



Linux

BEGINNING

NAMES

SBU - FALL 2022

What is Linux ?

- Linux is a free and open source operating system.
- At it's core, the Linux operating system is derived from the Unix OS.
 - **Unix** was created in the 1960s by Dennis Ritchie and Ken Thompson, both of them **also invented the C programming language**.
- Linux was initially named GNU and was developed by Richard Stallman
- **Linux** was the name of the kernel created in 1991 by Linux Torvalds, a student at the University of Helsinki.
- People started calling the GNU OS, Linux – because of the name of the kernel

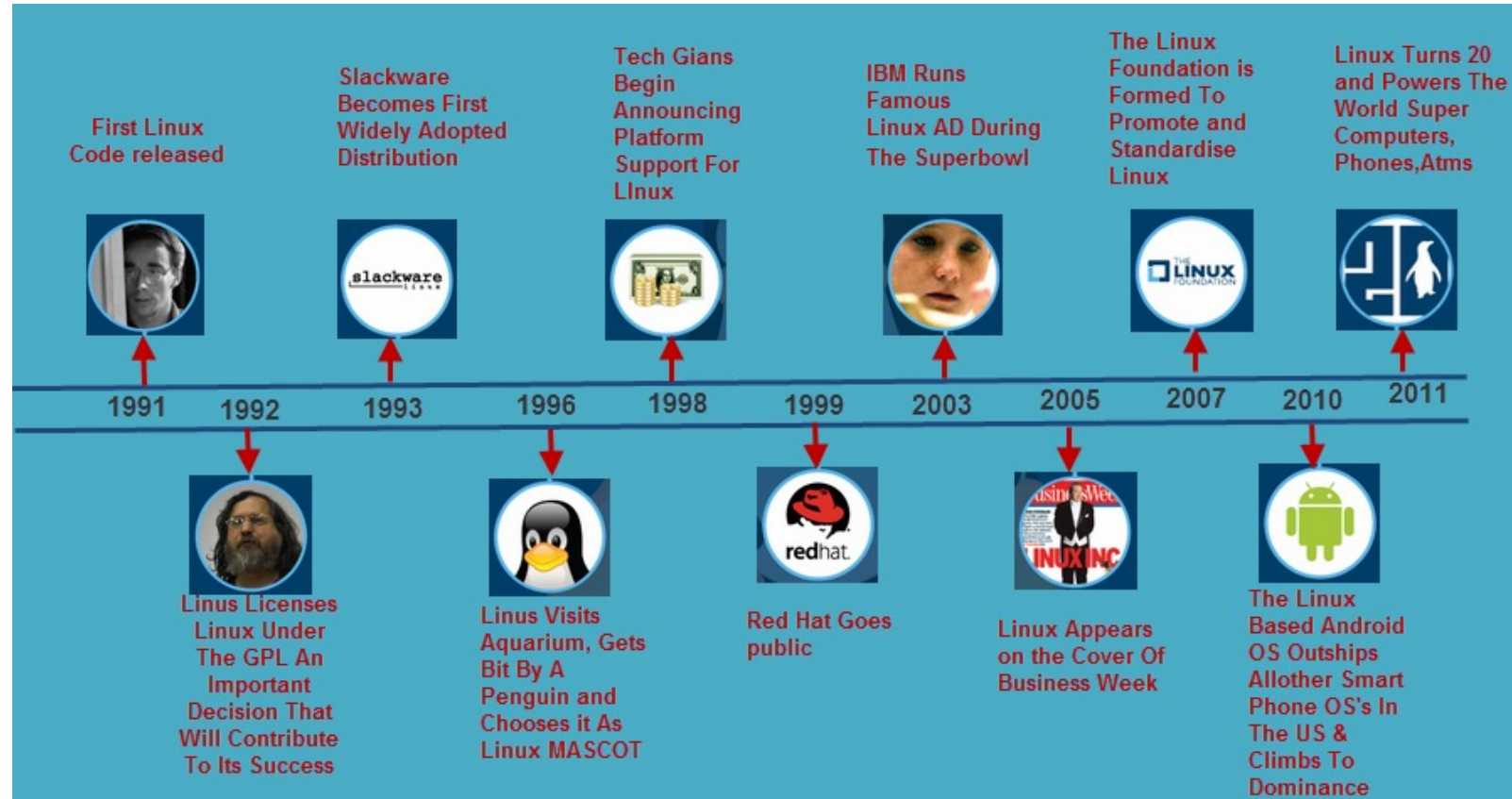


Kenneth L. Thompson

Dennis M. Ritchie



Timeline



What distributions of Linux do exist?

- Linux OS has multiple distributions that are derived from it's initial deployment.
- Most of the are FREE and offer full functionality:
 - **Examples:**
 - Debian
 - Ubuntu
 - CentOS
 - OpenSUSE
 - Mint
 - Gentoo
 - Slackware.



What distributions of Linux do exist (II)?

- Some examples of non-free (enterprise) Linux distros are:

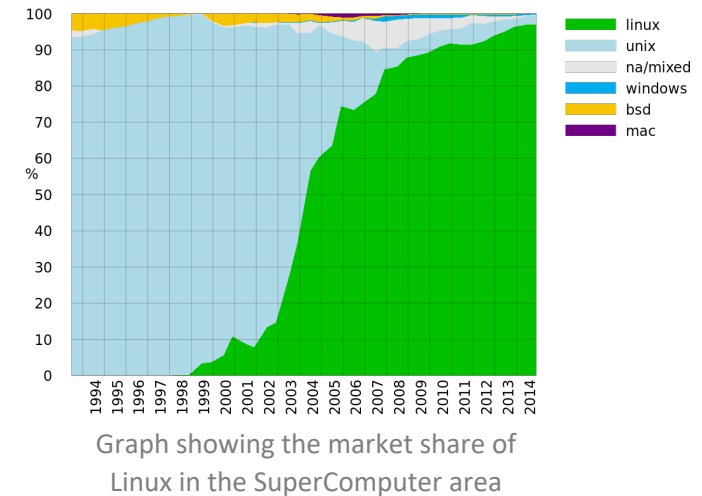
- **Red Hat** Enterprise Linux
 - **SUSE** Linux Enterprise Server
 - **Oracle** Linux
 - **Scientific** Linux
 - **Turbo** Linux
 - Linux **Mandriva**
-
- For more info about various types of Linux:

<http://distrowatch.com/>



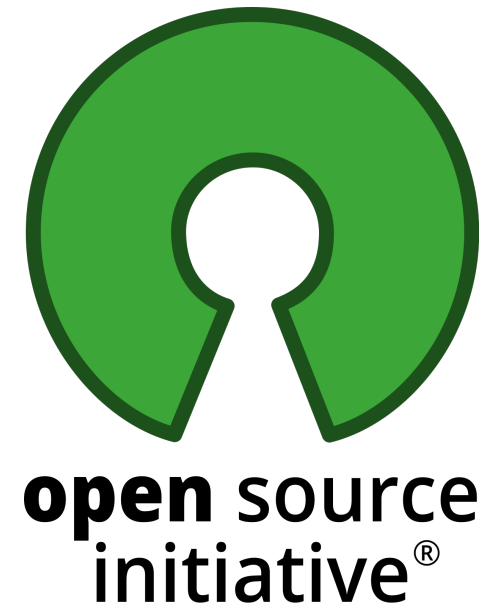
Why should I care about Linux ?

- In September 2008 Steve Ballmer (Microsoft CEO) claimed 60% of servers run Linux and 40% run Windows Server. According to IDC's report covering Q2 2013, Linux was up to 23.2% of worldwide server revenue.
- Linux is used as:
 - Server (HTTP, FTP, DNS, file server, etc)
 - Desktop (it's a free alternative to Microsoft's Windows XP, Vista, 7, 8 family)
 - Supercomputer operating system:
 - According to Wikipedia & top500.org, over 95% of Supercomputers use Linux as their host OS.
- You can also find Linux distros in:
 - Routers, firewalls, switches
 - Smartphones (see Android)
 - Gaming consoles



Why should I Use Linux ?

