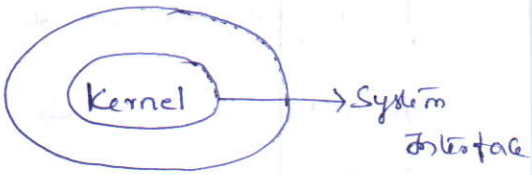
Index



Shell ⟶ user Interface

LINUX - Distributions

* Open Source Ubuntu
* Redhat
* feddoora
* Kali

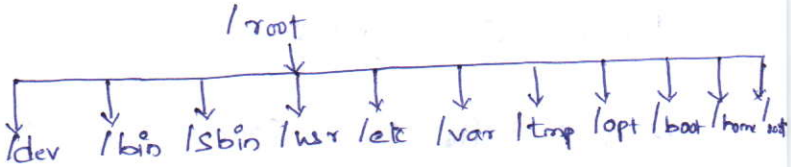
UNIX

* Solaries
* AIX
* Hp-UX

Advantages

* Open Source
* Security
* high Performance
* Multi tasking
* Scalability

Top level Hierarchy



* /bin: All normal Command info
* /Sbin: All admin user Command info
* /etc : Software Configuration files info
* /var : All log messages info
* /tmp : All temporary files info
* /opt : all 3rd Party packages info
* /boot : all booting files info
* /dev : all devices Content info
* /media: all removable media info
* /home: all default home directly for normal user
* /root : all default home directory for Root user.
* /user : all user info

user1 -> home /user1

Super user (admin) of

Power user -> full Previledge.

Reset VMs Commands

rht-vmctl → for resetting VMs

rht-vmctl reset Server (or) # rht-vmctl fullreset Server

rht-vmctl reset desktop (or) # rht-vmctl fullreset Server

rht-vmctl Start Server

rht-vmctl View Server

~ → home directory

→ root directory

whoami → display current user name

Hostname → display System name

pwd → display path of present working directory

BASIC COMMANDS

Touch : The touch command is the easiest way to create new, empty files. It is also used to change the timestamp (i.e dates and times of the most recent access and modified) on existing files and directories.

Syn:- # touch <file Name> To create file in pwd

touch /home/f1 under root folder

touch f1 f2 f3 f4

touch f1 f2 f3 f4

If you . dot infront of the file it will be hidden.

Cat :- (Concatinate file) The Cat Command can be used to join multiple files together and print the results on screen

It is used to create text files.

* To display content of the file

Cat <f1/N> // display the data

* for appending data

Cat >> <file Name>

ctrl +D for save file.

* # Mkdir <file name>

* To Create a directory

mkdir /root/d1

mkdir d1 d2 d3 /d4

mkdir dir{1...6}

* for hidden directories add '.' infront of directory

mkdir .d5

* To Create nested directories

mkdir -p /a/b/c/d

Change Directory (CD)

Desktop # cd /root/desktop /a/b/c/d/e

absolute Path

cd /a/b/c/d/e

relative Path

Umask Value

* It is used to provide security to file and directory.

The default umask value is 022 - Root user & normal user 002

read - 4 Execute - 1

Write - 2

Directory - (umask -755) → default Permission

file - (umask -644) → default Permission.

(d1)	user	group	others
directory	rwx	rwx	rwx
	7	7	7
	0	2	2
	7	5	5
	(rwx)	(r-x)	(r-x)

(f1)	user	group	others
file	rw-	rw-	rw-
	6	6	6
	0	2	2
	6	4	4
	(rw-)	(r--)	(r--)

Chmod : It is used to change the permission for directory / file. We have two modes in chmod

* Absolute mode (Numeric Mode)
* Symbolic Mode.

Note: root user umask value : 022 and normal user umask value : 002.

Absolute Mode:

$f_1 \rightarrow$ rw- r-- r-- $\rightarrow$ rwx r-x -wx (6 4 4 → 7 5 3)

Syn: # chmod <permission in numerical> <f/N>

chmod 753 f_1

Symbolic Mode:

$d_1 \rightarrow$ rwx r-x r-x $\rightarrow$ rw- rw- -w- (7 5 5 → 6 6 2)

Syn: # chmod u-x (nospace) g+w-x , o+w-rx d_1 <f/N>

+ : adding - : Subtracting

= : assigning.

chmod a=rwx d_1

Copy:-

Syn:- # cp <Source file Name> < Destination file Name>

cp f_1 /root/f_2

cp f_1 /root ⇒ /root/f_1

(If we didnot specify any file here name same file would be created in destination)

Move:- Mv is for cut and paste

Syn:- # mv <Source f/N> <Dest f/N>

mv f_1 f_2

remove: Syn # rm <f/N>

(or)

Sy # rm -f <f/N> (force Remove)

Copying directory

cp -r <Source dir> <dest dir>

* If destination directory is there Source dir copies the inside destination directory
* If destination directory is not there, new directory will be Created with the content of Souce file.

Moving directory

Syn:- # mv <Source d/N> <dest dir>

mv d_1 d_2

* If destination directory is the source directory moved into destination directory.

* If destination directory is not the source directory will be removed.

Remove directory

rm -r <d1/N>

forcely remove directory

rm -rf <d1/N>

ls - listing

ls → To display all files and directories in pwd

ls /root

for displaying listing all hidden files and all files and directories

ls -a

ls -a/

ls -a /root/

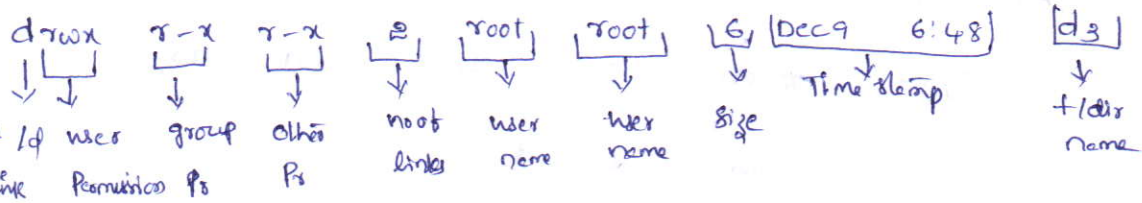
To field which is directory and files are

ls -p → The directory name end with /

* To get all the details of files and directories with Permission and owner info

ls -l (or) # ls

ls -l /



links (3)

* long list of particular file

```
#  ls -l  <f1/N>
```

```
#  ls - R          (To display the sub directories)
#  ls - t
#  ls - lt         (To display file and directories in desending order)
#  ls - ltr        (To display file and directories in asending order)
#  ls - ls         (desending order of size)
#  ls - lsr        (Asending order of size)
#  man ls
```

Inode Numbers

It is unique number used to identify the files. Inode number contain the Computer information of the files except the file name the file name is user name reference and inode is System Refer-ence.

```
#  ls -i  <f1/N>
```

Link : It is nothing but connect to a pointer ie Connection beetween files and directories

Two types of links ① Hard link ② Soft link

Hard link # ln f_1 f_2

* If you are Creating hard link between two files both will have same inode number
* If you update the source file the data will be updated in destination file and vice versa.
* The permission, size and timestamp is same for both files even be update the data in either on source or update
* If you delete the source file the data will be available in destination file.

disadvantages

* We can't Create Hard link between directories only for files.
* We can't identify whether link is Created or not.
* We can't Create hard link between the file System.

Soft link # ln -s <f/N source> <destination f/N>

* If you are Created Soft link files, both will have different inode numbers.
* If you update the source file data will update in destination file and vice versa.

* The size and stamp of the destination file is constant even though if you udate source and destination file the permission also different.
* If you delete source file the data will b is no more available in destination file because the destination file and vice versa is a just short cut of source file.
* [The size and stamp of the destination file is constant even though if you update source and destination file the permission also different.
* If you delete source file the data]

Advantages

* We can create soft link between two files and directories
* We can identify whether softlink is created or not
* We can create soft link between the file systems.

Vi Editor (Visual Editor)

We can edit and 3 different editions we have

1. Insert mode
2. Command line mode
3. Colon mode

Syn: vi <f/n>

To open a file in vi editor, if file is not there it will be created the new file automatically.

Esc + i	To insert mode
Esc + Shift + i	1st Position of Current line
Esc + A	last position of current line
Esc + R	To replace the entire file
Esc + x	To delete the Current Characters
Esc + nx	To delete the n number of Characters
Esc + dw	To delete the Current word
Esc + ndw	To delete the n number of words
Esc + dd	To delete the Current line
Esc + ndd	To delete the n number of lines
Esc + U	undo
Esc + yw	To Copy Current word
Esc + n yw	To Copy n number of Current words.
Esc + p	To paste
Esc + yy	To Copy Current line
Esc + nyy	To Copy n number of Current line
Esc + H	Go 1st position of 1st line of current page
Esc + G	Go to the 1st position of last line of current file
Esc + shift + ~	To Change lower Case to upper Case vice versa
Esc + /	To Search keyword -> Press get Enter is to get more options.
Esc + w	Next word of that position.
Esc + e	next word last position.

Colon mode: Esc + Shift + : Colon mode

: set nu → To display line numbers

: set nonu → To remove line numbers

: n → go to the nth line

: nd → To delete nth line

: 1,3d → To delete 1, 2, 3 lines

: 2 Co 6 → Copy line 2 to line 7

: 1,3 Co 4 → Copy lines 1, 2, 3 to line 5, 6, 7

: 2 move 5 → line 2 moved to 5

: 1,3 move 6 → to move lines 1, 2, 3 to 4 5 6

: w → Save

: w! → Save fource fully

: q → quit without saving

: q! → " " " fource fully

: wq! → Save and quit forcefully

Partitions and file Systems

Linux Kernel is assigning logical names to all the devices Connected to the System depending on the device class & manifacture

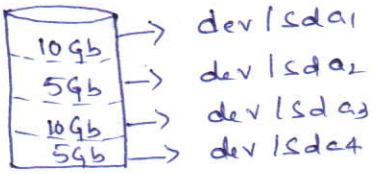
SCSI / SATA HDD → /dev/sda, sdb, ...

IDE HDD → /dev/hda, hdb ....

Virtual HDD → /dev/vda, vdb .....

SATA HARD DISKS

/dev/sda



Create Partitions

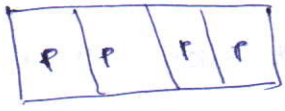
```
# fdisk -l          → list all available disks
# fdisk /dev/vdb
# m                 → To help
# n                 → Create a new partition
# P                 → Primary Partition
  : Partition No ←
  : 1st Sector ←
  : Last Sector ←   +2G
  : p → Print table
  : w → Save & exit
  : Part probe (up date changes in Kernel)
```

Delete Partition

```
# fdisk /dev/vdb
: p -> print table
: d -> delete Partition
: Enter Partition Number
: p
: w
# Part_probe
```

fdisk

* We Can Create 4 Primary Partitions



* We Can Create 3 primary partitions and 1 extended partitions. Under extended partition we Can Create 11 logical partitions.

* Total 15 partitions, we Can use only 14 and another one extended Partition Can't be used.

1. the main difference between fdisk and Parted is disk labeling
2. fdisk by default uses DOS labeling so that we can only Create max 4 Partitions
3. Parted uses gpt (guided partition table) labeling so that we Can Create 128 primary partitions.

Formatting :- It is process of adding file System. The difference file systems are available in Linux which are ext2, ext3 and ext4. xfs xfs.

Raw device :- The device which is not formatted that is not having file system is known as RAW Devices.

Block device :- The device which is formatted that is having file System known as Block device.

mkfs is a command used for adding file System

RAW device ⟶ Block device
format

mkfs.ext4 /dev/vdb1 (or) # mkfs -t ext4 /dev/vdb1

blkid /dev/vdb1

blkid This Command gives the block id for device after formatting with particular file system.

Mounting :- It is the process of attaching a block device to any directory under the root.

1. Creating directory # mkdir /sun

2. Mounting # mount /dev/vdb1 /sun
↓
Mount point

df -h → To checking mounted devices
(or)
df -hT → To check display fs & mounted devices

vi /etc/fstab

/dev/vdb1 /sun ext4 default 0 0

This is just temporary mounting to permanent add data to /etc/fstab

vi /etc/fstab

device Name	mount point	file system	mount option
/dev/vdb1	/sun	ext4	default

default Value	Node Value
0	0

fsck Pass : fsck is a simply a frontend to the various file system checkers (fsck.fstype) available under linux. fsck is searched for/in first, then in /etc/fs and /etc and the directories listed in the PATH

0 - No fsck (or) No errors

1 - Series of fsck (or) file system errors corrected

2 - parallel (or) system should be rebooted

4 - fls errors left un corrected

8 - operation error

Partition → formatting → etc/fstab → mount

1. Create partition { process for create & add file system to
2. Part probe Partitions step by step procedure }
3. format mkfs
4. Create dir
5. /etc/fstab
6. Mount # Mount <d/N> <mount point>

(or)

Mount <d/N> (or)

Mount <Mount point> (or)

Mount -a

UN-Mounting : It is process of detaching the device mount point / directory

Unmount is done usually two situations
1. deleting the partition 2. Maintenance activity

Syn: unmount <d/N> (or) <Mount Point>

* Remove entry from etc/fstab

vi /etc/fstab

Put # before line of fss+dd (To delete line)

* Delete the partition

fdisk /dev/vdb

: p → Print table
: d → delete the partition.
: w → Save & exit

partprobe

Deleting / unmounting busy file (Target is busy)

unmount /sun (Target is busy)

fuser -cu /sun

↳ list all the users & process accessing the directory

fuser -ck /sun

↳ Kill all the users & process consuming Particular files.

