

Set 10 as soft (min),and 12 as hard (max)

1.edquota -u raju (edit the inodes of file as given below)

Filesystem	blocks	soft	hard	inodes	soft	hard
/dev/sda7	56	0	0	7	10	12

2.su - raju

$> touch 1 2 3

$> ls -a |wc -l //Warns

$> touch 4 5

$> ls -a |wc -l //Stop

$> ctrl d

3. repquota -t /home

RAID (redundant array of indepedent disks)

1. raid 0 ->stripping

2. raid 1 ->mirroring

3. raid 2 onwards ->parity check

TO CREATE RAID

1. fdisk /dev/sda ->create 3 partitons and label as fd

2. partprobe

3. fdisk -l

4. mdadm -C /dev/md0 -n 2 -l 1 /dev/sda8 /dev/sda9 ->create raid device

where n ->num of device l ->raid level

5. cat /proc/mdstat ->status of raid device

6. mkfs.ext3 /dev/md0 ->format raid

7. mkdir /raid

8. mount /dev/md0 /raid ->mount raid

9. cp /etc/* /raid ->put some contents to raid dir

10.ls /raid/

11.mount

12.df -h

13.vim /etc/fstab ->mount raid permanently

/dev/md0 /raid ext3 defaults 1 2 (Add this line to /etc/fstab)

14.mount -a

TO VERIFY RAID

1. cat /proc/mdstat -->status of raid device

2. umount /raid

3. mdadm --stop /dev/md0 -->stop raid device

4. umount /raid

5. cat /proc/mdstat

6. mkdir /test1 -->create dir for mounting raid partitions

7. mkdir /test2

8. mount /dev/sda8 /test1 --> mount partiton on dir

9. mount /dev/sda9 /test2

10.ls /test1

11.ls /test2

12.umount /test1 -->umount the partitions

13.umount /test2

14.mdadm --assemble /dev/md0 /dev/sda8 /dev/sda9 -->assemble partitions

15.cat /proc/mdstat

16.mount /dev/md0 /raid --> mount back the raid

17.cat /proc/mdstat

TO RECOVER A FAILED PARTITION

1. cat /proc/mdstat

2. mdadm --fail /dev/md0 /dev/sda8 //Fail a partition

3. cat /proc/mdstat

4. mdadm --add /dev/md0 /dev/sda10 //Add a partition

5. mdadm --remove /dev/md0 /dev/sda8 //Remove faulty partition

6. cat /proc/mdstat

FIELDS OF /ETC/FSTAB

1.partition or label of partition

2.Directory to mount the partition

3.file system type

4.Defaults (rw,ro,acl,quota)

5.Dump (backup) 1->Take backup 0->dont take backup

6.File system check 1->first priority 2 ->second priority

0->Dont check file system

a>. TO TAKE BACK UP OF PARTITION USING DUMP COMMAND

1. dump -0uf 123 /boot ->Take backup of /boot in 123 file

2. du -h /boot/ ->check space of /boot and 123

3. du -h 123

4. mkdir /boot/extra ->add some more contents to /boot dir

5. cp /etc/* /boot/extra

6. du -h /boot/extra/

7. dump -1uf 456 /boot ->Again take backup of only newly added contents

8. du -h 456

9.du -h /boot/extra/

TO RESTORE BACK

1.cd abc ->go to dir where you want to restore

2.restore -rf /456 ->where /456 is place where we took backup

b>. TO TAKE BACK UP USING TAR COMMAND

1.tar -cvf backup file1 file2 file3 ->c-create v-verbose f-file,take backup of file1 to file3 in backup file

2.tar -tvf backup ->t-list num of files backed up

3.tar -xvf backup ->x-extract files backed up in pwd.

4.tar -rvf backup file4 file5 ->r-append more files to backed up file

5.tar -xvf backup -C /var ->Extract files inside /var dir

To uncompress and extract a file gzip file

1.gzip backup ->compress the file

2. tar zxvf backup ->uncompress + extract the file

To uncompress and extract a file bzip2 file

1.bzip2 backup ->compress the file

2. tar jxvf backup ->uncompress + extract the file

LVM (LOGICAL VOLUME MANAGER)

1. TO CREATE LVM

1.fdisk /dev/sda ->create 3 partitions + label to 8e

2.partprobe

3.fdisk -l

4.pvcreate /dev/sda8 /dev/sda9

5.pvdisplay /dev/sda8

6.pvdisplay /dev/sda9

7.vgcreate vg0 /dev/sda8 /dev/sda9

8.vgdisplay

9.lvcreate -L +200M -n /dev/vg0/home1

10.lvdisplay /dev/vg0/home1

11.lcreate -L +300M -n /dev/vg0/var1

12.lvdisplay /dev/vg0/var1

13.mkfs.ext3 /dev/vg0/var1

14 mkfs.ext3 /dev/vg0/home1

15.mkdir /home1

16.mkdir /var1

17.mount /dev/vg0/home1 /home1

18.mount /dev/vg0/var1 /var1

19.cp /etc/a* /home1

20.cp /etc/b* /var1

21.vim /etc/fstab

22.mount -a

TO EXTEND LVM

1.lvdisplay /dev/vg0/home1

2.lvextend -L +200M /dev/vg0/home1

3.ls /home1

4.resize2fs /dev/vg0/home1

5.ls /home1

TO REDUCE LVM

1. lvdisplay

2. umount /var1

3. e2fsck -f /dev/vg0/var1

4. resize2fs /dev/vg0/var1 100M

5. lvreduce -L -100M -n /dev/vg0/var1

6. mount /dev/vg0/var1 /var1

7. df -h

LVM SNAPSHOT

1. lvcreate -L 200M -s -n lv2 /dev/llc/lv1

TO EXTEND PV

1.pvcreate /dev/sda10

2.pvdisplay

TO EXTEND VG

1.vgextend vg0 /dev/sda10

2.vgdisplay vg0

TO REDUCE VG

1.vgreduce vg0 /dev/sda10

2.vgdisplay

TO REDUCE PV

1.pvremove /dev/sda10

2.pvdisplay

REMOVE LVM

1.lvdisplay

2.umount /dev/vg0/home1

3.umount /dev/vg0/var1

4.vim /etc/fstab

5.lvremove /dev/vg0/home1

6.lvremove /dev/vg0/var1

7.lvdisplay

REMOVE VG

1.vgdisplay

2.vgremove /dev/vg0

3.vgdisplay

REMOVE PV

1.pvdisplay

2.pvremove /dev/sda9

3.pvremove /dev/sda8

KICKSTART INSTALLATION

1. yum install system-config-kickstart vsftpd

2. system-config-kickstart

3. vim anaconda-ks.cfg (copy packages from this file and paste into ks.cfg)

4. cp ks.cfg /var/ftp/pub

5. service vsftpd restart

6. chkconfig vsftpd on

Client

1.Put the boot cd and reboot your machine

2.linux ks=ftp://192.168.0.37/pub/ks.cfg

VIRTULIZATION

1. yum groupinstall Virtualization

2. vim /boot/grub/grub.conf

default=0 (boot wih xen kernel)

