

# SRE Intern Assignment – Week 1

## Ubuntu on Multipass

Siddharth Kumar

Friday 25<sup>th</sup> July, 2025

## Contents

<b>1 Advanced System Management</b>	<b>3</b>
1.1 User, Group, and Sudo Management . . . . .	3
1.2 File System Permissions and ACLs . . . . .	4
1.3 Service Management (systemd Timers & Unit Files) . . . . .	5
1.4 Network Configuration (Static IP & DNS) . . . . .	6
<b>2 Logical Volume Management (LVM)</b>	<b>8</b>
2.1 Setting up LVM . . . . .	8
2.2 Extending an LV . . . . .	9
<b>3 Advanced Troubleshooting &amp; Monitoring</b>	<b>11</b>
3.1 Process Tracing with strace . . . . .	11
3.2 Open Files with lsof . . . . .	11
3.3 Journalctl Filtering . . . . .	13
<b>4 Security Enhancements</b>	<b>14</b>
4.1 Apparmor Basics . . . . .	14
4.2 File Attributes with chattr . . . . .	15
<b>5 Advanced Automation &amp; Scripting</b>	<b>16</b>
5.1 disk_monitor.sh Script . . . . .	16
5.2 user_report.sh Script . . . . .	18

## Introduction

This document presents solutions and explanations for the Week 1 SRE Intern Assignment, executed on a Multipass VM simulating a UBUNTU environment. Commands, outputs, and screenshots are provided as evidence.

# 1 Advanced System Management

## 1.1 User, Group, and Sudo Management

### Command

```
#1 useradd -m -s /bin/bash sre_admin  
#2 visudo  
Added following line in sudoers file  
#3 sre_admin ALL=(ALL:ALL) NOPASSWD:ALL
```

### Explanation

#### #1 Command:

useradd -> add user  
-m -> with a home directory  
-s /bin/bash -> shell to use

#### #2 Command:

visudo -> opens sudoers file

#### #3 Command:

NOPASSWD:ALL -> No password prompted to run sudo

### Screenshot

```
root@sre-vm:/# useradd -m -s /bin/bash sre_admin  
root@sre-vm:/# passwd sre_admin  
New password:  
Retype new password:  
passwd: password updated successfully  
root@sre-vm:/# visudo  
root@sre-vm:/# su - sre_admin  
sre_admin@sre-vm:~$ sudo apt update  
Hit:1 http://ports.ubuntu.com/ubuntu-ports jammy InRelease  
Hit:2 http://ports.ubuntu.com/ubuntu-ports jammy-updates InRelease  
Hit:3 http://ports.ubuntu.com/ubuntu-ports jammy-backports InRelease  
Hit:4 http://ports.ubuntu.com/ubuntu-ports jammy-security InRelease  
Reading package lists... Done  
Building dependency tree... Done  
Reading state information... Done  
25 packages can be upgraded. Run 'apt list --upgradable' to see them.
```

## 1.2 File System Permissions and ACLs

### Command

```
#1 sudo mkdir -p /data/shared_project  
#2 sudo chown root:root /data/shared_project/  
#3 sudo chmod g+s /data/shared_project/  
#4 sudo setfacl -d -m u:sre_admin:rwx /data/shared_project/  
#5 getfacl /data/shared_project
```

### Explanation

#### #1 Command:

mkdir -p -> makes folder recursively

#### #2 Command:

chown root:root -> assigned user as root and group as root

#### #3 Command:

chmod g+s -> used setgid bit,for sub-folders,files to have same group as parent

#### #4 Command:

setfacl -> to set acl(access control list) list

-d -> for default acl(access control list) list

-m -> modify

u:sre\_admin:rwx -> giving user sre\_admin read,write and execute permissions

#### #5 Command:

getfacl -> for checking acl list

### Screenshot

```
sre_admin@sre-vm:~$ sudo setfacl -d -m u:sre_admin:rwx /data/shared_project/  
sre_admin@sre-vm:~$ getfacl /data/shared_project/  
getfacl: Removing leading '/' from absolute path names  
# file: data/shared_project/  
# owner: root  
# group: root  
# flags: -s-  
user::rwx  
group::r-x  
other::r-x  
default:user::rwx  
default:user:sre_admin:rwx  
default:group::r-x  
default:mask::rwx  
default:other::r-x
```

### 1.3 Service Management (systemd Timers & Unit Files)

#### cleanup\_tmp.sh Script:

```
#!/bin/bash

find /tmp -type f -mtime +7 -exec rm -f {} \;
```

#### Service Unit:

```
[Unit]
Description=Clean 7 days older /tmp files

[Service]
Type=oneshot
ExecStart=/usr/local/bin/cleanup_tmp.sh
```

#### Timer Unit:

```
[Unit]
Description=Daily clean 7 days older files of /tmp at 3:00 AM

[Timer]
OnCalendar=**-* 03:00:00
Persistent=true
Unit=cleanup-tmp.service

[Install]
WantedBy=timers.target
```

#### Command

```
#1 sudo systemctl enable --now cleanup-tmp.timer
#2 systemctl status cleanup-tmp.timer
#3 systemctl list-timers --all | grep cleanup-tmp
```

#### Explanation

##### #1 Command:

useradd -> add user  
 -m -> with a home directory  
 -s /bin/bash -> shell to use

