GENERAL

linux system administration interview questions

Published: September 3, 2026 | Index ID: #-

Introduction

Linux system administration is one of the most sought-after skills in today's technology landscape. From cloud-native infrastructures to enterprise servers, Linux powers a vast majority of critical systems worldwide. As a result, organizations are constantly looking for seasoned professionals who can maintain, secure, and optimize these environments. Whether you're a seasoned administrator preparing for a senior role or a fresh graduate aiming to break into the field, mastering the most common **Linux system administration interview questions**can make the difference between landing the job and missing the opportunity.

In this article, we'll walk through the most frequently asked interview questions, organized by topic, and provide concise yet comprehensive answers. We'll also share preparation tips, real-world scenarios, and resources to help you feel confident and ready for the interview. By the end, you'll have a solid roadmap to showcase your expertise and demonstrate your problem solving mindset to hiring managers.

Core Linux Fundamentals

What is Linux, and how does it differ from other operating systems?

Linux is an open source, Unix-like operating system kernel that forms the foundation of many distributions such as Ubuntu, CentOS, Debian, and Red Hat Enterprise Linux (RHEL). Unlike proprietary systems like Windows, Linux is freely available, highly modular, and supports a wide range of hardware. Its kernel architecture allows users to compile only the modules they need, leading to efficient resource usage. Linux's strong emphasis on command line interfaces, scripting, and package management distinguishes it from GUI centric operating systems.

Explain the Linux boot process.

The boot sequence typically follows these steps:

1. BIOS/UEFI initializes hardware and loads the bootloader.
2. The bootloader (GRUB for most distributions) presents a menu and loads the kernel into memory.
3. The kernel initializes core subsystems, mounts the root filesystem, and starts the init system.
4. Init (systemd, upstart, or SysV) reads configuration files and launches system services in the correct order.
5. Once all services are up, the system presents a login prompt or graphical environment.

Understanding this flow is crucial for troubleshooting boot failures and configuring custom startup scripts.

What are the key differences between init, systemd, and upstart?

Each init system manages services and processes during boot and shutdown:

- SysV init uses shell scripts located in /etc/init.d and a runlevel system. It's simple but lacks parallel execution.
- Upstart introduced event driven job control, allowing services to start based on system events, but it's no longer actively maintained.
- Systemd is the modern default for most distributions. It uses unit files, supports parallel startup, socket activation, and provides a unified interface via systemctl.

Familiarity with systemd's unit files, dependencies, and cgroups is essential for modern Linux environments.

Command Line and Shell Proficiency

How would you list all processes running under a specific user?

Use ps -u username or ps -u username -f for full format. Alternatively, pgrep -u username returns PIDs only. For a more detailed view, top -u username or htop (if installed) can be helpful.

Explain the difference between grep and egrep.

grep searches for simple fixed string patterns, while egrep (or grep -E) supports extended regular expressions, allowing more complex pattern matching like alternation, repetition, and grouping.

How would you find the memory usage of a specific process?

Run ps -p PID -o rss,vsz,cmd to display resident set size (RSS) and virtual size (VSZ). For a real time view, top -p PID or htop can be used.

What is a symbolic link, and how do you create one?

A symbolic link (symlink) is a reference that points to another file or directory. Create one with ln -s /path/to/target /path/to/link. Symlinks are useful for aliasing configuration files or redirecting applications to newer binaries.

Explain how you would locate a file on the system.

Use find / -name filename for a recursive search, or locate filename if the database is updated. For speed, fd (a modern alternative) or rg can be employed.

System Management and Configuration

Describe the process of adding a new user.

Run useradd -m username to create the account and home directory. Set a password with passwd username. For group membership, use usermod -aG group username. Finally, verify with id username.

How do you change file permissions and ownership?

Use chmod for permissions (e.g., chmod 755 file) and chown for ownership (e.g., chown user:group file). For recursive changes, add the -R flag.

