



# INTRODUCTION TO LINUX AND VIRTUAL MACHINE

<p> Let's dive into it! </p>

# HELLO! I'm...

- Yee Shean
- A third-year Computer System and Network (CSN) student
- Currently doing my internship
- Nice to meet you!



# TABLE OF CONTENTS.



01

What is Linux and  
Virtual Machine?



02

Virtualization:  
Hypervisors



03

How to create a  
Bootable Disk?



04

Linux File Structure



05

Basic File Operations



06

Network Configuration  
Checking

# TABLE OF CONTENTS.



07

Text Processing Tools



08

Package Management



09

Manual and Info



10

Process Management



11

Bye Bye!



# 01 WHAT IS LINUX AND VIRTUAL MACHINE?

<p> Good question! </p>

# DID YOU **KNOW?**



## WHAT IS **LINUX**?

Linux is a free, open-source OS based on Unix.

Shell:

Allow users to interact with the OS (eg. Bash).

Use Cases:

Servers, desktop computers.



## WHAT IS **VM**?

VM is a software-based emulation of physical computer.

Hypervisor:

Software that creates and manages VMs.

Use Cases:

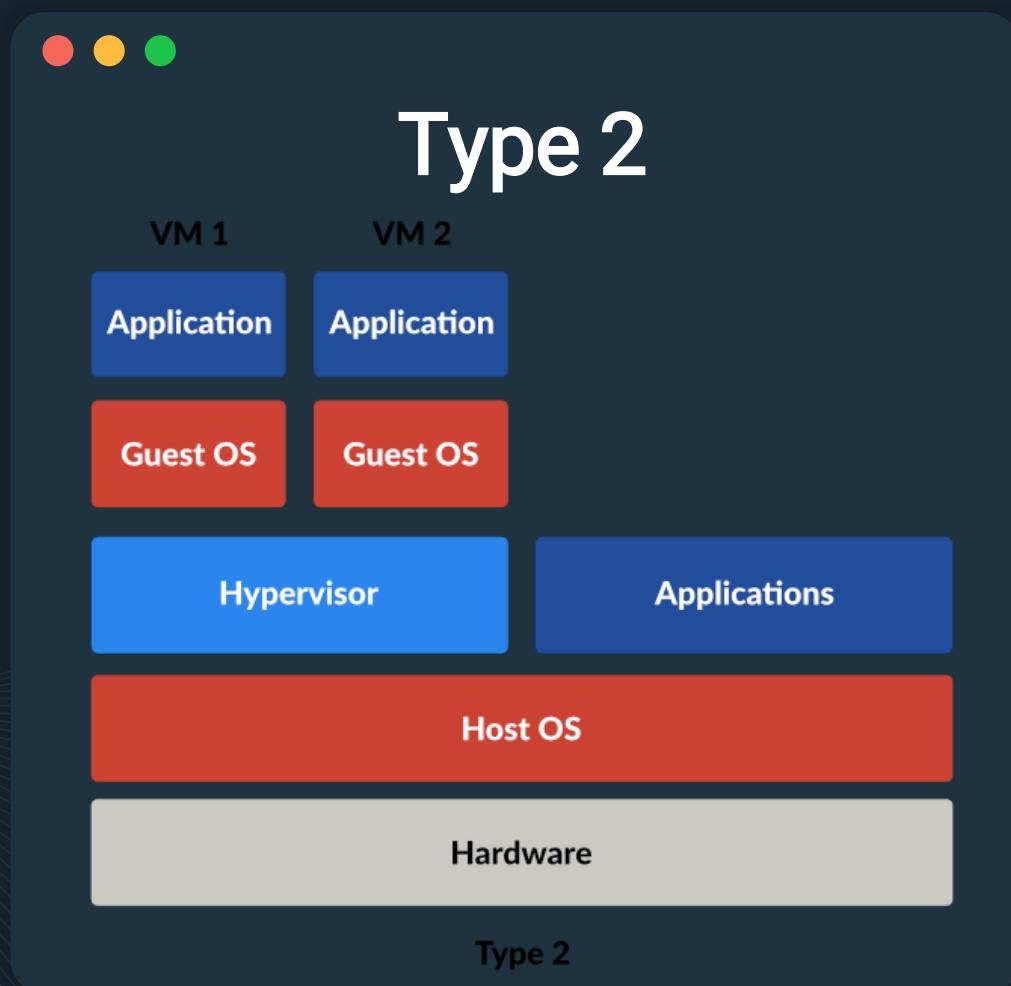
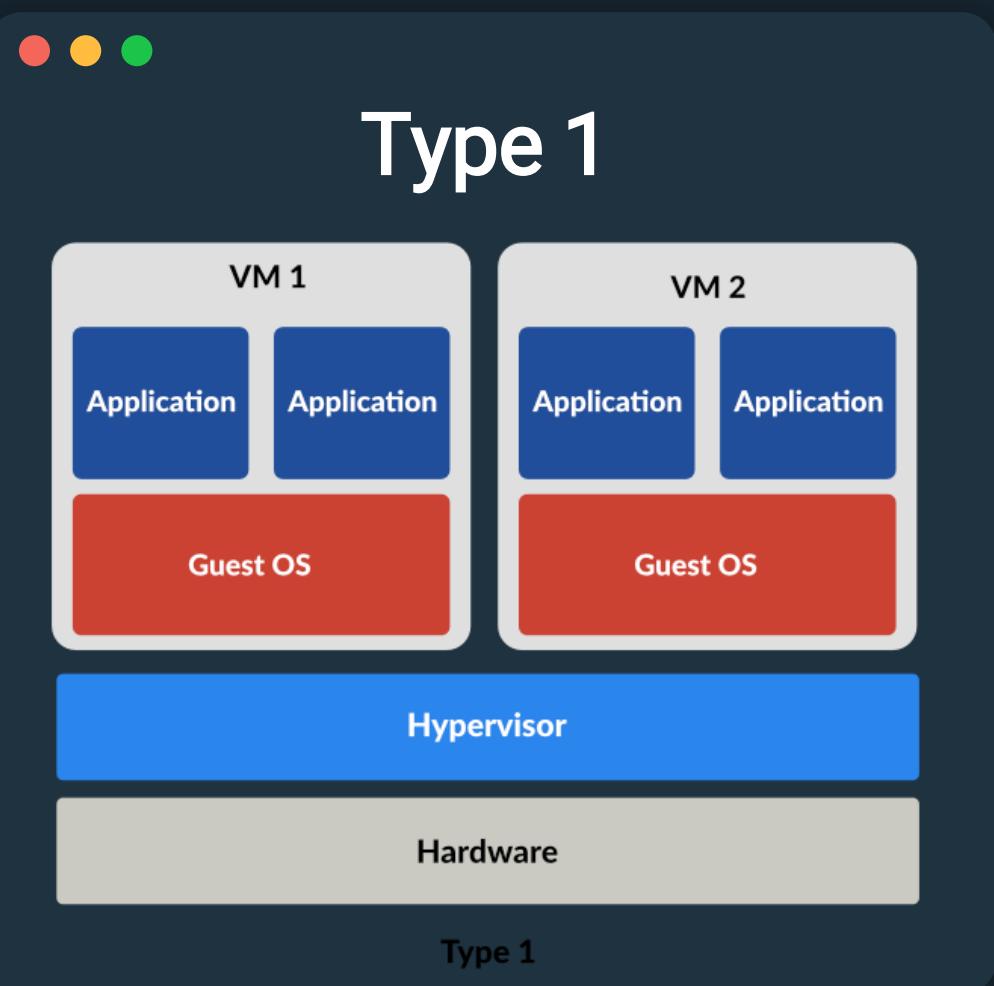
Development and testing environments.



