

Lab 6.2

By: Nathan Metens (metens@pdx.edu)

1. Security auditing	1
2. Applying some suggestions	2
3. Thoughts	4

1. Security auditing

```
root@cs591-metens-kali-1: /home/metens
File Actions Edit View Help
Follow-up:
-----
- Show details of a test (lynis show details TEST-ID)
- Check the logfile for all details (less /var/log/lynis.log)
- Read security controls texts (https://cisofy.com)
- Use --upload to upload data to central system (Lynis Enterprise users)

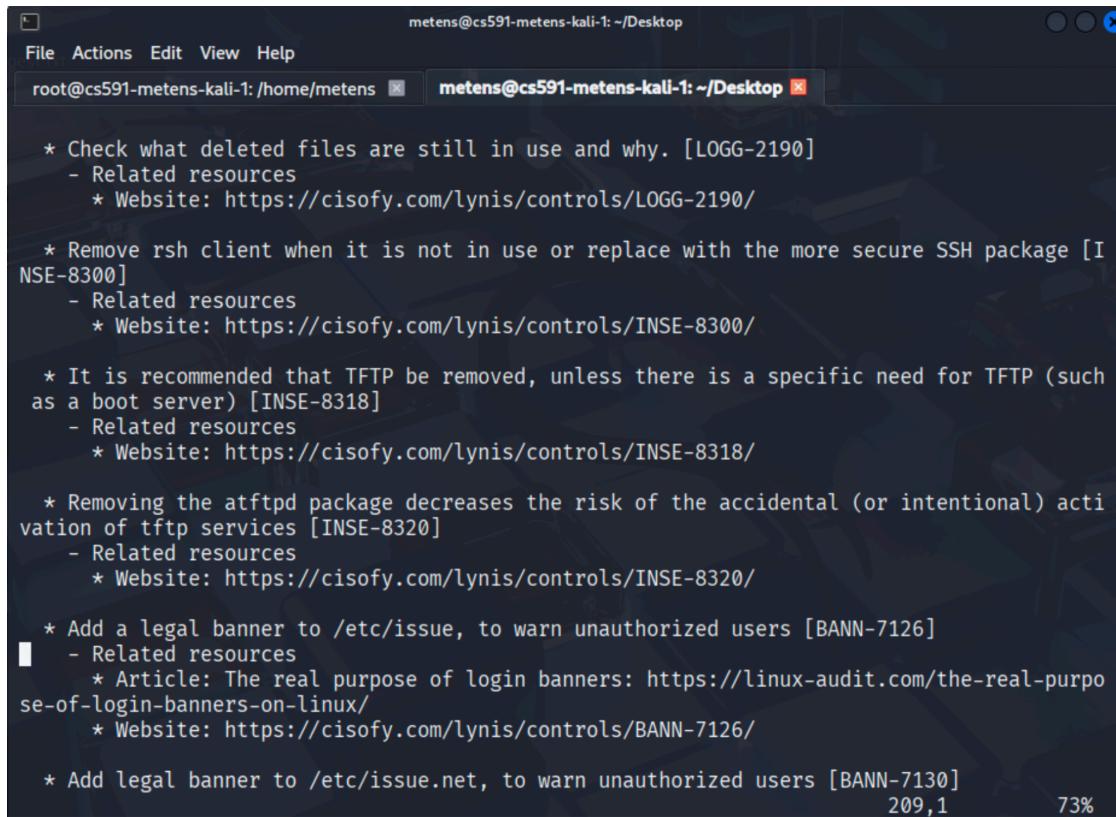
-----
Lynis security scan details:
Hardening index : 63 [#####
Tests performed : 277 ] ←
Plugins enabled : 1

Components:
- Firewall [V]
- Malware scanner [X]

Scan mode:
Normal [V] Forensics [ ] Integration [ ] Pentest [ ]
```

I searched the web for what the “Hardening Index” is and I found this: “The index gives the auditor an impression on how well a system is hardened. This number, however, is just an indicator on taken measures. One should not confuse it with a percentage of how “safe” a system might be.” ([source](#)).