3.Create a partition of 10-20G

4.Reboot

5.virt-manager (Tool to create guest o/s)

xm create <domain> ->create domain

xm shutdown <domain> ->shutdown domain

xm reboot <domain> ->reboot domain

xm save <domain> <file> ->take snapshot domain

xm restore <file> ->restore domain

cat /etc/xen/<domain> ->config file of the guest machine

NETWORK CONFIGURATION

1.eth0 ->first physical NIC card

2.eth1 ->second physical NIC card

eth0:0

eth0:1 ->Use this to assign virtual ip

To assign Multiple IP:-

1.To set multiple Ip temporarily

ifconfig eth0:0 192.168.0.11

ifconfig eth0:1 192.168.0.12 (Temp till service network restart)

2.To set multiple Ip permanently

system-config-network-tui (Create New device and name as eth0:0)

3. To set ip using tool in GUI

system-config-network-tui

4.dhclient ->reads the ip address from dhcp server

5.ifdown eth0   ->disable the card

ifup eth0   ->enable the card

6.To check physical status of nic card

ethtool eth0 (if yes link is present)

To change speed or duplex

ethtool -s eth0 autoneg off (before changing speed or duplex autoneg shuld b off

ethtool -s eth0 speed 10

ethtool -s eth0 duplex half

Steps of boot process

1. BIOS ->Do POST ->power on self test and Check boot priority,

2. MBR ->I stage boot loader,partition table, o/s signature

3. /etc/grub/grub.conf ->root (hd0,0)

kernel /vmlinuz..... ro root=LABEL=/

initrd /initrd.....img

4. /etc/inittab ->/etc/rc.d/rc.sysinit ->Set hostname

->Set clock

->Enable selinux,quota

->check filesystem

->Enable root in rw mode

5. /etc/inittab ->runlevel

6. /etc/rc.d/rc ->whenever runlevel changes

7. /etc/rc.d/rc.local ->last script executed in boot process

1. Problem -> MBR CORRUPTED

Corrupt -> dd if=/dev/zero of=/dev/sda bs=446 count=1

Error -> Stops at boot from for long time

Recovery -> Put first cd and reboot thn press linux rescue

chroot /mnt/sysimage

grub-install /dev/sda

2. Problem -> grub.conf file missing OR wrong entries in grub.conf

Corrupt -> mv /boot/grub/grub.conf /root

Error -> drops you to grub prompt

Recovery -> root (hd0,0)

kernel /vmlinuz..... ro root=LABEL=/

initrd /initrd.....img

3. Problem -> /vmlinuz or /initrd file missing from /boot dir.

Corrupt -> rm -rf /boot/vmlinuz..... rm -rf /boot/initrd.....

Error -> file not found

Recovery -> Put first cd and reboot thn press linux rescue

chroot /mnt/sysimage

cat /proc/sys/dev/cdrom/info ->check cdroms identity

mount /dev/hdb /media

cd /media/Server

rpm --ivh --force kernel.....

4. Problem -> /etc/rc.d/rc.sysinit OR /etc/inittab file missing

Corrupt -> rm -rf /etc/rc.d/rc.sysinit OR /etc/inittab

Error -> cannot that execute /etc/rc.d/rc.sysinit OR No process left

Recovery -> Put first cd and reboot thn press linux rescue

chroot /mnt/sysimage

cat /proc/sys/dev/cdrom/info ->check cdroms identity

mount /dev/hdb /media

cd /media/Server

rpm --ivh --force initscripts-8.45

5. Problem -> Wrong entry of runlevel 0 or 6

Corrupt -> vim /etc/inittab and edit runlevel to 0 OR 6

Error -> continously reboots

Recovery -> Reboot machine and go to single user mode

and edit vim /etc/initab file and change runlevel to

3 or 5. and press init q for changes.

6. Problem -> Wrong entries in the /etc/fstab

Corrupt -> vim /etc/fstab make some changes in the labels

Error -> Drops you to emergencey mode

Recovery -> First provide roots passwd

mount -o remount,rw /

vim /etc/fstab ->and correct the labels

If you dotn know label e2label /dev/sda1

keep checking for correct label e2label /dev/sda2

254

TO SET UP PRIMARY DNS SERVER

Set up static ip and host name for your machine

SERVER

1. yum install bind caching-nameserver -y

2. vim /etc/named.conf ->edit this file

3. named-checkconf /etc/named.conf

4. cd /var/named

5. cp localhost.zone forward.zone

6. vim forward.zone ->edit this file

7. named-checkzone <domainname>   forward.zone

8. cp forward.zone reverse.zone

9. vim reverse.zone   ->edit this file

10. named-checkzone <domainname>   reverse.zone

11.vim /etc/sysconfig/network-scripts/ifcfg-eth0 ->edit this file

PEERDNS=no

12.service network restart

13.vim /etc/resolv.conf   ->edit this file

14.service named restart

15.chkconfig named on

CLIENT

1. dig   <hostname>

2. host   <hostname>

3. host   <IP>

4. nslookup <IP>

5. nslookup <hostname>

$TTL 86400

@   IN SOA   server.matrix.com.   root (

42   ; serial (d. adams)

3H   ; refresh

15M   ; retry

1W   ; expiry

1D)   ; minimum

```
                IN NS           server.matrix.com.
server          IN A            192.168.0.59
client1         IN A            192.168.0.58
client2         IN A            192.168.0.57

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

zone "matrix.com"{
type master;
file "forward.zone";
};

zone "0.168.192.in-addr.arpa"{
type master;
file "reverse.zone";
};
search matrix.com
nameserver 192.168.0.59
$TTL  86400
@               IN SOA      server.matrix.com.     root (
                                    42              ; serial (d. adams)
                                    3H              ; refresh
                                    15M             ; retry
```

1W ; expiry

1D) ; minimum

IN NS server.matrix.com.

IN A 192.168.0.59

59 IN PTR server.matrix.com.

58 IN PTR client1.matrix.com.

57 IN PTR client2.matrix.com.

TO CONFIGURE SLAVE SERVER

MASTER MACHINE

1.vim /etc/named.conf

allow-transfer{ 192.168.0.43; }; ->put this line in masters named.

