



Certified Network Defender v3

MODULE 06

ENDPOINT SECURITY-LINUX SYSTEMS

EC-Council Official Curricula

This page is intentionally left blank.

LEARNING OBJECTIVES

The learning objectives of this module are:

- LO#01: Understand Linux OS and security concerns
- LO#02: Discuss Linux installation and patching
- LO#03: Discuss Linux OS hardening techniques
- LO#04: Discuss Linux user access and password management
- LO#05: Discuss Linux network and remote access security
- LO#06: Discuss various Linux security tools and frameworks

Copyright © by **EC-Council**. All Rights Reserved. Reproduction is Strictly Prohibited.

Learning Objectives

This module introduces the Linux OS, its security features, and the various techniques to harden the OS security. The objectives of this module are listed below:

- Understand Linux OS and security concerns
- Discuss Linux installation and patching
- Discuss Linux OS hardening techniques
- Discuss Linux user access and password management
- Discuss Linux network and remote access security
- Discuss various Linux security tools and frameworks



LO#01: Understand Linux OS and security concerns

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

LO#01: Understand Linux OS and Security Concerns

The objective of this section is to help you understand the security features and security concerns in Linux OS.

Linux OS

Linux is an **open-source OS** widely used across enterprises and government bodies

Components of Linux OS:

- **Hardware:** Consists of physical devices like monitor, RAM, HDD, CPU etc.
- **Kernel:** A core component of the OS having complete control over system resources
- **Shell:** An interface that takes inputs from the users, sends it to the kernel, and sends the output of the kernel back
- **Applications or utilities:** Utility programs that can be launched by running the shell. Utilities gives most of the functionalities provided by an operating system to the user
- **System libraries:** Special functions that do not require any access rights to the kernel modules to implement the functionality of the OS
- **Daemons:** Services that run to perform tasks like printing, scheduling them etc.
- **Graphical server:** Sub-system responsible for displaying graphics on the monitor and is referred as X

Linux System Architecture

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Linux OS

Linux is an open-source OS. It is widely used across enterprises and government bodies. It is one of the popular versions of the UNIX operating system. Therefore, its functionality is very similar to UNIX. Considered to be one of the most reliable and secure operating systems, it has been around since the mid-1990s.

A few major advantages of Linux are listed below:

- Linux is a freely available to public.
- It allows applications to be installed without rebooting the OS for their functioning.
- It allows customization to fix bugs rapidly as it is open source.

A historical timeline of Linux developments is summarized below:

| Year | Description |
|------|--|
| 1991 | First Linux code released. |
| 1992 | Linus licenses Linux under the GPL. An important decision that will contribute to its success. |
| 1993 | Slackware becomes first widely adopted distribution. |
| 1996 | Linus visits aquarium, gets bit by a penguin, and chooses it as Linux MASCOT. |
| 1998 | Tech. giants begin announcing platform support for Linux. |
| 1999 | Red Hat goes public. |

| | |
|-------------|---|
| 2003 | IBM runs the famous Linux advertisement during the Super Bowl. |
| 2005 | Linux appears on the cover of Business Week. |
| 2007 | The Linux Foundation is formed to promote and standardize Linux. |
| 2010 | The Linux-based Android OS outships all other smartphone OSes in the US and climbs to dominate. |
| 2011 | Linux turns 20 and powers the worlds' super computers, ATMS, phones, etc. |

Table 6.1: Timeline of Linux Developments

Components of Linux OS

The key components of Linux OS are listed below:

- **Hardware:** This component comprises physical devices such as monitor, RAM, HDD, CPU, etc.
- **Kernel:** It is the core component of the Linux OS. It has complete control over system resources and interacts directly with hardware (memory and peripheral devices).
- **Shell:** This is an interface to the kernel. It hides the complexity of the functions of the kernel from users. It takes commands from the user and executes the functions of the kernel.
- **Applications or utilities:** Utility programs are launched by running the shell. They provide most of the functionalities provided by the OS to the user. They are responsible for doing some specialized or individual tasks.

Most new Linux distributions feature App store-like tools that centralize and simplify application installations. For example, the Linux Software Center allows users to search for apps and also allows installing them from a centralized location.

- **System libraries:** These are special functions or programs. The applications or utilities use system libraries without access rights to access the kernel's modules and to implement most of the functionalities of the OS.
- **Daemons:** These are the background services run to perform specific tasks like printing, scheduling, sound, etc. They either start-up during boot or after logging into the desktop.
- **Graphical server:** This is the subsystem responsible for displaying graphics on the monitor and is referred to as X server or just X.

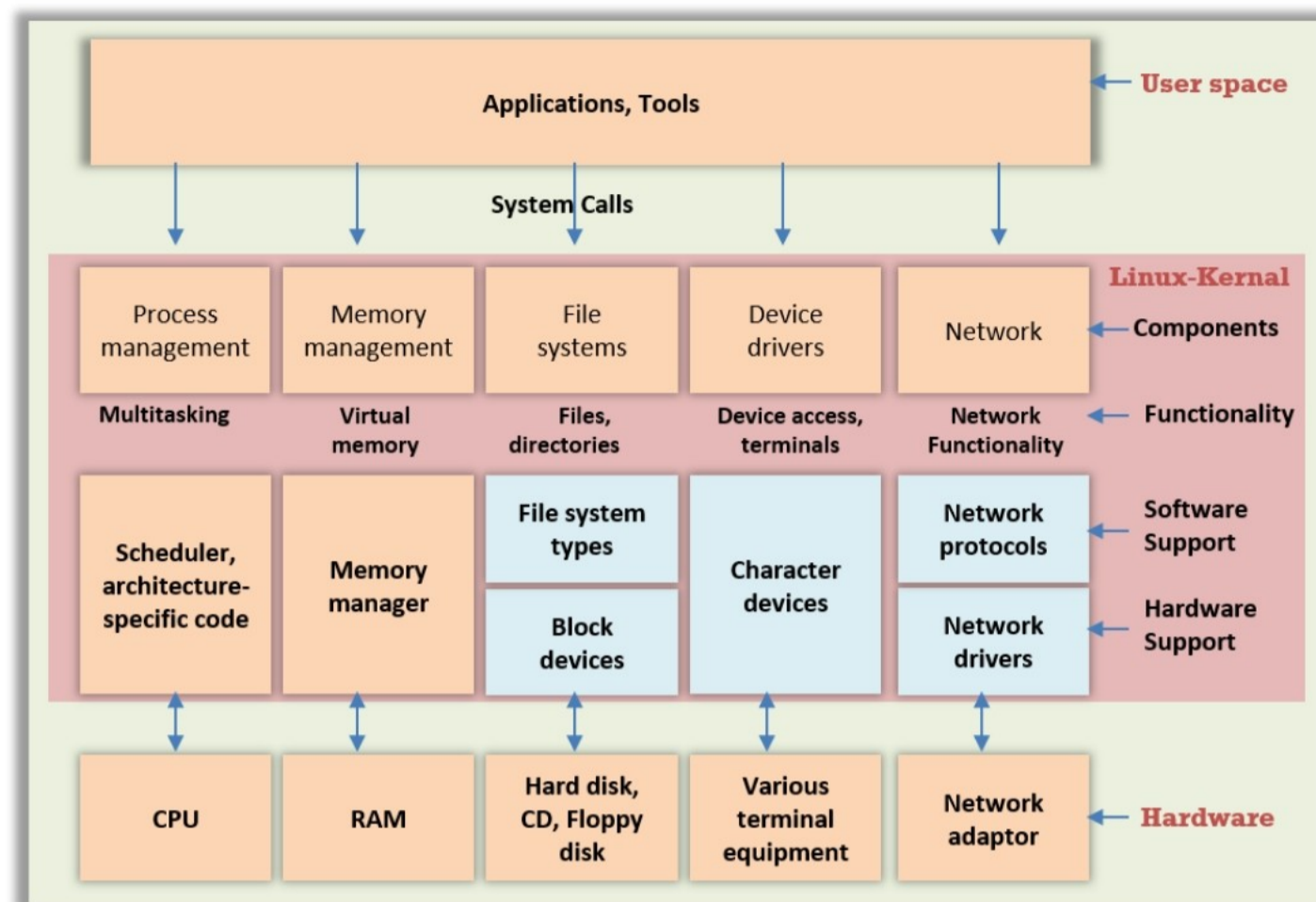


Figure 6.1: Linux System Architecture

Linux Features

CND

Certified Network Defender

| | |
|--------------------------|---|
| Portability | Linux kernel and applications can be installed on different hardware platforms |
| Open Source | Source code of Linux is available for free and it is a community-based development project |
| Multuser | Multiple users can access the various resources at the same time |
| Multiprogramming | Multiple number of applications/programs can run at the same time |
| Hierarchical File System | Linux standard hierarchical file structure arranges directories and files in a tree like structure |
| Shell | A special interpreter program used to execute programs or applications |
| Security | Linux provides security features like authentication, controlled access to files using passwords, data encryption |

Copyright © by **EC-Council**. All Rights Reserved. Reproduction is Strictly Prohibited.

Linux Features

- Portability: Linux works on different types of hardware in the same way.
- Open-source: OS Linux is a community-based development project, and its source code is free. Many people worldwide collaborate to improve Linux OS.
- Multi-user: Multiple users can access the system resources (memory, RAM, applications, etc.) at a time.
- Multiprogramming: OS Linux features running multiple programs at a time.
- Hierarchical file system: The system files and user files are arranged in a standard hierarchical file structure.
- Shell: It is a special interpreter program that is used to execute commands of the OS. It is used to perform different types of operations and call applications, etc.
- Security: Linux provides its users authentication (password protection, controlled access to special files, and encryption of data).

Linux Security Concerns

CND

Certified Network Defender

Linux security is becoming a **concern** as hackers have exploited many vulnerabilities in Linux in the recent past

Linux : Security vulnerabilities

Documentation

Search

Log in

CVEdetails.com

powered by SecurityScorecard

Vulnerabilities

By Date

By Type

Known Exploited

Assigners

CVSS Scores

EPSS Scores

Search

Vulnerable Software

Vendors

Products

Version Search

Vulnerability Intel.

Newsfeed

Open Source Vulns

Emerging CVEs

Linux : Security Vulnerabilities (Denial of service)

Published in: 2023 January February March April May June July August September October November December

CVSS Scores Greater Than: 0 1 2 3 4 5 6 7 8 9 In CISA KEV Catalog

Sort Results By: Publish Date Update Date CVE Number CVE Number CVSS Score EPSS Score

1465 vulnerabilities found

CVE-2023-42755

A flaw was found in the IPv4 Resource Reservation Protocol (RSVP) classifier in the Linux kernel. The xprt pointer may go beyond the linear part of the skb, leading to an out-of-bounds read in the 'rsvp_classify' function. This issue may allow a local user to crash the system and cause a denial of service.

Max Base Score 6.5

Published 2023-10-05

Updated 2023-11-02

EPSS 0.04%

CVE-2023-39198

Max Base Score 7.5

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Linux Security Concerns

Even though many people considered Linux to be inherently secure, as its code can be inspected for issues, attackers have exploited a number of security vulnerabilities in the recent past.

The following screenshot shows the latest CVE details of Linux security vulnerabilities

A screenshot of the CVEdetails.com website. The page title is "Linux : Security Vulnerabilities (Denial of service)". It shows a list of vulnerabilities with filters for "Published in:" (2023, January, February, March, April, May, June, July, August, September, October, November, December) and "CVSS Scores Greater Than:" (0, 1, 2, 3, 4, 5, 6, 7, 8, 9). There are also links to "In CISA KEV Catalog" and "Sort Results By:". The main content area shows "1465 vulnerabilities found" and a list of vulnerabilities. The first vulnerability is "CVE-2023-42755" with a description: "A flaw was found in the IPv4 Resource Reservation Protocol (RSVP) classifier in the Linux kernel. The xprt pointer may go beyond the linear part of the skb, leading to an out-of-bounds read in the 'rsvp_classify' function. This issue may allow a local user to crash the system and cause a denial of service." It has a "Max Base Score" of 6.5, was "Published" on 2023-10-05, "Updated" on 2023-11-02, and has an "EPSS" of 0.04%. The second vulnerability is "CVE-2023-39198" with a "Max Base Score" of 7.5.

Figure 6.2: CVE Details of Linux Security Vulnerabilities

Linux OS is a free and open-source software that can be modified and distributed by anyone. This can result in unexpected vulnerabilities that can be exploited by the attacker. Some vulnerabilities arise due to oversight by network defenders and poor configuration settings. However, the recent attacks show that Linux is being targeted for various malware attacks.

Page 873

Certified Network Defender Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.



LO#02: Discuss Linux installation and patching

Copyright © by **EC-Council**. All Rights Reserved. Reproduction is Strictly Prohibited.

LO#02: Discuss Linux Installation and Patching

The objective of this section is to help you understand the importance of security vigilance in Linux OS installation and patching. It also describes some of the standard security practices associated with Linux OS installation and patching.

Enable Minimal Installation Option



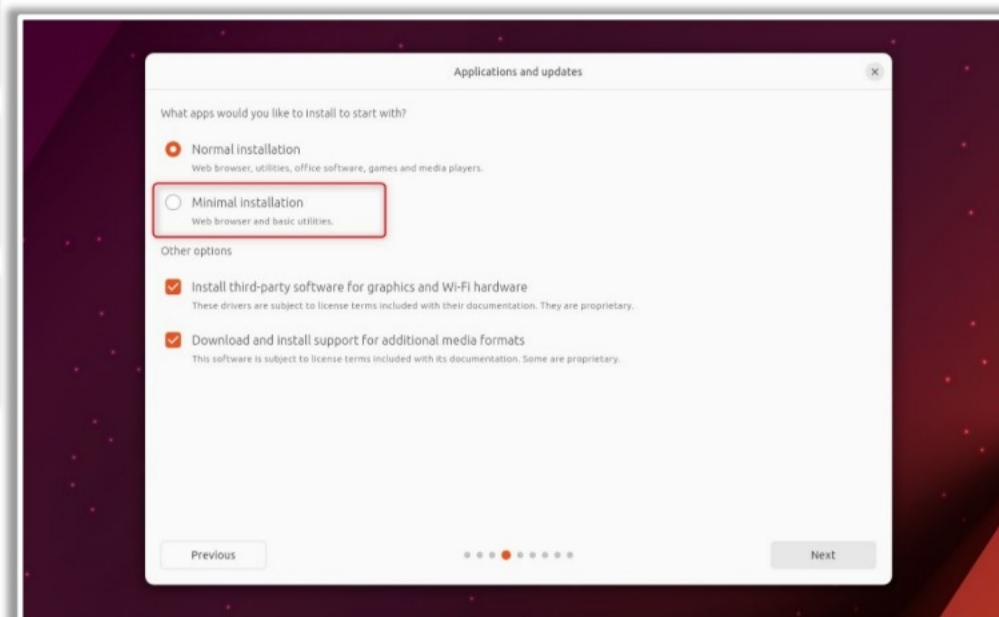
The Ubuntu Linux OS provides **minimal installation** option

The minimal installation option minimizes the number to packages being install during the operating system installation

This option prevents the Ubuntu OS from downloading:

Unnecessary packages, applications

Third-party application or **untrusted** application that may be vulnerable to new exploits



Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Enable Minimal Installation Option

- The Ubuntu Linux operating system provides the “Minimal Installation (Minimal)” option. The “minimal install” checkbox is added to Ubiquity (the Ubuntu installer) from Ubuntu version 18.04 onward. During the setup of the Bionic Beaver (Ubuntu codename for version 18.04), the user selects whether they need a full-fat Ubuntu install or a semi-skim version.
- The “Minimal installation” option appears in the section of the installer that prompts whether to minimize the number of packages or prevent the unnecessary packages being installed during the operating system installation.

For example: restricting installation of codecs to enable multimedia playback alongside the main desktop.

- Minimal installation features the same core Ubuntu experience, but with fewer apps. A minimal Ubuntu install comprises a desktop environment, a web browser, a few core systems tools, and nothing else. It removes around 80 packages from the default install. The system administrator can install any required application after the completion of the OS installation.

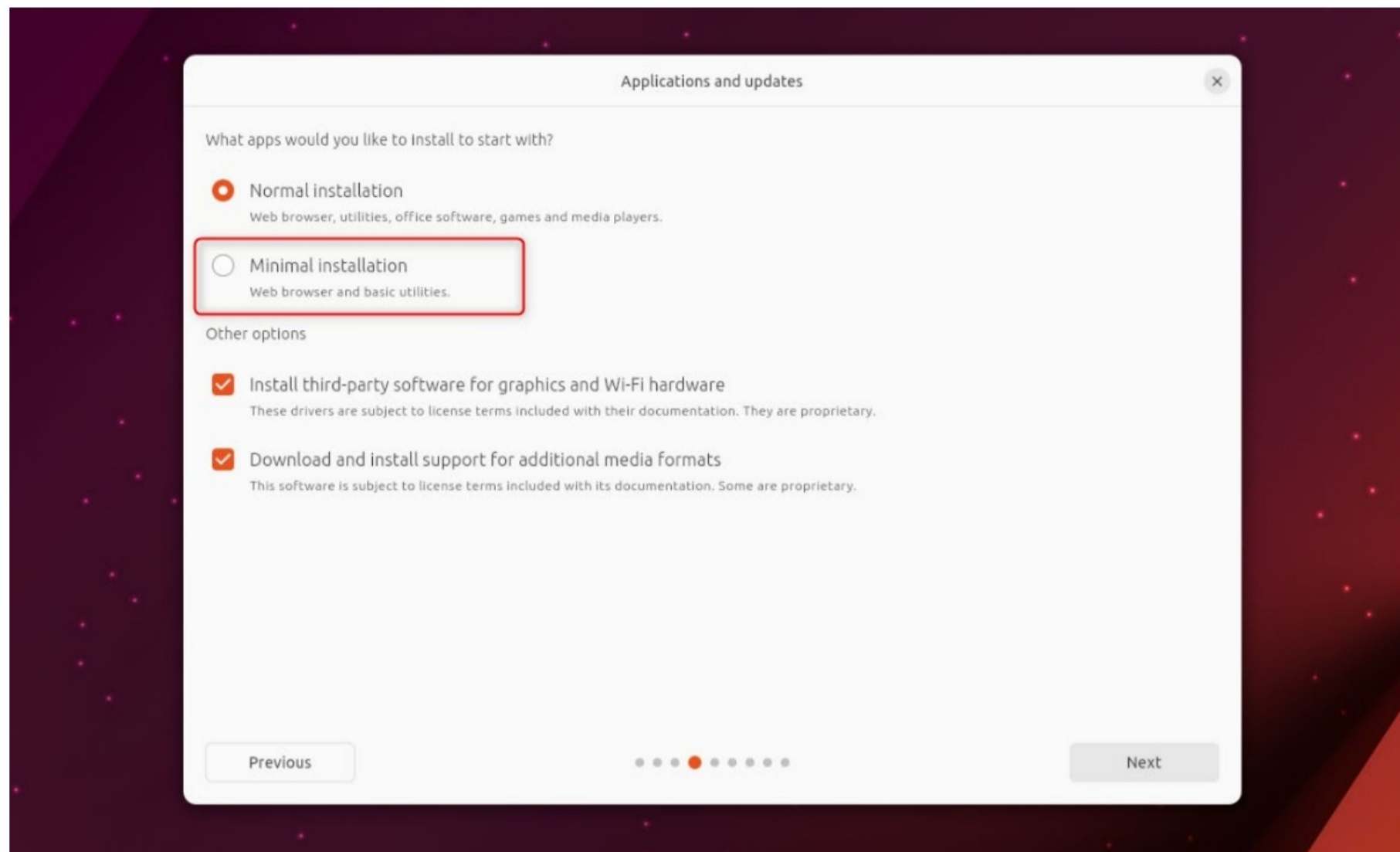


Figure 6.3: Enable Minimal Installation Option

Password Protect BIOS and Bootloader



- BIOS password protects unauthorized users from:
 - Changing the **BIOS settings**
 - Booting the system
- Navigate to BIOS configuration menu to add a password

- Bootloader password protects unauthorized users from:
 - Accessing **Single User mode**
 - Accessing **GRUB console**
 - Accessing **non-secure operating system** in case of presence of dual operating system
- GRUB** and **LILO** are two bootloaders found in Linux

Password Protecting GRUB

```
alice@alice-Virtual-Machine: ~/Desktop
alice@alice-Virtual-Machine:~/Desktop$ grub-mkpasswd-pbkdf2
Enter password:
Reenter password:
PBKDF2 hash of your password is grub.pbkdf2.sha512.10000.27BF
8EF7468CA1BB252FD6CB09C5F50F8C9D68158380DA30285B9E9E62D96841D
3135749A7F0B8CEBA71678D8E44CBB4C6C0F70342757FE8CC70B74E23279F
2C.4EC0D5E71B4AD6282ABD86AC5CB555856A1DC07CC92A29794349DBFC28
6D09C5A5447A55E6BA7784BE822782877B253AD07F6576EF46B2352CE9488
8D011948F
alice@alice-Virtual-Machine:~/Desktop$
```

```
Open 40_custom [Read-Only] Save
/etc/grub.d
type the
# menu entries you want to add after this comment. Be careful not to
change
# the 'exec tail' line above.
set superusers="alice"
password_pbkdf2 alice
grub.pbkdf2.sha512.10000.A99B6921EDAD3801CBF11E200EFF60EBB70CAF546BB5F12
sh Tab Width: 8 Ln 7, Col 23 INS
```

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Password Protect BIOS and Bootloader

BIOS

The Basic Input/Output System (BIOS) represents the firmware installed on the system's motherboard that is used to perform hardware initialization before booting the OS.

Protecting BIOS with a password can prevent unauthorized users from the following:

- Changing BIOS settings, which can block any physical unauthorized access to the BIOS to boot from a diskette or CD-ROM. Otherwise, the attacker enters rescue mode or single-user mode, which can allow them to start processes on the system or copy sensitive information.
- Booting system, in which the attacker is forced to enter a password before the BIOS starts the boot loader.

Setting a BIOS Password

The exact process of setting the BIOS password varies with the motherboard manufacturer. See the system's manual for specific instructions for setting or changing the BIOS password. In case the BIOS password has been forgotten, it can either be reset through jumpers on the motherboard or by disconnecting the CMOS battery. For this reason, lock the system case when needed or see the system's manual before disconnecting the CMOS battery.

Boot Loader

The boot loader is the software that loads the kernel. It is the first software that runs after the BIOS in order to boot the Linux system. When booting a system, the boot loader menu is displayed. The menu lists out boot entries (OS instances that are installed on the system). Two of the most common Linux Boot Loaders are GRUB and LILO.

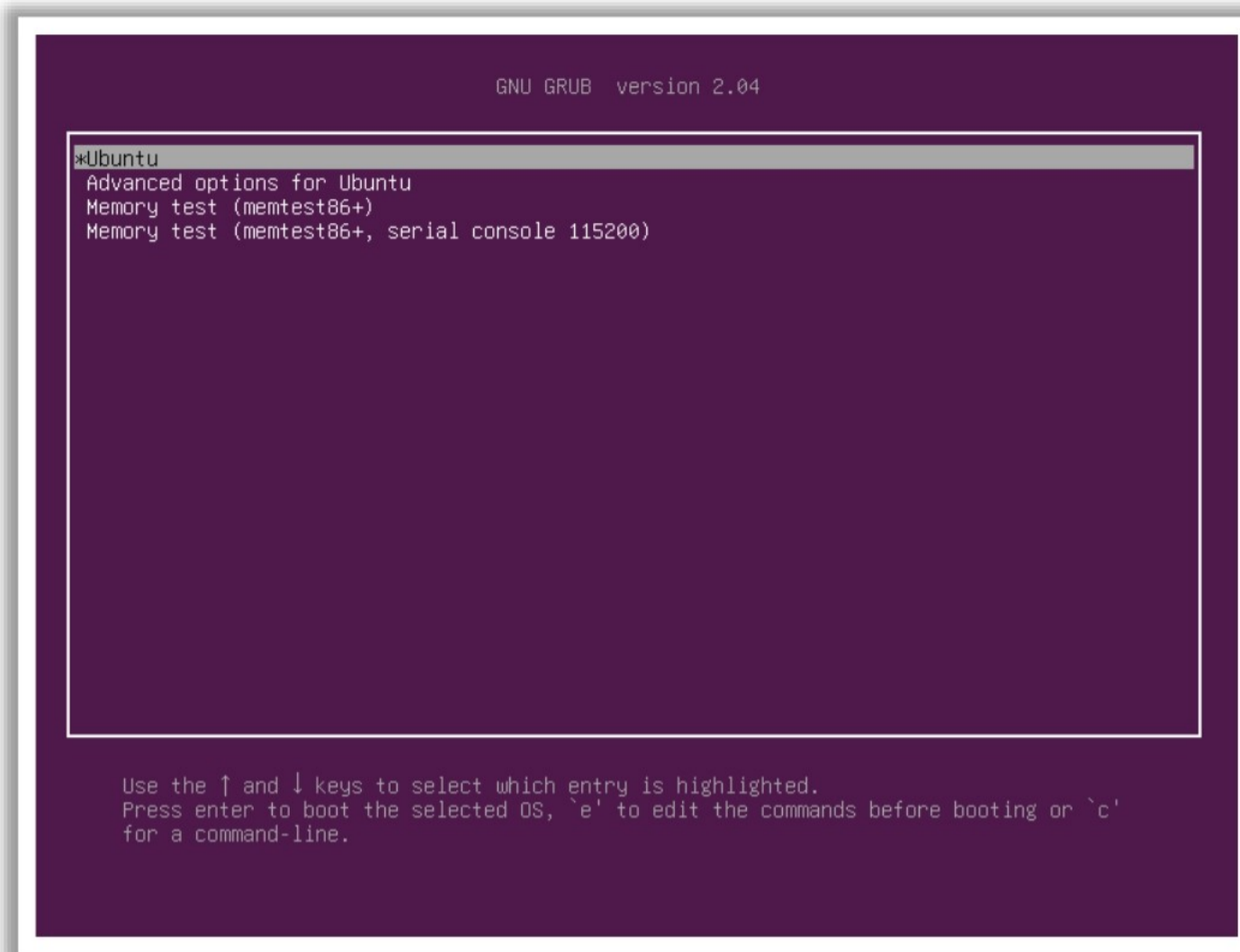


Figure 6.4: GRUB boot loader

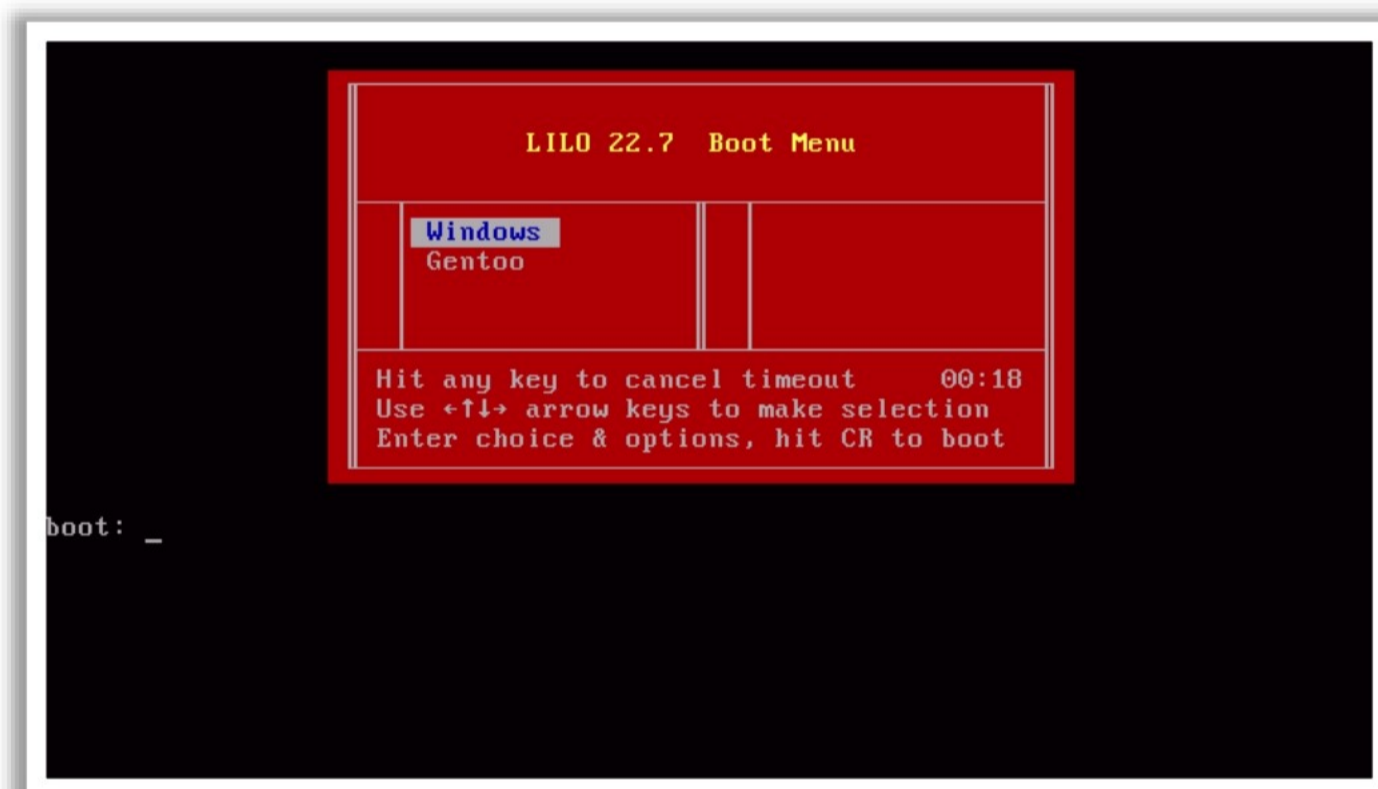


Figure 6.5: LILO boot loader

Anyone can override the existing OS by booting their OS through removable devices. Therefore, protect the boot loader by enabling password protection.

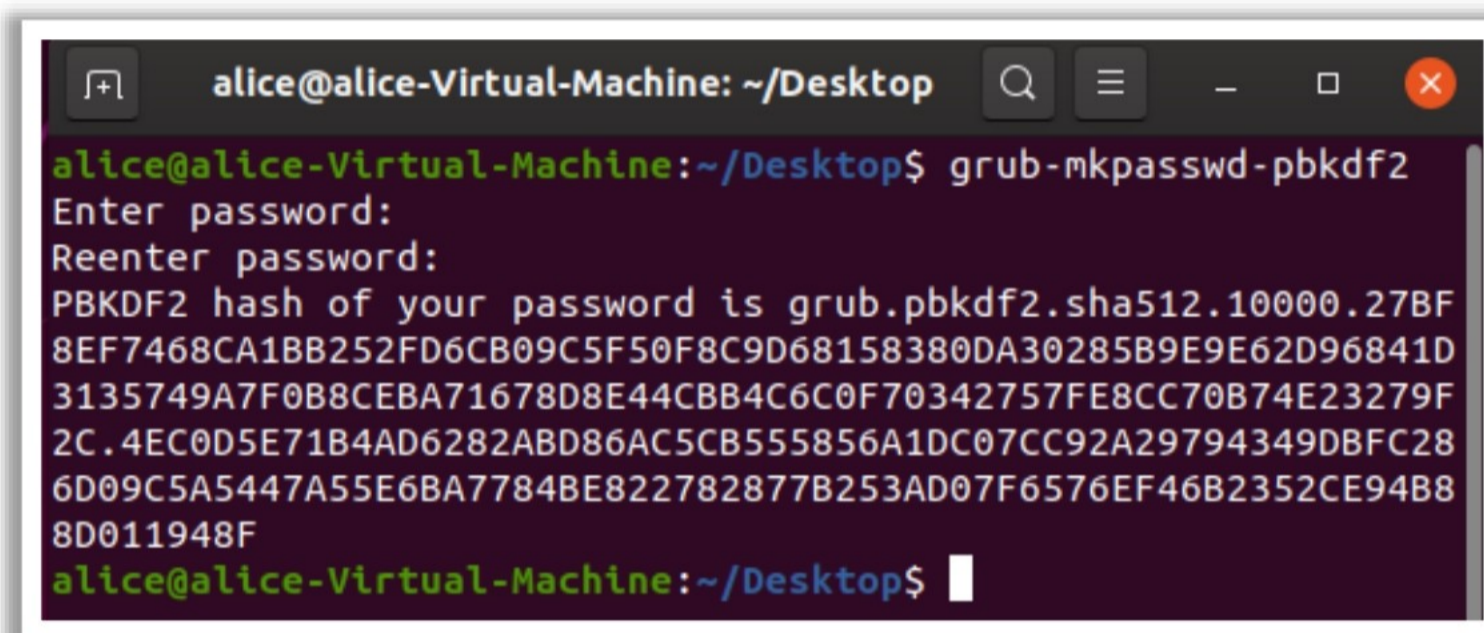
Protecting Linux boot loader with a password can prevent unauthorized users from the following:

- Accessing single-user mode, in which the attacker is blocked from automatic logging in the system as the root user without the root password.
- Accessing the GRUB console, in which the attacker is blocked from using the GRUB editor interface to edit its configuration or to get the information using the `cat` command.

- Accessing non-secure Oses, in which the attacker is blocked from selecting an OS at boot time if it is a dual-boot system. For example, selecting DOS at boot time as it ignores access controls and file permissions.

Steps to Password Protect Ubuntu's Boot Loader

- Generate password hash
 - Type `grub-mkpasswd-pbkdf2` in the Ubuntu terminal and press "Enter" to generate an obfuscated password for GRUB's configuration files. Copy the long string after the "Reenter password:."



```
alice@alice-Virtual-Machine: ~/Desktop
alice@alice-Virtual-Machine:~/Desktop$ grub-mkpasswd-pbkdf2
Enter password:
Reenter password:
PBKDF2 hash of your password is grub.pbkdf2.sha512.10000.27BF
8EF7468CA1BB252FD6CB09C5F50F8C9D68158380DA30285B9E9E62D96841D
3135749A7F0B8CEBA71678D8E44CBB4C6C0F70342757FE8CC70B74E23279F
2C.4EC0D5E71B4AD6282ABD86AC5CB555856A1DC07CC92A29794349DBFC28
6D09C5A5447A55E6BA7784BE822782877B253AD07F6576EF46B2352CE94B8
8D011948F
alice@alice-Virtual-Machine:~/Desktop$
```

Figure 6.6: Password Hash

Note: This step is optional. You can enter the password in plain text in Grub's configuration files, but it obfuscates it and provides extra security.

- Set a password
 - Type `sudo gedit /etc/grub.d/40_custom` to open the `40_custom` file in the Nano text editor, where the custom settings should be saved. Otherwise, they may be overwritten by newer GRUB versions if saved elsewhere.
 - Add a password entry at the bottom of the file:

Set `superusers="name" password_pbkdf2 name [long string from earlier]`



```
Open 40_custom [Read-Only] Save
/etc/grub.d
type the
# menu entries you want to add after this comment. Be careful not to
change
# the 'exec tail' line above.
set superusers="alice"
password_pbkdf2 alice
grub.pbkdf2.sha512.10000.A99B6921EDAD3801CBF11E200EFF60EBB70CAF546BB5F12:
sh Tab Width: 8 Ln 7, Col 23 INS
```

Figure 6.7: Set Password

Here, the superuser (who can edit boot entries) named “alice” is added. Save the file (Ctrl+O), press “Enter” and exit.

- Save the file by pressing Ctrl-O and “Enter,” then press Ctrl-X to exit.

Steps to Password Protect Linux’s LILO

- Restrict `/etc/lilo.conf` to root only.
`# chmod 600 lilo.conf`
- Password protect LILO by modifying the global section of `lilo.conf`.
`password=""`
`restricted`
- Re-run LILO to write the changes:
`# /sbin/lilo`

Give the root password when it prompts. It creates a `/etc/lilo.conf.shs` file, containing a password hash accessible only to root.

Linux Patch Management



- Apply **latest security patches** to keep the Linux kernel and software up-to-date

Method 1

Deploy the Patches Manually

- Download **updated packages** from a distribution's website and manually install it on your system

Method 2

Automate Patching

- Download and install updates using **third-party** patch management software

Commands to Manually Patch Debian-Based Linux OS

apt-get update fetches the list of available updates

```
alice@alice-Virtual-Machine: ~/Desktop
alice@alice-Virtual-Machine:~/Desktop$ sudo apt-get update
[sudo] password for alice:
Hit:1 http://in.archive.ubuntu.com/ubuntu eoan InRelease
Get:2 http://in.archive.ubuntu.com/ubuntu eoan-updates InRel
```

apt-get upgrade strictly upgrades the current packages

```
alice@alice-Virtual-Machine: ~/Desktop
alice@alice-Virtual-Machine:~/Desktop$ sudo apt-get upgrade
Reading package lists... Done
Building dependency tree
Reading state information... Done
```

apt-get dist-upgrade installs new updates

```
alice@alice-Virtual-Machine: ~/Desktop
alice@alice-Virtual-Machine:~/Desktop$ sudo apt-get dist-upgrade
Reading package lists... Done
Building dependency tree
Reading state information... Done
```

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Linux Patch Management

Linux patch management involves managing patches for the applications running on Linux systems. The process of Linux patch management includes the following:

- Scanning Linux endpoints for detecting missing patches
- Downloading patches from the vendor's sites
- Deploying patches to the client systems

Linux patch management helps maintain a productive and secure environment by improving the performance of the systems.

Methods to Manage Linux Systems Using Software Patching Tools

- Manual Patch Deployment: Download the updated packages from a distribution's website and manually install it on systems. Check the distribution's website for the latest patches and updates.

Commands to patch Linux systems manually

- For Debian-based Linux OSes (Debian, Ubuntu, Linux Mint, etc.):

```
sudo apt-get update
```

Fetches the list of available updates

```
alice@alice-Virtual-Machine: ~/Desktop
alice@alice-Virtual-Machine:~/Desktop$ sudo apt-get update
[sudo] password for alice:
Hit:1 http://in.archive.ubuntu.com/ubuntu eoan InRelease
Get:2 http://in.archive.ubuntu.com/ubuntu eoan-updates InRel
```

Figure 6.8: update command

`sudo apt-get upgrade` # Strictly upgrades the current packages

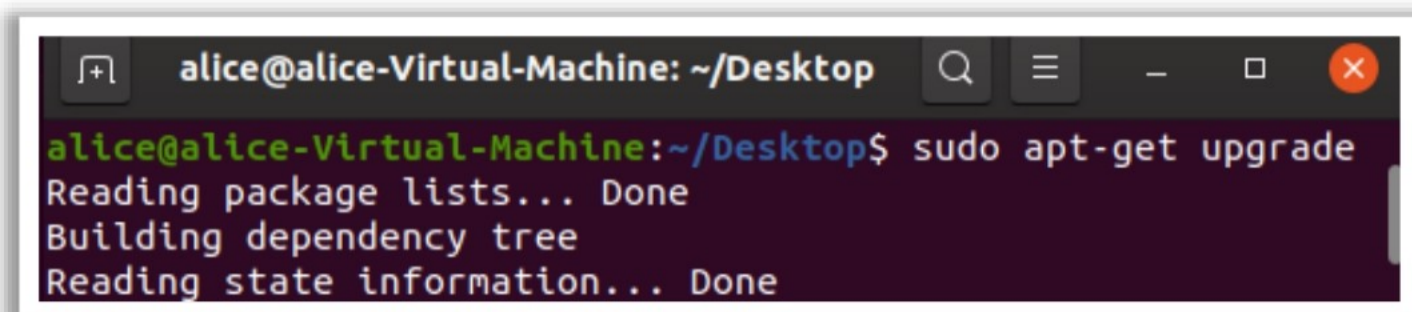
A terminal window titled 'alice@alice-Virtual-Machine: ~/Desktop' showing the command 'sudo apt-get upgrade' being executed. The output is: 'Reading package lists... Done', 'Building dependency tree', and 'Reading state information... Done'.

Figure 6.9: upgrade command

`sudo apt-get dist-upgrade` # Installs updates (new ones)

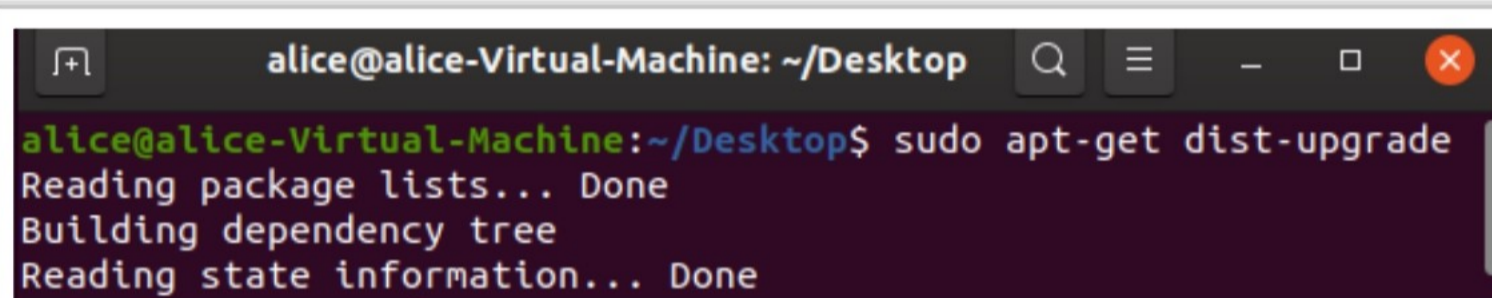
A terminal window titled 'alice@alice-Virtual-Machine: ~/Desktop' showing the command 'sudo apt-get dist-upgrade' being executed. The output is: 'Reading package lists... Done', 'Building dependency tree', and 'Reading state information... Done'.