# 02 VIRTUALIZATION: HYPERVISORS

<p> Hypervisor is a software that creates and manages VMs.</p>

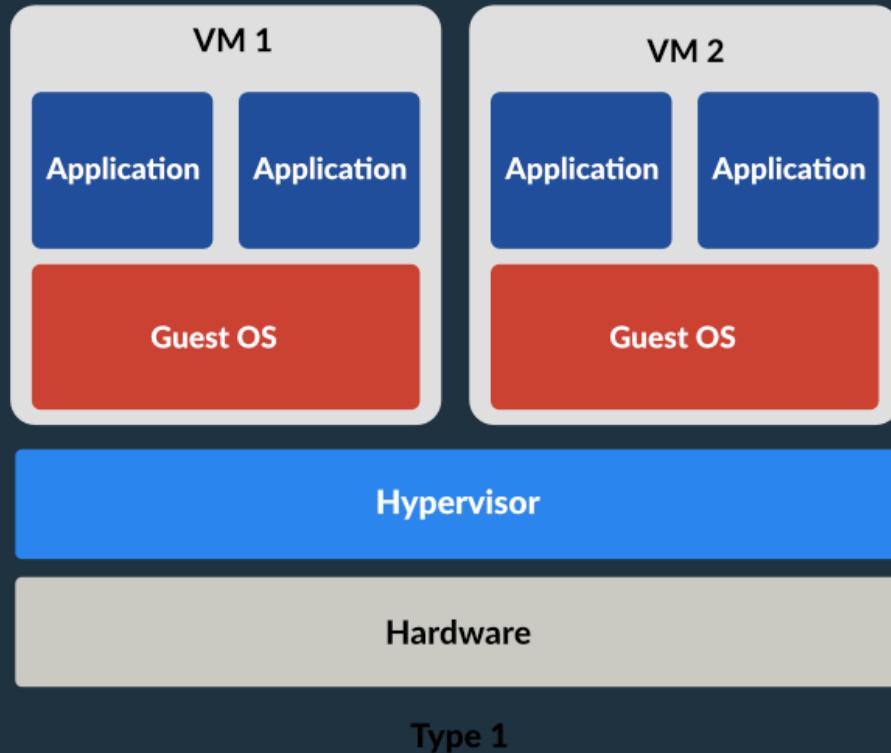
# Hypervisors: Type 1 vs Type 2





# Type 1 Hypervisor

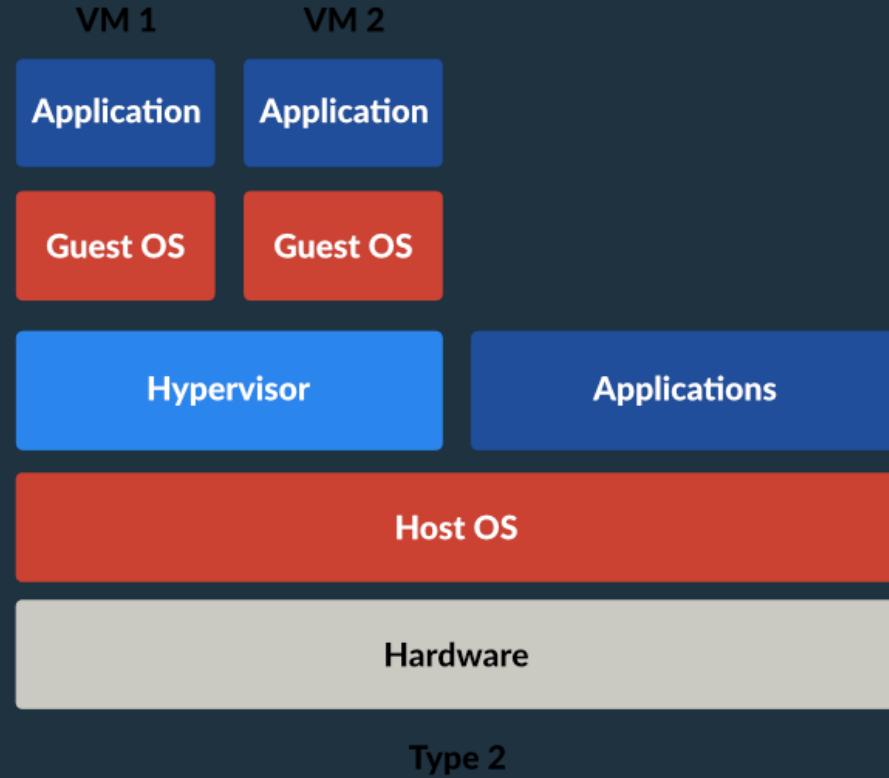
- Runs directly on the hardware of the host machine (bare-metal).
- Offers better performance and scalability for enterprise environments.
- Eg. VMware ESXi, Microsoft Hyper-V





