1. User and Group Administration

1. ***types of users available in Linux?***

   There are 5 types of users available in Linux.

   (i) System user (Admin user who control the whole system nothing but root user).
   (ii) Normal user (Created by the Super user. In RHEL - 7 the user id's from 1000 - 60000).
   (iii) System user (Created when application or software installed).
   In RHEL - 7 the System users are
   (!)Static system user id's from 1 - 200 and
   (ii) Dynamic system user user id'sfrom 201 - 999).
   (iv) Network user (Nothing but remote user, ie., who are login to the system trough network created in Windows Active Directory or in Linux LDAP or NIS).
   (v) Sudo user (The normal users who are having admin or Super user privileges)

   The types of users in Linux and their attributes:

Type of User	*Example*	*User ID*	*Group ID*	*Home Directory*	*Default Shell*
Super User	*Root*	*0*	*0*	*/root*	*/bin/bash*
Normal User	*ram, raju, gopal, ...etc.,*	*500 – 60000*	*500 - 60000*	*/home/<user name>*	*/bin/bash*
System User	*ftp, ssh, apache, nobody, ...etc.,*	*1 – 499*	*1 - 499*	*/vat/ftp, ...etc*	*/sbin/nolgin*
Network User	*Remote user like LDAP user*	*Same as normal users*	*Same as normal users*	*/home/guests/ldapuser*	*/bin/bash*
Sudo User	*Normal users with admin privileges*	*Same as normal users*	*Same as normal users*	*/home/<user name>*	*/bin/bash*

5. ***What are fields available in /etc/passwd file?***

   <user name> : x : <uid> : <gid> : <comment> : <user's home directory> : <login shell>
   *(where 'x' means link to password file ie., **/etc/shadow** file)*

6. ***What are fields available in /etc/shadow file?***

user name : password : last changed : min. days : max. days : warn days : inactive days : expiry days : reserved for future

7. ***What are the files that are related to user management?***

   - ***/etc/passwd*** *-----> Stores user's information like user name, uid, home directory and shell ...etc.,*
   - ***/etc/shadow*** *----> Stores user's password in encrypted form and other information.*
   - ***/etc/group*** *------> Stores group's information like group name, gid and other information.*
   - ***/etc/gshadow*** *---> Stores group's password in encrypted form.*
   - ***/etc/passwd-*** *---> Stores the /etc/passwd file backup copy.*

- ***/etc/shadow-*** *---> Stores the /etc/shadow file backup copy.*
- ***/etc/default/useradd*** *----> Whenever the user created user's default settings taken from this file.*
- ***/etc/login.defs*** *----> user's login defaults settings information taken from this file.*
- ***/etc/skell*** *------> Stores user's all environmental variables files and these are copied from this directory to user's home directory.*

9. What is the syntax of useradd command with full options?

# useradd -u <uid> -g <gid> -G <secondary group> -c <comment> -d <home directory> -s <shell><user name>

Example *: # useradd -u 600 -g 600 -G java -c "oracle user" -d /home/raju -s /bin/bash raju*

10. What is the syntax of adduser command with full options?

# adduser -u <uid> -g <gid> -G <secondary group> -c <comment> -d <home directory> -s <shell><user name>

Example *: # adduser -u 700 -g 700 -G linux -c "oracle user" -d /home/ram -s /bin/bash ram*

12. What is the syntax of userdel command with full options?

# userdel <options> <user name>

** The options are,*

-f -----> forcefully delete the user even through the user is login. The user's home directory, mail and message directories are also deleted.

-r -----> recursively means files in the user's home directory will be deleted and his home directory also deleted but the other files belongs to that user should be deleted manually.

13. How to check whether user is already created or not?

We can check in different ways.

(i) # id <user name> (It shows the user id group id and user name if that is already created)

(ii) # grep <user name> /etc/passwd

16. What is syntax of the usermod command with full options?

# usermod <options><user name>

** The options are,*

-L -----> lock the password
-U -----> unlock the password
-o ----->creates duplicate user modify the user's id same as other user
-u ----->modify user id
-g -----> modify group id
-G -----> modify or add the secondary group
-c -----> modify comment
-d -----> modify home directory
-s -----> modify user's login shell
-l -----> modify user's login name
-md ----> modify the users home directory and the old home directory

17. How to create the duplicate root user?

# useradd -o -u 0 -g root <user name>

19. What are the uses of .bash_logout, .bash_profile and .bashrc files?

.bash_logout : *is a user's logout ending program file. It will execute first whenever the user is logout.*

.bash_profile : *is user's login startup program file. It will execute first whenever the user is login. It consists the user's environmental variables.*

.bashrc : *This file is used to create the user's custom commands and to specify the umask values for that user's only.*

20. What is a group?

The collection of users is called a group. There are two types of groups.

***Primary group** : It will be created automatically whenever the user is created. User belongs to on group is called as primary group.*

***Secondary group :** It will not create automatically. The admin user should be created manually and users belongs to more than one group is called secondary group. A user can be assigned to max. 16 groups. ie., 1 primary group and 15 secondary groups.*

21. What is the command to check the user belongs to how many groups?

```
# groups    <user name>
```

22. What is the syntax to create the group?

```
# groupadd   <options><group name>
  The options are,        -f   -----> add the group forcefully
                          -g   -----> group id no.
                          -o   ----->non-unique (duplicate group id)
                          -p   -----> group password
                          -r   -----> system group
                          -R   -----> root group
```

23. What is the syntax to modify the group?

```
# groupmod   <options><group name>
  The options are,        -g  ------> group id
                          -n  ------> new name for existing one, ie., rename the group
                          -o  ------> non-unique (duplicate group id)
                          -p  ------> group passwd
                          -R  ------>root group
```

24. What is syntax to delete the group?

```
# groupdel    <group name>         (to delete the group without options)
# groupdel   -R   <group name>     (to delete the group and apply changes to the root directory)
```

25. How to assign the password to the group?

```
# gpasswd    <group name>          (to assign a password to the group without any options)
# gpasswd    <options><group name>
  The options are,        -a  ------>add users to the group
                          -d  ------>  delete the user from the group
                          -r  ------>  remove the group password
                          -R  ------>  restrict to access that group
                          -A  ------>  set the list of Administrative users
                          -M  ------>  set the list of group members
```

27. How to restore /etc/gshadow file if deleted by mistake?

*# grpconv (it creates the **/etc/gshadow** file from **/etc/group** file)*

28. How to change the password aging policies?

we can change the password policies in 2 ways.
*(i) First open the **/etc/login.defs** file and modify the current values.*
***Example :** # vim /etc/login.defs*

min - 0 -----> means the user can change the password to any no. of times.
min - 2 -----> means the user can change the password within 2 days. ie., he can change the password after 2 days.
max - 5 -----> means the user should change the password before or after 5 days. Otherwise the password will be expired after 5 days.
inactive - 2 -----> means after password expiry date the grace period another 2 days will be given to change the password.
warning - 7 -----> means a warning will be given to the user about the password expiry 7 days before expiry date.

(ii) second by executing the ***# chage*** *command.*

Example : *# chage <options><user name>*

The options are,
-d -----> last day
-E -----> expiry date
-I -----> inactive days
-l -----> list all the policies
-m -----> min. days
-M -----> max. days
-w -----> warning days

Note : *Whenever we change the password aging policy using* ***# chage*** *command, the information is will be modified in* ***/etc/shadow*** *file.*

30. *Explain the sudo user?*

Sudoers (nothing but sudo users) allows particular users to run various root user commands without needing a root password.

/etc/sudoers *is the configuration file for sudoers to configure the normal user as privileged user.*
It is not recommended to open this file using ***# vim*** *editor because this editor cannot check the syntax by default and whatever we typed in that file that will blindly save in this file.*

So, one editor is specially available for opening this file, i.e., ***# visudo*** *and all normal users cannot execute this command. Only root user can run this command.*

Once this file is opened nobody can open this file again on another terminal because ***"The file is busy"****message is displayed on the terminal for security reasons.*

31. *How to give different sudo permissions to normal users?*

Open the ***/etc/sudoers*** *file by executing*
#visudo *command and go to line no. 98 and type as*

```
 <User name>      <Machine>=        <Command>
    root               ALL=(ALL)           ALL
    raju               All=                ALL
  :wq    ----Save and exit this file.
```

32. ***In which location the sudo user commands history is logged?***

All the sudo users commands history is logged in ***/var/log/secure*** *file to make a record of sudo user commands.*

```
# cat   /var/log/secure       (to see the contents of this file)
# tailf  /var/log/secure      (to see the updates of this file continuously and press ctrl + c to quit the tailf)
```

<u>The other useful commands</u> :

```
# w        (this command gives the login user information like how many users currently login and full information )
# who                (to see users who are currently login and on which terminal they login)
# last               (see the list of users who are login and logout since the  /var/log/wtmp  file was created)
# lastb              (to see the list of the users who tried as bad logins)
# lastreboot         (to see all reboots since the log file was created)

# uptime   (to see the information from how long the system is running, how many users login and load  average)
   *  The load average is from   1 sec      :  5 secs  :  15 secs

# df                 (to see the mounted partitions, their mount points and amount of disk space)
# du                 (to see the disk usage of the each file in bytes)
# uname   -r         (gives the current kernel version)
# last   -x          (It shows last shutdown date and time)
# last   -x   grep  shutdown          (only shutdown time shows ie., grep will filter the 'last  -x'  command)
  *  grep:  It is used to search a word or sentence in file (ie., inside the file)
  *  find :   It is used to search a command or file inside the system)
# cat  /etc/shells   or  # chsh  -l      (to see how many shells that are supported by Linux)
          /bin/sh             ----->   default shell for Unix
          /bin/bash           ----->   default shell for Linux
          /sbin/nologin       ----->   users cannot login shell
          /bin/tcsh           ----->   c  shell to write  'C++'  language programs

# history                      (to see the history of the commands)
#history     -c                (to clear the history)
# history    -r                (to recover the history)
*   .bash_history   is the hidden file to store the history of the user commands. By default history size is 1000.

# whatis    <command>          (to see the short description of that command)
# whereis            <command>         (to see the location of that command and location of the
I                    document of that command)
# whoami                       (to see the current user name)
# passwd     <user name>       (to change the password of the user)
# id                           (to see the current user name, user id, group name and group id, .... etc.,)
# id   <user name>             (to see the specified user name, user id, group name and group id)
# su                           (to switch to root user without root user home directory)
# su   -                       (to switch to root user with root user home directory)
# su    <user name)            (to switch to the specified user without his home directory)
# su    -    <user name>       (to switch to the specified user with his home directory)
# du   -sh   /etc/              (to see the size of the   /etc   on the disk in KBs or MBs)
# ls   -l                       (to see the long listing of the files and directories)
 d  rwx  rwx  rwx   .  2   root    root    6     Dec 17  18:00     File name
 d       ----->  type of file
 rwx   ----->   owner permissions
 rwx   ----->   group permissions
 rwx   ----->  others permissions
 .       ----->  No ACL permissions applied
 root  ---->  owner of the file
 root  ----> group ownership
 6    ----->  size of the file
 Dec 7  18:00    ----->  Date and Time of the created or modified
 File name              ----->  File name of that file

# ls   -ld   <directory name>          (to see the long listing of the directories)
# stat   <file name/directory name>             (to see the statistics of the file or directory)
```

**35.	What are permission types available in Linux and their numeric representations?**

There are mainly three types of permissions available in Linux and those are,

```
read       -----   r   -----   4           null permission    ------  0
write      -----   r   -----   4
execute  -----   r   -----   4
```

Permissions	File	Directory
r	Read a file Ex. # cat <file name>	Read a directory contents Ex. ls /dir
w	Create, delete or modify the file contents	Create, delete or modify the files in a directory
x	Not required for file. It is required only for scripting files	Go to inside the directory Ex. # cd /dir

**36.	What is syntax of chmod command with full options?**

```
# chmod  <options><file/dir  name>  (to change the owner or permissions of the file/dir)
 The options are,  -c  ----->  changes
                   -f  ----->   silent  (forcefully)
                   -v  ----->  verbose
                   -R  ----->   recursive  (including sub directories and files)
To change the permissions the syntax is,
# chmod                      <who>   <what>  <which>            <file name or directory>
                             user (u)  add (+)    read (4)  or  (r)                    "
                             group(g)remove(-) write (2)  or  (w)                      "
                             other (o)equal (=)   execute  (1)  or  (x)                "
```

**37.	What is the syntax of chown command with full options?**

```
# chown   <options><file  name  or  directory>          (to change the ownership of the file or directory)
  The options are,           -c    ----->    changes
                             -f    ----->   silent  (forcefully)
                             -v   ----->    verbose
                             -h   ----->    no difference
                             -R   ----->    recursive  (including sub directories and files)
                             -H  ----->    symbolic link to a directory   (command line argument)
```

-L -----> symbolic link to a directory (all)
-p -----> do not traverse

# chown <username> : <group name> <file name or directory name> (to change owner and group ownership of the file or directory)

38. What is syntax of chgrp command with full options?

# chgrp <options><file name or directory> (to change group ownership of the file directory)

The options are, -c -----> changes
-f -----> silent (forcefully)
-v -----> verbose
-h -----> no difference
-R -----> recursive (including sub directories and files)
-H -----> symbolic link to a directory
-L -----> do not traverse-p -----> do not traverse

39. What are the default permissions of a file and directory?

The default permissions of a file = 6 6 6
The default permissions of a directory = 7 7 7

44. Explain about sticky bit?

It protects the data from other users when all the users having full permissions on one directory.
It can be applied on others level and applicable for directories only.

Example : *# chmod o+t <directory name> (to set the sticky bit permission on that directory)*
# ls -ld <directory name>
r w x r w x r w t <directory name>(where 't' is called the sticky bit)

47. Can we login to the user without password?

Yes, we can login.

49. How to restrict the users from login?

(i) By removing (deleting) the user we can restrict the user from login.
*(ii) Put the user's hostnames as entries in **/etc/hosts.deny** file (applying TCP wrappers).*
*(iii) **#passwd -l <user name>** (by locking his password we can restrict the users).*

50. How to put never expiry to a user?

# passwd -x -1 <user login name>

51. Which one is the default sticky bit directory?

***/tmp** is the default sticky bit directory.*

53. Can we mount/unmount the O/S file system?

No, we cannot mount or unmount the O/S file system.

54. How to find the users who are login and how to kill them?

# fuser -cu (to see who are login)
#fuser -ck <user login name> (to kill the specified user)

60. What is the syntax to assign read and write permissions to particular user, group and other at a time?

# setfacl -m u : <user name> : <permissions>, g : <user name> : <permissions>, o : <user name> : <permissions><file or directory>

<u>Useful commands</u> *:*

# setfacl -x u : <user name><file or directory name> (to remove the ACL permissions from the user)
# setfacl -x g : <user name><file or directory name>(to remove the ACL permissions from group)
# setfacl -x o : <user name><file or directory name> (to remove the ACL permissions from other)
# setfacl -b <file or directory> (to remove all the ACL permissions on that file directory)

2. Managing Partitions and File Systems

1. ***What is partition?***

A partition is a contiguous set of blocks on a drive that are treated as independent disk.

2. ***What is partitioning?***

7. ***What is file system?***

It is a method of storing the data in an organized fashion on the disk. Every partition on the disk except MBR and Extended partition should be assigned with some file system in order to make them to store the data. File system is applied on the partition by formatting it with a particular type of file system.

8. ***What are the different types of file systems supported in Linux?***

- *The Linux supported file systems are ext2, ext3, ext4, xfs, vfat, cdfs, hdfs, iso9660 ...etc.,*
- *The ext2, ext3, ext4 file systems are widely used in RHEL-6 and xfs file system is introduced on RHEL-7.*
- *The vfat file system is used to maintain a common storage between Linux and Windows O/S.*
- *The cdfs file system is used to mount the CD-ROMs and the hdfs file system is used to mount DVDs.*
- *The iso9660 file system is used to read CD/DVD.iso image format files in Linux O/S.*

9. *How to create different types of partitions?*

```
# fdisk -l
# fdisk /dev/sdc
Command (m for help) : n          (type n for new partition)
(p - primary) or e - extended) : p          (type p for primary partition or type e for extended partition)
First cylinder : (press Enter for default first cylinder)
Last cylinder : + <size in KB/MB/GB/TB>
Command (m for help) : t       (type  t  to change the partition id)
(for example:   8e for Linux LVM,   82 for Linux Swap   and 83 for Linux normal partition)
          Command (m for help) : w      (type  w  to save the changes into the disk)
# partprobe /partx  -a/kpartx  /dev/sdc1      (to update the partitioning information in partition table)
```

10. ***How to make a file system in Linux?***

```
# mkfs.ext2/ext3/ext4/xfs/vfat       <device name> ( for example/dev/sdc1)
```

11. ***How to mount the file systems temporarily or permanently?***

```
# mkdir /mnt/oracle
# mount /dev/sdc1  /mnt/oracle   (temporary mount)
# vim /etc/fstab
/dev/sdc1                  /mnt/oracle            xfs             defaults 0          0
Esc+:+wq!
# mount  -a   (permanent mount)
```

12. ***How to delete the partition?***

```
# fdisk  /dev/sdc
          Command (m for help) :d        (type  d  for delete the partition)
          Partition number :  (specify the partition number)
          Command (m for help) : w     (type  w  to write the changes into disk)
# partprobe/partx  -a/kpartx  /dev/sdc1(to update the partition table without restarting the system)
```

13. What is mounting and in how many types can we mount the partitions?

Attaching a parititon to a directory under root is known as mounting.
There two types of mountings in Linux/Unix.

- ❖ *Temporary Mounting :*
  In a temporary mounting first we create a directory and mount the partition on that directory. But this type mounting will last only till the system is up and once it is rebooted the mounting will be lost.
  Example:# mount <options><device name><directory name (mount point)>
- ❖ *Permanent Mounting :*
  In this also first we create the directory and open the /etc/fstab file and make an entry as below,
  <device name><mount point><file system type><mount options><take a backup or not><fsck value>
  Whenever the system reboots mount the partitions according to entries in /etc/fstab file. So, these type of mountings are permanently even after the system is rebooted.
  # mount -a to mount the partitions without reboot)

14. What are differences between the ext2, ext3, ext4 and xfs file systems?

S.No.	**Ext2**	**Ext3**	**Ext4**
1.	Stands for Second Extended file system.	Stands for Third Extended file system.	Stands for Fourth Extended file system.
2.	Does not having Journaling feature.	Supports Journaling feature.	Supports Journaling feature.
3.	Max. file size can be from 16 GB to 2 TB.	Max. file size can be from 16 GB to 2 TB.	Max. file size can be from 16 GB to 16 TB.
4.	Max. file system size can be from 2 TB to 32 TB	Max. file system size can be from 2 TB to 32 TB	Max. file system size can be from 2 TB to 1 EB *1EB = 1024 Peta bytes.

15. Which files are related to mounting in Linux?

- ➢ ***/etc/mtab*** *----> is a file which stores the information of all the currently mounted file systems and this file is dynamic and keep on changing.*
- ➢ ***/etc/fstab*** *----> is keeping information about the permanent mount points. If we want to make our mount point permanent then make an entry about the mount point in this file.*

/etc/fstab entries are:

1	2	3	4	5	6
device name	mount point	F/S type	mount options	Dump	FSCK

16. The partitions are not mounting even though there are entries in /etc/fstab. How to solve this problem?

F*irst check any wrong entries are there in /etc/fstab file. If all are ok then unmount all the partitions by executing the below command,*

- ✓ *# umount -a*

Then mount again mount all the partitions by executing the below command,

- ✓ *# mount -a*

17. When trying to unmounting it is not unmounting, how to troubleshoot this one?

Some times directory reflects error while unmounting because,
(i) you are in the same directory and trying to unmount it, check with ***# pwd****command.*
(ii) some users are present or accessing the same directory and using the contents in it, check this with

- ✓ *# fuser -cu <device name> (to check the users who are accessing that partition)*
- ✓ *# lsof <device name> (to check the files which are open in that mount point)*

✓ # fuser -ck <opened file name with path> (to kill that opened files)

Now we can unmount that partition using # umount <mount point>

18. How to see the usage information of mounted partitions?

```
# df   -hT      (to see device name, file system type, size, used, available size, use% and mount point)
```

19. How to see the size of the file or directory?

```
#du –sh <file 0r folder name>
```

23. What is the basic rule for swap size?

(i) If the size of the RAM is less than or equal to 2GB, then the size of the swap = 2 X RAM size.
(ii) If the size of the RAM is more than 2GB, then the size of the swap = 2GB + RAM size.

