

## Finding: Missing password strength module

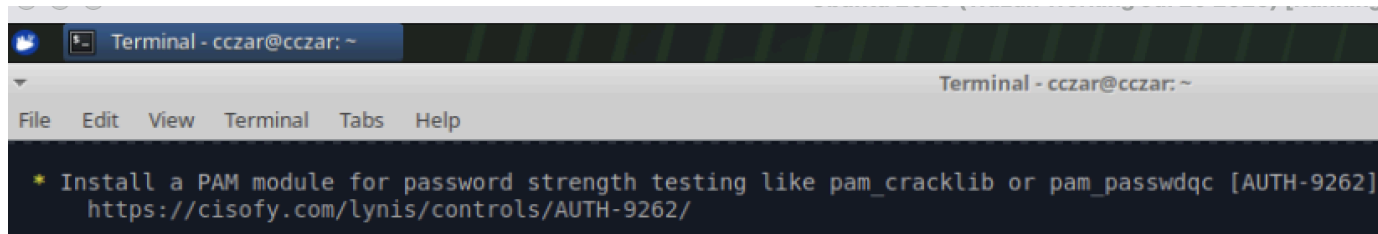
**NIST Control Mapping:** PR.AA-03 - Users, services, and hardware are authenticated

**Risk:** Weak passwords can compromise accounts.

**Severity:** High

**Remediation:** Implement a password strength module to enforce complexity requirements, including minimum length, a mix of uppercase and lowercase letters, numbers, and special characters.

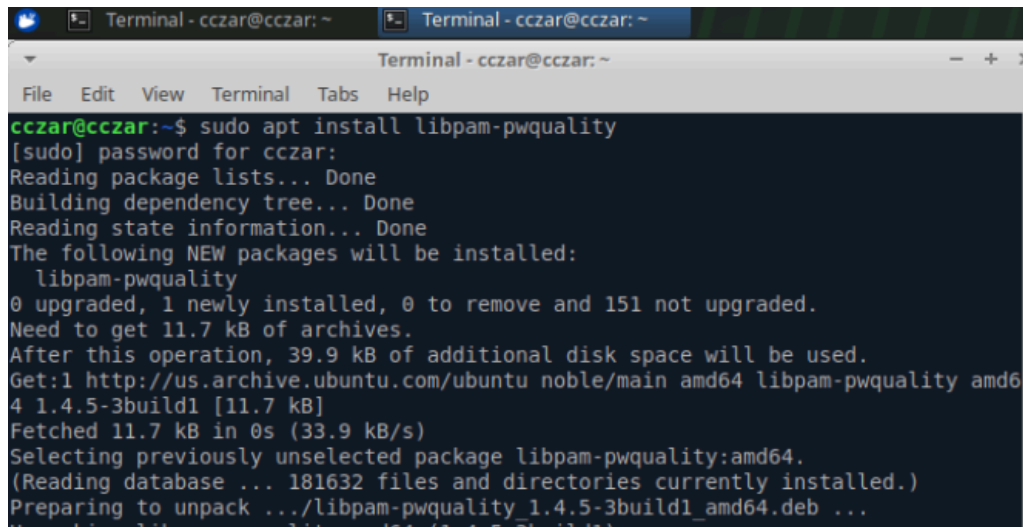
## INITIAL LYNIS SCAN RESULT



```
Terminal - cczar@cczar: ~  
File Edit View Terminal Tabs Help  
* Install a PAM module for password strength testing like pam_cracklib or pam_passwdqc [AUTH-9262]  
https://cisofy.com/lynis/controls/AUTH-9262/
```

## REMEDIATION PROCESS

### 1. Installed `pam_pwquality`



```
Terminal - cczar@cczar: ~  
File Edit View Terminal Tabs Help  
cczar@cczar:~$ sudo apt install libpam-pwquality  
[sudo] password for cczar:  
Reading package lists... Done  
Building dependency tree... Done  
Reading state information... Done  
The following NEW packages will be installed:  
  libpam-pwquality  
0 upgraded, 1 newly installed, 0 to remove and 151 not upgraded.  
Need to get 11.7 kB of archives.  
After this operation, 39.9 kB of additional disk space will be used.  
Get:1 http://us.archive.ubuntu.com/ubuntu noble/main amd64 libpam-pwquality amd64 1.4.5-3build1 [11.7 kB]  
Fetched 11.7 kB in 0s (33.9 kB/s)  
Selecting previously unselected package libpam-pwquality:amd64.  
(Reading database ... 181632 files and directories currently installed.)  
Preparing to unpack .../libpam-pwquality_1.4.5-3build1_amd64.deb ...  
Unpacking libpam-pwquality:amd64 (1.4.5-3build1)
```

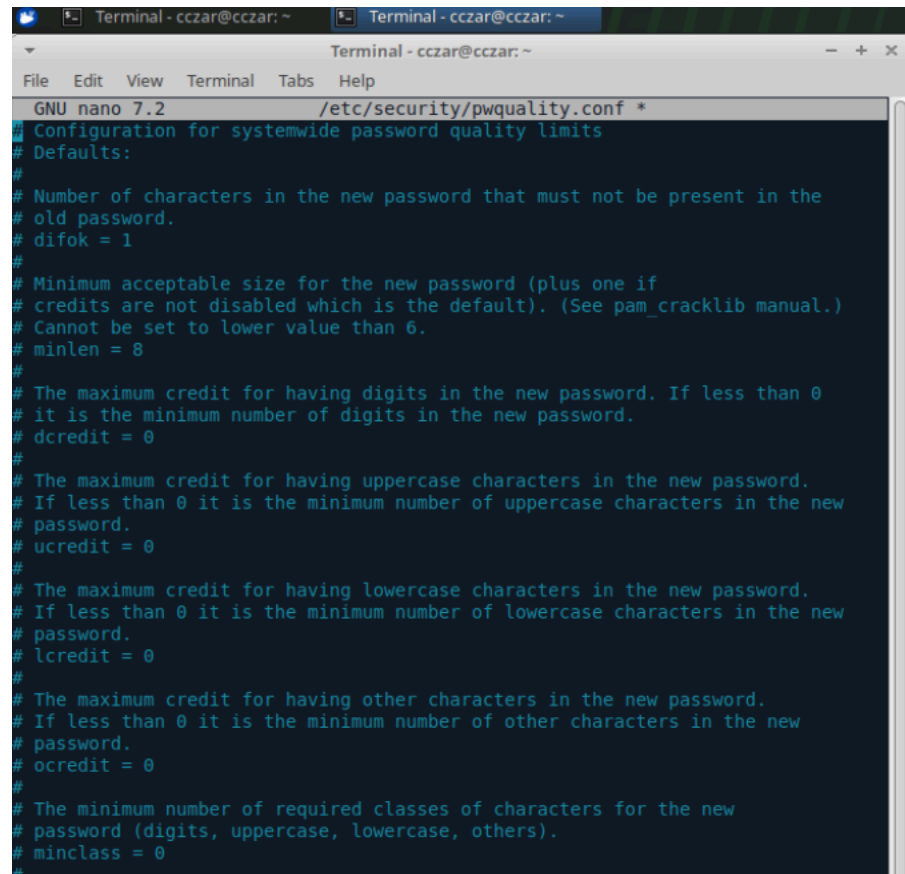
## Note on Password Quality Modules:

During remediation, `pam_pwquality` was installed and configured to enforce strong password policies. Although `pam_passwdqc` is an alternative PAM module for password quality, `pam_pwquality` was chosen for the following reasons:

- Modern and actively maintained – `pam_pwquality` is the preferred module on current Linux distributions, including Ubuntu 22.04 LTS.
- Centralized configuration – settings are stored in `/etc/security/pwquality.conf`, allowing clear control of password length, character classes, and differences from previous passwords.
- Ease of enforcement – integrates seamlessly with PAM (`/etc/pam.d/common-password`) to enforce policies system-wide.
- Legacy module – `pam_passwdqc` is primarily used for older or legacy systems

## 2. Configured `/etc/security/pwquality.conf`

Initial configuration:



```
Terminal - cczar@cczar: ~
GNU nano 7.2 /etc/security/pwquality.conf *
# Configuration for systemwide password quality limits
# Defaults:
#
# Number of characters in the new password that must not be present in the
# old password.
# difok = 1
#
# Minimum acceptable size for the new password (plus one if
# credits are not disabled which is the default). (See pam_cracklib manual.)
# Cannot be set to lower value than 6.
# minlen = 8
#
# The maximum credit for having digits in the new password. If less than 0
# it is the minimum number of digits in the new password.
# dcredit = 0
#
# The maximum credit for having uppercase characters in the new password.
# If less than 0 it is the minimum number of uppercase characters in the new
# password.
# ucredit = 0
#
# The maximum credit for having lowercase characters in the new password.
# If less than 0 it is the minimum number of lowercase characters in the new
# password.
# lcredit = 0
#
# The maximum credit for having other characters in the new password.
# If less than 0 it is the minimum number of other characters in the new
# password.
# ocredit = 0
#
# The minimum number of required classes of characters for the new
# password (digits, uppercase, lowercase, others).
# minclass = 0
#
```

The following configurations were updated:

**difok = 5**

- The new password must differ by at least **5 characters** from the previous one.

