

**SCHOOL OF COMPUTER SCIENCE ENGINEERING**

**AND INFORMATION SYSTEMS**

**WINTER SEMESTER 2024-2025**

**PMCA605L – CYBER SECURITY**

**DIGITAL ASSIGNMENT – 1**

**TOOL CHOSEN: LYNIS**

**SUBMITTED ON: 07 – FEB - 2025**

**SUBMITTED BY-**

**AKASH KUMAR BANIK**

**PROGRAM: MCA**

**REGISTER No.: 24MCA0242**

1. Identify any ONE Cyber Security tool of your choice for providing Security, Analysis, and Audit security for any organization. Expand the tool with a small example or application to understand the functionality of the tool. Write down the step-by-step process to use the tool with its use.

**Lynis: A Comprehensive Cybersecurity Tool for Security Auditing and Analysis**

**Introduction to Lynis**

Lynis is a powerful, open-source security auditing tool designed for Unix-based systems, including Kali Linux. It is widely used by system administrators, security professionals, and auditors to assess system security, analyze vulnerabilities, and ensure compliance with security policies. Unlike traditional antivirus programs, Lynis does not focus on detecting malware alone but performs a comprehensive security audit to identify misconfigurations, weak security settings, and potential vulnerabilities that may expose the system to attacks.

Lynis is commonly used for penetration testing, forensic analysis, compliance testing (ISO 27001, PCI-DSS, NIST, and HIPAA), and system hardening. Since Lynis does not require any agents to be installed, it provides a lightweight and efficient method for auditing systems without additional overhead.

**Key Features of Lynis**

1. Security Auditing – Lynis performs detailed system audits by analyzing security settings, installed software, and misconfigurations.
2. System Hardening Recommendations – After scanning the system, Lynis provides actionable recommendations to enhance security.
3. Compliance Testing – It helps organizations meet compliance requirements for ISO 27001, PCI-DSS, NIST, and HIPAA by detecting violations in security policies.
4. Malware and Rootkit Detection – Lynis scans the system for potential rootkits, malware, and unauthorized modifications in system files.
5. Log File Analysis and Forensics – The tool helps identify security incidents by analyzing logs and tracking suspicious activities.
6. Lightweight and Agentless – Lynis does not require installation of additional software agents, making it efficient for quick system audits.

**STEP-BY-STEP GUIDE TO USING LYNIS ON KALI LINUX**

**Step 1: Installing Lynis**

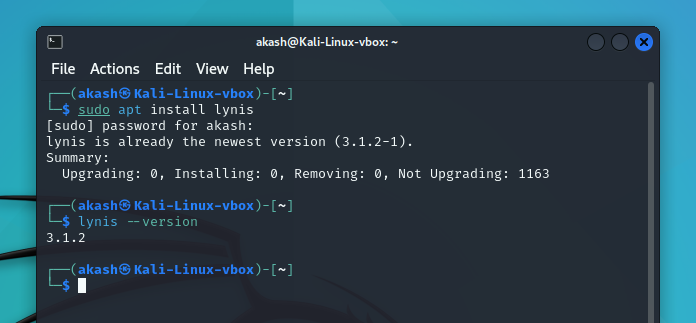
Lynis is pre-installed on Kali Linux, but if it is missing or needs an update, it can be installed using the following command:

*sudo apt update && sudo apt install lynis*

Once installed, the user can verify the installation by checking the version of Lynis:

*lynis --version*

If the installation is successful, the terminal will display the installed version of Lynis.