24. How to create a swap partition and mount it permanently?

```
# free   -m       (to see the present swap size)
# swapon  -s     (to see the swap usage)
# fdisk   <disk name>         (to make a partition)
Example:  # fdisk   /dev/sdb
Command (m for help) :  n    (to create a new partition)
First cylinder :  (press Enter to take as default value)
Last cylinder :  +2048M   (to create 2GB partition)
Command (m for help) :   t     (to change the partition id)
Enter the partition No.:  2      (to change the /dev/sdb2 partition id)
Enter the id :  82    (to change the partition id Linux to Linux Swap)
Command (m for help) :  w    (to save the changes into the disk)
# partprobe  /dev/sdb        (to update the partition table information)
# mkswap  <device or partition name>     (to  format the partition with swap file system)
Example :  # mkswap   /dev/sdb2        (to format the /dev/sdb2 partition with swap file system)
# swapon    <device or partition name>     (to activate the swap space)
Example :  # swapon   /dev/sdb2       (to activate /dev/sdb2  swap space)
# free   -m       (to see the swap size)
# vim /etc/fstab       (to make an entry to permanent mount the swap partition)
/dev/sdb2                       swap      swap      defaults  0           0
Esc+:+wq!     (to save and exit)
```

29. Why the file system should be unmount before running the fsck command?

*If we run **fsck** on mounted file systems, it leaves the file systems in unusable state and also deletes the data. So, before running the **fsck** command the file system should be unmounted.*

30. Which type of file system problems you face?

(i) File system full
(ii) File system corrupted

31. How to extend the root file system which is not on LVM?

*By using **# gparted** command we can extend the root partition, otherwise we cannot extend the file systems which is not on LVM.*

32. How to unmount a file system forcefully?

```
# umount    -f    <mount point>
# fuser    -ck    <mount point>
```

33. How to know the file system type?

```
# df  -hT      (command gives the file system type information)
```

34. How to know which file system occupy more space and top 10 file systems?

```
# df  -h  <device or partition name>  | sort  -r  |  head  -10
```

35. What is the command to know the mounted file systems?

```
# mount   or  # cat  /etc/mtab
```

36. How to know whether the file system is corrupted or not?

First unmount the file systems and then run ***fsck*** *command on that file system.*

38. How to create a file with particular size?

```
# dd if=/dev/zero  of=/orafile   bs=1MB   count=500 (to create  500MB size  /orafile with 4KB blocksize)
```

39. How to find how many disk are attached to the system?

```
# fdisk   -l    (to see how many disk are attached to the system)
```

42. How to create the file systems with the user specified superblock reserve space?

```
# mkfs.ext4   -m  <no.><partition  name>       (to format the partition with <no.>% of reserve space to superblock)
```

Whenever we format the file system, by default it reserve the 5% partition space for Superblock.

Important Commands **:**

```
✓ # fsck   <partition name>             (to check the consistency of the file system)
✓ # e2fsck  <partition name>            (to check the consistency of the file system in interactive mode)
✓ # e2fsck  -p   <partition name>       (to check the consistency of the file system without interact mode)
```

```
# umount  -a                  (to unmount all the file systems except ( / ) root file system)
# mount  -a                   (to mount all the file systems which are having entries in /etc/fstab file)
# fsck  -A                    (to run fsck on all file systems)
# mount  -o  remount, rw   /dev/sdb1       (to mount the partition with read and write permissions)
# mount  -o  remount, ro   /dev/sdb1       (to mount the partition with read only permissions)
# mount  < directory name>                  (to check whether this directory is mount/ normal directory)
# fdisk  -l                                (to list total hard disks attached to system and their partitions)
# fuser  -cu  <device or partition name>    (to see the users who are accessing that file system)
# fuser  -cK  <device or partition name>    (to kill the users processes who accessing the file systems)
```

Note: *Even though we kill those users processes sometimes we cannot unmount those partitions, so if this situation arises then first see the process id's of the user opened files by*

```
# lsof   <mount point>
# kill   -9   <process id> kill those  processes forcefully
```

3.Logical Volume Management and RAID Levels

1. What is LVM and why we go for LVM?

***Lvm** means Logical Volume Management. The combination of 2 or more physical disk in order to make a big logical disk is called Logical Volume.*

If normal Linux partition is full and an application requires some more disk space, then normal partition can not be extended for that application requirement. For this first we have to take a backup of that normal partition, delete that partition and again create that partition with more disk space, format and mount that partition and finally restore the application from the backup. This process requires down time.

So, to overcome this problem LVM concept is coming into the picture. Using this LVM we can extend or reduce the file systems as per requirement without loss of any data.

2. What are the components of the LVM?

Physical Volume (PV)
Physical Extent (PE)
Volume Group (VG)
Logical Volume (LV)
Logical Extent (LE)

<u>Physical Volume (PV)</u> :

*It is the standard partition that we add to the LVM. Normally a physical volume is a standard primary or logical partition with the partition code as **8e**.*

<u>Physical Extent (PE)</u> :

It is chunk of disk space. Every physical volume is divided into a number of equal sized PEs.

<u>Volume Group (VG)</u> :

It is composed of a group of physical volumes and logical volumes. It is the organizational group of LVM.

<u>Logical Volume (LV)</u> :

It is composed of a group of LEs. We can format (make a file system) and mount any file system on the logical volume. The size of these logical volumes can easily be increased or decreased as per the requirement.

<u>Logical Extent (LE)</u> :

It is also a chunk of disk space. Every logical extent is mapped to a specific physical extent.

3. How to create the LVM, make a file system and mount that permanently?

*(i) Take two physical disks for example **/dev/sdb** and **/dev/sdc**. if there is no second disk then make the required partitions using **# fdisk** command and change the partition code as **8e**.*

(ii) Convert the Physical disk into physical volumes by,

```
# pvcreate   /dev/sdb   /dev/sdc
```

(iii) Then create the volume group by combining physical volumes by,

```
# vgcreate  <volume group name><physical  volume names>   or
# vgcreate  -s  <PE size in MBs><volume  group name><physical  volume names>
```

(iv) Then create the logical volume on the above created volume group by,

```
# lvcreate  -L  +<size  in MBs>  -n <logical volume name><Volume  group name>  or
# lvcreate  -l  <no. of PEs>   -n  <logical volume name><volume  group name>
```

(v) Make a file system on the above created logical volume by,

```
# mkfs.ext2/ext3/ext4/xfs   /dev/<volume  group name>/<logical  volume name>
```

(vi) Create a mount point to mount the above created LVM file system by,

```
# mkdir   /mnt/<directory name>
```

(vii) Mount the LVM on the above created mount point temporarily by,

```
# mount   /dev/<volume  group name>/<logical  volume name><mount  point>or
```

Mount the LVM on mount point permanently by,

```
# vim  /etc/fstab
/dev/<VG name>/<LV name>        /mnt/<directory>           <file system type>          defaults    0       0
Esc+:+wq!
# mount   -a
# df   -hT       (to see the mounted partitions with file system types)
```

4. ***How to see the details of the Physical Volumes?***

```
# pvs                  (displays all physical volumes with less details)
# pvdisplay            (displays  all physical volumes with more details)
# pvdisplay  <physical volume name>          (displays the details of the specified physical volume)
# pvscan               (to scan all the physical volumes)
#pvscan    <PV name>       (to scan the specified physical volume)
```

5. ***How to see the details of the Volume Groups?***

```
# vgs                  (displays all volume groups with less details)
# vgdisplay                        (displays all volume groups with more details)
# vgdisplay  <VG name>    (displays the specified volume group with more details)
# vgscan                           (to scan all the volume groups)
# vgscan   <VG name>      (to scan the specified volume group)
```

6. ***How to see the details of the Logical Volumes?***

```
# lvs                  (displays all logical volumes with less details)
# lvdisplay                        (displays all logical volumes with more details)
# lvdisplay  <LV name>     (displays the specified logical volume details)
# lvscan               (to scan all the logical volumes)
# lvscan   <LV name>       (to scan the specified logical volume)
```

7. ***How to extend the Volume Group?***

Extending the volume group is actually adding a new physical volume to the volume group.

*To extend the volume group we need to create a new partition using **# fdisk** command and make sure that it's partition id should be **8e**, save the changes and update the partition table by **# partprobe***

*Create a physical volume on the newly created partition using **# pvcreate** command.*

*Add the partition to the volume group using **# vgextend** command*

```
Example :  # fdisk   /dev/sdb
                  Command (m for help) : n
                  First cylinder : press Enter for default one
                  Last cylinder : +500M                 (create 500MB partition)
                  Command (m for help) : t              (to change the partition id)
Select the partition : type the partition number
Specify the Hexa code : 8e
Command (m for help) : w             (to save the changes)
# partprobe   /dev/sdb1
# pvcreate   /dev/sdb1
# vgextend  <VG name>  /dev/sdb1
# vgdisplay  <VG name>   (to check the size of the volume group)
```

8. ***How to extend the logical volume and update it's file system?***

Sometimes the file system size may be full, so we need to increase the size of the logical volume to continue adding the data in it.

The size of the logical volume can be increased online, no downtime required.

*Check current size of the logical volume by **# lvdisplay <LV name>**and the size of the file system by **# df -hT** command.*

*Increase the size of the logical volume by **# lvextend or # lvresize** commands.*

*Then finally update the file system by **# resize2fs or # xfs_growfs** commands.*

```
Example :   # df   -hT
                  # lvextend   -L   +<size in MB></dev/vgname/lvname>     or
                  # lvresize   -L  +<size in MB></dev/vgname/lvname>
                  # resize2fs    </dev/vgname/lvname>
                  # lvdisplay  </dev/vgname/lvname>           (to check the size of the logical volume)
```

```
                # df    -hT                                 (to check the size of the file system)
```

14. How to restore the volume group which is removed mistakenly?

First unmount file system by **# umount <file system mount point>** command.
Check the volume group backup list by **# vgcfgrestore --list <volume group name>**command.
Then remove the logical volume by **# lvremove </dev/vgname/lvname>** command.
Copy the backup file which is taken backup before removed the volume group from the above backup list and paste it in this command **# vgcfgrestore -f <paste the above copied file name><vgname>**
The logical volume is created automatically after restoring the volume group but the volume group and logical volumes both will be in inactive state. So, check the state of the volume group by **#vgscan**and the logical volume state by **# lvscan** commands.
Then activate that volume group by **# vgchange -ay <volume group name>**commandand activate the logical volume by # **lvchange -ay <logical volume name>**command.
Mount the logical volume file system by **# mount -a** command.

```
Example :   # umount   <file system mount point>
            # vgcfgrestore   --list   <volume group name>       (copy the backup file from the list)
            # lvremove   </dev/vgname/lvname>
            # vgcfgrestore   -f   <paste the above copied file><volume group name>
            # vgscan                                   (to check the status of the volume group)
            # lvscan                                   (to check the status of the logical volume)
            # vgchange   -ay   <volume group name>     (activate the volume group if it is in inactive state)
            # lvchange   -ay   <logical volume name>   (activate the logical volume if it is in inactive state)
```

Note: The option **a** means active VG or LV and option **y** means yes.

```
            # mount   -a
```

19. What is the configuration file of the logical volume?

```
# cat  /etc/lvm/lvm.conf            (to see the contents of the LVM configuration file)
```

20. What are the locations of the logical volume and volume groups?

```
# cd  /etc/lvm/backup               (the logical volumes backup location)
# cd  /etc/lvm/archive              (the volume groups backup location)
```

21. How to know the current version of the LVM package?

```
# rpm  -qa   lvm*                   (to know the current version of the LVM package)
```

22. What are the attributes of the volume group?

```
# vgs                               (to see the attributes of the volume group)
```

23. How to extend the logical volume to max. disk space and half of the disk space?

```
# lvextend   -l  +100% FREE   <logical volume>        (to extend the logical volume by adding the
                                   volume group's       total available space)
```

30. What is RAID? What is the use of the RAID and how many types of RAIDs available?

RAID stands for Redundant Array of Independent Disks.
It provides fault tolerance, load balancing using stripping, mirroring and parity concepts.
There are mainly two types of RAIDs available.
(i) Hardware RAID (Depends on vendors and also more expensive)
(ii) Software RAID (Does not depends on vendors and less expensive when compared to Hardware RAID and also it is maintained by system administrator only.

31. How many types of software RAIDs available and their requirements?

(i) RAID - 0 ---- Stripping ---- Minimum 2 disks required
(ii) RAID - 1 ---- Mirroring ---- Minimum 2 disks required
(iii) RAID - (1+0) --- Mirroring + Stripping ---- Minimum 4 disks required
(iv) RAID - (0+1) --- Stripping + Mirroring ---- Minimum 4 disks required
(v) RAID - 5 ---- Stripping with parity ---- Minimum 3 disks required

43. What is a link file and how many types?

Link file is a short cut file to the original file. Creating and removing (deleting) inks between two files is known as managing links. There are two types of links files available in Linux.
(i)Soft link
(ii)Hard link

44. What is soft link and how to create it?

Soft link is nothing but a short cut file. If original file is deleted, no use of short cut file. ie., we cannot access the original data by selecting the link file. Soft link can be applied on both directories and files. These files can be stored in any of the file system. ie., the original file may be in one file system and the link file may be on another file system. If we edit any file, the link files are also updated automatically. When we create a soft link file, the permissions are full permissions. The soft link file and the original file inode no's are different. The size of the soft link file is same as the length of the original file name. The soft link can be created by

```
# ln    -s     <original file or directory><link file or directorywith path>          (to create a soft link)
# ln   -s   /root/script    /root/Desktop/script        (to create a link file for the script and stored on root Desktop)
```

45. What is hard link and how to create it?

Hard link in nothing but a backup file. If the original file is deleted, there is no effect on hard link file. i.e., we can access the original file data even though the link file is deleted. Hard links can be applied on files only not on directories. Hard link files can be stored in the same file system. ie., original and hard link files both should be in the same file system not on different file systems. The inode no's are same for original and hard link files. If the original is edited, the updations are applied on both original and hard link files. The size of the hard link file is same as the size of the original file.

46. What are the commands to search files and directories?

To search files and directories there are two commands.
(i) # locate
(ii) # find

47. Explain the locate command and how to use it?

locate always looks the locate database and not in a specific location. The data of the locate is stored in ***/var/lib/mlocate/mlocate.db*** *file. If the data is not updated in locate database or the locate database is available or locate database is deleted, we cannot locate the files and directories.* ***# updatedb*** *is the command to update the locate database. locate database cannot be find the newly created files and directories. It is not* *recommended to use on production servers because it impacts on performance of the servers. So, to overcome this problem we normally use* ***# find*** *command on production servers.*

```
# updatedb                              (to update the locate database)
# locate   <file name/directory name>   (to search the specified file or directory)
```

48. *Explain the find command and how to use it?*

find command required the specific location. Without specific location we cannot find the files or directories.

```
# find  <location><options><file  or  directory>        (to find the specific file or directory)
```

The options are,

-name	----->	search files and directories
-prem	----->	search for permissions
-size	----->	search for sizes
-user	----->	search for the owner
-uid	----->	search for files/directories of uid)
-gid	----->	search for files/directories of gid)
-group	----->	search for group owner
-empty	----->	search for empty files
-amin	----->	search for access time
-mmin	----->"	"
-cmin	----->"	"
-atime	----->	search for access day (access day, minutes, hrs, ...etc)
-mtime	----->	search for modify day (change the content)
-ctime	----->	search for change day (permissions, ...etc)

Examples *:*

```
# find  /  -name   <file name>                    (to search for file names in / directory)
# find  /  -name   <file name>  -type  f          (to find file names only)
# find  /  -name   <directory name>  -type  d     (to find directories with small letters only)
# find  /  -iname  <file/directory name>  -t  d   (to search for small or capital letter files/directories)
```

Network Configuration and Troubleshooting

1. *What is Network?*

Combination of two more computers connected together to share their resources each other by means of communication like cable is called Network.

2. *What is Networking?*

It is a connection between two or more computers to communicate with each other.

4. *Explain about NIC card?*

A Network Interface Card or controller is hardware component that connects a computer to a computer network. Each NIC card will be having MAC (Media Access Controller) address to avoid conflicts between same NIC adapters. In Linux these NIC adapter is represented by the word ***"eth"*** *. For example if two NIC cards are there in a system then it will be denoted as* ***"eho","eth1", ...****etc.,*

8. *What are the differences between TCP/IP and UDP protocols?*

TCP/IP	UDP
Transmission Control Protocol	User Datagram Protocol
It is connection oriented	It is connection less

Reliable	*Non-Reliable*
TCP Acknowledgement will be sent / received	*No Acknowledgement*
Slow communication	*Fast communication*
Protocol No. for TCP is 6	*Protocol No. for UDP is 17*
HTTP, FTP, SMTP, ...etc., uses TCP	*DNS, DHCP, ...etc., uses UDP*

9. What is an IP address?

Every Computer will be assigned an IP address to identify each one to communicate in the network. The IP address sub components are Classes of an IP address, Subnet masks and Gateway.

Classes of IP address *:*

The IP addresses are further divided into classes. The classes are A, B, C, D, E and the ranges are given below.

Class	*Start*	*End*	*Default Subnet mask*	*Classless Inter Domain Routing*
Class A	*0.0.0.0*	*127.255.255.255*	*255.0.0.0*	*/8*
Class B	*128.0.0.0*	*191.255.255.255*	*255.255.0.0*	*/16*
Class C	*192.0.0.0*	*223.255.255.255*	*255.255.255.0*	*/24*
Class D	*224.0.0.0*	*239.255.255.255*		
Class E	*240.0.0.0*	*255.255.255.255*		

10. What is loopback address?

A special IP number (127.0.0.1) is designated for the software loopback interface of a machine. 127.0.0.0 and 127.255.255.255 is also reserved for loopback and is used for internal testing on local machines.

11. What is multicasting?

Multicasting allows a single message to be sent to a group of recipients. Emailing and Teleconferencing are examples of multicasting. It uses the network infrastructure and standards to send messages.

14. What are important configuration files in network configuration?

# cat /etc/sysconfig/network *(This file keeps the information about the hostname assigned to the system and if we want to change the hostname permanently, we need to change the hostname in this file)*

# cat /etc/sysconfig/network-scripts/ *(This directory keeps the configuration of network devices connected to the system. Examples are* ***ifcfg-eht0, ifcfg-eth1, ifcfg-eth2,*** *.....etc.,)*

# cat /etc/hosts *(This file is responsible for resolving hostname into IP address locally. ie., local DNS if DNS server is not available)*

# cat /etc/resolve.conf *(This file keeps the address of the DNS server to which the clients will be accessing to resolve IP address to hostname and hostname to IP address)*

18. In how many ways we can configure the network?

There are two ways to configure the network.

(a) Static Network.

(b) Dynamic Network.

Static Network *:*

In this way we assign the IP address and hostname manually. Once we configure the IP address, it will not change.

Dynamic Network *:*

In this way we assign the IP address and hostname dynamically. This means the IP address will change at every boot.

