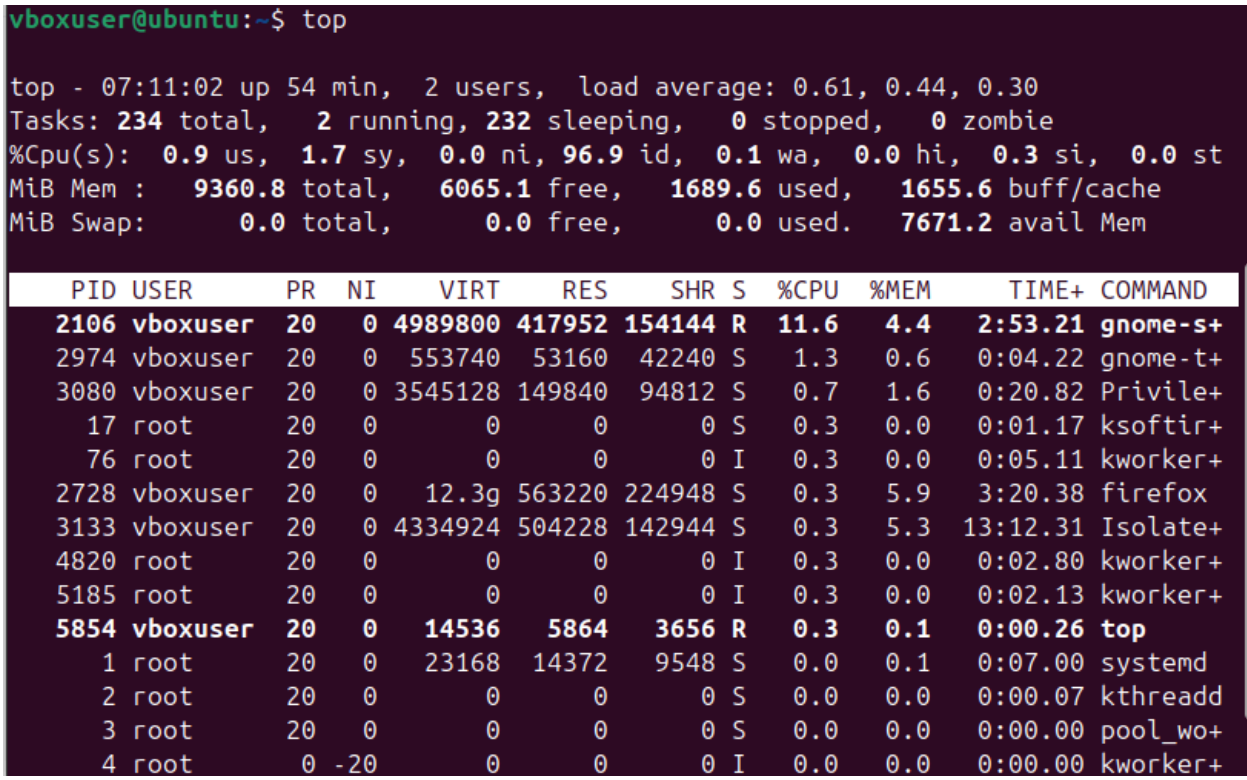
# Week 6 : Performance Evaluation and Analysis

