

# Week 2 Journal: Security Planning and Testing

## Methodology

Module: Operating Systems

Assessment: Week 2

Phase 2: Security Planning and Testing Methodology

### 1. Performance Testing Plan

For this week's task, my aim was to see how my Ubuntu system performs when it's under different types of loads like CPU, memory, disk, and network. I also wanted to set up remote monitoring from my Linux Mint machine so I could control and test Ubuntu from there. Both systems were connected through a **Host-only adapter** in VirtualBox, which made them communicate directly without internet.

### Tools Installed

On both Ubuntu and Linux Mint, I used the command below to install all the tools I needed:

```
sudo apt update && sudo apt install -y htop sysstat iotop iperf3 stress-ng ufw fail2ban  
unattended-upgrades
```

These tools helped me with monitoring system performance, running stress tests, and setting up security features.

### Remote Monitoring Setup

After checking that both machines had IP addresses on the same network (Ubuntu: 192.168.56.101, Mint: 192.168.56.102), I connected from Mint to Ubuntu using SSH. The connection worked successfully, which confirmed that remote access and monitoring were properly set up.

```
inet6 fe80::2ad4:b99f:e03f:8d4b/64 scope link noprefixroute
valid lft forever preferred lft forever
abulaze@VirtualBox:~$ ssh abulaze@192.168.56.101
abulaze@192.168.56.101's password:
Welcome to Ubuntu 24.04.3 LTS (GNU/Linux 6.14.0-33-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

Expanded Security Maintenance for Applications is not enabled.

6 updates can be applied immediately.
3 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

Last login: Tue Oct 28 18:02:59 2025 from 192.168.56.102
abulaze@Ubuntu:~$
```