19. How to assign the static IP address to the NIC card?

In RHEL - 6 *:*

# setup

(Move the cursor to Network configuration and press Enter key)

(Move the cursor to Device configuration and press Enter key)

(Select the NIC adapter ie., eth0 and press Enter key)
(Assign the above IP address and other details as per our requirements and move the cursor to "OK" and press Enter key)
(Move the cursor to "Save" to save the changes in device configuration and press Enter key)
(Once again move the cursor to "Save & Quit" button and press Enter key)
(Finally move the cursor to "Quit" button and press Enter key to quit the utility)
(Then restart the network service and check for the IP address by ***# service network restart*** *command)*
(If the change is not reflected with the above service, then restart the network manager by ***# service NetworkManager restart*** *command)*

```
# ifconfig                    (to see the IP address of the NIC card)
# ping  < IP address >        (to check whether the IP is pinging or not)
```

In RHEL - 7 **:**

```
# nmcli  connection  show                  (to see all the network connections)
# nmcli  device  show            (to see the network details if already configured manually or dynamically)
# nmcli connection add con-name  "System eth0"  ifname eth0 type ethernet   (to add the network connection)
# nmcli connection modify  "System eth0"  ipv4.addresses  '< IP address >/< netmask >< gateway >'
ipv4.dns < dns  server  IP address >  ipv4.dns-search   < domain name>  ipv4.method   <static  or  manually>
        (to assign IP address, gateway, dns, domain name and configure the network as static or manually)
# nmcli connection up  "System eth0"             (to up the connection)
# systemctl  restart  network                    (to restart the network service)
# systemctl  enable  network                     (to enable the network service)
# ifconfig                                       (to see the IP address of the NIC card)
# ping < IP address >                            (to check whether the IP is pinging or not)
```

20. ***What are the differences between RHEL - 6 and RHEL - 7 network configuration files?***

RHEL – 6	*RHEL - 7*
/etc/sysconfig/network-scripts *is the directory which contains the NIC configuration information.*	***/etc/sysconfig/network-scripts*** *is the directory which contains the NIC configuration information.*
/etc/sysconfig/network-scripts/ifcfg-<device name> *is the file which contains the NIC configuration details.*	***/etc/sysconfig/network-scripts/ifcfg-<device name>*** *is the file which contains the NIC configuration details.*
/etc/resolve.conf *is the file which contains DNS server IP and domain name location.*	***/etc/resolve.conf*** *is the file which contains DNS server IP and domain name location.*
/etc/sysconfig/network *is the hostname configuration file.*	***/etc/hostname*** *is the hostname configuration file.*
/etc/hosts *is the file which contains the local DNS server IP address.*	***/etc/hosts*** *is the file which contains the local DNS server IP address.*

21. ***What are the differences between Dynamic and Static configuration information?***

Dynamic configuration information	***Static configuration information***
Device *=<NIC device name>*	***Device*** *=<NIC device name>*
HWADDR*=02:8a:a6:30:45*	***HWADDR****=02:8a:a6:30:45*
Bootproto*=DHCP*	***Bootproto****=none (means static network)*
Onboot*=yes (****yes*** *means whenever we restart the system this connection will be activated and* ***no*** *means whenever we restart the system the connection will be deactivated)*	***Onboot****=yes*

Type=E*thernet*	***Type=E****thernet*
Userctl=*yes/no ----> If it is* ***yes*** *all normal users can disable the NIC card and If it is* ***no*** *except root user nobody can disable the NIC card.*	***Userctl=****yes/no ----> If it is* ***yes*** *all normal users can disable the NIC card and If it is* ***no*** *except root user nobody can disable the NIC card.*

22. ***How to set the hostname temporarily and permanently?***

<u>RHEL - 6</u> :

```
# hostname   <fully qualified domain name>(to set the hostname temporarily)
# vim  /etc/sysconfig/network                    (to set the hostname permanently)
  HOSTNAME=<fully qualified domain name>
  (save and exit this file)
# service network restart                        (to update the hostname in the network)
# chkconfig network on                           (to enable the connection at next reboot)
```

<u>RHEL - 7</u> :

```
# hostname   <fully qualified domain name>                (to set the hostname temporarily)
# hostnamectl  set-hostname    <fully qualified domain name>(to set the hostname permanently)
# systemctl  restart  network                                        (to update the hostname in the network)
# systemctl enable network                                           (to enable the connection at next reboot)
# service  NetworkManager  stop
# service  network  restart
# chkconfig  network  on
# service  NetworkManager  restart
```

25. ***What is the difference between TCP and UDP protocol?***

TCP is a connection oriented protocol and contain the information of sender as well as receiver.

<u>Example</u> : *HTTP, FTP, Telnet*
TCP is slower than UDP due to its error checking mechanism
UDP protocols are connection less packets have no information to where they are going. These type of ports are generally used for broadcasting.
<u>For example</u> : *DNS, DHCP,UDP are faster*

27. ***Mention all the network configuration files you would check to configure your ethernet card?***

*(i)/etc/sysconfig/network-scripts/ifcfg-eth**
(ii)/etc/sysconfig/network
(iii)/etc/resolve.conf
(iv)/etc/nsswitch.conf

28. ***What is the use of /etc/resolve.conf?***

It contains the details of nameserver, i.e., details of your DNS server which helps us connect to Internet.

29. ***What is the use of /etc/hosts file?***

To map any hostname to its relevant IP address.

32. ***What is the command to check all the listening ports and services of your machine?***

```
# netstat -ntulp
```

33. How can you make a service run automatically after boot?

```
# chkconfig <service  name>  on
```

34. What are the 6 run levels of linux? And how can you configure your script to run only when the system boots into GUI and not to any other runlevel?

0	*Power off*
1	*Single user*
2	*Multi user without network*
3	*Multiuser with network*
4	*Development purpose*
5	*GUI*
6	*Restart*

```
#chkconfig --level 5 service_name on
# chkconfig --level 1234 service_name  off
```

<u>Other useful commands</u> :

```
# ping <IP address  or  hostname>                (to check  the pinging)
# ifconfig                                       (to see or check all the NIC device names  and IP  addresses)
# hostname                                       (to see the hostname with fully qualified domain name)
# nslookup  <IP address>                         (to resolve  IP to name)

# ifconfig                                       (to check  the NIC card is enable or not)
# ifup  <NIC  device name>                       (to enable or up the NIC card)
#ifdown   <NIC  device name>                     (to disable or down the NIC card)
# route  -n                                      (to check  the gateway)
# cat  /etc/resolve.conf                         (to check  the dns server information)
# cat  /etc/sysconfig/network-scripts/ifcfg-<NIC  device name> (to  see the NIC device information)
# hostname   or  cat  /etc/sysconfig/network  (to check  the hostname  in RHEL  - 6)
# hostnamectl  status   or   cat  /etc/hostname          (to check  the hostname  in RHEL  - 7)
# ping  <IP address>                             (to check  the connection  communication)
# chkconfig   --list                      (to list all the services  which are running at boo time in RHEL  - 6 & 7)
# service  sshd status                    (to check  the sshd  status)
```

5. Managing SELinux

1. What is SELinux?

It is a one type of security that enhances the security that allows users and administrators more control over which users and applications can access which resources, such as files, Standard Linux access controls etc.,

It is mainly used to protect internal data (not from external data) from system services. In real time SELinux is disabled and instead of this IP tables are used. It protects all the services, files and directories by default if SELinux is enabled.

2. In how many ways we can implement the SELinux? Explain them.

We can implement the SELinux mainly in 2 modes.

(i) Enabled
(ii) Disabled (default mode)

Enabled **:**

Enabled means enabling the SELinux policy and this mode of SELinux is divided into two parts.

(a) Enforcing
(b) Permissive

Disabled **:**

Disabled means disabling the SELinux policy.

3. What is Enforcing mode in SELinux?

Enforcing means SELinux is on. It checks SELinux policy and stored a log. No can access the services by default but we can change the policy whenever we needed.

4. What is Permissive mode in SELinux?

SELinux is on and it don't check SELinux policy and stored the log. Everybody can access the services by default and we can also change the SELinux policy. It is also called as debugging mode or troubleshooting mode. In this mode SELinux policies and rules are applied to subjects and objects but actions are not affected.

8. What are the required files for SELinux?

```
#  vim  /etc/selinux/config            ----->   It is main file for SELinux.
# vim  /etc/sysconfig/selinux          ----->   It is a link file to the above file.
# vim  /var/log/audit/audit.log        ----->   SELinux log messages will be stored in this file.
```

9. what is the command to see the SELinux mode?

```
# getenforce                    (to check the SELinux mode)
```

10. What is command to set the SELinux mode temporarily?

```
# setenforce   0   or   1      (to set the SELinux mode. Where ' 0 ' -----> permissive and ' 1 ' -----> Enforcing)
```

Note **:** *(i) To change the SELinux mode from Permissive to Enforcing or Enforcing to Permissive modes the system restart is not required.*

(ii) To change Enforcing mode to Disabled mode or Disabled mode to Enforcing mode the system restart is required.

(iii) The above commands are changed the SELinux mode temporarily only. To make the selinux changes permanently then open ***/etc/selinux/config*** *and go to ,*

SELINUX=Enforcing or Permissive or Disabled *(save and exit this file)*

11. What is command to see the SELinux policy details?

```
# sestatus                              (to see the SELinux policy details)
```

6. Booting Procedure and Kernel parameters

1. Explain the booting procedure?

In Linux systems the booting is done in 6 stages.

BIOS
MBR
GRUB
Kernel
Init
Runlevel

BIOS :

BIOS stands for Basic Input and Output System. Whenever we power on the system , the system runs self diagnostic checks and detects all the connected input and out peripherals. This process is called POST (Power On Self Test). If any errors found it displays on the screen. Then BIOS locates the booting disk in the system and locates and loads the Primary boot loader nothing but MBR (Master Boot Record) into the memory. So, in simpleterms the BIOS loads the MBR into memory and executes the MBR.

MBR :

MBR stands for Master Boot Record. It is located in the 1st sector of the bootable disk (it may be /dev/hda or /dev/sda). The size of the MBR is 512 bytes and it contains three components.

(i) Primary boot loader information and its size is 446 bytes.
(ii) Partition table information and its size is 64 bytes.
(iii) MBR validation check and its size is 2 bytes. Its main purpose is whether the MBR is valid or not.

The primary boot loader contains the secondary boot loader nothing but GRUB or LILO (in old systems).

Then primary boot loader locates and loads the secondary boot loader into memory.

So, in simple terms the MBR loads and executes the GRUB boot loader.

GRUB or LILO :

GRUB stands for Grand Unified Boot loader. LILO stands for Linux Loader and is used in old Linux systems. If we have multiple kernel images installed in our system, we can choose which one to be executed. GRUB displays a splash screen, waits for few seconds. If we do not enter anything, it loads the default kernel image as specified in the grub configuration file. GRUB has the knowledge of the file system (the old LILO didn't understand the system). GRUB configuration file is ***/boot/grub/grub.conf*** *(****/etc/grub.conf*** *is a link to this). This file contains kernel and initrd images. So, in simple terms GRUB just loads and executes kernel and initrd images.*

Kernel :

Kernel initialises itself and loads the kernel modules and mounts the root file system as specified in the "root=" in grub.conf and then kernel executes the ***/sbin/init*** *program. Since init was the 1st program to be executed by Linux kernel, it has the process ID (PID) of 1. We can see this id by* ***# ps -ef | grep init*** *command. initrd stands for initial RAM Disk. initrd is used by kernel as temporary file system until kernel is booted and the real* *root the file system is mounted. It also contains necessary drivers compiled inside which helps it to access the hard drive partitions and other hardware.*

init level **:**

In this init program reads the ***/etc/inittab*** *file and put the system into specified run level. init identifies the default run level from* ***/etc/inittab*** *file and we can change the this default run level whenever we needed. We can find the default run level by* ***# grep "initdefault" /etc/inittab*** *command on our system. Normally the default run level in Linux is 3 in CLI (Command Line Interface) mode and 5 in GUI (Graphical User Interface) mode.*

Run Level Programs **:**

The following run levels are available in Linux systems.

0 -----> halt or shutdown the system
1 -----> Single user mode
2 -----> Multi user without NFS
3 -----> Full multi user mode but no GUI and only CLI mode
4 -----> Unused
5 -----> Full multi user mode with GUI (X11 system)
6 -----> reboot the system

Whenever we start the Linux system is booting we can see various services getting started. Those services are located in different run levels programs executed from the run level directory as defined by our default run level.

Depending on our default init level setting, the system will execute the programs from one of the following directories.

Run level 0 -----> /etc/rc.d/rc0.d
Run level 1 -----> /etc/rc.d/rc1.d
Run level 2 -----> /etc/rc.d/rc2.d
Run level 3 -----> /etc/rc.d/rc3.d
Run level 4 -----> /etc/rc.d/rc4.d
Run level 5 -----> /etc/rc.d/rc5.d
Run level 6 -----> /etc/rc.d/rc6.d

The above directories are also having symbolic links available for those directories under ***/etc/rc0.d, /etc/rc1.d,*** *....etc., So, the /etc/rc0.d is linked to /etc/rc.d/rc0.d*

Booting procedure in RHEL - 7:

Upto kernel the booting process is same as the above. ***/boot/grub2/grub.conf*** *is the GRUB configuration file in RHEL - 7.* ***systemd*** *is the initial process in RHEL - 7 and its process ID is* ***1****.*

linux16 *read the root (/) file system and then* ***initrd16****process will mount the root (/) file system in read & write mode and starts the* ***systemd****process. And the* ***systemd*** *process will read the* ***/etc/fstab*** *file and mount all the file systems. Then it reads the file* ***/etc/systemd/system/default.target*** *file and brings the system into the default run level according to the scripts the processes will start or stop.*

2. How to check the current run level of the system?

```
# who  -r                          (to see the present run level of the system)
```

14. How to check the present kernel version?

```
# uname  -r                        (it displays the present kernel version)
# uname  -a                        (it displays the present kernel version with other details)
# cat  /boot/grub/grub.conf        (in this file also we can find the kernel version)
```

16. How to check the version of the O/S ?

```
# cat  /etc/redhat-release         (gives the version of the O/S)
```

20. ***How to see the run level?***

# who -r *(to see the current run level)*

23. ***What is run level?***

(i) *Run level is nothing but to put the system in different levels to perform different maintenance modes.*

(ii) *There are 7 run levels. Those are 0, 1, 2, 3, 4, 5 and 6.*

(iii) *The above levels are used to put the system in different stages to avail different services.*

24. ***What is the default run level?***

(i) *When we boot the server the system automatically go to one particular run level. That run level is called the default run level.*

(ii) *In Linux the default run level is* ***5*** *in GUI and* ***3*** *in CLI.*

(iii) *We can modify the default run level by put an entry in* ***/etc/inittab*** *file.*

25. ***Which run level are you using?***

Run level ***3****.*

26. ***How to change the run level temporarily?***

# init <run level no.> *(to change the run level temporarily)*

Example : # init 0 or init 1 or init 2 or init 3 or init 4 or init 5 or init 6

7. *Job Automation*

3. *What are the differences between cron and at jobs?*

cron job :

(i) cron jobs are scheduling jobs automatically at a particular time, day of the week, week of the month and month of the year.

(ii) The job may be a file or file system.

(iii) We cannot get the information as a log file if the job was failed to execute ie., when it was failed and where is was failed and also cannot execute automatically the failed jobs.

at job :

(i) at jobs are executes only once.

(ii) Here also we cannot get the information if the job is failed and it is also do not execute the failed jobs automatically.

4. *What are the important files related to cron and at jobs?*

/etc/crontab *-----> is the file which stores all the scheduled jobs.*

/etc/cron.deny *-----> is the file used to restrict the users from using cron jobs.*

/etc/cron.allow ***----->*** *is used to allow only users whose names are mentioned in this file to use cron jobs and this file does not exist by default.*

/etc/at.deny *-----> same as cron.deny for restricting the users to use at jobs.*

/etc/at.allow *-----> same as cron.allow for allowing users to use at jobs.*

5. *What is the format of the cron job?*

```
# crontab   -e                              (to edit the cron job editor to create or remove the cron jobs)
<minutes><hours><day  of the month><month  of the year><day  of the week><job  or  script>
(0 - 59)      (0 - 23)          (1 - 31)   (1 - 12  or jan, feb, ...)        (0 - 6 or sun, mon, ...)
```

6. *How to check the assigned cron jobs of currently login user?*

```
# crontab   -l   -u   <user name>                (to check  the specified  user's  assigned cron jobs)
# crontab   -l   -u     raju                     (to check  the raju user's assigned cron jobs)
# crontab   -l                                   (to check  the root user's  assigned  cron  jobs)
```

7. *How to allow or deny cron jobs for a user?*

For allow	***For deny***
(i) Open ***/etc/cron.allow*** *file.*	*(i) Open* ***/etc/cron.deny*** *file.*
(ii) Put the entries of the user names whom do we want to allow the cron jobs.	*(ii) Put the entries of the user names whom do we want to deny the cron jobs.*

16. *Where is the location of the crontab and at jobs?*

/var/spool/cron -----> is the crontab file location.

/var/spool/at -----> is the at jobs file location.

```
# systemctl   restart  crond                 (to restart  the crond deamon  in RHEL - 7)
# systemctl   enable  crond                  (to enable the crond deamon at next boot in RHEL  - 7)
# systemctl   status  crond                  (to see the status of the crond deamon  in RHEL  - 7)
# systemctl   stop  crond                    (to stop  the  crond  deamon  in RHEL  - 7)
# systemctl   start  crond                   (to start  the cron  deamon  in RHEL  - 7)
# crontab     -lu     <user name>                              (to list all the cron jobs of the specified user)
```

8. Administrating Remote Systems (SSH)

2. What is SSH and explain it?

SSH is stands for Secure Shell. It was designed and created to provide the best security when accessing another computer remotely. Not only does it encrypt the session, it also provides better authentication facilities.

On windows systems install the putty software and through putty we can access the remote system by configuring ssh.

SSh is protocol which facilitates secured communication between two systems using Client-Server architecture and allows users to login to the server host systems remotely.

It is used to connect to remote system and perform administrative task or jobs. By default ssh takes password authentication mechanism and its port no. is 22. Through ssh the data will be transferred in encrypted format.

3. What is telnet?

Telnet is a mechanism to connect and to administrate the remote system from local system. This is the oldest program which is available on most network capable operating systems. Accessing a remote shell account through the telnet method is danger because in that everything that you send or receive over that telnet session is visible in plain text on your local network and the local network of the machine you are connecting to.

So, anyone can sniff the connection in-between can see our user name, password, email and other messages that we read and command that we run. For these reasons we need a more sophisticated program than telnet to connect to a remote host.

7. In how many ways we can connect the remote system?

(i)telnet	*(ii) ssh*
(iii) rlogin	*(iv) rcp*
(v)ftp	*(vi) scp*
(vii) sftp	*(viii) tftp*

8. What is the syntax for ssh?

```
# ssh   <IP address of the remote system>    -l    <user name>
# ssh   <user name>@<IP address of the remote system>
# ssh   <user name>@<remote hostname with fully qualified domain name>
```

** After executing any of the above commands, it may asks user name and password. Then type user name and passwords to connect the remote systems.*

10. How to prevent the remote login root user or how to configure the ssh to prevent the remote login for root?

(i) The location of the ssh configuration file is ***/etc/ssh/sshd_config***

(ii) Open the configuration file by ***# vim /etc/ssh/sshd_config***

-----> *go to line no.* ***42*** *(in RHEL - 6) or*

-----> go to line no. ***48*** *(in RHEL - 7)* ***PermitRootLogin yes***

*and uncomment that line and type as "**no**" in place of "**yes**" andsave and exit this file.*

(iii) Then restart the or reload the sshd deamon by

```
# service  sshd  restart                    (to restart the sshd deamon or service in RHEL - 6)
# systemctl restart sshd              (to restart the sshd deamon or service in RHEL - 7)
```

```
        # chkconfig  sshd  on                   (to enable the sshd deamon at next reboot in RHEL - 6)
        # systemctl  enable sshd                (to enable the sshd deamon at next reboot in RHEL - 7)
        # service  sshd  reload                 (to reload the sshd deamon in RHEL - 6)
        # systemctl  reload  sshd               (to reload the sshd deamon in RHEL - 7)
```

(iv) Then no root user cannot access our system remotely through ssh service.

Other useful commands **:**

```
# telnet   <IP address   or  hostname>         (to connect the specified remote system through telnet)
# ssh   <IP address   or   hostname>           (to connect the specified remote system through ssh)
        Username  : xxxxxx
        Password : xxxxxxx
# ssh   <IP address>    -l   <user name>       (to connect the remote system using user name)
        Password : xxxxxxx
# ssh   192.168.1.1  -l   root                 (to connect this remote system as root user)

# lastb                                        (to see the login failed tries)
```

9. Memory Management (Swap)

1. What is swap?

Swap space in Linux is used when the amount of the Physical memory (RAM) is full. If the system needs more memory resources and the RAM is full, inactive pages in the memory are moved from RAM to swap space. It helps the machines which are having small amount RAM and it should not be considered a replacement for more RAM. Swap is located on the hard disks which have slower access time than Physical memory.

2. What is the recommended swap space?

Generally the recommended swap space is double the RAM size, but the following table shows actual amount. Apart from the below recommendation a basic rule is applied to create the swap partition.