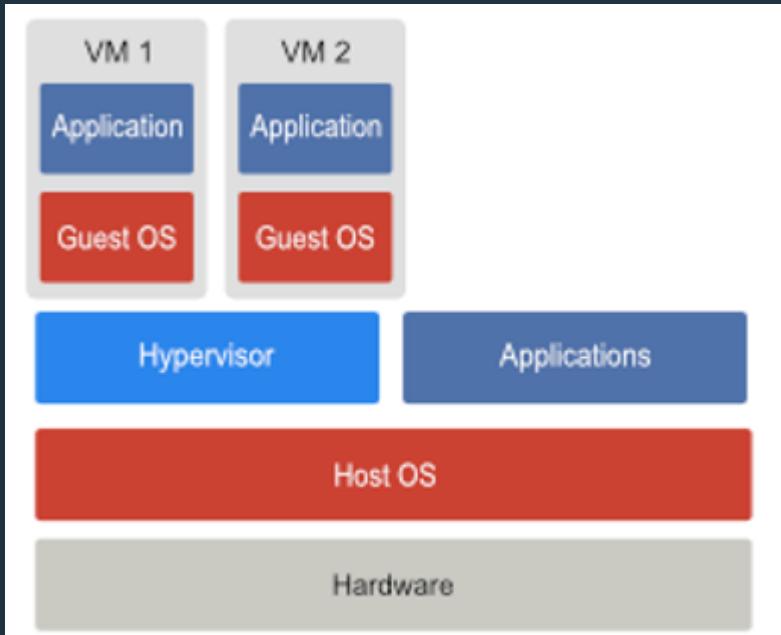
# Type 2 Hypervisor

- Runs on a host operating system (hosted).
- Easier to set up and typically used for development or personal uses.
- Eg. VMware Workstation, Oracle VirtualBox

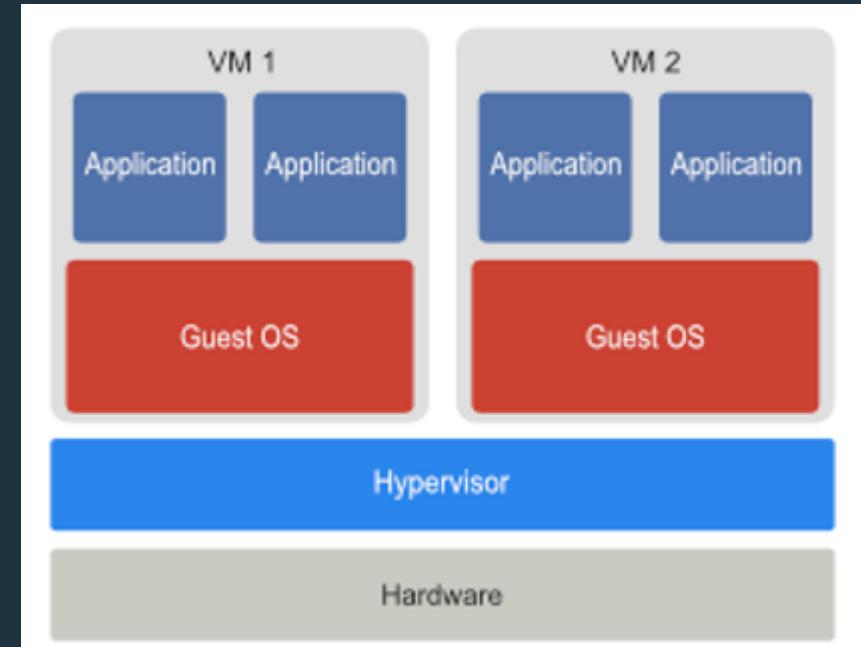




# Hypervisor Types



Type 2



Type 1



# 03 HOW TO CREATE A BOOTABLE DISK

<p> Used to install an OS in a physical machine. </p>

# HOW TO CREATE A BOOTABLE DISK



## **Bootable Disk Options:**

CD, DVD, or USB pendrive (flash drive)