**minlen = 12**

- Minimum password length = **12 characters**.

**ucredit = -1**

- Requires at least **1 uppercase letter**.
- Negative values mean “at least this many must exist.”

**lcredit = -1**

- Requires at least **1 lowercase letter**.

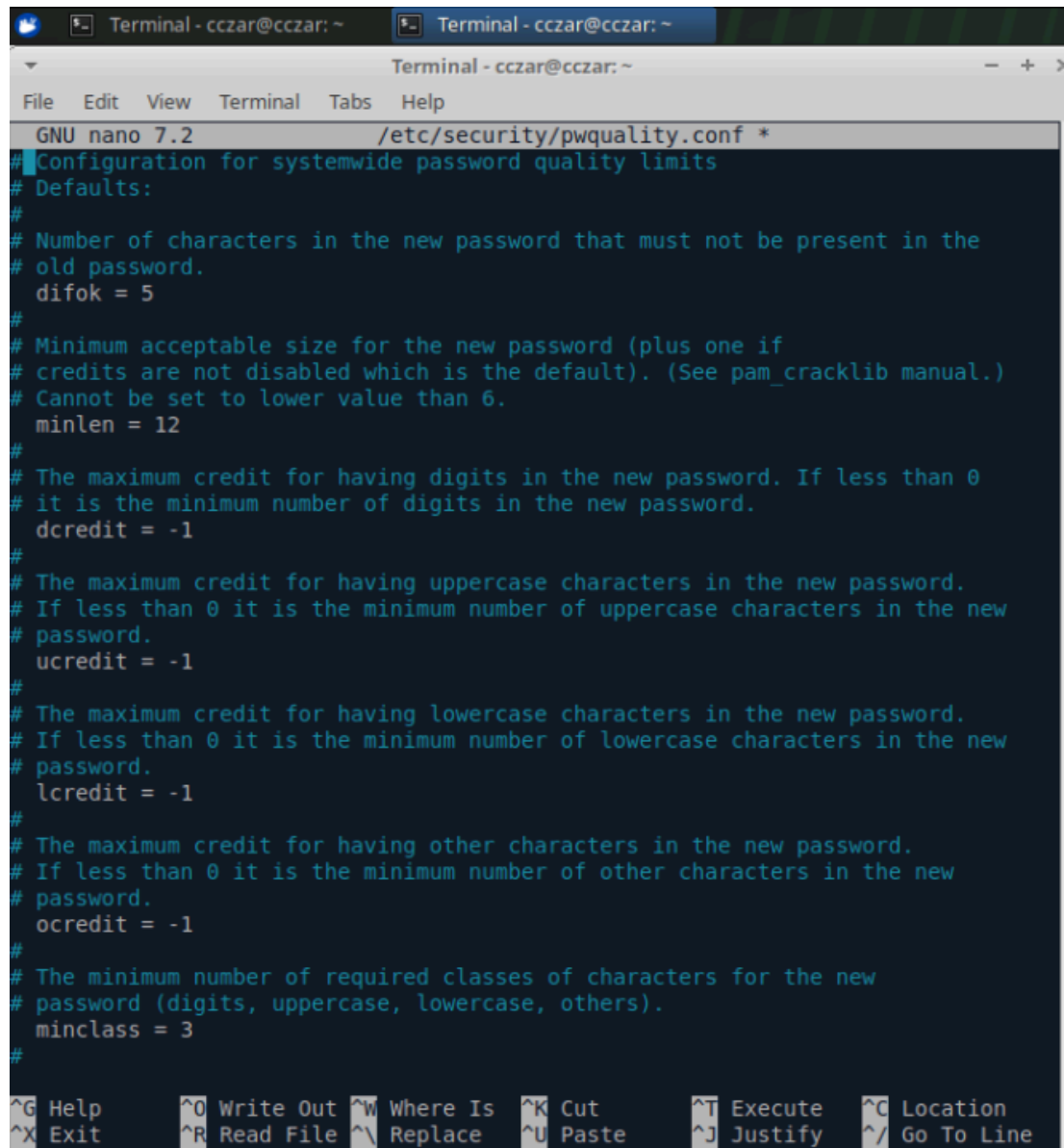
**dcredit = -1**

- Requires at least **1 digit**.

**ocredit = -1**

- Requires at least **1 special character** (non-letter, non-digit).

## Secure updated configuration:



The image shows a terminal window with two tabs, both titled "Terminal - cczar@cczar: ~". The active tab displays the GNU nano 7.2 editor editing the file /etc/security/pwquality.conf. The file contains configuration for systemwide password quality limits. The configuration includes several commented-out lines and active settings: difok = 5, minlen = 12, dcredit = -1, ucredit = -1, lcredit = -1, ocredit = -1, and minclass = 3. The bottom of the terminal shows a status bar with various keyboard shortcuts for nano, such as ^G Help, ^O Write Out, ^W Where Is, ^K Cut, ^T Execute, ^C Location, ^X Exit, ^R Read File, ^\ Replace, ^U Paste, ^J Justify, and ^\_ Go To Line.

```
GNU nano 7.2 /etc/security/pwquality.conf *
# Configuration for systemwide password quality limits
# Defaults:
#
# Number of characters in the new password that must not be present in the
# old password.
# difok = 5
#
# Minimum acceptable size for the new password (plus one if
# credits are not disabled which is the default). (See pam_cracklib manual.)
# Cannot be set to lower value than 6.
# minlen = 12
#
# The maximum credit for having digits in the new password. If less than 0
# it is the minimum number of digits in the new password.
# dcredit = -1
#
# The maximum credit for having uppercase characters in the new password.
# If less than 0 it is the minimum number of uppercase characters in the new
# password.
# ucredit = -1
#
# The maximum credit for having lowercase characters in the new password.
# If less than 0 it is the minimum number of lowercase characters in the new
# password.
# lcredit = -1
#
# The maximum credit for having other characters in the new password.
# If less than 0 it is the minimum number of other characters in the new
# password.
# ocredit = -1
#
# The minimum number of required classes of characters for the new
# password (digits, uppercase, lowercase, others).
# minclass = 3
#
^G Help  ^O Write Out  ^W Where Is  ^K Cut       ^T Execute   ^C Location
^X Exit  ^R Read File  ^\ Replace   ^U Paste     ^J Justify   ^_ Go To Line
```

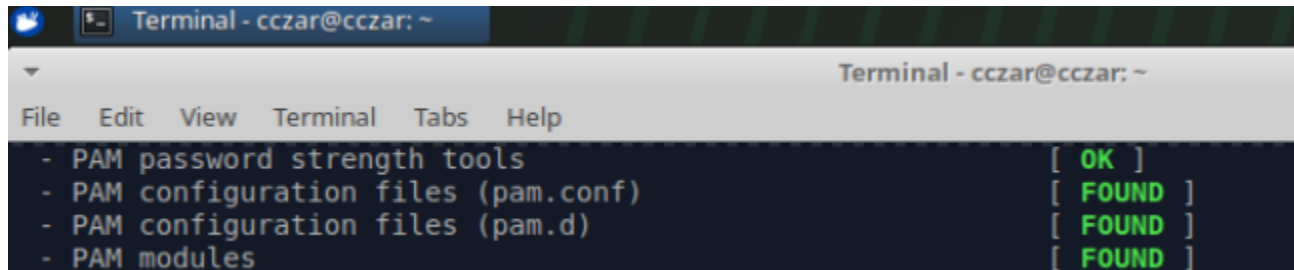
### 3. Validated configuration updates

- Attempted to set weak passwords 'testpass123!', Testpass1234, Testpass12!' → rejected, confirming enforcement.
- Successfully set password with required parameters.

```
cczar@cczar:~$ sudo nano /etc/security/pwquality.conf
[sudo] password for cczar:
cczar@cczar:~$ passwd cczar
Changing password for cczar.
Current password:
New password:
BAD PASSWORD: The password contains less than 1 uppercase letters
New password:
BAD PASSWORD: The password contains less than 1 non-alphanumeric characters
New password:
BAD PASSWORD: The password is shorter than 12 characters
passwd: Have exhausted maximum number of retries for service
passwd: password unchanged
cczar@cczar:~$
```

```
cczar@cczar:~$ passwd cczar
Changing password for cczar.
Current password:
New password:
Retype new password:
Sorry, passwords do not match.
New password:
Retype new password:
passwd: password updated successfully
cczar@cczar:~$
```

## LYNIS SCAN RESULT AFTER REMEDIATION



A terminal window titled "Terminal - cczar@cczar: ~" displays the output of a Lynis scan. The window has a menu bar with "File", "Edit", "View", "Terminal", "Tabs", and "Help". The scan results are listed as follows:

```
- PAM password strength tools [ OK ]  
- PAM configuration files (pam.conf) [ FOUND ]  
- PAM configuration files (pam.d) [ FOUND ]  
- PAM modules [ FOUND ]
```