2.service named restart

SLAVES MACHINE

1. yum install bind caching-nameserver -y

2. vim /etc/named.conf ->edit this file

3. setsebool -P named_disable_trans on

4. chmod a+w /var/named

5. cd /var/named

6. ls -l

7. vim /etc/resolv.conf ->edit this file

8. service named restart

9. ls -l ->zone files get created

CLIENT

1. nslookup <hostname> ->reads from master

2. service named stop ->Do this is in master

3. nslookup <hostname ->reads from slave

options {

directory "/var/named";

allow-transfer { 192.168.0.24; };

};

zone "matrix.com"{

type master;

file "forward.zone";

};

zone "0.168.192.in-addr.arpa"{

type master;

file "reverse.zone";

};

options {

directory "/var/named";

};

zone "matrix.com"{

type slave;

file "forward.zone";

masters { 192.168.0.59; };

};

zone "0.168.192.in-addr.arpa"{

type slave;

file "reverse.zone";

masters { 192.168.0.59; };

};

nameserver 192.168.0.59

nameserver 192.168.0.24

SUBDOMAIN

$TTL 86400

@ IN SOA server.example.com. root (

45 ; serial (d. adams)

3M ; refresh

3M ; retry

1W ; expiry

1D) ; minimum

IN NS server.example.com.

```
            IN NS          llc.subdomain.example.com.
server              IN A          192.168.0.42
slave         IN A          192.168.0.10
llc.subdomain     IN A          192.168.0.33
test          IN A          192.168.0.1
$TTL  86400
@             IN SOA        llc.subdomain.example.com.      root (
                                    42              ; serial (d. adams)
                                    3H              ; refresh
                                    15M             ; retry
                                    1W              ; expiry
                                    1D )            ; minimum

            IN NS        llc.subdomain.example.com.
llc           IN A           192.168.0.33
client1             IN A          192.168.0.30
client2             IN A          192.168.0.31
client3             IN A          192.168.0.32

options {
directory "/var/named";
forwarders { 192.168.0.42; };
};

zone "subdomain.example.com"{
type master;
```

file "forward.zone";

};

zone "0.168.192.in-addr.arpa"{

type master;

file "reverse.zone";

};

nameserver 192.168.0.33

$TTL 86400

@ IN SOA llc.subdomain.example.com. root (

42 ; serial (d. adams)

3H ; refresh

15M ; retry

1W ; expiry

1D) ; minimum

IN NS llc.subdomain.example.com.

33 IN PTR llc.subdomain.example.com.

30 IN PTR client1.subdomain.example.com.

31 IN PTR client2.subdomain.example.com.

32 IN PTR client3.subdomain.example.com.

TO CONFIGURE SPLIT

IN MASTER

1. vim /var/named/chroot/etc/named.conf ->edit this file

2. named-checkconf /var/named/chroot/etc/named.conf

3. cd /var/named/chroot/var/named

4. cp forward.zone external

5. vim external ->edit the file to change ips

6. chgrp named external

7. service named restart

From any internal ip

1. host server.matrix.com ->wll find internal ip

From any external ip

1. host server.matrix.com ->wll find external ip

```
search  example.com
nameserver 192.168.0.42
$TTL  86400
@               IN SOA      server.example.com.      root (
                                    45              ; serial (d. adams)
                                    3M              ; refresh
                                    3M              ; retry
                                    1W              ; expiry
                                    1D )            ; minimum

                IN NS       server.example.com.
                IN NS       llc.subdomain.example.com.
```

```
server                  IN A          192.168.10.42
slave          IN A          192.168.10.10
llc.subdomain      IN A          192.168.10.33
test           IN A          192.168.10.1
acl "internal" { 192.168.0.42; 192.168.0.34; };
options {
directory "/var/named";
#allow-transfer { 192.168.0.10; };
};

view "internal" {
match-clients { "internal"; };
zone "example.com"{
type master;
file  "forward.zone";
};

zone "0.168.192.in-addr.arpa"{
type master;
file  "reverse.zone";
};
};

view "external"  {
match-clients { "any"; };
zone "example.com" {
```

type master;

file "external";

};

};

To sync same time b/w server/client

1. yum install xinetd

2. chkconfig xinetd on

3. chkconfig --list

4. chkconfig time-stream on

To sync same time on client

1. rdate -s <Serversip>

<u>Configure RNDC SERVER</u>

1. rm -rf /etc/rndc.key

2. rndc-confgen -a

3. vim /etc/named.conf //Add entires to this file

4. vim /etc/rndc.conf //Add entries to this file

5. scp /etc/rndc.conf 192.168.0.20:/etc

6. pkill named

7. service named restart

8. rndc reload

Configure RNDC CLIENT

1. rdate -s <serverip>

2. yum install bind caching-nameserver -y

3. vim /etc/rndc.key

4. rndc reload

5. rndc stop

BOGUS

```
options {
        directory "/var/named";
        forwarders { 192.168.0.27; };
        allow-recursion { 192.168.0.64; };
        blackhole { 192.168.0.27; };
//      allow-query { 192.168.0.12; };
//      allow-transfer { key "abc"; };
        recursion no;
};

/*
key "abc" {
algorithm hmac-md5;
```

secret "aUkgS1un8SgqQ1DODU/Seg==";

};

***/**

server 192.168.0.27 { bogus yes; };

zone "example.com" IN {

type master;

file "localhost.zone";

};

zone "0.168.192.in-addr.arpa" IN {

type master;

file "named.local";

};

NIS SERVER

1. yum install ypserv -y

2. nisdomainname RHCSS

3. vim /etc/sysconfig/network

NISDOMAIN=RHCSS

4. service network restart

5. vim /etc/exports

/home *(rw,sync)

6. service portmap restart

7. service nfs restart

8. service ypserv restart

9. cd /var/yp

10. /usr/lib/yp/ypinit -m

NIS CLIENT

1. umount /home

2. comment home-dir line in /etc/fstab

3. authconfig-tui

4. ypwhich ->shows niservers hostname/ip

5. vim /etc/auto.master

/home /etc/auto.misc

6. /etc/auto.misc

u1 -fstype=nfs 192.168.0.82:/home/u1 //Login as u1

*** -fstype=nfs 192.168.0.82:/& //Login as any user**

7. service autofs restart

8. Log out and login as u1 user.

NIS SERVER

1. yum install ypserv -y

2. nisdomainname RHCSS

3. vim /etc/sysconfig/network

NISDOMAIN=RHCSS

4. service network restart

5. vim /etc/exports

/home *(rw,sync)

6. service portmap restart

7. service nfs restart

8. service ypserv restart

9. cd /var/yp

10. /usr/lib/yp/ypinit -m

NIS CLIENT

1. umount /home

2. comment home-dir line in /etc/fstab

3. authconfig-tui

4. ypwhich ->shows niservers hostname/ip

5. vim /etc/auto.master

/home /etc/auto.misc

6. /etc/auto.misc

u1 -fstype=nfs 192.168.0.82:/home/u1 //Login as u1

*** -fstype=nfs 192.168.0.82:/& //Login as any user**

7. service autofs restart

8. Log out and login as u1 user.

MASTER ENTRY WHILE SETTING UP SLAVE

1. vim /var/yp/Makefile

NOPUSH=False

2. /usr/lib/yp/ypint -m

3. service ypserv restart

SLAVE ENTRIES

1. yum install ypserv

2. vim /etc/sysconfig/network

NISDOMAIN=RHCE

3. service network restart

4. nisdomainname RHCE

5. service ypserv restart

6. /usr/lib/yp/ypinit -s <master'shostname) //Better configure Dns or hosts file

NIS CLIENT

1. authconfig-tui (Bind to both master,slave ip)

2. ypwhich ->shows masters ip

3. Stop ypserv in master

4. ypwhich ->shows slave ip

TO CONFIGURE NFS SERVER

1. mkdir /share ->create a dir under / which you want to share.