** If the RAM size is **less than or equal to 2 GB**, then the size of the **swap = 2 X RAM size**.*

** If the RAM size is **more than 2 GB**, then the size of the **swap = 2 GB + RAM size.***

Amount of RAM in the System	***Recommended Amount of Swap Space***
4 GB or less	*Min. 2 GB*
4 GB - 16 GB	*Min. 4 GB*
16 GB - 64 GB	*Min. 8 GB*
64 GB - 256 GB	*Min. 16 GB*
256 GB - 512 GB	*Min. 32 GB*

6. How to create the swap partition?

```
# fdisk   -l                                  (to see the available disks in the system)
# fdisk   /dev/sdb
          Command  (m   for   help) :  n              (to create  a new partition)
          First  cylinder :                           (press Enter  key)
          Last cylinder : +2048M
          Command   (m   or   help) :  t              (to change the hex code)
          Partition  no. (1-2) : 2                    (to change the partition  number hex code)
          Hex code :  82                              (82  is the hex code for Linux swap)
          Command   (m   for   help) :  w             (write  the changes to  the disk)
# partprobe    or  # partprobe   /dev/sdb      (to update the partition table information)
# mkswap             /dev/sdb2                (to convert  the raw disk to swap file system)
# swapon    /dev/sdb2                         (to turn on the swap partition)
# vim   /etc/fstab                            (to make the permanent  mount of swap partition)
  /dev/sdb2         swap      swap      defaults 0          0
  (save and exit this file)
# mount   -a                          (to mount all the partitions which are having entries  in  /etc/fstab  file)
# df   -hT                           (will not show the swap size)
# free    -m                         (to see  the total RAM  and  swap size)
```

7. How to remove the swap partition?

```
# swapon    -s                                (to see the swap partition  names  or  disks)
# swapoff    /dev/sdb2                        (to turn off the swap space)
# vim  /etc/fstab                             (open this file and remove the swap partition entry)
          (after removing  the swap partition save and exit this file)
# fdisk   /dev/sdb                            (to delete  the swap partition)
          Command   (m  for  help) : d                (d   for  to delete the partition)
          Partition no. (1-2) : 2
          Command   (m  for  help) : w                (to write  the changes into the disk)
# partprobe   or  # partprobe   /dev/sdb
# free    -m                                          (to see  the RAM as well  as swap sizes)
```

8. In how many ways can we create the swap spaces?

(i) By creating a new swap partition on the disk. (separate swap partition)

(ii) By creating swap file.

10. Software Management

1. What is software?

Software is a collection of programs to perform some tasks or manage systems, applications, databases ,...etc.,

2. What is package and package management?

Package is nothing but a software to perform some tasks. Software is the basic of any O/S allowing to install and use different utilities.

Package management means installing, updating, querying, repairing and removing packages. In Linux there are two tools to perform package management.

rpm -----> redhat package manager and yum -----> yellowdog updater modifier.

3. What is rpm?

*rpm is a package managing system (collection of tools to manage software packages). rpm is a powerful and most popular open source tool used for software management for installing, uninstalling (removing), verifying, querying and updating software packages. It is installed under **/var/lib/rpm** database directory. It deals with **.rpm** files, which contains the actual information about the packages. The rpm log messages will be stored in **/var/log/yum.log** file.*

4. What are the draw backs of rpm?

(i) rpm cannot resolve the dependency. It means, if we want to install any software, first the dependency packages should be installed.

(ii) There is no configuration file for rpm.

<u>Other useful rpm commands</u> :

```
# rpm  -ivh      <package name>              (to install the package)
# rpm  -qa                                   (to list all installed packages)
```

8. What is yum and explain the yum?

yum stands for yellow dog updater modified. yum is a package management application for computers running on Linux O/S.yum is a standard method of managing the installation and removal of software. It is from RHEL - 5 onwards. Packages are downloaded from collections called repositories, which may be online, on a network and or on installation media. yum is a front end tool for rpm. It is used to resolve the dependency which cannot be done by rpm. The yum command has access the repository where the packages are available and can install, update/upgrade, remove and query the packages automatically.

9. What are the important files that are related to yum?

***/etc/yum.conf** -----> is the yum configuration file.*
***/etc/yum.repos.d** -----> is the directory which contains the yum repository configuration file.*
***/etc/yum.repos.d/xxxxx.repo** ------> is the yum repository configuration file.*
***/var/lib/yum** -----> is the directory which contains the yum databases.*
***/var/log/yum.log** -----> is the file which stores the yum log messages.*

11. How to setup the yum client?

(i) Goto ***/etc/yum.repos.d*** *directory and create the repository file by* ***# vim linux.repo***

(ii) Type the entries as below,

```
[linux]                                   (Linux repo id)
name=yum repo client                      (yum repo client)
baseurl=ftp  or http://<IP  address of the server>/pub/rhel6
gpgcheck=0                       (0  means while installing it will not ask any signature keys of
              yum packages, If it is 1, then it will ask the signature keys while installing the packages)
enabled=1                    (if multiple repositories are there, then enable this only)
```

:wq (save and exit)

(iii) ***# yum clean all*** *(to clean the old one update the new repository)*

*(iv)****# yum repolist*** *(it displays no. of packages in that repository)*

<u>Other useful yum commands</u> :

```
# yum install  <package name>  -y            (to download and install the package and  y means yes)
# yum install  <package name>  -d            (to download the package)
# yum r  remove <package name>  -y     (to remove or uninstall the package and  y means yes)
# yum list installed                             (to display the list of all installed packages)
# yum list available                         (to list all the available packages to be installed)
# yum search <package name>                  (to search a particular package is available or not)
# yum info  <package name>                   (to display the information on that package)
# yum update  <package name>        (if the update version of the specified package is available, then update
that package)
# yum update all                 (to update all the packages nothing but whole system will be updated)

# yum clean all  (to clear the history, if we disable the repository id, then we have to clean the history, then
                                                                  only it will disable the repository)
```

11. Backup and Restore

4. What is tar and Explain it or how to take a backup using tar?

Archiving means collection of files and directories and to make a single file nothing but compression. tar means tape archiving. It is an archive file. By using tar command we can take a backup of files and directories. It cannot support file systems backup and also not support for large files more than 80GB. tar will not skip any single file including bad blocks also.

<u>Full syntax of tar</u> :

```
# tar   <options><destination  file name with path><source  file or directory  with path>
The options are,  -c  ----->  create
                              -v  ----->  verbose
                              -f  ----->  file name
                              -t  ----->  listing
                              -tv  ---->  long listing
                              -x  ----->  extract
                              -w  ---->interactive
                              -C  ----->  specific location  (Capital  C)
```

```
             -u  ----->  update means adding new contents  to the existing tar file
      --update   ----->             "                              "
      --delete       ----->  deletes  the contents from the tar file
             -p  ---->  preserve  the old permissions  of the files/directories  when extracting  the tar file
             -z  ---->  gzip (gun zip)  compression
             -j  ---->  bzip2 (bun zip) compression
             -J  ---->  xz  compression    (from  RHEL - 7)
```

Examples:

```
# tar  -cvf   /root/etc.tar     /etc/*           (to copy all the files and directories  from /etc  and make a single file
                                                         and place in the /root/etc.tar  file)
# tar  -tvf          /root/etc/tar                      (to long listing the contents of the  /root/etc.tar   file)
# tar  -xvf          /root/etc.tar      -C    /root1/     (to extract and copy the files in /root1/   location)
# tar  -xf           /root/etc.tar                      (to list the contents of the tar file)
# tar  -f  /root/etc.tar    --update   or   -u   <file name  or  directory>  (to add the new contents to the existing
                                                                                  tar file)
# tar  -f /root/etc.tar        --delete  <file name   or  directory>  (to delete the file from the tar)
# tar  -u   /root/etc.tar     /var               (to add the  /var  contents  into the   /root/etc.tar    file)
# tar  -cvf    mytar.tar    /  --xattrs          (to archive  the  contents  along with  SELinux  and  ACL  permissions)
# du   -h    /root/etc.tar                       (to see the size of the tar compressed  file)
```

5.   ***What are the compressing & uncompressing tools available for tar and explain them?***

Compressing Tools ***Uncompressing Tools***

```
# gzip   (.gz)                                              # gunzip
# gzip    <tar file name>                    (to compress  the size of the tar file and the output file is  .tar.gz)
# gunzip    < .gz  compressed  file name>    (to uncompress  the compressed  tar file and the output is  .tar only)
# bzip2     <tar  file  name>                (to compress  the size of the tar file and the output is  .tar.bz2)
# bunzip2   < .bz2  compressed  file name>   (to uncompress  the compressed  file and the output is   .tar only)
```

6.   ***What is scp, rsyncand how to use it?***

scp means   secure copy. ie.,   ssh + cp = scp which is used to copy the files/directories   into remote system.
scp will copy files/directories   into remote system blindly ie., if the file already exits, it will over write that file.
So, scp will take more time to copy when compared to # rsync tool.

```
# scp     <file  name><user  name>@  <IP address of the remote  system>:<location   to be copied>
# scp   anaconda*    root@192.168.1.1:/root (to copy anaconda file into /root of the remote  system)
# scp   -r    /etc/    root@192.168.1.1:/raju              (to copy   /etc/   directory  into   /raju of remote  system)
#scp    -av    /raju  root@192.168.1.1:/root    (to copy    /raju   into   /root  of the remote  system)
# scp   -r    root@192.168.1.1 :/etc      /home (to copy   /etc   of the remote  system  into   /home of thelocal system)
```

rsync *is also used to copy files/directories into remote systems. rsync tool will compare the new files or directories and copy only the changed or modified contents of the files into remote system. So, it takes less time to copy when compared to # scp tool.*

```
# rsync    -av    root@192.168.1.1:/etc    /home         (to copy   /etc   directory   changed contents into   /home)
rsync  options are,   -a     ----->    all  (copy the file with all permissions  except  SELinux  and  ACL  permissions)
                      -aA  -----> synchronize   ACL  permissions
                      -aAx ---->    synchronize   ACL  and  SELinux  permissions  also.
```

12. Managing Installed Services

1. ***What is service or deamon?***

*Service or deamon is program that stats at background and continuously run in the background. The service or deamon is ready for input or monitor the changes in our computer and respond to them. For example the Apache server has a deamon called **httpd** that listens on port no. 80 on our computer and when it receives a request for a page it sends the appropriate data back to the client machine.*

Example : apache, samba, NFS, FTP, ...etc.,

3. ***What are the differences between RHEL -6 and RHEL-7 services?***

RHEL -6	***RHEL -7***
*(a) The parent process ie., the starting process is **initd** and it's process id (pid) is 1.*	*(a) The parent process ie., the starting process is **systemd** and it's process id (pid) is 1.*
*(b) There two commands for starting the services . They are called **# service** and **# chkconfig***	*(b) Here only one command is used to start the service. That is **# systemctl***
*(c) **# service** command is used to start or stop the services temporarily and **# chkconfig** is used to start or stop the services at next booting time.*	*(c) **# systemctl** is the command to start or stop the services temporarily or next booting time.*
*(d) **/etc/init.d** is the location for all the services.*	*(d) **/usr/lib/systemd/system** is the location for all the services.*
(e) # service <service name> <start/stop/restart/reload/status >	*(e) # systemctl <start/stop/restart/reload/ status ><service name>*

4. ***What are the differences between initd and systemd deamons?***

Initd	***systemd***
(a) It is the starting process in RHEL - 4, 5 and 6.	*(a) It is starting process in RHEL - 7.*
(b) It's process id (pid) is 1.	*(b) It's process id (pid) is 1.*
(c) It will take more time to the system and services.	*(c) It will take less time to start the system and services when compared to RHEL - 6.*
(d) It will start the services one by one.	*(d) It will start the services parallel not one by one.*
(e) All the linux services are ends with letter d. Example : sshd, httpd, crond, ...etc.,	*(e) All the linux services are ends with letter d.service Example : sshd.service, httpd.service, ...etc.,*

# ps (to see the active process in the system)

# top (It will show a dynamic real-time view of a running system. ie., a summary of processes or threads currently managed by the Linux kernel)

# kill (It sends the specified signal to the specified process or process group)

# pgrep (to list the process id's which matches with the pgrep argument)

***RHEL - 6 commands* :**

# service <service name> status (to check the status of the service)

# service <service name> start (to start the service)

# service <service name> stop (to stop the service)

# service <service name> reload (to reload the service)

# service <service name> restart (to restart the service)

** These above commands will change the service statuses temporarily. So if we want to change statuses of the*

***RHEL - 7 commands* :**

# systemctl status <service name> (to check the status of the service)

# systemctl start <service name> (to start the service)

# systemctl stop <service name>(to stop the service)

# systemctl reload <service name> (to reload the service)

# systemctl restart <service name> (to restart the service)

13. Managing Process

1. ***What is process and explain it?***

2. ***How many process are run generally on Linux and explain them?***

There are generally three types of processes that run on Linux. They are,

(i) Interactive Processes
(ii) System Process or deamon
(iii) Automatic or batch.

Interactive Processes :

Interactive processes are those processes that are invoked by a user and can interact with the user. For example # vi or # vim are the interactive processes. Interactive processes may be run in foreground or background. The foreground process is the process that we are currently interacting with and is using the terminal as its stdin (standard input) and stdout (standard output). The background process is not interacting with the user and can be in one of two states, ie., paused or running.

System Processes or deamons :

Deamon is refer to processes that are running on the computer and provides services but do not interact with the console. Most server software is implemented as a deamon. For example Apache, samba, sshd are the deamons. Any process can become a deamon as long as it is run in the background and does not interact with the user.

3. ***What is parent process?***

The process which starts or creates another process is called the parent process. Every process will be having a parent process except initd process. The initd process is the parent process to all the remaining processes in Linux system because it is the first process which gets started by the kernel at the time of booting and it's PID is 1. Only after initd process gets started, the remaining processes are called by it, and hence it is responsible for all the remaining processes in the system. The parent process is identified by PPID (parent process ID).

4. ***What is child process?***

A process which started or created by the parent process is called child process and it is identified by PID.

Useful # ps commands :

# ps -aux (it displays all the terminals processes information including background processes with user names)
**** ?** (question mark) if it is appeared at tty column, it indicates that is a background process.*
# ps -ef (it displays the total processes information with parent process ID (PPID))

# ps -aux |grep firefox (to check whether the firefox is running or not)
** To communicate with the processes **# kill** and **# pkill** commands are used.*
# kill -----> It will kill the processes using PID's.
# pkill -----> It will kill the processes using process names.
** We can also give some signals while using the above commands and we get the signals information by **# kill -l** command. This command will list all the signals with no's and there are 64 signals to pass.*

7. ***How many process states are there?***

There are six process states and they are,
*(i)Running process (the process which is in running state and is indicated by "**r**").*
*(ii)Sleeping process (the process which is in sleeping state and is indicated by "**s**")*
*(iii) Waiting process (the process which is in waiting state and is indicated by "**w** ").*
*(iv) Stopping process (the process which is in stopping state and is indicated by "**T** ").*

(v) Orphan process (the process which is running without parent process and is indicated by "**o** ").
(vi) Zombie process (the process which is running without child process and is indicated by "**Z**").

8. What is Orphan process?

The processes which are running without parent processes are called Orphan processes. Sometimes parent process closed without knowing the child processes. But the child processes are running at that time. These child processes are called Orphan processes.

9. What is Zombie process?

When we start parent process, it will start some child processes. After some time the child process es will died because of not knowing the parent processes. These parent processes (which are running without child processes) are called Zambie processes. These are also called as defaunct processes.

11. What is top command and what it shows?

top is a command to see the processes states and statuses information continuously until we quit by pressing "**q**". By default top command will refresh the data for every **3** seconds.

When we need to see the running processes on our Linux in real time, the top command will be very useful. Besides the running processes the top command also displays other information like free memory both physical and swap.

The first line shows the current time, "**up 1 day**" shows how long the system has been up for, "**3 user**" how many users login, "**load average : 0.01, 0.00, 0.23**" the load average of the system **1, 5 and 15 minutes**.

The second line shows the no of processes and their current states.
The third line shows CPU utilization details like % of the users processes, % of the system processes, % of available CPU and % of CPU waiting time for I/O (input and output).

The fourth and fifth lines shows the total physical memory in the system, used physical memory, free physical memory, buffered physical memory, the total swap memory in the system, used swap memory, free swap memory and cached swap memory, ... etc.,

From sixth line onwards the fields are as follows.

PID Process ID
USER Owner of the process ie., which user executed that process
PR Dynamic Priority
NI Nice value, also known as base value
VIRT Virtual size of the task includes the size of processes executable binary
RES The size of RAM currently consumed by the task and not included the swap portion
SHR Shared memory area by two or more tasks
S Task Status
% CPU The % of CPU time dedicated to run the task and it is dynamically changed
% MEM The % of memory currently consumed by the task
TIME+ The total CPU time the task has been used since it started. + sign means it is displayed with hundredth of a second granularity. By default, TIME/TIME+ does not account the CPU time used by the task's dead children

17. What is the command to see the I/O statistics?

# iostat (to see the Input and Output statistics in the Linux system)

* This command is used to monitoring the system input and output statistics and processes transfer rate.
* It is also used to monitor how many kilo bytes read per second and how many kilo bytes read and write, shows CPU load average statistics since the last reboot in first line and most current data is shown in the second line.

18. How many CPUs are there in the system?

__# cat /proc/cpuinfo__ command will show no. of CPUs, no. of cores, no. of threads, no. of sockets and the CPU architecture, …etc., information.

__# nproc__ command will give the no. of CPUs present in the system.

__# lscpu__ command will give the information the architecture of the CPU (x86_64 or x86_32), no. of cores, no. of threads, no. of sockets, cache memory sizes (L 1, L 2, L 3, …etc) , CPU speed and the vendor of the CPU.

24. What is command to check the load average?

__# uptime__ is the command to check the system load, present time, from how many hours the system is running and load average.

** The load average shows three fields. The 1st field shows the load average from 1 minute, 2nd field shows the load average from 5 minutes and 3rd field shows the load average from 15 minutes.*

25. How to assign or shift the process to the particular CPU?

(i)First install __util-linux__ package by __# yum install util-linux -y__ command.

(ii)Check the specified process is assigned to which processor ie., which CPU by __# taskset -p <pid>__command.

(iii) Then shift the process to another available CPU by __# taskset -cp <cpu -list><pid>__ command.

Examples:

```
# taskset  -p  2125            (to check which processor is assigned to that process ID)
# taskset  -cp  0, 4  2125     (to shift the process to the CPUs 0 and 4)
# taskset  0  firefox          (to assign the firefox process to the CPU 0)
```

28. What is SAR utility and how to use it?

SAR stands for System Activity Report. Using SAR we can check the information of CPU usage, memory, swap, I/O, disk I/O, networking and paging. We can get the information of the present status and post status (history using the data) upto last 7 days because __HISTORY=7__ is there in the configuration file. The log messages are stored in __/var/log/sa/sa1, /var/log/sa/sa2, /var/log/sa/sa3__, ….etc., (where 1, 2, 3, ….etc., are dates). The SAR configuration is stored in __/etc/sysconfig/sysstat__ file. In this file the __HISTORY=7__ default option will be there. So, we can change the default 7 days to our required value.

Before using the SAR utility first we should install the SAR utility package by __# yum install sysstat -y__ command.*

<u>Examples</u> :

```
# sar  2  10                       (It will give the system report for every 2 seconds upto 10 times)
# sar  -p  2  10                   (to see the CPU utilization for every 2 seconds upto 10 times)
# sar  -p  ALL  -f  /var/log/sa/sa25       (to check the CPU utilization on 25th day of the current month)
# sar  -p  ALL  -f  /var/log/sa/sa10  -s  07:00:00  -e  15:00:00    (to check the CPU utilization on 10th day of
                   the current month from 7:00 to 15:00 hrs. where -s means start time -e end time)
# sar  -r  2  10                   (to see the memory utilization for every 2 seconds upto 10 times)
# sar  -r  -f  /var/log/sa/sa14    (to check the memory utilization on 14th day of the current month)
# sar  -r  -f  /var/log/sa/sa10  -s  07:00:00  -e  15:00:00    (to check the memory utilization on 10th day of
              the current month from 7:00 to 15:00 hrs. where -s means start time -e end time)

# sar  -m  2  10                   (to see the power management for every 2 seconds upto 10 times)
# sar  -b  2  10          (to see the disk input and output statistics for every 2 seconds upto 10 times)
```

29. What are the port no. for different services?

The Port no. list :

FTP (For data transfer)	*20*	*HTTP*	*80*
FTP (For connection)	*21*	*POP3*	*110*
SSH	*22*		
Telnet	*23*	*LDAP*	*389*
Send Mail or Postfix	*25*		
DNS	*53*	*HTTPS*	*443*
NFS	*2049*		
MySQL	*3306*		

** Ping is not used any port number. It is used ICMP (Internet Control Message Protocol) only.*

Other useful commands :

```
# uptime          (to see from how long the system is running and also gives the load average report)
*  The load average is having 3 fields. 1 - present status,  2 - 5 minutes back and 3 - 15 minutes back.
# iostat  5  2              (to monitor the input and output statistics for every 5 seconds upto 10 times)
# nproc                                  (to check how many processors (CPUs) are there in the system)
# top  1                                    (to see the no. processors (CPUs) are there in the system)
# lscpu                                          (to see the no. of CPUs present in the system)
# lsblk                                     (to see all the partitions or block devices information)
# cat /etc/redhat-release                        (to see the RHEL version of system)
# stat   <file name  or  directory name>         (to see the statistics of the file or directory)
```

14. FTP (File Transfer Protocol) Server

1 ***What is FTP?***

FTP stands for File Transfer Protocol used to transfer files from one host to another host over a TCP-based network.