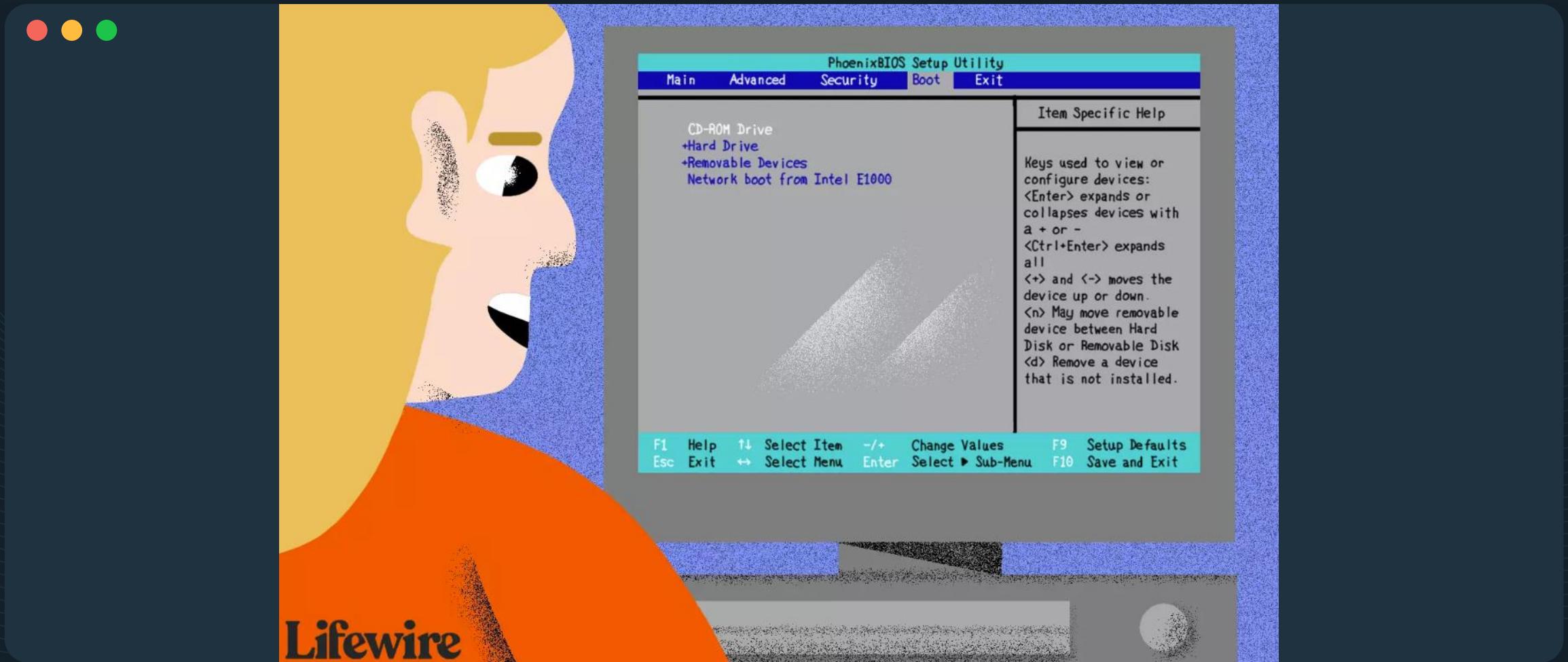
## **Process:**

- Select ISO file: Download the operating system installer in ISO format.
- Choose the drive: Insert the USB or DVD you want to make bootable.
- Flash the drive: Use software like Balena Etcher to mount the ISO onto the drive.

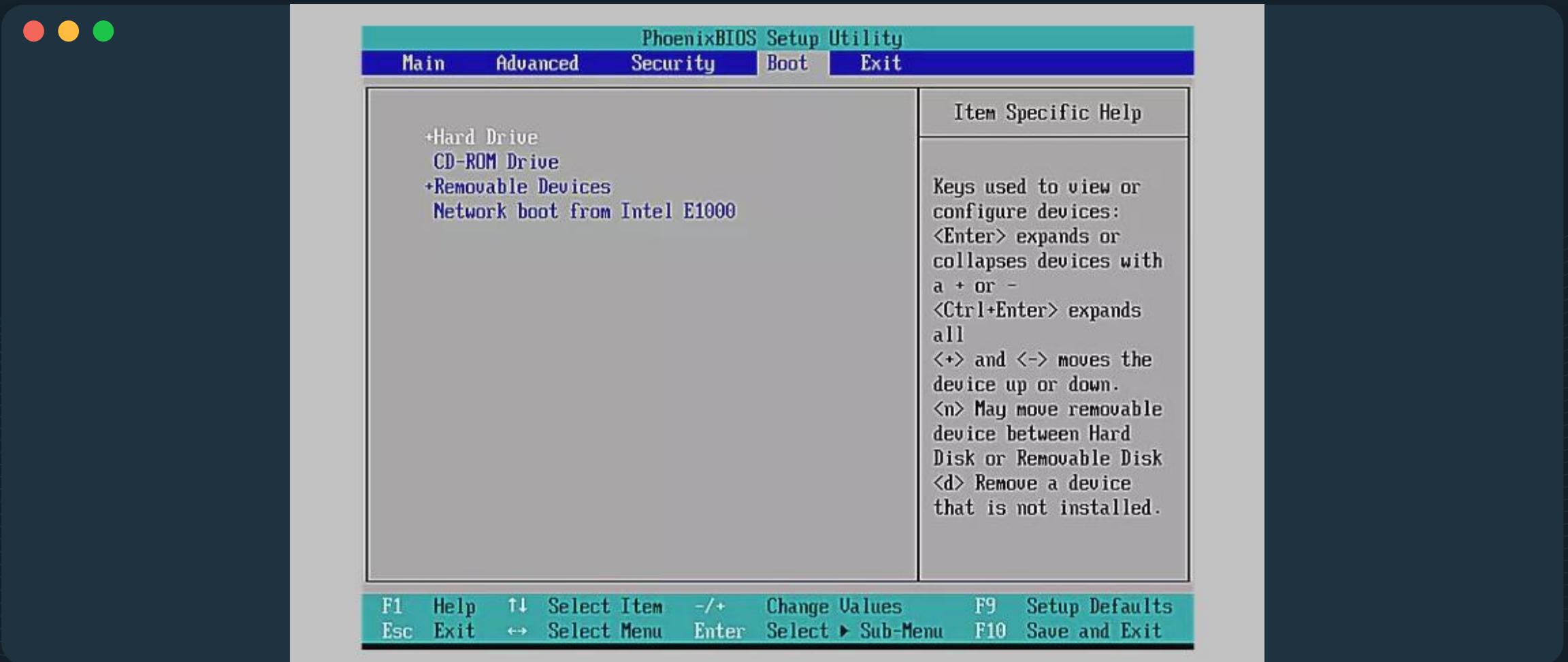
## **After Bootable Disk is created:**

- Change BIOS settings:
  - Change the boot order in the BIOS – set the bootable disk as the first boot device.