I copied the suggestions into a text file:



```
* Check what deleted files are still in use and why. [LOGG-2190]
- Related resources
  * Website: https://cisofy.com/lynis/controls/LOGG-2190/

* Remove rsh client when it is not in use or replace with the more secure SSH package [INSE-8300]
- Related resources
  * Website: https://cisofy.com/lynis/controls/INSE-8300/

* It is recommended that TFTP be removed, unless there is a specific need for TFTP (such as a boot server) [INSE-8318]
- Related resources
  * Website: https://cisofy.com/lynis/controls/INSE-8318/

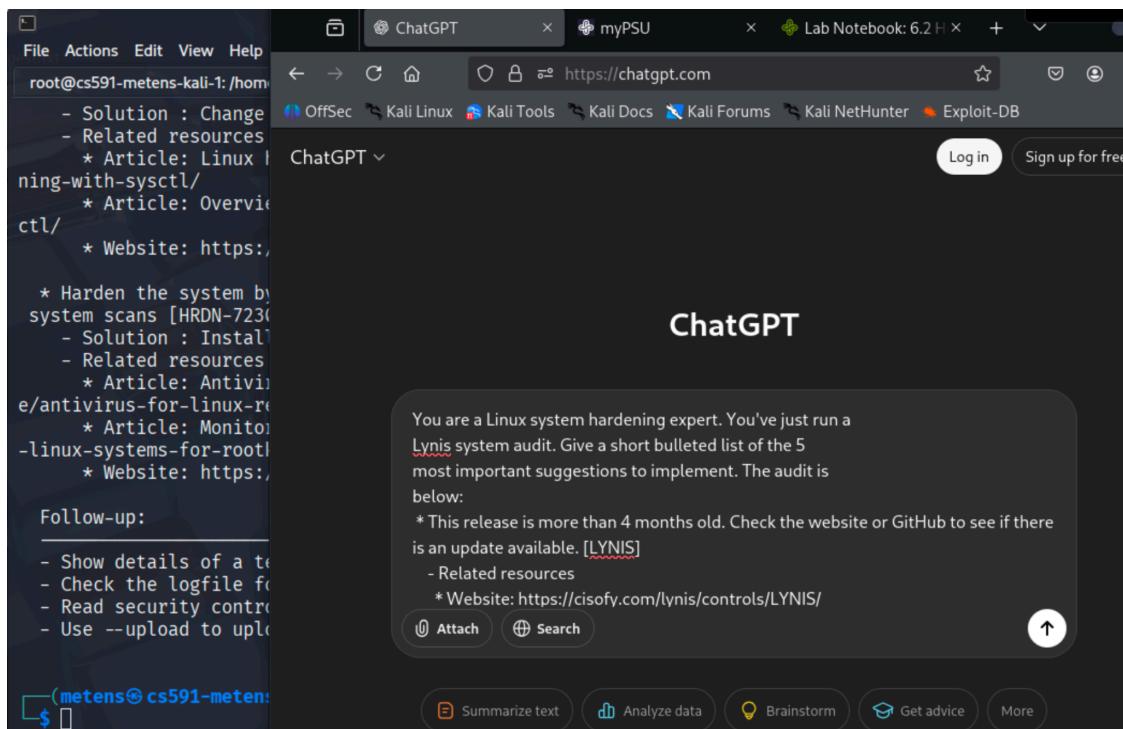
* Removing the atftpd package decreases the risk of the accidental (or intentional) activation of tftp services [INSE-8320]
- Related resources
  * Website: https://cisofy.com/lynis/controls/INSE-8320/

* Add a legal banner to /etc/issue, to warn unauthorized users [BANN-7126]
- Related resources
  * Article: The real purpose of login banners: https://linux-audit.com/the-real-purpose-of-login-banners-on-linux/
  * Website: https://cisofy.com/lynis/controls/BANN-7126/

* Add legal banner to /etc/issue.net, to warn unauthorized users [BANN-7130]
```

2. Applying some suggestions

Here, I ask ChatGPT to make a list of the 5 most important suggestions given the huge list I saved into a file earlier:



The screenshot shows a terminal window on the left and a ChatGPT interface on the right. The terminal window displays a portion of the Lynis audit report, specifically sections related to system hardening and rootkit detection. The ChatGPT interface has a message box containing the following text:

You are a Linux system hardening expert. You've just run a Lynis system audit. Give a short bulleted list of the 5 most important suggestions to implement. The audit is below:

* This release is more than 4 months old. Check the website or GitHub to see if there is an update available. [LYNIS]
- Related resources
 * Website: https://cisofy.com/lynis/controls/LYNIS/

The ChatGPT interface also includes standard AI interaction buttons like "Attach", "Search", and "Upvote".

