

Shabeeb Asghar

22i-2574

Section-C

Lab#5

Task#1:

sudo useradd -e 2025-03-22 Temp1

sudo chage -l Temp1

sudo nano /etc/security/limits.conf

sudo nano /etc/security/time.conf

```
sudo useradd -e 2025-03-22 Temp1
sudo chage -l Temp1
          : Feb 20, 2025
          : never
          : never
          : Mar 22, 2025
change    : 0
change    : 99999
rd expires : 7
sudo nano /etc/security/limits.conf
sudo nano /etc/security/limits.conf
sudo nano /etc/security/time.conf
█
```

```
GNU nano 7.2 /etc/security/limits.conf *
# - msgqueue - max memory used by POSIX message queues (bytes)
# - nice - max nice priority allowed to raise to values: [-20, 19]
# - rtprio - max realtime priority
# - chroot - change root to directory (Debian-specific)
#
#<domain>      <type>  <item>      <value>
#
#*              soft   core        0
#root           hard   core        100000
#*              hard   rss         10000
#@student       hard   nproc       20
#@faculty       soft   nproc       20
#@faculty       hard   nproc       50
#ftp            hard   nproc       0
#ftp            -       chroot       /ftp
#@student       -       maxlogins    4
#@Temp1         hard   maxlogins    1
# End of file

^G Help      ^O Write Out  ^W Where Is   ^K Cut        ^T Execute    ^C Location
^X Exit      ^R Read File  ^\ Replace    ^U Paste      ^J Justify    ^_ Go To Line
```

```
GNU nano 7.2 /etc/security/time.conf *
# for a rule to be active, ALL of service+ttys+users must be satisfied
# by the applying process.
#
#
# Here is a simple example: running blank on tty* (any ttyXXX device),
# the users 'you' and 'me' are denied service all of the time
#
#blank;tty* & !ttyp*;you|me;!A10000-2400
#
# Another silly example, user 'root' is denied xsh access
# from pseudo terminals at the weekend and on Mondays.
#xsh;ttyp*;root;!WdMo0000-2400
#login;*;Temp1;!A12200-0600
#
# End of example file.
#

^G Help      ^O Write Out  ^W Where Is   ^K Cut        ^T Execute    ^C Location
^X Exit      ^R Read File  ^\ Replace    ^U Paste      ^J Justify    ^_ Go To Line
```

Task#2

```
sudo groupadd projectX
sudo useradd -e 2025-03-22 dev1
sudo useradd -e 2025-03-22 dev2
sudo usermod -aG projectX dev1
sudo usermod -aG projectX dev2
sudo mkdir projectX
sudo chmod g+s projectX
ls -ld projectX
```

sudo chgrp projectX projectX

```
sudo groupadd projectX
sudo usermod -aG projectX dev1

sudo useradd -e 2025-03-22 dev1
sudo usermod -aG projectX dev2

sudo useradd -e 2025-03-22 dev2
sudo usermod -aG projectX dev1
sudo usermod -aG projectX dev2
sudo mkdir projectX
sudo chmod g+s projectX
ls -ld projectX
0 projectX
^C
sudo chgrp projectX /projectX
ch file or directory
sudo chgrp projectX projectX
```

moves the disk usage report to /var/log/disk_usage.log

: Every Sunday at 3 AM

Task#3

crontab -e

```
*/30 * * * * who > /var/log/user_activity.log
```

```
#0 0 * * * df -h > /var/log/disk_usage.log
```

```
#0 3 * * 0 rm -rf /tmp/old_logs/*
```

```
GNU nano 7.2 /tmp/crontab.2hAj9r/crontab
# and day of week (dow) or use '*' in these fields (for 'any').
#
# Notice that tasks will be started based on the cron's system
# daemon's notion of time and timezones.
#
# Output of the crontab jobs (including errors) is sent through
# email to the user the crontab file belongs to (unless redirected).
#
# For example, you can run a backup of all your user accounts
# at 5 a.m every week with:
# 0 5 * * 1 tar -zcf /var/backups/home.tgz /home/
#
# For more information see the manual pages of crontab(5) and cron(8)
#
# m h dom mon dow   command
*/30 * * * * who > /var/log/user_activity.log
#0 0 * * * df -h > /var/log/disk_usage.log
#0 3 * * 0 rm -rf /tmp/old_logs/*

^G Help      ^O Write Out  ^W Where Is   ^K Cut        ^T Execute    ^C Location
^X Exit      ^R Read File  ^\ Replace    ^U Paste      ^J Justify    ^_ Go To Line
```

```
# and what command to run for the task
#
# To define the time you can provide concrete values for
# minute (m), hour (h), day of month (dom), month (mon),
# and day of week (dow) or use '*' in these fields (for 'any').
#
# Notice that tasks will be started based on the cron's system
# daemon's notion of time and timezones.
#
# Output of the crontab jobs (including errors) is sent through
# email to the user the crontab file belongs to (unless redirected).
#
# For example, you can run a backup of all your user accounts
# at 5 a.m every week with:
# 0 5 * * 1 tar -zcf /var/backups/home.tgz /home/
#
# For more information see the manual pages of crontab(5) and cron(8)
#
# m h dom mon dow   command
*/30 * * * * who > /var/log/user_activity.log
#0 0 * * * df -h > /var/log/disk_usage.log
#0 3 * * 0 rm -rf /tmp/old_logs/*
```