**- Linux essential**

commands

clear = to clear screen ( clear previous text on commands from

your screen and take you to 1st row. )

# clear

**how to create a file**

command is = touch

how to use =

# touch filename.txt

eg = # touch 1.txt

this command will create a empty file.

**how to create a file and write content in it**

command is = cat > filename.txt

eg = # cat > 2.txt

after this your courser will come down and you are ready to

type text in that file and after completing you need to save

and exit

to save and exit = ctrl+d

**how to edit content of a any file**

we have different editors eg , gedit , nano , vi - editor

gedit = this mode is used in gui ( graphical mode only)

nano = this mode is used in text mode of linux and gui mode also

vi editor = this mode is used in text mode and gui mode also.

**how to use gedit :**

gedit filename.txt

eg =

# gedit 2.txt

this will open file in gui mode for editing.

**how to view content of any file**

cat filename.txt

eg =

# cat 2.txt

to view content of present dir

# ls

= list content of present dir

# ls -l

= this command is used for long listing of content of file

and folder with details ( date/time size in bytes , permissions etc)

ls -lh

( h = human readable )

it will show size of file in Kb, Mb, Gb

**how to create a folder**

mkdir = make dir

eg =

# mkdir f1

( f1 is folder name )

how to remove a file

rm filename

eg =

# rm 1.txt

how to remove a dir (folder)

rmdir dirname

eg =

# rmdir f2

( folder must be empty )

**how to use seq commnads**

=======================

seq (give option for numbers of lines to be printed in file) > filename.txt

eg:

# seq 1000 > 1thousand.txt

eg:

# seq 1000000 > 1million.txt

* if we want to view the content of file we use cat command

but using this command we will see last page of file.

cat 1thousand.txt | ( using this pipeline command will combine commands )

eg : cat 1thousand.txt | more

using "more" command will show output of file page wise and we can go to next

page but we can't go to previous page. how to go to next page just press "spacebar"

* if we want to go back to previous page and previous line we will use "less"

eg : cat 1thusand.txt | less

( using help command or

man (manual command to see more info about commands)

eg : man less ( this command will give output of less command in details )

by default seq will start number from 1 to 1000000

but if we want to start number from 10 to 100 how to use this command

eg :

# seq 10 100 > 1hundred.txt

if we want seq to be in steps

eg :

# seq 5 10 50 > 50.txt

5 = start numner

10 = step number

50 = end number

if we want to view 10 lines of file

but default using "head" commands will give output of 1st 10 lines of a file

if we want to view last 10 lines of a file

then using "tail" command will give output of last 10 lines of a file

eg :

# head 1thousand.txt

eg :

#tail 1thousand.txt

**how to get output of two or more file in a single file**

using cat command and >> ( combine and give output )

in file 11.txt 22.txt ( text of 11.txt is a b c d , text of

22.txt is 1 2 3 4

now you want to get and combine these file to a 33.txt file

= cat 11.txt 22.txt >> 33.txt

content of 11.txt will come first and then text of 22.txt and so on

stat ( show the status of file when last created/accessed/modified)

# stat 10thousand.txt

# uname : to know which OS you are using

# uname -a : details of Linux version ( eg: el5-ent edn linux 5)

rpm : redhat package management

( for installing any application )

# rpm -qa | grep httpd

( qa = query all )

( to search for httpd service is installed or not )

# ps : show current process

# ps -ef ( to see all the process )

# ps -aux ( to see detailed information of process )

# uptime ( shows system uptime )

free : ( check memory utilization )

# free -m

**Disk utilization**

df

# df -h ( human readable )

# top ( to view real time usagae of cpu and process )

this command will show process ID, eg PID 2637 )

( check pid and then kill ( kill means end task )

# kill 1212

go to /proc and then run below command to know more details

# cat cpuinfo

( this command will show information about cpu )

# kill PID

eg kill 3456

( this command will stop running that process like end task in task manager of windows )

**tar**

The tar command was originally developed for archiving data to tape drives. However, it's commonly used today for collecting a series of files, especially from a directory. For example, the following command backs up the information from the /home directory in the home.tar.gz file:

**# tar -czvf home.tar.gz /home**

This is one of the few commands that does not require a dash in front of the switch. This particular command creates (c) an archive, compresses (z) it, in verbose (v) mode, with the filename (f) that follows. Alternatively, you can extract (x) from that file with the following command:

**# tar -xzvf home.tar.gz /home**

The compression specified (z) is associated with the gzip command; if you wanted to use bzip2 compression, substitute the j switch.

File name Folder name

**# tar -xjvf home.tar.bz2 /home**

**how to compress a file**

File compression = to save disk space

in Linux we have 3 types of compression method

* gzip
* bzip2
* zip

eg: seq 1000000 > 1million.txt

ls -ltrh

l = long listing

t = sort time

r = reverse order

h = human readable format

to compress above file we will use gzip

# gzip -c 1m.txt > 1m.txt.gz

check man gzip

for -c .....(-c = is used to leave the original file as it is.)

% of compression

# gzip -l 1milion.txt.gz

**how to unzip**

zcat

to view to content of compress/zip file

# zcat 1million.txt.gz ( this command will dump o/p to screen)

-----------------------------------------

bzip2 = it is a advance utility then gzip ( improves the compression ratio )

# bzip2 -c 1million.txt > 1million.txt.bz2

**to decompress**

# bzip2 -d -c 1milliion.txt.bz2 > 1million.txt

check man bzip2

-d ......

to view the content of bzip2 file

# bzcat 1million.txt.bz2

( if the file is too big then you can use less or more command together with the above command )

---------------------------------------------

tar = use to bundles many files into a single file.( tar command can be used for backup data )

we will now bundle two files 1m.txt and 2m.txt

1m.txt = 6.6mb 2m.txt=15mb total=21.6mb

# tar -cvf 1m2m.txt.tar 1m.txt 2m.txt

output file looks like

milithousand.txt.tar ( this file contains 2 files inside )

creating tar and bzip file

# tar -cjvf 1m2m.txt.bz2 1m.txt 2m.txt

creating tar and gzip file

# tar -czvf 1m2m.txt.gz 1m.txt 2m.txt

# tar -xjvf milithousand.txt.bz2

---------------------------------------

for zip # zip -9 1m.txt.zip 1m.txt

for unzip

# unzip -x 1m.txt.zip

**how to use copy command**

cp =copy , cp <source path> <destination path>

eg : # cp /root/1.txt /root/Desktop/1.txt

**how to use move ( cut ) command**

by using this command original location of file is changed from one location to another location.

mv = move , mv <source path> <destination path>

but if you want to copy and paste any file or folder to another location

eg : s1 user want's to copy 1.txt file to s2 user on his Desktop

he must have permission to do so.

**File/folder permission**

======================

we have 10 bits permission settings including file/folder/link bits

- - - - - - - - - - file name is eg 1.txt

a b c d e f g h i j

- r w x r - - r - - for this file 1.txt we have full permission to owner

read permission to Group and read permission to everyone or world readable.

bit no. use for

a it indicates file/folder/link

d = folder

- = file

l = link ( shortcut )

b Read ( r) it is used for Read permission to Owner/user

c Write (w) it is used for Write/Edit permission to Owner

d Execute (x) it is used for execute permission to Owner/user

e Read ( r) it is used for Read permission to Group

f Write (w) it is used for Write/Edit permission to Group

g Execute (x) it is used for execute permission to Group

h Read ( r) it is used for Read permission to Everyone Or World readable

i Write (w) it is used for Write/Edit permission to Everyone Or World readable

j Execute (x) it is used for execute permission to Everyone Or World readable

how to assign permission to any file or folder

read ( r ) = 4

Write ( w ) = 2

execute ( x ) = 1

U G O

eg : chmod 7 7 7 1.txt

full permission to Owner/Group/Everyone

**7 = r 4 + w 2 + x 1**

output - r w x r w x r w x 1.txt

U G O

chmod 6 4 0 1.txt

read/write permission to Owner/user

only read permission to Group

no permission to Everyone

output - r w - r - - - - - 1.txt

chmod 000 1.txt

output - - - - - - - - - - 1.txt

**how to give permission using another method**

chmod u+rw 1.txt

( using this command indicates that u = user , + = add permission , rw = Read write )

