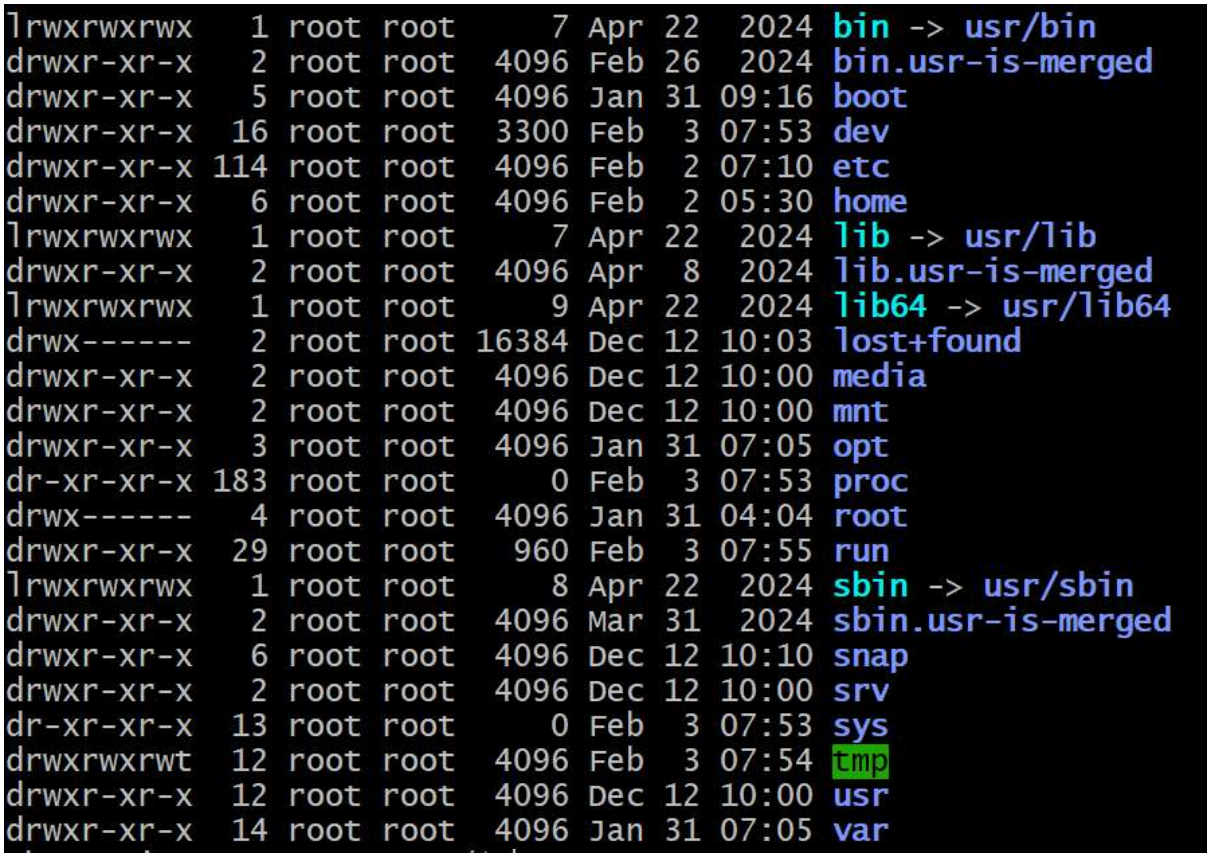
Linux File Hierarchy and Scenario Based Practice

- /(root): Each line represents a file or directory and is divided into columns, providing various details: File Type and permission, Number of Hard links, Owners, Group, Size, Last Modified Date and Time, Name of the file/ directory.