##### #2 Command:

visudo -> opens sudoers file

##### #3 Command:

NOPASSWD:ALL -> No password prompted to run sudo

## Screenshot

```
sre_admin@sre-vm:~$ sudo systemctl daemon-reload
sre_admin@sre-vm:~$ sudo systemctl enable --now cleanup-tmp.timer
Created symlink /etc/systemd/system/timers.target.wants/cleanup-tmp.timer → /etc/systemd/system/cleanup-tmp.timer.
sre_admin@sre-vm:~$ systemctl status cleanup-tmp.timer
● cleanup-tmp.timer - Daily clean 7 days older files of /tmp at 3:00 AM
   Loaded: loaded (/etc/systemd/system/cleanup-tmp.timer; enabled; vendor preset: enabled)
     Active: active (waiting) since Thu 2025-07-24 15:52:16 UTC; 43s ago
       Trigger: Fri 2025-07-25 03:00:00 UTC; 11h left
     Triggers: ●cleanup-tmp.service
sre_admin@sre-vm:~$ systemctl list -all | grep cleanup_tpm.timer
Unknown command verb list.
sre_admin@sre-vm:~$ systemctl list -all | grep cleanup-tmp.timer
Unknown command verb list.
sre_admin@sre-vm:~$ systemctl list-units -all | grep cleanup-tmp.timer
                               loaded  active  waiting  Daily clean 7 days older files of /tmp at 3:00 AM
cleanup-tmp.timer
sre_admin@sre-vm:~$ systemctl list-timers --all | grep cleanup-tmp
Fri 2025-07-25 03:00:00 UTC 11h left      n/a          n/a  cleanup-tmp.timer           cleanup-tmp.service
sre_admin@sre-vm:~$ systemctl list-units -all | grep cleanup-tmp
                               loaded  inactive  dead    Clean 7 days older /tmp files
cleanup-tmp.service
cleanup-tmp.timer
sre_admin@sre-vm:~$ systemctl list-timers --all | grep cleanup-tmp
Fri 2025-07-25 03:00:00 UTC 10h left      n/a          n/a  cleanup-tmp.timer           cleanup-tmp.service
sre_admin@sre-vm:~$
```

## 1.4 Network Configuration (Static IP & DNS)

### Netplan Yaml:

```
network:
  version: 2
  ethernets:
    default:
      match:
        macaddress: 52:54:00:75:8b:6a
      dhcp-identifier: mac
      dhcp4: false
      addresses:
        - 192.168.64.100/24
      routes:
        - to: 0.0.0.0/0
          via: 192.168.64.1
    nameservers:
      addresses:
        - 8.8.8.8
        - 8.8.4.4
```

### Command

```
#1 ip a
#2 ip route
#3 netplan apply
#4 resolvectl status
#5 ping google.com
```

### Explanation

#### #1 Command:

ip a → network details like ipv4, ipv6 etc

## #2 Command:

```
ip route -> gateway
```

## #3 Command:

```
netplan apply -> it applies the changes done in yaml file
```

## #4 Command:

```
resolvectl status -> to check status of DNS i.e nameservers
```

## #5 Command:

```
ping -> send data packets to see connections
```

## Screenshot

```
ubuntu@sre-vm:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00 brd 00:00:00:00:00:00
        inet 127.0.0.1/8 scope host lo
            valid_lft forever preferred_lft forever
        inet6 ::1/128 scope host
            valid_lft forever preferred_lft forever
2: enp0s1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 52:54:00:75:8b:6a brd ff:ff:ff:ff:ff:ff
        inet 192.168.64.100/24 brd 192.168.64.255 scope global enp0s1
            valid_lft forever preferred_lft forever
        inet6 fdd1:c079:7aa8:366d:5054:ff:fe75:8b6a/64 scope global dynamic mngtmpaddr noprefixroute
            valid_lft 2591917sec preferred_lft 604717sec
        inet6 fe80::5054:ff:fe75:8b6a/64 scope link
            valid_lft forever preferred_lft forever
ubuntu@sre-vm:~$ ip route
default via 192.168.64.1 dev enp0s1 proto static
192.168.64.0/24 dev enp0s1 proto kernel scope link src 192.168.64.100
ubuntu@sre-vm:~$ resolvectl status
Global
    Protocols: -LLMNR -mDNS -DNSOverTLS DNSSEC=no/unsupported
resolv.conf mode: stub

Link 2 (enp0s1)
Current Scopes: DNS
    Protocols: +DefaultRoute +LLMNR -mDNS -DNSOverTLS DNSSEC=no/unsupported
    DNS Servers: 8.8.8.8 8.8.4.4 fe80::bcd0:74ff:feea:564%65535
ubuntu@sre-vm:~$ ping google.com
PING google.com (142.250.77.110) 56(84) bytes of data.
64 bytes from pnmaaaa-aq-in-f14.1e100.net (142.250.77.110): icmp_seq=1 ttl=112 time=49.2 ms
64 bytes from pnmaaaa-aq-in-f14.1e100.net (142.250.77.110): icmp_seq=2 ttl=112 time=56.9 ms
64 bytes from pnmaaaa-aq-in-f14.1e100.net (142.250.77.110): icmp_seq=3 ttl=112 time=73.6 ms
64 bytes from pnmaaaa-aq-in-f14.1e100.net (142.250.77.110): icmp_seq=4 ttl=112 time=70.9 ms
64 bytes from pnmaaaa-aq-in-f14.1e100.net (142.250.77.110): icmp_seq=5 ttl=112 time=123 ms
464 bytes from pnmaaaa-aq-in-f14.1e100.net (142.250.77.110): icmp_seq=6 ttl=112 time=69.7 ms
64 bytes from pnmaaaa-aq-in-f14.1e100.net (142.250.77.110): icmp_seq=7 ttl=112 time=69.7 ms
64 bytes from pnmaaaa-aq-in-f14.1e100.net (142.250.77.110): icmp_seq=8 ttl=112 time=47.8 ms
```

## 2 Logical Volume Management (LVM)

### 2.1 Setting up LVM

etc/fstab:

```
UUID=d324df7b-80f5-40dc-989f-f7cfbc32539e /mnt/logs xfs defaults 0 0
UUID=25524ee4-dbf4-427b-ab56-64c7aa0f1a3e /mnt/apps xfs defaults 0 0
```