**to remove permission**

chmod u-rw 1.txt

( using this command indicates the u = user , - = remove permission, rw = read write)

**User management**

**How to create users**

useradd

# useradd (user name)

# useradd tina

cat /etc/login.defs

**How to Give password to that user**

passwd

# passwd (user name for which you want to set password)

# passwd tina

**UID ( user ID)**

by default 1st created by you will get UID 500 (in rhel5/6) uid 1000 RHEL7

User with UID lessthan < 500 (rhel5/6) are system users, in rhel7 <1000

you can create max of 60000 users

**Users info is stored in**

# cat /etc/passwd

1 2 3 4 5 6 7

student:x:501:501 :/home /student/

1= username

2= password

3= User ID (UID)

4= GID ( group ID)

5= comments

6= Home dir

7= shell

# cat /etc/shadow(**this command will show UID/GID etc...)**

`Create User's

user1

user2

user3

user4

user default information are stored in

# cat /etc/login.defs

**Managing group**

whenever you create a new user, a group by his name is also create

which is called as Primary Group:

Primary Group : every user has a primary group as his user name

( eg: user1(username) user1(groupname)

Suppliment group are created for file/folder

**how to create group ????**

# groupadd sales

# gpasswd sales

( to set password to group)

to check all groups available:

# cat /etc/group

-------------------------------------------------

Adding users to user2 group

usermod -G sales username

# usermod -G sales user1

# usermod -G user1 user2

how to check user1 is added to user2 group

# cat /etc/group

-----------------------------------------------

log-in as user2

Create myfile.txt

# chmod 740 myfile.txt

ls -lh

see the permission usr/bin/passwd

===================================

log-in as root

=============

set uid

Ability to execute a program as owner

log-in as root

# chmod 4755 myfile.txt

4 = is used before permission to set uid bit

**Set GID**

---------- ???? yes

Ability to enchance permission to a directory

# chmod 2775 /sales

2= is used before permission to set GID bit

**Sticky bit**

using this users can't delete other user file in the directory

eg /tmp directory

# chmod 3777 /accounts

3 = is used before permission to set sticky bit

( 4 3 2 USG (UDHAM SINGH) U=4 S=3 G=2 )

**Task:**

1. Create user's

USER NAME

user11

user12

user13

user14

2. Set password to users

for all users give password = pass123\*

3. Create following folder on /root/Desktop by name k1

# mkdir /root/Desktop/k1

note down 10 bits permission settings = drwxr-xr-x

5. open this k1 folder

# cd /root/Desktop/k1

create file by name f1.txt

# touch f1.txt

note down 10 bits permission settings of f1 = -rw-r--r--

6. Create folders on /root by name k2 and k3

# mkdir /root/k2 ( 10bit permission = drwxr-xr-x

# mkdir /root/k3 ( 10bit permission = drwxr-xr-x

go to k2

# cd /root/k2

here create file f2.txt

# touch f2.re txt ( 10 bit permission = -rw-r--r--

7. Set uid to k1 and f1.txt ( check you position by using pwd )

# chmod 4744 /root/Desktop/k1

( 10bit = drwsr--r-- )

# chmod 4744 /root/Desktop/k1/f1.txt

( 10bit = -rwsr--r-- )

8. Set Gid bit for k2

# chmod 2744 /root/k2

( 10bit = drwxr-sr-- )

9. Set Sticky bit for k1

# chmod 3744 /root/Desktop/k1

( 10bit = drwxr-sr-t )

log off from root and log-in as other user ( check permisson for /root/Desktop )

**SUID stands for Set User ID.**

SUID allows applications to run by normal user

with privileges of root user.

That means in case I have an application (eg. ping) whose owner is 'root' and it has its SUID bit set, then when I run this application as a normal user, that application would still run as root.

By default the SUID will be applied on ping so that the normal users will also can ping to other

systems.

**SUID commands**

USG

uid 4

sid 3

gid 2

**To remove SUID**

[root@comp1 ~]# chmod 0755 /bin/ping

**To add SUID**

[root@comp1 ~]# chmod 4755 /bin/ping

==============================================

SGID is used for group inheritance, when

SGID is applied to a directory, all sub directories

& files created by any user in that particular

directory would be owned by the specified

group, regardless of user..s group.

SGID

eg: root is creating one dir by name /test1

and he is going to create one file inside this dir

file1.txt

ls -lh

-rw-r--r-- root root

when we assign gid to dir /test1 and give

full permission to user's

chmod 777 /test1

now will 2777 /test1

now log-on with user1

and he can create file inside /test1 dir

file2.txt

-rwxr--r-- user1 user1

user1 root

===========================================

=====================================

Sticky bit is used to restrict ..others.. from

removing the files/dir..s. When applied only

owner of that particular file/dir can delete them.

Root / Superuser can also remove them, even if

it is applied.

Sticky Bit

chmod 3777 /test1

drwxrwsrwt

**How to mount cd/dvd-rom**

- All drives are listed in /dev directory

# cd /

# ls -l

- Open and check /dev directory for drives

# cd dev

if you are at /root or at another location then type

# cd /dev

Check which all drives are mounted

# mount

- Before mounting cd/dvd create directory by name

# mkdir /dvd

- type following command to mount cd/dvd-rom

# mount

# mount -l

# mount -t iso9660 /dev/cdrom /dvd

check dvd or cdrom is mounted or not

# mount

how to unmount dvd rom

# umount /dvd

then check by using mount command.

troubleshooting in mounting

When you get error like

" no medium found "

then check power cable and data cable of cd/dvd rom is connected properly.

then check if you have inserted cd or dvd in cdrom tray.

**File System**

**===========**

IDE Handling identified by hdxy

hd = IDE harddisk ( PATA )

x = IDE Device attached denoted by letter

( eg : a,b,c .... )

y = Partition number of a IDE harddisk denoted by number

( eg : 1,2,3 .... )

eg: /dev/hda1

/dev = this is a device directory in which all device are listed ( hdd/fdd/usb etc...)

In the same way SCSI handling is also done just you need to replace h to s

eg: /dev/sda1

here sd = SCSI drive/device

Every device is identified by a virtual file system

steps to follow after attaching any new harddisk

( when you attach a new harddisk it is known as RAW disk )

1st create partition

2nd format partition

3rd mount partition

**to check free disk space of all harddisk attached use**

# df -h

**to check details info of all hdd connected use**

# fdisk -l

**Post-Installation tools**

Fdisk

when will use fdisk -l will get details info of new hdd attached

eg : /dev/sdb or /dev/sdc

**to create partition use**

# fdisk /dev/sdb

press m for help on list of commands you can use

n = for creating new partition

p = for creating primary partition

+2G = for size of 2Gb

after you created any partition you must use

w = to write information to harddisk

to format a partition is known as making the file system

# mke2fs /dev/sdb1 ( mke2fs /dev/sdb1 this is your primary partition )

for ext2 ( 2nd extension file system )

# mkfs.ext3 -j /dev/sdb2

for ext3 ( extension to 2nd extension file system )

# mke2fs -t ext4 -j /dev/sdb5

for ext4 file system ( more security and bigger file/partition size )

in order to use that partition that file system should be mounted

create dir

# mkdir /data1 ( we are going to mount 1st pri part )

# mkdir /data2

# mkdir /data3

use mount command to mount sdb1 to /data1

# mount /dev/sdb1 /data1

after successfully mounting will find lost+found dir inside that dir (mount point)

the devices listed in the /etc/fstab file will be automatically mounted when the

system starts (fstab= file system table)

you must edit /etc/fstab

# vi /etc/fstab

To unmount the file system

[# umount <directory name>

=========================================================

**Converting from Ext2 to Ext3 filesystem**

# tune2fs -j <partition>

**To assign label**

# e2label <partition> <label\_name>

**To view existing label**

# e2label <partition>

**To view existing label**

# e2label <partition>

To see mounted partition with Label

# mount -l

==============================================================

**Task:**

1. Attach 2 harddisk of IDE ( 10 GB each )

2. Create partition's of

1st harddisk primary partition 2 nos of 4GB each and logical partition

of 2GB.

2nd harddisk primary partition 3 nos of 2 GB each and rest space leave

without partition.