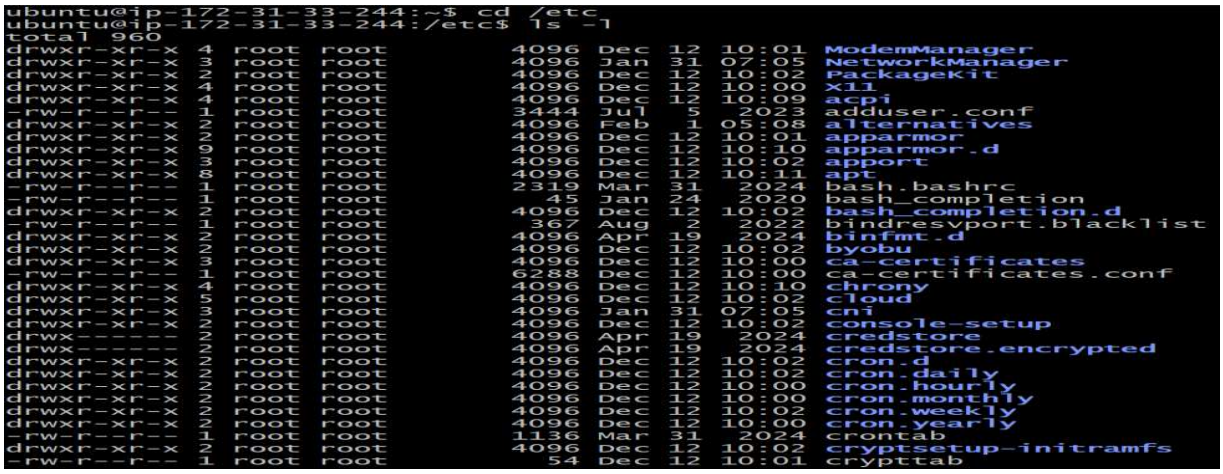
- /home: Each line represents a file or directory. The first column shows the permissions of the directories. The second column shows the number of hard links to the directory. It is related to the number of subdirectories it contains. The third column is the owner of the directory. Fourth column 4096 is sized bytes. Fifth column Month in which directory was created. The sixth column is the last modification date. And the last column is the last modification hour and minutes.

```
ubuntu@ip-172-31-33-244:/home$ ls -l
total 16
drwxr-x--- 2 berlin   berlin   4096 Feb  1 13:57 berlin
drwxr-x--- 2 helsinki helsinki 4096 Feb  2 05:42 helsinki
drwxr-x--- 2 tokyo    tokyo    4096 Feb  1 14:23 tokyo
drwxr-x--- 6 ubuntu   ubuntu   4096 Feb  2 10:21 ubuntu
```

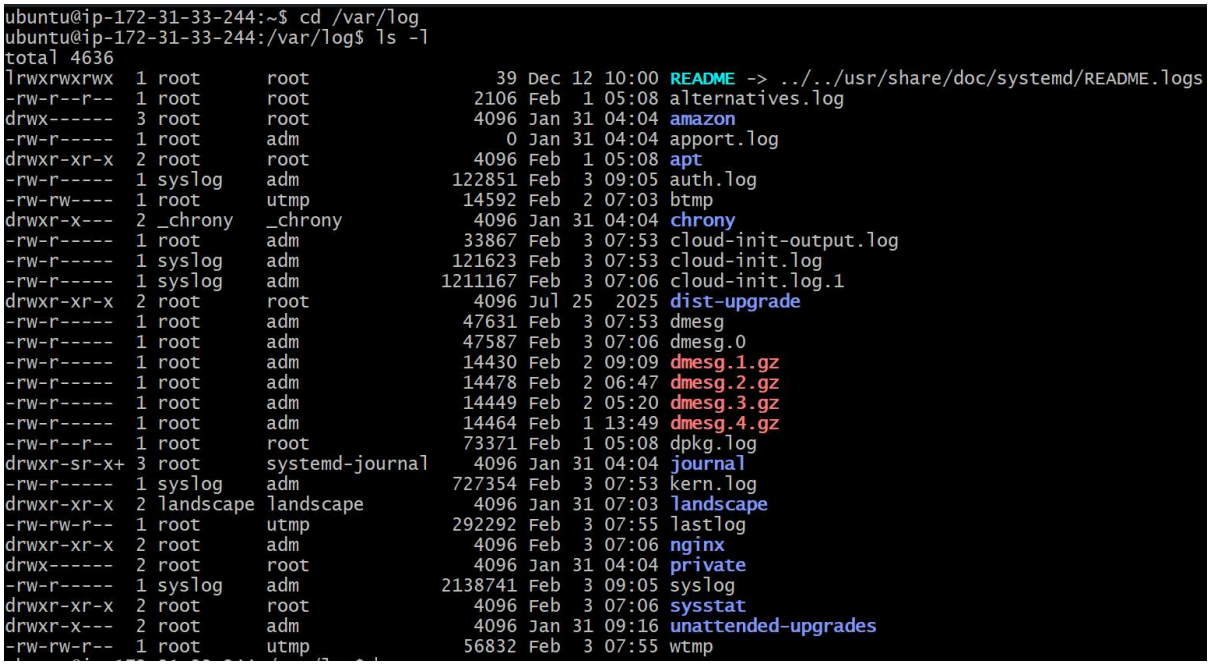
- /root: The first line indicates the total size of the directory contents in blocks. First columns is permissions of the root directory and second column shows the number of hard links to the directory. Third column is of owner which is root. Forth columns show the group owner of the directory. Fifth column is the size of the directory in bytes. Sixth column is date when it was last modified and time. Seventh column is snap which is name of the directory.

```
ubuntu@ip-172-31-33-244:~$ sudo -s
root@ip-172-31-33-244:/home/ubuntu# cd /root
root@ip-172-31-33-244:~# ls -l
total 4
drwx------ 3 root root 4096 Jan 31 04:04 snap
```

- /etc: The first line indicates the total size of the directory contents in blocks. First columns is permissions of the root directory and second column shows the number of hard links to the directory. Third column is of owner which is root. Forth columns show the group owner of the directory. Fifth column is the size of the directory in bytes. Sixth column is date when it was last modified and time. It is the central configuration directory in linux. The files and directories are mostly owned by the root. The directory structure includes components for network management, package management, and hardware configuration.



