

Linux Interview Questions

Lavatech Technology

2022

User and Group Administration

1. What is a user?

Ans. In Linux user is one who uses the system.

2. How many types of users available in Linux?

Ans. There are 5 types of users available in Linux.

- System user (Admin user who control the whole system nothing but root user).
- Normal user (Created by the Super user. In RHEL - 7 the user id's from 1000 - 60000).
- 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's from 201 - 999).
- Network user (Nothing but remote user, ie., who are login to the system through network created)
- Windows Active Directory or in Linux LDAP or NIS).
- Sudo user (The normal users who are having admin or Super user privileges)

3. What is user management?

Ans. User management means managing user. ie., Creating the users, deleting the users and modifying the users.

4. What are the important points related to users?

- Ans. Users and groups are used to control access to files and resources.
- Users can login to the system by supplying username and passwords to the system.
- Every file on the system is owned by a user and associated with a group.
- Every process has an owner and group affiliation.
- Every user in the system is assigned a unique user id (uid) and group id (gid).
- User names and user id are stored in /etc/passwd file.
- User's passwords are stored in /etc/shadow file in an encrypted form.
- Users are assigned a home directory and a shell to work with the O/S.
- Users cannot read, write and execute each other's files without permission.
- Whenever a user is created a mail box is created automatically in /var/spool/mail location.
- And some user environmental files like .bash_logout, .bash_profile, .bashrc , ...etc., are also copied from /etc/skel to his/her home directory (/home/<username>).

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

Ans. <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?

Ans. 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?

- **Ans./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/skel:** Stores user's all environmental variables files and these are copied from this directory to user's home directory

8. In how many ways can we create the users?

- Ans. **useradd** - <options><user name>
- (ii) **adduser** - <options><user name>
- (iii) **newusers** <file name> (In this file we have to enter the user details same as /etc/passwd file)

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

Ans. **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 userd /home/raju -s /bin/bash raju

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

Ans.**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 userd /home/ram -s /bin/bash ram.

11. **What is the syntax of newuser command?**

- Ans. **newusers** <file name> (This command will create multiple users at a time)
- First we should a file and enter user's data as fields same as the fields of /etc/passwd file for how many users do you want to create and mention that file as an argument for newusers command.

- When we execute this command new users will be created but their environmental files like

.bash_logout, .bash_profile, .bashrc and .bash_history files will not be copied from /etc/skel. Directory. So, we have to copied manually from /etc/skel directory.

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

Ans.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?

Ans.We can check in different ways:

- **id <user name>** (It shows the user id group id and user name if that is already created)
- **grep <user name> /etc/passwd**

14. How to verify or check the integrity of the password file?

Ans.**pwck <options> /etc/passwd** or

pwck <options> /etc/shadow The options are,

- **-q** :quiet
- **r** : read only
- **s** : sort the contents by uidin /etc/passwd and /etc/shadow files

15. How to verify or check the integrity of the group file?

- `grpck <options> /etc/group` or
- `grpck <options> /etc/gshadow`
- The options are, **-r-r**:read only **-s**:sort the contents by **gidin /etc/group** and **/etc/gshadow** files.

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

Ans. `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?

Ans.`useradd -o -u 0 -g root <user name>`

18. How to recover if the user deleted by mistake?

Ans.`pwunconv` (It creates the users according `/etc/passwd` file and deletes the `/etc/shadow` file)

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

Ans..`.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.

This file is used to create the user's custom commands and to specify the umask value.

20) What is a group?

Ans.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 a primary group.

- b) **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?**

Ans. groups <user name>

22. **What is the syntax to create the group?**

Ans. **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?**

Ans. 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** <group name> (to delete the group without options)

25. **How to assign the password to the group?**

Ans.**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

26. **How to check the integrity or consistency of the group?**

Ans.**grpck** (it will check the integrity or consistency in **/etc/gpasswd** and **/etc/gshadow** files).

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

Ans. grpconv (it creates the /etc/gshadow file from /etc/group file)

28. How to change the password aging policies?

Ans. we can change the password policies in 2 ways (i.e. configuration file and Chage Command) (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. i.e., 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.

29. **How add 45 days to the current system date?**

Ans. date -d "+ 45 days"

30. **Explain the sudo user?**

Ans.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?**

Ans. 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 Save and exit this file.

Note : When we trying to save this file if any syntax errors in this file, those errors are displayed with line no's and **What you do ?** (will be displayed, here press 'e' to edit this file and modify those errors or mistakes and save this file.

