RHEL9

Session 7 – 19th Nov 2022 Summary

- The "**useradd**" command is used to create users, behind the scene it updates the files such as "/etc/passwd", "/etc/shadow", "/etc/group" etc
- The link of "useradd" command is "**adduser**" which can also be used to create users
- The command to find the location of "adduser" command is "**which adduser**"

```
[root@localhost ~]# which adduser
/usr/sbin/adduser
```

- The command used to list the details of "adduser" command is "**ls –l /usr/sbin/adduser**" from this we see that "adduser" command is the link of "useradd" command, two different names but both the commands are same

```
[root@localhost ~]# ls -l  /usr/sbin/adduser
lrwxrwxrwx. 1 root root 7 Dec  3  2021 /usr/sbin/adduser -> useradd
[root@localhost ~]#
```

- The command used to create a user pop123 is "**useradd pop123**", this command behind the scene updates some of the files.

```
[root@localhost ~]# useradd  pop123
[root@localhost ~]#
```

- The "**/etc/passwd**" file has the database of all users, every line is the information of a particular user, it has **seven fields**

```
[root@localhost ~]# cat /etc/passwd
```

```
flatpak:x:990:985:User for flatpak system helper:/:/sbin/nologin
gdm:x:42:42::/var/lib/gdm:/sbin/nologin
cockpit-ws:x:989:984:User for cockpit web service:/nonexisting:/sbin/nologin
cockpit-wsinstance:x:988:983:User for cockpit-ws instances:/nonexisting:/sbin/no
login
gnome-initial-setup:x:987:982::/run/gnome-initial-setup/:/sbin/nologin
rpcuser:x:29:29:RPC Service User:/var/lib/nfs:/sbin/nologin
sshd:x:74:74:Privilege-separated SSH:/usr/share/empty.sshd:/sbin/nologin
chrony:x:986:981::/var/lib/chrony:/sbin/nologin
dnsmasq:x:985:980:Dnsmasq DHCP and DNS server:/var/lib/dnsmasq:/sbin/nologin
tcpdump:x:72:72::/:/sbin/nologin
systemd-oom:x:978:978:systemd Userspace OOM Killer:/:/usr/sbin/nologin
vimal:x:0:1000:vimal daga:/home/vimal:/bin/bash
tom:x:1001:1001::/home/tom:/bin/bash
apache:x:48:48:Apache:/usr/share/httpd:/sbin/nologin
yash:x:1002:1002::/home/yash:/bin/bash
pop:x:1003:1003::/home/pop:/bin/bash
jack:x:1004:1004::/home/jack:/bin/bash
eric:x:0:1005::/home/eric:/bin/bash
pop123:x:1005:1006::/home/pop123:/bin/bash
```

- The seven fields are separated by colon (:)
    - User name or Login name
    - Password (x) – links to other file "/etc/shadow"
    - UID(User Identifier)
    - GID(Group Identifier)
    - Comment – this field is usually empty but we can add details about the user
    - Home Directory- after user login, the OS will take user to this home directory
    - /bin/bash or /sbin/nologin shell
- To give details about the user
    - Create the account

```
[root@localhost ~]# useradd  tom123
[root@localhost ~]#
```

    - Open the file

```
[root@localhost ~]# vim /etc/passwd
```

    - Go to the line and add the details of the user

```
tom123:x:1006:1007:my tom admin:/home/tom123:/bin/bash
```

- The fifth field is the **comment field** is also called as GECOS field, in this one field we can store many information about the user.
- The "**chfn command**" by which we can store the detailed information about the user (chfn → change the finger database). It updates the information of the user in the "/etc/passwd" file. The command is "**chfn user_name**".

```
[root@localhost ~]# chfn vimal
Changing finger information for vimal.
Name [vimal daga]: vimal daga lw
Office []: LW
Office Phone []: 123456
Home Phone []: 982911124

Finger information changed.
[root@localhost ~]#
```