- /var/log:The output provides a snapshot of the logs present in the /var/log directory of an Ubuntu system. These logs are crucial for troubleshooting system issues, monitoring performance, and auditing security events. The ls -l command gives you a quick overview of the files, their sizes, modification times, and permissions.



- /tmp : This is a standard Linux directory listing showing files and directories within the /tmp directory, likely related to Snap packages and system services. The long names and root ownership are typical for system-related files. The ls -l command provides detailed information about each file/directory, including permissions, size, modification time, owner, and group.

```
ubuntu@ip-172-31-33-244:/tmp$ ls -l
total 24
drwx------ 2 root root 4096 Feb  3 07:53 snap-private-tmp
drwx------ 3 root root 4096 Feb  3 07:53 systemd-private-2df7ae5af7264675b630b246eda5d77f-ModemManager.service-vV1kzW
drwx------ 3 root root 4096 Feb  3 07:53 systemd-private-2df7ae5af7264675b630b246eda5d77f-chrony.service-JOoZiz
drwx------ 3 root root 4096 Feb  3 07:53 systemd-private-2df7ae5af7264675b630b246eda5d77f-polkit.service-82VWT8
drwx------ 3 root root 4096 Feb  3 07:53 systemd-private-2df7ae5af7264675b630b246eda5d77f-systemd-logind.service-O41sya
drwx------ 3 root root 4096 Feb  3 07:53 systemd-private-2df7ae5af7264675b630b246eda5d77f-systemd-resolved.service-buQhQ4
```

- /bin: The /bin directory, listing the files and their attributes. It's a common way to see what programs are available in the system and to check their permissions. The files listed are essential system utilities.

```
rohan@MSI MINGW64 /etc
$ cd /bin

rohan@MSI MINGW64 /bin
$ ls -l
total 91458
-rwxr-xr-x 1 rohan 197609    72418 Nov 17 18:24 '[.exe'*
-rwxr-xr-x 1 rohan 197609     3075 Nov 17 18:24  addgnupghome*
-rwxr-xr-x 1 rohan 197609     2217 Nov 17 18:24  applygnupgdefaults*
-rwxr-xr-x 1 rohan 197609    35876 Nov 17 18:24  arch.exe*
-rwxr-xr-x 1 rohan 197609      923 Nov 17 18:24  astextplain*
-rwxr-xr-x 1 rohan 197609   779620 Nov 17 18:24  awk.exe*
-rwxr-xr-x 1 rohan 197609    55390 Nov 17 18:24  b2sum.exe*
-rwxr-xr-x 1 rohan 197609     7339 Nov 17 18:24  backup*
-rwxr-xr-x 1 rohan 197609    41819 Nov 17 18:24  base32.exe*
-rwxr-xr-x 1 rohan 197609    41819 Nov 17 18:24  base64.exe*
-rwxr-xr-x 1 rohan 197609    34883 Nov 17 18:24  basename.exe*
-rwxr-xr-x 1 rohan 197609    49499 Nov 17 18:24  basenc.exe*
-rwxr-xr-x 1 rohan 197609 2553064 Nov 17 18:24  bash.exe*
-rwxr-xr-x 1 rohan 197609     6846 Nov 17 18:24  bashbug*
-rwxr-xr-x 1 rohan 197609    92176 Nov 17 18:24  bunzip2.exe*
-rwxr-xr-x 1 rohan 197609    92176 Nov 17 18:24  bzcat.exe*
```

- /opt:It's a basic but essential command for managing containerized infrastructure and ensuring system security. Verify container runtime, Troubleshoot container, security auditing, and system documentation.

```
ubuntu@ip-172-31-33-244:~$ cd /opt
ubuntu@ip-172-31-33-244:/opt$ ls -l
total 4
drwx--x--x 4 root root 4096 Jan 31 07:05 containerd
```