1024 Gb = 1 Tb
1024 Tb = 1 Pb
1024 Pb = 1 Hb
1024 Hb = 1 Zb

ext 3	ext 4	
32 Tb	1 Hb	← Max file System Size
2 Tb	16 Tb	← Max file Size
32000	64000	← No of Sub directories

File System Journaling

* fj is an advanced feature which has the capacity to recover the file System automatically in the event of read/write synchronization errors

* When we create a file system using mkfs command. It creates file system meta data like Super block & alternative Super block.

Super block:- The default location of Super block is 16-31 Sectors, It Contains the Complete information about the file System.

Alternative Super block:- It is duplicate Copy of Super block, there is no specific location for alternative Super block and Can be available in any part of the hard disk.

fsck (file System check)

The purpose of fsck is to bring the file System inconsistent (~~The purpose of fsck is to bring the file System~~) to consistent. fsck is needed to apply the file System. we can execute fsck commands in two modes.

1. Interactive mode (User inputs are taken)

```
# e2fsck  <devName>
# e2fsck  /dev/vdb1
```

2. Non-interactive mode

```
# e2fsck -y  <devName>
# e2fsck -y  /dev/vdb1
```

3. To check the status

```
# e2fsck -n  <devName>
# e2fsck -n  /dev/vdb1
```

To finding alternative Super blocks

```
# dumpe2fs  /dev/vdb1 | grep Super_block
```

for modifing alternative Superblock settings

```
# e2fsck -f -b  <alternative Super block ids>

# e2fsck -f -b  811679  /dev/vdb1
              ↓
        block number
      forcefully

# mount -a
```

Logical Volume Manager (LVM)

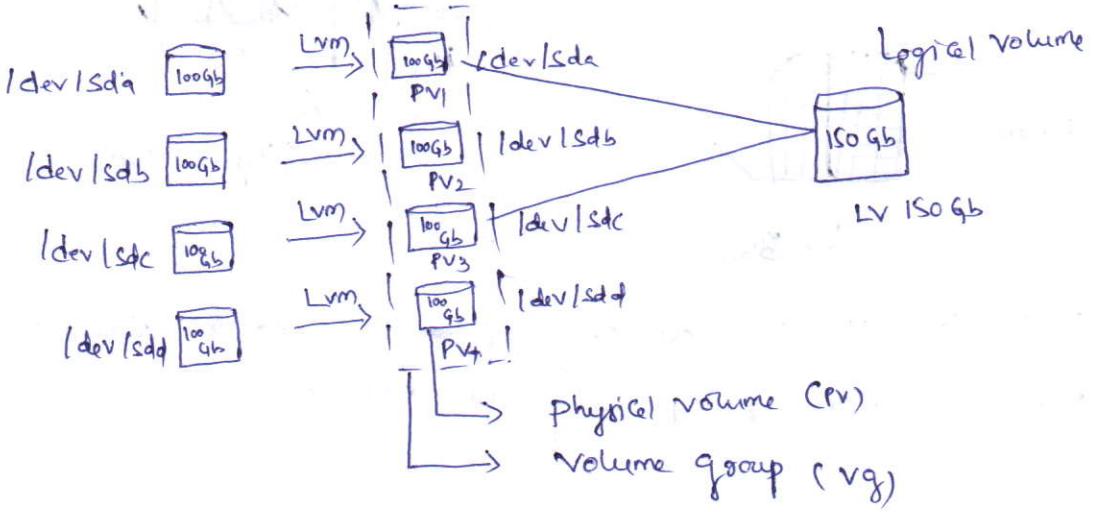
* It increases file system capacity online
* It increases file system performance online
* It increases file system redundancy online

Ex:- Svm - Solaris Volume manager - (Solaris Unix)
Vxvm - Veritas Volume manager - heterogenous (linux, windows and unix)

LVM object The objects which are coming under control of lvm is called as lvm objects.

There are 4 Lvm objects.

* physical volume (PV)
* volume group (vg)
* logical volume (LV)
* Extents < physical extents (PE)
  logical extents (LE)

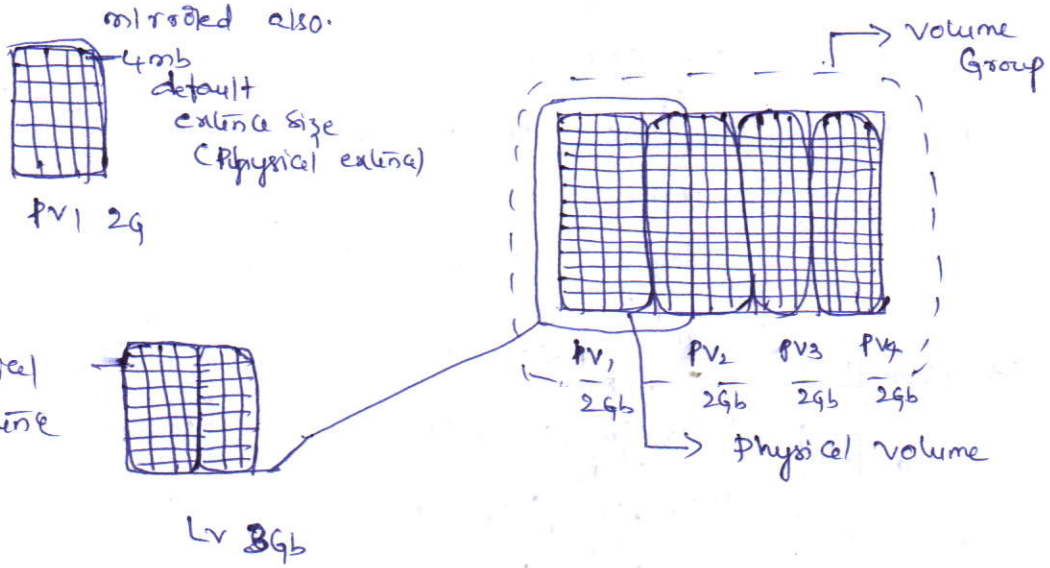


* Physical volume: The disk or partition which is under Lvm control is called physical volume

* Volume Group:- It is collection of physical volumes to Create a Volume group. min one physical volume is required. We can't use the physical volume until we add it to a volume group.

* Logical volume: It is virtual device created by taking the free space either one physical or more than one physical volume within a volume group.

Advantage: Lvm can be extended, shrink (reduce). It can be mirrored also.



Extinct:- The smallest unit of area used to store data store on volume group is called extinct.

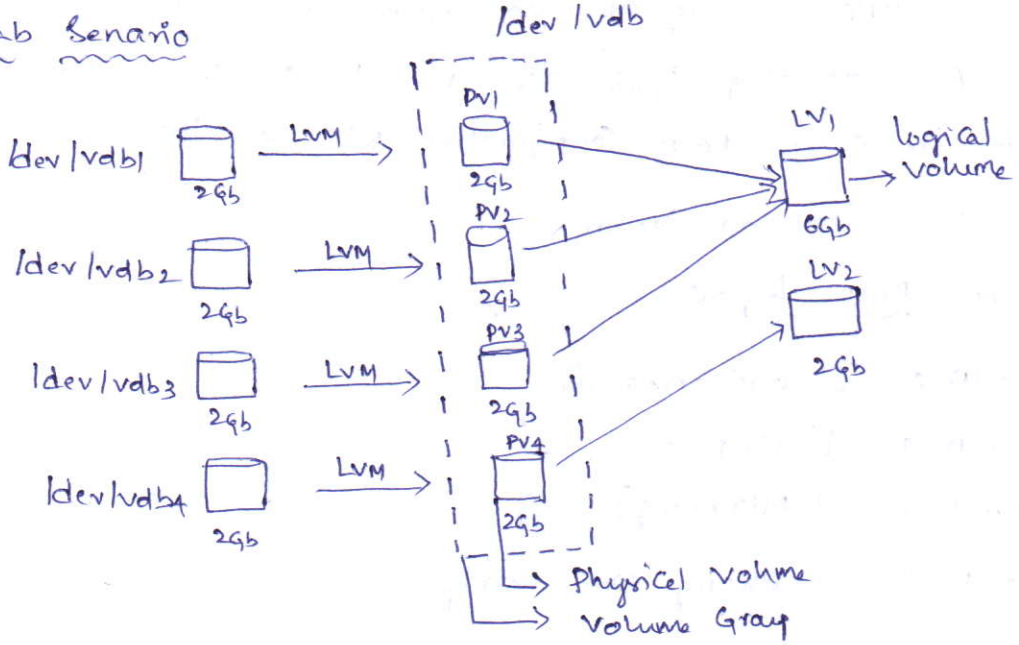
Physical extince

* Physical extince every physical volume is divided into multiple numbers of sub partitions is called as physical extince. The default physical extince size is 4mb. Physical extince size is calculated by 2^n mb n -> (2 to 9)

logical extince

logical extince every logical volume is divided into multiple number of sub partitions is called as logical partitions, The default size is depends on physical volume size.

Lab Senario



RAID (Redundent Array of Independent Disk)

Types of RAID

* Hardware RAID
* Software RAID

Hardware RAID :- If you doing RAID Configuration in hardware level i.e before installing operating System, it is known as hardware RAID. It is operating System independent. If operating System Crashes it will not effect the RAID Configuration.

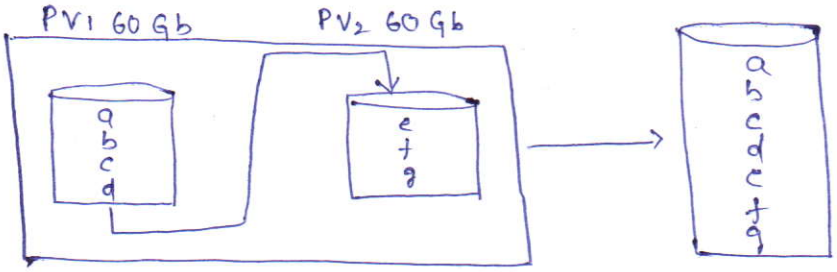
Software RAID :- If you are doing RAID Configuration in Software level i.e after installing operating System. It is known as Software RAID. It is operating System dependent. If OS Crashes (It will not effect) the entire RAID Configuration will be lost.

Ex:- lvm, Svm and VxVm

Different RAID layout

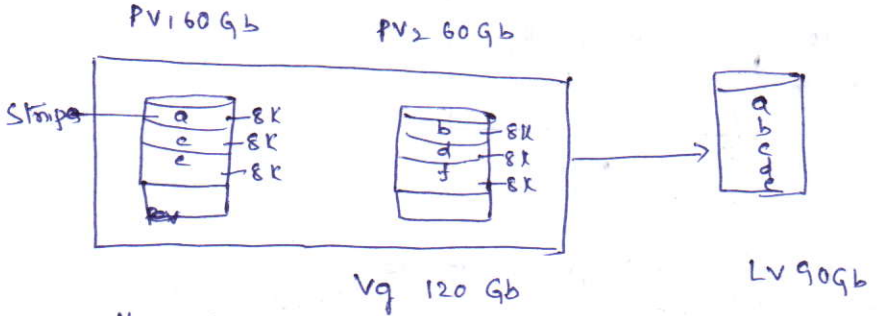
1. RAID 0 [Concatination]
2. RAID 0 [Striping]
3. RAID 1 [Mirroring]
4. RAID 5 [Striping with parity]

1. RAID 0 [Concatination]



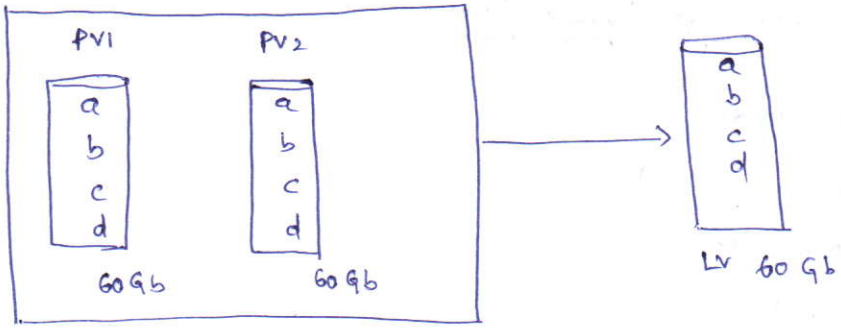
* Min 1 PV is required
* No data redundency
* writing is faster & Reading is slower

2. RAID 0 [Striping]



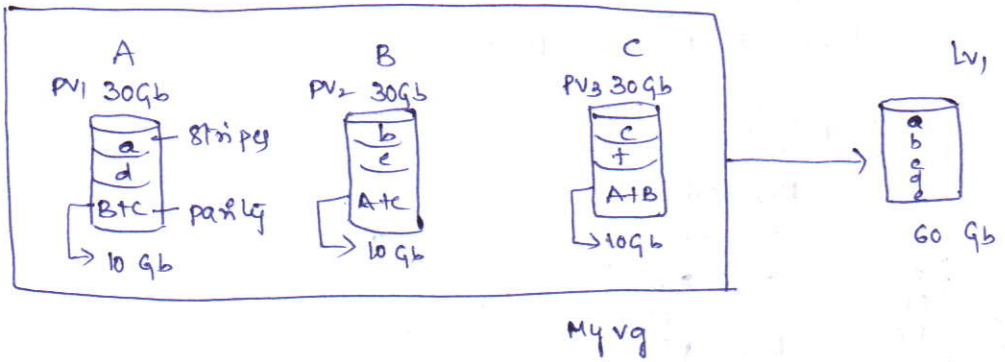
* min 2 PVs are required
* No data redundency
* writing is faster & Reading is slower
* RAID 0 Striping is faster than RAID 0 Concatination
* Stripes Size - 2^n Kb (n → 2 to 9)

3. RAID 1 [mirroring]



* minimum 2 PVs are required
* Data redundency
* writing is slower & reading is faster

RAID 5 (Striping with Parity)



Parity = $\frac{\text{Size of Smaller physical volume}}{\text{Total no of PV}} = \frac{30\,G}{3} = 10G$

* Min 3 PVs are required
* There is redundency for 1 PV failure
* Writing is slower & reading is faster
* RAID 5 is Cheaper than RAID 1.