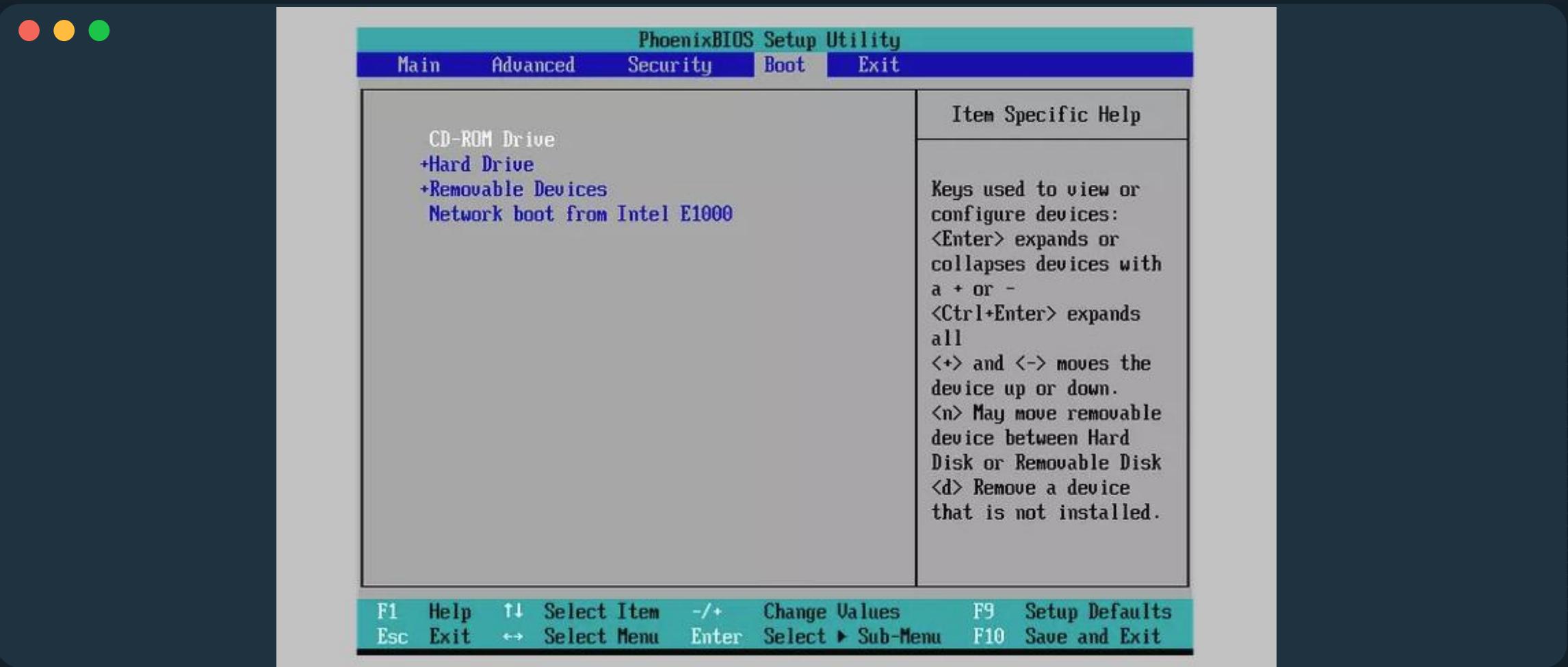
# Change BIOS Settings



# Change BIOS Settings



# Change BIOS Settings





# 04 LINUX FILE STRUCTURE

<p> Hierarchical structure to manage files and directories. </p>

# LINUX FILE STRUCTURE OVERVIEW



/etc

Stores configuration files

01



/bin

Contains essential binaries (commands)

02



/var

Stores variable data like logs, databases

03



/home

Contains user home directories

04



/usr

Contains user-installed software and utilities

05



/tmp

Stores temporary files that are cleared on reboot

06



# 05 BASIC FILE OPERATIONS

<p> Get ready with your fingers! </p>

# cd (Change directory)



**Move to a Specific  
Directory**

```
cd  
/path/to/directory
```

**Move to Home Directory**

```
cd ~  
OR  
cd
```

**Move Up One Level**

```
cd ..
```

Moves to the  
parent directory

# ls (List files)



## Basic Listing

```
ls
```

lists the files in  
the current  
directory

## Long Listing Format

```
ls -l
```

shows detailed  
info (permissions,  
ownership, size,  
and modification  
date)

## Include Hidden Files

```
ls -a
```

lists all files,  
including hidden  
files (those starting  
with a dot)