- It runs on anywhere
- Open-Source (!?)
- Customizable
- Educational
- Secure (Why?)
- Reliable
- Easy to Install
- Strong Community Support
- Easy to Use
- Free



Simplified architecture of Linux (I)

Kernel:

- The kernel is the heart of the operating system.
- It interacts with hardware and most of the tasks like memory management, task scheduling and file management.

Shell:

- The shell is the utility that processes your requests.
- When you type in a command at your terminal, the shell interprets the command and calls the program that you want.
- The shell uses standard syntax for all commands.
- C Shell, Bourne Shell and Korn Shell are most famous shells which are available with most of the Unix variants.

Simplified architecture of Linux (II)

Commands and Utilities:

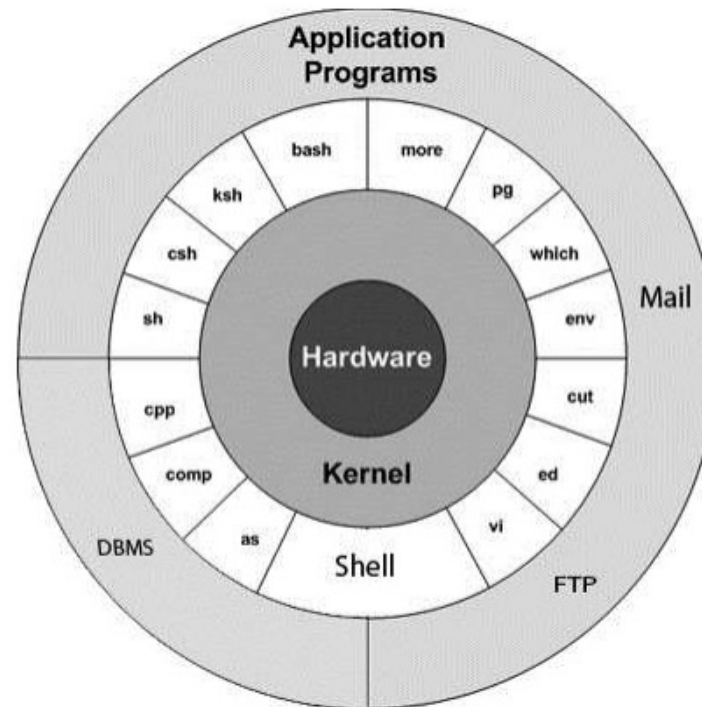
- There are various commands and utilities which you would use in your day to day activities.
- **cp**, **mv**, **cat** and **grep** are few examples of commands and utilities.
- There are over 250 standard commands plus numerous others provided through 3rd party software.
- All the commands come along with various optional options.

Files and Directories:

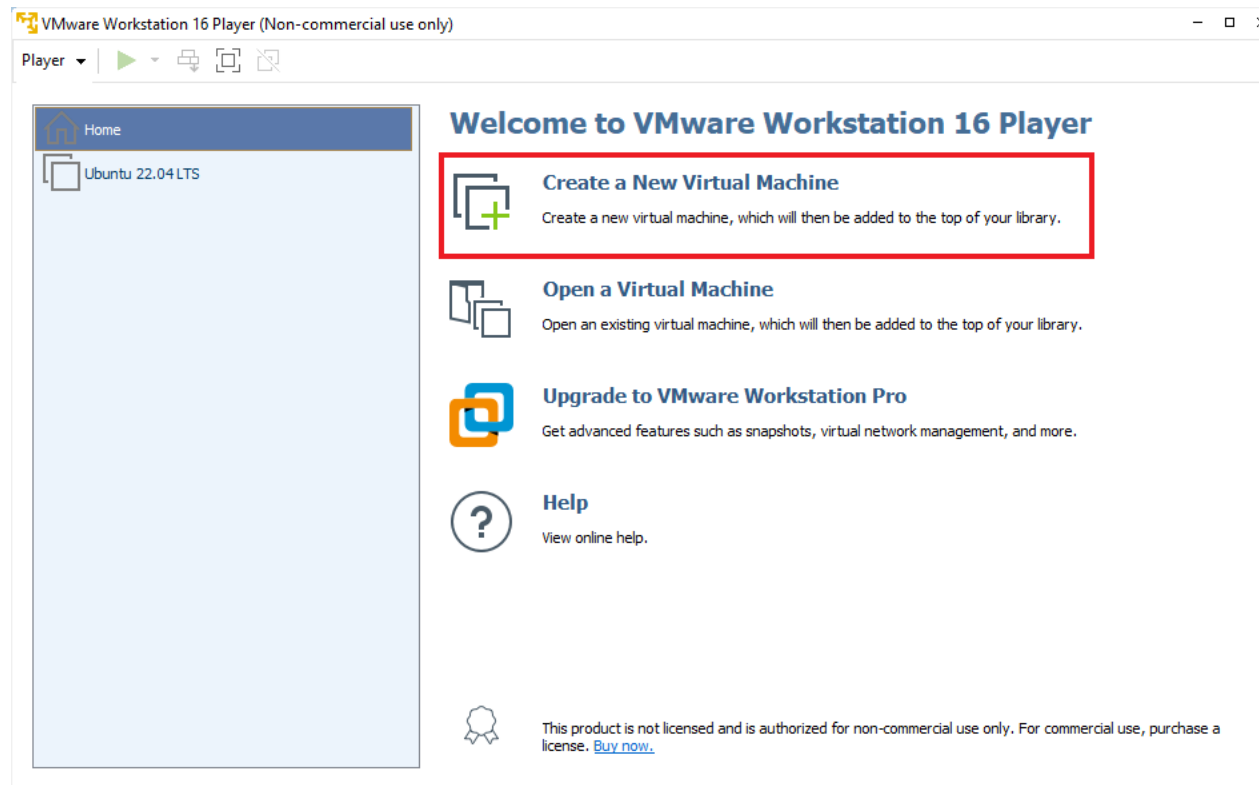
- All data in Linux is organized into files. All files are organized into directories.
- These directories are organized into a tree-like structure called the filesystem.

Simplified architecture of Linux (III)

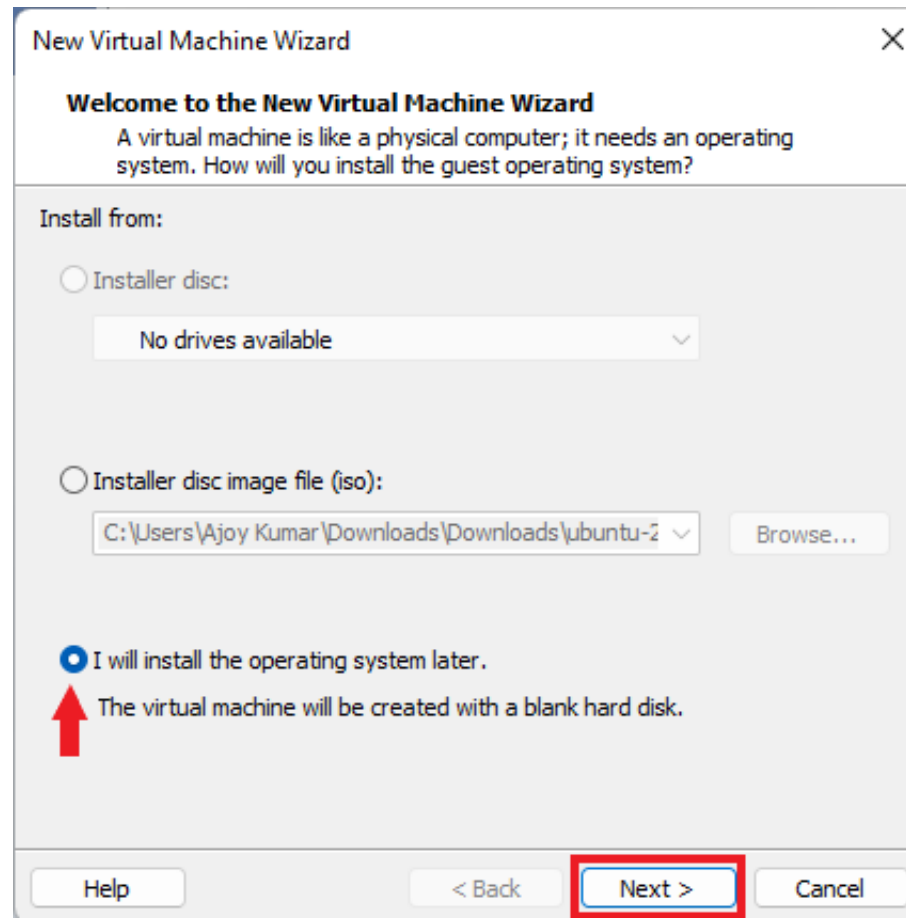
The diagram:



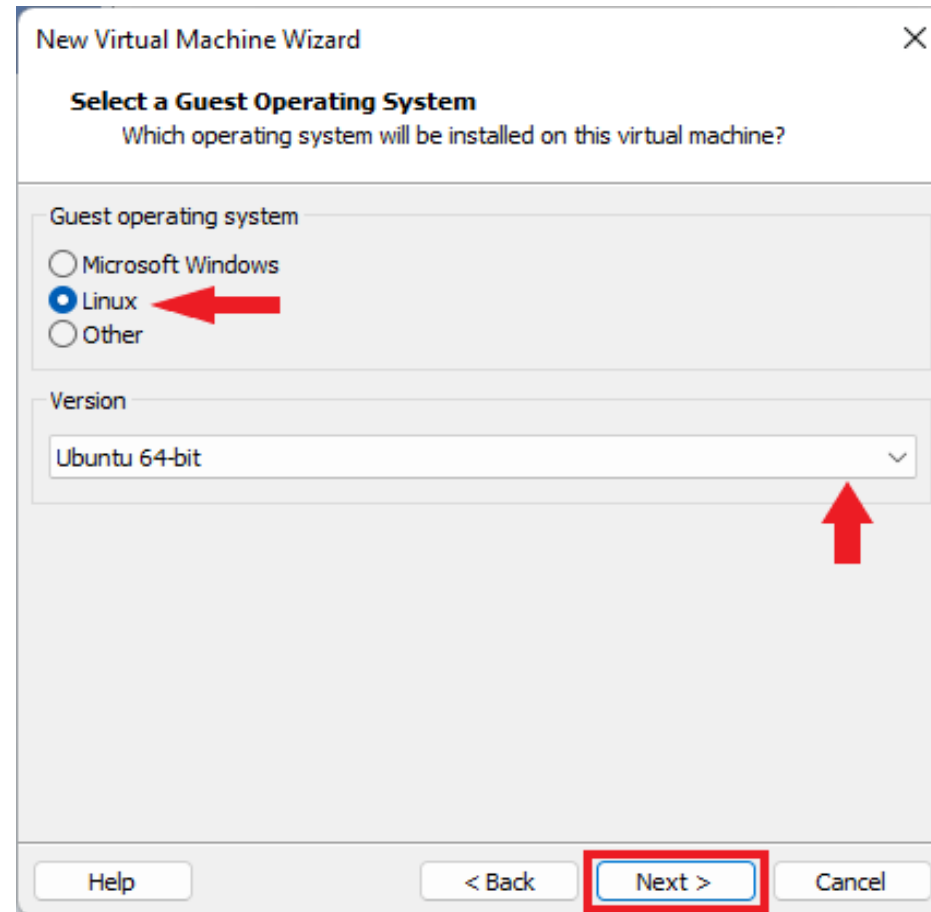
Installing Linux By Virtualization (I)



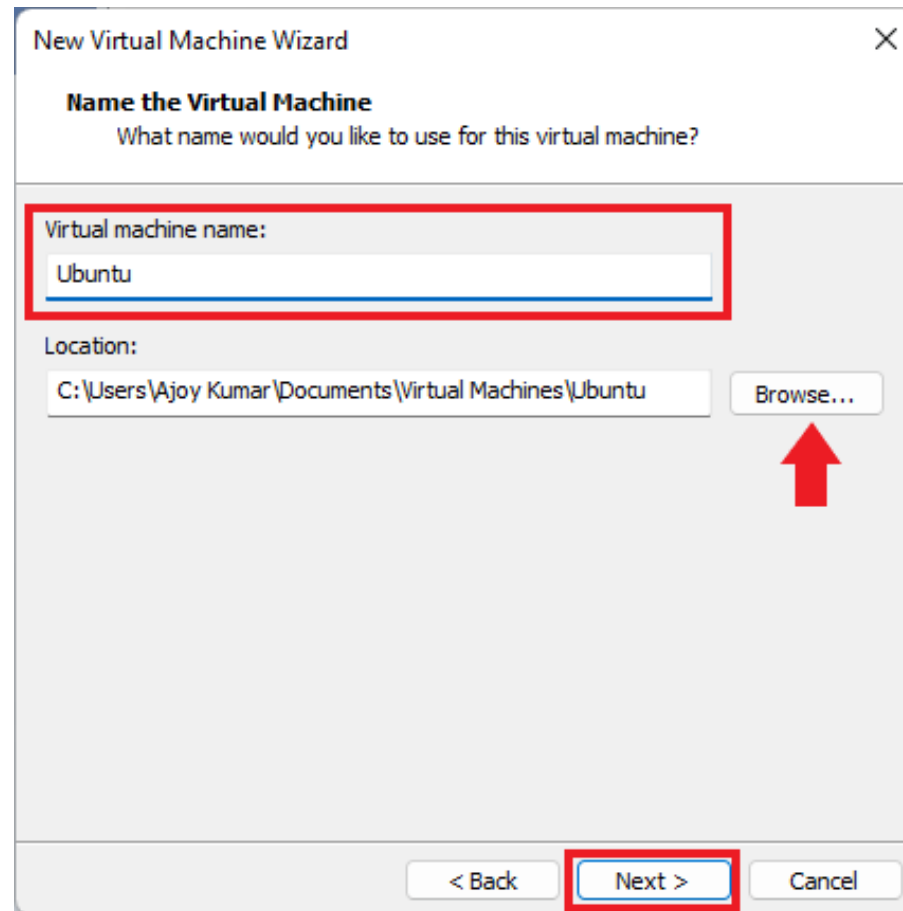
Installing Linux By Virtualization (II)



Installing Linux By Virtualization (III)



Installing Linux By Virtualization (IV)



Installing Linux By Virtualization (V)

The screenshot shows the 'New Virtual Machine Wizard' dialog box, specifically the 'Specify Disk Capacity' step. The title bar reads 'New Virtual Machine Wizard' with a close button. Below the title, the section is 'Specify Disk Capacity' with the subtitle 'How large do you want this disk to be?'. A paragraph explains: 'The virtual machine's hard disk is stored as one or more files on the host computer's physical disk. These file(s) start small and become larger as you add applications, files, and data to your virtual machine.' Below this, there is a text input field for 'Maximum disk size (GB):' with the value '50' and up/down arrows. A red rectangle highlights this field. Below the input field, it says 'Recommended size for Ubuntu 64-bit: 20 GB'. There are two radio button options: 'Store virtual disk as a single file' (unselected) and 'Split virtual disk into multiple files' (selected). A red arrow points to the 'Split virtual disk into multiple files' option. Below the selected option, a note states: 'Splitting the disk makes it easier to move the virtual machine to another computer but may reduce performance with very large disks.' At the bottom, there are four buttons: 'Help', '< Back', 'Next >', and 'Cancel'. The 'Next >' button is highlighted with a red rectangle.