Creating Partition for linux Lvm

fdisk /dev/vdb
: n (Create new disk)
: P (print table)
: first Sector ↵
: last Sector +2G ↵
: P (print table)
/dev/vdb1 .... 8e linux Lvm
: w (Save & exit)

Partprobe

Create, delete and display physical volume

Create :- # Pvcreate /dev/vdb1 <disk Name / Partition Name>

Display : # PVS (or) Pvdisplay (Pv Information)

Delete : # Pvremoved /dev/vdb1 <PV Name>

Create, delete and display, reduce & extend volume group

Create : # vgcreate myvg <vg Name>

Display : # vgs (or) vgdisplay

Delete : # vgremove myvg <vg Name>

Create VG with specific physical extence Size & /dev/vdb1

vg Create -S <P.E Size> <vg Name>

Extended vg

vg extended <vg name> <pv to be added>

Vg extended myvg /dev/vdb4

Reduce vg

vgreduce <vg Name> <pv to be removed>

Vg reduce myvg /dev/vdb2

Create, display, delete LV (logical volume)

Create LV: # lvcreate -L 5G <LV size> myvg <vg name>

(or)

lvcreate -L 5G <LV size> -n mylv <Lvname> myvg <vgname>

Display LV: # lvs (or) lvdisplay

Remove LV: # lvremove <LV Path>

lvremove /dev/myvg/mylv

RAID 0 [Concatination]

lvcreate -L <LV size> <vgname>

lvcreate -L 5G -n mylv myvg

RAID 0 [Stripping]

lvcreate -L <LV size> -n <LVname> <vgname> -i

<no of PVs to be added> -I <vgname> <stripe size>

lvcreate -L 5G -n mylv myvg -i 2

-I 8

RAID 1 [Mirroring]

lvcreate -L <LV size> -n <Lvname> <vgname>

-m

lvcreate -L 2G -n mylv myvg -m 1

RAID 5 (Parity with stripping)

lvcreate -L 2G <LV size> -n mylv <Lvname> myvg <vgname>

--type RAID5

Create a logical volume (LV) with 15 extents & extent size is 16 mb?

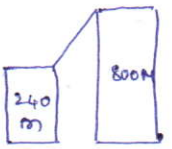
```
# fdisk /dev/vdb
        -> Create partition
: n
: P
: First Sector
  300m  Last Sector
: p     -> Print table
: t
: 8e    -> Hexa code
: p     -> Print table
  /dev/vdb1 ... 300m 8e Linux - Lvm
: w     -> Save & exit
: Partprobe

# PV create   /dev/vdb1
# vg create -s 16 myvg  /dev/vdb1
# Lv create -l 15 -n mylv myvg
```

format :- # mkfs.ext4 /dev/myvg/mylv
<Lv path>

Mount :-
```
# mkdir /sun
# mount /dev/myvg/mylv /sun
# df -hT
# vi /etc/fstab
  /dev/vdb1 myvg/mylv /sun ext4 default 0 0
: wq!
# mount -a
```

* Creating LV with +5 Extending the logical volume

*1 Extending Size of Lv



grow to # lvextended -L <dest size> <LVpath>

lvextended -L 800M /dev/myvg/mylv

grow up # lvextend -L +560M /dev/myvg/mylv
<+ size to extend> <LV path>

*2 Extend / resize file system

ext2, ext3 and ext4

resize2fs /dev/myvg/mylv <LV path>

xfs

xfs_growfs <LV path>

Note :- If you extending the lv size will be extended as per requirement but file system will not be extended so we have to extended file system also.

Reducing the logical volume

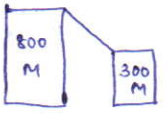
* If we can't reduce the logical volume with xfs file system

Unmounting

① umount /dev/myvg/mylv
<lv path>

(or)

/sun



② Checking file system

e2fsck -f /dev/myvg/mylv
<lv path>

③ resize2fs /dev/myvg/mylv 300m
<lv path> <new size>

④ resize of lv

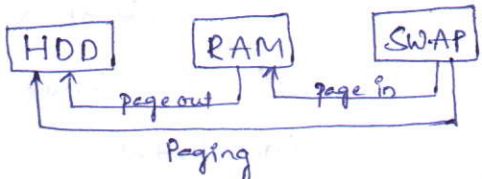
lvreduce -L 300M /dev/myvg/mylv
<new size> <LV path>

(5) Mount

Mount /sun (or) # mount -a

Swap Managment

Swap is nothing but virtual memory and which is part of hard disk when we writting data to hold initially it copies to the physical memory (RAM).



* If the RAM does not have sufficient free the remaining data copies temporary to Swap
* The data coping from swap to RAM is called as page in
* The data coping from RAM to hard disk is called as page out.
* The total prles changing the data from swap to RAM and RAM hard disk is called as paging

Note :- As server administrator it's our min responsibility to keep the utilization below the threshold value.

If unfortunately the Swap utilization is beyond the threshold value, we need to peoform the below the two tasks.

* Task 1 : list all the procures which are consuming more space and kill these procers which are not useful by getting approval from Procer owner. even after doing this if the Swap utilization is not coming down do the next step

* Task 2 : Extending the Swap space.

free -m (display RAM & Swap info)

fdisk /dev/vdb

: +2G

: p

/dev/vdb1 83 linux

: t (change it)

: L (list of all codes)

: 82 -> linux Swap / Solaris

/dev/vdb1 .... 82 linux Swap / Solaris

: w

: partprobe

mkswap <dev Name> /dev/vdb1 -> make Swap space

swapon <dev name> /dev/vdb1 -> To start the Swap

free -m (or) swapon -s -> display

vi /etc/fstab

/dev/vdb1 Swap swap default 0 0

: wq! -> (Save & exit)

Users Administration

Groups :- It is collection of users. It is used for easier administration. Every group will have a unique id known as group id (Gid). Gid → 0 to 1000 is system reserved.

Whole group information will be stored under /etc/group

Group Name ↑ GID ↑

Ex:- student : x : 1000

↓

pwd points

There are 2 types of groups

* primary * secondary

Primary Group :- Every user should and must be a part of a group called primary Group. minimum primary Group is (1) and maximum primary group is (1)

Secondary Group :- The same user can be part of multiple other groups called secondary groups.

minimum secondary groups Gatted is (0) and maximum secondary groups can be (15)

* Create a group with system defined GID

groupadd <group Name>
g1

* Create a group with user defined GID

groupadd -g <GID> 1500 <group Name> g2

* Change GID of a group

groupmod -g 2000 g2
(GID) <group Name>

* Change Group Name

groupmod -n <New group Name> g2 g5

<Group Name>

* Delete group Name

groupdel g5

<group Name>

User :- Every user has the unique id called user id (or) Uid

Different user types

* Super user 0
* System user 1 - 1000
* home user 1001 - .....

/etc/passwd

User Name | Pwd Points | Uid | P.Gid | discription | home directory

Ex:- student : x : 1000 : 1000 : student user : home/student : /bin/bash

* Create user with System defined attributes

useradd <U Name>

U1 : x : 1001 : 1002 : : /home/u1 : /bin/bash

* Set Password

Passwd U1

Enter passwd : ****

Renter Passwd : ****

* Switch user

su - <user name>

U1

logout → logout, exit or Ctrl+d

the user

* Display all groups of a user

groups <U/N>
U1

Ex:- U1 : U1 ——— Secondary groups
↓ Primary Group
User Name Name

* Display all ids of user ids of groups

id <U/N>
U1

Ex:- uid = 1003(U) gid = 1003(U) groups = 1003(U)

* Create user with user defined attributes

U/N : Tom

UID : 2396

P.G : Sap -2500

S.G : nas

User discription: lebadmin

home directory: /data/tom

Shell : /bin/sh

Note:- Before Create user please check prerequisites like group add (or) dir.

groupadd -g 2500 Sap

groupadd nas

mkdir /data

useradd -u 2396 -g <P.Group> (Sap) -G <S.Group> (Nas) -c <discription> (lebadmin)
-d <home dir> (/data/tom) -m -s <shell> (/bin/sh) <user Name> (tom)

Modifing the user attributes

-l User Name : Steve

-u user id (Uid) : 3396

-g P. group : Sapl

-G S. Group : nas1, nas2

-c description : Sysadmin

-d home directory : /oracle/Steve

-s shell : /bin/bash

```
# groupadd Sapl
# groupadd nas1, nas2
# mkdir -p /oracle/Steve
# useradd -u 3396 <New id>  tom <User Name>
#  "   -g Sapl <New PG>  "
#  "   -G Nas1, Nas2 <New SG>  "
#  "   -c Sysadmin <description>  "
#  "   -d /oracle/Steve <New hid>  "
#  "   -s /bin/bash <new shell>  "
#  "   -l Steve <New user>  "
```

Deleting user

```
# userdel -u <U/N>  -> Delete user
```

(or)

```
# userdel -r -u <U/N>
```

↳ Delete user along with its home directory.

groupmems list all users in a group

```
# groupmems -g <g/N> -p
                 nas
```

Adding user to a group

```
# groupmems -g <g/N> -a <u/N>
                 nas      u1
```

Remove user from a group

```
# groupmems -g <g/N> -d <u/N>
                 nas     slave
```

Note:- user id -o is not unique id

Create a user as a Super user

```
# useradd -u <user id> -o <user Name>
                 0(zero)      ram
                  ↓
              mode option
```

modify a user as a Super user

```
# usermod -u <user id> -o <user Name>
                0(zero)      raju
                 ↓
             mode option
```

Note: Once you make a normal as root user we can't make him as normal user and we can't delete

Shell It is interface between user & operating system if any user want to execute a command & if he want to login. he should have a valid shell.

- /bin/bash → Born again shell → Linux
- /bin/sh → Born shell → Soleries, Hp-Unix
- /bin/ksh → Korn shell → AIX
- /bin/csh → c shell
- /bin/zsh → z shell

echo $SHELL → Display default shell

echo $0 → Display present working shell

* Change from shell to another shell

/bin/sh

* Changing ownership of file & directory

u1 f1 & d1

chown <U/N> <f/d Name>

* Changing Group ownership of file & directory

g1 f1/d1

chgrp <New group> <file/dir Name>

* Changing group and ownership for file & directory at a time

f1/d1

chown:grp u1:g1 <f./d Name>

<U.N> <g.N>

Access Control List

ACL :- We can change a permission of a file & dir using chmod command. We can only change common permissions (u,g,o), not the customized permission. If you want to do customized permission, we want to use the acl method using acl we can set the permission for a particular user & a group in particular file & directory.