- The command used to open the "/etc/passwd" file is "vim /etc/passwd", we see that in the fifth single field multiple information of a user can be stored this is called as **GECOS or finger** database.

```
[root@localhost ~]# vim /etc/passwd
```

```
systemd-oom:x:978:978:systemd Userspace OOM Killer:/:/usr/sbin/nologin
vimal:x:0:1000:vimal daga lw,LW,123456,982911124:/home/vimal:/bin/bash
tom:x:1001:1001::/home/tom:/bin/bash
```

- The command used to open the manual for a file "**man 5 file_name**", here the file_name is "passwd"

```
[root@localhost ~]# man 5 passwd
```

```
    name:password:UID:GID:GECOS:directory:shell

The field are as follows:

name          This is the user's login name.  It should not contain capi-
              tal letters.

password      This is either the encrypted  user  password,  an  asterisk
              (*),  or the letter 'x'.  (See pwconv(8) for an explanation
              of 'x'.)

UID           The privileged root login account (superuser) has the  user
              ID 0.

GID           This is the numeric primary group ID for this user.  (Addi-
              tional groups for the user are defined in the system  group
              file; see group(5)).
```

```
GECOS       This  field  (sometimes  called the "comment field") is op-
            tional and used only for informational purposes.   Usually,
            it contains the full username.  Some programs (for example,
            finger(1)) display information from this field.

            GECOS stands for "General Electric Comprehensive  Operating
            System",  which was renamed to GCOS when GE's large systems
            division was sold to Honeywell.   Dennis  Ritchie  has  re-
            ported:  "Sometimes we sent printer output or batch jobs to
            the GCOS machine.  The gcos field in the password file  was
            a  place  to stash the information for the $IDENTcard.  Not
            elegant."

directory   This is the user's home directory:  the  initial  directory
            where  the  user  is placed after logging in.  The value in
            this field is used to set the HOME environment variable.

shell       This  is  the  program  to  run  at  login  (if  empty,  use
            /bin/sh).   If  set  to  a nonexistent executable, the user
            will be unable to login through  login(1).   The  value  in
            this field is used to set the SHELL environment variable.
```

- The "**finger command**" has the capability to convert the information of user into simple user readable format.
  - Create user

```
[root@ip-172-31-40-219 ~]# useradd  vimal
```

  - Print the file

```
[root@ip-172-31-40-219 ~]# cat /etc/passwd
```

```
vimal:x:1001:1001::/home/vimal:/bin/bash
```

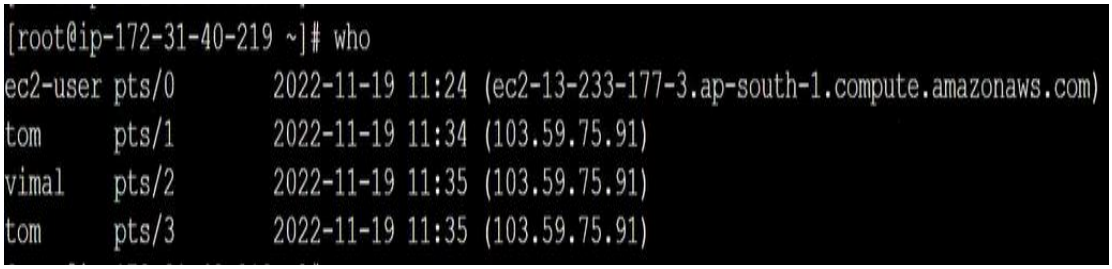
  - Use finger command to retrieve a particular user information

```
[root@ip-172-31-40-219 ~]# finger vimal
Login: vimal                            Name:
Directory: /home/vimal                  Shell: /bin/bash
Never logged in.
No mail.
No Plan.
```