2. Baseline Performance Test (Idle)

Before stressing the system, I recorded the idle performance to see how the machine behaves normally. I used vmstat, iostat, and htop. The CPU usage stayed below 10%, memory was mostly free, and the system was stable. This gave me a clean baseline for comparing later tests.

```
abulaze@Ubuntu: ~
0% 1.3% Tasks: 110, 343 thr, 90 kthr; 2 running
1% 0.7% Load average: 0.01 0.15 0.16
Mem: 993M/3.82G Uptime: 00:11:03
Swp: 0K/0K

Main I/O
PID USER PRI NI VIRT RES SHR S CPU%MEM% TIME+ Command
3408 abulaze 20 0 11592 5316 3652 R 8.4 0.1 0:01.18 htop
1877 abulaze 20 0 3877M 388M 149M S 1.3 9.9 0:12.63 /usr/bin/gnome-shell
1389 root 20 0 308M 9180 7644 S 0.6 0.2 0:00.66 /usr/libexec/upowerd
1907 abulaze 20 0 3877M 388M 149M S 0.6 9.9 0:02.52 /usr/bin/gnome-shell
1 root 20 0 24408 14340 9476 S 0.0 0.4 0:03.46 /sbin/init splash
249 root 19 -1 58840 17132 15724 S 0.0 0.4 0:00.63 /usr/lib/systemd/systemd-journald
321 root 20 0 38588 8428 4844 S 0.0 0.2 0:00.27 /usr/lib/systemd/systemd-udevd
396 systemd-oo 20 0 17560 7552 6656 S 0.0 0.2 0:00.84 /usr/lib/systemd/systemd-oomd
400 systemd-re 20 0 21584 12976 18800 S 0.0 0.3 0:00.20 /usr/lib/systemd/systemd-resolved
405 systemd-ti 20 0 91048 7824 6928 S 0.0 0.2 0:00.08 /usr/lib/systemd/systemd-timesyncd
514 systemd-ti 20 0 91048 7824 6928 S 0.0 0.2 0:00.00 /usr/lib/systemd/systemd-timesyncd
626 avahi 20 0 8672 4488 4104 S 0.0 0.1 0:00.28 avahi-daemon: running [Ubuntu.local]
627 messagebus 20 0 12164 7096 4536 S 0.0 0.2 0:01.27 @dbus-daemon --system --address=systemd: --nofork --nopidfile --systemd-activation --syslog
637 gnome-remo 20 0 350M 16408 8976 S 0.0 0.4 0:00.00 /usr/libexec/gnome-remote-desktop-daemon --system
649 polkitd 20 0 381M 12824 8224 S 0.0 0.3 0:00.45 /usr/lib/polkit-1/polkitd --no-debug
657 root 20 0 386M 7220 6580 S 0.0 0.2 0:00.04 /usr/libexec/power-profiles-daemon
667 root 20 0 1805M 39284 24784 S 0.0 1.0 0:00.19 /usr/lib/snapd/snapd
676 root 20 0 386M 7220 6580 S 0.0 0.2 0:00.00 /usr/libexec/power-profiles-daemon
677 root 20 0 386M 7220 6580 S 0.0 0.2 0:00.00 /usr/libexec/power-profiles-daemon
679 root 20 0 386M 7220 6580 S 0.0 0.2 0:00.01 /usr/libexec/power-profiles-daemon
685 root 20 0 385M 7720 6824 S 0.0 0.2 0:00.08 /usr/libexec/accounts-daemon
692 root 20 0 9424 2624 2496 S 0.0 0.1 0:00.04 /usr/sbin/cron -f -P
697 root 20 0 382M 6924 6284 S 0.0 0.2 0:00.06 /usr/libexec/switcheroo-control
708 root 20 0 1805M 39284 24784 S 0.0 1.0 0:00.39 /usr/lib/snapd/snapd
701 root 20 0 1805M 39284 24784 S 0.0 1.0 0:00.14 /usr/lib/snapd/snapd
702 root 20 0 1805M 39284 24784 S 0.0 1.0 0:00.00 /usr/lib/snapd/snapd
703 root 20 0 1805M 39284 24784 S 0.0 1.0 0:00.00 /usr/lib/snapd/snapd
704 root 20 0 18256 8888 7864 S 0.0 0.2 0:00.29 /usr/lib/systemd/systemd-logind
709 root 20 0 458M 14396 11708 S 0.0 0.4 0:00.12 /usr/libexec/udisks2/udisksd
712 root 20 0 382M 6924 6284 S 0.0 0.2 0:00.00 /usr/libexec/switcheroo-control
713 root 20 0 382M 6924 6284 S 0.0 0.2 0:00.00 /usr/libexec/switcheroo-control
715 root 20 0 385M 7720 6824 S 0.0 0.2 0:00.00 /usr/libexec/accounts-daemon
716 root 20 0 385M 7720 6824 S 0.0 0.2 0:00.10 /usr/libexec/accounts-daemon
F1Help F2Setup F3Search F4Filter F5Tree F6SortBy F7Nice F8Nice F9Kill F10Quit
```

3. CPU and Memory Stress Test

Next, I tested how the system reacts when CPU and memory are under pressure. I ran the stress-ng command, which pushed CPU to 100% and used about 50% of memory. The SSH connection stayed stable and after the test ended, everything went back to normal.

```
valid_lft 86242sec preferred_lft 14242sec
inet6 fe80::fe6e:feb4:b6a4:71e6/64 scope link noprefixroute
valid_lft forever preferred_lft forever
abdulazeer@Ubuntu:~$ stress-ng --cpu 2 --vm 1 --vm-bytes 50% --timeout 2m --metrics-brief
stress-ng: info: [3548] setting to a 2 mins, 0 secs run per stressor
stress-ng: info: [3548] dispatching hogs: 2 cpu, 1 vm
stress-ng: metrc: [3548] stressor      bogo ops real time  usr time  sys time  bogo ops/s  bogo ops/s
stress-ng: metrc: [3548]                (secs)  (secs)  (secs)  (real time) (usr+sys time)
stress-ng: metrc: [3548] cpu          398298   138.48   197.64    63.37    2876.28    1526.01
stress-ng: metrc: [3548] vm          345342   138.52    1.86    13.44    2493.08    22569.09
stress-ng: info: [3548] skipped: 0
stress-ng: info: [3548] passed: 3: cpu (2) vm (1)
stress-ng: info: [3548] failed: 0
stress-ng: info: [3548] metrics untrustworthy: 0
stress-ng: info: [3548] successful run completed in 2 mins, 18.53 secs
abdulazeer@Ubuntu:~$
```