There are two types of entry

* Trival entry
* Non_trival entry

Trival Enlry : The file or directory which is having normal chmod is called Trivel enlry. we can identify the enlry by looking at its permission.

f1

\# ls -l <f / d Name>

rw-r--r-- ⊙ Trivel enlry

Non-Trival Enlry :- The file or directory which is having special permission like acl is called Non-Trivel enlry

f1

\# ls -l <f/d Name>

· rw-rwxr-- ⊕ Non-Trivel enlry

Set Acl (or) modify Acl

u_1 - rwx	g_1 - rw
u_2 - rw	g_2 - rwx
u_3 - wx	g_3 - w

denote user ↑ User Name ↓ Permission ↑

\# Setfacl -m u:u1:rwx, u:u2:rw, u:u3:wx,

file1

group Name ↑

g:g_1:rw, g:g_2:rwx, g:g_3:w <file / directory>

↓ denote group ↓ Permission

\# getfacl <f / d Name>

Deleting acl for particular a user or a group

\# Setfacl -x u:u3, g:g_1 <file / dir Name>

(no space)

Delete all acl set in a file / dir

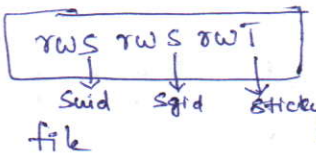
file1

\# Setfacl -b <file / dir Name>

Special Permissions

1. Suid → Set uid
2. Sgid → Set gid
3. Sticky

	Effect on file	Effect on dir
Suid	file executes as the user that owns the file, not the user that ran the files	No Effect
Sgid	file execute as the group that owns the file	files newly created in the directory have their group owner set to match the group owner of the directory
Sticky	No effect	user usage with right on the dir can only remove files that they own, they cannot remove or force saves to files owned by others



Suid (u+s) - 4 — file

Adding { # chmod u+s <file / d Name>
chmod 4644 <file / d Name>

Ex:- -rwSr--r--

Remove { # chmod u-s <file / directory Name>

Ex:-

Sgid (g+s) - 2

Adding { # chmod g+s <file / dir Name>
chmod 2755 <file / dir Name>

Ex:-

Remove { # chmod g-s <file / dir Name>

Ex:-

Sticky (o + t) → 1

Adding { # chmod o+t <file / dir Name>
chmod 1777 <file / dir Name>

Ex:-

Remove { # chmod o-t <file / dir Name>

Ex:-

grep Searching the lines with particular pattern (or) string in a file

```
# grep <pattern> root /etc/passwd
      <file name>
```

(or)

```
# cat /etc/passwd | grep <pattern> root
```

```
# grep -n root /etc/passwd (To display line numbers)
```

```
# grep -i root /etc/passwd (To ignore case sensitivity)
```

```
# grep -v root /etc/passwd (Display the lines, which is not having particular characters).
```

* Display the line starting with a character

```
# grep ^<pattern> <file name>
# grep ^root /etc/passwd
```

* Display the line ending with a character

```
# grep <pattern>$ <file name>
# grep /bin/bash$ /etc/passwd
```

*** To find the date and storing the output date into text file

```
# grep /bin/bash$ /etc/passwd > file.txt
```

* find It is used to find file or directory

1. Name based Searching
2. user based Searching
3. group based Searching
4. Permission based Searching
5. Size based Searching

Passwd

find / -name <file Name> // search in entire machine

Passwd

find /etc -name <file Name> // search in only under /etc directory

User based Searching

Display all the files / directories owned by a particular user

root

find / -user <u/n> // search in entire system

student

find /etc -user <u/n>

Group based Searching

Display all the files / directories owned by particular group

mail

find / -group <g/N>

mail

find /var -group <g/N>

Permission based Searching

Display all the files / directories a particular Permission

777

find / -perm <permission>

Size based Searching

find / -size 10M -> Search for exact size

find / -size +10M -> greater 10 mb

find / -size -10M -> Less 10 mb

* find the files and directories changing permission /root → 755
→ 777

find /root -perm 755 -exec chmod 777 {} \;

* student → /root/new

mkdir /root/new

find / -user student -exec cp -vf {} /root/new \;

Archeiving & Compressing

Archeiving & Compressing files are use full when we are creating backup and transfering the data ento the network. One of the oldest and most common command for creating and working with backup archeiving is the tar command. The tar file can be compressed using eithe gzip / bzip2 method.

du -sh /etc <d/N> // display the usage of size of the directory

tar

tar -cvf etc.tar /etc

c - create
v - verbose (output)
f - file

zip

gzip <f/N> etc.tar // etc.tar.gz

(or)

bzip2

bzip2 <f/N> etc.tar // etc.tar.bz2

unzip

gzip # gunzip <f1N> elc.tar.gz // elc.tar

bzip2 # bunzip2 <f1N> elc.tar.bz2 // elc.tar

Extracting

tar -xf <f1N> elc.tar => Extract to pwd

tar -xf <f1N> elc.tar -C /root <path>

⇓

Extract to particular dir.

tar -tf <tar file> elc.tar

⇓

Display all files in tar files

***

tar & gzip

tar -cvzf elc.tar.gz /elc

t - list
x - extract
z - gzip
j - bzip2

tar & bzip2

tar -cvjf elc.tar.bz2 /elc

Analyzing & storing logs

In RHEL7 the logs are stored in two different methods

1. rsyslogs → /var/log → Permanent → stored in disk
2. Systemd → Journals → temporarily → stored in memory

Note: Process and operating System Kernel need to be able to record a log of event that happend. These can be useful for auditing the System and troubleshooting problems. By Conventions the /var/log

directory is where these logs are stored

Log file	Purpose
/var/log/messages.log :	Most Syslog msgs are logged here. The exceptions are msgs related to authenticated and Imaging Processing, periodically run jobs and those which are purely debugging related
/var/log/secure.log :	The log file for security and authentication related msgs and errors.
/var/log/mail.log :	The log file with mail server related msgs
/var/log/cron.log :	The log file related to periodically executed tasks.
/var/log/boot.log :	The log file, msgs are related to system startup are logged here.

Log file rotation

Logs are rotated by log rotated utility to keep them from filling at the file system containing /var/log when a log file is rotated the all /var/log/messages file may become /var/log/messages - 2016 12 31. If it is rotated on dec 31 2016. once the log file is rotated a new log file is created. After a certain number of rotation typically after 4 weeks the all log file is discarded to free disk place.

tail /var/log/messages -n 1

Dec 29 15:40:01 — Time stamp when the log entry was rotated

localhost — The host from which the log msgs was sent

systemd: — Program or process that send by log msg

Started Session 14 of user root; — Actual msg which is sent.

Systemd – Journals

Journalctl

Journalctl --since "2016-12-31 12:00"

↳ Display all the log files from specific date

Journalctl --since "2016-12-31 12:00" --until "2017-12-31 12:00"

Journalctl _UID=0

↳ Display all the logs of particular user

Journalctl _PID=1

↳ Display all the logs of process with process id 1

Root Password breaking

exam ✱ Power on the System - while booting the System the Grub menu will be displayed, to boot the system using both follow these steps

1. Use arrow keys to select the boot entry you want to edit
2. Press e to start editing that entry
3. Use cursor keys to go to the line that starts with linux16

[linux16 ..... UTF-8 rd.break] Server

and Press ctl + X

↳ To go to emergency mode

classroom practices

[linux16 ..... UTF-8 rd.break console=tty1

and ctl+X]

4. Switch_root: /# mount -o rw,remount /sysroot

[sysroot is having default read permission. if you mention above write permission with read. it can able to changes to root user for password reset]

5. Switch_root: /# chroot /sysroot

sh-4.2 # passwd root

******

******

sh-4.2 # touch /.autorelabel -> Relabelling the Selinux target policy

Sh-4.2 # exit

Switch-root: / # exit.

Scheduled of Jobs

We can scheduling of jobs by using two type of jobs

* at Job
* Cron Job

At Job

If you are scheduling the job by at job it can be performed for one time only. By default any normal user can access the at job

Ex:- at > touch f1
at 10:30 am jan 31
: ctl+d → Save & exit

[at tea, at 3pm +4 day, at noon, at now, at midnight, at 1am tomorrow]

atq (or) at -l → To display all the at jobs in queue

atrm <job ids> } Delete Partitions
at -r <job ids> } Job from queue.

* To deny the user accessing the "at" job

If you want to denying the users accessing at job specify the user name in the file "/etc/at.deny"

```
# vi /etc/at.deny
u1
u2
u3
:wq!
```

* If you want to specify the users to access the 'at job'. You have to create a new file in "/etc/at.allow" and specify the users name inside the file. Then if users can access the at job.

If any user (or) same user having access the at.deny & at.allow then at.allow will over rule at.deny. The user can access.

Cron job

If you scheduling the jobs using 'cron' job. It can be performed in multiple enrolled of time. By default every normal user have the access to cron job.

```
# cat /etc/crontab
# which touch
/usr/bin/touch
```

crontab -e

	min (0 to 59)	Hour (0 to 23)	Day of month (1 to 31)	Month (1 to 12)	Day of week (0-6)	<Job to be> done
Jan 31 10:30 am	30	10	31	1	*	/usr/bin/touch f1
Every month 31 10:30 am	30	10	31	*	*	/usr/bin/touch /f1
Every day 10:30 am	30	10	*	*	*	" "
Jan & Dec 25 10:30 am 10:30 pm	30	10,22	25	1,12	*	" "
Every mon & Sat 11:30 pm	30	23	*	*	1,6	" "

crontab -l -> list of cron jobs queue

crond -> cron services

Ex: Scheduling Job for a particular user

```
# Crontab -eu student <U/N>
# Systemctl restart crond        -> Restart service
# Systemctl enable crond         -> enable service
# Contab -lu <U/N>               -> Display Job of a Particular user
```

* If you want to specify the users to access the 'cron'.tab' you have to create a new file in /etc/at cron.allow' and specify the users name inside the file.

* To deny the user accessing the "Crontab at" Job
If you want to denying the user accessing at Job specify the user name in the file "/etc/cron.deny"

* If same user is having access the at cron.allow & cron.deny then cron.allow will over rule and give the access to user, who are ever added in this file /etc/cron.allow

```
# vi /etc/cron.deny
u1
u2
u3
:wq!

# vi /etc/cron.allow
u1
u5
:wq!
```

Package Management

1. rpm [redhat Package Manager]
2. yum [yellowdog update Manager]

Package Example

tf5 - 0.17 - 66.e17 . X86-64 . rpm

- tf5 → Package Name
- 0.17 → version
- 66.e17 → release
- X86-64 → architecture

rpm

```
# mount -o loop /rhel7.iso /media
# cd /media
# ls
Packages
```

Note:- If you don't have Packages folder in your local system then mount Rhel7.iso to media for ~~fastest~~ Install Packages

Install

```
# rpm -ivh <PKN>
```

(or)

```
# rpm -ivh <full URL>
```

i = install
v = verbose
h = hash (%)

update

```
# rpm -Uvh <PKN>
```

Delete or uninstall

```
# rpm -e <PKN>
```

List all Installed Packages

```
# rpm -qa        q = query
                 a = all
```

List information about a package

```
# rpm -qi <PKN>
```

Display all files Created when Installing a Package

```
# rpm -ql <PKN>
```

Display the package responsible for a file

```
# rpm -qf <file name>
```

Note :- The main disadvantage of using rpm is dependence issue If you are Installing using rpm and if that particular Package have dependence pakages it will not be Installed automatically. where we need to Install it manually.

```
# rpm -ivh http <PKN> --nodeps
```

↳ Install the package without dependences

Yum

The main advantage of using the yum is to resolve the dependency issue. i.e., the dependency packages install automatically

Yum Server

```
# mount -o loop /rhel7.iso /media
# mkdir /rhel
# cp -ar /media/Packages /rhel
# cd /root
# ls
  Packages
# rpm -ivh (create repo * (or) full name of the package
# Createrepo -v /rhel   (Making rhel directory as repository)
```

Yum Server Configuration

```
# vim /etc/yum.repos.d/install.repo
[install]          -> repoid
name = yum server  -> repo name
enable = 1
gpgcheck = 0
baseurl = file:///rhel
:wq!
```

yum clean all → refreshing

yum repolist → list of all repository

yum install <PIN>

Yum client

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

[new] → repo id

name = yum client → repo name

enabled=1

gpgcheck=0

baseurl = http://content.example.com/rhel7.0/x86-64/dvd

:wq!

yum clean all → refreshing

yum repolist → list all repository

yum install <PIN>

Install package

yum install <PIN>

yum install <PIN> -y

Update

yum update <PIN>

Uninstall

yum remove <PIN>

Info of a package

yum info <PIN>

list all packages (installed & not installed)

yum list all

Display all installed

yum list installed

Display not-installed packages in repository

yum list available

Display package name responsible for file /var/www/html

yum provides <fIN>

Kernel updation

Exam # uname -r (or) uname -a → To check current version of Kernel

3.10.0-123.el7

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

[Kernel]

name = Kernel updation

enable = 1

gpgcheck = 0

baseurl = http://content.example.com/rhel7.0/x86-64/errata

:wq!

yum clean all

yum repolist

yum install Kernel -y

reboot

uname -r

Booting Process

Power-on
↓
POST
↓
BIOS
↓
Boot device
↓
1st 512 bytes

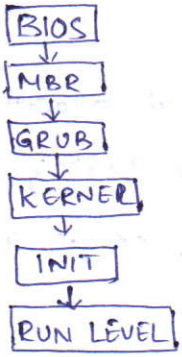
- 446 bytes — Primary boot loader info
- 64 bytes — partition & fs info
- 2 bytes — MBR validity check info

↓

Read /boot/grub/grub.conf (or) /etc/grub.conf
↓
find default kernel
↓
loads kernel to memory
↓
mounts root fs
↓
/sbin/init (first process id is 1)
↓
mounts all other fs
↓
start all other services
↓
start all ...

Read /etc/inittab to find default run level (5) is

go to current run level



* Booting process is very important to server administrators to understand the behaviour of operating system and its functionality
* we know the booting process it helps us to diagnous and fix the issue

we have Server booting phases

when we power on the server. It performs power on self test (POST) to all connected hardware devices. upon the successful of completion of Post it Contact to bios to identify the boot device and boot disk. The first 512 bytes of hard disk Contains MBR it is divided into three parts

1. The first 446 bytes contains the information about primary boot loader
2. 64 bytes Contains the information partitions and file System
3. The last 2 bytes Contains MBR validity check information

Grub (Grant unified boot loader)

Grub is default boot loader in linux. The primary responsibility of grub is to identify the default Kernel and loading to Kernel to memory by reading /boot/grub/grub.conf (or) /etc/grub.conf. and starts mounting the root life file System. Then Parent process /sbin/init starts and all other services and mount all others fs and finally boot the Server to default run level (run level 5) by reading /etc/init+tab.