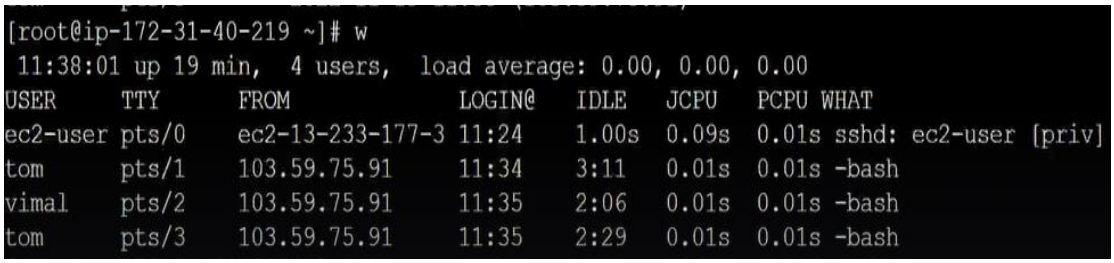
- The "**finger**" command is also used to keep track about all the users who are currently logged in

```
[root@ip-172-31-40-219 ~]# finger
Login    Name            Tty      Idle  Login Time   Office     Office Phone  Host
ec2-user EC2 Default User pts/0          Nov 19 11:24                        (ec2-13-233-177-3.ap-south-1.comput
e.amazonaws.com)
tom                      pts/1      1  Nov 19 11:34                        (103.59.75.91)
tom                      pts/3      1  Nov 19 11:35                        (103.59.75.91)
vimal    vimal daga      pts/2          Nov 19 11:35 LW    x1-2345          (103.59.75.91)
```

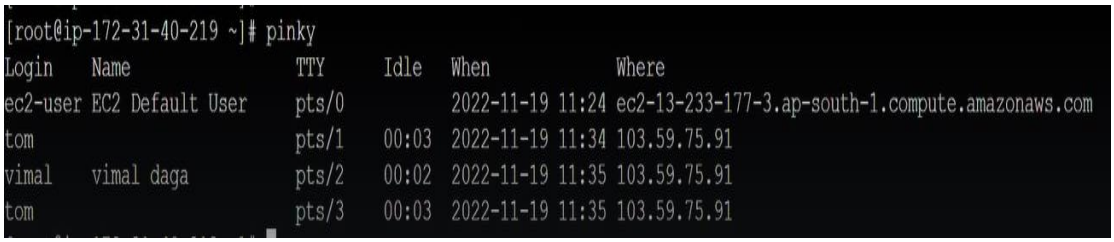
- The "**who command**" is also used to see about all user currently logged in



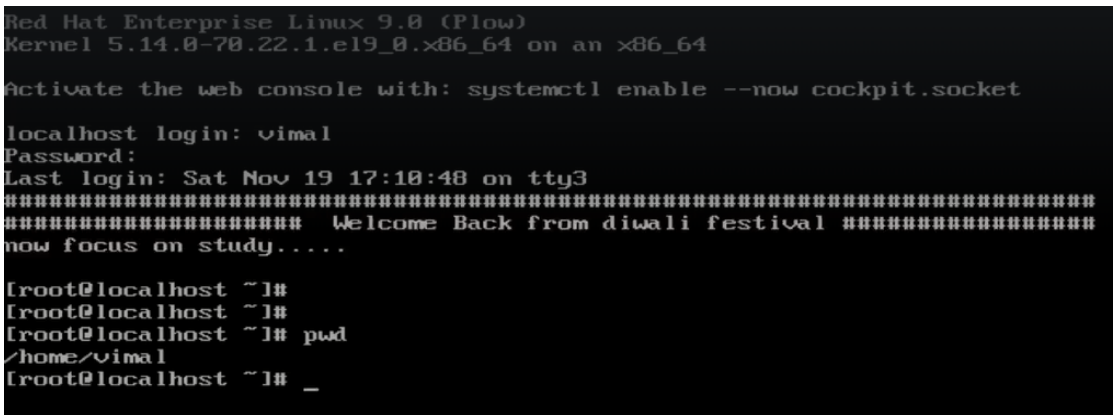
- The "w command" gives extra information about the system and all users currently logged in