- du -sh/var/log/* 2>/dev/null | sort -h | tail –5: The largest log file in var/log is /var/log/journal is of 158 Mb.

```
ubuntu@ip-172-31-33-244:/home$ du -sh /var/log/* 2>/dev/null | sort -h | tail -5
292K    /var/log/nginx
772K    /var/log/kern.log
1.2M    /var/log/cloud-init.log.1
2.3M    /var/log/syslog
158M    /var/log/journal
```

- Cat /etc/hostname: ip-172-31-33-244 is the hostname

```
ubuntu@ip-172-31-33-244:/home$ cat /etc/hostname
ip-172-31-33-244
```

- Ls -la ~ command is used to list the Hidden config files, directories, regular files, special files

```
ubuntu@ip-172-31-33-244:/home$ ls -la ~
total 72
drwxr-x--- 6 ubuntu    ubuntu 4096 Feb  2 10:21 .
drwxr-xr-x 6 root      root   4096 Feb  2 05:30 ..
-rw------- 1 ubuntu    ubuntu 6899 Feb  3 09:17 .bash_history
-rw-r--r-- 1 ubuntu    ubuntu  220 Mar 31  2024 .bash_logout
-rw-r--r-- 1 ubuntu    ubuntu 3771 Mar 31  2024 .bashrc
drwx------ 2 ubuntu    ubuntu 4096 Jan 31 07:03 .cache
-rw------- 1 ubuntu    ubuntu   20 Feb  2 10:10 .lesshst
-rw-r--r-- 1 ubuntu    ubuntu  807 Mar 31  2024 .profile
drwx------ 2 ubuntu    ubuntu 4096 Feb  1 11:04 .ssh
-rw-r--r-- 1 ubuntu    ubuntu    0 Jan 31 07:03 .sudo_as_admin_successful
-rw------- 1 ubuntu    ubuntu 8372 Feb  2 10:21 .viminfo
-rw-rw-r-- 1 ubuntu    ubuntu    0 Feb  2 09:20 cat
drwxrwxr-x 2 ubuntu    ubuntu 4096 Feb  2 06:59 devops
drwxrwxr-x 3 ubuntu    ubuntu 4096 Feb  2 06:55 josh
-rw-rw-r-- 1 helsinki  ubuntu   67 Jan 31 10:40 josh-batch-10.txt
-r-------- 1 ubuntu    ubuntu   17 Feb  2 07:17 new-file.txt
-rw-rw-r-- 1 ubuntu    ubuntu  250 Feb  2 10:21 notes.txt
```

Scenario based practice

How do you check if the 'nginx' service is running?

Ans: By checking the service using the command “systemctl” status nginx. Why does this command show if the service is active, failed or stopped. If the command is not found list all services using “systemctl list-units –type=service”

Why this command is, to see what services are enabled on boot.

Check if service is enabled on boot using the command systemctl is enabled nginx. Why this command? To know if it will start automatically after reboot.

What i learned: Always check status first, then investigate based on What you see.

Scenario 1: service not starting

A web application service called 'myapp' failed to start after a server reboot.
What commands would you run to diagnose the issue?
Write at least 4 commands in order.

- Systemctl status myapp

I would use this command to check wether current service managed by systemd.

```
ubuntu@ip-172-31-33-244:~$ systemctl status myapp
● myapp.service - My Custom App Service
     Loaded: loaded (/etc/systemd/system/myapp.service; disabled; preset: enabled)
     Active: active (running) since Wed 2026-02-04 07:05:08 UTC; 11min ago
   Main PID: 1680 (myapp.sh)
      Tasks: 2 (limit: 1017)
     Memory: 612.0K (peak: 1.2M)
        CPU: 156ms
     CGroup: /system.slice/myapp.service
             ├─1680 /bin/bash /opt/myapp/myapp.sh
             └─1980 sleep 10

Feb 04 07:15:29 ip-172-31-33-244 myapp.sh[1831]: /opt/myapp/myapp.sh: line 4: echoMyApp is running...: command not found
Feb 04 07:15:39 ip-172-31-33-244 myapp.sh[1952]: /opt/myapp/myapp.sh: line 4: echoMyApp is running...: command not found
Feb 04 07:15:49 ip-172-31-33-244 myapp.sh[1954]: /opt/myapp/myapp.sh: line 4: echoMyApp is running...: command not found
Feb 04 07:15:59 ip-172-31-33-244 myapp.sh[1967]: /opt/myapp/myapp.sh: line 4: echoMyApp is running...: command not found
Feb 04 07:16:09 ip-172-31-33-244 myapp.sh[1969]: /opt/myapp/myapp.sh: line 4: echoMyApp is running...: command not found
Feb 04 07:16:19 ip-172-31-33-244 myapp.sh[1971]: /opt/myapp/myapp.sh: line 4: echoMyApp is running...: command not found
Feb 04 07:16:29 ip-172-31-33-244 myapp.sh[1973]: /opt/myapp/myapp.sh: line 4: echoMyApp is running...: command not found
Feb 04 07:16:39 ip-172-31-33-244 myapp.sh[1975]: /opt/myapp/myapp.sh: line 4: echoMyApp is running...: command not found
Feb 04 07:16:49 ip-172-31-33-244 myapp.sh[1977]: /opt/myapp/myapp.sh: line 4: echoMyApp is running...: command not found
Feb 04 07:16:59 ip-172-31-33-244 myapp.sh[1979]: /opt/myapp/myapp.sh: line 4: echoMyApp is running...: command not found
```