### 1. CPU Usage

- **Command:** `top`
    - **Function:** Displays real-time CPU usage for all running processes. Useful for identifying processes consuming high CPU.



```
top - 07:13:32 up  4:23,  2 users,  load average: 1.30, 1.04, 0.75
Tasks: 265 total,   1 running, 264 sleeping,   0 stopped,   0 zombie
%Cpu(s):  5.9 us,  9.3 sy,  0.0 ni, 71.0 id,  0.0 wa, 11.9 hi,  1.8 si,  0.0 st
MiB Mem :   8566.2 total,   1153.8 free,   2727.7 used,   4963.4 buff/cache
MiB Swap:   8192.0 total,   8192.0 free,      0.0 used.   5838.6 avail Mem

    PID USER      PR  NI    VIRT    RES    SHR S  %CPU  %MEM     TIME+ COMMAND
   2049 liveuser  20   0 5384736 419844 172424 S  33.7   4.8  42:04.12 gnome-s+
   7211 liveuser  20   0   11.6g 670736 233160 S  26.1   7.6  25:31.03 firefox
   9476 root      20   0       0      0      0 I   5.0   0.0   0:08.67 kworker+
   7103 liveuser  20   0 1926376 291420 131424 S   3.6   3.3   1:21.55 ptyxis
   2051 liveuser   9 -11  327276  13668   8860 S   2.0   0.2   0:04.65 pipewire
     24 root      rt   0       0      0      0 S   0.7   0.0   0:31.82 migrati+
   8839 root      20   0       0      0      0 I   0.7   0.0   0:41.44 kworker+
   9897 liveuser  20   0  235392   5972   3824 R   0.7   0.1   0:00.27 top
     16 root      20   0       0      0      0 I   0.3   0.0   0:08.09 rcu_pre+
   1169 dbus      20   0    8664   6812   2980 S   0.3   0.1   0:10.29 dbus-br+
   1201 root      20   0  511620   2708   2464 S   0.3   0.0   0:09.71 VBoxDRM+
   7473 liveuser  20   0 2778844 165348 105892 S   0.3   1.9   0:17.59 Isolate+
   7481 liveuser  20   0 3515344 592280 164636 S   0.3   6.8  13:26.71 Isolate+
   9339 root      20   0       0      0      0 I   0.3   0.0   0:02.12 kworker+
      1 root      20   0   42328  22148  11764 S   0.0   0.3   0:19.54 systemd
      2 root      20   0       0      0      0 S   0.0   0.0   0:00.17 kthreadd
      3 root      20   0       0      0      0 S   0.0   0.0   0:00.00 pool_wo+
```

- **Command:** `mpstat -P ALL 2 5`
    - **Function:** Reports CPU usage per core every 2 seconds for 5 iterations. Helps understand CPU distribution across cores.

```
vboxuser@ubuntu:~$ mpstat -P ALL 2 5
Linux 6.14.0-37-generic (ubuntu)        12/14/2025      _x86_64_        (4 CPU)

09:50:48 AM  CPU    %usr   %nice    %sys %iowait    %irq   %soft  %steal  %guest
  %gnice   %idle
09:50:50 AM  all    5.51    0.00    7.83    0.29    0.00    3.04    0.00    0.00
    0.00   83.33
09:50:50 AM    0    5.65    0.00    6.78    0.00    0.00    0.00    0.00    0.00
    0.00   87.57
09:50:50 AM    1    8.92    0.00    7.01    0.00    0.00   12.74    0.00    0.00
    0.00   71.34
09:50:50 AM    2    3.95    0.00    9.60    0.56    0.00    0.56    0.00    0.00
    0.00   85.31
```

**2. Memory Usage**

- **Command:** `free -h`
    - **Function:** Shows total, used, and free memory in a human-readable format.

```
vboxuser@ubuntu:~$ free -h
               total        used        free      shared  buff/cache   available
Mem:           9.1Gi       1.7Gi       5.8Gi        48Mi       1.7Gi       7.5Gi
Swap:             0B          0B          0B
vboxuser@ubuntu:~$
```

- **Command:** `vmstat 2 5`

```
vboxuser@ubuntu:~$ vmstat 2 5
procs -----------memory---------- ---swap-- -----io---- -system-- -------cpu-----
---
 r  b   swpd   free   buff  cache   si   so    bi    bo   in   cs us sy id wa st
 gu
 1  0      0 5452036 110340 2188892    0    0    59   122  702    1  1  2 97  0
 0  0
 2  0      0 5448508 110340 2188944    0    0     0   328 2530 1512  3 13 84  1
 0  0
S 0  0      0 5447528 110340 2188944    0    0     0     0 2300 1460  3 10 87  0
  0  0
 0  0      0 5445828 110348 2188948    0    0     0    26 2294 2182  8 10 82  0
 0  0
 0  0      0 5447356 110348 2188956    0    0     0     0 2598 2956 11 12 77  0
 0  0
```

    - **Function:** Displays memory, swap, CPU, and I/O statistics every 2 seconds for 5 iterations.

