30 Essential Linux interview questions

Basic Linux Commands & File System (1–8)

1. What is the difference between relative and absolute paths in Linux?

- **Answer**:
  An **absolute path** starts from the root `/`, while a **relative path** starts from your current directory.

- **Explanation**:
  Absolute paths always work regardless of your current location. Relative paths depend on where you are.

- **Use Case**:
  In scripting, absolute paths ensure scripts run correctly from any location.

Example:

```
cd /home/user/docs     # Absolute
cd ../pictures         # Relative
```

2. How do you list files and directories, including hidden ones?

- **Answer**:
  Use `ls -la` to list all files, including hidden ones.

- **Explanation**:
  Hidden files start with a `.` and are skipped unless you use `-a`.

- **Use Case**:
  Useful for finding config files like `.bashrc`.

Example:

```
ls -la
```

3. What does the chmod command do, and how do you use it?

- **Answer**:
  `chmod` changes file or directory permissions.

- **Explanation**:
  Permissions are set as `r` (read), `w` (write), `x` (execute).

- **Use Case**:
  Give scripts execute permissions.

Example:

```
chmod +x script.sh
```

4. Explain the difference between touch and mkdir.

- **Answer**:
  `touch` creates empty files; `mkdir` creates directories.

- **Explanation**:
  Use `touch` for files and `mkdir` for folders.

- **Use Case**:
  When setting up file structures for an app.

Example:

```
touch file.txt
mkdir myfolder
```

---

5. What is the purpose of the man command?

- **Answer**:
  `man` displays the manual/help page for a command.

- **Explanation**:
  Useful to check syntax and options.

- **Use Case**:
  Understanding command flags in production.

Example:

```
man grep
```

6. How do you find a file by name in Linux?

- **Answer**:
  Use `find /path -name "<filename>"`.

- **Explanation**:
  `find` searches recursively.

- **Use Case**:
  Locating log files or configs.

Example:
find /var/log -name "auth.log"

7. How do you view the current working directory and disk usage?

- **Answer**:
  Use `pwd` to view current path, `df -h` for disk usage.

- **Explanation**:
  `pwd` shows where you are; `df -h` shows disk usage in human-readable format.

- **Use Case**:
  Troubleshoot storage issues.

Example:
Pwd
df -h

8. What are file permissions like rwxr-xr--, and how do you change ownership (chown)?

- **Answer**:
  File permissions define who can read/write/execute. Use `chown user:group <file>` to change ownership.

- **Explanation**:
  Permissions follow `user-group-others` order.

- **Use Case**:
  Giving correct ownership to web or log files.

- **Example**:
  chown apache:apache index.html

User & Process Management (9–14)

9. What is the difference between su and sudo?

- **Answer**:
  `su` switches to another user; `sudo` runs a command with elevated privileges.

- **Explanation**:
  `sudo` is safer and logs actions.

- **Use Case**:
  Running a root-only command like installing software.

Example:

```
sudo apt update
su - username
```

10. How do you add a user and set a password?

- **Answer**:
  Use `useradd <username>` and `passwd <username>`.

- **Explanation**:
  Creates and secures new accounts.

- **Use Case**:
  Onboarding new team members.

Example:

```
useradd devuser
passwd devuser
```

11. How do you check running processes (ps, top, htop) in Linux?

- **Answer**:
  Use `ps aux`, `top`, or `htop` to see running processes.

- **Explanation**:
  `htop` is more interactive; `ps` is good for scripting.

- **Use Case**:
  Monitoring memory-hungry processes.

Example:

```
ps aux | grep nginx
top
htop
```

12. How do you kill a running process safely?

- **Answer**:
  Use `kill <PID>` or `kill -9 <PID>`.

- **Explanation**:
  `kill` tries graceful exit, `-9` forces it.

- **Use Case**:
  Terminate hung apps like Apache.

Example:

```
kill 1234
kill -9 5678
```

13. What is the difference between a hard link and a soft link?

- **Answer**:
  Hard links share the same inode; soft links (symlinks) point to another file.

- **Explanation**:
  Deleting original breaks soft link but not hard link.

- **Use Case**:
  Managing configs or shared files.

Example:

```
ln file1 file2     # Hard link
ln -s file1 file2  # Soft link
```

14. How do you check which service or process is using the most memory or CPU?

- **Answer**:
  Use `top`, `htop`, or `ps aux --sort=-%mem`.

- **Explanation**:
  Sorted `ps` helps in automation/scripts.

- **Use Case**:
  Finding performance bottlenecks.

Example:
ps aux --sort=-%mem | head

🔹 Networking & Connectivity (15–19)

15. How do you check your IP address and network configuration in Linux?

- **Answer**:
  Use `ip a` or `ifconfig` (older systems).

- **Explanation**:
  `ip a` shows all IP addresses and interface states.

- **Use Case**:
  Debugging connectivity issues.

Example: `ip a`

16. What is SSH and how do you connect to a remote server?

Answer:
SSH (Secure Shell) is used to connect securely to remote systems:

```
ssh user@remote_ip
```

- **Explanation**:
  Encrypts communication between local and remote systems.

- **Use Case**:
  Managing cloud servers like EC2 or VPS.

Example: `ssh ubuntu@192.168.1.100`

17. How do you transfer files using scp or rsync?

- **Answer**:
  Use `scp` for quick transfer or `rsync` for optimized syncing.