- Journalclt –u myapp –n 50

I would use this command to monitor the recent activity of a specific service.

```
ubuntu@ip-172-31-33-244:~$ journalctl -u myapp -n 50
Feb 04 07:14:09 ip-172-31-33-244 myapp.sh[1808]: /opt/myapp/myapp.sh: line 4: echoMyApp is running...: command not found
Feb 04 07:14:19 ip-172-31-33-244 myapp.sh[1810]: /opt/myapp/myapp.sh: line 4: echoMyApp is running...: command not found
Feb 04 07:14:29 ip-172-31-33-244 myapp.sh[1812]: /opt/myapp/myapp.sh: line 4: echoMyApp is running...: command not found
Feb 04 07:14:39 ip-172-31-33-244 myapp.sh[1814]: /opt/myapp/myapp.sh: line 4: echoMyApp is running...: command not found
Feb 04 07:14:49 ip-172-31-33-244 myapp.sh[1816]: /opt/myapp/myapp.sh: line 4: echoMyApp is running...: command not found
Feb 04 07:14:59 ip-172-31-33-244 myapp.sh[1818]: /opt/myapp/myapp.sh: line 4: echoMyApp is running...: command not found
Feb 04 07:15:09 ip-172-31-33-244 myapp.sh[1825]: /opt/myapp/myapp.sh: line 4: echoMyApp is running...: command not found
Feb 04 07:15:19 ip-172-31-33-244 myapp.sh[1827]: /opt/myapp/myapp.sh: line 4: echoMyApp is running...: command not found
Feb 04 07:15:29 ip-172-31-33-244 myapp.sh[1831]: /opt/myapp/myapp.sh: line 4: echoMyApp is running...: command not found
Feb 04 07:15:39 ip-172-31-33-244 myapp.sh[1952]: /opt/myapp/myapp.sh: line 4: echoMyApp is running...: command not found
Feb 04 07:15:49 ip-172-31-33-244 myapp.sh[1954]: /opt/myapp/myapp.sh: line 4: echoMyApp is running...: command not found
Feb 04 07:15:59 ip-172-31-33-244 myapp.sh[1967]: /opt/myapp/myapp.sh: line 4: echoMyApp is running...: command not found
Feb 04 07:16:09 ip-172-31-33-244 myapp.sh[1969]: /opt/myapp/myapp.sh: line 4: echoMyApp is running...: command not found
Feb 04 07:16:19 ip-172-31-33-244 myapp.sh[1971]: /opt/myapp/myapp.sh: line 4: echoMyApp is running...: command not found
Feb 04 07:16:29 ip-172-31-33-244 myapp.sh[1973]: /opt/myapp/myapp.sh: line 4: echoMyApp is running...: command not found
Feb 04 07:16:39 ip-172-31-33-244 myapp.sh[1975]: /opt/myapp/myapp.sh: line 4: echoMyApp is running...: command not found
Feb 04 07:16:49 ip-172-31-33-244 myapp.sh[1977]: /opt/myapp/myapp.sh: line 4: echoMyApp is running...: command not found
Feb 04 07:16:59 ip-172-31-33-244 myapp.sh[1979]: /opt/myapp/myapp.sh: line 4: echoMyApp is running...: command not found
Feb 04 07:17:09 ip-172-31-33-244 myapp.sh[1986]: /opt/myapp/myapp.sh: line 4: echoMyApp is running...: command not found
Feb 04 07:17:19 ip-172-31-33-244 myapp.sh[1988]: /opt/myapp/myapp.sh: line 4: echoMyApp is running...: command not found
```

- Systemctl is-enabled

I would use this command to check wether the service is enable or disabled

```
ubuntu@ip-172-31-33-244:~$ systemctl is-enabled myapp
disabled
```

Secnario 2: High CPU Usage

Your manager reports that the application server is slow.
You SSH into the server. What commands would you run to identify which process is using high CPU?