- `su - raju` (to switch to raju user)
- `sudo useradd <useradd>` (The normal user raju can also add the users to the system)
- We can assign sudo permissions to 'n' no. of users by specifying names separated by commas (,) or line by line.
- Instead of giving all permissions to normal user we can give only some commands.
- **Example** student `ALL=/usr/sbin/useradd, /usr/sbin/usermod`
raju `ALL=NOPASSWD:/usr/sbin/useradd, /usr/sbin/usermod`
- We can also apply to one group or groups as follows.
- First create the users, assign one group to those users and also assign the passwords for that users.
- Open `/etc/sudoers` file by executing the command **visudo** and type as follows.
-

- We can also create one command alias and add some commands to that alias and mention that alias to users as follows.
- $\text{Cmnd}_{Alias} \text{NETWORKING} = /usr/sbin/route, /usr/sbin/ifconfig$
 $\text{username} > < \text{machines} > = < \text{command alias name} >$
 $\text{raju ALL} =$
- We can also create one user alias and add the users to that alias and assign some commands to that alias as follows.
- $\text{User}_{Alias} < \text{user alias name} > = < \text{user1} >, < \text{user2} >, < \text{user3} >$
Example : $\text{User}_{Alias} \text{OURTEAM} = \text{raju, shyam, ram, gopal}$
- $\text{OURTEAM ALL} = \text{ALL}$ (to give all permissions of sudo)
- Defaults $\text{timestamp}_{timeout} = 0$ (whenever the sudo user executes any command then it will ask for password)
 - Defaults : $< \text{user1} >, < \text{user2} >, < \text{user3} > \text{timestamp}_{timeout} = 0$ (the system will ask passwords for user1, user2, user3 to execute sudo commands)

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)

33. **How to assign the password to normal user by him whenever first login to the system?**

Ans. Whenever the user is created and that user is trying to login to the system, it will ask the password. If the root user is not assign the password to that user, then that normal user can assign the password by his own using the following commands.

- `useradd <user name>` (to create the user)
- `passwd -S <user name>` (to see the status of the password of that user. if root user is not assigned the password then the password status is locked)
- `passwd -d <user name>` (then delete the password for that user)
- `chage -d 0 <user name>` (it will change the password age policy)
- `su - <user name>` (Try to switch to that user then it will display the following message)
- Newpassword : (type new password for that user)
- Retype password : (retype the password again)

The other useful commands:

- `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**:cshell to write 'C++' language programs
- **/bin/csh** : c shell to write 'C' language programs
- **echo SHELL** :(to see the current shell)
- **chsh <user name>** (to change the user's shell)
- **Changing shell for <user name> :**

- New shell : <type new shell for example /bin/sh to change the current shell>New shell changed (But it will effect by restarting the server)
- date R :(to display the time only)
- date + x :(to display the date only)
- 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.
- echo HISTSIZE :(to check the current history size)
- export HISTSIZE=500 (to change the current history size to 500 temporarily)
- export HISTTIMEFORMAT=
- vim /etc/bashrc (open this file go to last line and type as follows to make history i size date time formats permanently)
- HISTSIZE=1000
- HISTTIMEFORMAT='
- <user name>:(to go to users home directory)
- what is <command>:(to see the short description of that command)
- where is <command>(to see the location of that command and location of the document of that command)
- reset :(to refresh the terminal)

- whoami :(to see the current user name)
- who a mi :(to see the current user with full details like login time and others)
- 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)
- lspci :(to list all the PCI slots present in the system)
- 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)

34. How many types of the files are there?

Ans. There are 7 types of files.

- a) -:regular file
- b) d : directory
- c) c : character device file (Ex. console file, open and close terminals, ...etc.,)
- d) b : block device file (Ex. device blocks like hard disks, CD/DVD disks)
- e) s : socket file (programmers will deal this file)
- f) p : pipe file (programmers will deal this file)
- g) l : linked file (nothing but short cut file)

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

Ans. 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

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

Ans. `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>` To change the permissions the syntax is, `chmod <who> <what> <which> <file name or directory>`

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

Ans. `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?

Ans.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?

Ans.

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

40. What is umask in linux?

Ans.