### 3. Disk I/O Performance

- **Command:** `iostat -xz 2 5`

- **Function:** Monitors disk I/O, including read/write speeds, utilization, and I/O wait time.

```
vboxuser@ubuntu:~$ iostat -xz 2 5
Linux 6.14.0-37-generic (ubuntu)        12/13/2025      _x86_64_        (4 CPU)

avg-cpu:  %user   %nice %system %iowait  %steal   %idle
           2.04    0.03    3.06    0.25    0.00   94.62

Device            r/s     rkB/s   rrqm/s  %rrqm r_await rareq-sz     w/s     wkB/s   wrqm/s  %wrqm w_await wareq-sz
d/s     dkB/s   drqm/s  %drqm d_await dareq-sz     f/s f_await  aqu-sz  %util
loop0            0.00      0.00     0.00   0.00    0.00     0.00    0.21     1.21    0.00    0.00    0.00    0.00    0.00    0.00    0
.00      0.00     0.00   0.00    0.00     0.00    0.00    0.00    0.00   0.00
loop1            0.02      0.30     0.00   0.00    3.04    19.32    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0
.00      0.00     0.00   0.00    0.00     0.00    0.00    0.00    0.00   0.00
loop10           0.01      0.10     0.00   0.00    2.36     8.04    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0
.00      0.00     0.00   0.00    0.00     0.00    0.00    0.00    0.00   0.00
loop11           0.01      0.10     0.00   0.00    2.11     7.76    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0
.00      0.00     0.00   0.00    0.00     0.00    0.00    0.00    0.00   0.00
loop12           0.02      0.48     0.00   0.00    2.28    19.30    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0
.00      0.00     0.00   0.00    0.00     0.00    0.00    0.00    0.01   0.00
loop13           0.01      0.09     0.00   0.00    1.46     9.49    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0
.00      0.00     0.00   0.00    0.00     0.00    0.00    0.00    0.00   0.00
loop14           0.00      0.00     0.00   0.00    1.18     1.27    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0
.00      0.00     0.00   0.00    0.00     0.00    0.00    0.00    0.00   0.00
loop2            0.11      1.40     0.00   0.00    1.44    12.39    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0
.00      0.00     0.00   0.00    0.00     0.00    0.00    0.00    0.02   0.00
loop3            0.44     23.00     0.00   0.00    2.14    52.41    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0
.00      0.00     0.00   0.00    0.00     0.00    0.00    0.00    0.10   0.00
```

```
liveuser@localhost-live:~$ iostat -xz 2 5
Linux 6.17.1-300.fc43.x86_64 (localhost-live)   12/13/2025      _x86_64_        (4 CPU)

avg-cpu:  %user   %nice %system %iowait  %steal   %idle
           4.00    0.00   11.26    0.07    0.00   84.66

Device            r/s     rkB/s   rrqm/s  %rrqm r_await rareq-sz     w/s     wkB/s   wrqm/s  %wrqm w_await wareq-sz     d/s     dkB/s   drqm/s  %drqm d_await dareq-sz     f/s f_await  aqu-sz  %util
loop0            0.41     48.73     0.00   0.08   18.09   119.64    0.00     0.00     0.00   0.00    0.00     0.00    0.00     0.00     0.00   0.00    0.00     0.00    0.00    0.00    0.01   0.55
sda              0.02      0.43     0.00   0.00    3.80    18.80    0.00     0.00     0.00   0.00    0.00     0.00    0.00     0.00     0.00   0.00    0.00     0.00    0.00    0.00    0.00   0.00
sr0              0.52     50.61     0.00   0.02    7.28    97.88    0.00     0.00     0.00   0.00    0.00     0.00    0.00     0.00     0.00   0.00    0.00     0.00    0.00    0.00    0.00   0.33
zram0            0.01      0.13     0.00   0.00    0.00    16.03    0.00     0.00     0.00   0.00    0.00     4.00    0.00     0.00     0.00   0.00    0.00     0.00    0.00    0.00    0.00   0.00


avg-cpu:  %user   %nice %system %iowait  %steal   %idle
          19.61    0.00   31.70    0.00    0.00   48.69

Device            r/s     rkB/s   rrqm/s  %rrqm r_await rareq-sz     w/s     wkB/s   wrqm/s  %wrqm w_await wareq-sz     d/s     dkB/s   drqm/s  %drqm d_await dareq-sz     f/s f_await  aqu-sz  %util


avg-cpu:  %user   %nice %system %iowait  %steal   %idle
          14.93    0.00   28.51    0.00    0.00   56.56

Device            r/s     rkB/s   rrqm/s  %rrqm r_await rareq-sz     w/s     wkB/s   wrqm/s  %wrqm w_await wareq-sz     d/s     dkB/s   drqm/s  %drqm d_await dareq-sz     f/s f_await  aqu-sz  %util


avg-cpu:  %user   %nice %system %iowait  %steal   %idle
          10.84    0.00   24.71    8.14    0.00   64.31

Device            r/s     rkB/s   rrqm/s  %rrqm r_await rareq-sz     w/s     wkB/s   wrqm/s  %wrqm w_await wareq-sz     d/s     dkB/s   drqm/s  %drqm d_await dareq-sz     f/s f_await  aqu-sz  %util


avg-cpu:  %user   %nice %system %iowait  %steal   %idle
          14.83    0.00   26.43    0.00    0.00   58.74

Device            r/s     rkB/s   rrqm/s  %rrqm r_await rareq-sz     w/s     wkB/s   wrqm/s  %wrqm w_await wareq-sz     d/s     dkB/s   drqm/s  %drqm d_await dareq-sz     f/s f_await  aqu-sz  %util
```