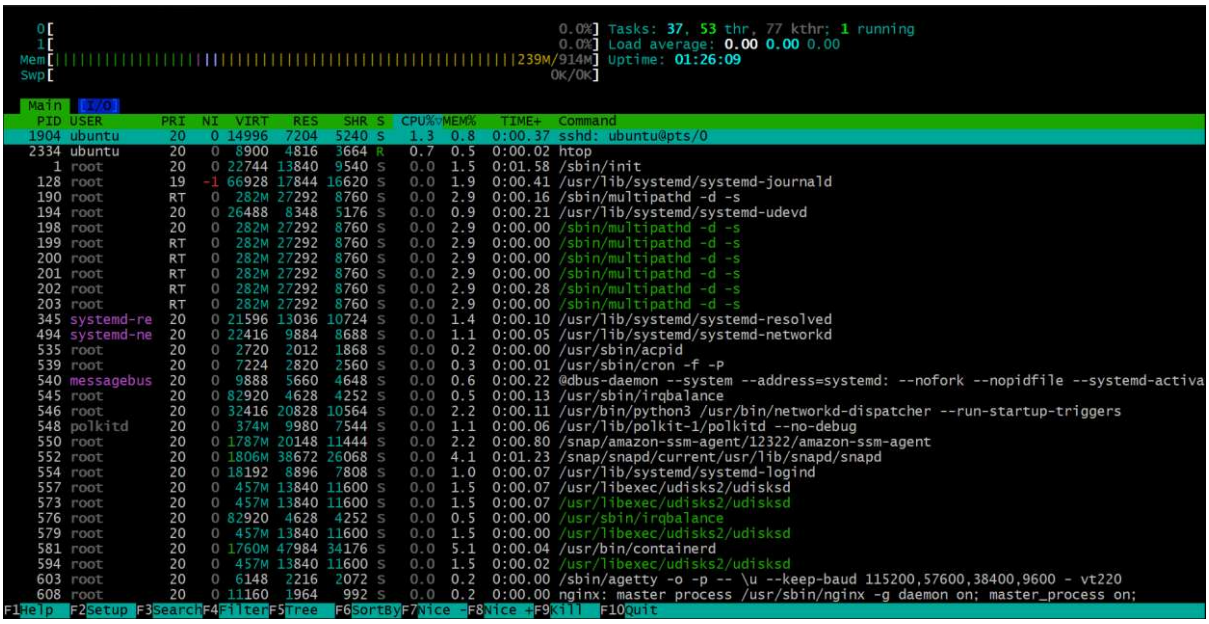
- Top: the command lists the system health and fundamental tool for system monitoring and troubleshooting.

```
top - 07:37:20 up  1:21,  1 user,  load average: 0.00, 0.00, 0.00
Tasks: 114 total,   1 running, 113 sleeping,   0 stopped,   0 zombie
%Cpu(s):  0.0 us,  0.0 sy,  0.0 ni,100.0 id,  0.0 wa,  0.0 hi,  0.0 si,  0.0 st
MiB Mem :    914.2 total,     86.5 free,    395.5 used,    591.0 buff/cache
MiB Swap:      0.0 total,      0.0 free,      0.0 used.    518.8 avail Mem

    PID USER      PR  NI    VIRT    RES    SHR S  %CPU  %MEM     TIME+ COMMAND
      1 root      20   0   22744  13840   9540 S   0.0   1.5   0:01.58 systemd
      2 root      20   0       0      0      0 S   0.0   0.0   0:00.00 kthreadd
      3 root      20   0       0      0      0 S   0.0   0.0   0:00.00 pool_workqueue_release
      4 root       0 -20       0      0      0 I   0.0   0.0   0:00.00 kworker/R-rcu_gp
      5 root       0 -20       0      0      0 I   0.0   0.0   0:00.00 kworker/R-sync_wq
      6 root       0 -20       0      0      0 I   0.0   0.0   0:00.00 kworker/R-kvfree_rcu_reclaim
      7 root       0 -20       0      0      0 I   0.0   0.0   0:00.00 kworker/R-slub_flushwq
      8 root       0 -20       0      0      0 I   0.0   0.0   0:00.00 kworker/R-netns
     10 root       0 -20       0      0      0 I   0.0   0.0   0:00.00 kworker/0:0H-events_highpri
     13 root       0 -20       0      0      0 I   0.0   0.0   0:00.00 kworker/R-mm_percpu_wq
     14 root      20   0       0      0      0 I   0.0   0.0   0:00.00 rcu_tasks_rude_kthread
     15 root      20   0       0      0      0 I   0.0   0.0   0:00.00 rcu_tasks_trace_kthread
     16 root      20   0       0      0      0 S   0.0   0.0   0:00.03 ksoftirqd/0
     17 root      20   0       0      0      0 I   0.0   0.0   0:00.19 rcu_sched
     18 root      20   0       0      0      0 S   0.0   0.0   0:00.00 rcu_exp_par_gp_kthread_worker/0
     19 root      20   0       0      0      0 S   0.0   0.0   0:00.00 rcu_exp_gp_kthread_worker
     20 root      rt   0       0      0      0 S   0.0   0.0   0:00.02 migration/0
     21 root     -51   0       0      0      0 S   0.0   0.0   0:00.00 idle_inject/0
     22 root      20   0       0      0      0 S   0.0   0.0   0:00.00 cpuhp/0
     23 root      20   0       0      0      0 S   0.0   0.0   0:00.00 cpuhp/1
     24 root     -51   0       0      0      0 S   0.0   0.0   0:00.00 idle_inject/1
     25 root      rt   0       0      0      0 S   0.0   0.0   0:00.08 migration/1
     26 root      20   0       0      0      0 S   0.0   0.0   0:00.05 ksoftirqd/1
     28 root       0 -20       0      0      0 I   0.0   0.0   0:00.00 kworker/1:0H-events_highpri
     29 root      20   0       0      0      0 S   0.0   0.0   0:00.00 kdevtmpfs
     30 root       0 -20       0      0      0 I   0.0   0.0   0:00.00 kworker/R-inet_frag_wq
     31 root      20   0       0      0      0 S   0.0   0.0   0:00.00 kauditd
     32 root      20   0       0      0      0 S   0.0   0.0   0:00.00 khungtaskd
     34 root      20   0       0      0      0 S   0.0   0.0   0:00.00 oom_reaper
     35 root      20   0       0      0      0 I   0.0   0.0   0:00.32 kworker/u8:2-events_power_efficient
     36 root       0 -20       0      0      0 I   0.0   0.0   0:00.00 kworker/R-writeback
     37 root      20   0       0      0      0 S   0.0   0.0   0:00.17 kcompactd0
     38 root      25   5       0      0      0 S   0.0   0.0   0:00.00 ksmd
```