2. chmod a+w /share ->Allow all to write to the /share dir

3. vim /etc/exports ->write entries of that share in this file

a. /abc *(rw,sync) //All Domain

b. /abc *.example.com(rw,sync)

/abc *.matrix.com(rw,sync) //Two Domains

c. /abc *.example.com(rw,sync) //Only Example domain

d. /abc 192.168.0.82(rw,sync) //Single Machine

e. /abc 192.168.0.82(rw,sync) 192.168.0.42(rw,sync) //Double Machine

f. /abc 192.168.0.1-10(rw,sync) //Range of ips

4. service portmap restart

5. service nfs restart

NFS CLIENT

showmount -e 192.168.0.38 -->shows dir shared by remote/server

1. TEMPORARY MOUNTING

mount 192.168.0.38:/songs /media

cd /media/

ls

mount

cd

umount /media

2. PERMANENT MOUNTING

->vim /etc/fstab

192.168.0.38:/songs /media nfs defaults 1 2

->mount -a

->mount

To configure httpd

1. IP BASED

1. yum install httpd -y

2. ifconfig eth0:0 192.168.0.32 ->set virtual ip

3. ifconfig eth0:1 192.168.0.33

4. vim /etc/httpd/conf/httpd.conf ->edit this files

5. vim /var/www/html/index.html ->create web page here

6. service httpd restart

7. elinks 192.168.0.30

8. elinks 192.168.0.32

9. elinks 192.168.0.33 //Displays webpage for all 3 ips

2. NAME BASED

1. vim /etc/httpd/conf/httpd.conf ->Edit line 972

2. mkdir /var/www/virtual

3. cd /var/www/virtual

4. vim index.html

5. vim /var/named/forward.zone //5th to 7th To be done in Dns server

www CNAME server.matrix.com

6. service named restart

7. host www.matrix.com

8. vim /etc/httpd/conf/httpd.conf ->Edit End of line

9. service httpd restart

10. elinks server.matrix.com

11. elinks www.matrix.com //Shows page in /var/www/virtual

12. elinks server.matrix.com //Shows page in /var/www/html

3. HOST AUTHENTICATION

1. vim /etc/httpd/conf/httpd.conf ->Edit the file

2. service httpd restart

3. elinks server.matrix.com ->Allows/Deny as per defined

4. USER AUTHENTICATION

1. vim /etc/httpd/conf/httpd.conf ->Edit the file

2. cd /var/www/html

3. vim .htaccess ->create this file and add contents

4. htpasswd -c /var/www/html/.htpasswd <user1>

5. htpasswd /var/www/html/.htpasswd <user2>

6. service httpd restart

7. elinks server.matrix.com ->Authenticate for user/passwd

5. HTTPS

1. yum install mod_ssl -y

2. cd /etc/pki/tls/certs

3. rm -rf localhost.crt

4. rm -rf ../private/localhost.key

5. make genkey

6. make testcert

7. vim /etc/httpd/conf/httpd.conf ->Edit this file

8. setsebool -P httpd_disable_trans on

9. service httpd restart

10. URL--> https://server.matrix.com ->Allow only https

6. USERHOMEDIR

1. vim /etc/httpd/httpd.conf ->Edit this file

line 355 ->Comment the line

line 362 ->UNComment the line

line 370 to 381 ->UNComment the line

2. service httpd restart

2. cd /home/u1 ->get into user u1's home dir

3. mkdir public_html

4. cd public_html

5. vim index.html

6. chmod a+x /home/u1

7. chcon -R --reference /var/www/ /home/u1/public_html

8. elinks server.matrix.com/~u1

VSFTPD SERVER

1. yum install vsftpd -y

2. service vsftpd restart

3. setsebool -P ftp_home_dir on ->Do this to allow user to connect

CLIENTS

ANONYMOUS LOGIN (connecting as anonymous or ftp user and blank passwor)

ftp <ftpserver>

name ftp

pass ..... (Blank Password)

ftp> pwd ->you connect to pub dir

ftp> ls ->shows remotes file list

ftp> !ls ->shows local file list

ftp> get <file> ->Copy/download files

ftp> put <file> ->Cannot UPLOAD files

ftp> cd /home ->NOT allowed to change dir

ftp> bye ->Log out from ftp prompt.

NON-ANONYMOUS LOGIN (connect as normal user of remote machine)

ftp <ftpserver>

name u1

pass 1

ftp> pwd ->you connect to home dir of u1

ftp> ls ->shows remotes file list

ftp> !ls ->shows local file list

ftp> get <file> ->Copy/download files

ftp> put <file> ->Copy/Upload files

ftp> cd /home ->Allowed to change dir

ftp> bye ->Log out from ftp prompt.

FTP USERS FILE

If you want to stop one or two users from using ftp

enter there names in this file. That stops that user

from connecting throu ftp.

To stop many users from using ftp connection

1. Add the users whom you want to stop using ftp to this file +

2. And add this to /etc/vsftpd/vsftpd.conf userlist_deny=YES

3. service vsftpd restart

OPTIONS FOR /ETC/VSFTPD/VSFTPD.CONF

1. anonymous_enable=YES ->Allow anonymous login

anonymous_enable=NO ->Stop anonymous login

2. local_enable=YES ->Allow normal user to login

local_enable=NO ->Stop normal user to login

3. write_enable=YES ->Allow normal user to create files by default

write_enable=NO ->Stop normal user to create files

4. To allow Anonymous user to UPLOAD file.

a> #anon_upload_enable=YES ->uncomment this line +

b> chmod a+w /var/ftp/pub   ->/var/ftp/pub dir should be writable

c> write_enable=YES   ->write enable should b yes

5. If Anonymous user uploads file,owner of file is ftp by default,

but if you want u1 to be owner of file change here.

#chown_uploads=YES

#chown_username=u1 ->uncomment this 2 lines + user=u1

6.no_anon_password=YES ->Wont prompt the password for anonymous user

no_anon_password=NO   ->Prompt the password for anonymous user

<u>TO CONFIGURE SAMBA SERVER</u>

1. mkdir /samba

2. chmod a+w /samba

3. yum install samba -y

4. vim /etc/samba/smb.conf

[public]

comment = Only users

path = /samba

public = yes

browseable=yes

writable = yes

printable = no

write list = +staff

5. service smb restart

OPTIONS THAT CAN B GIVEN IN FILE

1.If browseable=yes ->we CAN see the shared dir

If browseable=no -> we CANNOT see the shared dir

2.If public=yes ->Allows anonymous Login

If public=no ->Stops anonymous Login

3. writable = no ->uploading is denied for BOTH the users.

writable = yes ->uploading is allowed for BOTH the users

4a. writable =no + ->Allows only u1 to upload files, but

b. write list =u1 both writable=no and writelist=u1 shuld b enabled

5. hosts allow=127. 192.168.0.20 ->Allow only 192.168.0.20 ip to access share, other ips are denied

CLIENT COMMANDS

1.smbclient -L //192.168.0.48/share ->List directories shared

2. smbclient //192.168.0.48/share ->Anonymous Login

get <file>

put <file>

3a. To generat passwd to allow for non-anonymous login

1. smbpasswd -a u1

2. service smb restart

3b. smbclient //192.168.0.48/share -U u1 ->Non Anonymous Login

get <file>

put <file>

TO INSTALL VMWARE ON LINUX

1.mount 192.168.0.154:/dumps /media

2.cd /media/VMWARE

3.rpm -ivh VMware-server-1.0.1-29996.i386.rpm

4.yum install gcc kernel-devel xinetd -y

5.vmware-config.pl

6.cat vmware-server.key

TO EXTRACT WINDOWS IMAGE

1.scp nanju@192.168.0.154:/home/nanju/suma/raj.tar.bz2 .

