Most frequestly Asked Linux Questions in DevOps Interviews

1. What are the basic components of Linux architecture?

- **Answer:** Kernel, Shell, System Libraries, and Hardware.
  - Kernel → core that interacts with hardware.
  - Shell → command interpreter between user & kernel.

---

2. How do you check the current running processes in Linux?

- **Answer:** Using commands like:

  ```
  ps -ef      # detailed process info
  top         # live system processes
  htop        # interactive process viewer (if installed)
  ```

---

3. What is the difference between a hard link and a soft (symbolic) link?

- **Answer:**
  - Hard Link → Points directly to the inode of the file, continues to exist even if original file is deleted.
  - Soft Link → A shortcut (path-based), breaks if the target file is deleted.

  Example:

  ```
  ln file1 file1_hard    # hard link
  ln -s file1 file1_soft # soft link
  ```

---

4. How do you check disk usage and memory usage in Linux?

- **Answer:**

  ```
  df -h   # disk usage
  du -sh * # size of files/folders in current dir
  free -m # memory usage
  ```

---

5. How do you manage file permissions in Linux?

- **Answer:** Using `chmod`, `chown`, `umask`.

  ```
  chmod 755 file.txt  # rwxr-xr-x
  chown user:group file.txt
  ```

---

6. What are runlevels / systemd targets in Linux?

- **Answer:** They define the state of the system (multi-user, graphical, rescue).
  - Example: `systemctl get-default` → check current target.
  - Common ones: multi-user.target, graphical.target.

---

7. How do you search for a string inside files in Linux?

- **Answer:** Using `grep`.

  ```
  grep -r "error" /var/log/   # recursive search in logs
  ```

---

8. What is the difference between `cron` and `at`?

- **Answer:**
  - **cron** → Schedules recurring tasks.
  - **at** → Schedules a one-time task.

  ```
  crontab -e   # edit cron jobs
  at 10:30     # run command once at 10:30
  ```

---

9. How do you check open ports in Linux?

- **Answer:**

  ```
  netstat -tulnp   # list listening ports
  ss -tulnp        # modern alternative
  lsof -i :8080    # check which process is using port 8080
  ```

---

10. How do you check system logs in Linux?

- **Answer:**
  - Logs are stored in `/var/log/`.
  - Examples:

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

---

11. How do you check currently logged-in users?

- **Answer:**

```
who
w
users
```

---

12. Difference between kill, kill -9, and pkill?

- **Answer:**
    - `kill PID` → graceful termination.
    - `kill -9 PID` → force kill.
    - `pkill process_name` → kill process by name.

---

13. How do you check CPU usage and load?

- **Answer:**

```
top         # live CPU/memory usage
htop        # interactive viewer
uptime      # shows load average
mpstat      # CPU usage per core
```

---

14. How to find large files consuming disk space?

- **Answer:**

```
du -sh /* | sort -rh | head -n 10    # top 10 largest directories
find / -type f -size +100M            # files larger than 100MB
```

---

15. Difference between soft limit and hard limit?

- **Answer:**
    - **Soft limit:** temporary limit, can be increased up to hard limit.
    - **Hard limit:** maximum limit that cannot be exceeded.

```
ulimit -a   # check current limits
```

---

16. How to monitor real-time file changes?

- **Answer:**

```
tail -f /var/log/syslog
watch -n 5 "ls -lh /path/to/dir"
inotifywait -m /path/to/dir
```

---

17. Uses of symbolic links?

- **Answer:** Create shortcuts to files/folders for easy access.
- Useful for shared libraries, config files, versioned directories.

---

18. How do you check listening services and their processes?

- **Answer:**

```
netstat -tulnp
ss -tulnp
lsof -i
```

---

19. Difference between swap and RAM?

- **Answer:**
  - **RAM:** fast, temporary memory for running processes.
  - **Swap:** disk-based memory used when RAM is full.

---

20. How do you schedule recurring jobs with cron?

- **Answer:**

```
crontab -e          # edit cron jobs
crontab -l          # list cron jobs
```

==

21. In which Linux have you worked? On what version?

- **Answer:** I have worked on **Ubuntu (18.04, 20.04, 22.04 LTS), CentOS 7 & 8, and RHEL 7 & 8**. Most projects were on Ubuntu 20.04 LTS for DevOps pipelines.