2. ***How ftp works?***

FTP is built on client-server architecture and utilizes separate control and data connection between the client and server. FTP users may authenticate themselves using a clear-text sign-in protocol but can connect anonymously if the server is configured to allow it.

Usually, the FTP server, which stores files to be transferred, uses two ports for the transferring purpose. One port for commands and another port for sending and receiving data. Requesting from client computers are received at the port 21 of server. ie., it is exclusively reserved for sending commands, therefore it is called the Command Port.

Once an incoming request is received, the data requested or uploaded by the client computer is transferred through a separate port 22 and referred as Data Port. At this point, depending on the Active or Passive mode of the FTP connection, the port number used for the Data Transfer Varies.

3. ***What is Active FTP?***

In Active FTP connection, the connection is initiated by the Client, and the data connection is initiated by the Server. And as the server actively establishes the data connection with the client, hence it is called the Active FTP. Here the client opens up a port higher than 1024 and it connects to the server through port 21. Then the server opens its port 20 to establish a data connection.

4. ***What is Passive FTP?***

In Passive FTP connection, both command and data connections are established by the client. In this the server acts as entirely passive, that's why it is called the Passive FTP. Here the server listens for incoming requested connections from client through port 21 and the client also initiates the data connection at port 20.

5. ***What is the main difference between the Active FTP and Passive FTP?***

The main difference between the Active FTP and the Passive FTP is based on who initiates the data connection between the server and the client. If the data connection is initiated by the server, that is called Active FTP and if the data connection is initiated by the client, that is called Passive FTP.

6. ***What is the profile for FTP server?***

(i) *It is used for uploading and downloading the files and directories cannot be downloaded.*

(ii) *The FTP server packageis* ***vsftpd****.*

(iii) *The FTP client packages are* ***ftp*** *and* ***lftp****.*

(iv) *The FTP server deamon is* ***vsftpd*** *(Very Secure FTP deamon)*

(v) *The FTP scripting file is* ***/etc/initd/vsftpd***

(vi) *Port numbers 20 for data connection and 21 for FTP command connection.*

(vii) *The document root for FTP is* ***/var/ftp***

(viii) *The FTP home directory is* ***/var/ftp***

(ix) *The FTP configuration files are,*

(a) /etc/vsftpd/vsftpd.conf

(b) /etc/vsftpd/user_list

(c) /etc/vsftpd/ftpuser

(d) /etc/pam.d/vsftpd

7. How to configure the FTP server?

*(i) Install the FTP package by **# yum install vsftpd* -y** command.*

*(ii)Goto FTP document root directory and create some files by **# cd /var/ftp/pub***

# touch f(1..10}

*(iii) Restart the FTP service or deamon by **# service vsftpd restart** command in RHEL - 6.*

***# systemctl restart vsftpd** command in RHEL - 7.*

(iv) Make the FTP service or deamon enable even after reboot the server by

***# chkconfig vsftpd on** command in RHEL - 6 and **# systemctl enable vsftpd** command in RHEL - 7.*

(v) Add the FTP service to the IP tables (RHEL - 6) and Firewalld (RHEL - 7).

8. How to configure the FTP client and how to connect the ftp server?

*(i)Go to the client machine and install the **FTP** and **Lftp** packages.*

# yum install ftp* lftp* -y

9. How to configure the Secure FTP server?

*(i) Open the FTP configuration file by **# vim /etc/vsftpd/vsftpd.conf** command.*

*(ii) Go to line no : 12 and type as, **ananymous_enable=no** (save and exit the file)*

** ananymous_enable=yes (by default)*

It means anybody can login to the FTP server without any username and password.

If ananymous_enable=no, then we must provide the username and passwords when it prompts.

*(iii) Restart the ftp deamon by **# service vsftpd restart** command in RHEL - 6 or*

***# systemctl restart vsftpd** command in RHEL - 7.*

*(iv) Assign the FTP user password by **# passwd ftp** (type and retype the ftp user password)*

*(v) Go to client side and connect the FTP server by **# ftp 172.25.9.11** command.*

16. What are the difference between FTP and LFTP servers?

(i)The user name and password are required to access the FTP server but LFTP does not requires passwords.

*(ii) In ftp>prompt the **" Tab "** key will not work but in lftp> prompt the **" Tab "** key will work as usual.*

<u>Other useful FTP Commands</u> :

# ftp 172.25.9.11	*(to access the FTP server provide FTP user name and password)*
ftp > ls	*(to see all the files and directories in FTP root directory)*
ftp > bye or quit	*(to quit or exit from the FTP server)*
# lftp 172.25.9.11	*(to access the LFTP server without asking any passwords)*

15. <u>NFS (Network File System) Server and Autofs</u>

1. What is NFS? Explain it.

NFS stands for Network File system and it is way to share the local hard drive files between machines which are NFS compatible. That means we share the files between Linux and Unix machines but not between Linux and windows systems. NFS is used upd protocol.

Normally the NFS server exports one or more directories to the client system and the client system mount one or more of the shared directories called mount points. After the NFS is mounted, all I/O operations are written back to the server, and all the clients notice the change. A manual refresh is not needed because the client access the remote file systems same as local file system because access does not requires the IP address, user name and password. However we can provide the security using the kerberos security.

# service nfs restart	*(to restart the NFS service in RHEL - 6)*
# chkconfig nfs on	*(to on the nfs service in RHEL - 6)*
# systemctl enable nfs-server	*(to enable the nfs-server in RHEL - 7)*

24. What is LDAP server?

LDAP (Lightweight Directory Access Protocol) is a software protocol for enabling anyone to locate organizations, individuals, and other resources such as files and devices in a network, whether on the public Internet or on a corporate intranet.LDAP is lighter because in its initial version it did not include security features.

16. Samba Server

1. What is Samba and explain it?

(i)An open source implementation of the SMB file sharing protocolthat provides file and print services to SMB/CIFS clients. Samba allows a non-Windows server to communicate with the same networking protocol as the Windows products.

(ii)Samba allows Linux computers to share files and printers across a network connection by using SMB protocol. Samba will support DFS, NFS, ufs file systems to share files and directories. That's why Samba is used to share files and directories between different platforms.

(iii) Samba will support printer sharing and it requires authentication.

2. What are the differences between Samba and NFS?

(i) Samba supports to all O/S platforms, whereas NFS will support the same platforms only.

(ii) There is a security in Samba because Samba requires authentication, whereas in NFS there is no security if there is no kerberos because NFS does not requires authentication.

(iii) Samba will support printer sharing, whereas NFS will not support printer sharing.

3. What are the different file systems for sharing different O/S?

(i) Windows --- Windows -----> Distributed File system (DFS)
(ii) Linux --- Linux -----> Network File system (NFS)
(iii) Unix --- Unix -----> Network File system (NFS)
(iv) Apple MAC --- Apple MACs -----> Apple File sharing Protocol (AFP)
(v) Windows --- Linux -----> Common Internet File system (CIFS)

17. NTP (Network Time Protocol) or Chrony

1. What is NTP and Chrony?

NTP stands for Network Time Protocol in RHEL - 6 and Chrony is also a Network Time Protocol in RHEL - 7.These are used to synchronize the time on your Linux system with a centralized NTP or Chrony server.A local NTP or Chrony server on the network can be synchronized with an external timing source to keep all the servers in your organization in-sync with an accurate time.

2. What are the differences between NTP and Chrony?

NTP	***Chrony***
This is used in RHEL - 6.	*This is used in RHEL - 7.*
*Package is **ntp or system-config-date**.*	*Package is **chrony**.*
*It's deamon is **ntpd** and Port number is **123**.*	*It's deamon is **chronyd** and Port number is **123**.*
We have to install the package manually.	*By default this package is installed.*
***# ntpq -p** (to check ntp is configured or not).*	***# chronyc sources -v** (to check chrony is configured or not).*
*Configuration file is **/etc/ntp.conf***	*Configuration file is **/etc/chrony.conf***
*Log file is **/var/log/ntpstat***	*Log file is **/var/log/chrony***

3. ***How to configure the NTP and Chrony client?***

NTP :

(i)Install the ntp package by ***# yum install ntp* -y*** *or* ***# yum install system-config-date* -y*** *command.*

(ii)open the configuration file by ***# system-config-date or # vim /etc/ntp.conf*** *command.*

(# system-config-date command is used to configure the NTP in graphical mode)

** Make a comment on line numbers 21, 22 and 23. Then go to line number 24 and type as below.*

server <ntp server host name> *(save and exit this file)*

Example : *server classroom.example.com*

(iii)Restart the ntpd service by ***# service ntpd restart*** *command.*

(iv) Enable the ntp service at next boot by ***# chkconfig ntpd on*** *command.*

(v) Check whether the NTP is configured or not by ***# ntpq -p*** *command.*

Chrony :

(i)Chrony package is not installed because by default it is installed. If it not installed then install the package by ***# yum install chrony* -y*** *command.*

(ii)Open the chrony configuration file by ***# vim /etc/chrony.conf*** *command.*

** Make a comment on line numbers 3, 4 and 5. Then go to line number 6 and type as below.*

server <ntp server host name> iburst *(save and exit this file)*

Example : *server classroom.example.com iburst*

(iii)Restart the chrony service by ***# systemctl restart chronyd*** *command.*

(iv) Enable the chrony service at next boot by ***# systemctl enable chronyd*** *command.*

(v)Check whether the Chrony is configured or not by ***# chronyc sources -v*** *command.*

# timedatectl *(to check whether the client's time is synchronized to the server's time)*

# timedatectl list-timezones *(to list the different time zones)*

# timedatectl set-time <hh : mm : ss> *(to set the time)*

# timedatectl set-timezone Asia/Kolkata *(to set the time zone in RHEL - 7)*

# tzselect Asia/Kolkata *(to set the time zone in RHEL - 6)*

18. DNS (Domain Naming System)

1. ***What is DNS?***

DNS stands for Domain Naming System. The ***DNS****translates Internet domain and host names to IP addresses. DNS automatically converts the names we type in our Web browser address bar to the IP addresses of Web servers hosting those sites.*

DNS implements a distributed database to store this name and address information for all public hosts on the Internet. DNS assumes IP addresses do not change (are statically assigned rather than dynamically assigned).

2. ***What is DNS server and how it works?***

In any network, the hosts primarily communicate between each other through IP addresses. For example, if my computer is doing a google search, my computer is actually communicating with the IP address of one of the web servers of google.com. However, even if the computer is efficient with numbers, humans on the other hand work better with names. For this reason, the TCP/IP protocol includes the ***Domain Name System (DNS)*** *to link between IPs and computer names i.e. hostnames. The DNS is a distributed database of computers that is responsible for resolving hostnames against IP addresses and vice-versa.*

Any DNS query involves two parts.

(i) ***The Resolver:*** *The resolver forms up or initiates the query. The resolver itself does not run as a program.* ***/etc/resolve.conf*** *is an example of a resolver.*

(ii) ***Name Server:*** *The Name Server is the service running in the server that responds to the DNS query generated by the resolver i.e. answers to the question of the resolver.*

The working DNS :

(i)The client initiates a query to find a domain example.com. The client sends the query to the DNS server of the ISP. (The DNS Server IP in the client computer is set as the IP address of the DNS Server of the ISP)

*(ii)The DNS Server of the ISP first checks it's own cache to check whether it already knows the answer. But as the answer is not present, it generates another query. As the **Top Level Domain** of example.com is **.com**, so the DNS server queries the **Internet Registration Authority** to find who is responsible for example.com.*

(iii)The Internet Registration Authority responds to the ISP by answering the query.

*(iv) Once the ISP DNS Server knows the authoritative name servers, it contacts the authoritative name servers to find out the IP address for www.example.com i.e. the IP address of host **www** in the **domain** example.com.*

*(v)**example.com** responds to the ISP DNS Server by answering the query and providing the IP address of the web server i.e. www*

*(vi) The ISP DNS Server stores the answer in it's cache for future use and answers to the client by sending the IP address of the **www** server.*

*(vii) The client may store the answer to the DNS query in it's own cache for future use. Then the client communicates directly with the **www** server of domain **example.com** using the IP address.*

*(viii) The **www** server responds by sending the index.html page.*

3. What is the format of the domain name?

*Like a physical address, internet domain names are hierarchical way. If the Fully Qualified Domain Name is **www.google.co.in** , the **www** is the Hostname, **google** is the Domain, **co** is the Second Level Domain and **in** is the Top Level Domain.*

4. What are the files we have to edit to configure the DNS?

*There are four files to edit to configure the DNS. They are **/etc/named.conf, /etc/named.rfc1912.zones, Forward Lookup Zone**and **Reverse Lookup Zone**. DNS provides a centralised database for resolution. Zone is storage databasewhich contains all the records.*

*__**Forward Lookup Zone**__ is used to resolve**Hostnames** to **IP addresses**.*

*__**Reverse Lookup Zone**__ is used to resolve **IP addresses** to **Hostnames.***

5. What are the DNS record and explain them?

*(i) __**SOA Record**__ **:** (Start of Authority)*

SOA contains the general administration and control information about the domain.

*(ii) __**Host A Record**__ **:***

*(a) It is nothing but a**Forward Lookup Zone**.*

*(b) It maps **Hostname** to **IP address.***

*(iii) __**PTR**__ **:** (Pointer Record)*

*(a) It is nothing but a **Reverse Lookup Zone**.*

*(b) It maps **IP address** to **Hostname.***

*(iv) __**NS Record**__ **:** (Name Server Record)*

It stores the DNS server IP addresses.

*(v) __**MX Record**__ **:** (Mail Exchange Record)*

*It stores the records of the **Mail Server IP address**.*

*(vi) __**CNME Record**__ **:***

It is nothing but Host's Canonical name allows additional names or aliases to be used locate a system.

Client's configuration file :		***/etc/resolve.conf***
Port number	*:*	***53***

7. **How to configure the DNS server?**

```
# hostname   <fully qualified  domain  name>          (to change the hostname in  RHEL - 6)
# hostname   server9.example.com               (example for setting  hostname temporarily in  RHEL - 6)
# hostnamectl  set   <fully qualified domain  name>             (to change the hostname in  RHEL - 7)
# hostnamectl   set  server9.example.com         (example for setting  hostname temporarily in  RHEL - 7)
# vim   /etc/hosts            (open this file and go to last line  and  type as below  in  RHEL - 6  only)
<IP address>        <fully qualified domain name>       <hostname>
172.25.9.11         server9.example.com          server9                          (for example of the above syntax)
# vim   /etc/sysconfig/network   (open this file and go to last line  and  type as below  in  RHEL - 6  only)
HOSTNAME=<fully  qualified domain name>
 HOSTNAME=server9.example.com                                    (for example of the above syntax)
```

(xiv) Check the resolution with nslookup command.

```
  # nslookup    <hostname>                          (to check  the resolution with hostname)
  # nslookup    <IP address>                      (to check  the resolution with IP address)
```

Example **:**

```
  # nslookup        server9.example.com
  # nslookup        172.25.9.11
```

19. DHCP (Dynamic Host Configuration Protocol)

1. **What is DHCP and explain it?**

DHCP stands for Dynamic Host Configuration Protocol. DHCP is a network protocol that enables the server to assign an IP addresses to the clients in the network automatically from a defined range of IP addresses ie., scope configured for a given network.

DHCP allows a computer to join in an IP-based network without having a pre-configured IP address. DHCP is a protocol that assign unique IP addresses to devices, then releases and renews those addresses as devices leave and rejoin in the network.

Internet Service Providers (ISPs) usually use DHCP to help customers join their networks with minimum setup effort required. Likewise, home network equipment like broadband routers offers DHCP support to joining home computers to Local Area Networks (LANs).

In simple terms DHCP is used to assign the IP addresses to the remote hosts automatically. First client requests to the DHCP server, then DHCP server accepts the client's request and assign the next available IP address to the requested DHCP client.

2. **How the DHCP works?**

The process of requesting the IP address from the DHCP clients and assign the IP address by the DHCP server is called ***"D O R A"****.*

(i)When we switch on the system with DHCP client, the client system sends the ***broadcast request*** *looking for a DHCP server to answer. This process is called* ***DISCOVER*** *or* ***DHCP DISCOVER****.*

(ii)The router directs the ***DISCOVER*** *packet to the correct DHCP server.*

(iii) The server receives the ***DISCOVER*** *packet. Based on availability and usage policies set on the server, the server determines an appropriate address (if any) to give to the client. The server then temporarily reserves that address for the client and sends back to the client an* ***OFFER*** *or****DHCP OFFER*** *packet with that address information. The server also configures the client's DNS servers, WINS servers, NTP serves and sometimes other services also.*

(iv)Then the Client sends a **REQUEST** or **DHCP REQUEST** packet, letting the server know that it intends to use the address.

(v)Then the server sends an **ACK** or **DHCP ACK** packet, conforming that the client has been given a lease on the address for a server specified period of time.

3. What is the disadvantage to assign the Static IP address?

When a system uses a static IP address, It means that the system is manually configured to use a specific IP address. One problem with static assignment, which can result from user error or inattention to detail, occurs when two systems are configured with the same IP address. This creates a conflict that results in loss of service. Using DHCP to dynamically assign IP addresses to avoid these conflicts.

4. What is the profile of DHCP?

Package	:	**dhcp***
Script file	:	**/etc/init.d/dhcpd**
Configuration file	:	**/etc/dhcp/dhcpd.conf**
Deamon	:	**dhcpd**
Port numbers	:	**67 (dhcp server)** and **68 (dhcp client)**
Log messages	:	**/var/log/messages**

(v) Restart the DHCP services in RHEL - 6 and RHEL - 7.

```
# service  dhcpd  restart                (to restart  the  DHCP  service  in  RHEL - 6)
# chkconfig  dhcpd  on                   (to enable the  DHCP service  at next boot  in  RHEL - 6)
# systemctl  restart  dhcpd              (to restart  the  DHCP  service  in  RHEL - 7)
# systemctl  enable  dhcpd                (to enable the  DHCP service  at next boot  in  RHEL - 7)
```

6. How to configure the DHCP client?

(i) Change the IP addressing from static to dynamic if it is configured as static.

<u>In RHEL - 6</u> :

```
# setup
  Network  Configuration  ----->  Press Enter  ----->  Device  Configuration  ----->  Select  eth0  ----->
  Press Enter  ----->  Select  Use  DHCP  ----->  Press  Spacebar  ----->  OK  ----->  Save  ----->Save & Quit
  ----->  Quit
# service  NetworkManager  restart
# service  network  restart
```

<u>In RHEL - 7</u>:

```
# nmcli  connection  modify  "System  eth0"  ipv4.method   auto  or  dynamic
# nmcli  connection  down  "System  eth0"
# nmcli  connection  up  "System  eth0"
# systemctl  restart  network
```

(ii) Open /etc/sysconfig/network-scripts/ifcfg-eth0 file and edit the BOOTPROTO line.

```
# vim  /etc/sysconfig/network-scripts/ifcfg-eth0
  *  Go to  BOOTPROTO  line  and  edit that line as below.
  BOOTPROTO=dhcp                                                  (save  and  exit  this  file)
```

(iii) Get the IP address from the DHCP server.

```
# dhclient
# ifdown  eth0
# ifup  eth0
```

20. Web Server (Apache)

1. What is Web server and explain it?

A Web server is a system that delivers content or services to end users over the Internet. A Web server consists of a physical server, server operating system (OS) and software used to facilitate HTTP communication.

A computer that runs a Web site. Using the HTTP protocol, the Web server delivers Web pages to browsers as well as other data files to Web-based applications. The Web server includes the hardware, operating system, Web server software, TCP/IP protocols and site content (Web pages, images and other files). If the Web server is used internally and is not exposed to the public, it is an "intranet server" and if the Web server is used in the internet and is exposed to the public, it is an Internet server.

