

Day 7/90 – Linux File System Hierarchy & Scenario-Based Practice.

Today's goal is to understand where things live in Linux and practice troubleshooting like a DevOps engineer.

Part 1: Linux File System Hierarchy.

Document the purpose of these essential directories:

Core Directories (Must Know):

/ (root) - The starting point of everything

/home - User home directories

/root - Root user's home directory

/etc - Configuration files

/var/log - Log files (very important for DevOps!)

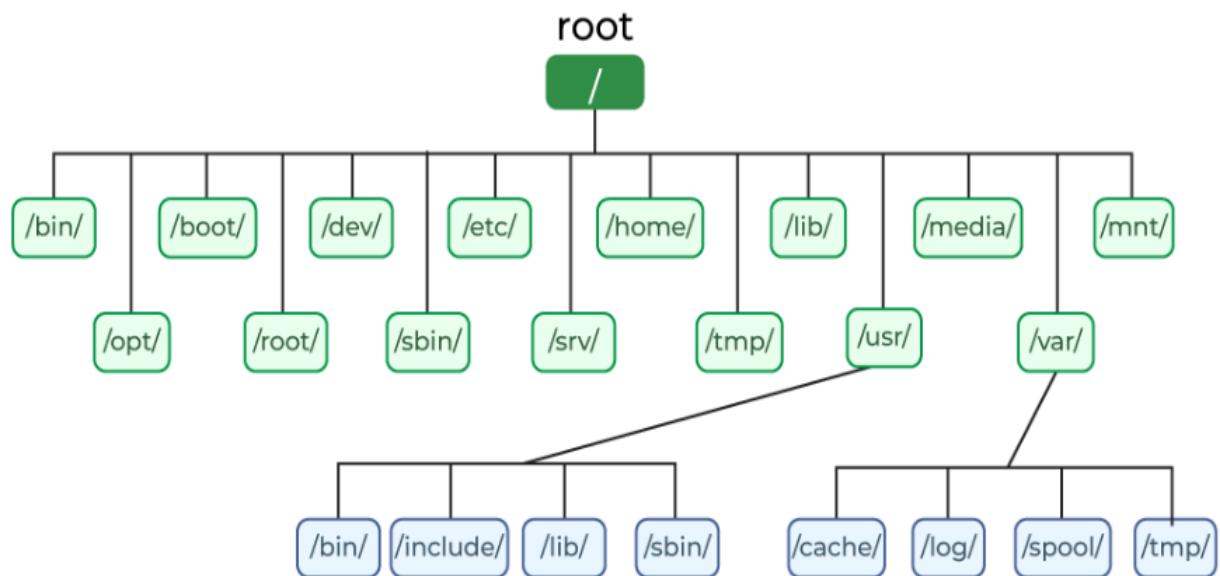
/tmp - Temporary files

Additional Directories (Good to Know):