- **Explanation**:
  `rsync` only copies changes, saving time.

- **Use Case**:
  Backing up logs or configs to a remote server.

Example:
```
scp file.txt user@remote:/tmp/

rsync -avz /data/ user@remote:/backup/
```

18. How do you check if a port is in use (netstat, ss, lsof)?

- **Answer**:
  Use `ss -tuln` or `lsof -i :<port>`.

- **Explanation**:
  Shows which service is listening on which port.

- **Use Case**:
  Troubleshooting port conflicts.

Example:
```
ss -tuln

lsof -i :80
```

19. How do you troubleshoot basic network issues (ping, traceroute)?

- **Answer**:
  Use `ping <host>`, `traceroute <host>`, `nslookup <domain>`.

- **Explanation**:
  Helps check connectivity, DNS resolution, and route.

- **Use Case**:
  Diagnosing internet or VPN issues.

Example:
```
ping google.com

traceroute google.com

nslookup google.com
```

System Administration & Services (20–24)

20. How do you start, stop, and restart a service using systemctl?

Answer:
`systemctl start|stop|restart <service>`

- **Explanation**:
  Manages systemd services.

- **Use Case**:
  Restarting web servers like Apache/Nginx.
- **Example**: `systemctl restart nginx`

21. How do you check logs using journalctl or tail -f?

- **Answer**:
  Use `journalctl -u <service>` or `tail -f <logfile>`.

- **Explanation**:
  `journalctl` works with systemd; `tail -f` shows live logs.

- **Use Case**:
  Debugging service failures or runtime errors.

Example:
```
journalctl -u sshd

tail -f /var/log/syslog
```

22. How do you schedule a cron job and list existing ones?

- **Answer**:
  Use `crontab -e` to edit and `crontab -l` to list.
- **Explanation**:
  Cron schedules repetitive tasks.
- **Use Case**:
  Automating backups or report generation.

Example:
```
crontab -e

# Add: 0 2 * * * /backup.sh
```

23. How do you configure a static IP address via CLI?

- **Answer**:
  On modern systems (Netplan), edit `/etc/netplan/*.yaml`.
- **Explanation**:
  CLI IP configs depend on distro (Netplan, NetworkManager, etc.).
- **Use Case**:
  Assigning static IPs to servers or VMs.

Example (Ubuntu):
```
network:

  ethernets:

    eth0:

      dhcp4: no

      addresses: [192.168.1.100/24]
```

```
      gateway4: 192.168.1.1
      nameservers:
        addresses: [8.8.8.8, 8.8.4.4]
  version: 2
```

24. How do you troubleshoot a failed or inactive systemd service?

Answer:
Check status and logs:

```
systemctl status <service>
journalctl -xeu <service>
```

- **Explanation**:
  Status shows failure reason; journalctl shows logs.
- **Use Case**:
  Debugging failed database or application services.

Example:
```
systemctl status apache2
journalctl -xeu apache2
```

Monitoring, Scripting & Troubleshooting (25–30)

25. How do you check system uptime, CPU, and memory usage?

- **Answer**:
  Use `uptime`, `top`, `free -h`.
- **Explanation**:
  Common tools for live system resource check.
- **Use Case**:
  Identifying overloaded systems.

Example:
```
uptime
```

```
top

free -h
```

26. How do you search for a keyword inside files (grep, find)?

- **Answer**:
  Use `grep -r "keyword" /path` or combine with `find`.

- **Explanation**:
  `grep` is for content, `find` is for names.

- **Use Case**:
  Searching logs or configs.

Example:

```
grep -r "error" /var/log

find /etc -name "*.conf"
```

27. What are environment variables and how do you set/view them?

- **Answer**:
  Use `export VAR=value` to set, `echo $VAR` to view.

- **Explanation**:
  Used to configure app settings and paths.

- **Use Case**:
  Setting JAVA_HOME, PATH, etc.

Example:

```
export APP_ENV=production

echo $APP_ENV
```

28. How do you handle “Permission Denied” errors when permissions seem correct?

- **Answer**:
  Check:

  - Ownership (`ls -l`)
  - Execution (`x` flag)
  - SELinux/AppArmor
  - Directory permissions

- **Explanation**:
  It's often ownership, SELinux, or missing execute permission on parent dir.

- **Use Case**:
  Debugging script or deployment failures.

Example:
```
ls -l script.sh
chmod +x script.sh
getenforce          # For SELinux
```

29. What steps do you take to troubleshoot a slow or unresponsive Linux server?

- **Answer**:

  1. Check CPU/mem: `top`, `htop`, `vmstat`
  2. Disk usage: `df -h`, `iostat`
  3. Logs: `journalctl`, `dmesg`
  4. Network: `ping`, `netstat`, `ss`

- **Explanation**:
  Check all system resources and logs step-by-step.

- **Use Case**:
  Real-time server issue troubleshooting.

Example:
```
top
df -h
```

```
journalctl -xe

ping 8.8.8.8
```

30. Write a basic shell script to take a filename as input and display the first 10 lines.

Answer:

```
#!/bin/bash

echo "Enter filename:"

read file

head -n 10 "$file"
```

- **Explanation**:
  Reads input and shows first 10 lines.

- **Use Case**:
  Used in log scanning or data previews.