## Finding: Vulnerable Packages (PKGS-7392)

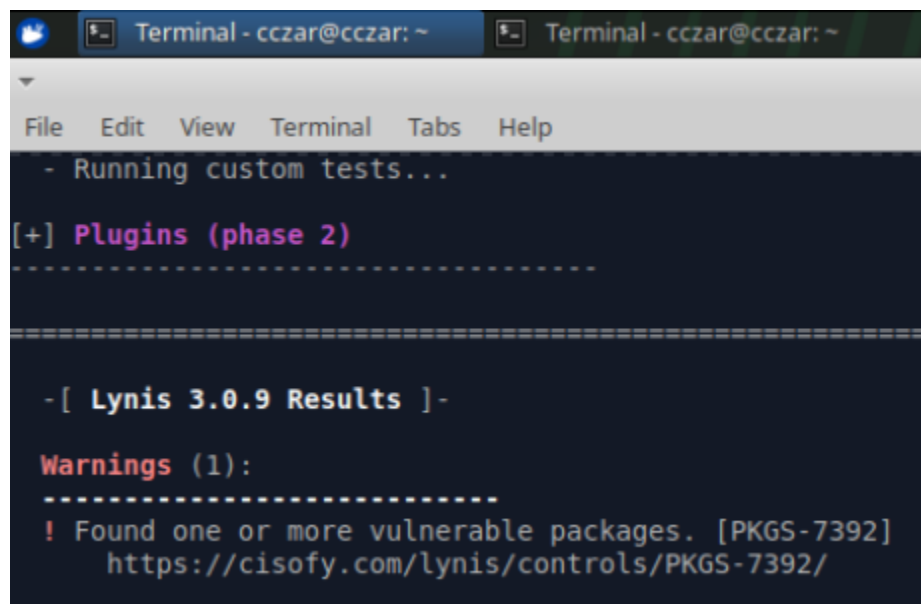
**NIST Control Mapping:** PR.PS-02 - Software is maintained, replaced, and removed commensurate with risk

**Risk:** Outdated packages increase the likelihood of compromise, as known vulnerabilities could be exploited.

**Severity:** High

**Remediation:** Identify and update all vulnerable packages to the latest stable versions.

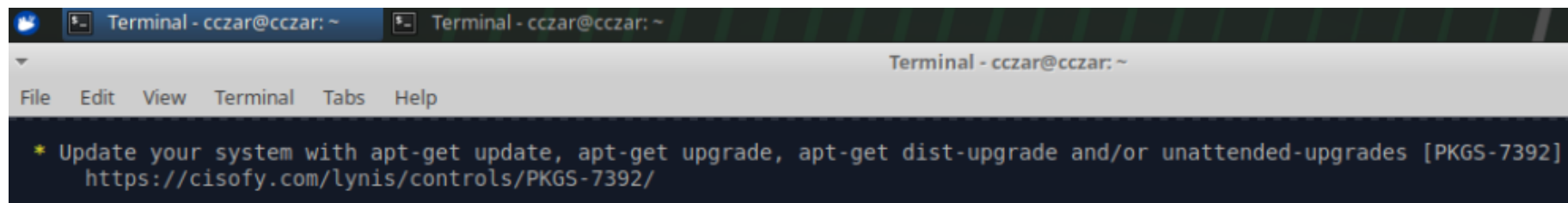
### INITIAL LYNIS SCAN RESULT



```
Terminal - cczar@cczar: ~
File Edit View Terminal Tabs Help
- Running custom tests...
[+] Plugins (phase 2)
-----

-[ Lynis 3.0.9 Results ]-

Warnings (1):
-----
! Found one or more vulnerable packages. [PKGS-7392]
  https://cisofy.com/lynis/controls/PKGS-7392/
```



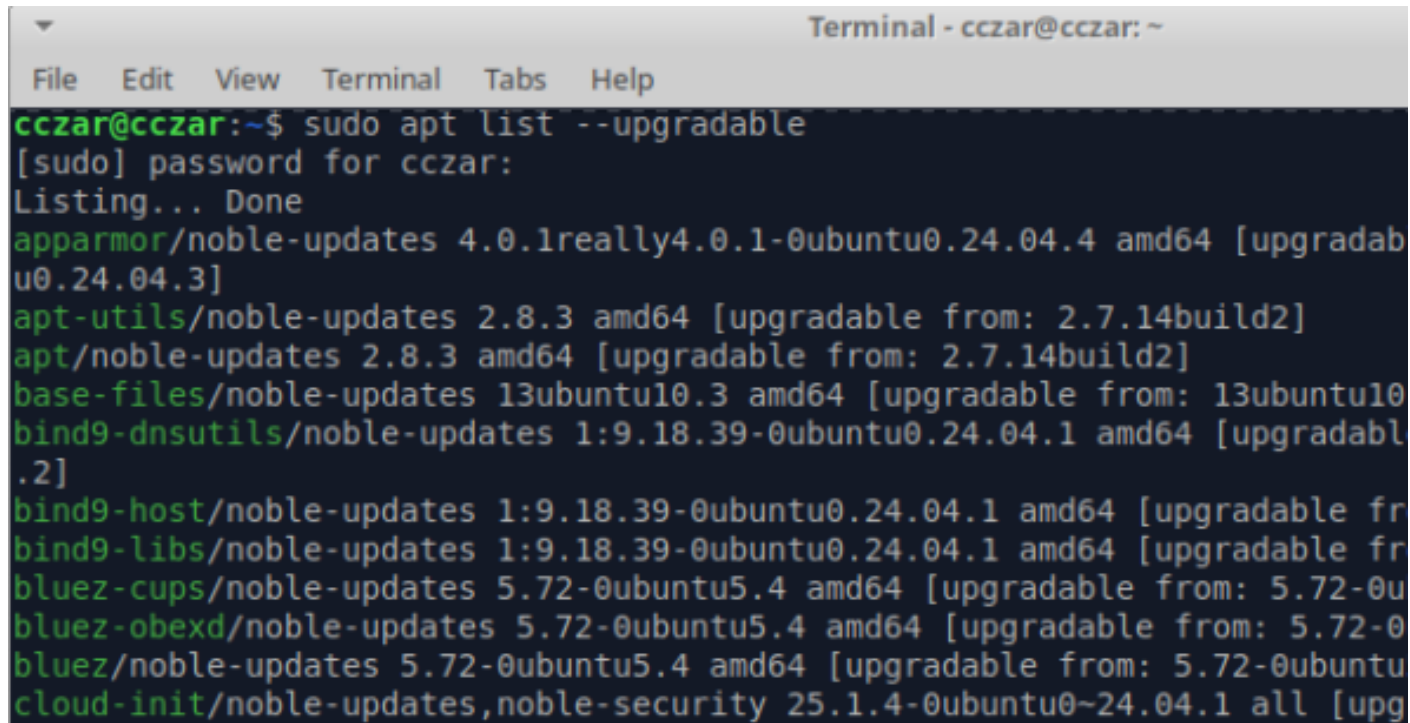
```
Terminal - cczar@cczar: ~
File Edit View Terminal Tabs Help

* Update your system with apt-get update, apt-get upgrade, apt-get dist-upgrade and/or unattended-upgrades [PKGS-7392]
  https://cisofy.com/lynis/controls/PKGS-7392/
```

## REMEDIATION PROCESS

### 1. Searched all vulnerable packages

- A total of 162 packages were found to have upgrades available, including critical system libraries, networking tools, and application utilities. Notable packages included [linux-image-generic](#), [cups](#), [python3.12](#), and [systemd](#).