Figure 6.10: dist-upgrade command

- For Red Hat-based Linux OSes (Red Hat, CentOS, Oracle Linux, etc.):
 - `yum check-update` # To check for the list of available updates
 - `yum update` # Installs updates for all the packages
- For SUSE-based Linux OSes (SUSE Linux Enterprise, OpenSUSE):
 - `zypper check-update` # To check for the list of available updates
 - `zypper update` # Installs updates for all the packages
- For Slackware-based Linux distributions, use `swaret`.
- For other RPM-based Linux distributions, use `autoupdate`.
- Automated Patching/Using Patch Management Software: Manual deployment of patches requires many steps and may result in errors. Moreover, it is difficult to identify where the mistake happens. It is also time-consuming due to the complexities involved. In these cases, third-party patch software enable automated patch management. They can automatically perform the following functions:
 - Scan for missing patches
 - Download patches
 - Test patches in a non-production environment
 - Approve patches to be rolled out in the production environment if they do not cause any issues.
 - Schedule reports

Examples of patch management software include ManageEngine Patch Manager Plus, SapphireIMS, GFI LanGuard, etc.

Linux Hardening Checklist: System Installation and Patching



- ✓ Use latest version for installing the OS and protect the new installed system from malicious network traffic till it is hardened
- ✓ Create a separate volume with the **nodev**, **nosuid**, and **noexec** options set for **/tmp**
- ✓ Create separate volumes for **/var**, **/var/log**, and **/home**
- ✓ Set sticky bit on all writable directories
- ✓ Configure the system to enable automatic software updates

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Linux Hardening Checklist: System Installation and Patching

The necessary steps to make the Linux host secure are listed below:

- Secure a newly installed system from hostile network traffic until finishing the installation and hardening of OS.
- Use the latest version of the OS. Go through the documentation of the vendor support to decide the lifecycle of the version and consider the service packs releases.
- Create separate volume with **nodev**, **nosuid**, and **noexec** options set for **/tmp** to avoid resource exhaustion.
 - Setting **nodev** avoids users from creating or using block or special character devices.
 - Setting **noexec** avoids users from running binary executables from **/tmp**.
 - Setting **nosuid** avoids users from creating **set userid** files in **/tmp**.
- Create individual volumes for **/var**, **/var/log**, and **/home**. Separate the write access of the non-administrative users from the root volume to restrict the impact of those volumes being filled.
- Set sticky bit on all world-writable directories to stop users with write access to the directory and deleting files of the other users.
- Configure the system to receive software updates. For all distributions, this step is a simple configuration change.



LO#03: Discuss Linux OS hardening techniques

Copyright © by **EC-Council**. All Rights Reserved. Reproduction is Strictly Prohibited.

LO#03: Discuss Linux OS Hardening Techniques

The objective of this section is to explain the various techniques that can be used for hardening Linux OS.

Disabling Unnecessary Services



The open ports of services running on the system can be used by the intruders. Hence, to protect the system, **disable** all unnecessary services

Disable unnecessary services using the service command (**systemctl**)

Disable services such as FTP, Telnet, Rlogin / Rsh, etc., if not in use

Command Syntax to stop a service

```
$ sudo systemctl stop [service]
```

Command Syntax to disable a service

```
$ sudo systemctl disable [service]
```

Command Syntax to kill a process

```
$ sudo sudo kill -9 [process_id]
```

List of All Services and Their Status in Ubuntu

| UNIT | LOAD | ACTIVE | SUB | DESCRIPTION |
|---------------------------|-----------|----------|---------|--------------------------|
| accounts-daemon.service | loaded | active | running | Accounts Service |
| acpid.service | loaded | active | running | ACPI event daemon |
| anacron.service | loaded | inactive | dead | Run anacron jobs |
| apparmor.service | loaded | active | exited | Load AppArmor profiles |
| apport-autoreport.service | loaded | inactive | dead | Process error reports wh |
| apport.service | loaded | active | exited | LSB: automatic crash rep |
| apt-daily-upgrade.service | loaded | inactive | dead | Daily apt upgrade and cl |
| apt-daily.service | loaded | inactive | dead | Daily apt download activ |
| auditd.service | not-found | inactive | dead | auditd.service |
| avahi-daemon.service | loaded | active | running | Avahi mDNS/DNS-SD Stack |
| bolt.service | loaded | active | running | Thunderbolt system servi |
| colord.service | loaded | active | running | Manage, Install and Gene |
| connman.service | not-found | inactive | dead | connman.service |
| console-setup.service | not-found | inactive | dead | console-setup.service |
| console-setup.service | loaded | active | exited | Set console font and key |
| cron.service | loaded | active | running | Regular background progr |
| cups-browsed.service | loaded | active | running | Make remote CUPS printer |
| cups.service | loaded | active | running | CUPS Scheduler |
| dbus.service | loaded | active | running | D-Bus System Message Bus |
| dmesg.service | loaded | inactive | dead | Save initial kernel mess |
| dns-clean.service | loaded | inactive | dead | Clean up any mess left b |
| emergency.service | loaded | inactive | dead | Emergency Shell |

Example: To Disable openvpn Service in Ubuntu

```
alice@alice-Virtual-Machine: ~/Desktop
alice@alice-Virtual-Machine:~/Desktop$ sudo systemctl stop openvpn
```

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Disabling Unnecessary Services

Several unwanted applications and services may get installed by default during the OS installation, eating away system resources without knowledge of the user. Open ports of the services running on the system can be exploited by attackers. Therefore, it is recommended to disable such services or applications to secure the system. Use the command `systemctl` to disable unnecessary services in the system.

- Syntax to stop a service: `$ sudo systemctl stop [service]`
- Syntax to disable a service: `$ sudo systemctl disable [service]`

For example: To disable openvpn service in Ubuntu, follow the command:

```
alice@alice-Virtual-Machine: ~/Desktop
alice@alice-Virtual-Machine:~/Desktop$ sudo systemctl stop openvpn
```

Figure 6.11: stop command

- Syntax to kill a process: `$ sudo kill -9 [process_id]`
For example: `# sudo kill -9 1511`

```
alice@alice-Virtual-Machine: ~/Desktop
alice@alice-Virtual-Machine:~/Desktop$ sudo kill 9 1511
[sudo] password for alice:
alice@alice-Virtual-Machine:~/Desktop$
```

Figure 6.12: kill command

To prevent the exploitation of known vulnerabilities, list out all the installed packages on Linux OS/Servers and remove the unnecessary ones. If it is a Linux server, it is strictly recommended to remove the following unnecessary services as servers require the least number of installed applications and services on them.

Telnet-Server

It comprises the telnetd daemon, which allows connections from users from other systems through the telnet protocol. The insecure and unencrypted telnet protocol could allow attackers to sniff network traffic, who can then steal credentials.

- Steps to remove telnet server in Red Hat:
 - Enter the command to know the telnet-server package is on the system.
`# rpm -q telnet-server package telnet-server is not installed`
 - Enter the command to remove the telnet-server from a Linux system.
`# yum erase telnet-server`

RSH-Server

The Berkeley rsh-server (rsh, rlogin, rcp) package comprises legacy services that exchange credentials in clear-text. This service comprises many security exposures that can be exploited.

- Steps to remove RSH-Server:
 - Enter the command to know the RSH-server package is on the system.
`# rpm -q rsh-server package rsh-server is not installed`
 - Enter the command to remove the RSH-server from a Linux system.
`# yum erase rsh-server`

NIS-Server

The client-server directory service protocol, network information server (NIS) (formerly, Yellow Pages), tends to distribute system configuration files. Also, it is an insecure system that is been vulnerable to DoS attacks, buffer overflows, and poor authentication for querying NIS maps.

- Steps to remove NIS-Server
 - Enter the command to know the NIS-Server package is on the system.
`# rpm -q ypserv package ypserv is not installed`
 - Enter the command to remove the NIS-server from a Linux system.
`# yum erase ypserv`

TFTP-Server

Trivial file transfer protocol (TFTP) is used to transfer configuration or boot machines from a boot server automatically. It does not support authentication or encryption to provide Confidentiality and Data Integrity protection.

- Steps to remove TFTP-Server:

- Enter the command to know the TFTP-Server package is on the system.
rpm -q tftp-server package tftp-server is not installed
- Enter the command to remove the TFTP-server from a Linux system.
yum erase tftp-server

TALK-Server

The TALK software allows sending and receiving messages across systems through a terminal session. It poses security risks as it uses unencrypted protocols for communication.

- Steps to remove TALK-Server:
 - Enter the command to know the Talk-Server package is on the system.
rpm -q talk-server package talk-server is not installed
 - Enter the command to remove the Talk-server from a Linux system.
yum erase talk-server

Remove or Uninstall Unnecessary Software/Packages



- Uninstall unnecessary software to protect the system from vulnerabilities in software
- To uninstall unnecessary software's, review the installed software using the package manager like **apt-get**, **dpkg**, or **yum** and delete all unwanted packages
- Use tools like **UnusedPkg diagnostics** and **Deborphan** to list out and remove the unused packages or libraries in a Linux distribution

Apt-get autoremove: Removes Libs and Packages that Were Installed Automatically

```
root@alice-Virtual-Machine: /home/alice# apt-get autoremove
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages will be REMOVED:
libavahi-core7
0 upgraded, 0 newly installed, 1 to remove and 33 not upgraded.
After this operation, 280 kB disk space will be freed.
Do you want to continue? [Y/n] y
(Reading database ... 221555 files and directories currently installed.)
Removing libavahi-core7:amd64 (0.7-4ubuntu5) ...
Processing triggers for libc-bin (2.30-0ubuntu2) ...
root@alice-Virtual-Machine: /home/alice#
```

dpkg: List Displays All Installed Packages

```
root@alice-Virtual-Machine: /home/alice/Desktop# dpkg --get-selections
Desired=Unknown/Install/Remove/Purge/Hold
| Status=Not/Inst/Conf-files/Unpacked/halF-conf/Half-inst/trig-Await/Trig-pend
| / Err?=(none)/Reinst-required (Status,Err: uppercase=bad)
||/ Name                  Version                     Architecture Description
+++-+-----+-----+-----+-----+
ii  accountsservice          0.6.55-0ubuntu10           amd64        query and manipulat
ii  acl                      2.2.53-4                   amd64        access control list
ii  acpi-support              0.143                      amd64        scripts for handlin
ii  acpid                    1:2.0.31-1ubuntu2          amd64        Advanced Configurati
ii  adduser                   3.118ubuntu1               all          add and remove user
ii  adwaita-icon-theme        3.34.0-1ubuntu1            all          default icon theme
ii  alsasound                 1:3.22.9-1                 amd64        GNOME solitaire car
ii  alsa-base                 1.0.25+dfsg-0ubuntu5       all          ALSA driver configu
ii  alsa-utils                1.1.9-0ubuntu1             amd64        Utilities for confi
ii  amd64-microcode           3.20191021.1+really3.20181128.1ubuntu2 amd64        Processor microcode
ii  anacron                   2.3-29                     amd64        cron-like program t
ii  apt                        2.2.3+dfsg-1.5             amd64        Automated Password
ii  app-install-data-partner  19.04                      all          Application Install
```

apt remove [package Name]: Uninstalls the Package

```
alice@alice-Virtual-Machine: ~/Desktop$ sudo apt remove ftp
[sudo] password for alice:
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages will be REMOVED:
ftp ubuntu-standard
0 upgraded, 0 newly installed, 2 to remove and 39 not upgraded.
After this operation, 196 kB disk space will be freed.
Do you want to continue? [Y/n] Y
(Reading database ... 219386 files and directories currently installed.)
Removing ubuntu-standard (1.440) ...
Removing ftp (0.17-34.1) ...
Processing triggers for man-db (2.8.7-3) ...
alice@alice-Virtual-Machine: ~/Desktop$
```

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Remove or Uninstall Unnecessary Software/Packages

Downloading packages from untrusted sources and leaving older versions of packages installed can introduce vulnerabilities or waste resources needed for more important tasks. Installing and uninstalling applications may leave many dependent files that have no use. Use the following commands to remove any partial package and remove any unused dependencies.

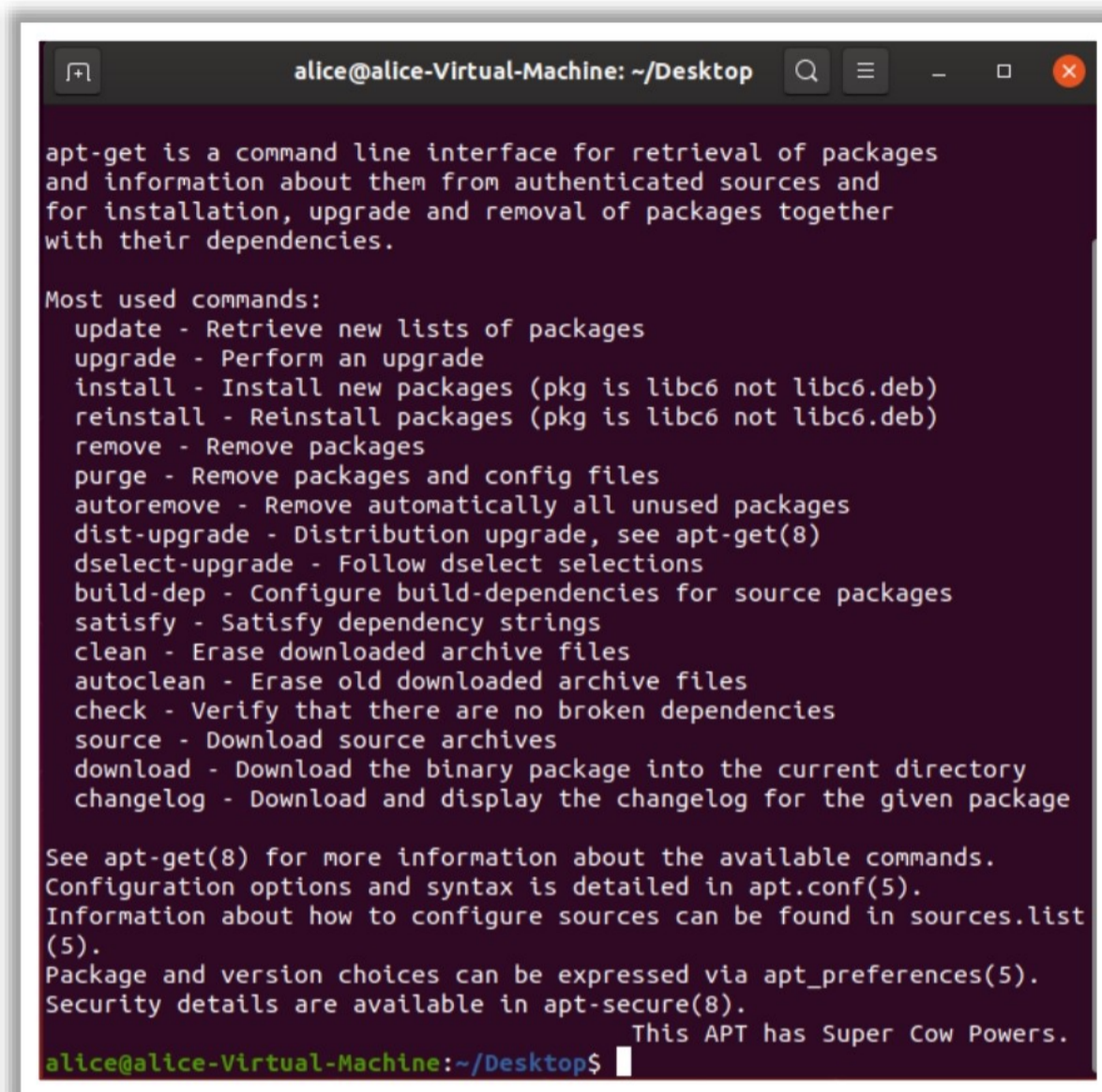
- Use command `sudo apt autoclean` to clean partial packages.
- Use command `sudo apt-get clean` to clean the apt cache.
- Use command `sudo apt autoremove application-name uninstall` or remove unnecessary packages.

```
root@alice-Virtual-Machine: /home/alice/Desktop# apt-get autoremove
Reading package lists... Done
Building dependency tree
Reading state information... Done
0 upgraded, 0 newly installed, 0 to remove and 39 not upgraded.
root@alice-Virtual-Machine: /home/alice/Desktop#
```

Figure 6.13: autoremove command

Using autoremove in Ubuntu

The administrator can use various techniques/tools to remove or uninstall unused packages in Linux using the `apt-get` command.

A terminal window titled 'alice@alice-Virtual-Machine: ~/Desktop' showing the help text for the 'apt-get' command. The text describes its purpose and lists various subcommands like 'update', 'install', 'remove', and 'autoremove'. It also mentions configuration files and security details.

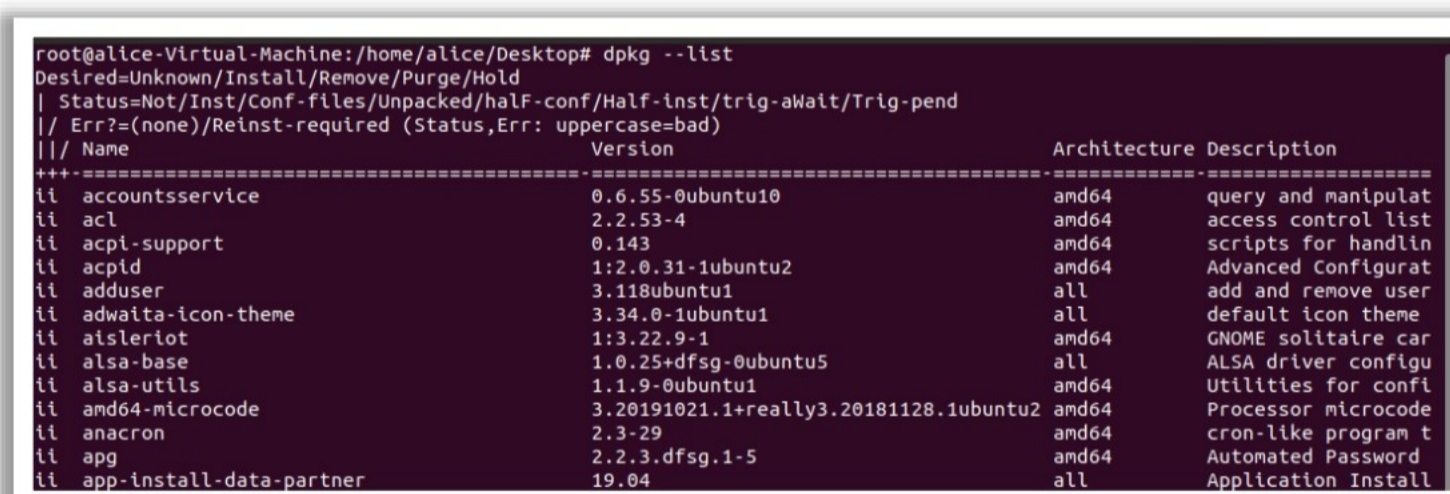
```
alice@alice-Virtual-Machine: ~/Desktop$ apt-get --help
apt-get is a command line interface for retrieval of packages
and information about them from authenticated sources and
for installation, upgrade and removal of packages together
with their dependencies.

Most used commands:
  update - Retrieve new lists of packages
  upgrade - Perform an upgrade
  install - Install new packages (pkg is libc6 not libc6.deb)
  reinstall - Reinstall packages (pkg is libc6 not libc6.deb)
  remove - Remove packages
  purge - Remove packages and config files
  autoremove - Remove automatically all unused packages
  dist-upgrade - Distribution upgrade, see apt-get(8)
  dselect-upgrade - Follow dselect selections
  build-dep - Configure build-dependencies for source packages
  satisfy - Satisfy dependency strings
  clean - Erase downloaded archive files
  autoclean - Erase old downloaded archive files
  check - Verify that there are no broken dependencies
  source - Download source archives
  download - Download the binary package into the current directory
  changelog - Download and display the changelog for the given package

See apt-get(8) for more information about the available commands.
Configuration options and syntax is detailed in apt.conf(5).
Information about how to configure sources can be found in sources.list
(5).
Package and version choices can be expressed via apt_preferences(5).
Security details are available in apt-secure(8).
This APT has Super Cow Powers.
alice@alice-Virtual-Machine:~/Desktop$
```

Figure 6.14: apt-get command

Use the script-based `dpkg -l` to list all the installed package in Linux

A terminal window showing the output of the 'dpkg -l' command. It lists installed packages with columns for Name, Version, Architecture, and Description. Packages listed include accountsservice, acl, acpi-support, acpid, adduser, adwaita-icon-theme, amleriot, alsa-base, alsa-utils, amd64-microcode, anacron, apt, and app-install-data-partner.

```
root@alice-Virtual-Machine:/home/alice/Desktop# dpkg --get-selections
Desired=Unknown/Install/Remove/Purge/Hold
| Status=Not/Inst/Conf-files/Unpacked/halF-conf/Half-inst/trig-aWait/Trig-pend
|/ Err?=(none)/Reinst-required (Status,Err: uppercase=bad)
||/ Name          Version          Architecture Description
+++=-----+-----+-----+-----+
ii  accountsservice  0.6.55-0ubuntu10 amd64      query and manipulat
ii  acl              2.2.53-4         amd64      access control list
ii  acpi-support     0.143           amd64      scripts for handlin
ii  acpid            1:2.0.31-1ubuntu2 amd64      Advanced Configurat
ii  adduser          3.118ubuntu1     all        add and remove user
ii  adwaita-icon-theme 3.34.0-1ubuntu1  all        default icon theme
ii  amleriot         1:3.22.9-1       amd64      GNOME solitaire car
ii  alsa-base        1.0.25+dfsg-0ubuntu5 all        ALSA driver configu
ii  alsa-utils       1.1.9-0ubuntu1   amd64      Utilities for confi
ii  amd64-microcode  3.20191021.1+really3.20181128.1ubuntu2 amd64      Processor microcode
ii  anacron          2.3-29          amd64      cron-like program t
ii  apt              2.2.3.dfsg.1-5   amd64      Automated Password
ii  app-install-data-partner 19.04          all        Application Install
```

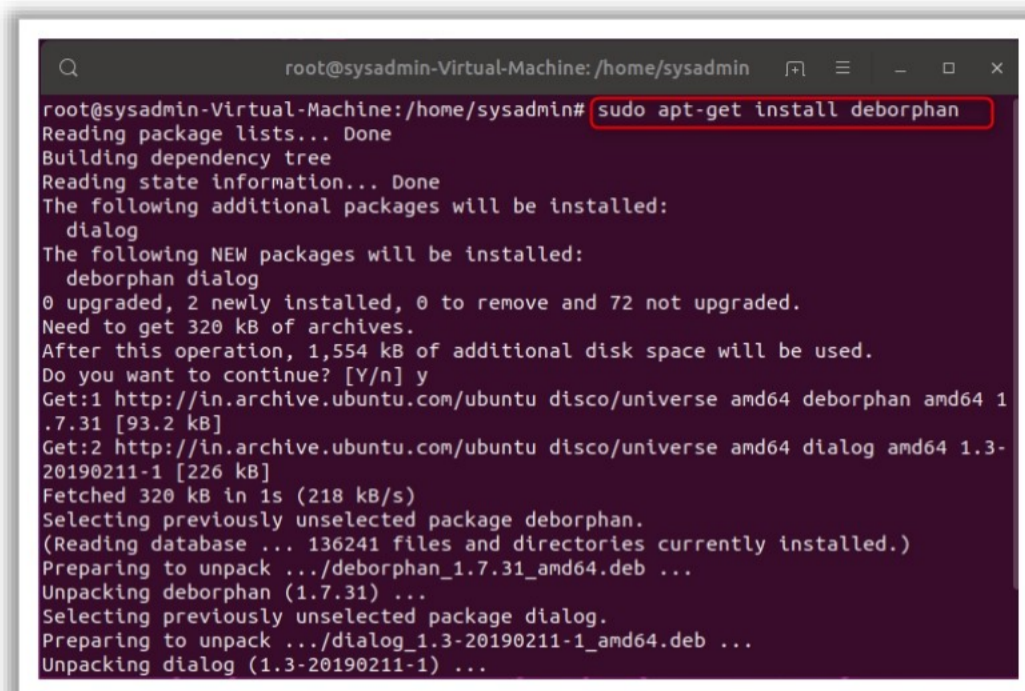
Figure 6.15: dpkg -l command

Use Debtorphan, a powerful tool that can list unused packages or libraries in a Linux distribution. It will check for packages that are no longer used so that the user can easily remove these entry points and keep the system clean.

Steps to install Debtorphan in Linux:

- Open terminal.

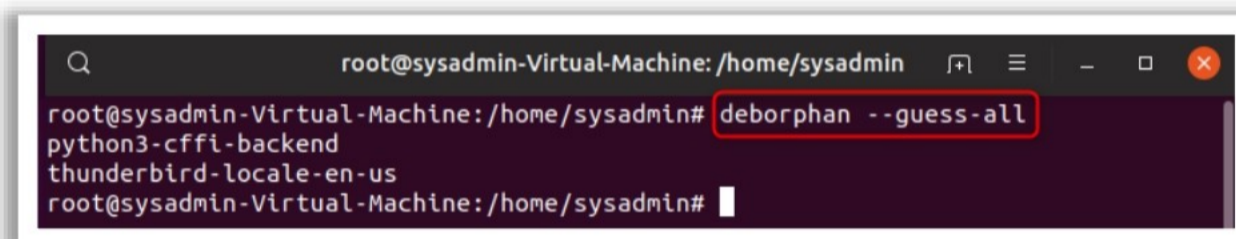
- Type the command `sudo apt-get install deborphan`.



```
root@sysadmin-Virtual-Machine: /home/sysadmin
root@sysadmin-Virtual-Machine: /home/sysadmin# sudo apt-get install deborphan
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  dialog
The following NEW packages will be installed:
  deborphan dialog
0 upgraded, 2 newly installed, 0 to remove and 72 not upgraded.
Need to get 320 kB of archives.
After this operation, 1,554 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://in.archive.ubuntu.com/ubuntu disco/universe amd64 deborphan amd64 1.7.31 [93.2 kB]
Get:2 http://in.archive.ubuntu.com/ubuntu disco/universe amd64 dialog amd64 1.3-20190211-1 [226 kB]
Fetched 320 kB in 1s (218 kB/s)
Selecting previously unselected package deborphan.
(Reading database ... 136241 files and directories currently installed.)
Preparing to unpack .../deborphan_1.7.31_amd64.deb ...
Unpacking deborphan (1.7.31) ...
Selecting previously unselected package dialog.
Preparing to unpack .../dialog_1.3-20190211-1_amd64.deb ...
Unpacking dialog (1.3-20190211-1) ...
```

Figure 6.16: install deborphan command

- Use the command `deborphan --guess-all` to list out the unused packages.



```
root@sysadmin-Virtual-Machine: /home/sysadmin
root@sysadmin-Virtual-Machine: /home/sysadmin# deborphan --guess-all
python3-cffi-backend
thunderbird-locale-en-us
root@sysadmin-Virtual-Machine: /home/sysadmin#
```

Figure 6.17: deborphan --guess-all command

- The following commands show how to remove the unused packages and library.

```
sudo deborphan --guess-data | xargs sudo aptitude -y purge
```

```
sudo deborphan | xargs sudo apt-get -y remove -purge
```

Ubuntu uses repositories (servers) for handling software installations. These repositories comprise packages (software and dependencies) audited by Canonical, Ubuntu developers, and the security teams.

The Ubuntu repositories are of four types: Main, Universe, Restricted, and Multiverse. The Ubuntu team does not audit all repositories. Of these four types, the Multiverse repository offers paid software, while the Restricted repository offers free downloads but does not provide full support. Using these repositories may harm the systems as they can provide harmful applications sometimes. Restricting downloading and updating packages or applications from these unsafe and restricted repositories can be the best option to avoid installing harmful packages.

Steps to disable unsafe repositories:

- Click “Show Applications” in Ubuntu Software Center.

- Select “Software & Updates”.



Figure 6.18: Software & Updates

- Check the following options under the “Ubuntu Software” tab.

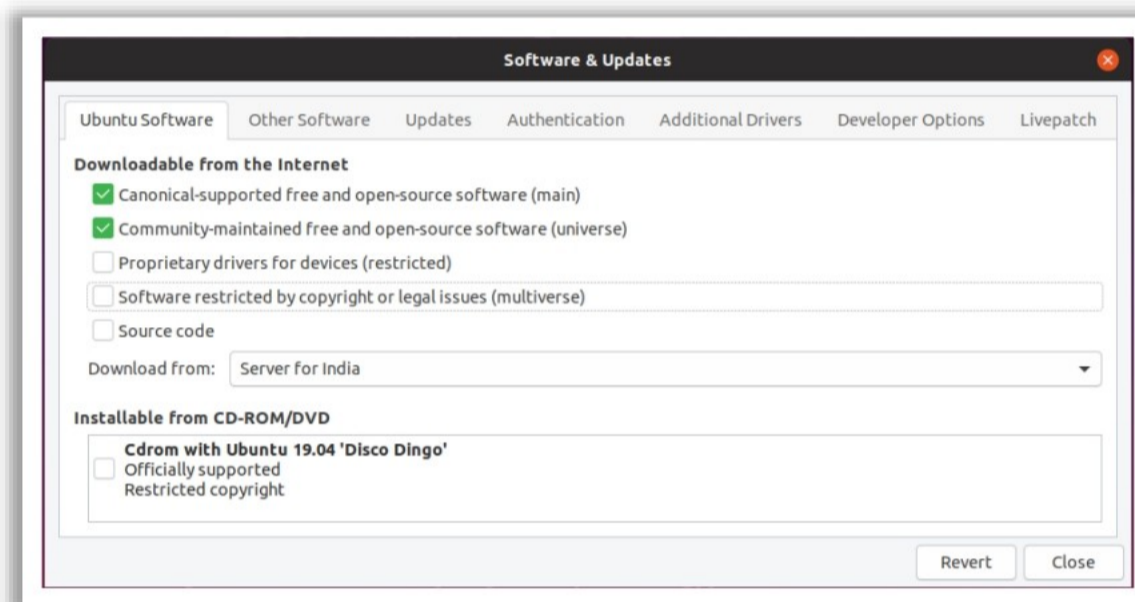


Figure 6.19: Ubuntu Software tab

- Click the “Other Software” tab and uncheck the “Canonical Partners” option.

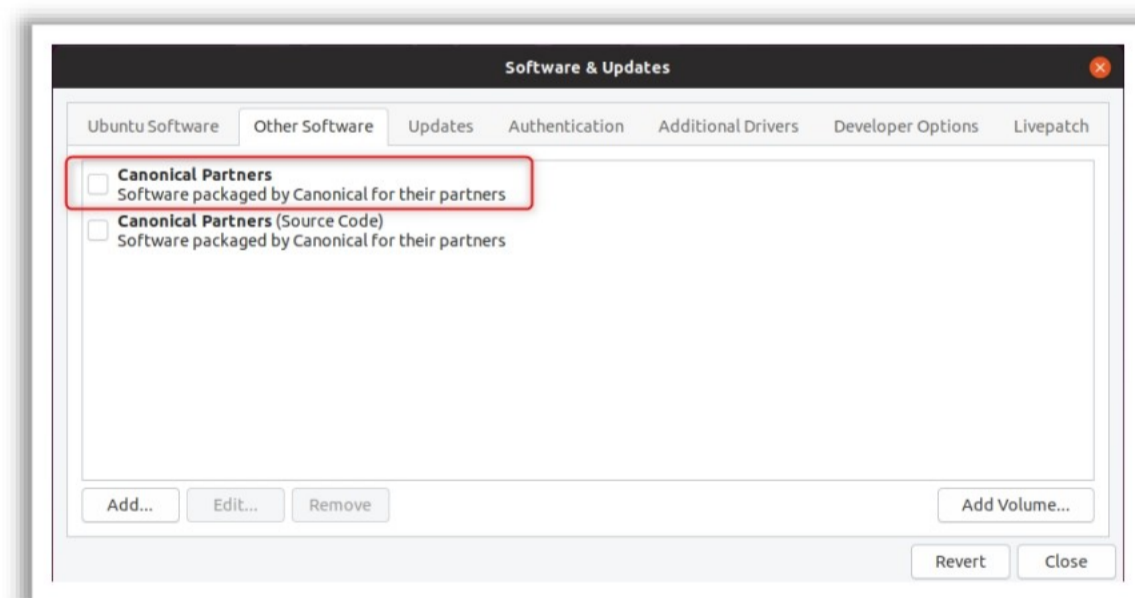


Figure 6.20: Other Software tab

- Close the window. If prompted for update, click to update.

Once the settings are finished, Ubuntu automatically downloads updates daily.

Install Antivirus



- Install antivirus software to protect **uploading infected files** to Linux

The image shows two terminal windows side-by-side. The left window is titled 'alice@alice-Virtual-Machine: ~' and shows the command 'clamscan --help' being executed. The output displays the ClamAV Bytecode Testing Tool version 0.103.9, copyright information for the ClamAV Team and Cisco Systems, Inc., and a list of command-line options including --help, --version, --debug, --force-interpret, --trust-bytecode, --info, --printsrc, --printbcir, --input, --trace, --no-trace-showsource, and --statistics-bytecode. The right window is titled 'root@alice-Virtual-Machine: ~' and shows the command 'clamscan' being executed. The output shows a scan summary for the root directory, indicating that all files scanned are OK, with a scan time of 5.353 seconds.

Antivirus Example: Clamscan

Antivirus Example: Clamscan

Source: <https://www.clamav.net>

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Install Antivirus

Installing an antivirus software can protect the Linux system from infected files. Both open source and commercial antivirus software are available for Linux. Standalone solutions and enterprise solutions (with centralized management) for larger scale deployments are also available.

ClamAV

Source: <https://www.clamav.net>

ClamAV is an open-source (GPLv2) antivirus engine for detecting Trojans, viruses, malware, and other malicious threats.

To install ClamAV on Linux, the user does not need all packages provided by different Linux distributions. Know which packages are needed when installing the bare minimum.

- On Debian, follow the commands.

```
apt-get update  
apt-get install clamav
```
- On RHEL/CentOS, follow the commands.

```
yum install -y epel-release  
yum install -y clamav
```
- On Fedora, follow the command.

```
yum install -y clamav clamav-update
```

Linux System Integrity Checking: Secure Boot



- ✓ Secure Boot feature ensures the **integrity** and **authenticity** of the bootloader and essential system files during the boot process
- ✓ It protects against **unauthorized or malicious code** being loaded before the operating system kernel
- ✓ To use Secure Boot on a Linux system, the system must support **UEFI firmware** with Secure Boot support
- ✓ Enable or disable Secure Boot in the BIOS or UEFI settings
- ✓ To boot a Linux system with Secure Boot enabled, use a **signed bootloader** (e.g., GRUB2) and a signed kernel. Secure Boot requires that these components have valid signatures from a trusted **certificate authority (CA)**
- ✓ Checking the **integrity** of the Linux bootloader is a crucial step in ensuring the security and stability of your system
 - Access the **Bootloader Configuration**
 - Check **Bootloader Configuration File**
 - Verify **File Integrity**
 - Check **Bootloader Options**
 - Verify **Bootloader Signature (UEFI Secure Boot)**

Copyright © by **EC-Council**. All Rights Reserved. Reproduction is Strictly Prohibited.

Linux System Integrity Checking: Secure Boot

Secure Boot feature ensures the integrity and authenticity of the bootloader and essential system files during the boot process. It protects against unauthorized or malicious code being loaded before the operating system kernel. To use Secure Boot on a Linux system, the system must support UEFI firmware with Secure Boot support. It can be enabled or disabled in the BIOS or UEFI settings. To boot a Linux system with Secure Boot enabled, use a signed bootloader (e.g., GRUB2) and a signed kernel. Secure Boot requires that these components have valid signatures from a trusted certificate authority (CA). Checking the integrity of the Linux bootloader is a crucial step in ensuring the security and stability of your system.

Steps to Use Secure Boot

- Access the **bootloader configuration** that is present in the **BIOS** or **UEFI settings**.
- Check the **bootloader configuration file** to ensure the secure boot is enabled.
- Verify file integrity by using checksums or digital signatures to confirm the authenticity of the installation files.
- Configure bootloader options that are only from sources that are trusted. It can be done using **BIOS/UEFI settings**. Disable the boot options that are not necessary.
- Verify the bootloader signature (**UEFI Secure Boot**) appended to all subsequent partition tables and app images before they are booted.

Linux System Integrity Checking using Package Integrity Verification



Identify Linux system integrity checking using package integrity verification involves verifying the authenticity and integrity of software packages installed on a Linux system to ensure they have not been tampered with or compromised

Package Managers

- **APT (Debian/Ubuntu):** APT uses GPG (GNU Privacy Guard) keys to verify package signatures. Ensure that the GPG keys for the repository are correctly configured
- **YUM/DNF (Red Hat/Fedora):** YUM and DNF use GPG keys as well. Ensure that the repository GPG keys are correctly imported
- **zypper (openSUSE):** zypper also relies on GPG keys for package verification. Make sure the keys are up to date

Verify Package Signatures Manually

- Manually verify the integrity of a package by checking its GPG signature
- To check the signature of a .rpm package:

```
rpm --checksig package.rpm
```
- To check the signature of a .deb package:

```
dpkg-sig --verify package.deb
```



Always use official or trusted repositories; downloading packages from untrusted sources may contain compromised or malicious packages

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Linux System Integrity Checking using Package Integrity Verification

Linux system integrity checking using package integrity verification involves verifying the authenticity and integrity of software packages installed on a Linux system to ensure they have not been tampered with or compromised. Package managers are fundamental components of the software management process on Linux operating systems. They facilitate the installation, update, and removal of software packages. One of the key characteristics of a package manager is its capability to validate packages before installation or update. Always use the official or trusted repositories; downloading packages from untrusted sources may contain compromised or malicious packages.

Package Managers

- **APT (Debian/Ubuntu):** Advanced Package Tool (APT) is a package management system for Linux distributions such as Debian and Ubuntu. It provides installation of new software packages, updating the package list index, and upgrading existing software packages. It uses GPG (GNU Privacy Guard) keys to verify package signatures. Ensure that the GPG keys for the repository are correctly configured.

Steps to Install and Remove a Package

- Using apt, install a package such as nmap (network scanner). Use the following command:

```
sudo apt install nmap
```
- Remove a package that is installed using the following command:

```
sudo apt remove nmap
```

Steps to Update Package Index and Upgrade Packages

- To update a package index with the latest changes made in the repositories, use the command:

```
Sudo apt update
```
- To upgrade the system, update the packages and use the following command:

```
sudo apt upgrade
```
- **YUM/DNF (Red Hat/Fedora):** YellowDog Updater Modified (YUM) is a packet manager used to install, remove, or update software packages for Linux distributors. Dandified YUM is an improved version of YUM and provides more functionality than YUM for installing, removing, or updating a software package. These managers use config files in /etc/yum.repos.d to specify a URL for the GPG key used to verify packages in that repository. Ensure that the repository GPG keys are correctly imported.
- **zypper (openSUSE):** zipper is a command-line package manager for SUSE Linux that is used for installing, updating, or removing packages. It is also used for managing repositories. It relies on GPG keys for package verification. Make sure the keys are up to date.

Verify Package Signatures Manually

Manually verify the integrity of a package by checking its GPG signature.

- To check the signature of a .rpm package:

```
rpm --checksig package.rpm
```
- To check the signature of a .deb package

```
dpkg-sig --verify package.deb
```

CND
Certified Network Defense

- ### Using chkrootkit

- ### Using rkhunter

- ```
alice@alice-Virtual-Machine: ~
alice@alice-Virtual-Machine: $ sudo chrootkit
rootkit is //
Checking and... not found
Checking basename... not infected
Checking diff... not found
Checking chfn... not infected
Checking chsh... not infected
Checking cpio... not infected
Checking crontab... not infected
Checking date... not infected
Checking dd... not infected
Checking diffname... not infected
Checking echo... not infected
Checking egrep... not infected
Checking find... not infected
Checking find... not infected
Checking fingerpd... not found
Checking gpg... not found
Checking grep... not infected
Checking hdparm... not infected
Checking ifconfig... not infected
Checking ifconfig... not found
Checking lnstd... not tested
Checking netcatconf... not found
Checking identd... not found
```

```
root@alice-Virtual-Machine: /h...
root@alice-Virtual-Machine: /home/alice# rkhunter -c
[Rootkit Hunter version 1.4.6]

Checking system commands...

Performing 'strings' command checks
Checking 'strings' command [OK]

Performing 'shared libraries' checks
Checking for preloading variables [None found]
Checking for preloaded libraries [None found]
Checking LD_LIBRARY_PATH variable [Not found]

Performing file properties checks
Checking for prerequisites [OK]
/usr/sbin/odometer [OK]
/usr/sbin/chroot [OK]
/usr/sbin/cron [OK]
/usr/sbin/dmccp [OK]
/usr/sbin/fuse [OK]
/usr/sbin/fusefsck [OK]
/usr/sbin/groupadd [OK]
/usr/sbin/groupdel [OK]
/usr/sbin/groupmod [OK]
/usr/sbin/grpck [OK]
/usr/sbin/lsnconf [OK]
```

## Running rkhunter to Detect Rootkits

Copyright © by **EC-Council**. All Rights Reserved. Reproduction is Strictly Prohibited.