5. What is Apache Web Server?

Apache is a open source web server. It is mostly used web server in the internet. httpd is the deamon that speaks the http or https protocols. It is a text based protocol for sending and receiving the objects over a network connection. The http protocol is sent over the wired network in clear text using default port number 80/tcp. To protect the website we can use https web server for data encryption.

6. What is the profile for Web server?

Package	:	***httpd***
script	:	***/etc/init.d/httpd***
Deamon	:	***httpd***
Configuration file	:	***/etc/httpd/conf/httpd.conf*** *(for http)*
		/etc/httpd/conf.d/ssl.conf *(for https)*
Document Root	:	***/var/www/html***
Log files	:	***/var/log/httpd/access_log***
		/var/log/httpd/error_log
Port Number	:	***80/http and 443/https***

7. How to make the http web server available to the cleint?

(a) *First assign the static IP address and hostname to the server.*

(b) *Check whether the server package by **# rpm -qa httpd*** command.*

(c) *If not installed, install the web server package by **# yum install httpd* -y** command.*

(d) *Start the web server and enable web server service at next boot.*

```
# service  httpd start                  (to start the webserver  deamon  in  RHEL - 6)
# chkconfig  httpd on                   (to enable the service at next boot  in  RHEL - 6)
# systemctl  restart httpd              (to start the webserver  deamon  in  RHEL - 7)
# systemctl  enable  httpd              (to enable the service at next boot  in  RHEL - 7)
# service  httpd start                  (to start the webserver  deamon  in  RHEL - 6)
# chkconfig  httpd on                   (to enable the service at next boot  in  RHEL - 6)
# systemctl  restart  httpd                    (to start the webserver  deamon  in  RHEL - 7)
# systemctl  enable  httpd                     (to enable the service at next boot  in  RHEL - 7)
```

9. ***How to configure the name based web hosting?***

(a) *Make a directory for virtual or named based hosting.*

```
# mkdir   /var/www/virtual
```

(b) *Go to the configuration file directory by* ***# cd /etc/httpd/conf.d***

(c) *Create the configuration for name based hosting.*

```
# vim   /etc/httpd/conf.d/virtual.conf
<VirtualHost    <IP address of the web server>  : 80>
ServerAdmin   root@<hostname of the web server>
ServerName    <virtual hostname of the web server>
DocumentRoot    /var/www/virtual
         </VirtualHost>

         <Directory     "/var/www/virtual">
AllowOverride   none
Require  All  Granted
</Directory>                                   (save  and  exit  this  file)
```

Example ***:***

```
# vim   /etc/httpd/conf.d/virtual.conf         (create  the configuration file)
<VirtualHost    172.25.9.11:80>
ServerAdmin         root@server9.example.com
ServerName          www9.example.com
DocumentRoot        /var/www/virtual
         </VirtualHost>

         <Directory     "/var/www/virtual">
AllowOverride  none
Require  All  Granted
</Directory>
```

(e) *Restart the web server deamon.*

```
# service  httpd start                         (to start the webserver  deamon  in  RHEL - 6)
# chkconfig  httpd on                          (to enable the service at next boot  in  RHEL - 6)
# systemctl  restart  httpd                    (to start the webserver  deamon  in  RHEL - 7)
# systemctl  enable  httpd                     (to enable the service at next boot  in  RHEL - 7)
```

(i) *Restart the web server deamon.*

```
# service  httpd start                         (to start the webserver  deamon  in  RHEL - 6)
# chkconfig  httpd on                          (to enable the service at next boot  in  RHEL - 6)
# systemctl  restart  httpd                    (to start the webserver  deamon  in  RHEL - 7)
# systemctl  enable  httpd                     (to enable the service at next boot  in  RHEL - 7)
```

12. ***How to restrict the web sites access from hosts or domains or networks?***

(a) *Go to the configuration file directory by* ***# cd /etc/httpd/conf.d***

(b) *Create the configuration for IP based hosting.*

```
# vim   /etc/httpd/conf.d/restrict.conf
<VirtualHost    172.25.9.11:80>
```

```
ServerAdmin    root@server9.example.com
ServerName     server9.example.com
DocumentRoot    /var/www/html
            </VirtualHost>

            <Directory     "/var/www/html">
AllowOverride   none
Require  All  Granted
Order  Allow,  Deny
Allow   from    172.25.9.0   or   172.25.0    (allows  172.25.9  network   or   172.25  network  to  access  the websites)
Deny  from    .my133t.org           (deny  all the systems  of   *.my133t.org   domain to access  the websites)
</Directory>
```

13.

(d) Restart the web server deamon.

```
# service  httpd  start                    (to start the webserver  deamon  in  RHEL - 6)
# chkconfig  httpd  on                   (to enable the service  at next boot  in  RHEL - 6)
# systemctl  restart  httpd                    (to start the webserver  deamon  in  RHEL - 7)
# systemctl  enable  httpd                    (to enable the service  at next boot  in  RHEL - 7)
```

15. How to configure the directory based web hosting?

(a) Go to the configuration file directory by ***# cd /etc/httpd/conf.d***

(b) Create the configuration for direct based hosting.

```
# vim   /etc/httpd/conf.d/confidential.conf
<VirtualHost     172.25.9.11:80>
ServerAdmin    root@server9.example.com
ServerName     server9.example.com
DocumentRoot    /var/www/html
            </VirtualHost>

            <Directory     "/var/www/html/confidential">
AllowOverride   none
Require  All  Granted
</Directory>                                             (save  and  exit  this  file)
```

(c) Create confidentialdirectory in /var/www/html.

```
# mkdir    /var/www/html/confidential
```

(c) Go to confidential directory and create the index.html file.

```
# cd   /var/www/html/confidential
# vim   index.html
            <html>
                        <H1>
                                    This is Alias based  Web Hosting
                        </H1>
            </html>                                  (save  and  exit  this  file)
```

(d) Restart the web server deamon.

```
# service  httpd  start                    (to start  the webserver  deamon  in  RHEL - 6)
# chkconfig  httpd  on                   (to enable the service  at next boot  in  RHEL - 6)
```

```
        # systemctl  restart  httpd                (to start the webserver  deamon  in  RHEL - 7)
        # systemctl  enable  httpd                 (to enable the service at next boot  in  RHEL - 7)
```

(e) Add the service to the IP tables and firewall.

(d) Restart the web server deamon.

```
        # service  httpd  start                 (to start the webserver  deamon  in  RHEL - 6)
        # chkconfig  httpd  on                  (to enable the service at next boot  in  RHEL - 6)
        # systemctl  restart  httpd             (to start the webserver  deamon  in  RHEL - 7)
        # systemctl  enable  httpd              (to enable the service at next boot  in  RHEL - 7)
```

Package	:	**mod_ssl**
Configuration file	:	**/etc/httpd/conf.d/ssl.conf**
Private key location	:	**/etc/pki/tls/private**
Public key location	:	**/etc/pki/tls/certs**
Authentication certificate	:	**/etc/pki/tls/certs**
Port number	:	**443**

* Private key extention is **" . key "** and public key extention is **" . crt "**

Other useful commands **:**

```
# httpd   -t                    (to check the web server configuration file for syntax errors)
```

21. Mail Server

1. What is mail server?

A mail server (sometimes also referred to an e-mail server) is a server that handles and delivers e-mail over a network, usually over the Internet. A mail server can receive e-mails from client computers and deliver them to other mail servers. A mail server can also deliver e-mails to client computers. A client computer is normally the computer where you read your e-mails, for example your computer at home or in your office. Also an advanced mobile phone or Smartphone, with e-mail capabilities, can be regarded as a client computer in these circumstances.

2. How many types of mail servers available in Linux?

There are two types of mail servers.

(i) **Sendmail server** (default in RHEL - 5, available in 6 and 7)

(ii) **Postfix** (default in RHEL - 6 and 7)

These both mail server are used to send and receive the mails, but we cannot used both mail servers at a time ie., we have to use only one server at a time. These mail servers are used as CLI mode. Outlook express in windows is used to send or receive the mails. Thunderbird is used to send or receive the mails using GUI mode in Linux. **# mail** is the command used to send the mails in CLI mode.

3. What are MUA, MTA, SMTP, MDA and MRAs?

MUA **:**

MUA stands for Mail User Agent. It is the e-mail client which we used to **create-draft-send** emails. Generally Microsoft Outlook, Thunderbird, kmail, ...etc., are the examples for MUAs.

MTA **:**

MTA stands for Mail Transfer Agent. It is used to transfer the messages and mails between senders and recipients. Exchange, Qmail, Sendmail, Postfix, ...etc., are the examples for MTAs.

SMTP**:**

SMTP stands for Simple Mail Transfer Protocol. It is used to transfer the messages and mails between the MTAs.

MDA **:**

MDA stands for Mail Delivery Agent. It is a computer software component that is responsible for the delivery of e-mail messages to a local recipient's mailbox. Within the Internet mail architecture, local message delivery is achieved *through a process of handling messages from the message transfer agent, and storing mail into the recipient's environment (typically a mailbox).*

MRA :

MRA stands for Mail Retrieval Agent. It is a computer application that retrieves or fetches e-mail from a remote mail server and works with a mail delivery agent to deliver mail to a local or remote email mailbox. MRAs may be external applications by themselves or be built into a bigger application like an MUA. Significant examples of standalone MRAs include fetchmail, getmail and retchmail.

4. What is the profile of mail server?

Package	:	***sendmail*** *(in RHEL - 5, 6 and 7) or* ***postfix*** *(in RHEL - 6 and 7).*
Configuration file	:	***/etc/postfix/main.cf, /etc/dovecot/dovecot.conf***
Log file	:	***/var/log/mail.log***
User's mails location	:	***/var/spool/mail/<user name>***
root user's mail location	:	***/var/spool/mail/root***
Deamons	:	***postfix***
Port number	:	***25***

5. How to configure the mail server?

The pre-requisite for mail server is DNS. ie., Domain Naming System should be configured first.

(i) Check the hostname of the server by ***# hostname*** *command.*

(ii) Install the mail server package by ***# yum install postfix* dovecot* -y*** *command.*

(iii) Open the mail configuration file and at last type as below.

```
# vim   /etc/postfix/main.cf
myhostname  =  server9.example.com
mydomain  =  example.com
myorigin  =  $mydomain
inet_interfaces   =  $myhostname,   localhost
mydestination  =  $myhostname,   localhost.$localdomain,  localhost,  $mydomain
home_mailbox  =  Maildir  /                                   (save  and  exit  this  file)
```

(iv) Open the another configuration file and at last type as below.

```
# vim   /etc/dovecot/dovecot.conf
protocols  =  imap    pop3    lmtp                    (save  and  exit  this  file)
```

(v) Restart the mail server services.

```
    # service  postfix  restart          (to restart  the postfix deamon in  RHEL  - 6)
    # service  dovecot  restart          (to restart  the dovecot deamon in  RHEL  - 6)
    # chkconfig  postfix  on      (to enable  the postfix deamon at next boot in  RHEL  - 6)
    # chkconfig  dovecot  on      (to enable  the dovecot deamon at next boot in  RHEL  - 6)
# systemctl  restart postfix  doveco0t (to restart  the  postfix  and dovecot  deamons in  RHEL - 6)
    # systemctl  enable  postfix  dovecot(to enable the  deamons at next boot in  RHEL  - 6)
```

(vi) Add the service to the IP tables and firewall.

```
# mail   -s   testmail    raju
Hi  this is a test  mail
ok  bye… bye ….                                   (exit  and  send the  mail by  Ctrl + d )
```

6. How to configure mail server as null client in RHEL - 7 ?

(i) Open the configuration file and at last type as below.

```
# vim   /etc/postfix/main.cf
relayhost  =  [client9.example.com]
inet_interfaces   =  loopback-only
mynetworks  =  127.0.0.0/8     [ : : 1]/128
```

```
myorigin  =  server9.example.com
mydestination  =
local_transport  = error :  local  delivery  disabled                    (save  and  exit  this  file)
```

(ii) Restart the postfix deamons.

# systemctl restart postfix
# systemctl enable postfix

(iv) Send a test mail to the user.

# mail -s testmail raju or # mutt -s testmail raju

```
 Hi  this is a test  mail
 ok  bye... bye ....                                              (exit  and  send the  mail by  Ctrl + d )
```

```
# mail   -s   "hello"<user  name>@<servername>  . <domain name>   (to send the mail to the remote  system)
# mailq                                                          (to see the mails in the queue)
```

* If the mail server is not configured or not running, then the sent mails will be in the queue.

```
# mail   -s   "hello"<user  name1><user  name2><<File  name>    (send the mail  with attached file to  the  2  users)
```

22. iSCSI (Remote Storage)

1. What is storage?

The memory where we can store the data, such as files, directories, ...etc., is called the storage. Storage is mainly two types. (i) Local storage and (ii) Remote Storage.

(i) ***Local storage* :**

Local storage is a storage which is directly connected to our system and ready to use.

***Example* :** Local hard disk, local pen drive, DAS (Direct Access Storage) ... etc.,

(ii) ***Remote storage* :**

The storage which is not connected to our system directly but allotted some space to our system in remote location is called remote storage.

***Example* :** iSCSI (Internet Small Computer System Interface), SAN (Storage Area Network), NAS (Network Area Storage)

2. What is iSCSI and explain it?

iSCSI is a way of connecting storage devices over a network using TCP/IP. It can be used over a local area network (LAN), a wide area network (WAN), or the Internet.

iSCSI devices are disks, tapes, CDs, and other storage devices on another networked computer that you can connect to. Sometimes these storage devices are part of a network called a **Storage Area Network** (SAN).

In the relationship between our computer and the storage device, our computer is called an **initiator** because it initiates the connection to the device, which is called a **target**.

iSCSI provides Remote Block or File Storage. Most data centers keep their storage in centralised SAN racks. iSCSI provides an inexpensive alternative to proprietary SAN hardware.

3. What is the terminology of iSCSI?

iSCSI supports sending SCSI commands from clients (initiators) over IP to SCSI storage devices (targets) on remote systems (servers). **iqn** is a iSCSI qualified name or number.

The format of iqn is **"iqn.yyyy-mm.<domain name in reverse order>**label is used to identify initiators and targets communicate through port number 3260.

23. MySQL Server or MariaDB

1. ***What is MySQL or MariaDB?***

MySQL or MariaDB is a database software to create and maintain the databases.
Upto RHEL - 6 the database software is MySQL and from RHEL - 7 onwards the database software is MariaDB.
If we want to do any transactions or database operations, we have to open the ***mysql >*** *or* ***mariadb >*** *prompt.*
*In MySQL or MariaDB all the database operation commands will end with a "****;*** ***"*** *(semicolon).*

2. ***What is the profile of MySQL or MariaDB?***

Package	:	***mysql**** *(in RHEL - 6) and* ***mariadb**** *(in RHEL - 7)*
Version	:	***5.0*** *(in RHEL - 6) and* ***5.5*** *(in RHEL - 7)*
Deamons	:	***mysqld*** *(in RHEL - 6) and* ***mariadb*** *(in RHEL - 7)*
Configuration file	:	***/etc/my.cnf***
Installation *Commands*	:	***mysql_secure_installation***

3. ***How to configure MySQL or MariaDB?***

(i) Install the MySQL or Mariadb software packages.

```
# yum groupinstall mysql*   -y          (to install MySQL in RHEL - 6)
# yum groupinstall mariadb*             (to install Mariadb in RHEL - 7)
```

(ii) Restart the mysqld and mariadb deamons.

```
# service mysqld restart       (to start the mysqld deamon in RHEL - 6)
# chkconfig mysqld on          (to enable the mysqld deamon at next boot in RHEL - 6)
# systemctl restart mariadb    (to start the mysqld deamon in RHEL - 7)
# systemctl enable mariadb     (to enable the mysqld deamon at next boot in RHEL - 7)
```

(iv) If we want to configure the database as localhost i.e., database will not be available to remote systems.

```
# vim /etc/my.cnf     (open this file and go to 2nd line, create an empty line and type as below)
  skip-networking=1
:wq     (save and exit this file)
```

(v) Restart the mysqld and mariadb deamons.

```
# service mysqld restart       (to start the mysqld deamon in RHEL - 6)
# chkconfig mysqld on          (to enable the mysqld deamon at next boot in RHEL - 6)
# systemctl restart mariadb    (to start the mysqld deamon in RHEL - 7)
# systemctl enable mariadb     (to enable the mysqld deamon at next boot in RHEL - 7)
```

(vi) Install the database engine. *(it works in both RHEL - 6 & 7)*

```
# mysql_secure_installation
  Enter current root password  :  (here do not enter any passwords and just press the Enter Key)
  Set root password [y/n]  :  y
  Remove ananymous users [y/n]  :  y
  Disallow root login remotely [y/n]  :  y
  Remove test database and access to it [y/n]  :  y
  Reload the privilages tables now [y/n]  :  y
```

(vii) Login into the mysql server as a root user.

```
# mysql  -u  root  -p       (where  u  ----->  user  and  p  ----->  using password)
   (we have to enter the password for root user)
```

(viii) See the default databases.

```
mysql > show databases;            (in RHEL - 6)
mariadb > show databases;          (in RHEL - 7)
```

(ix) Exit from the database by ***mysql > exit;*** *(in RHEL - 6) and* ***mariadb > exit;*** *(in RHEL - 7)*

4. ***How to create a database, create tables, enter the data into the tables and access that data?***

(i) Login into the database server by ***# mysql -u root -p*** *command.*

(ii) Create the database and connect the databases.

mysql or mariadb > create database <database name>; (to create the database)

mysql or mariadb>show databases; (to see all the databases in the server)

mysql or mariadb > use <database name>; (to connect to the specified database)

(iii) Create a table, enter the data and query the data.

mysql or mariadb > create table <table name> (field name1 data type (size),
field name2 data type (size),
field name3 data type (size));

<u>**Example**</u> **:** mysql or mariadb > create table mydetails (Name varchar (30), status varchar (10),
Address varchar (50), phone int (10));

(iv) See the structure of the table.

mysql or mariadb >describe<table name>; (to see the structure of the table)

<u>**Example**</u> **:** mysql or mariadb > describe mydetails;

(v) Insert or enter the data into the table.

mysql or mariadb > insert into mydetails values ("Raju", "Single", Hyderabad", 9848750755);

(vi) Query the table to get the data.

mysql or mariadb >select * from mydetails; (to see all the records of the tables)

mysql or mariadb > select name, phone from mydetails; (to select the wanted data ie., filtering the data)

5. How to take a backup of the database, drop the database and restore the database using backup?

To take a backup or restore of the database first we should comeout from the database server and then take a backup or restore the backup.

(i) Exit the from the database server.

mysql or mariadb > exit;

(ii) Take a backup of the database.

# mysqldump -u root -p <database name>><file name with full path>

<u>**Example**</u> **:** # mysqldump -u root -p mydetails > /root/mydetails.bak

(iii) Delete the database from the database server.

mysql or mariadb >drop database <database name>;

<u>**Example**</u> **:** mysql or mariadb > drop database mydetails;

(iv) Restore the deleted database using the backup copy.

mysql or mariadb >exit;

# mysql -u root -p <database name><<backup file name with path>

<u>**Example**</u> **:** # mysql -u root -p mydetails < /root/mydetails.bak

6. How to create the user in the database and make the user to do transactions or operations?

(i) To create the user in the database first login to the database and then create the user.

mysql or mariadb > create user <user name>@<host name> identified by "<password>";

<u>**Example**</u> **:** mysql or mariadb > create user raju@localhost or server9.example.com identified by
"raju123";

(ii) Make the user to do transactions on the database. (nothing but granting the permission)

mysql or mariadb > grant select, insert, update, delete on <database name>.* to <user name>;

or

mysql or mariadb > grant all on <database name> .* to <user name>;

<u>**Example**</u> **:** mysql or mariadb > grant select, insert, update, delete on mydetails .* to raju; or
mysql or mariadb > grant all on mydetails .* to raju;

(where database . * means granting permissions on all the contents like tables, indexes, views,
synonyms and others)

24. Log Server and Log Files

1. What is log server?

A log server represents a central log monitoring point on a network, to which all kinds of devices including Linux or Windows servers, routers, switches or any other hosts can send their logs over network. By setting up a log server, you can filter and consolidate logs from different hosts and devices into a single location, so that you can view and archive important log messages more easily.