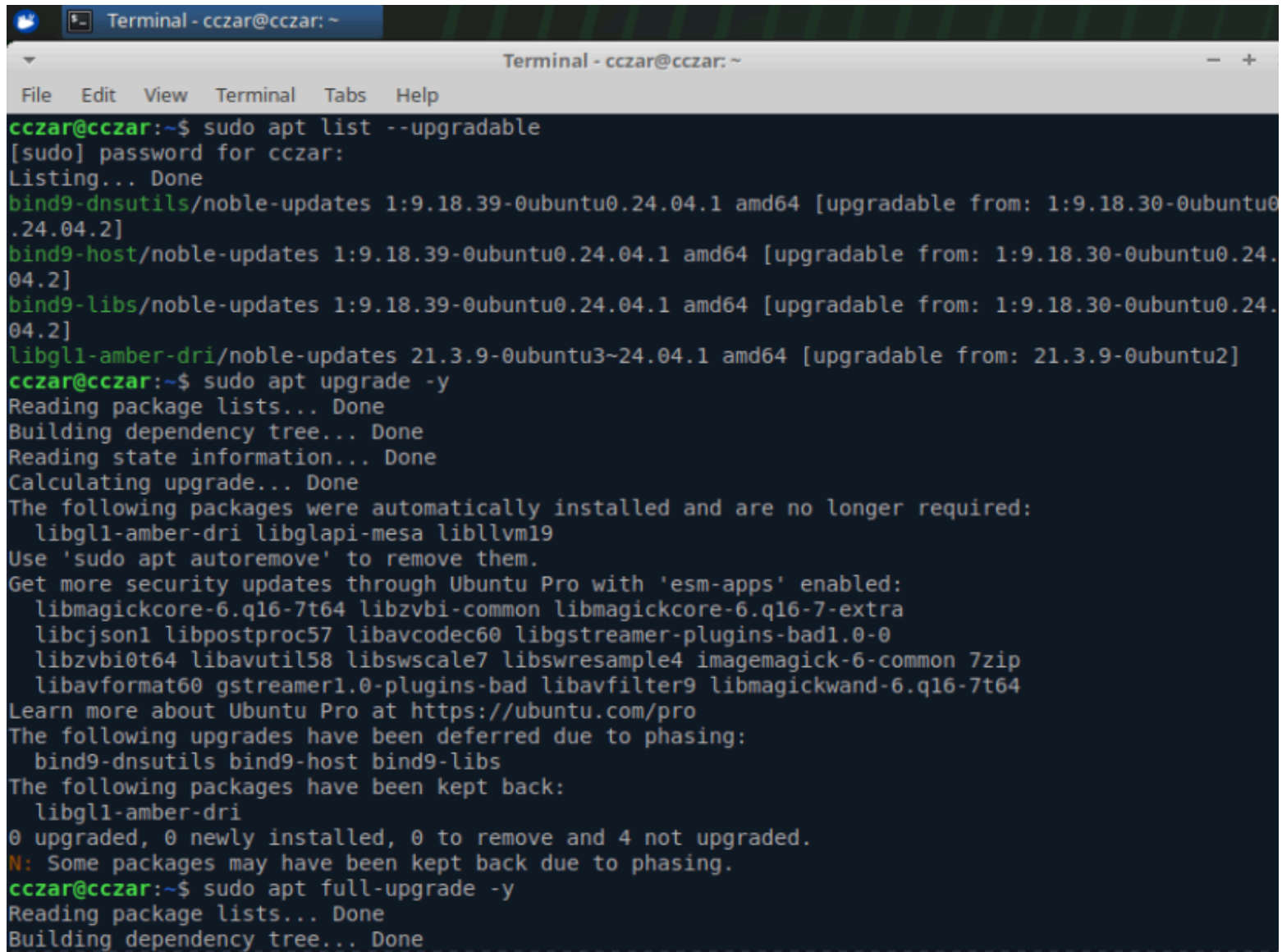
```
Terminal - cczar@cczar: ~
File Edit View Terminal Tabs Help
cczar@cczar:~$ sudo apt list --upgradable
[sudo] password for cczar:
Listing... Done
apparmor/noble-updates 4.0.1really4.0.1-0ubuntu0.24.04.4 amd64 [upgradab
u0.24.04.3]
apt-utils/noble-updates 2.8.3 amd64 [upgradable from: 2.7.14build2]
apt/noble-updates 2.8.3 amd64 [upgradable from: 2.7.14build2]
base-files/noble-updates 13ubuntu10.3 amd64 [upgradable from: 13ubuntu10
bind9-dnsutils/noble-updates 1:9.18.39-0ubuntu0.24.04.1 amd64 [upgradabl
.2]
bind9-host/noble-updates 1:9.18.39-0ubuntu0.24.04.1 amd64 [upgradable fr
bind9-libs/noble-updates 1:9.18.39-0ubuntu0.24.04.1 amd64 [upgradable fr
bluez-cups/noble-updates 5.72-0ubuntu5.4 amd64 [upgradable from: 5.72-0u
bluez-obexd/noble-updates 5.72-0ubuntu5.4 amd64 [upgradable from: 5.72-0
bluez/noble-updates 5.72-0ubuntu5.4 amd64 [upgradable from: 5.72-0ubuntu
cloud-init/noble-updates,noble-security 25.1.4-0ubuntu0~24.04.1 all [upg
```



## 2. Upgraded all installed packages to the latest security versions

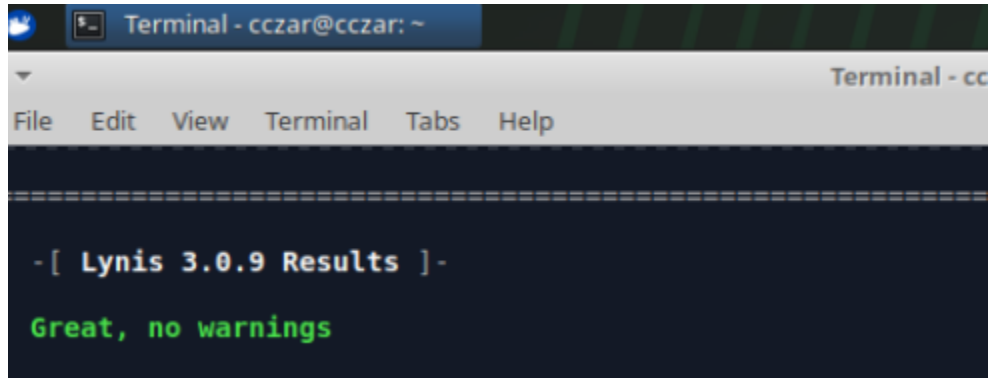
```
Terminal - cczar@cczar: ~
File Edit View Terminal Tabs Help
Get:16 http://us.archive.ubuntu.com/ubuntu noble-backports/restricted amd64 Components [216 B]
Get:17 http://us.archive.ubuntu.com/ubuntu noble-backports/universe amd64 Components [19.2 kB]
Get:18 http://us.archive.ubuntu.com/ubuntu noble-backports/multiverse amd64 Components [212 B]
Fetched 3,904 kB in 2s (1,874 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
162 packages can be upgraded. Run 'apt list --upgradable' to see them.
cczar@cczar:~$ sudo apt update
Hit:1 http://security.ubuntu.com/ubuntu noble-security InRelease
Hit:2 http://us.archive.ubuntu.com/ubuntu noble InRelease
Hit:3 http://us.archive.ubuntu.com/ubuntu noble-updates InRelease
Hit:4 http://us.archive.ubuntu.com/ubuntu noble-backports InRelease
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
162 packages can be upgraded. Run 'apt list --upgradable' to see them.
cczar@cczar:~$ sudo apt upgrade -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Calculating upgrade... Done
```

### 3. Verified that all packages were upgraded

A terminal window titled "Terminal - cczar@cczar: ~" with a menu bar (File, Edit, View, Terminal, Tabs, Help). The terminal shows the execution of 'sudo apt list --upgradable' and 'sudo apt upgrade -y'. It lists upgradable packages like bind9-dnsutils, bind9-host, bind9-libs, and libgll-amber-dri. After the upgrade, it lists packages to be removed (libgll-amber-dri, libglapi-mesa, libllvm19) and security updates available (libmagickcore-6.q16-7t64, libzvti-common, etc.). It also shows deferred upgrades (bind9-dnsutils, bind9-host, bind9-libs) and packages kept back (libgll-amber-dri). The final status is "0 upgraded, 0 newly installed, 0 to remove and 4 not upgraded." and the command 'sudo apt full-upgrade -y' is partially shown.

```
cczar@cczar:~$ sudo apt list --upgradable
[sudo] password for cczar:
Listing... Done
bind9-dnsutils/noble-updates 1:9.18.39-0ubuntu0.24.04.1 amd64 [upgradable from: 1:9.18.30-0ubuntu0.24.04.2]
bind9-host/noble-updates 1:9.18.39-0ubuntu0.24.04.1 amd64 [upgradable from: 1:9.18.30-0ubuntu0.24.04.2]
bind9-libs/noble-updates 1:9.18.39-0ubuntu0.24.04.1 amd64 [upgradable from: 1:9.18.30-0ubuntu0.24.04.2]
libgll-amber-dri/noble-updates 21.3.9-0ubuntu3~24.04.1 amd64 [upgradable from: 21.3.9-0ubuntu2]
cczar@cczar:~$ sudo apt upgrade -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Calculating upgrade... Done
The following packages were automatically installed and are no longer required:
  libgll-amber-dri libglapi-mesa libllvm19
Use 'sudo apt autoremove' to remove them.
Get more security updates through Ubuntu Pro with 'esm-apps' enabled:
  libmagickcore-6.q16-7t64 libzvti-common libmagickcore-6.q16-7-extra
  libcjson1 libpostproc57 libavcodec60 libgstreamer-plugins-bad1.0-0
  libzvti0t64 libavutil58 libswscale7 libswresample4 imagemagick-6-common 7zip
  libavformat60 gstreamer1.0-plugins-bad libavfilter9 libmagickwand-6.q16-7t64
Learn more about Ubuntu Pro at https://ubuntu.com/pro
The following upgrades have been deferred due to phasing:
  bind9-dnsutils bind9-host bind9-libs
The following packages have been kept back:
  libgll-amber-dri
0 upgraded, 0 newly installed, 0 to remove and 4 not upgraded.
N: Some packages may have been kept back due to phasing.
cczar@cczar:~$ sudo apt full-upgrade -y
Reading package lists... Done
Building dependency tree... Done
```