**Step 2: Performing a Full System Security Audit**

To conduct a comprehensive security audit, the following command is used:

*sudo lynis audit system*

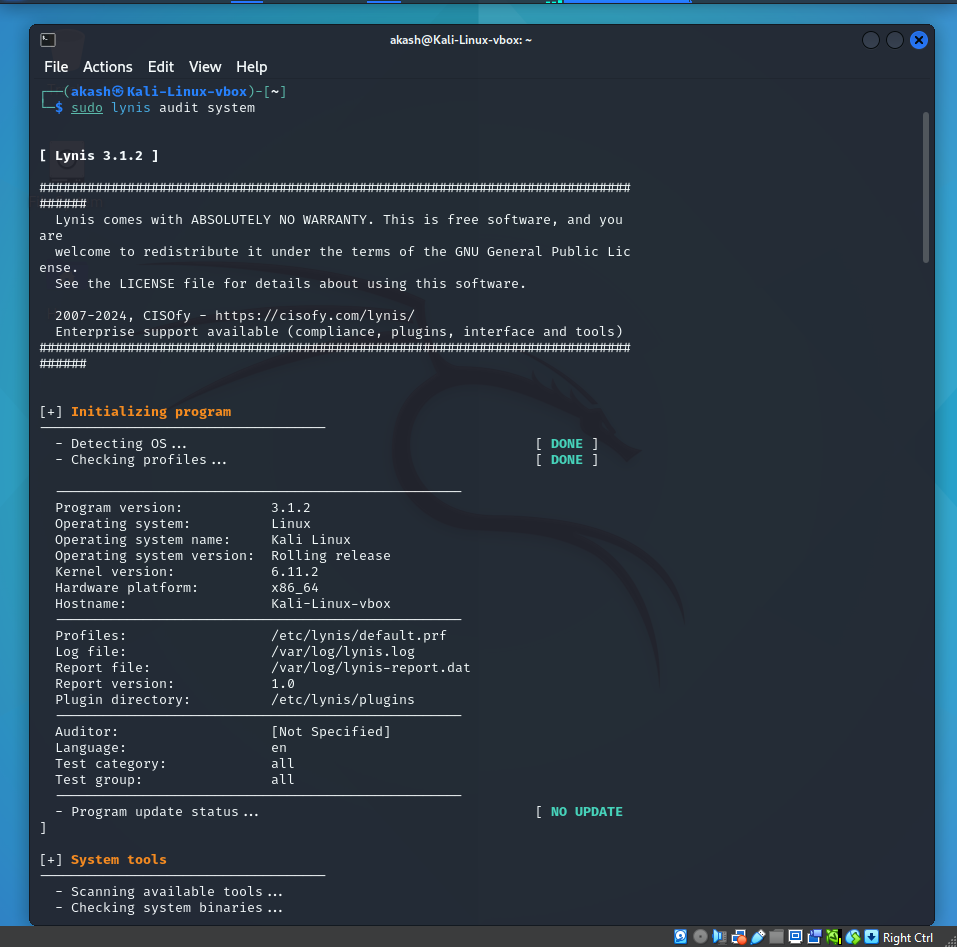
**What Happens During the Scan?**

Lynis begins analyzing system files, configurations, and installed software.

It runs hundreds of security tests, including checks for firewall status, SSH security, user authentication methods, system integrity, and log file analysis.

A detailed audit report is displayed in the terminal, highlighting warnings and security suggestions.

The results are stored in a log file **(/var/log/lynis.log)** for later review.