- The user file-creation mode mask (umask) is used to determine the file permissions for newly created files or directories. It can be used to control the default file or directory permissions for new files. It is a four-digit octal number. The umask value for normal user is **0002** and the umask value for root user is **0022**.
- So, the effected file permissions for normal users = $666 - 002 = 664$.
- The effected directory permissions for normal users = $777 - 002 = 775$.
- The effected file permissions for root user = $666 - 022 = 644$
- The effected directory permissions for root user = $777 - 022 = 755$
- `umask <value>` (to change the umask value temporarily)
- `vim /etc/bashrc` (open this file and change the umask value to effect the whole system)
- `source /etc/bashrc` (to updated the source file)
- `vim .bashrc` (open this file in user's home directory and at last type as follows)
- `umask <value>` (save and exit the file)
- `source .bashrc` or `logout` and `login` again (to the system to effect that umask value)

- If the **/etc/login.defs** file is corrupted then new users will be added and can be assigned the passwords but users cannot login.
- If the **/etc/login.defs** file is deleted then new users cannot be added

41. How change the permissions using numeric representation? Ans.

- a) The values for read = 4, write = 2, execute = 1 and null = 0. The total value = $4 + 2 + 1 = 7$
- b) `chmod <no.><no.><no.><file name or directory name>` **Example :** `chmod 7 7 4 file1` (to give read, write and execute to owner and read, write and execute to group and read permission to others)
- c) `chmod 6 6 0 file2` (to give read and write to owner and read and write to group and null (0) permission to others)

42. Explain about set uid (suid)? Ans.

- If we plan to allow all the users to execute the root users command then we go for set uid (suid).
- It can be applied for user level and is applicable for files only.
- `chmod u+s <file name>` (to set the suid on that file)
- `chmod u-s <file name>` (to remove the suid from that file)

- `ls -l` (if 'x' is replaced with 's' in owner's level permissions that means suid is applied on that file)
- `-rwxrwxrwx <file name>` (here 's' is called set uid or suid)
- **Example :** `chmod u+s /usr/sbin/init 6` (then any user can restart the system using this command `init 6`)
- `chmod u+s /sbin/fdis` (then any user can run the `fdisk` command)
- `strings <command name>` (to read the binary language of the command ie., the string command converts the binary language into human readable language)
- `strings mkfs` (to read the `mkfs` command's binary language into human readable language)
- Normally set uid (suid) permission will be given on scripting files only

43. Explain about set gid (sgid)?

Ans.If we plan to allow all the users of one group to get the group ownership permissions then we go for set gid (sgid). It can be applied for group level and is applicable on directories only. **Example:** `chmod g+s <directory name>` (to set the sgid on that directory)
`chmod g-s <directory name>` (to remove the sgid from that directory)

44. Explain about sticky bit?

Ans. 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)

45. What are the uses of passwd and shadow files?

Ans. **Passwd file:**

- a) When we create the user one entry is updated in password and shadow files.
- b) It represents and tell about that user login name , uid, gid, default home directory of the use and default shell.
- c) So, using this file we can easily get users information.

Shadow file ;

- a) This file tells about the login id, user's encrypted password, password when last changed, min. days the password valid, max. days valid, warning days, inactive days and expiry days.
- b) If shadow file is missed or deleted we can recover those entries of shadow file using password file.
- c) We can change the users encrypted passwords with the permissions of the higher authorities in case of emergency.

46. What is the use of group?

Ans.

- a) In an organization the whole work is divided into departments for easy maintenance and easy
- b) For each department is also represented as group and that group having so many users to do different works.
- c) So, if we create one group and assign that group to all the users in that department, then we can easily identify which user belongs to which group.
- d) We can share files, directories and execute some programs to that group and also give permissions to that group. So, each user of that group can easily share those directories and also can easily access, execute or even write in those shared files and directories.

47. Can we login to the user without password?

Ans. Yes, we can login.

48. How to recover the root password if missed or deleted?

Ans: RHEL - 6 :

- a) Restart the system.
- b) Select 1st option and press 'e'.
- c) Select 2nd option and press 'e'.
- d) At the end give one blank space and type 1 and press Enter key.

- e) Then press 'b' to boot the system in single user mode.
- f) Then prompt appears and type `passwd root` command.
New password : XXXXXX Retype password : XXXXXX
- g) Exit
- h) Then system starts as usual

RHEL - 7 :

- a) Restart the system.
- b) Using arrow keys select 1st line and press 'e' to edit.
- c) Go to **Linux 16** line press End key or **Ctrl + e** to go to the end of the line and give one space.
- d) Then type as **rd.break console=tty1 selinux=0**
- e) Then press Ctrl + x to start the computer in single user mode.
- f) After starting we get **switch_root : /** *prompt appears and then type as follows*
- g) chroot /sysroot** press Enter
- h) Then **sh - 4.2** prompt appears and type as
- i) **sh - 4.2 passwd root** New password : XXXXXX Retype password : XXXXXX
- j) **sh - 4.2 exit**
- k) **switch-root :/exit**
- l) Then the system starts and the desktop appears.

49. How to restrict the users from login?

Ans.

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

50. **How to put never expiry to a user?**

Ans. `passwd -x -1 <user login name>`