### Command

```
#1 fallocate -l 5G /var/lib/sre-vm/sre_disk.img
#2 losetup -fP /var/lib/sre-vm/sre_disk.img
#3 pvcreate /dev/loop3
#4 pvdisplay
#5 vgcreate data_vg /dev/loop3
#6 vgdisplay
#7 lvcreate -L 2G -n lv_logs data_vg
#8 lvdisplay
#9 mkfs.xfs /dev/data_vg/lv_logs
#10 df -h | grep /mnt
```

### Explanation

#### #1 Command:

fallocate -> creates a file of size 5GB

#### #2 Command:

losetup -> make it behave as a block device

#### #3 Command:

pvdisplay -> creates a physical volume

#### #4 Command:

pvdisplay -> checks and verify pv volume

#### #5 Command:

vgcreate -> create volume group

#### #6 Command:

vgdisplay -> checks and verify volume groups

#### #7 Command:

lvcreate -> create logical volumes

#### #8 Command:

lvdisplay -> checks and verify logical volumes

#### #9 Command:

mkfs.xfs -> formats lvs with xfs filesystem

#### #10 Command:

df -h -> used to check disk in human readable form

## Screenshot

```
root@sre-vm:/# umount /mnt/logs /mnt/apps
root@sre-vm:/# mount -a
root@sre-vm:/# df -h | grep /mnt
/dev/mapper/data_vg-lv_logs  2.0G   47M  2.0G   3% /mnt/logs
/dev/mapper/data_vg-lv_apps  2.0G   47M  2.0G   3% /mnt/apps
root@sre-vm:/# lvdisplay
--- Logical volume ---
LV Path          /dev/data_vg/lv_logs
LV Name          lv_logs
VG Name          data_vg
LV UUID          aa60RE-Sf35-3IiW-pjSp-EIdu-JZkJ-StC0ZZ
LV Write Access  read/write
LV Creation host, time sre-vm, 2025-07-25 10:38:52 +0000
LV Status        available
# open           1
LV Size          2.00 GiB
Current LE       512
Segments         1
Allocation       inherit
Read ahead sectors auto
- currently set to 256
Block device    253:0

--- Logical volume ---
LV Path          /dev/data_vg/lv_apps
LV Name          lv_apps
VG Name          data_vg
LV UUID          kFebV6-CMDX-3fj7-xaGv-6V86-27c8-MzDlwa
LV Write Access  read/write
LV Creation host, time sre-vm, 2025-07-25 10:39:01 +0000
LV Status        available
# open           1
LV Size          2.00 GiB
Current LE       512
Segments         1
Allocation       inherit
Read ahead sectors auto
- currently set to 256
Block device    253:1
```

## 2.2 Extending an LV

### Command

```
#1 lvextend -L +1G /dev/data_vg/lv_logs
#2 lvdisplay /dev/data_vg/lv_logs
#3 xfs_growfs /mnt/logs
```

## Explanation

### #1 Command:

lvextend -> used to extend lvs

### #2 Command:

lvdisplay -> check and verify lvs

### #3 Command:

xfs\_growfs -> resize xfs filesystems

## Screenshot

```
root@sre-vm:/# lvextend -L +1G /dev/data_vg/lv_logs
Insufficient free space: 256 extents needed, but only 255 available
root@sre-vm:/# lvextend -L +1020M /dev/data_vg/lv_logs
Size of logical volume data_vg/lv_logs changed from 2.00 GiB (512 extents) to <3.00 GiB (767 extents).
Logical volume data_vg/lv_logs successfully resized.
root@sre-vm:/# lvdisplay /dev/data_vg/lv_logs
--- Logical volume ---
  LV Path          /dev/data_vg/lv_logs
  LV Name          lv_logs
  VG Name          data_vg
  LV UUID          aa60RE-Sf35-3IiW-pjSp-EIdu-JZkJ-StC0ZZ
  LV Write Access  read/write
  LV Creation host, time sre-vm, 2025-07-25 10:38:52 +0000
  LV Status        available
  # open           1
  LV Size          <3.00 GiB
  Current LE       767
  Segments         2
  Allocation       inherit
  Read ahead sectors auto
  - currently set to 256
  Block device     253:0

root@sre-vm:/# xfs_growfs /mnt/logs
meta-data=/dev/mapper/data_vg-lv_logs isize=512    agcount=4, agsize=131072 blks
              = sectsz=512  attr=2, projid32bit=1
              =         crc=1   finobt=1, sparse=1, rmapbt=0
              =         reflink=1 bigtime=0 inobtcount=0
data        =         bsize=4096 blocks=524288, imaxpct=25
              =         sunit=0   swidth=0 blks
naming      =version 2         bsize=4096 ascii-ci=0, ftype=1
log         =internal log     bsize=4096 blocks=2560, version=2
              =         sectsz=512 sunit=0 blks, lazy-count=1
realtime    =none             extsz=4096 blocks=0, rtextents=0
data blocks changed from 524288 to 785408
root@sre-vm:/# df -h /mnt/logs
Filesystem            Size  Used Avail Use% Mounted on
/dev/mapper/data_vg-lv_logs  3.0G  54M  3.0G  2% /mnt/logs
```

### 3 Advanced Troubleshooting & Monitoring

### 3.1 Process Tracing with strace

## Command

```
#1 strace ls
```

## Explanation

## #1 Command:

```
strace -> used to observe system calls  
execve() -> executing a program  
openat() -> opening a file  
read() -> reading file content
```