---

22. What is enhanced in Ubuntu 24.04 Linux?

- **Answer:** Ubuntu 24.04 (LTS) includes:
  - Improved **performance & boot time**
  - Latest **Linux Kernel 6.x**
  - Better **cloud and container support**
  - Enhanced **security and system management tools**

---

23. What will happen if you lose the `.pem` file in AWS?

- **Answer:** You **cannot SSH** into the EC2 instance. Workarounds:
  1. Use **EC2 Instance Connect** (if enabled).
  2. Create a new key pair and replace the old key using **Systems Manager** or by mounting the EBS volume on another instance.

---

24. Suppose you have 100 GB of space in an RDS instance and want to reduce it to 25 GB. How?

- **Answer:** RDS does not allow reducing storage directly. Workarounds:
    1. **Take a snapshot** → Restore it with 25 GB.
    2. Export the data and import it into a new smaller RDS instance.

---

25. How do you check which Linux version is running?

- **Answer:**

```
cat /etc/os-release
lsb_release -a
uname -r   # kernel version
```

---

26. Difference between `apt` and `apt-get`?

- **Answer:**
    - `apt-get` → older, script-friendly.
    - `apt` → newer, user-friendly, shows progress, combines multiple commands.

---

27. How do you check running services in Linux?

- **Answer:**

```
systemctl list-units --type=service
service --status-all
```

---

28. How do you check file system type?

- **Answer:**

```
df -T
mount | column -t
```

---

29. How do you check network configuration?

- **Answer:**

```
ip addr show      # modern method
ifconfig          # older method
netstat -rn       # routing table
ping google.com   # connectivity test
```

---

30. How do you set environment variables?

- **Answer:**

```
export VAR_NAME=value                # temporary
echo "export VAR_NAME=value" >> ~/.bashrc  # persistent
source ~/.bashrc
```

---

31. How do you check disk inode usage?

- **Answer:**

```
df -i
```

---

32. How do you restart a service?

- **Answer:**

```
systemctl restart nginx.service
service nginx restart
```

---

33. How do you find large log files and truncate them?

- **Answer:**

```
find /var/log -type f -size +100M
> /var/log/largefile.log   # truncate without deleting
```

---

34. How do you monitor CPU and memory in real-time?

- **Answer:**

```
top
htop
vmstat 1
```

---

35. How do you add a new user and give sudo privileges?

- **Answer:**

```
useradd username
passwd username
usermod -aG sudo username   # Ubuntu
```

36. What is `awk` and how is it used in DevOps?

Answer:

- **`awk`** is a **text-processing tool** in Linux used to **extract, filter, and manipulate data from files or command outputs**.
- It is extremely useful in DevOps for **parsing logs, extracting metrics, and automating scripts**.

37. What is the difference between /bin, /sbin, /usr/bin, and /usr/sbin?

Answer:

- `/bin` → Essential user binaries (basic commands).
- `/sbin` → System binaries (admin tasks).
- `/usr/bin` → Non-essential user binaries (installed packages).
- `/usr/sbin` → Non-essential system binaries for root/admin.

---

38. How do you check which process is consuming the most memory?

Answer:

- `top` → Press `M` to sort by memory.
- `ps aux --sort=-%mem | head -n 10` → Top 10 memory-consuming processes.

---

39. What is the difference between a process and a thread in Linux?

Answer:

- Process → Independent program with its own memory space.
- Thread → Lightweight unit within a process, shares memory with other threads.

---

40. What are load averages in Linux (uptime command)?

Answer:

- Load average shows system load in 1, 5, and 15 minutes.
- Example: `1.00` means one CPU core fully utilized.
- On 4 cores → `4.00` = fully utilized.

---

41. How do you find which process is using the most I/O?

Answer:

- Use `iotop` (if installed).
- `pidstat -d 1` → per-process I/O.

---

42. Difference between su and sudo?

Answer:

- `su` → Switch user (requires root password).
- `sudo` → Run command as superuser (requires user’s password, controlled by sudoers).

