Practical Number: 2

CO/PO: CO1, PO1, PO2

Problem Definition

As a user administration tasks in a Linux-based operating system, including the creation, modification, and deletion of user accounts using tools like useradd, usermod, and userdel. This involves managing user and group IDs (UIDs/GIDs), configuring home directories, setting shell environments, and enforcing password policies. Additionally, it includes comprehensive group management through commands like groupadd, gpasswd, and groupdel to organize users based on roles and access levels. The task also covers the assignment and management of file and directory permissions using chmod, chown, and chgrp to enforce access control through permission bits (read, write, execute) and advanced permission schemes like SUID, SGID, and sticky bits. Regular monitoring of user activities using tools such as who, w, last, and log files (/var/log/auth.log, /var/log/secure) is also emphasized to ensure system security, policy compliance, and efficient resource utilization.

Key Questions to be evaluated during Implementation

Question(s)
How to delete a user with the home directory?
Ans: sudo userdel -r username
How to change a user's shell or home directory?
Ans: sudo usermod -s /bin/bash username /sudo usermod -d /new/home/path username
How to account expiry dates for users, and why is this important in enterprise environments?
Ans: sudo chage -E 2025-12-31 username Importance in enterprise environments: ○ Prevents unauthorized access after employees or contractors leave. ○ Enforces security compliance and audit requirements. ○ Helps reduce the risk of dormant accounts being exploited by attackers.
Describe the impact of SGID on directories. How does it affect file creation within a group-shared directory?
Ans: SGID (Set Group ID) on a directory ensures that new files or subdirectories created inside inherit the group ownership of the directory, not the primary group of the user creating the file. This is useful for collaborative environments where multiple users share a directory.
How to configure for monitoring specific file access or privilege escalation attempts?

Ans:
1. **Install & start auditd:**
`sudo apt install auditd -y`
`sudo systemctl enable --now auditd`

2. **Monitor specific file:**

```
sudo auditctl -w /path/to/file -p war -k file_watch
```

3. **Monitor privilege escalation (sudo/setuid):**

```
sudo auditctl -a always,exit -F arch=b64 -S execve -F
euid=0 -k priv_exec
```

4. **View logs:**

```
sudo ausearch -k file_watch
sudo ausearch -k priv_exec
```

Key Skills to be addressed

- Linux terminal usage
- Command-line user and group administration
- File system permission management
- Troubleshooting access issues

Applications

- Used in system administration
- Crucial for managing access in multi-user environments
- Foundational skill in DevOps and Cybersecurity

Learning Outcome

Student will be able to

- Understand the user management lifecycle in Linux
- Demonstrate the working of Linux administrative commands
- Learn how to enforce access control using permissions

Tools/Technology to Be Used

- Ubuntu/Linux Terminal
- Bash Shell
- Text Editors (nano/vim)

- **Total Hours of Problem Definition Implementation**

3 Hours

- **Total Hours of Engagement**

4 Hours
(*Includes implementation + modification + faculty testing*)

Post Laboratory Work Description

Part A:

1. Run id command to view the current user and group information.
2. display the current working directory.
3. print the value of HOME and PATH variable to determine the home directory and user's executable's path respectively.
4. Run su and su - command. Observe the output for the same.what is the main difference between them?
5. Run sudo su at the shell prompt to become the root user.
6. Run id command to view the current user and group information.
7. display the current working directory.
8. print the value of HOME and PATH variable to determine the home directory and user's executable's path respectively.
9. Exit the current user's shell to return to the student user's shell
10. Attempt to view the last five lines of /var/log/auth.log without using sudo
11. Attempt to view the last five lines of /var/log/auth.log using sudo
12. Attempt to make a copy of /etc/rpc as /etc/rpcOLD without using sudo
13. Attempt to make a copy of /etc/rpc as /etc/rpcOLD with sudo.
14. Attempt to delete /etc/rpcOLD without using sudo
15. Attempt to delete /etc/rpcdOLD with sudo

16. check the UID for root user, administrator and local users.
17. Adduser user01.
18. Create the group group01 with the GID of 10000.19. Create the group group02
20. Examine /etc/group to verify the supplemental group memberships.
21. Use the usermod -aG command to add a user to a supplementary group. Add user01 to the group created.
22. Observe /etc/group and /etc/passwd