## LYNIS SCAN RESULT AFTER REMEDIATION

A screenshot of a terminal window titled "Terminal - cczar@cczar: ~". The window has a menu bar with "File", "Edit", "View", "Terminal", "Tabs", and "Help". The terminal content shows a separator line of equals signs, followed by the text "-[ Lynis 3.0.9 Results ]-", and then "Great, no warnings" in green text.

```
=====
-[ Lynis 3.0.9 Results ]-
Great, no warnings
```

## Finding: GRUB Bootloader Unprotected (BOOT-1003)

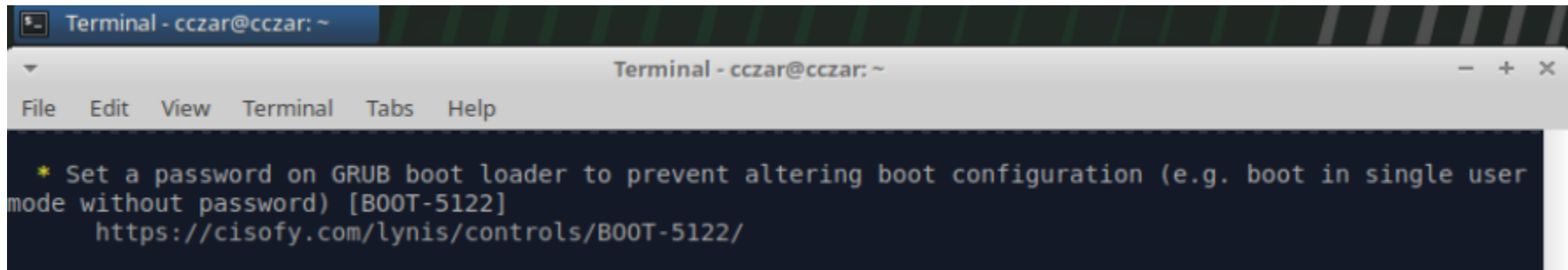
**NIST Control Mapping:** PR.PS-01: Configuration management practices are established and applied

**Risk:** Without a GRUB boot password, an attacker with physical access could modify boot parameters to bypass authentication controls, disable security mechanisms, or gain root access.

**Severity:** High

**Remediation:** Configure a GRUB boot password to ensure that only authorized administrators can modify boot parameters.

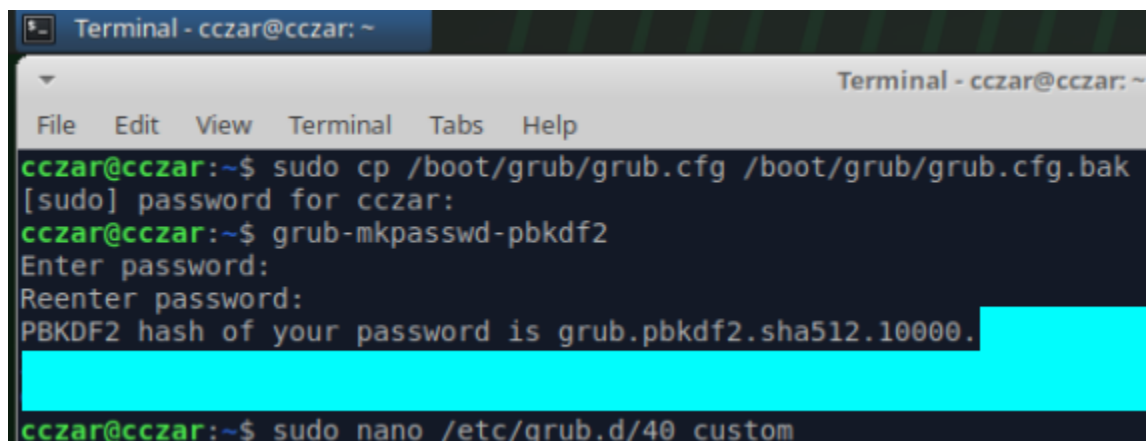
### INITIAL LYNIS SCAN RESULT



```
Terminal - cczar@cczar: ~  
File Edit View Terminal Tabs Help  
* Set a password on GRUB boot loader to prevent altering boot configuration (e.g. boot in single user mode without password) [BOOT-5122]  
https://cisofy.com/lynis/controls/BOOT-5122/
```

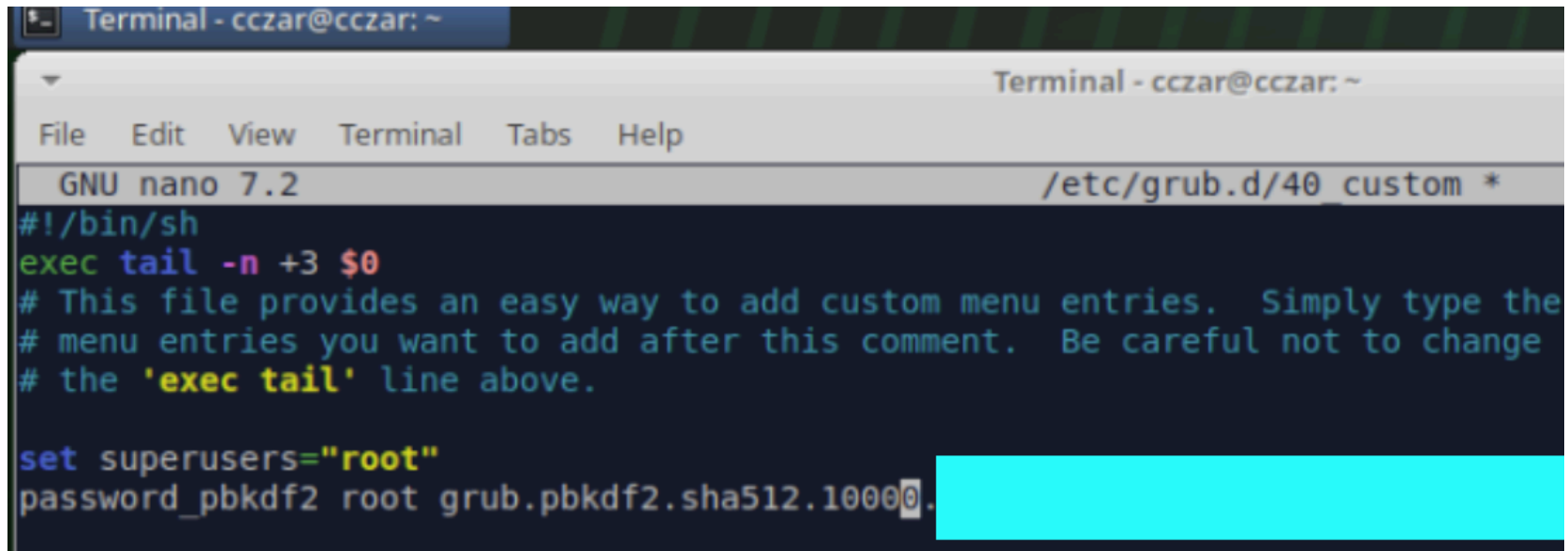
### REMEDIATION PROCESS

1. Generated a GRUB Password Hash (hash is hidden for security purposes)



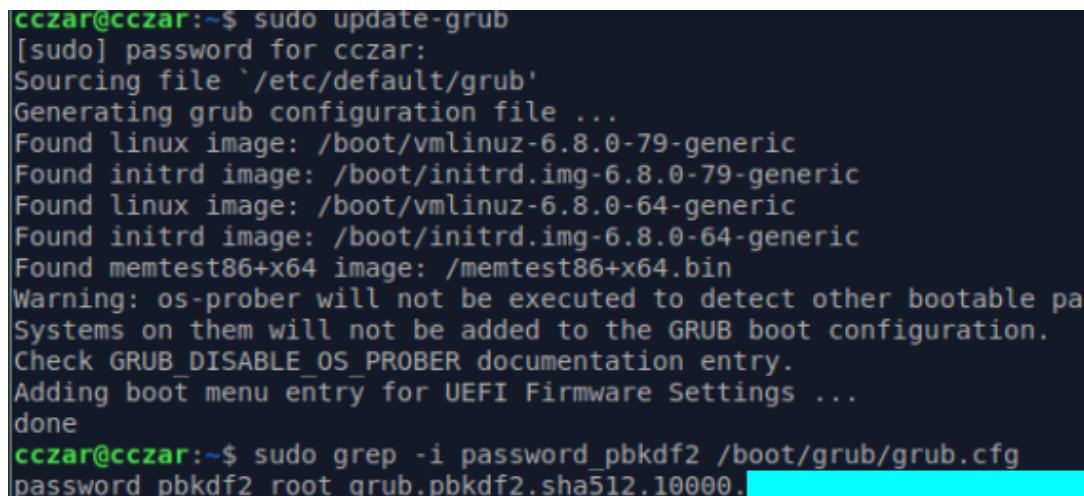
```
Terminal - cczar@cczar: ~  
File Edit View Terminal Tabs Help  
cczar@cczar:~$ sudo cp /boot/grub/grub.cfg /boot/grub/grub.cfg.bak  
[sudo] password for cczar:  
cczar@cczar:~$ grub-mkpasswd-pbkdf2  
Enter password:  
Reenter password:  
PBKDF2 hash of your password is grub.pbkdf2.sha512.10000.  
[REDACTED]  
cczar@cczar:~$ sudo nano /etc/grub.d/40_custom
```