## 4. Disk I/O Stress Test

Then I tested how the disk handles heavy writing operations using stress-ng and monitored it with iotop. I saw stress-ng writing heavily to disk but the system stayed responsive.

```
abdulazeer@Ubuntu: ~
Total DISK READ:      0.00 B/s | Total DISK WRITE:      0.00 B/s
Current DISK READ:    0.00 B/s | Current DISK WRITE:    0.00 B/s
  PID  PRIO  USER      DISK READ  DISK WRITE  COMMAND
  ----  ----  -
  249  be/3  root        0.00 B     1016.00 K  systemd-journald
  199  be/3  root        0.00 B     368.00 K  [jbd2/sda2-8]
  761  be/4  syslog      0.00 B     24.00 K  rsyslogd -n -iNONE
 3676  be/4  root        28.00 K     8.00 K  fwupd
  740  be/4  root        0.00 B     4.00 K  NetworkManager --no-daemon
    1  be/4  root        8.00 K     0.00 B  init splash
   31  be/4  root        2.11 M     0.00 B  [kworker/u10:0-events_unbound]
 3420  be/4  root       140.00 K     0.00 B  [kworker/u9:2-ext4-rsv-conversion]
 3603  be/4  root       176.00 K     0.00 B  [kworker/u10:3-events_power_efficient]

stress-ng: info: [3548] successful run completed in 2 mins, 18.53 secs
abdulazeer@Ubuntu:~$ stress-ng --hdd 1 --hdd-ops 20000 --timeout 2m --metrics-brief
stress-ng: info: [3586] setting to a 2 mins, 0 secs run per stressor
stress-ng: info: [3586] dispatching hogs: 1 hdd
stress-ng: metrc: [3586] stressor      bogo ops real time  usr time  sys time  bogo ops/s  bogo ops/s
stress-ng: metrc: [3586]                (secs)  (secs)  (secs)  (real time) (usr+sys time)
stress-ng: metrc: [3586] hdd          20000    3.38    0.53    2.05    5923.00    7776.88
stress-ng: info: [3586] skipped: 0
stress-ng: info: [3586] passed: 1: hdd (1)
stress-ng: info: [3586] failed: 0
stress-ng: info: [3586] metrics untrustworthy: 0
stress-ng: info: [3586] successful run completed in 3.38 secs
abdulazeer@Ubuntu:~$ sudo iotop -oPa
[sudo] password for abdulazeer:
abdulazeer@Ubuntu:~$
```