2.tar jxvf raj.tar.bz2 -C /var/lib/vmware

TO CONFIGURE SQUID SERVER

Squid Server

1. yum install squid -y

2. vim /etc/squid.conf //Configure server

3. Applications-->Internet-->Edit--Preference-->connectionsettings

(Give squid servers ip, So all connections to internet will pass

Through the squid server)

4. service squid restart

Squid Client

1. Applications-->Internet-->Edit--Preference-->connectionsettings

(Give squid servers ip, So all connections to internet will pass

Through the squid server)

Squid Authentication

1. line 1572 -> auth_param basicprogram /usr/lib/squid/pam_auth

line 2410 -> acl password proxy_auth REQUIRED

line 2527 -> http_access allow password.

2. Service squid restart

TO CONFIGURE PROC AS MDA AS FILTER AGENT TO BLOCK LOWER LEVEL SPAMS

touch /etc/procmailrc ->create file for root

touch /home/u1/.procmailrc ->create file for user u1

 # su - u1 ->login as u1

vim .procmailrc

:0

*** ^From.u2@server.matrix.com**

user 2

 # su - u2

mail u1@server.matrix.com

 # su - u1

ls

vim user2

Refer man page of proc -> man procmailex for more options

1>> TO CONFIGURE SMTP

1. vim /var/named/forward.zone (Dns should resolve mail server)

server IN MX 5 server.example.com.

2. service named restart

3. host -t MX server.example.com

4. yum install sendmail-cf -y

5. vim /etc/mail/sendmail.mc

line 115 ->comment

line 155 -> put your domain name

6. make -C /etc/mail

7. service sendmail restart

Verify -> netstat -tlpn |grep :25 (uses loopback after configuration uses ip)

2>> TO SEND MAILS

hhg

A> Using telnet

-> telnet server.matrix.com 25

-> mail from:u1@server.matrix.com

-> rcpt to: u2@server.matrix.com

-> data

-> subject: hello

-> aaaaaaaaaaaaaaaaaaaaaaaa

aaaaaaaaaaaaaaaaaaaaaaaa (then press enter and . to send)

quit

B> Using MAIL command

-> mail u1@server.matrix.com

aaaaaaaaaaaaaaaaaaaaaaaa (again . to send mail)

C> Using MUTT command

-> mutt u1@server.example.com

3>> TO CHECK MAILS

1. su - <user>

mail

2. mail -u <user>

3. tailf /var/spool/mail/<user>

just put your domain name here

1. server.matrix.com

2. service sendmail restart

1>> TO SET ALIASES TO USER,

Mail sent to a virtual group to be received by all users of virtual group

1. vim /etc/aliases

admin: u1,u2

2. newaliases

3. mail admin@server.matrix.com

4. mail -u u1

5. mail -u u2 ->both would haf received same mail

TO CONFIGURE VIRTUSER TABLE

Mail sent to u1 should be redirected to u2.

1. vim /etc/mail/virtusertable

u1@virtual.matrix.com u1@server.matrix.com

2. vim /etc/mail.local-host-names

server.matrix.com

virtual.matrix.com

3. make -C /etc/mail

4. service sendmail restart

5. mail u1@virtual.matrix.com

6. mail -u admin ->Receives mail

Any mail sent user of host virtual.matrix.com is received by admin user of server.matrix.com

TO RESTRICT OR ALLOW ACCESS TO CERTAIN USERS/DOMAIN

1a. To Reject mails FROM this IP

->vim /etc/mail/access

192.168.0.45 REJECT ->This IP cant send/receive mails

b. make -C /etc/mail

c. service sendmail restart

d. -> telnet server.matrix.com 25 (Try this frm 192.168.0.45 machine)

mail from:u2@server.matrix.com ->Not Allowed

2a. To Reject mails FROM this Domain

->vim /etc/mail/access

192.168.0 REJECT ->Any ips frm this domain cant access

2b. make -C /etc/mail

2c. service sendmail restart

2d. telnet server.matrix.com 25

mail from:u2@server.matrix.com ->Not Allowed

3a. To Reject mails TO u1

->vim /etc/mail/access

TO:u1@server.matrix.com REJECT ->u1 cant receive mail

3b. make -C /etc/mail

3c. service sendmail restart

3d. telnet server.matrix.com 25

mail from:u2@server.matrix.com ->ok

rcpt to :u1@server.matrix.com ->FAIL

4. To Reject mails FROM u1

-> vim /etc/mail/access

FROM:u1@server.matrix.com REJECT ->u1 cant send mail

4b. make -C /etc/mail

4c. service sendmail restart

4d. telnet server.matrix.com 25

mail from:u1@server.matrix.com ->FAIL

5. To Reject mails FROM u1 with a ERROR

-> vim /etc/mail/access

FROM:u1@server.matrix.com ERROR:STUPID ->display error message

5b. make -C /etc/mail

5c. service sendmail restart

5d. telnet server.matrix.com 25

mail from:u1@server.matrix.com -> Get error message

<u>TO MASQUERADE THE FROM ADDRESS OF MAIL</u>

1. vim /etc/mail/sendmail.mc (Add below lines to sendmail.mc file)

MASQUERADE_AS(`suma.virtual.com')dnl ->edit hostnmae

FEATURE(masquerade_envelope)dnl ->uncomment

FEATURE(masquerade_entire_domain)dnl - uncomment

MASQUERADE_DOMAIN(suma.virtual.com)dnl ->edit hostname

FEATURE(genericstable)dnl ->write below 3 lines

FEATURE(always_add_domain)dnl

GENERICS_DOMAIN_FILE(/etc/mail/local-host-names)dnlt

2. vim /etc/mail/local-host-names

server.example.com

suma.virtual.com

3. vim /etc/mail/genericstable

@server.matrix.com admin@suma.virtual.com

4. make -C /etc/mail

5. service sendmail restart

6. mail u1@server.matrix.com

7. mail -u u1 ->mail appears to come from admin@suma.virtual.com

thou mail was actually sent from root@server.matrix.com

TO CONFIGURE DOVECOT

1.yum install dovecot -y

2.vim /etc/dovecot.conf

uncomment line 17

3.service dovecot restart

TO CHECK MAIL USING POP3

1. telnet server.matrix.com 110

user <username> ->Type user and provide username

pass <password> ->Type pass and provide password

list ->List the mails

retr 1 ->Read the mail

quit

2. mutt -f pop3://u1@server.matrix.com

TO Check mail using IMAPS AND POP3S (Secured)

1. cd /etc/pki/tls/certs

2. make dovecot.pem

3. cp dovecot.pem /etc/pki/dovecot/certs

4. cp dovecot.pem /etc/pki/dovecot/private

5. vim /etc/dovecot.conf

uncomment line 87,88

6. service dovecot restart

2> mutt -f imaps://u1@server.matrix.com

TO CONFIGURE POSTFIX

1. yum install postfix system-switch-mail -y

2. system-switch-mail ->change to postfix

3. cd /etc/postfix

4. vim /etc/postfix/main-cf

line 69 ->myhostname=server.matrix.com

line 72 ->mydomain=matrix.com

line 92,93 ->uncomment both lines

line 107 ->uncomment

line 110 ->comment

line 155 ->remove localhosts entry

5. service postfix restart

you can send/receive mails

To configure Access Restrictions

1. vim /etc/postfix/access

copy line 387

smtpd_client_restrictions =check_client_access hash:/etc/postfix/access

2. vim /etc/postfix/main.cf

And paste above line in file

smtpd_client_restrictions =check_client_access hash:/etc/postfix/access

3. vim /etc/postfix/access

192.168.0.45 REJECT

4. postmap access

To redirect mail from one person to other.

