# File & directory commands

ls -l, du -sh, df -h, find / -name <file>

# User and permissions

useradd user1

passwd user1

chmod 755 file

chown root:root file

# System info

hostnamectl

uname -r

cat /etc/redhat-release

free -m

uptime

# Package management

yum repolist

yum install -y httpd

systemctl start httpd

systemctl enable httpd

how to namage user, service permission

tail -f /var/log/messages

journalctl –xe

Focus: Connectivity, services, and monitoring

# Networking

ip addr

ping 8.8.8.8

nslookup google.com

netstat -tulnp

ss -lntp

# Disk & process

df -h

ps aux | grep docker

top / htop

# Firewall & SELinux

systemctl status firewalld

setenforce 0 # temporary disable SELinux

getenforce

Practice short real-world telecom cases:

| **Scenario** | **Command** |
| --- | --- |
| Service not starting | systemctl status, journalctl -u <service> |
| Disk full | df -h, `du -sh /\* |
| Check system performance | top, vmstat 2, iostat |
| Schedule task | crontab -e |

DAY 2 – Docker Essentials for Telecom

Focus: Core Docker commands and concepts

# Check Docker service

systemctl status docker

docker version

docker info

# Pull and run images

docker pull centos:7

docker run -it --name test centos:7 /bin/bash

docker ps -a

# Manage containers

docker stop test

docker start test

docker exec -it test /bin/bash

docker logs test

docker inspect test

# Manage images

docker images

docker tag centos:7 mycentos:latest

docker rmi centos:7

afternoon

**Dockerfile + Troubleshooting**

**🔹 Create and test a simple Dockerfile**

FROM centos:7

RUN yum install -y httpd

EXPOSE 80

CMD ["/usr/sbin/httpd", "-D", "FOREGROUND"]

Build and run

docker build -t myweb:1.0 .

docker run -d -p 8080:80 myweb:1.0

docker ps

check and run

ccurl <http://localhost:8080>

Docker Scenarios for Interview

Practice these **telecom-relevant real cases** 👇

| **Scenario** | **Command or Action** |
| --- | --- |
| Container not starting | docker logs <id> |
| Remove unused images | docker system prune –a |
| Check container network | docker network ls, docker inspect |
| Run privileged container (for systemd) | docker run --privileged -d centos:7 /sbin/init |
| Copy files into container | docker cp file.txt container:/tmp/ |

### ****Bonus: Interview Q&A (Review These)****

1. What is the difference between **image** and **container**?
2. How do you check container logs?
3. What are Docker volumes and why are they used?
4. What happens when you use CMD vs ENTRYPOINT in Dockerfile?
5. How do you troubleshoot a failed Docker container?
6. How does Docker isolate processes in Linux?
7. What are namespaces and cgroups?
8. What command is used to check Docker resource usage? (docker stats)

## Final Tip (Night Before Interview)

* Revise 15 most-used Linux commands.
* Be confident explaining what each command does.
* Have 1–2 examples ready:

“In my lab, I created a CentOS container with Apache and tested it using curl.”  
(Shows practical experience!)

### ****Linux Admin****

* User and permission management (useradd, usermod, chmod, chown)
* Package management (yum, rpm)
* Service management (systemctl, service)
* Logs and monitoring (journalctl, /var/log, top, vmstat, iostat)
* Network commands (ip a, ping, netstat, ss, traceroute)
* Disk and filesystem (df -h, du -sh, mount, lsblk)
* Cron jobs and automation (crontab -e)
* SELinux and firewall basics (getenforce, firewalld)

### ****Docker****

* Docker architecture: Daemon, CLI, Registry, Containers, Images
* Commands: docker ps, docker images, docker run, docker exec, docker logs, docker inspect
* Dockerfile instructions: FROM, RUN, COPY, CMD, ENTRYPOINT, EXPOSE
* Volumes and bind mounts
* Container networking: bridge, host, overlay
* Container troubleshooting and logs
* Image management: docker tag, docker rmi, docker pull/push
* Systemd in containers (optional, if relevant)

### ****Scripting & Automation****

* Shell scripting basics (bash, for, if, while, awk, sed)
* Simple automation examples for Linux or Docker tasks

**Linux Admin Questions**

1. **Q:** How do you check disk usage on a Linux server?  
   **A:** Use df -h for filesystem usage and du -sh /path for folder sizes.
2. **Q:** How to check if a service is running and start it if not?  
   **A:**
3. systemctl status httpd
4. systemctl start httpd
5. systemctl enable httpd
6. **Q:** How to monitor CPU and memory usage?  
   **A:** top, htop, vmstat 2, free -m
7. **Q:** How do you schedule a backup job?  
   **A:** Use crontab -e, example:
8. 0 2 \* \* \* tar -czf /backup/home\_$(date +\%F).tar.gz /home

**Docker Questions**

1. **Q:** Difference between Docker image and container?  
   **A:** Image is a **read-only template**, container is a **running instance** of that image.
2. **Q:** How to see logs of a running container?  
   **A:** docker logs <container\_name\_or\_id>
3. **Q:** How do you run a container in detached mode?  
   **A:**
4. docker run -d -p 8080:80 nginx
5. **Q:** How to create a Docker image from a Dockerfile?  
   **A:**
6. docker build -t myimage:1.0 .
7. **Q:** How to troubleshoot a container that exits immediately?  
   **A:**
   * Check logs: docker logs <container>
   * Inspect: docker inspect <container>
   * Run interactively to debug: docker run -it <image> /bin/bash

**Scenario-Based Questions**

1. **Q:** A Docker container cannot access the network. What do you check?  
   **A:**
   * Inspect network: docker network ls, docker network inspect <network>
   * Check firewall rules on host
   * Ensure container IP is correct and DNS resolves
2. **Q:** Disk space is full on the Docker host. How do you clean it?  
   **A:**
3. docker system prune -a
4. docker volume prune
5. docker images
6. docker rmi <unused\_image>
7. **Q:** How do you persist data for a containerized database?  
   **A:** Use Docker volumes:
8. docker run -v /host/dbdata:/var/lib/mysql mysql

**Soft Skill Questions**

* How do you document changes and configurations?
* How do you collaborate with DevOps or application teams for deployments?
* Explain a time you troubleshooted a Linux or Docker issue.