## Screenshot

### 3.2 Open Files with lsof

## Command

```
#1 ps aux | grep sshd  
#2 lsof -p 792  
#3 lsof +D var/log
```

## Explanation

### #1 Command:

`ps aux -> to find Process id of sshd`

### #2 Command:

`lsof -p -> what files are opened by processes with given pid`

### #3 Command:

`lsof +D var/log -> what processes are using this directory`

## Screenshot

```
root@sre-vm:/# ps aux | grep sshd
root      792  0.0  0.4 15148  8372 ?        Ss  05:06  0:00 sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups
root      796  0.0  0.4 18120  9556 ?        Ss  05:06  0:00 sshd: ubuntu [priv]
ubuntu    939  0.1  0.3 18512  6524 ?        S   05:06  0:02 sshd: ubuntu@notty
root     1905  0.0  0.4 18120  9568 ?        Ss  05:06  0:00 sshd: ubuntu [priv]
ubuntu    1954  0.0  0.3 18416  6452 ?        S   05:06  0:00 sshd: ubuntu@pts/0
root     28353 0.0  0.0  6416  1860 pts/1   S+  05:35  0:00 grep --color=auto sshd
root@sre-vm:/# lsof -p 792
COMMAND PID USER   FD   TYPE DEVICE SIZE/OFF NODE NAME
sshd    792 root cwd   DIR            8,1    4096    2 /
sshd    792 root rtd   DIR            8,1    4096    2 /
sshd    792 root txt   REG            8,1  859632  2754 /usr/sbin/sshd
sshd    792 root mem   REG            8,1  141304  3647 /usr/lib/aarch64-linux-gnu/libpgp-error.so.0.32.1
sshd    792 root mem   REG            8,1  178472  3637 /usr/lib/aarch64-linux-gnu/libtirpc.so.3.0.0
sshd    792 root mem   REG            8,1  60104  5047 /usr/lib/aarch64-linux-gnu/libresolv.so.2
sshd    792 root mem   REG            8,1  18280  3921 /usr/lib/aarch64-linux-gnu/libkeyutils.so.1.9
sshd    792 root mem   REG            8,1  47712  3990 /usr/lib/aarch64-linux-gnu/libkrb5support.so.0.1
sshd    792 root mem   REG            8,1  174424  4726 /usr/lib/aarch64-linux-gnu/libk5crypto.so.3.1
sshd    792 root mem   REG            8,1  530880  4811 /usr/lib/aarch64-linux-gnu/libpcre2-8.so.0.10.4
sshd    792 root mem   REG            8,1  893312  3643 /usr/lib/aarch64-linux-gnu/libgcrypt.so.20.3.4
sshd    792 root mem   REG            8,1  34712  4747 /usr/lib/aarch64-linux-gnu/libcap.so.2.44
sshd    792 root mem   REG            8,1  112632  3932 /usr/lib/aarch64-linux-gnu/liblz4.so.1.9.3
sshd    792 root mem   REG            8,1  730992  4807 /usr/lib/aarch64-linux-gnu/libstd.so.1.4.8
sshd    792 root mem   REG            8,1  157936  3934 /usr/lib/aarch64-linux-gnu/liblzm.so.5.2.5
sshd    792 root mem   REG            8,1  22760  3632 /usr/lib/aarch64-linux-gnu/libcap-ng.so.0.0.0
sshd    792 root mem   REG            8,1  88976  4805 /usr/lib/aarch64-linux-gnu/libnsl.so.2.0.1
sshd    792 root mem   REG            8,1  1637400  5035 /usr/lib/aarch64-linux-gnu/libc.so.6
sshd    792 root mem   REG            8,1  14184  3920 /usr/lib/aarch64-linux-gnu/libcom_err.so.2.1
sshd    792 root mem   REG            8,1  798536  3926 /usr/lib/aarch64-linux-gnu/libkrb5.so.3.3
sshd    792 root mem   REG            8,1  313784  3924 /usr/lib/aarch64-linux-gnu/libgssapi_krb5.so.2.2
sshd    792 root mem   REG            8,1  161936  3989 /usr/lib/aarch64-linux-gnu/libselinux.so.1
sshd    792 root mem   REG            8,1  186296  3638 /usr/lib/aarch64-linux-gnu/libcrypt.so.1.1.0
sshd    792 root mem   REG            8,1  104608  3918 /usr/lib/aarch64-linux-gnu/libz.so.1.2.11
sshd    792 root mem   REG            8,1  4044960  4771 /usr/lib/aarch64-linux-gnu/libcrypto.so.3
sshd    792 root mem   REG            8,1  808816  4801 /usr/lib/aarch64-linux-gnu/libsystemd.so.0.32.0
sshd    792 root mem   REG            8,1  59320  3633 /usr/lib/aarch64-linux-gnu/libpam.so.0.85.1
sshd    792 root mem   REG            8,1  128880  3656 /usr/lib/aarch64-linux-gnu/libaudit.so.1.0.0
sshd    792 root mem   REG            8,1  40464  5052 /usr/lib/aarch64-linux-gnu/libwrap.so.0.7.6
sshd    792 root mem   REG            8,1  187776  5029 /usr/lib/aarch64-linux-gnu/ld-linux-aarch64.so.1
sshd    792 root  0r  CHR            1,3    0t0   6 /dev/null
sshd    792 root  1u  unix 0xffff0000032e1980  0t0  17239 type=STREAM
sshd    792 root  2u  unix 0xffff0000032e1980  0t0  17239 type=STREAM
sshd    792 root  3u  IPv4          17249  0t0   TCP *:ssh (LISTEN)
sshd    792 root  4u  IPv6          17260  0t0   TCP *:ssh (LISTEN)
root@sre-vm:/# lsof var/log
root@sre-vm:/# lsof +D var/log
COMMAND PID USER   FD   TYPE DEVICE SIZE/OFF NODE NAME
systemd-j 381  root mem   REG            8,1  8388608 75818 var/log/journal/321caecd54d4403090abdd864dc7cf55/user-1000.journal
systemd-j 381  root mem   REG            8,1  8388608 75676 var/log/journal/321caecd54d4403090abdd864dc7cf55/system.journal
systemd-j 381  root  25u  REG            8,1  8388608 75676 var/log/journal/321caecd54d4403090abdd864dc7cf55/system.journal
systemd-j 381  root  33u  REG            8,1  8388608 75818 var/log/journal/321caecd54d4403090abdd864dc7cf55/user-1000.journal
rsyslogd 694 syslog  7w  REG            8,1  3452 75781 var/log/auth.log
rsyslogd 694 syslog  8w  REG            8,1  95410 75814 var/log/syslog
rsyslogd 694 syslog  9w  REG            8,1  43671 75815 var/log/kern.log
unattende 760  root  3w  REG            8,1      0 75816 var/log/unattended-upgrades/unattended-upgrades-shutdown.log
root@sre-vm:/#
```