New Virtual Machine Wizard

Specify Disk Capacity
How large do you want this disk to be?

The virtual machine's hard disk is stored as one or more files on the host computer's physical disk. These file(s) start small and become larger as you add applications, files, and data to your virtual machine.

Maximum disk size (GB): 50

Recommended size for Ubuntu 64-bit: 20 GB

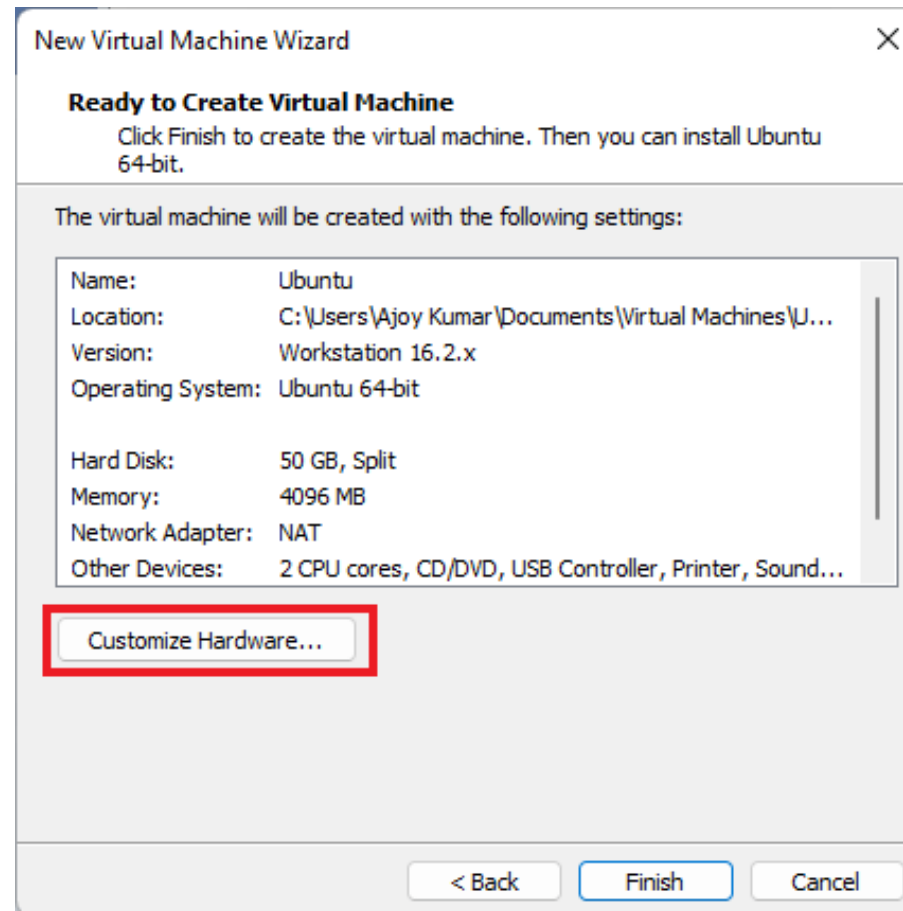
☐ Store virtual disk as a single file

☒ Split virtual disk into multiple files

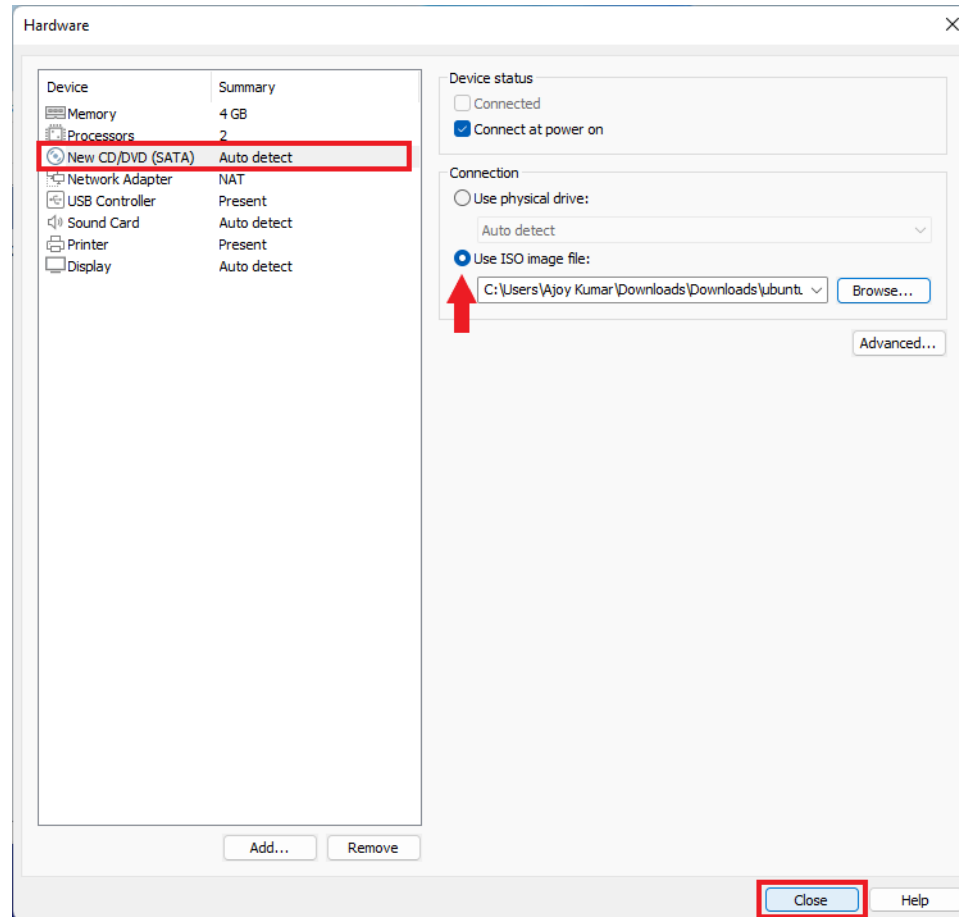
Splitting the disk makes it easier to move the virtual machine to another computer but may reduce performance with very large disks.