Here are some of the suggestions ChatGPT says are important to implement:

The terminal window shows a list of security hardening steps:

- Solution : Change
- Related resources
 - * Article: Linux Hardening-with-sysctl/
 - * Article: Overview of sysctl/
 - * Website: https://
- * Harden the system by system scans [HRDN-7230]
 - Solution : Install
 - Related resources
 - * Article: Antivirus-for-linux-rootless/
 - * Article: Monitoring-linux-systems-for-rootless/
 - * Website: https://

Follow-up:

- Show details of a te
- Check the logfile fo
- Read security contro
- Use --upload to uplo

(metens@cs591-meten: ~)

The ChatGPT interface shows the following suggestions:

2. Install and Configure Fail2ban

Category: [DEB-0880]

Why it's critical: It actively defends against brute-force attacks by banning malicious IPs.

Action:

- `sudo apt install fail2ban`
- Configure for SSH and other exposed services

3. Set GRUB Password to Prevent Boot-Time Bypass

Category: [BOOT-5122]

Why it's critical: Prevents attackers with physical access from entering single-user mode and bypassing authentication.

Action:

- Set a password in GRUB using `grub-mkpasswd-pbkdf2`
- Add the hashed password to `/etc/grub.d/40_custom`

4. Enable and Configure `auditd` for System Auditing

Ask anything

↓

↑

The terminal window shows a list of security hardening steps, identical to the one above:

- Solution : Change
- Related resources
 - * Article: Linux Hardening-with-sysctl/
 - * Article: Overview of sysctl/
 - * Website: https://
- * Harden the system by system scans [HRDN-7230]
 - Solution : Install
 - Related resources
 - * Article: Antivirus-for-linux-rootless/
 - * Article: Monitoring-linux-systems-for-rootless/
 - * Website: https://

Follow-up:

- Show details of a te
- Check the logfile fo
- Read security contro
- Use --upload to uplo

(metens@cs591-meten: ~)

The ChatGPT interface shows the following suggestions:

4. Enable and Configure `auditd` for System Auditing

Category: [ACCT-9628]

Why it's critical: Tracks file access and system events; essential for incident detection and compliance.

Action:

- `sudo apt install auditd`
- Configure rules for monitoring critical files and authentication attempts

5. Install a File Integrity Monitoring Tool (e.g., AIDE or OSSEC)

Category: [FINT-4350]

Why it's critical: Detects unauthorized changes to critical system files—key for identifying breaches early.

Action:

- Install AIDE: `sudo apt install aide`
- Initialize and schedule regular integrity checks

Let me know if you want a prioritized script or Ansible playbook to apply these automatically.

↓

↑

It suggested I install AIDE for file integrity monitoring: `sudo apt install aide`, which will detect unauthorized changes to critical system files. It also suggests installing and configuring Fail2ban: `sudo apt install fail2ban`, which helps prevent malicious IPs from brute force attacking.

Then I asked for the 5 easiest suggestions given the same audit, here were the results:

The terminal window shows a Lynis audit report for a Kali Linux system. The report lists various findings and recommendations. The ChatGPT interface is overlaid, asking for the 5 easiest suggestions to implement from the audit. The suggestions listed are:

- 1. Install `needrestart` to detect services needing a restart after updates**
Category: [DEB-0831]
 - Command: `sudo apt install needrestart`
 - Helps identify outdated services after upgrades—no config needed.
- 2. Install `apt-listbugs` to show critical bugs before package installation**
Category: [DEB-0810]
 - Command: `sudo apt install apt-listbugs`
 - Adds protection from broken updates via warnings.

The terminal window shows a Lynis audit report for a Kali Linux system. The report lists various findings and recommendations. The ChatGPT interface is overlaid, asking for the 5 easiest suggestions to implement from the audit. The suggestions listed are:

- 3. Install `apt-listchanges` to show changelogs before upgrades**
Category: [DEB-0811]
 - Command: `sudo apt install apt-listchanges`
 - Helps admins stay informed on package changes—zero config required.
- 4. Add legal login banners to `/etc/issue` and `/etc/issue.net`**
Categories: [BANN-7126] and [BANN-7130]
 - Commands:

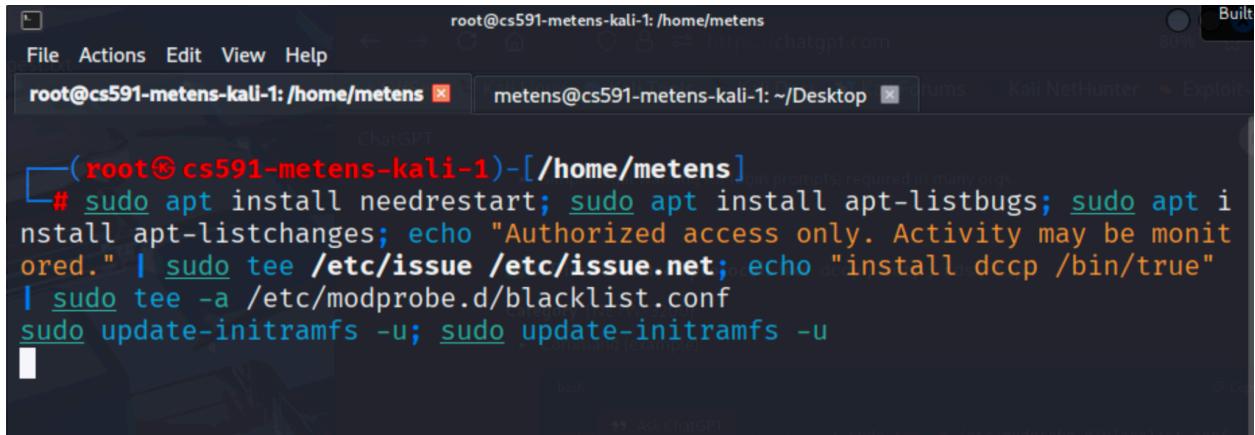
```
bash
echo "Authorized access only. Activity may be monitored." | sudo tee /etc/issue /etc/issue.net
```
 - Simple legal warning for login prompts; required in many orgs.
- 5. Disable unused protocols like `dccp`, `sctp`, `rds`, `tipc`**
Category: [NETW-3200]
 - Command (example):

```
sudo modprobe -r dccp sctp rds tipc
```

All in all, the 5 easiest suggestions were:

1. `sudo apt install nerdrestart`.
2. `sudo apt install apt-listbugs`
3. `sudo apt install apt-listchanges`
4. Add legal login banners in `/etc/issue` and `/etc/[issue.net](#)`
5. Disable unused protocols

I performed all of these easy suggestions by copying and pasting them in one line on the command line and then re-running the lynus audit.

