

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.

```
lrwxrwxrwx   1 root root    7 Apr 22  2024 bin -> usr/bin
drwxr-xr-x   2 root root  4096 Feb 26  2024 bin.usr-is-merged
drwxr-xr-x   5 root root  4096 Jan 31 09:16 boot
drwxr-xr-x  16 root root  3300 Feb  3 07:53 dev
drwxr-xr-x 114 root root  4096 Feb  2 07:10 etc
drwxr-xr-x   6 root root  4096 Feb  2 05:30 home
lrwxrwxrwx   1 root root    7 Apr 22  2024 lib -> usr/lib
drwxr-xr-x   2 root root  4096 Apr  8  2024 lib.usr-is-merged
lrwxrwxrwx   1 root root    9 Apr 22  2024 lib64 -> usr/lib64
drwx-----  2 root root 16384 Dec 12 10:03 lost+found
drwxr-xr-x   2 root root  4096 Dec 12 10:00 media
drwxr-xr-x   2 root root  4096 Dec 12 10:00 mnt
drwxr-xr-x   3 root root  4096 Jan 31 07:05 opt
dr-xr-xr-x 183 root root    0 Feb  3 07:53 proc
drwx-----  4 root root  4096 Jan 31 04:04 root
drwxr-xr-x  29 root root   960 Feb  3 07:55 run
lrwxrwxrwx   1 root root    8 Apr 22  2024/sbin -> usr/sbin
drwxr-xr-x   2 root root  4096 Mar 31  2024/sbin.usr-is-merged
drwxr-xr-x   6 root root  4096 Dec 12 10:10 snap
drwxr-xr-x   2 root root  4096 Dec 12 10:00 srv
dr-xr-xr-x  13 root root    0 Feb  3 07:53 sys
drwxrwxrwt  12 root root  4096 Feb  3 07:54 tmp
drwxr-xr-x  12 root root  4096 Dec 12 10:00 usr
drwxr-xr-x  14 root root  4096 Jan 31 07:05 var
```

- ◆ /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 column 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. Fourth column shows 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 column 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. Fourth column shows 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.

```

ubuntu@ip-172-31-33-244:~$ cd /etc
ubuntu@ip-172-31-33-244:/etc$ ls -l
total 960
drwxr-xr-x 4 root root 4096 Dec 12 10:01 ModemManager
drwxr-xr-x 3 root root 4096 Jan 31 07:05 NetworkManager
drwxr-xr-x 2 root root 4096 Dec 12 10:02 PackageKit
drwxr-xr-x 4 root root 4096 Dec 12 10:00 xtl
drwxr-xr-x 4 root root 4096 Dec 12 10:09 acpi
-rw-r--r-- 1 root root 3444 Jul 5 2023 adduser.conf
drwxr-xr-x 1 root root 4096 Feb 1 05:08 alternatives
drwxr-xr-x 1 root root 4096 Dec 12 10:01 apparmor
drwxr-xr-x 1 root root 4096 Dec 12 10:02 apparmor.d
drwxr-xr-x 8 root root 4096 Dec 12 10:11 apt
-rw-r--r-- 1 root root 2319 Mar 31 2024 bash.bashrc
-rw-r--r-- 1 root root 45 Jan 24 2020 bash_completion
drwxr-xr-x 1 root root 4096 Dec 12 10:02 bash_completion.d
-rw-r--r-- 1 root root 367 Aug 2 2022 bindresvport.blacklist
drwxr-xr-x 1 root root 4096 Apr 19 2024 binfmt.d
drwxr-xr-x 1 root root 4096 Dec 12 10:02 byobu
-rw-r--r-- 1 root root 6288 Dec 12 10:00 ca-certificates
drwxr-xr-x 1 root root 4096 Dec 12 10:10 ca-certificates.conf
drwxr-xr-x 1 root root 4096 Dec 12 10:02 chrony
drwxr-xr-x 1 root root 4096 Dec 12 10:02 cloud
drwxr-xr-x 1 root root 4096 Jan 31 07:05 cni
drwxr-xr-x 1 root root 4096 Dec 12 10:02 console-setup
drwxr-xr-x 1 root root 4096 Apr 19 2024 credstore
drwxr-xr-x 1 root root 4096 Apr 19 2024 credstore.encrypted
drwxr-xr-x 1 root root 4096 Dec 12 10:02 cron.d
drwxr-xr-x 1 root root 4096 Dec 12 10:02 cron.daily
drwxr-xr-x 1 root root 4096 Dec 12 10:00 cron.hourly
drwxr-xr-x 1 root root 4096 Dec 12 10:00 cron.monthly
drwxr-xr-x 1 root root 4096 Dec 12 10:02 cron.weekly
drwxr-xr-x 1 root root 4096 Dec 12 10:00 cron.yearly
-rw-r--r-- 1 root root 1136 Mar 31 2024 crontab
drwxr-xr-x 1 root root 4096 Dec 12 10:02 cryptsetup-initramfs
-rw-r--r-- 1 root root 54 Dec 12 10:01 crypttab