3. Format any one partition of 1st harddisk to ext2 and

format any one partition of 2nd harddisk to ext3

4. Mount 1st partition of 1st harddisk on /hdd1

and 1st partition of 2nd harddisk on /hdd2

5. show all mounted hdd details including partitions.

6. Mount cdrom and insert linux bootable (DVD) and show

content of Server folder inside of DVD

**Quota Management**

================

Quota : Limit the disk usage space to the user and group

v.imp b4 setting quota limit you have to enable quota in fstab

**now what is fstab**

fstab = file system table

**how to view fstab**

# cat /etc/fstab

**you need to edit this fstab to do this**

# vi /etc/fstab

**you have to add user quota to a quota after default by giving ,**

eg : defaults,usrquota

eg : defaults,grpquota

( no need to restart only remount file system )

**If not able to remount then run reboot**

( restart you pc for entry to listed in /etc/fstab file )

check using mount command

**To enable the new partition with usrquota and grpquota**

# mount -o usrquota, grpquota <part\_name> <mnt\_pt>

eg: # mount -o usrquota /dev/sdd1 /disk1

**To generate the database files inside the quota partition**

# quotacheck -mcuvg <quota\_mnt\_pt>

eg: # quotacheck -mcuvg /disk1

check once database is created

there is a file aquota.user file is created inside mount point

**To turn on the quota**

#quotaon <quota\_mnt\_pt>

eg: # quotaon /disk1

**To apply the quotas for the users on quota enabled partition**

# edquota -u <user name>

eg: # edquota -u user1

**( To apply the quotas for a group on quota enabled partition**

# edquota -g <group name>

eg: # edquota -g sales )

**to check mounted partition**

**# mount**

**To check how much disk space is used by all user's**

# repquota -a

limits are used as soft limits and hard limits (for dir mounted from 2nd sdb1=2Gb )

mount point is on root /disk1

max size of /disk1 is= 2Gb )

eg: hard limit = 20mb and 75% of is my soft limit = 15mb

soft limit is set so that the user is having a grace period of

7 days by default, he can continue till his grace period or till his hard limit.

to see the disk usage of all user's

once database is created

there is a file aquota.user file is created inside mount point

**Task:**

set soft limit to user1 = 20mb

set hard limit to user1 = 30mb

8bit = 1 byte

1024bytes = 1kb

1024 kb = 1mb

1024kb x 30 = ( blocks makes 30 mb size )

1 block = 1024bytes = 1KB

Limits are set in blocks

block size is soft= 20 hard = 30

log-on with user and try keeping/creating files )

------------------------------------

ext2 = 2nd extention file system

ext3 = extention to 2nd extention file system

( For RHEL6 )

After BIOS What

=================

- System is power ON

- POST ( Power ON Self Test )

- OS Search on HDD

- MBR ( Master Boot Loader )

Stage 1 = 512MB

Stage 1.5 = File system environment

Stage 2 = Initialize

1st process start in init (initialize) process

Init reads from /etc/inittab

As per default run level mentioned in inittab OS will start

all process ( program ) are started in /etc/initd

depending upon the runlevel selected.

**Run level** =========

0= halt

1= single user mode

2= multiuser without netwroking

3= full multiuser

4= not used

5= x11 ( Gui ) mode

6= restart

**You can write S or Single**

**to check run level**

# runlevel

you will get output something like this

N 5

previous Present

/etc/rc.d/rc.sysinit

this scrip that are common for all the runlevel

when you edit inittab with vi editior

you can change default runlevel

after runlevel information below in line

id:5:initdefault

replace/change 5 to required runlevel number

eg: id:3:initdefault

save and exit

**# tty(to check on which terminal you are working.)**

Linux provide 6 terminal

to go to another terminal keep pressing together below keys till you terminal

change.

Ctrl+Alt+F1

to come out from other terminal mode keep pressing together below keys till

you are out from that terminal

Ctrl+Alt+F7

**(RHEL 6)**

How to go to GUI mode from command/shell mode

temp switching of runlevel 3 to runlevel 5

GRUB = GRand Unified Boot loader

# startx

(for windows we change boot sequence and time by using cmd : bcdedit (boot configuration data editor)

open and grub-md5-cryptedit

# vi /boot/grub/grub.conf

change time to 10 sec

change and save/exit

reboot your pc

# reboot

**when red screen comes**

press 'e' and then select 2nd line and press 'e'

type and the end single ( runlevel 1 )

in this mode you can change your root password

# passwd root

it will ask you enter new password and confirm password

( both password must be same )

# reboot

and try login with new password for root.

now if you want to give password to grub.conf

( in order to stop changing root password and OS boot sequence also )

**Create encrupted password**

# grub-md5-crypt

type password and it will convert it back to crypt and copy this password

and open in another terminal window

# vi /boot/grub/grub.conf

paste above title

password --md5 (paste crypt password)

save/exit

# reboot

and then try to enter using 'e' it will ask you for password.

**Why should a Linux boot loader have password protection?**

The following are the primary reasons for password protecting a Linux boot loader:

1. Preventing Access to Single User Mode – If an attacker can boot into single user mode, he becomes the root user.

2. Preventing Access to the GRUB Console – If the machine uses GRUB as its boot loader, an attacker can use the GRUB editor interface to change its configuration or to gather information using the cat command.

3. Preventing Access to Non-Secure Operating Systems – If it is a dual-boot system, an attacker can select at boot time an operating system, such as DOS, which ignores access controls and file permissions.

**Password protecting GRUB2**

Follow the steps below to password protect GRUB2 in RHEL 7.

1. Remove –unrestricted from the main CLASS= declaration in /etc/grub.d/10\_linux file.

This can be done by using sed to replace the

# sed -i "/^CLASS=/s/ --unrestricted//" /etc/grub.d/10\_linux

use grub2-setpassword to set a password for the root user :

# grub2-setpassword(This creates a file /boot/grub2/user.cfg if not already present, which contains the hashed GRUB bootloader password. This utility only supports configurations where there is a single root user.)

Example /boot/grub2/user.cfg file :

# cat /boot/grub2/user.cfg

Recreate the grub config with grub2-mkconfig :

# grub2-mkconfig -o /boot/grub2/grub.cfg

Reboot the server and verify.

# shutdown -r now

Note that all defined grub menu entries will now require entering user & password each time at boot; henceforth, the system will not boot any kernel without direct user intervention from the console. When prompted for user, enter “root”. When prompted for password, enter whatever was passed to the grub2-setpassword command :

**Remove password protection**

To remove the password protection we can add the –unrestricted text in the main CLASS= declaration in /etc/grub.d/10\_linux file again. Another way is to remove the /boot/grub2/user.cfg file which stores the hashed GRUB bootloader password.

-----------------------

to set restrict user to edit grub menu :

to set restrict user to edit grub menu :

**Create password :**

use command:

# grub2-mkpasswd-pbkdf2

copy encrypted password and

now edit

#vi /etc/grub2.cfg

above menu entry type following:

set superusers="root"

export superusers

password\_pbkdf2 root <encrypted password paste here>

# reboot (and try edit grub2 menu )

Task- For changing runlevel and grub file

1. Check present runlevel and note down runlevel.

2. Find out and view the file used for changing runlevel

Note down the path of file used for runlevel.

3. Change runlevel to 3 ( full multiuser - in text mode )

4. Restart system and system must start in runlevel 3

5. Go to gui mode

6. And change runlevel to 5 and reboot pc.

7. View grub file and note down path of grub file.

8. Edit title and time to boot

9. Create crypt password for grub file. ( password must be pass123\*\* )

10. Give password to grub file.

11. Reboot pc and try to edit boot sequence or change hdd config settings.

12. Pc must ask for password ( enter password = pass123\*\* )

13. Remove password for grub file.

14. Do some changes in grub file and make os non operational

eg change to root (hd0,0) to (hd1,1)

15. Repaire it by using linux rescue mode.

**Configuring IP Address**

**Assigning IP address -Temporary**

[root@comp1 ~]# ifconfig eth0 <ipaddr> netmask <netmask>

Eg: # ifconfig eth0 192.168.80.20 netmask 255.255.255.0

**Assigning IP address -Permanent**

[root@comp1 ~]# netconfig

[root@comp1 ~]# service network restart

Restart network service after you change ip.