ScreenShots:

```
[yug@Yug ~]$ id
uid=1000(yug) gid=1000(yug) groups=1000(yug) context=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
[yug@Yug ~]$ pwd
/home/yug
[yug@Yug ~]$ echo $HOME
/home/yug
[yug@Yug ~]$ echo $PATH
/home/yug/.local/bin:/home/yug/bin:/usr/local/bin:/usr/local/sbin:/usr/bin:/usr/sbin
[yug@Yug ~]$ su
Password:
[root@Yug yug]# exit
exit
[yug@Yug ~]$ su -
Password:
Last login: Thu Jul 24 20:56:58 IST 2025 on pts/0
[root@Yug ~]# exit
logout
```

```
[root@Yug ~]# sudo su
[root@Yug ~]# id
uid=0(root) gid=0(root) groups=0(root) context=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
[root@Yug ~]# pwd
/root
[root@Yug ~]# echo $HOME
/root
[root@Yug ~]# echo $PATH
/root/.local/bin:/root/bin:/sbin:/bin:/usr/sbin:/usr/bin
[root@Yug ~]# exit
exit
```

```
[yug@Yug ~]$ tail -n 5 /var/log/secure
tail: cannot open '/var/log/secure' for reading: Permission denied
[yug@Yug ~]$ sudo tail -n 5 /var/log/secure
Jul 24 21:33:39 Yug su[6687]: pam_unix(su-l:session): session opened for user root(uid=0) by yug(uid=1000)
Jul 24 21:39:12 Yug su[6687]: pam_unix(su-l:session): session closed for user root
Jul 24 21:39:37 Yug sudo[6775]:     yug : TTY=pts/0 ; PWD=/home/yug ; USER=root ; COMMAND=/bin/whoami
Jul 24 21:39:37 Yug sudo[6775]: pam_unix(sudo:session): session opened for user root(uid=0) by yug(uid=1000)
Jul 24 21:39:37 Yug sudo[6775]: pam_unix(sudo:session): session closed for user root
[yug@Yug ~]$
```

```
[yug@Yug etc]$ cd ..
[yug@Yug /]$ cp /etc/rpc /etc/rpcOLD
cp: cannot create regular file '/etc/rpcOLD': Permission denied
[yug@Yug /]$ sudo cp /etc/rpc /etc/rpcOLD
[yug@Yug /]$ rm /etc/rpcOLD
rm: remove write-protected regular file '/etc/rpcOLD'?
[yug@Yug /]$ sudo rm /etc/rpcOLD
[yug@Yug /]$
```

```
[yug@Yug /]$ grep -E 'root|user' /etc/passwd
root:x:0:0:root:/root:/bin/bash
operator:x:11:0:operator:/root:/sbin/nologin
clevis:x:992:992:Clevis Decryption Framework unprivileged user:/var/cache/clevis:/usr/sbin/nologin
chrony:x:987:986:chrony system user:/var/lib/chrony:/sbin/nologin
```

```
[yug@Yug /]$ sudo useradd user1
[yug@Yug /]$ sudo passwd user1
Changing password for user user1.
New password:
BAD PASSWORD: The password is shorter than 8 characters
Retype new password:
passwd: all authentication tokens updated successfully.
[yug@Yug /]$ sudo groupadd -g 10000 group1
[yug@Yug /]$ sudo groupadd group2
```

```
user1:x:1001:
group1:x:10000:
group2:x:10001:
[yug@Yug /]$
```

```
[yug@Yug /]$ sudo usermod -aG group1 user1
[yug@Yug /]$ cat /etc/group
```

```
user1:x:1001:
group1:x:10000:user1
group2:x:10001:
[yug@Yug /]$
```

```
[yug@Yug /]$ cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
sync:x:5:0:sync:/sbin:/bin/sync
shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
halt:x:7:0:halt:/sbin:/sbin/halt
mail:x:8:12:mail:/var/spool/mail:/sbin/nologin
operator:x:11:0:operator:/root:/sbin/nologin
games:x:12:100:games:/usr/games:/sbin/nologin
ftp:x:14:50:FTP User:/var/ftp:/sbin/nologin
nobody:x:65534:65534:Kernel Overflow User:/:/sbin/nologin
systemd-coredump:x:999:999:systemd Core Dumper:/:/sbin/nologin
dbus:x:81:81:System message bus:/:/sbin/nologin
polkitd:x:998:998:User for polkitd:/:/sbin/nologin
avahi:x:70:70:Avahi mDNS/DNS-SD Stack:/var/run/avahi-daemon:/sbin/nologin
rtkit:x:172:172:RealtimeKit:/:/sbin/nologin
libstoragemgmt:x:996:996:daemon account for libstoragemgmt:/:/usr/sbin/nologin
geoclue:x:995:995:User for geoclue:/var/lib/geoclue:/sbin/nologin
tss:x:59:59:Account used for TPM access:/:/usr/sbin/nologin
colord:x:994:994:User for colord:/var/lib/colord:/sbin/nologin
sssd:x:993:993:User for sssd:/:/sbin/nologin
clevis:x:992:992:Clevis Decryption Framework unprivileged user:/var/cache/clevis:/usr/sbin/nologin
setroubleshoot:x:991:991:SELinux troubleshoot server:/var/lib/setroubleshoot:/usr/sbin/nologin
pipewire:x:990:990:PipeWire System Daemon:/run/pipewire:/usr/sbin/nologin
flatpak:x:989:989:Flatpak system helper:/:/usr/sbin/nologin
gdm:x:42:42:GNOME Display Manager:/var/lib/gdm:/usr/sbin/nologin
gnome-initial-setup:x:988:987::/run/gnome-initial-setup/:/sbin/nologin
sshd:x:74:74:Privilege-separated SSH:/usr/share/empty.sshd:/usr/sbin/nologin
chrony:x:987:986:chrony system user:/var/lib/chrony:/sbin/nologin
dnsmasq:x:986:985:Dnsmasq DHCP and DNS server:/var/lib/dnsmasq:/usr/sbin/nologin
tcpdump:x:72:72::/:/sbin/nologin
yug:x:1000:1000:Yug:/home/yug:/bin/bash
user1:x:1001:1001::/home/user1:/bin/bash
[yug@Yug /]$
```