## 2. Configured GRUB



```
Terminal - cczar@cczar: ~  
File Edit View Terminal Tabs Help  
GNU nano 7.2 /etc/grub.d/40 custom *  
#!/bin/sh  
exec tail -n +3 $0  
# This file provides an easy way to add custom menu entries. Simply type the  
# menu entries you want to add after this comment. Be careful not to change  
# the 'exec tail' line above.  
  
set superusers="root"  
password_pbkdf2 root grub.pbkdf2.sha512.10000.
```

3. Rebuilt the GRUB configuration to incorporate the changes from [40\\_custom](#), ran the update command, and verified password line was applied



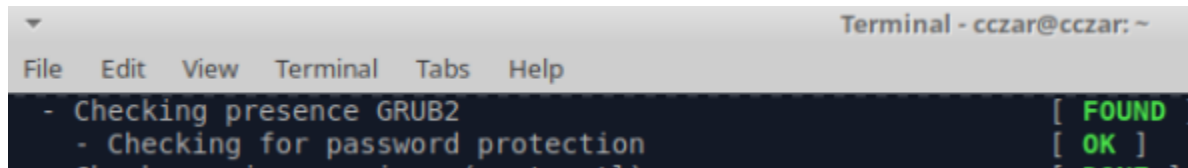
```
cczar@cczar:~$ sudo update-grub  
[sudo] password for cczar:  
Sourcing file '/etc/default/grub'  
Generating grub configuration file ...  
Found linux image: /boot/vmlinuz-6.8.0-79-generic  
Found initrd image: /boot/initrd.img-6.8.0-79-generic  
Found linux image: /boot/vmlinuz-6.8.0-64-generic  
Found initrd image: /boot/initrd.img-6.8.0-64-generic  
Found memtest86+x64 image: /memtest86+x64.bin  
Warning: os-prober will not be executed to detect other bootable pa  
Systems on them will not be added to the GRUB boot configuration.  
Check GRUB_DISABLE_OS_PROBER documentation entry.  
Adding boot menu entry for UEFI Firmware Settings ...  
done  
cczar@cczar:~$ sudo grep -i password_pbkdf2 /boot/grub/grub.cfg  
password_pbkdf2 root grub.pbkdf2.sha512.10000.
```

#### 4. Tested GRUB configuration

- After reboot, at the GRUB menu, pressing “e” to edit boot entry prompted for the password configured in [/etc/grub.d/40\\_custom](#). Entering the correct password allowed the system to boot normally, confirming that GRUB hardening is in effect.

**Note:** To further secure the system against unauthorized physical access, it is recommended to restrict root login on virtual and physical consoles (TTYs).

#### LYNIS SCAN RESULT AFTER REMEDIATION



```
Terminal - cczar@cczar: ~  
File Edit View Terminal Tabs Help  
- Checking presence GRUB2 [ FOUND ]  
- Checking for password protection [ OK ]  
- Checking for password protection (root login) [ FOUND ]
```

## Finding: Insecure Permissions on /etc/sudoers.d (PERM-1005)

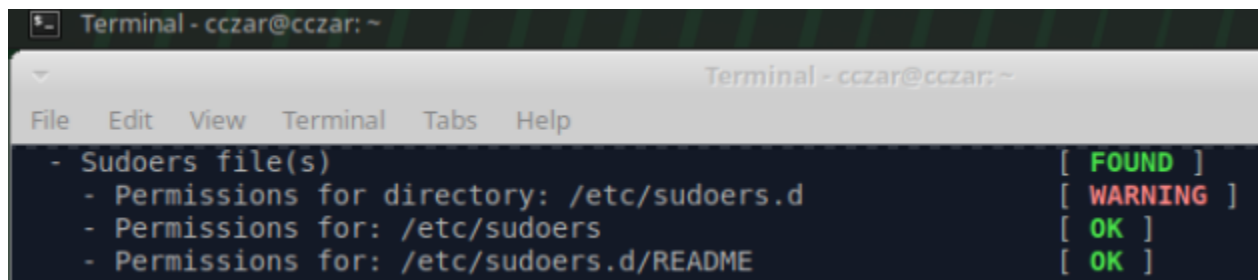
**NIST Control Mapping:** PR.AA-05 – Access permissions, entitlements, and authorizations are defined in a policy, managed, enforced, and reviewed, and incorporate the principles of least privilege and separation of duties

**Risk:** The [/etc/sudoers.d](#) directory has insecure permissions, which could allow unauthorized users to modify sudo configuration files, potentially leading to privilege escalation.

**Severity:** High

**Remediation:** Review and correct the permissions on the [/etc/sudoers.d](#) directory to ensure only authorized users can modify them.

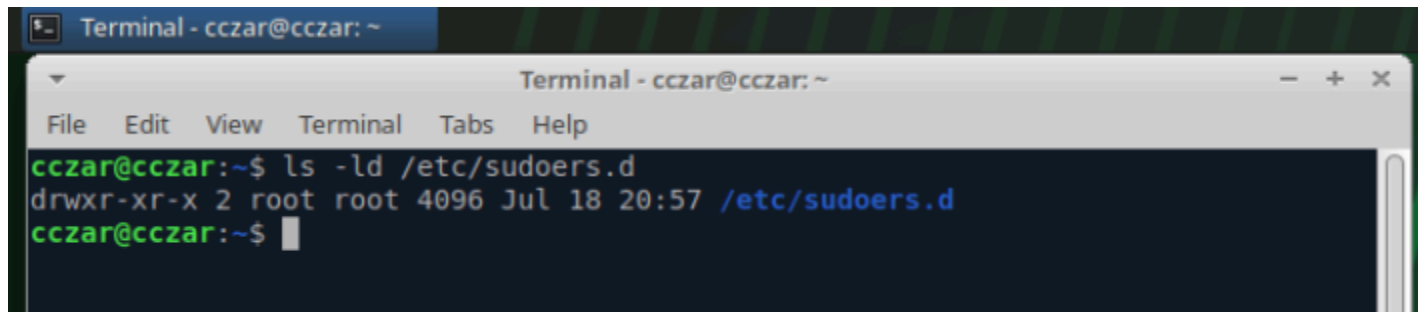
## INITIAL LYNIS SCAN RESULT



```
Terminal - cczar@cczar: ~  
File Edit View Terminal Tabs Help  
- Sudoers file(s) [ FOUND ]  
- Permissions for directory: /etc/sudoers.d [ WARNING ]  
- Permissions for: /etc/sudoers [ OK ]  
- Permissions for: /etc/sudoers.d/README [ OK ]
```

## REMEDIATION PROCESS

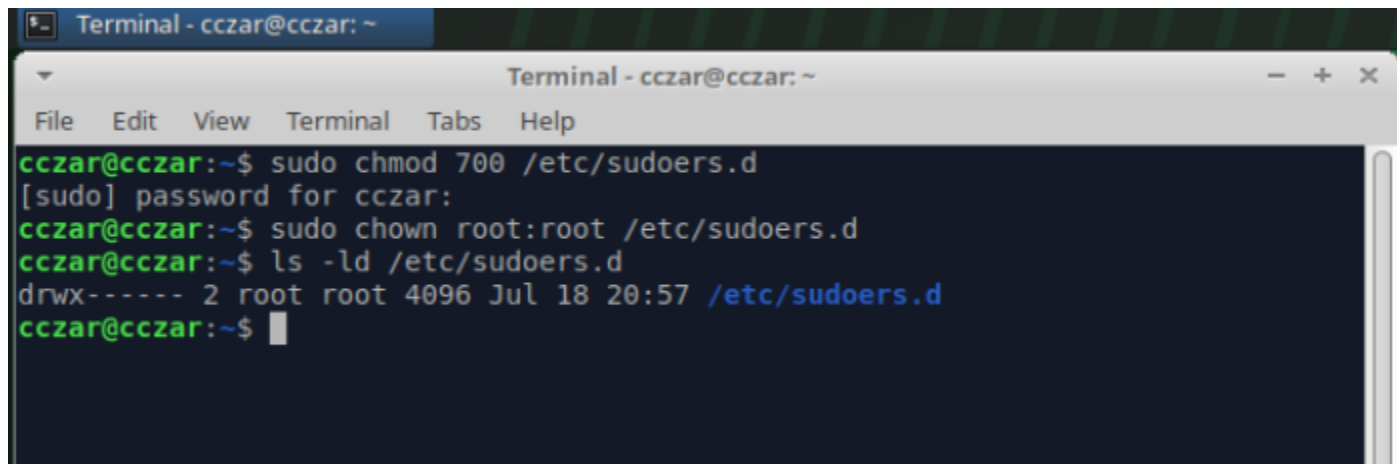
1. Reviewed current permissions, verifying that group and others have read/execute access to [/etc/sudoers.d](#) directory



```
Terminal - cczar@cczar: ~  
File Edit View Terminal Tabs Help  
cczar@cczar:~$ ls -ld /etc/sudoers.d  
drwxr-xr-x 2 root root 4096 Jul 18 20:57 /etc/sudoers.d  
cczar@cczar:~$
```