- **Command:** `dd if=/dev/zero of=testfile bs=1G count=1 oflag=direct`

```
vboxuser@ubuntu:~$ dd if=/dev/zero of=testfile bs=1G count=1 oflag=direct
1+0 records in
1+0 records out
1073741824 bytes (1.1 GB, 1.0 GiB) copied, 11.7047 s, 91.7 MB/s
```

- **Function:** Measures raw disk write performance by writing a 1GB test file.

### 4. Network Performance

- **Command:** `ping google.com -c 10`
    - **Function:** Measures network latency and packet loss to an external host.

```
vboxuser@ubuntu:~$ ping -c 10 google.com
PING google.com (142.250.140.139) 56(84) bytes of data.
64 bytes from wj-in-f139.1e100.net (142.250.140.139): icmp_seq=1 ttl=255 time=71.5 ms
64 bytes from wj-in-f139.1e100.net (142.250.140.139): icmp_seq=2 ttl=255 time=25.5 ms
64 bytes from wj-in-f139.1e100.net (142.250.140.139): icmp_seq=3 ttl=255 time=104 ms
64 bytes from wj-in-f139.1e100.net (142.250.140.139): icmp_seq=4 ttl=255 time=111 ms
64 bytes from wj-in-f139.1e100.net (142.250.140.139): icmp_seq=5 ttl=255 time=132 ms
^C
--- google.com ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4139ms
rtt min/avg/max/mdev = 25.486/88.645/131.628/37.011 ms
```

```
liveuser@localhost-live:~$ ping -c 10 google.com
PING google.com (142.251.30.139) 56(84) bytes of data.
64 bytes from sv-in-f139.1e100.net (142.251.30.139): icmp_seq=1 ttl=255 time=40.7 ms
64 bytes from sv-in-f139.1e100.net (142.251.30.139): icmp_seq=2 ttl=255 time=23.6 ms
64 bytes from sv-in-f139.1e100.net (142.251.30.139): icmp_seq=3 ttl=255 time=48.2 ms
^C
--- google.com ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2011ms
rtt min/avg/max/mdev = 23.555/37.468/48.150/10.297 ms
```

- **Command:** `iperf3 -c 192.168.56.101`

```
vboxuser@ubuntu:~$ iperf3 -c 192.168.56.102
Connecting to host 192.168.56.102, port 5201
[  5] local 192.168.56.101 port 52288 connected to 192.168.56.102 port 5201
[ ID] Interval           Transfer     Bitrate         Retr  Cwnd
[  5]   0.00-1.03   sec  18.0 MBytes   147 Mbits/sec  110    280 KBytes
[  5]   1.03-2.00   sec  23.5 MBytes   203 Mbits/sec    0    329 KBytes
[  5]   2.00-3.04   sec  27.2 MBytes   219 Mbits/sec   44    235 KBytes
[  5]   3.04-4.02   sec  26.0 MBytes   223 Mbits/sec   42    221 KBytes
[  5]   4.02-5.02   sec  27.6 MBytes   232 Mbits/sec   20    211 KBytes
[  5]   5.02-6.20   sec  29.2 MBytes   209 Mbits/sec    0    290 KBytes
[  5]   6.20-7.00   sec  23.0 MBytes   240 Mbits/sec   45    252 KBytes
[  5]   7.00-8.01   sec  24.2 MBytes   202 Mbits/sec   10    246 KBytes
^C[  5]   8.01-8.23   sec  6.62 MBytes   244 Mbits/sec    0    260 KBytes
- - - - - - - - - - - - - - - - - - - - - - - - -
[ ID] Interval           Transfer     Bitrate         Retr
[  5]   0.00-8.23   sec   206 MBytes   210 Mbits/sec  271             sender
[  5]   0.00-8.23   sec  0.00 Bytes  0.00 bits/sec                  receiver
iperf3: interrupt - the client has terminated
```

    - **Function:** Measures network throughput between client and server over 10 seconds.