Help < Back Next > Cancel

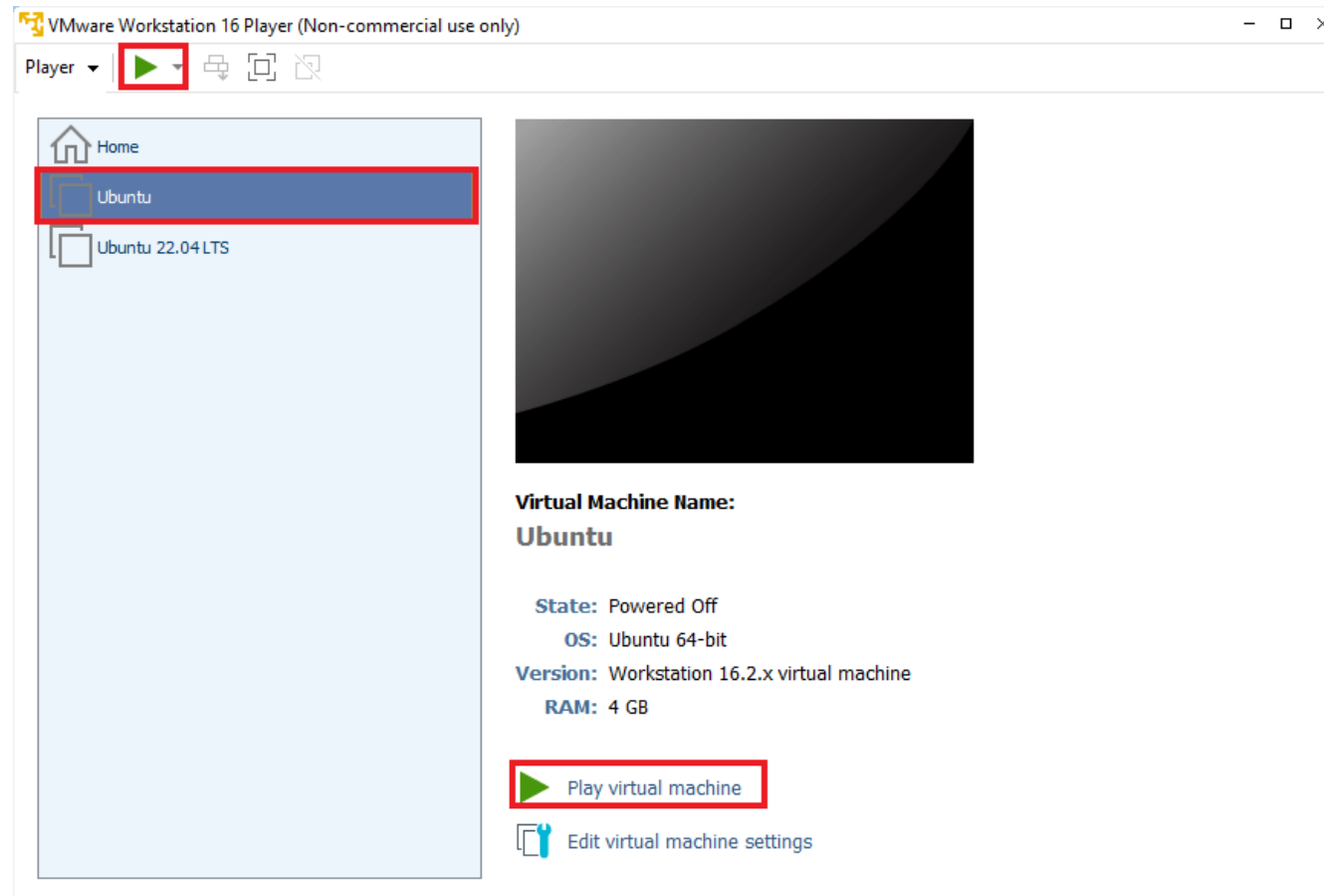
Installing Linux By Virtualization (VI)



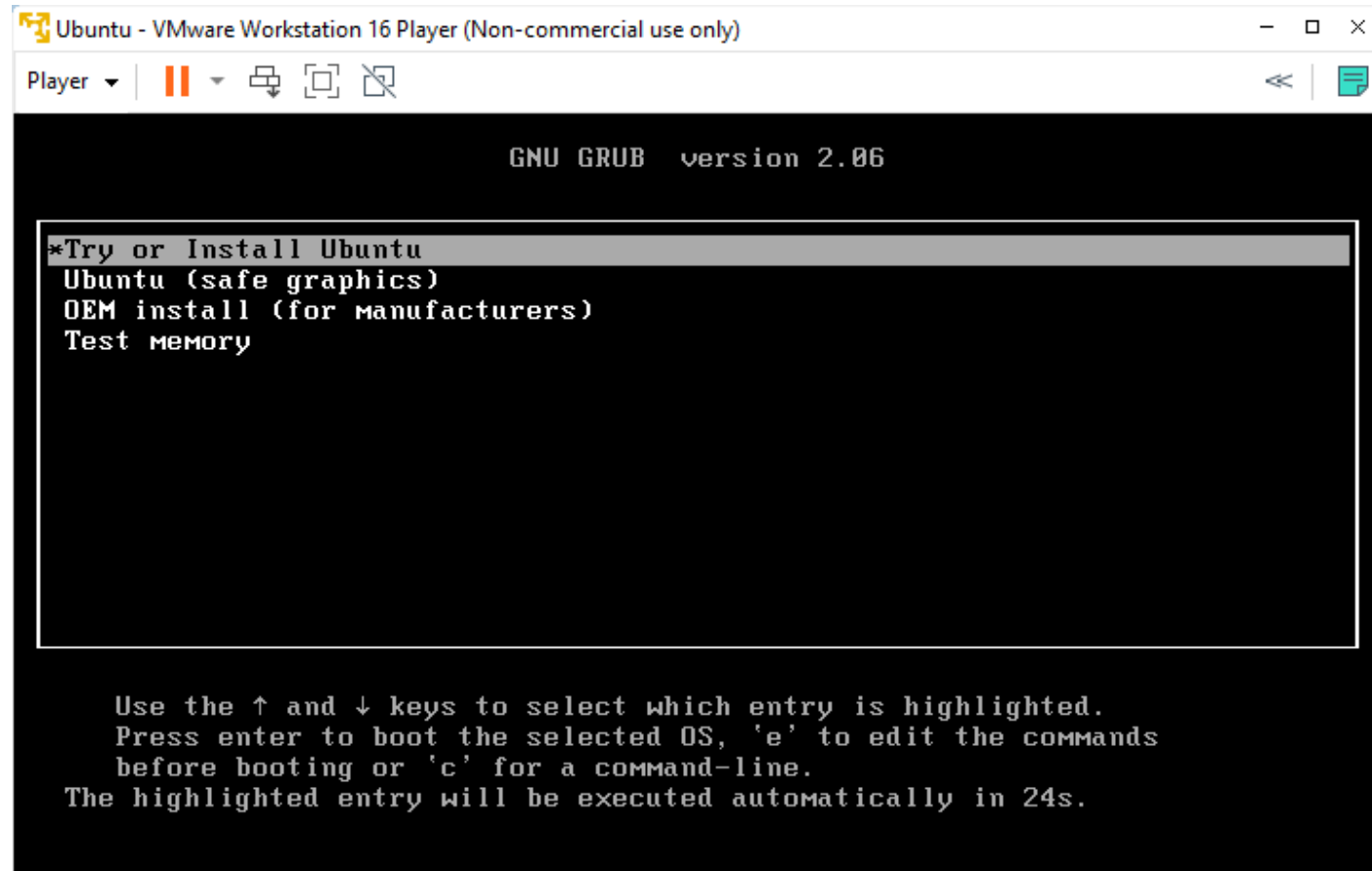
Installing Linux By Virtualization (VII)