# cat (Concatenate & Display Files)



## Create a New File

```
cat > file.txt
```

Create and  
overwrite the  
file

## Display File Content

```
cat file.txt
```

Print the content  
of file.txt

## Concatenate Files

```
cat file1.txt  
file2.txt > merged.txt
```

Merge two files into  
one

# touch & mkdir & rm



## Create a File

```
touch newfile.txt
```

Create an empty  
file

## Create a Directory

```
mkdir new_directory
```

## Remove Files or Directories

```
rm file.txt
```

```
rm -rf directory
```

Forcefully deletes  
a directory and  
its contents  
without  
confirmation

# **mv** (move or rename files)



## **Move a File**

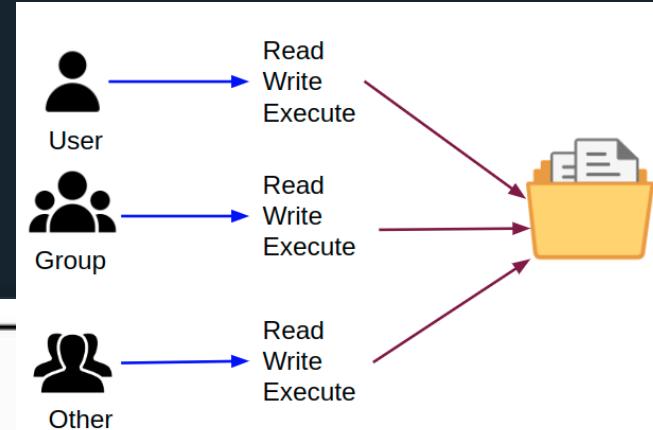
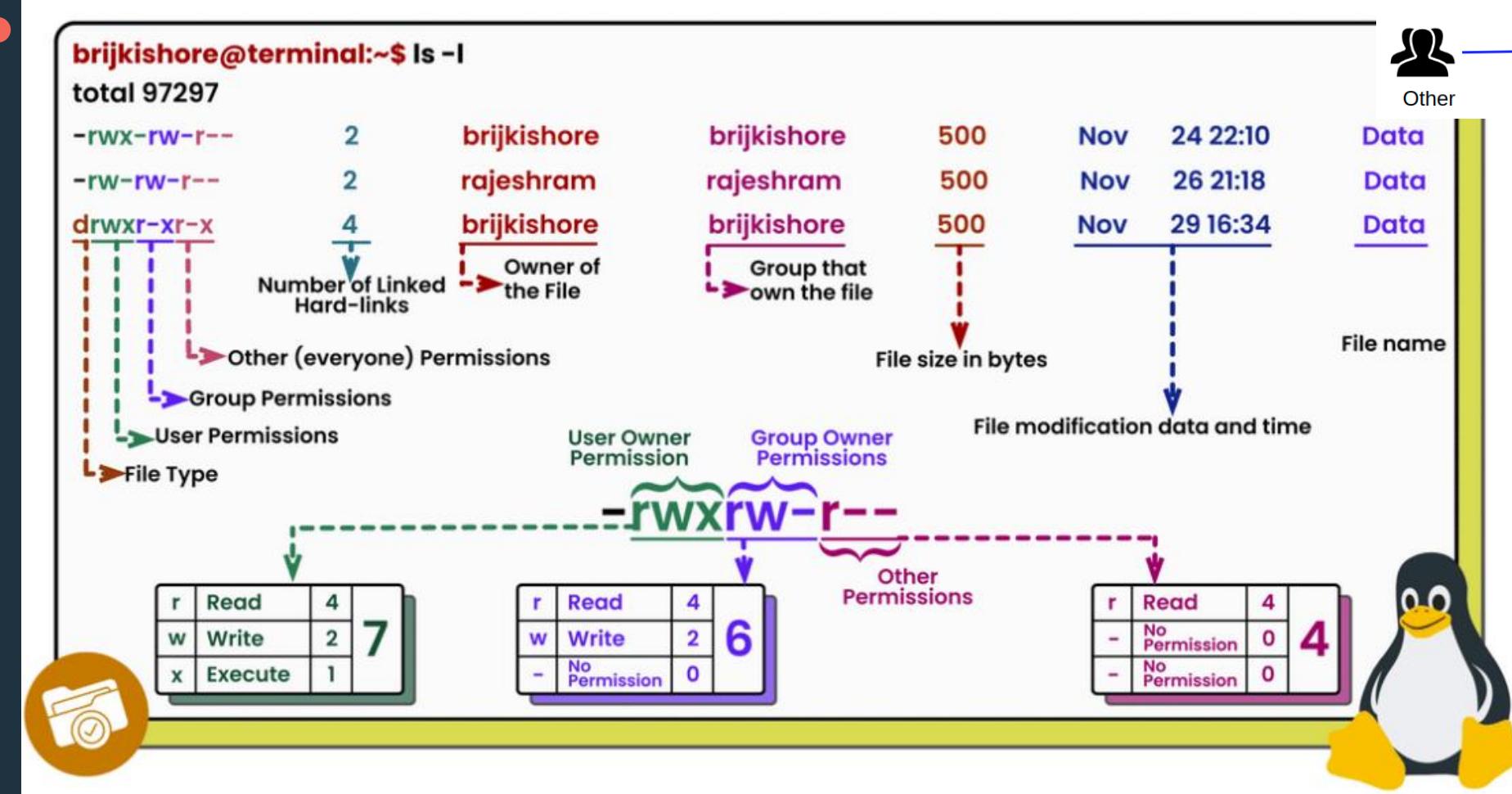
```
mv file.txt /new/location/
```

Moves the file to a  
different directory

## **Rename a File or Directory**

```
mv oldname.txt newname.txt
```

# LINUX FILE PERMISSION



# CHANGE FILE PERMISSION



## Numeric/Absolute Mode

r	Read	4
w	Write	2
x	Execute	1

Number	Permission Type	Symbol
0	No Permission	—
1	Execute	-x
2	Write	-w-
3	Execute + Write	-wx
4	Read	r-
5	Read + Execute	r-x
6	Read +Write	rw-
7	Read + Write +Execute	rwx

In this mode, file permissions are not represented as characters but a three-digit octal number.



## Checking Current File Permissions

```
ubuntu@ubuntu:~$ ls -l sample  
-rwx-rw-r-- 1 ubuntu ubuntu 15 Sep  6 08:00 sample
```

chmod 764 and checking permissions again

```
ubuntu@ubuntu:~$ chmod 764 sample  
ubuntu@ubuntu:~$ ls -l sample  
-rwxrw-r-- 1 ubuntu ubuntu 15 Sep  6 08:00 sample
```

# CHANGE FILE PERMISSION



## Symbolic Mode

User Denotations	
u	user/owner
g	group
o	other
a	all

Operator	Description
+	Adds a permission to a file or directory
-	Removes the permission
=	Sets the permission and overrides the permissions set earlier.

- In the Absolute mode, you change permissions for all 3 owners.
- In the Symbolic mode, you can modify permissions of a specific owner.
- It makes use of mathematical symbols to modify the Unix file permissions.