---

43. How do you check failed login attempts?

Answer:

- `/var/log/auth.log` (Debian/Ubuntu).
- `/var/log/secure` (RHEL/CentOS).
- Commands:
  - `lastb` → failed login attempts.
  - `faillog -a` → user failure logs.

---

44. How do you find zombie processes?

Answer:

- `ps aux | grep Z`
- Zombie = process finished execution but still in process table (waiting for parent to clean up).

---

45. What is the difference between SELinux and AppArmor?

Answer:

- SELinux → Mandatory Access Control (RHEL/CentOS).
- AppArmor → Profile-based security (Ubuntu/Debian).
- Both enhance Linux security but with different approaches.

---

46. How do you find which service is using a specific port?

Answer:

- `lsof -i :8080`
- `netstat -tulnp | grep 8080`
- `ss -ltnp | grep 8080`

---

47. How do you find which package a file belongs to?

Answer:

- Debian/Ubuntu → `dpkg -S /path/to/file`
- RHEL/CentOS → `rpm -qf /path/to/file`

---

48. What is the difference between ext3, ext4, and XFS?

Answer:

- ext3 → older journaling FS.
- ext4 → supports larger files, better performance.
- XFS → high-performance, good for large files and parallel I/O.

---

49. How do you troubleshoot high CPU usage in Linux?

Answer:

1. `top` / `htop` → check top processes.
2. `ps -eo pid,ppid,cmd,%cpu --sort=-%cpu | head`
3. Check kernel logs (`dmesg`).
4. Use `strace` on process if stuck.

---

50. What is cgroups in Linux?

Answer:
Control Groups → Kernel feature to limit and isolate resource usage (CPU, memory, I/O) for processes.

- Widely used in Docker/Kubernetes.

---

51. What is the difference between sticky bit, SUID, and SGID?

Answer:

- **SUID (Set User ID):** File runs with owner’s privileges.
- **SGID (Set Group ID):** File runs with group’s privileges.
- **Sticky Bit:** On directories, only owner can delete their files (e.g., `/tmp`).

---

52. How do you check network performance?

Answer:

- `ping` → connectivity.
- `iperf3` → bandwidth.
- `traceroute` → routing path.
- `netstat` / `ss` → connections.

---

53. How do you capture packets in Linux?

Answer:

- `tcpdump -i eth0 port 80` → capture HTTP traffic.
- `wireshark` → GUI-based packet analyzer.

---

54. How do you create persistent firewall rules in Linux?

Answer:

- Ubuntu/Debian → `ufw allow 22/tcp`, `ufw enable`.
- RHEL/CentOS → `firewall-cmd --permanent --add-port=22/tcp` → `firewall-cmd --reload`.

---

55. What is the difference between init.d, upstart, and systemd?

Answer:

- init.d → Old SysV init system.
- upstart → Event-driven init (Ubuntu 9–14).
- systemd → Modern system manager, default in most Linux distros now.

---

56. How do you analyze disk performance?

Answer:

- `iostat` → CPU + I/O stats.
- `iotop` → per-process disk usage.
- `sar -d` → disk activity report.

---

57. What is the difference between NFS and Samba?

Answer:

- NFS → Unix/Linux file sharing.
- Samba → Windows/Linux interoperability (SMB/CIFS).

---

58. How do you secure SSH access?

Answer:

- Disable root login.
- Use key-based authentication.
- Change default port.
- Restrict by IP.
- Use Fail2Ban for brute force protection.

---

59. How do you troubleshoot DNS issues in Linux?

Answer:

- `nslookup example.com`
- `dig example.com`
- `cat /etc/resolv.conf` → check DNS servers.
- `systemd-resolve --status`

---

60. How do you check which process opened a file?

Answer:

- `lsof /path/to/file` → list process using the file.

61. How do you find which process is consuming high network bandwidth?

Answer:

- Use `iftop` (real-time interface monitoring).
- Use `nethogs` to see per-process bandwidth usage.

---

62. What is the difference between CPU load and CPU utilization?

Answer:

- **CPU load** → number of processes waiting to run (from `uptime` or `top`).
- **CPU utilization** → percentage of CPU cycles actually used (from `mpstat`, `top`).

