Linux Server - Server Security and Automation

1. What are the key components of securing a Linux server?

Key components include:
- Regular system updates and patches
- User and permission management
- Firewall configuration (iptables/nftables)
- SSH hardening
- Intrusion detection systems (e.g., Fail2ban, Snort)
- Secure file permissions
- Audit logs and monitoring.

2. How can you secure SSH on a Linux server?

Steps to secure SSH include:
- Changing the default SSH port
- Disabling root login
- Using SSH keys instead of passwords
- Enabling SSH protocol version 2
- Limiting login attempts using Fail2ban or similar tools.

3. Describe a practical method to automate Linux system updates.

You can use a cron job with a package manager like `apt` or `yum`:
Example (for Debian/Ubuntu):
`sudo apt update && sudo apt upgrade -y`
Schedule it in crontab: `0 2 * * * /usr/bin/apt update && /usr/bin/apt upgrade -y`

4. What is SELinux and how does it enhance security?

SELinux (Security-Enhanced Linux) is a security module that provides mandatory access control. It defines access controls for applications, processes, and files to limit access based on policies, reducing the risk of vulnerabilities.

5. Scenario: You discover unauthorized login attempts on your server. What steps do you take to respond?

1. Check authentication logs (/var/log/auth.log)
2. Identify the source IP and block it using the firewall
3. Check for any compromised accounts
4. Change affected user passwords
5. Enable Fail2ban to block repeated login attempts
6. Notify relevant stakeholders and document the incident.

6. How would you automate backups on a Linux server?

Use a script and schedule it with cron:
Script example:
`tar -czf /backup/home_backup.tar.gz /home`
Cron entry: `0 3 * * * /path/to/backup_script.sh`

7. What is Ansible and how is it used in Linux server automation?

Ansible is an open-source IT automation tool. It is used for configuration management, application deployment, and task automation by using playbooks written in YAML to define tasks and roles.

8. Explain how to restrict user access to specific commands on a Linux server.

You can use the `sudoers` file to restrict access. Use `visudo` to safely edit the file:
`username ALL=(ALL) /usr/sbin/service apache2 restart` allows only the Apache restart command.

9. What are some best practices for log management on Linux servers?

- Centralize logs using tools like rsyslog or syslog-ng
- Rotate logs using logrotate
- Monitor logs for unusual activity
- Secure log files with proper permissions
- Back up important logs regularly

10. Scenario: Your automation script fails intermittently. How do you debug the issue?

1. Check script logs and output
2. Add verbose and debug output to the script
3. Verify environment variables and dependencies
4. Review cron or systemd logs if automated
5. Test manually in isolation to identify the failure point