- Navigate to the directory and run the following command to hunt rootkits in the system:

`./chkrootkit`

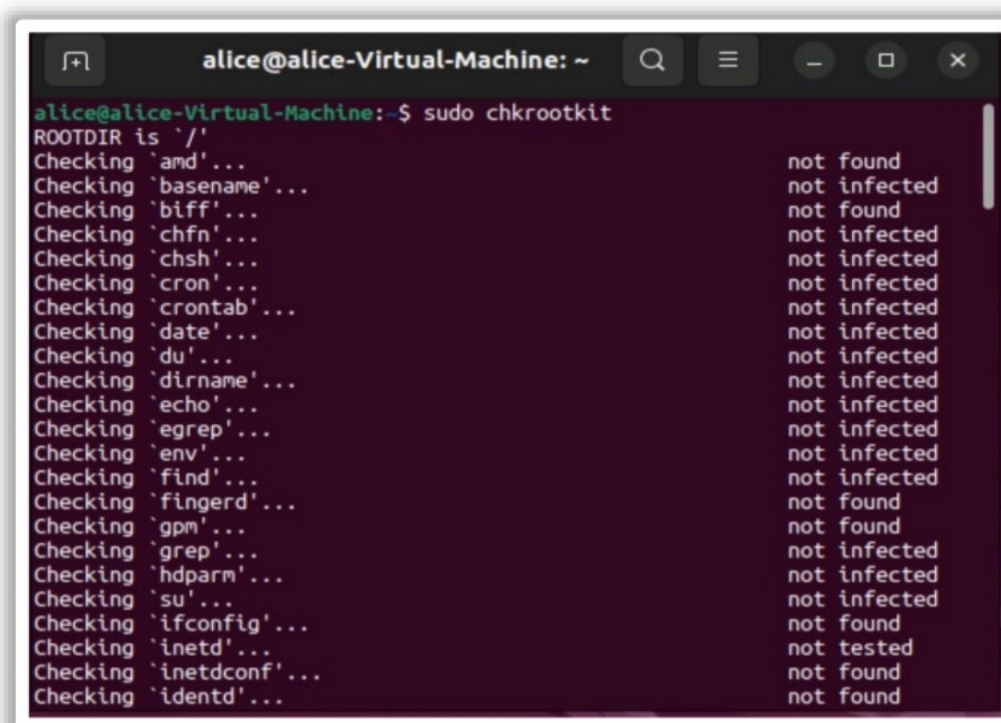


Figure 6.21: Chkrootkit Installation

### Rootkit Detection Using rkhunter

Alternatively, to find rootkits in Linux distributions, use the rkhunter command-line tool. It detects backdoors and remote exploits in the system by running the command rkhunter after installing the tool. It performs additional network tests and kernel module checks, which can be used for forensic analysis.

#### Steps for Using rkhunter

- Install rkhunter

It requires Extra Packages for the Enterprise Linux (EPEL) repository to be installed before installing rkhunter.

```
dnf install epel-release
```

```
dnf install rkhunter
```

- Configure rkhunter

To change the configure file, run

```
vi /etc/rkhunter.conf
```

and search for `#MAIL-ON-WARNING=me@mydomain root@mydomain` and remove it and change it to `root@whatever_the_server_name_is`

configure email and ensure it will work correctly

- Run rkhunter manually to generate a list of warnings to test your email settings

```
rkhunter --check
```

- To generate a clean rkhunter.dat file that rkhunter will use from this point forward as a baseline for further checks, run the following command:

```
rkhunter --propupd
```

## Linux Integrity Subsystem



Linux Integrity Subsystem is a security framework designed to enhance the **security** and **integrity** of a Linux-based operating system

It enables **monitoring**, **verifying**, and **protecting** the integrity of critical system files, processes, and configurations

The use of strong cryptographic techniques and key management is fundamental to its effectiveness in maintaining **system security**

Major components of the kernel integrity subsystem are as follows:

- ✓ **Integrity Measurement Architecture (IMA)**: it measures, records, and evaluates its hash in order to preserve its content
- ✓ **Extended Verification Module (EVM)**: It monitors changes in the file's extended attributes

Copyright © by **EC-Council**. All Rights Reserved. Reproduction is Strictly Prohibited.

### Linux Integrity Subsystem

Linux Integrity Subsystem is a security framework that helps Linux-based operating systems enhance their security and integrity. It provides features such as monitoring and protects the crucial file's configurations, integrity, and processes. This is more often used in combination with other security features in the operating system to take the security to the highest level and create a secure environment. In this subsystem, it is essential to use cryptographic techniques and key management as they are vital factors in a system's security and integrity. This enhances the trustworthiness of keys and data.

Linux integrity is comprised of two key components:


- **Integrity measurement architecture (IMA)**: This ensures keeping a check and record of all the activities on the operating system and maintaining a record of content. It also measures the integrity of files using cryptographic hash functions and provides logs for monitoring and verification.
- **Extended verification module (EVM)**: This is the extended version of IMA. It provides digital signatures and cryptographic verification to keep a check on integrity measurements. This protects the file attributes that are related to the system. It uses public keys to verify and sign the hashes to maintain and improve the trustworthiness of keys.


The other aspects of Linux subsystem security are as follows:

- **IMA policies**: This determines which part of the system or which file in the system should be monitored and prioritized. This helps the system to get control over the policies and integrity measurement.

- **Secure boot:** This ensures that the system boots with appropriate and trusted signed components and boot loaders.
- **Audit and forensic:** This monitors the system's activities and generates logs. These logs are invaluable for tracing security breaches and facilitating subsequent investigations.
- **Remote attestation:** This feature allows the system to demonstrate its integrity to remote entities. Such attestations can assist these entities in making critical decisions based on the verified integrity of the system.

## Kernel and Module Integrity Monitoring





Kernel and module integrity monitoring ensure the **integrity** and **security** of a computer's operating system kernel and loadable kernel modules

### Kernel Integrity Monitoring

- Kernel Integrity monitoring involves continuously monitoring the **kernel** of an operating system to detect any unauthorized or unexpected changes or modifications
- It enables identifying attempts to **tamper** the kernel, which is critical for system stability and security
- Kernel Integrity Monitoring Techniques:**
  - Checksums and Hashing
  - File Integrity Monitoring (FIM)
  - Secure Boot

### Module Integrity Monitoring

- Module integrity monitoring ensures that only **authorized** and **signed** kernel modules are loaded into the kernel
- In Linux environments, module integrity monitoring is part of security solutions such as SELinux (Security-Enhanced Linux) or **AppArmor**
- Module Integrity Monitoring Techniques**
  - Module Signing
  - Kernel Module Whitelisting
  - Security Policies

Copyright © by **EC-Council**. All Rights Reserved. Reproduction is Strictly Prohibited.

## Kernel and Module Integrity Monitoring

**Kernel Integrity Monitoring:** This feature continuously oversees the system's kernels and loadable kernel modules to ensure integrity, security, and compliance. It enables the kernel to detect any attempts to tamper with the system's security settings. This is particularly crucial in areas of the system where file modifications and changes to system components are a concern. The primary objective of kernel integrity monitoring is to maintain system stability and prevent unauthorized access and modifications, thereby avoiding disruptions in ongoing processes.

### Kernel Integrity Monitoring Techniques

- **Checksums and hashing:** This calculates the hashes and checksums of critical resources in the system and compares them with good known values. If the value does not match, then the file is considered modified.
- **File Integrity Monitoring:** These generate alerts by using File integrity tools to monitor unauthorized modifications if happen.
- **Secure Boot:** This ensures that the system boots with appropriate and trusted signed components and boot loaders.

### Module Integrity Monitoring and Techniques







This is a security practice that involves constant monitoring. This is implemented in the kernel and kernel's boot loader by restricting access to allow only the authorized modules. The pieces of code are dynamically added and removed from running the kernel and ensure no malicious content is passed. The following are the techniques:

- **Module signing:** Process of signing the kernel modules and allowing only trusted ones in the bootloader. Unsigned ones are rejected to maintain integrity.

- **Security Policies:** Use policies and access controls to restrict access and improve compliance within the environment.
- **Use cases:** Module integrity monitoring is commonly used in Linux systems. This is often part of security solutions such as SELinux (Security Enhanced Linux).
- **Kernel module Whitelisting:** Maintains and allows the list of signed and authorized modules into the loader.

## Linux File Integrity Checking: File Integrity Monitoring (FIM)



-  To enhance the security of the system and audit, privilege management for Linux includes file integrity monitoring (FIM)
-  FIM policies is set up in a centralized repository and is used to schedule **periodic integrity checks** of software applications, customer data, and operating systems
-  It produces detailed reports for **security alerts**, and **vulnerability assessments** by verifying file permission, cryptographic checksums
-  Organizations are assigned to a policy and the policy is automatically retrieved to compare the local file system against a **system baseline**
-  The **policy violations** to the file system are documented in a report and sent to the central repository
-  The system administrators are transmitted to the central informed of the security breach and the policy violations to the file system are logged in a report and repository

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

### Linux File Integrity Checking: File Integrity Monitoring (FIM)

To enhance the security of the system and audit, privilege management for Linux includes file integrity monitoring (FIM). FIM policies are set up in a centralized repository and are used to schedule periodic integrity checks of software applications, customer data, and operating systems. It produces detailed reports for security alerts, and vulnerability assessments by verifying file permission, and cryptographic checksums. Organizations are assigned to a policy and the policy is automatically retrieved to compare the local file system against a system baseline. The policy violations to the file system are documented in a report and sent to the central repository and the system administrators are alerted of this security transgression. The policy configuration of FIM is stored as JSON script. Each policy is named and the details such as which file systems and files are to be verified, and which aspects of a file to check are specified. It also assigns a risk level to a violation.

## File Integrity Monitoring In Linux with Tripwire



Initialize the Tripwire database, which stores information about the files to be monitored **`sudo tripwire --init`**

Create a policy file to define the files and directories to monitor and what changes are allowed. The policy file is customized as per the requirements **`sudo vi /etc/tripwire/twpol.txt`**

twadmin command generates a configuration file based on created policy file: **`sudo twadmin --create-cfgfile -S site.key /etc/tripwire/twcfg.txt`**

Generate an initial Tripwire database as per the policy and configuration files

Automate the process of checking file integrity and updating the database by creating a cron job **`sudo crontab -e`**

To execute Tripwire daily, use the following command:  
**`0 0 * * * /usr/sbin/tripwire --check`**

| Date                                              | Element                                                                    | Change Type | Attributes                                              | Users          |
|---------------------------------------------------|----------------------------------------------------------------------------|-------------|---------------------------------------------------------|----------------|
| <b>Node: Coruscant.galaxy.fra (Linux Server)</b>  |                                                                            |             |                                                         |                |
| 2/17/18 12:54 PM                                  | /etc/ntp.conf                                                              | Modified    | SHA-1, Size                                             | clanabi        |
| 2/17/18 12:54 PM                                  | /etc/ntp.conf                                                              | Modified    | SHA-1, Size                                             | equiptime      |
| 2/17/18 12:54 PM                                  | /etc/ntp.conf                                                              | Modified    | SHA-1, Size                                             | equiptime      |
| 2/18/18 12:54 PM                                  | /etc/ntp.conf                                                              | Modified    | Change, Growing, MD5, Modify, Package Data, SHA-1, Size | clanabi        |
| 2/18/18 12:54 PM                                  | /etc/ntp.conf                                                              | Modified    | SHA-1, Size                                             | equiptime      |
| 2/18/18 12:54 PM                                  | /etc/ntp.conf                                                              | Modified    | SHA-1, Size                                             | equiptime      |
| <b>Node: Dagobah.galaxy.fra (Windows Server)</b>  |                                                                            |             |                                                         |                |
| 2/16/18 3:03 AM                                   | Local Firewall Rules                                                       | Modified    | MD5                                                     |                |
| 2/16/18 3:03 AM                                   | System Services                                                            | Modified    | MD5                                                     |                |
| 2/16/18 3:03 AM                                   | Listening Ports                                                            | Modified    | MD5                                                     |                |
| 2/16/18 3:03 AM                                   | HKLM_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\EventLog\Start        | Modified    | SHA-1                                                   | DESKTOP\fraser |
| 2/16/18 3:03 AM                                   | firewall_rule                                                              | Modified    | MD5                                                     |                |
| 2/16/18 2:54 AM                                   | local-users                                                                | Modified    | MD5                                                     |                |
| 2/16/18 2:54 AM                                   | enumerate_group                                                            | Modified    | MD5                                                     |                |
| 2/16/18 2:46 AM                                   | Local Firewall Rules                                                       | Modified    | MD5                                                     |                |
| 2/16/18 2:46 AM                                   | Listening Ports                                                            | Modified    | MD5                                                     |                |
| <b>Node: Empire2016 (Active Directory Server)</b> |                                                                            |             |                                                         |                |
| 2/15/18 10:58 PM                                  | CN=Domain Admins,CN=Users,DC=Kessel                                        | Modified    | member                                                  | KESSEL\fraser  |
| 2/15/18 11:04 AM                                  | CN={2B226864-6444-4C7D-BE0D-4283D30A4444},CN= Policies,CN=System,DC=Kessel | Modified    | groupPolicyMD5                                          | KESSEL\fraser  |
| <b>Node: Empire2016 (Microsoft SQL Server)</b>    |                                                                            |             |                                                         |                |
| 2/15/18 11:00 PM                                  | Query-payroll table query                                                  | Modified    | SHA-1                                                   |                |
| <b>Node: Moth.galaxy.fra (Solaris Server)</b>     |                                                                            |             |                                                         |                |
| 2/14/18 2:24 PM                                   | /etc/passwd                                                                | Modified    | Modify, SHA-1, Size                                     | isa            |
| 2/14/18 2:24 PM                                   | /etc/passwd                                                                | Modified    | SHA-1, Size                                             | isa            |
| 2/14/18 2:24 PM                                   | /etc/passwd                                                                | Modified    | SHA-1, Size                                             | isa            |

Tripwire Page

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

## File Integrity Monitoring in Linux with Tripwire

Source: <https://www.tripwire.com>

The Tripwire Enterprise's File Integrity Monitoring (FIM) solution pairs with security configuration management (SCM) to provide real-time change intelligence and threat detection. It identifies potential security breaches through regular scans, baseline creation, reporting, and alerting. It uses automation to detect system changes and to remediate those that take a configuration out of policy. This solution is for those who want the power of Tripwire FIM without the other features included in Tripwire Enterprise.

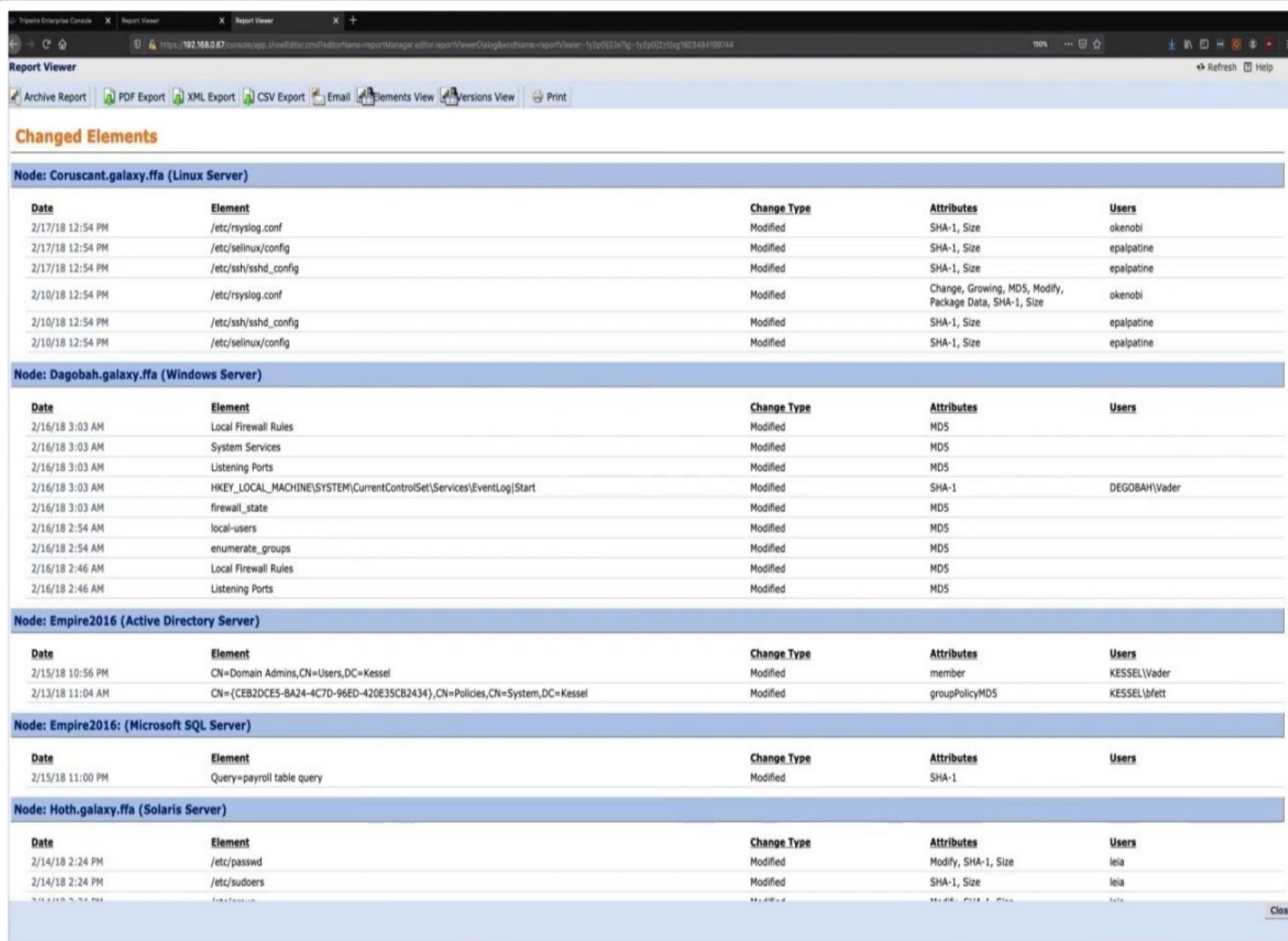
### Feature of Using Tripwire FIM:

- It reduces the signals-to-noise ratio.
- It allows capture of all the changes and details of who made the change and when.
- It provides customizing severities and scoring to reflect risk profile and business context.

### Steps to Use Tripwire FIM

- Initialize the Tripwire database, which stores information about the files to be monitored.  
`sudo tripwire --init`
- Create a policy file to define the files and directories to monitor and what changes are allowed. The policy file is customized as per the requirements.  
`sudo vi /etc/tripwire/twpol.txt`

- twadmin command generates a configuration file based on a created policy file.  
`sudo twadmin --create-cfgfile -S site.key /etc/tripwire/twcfg.txt`
- Generate an initial Tripwire database as per the policy and configuration files
- Automate the process of checking file integrity and updating the database by creating a cron job  
`sudo crontab -e`
- To execute Tripwire daily, use the following command:  
`0 0 * * * /usr/sbin/tripwire -check`



The screenshot shows the TripWire Report Viewer interface. It displays a table of 'Changed Elements' for several nodes. The table has columns for Date, Element, Change Type, Attributes, and Users. The nodes listed are Coruscant.galaxy.ffa (Linux Server), Dagobah.galaxy.ffa (Windows Server), Empire2016 (Active Directory Server), Empire2016: (Microsoft SQL Server), and Hoth.galaxy.ffa (Solaris Server).

| Date                                              | Element                                                                   | Change Type | Attributes                                              | Users         |
|---------------------------------------------------|---------------------------------------------------------------------------|-------------|---------------------------------------------------------|---------------|
| <b>Node: Coruscant.galaxy.ffa (Linux Server)</b>  |                                                                           |             |                                                         |               |
| 2/17/18 12:54 PM                                  | /etc/rsyslog.conf                                                         | Modified    | SHA-1, Size                                             | okenobi       |
| 2/17/18 12:54 PM                                  | /etc/selinux/config                                                       | Modified    | SHA-1, Size                                             | epalpatine    |
| 2/17/18 12:54 PM                                  | /etc/ssh/sshd_config                                                      | Modified    | SHA-1, Size                                             | epalpatine    |
| 2/10/18 12:54 PM                                  | /etc/rsyslog.conf                                                         | Modified    | Change, Growing, MD5, Modify, Package Data, SHA-1, Size | okenobi       |
| 2/10/18 12:54 PM                                  | /etc/ssh/sshd_config                                                      | Modified    | SHA-1, Size                                             | epalpatine    |
| 2/10/18 12:54 PM                                  | /etc/selinux/config                                                       | Modified    | SHA-1, Size                                             | epalpatine    |
| <b>Node: Dagobah.galaxy.ffa (Windows Server)</b>  |                                                                           |             |                                                         |               |
| 2/16/18 3:03 AM                                   | Local Firewall Rules                                                      | Modified    | MD5                                                     |               |
| 2/16/18 3:03 AM                                   | System Services                                                           | Modified    | MD5                                                     |               |
| 2/16/18 3:03 AM                                   | Listening Ports                                                           | Modified    | MD5                                                     |               |
| 2/16/18 3:03 AM                                   | HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\EventLog\Start       | Modified    | SHA-1                                                   | DEGOBAH\Vader |
| 2/16/18 3:03 AM                                   | firewall_state                                                            | Modified    | MD5                                                     |               |
| 2/16/18 2:54 AM                                   | local-users                                                               | Modified    | MD5                                                     |               |
| 2/16/18 2:54 AM                                   | enumerate_groups                                                          | Modified    | MD5                                                     |               |
| 2/16/18 2:46 AM                                   | Local Firewall Rules                                                      | Modified    | MD5                                                     |               |
| 2/16/18 2:46 AM                                   | Listening Ports                                                           | Modified    | MD5                                                     |               |
| <b>Node: Empire2016 (Active Directory Server)</b> |                                                                           |             |                                                         |               |
| 2/15/18 10:56 PM                                  | CN=Domain Admins,CN=Users,DC=Kessel                                       | Modified    | member                                                  | KESSEL\Vader  |
| 2/13/18 11:04 AM                                  | CN=(CEB2DCES-BA24-4C7D-96ED-420E35CB2434),CN=Policies,CN=System,DC=Kessel | Modified    | groupPolicyMD5                                          | KESSEL\Mett   |
| <b>Node: Empire2016: (Microsoft SQL Server)</b>   |                                                                           |             |                                                         |               |
| 2/15/18 11:00 PM                                  | Query=payroll table query                                                 | Modified    | SHA-1                                                   |               |
| <b>Node: Hoth.galaxy.ffa (Solaris Server)</b>     |                                                                           |             |                                                         |               |
| 2/14/18 2:24 PM                                   | /etc/passwd                                                               | Modified    | Modify, SHA-1, Size                                     | leia          |
| 2/14/18 2:24 PM                                   | /etc/sudoers                                                              | Modified    | SHA-1, Size                                             | leia          |
| 2/14/18 2:24 PM                                   | /etc/passwd                                                               | Modified    | SHA-1, Size                                             | leia          |

Figure 6.22: TripWire page

## Linux File Integrity Checking: AIDE



- Advanced Intrusion Detection Environment (AIDE) is a utility that ensures file integrity by establishing database of files in the system. This file is used to identify **intrusions** and verify **file integrity**
- It uses rules to verify whether the files or directories are modified and detects a threat if the file is modified

### Steps to use AIDE

- 1 Initialize the AIDE database using `sudo aide -init` command
- 2 Edit the AIDE configuration file (`/etc/aide/aide.conf`) to define the files and directories to be monitored and what changes are allowed
- 3 Update the AIDE database using `sudo aide -update`
- 4 Create a cron job to regularly check file integrity using `sudo crontab -e`
- 5 Add the following line to run AIDE daily - `0 0 * * * /usr/sbin/aide --check`

```
aide --check
Start timestamp: 2018-07-11 12:41:20 +0200 (AIDE 0.16)
AIDE found differences between database and filesystem!!
...
[trimmed for clarity]
```

### Manual Check using AIDE

Copyright © by **EC-Council**. All Rights Reserved. Reproduction is Strictly Prohibited.

## Linux File Integrity Checking Tool: Advanced Intrusion Detection Environment (AIDE)

**Source:** [www.redhat.com](http://www.redhat.com)

The Advanced Intrusion Detection Environment (AIDE) is a potent open-source intrusion detection tool that employs predefined rules to validate the integrity of Linux operating system files and directories. It protects the system from malware and viruses and detects unauthorized activity. To assure file integrity and find system intrusions, it creates a database of the file system and compares it to the system. It keeps track of recently modified or changed files. When someone tries to edit or change a file or directory, the user can keep track of that activity.

### Steps to Check File Integrity with AIDE:

- Initialize the AIDE database using  
`sudo aide -init`
- Edit the AIDE configuration file (`/etc/aide/aide.conf`) to define the files and directories to be monitored and what changes are allowed.
- Update the AIDE database using  
`sudo aide -update`
- Create a cron job to regularly check file integrity using  
`sudo crontab -e`
- Add the following line to run AIDE daily  
`0 0 * * * /usr/sbin/aide -check`

- Check manually.

```
aide --check
Start timestamp: 2018-07-11 12:41:20 +0200 (AIDE 0.16)
AIDE found differences between database and filesystem!!
...
[trimmed for clarity]
```

Figure 6.23: File Integrity Checking Manually

## Linux File Integrity Checking: Samhain



Samhain is an open-source host-based intrusion detection system (HIDS) with file integrity checking features

### Steps to use Samhain

- Customize the configuration file `/etc/samhain/samhainrc`, include the files and directories to be monitored and define other configuration settings
- Use the below configuration settings:
  - FILE\_CHECKS**: Specify the files and directories to monitor
  - HIDE\_MODIFIED**: Hide modified files from listings
  - IGNORE\_LIST**: Define files or directories to ignore during checks
  - REPORT\_LEVEL**: Reporting level (1 for minimal, 3 for detailed)
  - SYSLOG\_FACILITY**: Log location (e.g., LOG\_LOCAL4)
- View the reports in the Samhain log file, located at `/var/log/samhain/samhain.log`
- Check the most recent report using `- sudo samhain -t check`

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

## Linux File Integrity Checking Tool: Samhain

Source: <https://www.la-samhna.de>

Samhain's performs file integrity checks, log file monitoring and analysis, rootkit detection, port monitoring, rogue SUID executable detection, and hidden process detection. It is an open-source multi-platform for Linux systems. It offers PGP-signed database and configuration files, a stealth mode, and several more features to protect its integrity.

### Steps to Check Linux File Integrity Using Samhain

The steps to perform file integrity in Samhain tool are as follows:

- To install Samhain, use the distribution package manager. To install on Ubuntu/Linux, initially, update the OS with the command **sudo apt-get update -y**. For simple installation, use **sudo apt-get install samhain** command.
- To configure Samhain, customize the configuration according to the requirements. The samhain's configuration file is located in `/etc/Samhain/samhainrc`. Using the configuration file, the user can choose the files and directories to monitor to maintain integrity. Using the settings mentioned below, the user can customize monitoring based on what is needed.

Configuration Settings  
**FILE\_CHECKS:** Specify the files and directories to monitor  
**HIDE\_MODIFIED:** Hide modified files from listings  
**IGNORE\_LIST:** Define files or directories to ignore during checks  
**REPORT\_LEVEL:** Reporting level (1 for minimal, 3 for detailed)  
**SYSLOG\_FACILITY:** Log location (e.g., LOG\_LOCAL4).

Figure 6.24: Configuration Settings for Samhain

- Now, initialize Samhain before executing the file integrity services. To initialize Samhain, use the command `samhain -t init, --set-checksum-test=init`.
- Find the reports in the path **/var/log/samhain.log**. This periodically checks file integrity and records the log results. To verify an audit trail, use `samhain [-j | --just-list] -L logfile| --verify-log=logfile`. View the reports often to find any unauthorized changes to keep a check on integrity violations.

```
samhain - check file integrity
```

Figure 6.25: Samhain File Integrity Checking

```
samhain [-j | --just-list] -L logfile| --verify-log=logfile
```

Figure 6.26: Verifying the Integrity of a Signed Logfile

## Linux File Integrity Checking using OSSEC



- 01 OSSEC a host-based intrusion detection system (HIDS) can be used for monitoring the integrity of files and directories on a Linux system
- 02 OSSEC uses rules to define which files and directories to monitor and what actions to take when integrity violations occur
- 03 OSSEC rules can be found in `/var/ossec/etc/rules/local_rules.xml` and `/var/ossec/etc/rules/decoder.xml`
- 04 The OSSEC rules `local_rules.xml` can be customized to specify which files and directories to monitor and define the appropriate actions
- 05 View alerts and reports in real-time by tailing the OSSEC alert log `sudo tail -f /var/ossec/logs/alerts/alerts.log`

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

### Linux File Integrity Checking Using OSSEC

OSSEC is a host-based intrusion detection system (HIDS) and an effective tool for observing the consistency of files and directories on a Linux system. Linux server's security can be improved by configuring and customizing it to meet user's requirements and quickly spotting unauthorized changes. It helps enhance the Linux file integrity checking and provides flexibility in scheduling tasks.

#### OSSEC Features

- **Log-based intrusion detection system (LIDS):** Actively monitors data from numerous data points.
- **File integrity monitoring:** Keeps a forensic copy of data and spots changes to the system.
- **Active response:** Includes firewall integration with third-party portals to respond to attacks and changes in the system. It also provides a self-healing system.
- **Compliance auditing:** Supports auditing applications and systems at the level of many common standards, including PCI-DSS and CIS benchmarks.
- **Rootkit and malware detection:** It confines files to detect malicious applications and rootkits.
- **System inventory:** Keeps a check on data such as installed software, network listeners, and hardware.

### Step to Use Linux File Integrity Checking Using OSSEC

- Find the OSSEC rules in **/var/ossec/etc/rules/local\_rules.xml** path.
- In OSSEC, go to the path **/var/ossec/etc/rules/ocal\_riles.xml** that holds the custom decoders that are added to the **local\_decoder.xml** file.
- Find the audits and logs in the file alert.log. To access it, use the `sudo tail -f /var/ossec/logs/alerts/alerts.log` command.

## Linux File Integrity Checking Using Integrity Measurement Architecture (IMA)



- Linux kernel feature IMA helps verify the **integrity of files and executables** using digital signatures

- It ensures the integrity of critical system files and detects unauthorized changes

- To enable IMA, use the below configuration:

- `CONFIG_INTEGRITY=y`

- `CONFIG_IMA=y`

- IMA policies are defined in the **/etc/ima/ima-policy** file

- Example IMA policy that measures all files and executables **measure func=H**

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

### Linux File Integrity Checking Using Integrity Measurement Architecture (IMA)

Integrity Measurement Architecture (IMA) is responsible for calculating hashes of data before they are loaded. It has two subsystems in it: **measure** and **appraise**. The IMA measurement subsystem can interact with the TPM chip in the system to protect the collected hashes from alteration by a malicious administrator or application, supporting the files' integrity.

To enable IMA subsystems, enter the command `CONFIG_INTEGRITY=y` and `CONFIG_IMA=y`.

This Linux kernel feature supports file integrity validation by using digital signatures. IMA policies are defined in the **/etc/ima/ima-policy** file. For example, the IMA policy that can measure all files and executables is **func=H**.

It can store the audited file in the path **/var/log/audit/audit.log**. To view the file, use the command `grep "ima:" /var/log/messages`.

## File Integrity Monitoring: Filesystem Monitoring with inotifywait Tool



1

inotifywait is a command line tool that watches for changes to **files** or **directories** by using Inotify Linux kernel subsystem

2

inotifywait outputs the **event** and the file or directory where the change occurred to the console when a change is discovered

```
inotifywait /path/to/file
```

Watch a Specific File for Events

```
inotifywait --monitor /path/to/file
```

Without Exiting, Continuously Watch a Specific File for Events

```
inotifywait --event modify /path/to/file
```

Watch a File for File Modification Events

```
$./a.out /tmp /home/user/temp
Press enter key to terminate.
Listening for events.
IN_OPEN: /home/user/temp/foo [file]
IN_CLOSE_WRITE: /home/user/temp/foo [file]
IN_OPEN: /tmp/ [directory]
IN_CLOSE_NOWRITE: /tmp/ [directory]
```

Listening for events stopped.

Listening for Events

Copyright © by **EC-Council**. All Rights Reserved. Reproduction is Strictly Prohibited.

### Linux File Integrity Checking Tool: inotifywait

The inotifywait command-line tool offers a monitoring system for filesystem events. It can be utilized to track individual files or directories. When monitoring a directory, it returns events for both the directory and any files within the directory.

#### Steps to Use inotifywait for Linux File Integrity Checking

- Enter the following command to monitor specific events that are recorded:

```
inotifywait /path/to/file
```

- Enter the following command to monitor events continuously without exiting:

```
inotifywait --monitor /path/to/file
```

- Enter the following command to monitor the file and detect modification activities:

```
inotifywait --event modify /path/to/file
```

## Monitor File Integrity In Linux Using auditd (Linux Auditing System)



- Linux kernel feature Auditd enables **monitoring** file changes by configuring audit rules to monitor specific files or directories for changes
- It generates detailed **logs** that can be reviewed and analyzed
- To enable and start auditd, use the commands **sudo systemctl enable auditd** and **sudo systemctl start auditd**
- Use tools such as **ausearch** or **aureport** to query and analyze audit logs for
- To view log events of the password file **sudo ausearch -k passwd\_changes**
- Rule to monitor changes to /etc/passwd file use **sudo auditctl -w /etc/passwd -p wa -k passwd\_changes**
  - **-w**: Specifies the file or directory path to monitor
  - **-p wa**: Defines the permissions to monitor
  - **w**- write
  - **a**- attribute changes (e.g., permissions, ownership)
  - **k**: Assigns a unique key (used to identify related events in the audit logs) to the rule (e.g., passwd\_changes)

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

### Monitoring File Integrity in Linux Using auditd (Linux Auditing System)

The Linux Auditing System (auditd) simplifies the task of monitoring system processes and access. It allows for file monitoring based on user-defined priorities and generates detailed logs for verification and review. These logs serve as invaluable resources for ensuring system integrity and compliance.

#### Steps to Review Logs Generated by auditd

- To enable auditd, enter the `sudo systemctl enable auditd` command; it will start at boot time.
- To view the currently configured audit rules, enter the `sudo auditctl -l` command.
- To analyze the audits and data in detail, use the `aureport` and `ausearch` commands; these can query the audits.

```
$ sudo ausearch -i -k user-modify
```

```
$ sudo aureport -x
```

- To monitor the changes to the **/etc/passwd** file, use the `sudo auditctl -w /etc/passwd -p wa -k passwd_changes` command, where “w” specifies the file or directory path to monitor, “-p wa” specifies the permissions to monitor, “w-” writes an attribute change (e.g., permissions, ownership), and “k” assigns a unique key (used to identify related events in the audit logs) to the rule (e.g., passwd\_changes)

```
auditctl -w path_to_file -p permissions -k key_name
```

## File Integrity Checking In Linux Using Rootkit



- Rootkit can be used for basic file integrity checking on a Linux system
- Perform a file integrity check using `rkhunter- sudo rkhunter --check`
- rkhunter comes with a configuration file located at `/etc/rkhunter.conf`. You can customize this file to adjust the scanning parameters and file paths to meet your specific needs
- Rkhunter scans system files and directories for changes and reports any discrepancies
- It generates a report by default stored in the `/var/log/rkhunter/` directory
- Review the report for any warnings or suspicious changes
- Customize the `/etc/rkhunter.conf` file to modify the scanning parameters. Specify additional files or directories to monitor
- Run cron job to schedule rkhunter by the command `sudo crontab -e`

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

### File Integrity Checking in Linux Using Rootkit

Rootkit is an open-source Linux-based tool that is used to detect malware, rootkits, or backdoor exploits in the system. This tool can scan for malware and unauthorized permissions set on binaries in Linux and hunts if any suspicious strings are attached to the kernel. It also searches for any modifications or discrepancies that are made to any file or directory in the Linux system and generates a report. Review the report for any warnings or suspicious changes by going through the stored report stored in the path `/var/log/rkhunter/directory`.

Use the configuration file `/etc/rkhunter.conf` to customize the parameters based on the user's requirements such as specifying additional files or directories to monitor.

#### Steps to Run Rootkit

- To start rootkit hunter, enter the following command.

```
sudo rkhunter --check
```

- If needed, schedule the rootkit to start. Enter the command `sudo crontab -e` to schedule a task.

## Linux Hardening Checklist: OS Hardening



- ✓ Restrict core dump
- ✓ Remove legacy services
- ✓ Disable any services and applications started by **xinetd** or **inetd** that are not being utilized. Remove xinetd
- ✓ Disable or remove **server services** (FTP, DNS, LDAP, SMB, DHCP, NFS, SNMP, etc) that are not used
- ✓ Ensure syslog (rsyslog, syslog, syslogng) service is running
- ✓ Enable a **network time protocol (NTP)** service to ensure clock accuracy
- ✓ Restrict the use of the **cron** and at services

Copyright © by **EC-Council**. All Rights Reserved. Reproduction is Strictly Prohibited.

## Linux Hardening Checklist: OS Hardening (Cont'd)



- ✓ Enable SELinux (Security-Enhanced Linux)
- ✓ Put honeypots and honeynets to use
- ✓ Disable SSH Password Authentication and SSH Root Access
- ✓ Change the SSH Default Port
- ✓ Create proper Nftables or Iptables rules
- ✓ Encrypt storage devices and partitions
- ✓ Defend the system against rootkits and secure the BIOS

Copyright © by **EC-Council**. All Rights Reserved. Reproduction is Strictly Prohibited.

### Linux Hardening Checklist: OS Hardening

- Restrict/disable core dumps, which help understand why a program aborted but contain confidential data from memory.
- If there is no exception, remove legacy services that provide or rely on unencrypted authentication. Those services include those discussed previously:
  - `telnet-server`

- `rsh`, `rlogin`, `rcp`
- `ypserv`, `ypbind`
- `tftp`, `tftp-server`
- `talk` and `talk-server`
- Disable the unused services and applications started by `xinetd` (remove if possible) or `inetd`. The `inetd` or `xinetd` service allows running a program when a connection is sent to a designated network port.
- Disable/remove server services if you do not use them. For example, FTP, DNS, LDAP, SMB, DHCP, NFS, SNMP, etc.
- Ensure running `syslog` (`rsyslog`, `syslog`, `syslog-ng`) service as it manages logs in `/var/log/` and supports remote log forwarding.
- Enable a network time protocol (NTP) service to ensure clock accuracy. Keeping time synchronized across devices is important for many authentication and encryption services and allows for analysis of logs across multiple systems.
- Restrict the use of the `cron` and `at` services as these can run commands on the system. These should only be allowed to accounts that need to run commands.
- Enable SELinux (Security-Enhanced Linux) allows blocking unauthorized access and enforces separation of privilege.
- Put honeypots that creates a virtual trap to lure attackers. In addition, put honeynets the act as a decoy network containing one or more honeypots.
- Disable SSH Password Authentication and SSH Root Access since connecting to the servers using these mechanisms is unsafe.
- Change the SSH Default Port since leaving the default port 22 for SSH makes the server vulnerable to brute-force attacks.
- Create proper Nftables or Iptables rules that operate by comparing network traffic against a set of rules.
- Encrypt storage devices and partitions to ensure data integrity and security.
- Defend the system against rootkits and secure the BIOS since rootkits target a vulnerability in a machine's operating system (OS) or application on the machine.



---

### LO#04: Discuss Linux user access and password management


---

Copyright © by **EC-Council**. All Rights Reserved. Reproduction is Strictly Prohibited.

## **LO#04: Discuss Linux User Access and Password Management**

The objective of this section is to explain the user access and password management features in Linux.