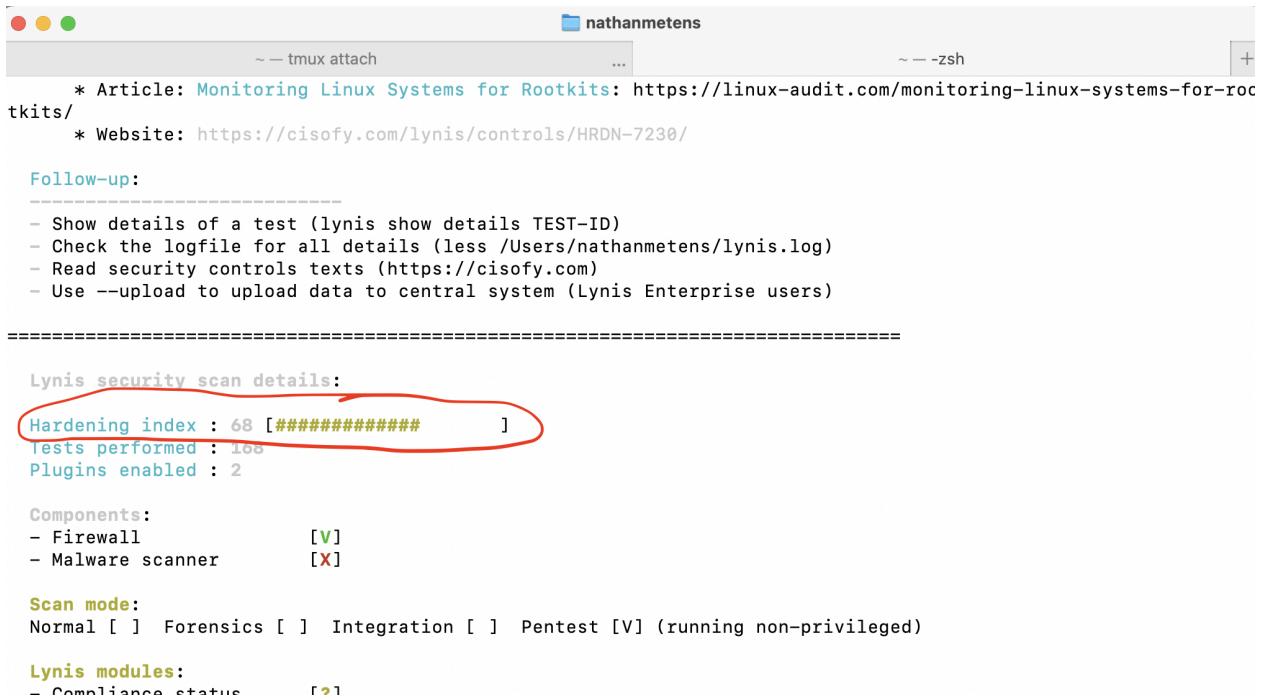


The screenshot shows a terminal window titled "root@cs591-metens-kali-1: /home/metens". The user has run several commands to prepare for the audit:

```
(root@cs591-metens-kali-1)-[~/home/metens]
# sudo apt install needrestart; sudo apt install apt-listbugs; sudo apt i
nstall apt-listchanges; echo "Authorized access only. Activity may be monit
ored." | sudo tee /etc/issue /etc/issue.net; echo "install dccp /bin/true"
| sudo tee -a /etc/modprobe.d/blacklist.conf
sudo update-initramfs -u; sudo update-initramfs -u
```

*** Well, right after you left, the Kali Linux internet connection dropped again, and I couldn't seem to fix it for the life of me. So I went back and redid all the steps on my MacBook. ***

Here is the original hardening index on my Mac:



The screenshot shows the Lynis security scan results on a Mac. The "Hardening index" is highlighted with a red oval:

```
nathanmetens
* Article: Monitoring Linux Systems for Rootkits: https://linux-audit.com/monitoring-linux-systems-for-rootkits/
* Website: https://cisofy.com/lynis/controls/HRDN-7230/

Follow-up:
- Show details of a test (lynis show details TEST-ID)
- Check the logfile for all details (less /Users/nathanmetens/lynis.log)
- Read security controls texts (https://cisofy.com)
- Use --upload to upload data to central system (Lynis Enterprise users)

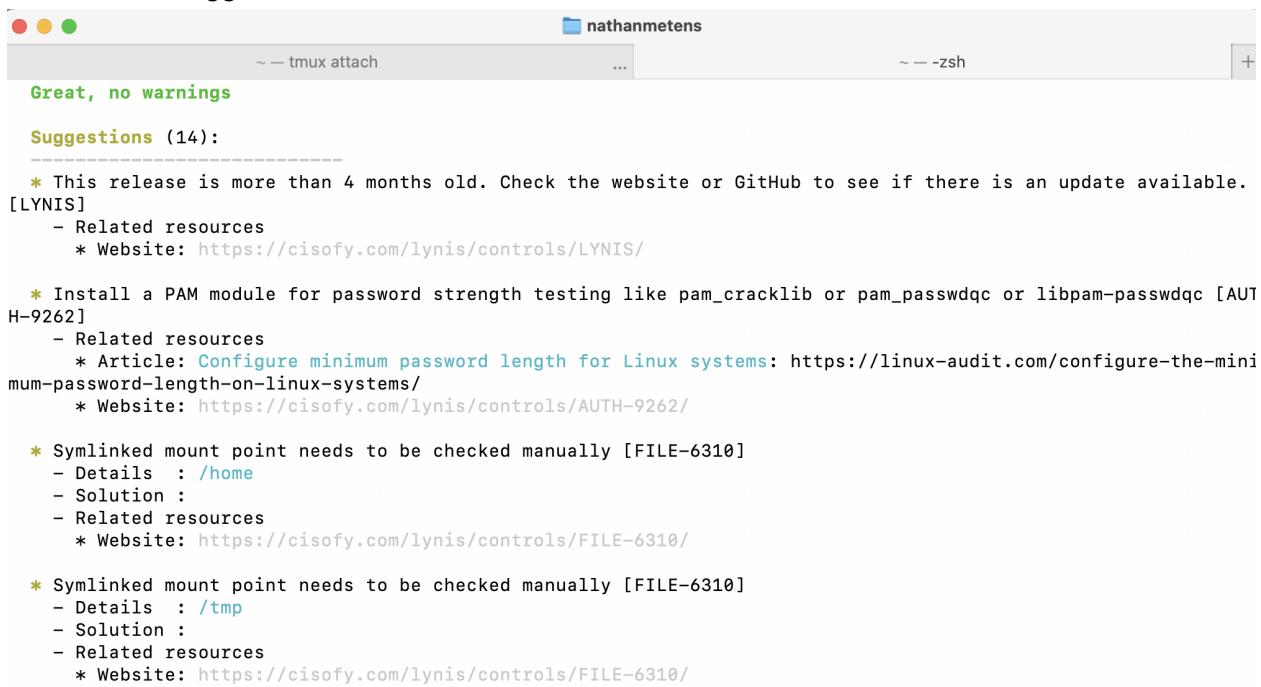
=====
Lynis security scan details:
Hardening index : 68 [#####
] Tests performed : 100 Plugins enabled : 2

Components:
- Firewall [V]
- Malware scanner [X]

Scan mode:
Normal [ ] Forensics [ ] Integration [ ] Pentest [V] (running non-privileged)

Lynis modules:
- Compliance status
```