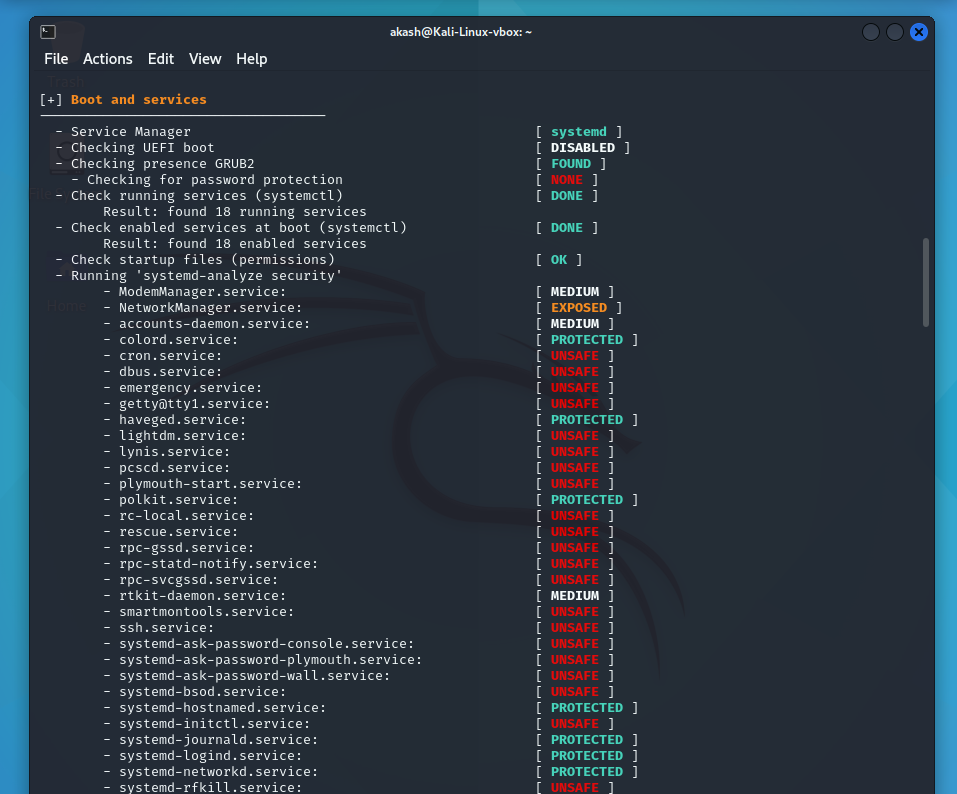


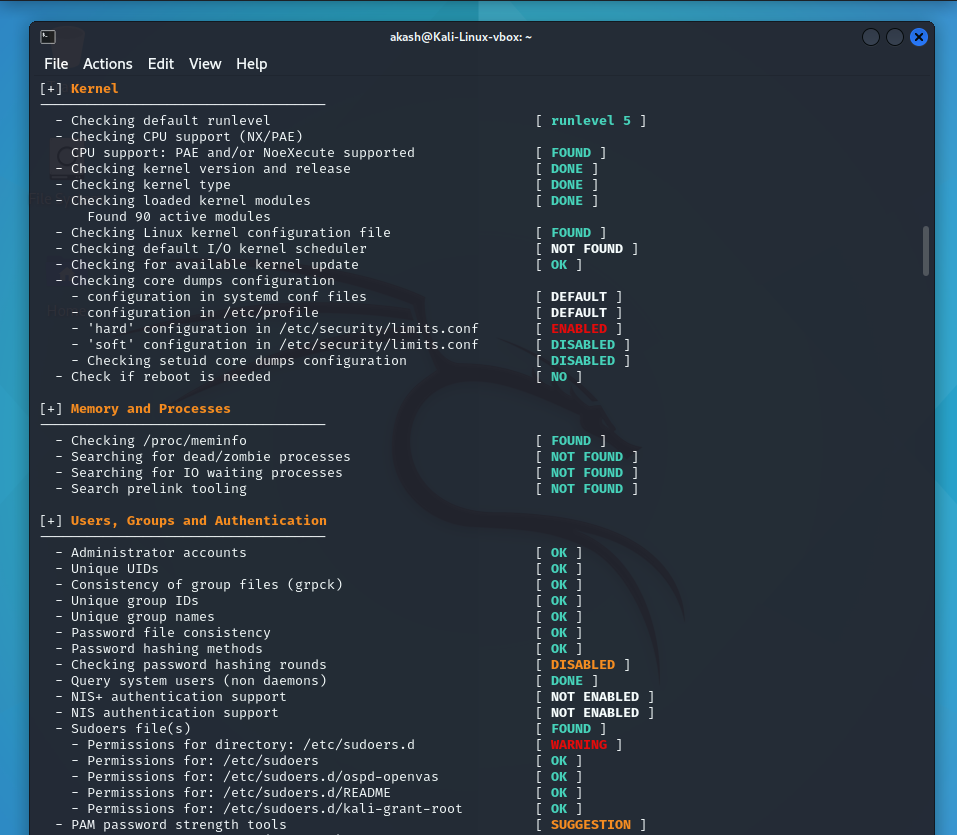
**Step 3: Understanding the Audit Results**