### Current File Permissions

```
home@VirtualBox:~$ ls -l sample  
-rw-rw-r-- 1 home home 55 2012-09-10 10:59 sample
```

### Setting permissions to the 'other' users

```
home@VirtualBox:~$ chmod o=rwx sample  
home@VirtualBox:~$ ls -l sample  
-rwxrwxrwx 1 home home 55 2012-09-10 10:59 sample
```

### Adding 'execute' permission to the user/group

```
home@VirtualBox:~$ chmod g+x sample  
home@VirtualBox:~$ ls -l sample  
-rwxrwxrwx 1 home home 55 2012-09-10 10:59 sample
```

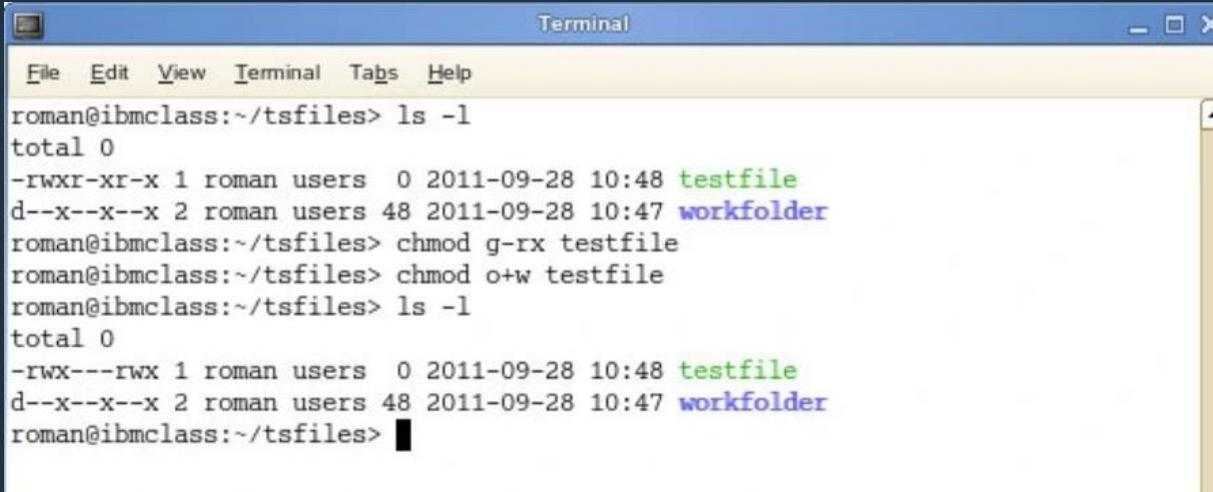
### Removing 'read' permission for 'user'

```
home@VirtualBox:~$ chmod u-r sample  
home@VirtualBox:~$ ls -l sample  
--w-rwxrwx 1 home home 55 2012-09-10 10:59 sample
```

# CHANGE FILE PERMISSION

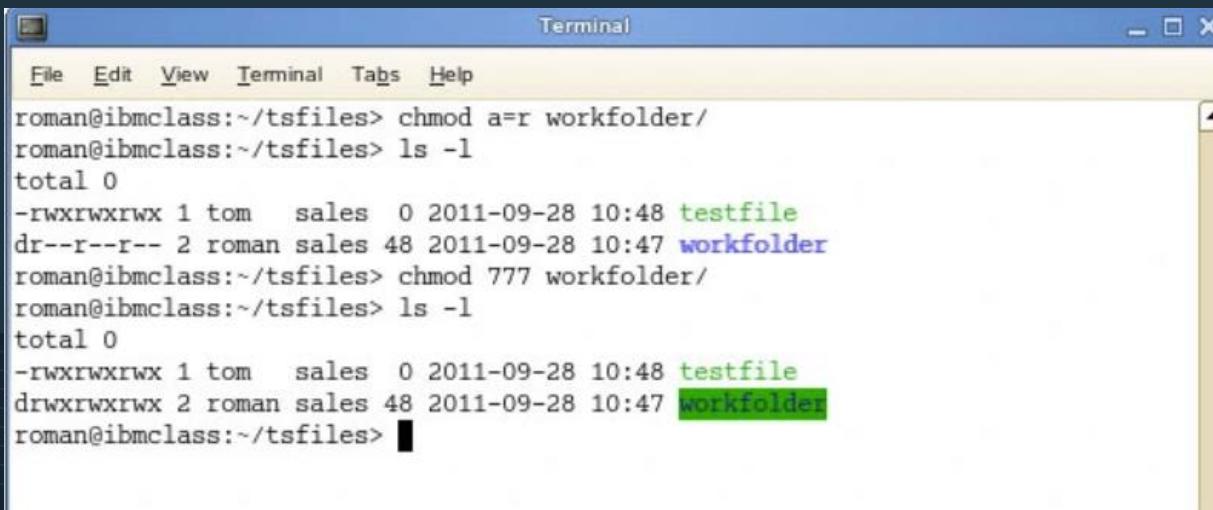


## Symbolic



```
Terminal
File Edit View Terminal Tabs Help
roman@ibmclass:~/tsfiles> ls -l
total 0
-rwxr-xr-x 1 roman users 0 2011-09-28 10:48 testfile
d--x--x--x 2 roman users 48 2011-09-28 10:47 workfolder
roman@ibmclass:~/tsfiles> chmod g-rx testfile
roman@ibmclass:~/tsfiles> chmod o+w testfile
roman@ibmclass:~/tsfiles> ls -l
total 0
-rwx---rwx 1 roman users 0 2011-09-28 10:48 testfile
d--x--x--x 2 roman users 48 2011-09-28 10:47 workfolder
roman@ibmclass:~/tsfiles>
```

## Numeric/Absolute



```
Terminal
File Edit View Terminal Tabs Help
roman@ibmclass:~/tsfiles> chmod a=r workfolder/
roman@ibmclass:~/tsfiles> ls -l
total 0
-rwxrwxrwx 1 tom sales 0 2011-09-28 10:48 testfile
dr--r--r-- 2 roman sales 48 2011-09-28 10:47 workfolder
roman@ibmclass:~/tsfiles> chmod 777 workfolder/
roman@ibmclass:~/tsfiles> ls -l
total 0
-rwxrwxrwx 1 tom sales 0 2011-09-28 10:48 testfile
drwxrwxrwx 2 roman sales 48 2011-09-28 10:47 workfolder
roman@ibmclass:~/tsfiles>
```

## chmod

Changes permissions using symbolic or numeric modes.