1. vim /etc/postfix/virtual

copy line 149

virtual_alias_maps = hash:/etc/postfix/virtual

2. vim /etc/postfix/main.cf

And paste below line in file

virtual_alias_maps = hash:/etc/postfix/virtual

3. vim /etc/postfix/virtual

abc@server.matrix.com suma@server.matrix.com

4. postmap virtual

XINETD

1. yum install xinetd

2. yum install telnet-server

3. service xinetd restart

4. chkconfig xinetd on

5. chkconfig --list all |grep telnet

6. chkconfig telnet on

1. To connect to telnet server

---> telnet <server.ip>

Login: user1

passwd: 1

2. To Allow only particular client to connect to your server

---> vim /etc/xinetd.d/telnet

only_from =<clients.ip> -->only this client is allowed to telnet

3. To stop a Client to access any type of xinetd based application

--> vim /etc/xinetd.conf

no_access =<clients.ip> ->this client cant access any xinetd application

--> service xinetd restart

---> telnet <server.ip> ->Not allowed

4. To stop a Client to access xinetd services in particular time

--> vim /etc/xinetd.conf

access_times=10.00-12.00

--> service xinetd restart

---> telnet <server.ip> ->Not allowed if its not 10.00 to 12.00

---> telnet <server.ip> ->Allowed if time is b/w 10.00 to 12.00

--> tailf /var/log/secure ->to check client connecting to your server.

PORT MONITORING TOOLS

---> SHOWS all applications running in local machine:

1. pkg:-net-tools-1.60-73 -> installed by default

netstat -tulpn |grep <service>

t->tcp u->udp l->listening p->pid n->numericip

2. pkg:-nmap-4.11-1.1 -> installed by default

nmap <ip.address> |grep <service>

---> SHOWS all applications running in entire network:

1. yum install wireshark

Applications-->Internet-->wireshark-->capture

2. pkg:-tcpdump-3.9.4-11.el5 -> installed by default

tcpdump -c <ip.address> |grep <service>

TCPWRAPPERS

Services which contain libwrap module can use hosts.deny to control Access

ldd /usr/sbin/vsftpd |grep libwrap

ldd /usr/sbin/sendmail |grep libwrap

ldd /usr/sbin/sshd |grep libwrap

To Restrict a host/network to control access to a Service.

1. Using Hostname/Domainname

vim /etc/hosts.deny

-> vsftpd *.example.com ->All hosts in example.com denied to access ftp

-> vsftpd server.example.com ->Host server in example.com denied to access

2. Using Ipaddress/Network

vim /etc/hosts.deny

-> vsftpd 192.168.1.0/255.255.255.0 ->All hosts in 1.0 N/W denied.

-> vsftpd 192.168.1.4 ->Host 1.4 denied.

3. To Deny all Except few.

vim /etc/hosts.deny

-> sshd:ALL EXCEPT matrix.com ->Any domain other than matrix.com are denied the Access to ssh.

4. To Allow all Except few.

vim /etc/hosts.allow

-> ALL *.example.com EXCEPT *.matrix.com ->Any domain other than matrix.com are Allowed to Access.

Both entries allow/deny can be given in either hosts.allow or hosts.deny file

SSH

--> Pkg -openssh

Daemon -sshd

Portnum -22

Files -/etc/ssh/sshd_config

.ssh/*

---> vim /etc/ssh/sshd_config

1. line 13 -> change port num.

Port 53

service sshd restart

client connecting to your machine should connect giving like this

ssh -p 53 <server ip> ,Only then it connects.

2. line 37 -> Allow/stop user to ssh

AllowUsers u1

DenyUsers u2

service sshd restart

This stops a client to connect as u2 and can connect as only u1 user.

3. line 37 -> Allow/stop user to ssh

AllowGroups asia

DenyGroups america

service sshd restart

This stops a client to connect as any members of america, and can connect

as any member os america.

4. line 38 -> Login grace time.

LoginGraceTime 1m

service sshd restart

Once you connect to sshserver, you haf to provide passsword within a min,

or connection fails.

5. line 39 -> Root login allowed/not-allowed

PermitRootLogin no

service sshd restart

This stops a client to ssh as root user,anb can connect as normal user only

6. line 41 -> Password prompts

MaxAuthTries 1

service sshd restart

Password is prompted only twice within which he has to give right password

to authenticate.

7. line 96 -> Stop Gui Access

X11Forwarding no

service sshd restart

Thou the client connects to your server using ssh -X <server.ip>, they

wont be able to connect to GUI of Server

8. Generating Public/Private key

--> To generate the key ->Generates id_dsa,id_dsa.pub files under .ssh dir

ssh-keygen -t dsa

--> To copy key to client machine ->copies id_dsa to .ssh of clients machine

ssh-copy-id -i /root/.ssh/id_dsa <clients.ip>

PAM

A. First field

1.auth -> Module authenticate the user,by checking the password

2.account -> Module verifies that access is allowed,by checking if users account is valid,expired,allowed to access in this time of day.

3.passwd -> Module sets and verify the passwords

4.session -> Module configure and manage user sessions. Represents the enviornment of a user.

B. Control Flag

1. required -> Module result must be successfull for authentication to continue.

if result failed returns failure only after remaining modules are invoked.

2. requisite -> Module result must be successfull for authentication to continue.

But if module result is fail user is noticed immediately with

message reflecting failed required or requiste module

3. Sufficient -> If module results fail , it is ignored, but if its successful and no required modules is failed ,then user is authenticated to

service.

4. Optional -> If module results fail,it is ignored.If module result is successful it does not play role in overall success/failure for module.

/etc/pam.d/login

1.account required pam_nologin.so

Checks for /etc/nologin file, If exists stops normal user from logging in.

If module is commented, allows user to login even if file /etc/nologin exists

2.auth requisite pam_access.so

Checks for user name, If username is correct only then it prompts for password

if user incorrect,Displays login incorrect.But if this module is commented though

username is wrong it prompts for password

3.session required pam_mkhomedir.so

Create user without homedir --> useradd -M suma

If homedir doesnt exist,this module creates home dir as soon as user is logged in.

But if this module is commented if user is created without homedir,it wont create

home dir.