```

- ♦ /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.

```

ubuntu@ip-172-31-33-244:~$ cd /var/log
ubuntu@ip-172-31-33-244:/var/log$ ls -l
total 4636
lrwxrwxrwx 1 root root 39 Dec 12 10:00 README -> ../../usr/share/doc/systemd/README.logs
-rw-r--r-- 1 root root 2106 Feb 1 05:08 alternatives.log
drwx----- 3 root root 4096 Jan 31 04:04 amazon
-rw-r----- 1 root adm 0 Jan 31 04:04 apport.log
drwxr-xr-x 2 root root 4096 Feb 1 05:08 apt
-rw-r----- 1 syslog adm 122851 Feb 3 09:05 auth.log
-rw-rw---- 1 root utmp 14592 Feb 2 07:03 bttmp
drwxr-x--- 2 _chrony _chrony 4096 Jan 31 04:04 chrony
-rw-r----- 1 root adm 33867 Feb 3 07:53 cloud-init-output.log
-rw-r----- 1 syslog adm 121623 Feb 3 07:53 cloud-init.log
-rw-r----- 1 syslog adm 1211167 Feb 3 07:06 cloud-init.log.1
drwxr-xr-x 2 root root 4096 Jul 25 2025 dist-upgrade
-rw-r----- 1 root adm 47631 Feb 3 07:53 dmesg
-rw-r----- 1 root adm 47587 Feb 3 07:06 dmesg.0
-rw-r----- 1 root adm 14430 Feb 2 09:09 dmesg.1.gz
-rw-r----- 1 root adm 14478 Feb 2 06:47 dmesg.2.gz
-rw-r----- 1 root adm 14449 Feb 2 05:20 dmesg.3.gz
-rw-r----- 1 root adm 14464 Feb 1 13:49 dmesg.4.gz
-rw-r--r-- 1 root root 73371 Feb 1 05:08 dpkg.log
drwxr-sr-x+ 3 root systemd-journal 4096 Jan 31 04:04 journal
-rw-r----- 1 syslog adm 727354 Feb 3 07:53 kern.log
drwxr-xr-x 2 landscape landscape 4096 Jan 31 07:03 landscape
-rw-rw-r-- 1 root utmp 292292 Feb 3 07:55 lastlog
drwxr-xr-x 2 root adm 4096 Feb 3 07:06 nginx
drwx----- 2 root root 4096 Jan 31 04:04 private
-rw-r----- 1 syslog adm 2138741 Feb 3 09:05 syslog
drwxr-xr-x 2 root root 4096 Feb 3 07:06 sysstat
drwxr-x--- 2 root adm 4096 Jan 31 09:16 unattended-upgrades
-rw-rw-r-- 1 root utmp 56832 Feb 3 07:55 wtmp

```

- ♦ /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-j0oziz
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-041sya
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 bzipcat.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 .lessht
-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
-rw-rw-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 whether 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

```


- ◆ Journalctl -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 whether the service is enabled or disabled

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

Scenario 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-slab_flushwq
  8 root         0 -20    0         0         0  I   0.0   0.0   0:00.00 kworker/R-nets
 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 ksm
```