## Enforce Strong Password Management

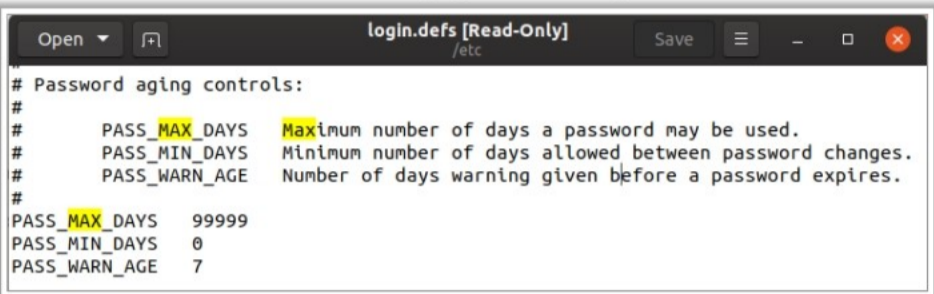


Enforce strong password policy using **PAM** (pluggable authentication module)

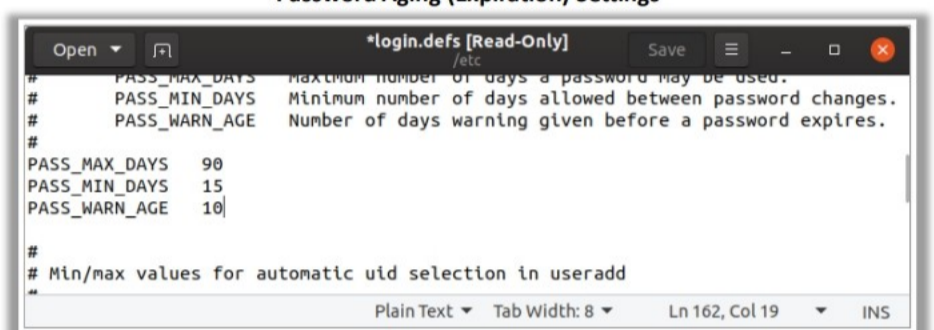
### Password Policy Settings

- Password max days
- Password min days
- Password warning days
- Password minimum length
- Minimum uppercase characters
- Minimum lowercase characters
- Minimum digits in password
- Minimum other characters (symbols)
- Account lock – retries
- Account unlock time

#### Password Aging Default Settings



#### Password Aging (Expiration) Settings



Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

## Enforce Strong Password Management

By default, the system implements simple password rules. Linux distributions allows the administrators to set password policies. Implementing a strong password policy is one of the key user management tasks of Linux system administration, and it restricts unauthorized access to systems. If a user account has a simple password, attackers may be able to brute-force or guess the password easily and access the system. The network defender should ensure that the password rules are set as per the organization's policies.

To view and change the default password policy settings, use the following command.

```
sudo gedit /etc/logins.defs
```

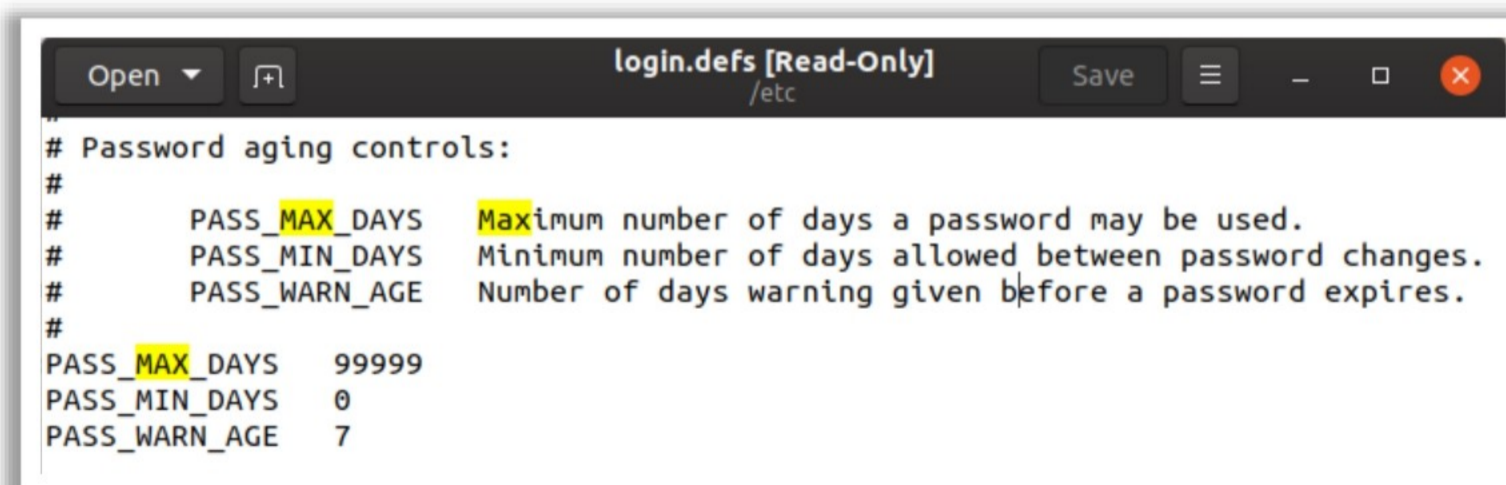


Figure 6.27: Viewing and changing password policy settings

## Enforcing Strong Password Policy on Linux Systems Using PAM

The file PAM (pluggable authentication module) can be found in the following locations.

- Red Hat-based systems @ /etc/pam.d/system-auth.
- Debian-based systems @ /etc/pam.d/common-password.

## Parameters and Commands Used in Implementing Password Policies in Linux

- “Password max days” (applicable for new accounts only) restricts the maximum number of days a password can be used. For example:

```
vi /etc/login.defs
PASS_MAX_DAYS 90
```

- “Password Min days” (applicable for new accounts only) limits the minimum number of days after the password can be changed. For example, if this parameter is set to 20 and the password is set today, the user cannot change or set the password before 20 days from today. For example:

```
vi /etc/login.defs
PASS_MIN_DAYS 20
```

- “Password warning days” (applicable for new accounts only) controls the password warning days and warns the user when the password is going to expire. Till the warning days end, a warning message will be displayed to the user regularly, allowing the user to change the password before expiry. For example:

```
vi /etc/login.defs
PASS_WARN_AGE 10
```

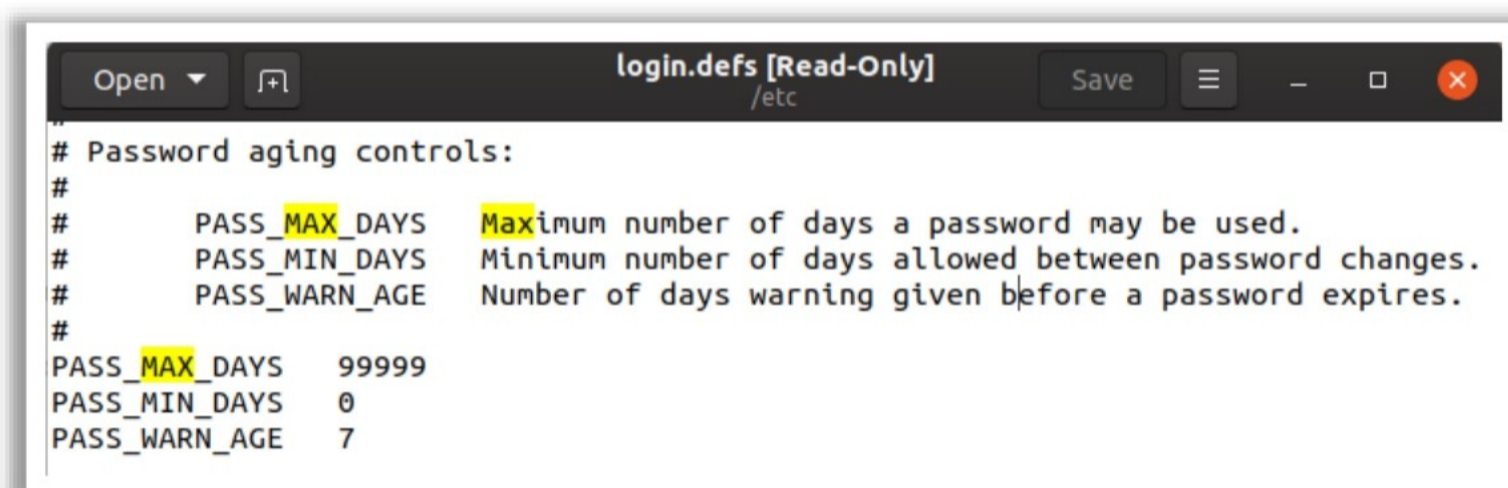


Figure 6.28: Password warning days settings

## Password aging default settings

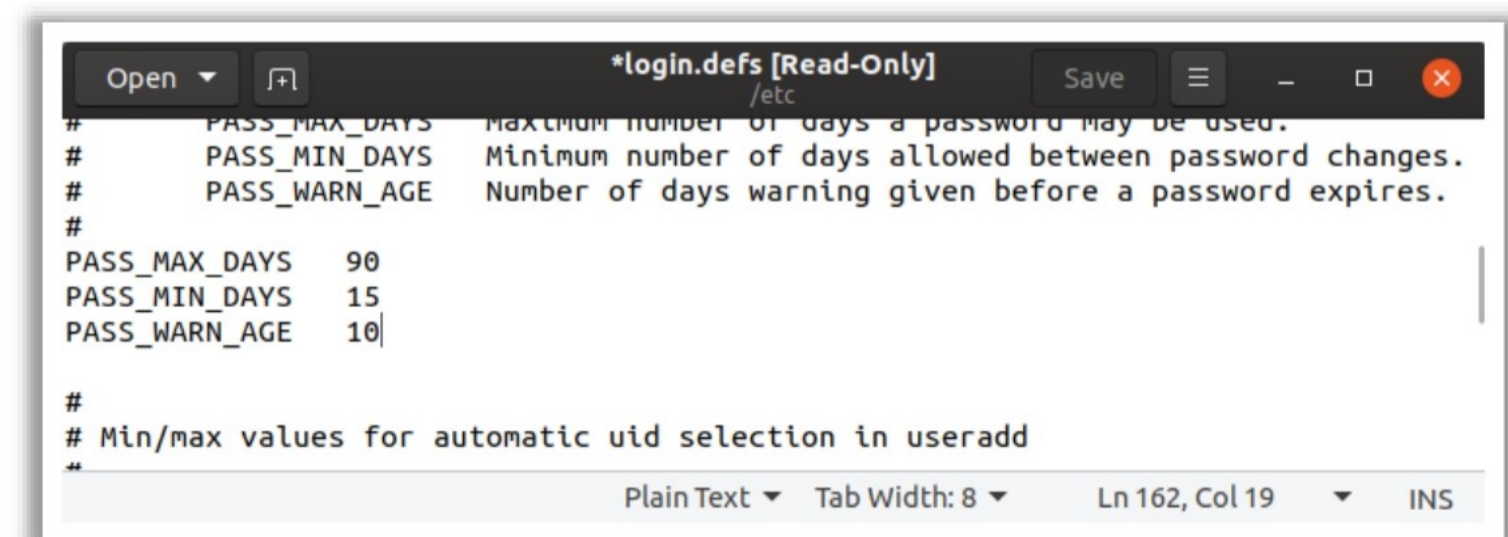


Figure 6.29: Password aging default settings

## Password aging (expiration) settings

- “Password history or deny re-used passwords” records the history of used passwords. It checks the password history and warns the user if they have use an old password while setting a new one. For example:

```
vi /etc/pam.d/system-auth
password sufficient
pam_unix.so sha512 shadow nullok try_first_pass use_authtok
remember=5
```

- “Password minimum length” controls the minimum password length condition. It checks against this parameter and warns the user if they try to set the password length less than the set password minimum length when the user sets a new password. For example:

```
vi /etc/pam.d/system-auth
password requisite pam_cracklib.so try_first_pass retry=4
minlen=10
```

- “Minimum uppercase characters” controls how many minimum uppercase characters should be added in the password. It checks against this parameter and warns the user if they do not include any uppercase characters in the password while creating a new password. For example:

```
vi /etc/pam.d/system-auth
password requisite pam_cracklib.so try_first_pass retry=3 minlen=12
ucredit=2
```

- “Minimum lowercase characters” controls how many lowercase characters should be added in the password. It checks against this parameter and warns the user if they do not include any lowercase characters in the password while creating a new password. For example:

```
vi /etc/pam.d/system-auth
password requisite pam_cracklib.so try_first_pass retry=3 minlen=12
lcredit=-1
```

- “Minimum digits in password” controls how many digits should be added in the password. It checks against this parameter and warns the user if they do not include any digits in the password while creating a new password. For example:

```
vi /etc/pam.d/system-auth
Password requisite pam_cracklib.so try_first_pass retry=3 minlen=12
dcredit=-1
```

- “Minimum other characters (symbols)” keeps how many special characters should be added in the password. It checks against this parameter and warns the user if they do not include any symbols in the password while creating a new password. For example:

```
vi /etc/pam.d/system-auth
Password requisite pam_cracklib.so try_first_pass retry=3 minlen=12
ocredit=-1
```

- “Account lock – retries” locks the user account after reaching the given number of failed login attempts. For example:

```
vi /etc/pam.d/system-auth
auth required pam_tally2.so onerr=fail audit silent deny=5
account required pam_tally2.so
```

- “Account unlock time” sets the time to unlock the user account if the user account is locked after consecutive failed authentications. For example:

```
vi /etc/pam.d/system-auth
auth required pam_tally2.so onerr=fail audit silent deny=5
unlock_time=900
account required pam_tally2.so
```

### Setting Secure Password Policy

The steps the users should follow to set a secure password policy on Debian/Ubuntu are listed below.

- Install `pwquality/pam_pwquality` PAM module package on Ubuntu / Debian system using the following command.
- After installation of the package, edit the `/etc/pam.d/common-password` (for Debian-based systems) file or edit `/etc/pam.d/system-auth` file (Red Hat-based systems) to set password requirements using the following command.

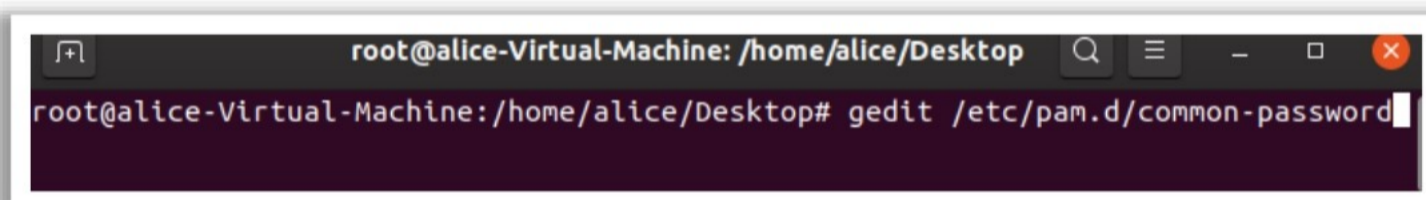


Figure 6.30: common-password command

- Change line 25 from `password requisite pam_pwquality.so retry=3` to as marked in the screenshot.

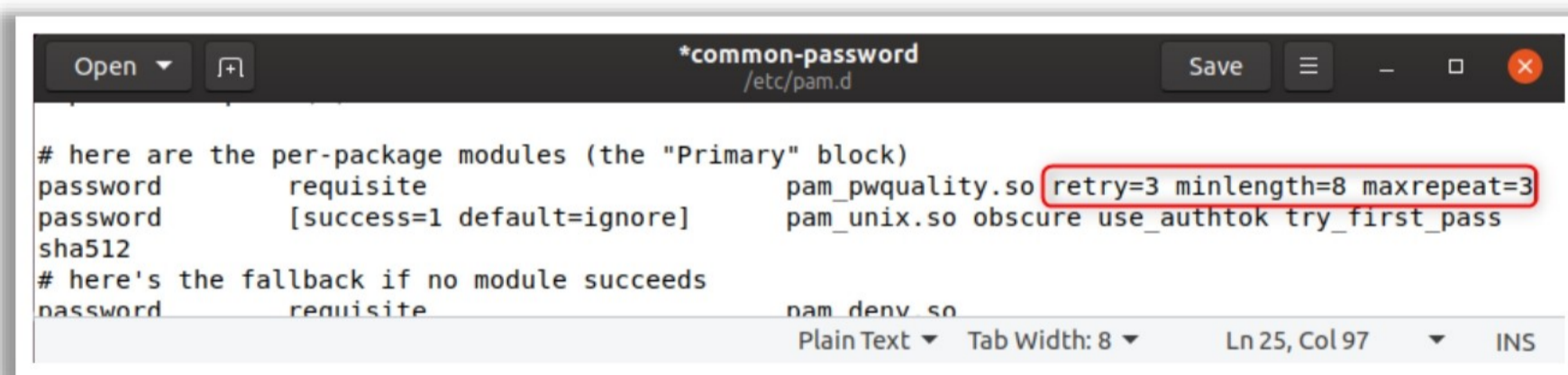


Figure 6.31: Password quality

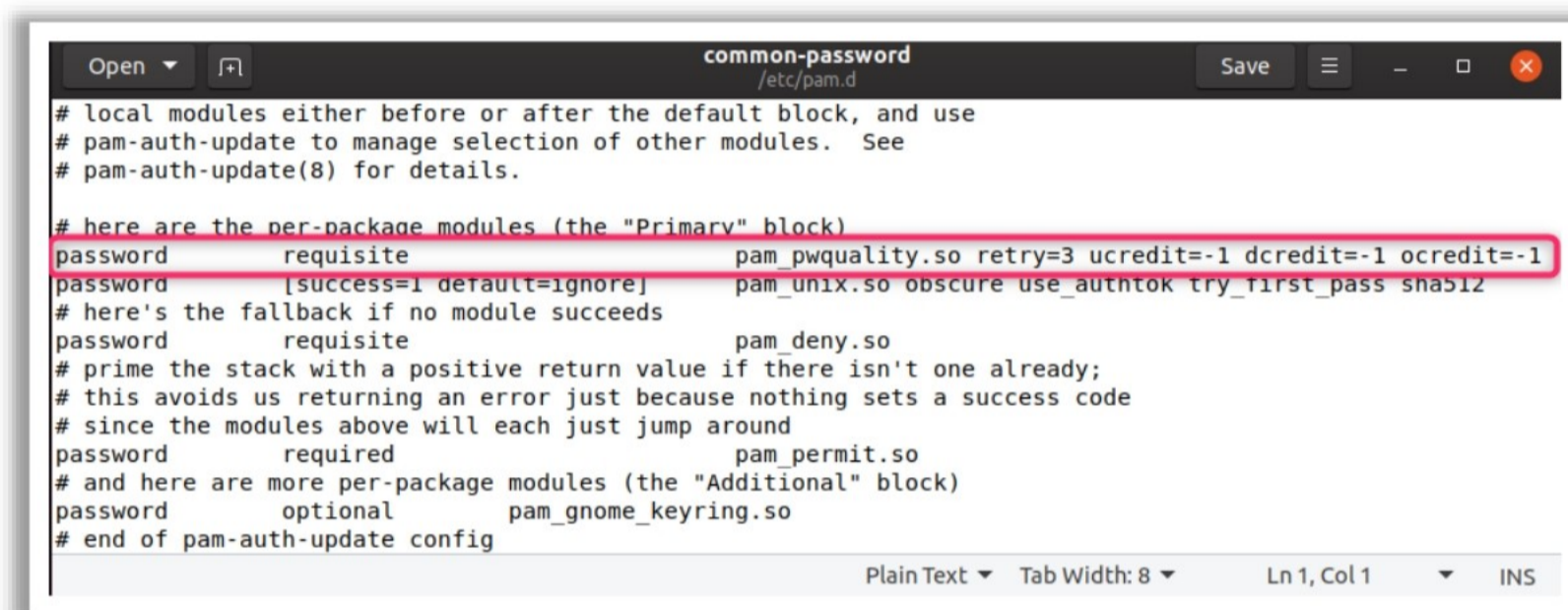
- Change the settings that suit the desired password policy. For example,
  - `retry=3`: Prompt user 3 times before displaying error
  - `minlen=8`: Password cannot be less than 8 character

- `maxrepeat=3`: Maximum of 3 characters are allowed to repeat
- Reboot your system using the command `sudo reboot`.
- Now the system will restrict creating a user account with a weak password.

## Setting Password Complexity

To set password complexity on Debian/Ubuntu edit `/etc/pam.d/common-password` file and set the following password complexity settings.

- `ucredit=-1`: Require at least one uppercase letter in the password.
- `lcredit=-1`: Require at least one lowercase letter in the password.
- `dcredit=-1`: Require at least one special character in the password.



```
local modules either before or after the default block, and use
pam-auth-update to manage selection of other modules. See
pam-auth-update(8) for details.

here are the per-package modules (the "Primary" block)
password requisite pam_pwquality.so retry=3 ucredit=-1 dcredit=-1 ocredit=-1
password [success=1 default=ignore] pam_unix.so obscure use_authtok try_first_pass sha512
here's the fallback if no module succeeds
password requisite pam_deny.so
prime the stack with a positive return value if there isn't one already;
this avoids us returning an error just because nothing sets a success code
since the modules above will each just jump around
password required pam_permit.so
and here are more per-package modules (the "Additional" block)
password optional pam_gnome_keyring.so
end of pam-auth-update config
```

Figure 6.32: Password complexity

## Restrict User from Using Previous Passwords



- For a secure password policy, it is recommended to restrict user from using previous passwords
- To prevent users from reusing old passwords, use the **remember** option for the PAM module

```
/etc/pam.d/common-password - password-related modules common to all services
#
This file is included from other service-specific PAM config files,
and should contain a list of modules that define the services to be
used to change user passwords. The default is pam_unix.
#
Explanation of pam_unix options:
#
The "sha512" option enables salted SHA512 passwords. Without this option,
the default is Unix crypt. Prior releases used the option "md5".
#
The "obscure" option replaces the old "OBSOLETE_CHECKS_ENAB" option in
login.defs.
#
See the pam_unix manpage for other options.
#
As of pam 1.0.1-6, this file is managed by pam-auth-update by default.
To take advantage of this, it is recommended that you configure any
local modules either before or after the default block, and use
pam-auth-update to manage selection of other modules. See
pam-auth-update(8) for details.
#
here are the per-package modules (the "Primary" block)
password requisite pam_pwquality.so retry=3 remember=10
password [success=1 default=ignore] pam_unix.so obscure use_authtok try_first_pass sha512
here's the fallback if no module succeeds
password requisite pam_deny.so
prime the stack with a positive return value if there isn't one already;
this avoids us returning an error just because nothing sets a success code
since the modules above will each just jump around
password required pam_permit.so
and here are more per-package modules (the "Additional" block)
password optional pam_anoma_kaurion.so
```

```
alice@alice-Virtual-Machine:~$ passwd
Changing password for alice.
Current password:
Changing password for alice.
New password:
BAD PASSWORD: The password is the same as the old one
New password:
Retype new password:
passwd: password updated successfully
alice@alice-Virtual-Machine:~$ passwd
Changing password for alice.
Current password:
Changing password for alice.
New password:
BAD PASSWORD: The password is just rotated old one
New password:
```

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

## Restrict User from Using Previous Passwords

It is recommended to restrict the user from using previous passwords. To prevent users from reusing old passwords, use the **remember** parameter of the **pam\_unix** module. This module maintains a list of old passwords and restricts the user from using previous passwords. The list of old passwords is stored in **/etc/security/opasswd** file. The account information is set using **/etc/passwd**.

### To Restrict User from Using Previous Passwords

#### Step 1: Limiting password reuse

- Debian/Ubuntu Linux users need to open **/etc/pam.d/common-password** file, then run:  

```
##cp/etc/pam.d/common-password/root/common-password.bak
vi/etc/pam.d/common-password
```
- To prevent the user from utilizing the last 13 passwords, edit/add password line  
**remember=13.**

```
password sufficient pam_unix.so use_authtok sha512 shadow
remember=13
```

- Save and close the file.

#### Step 2: Enable password aging

- Edit **/etc/login.defs**, enter:  

```
vi /etc/login.defs
```

The user can set the minimum number of days to change password since last change (PASS\_MIN\_DAYS = 7).

- Save and close the file.

Step 3: /etc/security/opasswd

The file that store the list of old password does not exist /etc/security/opasswd, the user can create the file utilizing touch or shell redirection command:  
# [ ! -f /etc/security/opasswd ] && touch/etc/security/opasswd

## Ensure No Accounts Have Empty Passwords



All accounts should have **passwords** to prevent misuse of the account by an unauthorized user

Type # `awk -F: '($2 == "") {print}' /etc/shadow` command to list the accounts with empty passwords

```
root@alice-Virtual-Machine:/home/alice# awk -F: '($2 == "") {print}' /etc/shadow
alice::18327:0:99999:7:::
```

```
root@alice-Virtual-Machine:/home/James# awk -F: '($3 == "0") {print}' /etc/passwd
root:x:0:0:root:/root:/bin/bash
root@alice-Virtual-Machine:/home/James#
```

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

### Ensure No Accounts Have Empty Passwords

An account with an empty password can be used to login without a password. Hence, ensure that no accounts have empty passwords:

- Type the below appended command  
# `awk -F: '($2 == "") {print}' /etc/shadow`

```
root@alice-Virtual-Machine:/home/alice# awk -F: '($2 == "") {print}' /etc/shadow
alice::18327:0:99999:7:::
```

Figure 6.33: Checking for empty password

- Locking the empty password accounts  
# `passwd -l accountName`

### Disable Login for Users with Null Password

With a null password, the adversary can log in and exploit the system. Therefore, it is essential to disable login for users with a null password.

Use the following command to set up null password: # `usermod -p "" username`

The `nullok` module of PAM configuration allows the null password; therefore, remove the `nullok` from any authentication module.

### Disabling Null Password in Debian Linux

- The two files that are used by Debian Linux are

- o /etc/pam/common-auth
- o /etc/pam.d/common-password

Use `cp` command for files backup

a) Open /etc/pam/common-auth:

```
#cp/etc/pam/common-auth /etc/pam/common-auth.ORI
vi /etc/pam/common-auth
```

Find and remove nullok

```
password required pam_unix.so nullok obscure min=4 max=8 md5
password required pam_unix.so obscure min=4 max=8 md5
```

Save and exit the file.

b) Open file /etc/pam.d/common-password:

```
cp/etc/pam.d/common-password
vi /etc/pam.d/common-password
```

Fine and remove nullok\_secure

- auth required pam\_unix.so nullok\_secure
- auth required pam\_unix.so

Save and exit the file.

Now, the adversary cannot utilize the null password to gain access to the system.

### Disabling Null Password in Red Hat/Fedora Linux

▪ Open/etc/pam.d/system-auth:


```
cp /etc/pam.d/system-auth /etc/pam.d/system-auth.ORI
vi /etc/pam.d/system-auth
```

▪ Fine and remove nullok

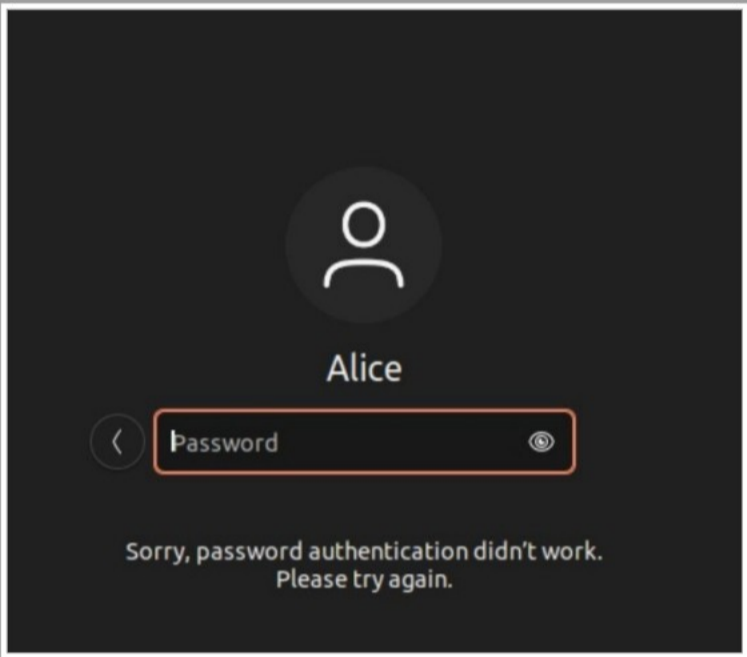
```
auth sufficient /lib/security/pam_unix.so likeauth nullok
auth sufficient /lib/security/pam_unix.so likeauth
```

▪ Save the file.

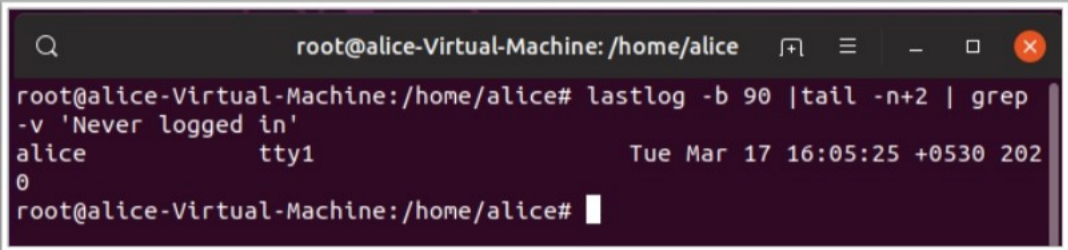
## Disable Unnecessary Accounts



- Disable **inactive user account** that have not been used over a long period of time
- Do remember to disable **user account of employees** who have resigned from the organization
- Attacker can gain access to system through compromised unused/inactive user accounts

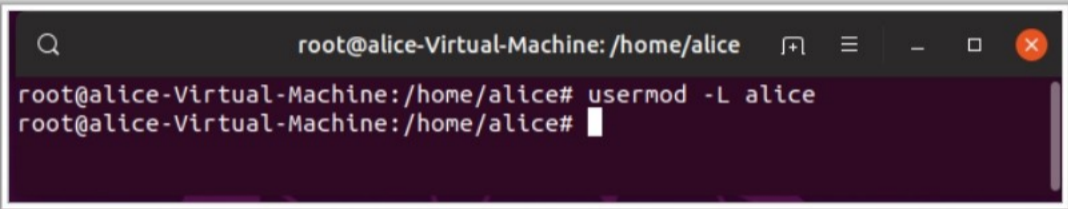


Command to lists users who have not logged in in the past 90 days, eliminating the header row and excluding users who never logged in



```
root@alice-Virtual-Machine:/home/alice# lastlog -b 90 | tail -n+2 | grep -v 'Never logged in'
alice tty1 Tue Mar 17 16:05:25 +0530 202
0
```

Command to Disable a User



```
root@alice-Virtual-Machine:/home/alice# usermod -L alice
root@alice-Virtual-Machine:/home/alice#
```

Copyright © by **EC-Council**. All Rights Reserved. Reproduction is Strictly Prohibited.

## Disable Unnecessary Accounts

To reduce the system's attack surface, it is recommended that inactive accounts (accounts not used over a long period of time) should be disabled. Attackers can gain access to a system through compromised unused/inactive user accounts. Moreover, an inside attacker may misuse the account of a previous employee to perform malicious activities; therefore disable user accounts of employees who have resigned from the organization. Identify the inactive accounts in the system and disable them.

Command to lists users who have not logged in in the past 90 days:

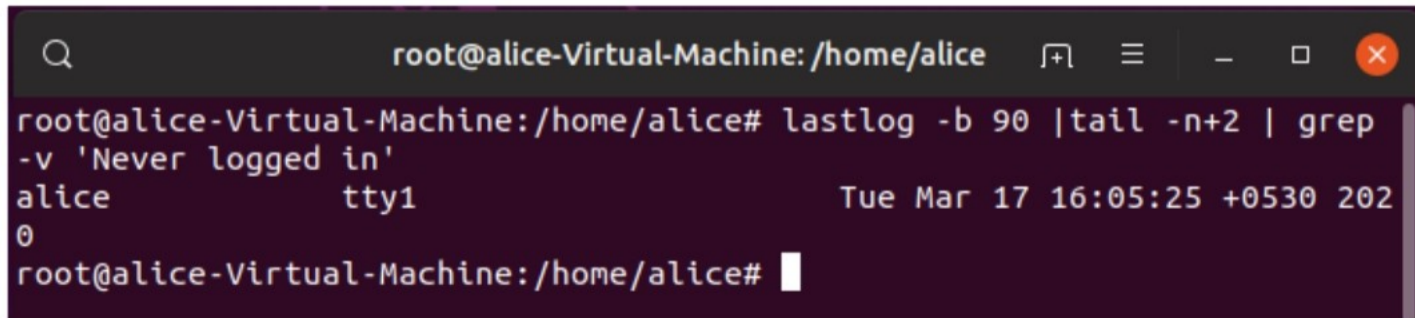
```
lastlog -b 90
```

Command to lists users who have not logged in in the past 90 days, eliminating the header row:

```
lastlog -b 90 | tail -n+2
```

Command to lists users who have not logged in in the past 90 days, eliminating the header row and excluding system users:

```
lastlog -b 90 | tail -n+2 | grep -v 'Never logged in'
```



```
root@alice-Virtual-Machine:/home/alice# lastlog -b 90 | tail -n+2 | grep -v 'Never logged in'
alice tty1 Tue Mar 17 16:05:25 +0530 202
0
root@alice-Virtual-Machine:/home/alice#
```

Figure 6.34: Checking for unnecessary accounts

Command to disable a user in Ubuntu:

Usermod -L <username>

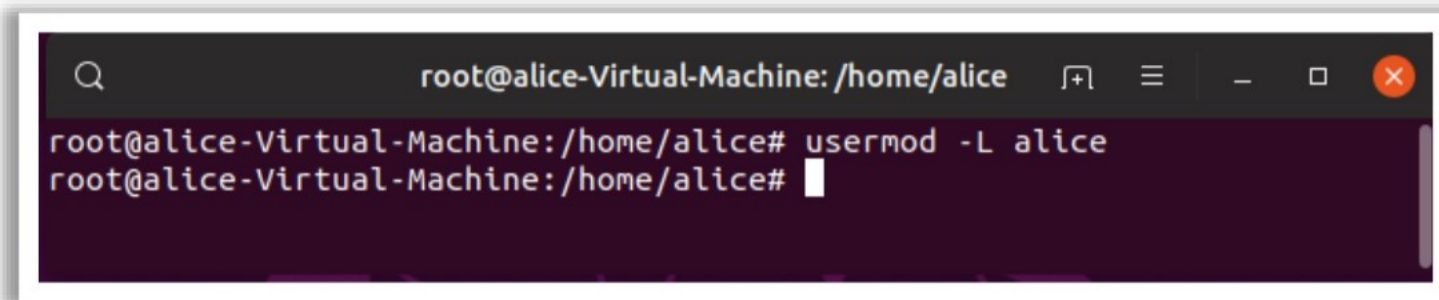


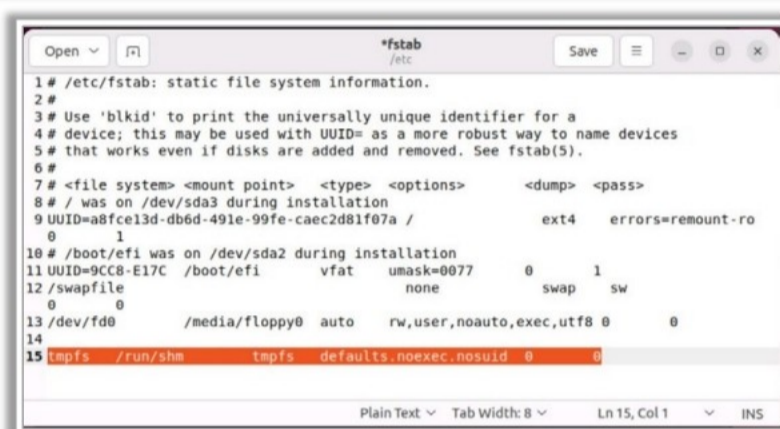
Figure 6.35: disabling user

## Secure Shared Memory



- By default, shared memory mounted with read/write permissions, making the **/run/shm** (implementation of shared memory) space exploitable
- Shared memory can be exploitable to attacks targeted towards services like apache2, httpd, etc.
- To make shared memory secure, mount **/run/shm** in read-only mode without the permission to execute programs
- Edit **/etc/fstab** file and include the following line of code to set **/run/shm** to read-only:

```
tmpfs /run/shm tmpfs defaults,noexec,nosuid 0 0
```



Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

## Secure Shared Memory

Shared Memory is a feature supported by Linux. One or more programs use the same memory space for handling the execution processes as per schedule. However, this feature can be exploited. By default, the shared memory is mounted with read and write permission in the Linux system. The mounted point of the shared space is **/run/shm**. This mount point can be read without permission. An attacker may use it to execute programs, change UID of the program, etc. To avoid such breaches, network defenders should protect shared memory.

### Steps to Secure Shared Memory

- Open the terminal and type the command `sudo nano /etc/fstab` and hit Enter.
- File will open in the terminal and add the below line at the bottom of the file.

```
tmpfs /run/shm tmpfs defaults,noexec,nosuid 0 0
```

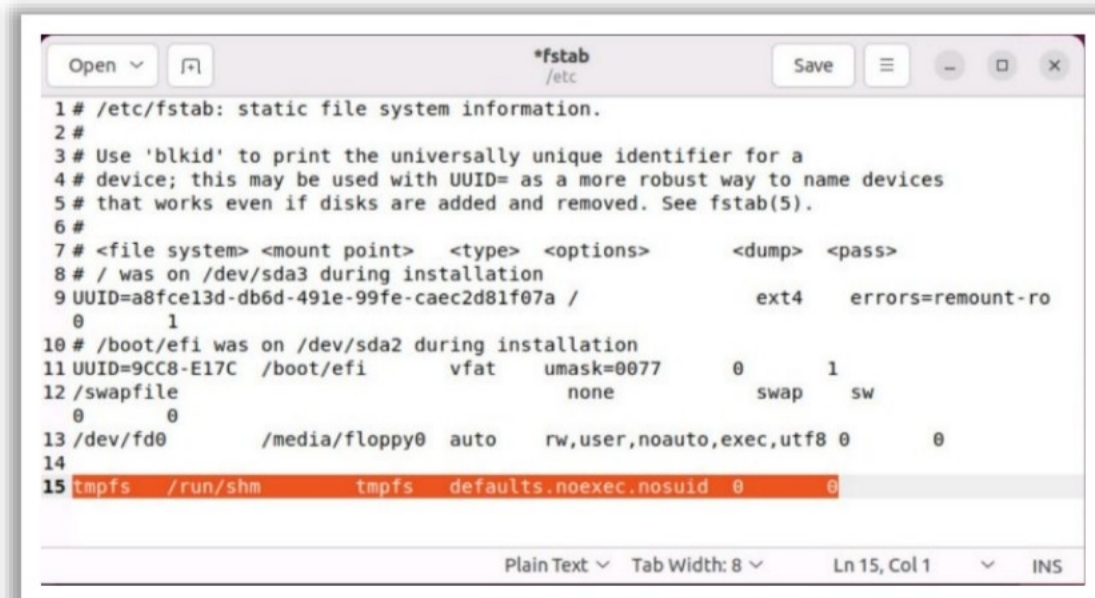


Figure 6.36: Securing shared memory

- Press Ctrl+o and hit Enter to save the file.
- Press Ctrl+x to exit the file.
- Reboot the system for the changes to take effect.

## Delete X Window Systems (X11)



- CentOS / RHEL 5.x / Fedora Linux comes with X Windows system
- X Window System is the **graphical interface** for Linux is not required for dedicated Linux based mail and Apache/Nginx web server. Vulnerabilities in X Window can help non-root users to escalate his/her privilege to higher level

- Disable** and **remove** X Windows to prevent from exploiting vulnerability in the X11 code

To disable X Windows at System Boot, edit /etc/inittab and set run level to 3

- Open inittab file in editor: `vi /etc/inittab`
- Find line: `id:5:initdefault:`
- Replace with `id:3:initdefault:`

To remove X Windows, type the following command to remove X Windows

```
yum groupremove "X Window System"
```

```
root@alice-Virtual-Machine: /
GNU nano 6.2 /etc/inittab *
#
#Terminal gettys are handled by /etc/init/tty.conf and /etc/init/serial.conf,
#with configuration in /etc/sysconfig/init.
#
#For information on how to write upstart event handlers, or how
#upstart works, see init(5), init(8), and initctl(8).
#
#Default runlevel. The runlevels used are:
0 - halt (Do NOT set initdefault to this)
1 - Single user mode
2 - Multiuser, without NFS (The same as 3, if you do not have networking)
3 - Full multiuser mode
4 - unused
5 - X11
6 - reboot (Do NOT set initidefault to this)
#
id:5:initdefault:
File Name to Write: /etc/inittab
^G Help ^M-D DOS Format ^M-A Append ^M-B Backup File
^C Cancel ^M-M Mac Format ^M-P Prepend ^T Browse
```

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

### Delete X Window Systems (X11)

CentOS/RHEL 5.x/Fedora Linux comes with X Windows System, the graphical interface for Linux, which may not be required for dedicated Linux-based mail and Apache/Nginx web servers. Vulnerabilities in X Window System can help non-root users escalate their privilege to a higher level. Disable and remove X Windows to prevent vulnerabilities arising from the X11 code.