Disk Quota

- Disk quotas are nothing but allocating and limiting the free space to normal users for storing the personal files
- Disk quotas are disabled by normal users

```
# fdisk /dev/vdb
    /dev/vdb1  -sg
# Partprobe
# mkfs.ext4  /dev/vdb1
# mkdir /sun
# vi /etc/fstab
  /dev/vdb1  /sun  ext4  defaults,usrquota,grpquota  0 0
: wq!
# Mount -a
# ls /sun
# quotacheck . -cugf /sun
# ls /sun
  quota.group   quota.user
# user add u1
# edquota u1
```

c - Create
u - user } quota
g - group
f - forcefully

filesystem	blocks	soft	hard	inodes	soft	hard
/dev/vdb1	0	0	0	0	0	0

don't change value, value will be automatically update

```
# edquota u1
```

fs
/dev/vdb1

blocks: SL 30, HL 40

inodes: SL 10, HL 15

```
: wq!
```

quotaon -v /sun

update the quota

fs	block		inodes	
	SL	HL	SL	HL
/dev/vdb1	30	40 (50)	10	15 (20)

:wq!

quotaon -v /sun

Delete quota of a particular user

edquota u1

fs	block		inodes	
/dev/vdb1	30 (0)	50 (0)	15 (0)	20 (0)

quotaon -v /sun

Delete all quota of a user

repquota -v /sun <dir name>

Daemon

a daemon is process, which run continously in back ground and provide services to end users as per the demand

Systemd : It is a daemon, which is depleced by init in Rhel7. It is parent process i.e, the first process to be started in the system. the process id (PID) is 1. Systemd takes responsibilities manage system units like services, sockets and targets (/usr/lib/systemd/system)

```
# Systemctl list-units --type=service -a
# Systemctl list-units --type=socket -a
# Systemctl list-units --type=target -a
```

* In Rhel7 the run levels are depleced by system targets

* Status of a service

```
# Systemctl status sshd
```
<ServiceName>

* Start of service

```
# systemctl start sshd
```
<Service Name>

* Restart a service

```
# Systemctl restart sshd
```

* Stop a service

```
# Systemctl stop sshd
```

* Enabled
  └> Start Service after reboot

  # Systemctl enable sshd

* disable
  └> Stop Service after reboot

  # Systemctl disable sshd

* Check a Service is enabled / disable
  # systemctl is-enabled sshd

* mask
  └> Permanent disabled a Service

  # Systemctl mask sshd

* unmask
  └>
  # Systemctl unmask sshd

Run levels in Rhel 6

0 - halt
1 - Single user mode
2 - multi user mode without Nfs
3 - multi user mode with Nfs
4 - Not used
5 - Graphical
6 - reboot

init 0 → shutdown (R6) (or) # systemctl poweroff (R7)
(or)
poweroff

init 6 → reboot (R6) # systemctl reboot (R7)
(or)
reboot

* To display list-units of target

systemctl list-units --type = target

- graphical.target
- multi-user.target

systemctl get-default → display default targets in your system

* To temporarily change target

multi-user.target

systemctl isolate <target Name>

* How to change the target permanently

systemctl get-default

ls -l /etc/systemd/system/default.target

/ ——— default.target < graphical.target

rm -f /etc/systemd/system/default.target

ln -sf /usr/lib/systemd/system/multi-user.target

/etc/systemd/system/default.target

ls -l /etc/systemd/system/default.target

/ ——— default.target < multi-user.target

systemctl get-default

multi-user target

(or) Single step

systemctl set-default multi-user.target

reboot.

Firewall

The daemon response for firewall is firewalld. In Rhel 6, we have IP tables, which is managed by net filtering. In Rhel 7 IP tables are replaced with firewall.

Note: Make sure that IP tables are permanently disabled inorder to Configure firewall in Rhel 7. It is not recommended to Configure both IP tables and firewalling in same operating System.

* Permanently stop the services of iptables & IP 6 tables

```
# System ctl mask iptables
# System ctl mask ip 6 tables
```

* To display the firewall configuration, which are opened in Services

```
# firewall-cmd --list-all
public (default, active)
interface
.....
```

* To open port/services (remove service / port from security layer)

```
# firewall-cmd --add-service=ftp
# firewall-cmd --add-port=80/tcp
                         /udp
```

* close the ports / services (add services / ports to security layer)

firewall-cmd --remove-service=ftp

firewall-cmd --remove-port=80/tcp

Note:- If you add / open service / ports are temporarily

* If you want to permanent the ports / services

* Open the ports / services (remove from security layer)

1. # firewall-cmd --permanent --add-service=ftp

firewall-cmd --reloads

2. # firewall-cmd --permanent --add-port=80/tcp

firewall-cmd --reloads

* close the ports / services (adding to security layer)

1. # firewall-cmd --permanent --remove-service=ftp

firewall-cmd --reloads

2. # firewall-cmd --permanent --remove-port=80/tcp

firewall-cmd --reloads

* To display Graphical environment of firewall Configuration.

```
# firewall --config
```

* To display Current zone

```
# firewall-cmd --get-default-zone
```

* To change Current zone to another zone

```
# firewall-cmd --get-default-zone=work
```

* To display of particular zone Configuration

```
# firewall-cmd --list-all --zone=home
```

* Open the services / port from security layer to particular zone

```
# firewall-cmd --add-service=ftp --zone=home
```

(Or)

```
# firewall-cmd --add-port=21 --zone=home
```

Selinux

(Security and Enhancement linux)

It is top most security layer in linux operating systems. to protect the data of a file and contents of the directory from unauthorized process accessing.

We generally protects the files and directories from users, groups by configuring strict permissions but these permisions cannot stop unauthorized process for accessing the file and directory.

In order to limit on unauthorized process not to access file and directories then we have to protect it with selinux.

However for file and directory we have permissions the same way we have something called security context. The sec - context of files or directories decides wether a process is allowed to access the data of a file (or) the content of the directory

Note: When we install a linux operating system packages responsible for selinux (or) default installed and enabled.

* Selinux Modes

1) Enforcing - Selinux policies are strictly enabled
2) permissive - warning will be provided, but no security
3) disabled - Selinux policies are not enabled

* To display the present Selinux mode

getenforce

Enforcing

Setenforce 0 / To change enforcing to permissive (temporarily)

Setenforce 1 / To change permissive to enforce (temp)

* To change the Selinux mode permanently

Exam

vim /etc/selinux/config (or) /etc/sysconfig/selinux

Selinux = permissive

:wq!

reboot // once reboot, mode will be updated

getenforce

* Display the Security Context of file & directory

ls -Z <f1/N> / To check the Context of the file/dir

ls -Zd <d1> / To check the Context of the directory

---- unconfined_u: object_r, default_t: s0 /d1

user role type Sensivity

↳ Samba.

Ex: samba_share_t

* Changing Security Context of a file / dir
  * Chcon
  * Semanage

Chcon : We can change security context of a directory / file, but it is temporarily.

Semanage :-

ls -zd /d1
---default_t--

chcon -t samba_share_t /d1
---samba_share_t--

Semanage :- We can change security context of a directory & file, we there we can change by using this cmd change permanently.

ls --zd /d2
---default_t--

semanage fcontext -a -t samba_share_t "/d2 (/.*)?"

restorecon -Rv /d2
↳ Recursive

ls -zd /d2
-- samba_share_t . .

IPV4 (Internet Protocol Version 4)

* 32 bit address, (4 octets), Binary expressed in decimals.

```
00000000.00000000.00000000.00000000   0.0.0.0
11111111.11111111.11111111.11111111   255.255.255.255
```

Class A	0 to 126	0.0.0.0	126.255.255.255	} Unicast address
Class B	128 to 191	128.0.0.0	191.255.255.255	
Class C	192 to 223	192.0.0.0	223.255.255.255	
Class D	224 to 239	224.0.0.0	239.255.255.255	} multicast address
Class E	240 to 255	240.0.0.0	255.255.255.255	} Broadcast

127 - loopback

Class	A	B	C
	N.H.H.H	N.N.H.H	N.N.N.H
Netmask & Subnet	255.0.0.0	255.255.0.0	255.255.255.0
IP address	100.1.2.3	172.25.0.11	192.168.1.2
Prefix/ CIDR	100.1.2.3/255.0.0.0 100.1.2.3/8	172.25.0.11/255.255.0.0 172.25.0.11/16	192.168.1.2/255.255.255.0 192.168.1.2/24
Network IP	100.0.0.0/8	172.25.0.0/16	192.168.1.0/24
Start usable IP	100.0.0.1	172.25.0.1	192.168.1.0
Usable IPs	⋮	⋮	⋮
Last usable IP	100.0.0.254	172.25.0.254	192.168.1.254
Broadcast IP	100.0.0.255	172.25.0.255	192.168.1.255
Total IPs	(256x256x256-2)	(256x256-2)	(256*2)

IPV6

- 128 bits , Hexadecimal (0-9, a-f), 8 bits - N, 64 bits - H

0000 · 0000 · 0000 · 0000 · 0000 · 0000 · 0000 · 0000

Ex1:- 00ab : 0000 : 0000 : aef8 : 00bd : 0 : 0ae8 /64

First Rule : Remove the zeros before Hexadecimal numbers

ab : 0000 : 0000 : aef8 : bd : 0 : ae8

Second Rule : More than two zeros we can write single zero

ab : 0 : 0 : aef8 : bd : 0 : ae8

Third Rule :- more than two zeros we can mention "::"

ab : : aef8 : bd : 0 : ae8

Fourth Rule Do not remove zeros after applied the third rule at a time.

Ex2 :- ab : 0 : 0 : 0 : aef8 : 0 : 0 : ae8

ab : : aef8 : 0 : 0 : ae8
1 ↓ 2 3 4 5
(3)

Loop back IPV6 : : 1/128

* To check IP address information

If Config

Network Manager

Device :- It is network interface adapter (NIC)

Connection :- Connection is set of settings we configure on a device

Network Manager :- The device or tool, which is used to manage the Connections. The Configuring to create, modify or delete a Connection is called network manager.

When we create a new connection (or) for every connection a new configuration file is created under /etc/sysconfig/network-scripts with the same connection name.

Con1 /etc/sysconfig/network-scripts/ifcfg-con1

Con2 /etc/sysconfig/network-scripts/ifcfg-con2

To display the device information and IP of the device

```
# ifconfig
    eth0 ...    172.25.0.2
```

To display the device config & IP of the device

```
# ip addr show    (or)    # ip a s
```

* To display particular device information

nmcli dev show <dev name> (eth0)

* To display all the Connection in our System

nmcli Con show (or) # nmcli Con s

* To display the information about particular Connection

nmcli Con s Con1
nmcli Con s <Con name> "System eth0"

* To display active Connection

nmcli Con s --active (or) # nmcli Con s -a

* Bring down Connection

nmcli Con down "System eth0"

* Bring up the Connection

nmcli Con up "System eth0"

* Delete a Connection "System eth0"

nmcli Con del "Con1 name".

* Create a Connection (or) modify a connection

nmcli Con add Con-name Con1 type ethernet ifname eth0

nmcli Con ~~add~~ mod Con1 ipv4.addresses "192.168.1.2/24 192.168.1.1" ipv4.method manual

nmcli Con mod Con1 ipv4.dns "8.8.8.8"

One single command to create ip address, dns

nmcli con add con-name con2 type ethernet ifname eth0 ipv4.addresses "192.168.1.3/24 192.168.1.1" ipv4.method manual connection.autoconnect yes

Exam:- IP address : 192.168.1.100
Gateway : 192.168.1.1
DNS : 8.8.8.8

nmcli con mod "System eth0" ipv4.addresses "192.168.1.100/24 192.168.1.1" IPV4.method manual IPV4.dns "8.8.8.8" Connection.auto connect yes

Restart service of down & up the device

Systemctl restart network (or) nmcli con down <Con name>
nmcli con up <Con name>

Set the IP address of IPV6

nmcli con mod "System eth0" IPV6.addresses "2001:ab/64" IPV6.method manual.

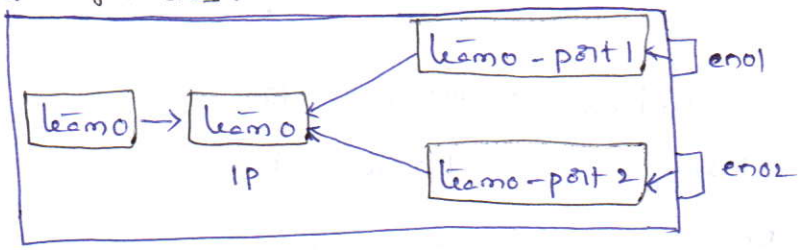
* You can set IP address, gateway and dns using another command by using "nmtui" manual enter.

Note:- Do not use to set ip address to IPV6 by using this "nmtui" command.

NIC Teaming / Link Aggregation

IP multipathing

It is used to provide network redundancy in the event of physical adapter failures.