Once the audit is complete, Lynis provides a security report with warnings and recommendations. Each finding is assigned a status, such as:

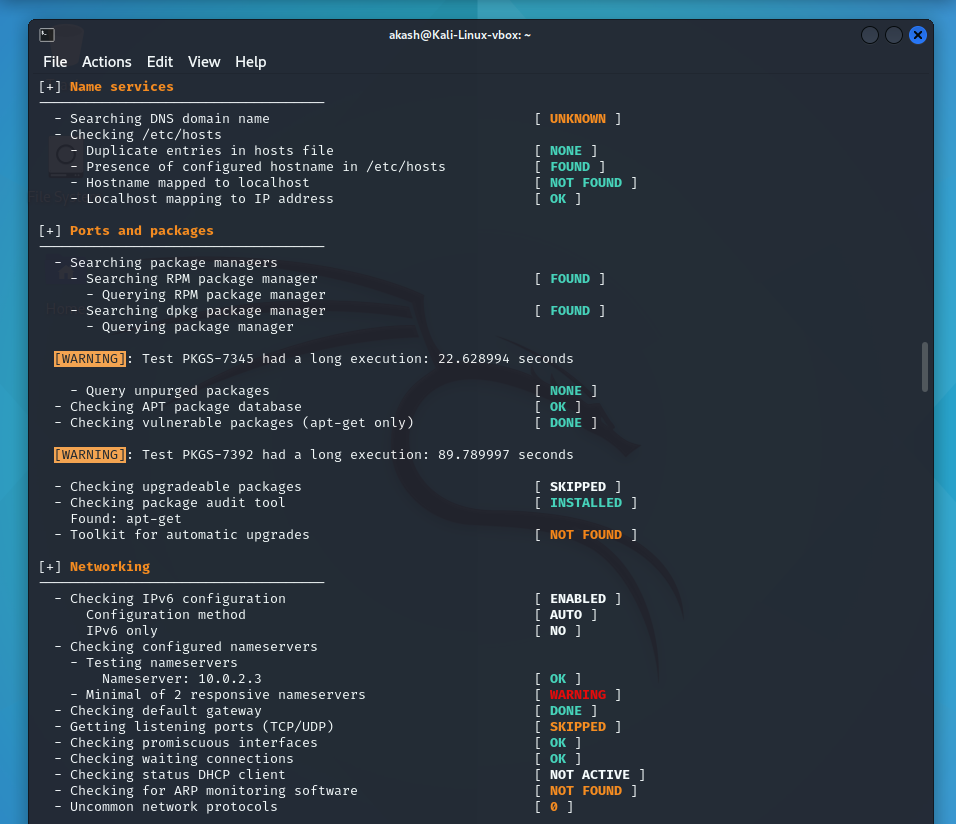
* [OK] – No issues detected.
* [WARNING] – A potential security risk that should be reviewed.
* [SUGGESTION] – Recommended improvement for better security.