To disable X Windows Systems at system boot, edit /etc/inittab and set run level to 3

- Open `init` file in editor `#vi /etc/inittab`
- Find line `id:5:initdefault:`
- Replace with `id:3:initdefault:`

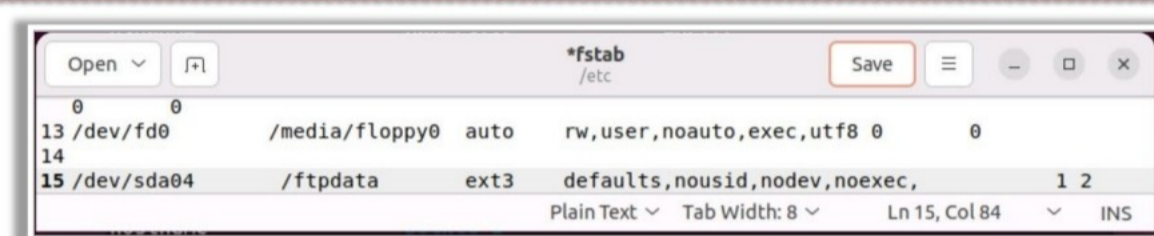
To remove X Window System, execute the following command:

```
yum groupremove "X Window System"
```

## Create Separate Disk Partitions For Linux System



- Separate OS files from user files for higher data security
- Ensure that the following file systems are mounted on separate partitions:
  - /usr
  - /home
  - /var and /var/tmp
  - /tmp
- Create separate partitions for **Apache** and **FTP** server roots
- Edit and update following configuration settings in **/etc/fstab** file
  - **noexec** : Do not set execution of any binaries on this partition (prevents execution of binaries but allows scripts)
  - **nodev**: Do not allow character or special devices on this partition (prevents use of device files such as zero, sda, etc.)
  - **nosuid** : Do not set SUID/SGID access on this partition (prevent the setuid bit)



Example: /etc/fstab configuration to restrict user access on /dev/sda04 ftp server root directory

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

## Create Separate Disk Partitions for Linux System

Disk partitions on a Linux system allow for easier recovery in the event of a corrupted file system, increases performance, and enhances both disk space efficiency and system security. Disk partition limits each file system's ability to expand. Hence, it is recommended to separate operating system files from user files to enhance data security. Keeping system and user files on different partitions helps eliminate the chance of attackers exploiting SUID programs and accessing restricted areas. Ensure that the following file systems are mounted on separate partitions:

- /usr: This partition stores the executable binaries, kernel source tree, and documentation.
- /home: Stores home directories.
- /var: Stores spool directory such as mail and printing and error log directory.
- /var/tmp: Stores temporary files or directories of programs between system reboots.
- /tmp: Stores temporary data files.

For Apache and FTP server roots, create separate partitions. Edit `/etc/fstab` file and add the following configurations:

- **noexec**: This partition allows the execution of scripts but not binaries.
- **nodev**: This partition prevents character or special devices. The device files such as zero, sda, etc., cannot be executed.
- **nosuid**: The user should not be able to set SUID/SGID access on this partition.

Example: `/etc/fstab` configuration to restrict user access on `/dev/sda04` ftp server root directory.

## Enable Disk Quota for All Users



- Implement **disk quotas** for users to limit the number of files a user can create on the system

```
alice@alice-Virtual-Machine:~$ sudo quota alice
alice@alice-Virtual-Machine:~$

GNU nano 3.2 /tmp//EdP.algwy3
Disk quotas for user James (uid 1001):
Filesystem blocks soft hard inodes soft$
/dev/sda1 12208 100 110 238 0$

alice@alice-Virtual-Machine:~$ sudo quota -vs alice
alice@alice-Virtual-Machine:~$
```

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

### Enable Disk Quota for All Users

Disk quota restricts the disk space and sends an alert to the system administrator before the partition becomes full or the user consumes excess disk space. Implement disk quotas for users to limit the number of files a user can create on the system. Disk quota can be configured for individual users or user groups.

To enable disk quota, follow the steps outlined below.

#### Step 1: Installing quota in Ubuntu

- Install quota package on Ubuntu using the following command.  

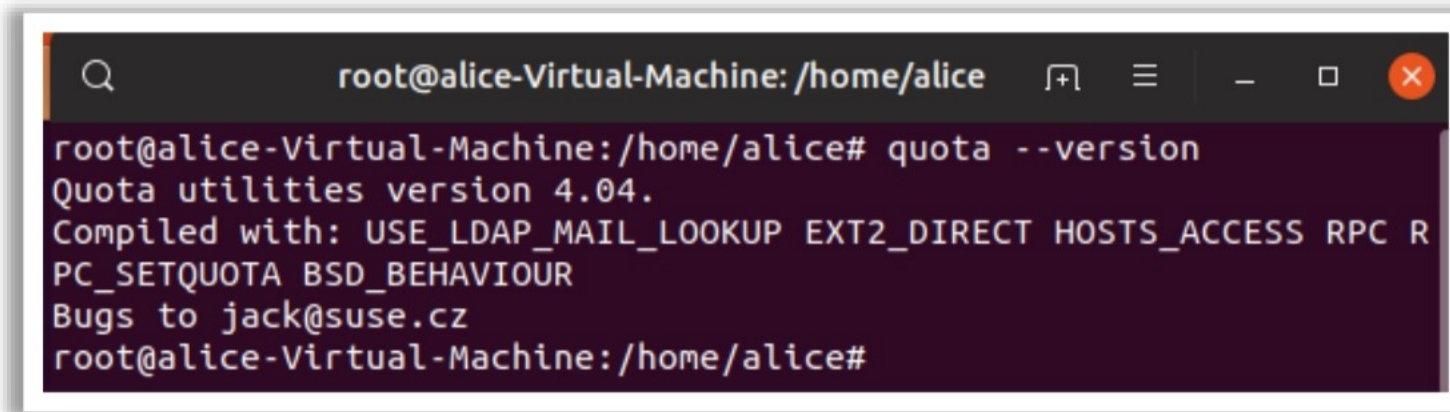
```
$ sudo apt install quota
```
- Enter 'Y' to continue the installation process.

```
root@alice-Virtual-Machine:/home/alice# sudo apt install quota
Reading package lists... Done
Building dependency tree
Reading state information... Done
Suggested packages:
 libnet-ldap-perl rpcbind
The following NEW packages will be installed:
 quota
0 upgraded, 1 newly installed, 0 to remove and 174 not upgraded.
Need to get 260 kB of archives.
After this operation, 1,577 kB of additional disk space will be used.
Get:1 http://in.archive.ubuntu.com/ubuntu disco/main amd64 quota amd64 4.04-2ubuntu2 [260 kB]
Fetched 260 kB in 0s (890 kB/s)
debconf: unable to initialize frontend: Dialog
```

Figure 6.37: Installing quota

- Use the following command to check the version.

```
$ quota --version
```



```
root@alice-Virtual-Machine: /home/alice
root@alice-Virtual-Machine: /home/alice# quota --version
Quota utilities version 4.04.
Compiled with: USE_LDAP_MAIL_LOOKUP EXT2_DIRECT HOSTS_ACCESS RPC R
PC_SETQUOTA BSD_BEHAVIOUR
Bugs to jack@suse.cz
root@alice-Virtual-Machine: /home/alice#
```

Figure 6.38: Checking quota version

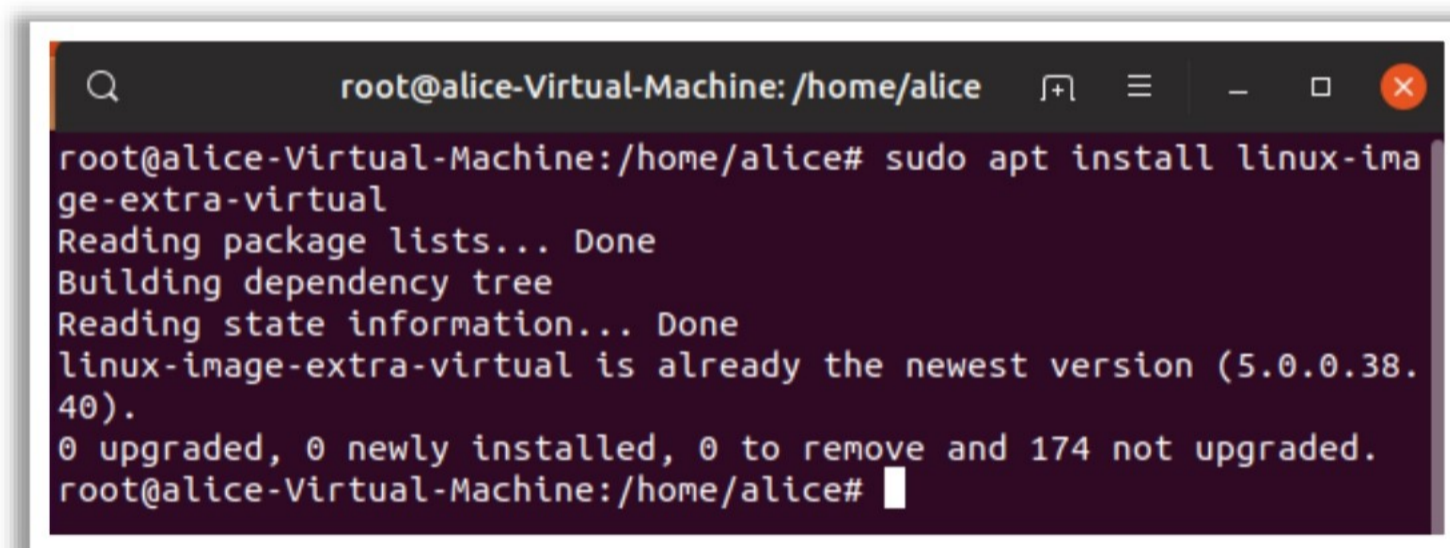
## Step 2: Installing the module for quota kernel

- Use the `find` tool for confirming the two modules `quota_v1` and `quota_v2` present in the `/lib/modules` directory.

```
$ find /lib/modules/ `uname -r` -type f -name '*quota_v*.ko*'
```

- If the two modules are not found, then use the following command to install them.

```
$ sudo apt install linux-image-extra-virtual
```



```
root@alice-Virtual-Machine: /home/alice
root@alice-Virtual-Machine: /home/alice# sudo apt install linux-image-extra-virtual
Reading package lists... Done
Building dependency tree
Reading state information... Done
linux-image-extra-virtual is already the newest version (5.0.0.38.40).
0 upgraded, 0 newly installed, 0 to remove and 174 not upgraded.
root@alice-Virtual-Machine: /home/alice#
```

Figure 6.39: Installing module for quota kernel

- Run the command to install the quota kernel module if not available.

## Step 3: Updating file system mount options

- The file system entry present in `/etc/fstab` file should be updated, which makes the quota active on a specific system by mounting it with related quota options.

```
$ sudo nano /etc/fstab
```

Depending on the representation of `/` or root file system, the path of `fstab` file and the desktop location will differ. Replace `/` pointing to the root system using the following command, which allows access to `userquota` and `grpquota`.

```
LABEL=cloudimg-rootfs/ext4 usrquota,grpquota 0 0
```

- To implement the changes, remount the file system.

```
$ sudo mount -o remount /
```

- The `grep` command is used to verify the use of the new options when mounting the file system in the `/proc/mounts` file. Use the following command to locate the root file system entry.

```
$ sudo cat /proc/mounts | grep ' / '
```

#### Step 4: Enabling disk quotas on Ubuntu

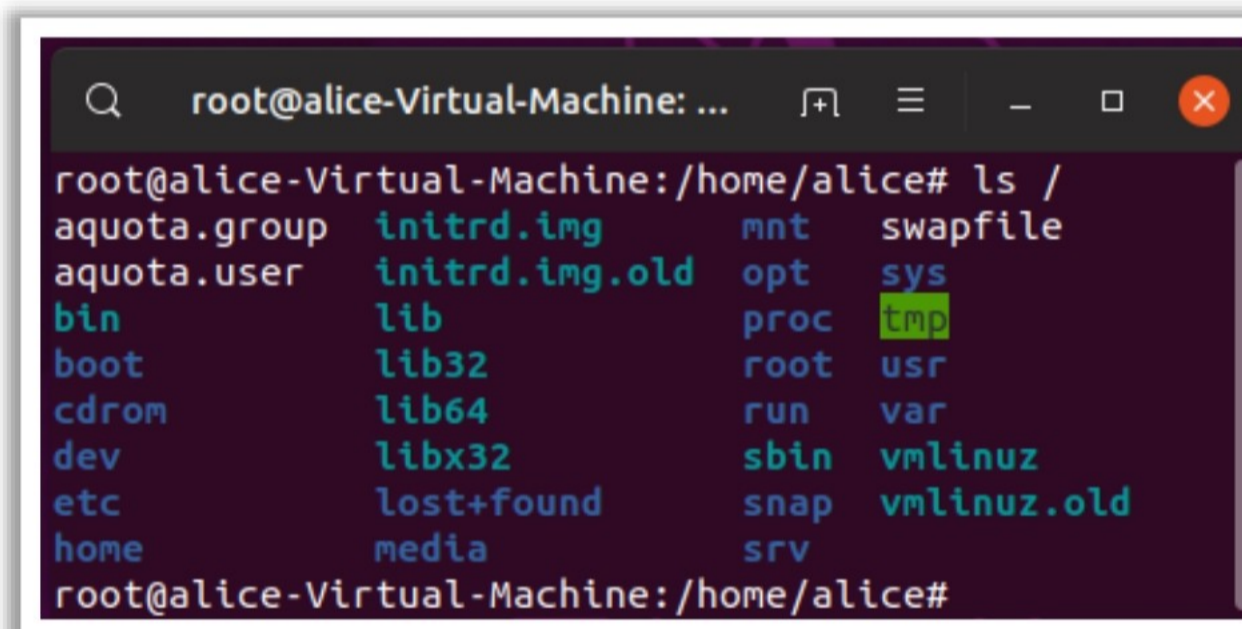
- Run the `quotacheck` command, which would create a user quota file and group quota file, which contain information about the limit and usage of the file system.

```
$ sudo quotacheck -ugm /
```

- Do not run the `quotacheck` command if the user does not want to enable the user or group quota files. Confirm by running the following command.

```
$ ls /
```

- The output will be as follows:



```
root@alice-Virtual-Machine: /home/alice# ls /
aquota.group initrd.img mnt swapfile
aquota.user initrd.img.old opt sys
bin lib proc tmp
boot lib32 root usr
cdrom lib64 run var
dev libx32 sbin vmlinuz
etc lost+found snap vmlinuz.old
home media srv
```

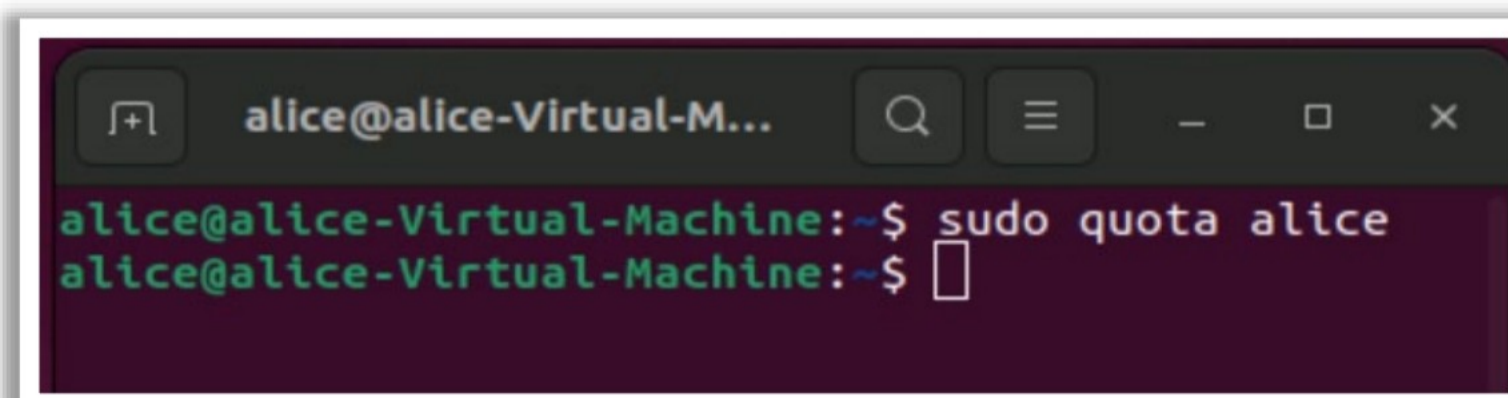
Figure 6.40: Enabling disk quota output

- Turn on the quota on the root file system with the following command.

```
$ sudo quotaon -v /
```

#### Step 5: Configure quotas for a single user

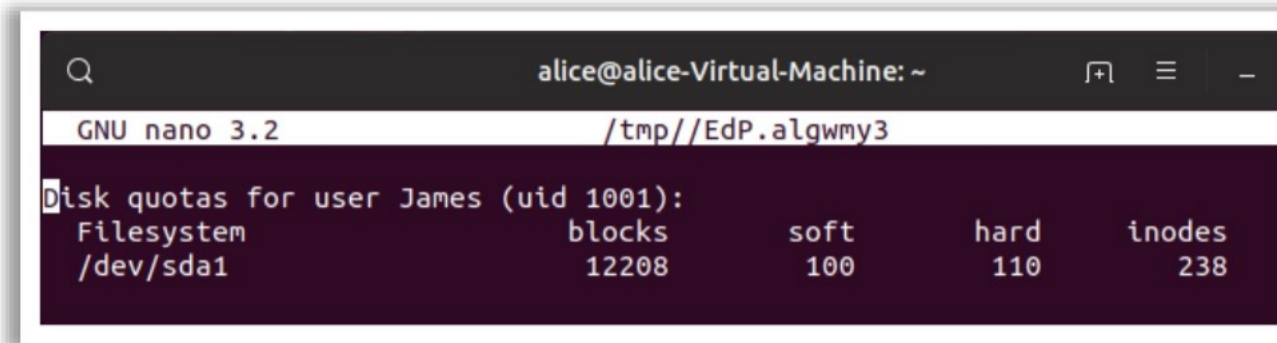
- Use `edquota` command to edit quota belonging to the user James.



```
alice@alice-Virtual-Machine: ~$ sudo quota alice
alice@alice-Virtual-Machine: ~$
```

Figure 6.41: Editing quota

- The output consists of a username, file system, usage of blocks, inodes, etc. The block-based quota controls the disk space.



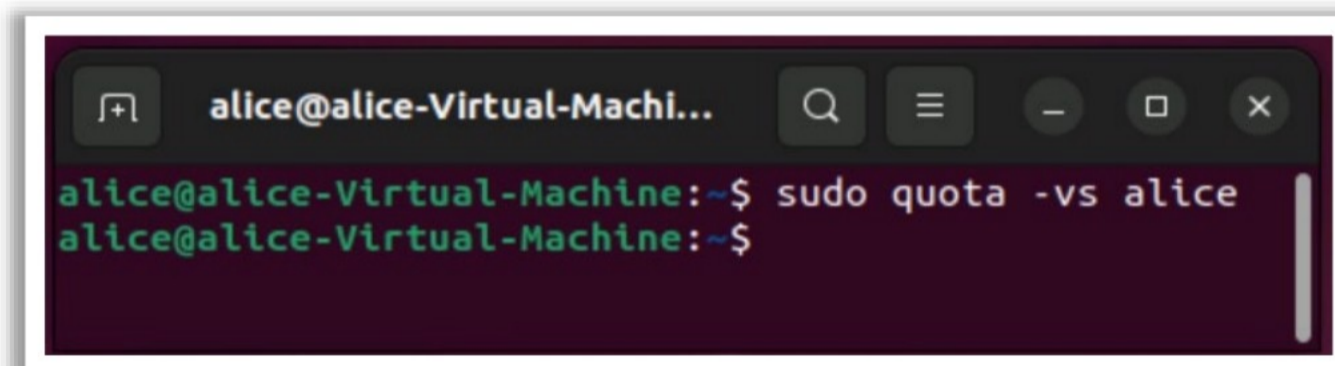
The screenshot shows a terminal window titled 'alice@alice-Virtual-Machine: ~'. The user is in the 'nano' editor editing a file at '/tmp//EdP.algwhy3'. The output of the 'quota' command for user James (uid 1001) is displayed as follows:

| Disk quotas for user James (uid 1001): |        |      |      |        |
|----------------------------------------|--------|------|------|--------|
| Filesystem                             | blocks | soft | hard | inodes |
| /dev/sda1                              | 12208  | 100  | 110  | 238    |

Figure 6.42: Editing quota output

- Update user block quota as per requirement. After necessary modification, close the file.
- Use the following command to check the new user quota.

```
$ sudo quota -vs james
```



The screenshot shows a terminal window titled 'alice@alice-Virtual-Machi...'. The user has entered the command 'sudo quota -vs alice' and the output is displayed on the next line.

```
alice@alice-Virtual-Machine:~$ sudo quota -vs alice
alice@alice-Virtual-Machine:~$
```

Figure 6.43: Checking for new user quota

## Understanding and Checking Linux File Permissions



Type **ls -l** command to display the list of files and their permissions under home directory

### Types of Permissions:

- r → denotes read permission
- w → denotes write permission
- x → denotes execute permission
- - → refers to No permission

### Permission Details:

- The first character in the directory list denotes file type d (if directory) else a file
- The next three characters denote user permissions.
- The next three characters denote group permissions.
- The final three characters denote other permissions

### Permission Groups: Owner and Group

- First name after number is owner name
- Second name after number ID group name

```
-Virtual-Machine:~
-Virtual-Machine:~$ ls -l
total
drwxr-xr-x 2 Owner Group 4096 Jul 5 10:21 Desktop
drwxr-xr-x 2 4096 Jul 5 10:21 Documents
drwxr-xr-x 2 4096 Jul 5 10:21 Downloads
-rw-r--r-- 1 8980 Jul 5 10:10 examples.desktop
drwxr-xr-x 2 4096 Jul 5 10:21 Music
drwxr-xr-x 2 4096 Jul 5 10:21 Pictures
drwxr-xr-x 2 4096 Jul 5 10:21 Public
drwxr-xr-x 2 4096 Jul 5 10:21 Templates
drwxr-xr-x 2 4096 Jul 5 10:21 Videos
```

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

## Understanding and Checking Linux File Permissions

Linux is a multi-user OS. This feature can raise security concerns as attackers can corrupt, change, or remove important data. For ensuring strong security, Linux categorizes the file authorization into two levels, ownership and permission, for effective security.

To understand and check the Linux file permissions and ownerships, the **ls -l** command lists out all the files and their permissions under the **home** directory. The lines give a lot of information about files and directories. Each line starts with ten characters. A few lines start with a '-' (file) and the remaining start with a 'd' (directory).

### Ownership

Each file and directory on the Linux system is assigned three types of users:

- User refers to the user permissions that only apply to the owner of the file/directory. These permissions do not impact the other users' actions. In the lines, the name after the digit is the owner's name. By default, the person who creates a file is called its owner.

For example, if the user's permissions for a few files is set as **rw-** (first set of three characters), it denotes the file's owner can only read (see) and write (change the contents) of the file's contents.

- Group refers to the group permissions that apply only to the group of the file/directory. These permissions do not impact other users' actions. The name after the owner/username denotes the group's owner in the lines. A user-group comprises multiple users and they get the same access permissions to the files.

For example, if the group's permissions for a few files is set as `r-x` (second set of three characters), it denotes the group's members can only read (only see) and execute those files but cannot change the file's contents.

- Others refers to the permissions that apply to all the other users (except owner and groups) on the system. These other users have neither created the file nor belong to a group that owns the file.

For example, if this permission for a few files is set as `r-` (last set of three characters), it denotes the other users who have accounts on the Linux system are allowed to read (only see) the contents of the files, but they are not allowed to change or execute the files.

## Permissions


By setting permissions, Linux distinguishes among the three user types (user, group, and others) to avoid a user from affecting the file contents of another user.

The next nine characters (example, `rw-rw-rw-`) in the lines reveal the security permissions for each file or directory. Assume those nine characters as three sets of three characters. Each of the three `rw-` refers to the operations the user/group/other can perform on the file/directory.

- '`r`' (read) permission allows reading the file contents.
- '`w`' (write) permission allows writing or modifying the file's contents.
- '`x`' (execute) permission allows executing a file if it is a program.
- If the character '`-`' appears instead of any of the `rw-` characters, it means the permissions are revoked.

The column after the group owner denotes the size of each file in bytes. The next three columns show the date and time at which the file was last modified. And the final column denotes the file name/directory name.

## Changing File Permissions



- 📌 Check for permission on **sensitive files**
- 📌 Use **chmod** command to change the permissions of a file or directory
- 💻 `chmod [permission Value] [File Name]`

Common Directory Permission Settings

| Value | Meaning                                                                                                                                                                                         |
|-------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 777   | (rwxrwxrwx) No restrictions on permissions. Anybody can list files, create new files in the directory, and delete files in the directory                                                        |
| 755   | (rwxr-xr-x) The directory owner has full access. All others can list the directory but cannot read or delete it. This setting is useful for directories that you wish to share with other users |
| 700   | (rwx-----) The directory owner has full access. Nobody else has any rights. This setting is useful for directories that only the user can use and must be kept private from others              |

Common File Permission Settings

| Value | Meaning                                                                                                                                                                                  |
|-------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 777   | (rwxrwxrwx) No restrictions on anything. Anybody can do anything. Generally, not a desirable setting                                                                                     |
| 755   | (rwxr-xr-x) The file owner may read, write, and execute the file. Others can read and execute the file. This setting is useful for all programs that are used by all users               |
| 700   | (rwx-----) The file owner may read, write, and execute the file. Nobody else has any rights. This setting is useful for programs that only user may use and are kept private from others |
| 666   | (rw-rw-rw-) All users can read and write the file                                                                                                                                        |
| 644   | (rw-r--r--) The owner can read and write a file, while others may only read the file. A very common setting where everybody may read but only the owner can make changes                 |
| 600   | (rw-----) Owner can read and write a file. Others have no rights. A common setting for files that the owner wants to keep private                                                        |

Copyright © by **EC-Council**. All Rights Reserved. Reproduction is Strictly Prohibited.

## Changing File Permissions

Changes in file permissions are needed to avoid unauthorized access to file contents by other users. By using the command `chmod` (change mode), permissions (read/write/execute) can be set on files/directories for the users (owner/group/others).

The syntax of the command is `chmod permissions filename`.

### chmod Command Modes

- **Absolute/numeric mode:** This method allows a three-digit octal number instead of characters for file permissions.

| Number | Permission Type          | Symbol |
|--------|--------------------------|--------|
| 0      | No permission            | ---    |
| 1      | Execute                  | --X    |
| 2      | Write                    | -W-    |
| 3      | Execute and write        | -WX    |
| 4      | Read                     | r--    |
| 5      | Read and execute         | r-X    |
| 6      | Read and write           | rw-    |
| 7      | Read, write, and execute | Rwx    |

Table 6.2: Numbers for All Permission Types

| Value | Meaning                                                                                                                                                                                                         |
|-------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 777   | ( <code>rwXrwXrwX</code> ) No restrictions on permissions. Anybody can list files, create new files in the directory, and delete files in the directory.                                                        |
| 755   | ( <code>rwXr-Xr-X</code> ) The directory owner has full access. All others can list the directory but cannot read or delete it. This setting is useful for directories that you wish to share with other users. |
| 700   | ( <code>rwX-----</code> ) The directory owner has full access. Nobody else has any rights. This setting is useful for directories that only the user can use and must be kept private from others.              |

Table 6.3: Common Directory Permission Settings

| Value | Meaning                                                                                                                                                                                                  |
|-------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 777   | ( <code>rwXrwXrwX</code> ) No restrictions on anything. Anybody can do anything. Generally, not a desirable setting.                                                                                     |
| 755   | ( <code>rwXr-Xr-X</code> ) The file owner may read, write, and execute the file. Others can read and execute the file. This setting is useful for all programs that are used by all users.               |
| 700   | ( <code>rwX-----</code> ) The file owner may read, write, and execute the file. Nobody else has any rights. This setting is useful for programs that only user may use and are kept private from others. |
| 666   | ( <code>rw-rw-rw</code> ) All users can read and write the file.                                                                                                                                         |
| 644   | ( <code>rw-r-r--</code> ) The owner can read and write a file, while others may only read the file. A very common setting where everybody may read but only the owner can make changes.                  |
| 600   | ( <code>rw-----</code> ) Owner can read and write a file. Others have no rights. A common setting for files that the owner wants to keep private.                                                        |

Table 6.4: Common File Sharing Settings

■ Symbolic mode:

This method uses mathematical symbols to modify the file/directory permissions on the three user types.

Steps to use symbolic mode:

- Use arguments (`u` for user, `g` for group, `o` for others, and `a` for all) or a combination of arguments (`ugo`, `ug`, etc.) to modify the user types. Use a '+' for adding, a '-' for removing, and a '=' for assigning permission.

- Specify permissions (`r`, `w`, `x`) or a combination of permissions to modify the existing or add permissions for files/directories.
- Use commas to modify more permissions.
- Finally, specify the name of the file.

For example:

- `chmod o+x abc.txt` refers to adding execution permissions for other users for `abc.txt`.
- `chmod ugo-rwx abc.txt` refers to removing all permissions for everyone for `abc.txt`.
- `chmod ug+rw,o-x xyz.mp4` refers to adding read and write permissions for user and group while removing execute permission for others for `xyz.mp4`.
- `chmod ug=rx,o+r abc.txt` refers to assigning read and write permissions for user and group for `abc.txt`.

### Changing Ownership and Group

For changing the ownership of a file/directory, use the command `chown user`. To change the user as well as the group for a file or directory use the command `chown user:group filename`.

To change group owner only, use the command `chgrp group_name filename`. Here, `chgrp` stands for change group.

## Check and Verify Permissions for Sensitive Files and Directories



| Permission | File Pathname       | Description                                                                          |
|------------|---------------------|--------------------------------------------------------------------------------------|
| 600        | /boot/grub/menu.lst | GRUB boot loader menu file                                                           |
| 400        | /etc/cron.allow     | List of users permitted to use cron to submit periodic jobs                          |
| 400        | /etc/cron.deny      | List of users who cannot use cron to submit periodic jobs                            |
| 644        | /etc/crontab        | System-wide periodic jobs                                                            |
| 644        | /etc/hosts.allow    | List of hosts allowed to use internet services that are started using TCP wrappers   |
| 644        | /etc/hosts.deny     | List of hosts denied access to internet services that are started using TCP wrappers |
| 644        | /etc/logrotate.conf | File that controls how log files rotate                                              |
| 644        | /etc/xinetd.conf    | Configuration file for xinetd server                                                 |
| 755        | /etc/xinetd.d       | Directory containing configuration files for specific                                |
| 755        | /var/log            | Directory with all log files                                                         |
| 644        | /var/log/lastlog    | Information about all previous logins                                                |
| 644        | /var/log/messages   | Main system message log file                                                         |
| 664        | /var/log/wtmp       | Information about current logins                                                     |
| 755        | /etc/pam.d          | Directory with configuration files for pluggable authentication modules (PAMs)       |

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

## Check and Verify Permissions for Sensitive Files and Directories (Cont'd)



| Permission | File Pathname        | Description                                                                                                                                                        |
|------------|----------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 644        | /etc/passwd          | Old-style password file with user account information but not the passwords                                                                                        |
| 755        | /etc/rc.d            | Directory with system-startup scripts                                                                                                                              |
| 600        | /etc/securetty       | TTY interfaces (terminals) from which root can log in                                                                                                              |
| 755        | /etc/security        | Policy files that control system access                                                                                                                            |
| 400        | /etc/shadow          | Files with encrypted passwords and password expiration information                                                                                                 |
| 400        | /etc/shutdown.allow  | Users who can shut down or reboot by pressing Ctrl+Alt+Delete                                                                                                      |
| 755        | /etc/ssh             | Directory with configuration files for the Secure Shell (SSH)                                                                                                      |
| 755        | /etc/sysconfig       | System configuration files                                                                                                                                         |
| 644        | /etc/sysctl.conf     | Kernel configuration parameters                                                                                                                                    |
| 644        | /etc/syslog.conf     | Configuration file for the syslogd server that logs messages                                                                                                       |
| 644        | /etc/udev/udev.conf  | Configuration file for udev – the program that provides the capability to dynamically name hot-pluggable devices and create the device files in the /dev directory |
| 600        | /etc/vsftpd          | Configuration file for the very secure FTP server                                                                                                                  |
| 600        | /etc/vsftpd.ftpusers | List of users who are not allowed to use FTP to transfer files                                                                                                     |

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

## Check and Verify Permissions for Sensitive Files and Directories

The table shown here includes the typical numeric permission settings for important system files in Linux. This may slightly vary depending on the Linux distribution.

After knowing the numeric permission values for common file and directory permission settings, it becomes quicker to identify the permissions given or changes in the permission values for sensitive files and directories of Linux. Network defender should compare and identify permission value allocations and changes in permission for the Linux hosts on their network.

## Disable Unwanted SUID and SGID Binaries



- SUID/SGID bits, **if enabled**, helps local or remote users in getting root privileges by exploiting the existing vulnerabilities in the file
- Find and remove "s" bits from the files to **disable** SUID/SGID bits

```
root@alice-Virtual-Machine: /home/alice
root@alice-Virtual-Machine: /home/alice# find / -perm /4000
/snap/core22/634/usr/bin/chfn
/snap/core22/634/usr/bin/chsh
/snap/core22/634/usr/bin/gpasswd
/snap/core22/634/usr/bin/mount
/snap/core22/634/usr/bin/newgrp
/snap/core22/634/usr/bin/passwd
/snap/core22/634/usr/bin/su
/snap/core22/634/usr/bin/sudo
/snap/core22/634/usr/bin/umount
/snap/core22/634/usr/lib/dbus-1.0/dbus-daemon-launch-helper
/snap/core22/634/usr/lib/openssh/ssh-keysign
/snap/core22/634/usr/lib/snapd/snap-confine
/snap/core20/1891/usr/bin/chfn
/snap/core20/1891/usr/bin/chsh
/snap/core20/1891/usr/bin/gpasswd
/snap/core20/1891/usr/bin/mount
/snap/core20/1891/usr/bin/newgrp
/snap/core20/1891/usr/bin/passwd
/snap/core20/1891/usr/bin/su
/snap/core20/1891/usr/bin/sudo
/snap/core20/1891/usr/bin/umount
/snap/core20/1891/usr/lib/dbus-1.0/dbus-daemon-launch-helper
```

View all files with SUID set

```
root@alice-Virtual-Machine: /home/alice
root@alice-Virtual-Machine: /home/alice# find / -perm /2000
/snap/core22/634/usr/bin/chage
/snap/core22/634/usr/bin/expiry
/snap/core22/634/usr/bin/ssh-agent
/snap/core22/634/usr/bin/wall
/snap/core22/634/usr/sbin/pam_extrausers_chkpwd
/snap/core22/634/usr/sbin/unix_chkpwd
/snap/core22/634/var/mail
/snap/core22/864/usr/bin/chage
/snap/core22/864/usr/bin/expiry
/snap/core22/864/usr/bin/ssh-agent
/snap/core22/864/usr/bin/wall
/snap/core22/864/usr/sbin/pam_extrausers_chkpwd
/snap/core22/864/usr/sbin/unix_chkpwd
/snap/core22/864/var/mail
/snap/core20/2015/usr/bin/chage
/snap/core20/2015/usr/bin/expiry
/snap/core20/2015/usr/bin/ssh-agent
/snap/core20/2015/usr/bin/wall
/snap/core20/2015/usr/sbin/pam_extrausers_chkpwd
/snap/core20/2015/usr/sbin/unix_chkpwd
/snap/core20/2015/var/mail
/snap/core20/1891/usr/bin/chage
```

View all files with SGID set

```
root@alice-Virtual-Machine: /usr
root@alice-Virtual-Machine: /usr# chmod a-s /usr/bin/chfn
root@alice-Virtual-Machine: /usr#
```

Remove the setuid bit from a file

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

## Disable Unwanted SUID and SGID Binaries

Binaries with `setuid` and `setgid` turned on can be of potential risk as an attacker could exploit any vulnerability in the binary to gain elevated or root privileges. Programs with SUID bit set execute with the privileges of the program file's owner, and programs with SGID bit set are executed with the privileges of the file's group owner. To prevent misuse of SUID/SGID bits enabled file containing vulnerabilities or bugs by local or remote users, it is recommended to find and disable all unnecessary SUID/SGID bits.

- The user needs to find all the files by using the following commands.

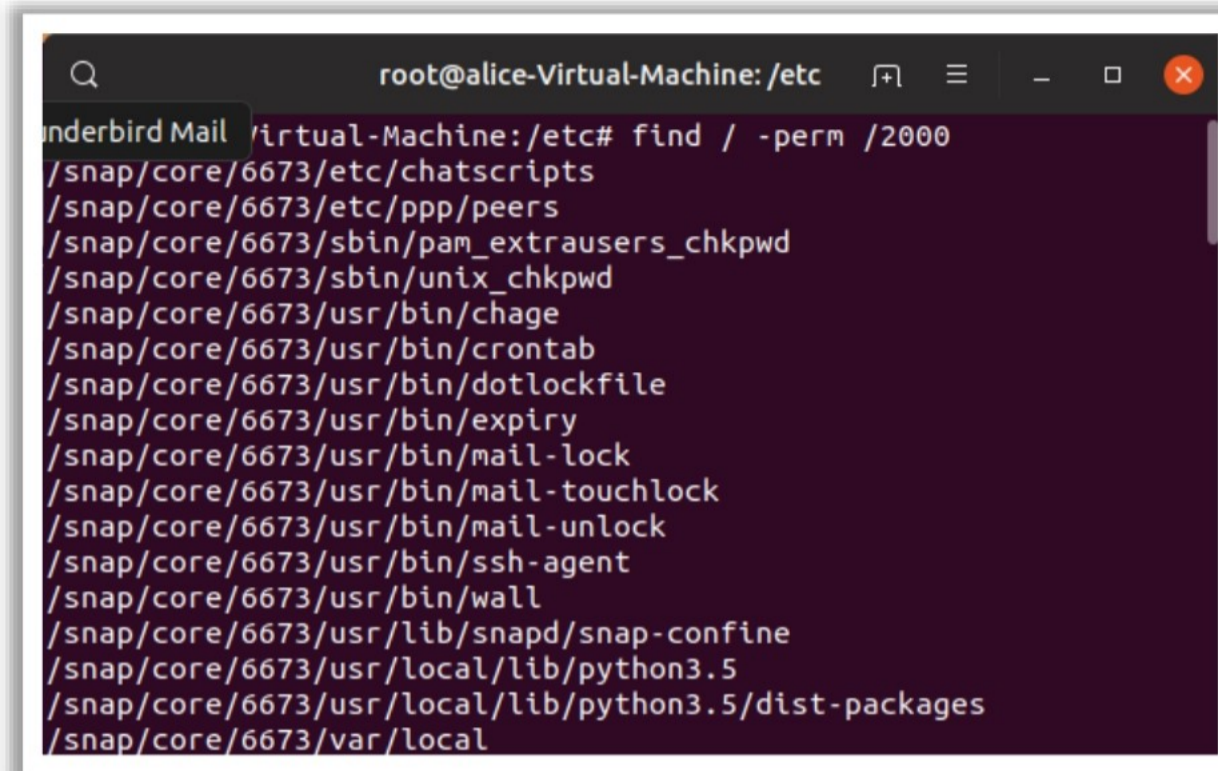
#See all set user id files:

```
find / -perm +4000
```

```
root@alice-Virtual-Machine: /etc
root@alice-Virtual-Machine: /etc# find / -perm /4000
/snap/core/6673/bin/mount
/snap/core/6673/bin/ping
/snap/core/6673/bin/ping6
/snap/core/6673/bin/su
/snap/core/6673/bin/umount
/snap/core/6673/usr/bin/chfn
/snap/core/6673/usr/bin/chsh
/snap/core/6673/usr/bin/gpasswd
/snap/core/6673/usr/bin/newgrp
/snap/core/6673/usr/bin/passwd
/snap/core/6673/usr/bin/sudo
/snap/core/6673/usr/lib/dbus-1.0/dbus-daemon-launch-helper
/snap/core/6673/usr/lib/openssh/ssh-keysign
/snap/core/6673/usr/lib/snapd/snap-confine
/snap/core/6673/usr/sbin/pppd
/snap/core/8689/bin/mount
/snap/core/8689/bin/ping
```

Figure 6.44: Finding user id files

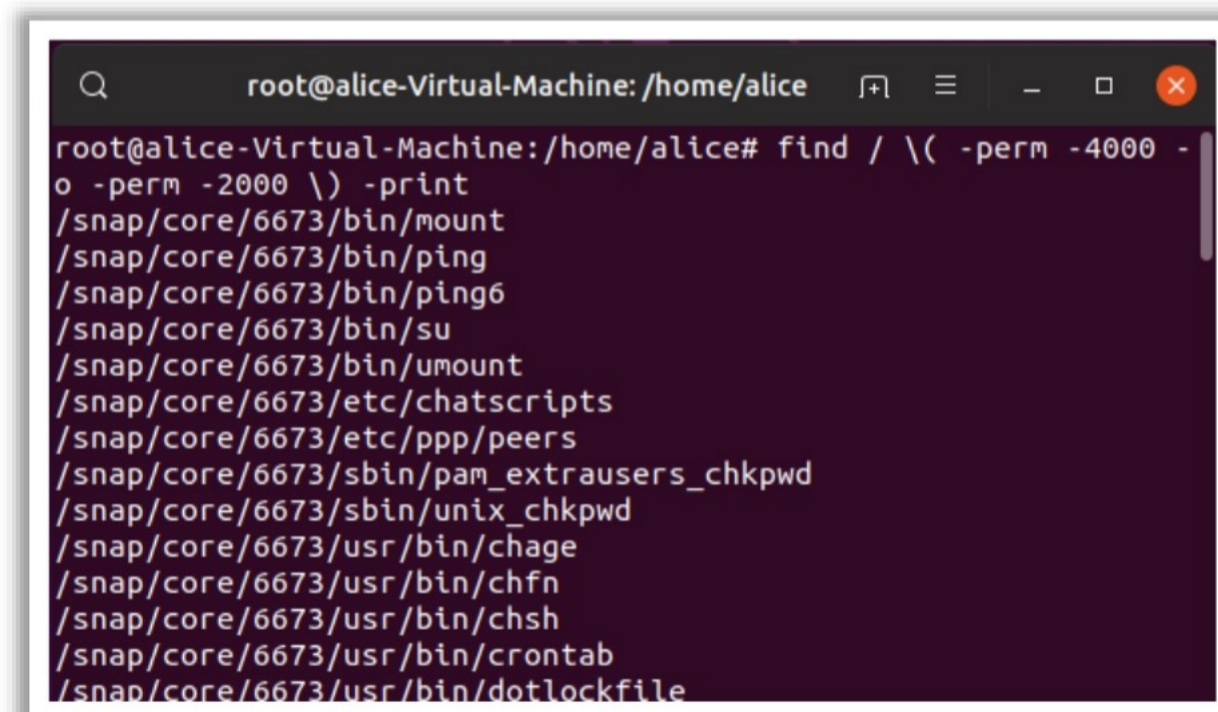
```
See all group id files
find / -perm +2000
```

A terminal window titled 'root@alice-Virtual-Machine: /etc' showing the output of the command 'find / -perm /2000'. The output lists various system files and directories, including chatscripts, peers, chkpwd, chage, crontab, dotlockfile, expiry, mail-lock, mail-touchlock, mail-unlock, ssh-agent, wall, snapd/snap-confine, python3.5, dist-packages, and var/local.

```
root@alice-Virtual-Machine: /etc# find / -perm /2000
/snap/core/6673/etc/chatscripts
/snap/core/6673/etc/ppp/peers
/snap/core/6673/sbin/pam_extrausers_chkpwd
/snap/core/6673/sbin/unix_chkpwd
/snap/core/6673/usr/bin/chage
/snap/core/6673/usr/bin/crontab
/snap/core/6673/usr/bin/dotlockfile
/snap/core/6673/usr/bin/expiry
/snap/core/6673/usr/bin/mail-lock
/snap/core/6673/usr/bin/mail-touchlock
/snap/core/6673/usr/bin/mail-unlock
/snap/core/6673/usr/bin/ssh-agent
/snap/core/6673/usr/bin/wall
/snap/core/6673/usr/lib/snapd/snap-confine
/snap/core/6673/usr/local/lib/python3.5
/snap/core/6673/usr/local/lib/python3.5/dist-packages
/snap/core/6673/var/local
```

Figure 6.45: Finding group id files

```
Or combine both in a single command
find / \(-perm -4000 -o -perm -2000 \) -print
```

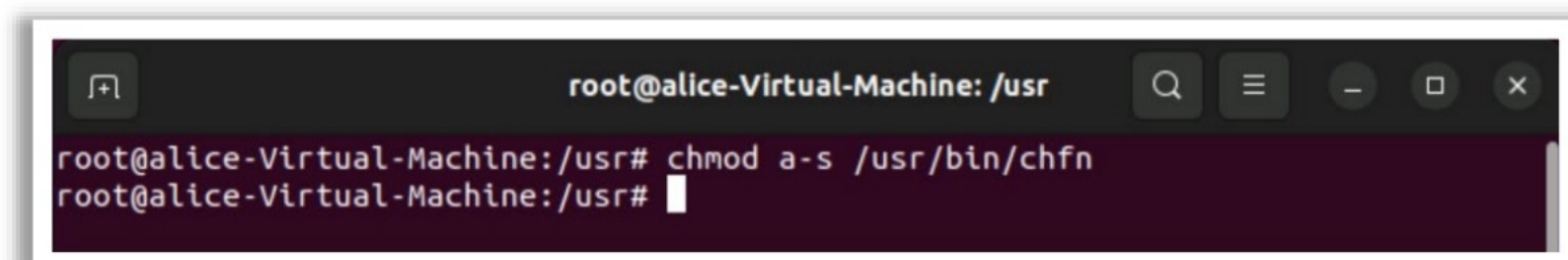
A terminal window titled 'root@alice-Virtual-Machine: /home/alice' showing the output of the command 'find / \( -perm -4000 -o -perm -2000 \) -print'. The output lists various system files and directories, including mount, ping, ping6, su, umount, chatscripts, peers, chkpwd, chfn, chsh, crontab, and dotlockfile.

```
root@alice-Virtual-Machine: /home/alice# find / \(-perm -4000 -o -perm -2000 \) -print
/snap/core/6673/bin/mount
/snap/core/6673/bin/ping
/snap/core/6673/bin/ping6
/snap/core/6673/bin/su
/snap/core/6673/bin/umount
/snap/core/6673/etc/chatscripts
/snap/core/6673/etc/ppp/peers
/snap/core/6673/sbin/pam_extrausers_chkpwd
/snap/core/6673/sbin/unix_chkpwd
/snap/core/6673/usr/bin/chage
/snap/core/6673/usr/bin/chfn
/snap/core/6673/usr/bin/chsh
/snap/core/6673/usr/bin/crontab
/snap/core/6673/usr/bin/dotlockfile
```

Figure 6.46: combining commands

- Remove the SUID/SGID binaries.