Part B:

1. Check the permission of files created.
2. Check the permission of directories created.
3. Set read and write permissions for others with numeric mode to file1.txt
4. Remove write permission for user, group and others to folder CE.
5. Create a directory 5CE under CE. Observe the response.
6. Set read, write and execute permissions for user, group and others to 5CE.
7. Set read and execute permission for group and no permission for other to file2.txt.
8. Change the ownership of file to user01
9. Change the group ownership of file to group01
10. Change the ownership of both group and user at the same time.
11. Set the special permissions on directory.
    a. The *setuid* permission on an executable file means that commands run as the user owning the file, not as the user that ran the command. One example is the passwd command:run ls -l /usr/bin/passwd
    b. The special permission *setgid* on a directory means that files created in the directory inherit their group ownership from the directory, rather than inheriting it from the creating user. run ls -ld /run/log/journal
    c. the *sticky bit* for a directory sets a special restriction on deletion of files. Only the owner of the file (and root) can delete files within the directory. run ls -ld /tmp
12. Set the setusid, setgid and sticky bit for different files and perform the operations accordingly.
13. Display the current value of shell's mask.
14. Check the permission of directories.
15. Check the permission of files.
16. Set the umask to 542.
17. Check the permission of files and directories.
18. Try to open the file and directory created.
19. Try to open the file as other user.

```
[yug@Yug /]$ ls -l
total 28
dr-xr-xr-x.   2 root root    6 Jun 25  2024 afs
lrwxrwxrwx.   1 root root    7 Jun 25  2024 bin -> usr/bin
dr-xr-xr-x.   7 root root 4096 Jul 24 20:51 boot
drwxr-xr-x.   2 root root    6 Jul 24 22:09 CE
drwxr-xr-x.  18 root root 3240 Jul 24 20:50 dev
drwxr-xr-x. 129 root root 8192 Jul 24 21:56 etc
-rw-r--r--.   1 root root    0 Jul 24 22:08 file1.txt
drwxr-xr-x.   4 root root   30 Jul 24 21:53 home
lrwxrwxrwx.   1 root root    7 Jun 25  2024 lib -> usr/lib
lrwxrwxrwx.   1 root root    9 Jun 25  2024 lib64 -> usr/lib64
drwxr-xr-x.   2 root root    6 Jun 25  2024 media
drwxr-xr-x.   2 root root    6 Jun 25  2024 mnt
drwxr-xr-x.   2 root root    6 Jun 25  2024 opt
dr-xr-xr-x. 259 root root    0 Jul 24 20:50 proc
dr-xr-x---.   4 root root 4096 Jul 24 21:39 root
drwxr-xr-x.  45 root root 1160 Jul 24 20:54 run
lrwxrwxrwx.   1 root root    8 Jun 25  2024 sbin -> usr/sbin
drwxr-xr-x.   2 root root    6 Jun 25  2024 srv
dr-xr-xr-x.  12 root root    0 Jul 24 20:50 sys
drwxrwxrwt.  17 root root 4096 Jul 24 21:34 tmp
drwxr-xr-x.  12 root root  144 Jul 23 12:21 usr