### 3.3 Journalctl Filtering

#### Command

```
#1 logger -p user.error "Custom error"
#2 journalctl -p err -b
#3 journalctl -u apache2 --since "30 min ago"
```

#### Explanation

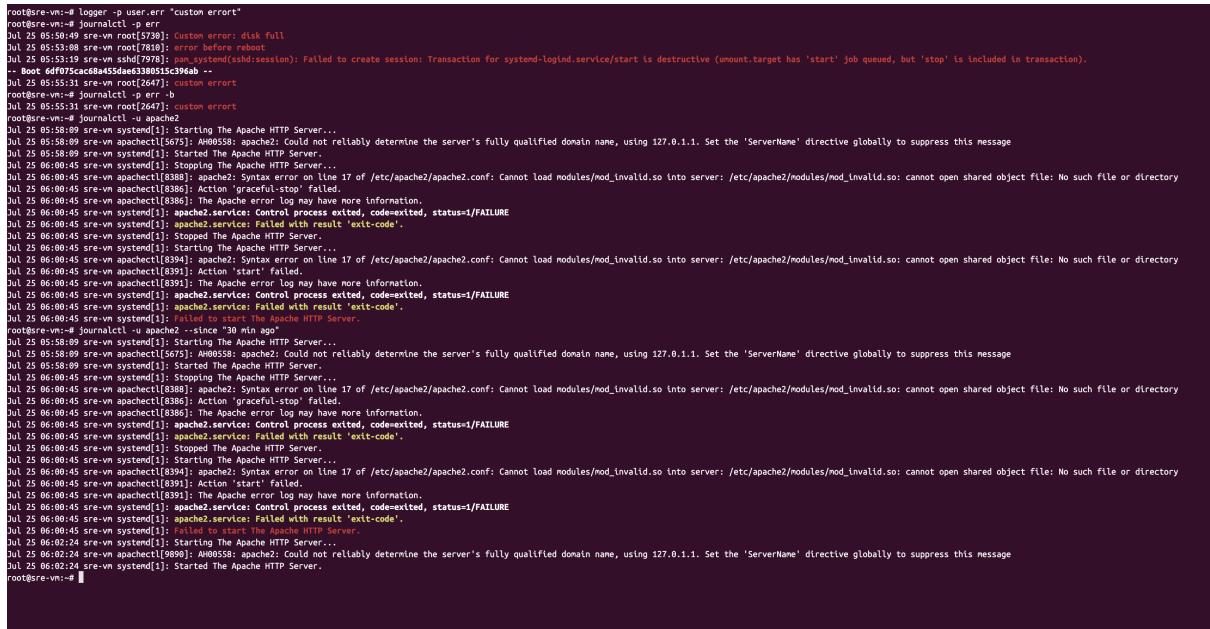
##### #1 Command:

logger -> to generate custom error

##### #2 Command:

journalctl -> to query logs  
 -p -> priority like err, debug, warn etc  
 -u -> users like apache2, firewalld etc