Teb teem bridge setup

Step 1: Con-name → team0
Type → team
Ifname → team0
Config → runner-name active backup
// switching will happen when one device will fail

Step 2: Set IP for master

Step 3: Con-name → team0-port1
Ifname → eno1
type → team-slave
master → team0

Step 4: Con-name → team0-port2
Ifname → eno2
Type → team-slave
master → team0

Note: If want to check examples of any command you can use command below mentioned.

man nmcli-examples (or) # man teamd.conf

Step 1: nmcli con add con-name teamo type team ifname teamo config '{"runner":{"name":"activebackup"}}'

Step 2: nmcli con add mod teamo ipv4.addresses 192.168.12.24 ipv4.method manual connection.autoconnect yes

Step 3: nmcli con add con-name teamo-port1 ifname eno1 type team-slave master teamo

Step 4: nmcli con add con-name teamo-port2 ifname eno2 type team-slave master teamo

To check the state of the device

teamdctl teamo state

check the failover testing

nmcli con down teamo-port1

teamdctl teamo state

SSH (Secure Shell)

The main purpose of SSH is used to connect from one server to another server. Using SSH the data transfer will be encrypted formatted. The port for SSH is 22. By default Packages are responsible for ssh is not installed. We need to install it manually. By default the root login are enable in ssh.

Sys1	Sys2
Desktop / client	Server
172.25.X.10	172.25.X.11
desktopX.example.com	ServerX.example.com

Server Side Configuration

```
# yum install -y openssh
# systemctl restart sshd
# systemctl enable sshd
# firewall-cmd --permanent --add-service=ssh
# firewall --reload
```

Client - side

yum install -y openssh

systemctl restart sshd

systemctl enable sshd

hostname

desktopX.example.com

ssh <u/n> @ <host name>
@ <IP address>

Server Side

#vim /etc/ssh/sshd_config // To display root login

Esc+/ (Enter Permit Root)

PermitRoot login yes (remove '#')
↓
NO

:wq!

systemctl restart sshd

Password less Authentication

* public key Authentication
* Private key Authentication

→ Client Side (desktop)

ssh-keygen

Generating public / private rsa key pair

Enter file in which (/root/.ssh/id-rsa):

Pass phrase ↵

cd /root/.ssh

ls

id_rsa.pub

Server Side

cd /root/.ssh

ls

authorised_keys

Cat authorised_keys.

SAMBA SERVER

The main purpose of Samba Server is used to upload & download files and directories under heterogenious (Heterogeneous) environment.

Samba clients may be windows / linux clients

FTP	SAMBA
* User based authentication	* User based authentication
* We can upload & download any files	* We can upload & download both files and directories
* No sharing of resources	* We can share the resources like CD Rom (or) printers
* By default anonymous ~~anonymous~~ login are enabled ~~normal~~	* By default anonymous login are enabled
* Any normal user can access the ftp server	* Only valid Samba users can access Samba servers.

* Share the following directories with samba users with permissions

/dir1 -> 172.25.X.10
/dir2 -> 172.25.X.0/24
} U1, U2, U3

/dir1 -> U1, U2, U3 } r | /dir2 -> U1, U2 } r | /dir2-U3 } rw

755

Server Side

* 1. Install Package
2. Restart Service and enable Service
3. firewall
4. Create directory
5. Change Selinux Context
6. Create users
7. make user to Samba users
8. Set acl (If 'w' is required)
   ↳ Permission
9. Samba Configuration
10. Restart Service

Server Side Configuration

```
# yum install -y samba*          // install package
# Systemctl restart smb nmb     // restart services
# Systemctl enable smb nmb      // Enable services
# firewall-cmd --permanent --add-service=samba
# firewall-cmd --reload         // firewall config & service reload
#mkdir /dir1 /dir2              //Create directories
# touch /dir1/f{1..5} /dir2/f{1..5}   // Create files under directories for checking
```

// Change Context

```
# Semanage fcontext -a -t Samba_Share_t "/dir1(/.*)?"
# Semanage fcontext -a -t Samba_Share_t "/dir2(/.*)?"
# restore: Con -Rv /dir1 /dir2
(nospace)
# ls -Zd /dir1 /dir2
```

```
# useradd U1
# useradd U2
# Use radd U3
```

} // Create User

```
# smbpasswd -a U1↵
enter the passwd
# smbpasswd -a U2↵
enter the passwd
# smbpasswd -a U3↵
enter the passwd
```

```
# pdbedit -L        // To check the Samba users list
# setfacl -m U:U3:rwx /dir2
# vim /etc/samba/smb.conf
```

go to last line

```
[Share1]
Comment = Read Permission
Path = /dir1
browsable = yes
hosts allow = 172.25.X.10
```

// allow the host or network Systems to access Samba Server.

[share2]

Comment = Read Permission

Path = /dir2

browsable = yes

host allow = 172.25.X.0/24

writ list = u3

:wq!

(10) systemctl restart smb nmb

Client Side Configuration (172.25.X.10)

yum install -y samba-client

smbclient -L 172.25.X.11

Server IP

root pwd! ↵ (Don't give, ↵)

smbclient //172.25.X.11/share1 -U u1

smbclient //172.25.X.11/share1 -U u2

smbclient //172.25.X.11/share1 -U u3

smb: \> ls

smb: \> mkdir id1

(00)

smbclient //172.25.X.11/share2 -U u1

smbclient //172.25.X.11/share2 -U u2

smbclient //172.25.X.11/share2 -U u3

smb: \> ls

smb: \> mkdir id1

Client Side

yum install -y cifs-utils

mkdir /test

vim /root/cred

username=u3

password=123

:wq!

vi /etc/fstab

//172.25.X.11/share1 /test cifs credentials=/root/cred,multiuser,sec=ntlmssp 0 0

mount -a

df -h

To check the shared directory (in client side in) test

cd test

ls -l

NFS (Network file System)

The purpose of NFS is to share file Systems under a network of machines running with heterogenous environment

Advantages

* All Clients can access shared fs simultaneously
* Reduce admin over heads

Note: The protocol, which is responsible of communicate between NFS Server and NFS client is Rpc (Remote procedure Call) and port no is 2049.

NFS Versions - 2, 3, 4 (present version in lab V4.2)

Difference between version 2 and version 3

V2	V3
* maximum size of file System we can share is 2gb	* There is no limit
* Data waiting period between NFS Ser & NFS Client is 30sec	* It is less than 30 Sec
* Data accessing speed between NFS Server & NFS Client is 8 kps	* It is increased 4 times i.e 32 kpbs

Enhancements of NFS version 4

* Improved firewall port (2049)
* Stateful Connection (No RPC problems are faced)
* Easy to administration and trouble shoot

Scenario

/data → 172.25.X.10 [r] 172.25.3.0/24 [rw]

1. Install Package
2. Restart & enable Services
3. firewall
4. Create directory and file System
5. Change ownership to nfsnobody (if 'w' permission is required)
6. NFS Config (export)
7. exporting
8. Restart Services

```
# yum install -y nfs*
# System ctl restart nfs-Server
# System ctl enable nfs-server
# firewall-cmd --permanent --add-service = nfs
# firewall-cmd --reload
```

If directory

#mkdir /data (or)

If partition

fdisk /dev/vdb

: 3G

partprobe

mkfs.ext4 /dev/vdb1

mkdir /data

mount /dev/vdb1 /data

vim /etc/exports

/data 172.25.X.10 (ro, sync) 172.25.X.0/24 (rw, sync)

: wq!

exportfs -r

systemctl restart nfs-server

touch /data/f{1..}

chown nfsnobody /data // change the permission to use & nfsnobody (ownership)

showmount -e // display the export list of current system

Client Side Configuration

yum install -y nfs-utils

mkdir /test

mount -t nfs 172.25.X.11:/data /test // Temporary

vim /etc/fstab // Permanent

172.25.X.11:/data /test nfs defaults 0 0

: wq!

mount -a

#mkdir d1 (X) -> Permission denied (or) Read only Authentication

Note: Instead of change ownership of nobody, give no_root_squash entry in exports file.

Step 5

(NFS Step) # vi /etc/exports

Srv Side Config (without changing ownership) /data 172.25.X.10(rw,sync) 172.25.X.0/24 (rw,sync,no_root_squash)

:wq!

wget :- It is used to downloading from http Server

wget http://classrooms.examples.com/abc

abc downloaded to pwd (present working directory)

wget -O /root/abc http://classroom.examples/abc

abc downloaded to under /root directory

wget -O /root/xyz http://classroom.examples/abc

abc downloaded to /root with new name xyz

Secure NFS

Scenario 172.25.X.11

/oracle -> 172.25.X.10 (r)

Server-Side

1. Install package
2. Download keys
3. Restart & enable services
4. firewall
5. Specify version in sysconfig file
6. Create directory & file systems
7. Change ownership (if w is required)
8. nfs Configuration (exports)
9. exportfs
10. Remote Service

Server Side Config

lab nfskrb5 setup // only lab setup. not for exam purpose

yum install -y nfs*

wget -O /etc/krb5keytabs/serverX.keytab ↱ System Subnet network

systemctl restart nfs-secure-server

systemctl enable nfs-secure-server

firewall-cmd --permanent --add-service=nfs

firewall-cmd --reload

vim /etc/sysconfig/nfs

RPCNFSDARGS = "-V 4.2" (line number is 13)

:wq!

mkdir /oracle

touch /oracle/f{1..5}

vi /etc/exports

/oracle 172.25.X.10(rw,sync,sec=krb5p)

:wq!

exportfs -r

systemctl restart nfs-secure-server

client side

install package

2. Download keys

3. Restart & enable service

4. Create directory

5. mount

lab nfskrb5 setup

yum install -y nfs-utils

wget -O /etc/krb5.keytab http://classroom.examples.com/pub/keytabs/desktopX.keytab

:wq!

systemctl declare nfs-secure

systemctl enable nfs-secure

mkdir /test

Temp

mount -t nfs 172.25.X.11:/data /test

Permanent

vim /etc/fstab

172.25.X.11:/oracle /test nfs defaults,sec=krb5 0 0

:wq!

mount -a

df -h

cd /test

ls ✓

mkdir d1 ✗

LDAP

The purpose of ldap is to provide the user authentication. If we have ldap server without Nfs. The user can not get access to his file, so this case though the centralized authentication access is happening but no use. So mandatory (access is) ldap serve and Nfs server should be integrated.

ldap server should be take the responsibilities for centralized user with authentication where as Nfs take responsibility for sharing user home directory.

Note :- If you want to login as ldap user from any system that system has to be configured as ldap client under ldap server.

Server setup already installed in lab. So we have to configure and study for ldap client configuration.

Ldap client Configuration

#yum install -y authconfig-gtk sssd

#authconfig -gtk

Identity Authentication

User Account Configuration

User Account DB LDAP

LDAP Search base DN dc=example, dc=com

LDAP Server classroom.example.com

☑ use TLS to encrypt connections

Download CA Certificate

http://classroom.example.com/pub/example-ca.crt Authentication

Configuration

Authentication method - LDAP password

Apply

#systemctl restart sssd
systemctl enable sssd // enable & restart sssd service

getent passwd student ldapuser0
su - ldapuser0
-bash-4.2$ whoami
ldapuser0
-bash-4.2$ pwd
/root/desktop

If you are switching to ldap user, switching is happening but we are not able access the home directory. If you want to access the home dir. we have to install NFS client & autofs configuration.

Autofs

```
# showmount -e classroom.example.com
                    // To check export list of ldap server

# yum install -y autofs
# systemctl restart autofs
# systemctl enable autofs

# vi /etc/auto.master.d/new.autofs
    /home/guests  /etc/auto.misc

  :wq!

# vi /etc/auto.misc
  ldapusero  -rw  classroom.example.com:/home/guests/ldapusero

  :wq!

# systemctl restart autofs
# su -l ldapusero          // Switch to ldap user
```

Mounting all ldap users home directory

#vim /etc/autofs.misc

* -rw classroom.example.com:/home/guests/&

:wq!

systemctl restart autofs

NTP client

To synchroning the time between the server & client

[# timedatectl set-ntp [true (or) yes]]

timedatectl set-timezone Asia/Kolkata // change the time zone

timedatectl set-ntp true (or) yes // Change ntp enable

vim /etc/chrony.conf

put (#log) before # Server 0: —
Server 1: —
Server 2: —
Server 3: —
Server classroom.example.com iburst

:wq!

systemctl restart chronyd
systemctl enable chronyd
timedatectl
—— Asia/Kolkata (IST, +0530)
—— yes
—— yes

Providing Block Storage Iscsi

(Iscsi : Internet Small Computer System Interface protocol)