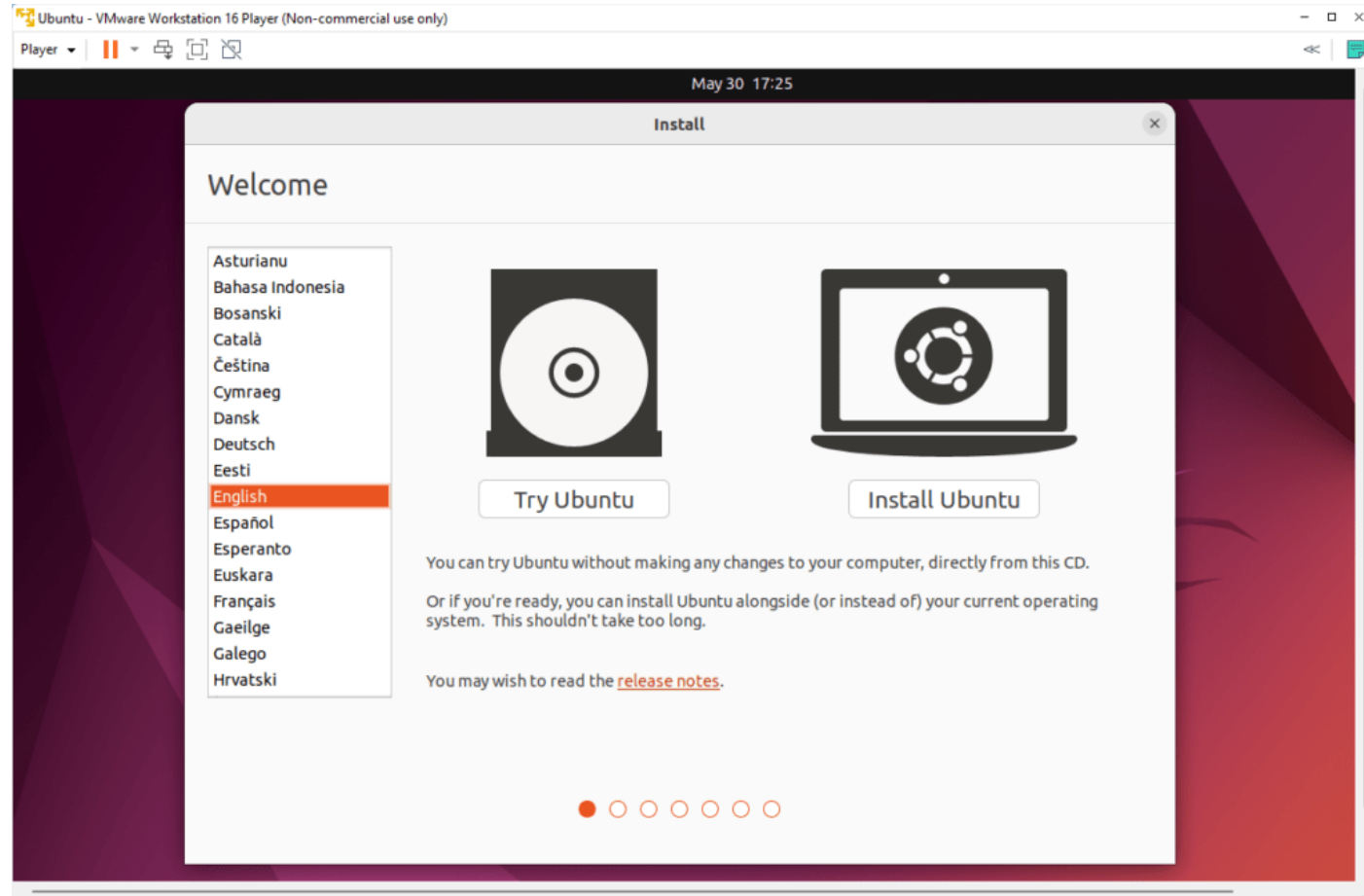
Installing Linux By Virtualization (VIII)



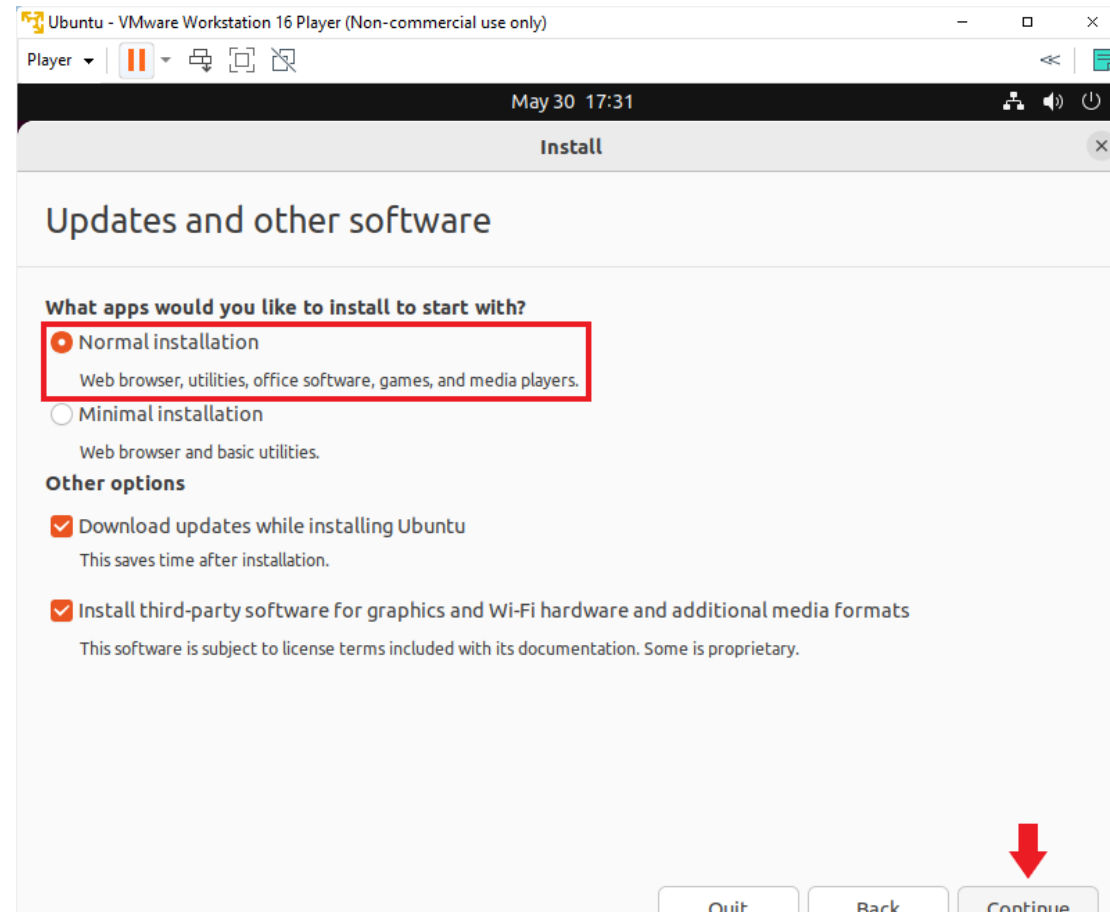
Installing Linux By Virtualization (IX)