For example, to remove the SUID `/usr/bin/chfn` obtained from view all files with SUID set, execute the following command `chmod a-s /usr/bin/chfn`.

A terminal window titled 'root@alice-Virtual-Machine: /usr' showing the command 'chmod a-s /usr/bin/chfn' being executed.

```
root@alice-Virtual-Machine: /usr# chmod a-s /usr/bin/chfn
root@alice-Virtual-Machine: /usr#
```

Figure 6.47: Removing SUID/SGID binaries

## Remove or Rectify Permissions for World-Writable Files



- Any user can **edit** the world-writable files which can pose security risk to system
- View all world-writable file and **set** correct user and group permission to the required files or delete the unnecessary files

View World-writable Files without Sticky Bit

```
root@alice-Virtual-Machine: /home/alice
root@alice-Virtual-Machine:/home/alice# find /home/alice -xdev -type d \
(-perm -0002 -a ! -perm -1000 \) -print
```

View Noowner Files

```
root@alice-Virtual-Machine: /home/alice
root@alice-Virtual-Machine:/home/alice# find /home/alice -xdev \(-nouser -o -nogroup \) -print
root@alice-Virtual-Machine:/home/alice#
```

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

## Remove or Rectify Permissions for World-Writable Files

World-writable files and directories can, as the name suggests, be edited by any user on the system. This can present a security issue, particularly if system files or configuration are left as world writable. View all world-writable files, correct the user and group permission to the required files or delete the necessary files. You should regularly monitor for recurring world-writable files, as they may be a symptom of a misconfigured application/user account.

Steps to find world-writable files are listed below:

- Use the below command to locate world-writable files and directories on the machine.

```
find /dir -xdev -perm +o=w ! \(-type d -perm +o=t \) ! -type l -print
```

```
root@alice-Virtual-Machine: /home...
root@alice-Virtual-Machine:/home/alice# find /etc -xdev
-perm +o=w ! \(-type d -perm +o=t \) ! -type l -print
root@alice-Virtual-Machine:/home/alice#
```

Figure 6.48: Locating world-writable files

- Command to locate all directories that are world writable and do not have their sticky bits set. This command will discover and print these for /home/alice directory.

```
root@alice-Virtual-Machine: /home/alice
root@alice-Virtual-Machine:/home/alice# find /home/alice -xdev -type d \
(-perm -0002 -a ! -perm -1000 \) -print
```

Figure 6.49: Locating directories

If the above command produces any output, fix each reported directory `/dir`

- Using the `chmod` command.

```
find /webroot -xdev -type d \(-perm -0002 -a ! -perm -1000 \) -
print0 | xargs -0 chmod +t
```

- Or, review each directory and set permission as per requirements:

```
chmod +t /path/to/dir
```

- Command to disable world write access to a file.

```
chmod o-w file
```

- Command to find and interactively fix world-writable files.

```
find /dir -xdev -perm +o=w ! \(-type d -perm +o=t \) ! -type l -
ok chmod -v o-w {} \;
```


- Command to prevent newly created files from being world writable.

```
umask 002
```

- Command for files that are not owned by someone.

```
Find /home/alice -xdev \(-nouser -o -nogroup \) -print
```

## Disable USB Storage



By **default**, Linux allows mounting of removable devices to the system

**Disable** USB storage in Linux to prevent data theft using a removable media

01

Disable USB storage using the **system BIOS configuration** option

02

Disable **kernel support** for USB via GRUB

03

In **Debian-based distribution**  
**Block usb-storage** module from loading into Linux kernel

04

In **Red Hat-based distribution**  
Disable USB storage With using **fake install** or **Blacklist usb-storage**

### Block usb-storage Module

```
alice@alice-Virtual-Machine: /lib/modu...
alice@alice-Virtual-Machine: /lib/modules/5.3.0-24-generic/kernel/drivers/usb/storage$ sudo mv usb-storage.ko usb-storage.ko.blacklist
alice@alice-Virtual-Machine: /lib/modules/5.3.0-24-generic/kernel/drivers/usb/storage$ ls
uas.ko ums-freecom.ko ums-realtek.ko
ums-alauda.ko ums-isd200.ko ums-sddr09.ko
ums-cypress.ko ums-jumpshot.ko ums-sddr55.ko
ums-datafab.ko ums-karma.ko ums-usbata.ko
ums-eneub6250.ko ums-onetouch.ko usb-storage.ko.blacklist
alice@alice-Virtual-Machine: /lib/modules/5.3.0-24-generic/kernel/drivers/usb/storage$
```

### Blacklist usb-storage

```
*blacklist.conf [Read-Only]
/etc/modprobe.d
This file lists those modules which we don't want to be loaded by
alias expansion, usually so some other driver will be loaded for the
device instead.

evbug is a debug tool that should be loaded explicitly
blacklist evbug

these drivers are very simple, the HID drivers are usually preferred
blacklist usbmouse
blacklist usbkbd
blacklist usb-storage
```

Copyright © by **EC-Council**. All Rights Reserved. Reproduction is Strictly Prohibited.

## Disable USB Storage

Linux allows mounting of removable devices to the system and shows files and folders contained in devices by default. This allows attackers to copy files or to run a malicious script on the system. With a few configuration changes, it is possible to disable USB on Linux machines for unprivileged users.

Methods to disable USB storage are listed below.

- Methods to Disable USB Storage based on Red Hat distribution:
  - Using fake install:
    - Create and open the blacklist file named `block_usb.conf` in the folder `/etc/modprobe.d`,  

```
$ sudo vim /etc/modprobe.d/block_usb.conf
```
    - Enter `install usb-storage /bin/true` command.
    - Save the file and restart the system to apply changes.
  - Blacklisting USB-storage:
    - Create the file `/etc/modprobe.d/blacklist.conf` to blacklist `usb-storage`.  

```
$ sudo vim /etc/modprobe.d/blacklist.conf
```

- Enter the following line to blacklist the usb.

```
blacklist usb-storage
```

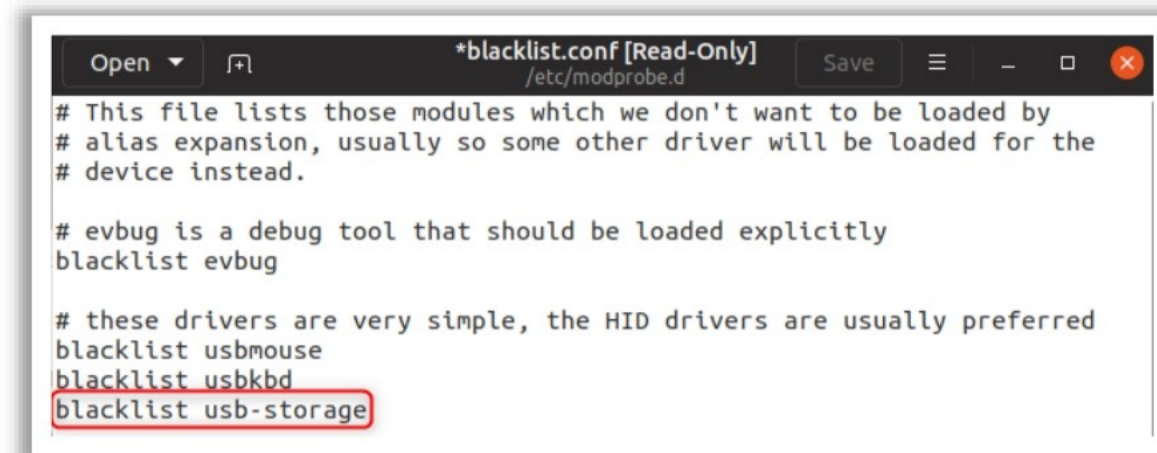


Figure 6.50: Blacklisting usb storage

- Save file and exit.
- Restart the system to take effect changes.

Note: Other privileged users may be able to load the usb-storage module again by executing the following command.

```
$ sudo modprobe usb-storage
```

- Removing/moving the USB driver: Change directory to kernel usb storage modules path and rename the usb-storage.ko.xz module to usb-storage.ko.xz.blacklist by issuing the below commands to block USB storage module from loading into kernel.

```
cd /lib/modules/`uname -r`
/kernel/drivers/usb/storage/
ls
#sudo mv usb-storage.ko usb-storage.ko.xz.blacklist
```

- In Debian-based Linux distributions: Enter the below commands to block usb-storage module from loading into Linux kernel.

```
cd /lib/modules/ `uname -r` /kernel/drivers/usb/storage/
ls
#sudo mv usb-storage.ko usb-storage.ko.blacklist
```

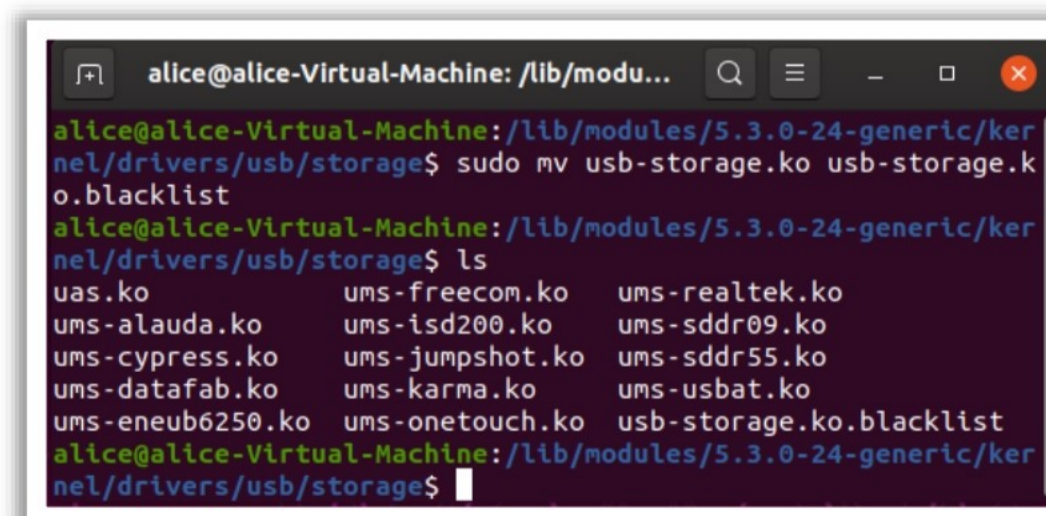


Figure 6.51: Blocking usb-storage module

## Block USB in Debian and Ubuntu

- BIOS option: Disable USB from system BIOS configuration option. Ensure BIOS is password protected. Nobody can boot the system from USB.
- GRUB option: To disable kernel support for USB through GRUB, open `groub.conf` or `menu.lst` and add “nousb” to the kernel line.

```
kernel /vmlinuz-2.6.18-128.1.1.el5 ro root=LABEL=/ console=tty0
console=ttyS1,19200n8 nousb
```

## Linux Hardening Checklist: User Access and Passwords



Create an account for each user who should access the system



Enforce the use of **strong passwords**



Use **sudo** to delegate admin access

Copyright © by **EC-Council**. All Rights Reserved. Reproduction is Strictly Prohibited.

### Linux Hardening Checklist: User Access and Passwords

- Create an account for every user who should access the system instead of shared accounts/passwords. This can make it easy to keep an audit trail and remove access when not needed.
- Enforce the use of strong passwords using password security rules that can be set in `/etc/pam.d/password-auth`.
- Use `sudo` to delegate administrator access by editing the configuration file `/etc/sudoers` with the command `visudo`. This can create fine-grained access to run commands as root/other user IDs.



---

### LO#05: Discuss Linux network security and remote access

---

Copyright © by **EC-Council**. All Rights Reserved. Reproduction is Strictly Prohibited.

## **LO#05: Discuss Linux Network Security and Remote Access**

This section enlists security measures required to improve Linux network and remote access security.

## Configure sysctl to Secure Linux Kernel



- System Control Interface (**sysctl**) help you make changes to a running Linux kernel. Configure Linux kernel for certain security settings to secure Linux kernel
- /etc/sysctl.conf** a text file contains sysctl values that are set and read during booting

Edit **/etc/sysctl.conf** to:

- Restrict network-transmitted configuration for IPv4
- Restrict network-transmitted configuration for IPv6
- Turn on execshield protection
- Prevent syn flood attack
- Turn on source IP address verification
- Prevent spoofing attack against the IP address of the server
- Logs various suspicious packets (spoofed packets, source-routed packets, and redirects)

Copyright © by **EC-Council**. All Rights Reserved. Reproduction is Strictly Prohibited.

## Configure sysctl to Secure Linux Kernel (Cont'd)



Sample **/etc/sysctl.conf** for Linux Server Hardening

```
#####
Avoid a smurf attack
net.ipv4.icmp_echo_ignore_broadcasts = 1

Turn on protection for bad icmp error messages
net.ipv4.icmp_ignore_bogus_error_responses = 1

Turn on syncookies for SYN flood attack protection
net.ipv4.tcp_syncookies = 1

Turn on and log spoofed, source routed, and redirect packets
net.ipv4.conf.all.log_martians = 1
net.ipv4.conf.default.log_martians = 1

No source routed packets here
net.ipv4.conf.all.accept_source_route = 0
net.ipv4.conf.default.accept_source_route = 0

Turn on reverse path filtering
net.ipv4.conf.all.rp_filter = 1
net.ipv4.conf.default.rp_filter = 1

Make sure no one can alter the routing tables
net.ipv4.conf.all.accept_redirects = 0
net.ipv4.conf.default.accept_redirects = 0
net.ipv4.conf.all.secure_redirects = 0
net.ipv4.conf.default.secure_redirects = 0

Don't act as a router
net.ipv4.ip_forward = 0
net.ipv4.conf.all.send_redirects = 0
net.ipv4.conf.default.send_redirects = 0

Turn on execshield for reducing worm or other automated remote attacks
kernel.exec-shield = 1
kernel.randomize_va_space = 1
```

```
Tune IPv6
net.ipv6.conf.default.router_solicitations = 0
net.ipv6.conf.default.accept_ra_rtr_pref = 0
net.ipv6.conf.default.accept_ra_pinfo = 0
net.ipv6.conf.default.accept_ra_defrtr = 0
net.ipv6.conf.default.autoconf = 0
net.ipv6.conf.default.dad_transmits = 0
net.ipv6.conf.default.max_addresses = 1

Increase system file descriptor limit
fs.file-max = 65535

Allow for more PIDs (Prevention of fork() failure error message)
kernel.pid_max = 65536

Increase system IP port limits
net.ipv4.ip_local_port_range = 2000 65000

Tuning Linux network stack to increase TCP buffer size. Set the max OS
send buffer size (wmem) and receive buffer size (rmem) to 12 MB for
queues on all protocols.
net.core.rmem_max = 8388608
net.core.wmem_max = 8388608

set minimum size, initial size and max size
net.ipv4.tcp_rmem = 10240 87380 12582912
net.ipv4.tcp_wmem = 10240 87380 12582912

Value to set for queue on the INPUT side when incoming packets are
faster then the kernel process on them.
net.core.netdev_max_backlog = 5000

For increasing transfer window, enable window scaling
net.ipv4.tcp_window_scaling = 1
```

Copyright © by **EC-Council**. All Rights Reserved. Reproduction is Strictly Prohibited.

### Configure sysctl to Secure Linux Kernel

The System Control interface or utility in Linux distribution `sysctl` helps to edit Linux kernel parameters (system settings) at runtime. The file `/etc/sysctl.conf` is a text file that comprises `sysctl` values to be read in and set by `sysctl` at boot time. The network defender can change the parameter values and make the system secure. The configuration of `sysctl`

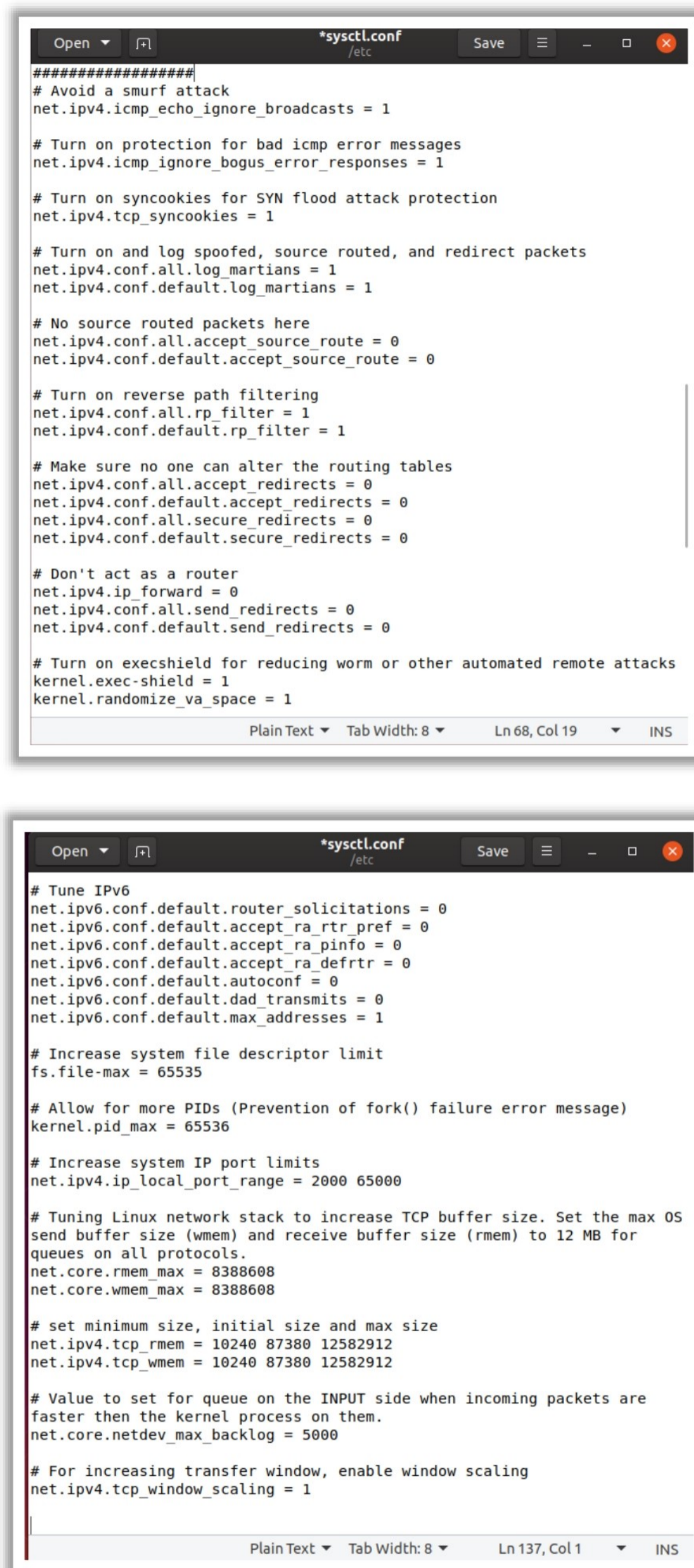
provides additional security and helps protect the host from different network-level attacks such as a man-the-middle attack, spoofing attack, etc.

Configure `/etc/sysctl.conf` for various Linux networking and system settings such as:

- Limits configuring network-transmitted for IPv4
- Limits configuring network-transmitted for IPv6
- Turns on execshield protection
- Prevents the common “syn flood attack”
- Turns on source IP address verification
- Avoids an attacker from using a spoofing attack against the IP address of the server
- Logs various types of suspicious packets (spoofed packets, source-routed packets, and redirects)

Steps to configure `sysctl` securely in Linux distribution by editing the configuration file:

- Use the command `Sudo nano /etc/sysctl.conf` to access the configuration file.
  - Open the file. The administrator can make the following basic security configuration.
    - Disable IP forwarding
    - Disable the send packet redirect
    - Disable ICMP redirect acceptance
    - Enable bad Error Message protection
  - Steps to change the parameter values in `/etc/sysctl.conf` file.
    - Enter the command `sudo gedit /etc/sysctl.conf` (enter the required administrator password when prompted for password).
    - Press enter; the `sysctl.conf` file will open in terminal. Configure the `sysctl` file as shown in the following figures.



```
#####
Avoid a smurf attack
net.ipv4.icmp_echo_ignore_broadcasts = 1

Turn on protection for bad icmp error messages
net.ipv4.icmp_ignore_bogus_error_responses = 1

Turn on syncookies for SYN flood attack protection
net.ipv4.tcp_syncookies = 1

Turn on and log spoofed, source routed, and redirect packets
net.ipv4.conf.all.log_martians = 1
net.ipv4.conf.default.log_martians = 1

No source routed packets here
net.ipv4.conf.all.accept_source_route = 0
net.ipv4.conf.default.accept_source_route = 0

Turn on reverse path filtering
net.ipv4.conf.all.rp_filter = 1
net.ipv4.conf.default.rp_filter = 1

Make sure no one can alter the routing tables
net.ipv4.conf.all.accept_redirects = 0
net.ipv4.conf.default.accept_redirects = 0
net.ipv4.conf.all.secure_redirects = 0
net.ipv4.conf.default.secure_redirects = 0

Don't act as a router
net.ipv4.ip_forward = 0
net.ipv4.conf.all.send_redirects = 0
net.ipv4.conf.default.send_redirects = 0

Turn on execshield for reducing worm or other automated remote attacks
kernel.exec-shield = 1
kernel.randomize_va_space = 1

Tune IPv6
net.ipv6.conf.default.router_solicitations = 0
net.ipv6.conf.default.accept_ra_rtr_pref = 0
net.ipv6.conf.default.accept_ra_pinfo = 0
net.ipv6.conf.default.accept_ra_defrtr = 0
net.ipv6.conf.default.autoconf = 0
net.ipv6.conf.default.dad_transmits = 0
net.ipv6.conf.default.max_addresses = 1

Increase system file descriptor limit
fs.file-max = 65535

Allow for more PIDs (Prevention of fork() failure error message)
kernel.pid_max = 65536

Increase system IP port limits
net.ipv4.ip_local_port_range = 2000 65000

Tuning Linux network stack to increase TCP buffer size. Set the max OS
send buffer size (wmem) and receive buffer size (rmem) to 12 MB for
queues on all protocols.
net.core.rmem_max = 8388608
net.core.wmem_max = 8388608

set minimum size, initial size and max size
net.ipv4.tcp_rmem = 10240 87380 12582912
net.ipv4.tcp_wmem = 10240 87380 12582912

Value to set for queue on the INPUT side when incoming packets are
faster then the kernel process on them.
net.core.netdev_max_backlog = 5000

For increasing transfer window, enable window scaling
net.ipv4.tcp_window_scaling = 1
```

Figure 6.52: Sample /etc/sysctl.conf for Linux Server Hardening

## Host-based Firewall Protection with Iptables



- Iptables is a **built-in** firewall utility for Linux OSes
- Iptables comes pre-installed on any Linux distribution. However, you can update/install it with `sudo apt-get install iptables` command

```
alice@alice-Virtual-Machine: ~/Desktop
alice@alice-Virtual-Machine:~/Desktop$ iptables -h
iptables v1.8.3

Usage: iptables -[ACD] chain rule-specification [options]
iptables -I chain [rulenum] rule-specification [options]
iptables -R chain rulenum rule-specification [options]
iptables -D chain rulenum [options]
iptables -[LS] [chain [rulenum]] [options]
iptables -[FZ] [chain] [options]
iptables -[NX] chain
iptables -E old-chain-name new-chain-name
iptables -P chain target [options]
iptables -h (print this help information)

Commands:
Either long or short options are allowed.
--append -A chain Append to chain
--check -C chain Check for the existence of a rule
--delete -D chain Delete matching rule from chain
--delete -D chain rulenum Delete rule rulenum (1 = first)
 from chain
--insert -I chain [rulenum] Insert in chain as rulenum (default 1=first)
--replace -R chain rulenum Replace rule rulenum (1 = first)
```

| Task                        | Iptable Commands                                                           |
|-----------------------------|----------------------------------------------------------------------------|
| Filtering non TCP packets   | <code>iptables -A INPUT -p tcp ! --syn -m state --state NEW -j DROP</code> |
| Blocking XMAS scan Attack   | <code>iptables -A INPUT -p tcp --tcp-flags ALL -j DROP</code>              |
| Drop any NULL packets       | <code>iptables -A INPUT -f -j DROP</code>                                  |
| Drop any fragmented packets | <code>iptables -A INPUT -f -j DROP</code>                                  |

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

## Host-based Firewall Protection with Iptables (Cont'd)



Existing rules can be checked using  
`sudo iptables -L -n -v`  
command

Specific IP address can be block using  
Iptables Firewall  
`iptables -A INPUT -s 10.10.10.55 -j DROP`

```
root@alice-Virtual-Machine: /home/alice
root@alice-Virtual-Machine:/home/alice# iptables -L -n -v
iptables v1.8.7 (nf_tables)
root@alice-Virtual-Machine:/home/alice# iptables -L -n -v
Chain INPUT (policy ACCEPT 0 packets, 0 bytes)
 pkts bytes target prot opt in out source destination

Chain FORWARD (policy ACCEPT 0 packets, 0 bytes)
 pkts bytes target prot opt in out source destination

Chain OUTPUT (policy ACCEPT 0 packets, 0 bytes)
 pkts bytes target prot opt in out source destination
root@alice-Virtual-Machine:/home/alice#
```

```
root@alice-Virtual-Machine: /home/alice
root@alice-Virtual-Machine:/home/alice# iptables -A INPUT -s 10.10.10.55 -j DROP
root@alice-Virtual-Machine:/home/alice#
```

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

### Host-based Firewall Protection with Iptables

Host-based firewalls provide enhanced security against threats. Linux systems support a kernel-based packet filter that is suitable for using host-based firewalls.

The advantages of host-based firewalls are listed below:

- It protects against firewall failure. Adding another firewall can be helpful if the primary firewall fails because the attack that causes the primary firewall failure cannot affect the host-based firewall.
- It is simple to configure a host-based firewall because the host requires support for just a few protocols to function. Simplicity makes verification of the ruleset simpler.
- It can help secure against the threats originating within a corporate network and reduce the risks of misconfigured software on a host.
- It can be configured to support a single set of applications and to block everything.

## Iptables

Iptables is a command-line firewall utility that can allow or deny traffic. Iptables is preinstalled in a Linux system. In order to update or install iptables, the user needs to regain the iptables package using the command:

```
sudo apt-get install iptables
```

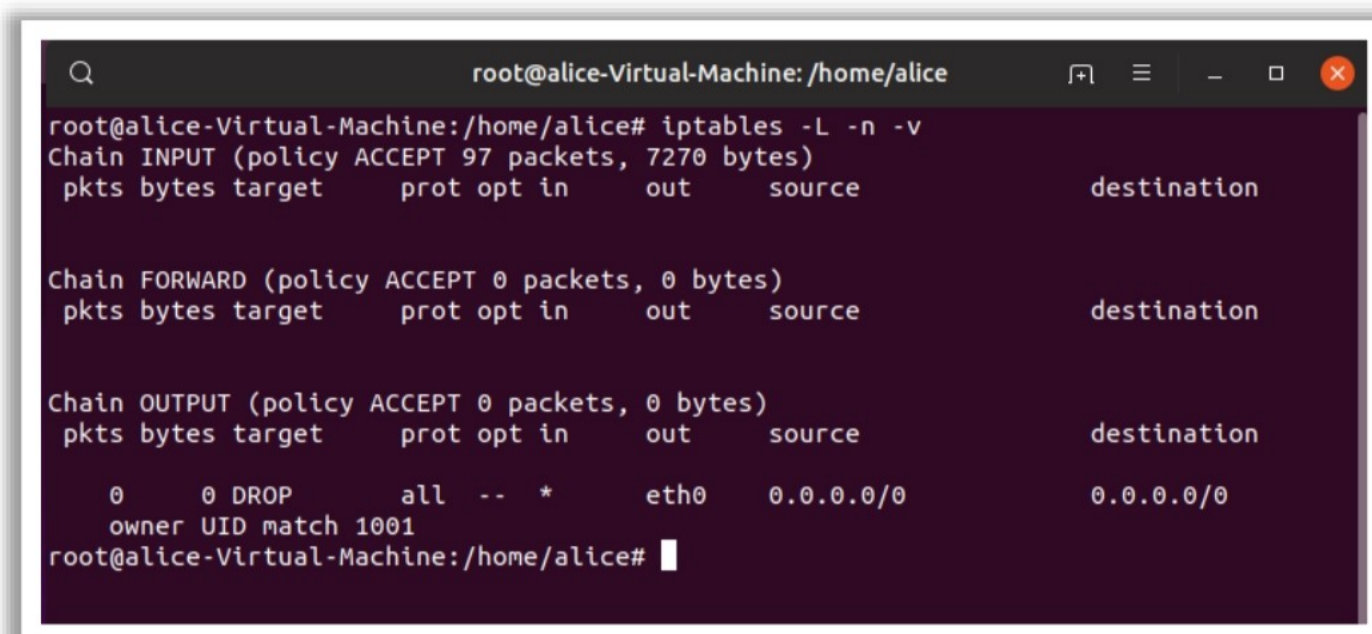
Every packet traversing through the filter system is assigned to an appropriate table depending on the tasks performed by the packet. The table contains chains that display the details of the destination of the packet. The tables can be used to create rules and the user has the facility to create their own chains and link them from the built-in chains. This facilitates the ability to create complex rules. However, the user needs to be extra alert while using the `iptables` command as any small error in the command can lock the system and may require the user to fix the error manually.

There are three different types of chains:

- Input: The input chain verifies the incoming connections and its behavior. Iptables compares the IP address and port of the incoming connection to a rule in the chain.
- Forward: The forward chain mainly forwards the incoming connections to its destination. The command `iptables -L -v` verifies whether an incoming connection needs a forward chain.
- Output: The output chain is used for output connections, wherein the chain checks for the output chain and decides whether to allow or deny the output request.

## Example iptables firewall rules:

- Check the existing rules using the `sudo iptables -L -n -v` command.



```

root@alice-Virtual-Machine: /home/alice
root@alice-Virtual-Machine:/home/alice# iptables -L -n -v
Chain INPUT (policy ACCEPT 97 packets, 7270 bytes)
 pkts bytes target prot opt in out source destination

Chain FORWARD (policy ACCEPT 0 packets, 0 bytes)
 pkts bytes target prot opt in out source destination

Chain OUTPUT (policy ACCEPT 0 packets, 0 bytes)
 pkts bytes target prot opt in out source destination

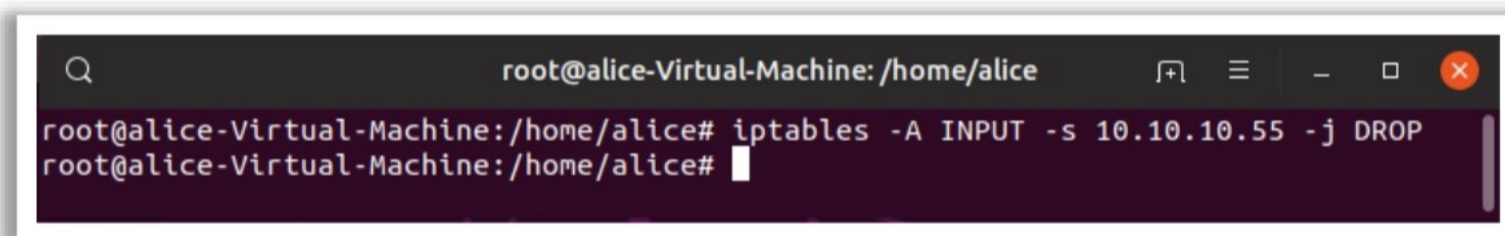
 0 0 DROP all -- * eth0 0.0.0.0/0 0.0.0.0/0
 owner UID match 1001
root@alice-Virtual-Machine:/home/alice#

```

Figure 6.53: iptables firewall rules

- Check the rules for a specific table using the command `# iptables -t nat -L -v -n`.
- Block the specified IP address using iptables firewall.

`Iptables -A INPUT -s 10.10.10.55 -j DROP`



```

root@alice-Virtual-Machine: /home/alice
root@alice-Virtual-Machine:/home/alice# iptables -A INPUT -s 10.10.10.55 -j DROP
root@alice-Virtual-Machine:/home/alice#

```

Figure 6.54: Blocking specific IP address

- Block specific port on iptables firewall using the command `# iptables -A OUTPUT -p tcp --dport xxx -j DROP`.
- Block Facebook on Iptables firewall using the command `# iptables -A OUTPUT -p tcp -d 66.220.144.0/20 -j DROP`.

| Task                               | Iptables Commands                                                                                        |
|------------------------------------|----------------------------------------------------------------------------------------------------------|
| Filtering non-TCP packets          | <code>iptables -A INPUT -p tcp ! --syn -m state --state NEW -j DROP</code>                               |
| Blocking XMAS scan attack          | <code>iptables -A INPUT -p tcp --tcp-flags ALL ALL -j DROP</code>                                        |
| Drop any NULL packets              | <code>iptables -A INPUT -f -j DROP</code>                                                                |
| Drop any fragmented packets        | <code>iptables -A INPUT -f -j DROP</code>                                                                |
| Block network flood on Apache port | <code>iptables -A INPUT -p tcp --dport 80 -m limit --limit 100/minute --limit-burst 200 -j ACCEPT</code> |

|                                        |                                                                 |
|----------------------------------------|-----------------------------------------------------------------|
| Block incoming ping requests           | # iptables -A INPUT -p icmp -i eth0 -j DROP                     |
| Block access to a specific MAC address | iptables -A INPUT -m mac --mac-source 00:00:00:00:00:00 -j DROP |
| Block connection on network interface  | iptables -A INPUT -i eth0 -s xxx.xxx.xxx.xxx -j DROP            |
| Disable outgoing mails                 | iptables -A OUTPUT -p tcp --dports 25,465,587 -j REJECT         |

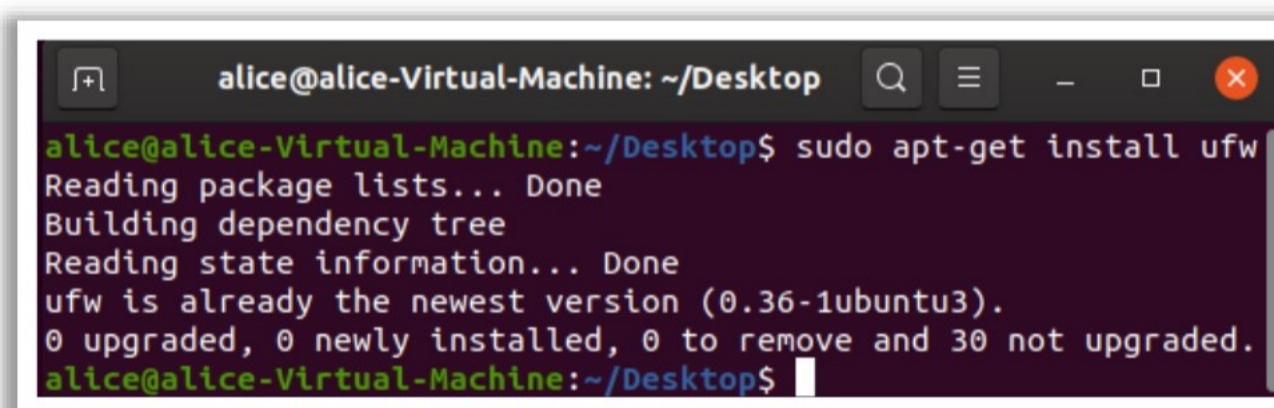
Table 6.5: Other iptables commands for various tasks

## UFW

UFW (uncomplicated firewall) is an interface to iptables. For beginners, it is difficult to use iptables for configuring a firewall. UFW can help them by simplifying the process of configuring a firewall to make the system secure in the network. Enable UFW to protect unusual traffic

### Steps to Set Up a Firewall with UFW

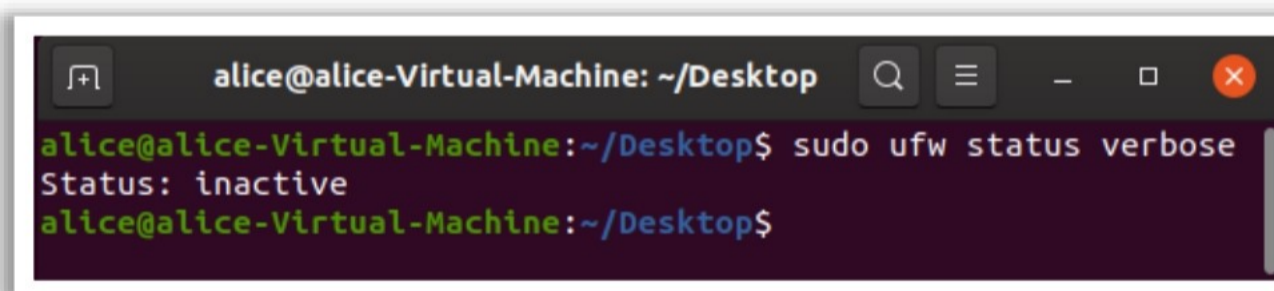
- Install UFW using the `sudo apt-get install ufw` command.



```
alice@alice-Virtual-Machine: ~/Desktop
alice@alice-Virtual-Machine:~/Desktop$ sudo apt-get install ufw
Reading package lists... Done
Building dependency tree
Reading state information... Done
ufw is already the newest version (0.36-1ubuntu3).
0 upgraded, 0 newly installed, 0 to remove and 30 not upgraded.
alice@alice-Virtual-Machine:~/Desktop$
```

Figure 6.55: Installing UFW

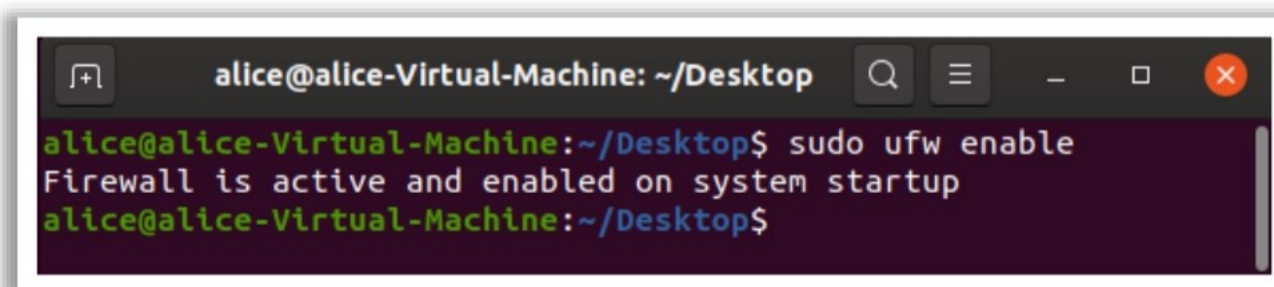
- Check the status of UFW using the `sudo ufw status verbose` command. The output will be active or inactive. The default status of UFW is disabled.



```
alice@alice-Virtual-Machine: ~/Desktop
alice@alice-Virtual-Machine:~/Desktop$ sudo ufw status verbose
Status: inactive
alice@alice-Virtual-Machine:~/Desktop$
```

Figure 6.56: Checking status of UFW

- Enable UFW using the `sudo ufw enable` command.



```
alice@alice-Virtual-Machine: ~/Desktop
alice@alice-Virtual-Machine:~/Desktop$ sudo ufw enable
Firewall is active and enabled on system startup
alice@alice-Virtual-Machine:~/Desktop$
```

Figure 6.57: Enabling UFW

- Set default policies using the following commands.

```
sudo ufw default deny incoming
sudo ufw default allow outgoing
```

### Add UFW Rules

There are two ways to add rules: denoting the port number and using the service name.