On most Linux distributions, `rsyslog` *is the standard syslog daemon that comes pre-installed. Configured in a client/server architecture,* `rsyslog` *can play both roles; as a syslog server* `rsyslog` *can gather logs from other devices, and as a syslog client,* `rsyslog` *can transmit its internal logs to a remote syslog server.*

When logs are collected with syslog mechanism, three important things must be taken into consideration:

***Facility level:** what type of processes to monitor*

***Severity (priority) level:** what type of log messages to collect*

***Destination:** where to send or record log messages*

2. What is the profile of log server?

This is also called as rsyslog server. The requirements are given below.

(i)	Package	:	**rsyslog***
(ii)	Deamon	:	**rsyslog**
(iii)	Port No.	:	**514**
(iv)	Configuration file	:	**/etc/rsyslog.conf**

3. How to configure the log server?

(i) *Install rsyslog package by* **# yum install rsyslog* -y** *command.*

(ii) *Open the log server configuration and file and edit as per requirements.*

```
# vim /etc/rsyslog.conf
```

Go to line no. : 15 & 16 and uncomment on those lines. *(save and exit this file)*

(iii) *Restart the log server deamon in RHEL - 6 and RHEL - 7.*

```
# service rsyslog restart        (to restart the log server deamon in RHEL - 6)
# chkconfig rsyslog on           (to enable the log server deamon at next boot in RHEL - 6)
# systemctl restart rsyslog      (to restart the log server deamon in RHEL - 7)
# systemctl enable rsyslog       (to enable the log server deamon at next boot in RHEL - 7)
```

(iv) *Verify whether the log server is listening or not.*

```
# netstat -ntulp | grep 514
```

5. What is log file?

Log file is file that contains messages about that system, including the kernel, services and applications running on it, ...etc., There are different log files for different information. These files are very useful when trying to troubleshoot a problem with systems.

Almost all log messages are stored in ***/var/log*** *directory. Only root user can read these log messages. We can use less or more commands to read these log files. The messages will be generated only when rsyslog service is running, otherwise the log messages will not be generated.*

<u>The different types of log files and their locations</u> :

/var/log/messages *-----> System and general messages and DHCP log messages.*

/var/log/authlog *-----> Authentication log messages.*

/var/log/secure *-----> Security and authentication and user log messages.*

/var/log/maillog *-----> Mail server log messages.*

/var/log/cron *-----> Cron jobs log messages.*

/var/log/boot.log *-----> All booting log messages.*

/var/log/httpd *-----> All Apache web server log messages.*

/var/log/mysqld.log *-----> Mysql database server log messages.*

/var/log/utmp *or* ***/var/log/wtmp*** *-----> All the user's login messages.*

/var/log/Qmail *-----> Qmail log messages.*

/var/log/kernel.log *-----> All kernel related log messages.*

/var/log/samba *-----> All samba server log messages.*

/var/log/anakonda.log *-----> Linux installation log messages.*

/var/log/lastlog *-----> Recent login information for all users.*

# lastlog *(to see the log messages of the above log file)*

/var/log/yum.log *-----> All package installation log messages generated by # yum or # rpm commands.*

/var/log/cups *-----> All printer and printing related log messages.*

/var/log/ntpstat *-----> All ntp server and services log messages.*

/var/log/spooler *-----> Mail, printer and cron jobs spooling messages.*

/var/log/sssd *-----> System security service deamon log messages.*

/var/log/audit.log *-----> SELinux log messages.*

# dmesg *(to see the boot log messages)*

# tailf or # tail -f /var/log/secure *(to check or watch the log files continuously)*

25. Configuring IP tables and Firewall

1. ***What are IPtables or firewalls?***

IP tables is a command-line firewall utility that uses policy chains to allow or block traffic. When a connection tries to establish itself on your system, iptables looks for a rule in its list to match it to. If it doesn't find one, it resorts to the default action. IP tables almost always comes pre-installed on any Linux distribution.

We can update/Reinstall the IP tablespackage by ***# yum install iptables* -y*** *command.*

2. ***What are the types of firewalls?***

7. ***In how many ways can we protect the network?***

There are 4 ways to protect the network.

(i)SELinux
(ii)IP tables
(iii) Firewalld
(iv) TCP wrappers

IP tables and firewalld both are used to protect our systems services from outside. But we can use only one way at a time.

8. ***How to configure the firewalld?***

(i) Install the firewalld package by ***# yum install firewalld* -y*** *command.*

(ii) Check whether the firewalld package is installed or not by ***# rpm -qa firewalld*** *command.*

(iii) Check the status of the firewalld by executing the below commands.

# systemctl status firewalld *or* ***# firewall-cmd --status***

<u>Examples of IP tables commands</u> :

# service iptables status	*(to check the IP tables status)*
# service iptables start	*(to start the IP tables)*
# service iptables stop	*(to stop the IP tables)*
# service iptables restart	*(to restart the IP tables)*
# service iptables save	*(to save the iptable rules permanently)*
# chkconfig iptables on	*(to enable the iptables at next boot)*
# chkconfig iptables off	*(to disable the iptables at next boot)*

26. Virtualization

1. What is virtualization?

Virtualization allows multiple operating system instances to run concurrently on a single computer;it is a means of separating hardware from a single operating system. Each "guest" OS is managed bya Virtual Machine Monitor (VMM), also known as a hypervisor. Because the virtualization system sitsbetween the guest and the hardware, it can control the guests' use of CPU, memory, and storage,even allowing a guest OS to migrate from one machine to another.

2. What are types of virtualizations available in Linux?

RHEL - 5 **:**	***RHEL - 6 & 7*** **:**
xen	*kvm*
64 bit	*64 bit*
VT-Enabled	*VT-Enabled*
Intel/AMD	*Intel/AMD*
2 GB RAM	*2 GB RAM*
6 GB Hard disk	*6 GB Hard disk*

Other useful commands **:**

# lscpu	*(to list the CPU information)*
# cat /proc/cpuinfo	*(to display the CPU information)*

27. General Questions

1. ***Tell me about yourself ?***

   (i) Tell your personal details

   (ii) Technical (Educational details)

2. ***Tell me about your profile?***

   (i) Tell your personal details.

   (ii) Educational details.

   (iii) Work history (previous companies).

   (iv) Profile (Present company) :

   (a) ***Coming to Linux :*** *(upto till date)*

   (1) O/S installation.

   (2) File system creation.

   (3) User administration like user creation, user permissions, profiles, setting environment to user, giving special permissions (sudo and ACLs) to them and user troubleshooting issues like user unable to login password requests.

   (4) Hardware related issues like adding disks, NIC cards, processor replacement, memory replacement, increase memory and power supply replacement, ….etc.,

   (5) Network related issues like providing networking, setting NIC card parameters, troubleshooting issues.

   (6) Some internal backups.

   (7) O/S patching and package administration whenever needed using ***rpm*** *and* ***yum****.*

   (8) I also supports process related issues like memory utilization full (90%), CPU utilization full (90%) and file system full, …etc.,

   (9) I also support for system troubleshooting issues like system not responding, node down, starting and stopping services and deamons.

   (b) ***Coming to Veritas Volume Manager :*** *(from the last 1 year)*

(1) We get requests from production, database, Q A people like creating volumes, file system creation, increase and (or) decrease the volume sizes, provide permissions, redundancy, put the volume into cluster to provide high availability,

(2) sometimes destroy or remove the volumes, backup and restore whenever necessary,

(3) We also get some troubleshooting issues like volume not started, volume not accessible, file system crashed, mount point deleted, disks failed, volume manager deamons are not working, configuratio files missed, crashed, disk groups not deporting and not importing, volume started but users are unable to access file systems on those volumes,…etc.,

   (c) ***Coming to Veritas Cluster :*** *(from 6 months)*

   (1) We get requests like node adding, resource adding, service group adding, adding service groups and resources to existing service groups, mount points adding, adding NIC cars, IP addresses, adding volumes, disk groups, freezing and unfreezing services groups and also get some troubleshooting issues like cluster not running, if resources faulted then restart the service groups, communication failed between two systems, Gab is not running, llt not running, and configuration files main.cf crashed or missed and resources are not started, … etc.

   (d) I also write small scripts to perform internal routine jobs, document preparation, handover mails checking, how many tickets issued, how many tickets solved and how many jobs pending, ….etc.,

   (e) I also supports in application deployment, database deployment and others.

3. ***What are the tools you are using?***

   (i) netstat, vmstat, iostat, nmap and top for performance monitoring tools.

   (ii) cron and at for job scheduling.

   (iii) Remedy tool for ticketing system.

   (iv) Veritas Netbackup, Tivoli, …. etc., for backing purpose

   (v) Outlook for internal mailing.

5. What are the Applications are you using?

(i) Databases (Oracle 10g, 11g and Mysql).
(ii) Oracle Applications like ERP packages (Oracle 11i and 12).
(iii) SAP applications.
(iv) Datawarehousing, ….etc.,

6. What is your company hierarchy?

Me -----> Team Lead or Tech Lead -----> Manager -----> Delivery Manager -----> Asia head

8. What are your shift timings?

General shift -----> 09:00 - 18:00 hrs.

9. What is your team size?

Total 18 members. For each shift 5 members each and 3 members on weekly off.

10. What about tickets issues and tickets frequency?

(i) 7 - 8 tickets daily and Max. 10 per day.
In those 85 - 90% are CPU utilization full, memory full, file system full, login problems and sometimes node down issues.
(ii)General tickets severity - 3, severity - 2, severity - 1.
We are not resolved severity level - 1 tickets.
(iii) Incidents :
Severity level - 1 should be solved within 1 hour (Immediate).
Severity level - 2 should be solved within 6 hours.
Severity level - 1 should be solved within 24 hours.
Severity level - 1 should be solved within 2 days.
Request priority ----> High, medium and low

11. What is your notice period?

25 - 30 days.
(i) Databases.
(ii) Banking.
(iii) Finance.
(iv) Logistics.
(v) Hotel and Tourism, …..etc.,

36. What is grep?

(i) **grep** means Globally search for Regular Expression.
(ii) Using grep we can filter the results to get a particular information.
(iii) We can get only information about what string we have specified in grep command.

40. What is Linux Kernel?

It acts as an interpreter between Linux OS and its hardware. It is the fundamental component of Linux OS and contains hardware drivers for the devices installed on the system. The kernel is a part of the system which loads first and it stays on the memory.

48. What do you understand the Load Average?

If the number of active tasks utilizing CPU is less as compared to available CPU cores then the load average can be considered normal but if the no. of active tasks starts increasing with respect to available CPU cores then the load average will start rising.**For example,**
#uptime 00:43:58 up 212 days, 14:19, 4 users, load average: 6.07, 7.08, 8.07

49. How to check all the current running services in Linux?

<u>To find the status of any single service</u>

```
# service vsftpd status

  vsftpd (pid 5909) is running...
```

To get the status of all the running services ***:***

```
# service --status-all | grep running
```

56. Which log file will you check for all authentication related messages?
/var/log/secure

66. What is the difference between A record and CNAME record in DNS?

A record ***:***

It is the Address records also known as host records
Points to the IP address reflecting the domain
Used for forward lookup of any domain name

For example:
Our website is configured on 50.63.202.15 IP so the A record of my domain name will point towards that IP. Every time a query for golinuxhub.com is made the internet will lookup for contents stored on the machine with 50.63.202.15 this IP.

CNAME Record ***:***

It is short abbreviation for Canonical Name
Provides an alias name for same hostname
Helps create subdomains

NOTE: *You cannot create a CNAME record for the domain name itself (it should be done with A record)*

For example:
golinuxhub.com is a domain name whereas www.golinuxhub.com is a sub domain name.

(f)Configure the yum server.

```
# vim /etc/yum.repos.d/linux.repo
  [linux]
  name=Linux yum server
   baseurl=ftp://172.25.9.11/pub/rhel6          (Specify the FTP server IP address)
   gpgcheck=0
   enabled=1                                          (save and exit the file)
# yum clean all
# yum repolist
```

(g) Configure the DHCP server.

```
# yum install dhcp* -y
# cp -rvpf /usr/share/doc/dhcp-4.1.1/dhcpd.conf.sample /etc/dhcp/dhcpd.conf
```

(h) Configure the TFTP server.

```
# yum install tftp* syslinux* -y
# cp -rvpf /media/RHEL6/isolinux/*.* /var/lib/tftpboot
# mkdir /var/lib/tftpboot/pxelinux.cfg
# cp /var/lib/tftpboot/isolinux.cfg /var/lib/ftfpboot/pxelinux.cfg/default
# cp -rvpf /usr/share/syslinux/pxelinux.0 /var/lib/tftpboot
```

(k) Restart all the services once again.

```
# service network restart
# chkconfig network on
# service vsftpd restart
# chkconfig vsftpd on
# service dhcpd restart
# chkconfig dhcpd on
```

31. Examples of top command

top is one of the tool for monitoring system usage and also to make any change for improving system performance.

Introduction:

The top program provides a dynamic real-time view of a running system. It can display system summary information as well as a list of tasks currently being managed by the Linux kernel. The types of system summary information shown and the types, order and size of information displayed for tasks are all user configurable and that configuration can be made persistent across restarts.

1. *Without any arguments :*

```
     #                                                                   top
     top - 17:51:07 up 1 day,  2:56, 27 users,  load average: 5.33, 29.71,
28.33
     Tasks: 1470 total,   1 running, 1469 sleeping,   0 stopped,   0 zombie
     Cpu(s):             0.0%us,              0.1%sy,              0.0%ni,
99.9%id,        0.0%wa,         0.0%hi,         0.0%si,         0.0%st
     Mem:  264114424k total, 253006956k used, 11107468k free,     66964k
buffers
     Swap:  33554424k total,      3260k used, 33551164k free, 245826024k
cached

     PID   USER       PR  NI  VIRT  RES  SHR  S %CPU %MEM    TIME+  COMMAND
      1960     deepak  15   0 30452 3220 1540 R  2.3  0.0   0:00.78 top
     2457 root       11  -5     0    0    0 S  2.3  0.0  11:36.93 kacpid
     2493 pmartprd   16    0  1397m  289m  9.8m  S   0.3   0.1   18:36.07
pmrepagent
     4639 pmartprd  15   0  787m  54m 4080 S  0.3  0.0   5:19.55 pmserver
     14402     root       RT   0  151m 5256 2872 S  0.3  0.0   1:41.40
multipathd
     17886     root       10  -5     0    0    0 S  0.3  0.0   0:07.41
kondemand/11
```

Generally we use top without any arguments, but the magic is mostly done from the top command line which must of us skip. Well before taking you to that part let me explain you the various system related features which are shown by top command.

***NOTE:** You can enable or disable the marked blue line by pressing "**l**" once top is running.*

```
     top - 17:51:07 up 1 day,  2:56, 27 users,  load average: 5.33, 29.71,
28.33
     Tasks: 1470 total,   1 running, 1469 sleeping,   0 stopped,   0 zombie
     Cpu(s):             0.0%us,              0.1%sy,             0.0%ni,
99.9%id,        0.0%wa,         0.0%hi,         0.0%si,         0.0%st
     Mem:  264114424k total, 253006956k used, 11107468k free,     66964k
buffers
```

```
Swap:  33554424k  total,       3260k  used,  33551164k  free,  245826024k
cached
```

Explanation: This line tells you about the uptime of your system along with load average value.

NOTE: You can enable/disable the marked blue line by pressing "**t**".

```
top - 17:51:07 up 1 day,  2:56, 27 users,  load average: 5.33, 29.71,
28.33
Tasks: 1470 total,   1 running, 1469 sleeping,   0 stopped,   0 zombie
Cpu(s):              0.0%us,              0.1%sy,              0.0%ni,
99.9%id,         0.0%wa,          0.0%hi,          0.0%si,          0.0%st
Mem:  264114424k total, 253006956k used, 11107468k free,      66964k
buffers
Swap:  33554424k  total,       3260k  used,  33551164k  free,  245826024k
cached
```

Explanation: This line gives us a brief detail of all the tasks running/sleeping/stopped currently in the system along with the CPU Usage

Value	Meaning
us	user cpu time (or) % CPU time spent in user space
sy	system cpu time (or) % CPU time spent in kernel space
ni	user nice cpu time (or) % CPU time spent on low priority processes
id	idle cpu time (or) % CPU time spent idle
wa	io wait cpu time (or) % CPU time spent in wait (on disk)
hi	hardware irq (or) % CPU time spent servicing/handling hardware interrupts
si	software irq (or) % CPU time spent servicing/handling software interrupts
st	steal time - - % CPU time in involuntary wait by virtual cpu while hypervisor is servicing another processor (or) % CPU time stolen from a virtual machine

NOTE: You can enable/disable the marked blue line by pressing "**m**".

```
top - 17:51:07 up 1 day,  2:56, 27 users,  load average: 5.33, 29.71,
28.33
Tasks: 1470 total,   1 running, 1469 sleeping,   0 stopped,   0 zombie
Cpu(s):              0.0%us,              0.1%sy,              0.0%ni,
99.9%id,         0.0%wa,          0.0%hi,          0.0%si,          0.0%st
Mem:  264114424k total, 253006956k used, 11107468k free,      66964k
buffers
Swap:  33554424k  total,       3260k  used,  33551164k  free,  245826024k
cached
```

Explanation: The next line shows your memory(RAM and swap) usage and capacity.

```
PID  USER                   PR        NI        VIRT          RES        SHR
S               %CPU             %MEM           TIME+                COMMAND
13916     stmprd    18   0  903m 129m 9936 S 51.4  0.1  3:07.01 java
13921     stmprd    18   0  901m 128m 9936 S 49.8     0.0        3:02.92
java
13825     stmprd    18   0  951m 190m 9932 S 49.5     0.1        3:07.13
java
13856     stmprd    20   0  978m 197m 9936 S 49.2     0.1        3:05.89
java
```

```
13853     stmprd    18   0  921m 150m 9932 S 48.5     0.1       3:09.14
java
13875     stmprd    18   0  907m 132m 9940 S 48.5     0.1       3:09.49
java
13937     stmprd    25   0  926m 165m 9936 S 48.2     0.1       3:10.31
java
13919     stmprd    18   0  917m 153m 9936 S 47.5     0.1       3:05.92
java
13879     stmprd    25   0  921m 160m 9936 S 47.2     0.1       3:08.43
java
13908     stmprd    25   0  901m 131m 9932 S 47.2     0.1       3:12.23
java
13905     stmprd    25   0  907m 137m 9932 S 46.6     0.1       2:59.85
java
```

The left sections shows you the details of the process running along with the below details.