### symbolic

Defines a target, action, and permissions.

Target (u/g/o)

chmod u +x filename

Action - add (+), remove (-), or set (\*)

Permission (r/w/x)

### Numeric

Uses the octal numbering system, which uses digits 0 through 7. Read is 4, write is 2, execute is 1, and permission is 0

These numbers are added together to combine permissions. For example, a read and execute permission would be 5 (4+0+1), whereas read and write is 6 (4+2+0)

User (owner)

Other

chmod 600 filename

Group



# 06 NETWORK CONFIGURATION CHECKING

<p> How to check or verify? </p>

# Check IP, Ports & Routing Table.



## IP Address

ip addr

ifconfig  
(older command)



## Open Ports

netstat -tuln

ss -tuln



## Routing Table

route -n



# 07 TEXT PROCESSING TOOLS

<p> There are many text processing tools! </p>

# TEXT PROCESSING TOOLS



## grep

grep "pattern" file.txt  
To find lines containing the pattern

## sed

sed 's/old/new/g' file.txt  
To replace old with new  
  
sed '/pattern/d' file.txt  
To delete lines with pattern

sed -i 's/old/new/g' file.txt  
To modify the file directly

# TEXT PROCESSING TOOLS



## sort

```
sort file.txt
```

To sort lines alphabetically

```
sort -r file.txt
```

To sort in reverse order

## tr

```
tr 'a-z' 'A-Z'
```

Character replacement

```
tr -d 'a'
```

To delete all occurrences of  
the letter 'a'

```
echo "HELLO WORLD" | tr 'A-Z'  
'a-z' > output.txt
```

# TEXT EDITING TOOLS



## nano

- Basic text editor
- **Good for Beginners:** Nano is easy to use for quick text edits without needing advanced knowledge.

## vim

- Advanced text editor
- **Good for Advanced Users:** Vim is more powerful, with complex commands for efficient editing.



# 08 PACKAGE MANAGEMENT

<p> All your packaging need! </p>

# Package Manager Usage



## \*\*\*Debian-based Systems\*\*\*

```
curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo tee  
/usr/share/keyrings/docker-archive-keyring.gpg > /dev/null
```

**Step 1: Download the GPG key**

```
echo "deb [arch=amd64 signed-by=/usr/share/keyrings/docker-archive-keyring.gpg]  
https://download.docker.com/linux/ubuntu $(lsb_release -cs) stable" | sudo tee  
/etc/apt/sources.list.d/docker.list > /dev/null
```

**Step 2: Add the repository with key reference**

```
sudo apt-get update
```

**Step 3: Refreshes the list of available packages from all repositories**

# Package Manager Usage



```
sudo apt-get install docker-ce
```

**Step 4: Install the specified package like Docker from the newly added repository**

```
sudo apt-get remove package_name
```

**(Optional) Remove the package but keeps configuration files.**

**\*\*\*RPM-based Systems (Red Hat, CentOS)\*\*\***

Install a Package: sudo yum install package\_name or sudo dnf install package\_name

Remove a Package: sudo yum remove package\_name or sudo dnf remove package\_name



# 09 MANUAL

<p> When you don't know how to use a command. </p>



# man command

Shows the manual page for a command, providing details on its usage, options, and examples



Pro tips: To search within man pages, press '/' followed by the search term



# 10 PROCESS MANAGEMENT

<p> Keep all your process in check! <p>

# CHECK RUNNING PROCESSES



`ps aux`

- To list out all the currently running processes



`top or htop`

- To provide a real-time, dynamic view of running processes
- `htop`: An enhanced version of top with a more user-friendly interface

# KILLING A PROCESS



## Graceful Termination

`kill PID`

`kill -SIGTERM PID`

## Forceful Termination

`kill -9 PID`

`kill -SIGKILL PID`



## Finding a Process by Name:

To find the PID by its name

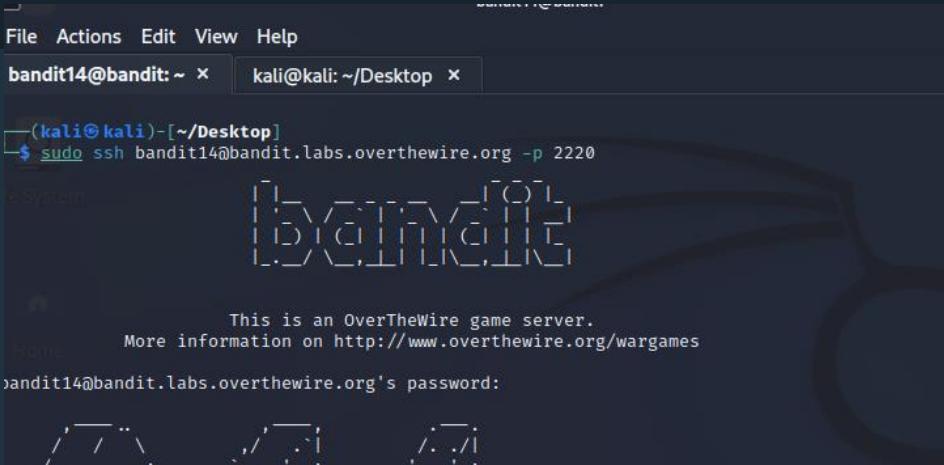
`pgrep process_name`

To forcefully kill all processes with a specific name

`pkill -9 process_name`

# Bandit Wargame

- <https://overthewire.org/wargames/bandit/>



A screenshot of a terminal window titled "bandit14@bandit: ~" and "kali@kali: ~/Desktop". The user is running a command to connect to an OverTheWire game server: `sudo ssh bandit14@bandit.labs.overthewire.org -p 2220`. The terminal shows a welcome message from the game server, which includes a small ASCII art logo of a cat. It also asks for the password of the user "bandit14@bandit.labs.overthewire.org".





# THANK YOU!



See u next time!