---

63. How do you identify which user ran a specific command?

Answer:

- Check `.bash_history` of the user.
- Check `/var/log/auth.log` or `/var/log/secure`.
- Enable auditing with `auditd` for sensitive commands.

---

64. What are Linux namespaces?

Answer:
Namespaces isolate resources (process IDs, network, mounts, users, etc.) for processes.

- Used heavily in **containers (Docker, Kubernetes)**.

---

65. What is the difference between nice and renice?

Answer:

- `nice` → starts a process with a priority.
- `renice` → changes priority of an already running process.

---

66. How do you check SELinux mode?

Answer:

- `getenforce` → shows current mode (Enforcing, Permissive, Disabled).
- `sestatus` → detailed SELinux status.

---

67. How do you troubleshoot when a Linux server is very slow?

Answer:

1. `uptime` → load average.
2. `top` / `htop` → CPU/memory usage.
3. `iostat` → disk I/O.
4. `iftop` → network usage.
5. `dmesg` / logs → hardware/kernel issues.

---

68. How do you analyze kernel logs?

Answer:

- `dmesg` → kernel ring buffer.
- `journalctl -k` → systemd kernel logs.

---

69. How do you find which shared libraries a binary depends on?

Answer:

- `ldd /path/to/binary`

---

70. What’s the difference between grep, egrep, and fgrep?

Answer:

- `grep` → basic regex.
- `egrep` → extended regex (supports `+`, `?`, `|`).
- `fgrep` → fixed string search (no regex).

---

71. How do you find top 10 CPU-consuming processes over time?

Answer:

- `ps -eo pid,ppid,cmd,%cpu --sort=-%cpu | head -n 10`
- Use `sar -u 1 10` for CPU history.

---

72. What is the difference between synchronous and asynchronous I/O?

Answer:

- **Synchronous I/O** → Process waits until operation finishes.
- **Asynchronous I/O** → Process continues without waiting, kernel signals when done.

---

73. How do you list open files by a process?

Answer:

- `lsof -p <PID>`

---

74. How do you troubleshoot high memory usage?

Answer:

- `free -m` → total memory usage.
- `ps aux --sort=-%mem | head -n 10` → processes using memory.
- `vmstat 1` → memory paging.
- `smem` → detailed memory breakdown.

---

75. How do you check TCP connections and their states?

Answer:

- `ss -s` → summary of socket connections.
- `netstat -ant | grep ESTABLISHED` → active connections.

---

76. What is the difference between IPv4 and IPv6 in Linux configuration?

Answer:

- IPv4 → 32-bit addressing (`192.168.1.1`).
- IPv6 → 128-bit addressing (`2001:db8::1`).
- Configured via `/etc/network/interfaces`, `netplan`, or `NetworkManager`.

---

77. How do you check which process opened a network port?

Answer:

- `lsof -i :22`
- `ss -ltnp | grep :22`

---

78. How do you limit CPU/memory usage for a process in Linux?

Answer:

- `ulimit` → set resource limits.
- `cgroups` → precise resource control (used in containers).

---

79. How do you check which kernel modules are loaded?

Answer:

- `lsmod` → list loaded modules.
- `modinfo <module>` → module details.

---

80. What is the difference between journald and syslog?

Answer:

- **syslog** → traditional plain-text log system.
- **journald** → systemd logging with structured, indexed logs.

---

81. How do you find all processes started by a specific user?

Answer:

- `ps -u username`
- `pgrep -u username`

---

82. What is hugepages in Linux?

Answer:
Hugepages allow memory pages larger than default (4KB) to improve performance for large-memory apps like databases.

---

83. How do you find which file descriptors a process has opened?

Answer:

- `ls -l /proc/<PID>/fd/`

---

84. How do you permanently change hostname in Linux?

Answer:

- `hostnamectl set-hostname newname` (systemd-based).
- Edit `/etc/hostname` (older systems).

---

85. What is swapiness in Linux?

Answer:

- Kernel parameter (`vm.swappiness`) that defines swap usage preference.
- Value 0 → avoid swap, 100 → aggressive swap.