drwxr-xr-x.  20 root root 4096 Jul 23 12:26 var
[yug@Yug /]$ ls -ld
dr-xr-xr-x. 19 root root 262 Jul 24 22:09 .
```

```
[yug@Yug /]$ sudo chmod o+rw file1.txt
[yug@Yug /]$ sudo chmod a-w CE
[yug@Yug /]$ sudo mkdir CE/5CE
[yug@Yug /]$ sudo chmod 777 CE/5CE
[yug@Yug /]$
```

```
[yug@Yug /]$ sudo chmod o+rw file1.txt
[yug@Yug /]$ sudo chmod a-w CE
[yug@Yug /]$ sudo mkdir CE/5CE
[yug@Yug /]$ sudo chmod 777 CE/5CE
[yug@Yug /]$ sudo touch file2.txt
[yug@Yug /]$ sudo chmod g+rw,o-rwx file2.txt
[yug@Yug /]$ sudo chown user1 file1.txt
[yug@Yug /]$ sudo chgrp group1 file1.txt
[yug@Yug /]$ sudo chown user1:group1 file1.txt
[yug@Yug /]$ ls -l /usr/bin/passwd
-rwsr-xr-x. 1 root root 69296 Aug 10  2021 /usr/bin/passwd
[yug@Yug /]$ ls -ld /run/log/journal
drwxr-sr-x+ 3 root systemd-journal 60 Jul 24 20:50 /run/log/journal
[yug@Yug /]$ ls -ld /tmp
drwxrwxrwt. 17 root root 4096 Jul 24 21:34 /tmp
[yug@Yug /]$
```

```
[yug@Yug /]$ sudo chmod u+s myfile
[yug@Yug /]$ sudo chmod g+s mydir
[yug@Yug /]$ sudo chmod +t mydir
[yug@Yug /]$ umask
0022
[yug@Yug /]$ ls -ld *
dr-xr-xr-x.   2 root  root      6 Jun 25  2024 afs
lrwxrwxrwx.   1 root  root      7 Jun 25  2024 bin -> usr/bin
dr-xr-xr-x.   7 root  root   4096 Jul 24 20:51 boot
dr-xr-xr-x.   3 root  root     17 Jul 24 22:12 CE
drwxr-xr-x.  18 root  root   3240 Jul 24 20:50 dev
drwxr-xr-x. 129 root  root   8192 Jul 24 21:56 etc
-rw-r--rw-.   1 user1 group1    0 Jul 24 22:08 file1.txt
-rw-rw----.   1 root  root      0 Jul 24 22:14 file2.txt
drwxr-xr-x.   4 root  root     30 Jul 24 21:53 home
lrwxrwxrwx.   1 root  root      7 Jun 25  2024 lib -> usr/lib
lrwxrwxrwx.   1 root  root      9 Jun 25  2024 lib64 -> usr/lib64
drwxr-xr-x.   2 root  root      6 Jun 25  2024 media
drwxr-xr-x.   2 root  root      6 Jun 25  2024 mnt
drwxr-sr-t.   2 root  root      6 Jul 24 22:22 mydir
-rwSr--r--.   1 root  root      0 Jul 24 22:22 myfile
drwxr-xr-x.   2 root  root      6 Jun 25  2024 opt
dr-xr-xr-x. 259 root  root      0 Jul 24 20:50 proc
dr-xr-x---.   4 root  root   4096 Jul 24 21:39 root
drwxr-xr-x.  45 root  root   1160 Jul 24 20:54 run
lrwxrwxrwx.   1 root  root      8 Jun 25  2024 sbin -> usr/sbin
drwxr-xr-x.   2 root  root      6 Jun 25  2024 srv
dr-xr-xr-x.  12 root  root      0 Jul 24 20:50 sys
drwxrwxrwt.  17 root  root   4096 Jul 24 21:34 tmp
drwxr-xr-x.  12 root  root    144 Jul 23 12:21 usr
drwxr-xr-x.  20 root  root   4096 Jul 23 12:26 var
```

```
[yug@Yug /]$ ls -l
total 28
dr-xr-xr-x.   2 root  root      6 Jun 25  2024 afs
lrwxrwxrwx.   1 root  root      7 Jun 25  2024 bin -> usr/bin
dr-xr-xr-x.   7 root  root   4096 Jul 24 20:51 boot
dr-xr-xr-x.   3 root  root     17 Jul 24 22:12 CE
drwxr-xr-x.  18 root  root   3240 Jul 24 20:50 dev
drwxr-xr-x. 129 root  root   8192 Jul 24 21:56 etc
-rw-r--rw-.   1 user1 group1    0 Jul 24 22:08 file1.txt