- ◆ 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

```

0[ 0.0%] Tasks: 37, 53 thr, 77 kthr; 1 running
1[ 0.0%] Load average: 0.00 0.00 0.00
Mem[ 239M/914M] Uptime: 01:26:09
Swp[ 0K/0K]

Main 1/20
PID USER PRI NI VIRT RES SHR S CPU% MEM% TIME+ Command
1904 ubuntu 20 0 14996 7204 5240 S 1.3 0.8 0:00.37 sshd: ubuntu@pts/0
2334 ubuntu 20 0 8900 4816 3664 R 0.7 0.5 0:00.02 htop
1 root 20 0 22744 13840 9540 S 0.0 1.5 0:01.58 /sbin/init
128 root 19 -1 66928 17844 16620 S 0.0 1.9 0:00.41 /usr/lib/systemd/systemd-journald
190 root RT 0 282M 27292 8760 S 0.0 2.9 0:00.16 /sbin/multipathd -d -s
194 root 20 0 26488 8348 5176 S 0.0 0.9 0:00.21 /usr/lib/systemd/systemd-udev
198 root 20 0 282M 27292 8760 S 0.0 2.9 0:00.00 /sbin/multipathd -d -s
199 root RT 0 282M 27292 8760 S 0.0 2.9 0:00.00 /sbin/multipathd -d -s
200 root RT 0 282M 27292 8760 S 0.0 2.9 0:00.00 /sbin/multipathd -d -s
201 root RT 0 282M 27292 8760 S 0.0 2.9 0:00.00 /sbin/multipathd -d -s
202 root RT 0 282M 27292 8760 S 0.0 2.9 0:00.28 /sbin/multipathd -d -s
203 root RT 0 282M 27292 8760 S 0.0 2.9 0:00.00 /sbin/multipathd -d -s
345 systemd-re 20 0 21596 13036 10724 S 0.0 1.4 0:00.10 /usr/lib/systemd/systemd-resolved
494 systemd-ne 20 0 22416 9884 8688 S 0.0 1.1 0:00.05 /usr/lib/systemd/systemd-networkd
535 root 20 0 2720 2012 1868 S 0.0 0.2 0:00.00 /usr/sbin/acpid
539 root 20 0 7224 2820 2560 S 0.0 0.3 0:00.01 /usr/sbin/cron -f -P
540 messagebus 20 0 9888 5660 4648 S 0.0 0.6 0:00.22 @dbus-daemon --system --address=systemd: --nofork --nopidfile --systemd-activa
545 root 20 0 82920 4628 4252 S 0.0 0.5 0:00.13 /usr/sbin/irqbalance
546 root 20 0 32416 20828 10564 S 0.0 2.2 0:00.11 /usr/bin/python3 /usr/bin/networkd-dispatcher --run-startup-triggers
548 polkitd 20 0 374M 9980 7544 S 0.0 1.1 0:00.06 /usr/lib/polkit-1/polkitd --no-debug
550 root 20 0 1787M 20148 11444 S 0.0 2.2 0:00.80 /snap/amazon-ssm-agent/12322/amazon-ssm-agent
552 root 20 0 1806M 38672 26068 S 0.0 4.1 0:01.23 /snap/snapd/current/usr/lib/snapd/snapd
554 root 20 0 18192 8896 7808 S 0.0 1.0 0:00.07 /usr/lib/systemd/systemd-logind
557 root 20 0 457M 13840 11600 S 0.0 1.5 0:00.07 /usr/libexec/udisks2/udisksd
573 root 20 0 457M 13840 11600 S 0.0 1.5 0:00.07 /usr/libexec/udisks2/udisksd
576 root 20 0 82920 4628 4252 S 0.0 0.5 0:00.00 /usr/sbin/irqbalance
579 root 20 0 457M 13840 11600 S 0.0 1.5 0:00.00 /usr/libexec/udisks2/udisksd
581 root 20 0 1760M 47984 34176 S 0.0 5.1 0:00.04 /usr/bin/containerd
594 root 20 0 457M 13840 11600 S 0.0 1.5 0:00.02 /usr/libexec/udisks2/udisksd
603 root 20 0 6148 2216 2072 S 0.0 0.2 0:00.00 /sbin/agetty -o -p -- \u --keep-baud 115200,57600,38400,9600 - vt220
608 root 20 0 11160 1964 992 S 0.0 0.2 0:00.00 nginx: master process /usr/sbin/nginx -g daemon on; master_process on;
F1Help F2Setup F3Search F4Filter F5Tree F6SortBy F7Nice F8Nice F9Kill F10Quit