A few examples and corresponding commands are discussed below.

- Allow both incoming and outgoing connections on port 22 for SSH.

```
sudo ufw allow ssh
```

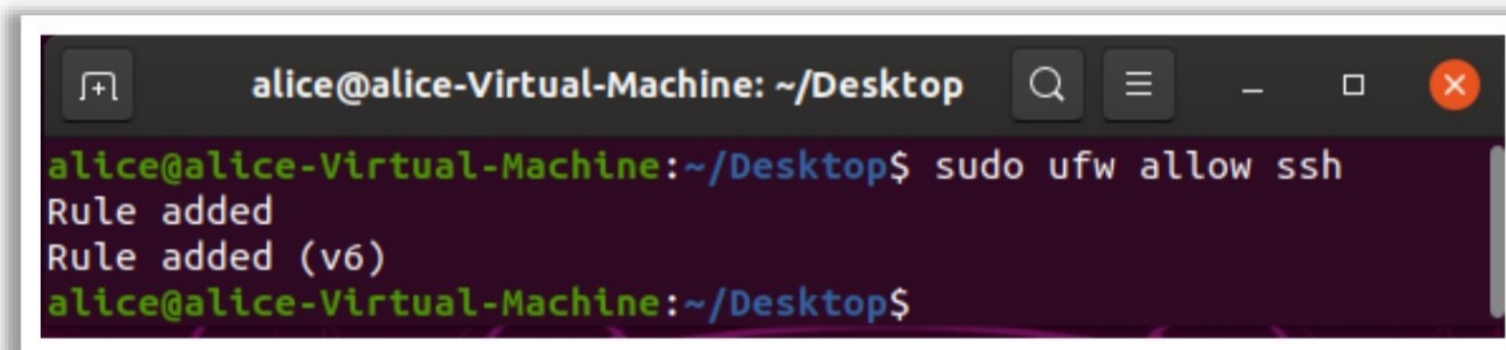


Figure 6.58: Adding UFW rules

(or)

```
sudo ufw allow 2000
```

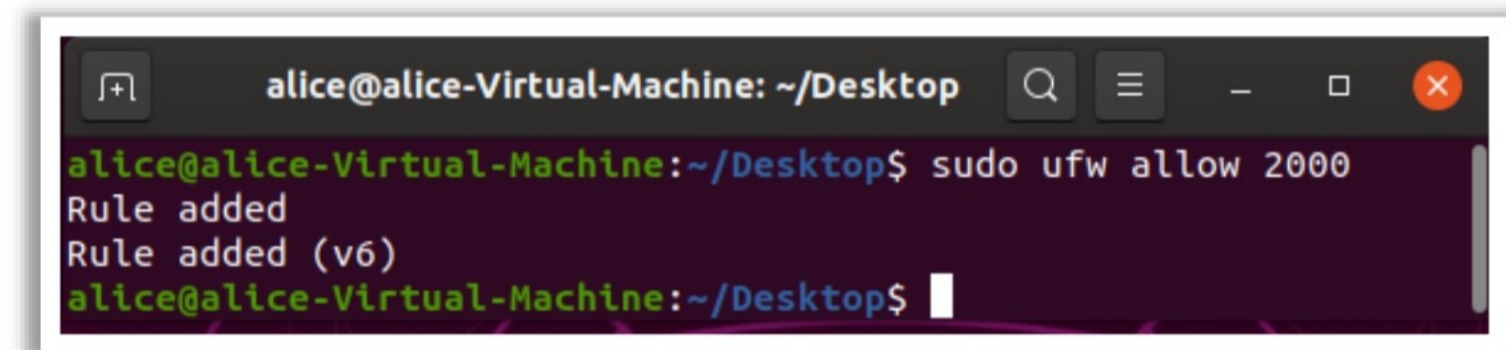


Figure 6.59: Adding UFW rules

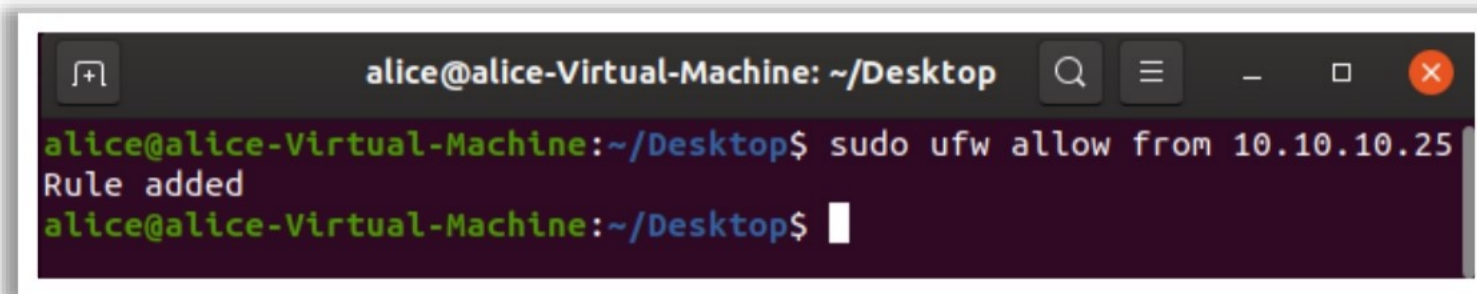
- Deny traffic on a specific port.  

```
sudo ufw deny 22.
```
- Allow packets based on TCP or UDP.

```
sudo ufw allow 80/tcp
sudo ufw allow http/tcp
sudo ufw allow 1725/udp
```

- Allow connections from an IP address.

```
sudo ufw allow from 10.10.10.25
```

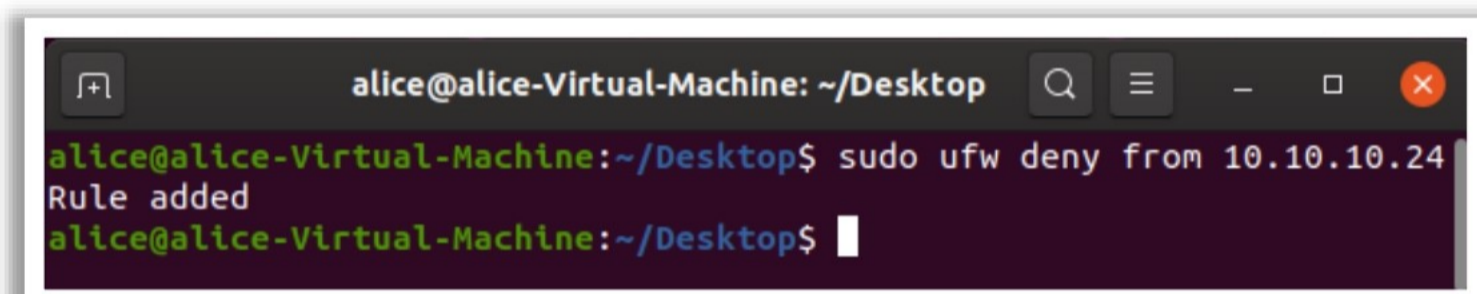


```
alice@alice-Virtual-Machine: ~/Desktop
alice@alice-Virtual-Machine:~/Desktop$ sudo ufw allow from 10.10.10.25
Rule added
alice@alice-Virtual-Machine:~/Desktop$
```

Figure 6.60: Allowing connections

- Deny connections from an IP address

```
Sudo ufw deny from 10.10.10.24.
```



```
alice@alice-Virtual-Machine: ~/Desktop
alice@alice-Virtual-Machine:~/Desktop$ sudo ufw deny from 10.10.10.24
Rule added
alice@alice-Virtual-Machine:~/Desktop$
```

Figure 6.61: Denying connections

- Allow connections from a specific subnet.

```
sudo ufw allow from 198.51.100.0/24
```

- Allow a specific IP address/port combination.

```
sudo ufw allow from 198.51.100.0 to any port 22 proto tcp
```

- When more advanced or specific rules need to be added/removed:

- Add the rules to the `/etc/ufw/before.rules` (`before6.rules` for IPv6) file to execute the rules.
- There exists `after.rule` and an `after6.rule` files to add any rules that would need to be added after UFW runs the command-line-added rules.
- An additional configuration file that is located at `/etc/default/ufw` allows the user to disable or enable IPv6, to set default rules, and set UFW to manage built-in firewall chains.

## Remove UFW Rules

Delete rules using port number or service name. Use `delete` in the command while removing a rule. For example, the command to delete allowing HTTP traffic from port number 80 is `sudo ufw delete allow 80`.

## TCP Wrappers



- TCP Wrappers or TCPD is a host-based networking access control list (ACL) system that provides firewall services by monitoring network traffic
- TCP Wrappers authorizes the entities to support the connection over the network
- It allows the entities according to **/etc/hosts.allow** rule and denies the connection requests based on **/etc/hosts.deny** rule
- A given service is said to be TCP wrapped only when it gives output for `ldd [/path/to/binary] | grep libwrap` command

```
root@alice-Virtual-Machine: /home/alice/Desktop
root@alice-Virtual-Machine:/home/alice/Desktop# ldd $(which sshd) | grep libwrap
libwrap.so.0 => /lib/x86_64-linux-gnu/libwrap.so.0 (0x00007fae9df7f000)
root@alice-Virtual-Machine:/home/alice/Desktop# ldd $(which ufw) | grep libwrap
root@alice-Virtual-Machine:/home/alice/Desktop#
```

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

## TCP Wrappers

TCP Wrappers or TCPD is a host-based access control list that provides firewall services for UNIX servers by monitoring incoming packets. It is not an alternate to a properly configured pf firewall or netfilter as the firewall cannot check encrypted connections. However, TCPD provides additional security. It checks and allows data packets to pass through only when the external system or host has the authorization to connect to the network. It supports common services POP3, FTP, SSHD, telnet and R services.

### Advantages

- TCPD has been widely deployed for host-level security, ensuring good IP addresses are allowed (whitelisted) or are bad ones are blocked (blacklisted).
- The connections monitored by TCPD are reported by the syslog feature (logging).
- It supports a simple form of access control that is based on pattern matching. When a pattern matches, you can execute shell commands or scripts.
- It works at the application Layer (Layer 7), allowing filters to be applied even when encryption is used (e.g., HTTPS).
- It protects against spoofing.

### Note:

- Use both firewall and TCPD to fight against attackers.
- Do not configure TCPD on the firewall host.
- Keep TCPD on all UNIX/Linux/BSD workstations.

- Do not use NIS (YP) netgroups in TCPD rules.
- Keep TCPD Wrappers behind a firewall system.

### **hosts.allow and hosts.deny**

When a network request reaches the server, TCPD uses `hosts.allow` and `hosts.deny` to decide whether the client should be allowed to use the given service. By default, both files are empty. Therefore, everything is allowed through the TCPD layer, and the system relies on the firewall for full protection. But ensure both the files exist by entering the command `# ls -l /etc/hosts.allow /etc/hosts.deny`. Only the first rule will be considered if two rules apply to the same service.

Not all network services support the use of TCPD. To know whether a given service supports TCPD, user the `#ldd /path/to/binary | grep libwrap` command. If any output is received, the service can be TCP-Wrapped. For example, `sshd` and `vsftpd`.

Steps to use TCPD to restrict access to services are listed below:

- Ensure you add a new line by pressing Enter after the last non-empty line in `/etc/hosts.allow` and `/etc/hosts.deny`.
- To allow SSH and FTP access only to 192.168.0.201 and localhost and deny all others, add the flowing lines in `/etc/hosts.deny`.

```
sshd, vsftpd : ALL
ALL : ALL
```

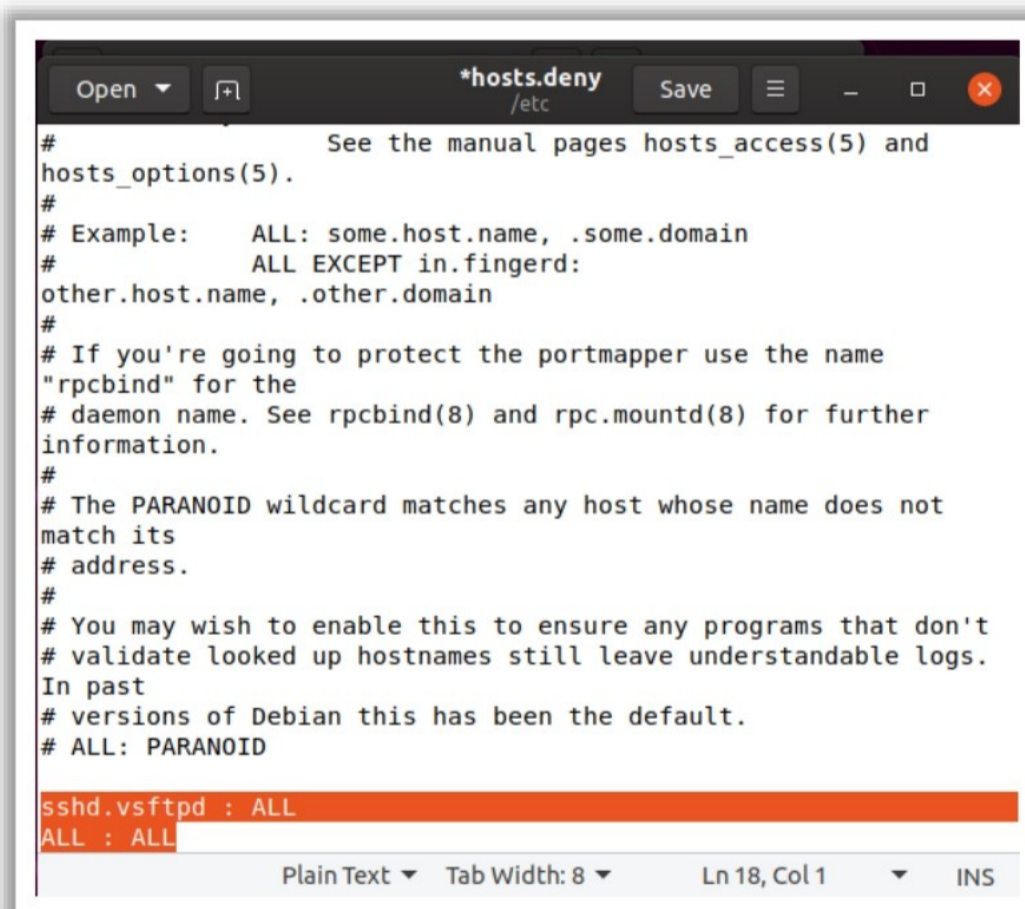
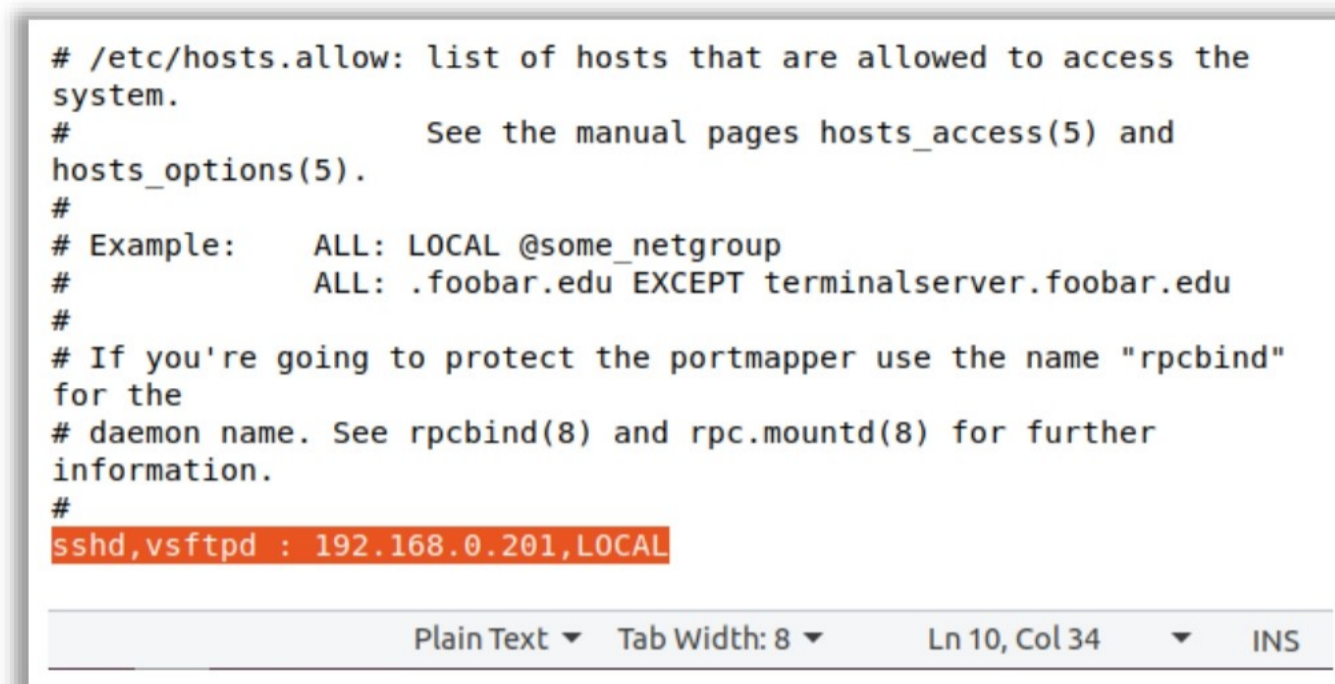


Figure 6.62: Adding in `/etc/hosts.deny`

- Add the following line in `/etc/hosts.allow`.

`sshd, vsftpd : 192.168.0.102,LOCAL`



```
/etc/hosts.allow: list of hosts that are allowed to access the
system.
See the manual pages hosts_access(5) and
hosts_options(5).
#
Example: ALL: LOCAL @some_netgroup
ALL: .foobar.edu EXCEPT terminalserver.foobar.edu
#
If you're going to protect the portmapper use the name "rpcbind"
for the
daemon name. See rpcbind(8) and rpc.mountd(8) for further
information.
#
sshd,vsftpd : 192.168.0.201,LOCAL
```

Figure 6.63: Adding in `/etc/hosts.allow`

- The changes take effect immediately without a restart.
- See the effect of removing the word `LOCAL` from the last line: the FTP server will become unavailable for the localhost.
- To allow all services to host where the name is `example.com`, add the following line in `hosts.allow`.

`ALL : .example.com`

- To deny access to `vsftpd` to machines on `10.0.1.0/24`, add the following line in `hosts.deny`.

`Vsftpd : 10.0.1.`

# Monitor Open Ports and Services



- Monitoring **open ports** and the **associated services** running on each port helps understand the associated vulnerabilities and hidden security risks
- Use **netstat -tulpn** or **ss -tulpn** command displays all open ports and associated services

```
root@alice-Virtual-Machine: /home/alice
root@alice-Virtual-Machine:/home/alice# netstat -tulpn
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address Foreign Address State Timer
tcp 0 0 127.0.0.1:53 0.0.0.0:* LISTEN off (0.00/0/0)
tcp 0 0 0.0.0.0:25 0.0.0.0:* LISTEN off (0.00/0/0)
tcp 0 0 127.0.0.1:631 0.0.0.0:* LISTEN off (0.00/0/0)
tcp6 0 0 :::1:631 :::* LISTEN off (0.00/0/0)
tcp6 0 0 :::25 :::* LISTEN off (0.00/0/0)
udp 0 0 0.0.0.0:5353 0.0.0.0:* off (0.00/0/0)
udp 0 0 0.0.0.0:59586 0.0.0.0:* off (0.00/0/0)
udp 0 0 127.0.0.1:53 0.0.0.0:* off (0.00/0/0)
udp6 0 0 :::5353 :::* off (0.00/0/0)
udp6 0 0 :::51311 :::* off (0.00/0/0)
root@alice-Virtual-Machine:/home/alice#
```

```
root@alice-Virtual-Machine: /home/alice
root@alice-Virtual-Machine:/home/alice# ss -tulpn
Netid State Recv-Q Send-Q Local Address:Port Peer Address:Port
Process
udp UNCONN 0 0 0.0.0.0:5353 0.0.0.0:*
users:(("avahi-daemon",pid=447,fd=12))
udp UNCONN 0 0 0.0.0.0:59586 0.0.0.0:*
users:(("avahi-daemon",pid=447,fd=14))
udp UNCONN 0 0 127.0.0.1:53 0.0.0.0:*
users:(("systemd-resolve",pid=406,fd=13))
udp UNCONN 0 0 0.0.0.0:631 0.0.0.0:*
users:(("cups-browsed",pid=2539,fd=7))
udp UNCONN 0 0 [::]:5353 [::]:*
users:(("avahi-daemon",pid=447,fd=13))
udp UNCONN 0 0 [::]:51311 [::]:*
users:(("avahi-daemon",pid=447,fd=15))
tcp LISTEN 0 4096 127.0.0.1:53 0.0.0.0:*
users:(("systemd-resolve",pid=406,fd=14))
tcp LISTEN 0 128 127.0.0.1:631 0.0.0.0:*
users:(("cupsd",pid=2537,fd=7))
tcp LISTEN 0 100 0.0.0.0:25 0.0.0.0:*
users:(("master",pid=6676,fd=13))
tcp LISTEN 0 128 [::]:631 [::]:*
users:(("cupsd",pid=2537,fd=6))
tcp LISTEN 0 100 [::]:25 [::]:*
users:(("master",pid=6676,fd=14))
root@alice-Virtual-Machine:/home/alice#
```

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

## Monitor Open Ports and Services

Monitoring open ports and the associated services running on each port help understand the related vulnerabilities and hidden security risks. Use the following commands to display all open ports and related services.

Monitoring open ports using the `netstat` command (for Ubuntu):

- Install `netstat` using the `sudo apt install net-tools` command.
- List all ports available on a server using the `netstat -tulpn` command.

```
root@alice-Virtual-Machine: /home/alice
root@alice-Virtual-Machine:/home/alice# netstat -tulpn
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address Foreign Address State Timer
tcp 0 0 127.0.0.1:53 0.0.0.0:* LISTEN off (0.00/0/0)
tcp 0 0 0.0.0.0:25 0.0.0.0:* LISTEN off (0.00/0/0)
tcp 0 0 127.0.0.1:631 0.0.0.0:* LISTEN off (0.00/0/0)
tcp6 0 0 :::1:631 :::* LISTEN off (0.00/0/0)
tcp6 0 0 :::25 :::* LISTEN off (0.00/0/0)
udp 0 0 0.0.0.0:5353 0.0.0.0:* off (0.00/0/0)
udp 0 0 0.0.0.0:59586 0.0.0.0:* off (0.00/0/0)
udp 0 0 127.0.0.1:53 0.0.0.0:* off (0.00/0/0)
udp6 0 0 :::5353 :::* off (0.00/0/0)
udp6 0 0 :::51311 :::* off (0.00/0/0)
root@alice-Virtual-Machine:/home/alice#
```

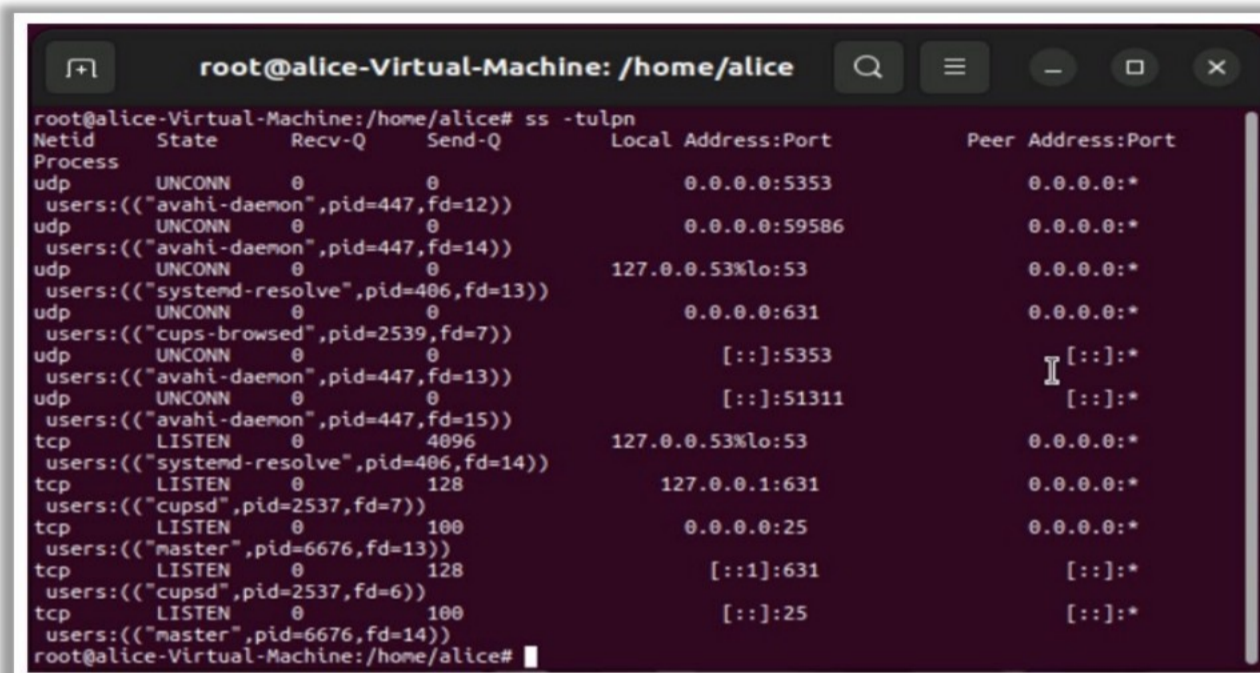
Figure 6.64: netstat -tulpn command

- For detail command options, use the list below:
  - t to show TCP ports

- `-u` to show UDP ports
- `-n` to show numerical addresses instead of resolving hosts
- `-l` to show only listening ports
- `-p` to show the PID and name of the listener's process

Monitoring open ports using the `ss` command (for Linux):

- Run the `ss -tulpn` command.



```
root@alice-Virtual-Machine: /home/alice# ss -tulpn
Netid State Recv-Q Send-Q Local Address:Port Peer Address:Port
Process
udp UNCONN 0 0 0.0.0.0:5353 0.0.0.0:*
users:((("avahi-daemon",pid=447,fd=12))
udp UNCONN 0 0 0.0.0.0:59586 0.0.0.0:*
users:((("avahi-daemon",pid=447,fd=14))
udp UNCONN 0 0 127.0.0.53%lo:53 0.0.0.0:*
users:((("systemd-resolve",pid=406,fd=13))
udp UNCONN 0 0 0.0.0.0:631 0.0.0.0:*
users:((("cups-browsed",pid=2539,fd=7))
udp UNCONN 0 0 [::]:5353 [::]:*
users:((("avahi-daemon",pid=447,fd=13))
udp UNCONN 0 0 [::]:51311 [::]:*
users:((("avahi-daemon",pid=447,fd=15))
tcp LISTEN 0 4096 127.0.0.53%lo:53 0.0.0.0:*
users:((("systemd-resolve",pid=406,fd=14))
tcp LISTEN 0 128 127.0.0.1:631 0.0.0.0:*
users:((("cupsd",pid=2537,fd=7))
tcp LISTEN 0 100 0.0.0.0:25 0.0.0.0:*
users:((("master",pid=6676,fd=13))
tcp LISTEN 0 128 [::]:631 [::]:*
users:((("cupsd",pid=2537,fd=6))
tcp LISTEN 0 100 [::]:25 [::]:*
users:((("master",pid=6676,fd=14))
root@alice-Virtual-Machine: /home/alice#
```

Figure 6.65: `ss -tulpn` command

Use the same command options the `netstat` command uses.

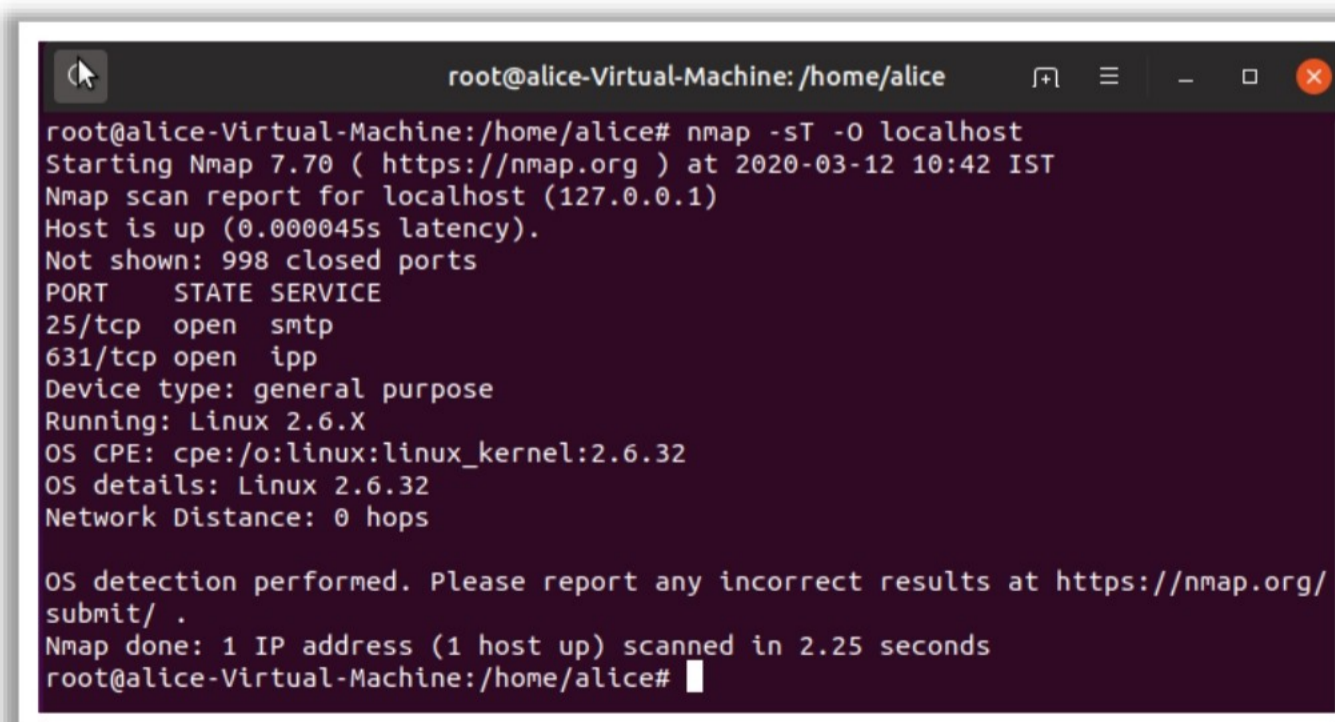
Monitoring open ports using the `lsof` command:

Run the `lsof` command with the options `sudo lsof -nP -iTCP -sTCP:LISTEN`.

Monitoring open ports using the Nmap utility:

- Run the below command to scan for open TCP ports.

```
nmap -sT -O localhost
```



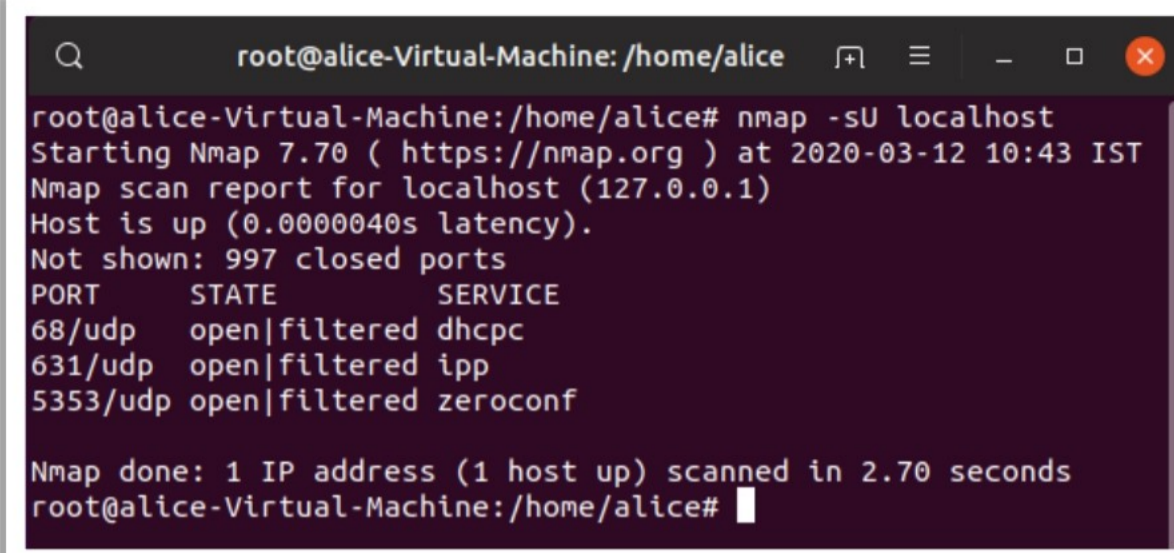
```
root@alice-Virtual-Machine: /home/alice# nmap -sT -O localhost
Starting Nmap 7.70 (https://nmap.org) at 2020-03-12 10:42 IST
Nmap scan report for localhost (127.0.0.1)
Host is up (0.000045s latency).
Not shown: 998 closed ports
PORT STATE SERVICE
25/tcp open smtp
631/tcp open ipp
Device type: general purpose
Running: Linux 2.6.X
OS CPE: cpe:/o:linux:linux_kernel:2.6.32
OS details: Linux 2.6.32
Network Distance: 0 hops

OS detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 2.25 seconds
root@alice-Virtual-Machine: /home/alice#
```

Figure 6.66: TCP port scan

- Run the below command to scan for open UDP ports.

```
nmap -sU localhost
```

A terminal window titled 'root@alice-Virtual-Machine: /home/alice' showing the output of the command 'nmap -sU localhost'. The output indicates that the host is up and lists three open UDP ports: 68/udp (dhcpc), 631/udp (ipp), and 5353/udp (zeroconf).

```
root@alice-Virtual-Machine:/home/alice# nmap -sU localhost
Starting Nmap 7.70 (https://nmap.org) at 2020-03-12 10:43 IST
Nmap scan report for localhost (127.0.0.1)
Host is up (0.0000040s latency).
Not shown: 997 closed ports
PORT STATE SERVICE
68/udp open|filtered dhcpc
631/udp open|filtered ipp
5353/udp open|filtered zeroconf

Nmap done: 1 IP address (1 host up) scanned in 2.70 seconds
root@alice-Virtual-Machine:/home/alice#
```

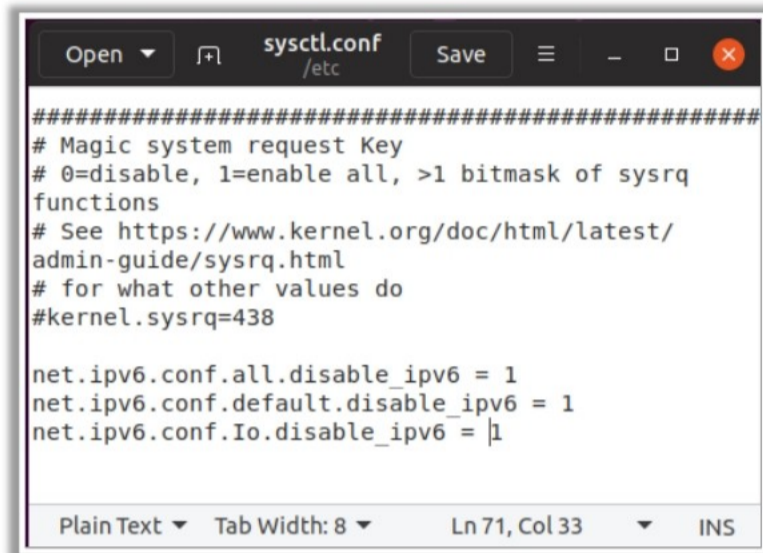
Figure 6.67: UDP port scan

## Turn Off IPv6 if Not In Use



Running misconfigured IPv6 leaves the system exposed to various attacks

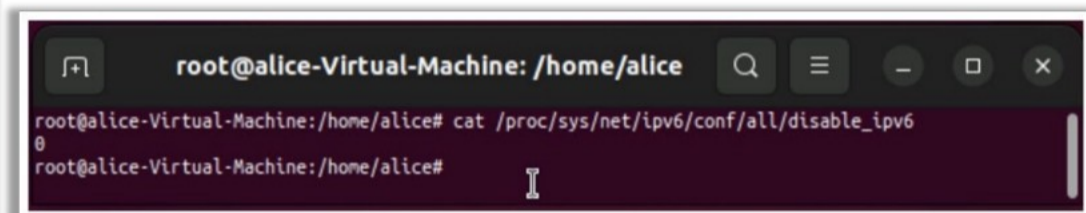
It is recommended to **switch off IPv6** if not in use



```

Magic system request Key
0=disable, 1=enable all, >1 bitmask of sysrq
functions
See https://www.kernel.org/doc/html/latest/
admin-guide/sysrq.html
for what other values do
#kernel.sysrq=438

net.ipv6.conf.all.disable_ipv6 = 1
net.ipv6.conf.default.disable_ipv6 = 1
net.ipv6.conf.lo.disable_ipv6 = 1
```



```
root@alice-Virtual-Machine: /home/alice
root@alice-Virtual-Machine:/home/alice# cat /proc/sys/net/ipv6/conf/all/disable_ipv6
1
root@alice-Virtual-Machine:/home/alice#
```

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

### Turn Off IPv6 if Not In Use

Although IPv6 incorporates some security improvements over IPv4, running misconfigured IPv6 leaves the system exposed to various attacks. If IPv6 is enabled by default, ensure it is turned off until properly configured or when it is not in use.

Steps to disable IPv6 protocol on a Debian-based system are listed below:

- Enter the command `sudo nano /etc/sysctl.conf` in the terminal.
- Add the following lines at the bottom of the file.

```
net.ipv6.conf.all.disable_ipv6 = 1
net.ipv6.conf.default.disable_ipv6 = 1
net.ipv6.conf.lo.disable_ipv6 = 1
```
- Save and close the file and restart the system.