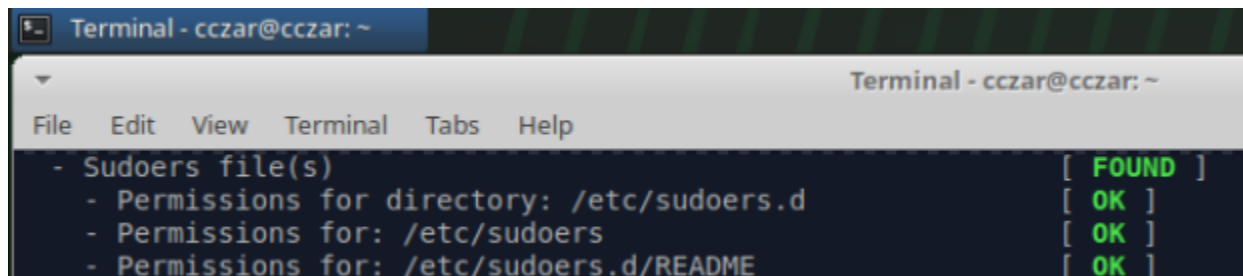
## 2. Restricted permissions to root-only, with group and others having no access, and verified permission change.

**Note:** Restricted the `/etc/sudoers.d` directory permissions so that only root can read, write, or execute. After updating permissions, `/etc/sudoers.d` is owned by root with access limited to root-only. Listing the directory confirms permissions set to `700`, ensuring that unauthorized users cannot modify sudo configuration files and mitigating the risk of privilege escalation.



```
Terminal - cczar@cczar: ~  
cczar@cczar:~$ sudo chmod 700 /etc/sudoers.d  
[sudo] password for cczar:  
cczar@cczar:~$ sudo chown root:root /etc/sudoers.d  
cczar@cczar:~$ ls -ld /etc/sudoers.d  
drwx----- 2 root root 4096 Jul 18 20:57 /etc/sudoers.d  
cczar@cczar:~$
```

### LYNIS SCAN RESULT AFTER REMEDIATION



```
Terminal - cczar@cczar: ~  
- Sudoers file(s) [ FOUND ]  
- Permissions for directory: /etc/sudoers.d [ OK ]  
- Permissions for: /etc/sudoers [ OK ]  
- Permissions for: /etc/sudoers.d/README [ OK ]
```



## Finding: USB Storage Driver Enabled (USB-1001)

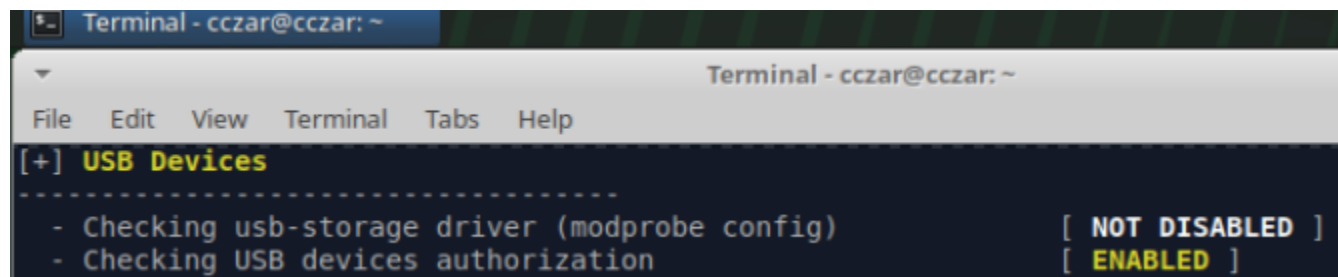
**NIST Control Mapping:** PR.DS-01 – The confidentiality, integrity, and availability of data-at-rest are protected

**Risk:** The USB storage driver is enabled, allowing unauthorized users to transfer sensitive data via USB devices.

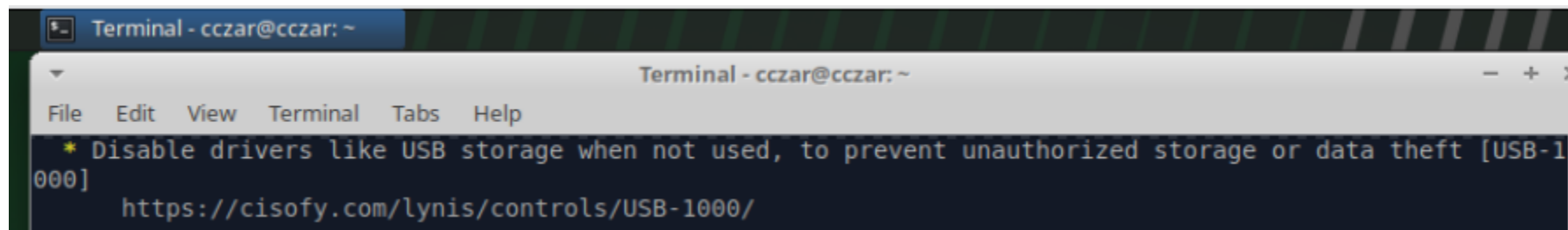
**Severity:** High

**Remediation:** Disable the USB storage driver to prevent data exfiltration via USB.

### INITIAL LYNIS SCAN RESULT



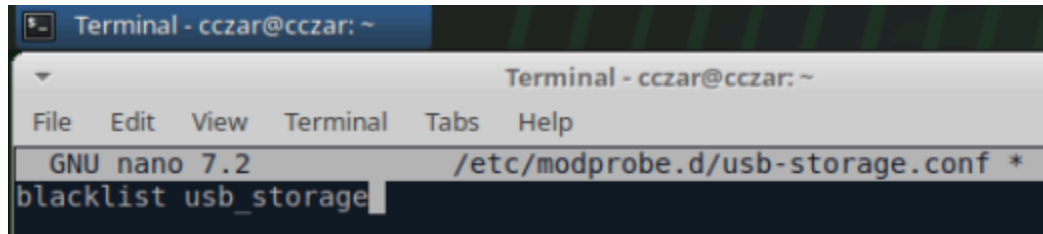
```
Terminal - cczar@cczar: ~
File Edit View Terminal Tabs Help
[+] USB Devices
-----
- Checking usb-storage driver (modprobe config) [ NOT DISABLED ]
- Checking USB devices authorization [ ENABLED ]
```



```
Terminal - cczar@cczar: ~
File Edit View Terminal Tabs Help
* Disable drivers like USB storage when not used, to prevent unauthorized storage or data theft [USB-1000]
https://cisofy.com/lynis/controls/USB-1000/
```

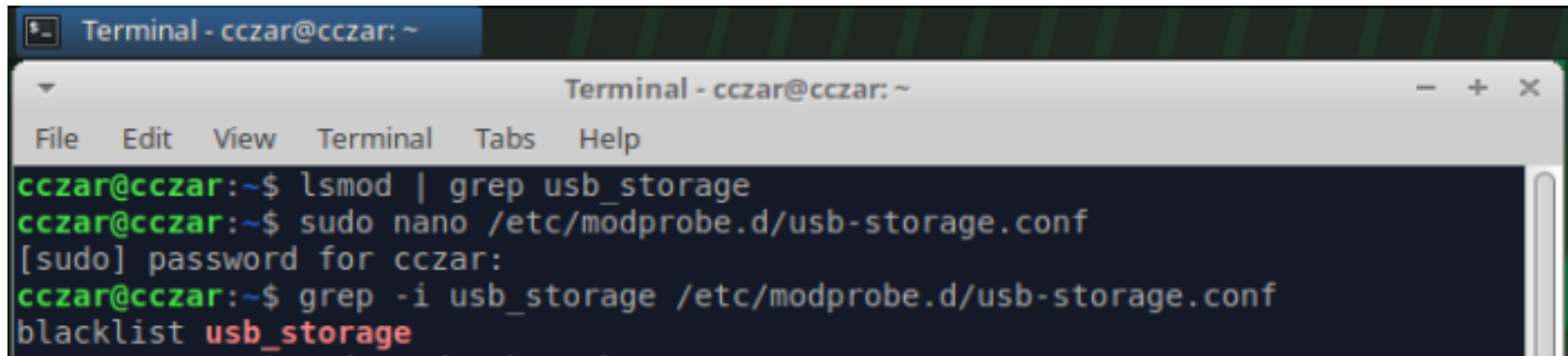
## REMEDIATION PROCESS

1. Verified USB storage driver was unloaded and created blacklist file for modprobe to prevent USB storage driver from loading



```
Terminal - cczar@cczar: ~
GNU nano 7.2 /etc/modprobe.d/usb-storage.conf *
blacklist usb_storage
```