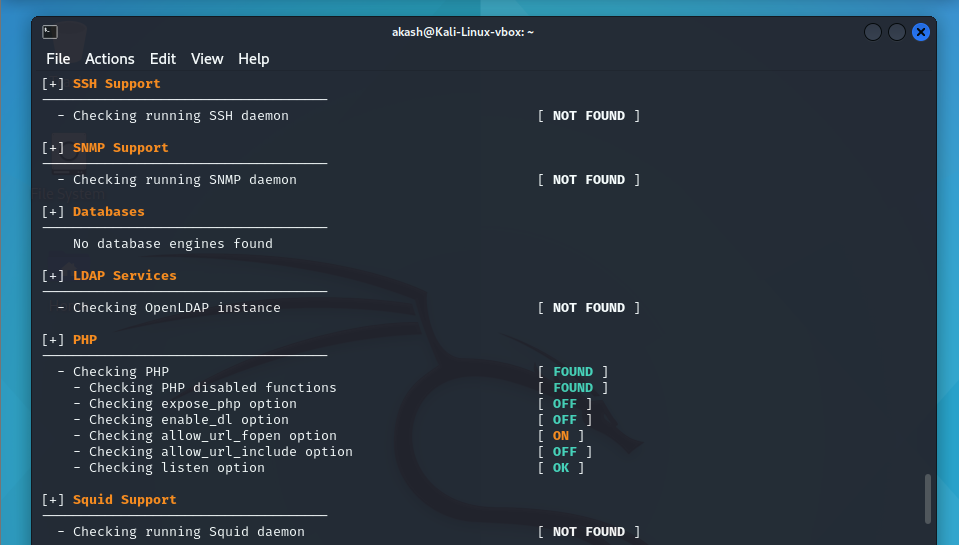
**AUDIT REPORT OUTPUT:**

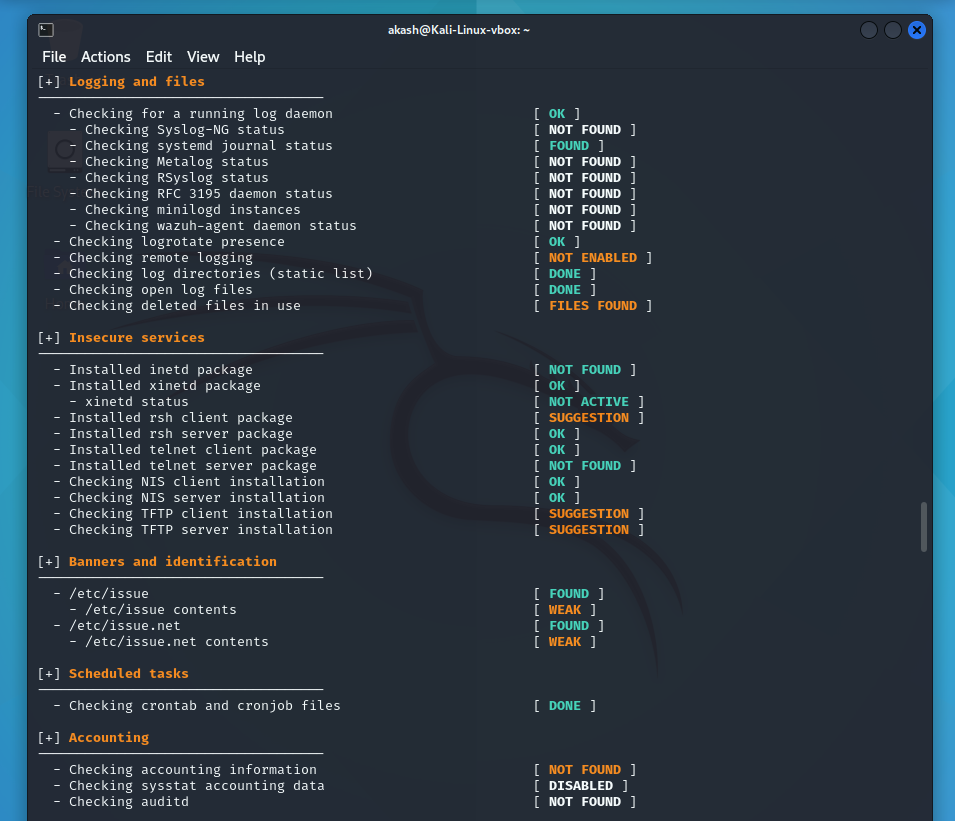








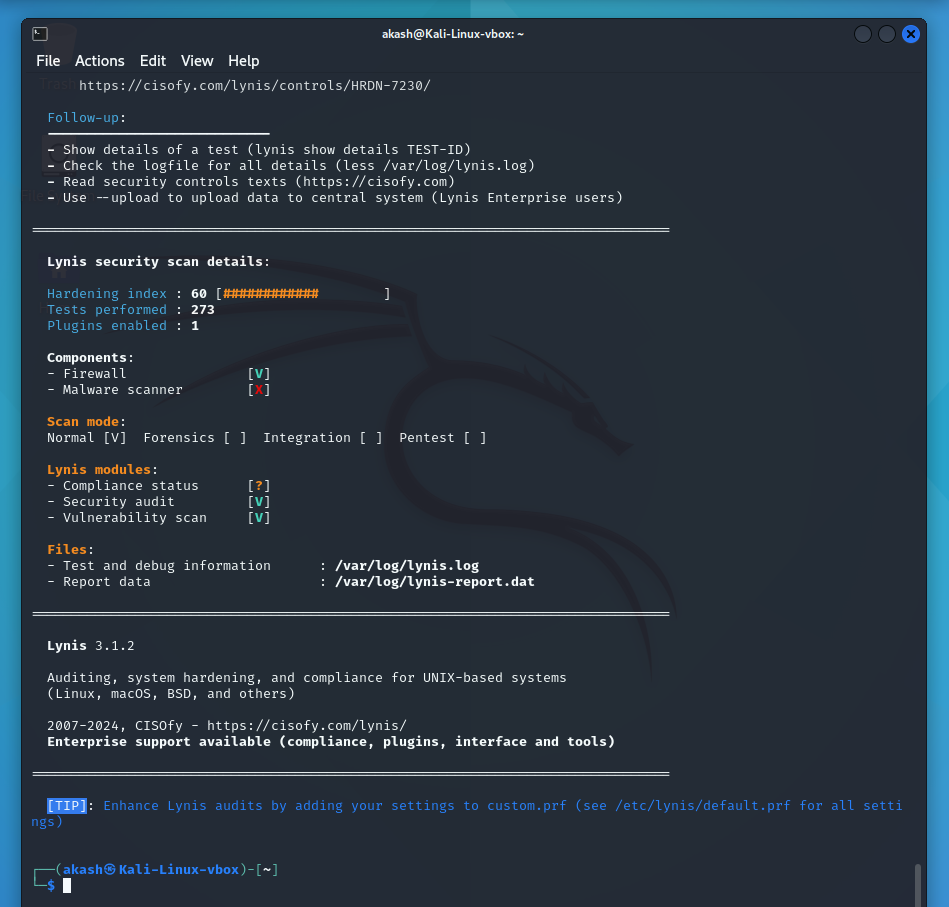


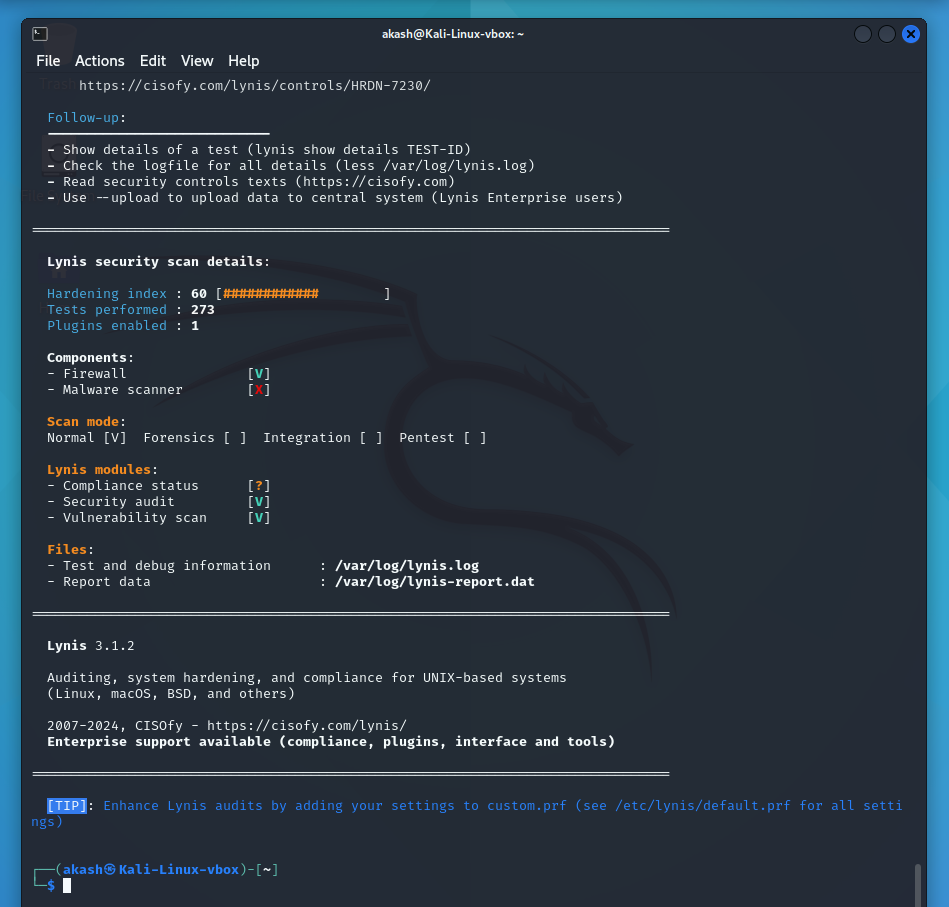












**INTERPRETING THE RESULTS**

**Boot and Services**

* systemd is used as the service manager.
* UEFI boot is disabled, reducing security benefits.
* GRUB2 lacks password protection, making it vulnerable.
* 18 services are running at boot; some (cron.service, dbus.service, lightdm.service, ssh.service) are marked UNSAFE.
* Protected services include polkit.service and systemd-timesyncd.service.

**Kernel Security**

* Kernel is up to date, but 90 active modules should be reviewed.
* Setuid core dumps are disabled.
* Some kernel parameters (kernel.kptr\_restrict, net.ipv4.conf.all.accept\_redirects) need adjustment for security.