4. auth auth [user_unknown=ignore success=ok ignore=ignore default=bad]

pam_securetty.so

Checks if any terminal is commented in the /etc/securetty, if any terminal is

commented root cannot login in that terminal,But if this module is commented

root can login in a terminal, though that particular terminal is commented in

/etc/securetty file.

suma ..... maxlogins 3

Allows this user to login in only 3 terminals, after 3 logins, we cant even

ssh as that particular user. Even ssh not allowed because USEPAM yes in

/etc/ssh/sshd_config

/etc/pam.d/su

auth required pam_rootok.so

Change above line from sufficent to required,So even root user needs password

when he tries to do su to noabove line from sufficent to required,So even root user needs password

when he tries to do su to normal user

/etc/pam.d/system_auth

passwd required pam_passwdqc.so

When even root tries to set password for normal user, even root is forced

to set complex password for normal user.

/etc/pam.d/vsftpd

1. auth required pam_listfile.so item=user sense=deny file=/etc/vsftpd/suma

onerr=succedd

Create file suma under /etc/vsftpd, and add some users to the file /etc/vsftpd/suma

So tht users are not allowed to login thru ftp.Default file is /etc/vsftpd/ftpusersSo any users put into tht file are denied to ftp.

SYSLOG

1. To Configure Syslogserver

a. vim /etc/syscongig/syslog

sysLOGD_OPTIONS = "r"

b. service syslog restart

c. tailf /var/log/message

2. To redirect all clients log message to file called remote

vim /etc/syslog.conf

***.* /var/log/remote**

Clients to redirect there logs to syslogserver(192.168.0.45)

--> vim /etc/syslog.conf

***.* @192.168.0.45**

--> service syslog restart

. rotate

daily

1. **TYPES OF IPTABLES**
2. **2.**

1. **FILTER**
2. **NAT**
3. **MANGLE**

-> Types of Filter

1. **Forward**

2. Input

3. Output

-> Types of NAT

1. Pre-routing

2. Post-routing

3. Output

-> Types of Mangle

1. Forward

2. INPUT

3. OUTPUT

4. Pre-routing

5. Post-routing

2. TYPES OF TARGET

1. Accept

2. Drop

3. Log

4. Reject

--------->

1.-t -> table

2.-s source

3.-d destination

4.-i interface

5.--d -> service's port num

6.-i interface

7.-j target

---------->

1.-A Append

2.-F Flush

3.-R Replace

4.-D Delete

5.-L List

6.-I Insert

pkg   : iptables

daemon : iptables

file   : /etc/sysconfig/iptables

lokkit ->Enable firewall

1. To List all iptables

iptables -L

2. To Flush all iptables

iptables -F

3. To List only Input iptables

iptables -t filter -L INPUT

4. No System can access any services in server using tcp connection .

iptables -t filter -A INPUT -s 0.0.0.0/0 -j REJECT

5. To Control access to ping for single IP to stop pinging(ICMP)

a> REJECT ->Reject pinging with ACK

iptables -t filter -A INPUT -s 192.168.0.20 -p icmp -j REJECT

iptables -F

b> DROP ->Drop pinging with no ACK

iptables -t filter -A INPUT -s 192.168.0.20 -p icmp -j DROP

iptables -F

c> LOG ->Allow pinging + Log messages

iptables -t filter -A INPUT -s 192.168.0.20 -p icmp -j LOG

tailf /var/log/messages

iptables -F

d> ACCEPT ->Allow pinging

iptables -t filter -A INPUT -s 192.168.0.20 -p icmp -j ACCEPT

6. To Control access to ping for Entire N/W to stop ping(ICMP)

iptables -t filter -A INPUT -s 192.168.0.0/0 -p icmp -j REJECT

7. To Deny everybody to access other than this network

iptables -t filter -A INPUT -s! 192.168.0.0/0 -p icmp -j REJECT

8. To Number the iptables

iptables -t filter -L INPUT --line-numbers -n

9. To Delete individual iptable

iptables -t filter -D INPUT 2

10. To Replace an iptables with other iptables

iptables -t filter -R INPUT 2 -s 192.168.0.53 -p icmp -j REJECT

11. To Insert iptables B/W 2 iptables

iptables -t filter -I INPUT 2 -s 192.168.0.53 -p icmp -j REJECT

12. To Control access to ssh

iptables -t filter -A INPUT -s 192.168.0.20 -p tcp --dport 22 -j REJECT

13. To Control access to ssh

iptables -t filter -A INPUT -s 192.168.0.20 -p tcp --dport 21 -j REJECT

14. To Control access to pop3

iptables -t filter -A INPUT -s 192.168.0.20 -p tcp --dport 110 -j REJECT

15. To Control access to imaps

iptables -t filter -A INPUT -s 192.168.0.20 -p tcp --dport 993 -j REJECT

OUTPUT EXAMPLES

1.TO Stop our server to ping to any machine

iptables -t filter -A OUTPUT -s 192.168.0.28 -p icmp -j DROP

2.TO Stop our server to ssh to any machine

iptables -t filter -A OUTPUT -s 192.168.0.28 -p tcp --dport 22 -j DROP

3.TO Stop our server to ftp to any machine

iptables -t filter -A OUTPUT -s 192.168.0.28 -p tcp --dport 21 -j DROP

CONFIGURE ROUTER TO DO FORWARDING EXAMPLES

1. Set 2 diff ips of 2 diff n/w in router/server

2. vim /etc/sysctl.conf

3. sysctl -p

-> To stop Pinging b/w diff n/w.Thou they pass thru routers

iptables - filter -A FORWARD -s 10.0.0.20 -d 192.168.0.20 -p icmp -j REJECT

-> To stop ssh b/w diff n/w.

iptables - filter -A FORWARD -s 10.0.0.20 -d 192.168.0.20 -p tcp --dport 22 -j REJECT

-> To stop ftp b/w diff n/w.

iptables - filter -A FORWARD -s 10.0.0.20 -d 192.168.0.20 -p tcp --dport 21 -j REJECT

POSTROUTING OR SNAT

Anybody from public ip wants to connect to private ip.
But they cant connect using public ip, So use routers
private ip.

1. iptables -t nat -A POSTROUTING -j SNAT --to-source 10.0.0.1
2. From 192.168.0.20
 # ssh 10.0.0.20
w

3. iptables -t nat -L POSTROUTING
4. iptables -t nat -F POSTROUTING

PREROUTING OR DNAT

Anybody from private wants to connect to public ip.
But they cant connect using private ip, So use routers
public ip.