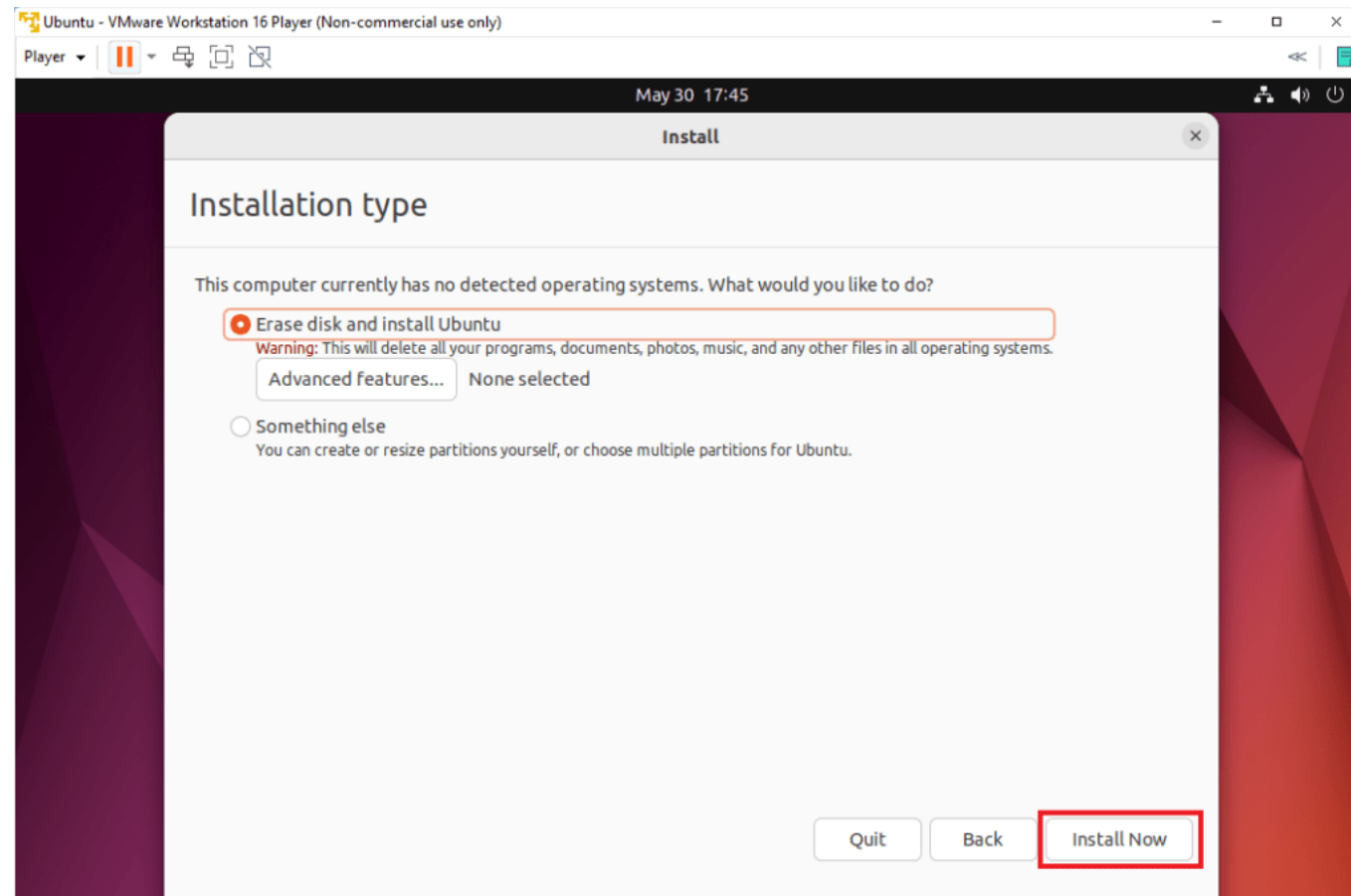
Installing Linux By Virtualization (X)



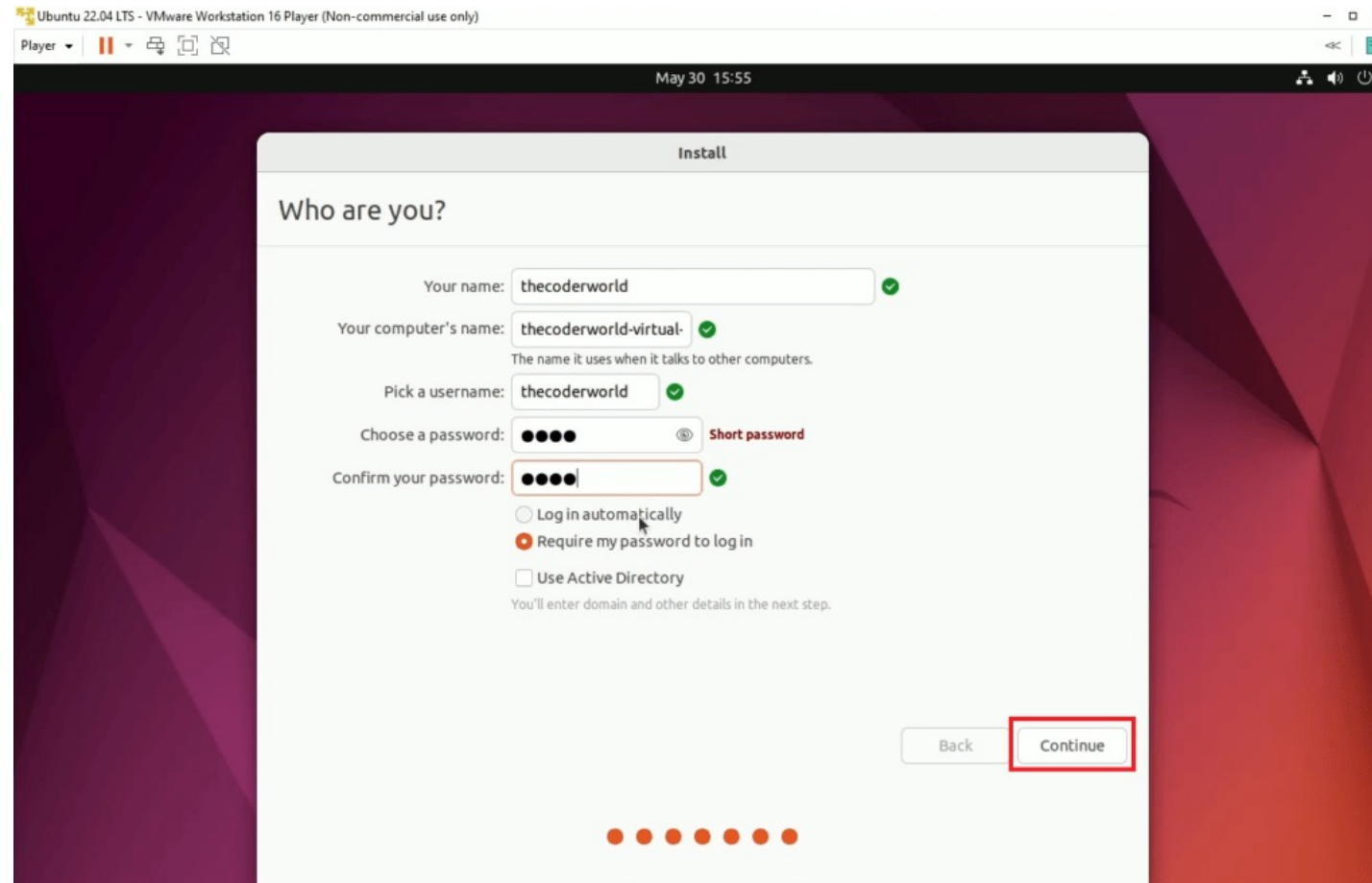
Installing Linux By Virtualization (XI)



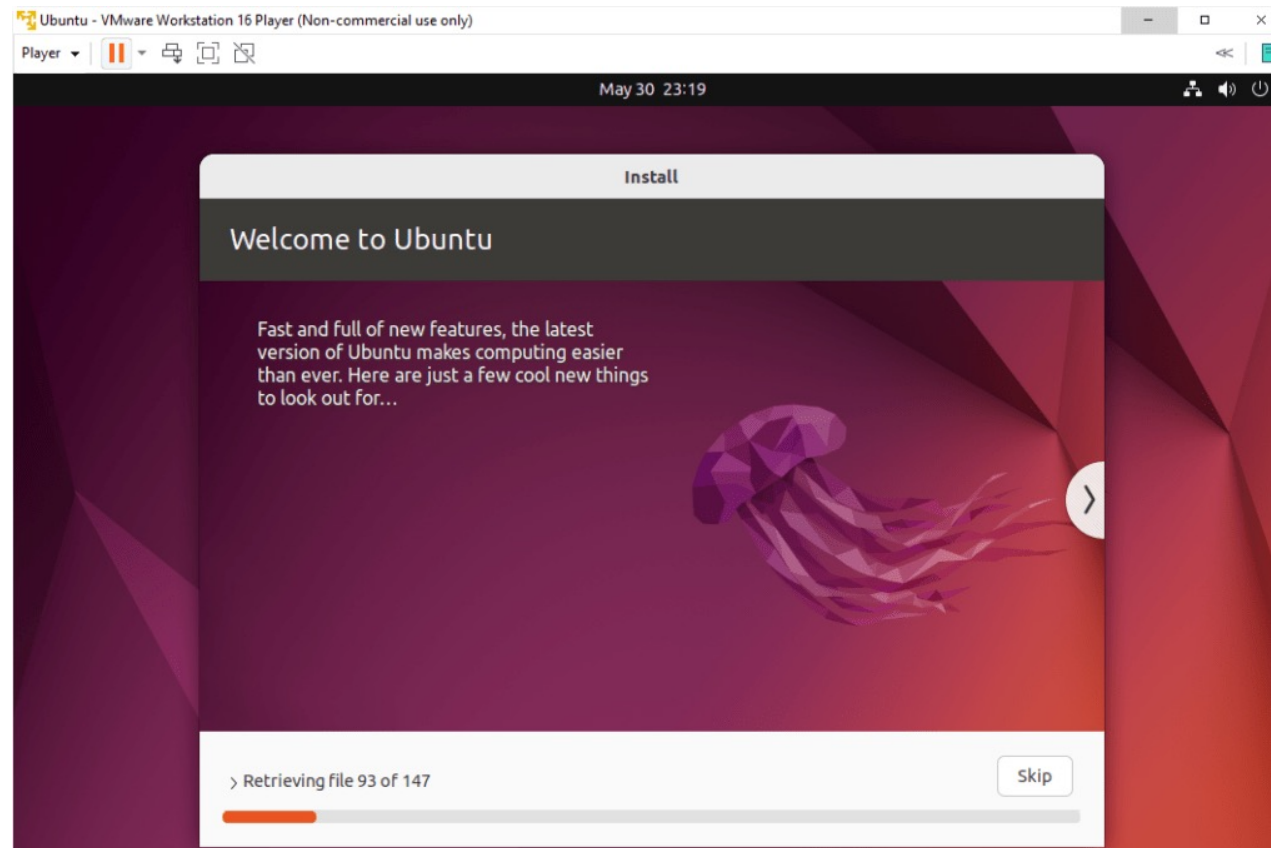
Installing Linux By Virtualization (XII)