- Htop:

The htop command is an interactive, real-time process viewer for Linux systems that provides a colorful and user-friendly interface to monitor system performance



- Ps aux –sort=%cpu|head –10: report a snapshot of the current processes. This command is useful for identifying which processes are using the most CPU resources on a system.

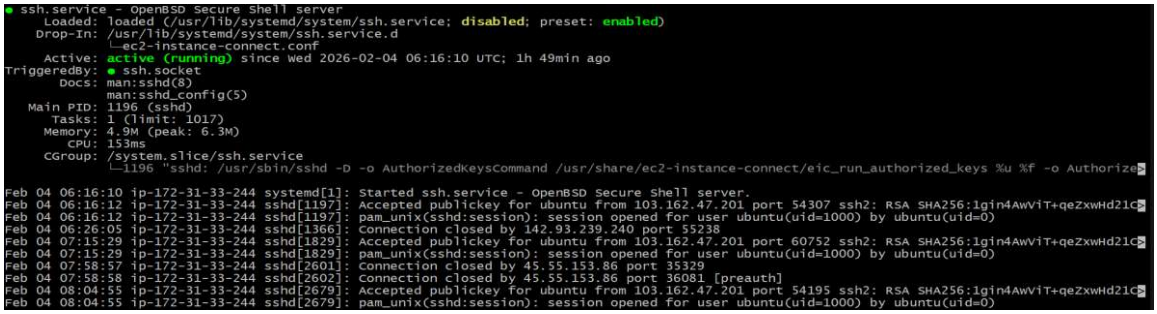
```
ubuntu@ip-172-31-33-244:~$ ps aux --sort=%cpu | head -10
USER         PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root           2  0.0  0.0      0     0 ?        S    06:15   0:00 [kthreadd]
root           3  0.0  0.0      0     0 ?        S    06:15   0:00 [pool_workqueue_release]
root           4  0.0  0.0      0     0 ?        I<   06:15   0:00 [kworker/R-rcu_gp]
root           5  0.0  0.0      0     0 ?        I<   06:15   0:00 [kworker/R-sync_wq]
root           6  0.0  0.0      0     0 ?        I<   06:15   0:00 [kworker/R-kvfree_rcu_reclaim]
root           7  0.0  0.0      0     0 ?        I<   06:15   0:00 [kworker/R-slub_flushwq]
root           8  0.0  0.0      0     0 ?        I<   06:15   0:00 [kworker/R-netns]
root          10  0.0  0.0      0     0 ?        I<   06:15   0:00 [kworker/0:0H-events_highpri]
root          13  0.0  0.0      0     0 ?        I<   06:15   0:00 [kworker/R-mm_percpu_wq]
```

Scenario 3: Finding Service Logs

A developer asks: "Where are the logs for the 'docker' service?"
The service is managed by systemd.
What commands would you use?