Here are the suggestions:



Great, no warnings

Suggestions (14):

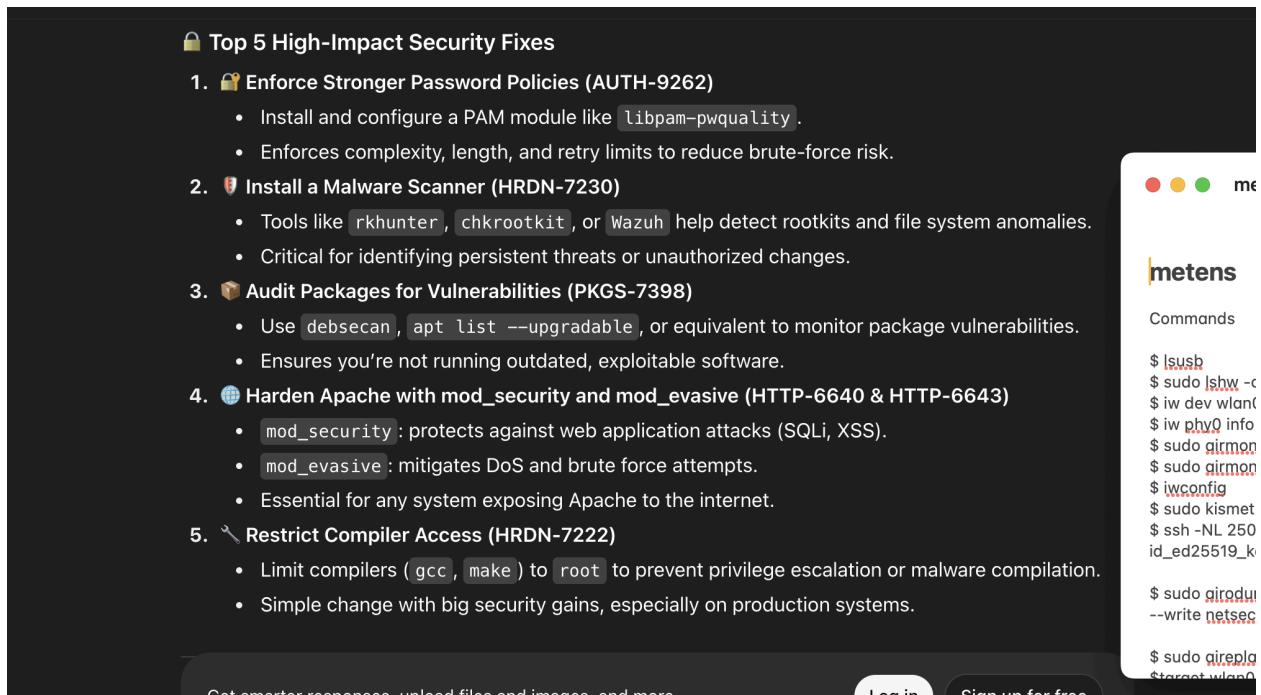
```
* This release is more than 4 months old. Check the website or GitHub to see if there is an update available. [LYNIS]
- Related resources
  * Website: https://cisofy.com/lynis/controls/LYNIS/

* Install a PAM module for password strength testing like pam_cracklib or pam_passewdqc or libpam-passwdqc [AUTH-9262]
- Related resources
  * Article: Configure minimum password length for Linux systems: https://linux-audit.com/configure-the-minimum-password-length-on-linux-systems/
  * Website: https://cisofy.com/lynis/controls/AUTH-9262/

* Symlinked mount point needs to be checked manually [FILE-6310]
- Details : /home
- Solution :
- Related resources
  * Website: https://cisofy.com/lynis/controls\(FILE-6310/

* Symlinked mount point needs to be checked manually [FILE-6310]
- Details : /tmp
- Solution :
- Related resources
  * Website: https://cisofy.com/lynis/controls\(FILE-6310/
```