## 5. Network Throughput Test (Host-only Connection)

The last part of the performance test was checking network speed between Mint and Ubuntu using iperf3. Due to some connection errors, the test wasn't fully complete at first, but the setup and commands were correctly configured.

```
azeez@azeez-VirtualBox:~$
[ 9] 25.01-26.06 sec 104 MBytes 474 Mb/s/sec 384 174 KBytes
[11] 25.01-26.06 sec 77.2 MBytes 351 Mb/s/sec 419 63.6 KBytes
[SUM] 25.01-26.06 sec 377 MBytes 1.71 Gb/s/sec 1332
-----
[ 5] 26.06-27.00 sec 7.08 MBytes 455 Mb/s/sec 82 126 KBytes
[ 7] 26.06-27.00 sec 4.25 MBytes 245 Mb/s/sec 0 126 KBytes
[ 9] 26.06-27.00 sec 7.12 MBytes 411 Mb/s/sec 0 286 KBytes
[11] 26.06-27.00 sec 3.02 MBytes 209 Mb/s/sec 0 93.3 KBytes
[SUM] 26.06-27.00 sec 22.9 MBytes 1.32 Gb/s/sec 82
-----
[ 5] 27.00-28.01 sec 36.6 MBytes 305 Mb/s/sec 154 140 KBytes
[ 7] 27.00-28.01 sec 54.6 MBytes 450 Mb/s/sec 130 147 KBytes
[ 9] 27.00-28.01 sec 43.5 MBytes 363 Mb/s/sec 191 171 KBytes
[11] 27.00-28.01 sec 41.5 MBytes 346 Mb/s/sec 171 177 KBytes
[SUM] 27.00-28.01 sec 176 MBytes 1.47 Gb/s/sec 646
-----
[ 5] 28.01-29.01 sec 21.9 MBytes 435 Mb/s/sec 191 91.9 KBytes
[ 7] 28.01-29.01 sec 51.6 MBytes 432 Mb/s/sec 180 137 KBytes
[ 9] 28.01-29.01 sec 38.8 MBytes 325 Mb/s/sec 226 150 KBytes
[11] 28.01-29.01 sec 44.6 MBytes 375 Mb/s/sec 180 229 KBytes
[SUM] 28.01-29.01 sec 187 MBytes 1.57 Gb/s/sec 777
-----
[ 5] 29.01-30.02 sec 30.8 MBytes 255 Mb/s/sec 217 189 KBytes
[ 7] 29.01-30.02 sec 56.1 MBytes 465 Mb/s/sec 180 165 KBytes
[ 9] 29.01-30.02 sec 41.8 MBytes 346 Mb/s/sec 173 115 KBytes
[11] 29.01-30.02 sec 49.4 MBytes 409 Mb/s/sec 165 187 KBytes
[SUM] 29.01-30.02 sec 178 MBytes 1.48 Gb/s/sec 735
-----
[ ID] Interval Transfer Bitrate Retr
[ 5] 0.00-30.02 sec 1.24 GBytes 356 Mb/s/sec 5465 sender
[ 8] 0.00-30.02 sec 1.24 GBytes 355 Mb/s/sec receiver
[ 7] 0.00-30.02 sec 1.27 GBytes 363 Mb/s/sec 5393 sender
[ 9] 0.00-30.02 sec 1.27 GBytes 363 Mb/s/sec 5003 receiver
[11] 0.00-30.02 sec 1.24 GBytes 355 Mb/s/sec sender
[ 5] 0.00-30.05 sec 1.24 GBytes 354 Mb/s/sec receiver
[ 8] 0.00-30.05 sec 1.28 GBytes 367 Mb/s/sec 4931 sender
[11] 0.00-30.05 sec 1.28 GBytes 366 Mb/s/sec receiver
[SUM] 0.00-30.02 sec 5.03 GBytes 1.44 Gb/s/sec 20762 sender
[SUM] 0.00-30.05 sec 5.03 GBytes 1.44 Gb/s/sec receiver
iperf Done.
azeez@azeez-VirtualBox:~$
```

```
[10] 27.00-28.00 sec 42.8 MBytes 358 Mb/s/sec
[12] 27.00-28.00 sec 40.5 MBytes 339 Mb/s/sec
[SUM] 27.00-28.00 sec 174 MBytes 1.46 Gb/s/sec
-----
[ 5] 28.00-29.00 sec 51.4 MBytes 431 Mb/s/sec
[ 8] 28.00-29.00 sec 52.1 MBytes 438 Mb/s/sec
[10] 28.00-29.00 sec 38.9 MBytes 326 Mb/s/sec
[12] 28.00-29.00 sec 45.2 MBytes 380 Mb/s/sec
[SUM] 28.00-29.00 sec 188 MBytes 1.58 Gb/s/sec
-----
[ 5] 29.00-30.00 sec 30.5 MBytes 256 Mb/s/sec
[ 8] 29.00-30.00 sec 56.0 MBytes 469 Mb/s/sec
[10] 29.00-30.00 sec 41.8 MBytes 350 Mb/s/sec
[12] 29.00-30.00 sec 48.5 MBytes 407 Mb/s/sec
[SUM] 29.00-30.00 sec 177 MBytes 1.48 Gb/s/sec
-----
[ 5] 30.00-30.05 sec 1.38 MBytes 251 Mb/s/sec
[ 8] 30.00-30.05 sec 1.88 MBytes 342 Mb/s/sec
[10] 30.00-30.05 sec 2.00 MBytes 365 Mb/s/sec
[12] 30.00-30.05 sec 2.25 MBytes 410 Mb/s/sec
[SUM] 30.00-30.05 sec 7.50 MBytes 1.37 Gb/s/sec
-----
[ ID] Interval Transfer Bitrate
[ 5] 0.00-30.05 sec 1.24 GBytes 355 Mb/s/sec
[ 8] 0.00-30.05 sec 1.27 GBytes 363 Mb/s/sec
[10] 0.00-30.05 sec 1.24 GBytes 354 Mb/s/sec
[12] 0.00-30.05 sec 1.28 GBytes 366 Mb/s/sec
[SUM] 0.00-30.05 sec 5.03 GBytes 1.44 Gb/s/sec
-----
Server listening on 5201 (test #2)
-----
```