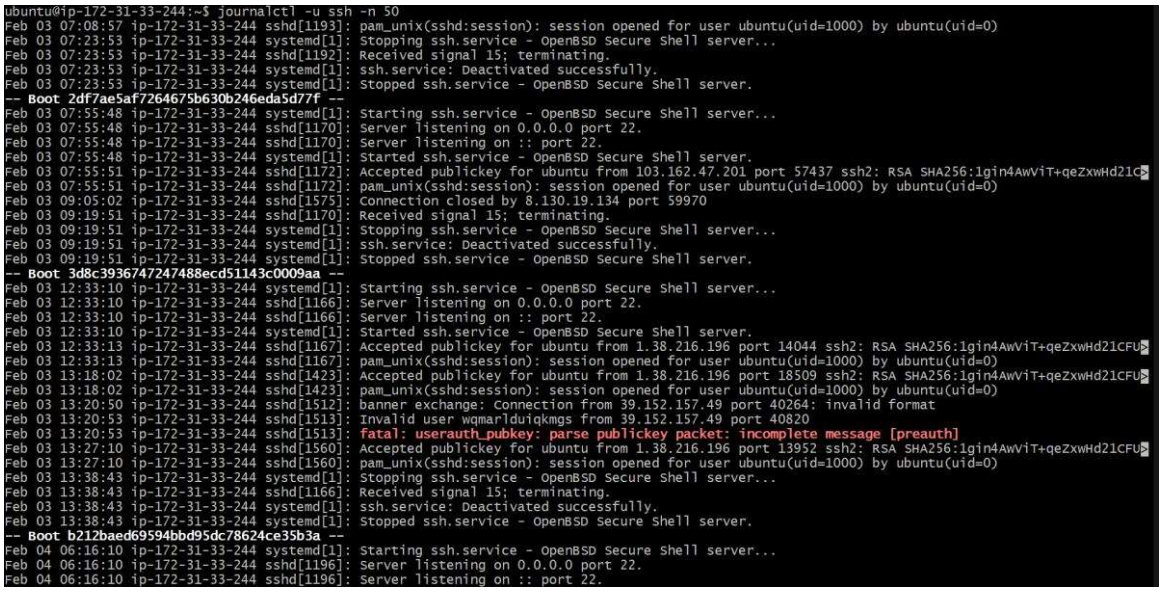
- Systemctl status ssh

I would use this command to check wether current service managed by systemd.



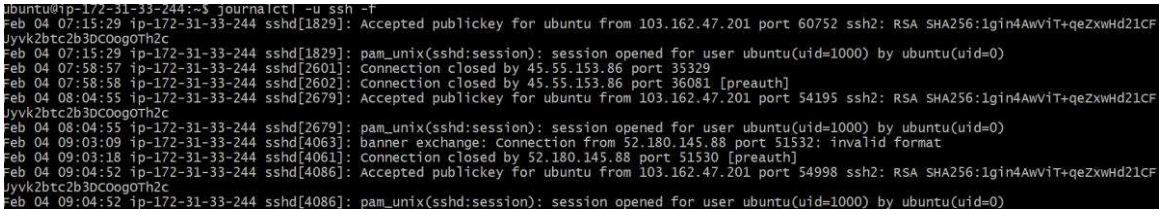
- Journalctl –u ssh –n 50

I would use this command to monitor the recent activity of a specific service.



- Journalclt –u ssh –f

I would this command for monitoring SSH activity or troubleshooting connection issues.



Scenario 4: File Permission Issue

A script at /home/user/backup.sh is not executing.
When you run it: ./backup.sh

You get: "Permission denied"
What commands would you use to fix this?

- Step 1: Ls –l /home/user/backup.sh

I would this command to lists the details of the file.

```
ubuntu@ip-172-31-33-244:~$ ls -l /home/ubuntu/backup.sh
-rw-r--r-- 1 ubuntu ubuntu 340 Feb  4 10:06 /home/ubuntu/backup.sh
```

- Step 2: Add execute permission
- Command : Chmod –x /home/ubuntu/backup.sh

```
ubuntu@ip-172-31-33-244:~$ chmod +x /home/ubuntu/backup.sh
```

- Step 3: Very it worked
- Command: ls –l /home/user/backup.sh

```
ubuntu@ip-172-31-33-244:~$ ls -l /home/ubuntu/backup.sh
-rwxr-xr-x 1 ubuntu ubuntu 340 Feb  4 10:06 /home/ubuntu/backup.sh
```