/bin - Essential command binaries

/usr/bin - User command binaries

/opt - Optional/third-party applications.



```
sri@LAPTOP-KH3CPCFO: ~  
sri@LAPTOP-KH3CPCFO:~$ du -sh /var/log/* 2>/dev/null | sort -h | tail -5  
832K /var/log/kern.log  
1.8M /var/log/syslog.1  
1.9M /var/log/syslog  
7.6M /var/log/installer  
801M /var/log/journal  
sri@LAPTOP-KH3CPCFO:~$ cat /etc/hostname  
LAPTOP-KH3CPCFO  
sri@LAPTOP-KH3CPCFO:~$ ls -la ~  
total 56888  
drwxr-x--- 31 sri sri 4096 Feb 14 15:44 .  
drwxr-xr-x 5 root root 4096 Feb 12 18:49 ..  
drwxr-xr-x 4 sri sri 4096 Jan 27 15:51 .ansible  
lrwxrwxrwx 1 sri sri 31 Sep 1 2024 .aws -> /mnt/c/Users/VERRA_VL  
/mnt/c/.aws  
lrwxrwxrwx 1 sri sri 33 Sep 1 2024 .azure -> /mnt/c/Users/VERRA_VL  
/mnt/c/.azure  
-rw----- 1 sri sri 23171 Feb 14 15:48 .bash_history  
-rw-r--r-- 1 sri sri 220 Aug 2 2024 .bash_logout  
-rw-r--r-- 1 sri sri 3771 Aug 2 2024 .bashrc  
drwx----- 4 sri sri 4096 Jul 7 2025 .cache  
drwx----- 4 sri sri 4096 Feb 11 23:56 .config  
drwxr-xr-x 5 sri sri 4096 Sep 1 2024 .docker  
drwxr-xr-x 3 sri sri 4096 Jul 7 2025 .dotnet  
-rw----- 1 sri sri 12288 Feb 17 2025 .firstfile.txt.swo  
drwxr-xr-x 8 sri sri 4096 Sep 4 2024 .git  
-rw-r--r-- 1 sri sri 59 Aug 26 2024 .gitconfig  
-rw-r--r-- 1 sri sri 20 Feb 14 15:44 .lessht  
drwxr-xr-x 4 sri sri 4096 Sep 3 2024 .local  
-rw-r--r-- 1 sri sri 0 Feb 14 15:32 .motd_shown  
-rw-r--r-- 1 sri sri 87 Aug 12 2024 .mysql_history  
drwxr-xr-x 4 sri sri 4096 Jul 8 2025 .npm  
-rw-r--r-- 1 sri sri 807 Aug 2 2024 .profile  
drwx----- 3 sri sri 4096 Jan 27 15:35 .ssh  
-rw-r--r-- 1 sri sri 0 Aug 12 2024 .sudo_as_admin_successful  
drwxr-xr-x 2 sri sri 4096 Jan 31 18:28 .terraform.d  
drwxr-xr-x 7 sri sri 4096 Feb 18 2025 .vagrant.d  
-rw----- 1 sri sri 6109 Jan 27 13:27 .viminfo  
drwxr-xr-x 5 sri sri 4096 Jul 7 2025 .vscode-server  
-rw-r--r-- 1 sri sri 234 Jan 31 17:59 .wget-hsts  
-rw-r--r-- 1 sri sri 1933 Feb 13 00:12 .wsl-config
```

Part 2: Scenario-Based Practice.

Solved scenario: understanding how to approach scenarios.

Example scenario : Check if a service is running.

Question: How do you check if the nginx service is running ?

1. Check service status : `systemctl status nginx`

Why this command? To see what service is active, failed, or stopped.

2. If service is not found, list all service : `systemctl list-units --type=service`

Why this command? To see what service exist on the system.

3. check if service is enabled on boot : `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.

1. `systemctl status myapp` (Why: Check if active, failed, or stopped)
2. `journalctl -u myapp -n 50` (Why: Read recent logs to find error messages)
3. `systemctl is-enabled myapp` (Why: Verify if it starts on boot.)
4. `systemctl list-units --type=service` (Why: Confirm service exists and is recognized)

- 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?

1. Use a command that shows live CPU usage : `top`.
2. Look for processes sorted by CPU percentage : `ps aux --sort=-%cpu | head -10`
3. Easier interactive monitoring : `htop`.
4. Note the PID (process ID) of the top process.

- Scenario 3 : Finding Docker service Logs.

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

1. `systemctl status docker` (why: confirm service state)
2. `journalctl -u docker -n 50` (why: view last 50 log lines)
3. `journalctl -u docker -f` (why: follow logs in real-time)

- Scenario 4 : Permission Denied Script

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?

1. Check current permissions : `ls -l /home/user/backup.sh` (Look for `-rw-r--r--`—notice no 'x' = not executable).
2. check for execute permission. `chmod +x /home/user/backup.sh` :
3. verify it worked : `ls -l /home/user/backup.sh` (Look for : `-rwx-r-x-r-x`)
4. Try running it : `./backup.sh`