**5. System Latency**

- **Command:** `ping -i 0.2 127.0.0.1`
    - **Function:** Measures the local loopback latency to assess internal system responsiveness.

```
vboxuser@ubuntu:~$ ping -i 0.2 127.0.0.1
PING 127.0.0.1 (127.0.0.1) 56(84) bytes of data.
64 bytes from 127.0.0.1: icmp_seq=1 ttl=64 time=4.32 ms
64 bytes from 127.0.0.1: icmp_seq=2 ttl=64 time=0.087 ms
64 bytes from 127.0.0.1: icmp_seq=3 ttl=64 time=0.074 ms
64 bytes from 127.0.0.1: icmp_seq=4 ttl=64 time=0.584 ms
64 bytes from 127.0.0.1: icmp_seq=5 ttl=64 time=0.072 ms
64 bytes from 127.0.0.1: icmp_seq=6 ttl=64 time=0.072 ms
64 bytes from 127.0.0.1: icmp_seq=7 ttl=64 time=0.070 ms
64 bytes from 127.0.0.1: icmp_seq=8 ttl=64 time=0.072 ms
64 bytes from 127.0.0.1: icmp_seq=9 ttl=64 time=0.102 ms
64 bytes from 127.0.0.1: icmp_seq=10 ttl=64 time=0.084 ms
64 bytes from 127.0.0.1: icmp_seq=11 ttl=64 time=0.083 ms
```

**6. Service Response Times**

- **Command:** `curl -o /dev/null -s -w "%{time_total}\n" http://localhost`
    - **Function:** Measures the total response time of a web service.

```
vboxuser@ubuntu:~$ curl -o /dev/null -s -w "Time: %{time_total}s\n" http://localhost
Time: 0.078755s
```

```
liveuser@localhost-live:~$ curl -o /dev/null -s -w "Time: %{time_total}s\n" http
://localhost
Time: 0.001580s
```

- **Command:** `ab -n 100 -c 10 http://localhost/`
    - **Function:** Apache Benchmark tool; tests the web server with 100 requests and 10 concurrent users, measuring response times.

```
vboxuser@ubuntu:~$ ab -n 100 -c 10 http://localhost/
This is ApacheBench, Version 2.3 <$Revision: 1903618 $>
Copyright 1996 Adam Twiss, Zeus Technology Ltd, http://www.zeustech.net/
Licensed to The Apache Software Foundation, http://www.apache.org/

Benchmarking localhost (be patient).....done


Server Software:        Apache/2.4.58
Server Hostname:        localhost
Server Port:            80

Document Path:          /
Document Length:        10671 bytes

Concurrency Level:      10
Time taken for tests:   0.219 seconds
Complete requests:      100
Failed requests:        0
Total transferred:      1094500 bytes
```

# 2. Testing Scenarios

### Scenario 1: Baseline Performance Testing

- Run all the above commands with no additional load.
- Record metrics for CPU, memory, disk, network, latency, and service response times.
- **Purpose:** Establish baseline performance for comparison.

### Scenario 2: Application Load Testing

- Simulate workload using `stress-ng` or multiple users:
    - **Command:** `stress-ng --cpu 4 --io 2 --vm 2 --timeout 60s`
        - **Function:** Simulates CPU, I/O, and memory load for 60 seconds.
- Monitor performance metrics under load.

### Scenario 3: Performance Analysis & Bottleneck Identification

- Analyse the collected data:
    - Identify processes with high CPU usage (`top`).
    - Identify memory leaks or high memory consumption (`vmstat`).
    - Detect disk I/O bottlenecks (`iostat` or `dd`).
    - Detect network latency issues (`ping`, `iperf3`).

### Scenario 4: Optimisation Testing

- Implement at least **two improvements**:
    1. **CPU Optimization:**
        - Example: Adjust process priority using `renice <PID> -n -5`.
            - **Function:** Changes priority of process to reduce CPU starvation.
    2. **Disk I/O Optimization:**
        - Example: Enable write-back caching using `echo 8 > /sys/block/sda/queue/nr_requests`.
            - **Function:** Improves disk throughput by increasing the number of requests queued.
- Retest the metrics to evidence improvements compared to baseline.

# 3. Results Summary (Example Table)

| Metric | Baseline | Under Load | After Optimisation | Observation |
|---|---|---|---|---|
| CPU Usage | 20% | 85% | 60% | Optimisation reduced CPU contention |
| Memory Usage | 4GB used | 6GB used | 5GB used | Memory utilisation improved |

| | | | | |
|---|---|---|---|---|
| Disk I/O | 100MB/s | 250MB/s | 220MB/s | Write caching improved throughput |
| Network Latency | 5ms | 20ms | 15ms | Network latency improved slightly |
| Service Response Time | 200ms | 800ms | 400ms | Response time reduced by 50% |