#### Screenshot



```
root@srivm:~# logger -p user.error "Custom error"
root@srivm:~# journalctl -p err
Jul 25 05:50:49 srivm root[5730]: Custom error: disk full
Jul 25 05:53:08 srivm root[7810]: error before reboot
Jul 25 05:53:08 srivm root[7810]: Failed to create session: Transaction for systemd-logind.service/start is destructive (umount.target has 'start' job queued, but 'stop' is included in transaction).
-- Boot: 64f075cacca4a43380515c3964b
Jul 25 05:55:31 srivm root[2647]: custom error
root@srivm:~# journalctl -p err -b
Jul 25 05:55:31 srivm root[2647]: custom error
root@srivm:~# journalctl -u apache2
Jul 25 05:58:09 srivm systemd[1]: Starting The Apache HTTP Server...
Jul 25 05:58:09 srivm apache2[15615]: AH00558: apache2: Could not reliably determine the server's fully qualified domain name, using 127.0.1.1. Set the 'ServerName' directive globally to suppress this message
Jul 25 05:58:09 srivm systemd[1]: Started The Apache HTTP Server.
Jul 25 06:00:45 srivm systemd[1]: Stopping The Apache HTTP Server...
Jul 25 06:00:45 srivm apache2[18388]: apache2: Syntax error on line 17 of /etc/apache2/apache2.conf: Cannot load modules/mod_invalid.so into server: /etc/apache2/modules/mod_invalid.so: cannot open shared object file: No such file or directory
Jul 25 06:00:45 srivm apache2[18386]: The Apache error log may have more information.
Jul 25 06:00:45 srivm apache2[18386]: Action 'graceful-stop' failed.
Jul 25 06:00:45 srivm systemd[1]: apache2.service: Failed with result 'exit-code'.
Jul 25 06:00:45 srivm systemd[1]: Stopped The Apache HTTP Server.
Jul 25 06:00:45 srivm systemd[1]: Starting The Apache HTTP Server...
Jul 25 06:00:45 srivm apache2[18388]: apache2: Syntax error on line 17 of /etc/apache2/apache2.conf: Cannot load modules/mod_invalid.so into server: /etc/apache2/modules/mod_invalid.so: cannot open shared object file: No such file or directory
Jul 25 06:00:45 srivm apache2[18386]: Action 'start' failed.
Jul 25 06:00:45 srivm apache2[18386]: The Apache error log may have more information.
Jul 25 06:00:45 srivm systemd[1]: apache2.service: Control process exited, code=exited, status=1/FAILURE
Jul 25 06:00:45 srivm systemd[1]: apache2.service: Failed with result 'exit-code'.
Jul 25 06:00:45 srivm systemd[1]: Failed to start The Apache HTTP Server.
root@srivm:~# journalctl -u apache2 --since "30 min ago"
Jul 25 05:58:09 srivm systemd[1]: Starting The Apache HTTP Server...
Jul 25 05:58:09 srivm apache2[15615]: AH00558: apache2: Could not reliably determine the server's fully qualified domain name, using 127.0.1.1. Set the 'ServerName' directive globally to suppress this message
Jul 25 05:58:09 srivm systemd[1]: Started The Apache HTTP Server.
Jul 25 06:00:45 srivm systemd[1]: Stopping The Apache HTTP Server...
Jul 25 06:00:45 srivm apache2[18388]: apache2: Syntax error on line 17 of /etc/apache2/apache2.conf: Cannot load modules/mod_invalid.so into server: /etc/apache2/modules/mod_invalid.so: cannot open shared object file: No such file or directory
Jul 25 06:00:45 srivm apache2[18386]: Action 'graceful-stop' failed.
Jul 25 06:00:45 srivm systemd[1]: apache2.service: Control process exited, code=exited, status=1/FAILURE
Jul 25 06:00:45 srivm systemd[1]: apache2.service: Failed with result 'exit-code'.
Jul 25 06:00:45 srivm systemd[1]: Stopped The Apache HTTP Server.
Jul 25 06:00:45 srivm systemd[1]: Starting The Apache HTTP Server...
Jul 25 06:00:45 srivm apache2[18388]: apache2: Syntax error on line 17 of /etc/apache2/apache2.conf: Cannot load modules/mod_invalid.so into server: /etc/apache2/modules/mod_invalid.so: cannot open shared object file: No such file or directory
Jul 25 06:00:45 srivm apache2[18386]: Action 'start' failed.
Jul 25 06:00:45 srivm apache2[18386]: The Apache error log may have more information.
Jul 25 06:00:45 srivm systemd[1]: apache2.service: Control process exited, code=exited, status=1/FAILURE
Jul 25 06:00:45 srivm systemd[1]: apache2.service: Failed with result 'exit-code'.
Jul 25 06:02:24 srivm systemd[1]: Failed to start The Apache HTTP Server.
Jul 25 06:02:24 srivm systemd[1]: Starting The Apache HTTP Server...
root@srivm:~#
```

## 4 Security Enhancements

### 4.1 Apparmor Basics

#### Command

```
#1 aa-status
#2 aa-complain /usr/bin/man
#3 aa-enforce /usr/bin/man
```

#### Explanation

##### #1 Command:

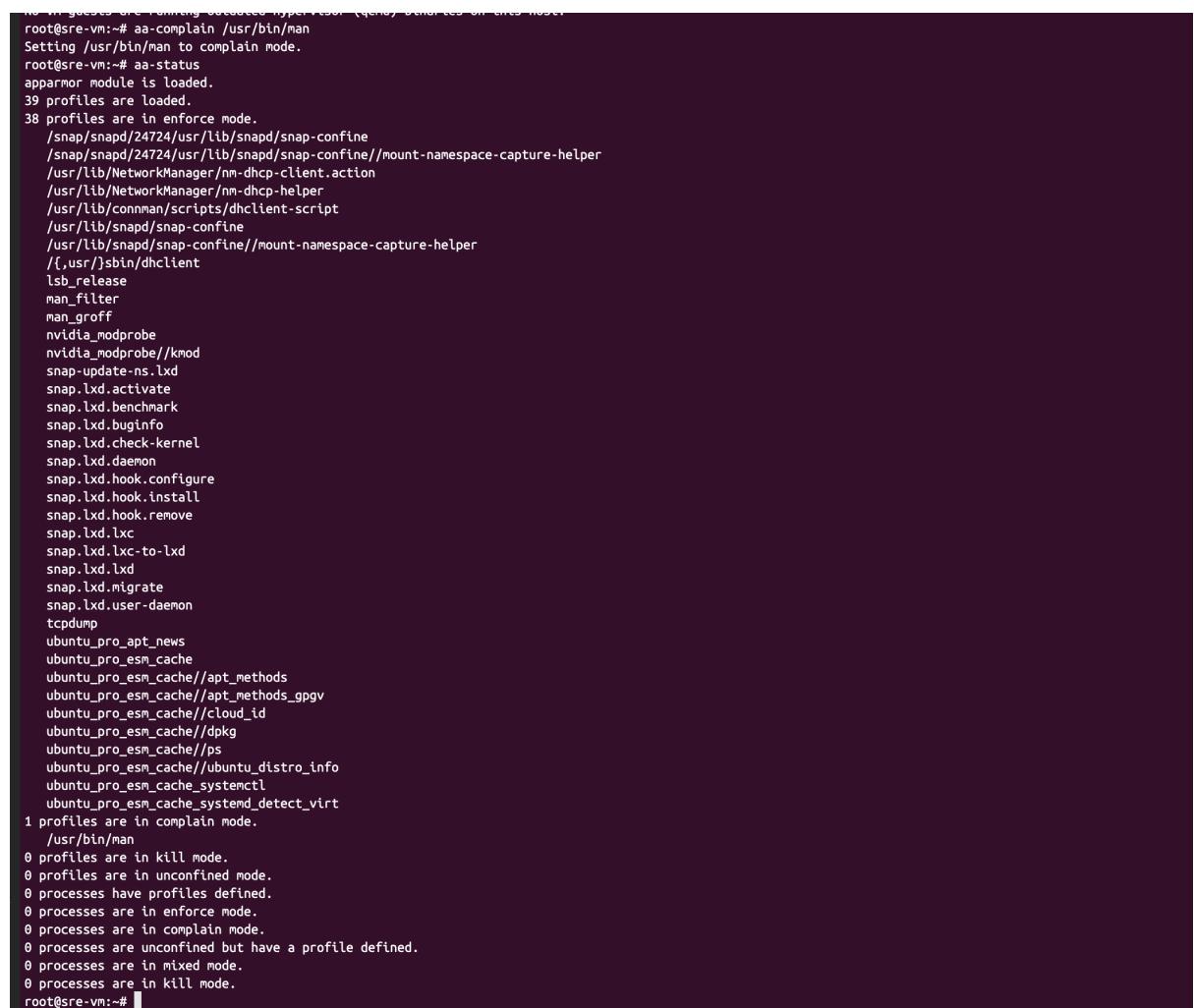
aa-status -> gives status what file are in complain or enforce mode