Here is what ChatGPT suggests are the most important fixes given the prompt:



Top 5 High-Impact Security Fixes

- Enforce Stronger Password Policies (AUTH-9262)
 - Install and configure a PAM module like libpam-pwquality.
 - Enforces complexity, length, and retry limits to reduce brute-force risk.
- Install a Malware Scanner (HRDN-7230)
 - Tools like rkHunter, chkrootkit, or Wazuh help detect rootkits and file system anomalies.
 - Critical for identifying persistent threats or unauthorized changes.
- Audit Packages for Vulnerabilities (PKGS-7398)
 - Use debsecan, apt list --upgradable, or equivalent to monitor package vulnerabilities.
 - Ensures you're not running outdated, exploitable software.
- Harden Apache with mod_security and mod_evasive (HTTP-6640 & HTTP-6643)
 - mod_security : protects against web application attacks (SQLi, XSS).
 - mod_evasive : mitigates DoS and brute force attempts.
 - Essential for any system exposing Apache to the internet.
- Restrict Compiler Access (HRDN-7222)
 - Limit compilers (gcc, make) to root to prevent privilege escalation or malware compilation.
 - Simple change with big security gains, especially on production systems.

Here are the top 5 easiest suggestions:

Great question. From your Lynis audit, here are the easiest and quickest improvements you can implement—requiring minimal configuration, technical complexity, or risk.

⚡ Top 5 Easiest Hardening Suggestions to Implement

1. 📡 Update the System & Lynis Tool (LYNIS)

- Keep the OS and Lynis up to date.
- Commands:

```
bash                                     ⌂ Copy ⌂ Edit  
sudo apt update && sudo apt upgrade      # Debian/Ubuntu  
sudo yum update                          # RHEL/CentOS  
sudo lynis update info
```

2. 📡 Add Hostname and FQDN to /etc/hosts (NAME-4404)

- Helps ensure reliable local name resolution.
- Edit /etc/hosts :

```
bash                                     ⌂ Copy ⌂ Edit  
sudo nano /etc/hosts
```

Add a line like:

```
127.0.1.1   yourhostname.yourdomain.com   yourhostname
```

3. 🔎 Check for Deleted Files in Use (LOGG-2190)

- Useful for cleanup or troubleshooting.
- Command:

```
bash                                     ⌂ Copy ⌂ Edit  
sudo lsof | grep deleted
```

4. 📡 Check and Fix Home Directory Permissions (HOME-9304)

- Prevents users from accidentally exposing private data.
- Command:

```
bash                                     ⌂ Copy ⌂ Edit  
sudo chmod 700 /home/username
```

5. 🛡 Restrict Access to Compilers (HRDN-7222)

- Prevent non-root users from compiling or executing binaries.
- Commands:

```
bash                                     ⌂ Copy ⌂ Edit  
sudo chmod 700 /usr/bin/gcc  
sudo chmod 700 /usr/bin/make
```

These steps are low-effort, but contribute meaningfully to tightening security—great for a quick win or the start of a hardening checklist.



metens

Commands

\$ lsusb
\$ sudo lshw
\$ iw dev wlan
\$ iw phy0 info
\$ sudo airmc
\$ sudo airmc
\$ iwconfig
\$ sudo kismet
\$ ssh -NL 25
id_ed25519_

\$ sudo airod
--write netse

I converted these steps for Mac instead of Linux:

⚡ Top 5 Easiest macOS Hardening Suggestions (Lynis-style)

1. 📁 Ensure Proper Local Name Resolution

Lynis Ref: [NAME-4404]