```

- ◆ `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 whether current service managed by systemd.


```

ssh.service - openBSD Secure Shell server
Loaded: loaded (/usr/lib/systemd/system/ssh.service; disabled; preset: enabled)
Drop-In: /usr/lib/systemd/system/ssh.service.d
└─ec2-instance-connect.conf
Active: active (running) since wed 2026-02-04 06:16:10 UTC; 1h 49min ago
TriggeredBy: ● ssh.socket
Docs: man:ssh(8)
       man:ssh_config(5)
Main PID: 1196 (sshd)
Tasks: 1 (limit: 1017)
Memory: 4.9M (peak: 6.3M)
CPU: 153ms
CGroup: /system.slice/ssh.service
└─1196 "sshd: /usr/sbin/sshd -D -o AuthorizedKeysCommand=/usr/share/ec2-instance-connect/eic_run_authorized_keys %u %f -o Authorize
Feb 04 06:16:10 ip-172-31-33-244 systemd[1]: Started ssh.service - OpenBSD Secure Shell server.
Feb 04 06:16:12 ip-172-31-33-244 sshd[1197]: Accepted publickey for ubuntu from 103.162.47.201 port 54307 ssh2: RSA SHA256:1gin4AwViT+qeZxwHd21C
Feb 04 06:16:12 ip-172-31-33-244 sshd[1197]: pam_unix(sshd:session): session opened for user ubuntu(uid=1000) by ubuntu(uid=0)
Feb 04 06:26:05 ip-172-31-33-244 sshd[1366]: Connection closed by 142.93.239.240 port 55238
Feb 04 07:15:29 ip-172-31-33-244 sshd[1829]: Accepted publickey for ubuntu from 103.162.47.201 port 60752 ssh2: RSA SHA256:1gin4AwViT+qeZxwHd21C
Feb 04 07:15:29 ip-172-31-33-244 sshd[1829]: pam_unix(sshd:session): session opened for user ubuntu(uid=1000) by ubuntu(uid=0)
Feb 04 07:58:57 ip-172-31-33-244 sshd[2601]: Connection closed by 45.55.153.86 port 35329
Feb 04 07:58:58 ip-172-31-33-244 sshd[2602]: Connection closed by 45.55.153.86 port 36081 [preauth]
Feb 04 08:04:55 ip-172-31-33-244 sshd[2679]: Accepted publickey for ubuntu from 103.162.47.201 port 54195 ssh2: RSA SHA256:1gin4AwViT+qeZxwHd21C
Feb 04 08:04:55 ip-172-31-33-244 sshd[2679]: pam_unix(sshd:session): session opened for user ubuntu(uid=1000) by ubuntu(uid=0)

```

◆ Journalctl -u ssh -n 50

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

```

ubuntu@ip-172-31-33-244:~$ journalctl -u ssh -n 50
Feb 03 07:08:57 ip-172-31-33-244 sshd[1193]: pam_unix(sshd:session): session opened for user ubuntu(uid=1000) by ubuntu(uid=0)
Feb 03 07:23:53 ip-172-31-33-244 systemd[1]: Stopping ssh.service - openBSD Secure Shell server...
Feb 03 07:23:53 ip-172-31-33-244 sshd[1192]: Received signal 15; terminating.
Feb 03 07:23:53 ip-172-31-33-244 systemd[1]: ssh.service: Deactivated successfully.
Feb 03 07:23:53 ip-172-31-33-244 systemd[1]: Stopped ssh.service - openBSD Secure Shell server.
-- Boot 2df7ae5af7264675b630b246eda5d77f --
Feb 03 07:55:48 ip-172-31-33-244 systemd[1]: Starting ssh.service - OpenBSD Secure Shell server...
Feb 03 07:55:48 ip-172-31-33-244 sshd[1170]: Server listening on 0.0.0.0 port 22.
Feb 03 07:55:48 ip-172-31-33-244 sshd[1170]: Server listening on :: port 22.
Feb 03 07:55:48 ip-172-31-33-244 systemd[1]: Started ssh.service - OpenBSD Secure Shell server.
Feb 03 07:55:51 ip-172-31-33-244 sshd[1172]: Accepted publickey for ubuntu from 103.162.47.201 port 57437 ssh2: RSA SHA256:1gin4AwViT+qeZxwHd21C
Feb 03 07:55:51 ip-172-31-33-244 sshd[1172]: pam_unix(sshd:session): session opened for user ubuntu(uid=1000) by ubuntu(uid=0)
Feb 03 09:05:02 ip-172-31-33-244 sshd[1575]: Connection closed by 8.130.19.134 port 59970
Feb 03 09:19:51 ip-172-31-33-244 sshd[1170]: Received signal 15; terminating.
Feb 03 09:19:51 ip-172-31-33-244 systemd[1]: Stopping ssh.service - OpenBSD Secure Shell server...
Feb 03 09:19:51 ip-172-31-33-244 systemd[1]: ssh.service: Deactivated successfully.
Feb 03 09:19:51 ip-172-31-33-244 systemd[1]: Stopped ssh.service - OpenBSD Secure Shell server.
-- Boot 3d8c3936747247488ecd51143c0009aa --
Feb 03 12:33:10 ip-172-31-33-244 systemd[1]: Starting ssh.service - OpenBSD Secure Shell server...
Feb 03 12:33:10 ip-172-31-33-244 sshd[1166]: Server listening on 0.0.0.0 port 22.
Feb 03 12:33:10 ip-172-31-33-244 sshd[1166]: Server listening on :: port 22.
Feb 03 12:33:10 ip-172-31-33-244 systemd[1]: Started ssh.service - OpenBSD Secure Shell server.
Feb 03 12:33:13 ip-172-31-33-244 sshd[1167]: Accepted publickey for ubuntu from 1.38.216.196 port 14044 ssh2: RSA SHA256:1gin4AwViT+qeZxwHd21C
Feb 03 12:33:13 ip-172-31-33-244 sshd[1167]: pam_unix(sshd:session): session opened for user ubuntu(uid=1000) by ubuntu(uid=0)
Feb 03 13:18:02 ip-172-31-33-244 sshd[1423]: Accepted publickey for ubuntu from 1.38.216.196 port 18509 ssh2: RSA SHA256:1gin4AwViT+qeZxwHd21C
Feb 03 13:18:02 ip-172-31-33-244 sshd[1423]: pam_unix(sshd:session): session opened for user ubuntu(uid=1000) by ubuntu(uid=0)
Feb 03 13:20:50 ip-172-31-33-244 sshd[1512]: banner exchange: Connection from 39.152.157.49 port 40264: invalid format
Feb 03 13:20:53 ip-172-31-33-244 sshd[1513]: Invalid user wqmarlduiqkms from 39.152.157.49 port 40820
Feb 03 13:20:53 ip-172-31-33-244 sshd[1513]: fatal: userauth_pubkey: parse publickey packet: incomplete message [preauth]
Feb 03 13:27:10 ip-172-31-33-244 sshd[1560]: Accepted publickey for ubuntu from 1.38.216.196 port 13952 ssh2: RSA SHA256:1gin4AwViT+qeZxwHd21C
Feb 03 13:27:10 ip-172-31-33-244 sshd[1560]: pam_unix(sshd:session): session opened for user ubuntu(uid=1000) by ubuntu(uid=0)
Feb 03 13:38:43 ip-172-31-33-244 systemd[1]: Stopping ssh.service - OpenBSD Secure Shell server...
Feb 03 13:38:43 ip-172-31-33-244 sshd[1166]: Received signal 15; terminating.
Feb 03 13:38:43 ip-172-31-33-244 systemd[1]: ssh.service: Deactivated successfully.
Feb 03 13:38:43 ip-172-31-33-244 systemd[1]: Stopped ssh.service - OpenBSD Secure Shell server.
-- Boot b212baed69594bbd95dc78624ce35b3a --
Feb 04 06:16:10 ip-172-31-33-244 systemd[1]: Starting ssh.service - OpenBSD Secure Shell server...
Feb 04 06:16:10 ip-172-31-33-244 sshd[1196]: Server listening on 0.0.0.0 port 22.
Feb 04 06:16:10 ip-172-31-33-244 sshd[1196]: Server listening on :: port 22.