##### #2 Command:

aa-complain -> violation are logged not blocked

aa-enforce -> violation are blocked

#### Screenshot



```
root@sre-vm:~# aa-complain /usr/bin/man
Setting /usr/bin/man to complain mode.
root@sre-vm:~# aa-status
apparmor module is loaded.
39 profiles are loaded.
38 profiles are in enforce mode.
/snap/snapd/24724/usr/lib/snapd/snap-confine
/snap/snapd/24724/usr/lib/snapd/snap-confine//mount-namespace-capture-helper
/usr/lib/NetworkManager/nm-dhcp-client.action
/usr/lib/NetworkManager/nm-dhcp-helper
/usr/lib/connman/scripts/dhcclient-script
/usr/lib/snapd/snap-confine
/usr/lib/snapd/snap-confine//mount-namespace-capture-helper
/[,usr/]sbin/dhcclient
lsb_release
man_filter
man_groff
nvidia_modprobe
nvidia_modprobe//kmod
snap-update-ns.lxd
snap.lxd.activate
snap.lxd.benchmark
snap.lxd.buginfo
snap.lxd.check-kernel
snap.lxd.daemon
snap.lxd.hook.configure
snap.lxd.hook.install
snap.lxd.hook.remove
snap.lxd.lxc
snap.lxd.lxc-to-lxd
snap.lxd.lxd
snap.lxd.migrate
snap.lxd.user-daemon
tcpdump
ubuntu_pro_apt_news
ubuntu_pro_esm_cache
ubuntu_pro_esm_cache//apt_methods
ubuntu_pro_esm_cache//apt_methods_gpgv
ubuntu_pro_esm_cache//cloud_id
ubuntu_pro_esm_cache//dpkg
ubuntu_pro_esm_cache//ps
ubuntu_pro_esm_cache//ubuntu_distro_info
ubuntu_pro_esm_cache_systemctl
ubuntu_pro_esm_cache_systemd_detect_virt
1 profiles are in complain mode.
/usr/bin/man
0 profiles are in kill mode.
0 profiles are in unconfined mode.
0 processes have profiles defined.
0 processes are in enforce mode.
0 processes are in complain mode.
0 processes are unconfined but have a profile defined.
0 processes are in mixed mode.
0 processes are in kill mode.
root@sre-vm:~#
```

## 4.2 File Attributes with chattr

### Command

```
#1 chattr +i /etc/passwd  
# chattr -i /etc/passwd
```

### Explanation

#### #1 Command:

chattr +i -> chattr is used to manage inode metafile system flags.

+i -> make file immutable

-i -> removes immutability

### Screenshot



## 5 Advanced Automation & Scripting

### 5.1 disk\_monitor.sh Script

Script:

```
#!/bin/bash

THRESHOLD=10
LOG_FILE="/var/log/disk_monitor.log"
EMAIL_LOG="/var/log/disk_monitor_email.log"
TO_EMAIL="s_kumar4@ph.iitr.ac.in"

USAGE=$(df / | grep / | awk '{print $5}' | sed 's/%//')
TIMESTAMP=$(date "+%Y-%m-%d %H:%M:%S")

if [ "$USAGE" -gt "$THRESHOLD" ]; then
    echo "[${TIMESTAMP}] WARNING: Disk usage is at ${USAGE}% on /"
    echo "[${TIMESTAMP}] WARNING: Disk usage is at ${USAGE}% on /" >> "$LOG_FILE"

    echo "Disk usage alert: ${USAGE}% used on root (/) as of ${TIMESTAMP} \
        | mail -s \"Disk usage Warning on $(hostname)\" \"$TO_EMAIL\""

    echo "[${TIMESTAMP}] Email alert sent to $TO_EMAIL FOR DISK USAGE ${USAGE}%" >>
        "$EMAIL_LOG"
fi
```

### Cron Job

```
# Cron entry for every 5 mins
*/5 * * * * /scripts/disk_monitor.sh
```

### Command

```
#1 crontab -e
#2 apt install mailutils
#3 df / | grep / | awk '{print $5}' | sed 's/%//'
#4 echo "msg" | mail -s "subject" my@example.com
```