**Assigning Virtual IP address**

[root@comp1 ~]# netconfig--device <interface>:<n>

[root@comp1 ~]# service network restart

**Check your IP address assigned on all network adapter's**

# ifconfig

Configuring IP Address

Graphical tool for assigning IP address

[root@comp1 ~]# neat

OR

[root@comp1 ~]# system-config-network

Graphical tool for assigning IP address

**Configuring Network Interface**

To check whether NIC is detected

[root@comp1 ~]# ethtool <interface>

**Enabling Network Interface**

[root@comp1 ~]# ifup<interface>

**Disabling Network Interface**

[root@comp1 ~]# ifdown <interface>

Locations For IP address

[root@comp1 ~]# cd /etc/sysconfig/network-scripts

[root@comp1 ~]# vi ifcfg-eth0

For virtual IP address

[root@comp1 ~]# vi ifcfg-eth0:1

For Primary DNS

` **RHCE-253**

**Networking in linux**

Assigning IP address to pc

1) To check what is the present IP address of your PC use command

# ifconfig

2) To set or assign IP address to PC

# system-config-network(This command will allow you to assign ip in gui mode)

3) To start/stop networking services

# service network start / stop / restart

4) To check all services status in runlevel's

# chkconfig --list

5) To Stop and start any service in runlevel's

# chkconfig --level 5 network on

6) To check hostname of your pc

# hostname

**HOW TO TAKE REMOTE DESKTOP OF WIN XP/SER 2003 PC**

1. CHECK REMOTE DESKTOP ENABLE ON CLIENT OR WIN XP/SER 2003 PC

2. ON LINUX PC INSTALL "rdesktop ...... .rpm" this rpm is available

in Linux installation DVD in "Server" folder. (rhel 5)

rhel 6 "package" dir

3. On Terminal shell type

# rdesktop -u <username> IP

( username is the user of Windows xp / server 2003 pc )

eg:

# rdesktop -u administrator 192.168.100.21

( by default this will open in full screen mode )

( but we can define the window size by using -g HXL

Eg:

# rdesktop -g 550X450 -u administrator 192.168.100.21

-----------------------------------------------------------------------------------

**How to check rdesktop is installed or not**

# rpm -qa | grep rdesktop

q = querry

a = all

grep to search service for rdesktop or dhcp or any application is installed or not

to check

# rpm -q rdesktop

if you got some reply indicating rdesktop information/rpm

then no need to install rpm for this rdesktop.

or if nothing comes then you need to install rpm for rdesktop

Now to install any package ( rpm = redhat package management )

go to DVD drive and go to Server/Package director and run below command

# rpm -ivh rdesktop\*\*\*\*.rpm

info about -ivh

check man page of rpm. ( man rpm )

to check how to un-install any package from your pc

# rpm -ev rdesktop

--------------------------------------------------------------

you can install packages by using "yum" also

CHECK man page of "yum" ( man yum )

YELLOWDOG UPDATE MANAGER

**NFS ( Network file system )**

- Designed by IBM

- Applicable to Linux & Unix

- Transport access to remote file system

- User RPC for communication

- Installed by default

**Task:**

**Check NFS (package) is installed**

# rpm -qa | grep nfs

**to check for NFS utilities are installed, check the list of files**

# rpm -ql nfs-utilS

/nfs1 must be available on remote system

/etc/exports ( this file has details of all dir )

**edit above file and give path /nfs1 ip: 192.168.10.xx**

( rw = read / write )

On the server that is exporting NFS file systems, the nfs service must be running.

View the status of the NFS daemon with the following command:

# /sbin/service nfs status

# /sbin/service nfs on

chkconfig --level 53 nfs on

if you need to use internet port no.80 http://www.google.com

by default http port no.80 is closed

# chkconfig --level 5 httpd on (to make this port open during startup)

Start the NFS daemon with the following command:

/sbin/service nfs start

Stop the NFS daemon with the following command:

/sbin/service nfs stop

To start the nfs service at boot time, use the command:

/sbin/chkconfig --level 345 nfs on

**NFS Sharing**

===========

Need to run following commands on server side

if you need to share folder /nfs1 on linux network

# mkdir /nfs1

keep some data inside this dir ( 1.txt , 222 )

**open exports file and**

if you open to view content of exports file

this file is blank.....

you need to type following inside file by any file editor

vi = Visual Interface

vim = Visual interface IMproved

# vi /etc/exports

**type following in this file**

/nfs1 192.168.20.24(rw,sync)

IP is ip address of another client to whom we need to give access to

/nfs1 folder created on server side.

or access from any pc in that network use following cmd

/nfs1 192.168.10.0/24(rw,sync)

or

/nfs1 \*(rw,sync)

Or

/nfs1 \*.proactiveitservicess.com(rw,sync)

save and exit

**start following services**

nfs on server pc

# service nfs status

# systemctl status nfs

# systemctl start nfs

# service nfs start ( or reload )

( keep the firewall off )

and then go to client pc and mount using follwing command

before using mount command create /nfs1 folder

# mkdir /hodata

# mount -t nfs 192.168.20.20:/nfs1 /hodata

( IP address of server on which /nfs1 is shared )

once nfs is configured on main server

then you need to disable firewall and selinux

# system-config-firewall

select disable .

for selinux linux ( security enhanced linux)

# lokkit --nfs selinux=disabled

and then go to other pc

# mount -t nfs 192.168.20.20:/nfs1 /hodata

-----------------------------------------------

**check dhcp is installed or not :**

# rpm -qa | grep dhcp

you will find dhcp client is available, you need to install dhcp server

go to cd-rom drive .

first mount the dvd .

# cd /dvd

# cd Packages

# rpm -ivh dhcp\*.rpm

( \* indicates version eg 3.0.2.1 ..)

# cd /etc ( this is for rhel5 )

cd /etc/dhcpd.conf

# cd /etc/dhcp/dhcpd.conf (rhel 6)

# vi dhcpd.conf

NOTE : note down the path

/usr/share/doc/dhcp\*/dhcpd.conf.sample

come out from this file and

go to this path and copy this file "dhcpd.conf.sample"

and then paste it /etc/dhcp/dhcpd.conf (by the name "dhcpd.conf")

# cd /usr/share/doc/dhcp\*(tab)

# cp dhcpd.conf.sample /etc/dhcp/dhcpd.conf

( it will ask you to overwrite type "yes" )

Then edit dhcpd.conf to set range etc....

# vi /etc/dhcpd.conf

Check status of dhcp server running or not

# service dhcpd status

Restart or reload this dhcpd service

# service dhcpd restart

**SWAP**

**What is Swap Space?**

Swap space in Linux is used when the amount of physical memory (RAM) is full. If the system needs

more memory resources and the physical memory is full, inactive pages in memory are moved to the

swap space.

The size of your swap space should be equal to twice your computer's RAM, or 32 MB, whichever

amount is larger, but no more than 2048 MB (or 2 GB).

**Adding Swap Space**

Sometimes it is necessary to add more swap space after installation. For example, you may upgrade

the amount of RAM in your system from 64 MB to 128 MB, but there is only 128 MB of swap

space.

To add a swap partition (assuming /dev/hdb1 is the swap partition you want to add):

before creating swap partition

# cat /proc/swaps

**At a shell prompt as root, type the command**

# parted /dev/hdb, where /dev/hdb is the device name for the hard drive with free space.

**At the (parted) prompt, type**

# print

to view the existing partitions and the amount of free space. The start and end values are in megabytes. Determine how much free space is on the hard drive and how much you want to allocate for a new swap partition.

At the (parted) prompt, type

# mkpartfs primary linux-swap 1 128 (rhel6)

# mkpart primary linux-swap 1 128 (rhel7)

**Check using print command again**

# print

Now that you have the swap partition, use the command mkswap to setup the swap partition. At a shell prompt as root, type the following:

# mkswap /dev/hdb1

**To enable the swap partition immediately, type the following command:**

# swapon /dev/hdb1

**To enable it at boot time, edit /etc/fstab to include:**

# /dev/hdb2 swap swap defaults 0 0

The next time the system boots, it will enable the new swap partition.