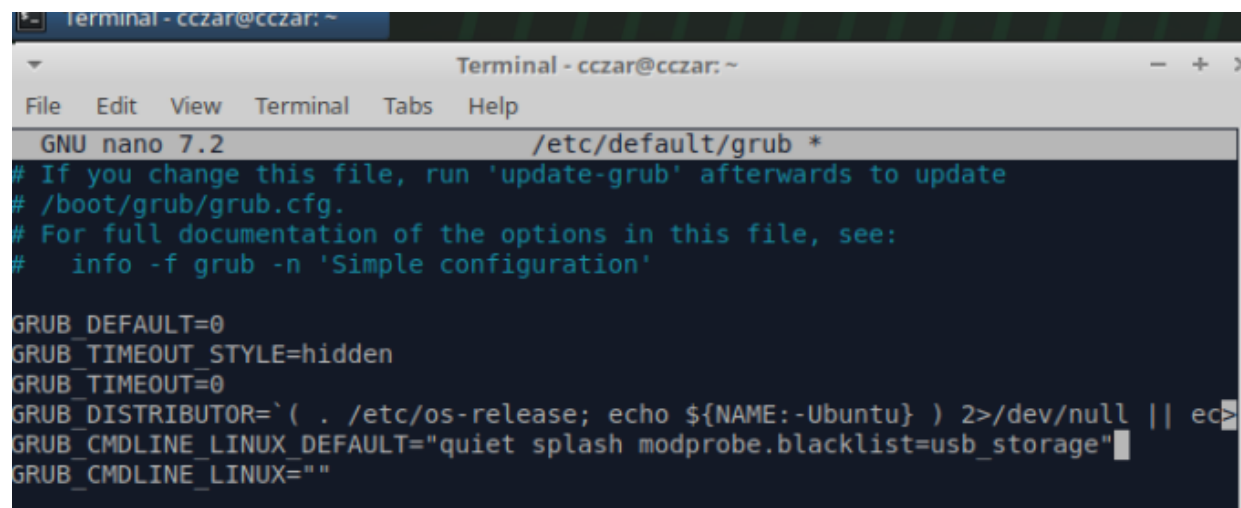
2. Confirmed blacklist status of USB storage driver



```
Terminal - cczar@cczar: ~
cczar@cczar:~$ lsmod | grep usb_storage
cczar@cczar:~$ sudo nano /etc/modprobe.d/usb-storage.conf
[sudo] password for cczar:
cczar@cczar:~$ grep -i usb_storage /etc/modprobe.d/usb-storage.conf
blacklist usb_storage
```

### 3. Disabled the USB module via kernel parameters

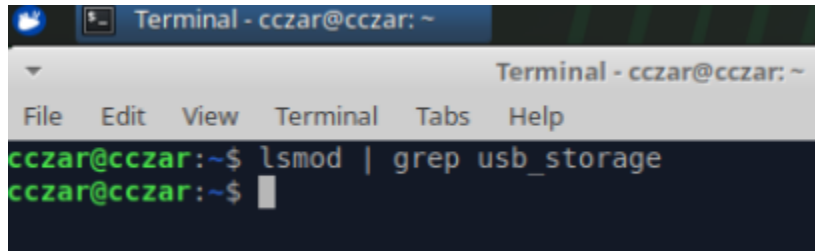
```
cczar@cczar:~$ grep -i usb_storage /etc/modprobe.d/usb-storage.conf
blacklist usb_storage
cczar@cczar:~$ sudo modprobe usb_storage
cczar@cczar:~$ lsmod | grep usb_storage
usb_storage      86016  0
cczar@cczar:~$ sudo nano /etc/default/grub
cczar@cczar:~$ lsmod | grep usb_storage
usb_storage      86016  0
cczar@cczar:~$ sudo update-grub
Sourcing file `/etc/default/grub'
Generating grub configuration file ...
Found linux image: /boot/vmlinuz-6.8.0-79-generic
Found initrd image: /boot/initrd.img-6.8.0-79-generic
Found linux image: /boot/vmlinuz-6.8.0-64-generic
Found initrd image: /boot/initrd.img-6.8.0-64-generic
Found memtest86+x64 image: /memtest86+x64.bin
Warning: os-prober will not be executed to detect other bootable partitions.
Systems on them will not be added to the GRUB boot configuration.
Check GRUB_DISABLE_OS_PROBER documentation entry.
Adding boot menu entry for UEFI Firmware Settings ...
done
```



```
Terminal - cczar@cczar: ~
GNU nano 7.2 /etc/default/grub *
# If you change this file, run 'update-grub' afterwards to update
# /boot/grub/grub.cfg.
# For full documentation of the options in this file, see:
#   info -f grub -n 'Simple configuration'

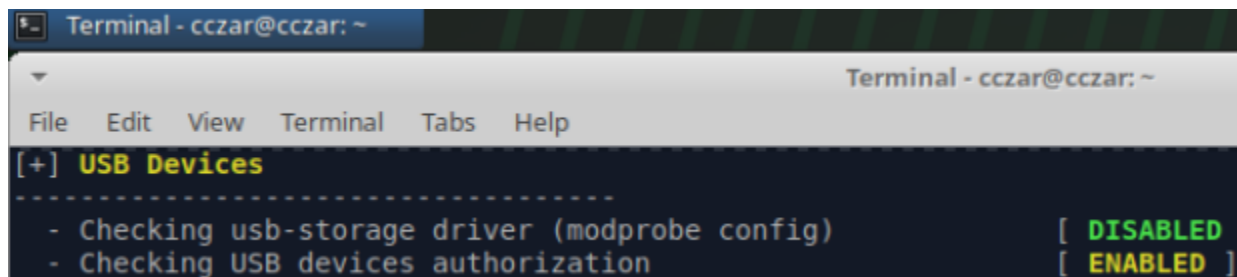
GRUB_DEFAULT=0
GRUB_TIMEOUT_STYLE=hidden
GRUB_TIMEOUT=0
GRUB_DISTRIBUTOR=`(. /etc/os-release; echo ${NAME:-Ubuntu}) 2>/dev/null || ec>
GRUB_CMDLINE_LINUX_DEFAULT="quiet splash modprobe.blacklist=usb_storage"
GRUB_CMDLINE_LINUX=""
```

4. After applying the USB storage module blacklist and updating the kernel parameters, a reboot shows `lsmod | grep usb_storage` does not return the module, confirming the module is no longer loaded at boot and the hardening is effective.



```
Terminal - cczar@cczar: ~  
File Edit View Terminal Tabs Help  
cczar@cczar:~$ lsmod | grep usb_storage  
cczar@cczar:~$
```

## LYNIS SCAN RESULT AFTER REMEDIATION



```
Terminal - cczar@cczar: ~  
File Edit View Terminal Tabs Help  
[+] USB Devices  
-----  
- Checking usb-storage driver (modprobe config) [ DISABLED ]  
- Checking USB devices authorization [ ENABLED ]
```

# Conclusion

The assessment evaluated the overall security posture of the environment, identifying risks across authentication, package management, bootloader security, access controls, and peripheral device use. Several high-risk findings—including authentication hardening, package updates, GRUB boot protection, restricted console access, and USB storage controls—have been remediated and verified.

## Key Improvements

- Improved alignment with NIST CSF 2.0 and CIS benchmarks.
- Reduced the attack surface and enhanced system resilience.
- Established a foundation for ongoing security governance and monitoring.

## Action Items

1. Enforce secure password standards for all accounts; regularly review and update the module to align with current best practices.
2. Remove unnecessary or unsupported software and implement a regular patch management process to mitigate exposure to known vulnerabilities.
3. Configure GRUB so only authorized users can modify boot options or enter single-user mode; verify password configuration and restrict console access.
4. Enforce least privilege for `/etc/sudoers.d` and regularly audit sudo configuration to detect unauthorized changes.
5. Restrict USB device use through endpoint controls and monitoring; regularly review USB access policies to protect sensitive data.

## Remaining Risks

System issues persist that could compromise security if not addressed promptly. Immediate remediation is recommended to maintain a hardened environment.

## Future Recommendations

- Regular patching and configuration audits.
- Integration of configuration management and monitoring tools.
- Continuous adherence to industry security standards and best practices.