Fields/Column	***Description***
PID	*Process Id*
USER	*The effective user name of the task's owner*
PR	*The priority of the task*
NI	*The nice value of the task. A negative nice value means higher priority, whereas a positive nice value means lower priority. Zero in this field simply means priority will not be adjusted in determining a task's dispatchability*
%CPU	*The task's share of the elapsed CPU time since the last screen update, expressed as a percentage of total CPU time.*
%MEM	*A task's currently used share of available physical memory*
TIME+	*Total CPU time the task has used since it started*
S	*The status of the task which can be one of:* *'D' = uninterruptible sleep* *'R' = running* *'S' = sleeping* *'T' = traced or stopped* *'Z' = zombie*
RES	*The non-swapped physical memory a task has used*
SHR	*The amount of shared memory used by a task*
Command	*Display the command line used to start a task or the name of the associated program*

2. *Arrange Tasks with High to Low CPU Usage :*

*Press "**P**" or "**shift+p**" once top is running to arrange all the tasks with **High to Low CPU Usage** as shown below.*

```
     top - 18:03:00 up 1 day,  3:08, 27 users,  load average: 12.54, 32.34,
32.75
     Tasks: 1485 total,   3 running, 1482 sleeping,   0 stopped,   0 zombie
     Cpu(s):            41.2%us,             0.8%sy,             0.0%ni,
56.6%id,       1.4%wa,        0.0%hi,         0.0%si,         0.0%st
```

```
     Mem:  264114424k total, 258863028k used,  5251396k free,    76308k
buffers
     Swap: 33554424k total,      3256k used, 33551168k free, 250950544k
cached
     PID  USER              PR       NI       VIRT         RES         SHR
S          %CPU            %MEM            TIME+                  COMMAND
      9663     stmprd    22   0      902m 301m 9888 S 2578.3      0.1
     2:27.04   java
     32117     etlprd    18  -1      32416     5908 1716 R    6.2
     0.0       0:04.84                                   cleanup_dirfile
     10053     root       18  -1     27100     1936 1460 S    4.9
      0.0      0:00.15    ps
     5456 pmartprd  16   0     1182m     130m 8560 S  3.9    0.1
     38:39.72  pmserver
     17492     deepak  16   0  30592     3388 1544 R    3.6  0.0
     0:17.11   top
      2843     pmartprd  15   0     730m       48m  4052 S    3.3
      0.0      4:40.33    pmserver
      2457     root       11  -5         0    0    0    S    2.9  0.0
     11:42.39  kacpid
      3731     tdmsprd   15   0     370m       49m  32m  S    2.3
     0.0       0:00.64    pmdtm.orig
```

3. *Arrange Tasks with High to Low Memory Usage.*

*Press "**M**" or "**shift+m**"once top is running to arrange all the tasks with **High to Low Memory Usage** as shown below.*

```
     top - 18:04:26 up 1 day,  3:09, 27 users,  load average: 37.12, 34.56,
33.44
     Tasks: 1676 total,   1 running, 1675 sleeping,   0 stopped,   0 zombie
     Cpu(s):             2.3%us,             76.7%sy,              0.0%ni,
19.7%id,         1.3%wa,         0.0%hi,         0.0%si,         0.0%st
     Mem:  264114424k total, 262605184k used,  1509240k free,    77924k
buffers
     Swap: 33554424k total,      3256k used, 33551168k free, 252198368k
cached
     PID  USER              PR       NI       VIRT         RES         SHR
S       %CPU      %MEM       TIME+        COMMAND
     1852 pmartprd  18   0     2005m     319m 4776 S    6.9  4.1
     28:34.32  java
     2493 pmartprd  16   0     1397m     289m 9.8m S    0.0  4.0
     18:37.79                                              pmrepagent
     20557     etlprd    15   0      911m 201m 3024 S    0.0  3.0
     17:09.02                                              pmdtm.orig
     18778     root      RT   0      286m 188m 156m S    0.0  2.1
     13:24.98  aisexec
     5456 pmartprd  15   0     1182m     130m 8560 S    6.2
     1.1  38:40.58                                            pmserver
     16004     etlprd    14  -1      179m 83m  2636 S    0.0
     0.1  9:41.36    db2bp
```

```
    11272   stmprd   25   0    906m     67m  9736 S    99.7
    0.0  0:48.11   java
```

4. Change the nice value (priority) of any task

To understand what is nice value follow the below link
What is nice and how to change the priority of any process in Linux?

Press "**r**" when top is running on the terminal. You should get a prompt as shown below in blue color.

```
    top - 18:08:38 up 115 days,  8:44,  4 users,  load average: 0.03,
0.03,                                                              0.00
    Tasks: 325 total,   2 running, 323 sleeping,   0 stopped,   0 zombie
    Cpu(s):           0.1%us,           6.4%sy,            0.0%ni,
93.3%id,       0.3%wa,        0.0%hi,        0.0%si,          0.0%st
    Mem:  49432728k total,  2063848k used, 47368880k free,   310072k
buffers
    Swap:  2097144k total,        0k used,  2097144k free,  1297572k
cached
    PID to renice:1308 [Hit Enter]
    PID  USER              PR      NI      VIRT        RES         SHR
S     %CPU     %MEM      TIME+        COMMAND
    5359 root      39      19       0 0    0    R    100.1
    0.0      94:31:35   kipmi0
    1308 deepak  16  0   29492     2292 1512 S    0.7  0.0     0:00.33
    top
    6116 root     15   0     369m     30m  11m  S    0.7
    0.1  77:24.97   cimserver
```

Give the **PID** whose nice value has to be changed and hit "Enter". Then give the **nice value** for the PID

```
    top - 18:08:38 up 115 days,  8:44,  4 users,  load average: 0.03,
0.03,                                                              0.00
    Tasks: 325 total,   2 running, 323 sleeping,   0 stopped,   0 zombie
    Cpu(s):           0.1%us,           6.4%sy,            0.0%ni,
93.3%id,       0.3%wa,        0.0%hi,        0.0%si,          0.0%st
    Mem:  49432728k total,  2063848k used, 47368880k free,   310072k
buffers
    Swap:  2097144k total,        0k used,  2097144k free,  1297572k
cached
    Renice PID 1308 to value:-1 [Hit Enter]
    PID  USER              PR      NI      VIRT        RES         SHR
S     %CPU     %MEM      TIME+        COMMAND
     5359     root      39      19       0      0 0    R        100.1
    0.0      9431:35   kipmi0
    1308 deepak  16  0   29492     2292 1512 S    0.7  0.0
    0:00.33   top
    6116 root      15   0    369m     30m  11m  S    0.7  0.1
    77:24.97 cimserver
```

Verify the changes :

```
    top - 18:09:06 up 115 days,  8:45,  4 users,  load average: 0.13,
0.06,                                                              0.01
    Tasks: 325 total,   1 running, 324 sleeping,   0 stopped,   0 zombie
    Cpu(s):           0.0%us,           0.1%sy,            0.0%ni,
99.8%id,          0.1%wa,          0.0%hi,          0.0%si,          0.0%st
     Mem:  49432728k  total,  2063276k  used,  47369452k  free,    310072k
buffers
     Swap:  2097144k  total,          0k  used,  2097144k  free,  1297588k
cached
     PID  USER               PR       NI       VIRT          RES          SHR
S      %CPU       %MEM       TIME+          COMMAND
     1308  deepak  15  -1  29492      2292  1512  S     0.7  0.0
     0:00.42    top
     5359  root       34       19     0     0     0     S     0.7  0.0
     9431:42    kipmi0
     1        root       15   0     10352     692   580   S     0.0
     0.0   0:02.16    init
     2        root       RT  -5          0     0     0     S     0.0
     0.0   0:02.37 migration/0
     3        root       34  19          0    0    0     S     0.0
     0.0   0:00.00 ksoftirqd/
```

5. Kill any task

*Press "**k**" on the terminal when top is running. You should get a prompt as shown below in blue color*

```
     top - 18:09:31  up  115  days,   8:45,   4  users,   load  average:  0.08,
0.05,                                                                       0.01
     Tasks: 325  total,   1  running, 324  sleeping,   0  stopped,   0  zombie
     Cpu(s):              0.1%us,              0.1%sy,              0.0%ni,
99.8%id,          0.1%wa,          0.0%hi,          0.0%si,          0.0%st
     Mem:  49432728k  total,  2062036k  used,  47370692k  free,    310072k
buffers
     Swap:  2097144k  total,          0k  used,  2097144k  free,  1297596k
cached
     PID to kill:1308
     PID  USER               PR       NI       VIRT          RES          SHR
S              %CPU              %MEM              TIME+                COMMAND
     5359  root       34    19     0         0     0     S     1.3
     0.0   9431:42       kipmi0
     6460  root       24    0     179m      30m   3976  S     1.0   0.1
     79:04.77         java
     1308  deepak     15    -1    29492      2292  1512  S     0.7   0.0
     0:00.49          top
     1434  root       15    0     29492      2288  1516  R     0.7   0.0
     0:00.13          top
     top - 18:09:31  up  115  days,   8:45,   4  users,   load  average:  0.08,
0.05,                                                                       0.01
     Tasks: 325  total,   1  running, 324  sleeping,   0  stopped,   0  zombie
     Cpu(s):              0.1%us,              0.1%sy,              0.0%ni,
99.8%id,          0.1%wa,          0.0%hi,          0.0%si,          0.0%st
     Mem:  49432728k  total,  2062036k  used,  47370692k  free,    310072k
buffers
     Swap:  2097144k  total,          0k  used,  2097144k  free,  1297596k
cached
```

```
     Kill   PID   1308   with   signal   [15]:[Hit   Enter   for   default]
     PID  USER            PR      NI      VIRT        RES         SHR
S      %CPU     %MEM      TIME+        COMMAND
      5359     root      34   19     0        0    0    S    1.3
     0.0       9431:42   kipmi0
      6460     root      24    0   179m      30m  3976  S    1.0  0.1
     79:04.77       java
     1308 deepak     15   -1   29492      2292 1512  S    0.7  0.0
     0:00.49        top
```

6. View all the processes running by a user

Press "**u**" on the terminal when top is running. You should get a prompt as shown below in blue color

```
     top - 18:12:24 up 115 days,  8:48,  4 users,  load average: 0.06,
0.05,                                                                0.00
     Tasks: 328 total,   1 running, 327 sleeping,   0 stopped,   0 zombie
     Cpu(s):            0.0%us,            0.4%sy,            0.0%ni,
99.6%id,        0.0%wa,        0.0%hi,        0.0%si,        0.0%st
     Mem:  49432728k  total,  2063268k  used, 47369460k  free,   310072k
buffers
     Swap:  2097144k  total,         0k  used,  2097144k  free,  1297660k
cached
     Which     user     (blank     for     all):deepak [Hit     Enter]
     PID  USER            PR      NI      VIRT        RES         SHR
S      %CPU     %MEM      TIME+        COMMAND
      1729     root      15    0   29488      2196 1432  R    2.0
     0.0       0:00.01   top
     1    root      15    0    10352      692   580   S    0.0  0.0
     0:02.16   init
     2    root      RT    -5    0        0    0    S    0.0  0.0
     0:02.37   migration/0
     3    root      34    19    0        0    0    S    0.0  0.0
     0:00.00   ksoftirqd/0
     4    root      RT    -5    0        0    0    S    0.0  0.0
     0:00.00   watchdog/0
     top - 18:12:41 up 115 days,  8:48,  4 users,  load average: 0.04,
0.05,                                                                0.00
     Tasks: 328 total,   1 running, 327 sleeping,   0 stopped,   0 zombie
     Cpu(s):            0.0%us,            0.1%sy,            0.0%ni,
99.9%id,        0.0%wa,        0.0%hi,        0.0%si,        0.0%st
     Mem:  49432728k  total,  2062356k  used, 47370372k  free,   310072k
buffers
     Swap:  2097144k  total,         0k  used,  2097144k  free,  1297672k
cached
     PID  USER            PR      NI      VIRT        RES         SHR
S      %CPU     %MEM      TIME+        COMMAND
     1561 deepak     17    0    3984       780  468   S    0.0  0.0
     0:00.00   man
     1564 deepak     19    0    8704       964  816   S    0.0  0.0
     0:00.00        sh
```

```
1566 deepak     23   0    8704       464  316  S    0.0  0.0
0:00.00         sh
1571 deepak     16   0    8452       892  712  S    0.0  0.0
0:00.01         less
31328     deepak     15   0    110m 2348 1264 S    0.0  0.0
0:00.20         sshd
31329     deepak     16   0    27676      2564 1816 S    0.0
0.0       0:00.02   bash
31422     deepak     15   0    109m 2360 1260 S    0.0  0.0
0:00.14   sshd
31423     deepak     15   0    27548      2500 1784 S    0.0
0.0       0:00.02   bash
```

7. *Change delay between terminal refresh*

By default the top terminal is set for auto refresh after every 3 seconds but if you want you can change it as per your requirement.

*Press "**d**" when top is running. You should get a prompt as shown below in blue color.*

```
    top - 18:14:55 up 115 days,  8:50,  4 users,  load average: 0.01,
0.04,                                                           0.00
    Tasks: 328 total,   1 running, 327 sleeping,   0 stopped,   0 zombie
    Cpu(s):           0.0%us,            0.1%sy,            0.0%ni,
99.9%id,        0.0%wa,         0.0%hi,         0.0%si,         0.0%st
    Mem:  49432728k  total,  2063828k  used, 47368900k  free,    310072k
buffers
    Swap:  2097144k  total,         0k  used,  2097144k  free,  1297728k
cached
    Change      delay      from      3.0      to:2.0 [Hit       Enter]
    PID  USER                PR       NI       VIRT          RES         SHR
S       %CPU      %MEM       TIME+        COMMAND
    5359 root       34   19        0    0    0    S    0.7  0.0
    9431:58   kipmi0
    1795 root       15   0 29492 2300 1524     R    0.3  0.0
    0:00.20         top
    1    root       15   0 10352  692    580  S    0.0  0.0
    0:02.16         init
```

Verify the changes. You must see the screen buffer getting refresh much earlier or just to verify you can provide a higher value of delay and observer the refresh rate on the terminal

8. *No. of task to be displayed*

By default this option is set to unlimited that is the reason your terminal is fully covered with list of tasks when you run the top command. Any how you can list the no of tasks to be visible once you run top command.

*Press "**n**"when top is running. You should get a prompt as shown below in blue color*

```
    top - 18:18:07 up 115 days,  8:54,  4 users,  load average: 0.01,
0.03,                                                           0.00
    Tasks: 328 total,   1 running, 327 sleeping,   0 stopped,   0 zombie
    Cpu(s):           0.0%us,            0.2%sy,            0.0%ni,
99.7%id,        0.1%wa,         0.0%hi,         0.0%si,         0.0%st
    Mem:  49432728k  total,  2063348k  used, 47369380k  free,    310072k
buffers
    Swap:  2097144k  total,         0k  used,  2097144k  free,  1297804k
```

```
cached
     Maximum  tasks  =  0,  change  to  (0  is  unlimited):2  [Hit  Enter]
     PID  USER                 PR       NI        VIRT          RES          SHR
S      %CPU      %MEM       TIME+         COMMAND
     5359 root       34   19         0    0    0    S    2.3  0.0
     9432:08   kipmi0
     1795 root       15   0 29492 2304    1528 R    0.7  0.0
     0:00.65         top
     1    root       15   0 10352  692    580  S    0.0  0.0
     0:02.16         init
     2    root       RT  -5          0    0    0    S    0.0  0.0
     0:02.37   migration/0
     top - 14:48:40 up 116 days,  5:24,  3 users,  load average: 0.05,
0.04,                                                                  0.00
     Tasks: 318 total,   1 running, 317 sleeping,   0 stopped,   0 zombie
     Cpu(s):            0.0%us,             0.1%sy,               0.0%ni,
99.9%id,         0.0%wa,          0.0%hi,          0.0%si,          0.0%st
     Mem:  49432728k  total,  2051952k  used, 47380776k  free,    310176k
buffers
     Swap:  2097144k  total,          0k  used,  2097144k  free,  1293800k
cached
     PID  USER                 PR       NI        VIRT          RES          SHR
S      %CPU      %MEM       TIME+         COMMAND
     5359 root       34  19          0    0    0    S    1.0  0.0
     9502:15   kipmi0
     25009 prasadee  15   0 29492 2280    1516 R    0.3  0.0
     0:01.88         top
```

9. ***<u>View live individual CPU processor performance</u> :***

*By default top command shows you the average of all the available CPUs in the machine. In case you want to see report of all the individual CPUs press "**1**" once you are running top command and you will get to see something like below*

```
     top - 00:00:58 up 215 days, 13:36,  5 users,  load average: 4.07,
4.04,                                                                  4.00
     Tasks: 339 total,   5 running, 334 sleeping,   0 stopped,   0 zombie
     Cpu0          :          0.0%us,          0.2%sy,          0.0%ni,
99.8%id,        0.0%wa,        0.0%hi,        0.0%si,          0.0%st
     Cpu1          :          0.0%us,          0.4%sy,          0.0%ni,
99.6%id,        0.0%wa,        0.0%hi,        0.0%si,          0.0%st
     Cpu2          :          0.0%us,          0.1%sy,          0.0%ni,
99.5%id,        0.4%wa,        0.0%hi,        0.0%si,          0.0%st
     Cpu3          :          0.1%us,          0.2%sy,          0.0%ni,
99.8%id,        0.0%wa,        0.0%hi,        0.0%si,          0.0%st
     Cpu4          :          0.5%us,          0.1%sy,          0.0%ni,
99.4%id,        0.0%wa,        0.0%hi,        0.0%si,          0.0%st
     Cpu5          :          0.2%us,          0.4%sy,          0.0%ni,
99.4%id,        0.0%wa,        0.0%hi,        0.0%si,          0.0%st
     Cpu6          :          0.0%us,          0.5%sy,          0.0%ni,
99.5%id,          0.0%wa,          0.0%hi,          0.0%si,          0.0%st
     Cpu7          :          0.1%us,          1.4%sy,          0.0%ni,
98.5%id,          0.0%wa,          0.0%hi,          0.0%si,          0.0%st
     Mem:  49432728k  total,  11891852k  used,  37540876k  free,   7996596k
buffers
     Swap:  2097144k  total,          0k  used,  2097144k  free,  1850540k
cached
     PID  USER                 PR       NI       VIRT         RES          SHR
S        %CPU       %MEM        TIME+           COMMAND
     23102     root      25   0    23724     2024  1168  R    100.1
     0.0  64446:36       pvs
     23338     root      25   0    23724     2028  1168  R    100.1
     0.0  64444:22       pvs
     23959     root      25   0    23724     2016  1168  R    100.1
     0.0  135691:00      pvs
     28698     root      25   0    23724     2024  1168  R    99.3
     0.0  128828:38      pvs
      5359     root      34   19   0    0    0    S    0.8  0.0
     17610:33  kipmi0
```

10. *<u>Add a new field in top output</u> :*

By default you see limited set of output when you use the top command. But apart from those there are a other list of field which can be added to the top output. To view all the list f field which can be added follow the below steps.Run top command and then,

*Press "**f**" which will take you the list of available fields under top command.*

All the field initials stated in BLOCK letters are visible by default when you issue top command. To add a new field press the field initial as shown in the first column.

```
     * A: PID        =    Process Id                 0x00000001
     PF_ALIGNWARN
     * E: USER       =    User Name                          0x00000002
     PF_STARTING
     * H: PR         =    Priority
     0x00000004      PF_EXITING
     * I: NI         =    Nice value                           0x00000040
     PF_FORKNOEXEC
     * O: VIRT       =    Virtual Image (kb)    0x00000100
     PF_SUPERPRIV
     * Q: RES        =    Resident size (kb)              0x00000200
     PF_DUMPCORE
     * T: SHR        =    Shared  Mem  size  (kb)           0x00000400
     PF_SIGNALED
     * W: S          =    Process Status                     0x00000800
     PF_MEMALLOC
     * K: %CPU       =    CPU usage                            0x00002000
     PF_FREE_PAGES                                                  (2.5)
     * N: %MEM  =    Memory usage (RES)    0x00008000      debug       flag
(2.5)
     * M: TIME+      =    CPU Time, hundredths     0x00024000   special
threads                                                             (2.5)
```

```
b: PPID          =     Parent Process Pid               0x001D0000
special                          states                            (2.5)
c: RUSER         =     Real user name                   0x00100000
PF_USEDFPU                          (thru                          2.4)
d: UID           =     User                                         Id
f: GROUP         =     Group                                      Name
g: TTY           =     Controlling                                 Tty
 j: P            =     Last            used          cpu         (SMP)
* P: SWAP        =     Swapped                 size               (kb)
 l: TIME         =     CPU                                        Time
 r: CODE         =     Code                  size                 (kb)
 s: DATA         =     Data+Stack                 size            (kb)
 u: nFLT         =     Page                Fault                 count
 v: nDRT         =     Dirty                Pages                count
 y: WCHAN      =       Sleeping               in              Function
 z: Flags        =     Task               Flags              <sched.h>
* X: COMMAND     =     Command name/line
```

For example to add "**swap**" field press "**p**" (in small letters). As soon as you press "**p**" it should turn into block letter notifying that it has been added to top output.Once done hit enter and it will take you back to top output

You should see something like below screen

```
PID   USER                  PR        NI        VIRT          RES          SHR
S          %CPU        %MEM        TIME+            COMMAND
20472 prasadee   15    0 30748 2412 1620     R     0.8   0.0
0:00.43    27m top
22568      root        17    0   296m 5300 3536     S     0.4   0.1
3:00.30 291m eventlogd
1     root        15    0 10348  644  544     S     0.0   0.0
2:28.66    9704 init
2     root        RT    -5        0     0     0     S     0.0   0.0
0:38.59          0 migration/0
3     root        34    19        0     0     0     S     0.0   0.0
0:00.43          0 ksoftirqd/0
4     root        RT    -5        0     0     0     S     0.0   0.0
0:00.00          0 watchdog/0
5     root        RT    -5        0     0     0     S     0.0   0.0
0:28.64          0 migration/1
6     root        34    19        0     0     0     S     0.0   0.0
0:00.67          0 ksoftirqd/1
7     root        RT    -5        0     0     0     S     0.0   0.0
0:00.00          0 watchdog/1
8     root        RT    -5        0     0     0     S     0.0   0.0
0:28.00          0 migration/2
9     root        34    19        0     0     0     S     0.0   0.0
0:00.73          0 ksoftirqd/2
```