**After adding the new swap partition and enabling it, make sure it is enabled by viewing the output of the command**

# cat /proc/swaps

**To add a swap file:**

1. Determine the size of the new swap file and multiple by 1024 to determine the block size. For example, the block size of a 64 MB swap file is 65536.

2. At a shell prompt as root, type the following command with count being equal to the desired block size:

# dd if=/dev/zero of=/swapfile bs=1024 count=65536

3. Setup the swap file with the command:

# mkswap /swapfile

4. To enable the swap file immediately but not automatically at boot time:

# swapon /swapfile

5. To enable it at boot time, edit /etc/fstab to include:

# /swapfile swap swap defaults 0 0

The next time the system boots, it will enable the new swap file.

6. After adding the new swap file and enabling it, make sure it is enabled by viewing the outputof the command

# cat /proc/swaps or free.

------------------------------------------------------------------

**Removing Swap Space**

At a shell prompt as root, execute the following command to make sure the swap partition is disabled (where /dev/hdb1 is the swap partition):

# swapoff /dev/hdb1

Remove its entry from /etc/fstab.

Remove the partition using parted or fdisk. Only parted will be discussed. To remove the partition with parted:

**At a shell prompt as root, type the command**

# parted /dev/hdb

where /dev/hdb is the device name for the hard drive with the swap space to be removed.

**At the (parted) prompt, type**

# print

to view the existing partitions and determine the minor number of the swap partition you wish to delete.

**At the (parted) prompt, type**

# rm MINOR, where MINOR is the minor number of the partition

you want to remove.

Type

# quit

to exit parted.

**To remove a swap file:**

1. At a shell prompt as root, execute the following command to disable the swap file (where/swapfile is the swap file):

# swapoff /swapfile

2. Remove its entry from /etc/fstab.

3. Remove the actual file:

# rm /swapfile

-----------------------------------------------

**LVM**

Logical volume management

Features

Ability to create volume sets

LVM masks the physical technology

( IDE, SCSI, SATA, PATA types of drives )

IDE (slot) = Integrated Drive/Device Electronics

SCSI = Small Computer System Interface

( these type of harddisk can be attached / detached when power is on, these are also called as hotswapable drives)

1= sda1 10GB, 2= sdb2 5gb , 3= sdc3 4gb

15gb

SATA = Serial Adv Tech Attachment

PATA = Parallel Adv Tech Attachment

Ability to increase and decrease size of the volume

PV = Physical Volume

VG = Volume Group

LV = Logical Volume

attach sdb, sdc

- Create partition using Fdisk

1= sdc1 10GB, 2= sdd1 5gb ( don't formate it)

Create physical volumes

# pvcreate /dev/sdc1 /dev/sdd1 ....

Assign physical volumes to volume groups

# vgcreate vg1 /dev/sdc1 /dev/sdd1 ....

Create logical volumes from volume groups

# lvcreate -L 14G -n data vg1

-L = Size from volume group , -n = name of logical volume)

format logical volume

# mkfs.ext3 -j /dev/vg1/data

and then mount it ( dir /lvm1 )

# mount /dev/vg1/data /lvm1

Check volume group informtion

# vgdisplay ( display volume group )

**Cron and AT**

==========

cron = it is schedular

cron wakes up every minutes to see that is there any job configured then it is executed

= default installed

= rules based

= rules based on time

a = minutes ( 0 - 59 )

b = hour ( 0 -23 )

c = day of the month ( 1 - 31 )

d = month ( 1- 12 )

e = day of week ( 0 - 7 ) 0 and 7 is Sunday

( note: you need to create script by file name test.sh

and to run that script file type ./ before file name

# cat > ping1.sh

how to create script

create file by using cat command and type following content

ping -c 5 10.10.10.27 >> /root/Desktop/cron1.txt )

save and edit

ctrl+d

try run script

# ./ping1.sh

now you need to set this file execute bit permission

# chmod 777 ping1.sh

( colour of this file will turn to green )

for your reference below is example of position of \*

when the file will run :

a b c d e

\* \* \* \* \* /root/Desktop/test.sh

5 8 27 10 1 /root/test.sh

1 2 \* \* \* /root/test.sh

file entry will be done in /etc/crontab in order mentioned above.

-----------------------------------

AT command

using this command will execute job once or time defined.

eg: create file by touch command at1.txt

# at now +10 minute

when you press enter you will come to prompt >

> ping -c 5 10.10.10.27 > at1.txt

for save and quit task

ctrl+d

this will save job and ready for execute at pre-defined time.

**HTTP(Apache Server)**

1st check the ip address of the computer

#ifconfig

then check the httpd status

#systemctl status httpd.service

check by (

#chkconfig --list

if its not on then

#chkconfig --level httpd 35 on

then start/restart the service

#service httpd start/restart)

then go to /var/www/html folder

#cd var

in that www

#cd www

in what you will find html folder

#cd html

create index.html file and type

inside (Welcome to ProActive Web Server......)

after this restart the service

#systemctl restart httpd.service

go to url and type the ip address of the computer

it will open the page which we have created.

-------------------------------------------------------------

How to check who has monitored your web ?

go to /var/www in that by

ls command

goto

#cd usage

#ls

run command

# webalizer

after this restart the service

#service httpd restart

then go to internet explorer in web browser type

localhost/usage

it will show you all the usage summary ,monthly stats,daily stats,hourly stats etc.

**Kickstart**

Starting the Installation with a Kickstart File

No matter where you choose to put the kickstart file, you can boot with a USB key or a specialized installation CD. You can put the kickstart file on the boot media. Just copy the configuration file as ks.cfg from wherever you've saved your kickstart configuration file.

If you're booting from a CD, and the kickstart file is also on that CD, enter the following command at the boot: prompt. This assumes the ks.cfg file is in the top level directory of the CD.

boot: linux ks=cdrom:/ks.cfg

If you're booting from the Red Hat installation CD, you can still refer to a kickstart configuration file on a USB key or hard disk, respectively, possibly with one of the following commands:

boot: linux ks=hd:sdb:/ks.cfg

boot: linux ks=hd:hda2:/home/mj/ks.cfg

This assumes the kickstart configuration file is called ks.cfg and is located on the USB key detected as /dev/sdb, or the second partition of the first IDE drive in the /home/mj directory.

You don't need to get a kickstart file from a DHCP server. To boot from a specific NFS or HTTP server on the network, say with an IP address of 192.168.17.18, from the /kicks/ks.cfg file, type one of the following commands:

boot: linux ks=nfs:192.168.17.18:/kicks/ks.cfg

boot: linux ks=http:192.168.17.18:/kicks/ks.cfg

Or if it's on the first floppy drive, try

boot: linux ks=hd:fd0:/ks.cfg

**Kickstart Installation**

To install kickstart you need to install following rpm's

to go "Server or Package" dir

and run following rpm's

rpm -ivh system-config-kickstart-2.8.6.2-1.el6.noarch.rpm

( you need to install following dependencies )

rpm -ivh pykickstart-1.74.1-1.el6.noarch.rpm

rpm -ivh system-config-language-1.3.4-5.el6.noarch.rpm

rpm -ivh anaconda-yum-plugins-1.0-5.1.el6.noarch.rpm

rpm -ivh createrepo-0.9.8-4.el6.noarch.rpm

rpm -ivh deltarpm-3.5-0.5.20090913git.el6.x86\_64.rpm

rpm -ivh python-deltarpm-3.5-0.5.20090913git.el6.x86\_64.rpm

rpm -ivh createrepo-0.9.8-4.el6.noarch.rpm

rpm -ivh isomd5sum-1.0.6-1.el6.x86\_64.rpm

rpm -ivh makebootfat-1.4-10.el6.x86\_64.rpm

rpm -ivh pyparted-3.4-2.el6.x86\_64.rpm

rpm -ivh python-cryptsetup-0.0.11-1.el6.x86\_64.rpm

rpm -ivh python-nss-0.8-3.el6.x86\_64.rpm

rpm -ivh python-pyblock-0.48-1.el6.x86\_64.rpm

rpm -ivh squashfs-tools-4.0-3.el6.x86\_64.rpm

rpm -ivh syslinux-3.86-1.1.el6.x86\_64.rpm

rpm -ivh perl-Crypt-PasswdMD5-1.3-6.el6.noarch.rpm

rpm -ivh syslinux-3.86-1.1.el6.x86\_64.rpm

rpm -ivh tigervnc-server-1.0.90-0.10.20100115svn3945.el6.x86\_64.rpm

rpm -ivh anaconda-13.21.82-1.el6.x86\_64.rpm

rpm -ivh anaconda-yum-plugins-1.0-5.1.el6.noarch.rpm

once all above rpm's is installed then run

rpm -ivh system-config-kickstart-2.8.6.2-1.el6.noarch.rpm

then go to "applications" and "system tools" there you will find kickstart

select it and you will see graphical wizard to create kickstart file.