- Action: Add your hostname and FQDN to `/etc/hosts` (useful for local services and dev environments).
- Steps:

```
bash
sudo nano /etc/hosts
```

Add a line like:

```
bash
127.0.0.1 Nathans-MacBook-Air.local Nathans-MacBook-Air
```

(Replace with your actual hostname if different — you can get it with `scutil --get HostName`)

2. 🔎 Check for Deleted Files Still in Use

Lynis Ref: [LOGG-2190]

- macOS Equivalent: See what files are deleted but still held open (like logs by a running process).
- Command:

```
bash
lsof | grep deleted
```

3. 🛡 Lock Down Home Directory Permissions

Lynis Ref: [HOME-9304]

- Action: Restrict permissions on your home directory so others can't snoop (especially on multi-user Macs).
- Command:

```
bash
chmod 700 ~
```

4. 🔒 Restrict Compiler Access (Advanced Option)

Lynis Ref: [HRDN-7222]

- macOS Note: You can restrict access to tools like `clang` if others use your Mac.
- Commands (be cautious — only do this if you understand the impact):

```
bash
sudo chmod 700 /usr/bin/clang
sudo chmod 700 /usr/bin/gcc
```
- You can undo it with:

```
bash
sudo chmod 755 /usr/bin/clang
```

5. 📈 Keep Homebrew & System Updated

Lynis Ref Equivalent: [LYNIS] (keep system and tools up to date)

- Command:

```
bash
brew update && brew upgrade
softwareupdate --all --install --force
```

✓ Bonus Tip: Run a Basic Security Audit on macOS

macOS has a built-in security audit tool:

```
bash
sudo system_profiler SPConfigurationProfileDataType
```

Or consider using:

```
bash
# (Optional) Run a basic security audit
sudo system_profiler SPConfigurationProfileDataType
```



metens

Commands

```
$ lsusb
$ sudo lshw -c
$ iw dev wlan0
$ iw phy0 info
$ sudo girmon
$ sudo girmon
$ iwconfig
$ sudo kismet
$ ssh -NL 250:id_ed25519_ko
```

```
$ sudo airodump
--write netsec
```

```
$ sudo aireplay
$ target wlan0
```

I tried performing a lot of them, but most of them said permission denied, even though I was using sudo. After another audit, here is the updated hardening index:

```
██████████ nathanmetens
~ — tmux attach ... ~ — -zsh
- Read security controls texts (https://cisofy.com)
- Use --upload to upload data to central system (Lynis Enterprise users)

=====
Lynis security scan details:
Hardening index : 72 [#####]
Tests performed : 119
Plugins enabled : 2

Components:
- Firewall [V]
```

It increased by 4 points, which is nice. I feel safer now.

Here is what ChatGPT said after I showed the updated suggestions: There were 13 now instead of 14:

```
https://linux-audit.com/software/why-remove-compilers-
from-your-system/
* Website: https://cisofy.com/lynis/controls/HRDN-7222/
```

Thanks for sharing the updated audit output! It looks like Lynis is still reporting many of the same suggestions, which usually means the issues were **not fully resolved** or Lynis detected the underlying condition remains.

Let's review key points to verify if your commands worked or not:

1. This release is more than 4 months old

- Did you update macOS with `softwareupdate` and brew packages with `brew upgrade`?
- Check macOS version: `sw_vers`
- Check brew package updates: `brew outdated`

If not updated recently, this will persist.

2. Install PAM module for password strength testing

- On macOS, `pwpolicy` settings apply.
- Did you run something like:

Ask anything

metens

Commands

```
$ lsusb
$ sudo lshw -
$ iw dev wlan0
$ iw phy0 info
$ sudo airmesh
$ sudo airmesh
$ iwconfig
$ sudo kismet
$ ssh -NL 25
id_ed25519_
$ sudo airodump
--write netse
$ sudo airodump
$ target_wlan
```

3. Thoughts

The results I got from asking ChatGPT to tell me what to fix and how to fix it were not extremely helpful. Although it did help increase the hardening score by 4 points, it took me an hour to get this done. I need to continually ask ChatGPT to convert to macOS and to give me a different answer after I run a command that fails. Overall, I think ChatGPT is a good guide, but it isn't always the best. If I wanted to, I could have read through the 14 suggestions and searched the web for details on how to do them myself. ChatGPT was a guide, not the answer.