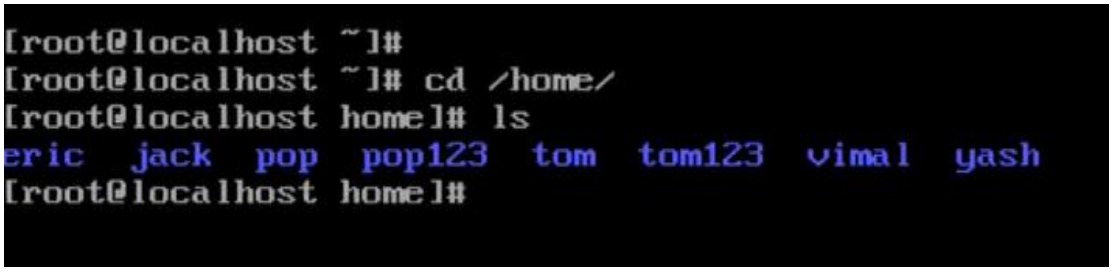
- The "**pinky command**" is also used to see about all user currently logged in



- The sixth field is the "**home directory**" , when a user login to OS, the user will be in some directory or folder
- After user login, the first directory, where the user lands is called the **home directory**

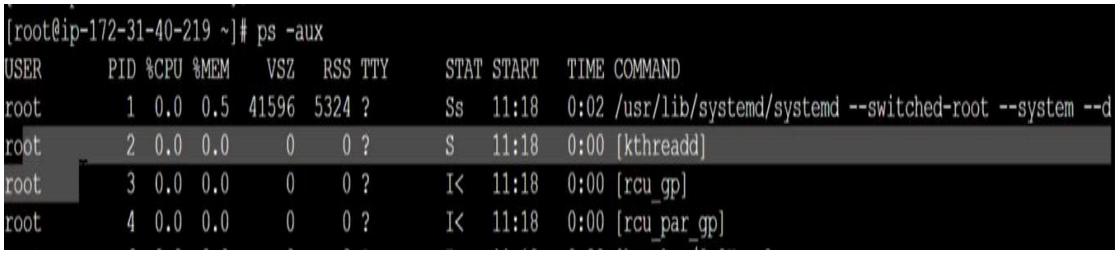


- We have multiple users, each user has personal data to manage, to store user data separately there is home directory
- The "**/home"** is a pre-created directory, where we see all users home directory



- When we create a user krish using command "**useradd krish**", the useradd command automatically creates the directory krish for the user, this is the home directory for krish. It also sets a security.

- The seventh field is **"/bin/bash or /sbin/nologin**", when a user login in CLI, we have a command prompt, there is a cursor blinking, where a command can be entered. A program behind the scene takes the command and runs it and gives the output. This program is called as **Shell Program.** There are different types of shell bash, csh, sh etc. The very famous shell is the "**bash**" shell. A user is able to run a program because bash shell is given to the user. These users are called **interactive users**.
- There is another shell "**nologin**" shell – where in some situations there are users, who do not interact with OS, such users are called **non-interactive users**.
- When we run a program it becomes a process, every process is run by a user that is every process is associated by a user.



- In server program, when we start Apache Webserver, an httpd process starts, the user "**apache**" starts this process. The purpose of this user is to start the httpd process. These users are called as "**system or service users**".

```
[root@localhost ~]# ps  -aux   |  grep httpd
root         928  0.0  0.1  20048 11436 ?        Ss   16:24   0:00 /usr/sbin/httpd -DFOREGROUND
apache       959  0.0  0.0  21516  7304 ?        S    16:24   0:00 /usr/sbin/httpd -DFOREGROUND
apache       960  0.0  0.1 1669160 12828 ?       Sl   16:24   0:01 /usr/sbin/httpd -DFOREGROUND
apache       961  0.0  0.1 1538024 14868 ?       Sl   16:24   0:00 /usr/sbin/httpd -DFOREGROUND
apache       962  0.0  0.1 1538024 14868 ?       Sl   16:24   0:00 /usr/sbin/httpd -DFOREGROUND
root        7074  0.0  0.0 221800  2268 pts/0    S+   17:46   0:00 grep --color=auto httpd
[root@localhost ~]#
```

- For such users a non-interactive shell called "**nologin shell"** has been given