Fill/select all informations and save as this file by name "ks.cfg"

use this file for unattended installation of OS.

Starting the Installation with a Kickstart File

No matter where you choose to put the kickstart file, you can boot with a USB key or a specialized installation CD. You can put the kickstart file on the boot media. Just copy the configuration file as ks.cfg from wherever you've saved your kickstart configuration file.

If you're booting from a CD, and the kickstart file is also on that CD, enter the following command at the boot: prompt. This assumes the ks.cfg file is in the top level directory of the CD.

boot: linux ks=cdrom:/ks.cfg

If you're booting from the Red Hat installation CD, you can still refer to a kickstart configuration file on a USB key or hard disk, respectively, possibly with one of the following commands:

boot: linux ks=hd:sdb1:/ks.cfg

boot: linux ks=hd:hda2:/home/mj/ks.cfg

This assumes the kickstart configuration file is called ks.cfg and is located on the USB key detected as /dev/sdb, or the second partition of the first IDE drive in the /home/mj directory.

You don't need to get a kickstart file from a DHCP server. To boot from a specific NFS or HTTP server on the network, say with an IP address of 192.168.17.18, from the /kicks/ks.cfg file, type one of the following commands:

boot: linux ks=nfs:192.168.17.18:/nfs1/ks.cfg

boot: linux ks=http:192.168.17.18:/kicks/ks.cfg

Or if it's on the first floppy drive, try

boot: linux ks=hd:fd0:/ks.cfg

**SAMBA And SWAT**

Microsoft's CIFS was built on the Server Message Block (SMB) protocol. SMB was developed in the 1980s by IBM, Microsoft, and Intel as a way to share files and printers over a network.

SMB network communication over a Microsoft-based network is also known as NetBIOS over TCP/IP. Through the collective works of Andrew Tridgell and the Samba team, Linux systems provide transparent and reliable SMB support over TCP/IP via a package known as Samba. You can do four basic things with Samba:

Share a Linux directory tree with Windows and Linux/Unix computers

Share a Windows directory with Linux/Unix computers

Share a Linux printer with Windows and Linux/Unix computers

Share a Windows printer with Linux/Unix computers

These are the four Samba RPM packages that you need:

The samba RPM package includes the basic SMB server software for sharing files and printers.

The samba-client RPM package provides the utilities needed to connect to shares from Microsoft computers.

The system-config-samba package installs the Red Hat Samba Server Configuration utility.

The samba-common RPM package contains common Samba configuration files.

It's easy to start the Samba Server Configuration tool. You can do so from a command line interface in the GUI with the system-config-samba command.

Configuring Samba to Start

Configuring Samba to start when Linux boots is a straightforward process. Configure Samba to start in at least runlevels 3 and 5, and then start the Samba server. The following two commands activate Samba in runlevels 2, 3, 4, and 5, and start the service:

# chkconfig smb on

# service smb start

The command that tests smb.conf for syntax errors is

# testparm

The command that forces the Samba service to reread the configuration file-without having to disconnect your Microsoft users or restarting the service-is

# service smb reload

You can see if Samba is already running with the service smb status command. If it's stopped, you can start it with the service smb start command. If it's running, you can make Samba reread your configuration file with the following command:

# service smb reload

**very secure FTP (vsFTP)**

Check your system to determine whether the vsFTP server is installed. The easiest way is with the following command:

# rpm -q vsftpd

If it isn't already installed, use the techniques discussed earlier to install the vsFTP RPM package.

Activate the vsFTP server with the service vsftpd start command.

Make sure this server is automatically activated the next time you boot Linux with the following command:

# chkconfig vsftpd on

Log into the vsFTP server as a regular user. You should preferably log in from a remote computer.

Once you're logged in, run the cd ? command twice (remember the space between the command and the two dots). Explore the local directory. You should see a danger here, as this is the root directory for the FTP server computer.

Close the FTP client session.

If you're concerned about the security issues, deactivate the vsFTP server. The best way to secure your system from FTP-based attacks is to uninstall the vsFTP server.

RHEL includes the vsFTP server. The default configuration allows anonymous and real user access.

You can customize vsFTP through the /etc/vsftpd/vsftpd.conf configuration file. It also uses authentication files in the /etc/vsftpd/ directory: ftpusers, user\_list, and chroot\_list.

The default directive in /etc/vsftpd/vsftpd.conf that you should disable if you don't want users logging into their accounts through the vsFTP server is local\_enable=YES

The directive you should add if you want to keep regular users from getting to the top-level root directory (/) on your computer is chroot\_user=YES

Based on the default RHEL 5 configuration, both ftpusers and user\_list in the /etc/vsftpd directory include a list of users not allowed to log into the vsFTP server

The additional directives you need to add to the default vsFTP configuration file to allow security using PAM and TCP wrappers are pam\_service\_name=vsftpd and tcp\_wrappers=YES

--------------------------------

Access to User's:

Test local user access and update SELinux configuration

# getsebool -a | grep ftp'

- dumps FTP-related SELinux booleans

# setsebool -P ftp\_home\_dir 1

Note: RHEL6 enables SELinux in 'enforcing' mode, requiring a slight change to the booleans to permit VSFTPD or any FTPD daemon to transition user into their: $HOME directory.

**RAID**

Ability to increase availability and reliability

Multiple disks grouped together into "arrays"

to provide better performance, redundancy or

both.

mdadm - provides the administration interface to software RAID.

Many "RAID Levels" supported, including

RAID O, 1 and 5.

Spare disks add extra redundancy

RAID devices are named, /dev/md0,

/dev/md1, /dev/md2, /dev/md3 and so on.

Steps for creating RAID Volume

- Create partition = fdisk

- Create Raid volume = mdadm

# mdadm --create /dev/md0 --level=1 --raid-devices=2 \ /dev/hdb1 /dev/hdb2

Create and define RAID devices using mdadm

# mdadm -C /dev/md0 -a yes -l 1 -n 2 -x 1 /dev/hda1 /dev/hdb1 /dev/hdc1

- Create file system ( format using mke2fs )

Format each RAID device with a filesystem

# mkfs.ext3 -j /dev/md0

- mount file system = mount

- update /etc/fstab = vi /etc/fstab

mdadm allows you to check the status of your RAID devices

# mdadm --detail /dev/md0

Simulating disk failures

# mdadm /dev/md0 -f /dev/hda1

Recovering from a software RAID disk failure

replace the failed hard drive and power on

reconstruct partitions on the replacement drive

# mdadm /dev/md0 -a /dev/hda1

**Configuring iptables**

The philosophy behind iptables is based on "chains." These are sets of rules applied to each network packet. Each rule does two things: it specifies the conditions a packet must meet to match the rule, and it specifies the action if the packet matches.

The iptables command uses the following basic format:

iptables -t tabletype <action direction> <packet pattern> -j <what to do>

Now analyze this command, step by step. First is the -t tabletype switch. There are two basic tabletype options for iptables:

filter: Sets a rule for filtering packets.

nat : Configures Network Address Translation, also known as masquerading, discussed later in this chapter.

The default is filter; if you don't specify a -t tabletype, the iptables command assumes that you're trying to affect a filtering rule.

Next is the <action direction>. There are four basic actions associated with iptables rules:

-A (--append) Appends a rule to the end of a chain.

-D (--delete) Deletes a rule from a chain. Specify the rule by the number or the packet pattern.

-L (--list) Lists the currently configured rules in the chain.

-F (--flush) Flushes all of the rules in the current iptables chain.

If you're appending to (-A) or deleting from (-D) a chain, you'll want to apply it to network data traveling in one of three directions:

INPUT All incoming packets are checked against the rules in this chain.