### Explanation

#### #1 Command:

crontab -e -> used for scheduling task

#### #2 Command:

mailutils -> helps to send mail

#### #3 Command:

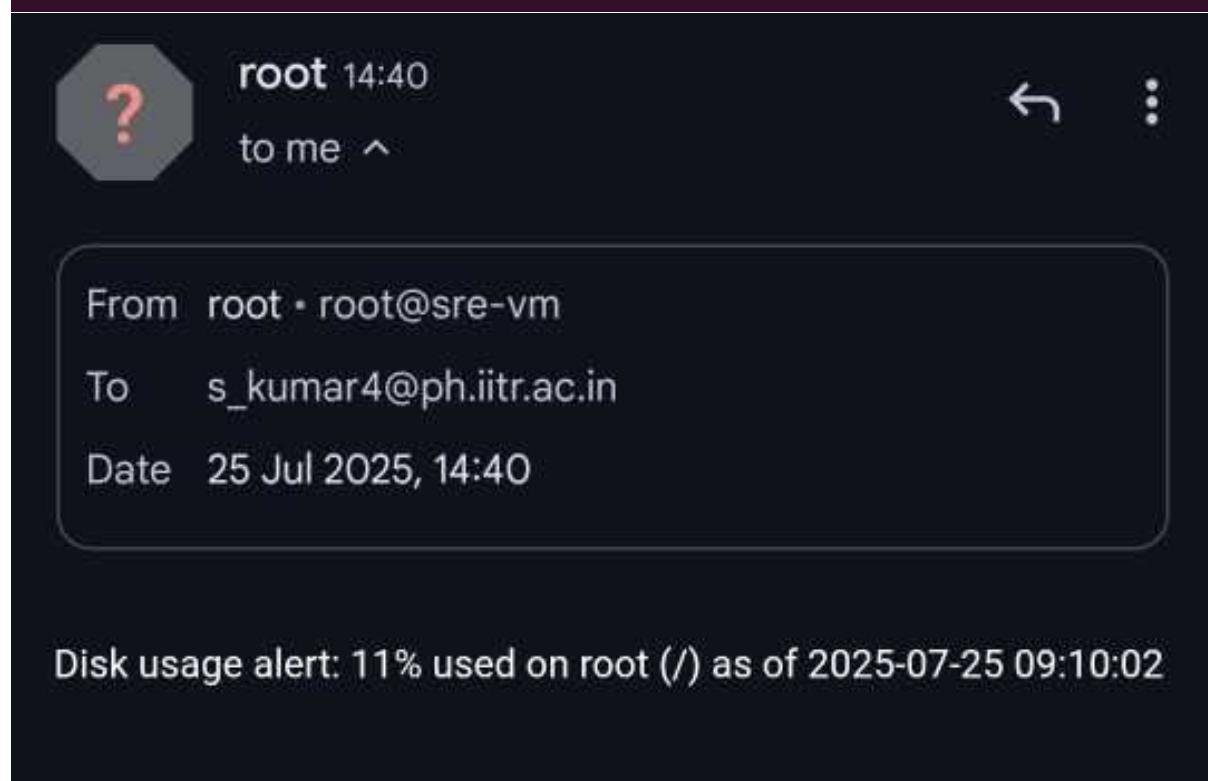
df / -> disk utilization information  
awk '{print \$5}' / -> to get data from column 5  
sed 's/%//' -> to filter number from percentage

**#4 Command:**

```
mail -s "subject" $email -> this is used with echo to send mail to given email
```

**Screenshot**

```
You have new mail in /var/mail/root
root@sre-vm:/# cat var/log/disk_monitor_email.log
[2025-07-25 07:58:45] Email alert sent to s_kumar4@ph.iitr.ac.in FOR DISK USAGE 11%
[2025-07-25 08:10:01] Email alert sent to s_kumar4@ph.iitr.ac.in FOR DISK USAGE 11%
[2025-07-25 09:00:01] Email alert sent to siddharth5.intern@phonepe.com FOR DISK USAGE 11%
[2025-07-25 09:05:01] Email alert sent to siddharth5.intern@phonepe.com FOR DISK USAGE 11%
[2025-07-25 09:10:02] Email alert sent to s_kumar4@ph.iitr.ac.in FOR DISK USAGE 11%
root@sre-vm:/# cat var/log/disk_monitor.log
[2025-07-25 07:58:45] WARNING: Disk usage is at 11% on /
[2025-07-25 08:10:01] WARNING: Disk usage is at 11% on /
[2025-07-25 09:00:01] WARNING: Disk usage is at 11% on /
[2025-07-25 09:05:01] WARNING: Disk usage is at 11% on /
[2025-07-25 09:10:02] WARNING: Disk usage is at 11% on /
root@sre-vm:/#
```



## 5.2 user\_report.sh Script

Script:

```
#!/bin/bash

get_user_info() {
local username=$1

if id "$username" &>/dev/null; then
echo "User: $username"
echo "UID: $(id -u $username)"
echo "GID: $(id -g $username)"
echo "Groups: $(id -Gn $username)"
echo "Home directory: $(getent passwd $username | cut -d: -f6)"
echo "Shell: $(getent passwd $username | cut -d: -f7)"
else
echo "Error: User '$username' does not exist."
fi

}

if [ -z "$1" ]; then
echo "Usage: $0 <username>"
exit 1
fi

get_user_info "$1"
```

### Explanation

#### #1 Command:

\$1 -> first argument

#### #2 Command:

id "\$username" -> checks if username exists  
-u -> gives UID  
-g -> gives GID  
-Gn -> primary and secondary groups

#### #3 Command:

-z "\$1" -> checks if first argument is empty

#### #4 Command:

exit 1 -> exits the script

## Screenshots

```
root@sre-vm:/# /scripts/user_reports.sh root
User: root
UID: 0
GID: 0
Groups: root
Home directory: /root
Shell: /bin/bash
root@sre-vm:/# /scripts/user_reports.sh sid1
User: sid1
UID: 1001
GID: 1001
Groups: sid1
Home directory: /home/sid1
Shell: /bin/sh
root@sre-vm:/# /scripts/user_reports.sh sid
Error: User 'sid' does not exist.
root@sre-vm:/# /scripts/user_reports.sh user1
Error: User 'user1' does not exist.
root@sre-vm:/#
```

## Conclusion

This document demonstrates the execution and understanding of essential Linux system administration tasks on a UBUNTU-like environment using Multipass.