-rw-rw----.   1 root  root      0 Jul 24 22:14 file2.txt
drwxr-xr-x.   4 root  root     30 Jul 24 21:53 home
lrwxrwxrwx.   1 root  root      7 Jun 25  2024 lib -> usr/lib
lrwxrwxrwx.   1 root  root      9 Jun 25  2024 lib64 -> usr/lib64
drwxr-xr-x.   2 root  root      6 Jun 25  2024 media
drwxr-xr-x.   2 root  root      6 Jun 25  2024 mnt
drwxr-sr-t.   2 root  root      6 Jul 24 22:22 mydir
-rwSr--r--.   1 root  root      0 Jul 24 22:22 myfile
drwxr-xr-x.   2 root  root      6 Jun 25  2024 opt
dr-xr-xr-x. 259 root  root      0 Jul 24 20:50 proc
dr-xr-x---.   4 root  root   4096 Jul 24 21:39 root
drwxr-xr-x.  45 root  root   1160 Jul 24 20:54 run
lrwxrwxrwx.   1 root  root      8 Jun 25  2024 sbin -> usr/sbin
drwxr-xr-x.   2 root  root      6 Jun 25  2024 srv
dr-xr-xr-x.  12 root  root      0 Jul 24 20:50 sys
drwxrwxrwt.  17 root  root   4096 Jul 24 21:34 tmp
drwxr-xr-x.  12 root  root    144 Jul 23 12:21 usr
drwxr-xr-x.  20 root  root   4096 Jul 23 12:26 var
[yug@Yug /]$
```

```
[yug@Yug /]$ umask 542
[yug@Yug /]$ sudo touch testfile
[yug@Yug /]$ sudo mkdir testdir
[yug@Yug /]$ ls -l
total 28
dr-xr-xr-x.   2 root  root      6 Jun 25  2024 afs
lrwxrwxrwx.   1 root  root      7 Jun 25  2024 bin -> usr/bin
dr-xr-xr-x.   7 root  root   4096 Jul 24 20:51 boot
dr-xr-xr-x.   3 root  root     17 Jul 24 22:12 CE
drwxr-xr-x.  18 root  root   3240 Jul 24 20:50 dev
drwxr-xr-x. 129 root  root   8192 Jul 24 21:56 etc
-rw-r--rw-.   1 user1 group1    0 Jul 24 22:08 file1.txt
-rw-rw----.   1 root  root      0 Jul 24 22:14 file2.txt
drwxr-xr-x.   4 root  root     30 Jul 24 21:53 home
lrwxrwxrwx.   1 root  root      7 Jun 25  2024 lib -> usr/lib
lrwxrwxrwx.   1 root  root      9 Jun 25  2024 lib64 -> usr/lib64
drwxr-xr-x.   2 root  root      6 Jun 25  2024 media
drwxr-xr-x.   2 root  root      6 Jun 25  2024 mnt
drwxr-sr-t.   2 root  root      6 Jul 24 22:22 mydir
-rwSr--r--.   1 root  root      0 Jul 24 22:22 myfile
drwxr-xr-x.   2 root  root      6 Jun 25  2024 opt
dr-xr-xr-x. 259 root  root      0 Jul 24 20:50 proc
dr-xr-x---.   4 root  root   4096 Jul 24 21:39 root
drwxr-xr-x.  45 root  root   1160 Jul 24 20:54 run
lrwxrwxrwx.   1 root  root      8 Jun 25  2024 sbin -> usr/sbin
drwxr-xr-x.   2 root  root      6 Jun 25  2024 srv
dr-xr-xr-x.  12 root  root      0 Jul 24 20:50 sys
d-w---xr-x.   2 root  root      6 Jul 24 22:26 testdir
--w----r--.   1 root  root      0 Jul 24 22:26 testfile
drwxrwxrwt.  17 root  root   4096 Jul 24 21:34 tmp
drwxr-xr-x.  12 root  root    144 Jul 23 12:21 usr
drwxr-xr-x.  20 root  root   4096 Jul 23 12:26 var
[yug@Yug /]$ ls -ld testdir
d-w---xr-x. 2 root root 6 Jul 24 22:26 testdir
[yug@Yug /]$ cat testfile
[yug@Yug /]$ cd testdir/
[yug@Yug testdir]$ cd ..
[yug@Yug /]$ sudo -u user1 cat testfile
[yug@Yug /]$
```