OUTPUT All outgoing packets are checked against the rules in this chain.

FORWARD All packets being sent to another computer are checked against the rules in this chain.

Next, you need to configure a <packet pattern>. All iptables firewalls check every packet against this pattern. The simplest pattern is by IP address:

-s ip\_address All packets are checked for a specific source IP address.

-d ip\_address All packets are checked for a specific destination IP address.

Packet patterns can be more complex. In TCP/IP, packets are transported using the TCP, UDP, or ICMP protocol. You can specify the protocol with the -p switch, followed by the destination port (--dport). For example, the -p tcp --dport 80 extension affects users outside your network who are trying to use an HTTP connection.

Once the iptables command finds a packet pattern match, it needs to know what to do with that packet, which leads to the last part of the command, -j <what to do>. There are three basic options:

DROP The packet is dropped. No message is sent to the requesting computer.

REJECT The packet is dropped. An error message is sent to the requesting computer.

ACCEPT The packet is allowed to proceed as specified with the -A action: INPUT, OUTPUT, or FORWARD.

Take a look at some examples of how you can use iptables commands to configure a firewall. The first step is always to see what is currently configured, with the following command:

# iptables -L

list the default rules

to flush all rules of iptables

# iptables -F

If iptables is properly configured, it should return chain rules in three different categories: INPUT, FORWARD, and OUTPUT.

The following command defines a rule that rejects all traffic from the 192.168.75.0 subnet, and it sends a "destination unreachable" error message back to any client that tried to connect:

# iptables -A INPUT -s 192.168.0.0/24 -j REJECT

to reject all computer in this network 192.168.0 network

how to delete this rule

# iptables -D INPUT -s 192.168.0.0/24 -j REJECT

To block a perticular computer with ip 192.168.0.51 , drop packets from this pc.

# iptables -A INPUT -s 192.168.0.51 -p icmp -j DROP

to delete the above rule

use -D in place of -A

This rule stops users from the computer with an IP address of 192.168.25.200 from "pinging" your system (remember that the ping command uses the ICMP protocol):

# iptables -A INPUT -s 192.168.20.5 -p icmp -j DROP

The following command guards against TCP SYN attacks from outside your network. Assume that your network IP address is 192.168.1.0. The exclamation point (!) inverts the meaning; in this case, the command applies to all IP addresses except those with a 192.168.1.0 network address (and a 255.255.255.0 subnet mask).

# iptables -D INPUT -s !192.168.20.0/24 -p tcp -j DROP

.

Then, if you want to delete the rule related to the ping command in this list, use the following command:

# iptables -D INPUT -s 192.168.25.200 -p icmp -j DROP

The default rule for INPUT, OUTPUT, and FORWARD is to ACCEPT all packets. One way to stop packet forwarding is to add the following rule:

# iptables -A FORWARD -j DROP

**DNS Configuration**

**To Check IP**

[root@www Desktop]# ifconfig

eth0 Link encap:Ethernet HWaddr 00:0C:29:84:6D:8C

inet addr:10.90.12.1 Bcast:10.90.12.255 Mask:255.255.255.0

inet6 addr: fe80::20c:29ff:fe84:6d8c/64 Scope:Link

UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1

RX packets:6624 errors:0 dropped:0 overruns:0 frame:0

TX packets:1474 errors:0 dropped:0 overruns:0 carrier:0

collisions:0 txqueuelen:1000

RX bytes:442710 (432.3 KiB) TX bytes:1901220 (1.8 MiB)

Interrupt:19 Base address:0x2000

eth1 Link encap:Ethernet HWaddr 00:0C:29:84:6D:96

inet addr:10.23.151.66 Bcast:10.23.159.255 Mask:255.255.224.0

inet6 addr: fe80::20c:29ff:fe84:6d96/64 Scope:Link

UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1

RX packets:13927 errors:0 dropped:0 overruns:0 frame:0

TX packets:7518 errors:0 dropped:0 overruns:0 carrier:0

collisions:0 txqueuelen:1000

RX bytes:9215651 (8.7 MiB) TX bytes:948169 (925.9 KiB)

Interrupt:19 Base address:0x2080

lo Link encap:Local Loopback

inet addr:127.0.0.1 Mask:255.0.0.0

inet6 addr: ::1/128 Scope:Host

UP LOOPBACK RUNNING MTU:16436 Metric:1

RX packets:8 errors:0 dropped:0 overruns:0 frame:0

TX packets:8 errors:0 dropped:0 overruns:0 carrier:0

collisions:0 txqueuelen:0

RX bytes:480 (480.0 b) TX bytes:480 (480.0 b)

**To Set DNS Server IP**

vi /etc/sysconfig/network-scripts/ifcfg-eth0

EVICE=eth0

HWADDR=00:0c:29:84:6d:8c

NM\_CONTROLLED=no

ONBOOT=yes

IPADDR=10.90.12.1

BOOTPROTO=none

NETMASK=255.255.255.0

DNS=10.90.12.1

TYPE=Ethernet

IPV6INIT=no

USERCTL=no

save :wq

**To Set Host Name**

[root@station Desktop]# vim /etc/sysconfig/network

NETWORKING=yes

HOSTNAME=station.example.com

save :wq

[root@station Desktop]# vim /etc/hosts

0.90.12.1 station.example.com station

save :wq

[root@station Desktop]# vim /etc/resolv.conf

search station.example.com

nameserver 10.90.12.1

save :wq

[root@station Desktop]# hostname station.example.com

[root@station Desktop]# hostname

station.example.com

**To Install Package**

[root@station Desktop]# yum install bind\*

Loaded plugins: fastestmirror, refresh-packagekit, security

Repository 'yum' is missing name in configuration, using id

Loading mirror speeds from cached hostfile

Setting up Install Process

Package 32:bind-utils-9.7.3-8.P3.el6.i686 already installed and latest version

Package 32:bind-libs-9.7.3-8.P3.el6.i686 already installed and latest version

Resolving Dependencies

--> Running transaction check

---> Package bind.i686 32:9.7.3-8.P3.el6 will be installed

---> Package bind-chroot.i686 32:9.7.3-8.P3.el6 will be installed

---> Package bind-dyndb-ldap.i686 0:0.2.0-7.el6 will be installed

--> Finished Dependency Resolution

**Dependencies Resolved**

================================================================================

Package Arch Version Repository Size

================================================================================

Installing:

bind i686 32:9.7.3-8.P3.el6 yum 3.9 M

bind-chroot i686 32:9.7.3-8.P3.el6 yum 67 k

bind-dyndb-ldap i686 0.2.0-7.el6 yum 49 k

**Transaction Summary**

================================================================================

Install 3 Package(s)

Total download size: 4.0 M

Installed size: 7.1 M

Is this ok [y/N]: y

Downloading Packages:

(1/3): bind-9.7.3-8.P3.el6.i686.rpm | 3.9 MB 00:00

(2/3): bind-chroot-9.7.3-8.P3.el6.i686.rpm | 67 kB 00:00

(3/3): bind-dyndb-ldap-0.2.0-7.el6.i686.rpm | 49 kB 00:00

--------------------------------------------------------------------------------

Total 20 MB/s | 4.0 MB 00:00

Running rpm\_check\_debug

Running Transaction Test

Transaction Test Succeeded

Running Transaction

Warning: RPMDB altered outside of yum.

Installing : 32:bind-9.7.3-8.P3.el6.i686 1/3

Installing : 32:bind-chroot-9.7.3-8.P3.el6.i686 2/3

Installing : bind-dyndb-ldap-0.2.0-7.el6.i686 3/3

Installed:

bind.i686 32:9.7.3-8.P3.el6 bind-chroot.i686 32:9.7.3-8.P3.el6

bind-dyndb-ldap.i686 0:0.2.0-7.el6

Complete!