Steps to disable IPv6 using GRUB are listed below:

- Edit `/etc/default/grub` to configure the GRUB to pass kernel parameters at boot time.

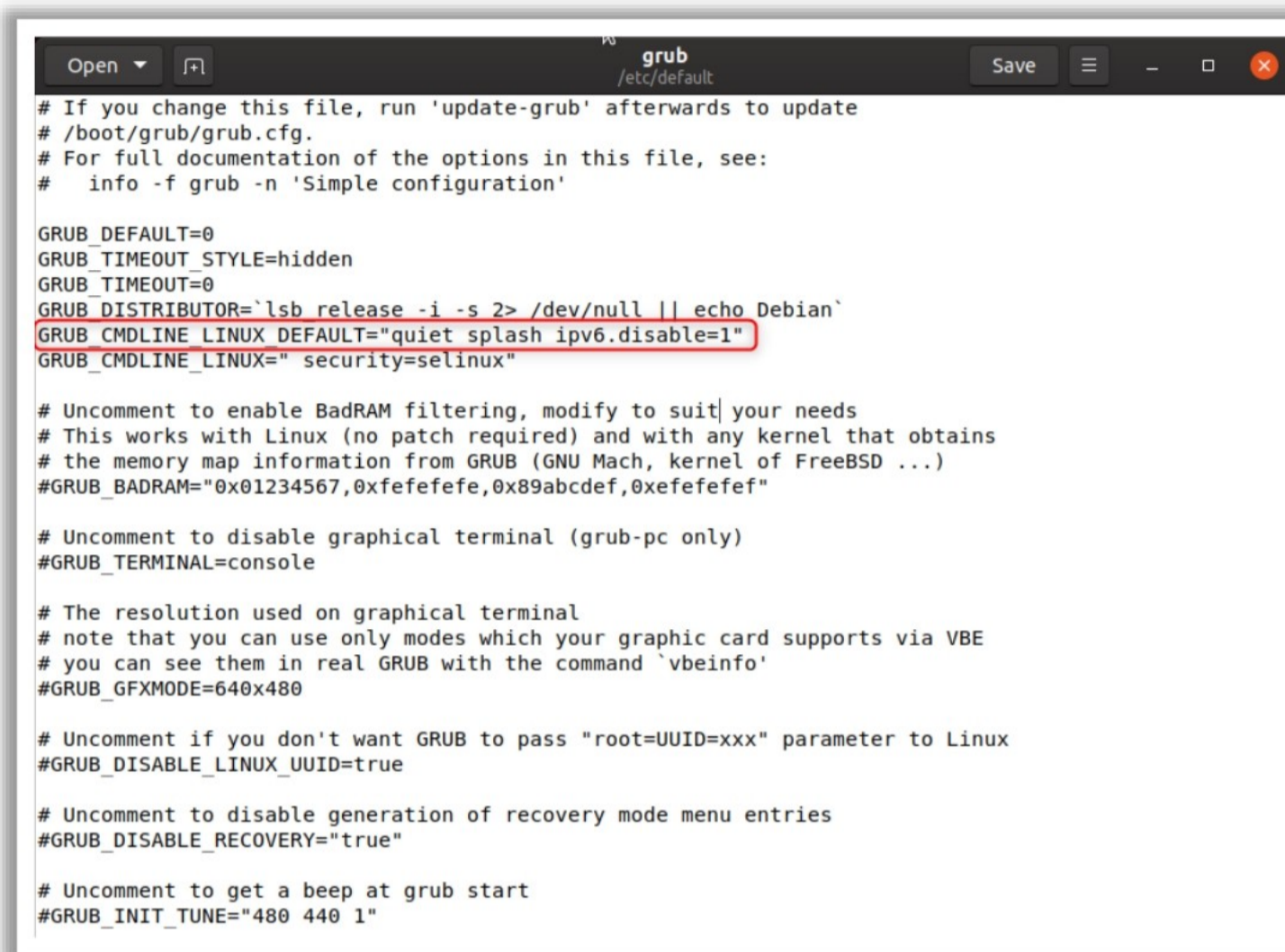


Figure 6.68: Editing `/etc/default/grub`

- Modify `GRUB_CMDLINE_LINUX_DEFAULT` and `GRUB_CMDLINE_LINUX` to disable IPv6 on boot.

```
GRUB_CMDLINE_LINUX_DEFAULT="quiet splash ipv6.disable=1"
```

```
GRUB_CMDLINE_LINUX="ipv6.disable=1"
```

- Save the file and run the `update-grub` command.

```
sudo update-grub
```

- Reboot the system.

Steps to disable the IPv6 protocol on a Red Hat-based system are listed below:

- Enter the following commands.

```
sysctl -w net.ipv6.conf.all.disable_ipv6=1
```

```
sysctl -w net.ipv6.conf.default.disable_ipv6=1
```

Note:

- When using X Forwarding through ssh, disabling IPv6 can break the system. To fix this, follow the steps listed below:
  - Open the `/etc/ssh/sshd_config` file.
  - Change the `#AddressFamily any` to `AddressFamily inet`.

- Save that file and restart `sshd`.
- When using postfix, you may encounter issues in starting the service. To fix this, follow the steps listed below:
  - Use an IPv4 loopback.
  - Open the `/etc/postfix/main.cf` file.
  - Comment out the `localhost` line, and add the IPv4 loopback.

```
#inet_interfaces = localhost
inet_interfaces = 127.0.0.1
```

## Secure SSH Login Root Login



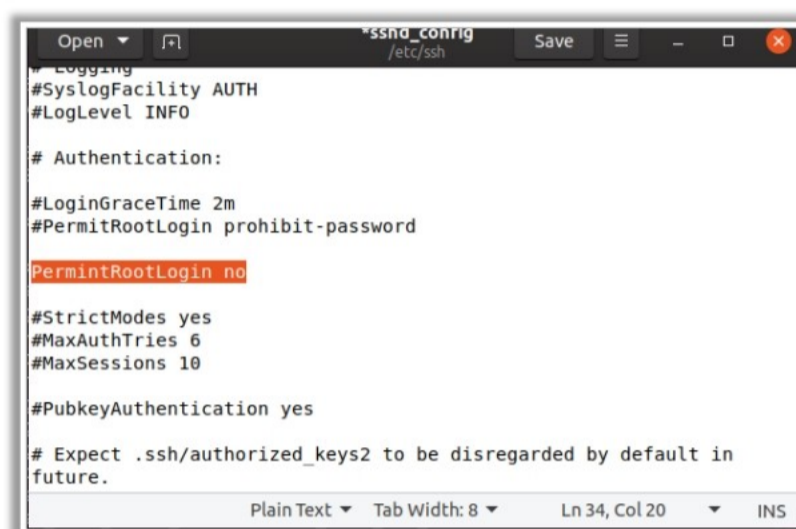
- SSH enables secure **data transfer communication** between client and server
- Attackers can use SSH to attack the operating system

### 1 Disable SSH Root login

To prevent users from logging in directly as root:

**Disable** SSH root login using any of the following methods

- Disable **PermitRootLogin** in the `/etc/ssh/sshd_config` file
- Add the root user in **DenyUsers** list on `etc/ssh/sshd_config` file
- Deny the root user via `/etc/ssh/sshd.deny` file



```
Open *sshd_config /etc/ssh Save
Logging
#SyslogFacility AUTH
#LogLevel INFO

Authentication:

#LoginGraceTime 2m
#PermitRootLogin prohibit-password
PermitRootLogin no

#StrictModes yes
#MaxAuthTries 6
#MaxSessions 10

#PubkeyAuthentication yes

Expect .ssh/authorized_keys2 to be disregarded by default in
future.

Plain Text Tab Width: 8 Ln 34, Col 20 INS
```

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

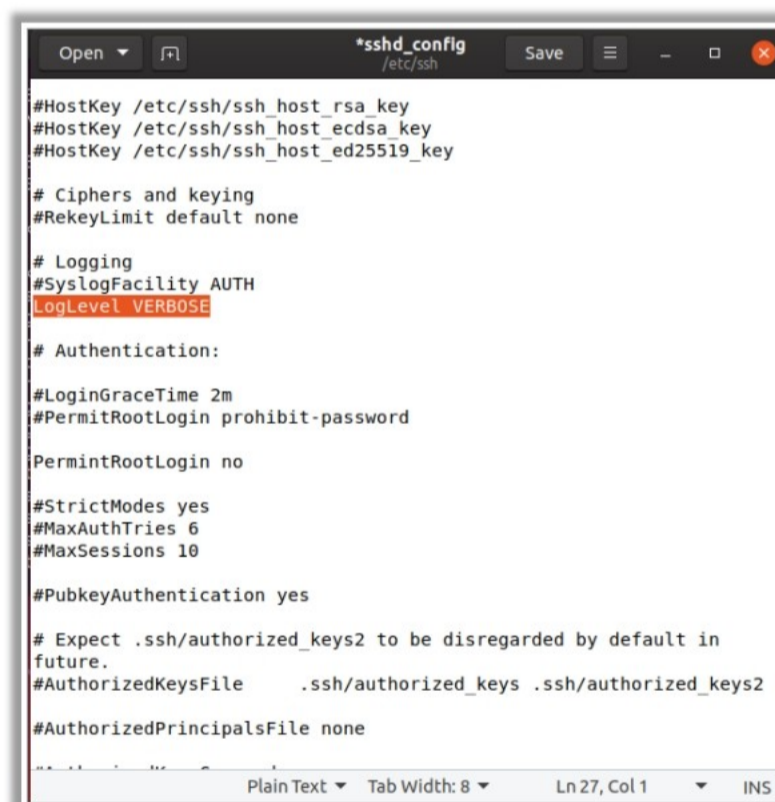
## Secure SSH Login Root Login (Cont'd)



### 2 Enable Detailed Logging for SSH

To enable detailed login:

- Edit the `/etc/ssh/sshd_config` file
- Change the LogLevel parameter value to **verbose**



```
Open *sshd_config /etc/ssh Save
#HostKey /etc/ssh/ssh_host_rsa_key
#HostKey /etc/ssh/ssh_host_ecdsa_key
#HostKey /etc/ssh/ssh_host_ed25519_key

Ciphers and keying
#RekeyLimit default none

Logging
#SyslogFacility AUTH
LogLevel VERBOSE

Authentication:

#LoginGraceTime 2m
#PermitRootLogin prohibit-password
PermitRootLogin no

#StrictModes yes
#MaxAuthTries 6
#MaxSessions 10

#PubkeyAuthentication yes

Expect .ssh/authorized_keys2 to be disregarded by default in
future.
#AuthorizedKeysFile .ssh/authorized_keys .ssh/authorized_keys2

#AuthorizedPrincipalsFile none

Plain Text Tab Width: 8 Ln 27, Col 1 INS
```

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

## Secure SSH Login Root Login

By default, Linux systems permit remote access using the highly privileged root user account. Hackers can try to brute-force user password and access the system by using the enabled ssh root access. To protect against this, disable root login through SSH. Before doing that, create a separate user account (use `adduser` command) for regular use of the system and ensure that

the new account is included in sudoers, allowing it to execute commands just as the super user (root account) when using the sudo command.

**Steps to disable SSH root login are listed below:**

- Open the main ssh configuration file `/etc/ssh/sshd_config`.  
`# gedit /etc/ssh/sshd_config`
- Search for the line in the file - `#PermitRootLogin no`
- Remove the '#' from the beginning of the line.  
`PermitRootLogin no`



Figure 6.69: Disabling SSH root login

- Restart the SSH daemon service by using one of the below commands.  
`# /etc/init.d/sshd restart`  
`# systemctl restart sshd`  
`# service sshd restart`
- Try to login with the root user. "Permission Denied" error will be shown.

Secure SSH configuration settings in `/etc/ssh/sshd_config`:

- `PermitRootLogin no`
- `AllowUsers [username]`
- `IgnoreRhosts yes`
- `HostbasedAuthentication no`
- `PermitEmptyPasswords no`
- `X11Forwarding no`
- `MaxAuthTries 5`
- `Ciphers aes128-ctr,aes192-ctr,aes256-ctr`
- `UsePAM yes`

- `ClientAliveInterval 900`
- `ClientAliveCountMax 0`
- Save the file and Exit.

### Steps to Enable Detailed Logging for SSH

- Edit the `/etc/ssh/sshd_config` file
- Change the **LogLevel** parameter value to verbose

## Setup Chroot SFTP



- By default, SFTP logged in users can browse other user's directories like SCP, SSH etc.
- You should create a **chroot directory** to avoid access to their SFTP home directory:
  - To create a chroot directory, execute `sudo mkdir /sftp/(name)` command and to give root user rights to it, Use `sudo chown root:root /sftp/` command
  - Create sftponly group by executing `sudo groupadd sftponly` command and add users using `sudo useradd -g sftponly -d /(name) -s /sbin/nologin (new user)`
  - Create a password for that account `sudo passwd (user)`
  - Execute the following commands to restrict the rights to access the home directory

```
sudo chown senthil:sftponly /sftp/ostechnix
sudo chmod 700 /sftp/ostechnix/
```
  - Edit `/etc/ssh/sshd_config` file by commenting out the following line by adding hash (#) in front of it:

```
#Subsystem sftp /usr/lib/openssh/sftp-server
```
  - Add `Subsystem sftp internal-sftp, Match group sftponly, ChrootDirectory, /sftp/ X11Forwarding no, AllowTcpForwarding no, ForceCommand internal-sftp` at the end of the file

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

## Setup Chroot SFTP

SFTP (SSH File Transfer Protocol or Secure File Transfer Protocol) provides file access, file transfer, and file management feature over any reliable data stream. By default, the users that can log in to the system through SSH, SFTP, and SCP can browse the entire file system, including the other user's directories. Configuring SFTP in the chroot environment can limit the users to access their home directory and adds an extra layer of security especially on systems with multiple users.

Steps to setup SFTP chroot jail environment that work for any modern Linux distribution are listed below:

- Create chrooted directory using the command `$ sudo mkdir /sftp`.
  - Apply root privileges to the created directory using the command `$ sudo chown root:root /sftp/`.
  - Create separate directories for each user under the created directory. For example, `/sftp/user1`, `/sftp/user2`, `/sftp/user3`, etc.

```
$ sudo mkdir /sftp/ostechnix
```
  - Users can save their data in this directory (`$HOME` directory) but cannot go beyond this directory.
- Create sftp group sftponly.

```
$sudo groupadd sftponly
```

  - Create SFTP users to access the sftponly chrooted directory and assign them to the group sftponly, setup their home directory as `/sftp/ostechnix` and the default shell as `/sbin/nologin`.

- Set a password for the new user using the `$ sudo passwd Alex` command.

- ```
$ sudo chmod 700 /sftp/ostechnix/
```

- Configure Chrooted SFTP.

- ```
$ sudo vi /etc/ssh/sshd config
```

- Find and comment out the following lines (i.e., add # in-front of them).

```
#Subsystem sftp /usr/libexec/openssh/sftp-server
```

- In Ubuntu 19.04 , find and comment the following line:

```
#Subsystem sftp /usr/lib/openssh/sftp-server
```

- Add the following lines at the end of the file:

```
Subsystem sftp internal-sftp
Match group sftponly
ChrootDirectory /sftp/
X11Forwarding no
AllowTcpForwarding no
ForceCommand internal-sftp
```

- Save and exit the file.

- Restart the ssh service to update the changes.

```
$ sudo systemctl restart sshd
```

- Try to SSH to this system from any other system on the network using the sftp user (i.e., Alex in our case).

```
$ ssh alex@192.168.55.1
```

- The reply will be the following error message.

```
alex@192.168.55.1's password:
This service allows sftp connections only.
Connection to 192.168.55.1 closed.
```

Here, 192.168.43.2 is the remote system's IP address where the SFTP is configured. You can only access the remote system using sftp.

```
$ sftp alex@192.168.55.1
alex@192.168.55.1's password:
Connected to 192.168.55.1.
sftp>
```

- **Add** Subsystem `sftp internal-sftp`, Match group `sftponly`, ChrootDirectory, `/sftp/` X11Forwarding `no`, AllowTcpForwarding `no`, ForceCommand `internal-sftp` **at the end of the file.**

- **Testing the Configuration.**

- Log in to the remote machine via SFTP using the credentials of the chrooted user.
- Open an SFTP connection using the `sftp` command followed by the remote server username and the server IP address or domain name:

```
Sftp alex@192.168.55.1
```

- Enter the password and remote server will show a confirmation message and the `sftp>` prompt:

```
alex@192.168.55.1's password:
sftp>
```

- Run the `pwd` command to know everything is working fine (it will return `/`).

```
sftp> pwd
Remote working directory: /
```

## Linux Hardening Checklist: Network Security and Remote Access



Implement firewall and other access controls to restrict connections to services running on the host to authorized users of the service



**Disable** IP forwarding, send packet redirects, source routed packet acceptance, and ICMP redirect acceptance



**Enable** IP ignore broadcast requests, bad error message protection, TCP/SYN cookies



Ensure that SSH server configuration Protocol version is set to 2, **LogLevel** is set to INFO, and **PermitEmptyPasswords** is set to **No**



Disable **root login** over SSH

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

### Linux Hardening Checklist: Network Security and Remote Access

- Restrict connections to the services running on the host to authorized users of the service using firewalls (for example, `iptables`) and other access control techniques.
- Set the following kernel parameters in `/etc/sysctl.conf`:
  - Disable: IP forwarding, send packet redirects, source-routed packet acceptance, and ICMP redirect acceptance.
  - Enable: Ignore broadcast requests, bad error message protection, and TCP/SYN cookies.
- Ensure whether SSH server configuration is set by default (`Protocol version=2`, `LogLevel=INFO`, and `PermitEmptyPasswords=No`) as setting them to other values impacts the SSH server's security.
- Disable root login over SSH. Instead, the users should authenticate with their account and use `su` or `sudo` command when required.
- Deploy an intrusion prevention system (IPS) to block remote systems generating authentication failures and to combat brute-force password attempts.



---

### LO#06: Discuss various Linux security tools and frameworks

---

Copyright © by **EC-Council**. All Rights Reserved. Reproduction is Strictly Prohibited.

## **LO#06: Discuss Various Linux Security Tools and Frameworks**

The objective of this section is to help you understand the various Linux security tools and frameworks.

## Security Auditing and System Hardening using Lynis



- Lynis performs an extensive **health scan** of the systems to support **system hardening** and **compliance testing**

### Lynis is used for:

- 1 Security auditing
- 2 Compliance testing (e.g. PCI, HIPAA, SOX)
- 3 Penetration testing
- 4 Vulnerability detection
- 5 System hardening

Source: <https://cisofy.com/>

Copyright © by **EC-Council**. All Rights Reserved. Reproduction is Strictly Prohibited.

## Security Auditing and System Hardening using Lynis

### Lynis

Source: [www.cisofy.com](http://www.cisofy.com)

A reliable and effective open-source (GPL license) security tool, Lynis can perform an extensive health scan of systems to support system hardening and compliance testing.

Lynis' scanning is modular and opportunistic in which it only uses and tests the components (system tools and its libraries) that it can find, requiring no installation of other tools. Therefore, it can keep the system clean.

Figure 6.70: Security scan with Lynis

## Turn On AppArmor



AppArmor is a **Linux kernel security** module that allows the network defender to restrict programs' capabilities through per-program profiles



It is **Mandatory Access Control** (MAC) system implemented upon Linux Security Modules (LSM)



It allows the network defender apply MAC to a limit access to set of resources

```
root@alice-Virtual-Machine: /
root@alice-Virtual-Machine:/#
root@alice-Virtual-Machine:/# aa-enforce /usr/bin/ping
Setting /usr/bin/ping to enforce mode.
root@alice-Virtual-Machine:/# apparmor_status
apparmor module is loaded.
21 profiles are loaded.
20 profiles are in enforce mode.
 /snap/core/8268/usr/lib/snapd/snap-confine
 /snap/core/8268/usr/lib/snapd/snap-confine//mount-namespace-capture-helper
 /usr/bin/man
 /usr/bin/ping
 /usr/lib/snapd/snap-confine
 /usr/lib/snapd/snap-confine//mount-namespace-capture-helper
 libreoffice-soffice//gpg
 man_filter
 man_groff
 ping
 snap-update-ns.amass
 snap-update-ns.core
 snap-update-ns.gnome-calculator
 snap-update-ns.gnome-characters
 snap-update-ns.gnome-logs
 snap.amass.amass
 snap.core.hook.configure
 snap.gnome-calculator.gnome-calculator
 snap.gnome-characters.gnome-characters
 snap.gnome-logs.gnome-logs
1 profiles are in complain mode.
 libreoffice-soffice
0 processes have profiles defined.
0 processes are in enforce mode.
0 processes are in complain mode.
0 processes are unconfined but have a profile defined.
root@alice-Virtual-Machine:/#
```

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

## Turn On AppArmor

AppArmor is a mandatory access control (MAC) system that is considered to be an enhancement to the Linux Security Modules (LSM) kernel. It is utilized by the network defender to restrict programs to a limited set of resources. AppArmor provides fine-grained permission and safeguards the OS and applications against internal and external threats.

If there is any security threat in the Linux kernel, AppArmor prevents access to the vulnerable critical paths. There are various default security policies in AppArmor. The security policies of AppArmor regulate application access to the system resources.

AppArmor uses the profiles of applications, which are the text files located in `/etc/apparmor.d/`. Few packages installed their profiles and the rest of the profiles are found in `apparmor-profiles` package.

The user can install `apparmor-profiles` package using the following command:

```
sudo apt install apparmor-profiles
```

If AppArmor security module is not enabled, then the user can enable it by executing the following command:

```
security=apparmor
```

## Turn On Security-Enhanced Linux (SELinux)



- SELinux (security-enhanced Linux) is a **kernel level MAC** (Mandatory Access Control) implementation for Linux
- It uses the LSM framework to implement MAC
- Protects against damages caused by unknown or unpatched exploits
- Enables fine-grained access control in the system
- Allows implementation of a customized level of security

```
[root@node1 ~]# sestatus
SELinux status: enabled
SELinuxfs mount: /selinux
Current mode: enforcing
Mode from config file: enforcing
Policy version: 24
Policy from config file: targeted

[root@node1 ~]# cat /etc/se
securetty security/ services sestatus.conf
[root@node1 ~]# cat /etc/se
securetty security/ services sestatus.conf
[root@node1 ~]# cat /etc/selinux/
config restorecond.conf restorecond_user.conf senanage.conf targeted/
[root@node1 ~]# cat /etc/selinux/
config restorecond.conf restorecond_user.conf senanage.conf targeted/
[root@node1 ~]# cat /etc/selinux/config

This file controls the state of SELinux on the system.
SELINUX= can take one of these three values:
enforcing - SELinux security policy is enforced.
permissive - SELinux prints warnings instead of enforcing.
disabled - No SELinux policy is loaded.
SELINUX=enforcing
SELINUXTYPE= can take one of these two values:
targeted - Targeted processes are protected,
mls - Multi Level Security protection.
SELINUXTYPE=targeted

[root@node1 ~]#
```

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

### Turn On Security-Enhanced Linux (SELinux)

The open-source application SELinux (security-enhanced Linux) is a MAC module that resides at the kernel level in Linux systems. It decides which process can access which files, directories, and ports; it provides an additional layer of system security.

SELinux uses Linux Security Modules (LSM), which is a framework of components used to implement MAC modules without requiring any change in the kernel source code.

SELinux security policy is implemented using the following:

- Type enforcement (TE), which assigns every object a type on the system. The network defender then defines policies. These policies state what access is allowed between pairs of types. TE allows applying fine-grained permissions on files and processes.
- RBAC (role-based access controls), which applies a specific set of roles to the user to allow or restrict specific actions to control system activity. The network defender can allow the user some of the access privileges of the root account using RBAC (without granting full control of the system). For example, the network defender can allow access to a server for interacting with web application-related activity without allowing full root access.
- Multi-level security (MLS), which allows users to share classified information at different security clearance levels. MLS is helpful for governments that possess classified information.

Steps for using SELinux on Linux Distribution are listed below:

- Ubuntu already contains “AppArmor.” Disable it before installing SELinux to avoid unnecessary conflicts using the following commands.

```
sudo /etc/init.d/apparmor stop
```

```
apt-get update && upgrade -yuf
```

- Enter the command `apt-get install selinux` in the terminal.
- Open the configuration file using the following command after finishing the installation of SELinux.

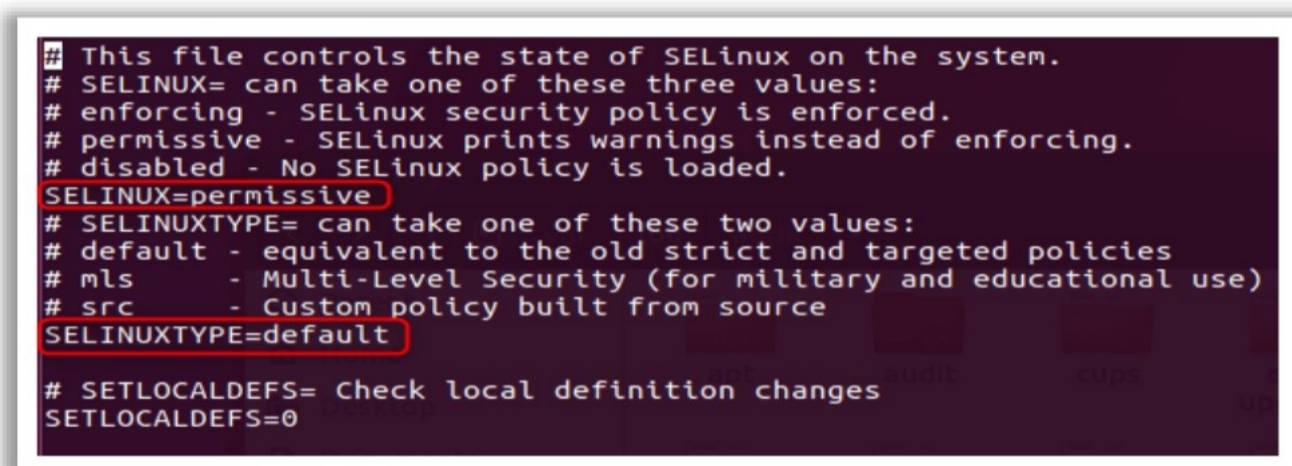
```
nano /etc/selinux/config
```

- SELinux provides two global level management rules/modes that help apply policies as per the need of the network defender:
  - Permissive logs the violated rules.
  - Enforcing denies access to a particular request sent by a process.

Here, set parameter values as permissive and default.

```
SELINUX=permissive
```

```
SELINUXTYPE=default
```



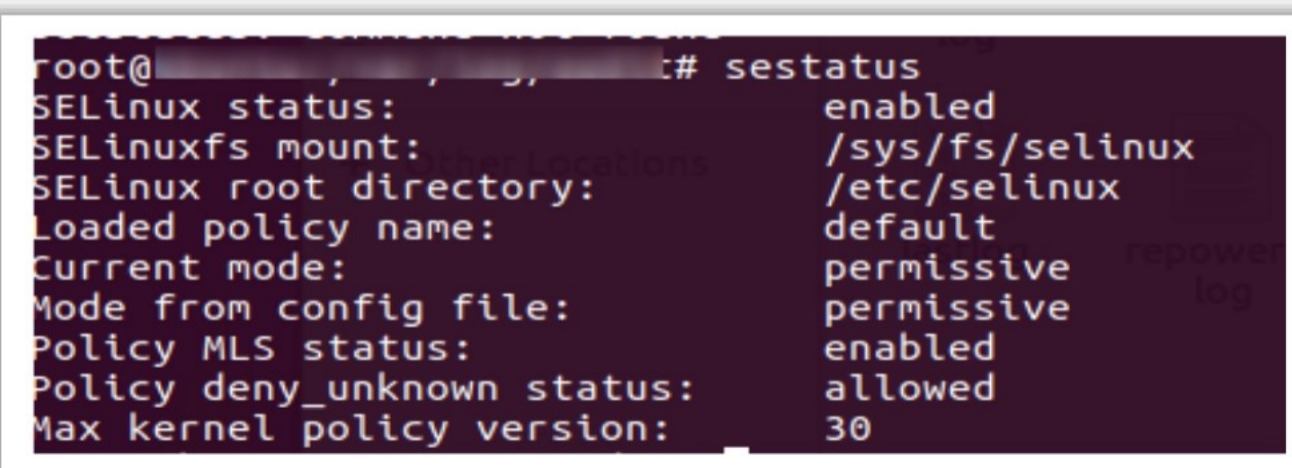
```
This file controls the state of SELinux on the system.
SELINUX= can take one of these three values:
enforcing - SELinux security policy is enforced.
permissive - SELinux prints warnings instead of enforcing.
disabled - No SELinux policy is loaded.
SELINUX=permissive
SELINUXTYPE= can take one of these two values:
default - equivalent to the old strict and targeted policies
mls - Multi-Level Security (for military and educational use)
src - Custom policy built from source
SELINUXTYPE=default
SETLOCALDEFS= Check local definition changes
SETLOCALDEFS=0
```

Figure 6.71: Setting parameter values as permissive and default

- Save and close the file.
- Reboot the system.

Note: If SELinux is not properly configured before rebooting the system, it will make the entire OS unbootable.

- Once the system starts, enter the command `sestatus` to check the status of the running status of SELinux.



```
root@_____# sestatus
SELinux status: enabled
SELinuxfs mount: /sys/fs/selinux
SELinux root directory: /etc/selinux
Loaded policy name: default
Current mode: permissive
Mode from config file: permissive
Policy MLS status: enabled
Policy deny_unknown status: allowed
Max kernel policy version: 30
```

Figure 6.72: sestatus command

This command shows basic details of SELinux. The default mode, enforcing, blocks all the requests. However, for the convenience of a normal user, it is set to the permissive mode, which allows the request but logs any violated rule set by the administrator. The logs can be checked in the audit.log file located in /var/log/audit.

## Audit Linux System for Security Compliance using OpenSCAP



**SCAP (security content automation protocol)** is a multi-purpose framework of specifications that supports automated configuration, vulnerability and patch checking, technical control compliance activities, and security measurement

To **Install OpenSCAP** use the following command:

On Fedora:

```
dnf install openscap-scanner
```

On RHEL 6, RHEL7, CentOS 6, and CentOS 7:

```
yum install openscap-scanner
```

On Debian and Ubuntu:

```
apt-get install libopenscap8
```

```
oscap xccdf eval --profile xccdf_org.ssgproject.content_profile_rht-cpp --results-arf arf.xml --report report.html /usr/share/xml/scap/ssg/content/ssg-rhel6-ds.xml
Title Ensure /tmp Located On Separate Partition
Rule xccdf_org.ssgproject.content_rule_partition_for_tmp
Ident CCE-26435-8
Result fail

Title Ensure /var Located On Separate Partition
Rule xccdf_org.ssgproject.content_rule_partition_for_var
Ident CCE-26639-5
Result fail

Title Ensure /var/log Located On Separate Partition
Rule xccdf_org.ssgproject.content_rule_partition_for_var_log
Ident CCE-26215-4
Result fail

Title Ensure /var/log/audit Located On Separate Partition
Rule xccdf_org.ssgproject.content_rule_partition_for_var_log_audit
Ident CCE-26436-6
Result fail

Title Ensure Red Hat GPG Key Installed
Rule xccdf_org.ssgproject.content_rule_ensure_redhat_gpgkey_installed
Ident CCE-26596-6
Result fail

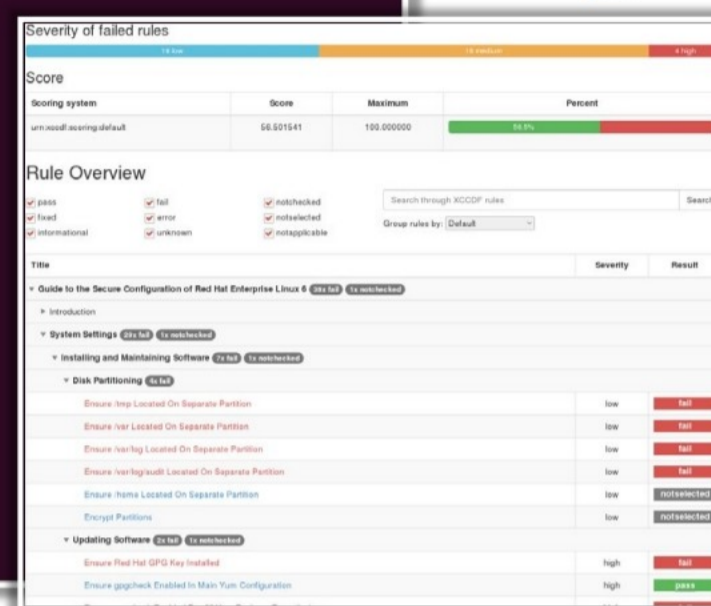
Title Ensure gpgcheck Enabled In Main Yum Configuration
Rule xccdf_org.ssgproject.content_rule_ensure_gpgcheck_globally_activated
Ident CCE-26709-6
Result pass

Title Ensure gpgcheck Enabled For All Yum Package Repositories
Rule xccdf_org.ssgproject.content_rule_ensure_gpgcheck_never_disabled
Ident CCE-26647-8
Result fail

Title Ensure Software Patches Installed
Rule xccdf_org.ssgproject.content_rule_security_patches_up_to_date
Ident CCE-27635-2
Result notchecked
```

Result Printed using Command Line

HTML Report



Source: <https://www.open-scap.org>

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

## Audit Linux System for Security Compliance using OpenSCAP

### OpenSCAP

Source: <https://www.open-scap.org>

SCAP (security content automation protocol), pronounced “S Cap or SCAP,” is a standard security specification maintained by National Institute of Standard and Technology (NIST). It can handle multiple security issues on the host machine. SCAP supports various activities for host safety such as patch checking, vulnerability checking, technical control and compliance activities, and security measurement. NIST recommends SCAP for security automation and policy compliance activities.

The OpenSCAP project provides a useful set of open-source tools for implementing SCAP.

SCAP is a suite of selected open standards, which performs the following functions:

- Lists flaws in software, configuration issues in security, and product names.
- Measures systems to decide the existence of the vulnerability.
- Provides ranking the results of the measurements to evaluate the impact of the discovered security issues.

### SCAP Components

SCAP standard family comprises of multiple component standards. The components are designed to work together toward a common goal. For each component the standard defines a document format with syntax and semantics of the internal data structures. All the component standards are based on extensible markup language (XML) and each component standard defines its own

XML namespace. Different versions of the same component standard (language) may also be distinguished by different XML namespaces.

SCAP standard consists of these components:

- Common Vulnerabilities and Exposures (CVE)
- Common Configuration Enumeration (CCE) (prior website at Mitre)
- Common Platform Enumeration (CPE)
- Common Vulnerability Scoring System (CVSS)
- Extensible Configuration Checklist Description Format (XCCDF)
- Open Vulnerability and Assessment Language (OVAL)
- Open Checklist Interactive Language (OCIL) Version 2.0
- Asset Identification (AID)
- Asset Reporting Format (ARF)
- Common Configuration Scoring System (CCSS)
- Trust Model for Security Automation Data (TMSAD)

### **Audit Linux System for Security Compliance with OpenSCAP**

OpenSCAP helps to implement SCAP and supports features for implementing security policies. It is a standardized compliance checking solution and a line of specifications maintained by NIST for maintaining system security.

#### **Advantages**

- Provides automatic vulnerability checking
- Customizes policies
- Allows easy implementation
- Prevents attacks on the host

#### **Key Features**

- Security compliance ensures that the applied security policy is in line with the system.
- Vulnerability assessment allows classification of vulnerabilities and scans the system. OpenSCAP defines what new vulnerabilities exist.

#### **OpenSCAP Tools**

- OpenSCAP Base: This command-line tool allows different SCAP capabilities such as:
  - Showing information about specific security contents
  - Vulnerability and configuration scanning
  - Converting between different SCAP formats

- OpenSCAP Daemon
- SCAP Workbench: The GUI SCAP Workbench allows user to tailor SCAP content easily, perform local or remote scans, and export results.
- SCAPTimony
- OSCP Anaconda add-on

To use OpenSCAP in Ubuntu, follow the steps outlined below:

- Enter the following command `sudo apt-get install libopenscap8` in the terminal.

- Enter the following command to get the OVAL file.

```
wget https://people.canonical.com/~ubuntu-security/oval/com.ubuntu.xenial.cve.oval.xml
```

- Run OpenSCAP using the following command.

```
oscap oval eval --results /tmp/results-xenial.xml --report /tmp/report-xenial.html com.ubuntu.xenial.cve.oval.xml
```

- See the results by accessing Firefox `/tmp/report-xenial.html`.

The screenshot displays the OVAL Results web interface in a browser window. The address bar shows the file path `file:///tmp/report-xenial.html`. The interface is divided into several sections:

- OVAL Results Generator Information:** A table with columns Schema Version (5.11.1), Product Name (cpe:/a:open-scap:oscap), Product Version (1.2.8), Date (2017-03-02), and Time (01:00:17).
- OVAL Definition Generator Information:** A table with columns Schema Version (5.11.1), Product Name (Canonical CVE OVAL Generator), Product Version (1), Date (2017-03-02), and Time (00:49:44). It also includes counts for Definitions (6956 Total), Tests (13790), Objects (1033), States (2284), and Variables (0).
- System Information:** A table listing Host Name (ru-VirtualBox), Operating System (Linux), Operating System Version (#85-Ubuntu SMP Mon Feb 20 11:50:30 UTC 2017), and Architecture (x86\_64). It also includes a detailed list of network interfaces (lo, eth0, eth1, eth2) with their respective IP addresses and MAC addresses.
- OVAL System Characteristics Generator Information:** A table with columns Schema Version (5.11.1), Product Name (cpe:/a:open-scap:oscap), Product Version (1), Date (2017-03-02), and Time (01:00:17).
- OVAL Definition Results:** A table with columns ID, Result, Class, Reference ID, and Title. It lists several vulnerabilities, including CVE-2017-5969, CVE-2017-5940, CVE-2017-5932, CVE-2017-5931, CVE-2017-5896, CVE-2017-5849, CVE-2017-5618, and CVE-2017-5604, all with a result of 'true' and a class of 'vulnerability'.

Figure 6.73: OpenSCAP


To install OpenSCAP on Fedora, execute the following command:







```
Dnf install openscap-scanner
```

To install OpenSCAP on RHEL 6, RHEL 7, CentOS 6, and CentOS 7, use the following command:

```
Yum install openscap-scanner
```

## Additional Linux Hardening Tools



|                                                                                                                                                                           |                                                                                                                                                                       |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|  <b>Bastille Linux</b><br><a href="https://sourceforge.net">https://sourceforge.net</a>  |  <b>nixarmor</b><br><a href="https://github.com">https://github.com</a>            |
|  <b>JShielder</b><br><a href="https://github.com">https://github.com</a>                 |  <b>bane</b><br><a href="https://github.com">https://github.com</a>                |
|  <b>Comodo Antivirus</b><br><a href="https://www.comodo.com">https://www.comodo.com</a> |  <b>Grsecurity</b><br><a href="https://grsecurity.net">https://grsecurity.net</a> |

Copyright © by **EC-Council**. All Rights Reserved. Reproduction is Strictly Prohibited.

## Additional Linux Hardening Tools

### Bastille Linux

Source: [www.sourceforge.net](http://www.sourceforge.net)

Bastille Linux hardens a Linux OS-based system through its default hardening mode. It builds a policy- based on the interaction with the user and applies the policy to the system. During assessment, it builds a report to explain the user about the security settings and let the user know which settings are tightened.

### JShielder

Source: [www.github.com](http://www.github.com)

JShielder automates the process of installing the required packages to harden a Linux server after interaction with the user.

### nixarmor

Source: [www.github.com](http://www.github.com)

This Linux project allows security professionals and network defenders to automate system hardening.

### bane

Source: [www.github.com](http://www.github.com)

Bane is an AppArmor profile generator for docker containers. It protects applications by setting restrictions on the resources they access or modify. It is used for security monitoring, system hardening, and application security.

## **Grsecuity**

Source: [www.grsecurity.net](http://www.grsecurity.net)

An extensive security enhancement to the Linux kernel, Grsecurity can defend against security threats using its intelligent access control, memory corruption-based exploit prevention, and a host of another system hardening that generally requires no configuration.

## Module Summary



- Linux is an open-source OS widely used across enterprises and government bodies
- Linux security is becoming a concern as hackers have exploited many of its vulnerabilities in the recent past
- The use of latest version for installing the OS help you prevent from many attacks
- Removing or disabling unnecessary services and software packages helps you reduce the attack surface
- Appropriate user access permission and strong password management policies discourages any unauthorized access

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

## Module Summary

This module described various security features of Linux that should be implemented to secure a Linux system. The key highlighted points in this module are listed below:

- Linux is an open-source operating system widely used across enterprises and government bodies.
- Linux security is becoming a concern as hackers have been exploited its vulnerabilities in the recent past.
- The use of latest version for installing the OS helps prevent many types of attacks.
- Removing or disabling unnecessary services and software packages helps reduce the attack surface.
- Appropriate user access permission and strong password management policies discourages any unauthorized access.
- AppArmor is a Linux kernel security module that allows the network defenders to restrict programs' capabilities through per-program profiles.
- SELinux (security-enhanced Linux) is a kernel-level MAC implementation for Linux.