Explain the concept of a mount point and how you would mount a filesystem.

A mount point is a directory where a filesystem is attached. To mount, use mount -t ext4 /dev/sdb1 /mnt/data. To make it persistent across reboots, add an entry to /etc/fstab specifying the device, mount point, filesystem type, and options.

What is a swap file, and how do you configure one?

A swap file provides virtual memory. Create it with fallocate -l 2G /swapfile, set permissions chmod 600 /swapfile, format it mkswap /swapfile, and enable swapon /swapfile. Add to /etc/fstab for persistence. Adjust swappiness in /etc/sysctl.conf as needed.

How would you manage package installations on RHEL-based and Debian-based systems?

On RHEL/CentOS, use yum or dnf (e.g., dnf install nginx). On Debian/Ubuntu, use apt (e.g., apt install nginx). For both, you can query, update, and remove packages using their respective package managers. Keep repositories up to date with yum update or apt update && apt upgrade.

Networking and Remote Access

How do you view the current network configuration?

Use ip addr show for interface details, ip route show for routing tables, and netstat -tulnp or ss -tulnp for listening sockets. For DNS resolution, check /etc/resolv.conf and use dig or nslookup.

Explain how you would set up SSH key-based authentication.

Generate a key pair with ssh-keygen -t rsa (or ed25519). Copy the public key to the remote host's ~/.ssh/authorized_keys using ssh-copy-id user@host. Ensure proper permissions: chmod 700 ~/.ssh and chmod 600 ~/.ssh/authorized_keys. Disable password authentication in /etc/ssh/sshd_config by setting PasswordAuthentication no and reload the service.

What is a firewall, and how do you configure it with firewalld?

A firewall controls inbound and outbound traffic. Firewalld provides zone-based management. To open HTTP, run firewall-cmd --zone=public --add-service=http --permanent, then firewall-cmd --reload. To view active zones, firewall-cmd --get-active-zones. For advanced rules, use firewall-cmd --add-rich-rule="rule family="ipv4" source address="192.168.1.0/24" port port="22" protocol="tcp" accept".

Describe how you would troubleshoot a network connectivity issue.

Follow a systematic approach:

1. Check physical connectivity (cables, switches).
2. Verify interface status with ip link show.
3. Confirm IP configuration using ip addr show and ifconfig.
4. Test reachability with ping to gateway and external hosts.
5. Inspect routing tables via ip route.
6. Check firewall rules with firewall-cmd --list-all.
7. Review logs in /var/log/messages or journalctl -u NetworkManager.
8. Use traceroute or mtr for path analysis.

Security and Hardening

What is SELinux, and how does it differ from AppArmor?

SELinux (Security-Enhanced Linux) is a mandatory access control (MAC) system that enforces fine-grained policies on processes, files, and network resources. It operates via labeling and policy rules. AppArmor is a simpler MAC system that uses profile files to restrict applications. SELinux offers more granular control but can be more complex to manage.

How would you check the status of SELinux and change it to permissive mode?

Check status with sestatus or getenforce. To set permissive temporarily, run setenforce 0. To make it permanent, edit /etc/selinux/config and set SELINUX=permissive, then reboot.

Explain the steps to secure a Linux server.

Key steps include:

- Apply the latest security patches with yum update or apt upgrade.
- Disable unnecessary services and ports.
- Implement firewall rules with firewalld or iptables.

Baca Juga (Artikel Terkait)

- **automaker mini software version v8 erptoboss**
  URL: http://art-consciousness.com/automaker-mini-software-version-v8-erptoboss.pdf
- **automata and computability**
  URL: http://art-consciousness.com/automata-and-computability.pdf
- **automata and languages theory and applications**
  URL: http://art-consciousness.com/automata-and-languages-theory-and-applications.pdf
- **automata and mechanical toys**
  URL: http://art-consciousness.com/automata-and-mechanical-toys.pdf

Tags: #linux #system #administration #interview #questions