---

86. How do you check DNS cache in Linux?

Answer:

- `systemd-resolve --statistics`
- `nscd -g` (if Name Service Cache Daemon used).

---

87. How do you debug a process stuck in uninterruptible sleep (D state)?

Answer:

- Use `ps -o state -p <PID>` → if `D`, it's waiting for I/O.
- Check `dmesg` and `iostat` for disk/NFS issues.

---

88. What is the difference between soft mount and hard mount in NFS?

Answer:

- **Soft mount** → Returns error if server is unreachable.
- **Hard mount** → Keeps retrying until server responds (default, safer for data).

---

89. How do you schedule a job to run every 5 minutes in cron?

Answer:

- `*/5 * * * * /path/to/script.sh`

---

90. How do you capture and analyze system calls of a process?

Answer:

- `strace -p <PID>` → trace system calls.

- Useful for debugging stuck/hanging processes.

91. Scenario:

Your application is randomly freezing. How would you check if it's caused by kernel-level I/O blocking?
Answer:

- Use `dstat` / `iostat -x` to monitor I/O wait.
- Use `strace -p <PID>` to check blocked system calls.
- Inspect `/proc/<PID>/stack` for kernel-level wait states.

---

92. Scenario:

You see very high **load average**, but CPU utilization is low. What does it mean?
Answer:

- Load average includes processes waiting on **I/O, disk, or locks**.
- Likely cause: Disk bottleneck, NFS mount issues, or memory pressure.
- Debug using `iostat`, `vmstat`, and `dmesg`.

---

93. Scenario:

A server with 64GB RAM is still swapping aggressively. How would you handle this?
Answer:

- Check `vm.swappiness` (default 60).
- Reduce it (`sysctl -w vm.swappiness=10`).
- Check for memory leaks via `smem` or `/proc/<PID>/smaps`.
- Use `oom_score_adj` to protect critical processes.

---

94. Scenario:

Your production server is facing TCP connection leaks. How would you debug?
Answer:

- `ss -ant state established` → check TCP sessions.
- `netstat -s` → TCP statistics (retransmissions, failures).
- Enable `tcpdump` to capture packets.
- Inspect `/proc/net/tcp` for orphaned connections.

---

95. Scenario:

A process is using 100% CPU but doing no useful work. How do you find why?
Answer:

- Attach `strace -p <PID>` → see syscalls.
- Use `perf top` → identify hot functions.
- Might be a **spinlock**, busy loop, or kernel bug.

---

96. Scenario:

Your containerized app inside Kubernetes shows **“Too many open files”** error. How do you fix?
Answer:

- Check `ulimit -n` (file descriptor limit).
- Increase limits in systemd service (`LimitNOFILE=`).
- Configure Docker/Kubernetes pod spec with higher `nofile`.
- Verify via `/proc/<PID>/limits`.

---

97. Scenario:

During a deployment, the filesystem went **read-only**. What could be the reason?
Answer:

- Kernel remounted filesystem as read-only due to disk errors.
- Check `dmesg | grep EXT4` or `XFS`.
- Run `fsck` after reboot.
- If frequent → hardware/disk replacement needed.

---

98. Scenario:

You need to trace **why DNS lookups are slow** on your Linux server. How would you do it?
Answer:

- Check `/etc/resolv.conf` for misconfigured DNS.
- Use `dig +trace` to debug.
- Use `strace -e trace=network -p <PID>` to see DNS calls.
- Check if `nscd` or `systemd-resolved` caching is working.

---

99. Scenario:

A developer complains their cron job didn't run, but cron is active. How do you debug?
Answer:

- Check `/var/log/cron` or `journalctl -u cron`.
- Ensure script has **execute permission**.
- Check environment differences (`cron` runs with limited PATH).
- Confirm correct newline/format in crontab.

---

100. Scenario:

Your application shows random latency spikes. How do you trace kernel-level performance issues?
Answer:

- Use **`perf` tools**: `perf top`, `perf record`, `perf report`.
- Use **`eBPF`/bcc tools** (`execsnoop`, `opensnoop`, `biolatency`).
- Check CPU scheduling latency via `latencytop`.
- Debug NUMA imbalance with `numactl --hardware`.