```

◆ Journalctl -u ssh -f

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

```

ubuntu@ip-172-31-33-244:~$ journalctl -u ssh -f
Feb 04 07:15:29 ip-172-31-33-244 sshd[1829]: Accepted publickey for ubuntu from 103.162.47.201 port 60752 ssh2: RSA SHA256:1gin4AwViT+qeZxwHd21C
Feb 04 07:15:29 ip-172-31-33-244 sshd[1829]: pam_unix(sshd:session): session opened for user ubuntu(uid=1000) by ubuntu(uid=0)
Feb 04 07:58:57 ip-172-31-33-244 sshd[2601]: Connection closed by 45.55.153.86 port 35329
Feb 04 07:58:58 ip-172-31-33-244 sshd[2602]: Connection closed by 45.55.153.86 port 36081 [preauth]
Feb 04 08:04:55 ip-172-31-33-244 sshd[2679]: Accepted publickey for ubuntu from 103.162.47.201 port 54195 ssh2: RSA SHA256:1gin4AwViT+qeZxwHd21C
Feb 04 08:04:55 ip-172-31-33-244 sshd[2679]: pam_unix(sshd:session): session opened for user ubuntu(uid=1000) by ubuntu(uid=0)
Feb 04 09:03:09 ip-172-31-33-244 sshd[4063]: banner exchange: Connection from 52.180.145.88 port 51532: invalid format
Feb 04 09:03:18 ip-172-31-33-244 sshd[4061]: Connection closed by 52.180.145.88 port 51530 [preauth]
Feb 04 09:04:52 ip-172-31-33-244 sshd[4086]: Accepted publickey for ubuntu from 103.162.47.201 port 54998 ssh2: RSA SHA256:1gin4AwViT+qeZxwHd21C
Feb 04 09:04:52 ip-172-31-33-244 sshd[4086]: pam_unix(sshd:session): session opened for user ubuntu(uid=1000) by ubuntu(uid=0)

```

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
```