[root@station Desktop]#

**To Copy named.conf file**

[root@station Desktop]# cp /etc/named.conf /var/named/chroot/etc/named.conf

**To Change directory**

cd /var/named/chroot/etc/

**To edit configuration file**

[root@station etc]#vim named.conf

options {

directory "/var/named";

};

zone "example.com" IN {

type master;

file "for.zone";

};

zone "12.90.10.in-addr.arpa" IN {

type master;

file "rev.zone";

};

save :wq

**To Change Group Name**

[root@station etc]# chgrp named named.conf

**To Copy File same Location**

[root@station etc]# cp /var/named/named.localhost /var/named/chroot/var/named/for.zone

[root@station etc]# cp /var/named/named.loopback /var/named/chroot/var/named/rev.zone

**To change directory**

[root@station etc]# cd /var/named/chroot/var/named/

**To edit configuration file**

[root@station named]# vim for.zone

$TTL 1D

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

0 ; serial

1D ; refresh

1H ; retry

1W ; expire

3H ) ; minimum

@ IN NS station.example.com.

station IN A 10.90.12.1

save :wq

**To edit configuration file**

[root@station named]# vim rev.zone

$TTL 1D

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

0 ; serial

1D ; refresh

1H ; retry

1W ; expire

3H ) ; minimum

@ IN NS station.example.com.

1 IN PTR station.example.com.

save :wq

**To Change Group Name**

[root@station named]# chgrp named for.zone

[root@station named]# chgrp named rev.zone

[root@station named]# ll

total 8

-rw-r-----. 1 root named 190 Jun 1 19:12 for.zone

-rw-r-----. 1 root named 196 Jun 1 19:15 rev.zone

[root@station named]#

To Restart Service & On

[root@station named]# service named restart

Stopping named: [ rajesh ]

Starting named: [ rajesh ]

[root@station named]# chkconfig named on

To Check Named Server

[root@station named]# dig 10.90.12.1

; <<>> DiG 9.7.3-P3-RedHat-9.7.3-8.P3.el6 <<>> 10.90.12.1

;; global options: +cmd

;; Got answer:

;; ->>HEADER<<- opcode: QUERY, status: NXDOMAIN, id: 23819

;; flags: qr rd ra; QUERY: 1, ANSWER: 0, AUTHORITY: 1, ADDITIONAL: 0

;; QUESTION SECTION:

;10.90.12.1. IN A

;; AUTHORITY SECTION:

. 10800 IN SOA a.root-servers.net. nstld.verisign-grs.com. 2012060501 1800 900 604800 86400

;; Query time: 193 msec

;; SERVER: 113.193.1.14#53(113.193.1.14)

;; WHEN: Fri Jun 1 19:17:27 2012

;; MSG SIZE rcvd: 103

[root@station named]# dig station.example.com

; <<>> DiG 9.7.3-P3-RedHat-9.7.3-8.P3.el6 <<>> station.example.com

;; global options: +cmd

;; Got answer:

;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 24133

;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 1, ADDITIONAL: 0

;; QUESTION SECTION:

;station.example.com. IN A

;; ANSWER SECTION:

station.example.com. 86400 IN A 10.90.12.1

;; AUTHORITY SECTION:

example.com. 86400 IN NS station.example.com.

;; Query time: 1 msec

;; SERVER: 10.90.12.1#53(10.90.12.1)

;; WHEN: Fri Jun 1 19:17:47 2012

;; MSG SIZE rcvd: 67

[root@station named]#

**BIND**

Install bind packages.

#rpm -ivh bind-\* rpm

#cd /var/named/chroot/etc

copy the file /usr/share/doc/bind-9.7.0/named.conf.default to /var/named/chroot/etc

#cp /usr/share/doc/bind-9.7.0/named.conf.default /var/named/chroot/etc/named.conf

#vi named.conf

edit following lines (Add your Ip address at 192.168.80.70)..

//

// named.conf

//

// Provided by Red Hat bind package to configure the ISC BIND named(8) DNS

// server as a caching only nameserver (as a localhost DNS resolver only).

//

//

See /usr/share/doc/bind\*/sample/ for example named configuration files.

//

options {

listen-on port 53 { 127.0.0.1; 192.168.80.70; };

listen-on-v6 port 53 { ::1; };

directory "/var/named";

dump-file "/var/named/data/cache\_dump.db";

statistics-file "/var/named/data/named\_stats.txt";

memstatistics-file "/var/named/data/named\_mem\_stats.txt";

allow-query { localhost; 192.168.80.0/24; };

recursion yes;

dnssec-enable yes;

dnssec-validation yes;

dnssec-lookaside auto;

/\* Path to ISC DLV key \*/

bindkeys-file "/etc/named.iscdlv.key";

};

logging {

channel default\_debug {

file "data/named.run";

severity dynamic;

};

};

zone "." IN {

type hint;

file "named.ca";

};

include "/etc/named.rfc1912.zones";

:wq

#chkconfig --level 35 named on

#service named restart

Now verify with following command

#dig -x 192.168.80.70

#ping server.proactive.com

**OpenOfice Installation**

To install Openoffice suite for RHEL 6 ,

just stick with the following steps…

Download setup file,

After your download completes,

Open Terminal and Execute the following…

# tar -xzf OOo\_3.3.0\_Linux\_x86\_install-rpm-wJRE\_en-US.tar.gz

Now Change your directory into OOo\_3.3.0\_Linux\_x86\_install-rpm-wJRE\_en-US (the folder which you extracted)

On Same Directory You will find Folder call RPMs,

now move to that folder using follwoing command.

#cd RPM We go,need to Install All RPMS to Install OpenOffice 3.2.

Execute Following Command

# rpm -ivh \*.rpm

If everything goes fine, it means OpenOffice gets installed.

But By performing above steps OpenOffice got installed, but you’ll not able to see OpenOffice Selection in Applications Menu. For this we have to Perform another few Steps.

In RPMs directory there is one more directory called “desktop-integrates“.

Now Lets Switch to that Directory and install Red Hat Related Desktop Integration RPM.

# rpm -ivh openoffice-desktop-redhat-xxx.rpm

Once Desktop Integration RPM get installed You’ll able to See Open Office Selection in Application Menu.

This is for 32Bit Linux Operating system.

**Analyzing And Storing Logs**

Many programs use a standard protocol to send messages to "rsyslogd"

Each message is described by a

'facility' ( the type of message it is) and

'severity' (how important it is)

# cat /etc/rsyslog.conf

( This file uses the facility and severity of a log message to determine where it should be sent)

Rotating logs

============

Logs are "rotated" to keep them from filling up the file system containing

when a log file is rotated, it is renamed with an extension indicating the date on which it was rotated eg

old file /var/log/message file may become /var/log/messages-20120421 (yyyymmdd)

once the old log file is rotated, a new log file is created and the service that writes to it is notified.

After a certain number of rotations (typically after four weeks) the old log file is discarded to conserve disk space.

A cron job runs the logrotate propram daily to see if any logs need to be rotated.

most log files are rotated weekly or when they reach a certain size.

rsyslog manual location: /usr/share/doc/rsyslog-\*/manual.html

check rsyslog.conf and logrotate man pages.

**Access Control List**

.. To configure different set of file permissions for

different users on a single resource (file/folder),

ACL..s are implemented.

.. ACL..s can be implemented only on ..ACL enabled

partitions..

.. ACL..s can be applied on

.. Users

.. Groups

root

/aclmount

Example

Quotation.txt

Hello

File Created by root

User : root : rw\_ , Group : root : r\_ \_ , Others : r\_ \_

Steps

Create a new partition

#fdisk /dev/sdc

Format the Partition

#mkfs.ext3 /dev/hda9

Create a new mount point

#mkdir /aclmount

Steps

Mount New partition with ACL ..option..

#mount -o acl /dev/sdb2 /aclmount

Create Users

#useradd usr1

#useradd usr2

#useradd usr3

Create Groups

# groupadd sales

Steps

Add some users to group

# gpasswd -M usr1, usr2 sales

Create files into the ACL enabled partition

#vi /aclmount/quotation.txt

The default permissions for the directory / file will be

..rw\_r\_\_r\_\_..

Applying ACL

ACL permissions to the directory for the user

#setfacl -m u:usr1:- /aclmount/quotation.txt

setfacl = set file access control list

m= modify option for acl to dir

ACL permissions to the directory for the group

#setfacl -m g:sales:rw /aclmount/quotation.txt

ACL

To list applied ACL applied on a File

#getfacl /aclmount/quotation.txt

now test acl by log-on with usr2 and usr1 and check by cat .

Remove ACL from a file

#setfacl -x u:usr1 /aclmount/quotation.txt

---------------------------------------------------------------------