1. iptables -t nat -A PREROUTING -p tcp --dport 22 -j SNAT

--to-source 10.0.0.1

2. From 10.0.0.20

 # ssh 10.0.0.10

 # ifconfig

3. iptables -t nat -L PREROUTING

4. iptables -t nat -F PREROUTING

FIREWALL

iptables -t filter -A INPUT -s 192.168.0.100 --icmp-type echo-request -j DROP

iptables -t filter -A INPUT -s 192.168.0.100 -icmp-type echo-request -j DROP

iptables -t filter -A INPUT -s 192.168.0.100 -p tcmp -icmp-type echo-request -j DROP

iptables -t filter -A INPUT -s 192.168.0.100 -p icmp -icmp-type echo-request -j DROP

iptables -t filter -A INPUT -s 192.168.0.100 -p icmp --icmp-type echo-request -j DROP

iptables -L

iptables -t filter -A INPUT -s 192.168.0.100 -p icmp -j REJEC

iptables -F

ifconfig

iptables -t filter -A INPUT -s 192.168.0.71 -j DROP

iptables -F

pkill ssh

iptables -t filter -A INPUT -s 192.168.0.71 -p icmp -j DROP

iptables -F

iptables -t filter -A INPUT -s 192.168.0.71 -p icmp -j REJECT

iptables -t filter -A INPUT -s 192.168.0.71 -p icmp -j REJECT --reject-with icmp-host-unreachable

iptables -F

iptables -t filter -A INPUT -s 192.168.0.71 -p icmp -j REJECT --reject-with icmp-net-unreachable

iptables -F

iptables -t filter -A INPUT -s 192.168.0.71 -p icmp -j REJECT --reject-with icmp-host-unreachable

iptables -F

iptables -F

iptables -t filter -A INPUT -m mac --mac-source 00:21:91:8E:33:F0 -j DROP

ping 192.168.0.71

iptables -F

iptables -t filter -A INPUT -m mac --mac-source 00:21:91:8E:20:BB -j DROP

iptables -F

iptables -t filter -A INPUT -m pkttype --pkt-type broadcast   -j DROP

iptables -F

iptables -t filter -A INPUT -m pkttype --pkt-type unicast   -j DROP

iptables -F

iptables -F

iptables -F -t nat

iptables -t filter -A INPUT -m length --length 1000   -j DROP

iptables -t filter -A INPUT -m length --length 1000   -j DROP

iptables -F -t nat

iptables -F

iptables -t filter -A INPUT -m length --length 1000:2000   -j DROP

iptables -F

iptables -t filter -A OUTPUT   -m owner --uid-owner user1 -j DROP

iptables -t filter -A OUTPUT   -m owner --owner user1 -j DROP

man iptables

iptables -t filter -A OUTPUT -p tcp -m owner --owner user1 -j DROP

iptables -t filter -A OUTPUT -p tcp -m owner --uid-owner user1 -j DROP

iptables -t filter -A OUTPUT -p tcp -m owner --owner 501 -j DROP

iptables -t filter -A OUTPUT -p tcp -m owner --uid-owner 501 -j DROP

iptables -F

iptables -t filter -A OUTPUT -p tcp -m owner --uid-owner 501 -j DROP

useradd user1

useradd user2

iptables -F

iptables -t filter -A OUTPUT -p tcp -m owner --uid-owner user1 -j DROP

passwd user1

passwd user2

iptables -F

iptables -t filter -A OUTPUT -p tcp -m owner --uid-owner user1 -j DROP

iptables -t filter -A FORWARD -p tcp --tcp-flags SYN SYN   -j TCPMSS --clamp-mss-to-pmtu

iptables -t filter -A FORWARD -p tcp   -j TCPMSS --clamp-mss-to-pmtu

iptables -F

iptables -t mangle -A FORWARD -s 192.168.0.71 -m ttl --ttl 24 -j

iptables -t mangle -A FORWARD -s 192.168.0.71 -m ttl --ttl 24 -j   DROP

iptables -F

iptables -t mangle -A INPUT -s 192.168.0.71 -m ttl --ttl 24 -j DROP

iptables -F

iptables -t mangle -A INPUT -s 192.168.0.71 -m tos --tos 0x4 -j DROP

iptables -F

iptables -F

iptables -F -t mangle

iptables -t mangle -A INPUT -s 192.168.0.71 -j MARK --set-mark 0x2

iptables -t mangle -A INPUT -m mark --mark -j DROP

iptables -t mangle -A INPUT -m mark --mark 0x2 -j DROP

iptables -F -t mangle

iptables -L -t mangle

iptables -t mangle -A INPUT -s 192.168.0.71 -j DROP

iptables -t filter -A INPUT -s 192.168.0.71 -j ACCEPT

iptables -F -t mangle

iptables -F

iptables -t mangle -A INPUT -s 192.168.0.71 -j ACCEPT

iptables -t filter -A INPUT -s 192.168.0.71 -j DROP

pkill ssh

iptables -F

iptables -F -t mangle

vi /etc/ssh/sshd_config

service sshd restart

service sshd restart

iptables -t mangle -A INPUT -s 192.168.0.71 -p tcp --dport 22 -j REDIRECT --to-port 122

iptables -t mangle -A PREROUTING -s 192.168.0.71 -p tcp --dport 22 -j REDIRECT --to-port 122

iptables -t nat -A PREROUTING -s 192.168.0.71 -p tcp --dport 22 -j REDIRECT --to-port 122

iptables -t nat -A PREROUTING -s 192.168.0.71 -p tcp --dport 22 -j REDIRECT --to-port 122

iptables -F -t mangle

iptables -F -t nat

vi /etc/syslog.conf

service syslog restart

iptables -t filter -A 192.168.0.71 -j LOG --log-level info --log-prefix " My Packet : "

iptables -t filter -A INPUT 192.168.0.71 -j LOG --log-level info --log-prefix " My Packet : "

iptables -t filter -A INPUT 192.168.0.71 -j LOG --log-level info --log-prefix " My Packet : "

iptables -t filter -A INPUT -s 192.168.0.71 -j LOG --log-level info --log-prefix " My Packet : "

iptables -F

iptables -t filter -A INPUT -s 192.168.0.71 -m limit --log-limit 10/m -j LOG --log-level info --log-prefix " My Packet : "

iptables -t filter -A INPUT -s 192.168.0.71 -m limit --limit-rate 10/m -j LOG --log-level info --log-prefix " My Packet : "

iptables -t filter -A INPUT -s 192.168.0.71 -m limit --limit 10/m -j LOG --log-level info --log-prefix " My Packet : "

iptables -F

iptables -t filter -A INPUT -s 192.168.0.71 -m limit --limit 10/m --limit-burst 1 -j LOG --log-level info --log-prefix " My Packet : "

iptables -F

iptables -

iptables -t filter -A INPUT -s 192.168.0.71 -m limit --limit 10/m --limit-burst 1 -j LOG --log-level info --log-prefix " My Packet : "

iptables -L

iptables -L

iptables -L -v

iptables -Z

iptables -L -v

iptables -L -v

iptables -Z

iptables -L -v

iptables -F

iptables -L

iptables -L

iptables -N llc

iptables -L

iptables -A 192.168.0.71 -j llc

iptables -L

iptables -A 192.168.0.71 -j llc

iptables -t llc -A 192.168.0.71 -j DROP

iptables -A llc 192.168.0.71 -j DROP

iptables -A llc -s 192.168.0.71 -j DROP

iptables -L

iptables -A filter -s 192.168.0.0/24 -j llc

iptables -t filter -A INPUT -s 192.168.0.0/24 -j llc

iptables -L

iptables -F

iptables -L

iptables -X llc

iptables -L