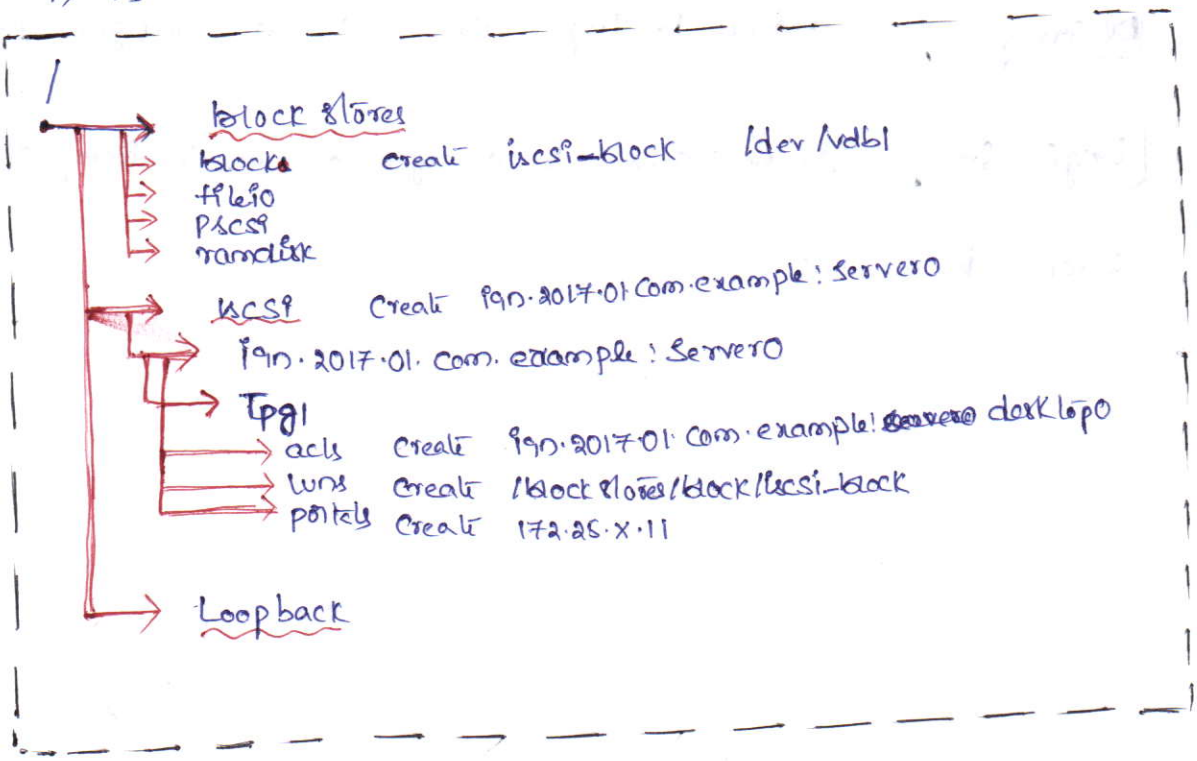
The iscsi is tcp/Ip based protocol used for emulating of scsi high performance local storage bus over IP networks.

Target :- It is Iscsi Server

Initiator :- It is Iscsi client

Iqn :- (Iscsi qualified name). It is world wide unique name used to identify by both initiator and target it should be in the format `iqn.yyyy-mm.com.<reverse domain>[:optional string]`

```
#targetcli
/> ls
```



Tpg :- Target Portal group

It is set of interface IP address tcp port to be which a specific iscsi target listen

Acl :- It is an access restriction using the node using the iqn to validate the access permission for an initiator

Portals :- An IP address and port an target & initiator used to establish Connection.

Lun :- Logical Unit Number

It is numbered block device attached to available through a target

Discovery :- queries a target server to list configure target

Login :- Authentication to a target & lun to being client block device use

Iscsi target Configuration

Server Configuration

1. Create Partition or Lvm
2. Install packages
3. Restart & enable service
4. firewall
5. Config
6. Restart

```
# fdisk /dev/vdb

    /dev/vdb1  + 5300m  → &e

# partprobe

# pvcreate  /dev/vdb1
# vgcreate  iscsi_vg  /dev/vdb1
# lvcreate -L 5G  -n  iscsi_lv  iscsi_vg

# yum  install -y  targetcli
# systemctl  restart  target
# systemctl  enable  target
# firewall-cmd  --permanent  --add-port=3260/tcp
# firewall-cmd  ---reload
```

#targetcli

/> ls

/>cd backstores/block

/backstores/block> Create iscsi_block /dev/iscsi-vg/iscsi-lv

/> ls

/> cd /iscsi

/iscsi> Create iqn.2017-01.com.example:servero

/iscsi> cd iqn.2017-01.com.example:servero/tpg1/acls

/iscsi/iqn.2017-01.com.example:servero/tpg1/acls> Create iqn.2017-01.com.example:desktop0

/iscsi/iqn.2017-01.com.example:servero/tpg1/acls> cd ../luns

/iscsi/iqn.2017-01.com.example:servero/tpg1/luns> Create /backstores/block/iscsi_block

/iscsi/iqn.2017-01.com.example:servero/tpg1/luns> cd ../portals

/iscsi/iqn.2017-01.com.example:servero/tpg1/portals> Create 172.25.x.11

/iscsi/iqn.2017-01.com.example:servero/tpg1/portals> cd /

/> ls

/> saveconfig

/> exit

systemctl restart target

iscsi Client Configuration

1. Install package
2. Restart service & enable service
3. set initiator name (acl)
4. Discovery
5. restart service
6. Login
7. Create partition & mount
8. logout // one time
9. ~~Re~~ Reboot

yum install -y iscsi-initiator-utils

systemctl restart iscsid

systemctl enable iscsid

vim /etc/iscsi/initiatorname.iscsi

InitiatorName = iqn.2017-01.com.example:desktop

:wq!

man iscsiadm

iscsiadm --mode discovery --type sendtargets --portal

172.25.X.11 --discover

systemctl restart iscsid

iscsiadm --mode node --targetname iqn.2017-01.com.example.server0 --portal 172.25.X.11:3260 --login

-- Success

#

fdisk /dev/sda

+1G

:w

partprobe

mkfs.xfs /dev/sda1

mkdir /sun

vi /etc/fstab

/dev/sda1 /sun xfs _netdev 0 0

:wq!

mount -a

df -h

iscsiadm --mode node --targetname iqn.2017-01.com.example.server0 --portal 172.25.X.11:3260 --logout

reboot

Apache web Server

(http) (httpd) – deamon

http → 80

https → 443

config file → /etc/httpd/conf/httpd.conf

→ /etc/httpd/conf.d/http-vhost.conf

Document Root → /var/www/html

* Ip based web Server : http://ServerX.example.com
* Name based web Server : http://wwwX.example.com (virtual)
* Sub Pages

http://ServerX.example.com/secret

* protect the webserver from the System

http://ServerX.example.com/secret

↓

① only ServerX.com can access

② All can access except ServerX

https

https://ServerX.example.com (or)

Dynamic webpages

http://WebappX.example.com:5543

* Package
* Restart & enable
* firewall
* Create file
* Config
* Restart service

* ip based web server

#yum install -y httpd*

#systemctl restart httpd
#systemctl enable httpd

#firewall-cmd --permanent --add-service=http

#firewall-cmd --reload

#vim /var/www/html/index.html

***** ip based web server ******

:wq!

#vim /etc/httpd/conf.d/Server.conf

~~<VirtualHost>~~

<VirtualHost 172.25.X.11:80>
ServerAdmin root@serverX.example.com
DocumentRoot /var/www/html
ServerName ServerX.example.com

</VirtualHost>

:wq!

Systemctl restart httpd

firefox
↓
localhost/manual
↓
user's Guide
↓
Config Section
↓
virtual host

* Name based web server (virtual)

mkdir /var/www/virtual
vi /var/www/virtual/index.html

*** Name based web server ***

:wq!

cp /etc/http/conf.d/server.conf /etc/httpd/conf.d/www.conf

vim /etc/http/conf.d/www.conf

<VirtualHost 172.26.X.11:80>
ServerAdmin root@server0.example.com
DocumentRoot /var/www/virtual
ServerName wwwX.example.com

</VirtualHost>
:wq!
Systemctl restart httpd

* Subpage

#mkdir /var/www/html/secret

vi /var/www/html/secret/index.html

**** Secret page ****

:wq!

* Protecting web pages

Only ServerX Can Server

#vim /etc/httpd/conf.d/server.conft

<VirtualHost>

.....

</VirtualHost>

<Directory /var/www/html/secret>

Require ip 172.25.X.11

</Directory>

:wq!

Systemctl restart httpd

* All Can access except ServerX

#vim /etc/httpd/conf.d/server.conf

<VirtualHost>

.....

</VirtualHost>

<Directory /var/www/html/secret>

Order Allow,Deny

Allow from all

Deny from 172.25.X.11

</Directory>

:wq!

#Systemctl restart httpd

Https

http + TLS/SSL

↓

Certs & keys

2 Certs & key

/etc/pki/tls/Certs

/etc/pki/tls/private

#yum install -y mod_ssl

#firewall-cmd --permanent --add-service=https

#firewall-cmd --reload

#cd /etc/pki/tls/Certs

#wget http://classroom.example.com/pub/example-ca.crt

#wget http://classroom.example.com/tls/Certs/ServerX.crt

#cd ../private

#wget http://classroom.example.com/pub/tls/private/ServerX.key

#vim /etc/httpd/conf.d/Server.conf

<VirtualHost>

</VirtualHost>

<Directory>

</Directory>

<VirtualHost 172.25.X.0:443>

DocumentRoot /var/www/html/

ServerName ServerX.example.com

SSLEngine on

SSLProtocol all -SSLV2 -SSLV3

SSLCipherSuite HIGH:MEDIUM:!aNULL:!MD5

SSL HandShake Operation Order on

SSLCertificateFile /etc/pki/tls/certs/ServerX.crt

SSLCertificateKeyFile /etc/pki/tls/private/ServerX.key

SSLCertificateChainFile /etc/pki/tls/certs/example-ca.crt

</VirtualHost>

Open another teb to find above all paths

#vim /etc/httpd/conf.d/ssl.conf

Copy SSL Engine on

Next phara last line (7)

2 cert & key

:wq!

#systemctl restart httpd

* Dynamic Page *

yum install -mod-wsgi

firewall-cmd --permanent --add-port=5542/tcp

firewall-cmd --reload

ls /home/student
webapp.wsgi

mkdir /var/www/html/new

cp /home/student/webapp.wsgi /var/www/html/new

ls /var/www/html/new

chown apache:apache /var/www/html/new/webapp.wsgi

vim /etc/httpd/conf.d/webapp.conf

Listen 5542

<VirtualHost 172.26.X.11:5542>

ServerName webappx.example.com

WSGIScriptAlias / /var/www/html/new/webapp.wsgi

</VirtualHost>

:wq!

Semanage port |grep http

man semanage-port

[Example:- Semanage port -a -t http_port_t -p tcp 81]

Semanage port -a -t http_port_t -p tcp 5542

Systemctl restart httpd

* Mariadb *

yum install -y mariadb*
Systemctl restart mariadb
Systemctl enable mariadb
firewall-cmd --permanent --add-service=mysql
firewall-cmd --reload

* How to secure mariadb installation

mysql_secure_installation

Enter
Current root pwd
Set root passwd (Y/N) (Y)
New password : ****
Re-enter password : ****

* Remove anonymous users? (Y/N) (Y)
* Disallow root login remotely? (Y/N) (Y)
* Remove test database and access to it [Y/N] (Y)
* Reload privilege table now? [Y/N] (Y)

* login to Mariadb *

mysql -u root -p

Enter the password : ****

- MariaDB [none] >
- MariaDB [none] > show databases; // To display all databases in maria data base

-M > Create database rhce;

// To Create new database

-M > use rhce;

// To Change & enter into the database

-M> Show tables;

// To display all the table in database

-M > Create table student (id int, name varchar (20),
place varchar (20));

-M > describe student // To display the fields of student table

-M> Insert into student (id, name, place) values (100, 'tom', 'Delhi');

// Inserting data int student table

-M> insert into student (id, name, place) values (101, 'tony', 'Hyd');

-m> Select * from student;

-m> Select name from student;

-m> Select name from student where place = 'Kolkatta';

-m> Select * from student where place = 'Kolkatta';

-m> Select place from student where id = 100;

* Taking dump / backup file *

```
# mysqldump -u root -p rhce > /root/rhce.dump
  Enter password: ↵
# ls /root
  .. rhce.dump
```

* Restoring the backup *

```
# mysql -u root -p
  new password : ****
M> Create database newrhce
m> exit
# mysql -u root -p newrhce < /root/rhce.dump
  Enter password: ****
```

*

User Name	Password	login type	privilege
tom	123	localhost	Select on student table in rhcedb
toy	123	any host	Select insert, delete and table in rhce db
sam	123	any host	All privilege on all table in all db.