6. Security Configuration Checklist

After performance testing, I moved on to security setup on my Ubuntu (target) system.

Security Task	Command / Action	Description
SSH Hardening	Edit /etc/ssh/sshd_config	Disabled root login, set password auth to no, limited tries
Firewall Setup	sudo ufw default deny incoming sudo ufw allow OpenSSH sudo ufw enable	Allowed only SSH traffic, blocked all others
Fail2ban	sudo apt install -y fail2ban sudo systemctl enable --now fail2ban	Blocked repeated failed login attempts
Automatic Updates	sudo apt install -y unattended-upgrades	Keeps security patches up-to-date
AppArmor	sudo aa-status	Verified AppArmor profiles were active

Least Privilege	sudo -l	Checked minimal sudo privileges
-----------------	---------	---------------------------------

```
abdulazeer@Ubuntu:~$ sudo ufw enable
Firewall is active and enabled on system startup
abdulazeer@Ubuntu:~$ sudo ufw status
Status: active

To Action From
--
OpenSSH ALLOW Anywhere
OpenSSH (v6) ALLOW Anywhere (v6)

abdulazeer@Ubuntu:~$ sudo apt install -y fail2ban
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
fail2ban is already the newest version (1.0.2-3ubuntu0.1).
The following package was automatically installed and is no longer required:
 liblvm19
Use 'sudo apt autoremove' to remove it.
0 upgraded, 0 newly installed, 0 to remove and 6 not upgraded.
abdulazeer@Ubuntu:~$ sudo systemctl enable --now fail2ban
sudo: systemctl: command not found
abdulazeer@Ubuntu:~$ sudo system1 enable --now fail2ban
sudo: system1: command not found
abdulazeer@Ubuntu:~$ sudo systemtl enable --now fail2ban
sudo: systemtl: command not found
abdulazeer@Ubuntu:~$ sudo systemctl enable --now fail2ban
Synchronizing state of fail2ban.service with SysV service script with /usr/lib/systemd/systemd-sysv-install.
Executing: /usr/lib/systemd/systemd-sysv-install enable fail2ban
abdulazeer@Ubuntu:~$ sudo systemctl status fail2ban
● fail2ban.service - Fail2Ban Service
   Loaded: loaded (/usr/lib/systemd/system/fail2ban.service; enabled; preset: enabled)
   Active: active (running) since Sat 2025-11-01 11:32:06 UTC; 50min ago
     Docs: man:fail2ban(1)
```

7. Threat Model and Mitigation

I created a simple table showing common threats and how I mitigated them.

Threat	Description	Mitigation
Brute-force SSH attacks	Attackers try random passwords to log in	Disabled password login, enabled Fail2ban
Unpatched vulnerabilities	Hackers exploit outdated packages	Enabled automatic updates
Privilege escalation	Normal user gains root access	Applied least privilege and limited sudo
Open ports exposure	Attackers scan for open services	Configured UFW to allow only SSH

Configuration tampering	Unauthorized system changes	Restricted config access and backups
-------------------------	-----------------------------	--------------------------------------

## 8. Conclusion

In this week's task, I learned how to monitor system performance, apply stress tests, and set up a basic security plan for a Linux system. The hardest part was getting the network connection to work for the iperf3 test, but I understood the full setup process and command usage. This gave me a better idea of how real admins test performance and secure their systems.