1. Linux Hardening Checklist

A. System Updates & Patch Management

Update system regularly using package manager (apt update, yum update).

Enable automatic security updates.

Remove unused packages and old kernels.

Check for vulnerabilities using tools like Lynis or OpenVAS.

B. User Accounts & Authentication

Create strong password policies (minimum length, complexity).

Disable root login via SSH.

Use sudo for administrative access instead of root.

Remove or disable unused user accounts.

Enable account lockout after multiple failed login attempts.

Implement multi-factor authentication

(MFA) if possible.
C. File Permissions & Ownership
Set proper permissions using chmod, chown.
Restrict access to sensitive files (e.g., /etc/passwd, /etc/shadow).
Avoid giving 777 permissions.
Regularly audit file permissions.
Enable sticky bit on shared directories.
D. Firewall Configuration
Enable firewall (UFW, firewalld, or iptables).
Allow only required ports (e.g., 22 for SSH).
Block unused ports and services.
Monitor firewall logs regularly.
E. SSH Security
Change default SSH port (optional but recommended).
Disable root login (PermitRootLogin no).
Use SSH key-based authentication instead of passwords.
Limit login attempts.

Allow only specific users via SSH.

F. Service & Process Management

Disable unnecessary services (e.g., FTP, Telnet).

Check running processes using ps, top.

Use systemctl to manage services.

Monitor suspicious processes.

G. Logging & Monitoring

Enable system logging (rsyslog, journald).

Monitor logs in /var/log/.

Use intrusion detection tools (Fail2ban, AIDE).

Set up log rotation and backup.

H. Network Security

Disable unused network interfaces.

Use secure protocols (SSH, HTTPS).

Disable insecure services like Telnet.

Monitor network traffic using netstat, ss.

I. Backup & Recovery

Schedule regular backups.

Store backups securely (offline/cloud).

Test backup restoration.
Use automated backup tools.
J. Malware & Intrusion Protection
Install antivirus (ClamAV).
Use Fail2ban to block brute-force attacks.
Use SELinux or AppArmor for access control.
Scan system regularly for malware.
2. Security Configuration Summary
The Linux system was secured by implementing essential security configurations to protect against unauthorized access, malware, and network attacks.
First, the system was updated with the latest security patches and unnecessary packages were removed to reduce vulnerabilities. Strong password policies and user access controls were applied to ensure only authorized users can access the system. Root login was disabled and

sudo access was configured for administrative tasks.
File permissions and ownership were properly configured to protect sensitive data. A firewall was enabled to allow only required network traffic and block unauthorized connections. SSH security was strengthened by disabling root login and enabling key-based authentication. Unnecessary services and ports were disabled to reduce the attack surface. System logs and processes were continuously monitored for suspicious activity. Tools like Fail2ban and antivirus software were used for intrusion and malware protection.
Regular backups were scheduled to ensure data recovery in case of system failure or attack. Overall, these security configurations improved system confidentiality, integrity, and availability,

making the Linux system more secure and resilient against cyber threats.