M> Create user 'tom'@'localhost' identified by '123';

M> Create user 'jay'@'%' identified by '123';

M> Create user 'sam'@'%' identified by '123';

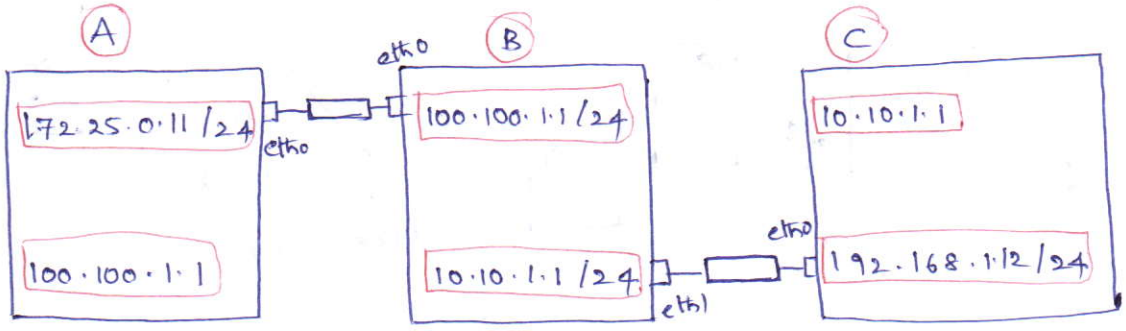
M> Grant select on mbc.student to 'tom'@'localhost';

M> Grant select, insert, update, delete on mbc.* To 'ray'@'%';

M> Grant All on *.* To 'sam'@'%';

m> flush privileges;

Masquerading



A

ping 172.25.X.11 ✓
ping 100.100.1.1 ✓
ping 192.168.1.2 X

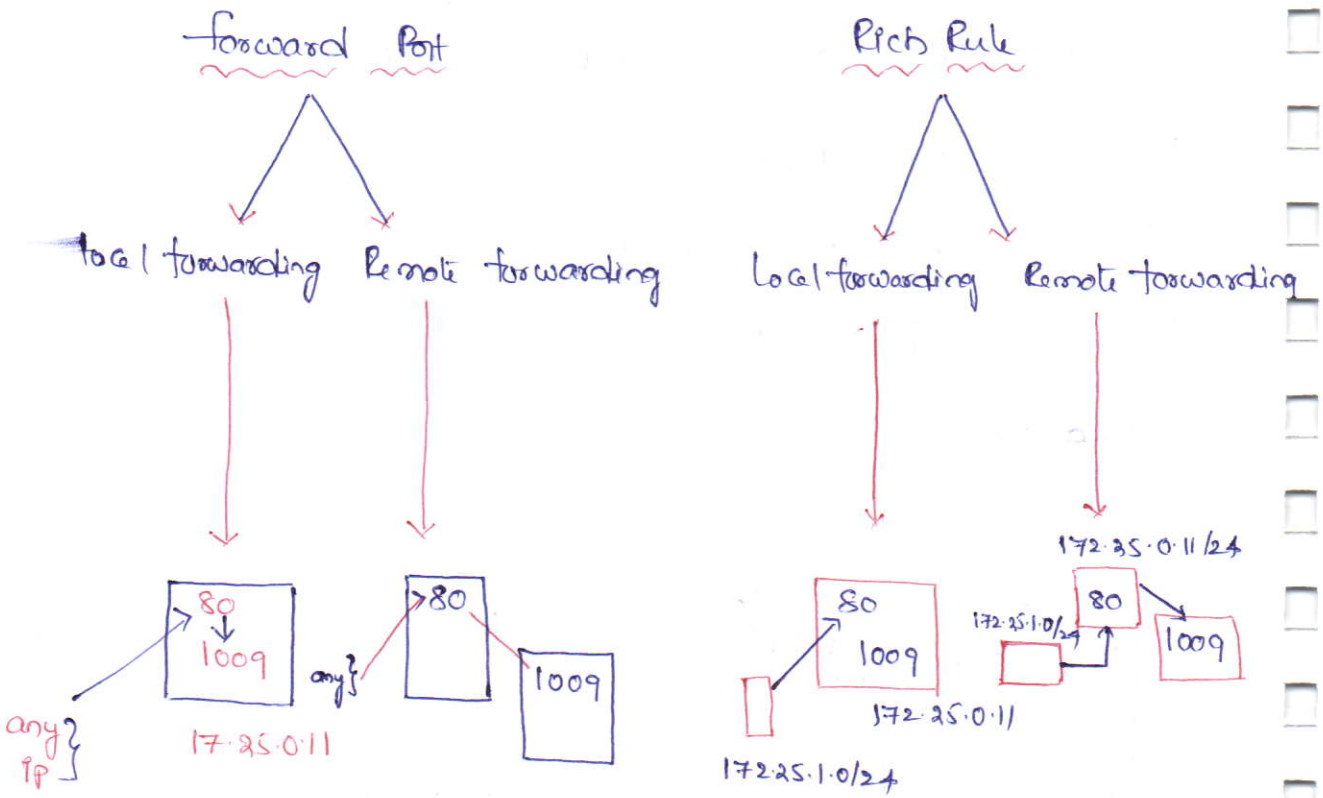
C

ping 10.10.1.1 ✓
ping 192.168.1.2 ✓
ping 172.25.X.11 X

firewall-cmd --permanent --add-masquerade
firewall-cmd --reload
firewall-cmd --list-all
masquerade: yes

Port forwarding : It is used to forward the traffic coming one port to another port

* forward port * Rich Rule



forward Port

local forwarding

firewall-cmd -- ~~permanent~~ Permanent --add-forward-port=port=1009:

Proto=tcp : toport=80

firewall-cmd --reload

firewall-cmd --list-all

Remove

firewall-cmd --permanent --remove-forward-port=

port=port=1009: proto=tcp: toport=80

forward Port

Remote forwarding

firewall-cmd --permanent --add-forward-port= port=1009:

proto=tcp: toport=80: toaddr=172.25.0.10

Remove

firewall-cmd --permanent --~~add~~remove-forward-port=

port=1009: proto=tcp: toport=80: toaddr=172.25.0.10

Rich Rule

Local forwarding

man firewalld.rich languages

example 5

firewall-cmd --permanent --add-rich-rule 'rule family

="IPV4" source address="172.25.1.0/24" forward-port ~~to=~~

to-port="80" protocol="tcp" port="1009" /

firewall-cmd --reload

firewall-cmd --list-all

Remove Rich Rule

Remote forwarding

firewall-cmd --permanent --add-rich-rule 'rule family = "IPV4" source address = "172.25.1.0/24" forward-port to-addr = "172.25.0.10" to-port = "80" protocol = "tcp" port = "1009" '

firewall-cmd --reload

firewall-cmd --list-all

Mail Server

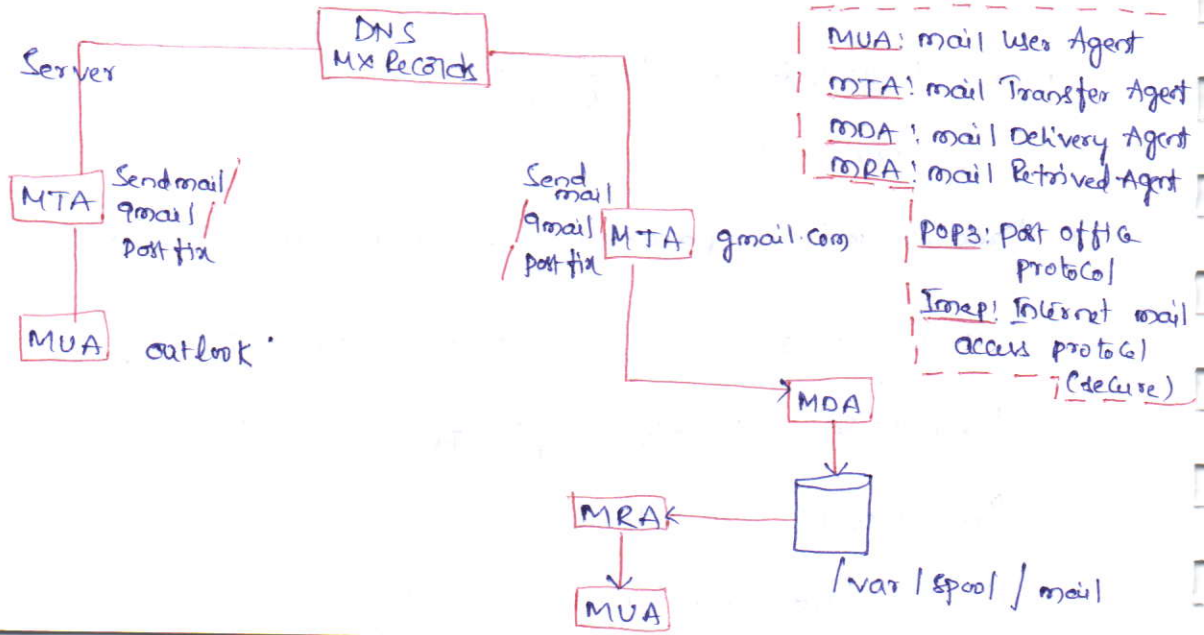
Send email

gmail

Postfix < mail gateway / Null client

email Server is used to Sending & Receive mails.

We Can Configure mail Server either using Send mail / gmail / postfix method. In Rhel7 by default we Can use postfix mail



Postfix

* mail gateway (We can send & receive mail)
* NULL client (We can send but we can't receive mail)

Mail gateway

```
# yum install -y postfix
# systemctl restart postfix
# systemctl enable postfix
# firewall-cmd --permanent --add-service smtp
# firewall-cmd --reload
# vim /etc/postfix/main.cf
```

```
myhostname = serverX.example.com          76 line (edit) remove # tg
mydomain = example.com                    -83
myorigin = $mydomain                      -98,99 remove # tg
inet_interfaces = all                     -116
inet_protocols = all
mydestination = localhost ....            -164
mynetworks = 172.25.X.0/24, 127.0.0.0/8,  -264
             [::1]/128
relayhost = [smtp.example.com]            314
:wq!
```

```
# systemctl restart postfix
```

* Send a mail to particular user

```
# mail student@serverX.example.com
Subject: Test                    [Subject]
This is atleast mail .....       [body]
.↵
EOT
```

su - student

student #mail

$1 (mail number)

$q

Null Client

yum install -y postfix

systemctl restart postfix

systemctl enable postfix

firewall-cmd --~~permanent~~ --add-service=smtp (Permanent)

firewall-cmd --reload

vim /etc/postfix/main.cf

myhostname = desktopX.example.com

mydomain = example.com

myorigin = $mydomain

inet_interfaces = loopback-only

inet_protocols = all

mydestination = 169 -empty

mynetworks = 127.0.0.0/8, (264)
[::1]/128

relay host = [smtp.example.com]

:wq!

systemctl restart postfix

root@desktop0 Desktop # mail student@desktop0.example.com

Subject : test

↵

Ks (Kick start)

If you want to install Ks Configuration through network to all clients below process

```
# yum install -y pykickstart
# yum install -y system-config-kickstart
# system-config-kickstart → Graphical window is opened
```

Kvm [Kernel virtualization Manager]

- It is used to create virtual machines.

```
# yum install -y virt-manager
# yum install -y libvirt-daemon-config-network
# virt-manager
```

Scripting

BASH Scripting

A bash shell script is simply an executable file. Composed of a list of commands.

The first line of a bash shell script begin with #! which is known as sharp-bang (sha-bang) or magic pattern.

It indicates that the file is an executable shell script the path name that follows is the command interpreter, the program that should be executable the script.

```
# vim script.sh
  #! /bin/bash
  echo "hello"
  :wq!
# ls -l script.sh
# chmod +X script.sh
# ./script.sh
# sh script.sh
```

Note: If you want to make as script as command copy script file into /bin

```
# cp script.sh /bin
# script.sh.
```

*1

```
# Vim  script_add
#!/bin/bash
    no1=10
    no2=20
    echo 'the sum is $[$no1 + $no2]'
:wq!
# ./sh  script_add
```

* Adding two numbers by taking from inputs

```
#Vim  script2.sh
#!/bin/bash
    echo "the first number is"
    Read a
    echo "the second number is"
    Read b
    echo "the sum is $[$no1 + $no2]"
    :wq!
#./sh  script2.sh
```

* greaterthan -gt
* greaterthan or equal -ge
* lessthan -lt
* lessthan or equal -le
* equal to -eq or ==
* not equal to -ne

```
#vim numbers.sh
#!/bin/bash
echo "1st no"
read a
echo "2nd no"
read b
if [ $a -gt $b ]; then
echo "$a is great than $b"
else
if [ $b -gt $a ]; then
echo "$b is great than $a"
else
echo "both are equal"
fi
fi
:wq!
#Chmod +X numbers.sh
```

Positional Parameters

Positional Parameters which stores the values of Command line arguments to a script.

* The var0 is Predefined with script name itself i.e $0
* The var1 is predefined with script name first arg. i.e $1
* The var2 Contains predefined with script name second arg i.e $2
* $* is used store entire args list i.e $*

* $# → It represent the total no of cmd arguments passed to the scripts. i.e $#

* How to use positional parameters

```
#Vim new.sh
#!/bin/bash
  If [ $# ==0]; then
  echo " usage: $0 <hi / foo>"
  else
  if [ $1 = = "hi" ]; then
  else echo "foo"
   else
   If [ $1 == "foo" ]; then
      echo "hi"
    else
      echo " $x not found"
   fi fi fi
  ;wq!
#Chmod +X new.sh
#./new.sh
  (a) #./new.sh hi
        foo
  (b)
      #./new.sh foo
        hi
  (c)
```

#./newsh fool abc
fool abc not found.

Ex:

```
# vim file1
  Sam
  tom
  roy
:wq!
# vim user.sh
  !/bin/bash
  If [ $# == 0 ]; then
  echo " usage: $0 <file name>"
  else
  If [ -f $* ]; then
  for user in $(cat $*);
  do
  done
  else
  echo "$* not found"
  fi
  fi
  :wq!
#Chmod +x user.sh
#./user.sh ↵
    ./user.sh < file name1
# ./user.sh file2
      file2 is not found
# ./user.sh file
# tail /etc/passwd
```