**Memory and Processes**

* No zombie or IO waiting processes.
* Prelink tooling is absent but not critical.

**Ports and Packages**

* RPM and dpkg are functional; no unpurged packages.
* APT database is OK, but vulnerability checks are slow.
* Automatic updates are disabled.

**Networking**

* IPv6 is enabled.
* Only one nameserver detected; two recommended.
* Default gateway and DHCP settings are correct.
* No promiscuous mode detected.

**Security Framework**

* AppArmor is disabled.
* No SELinux, grsecurity, or MAC frameworks found.

**File Integrity and Permissions**

* dm-integrity and dm-verity are disabled.
* Files like /etc/crontab and /etc/ssh/sshd\_config need permission adjustments.
* Home directory permissions are misconfigured.

**Malware and IDS/IPS**

* No malware detection or IDS/IPS tools installed.

**Kernel Hardening**

* Some sysctl settings (kernel.kptr\_restrict, net.core.bpf\_jit\_harden) need adjustments.
* Others (fs.protected\_hardlinks, kernel.randomize\_va\_space) are properly configured.

Based on these results, the system administrator can apply security fixes to improve system security.

**Recommendations:**

* Enable UEFI boot and GRUB2 password protection.
* Secure unsafe services (cron, ssh, lightdm).
* Adjust kernel parameters for better security.
* Enable AppArmor and consider additional security frameworks.
* Install malware detection and IDS/IPS tools.
* Use at least two nameservers.
* Fix file permissions for sensitive system files and home directories.

Implementing these changes will enhance security.

**Step 4: Running a Specific Security Check**

Instead of performing a full audit, Lynis allows users to run specific security checks. For example, if the analyst wants to check only for malware and rootkits, the following command is used:

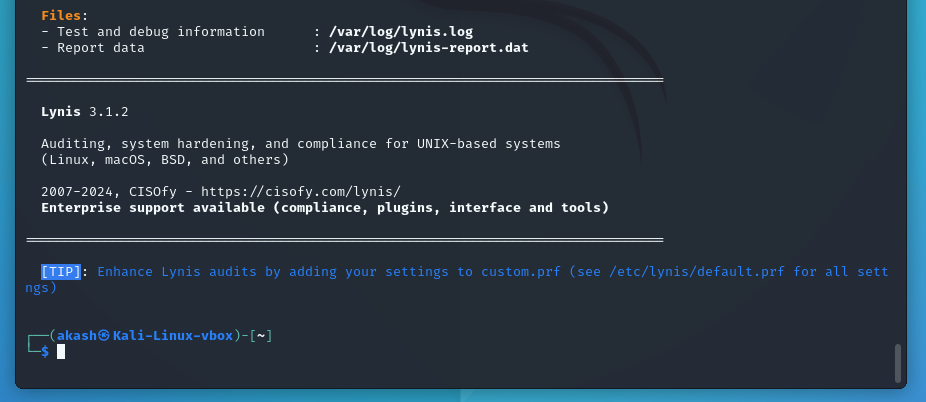
*sudo lynis audit system --tests-from-group malware*

This command focuses only on malware-related tests, reducing scan time while providing targeted insights.









**Summary of Scan Results:**

* Malware software components: NOT FOUND (No detected malware)
* Warnings: None (which is a good sign!)
* Suggestions (6):
  + Lynis version is outdated (Consider updating)
  + Install libpam-tmpdir (Helps set secure temp directories)
  + Install apt-listbugs (Shows critical bugs before installing packages)
  + Install apt-listchanges (Displays significant package changes before upgrades)
  + Install needrestart (Checks which services need restarting after updates)
  + Install fail2ban (Protects against brute-force attacks)

**Step 5: Exporting and Reviewing the Audit Report**

After performing a security audit with Lynis, it is crucial to save and analyze the findings for further investigation and future security improvements. The Lynis report contains valuable insights about system vulnerabilities, recommended security enhancements, and compliance status.

To save the audit results for further analysis, the analyst can export the report using:

*sudo lynis show report > lynis-security-report.txt*

This generates a text file (lynis-security-report.txt) containing all audit findings, allowing system administrators to track security improvements over time.

OR,

We can also use the following command:

# Copy the Lynis report to a more accessible location

*sudo cp /var/log/lynis-report.dat ~/lynis-security-report.txt*

# Change file ownership to the current user for easy access

*sudo chown $USER:$USER ~/lynis-security-report.txt*

**Why This Approach?**

**Direct redirection (> issue fix):**

* Running sudo lynis show report > lynis-security-report.txt may not work as expected because lynis show report displays the report interactively and does not output text in a structured format.
* Instead, copying the Lynis report file (/var/log/lynis-report.dat) ensures that all results are preserved.

**Making the file accessible:**

* By default, Lynis logs are stored in /var/log/, a directory that requires root permissions to access.
* Copying the file to the user's home directory (~/) and changing ownership (chown) allows a non-root user to read and analyze it without using sudo repeatedly.

**Viewing the Report**

Once the report is saved, we can review it using various methods:

* View the report in the terminal:

*cat ~/lynis-security-report.txt*

* Open it in a text editor (nano, vim, or less):

*nano ~/lynis-security-report.txt # Edit in nano*

*vim ~/lynis-security-report.txt # View in Vim*

*less ~/lynis-security-report.txt # Paginated view*

**Analyzing the Report**

After reviewing the report, system administrators can:

* Identify security gaps (e.g., outdated software, weak configurations).
* Follow Lynis recommendations to improve hardening scores.
* Track security enhancements over time by comparing previous audit reports.
* Export results for compliance reporting in security audits.

**Conclusion**

By following the step-by-step guide above, users can leverage Lynis to conduct comprehensive security audits, identify risks, and implement recommended security measures to strengthen their Linux systems. Regular audits with Lynis ensure that the system remains secure, hardened, and compliant with industry best practices.