Installing Linux By Virtualization (XIII)



Installing Linux By Virtualization (XIII)

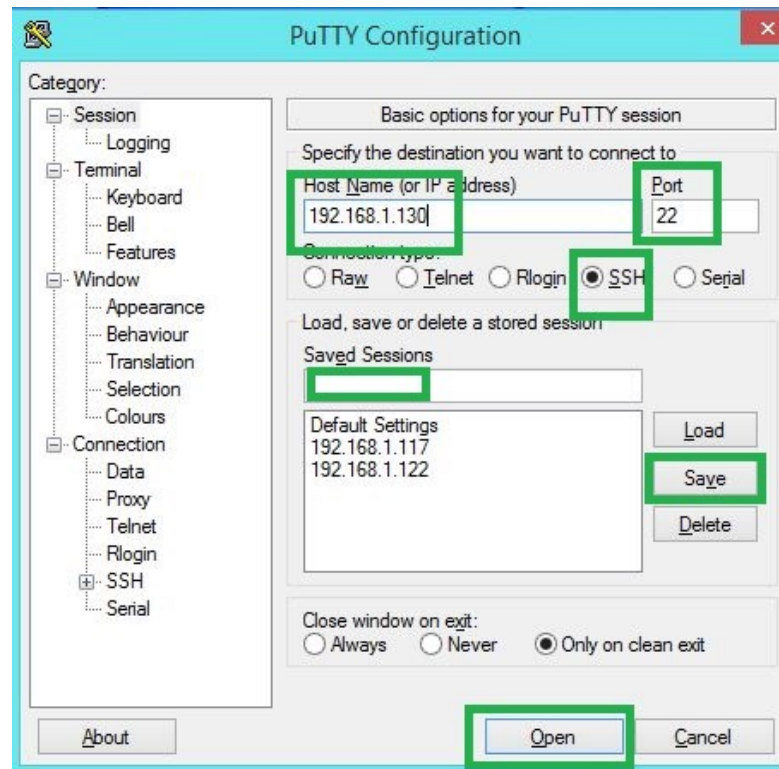


Remote access to a linux server (I)

- Usually is done via SSH
- The SSH server can be installed like this:
 - `sudo apt-get install openssh-server` // in Ubuntu
 - `yum install openssh-server` // In RedHat, CentOS
- Start the SSH server:
 - `sudo service ssh restart` // in Ubuntu
 - `service sshd start` //in Redhat, CentOS
- Download a terminal emulator client:
 - putty or Ericom Interconnect

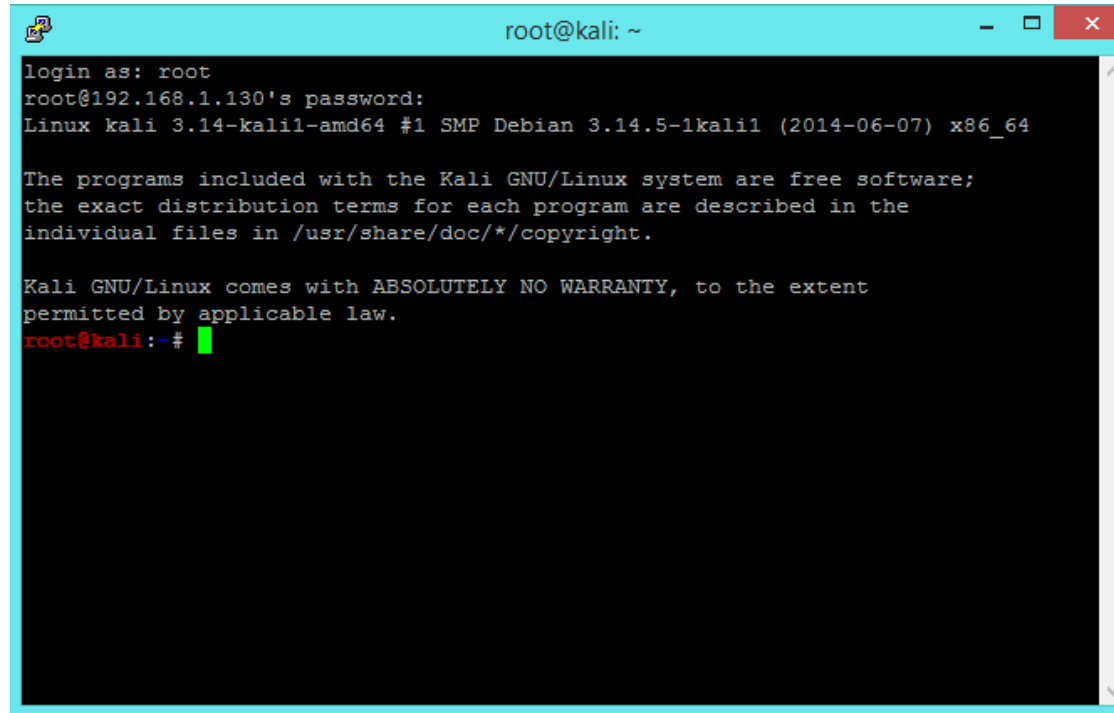
Remote access to a linux server (II)

Run putty, enter the hostname/IP, the port (default is 22) and hit “Open”.



Remote access to a linux server (III)

Enter the user/password and you are connected to the Linux BASH environment



```
root@kali: ~  
login as: root  
root@192.168.1.130's password:  
Linux kali 3.14-kali1-amd64 #1 SMP Debian 3.14.5-1kali1 (2014-06-07) x86_64  
  
The programs included with the Kali GNU/Linux system are free software;  
the exact distribution terms for each program are described in the  
individual files in /usr/share/doc/*/copyright.  
  
Kali GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent  
permitted by applicable law.  
root@kali:~#
```

BASH – the Linux shell

- BASH is a programming/scripting language
- BASH shell is the Linux equivalent of the Windows **cmd**
- BASH is a command processor that typically runs in a text window, where the user types commands that cause actions
- BASH runs scripts (python, perl, etc)
- It has been ported to Windows (via Cygwin)

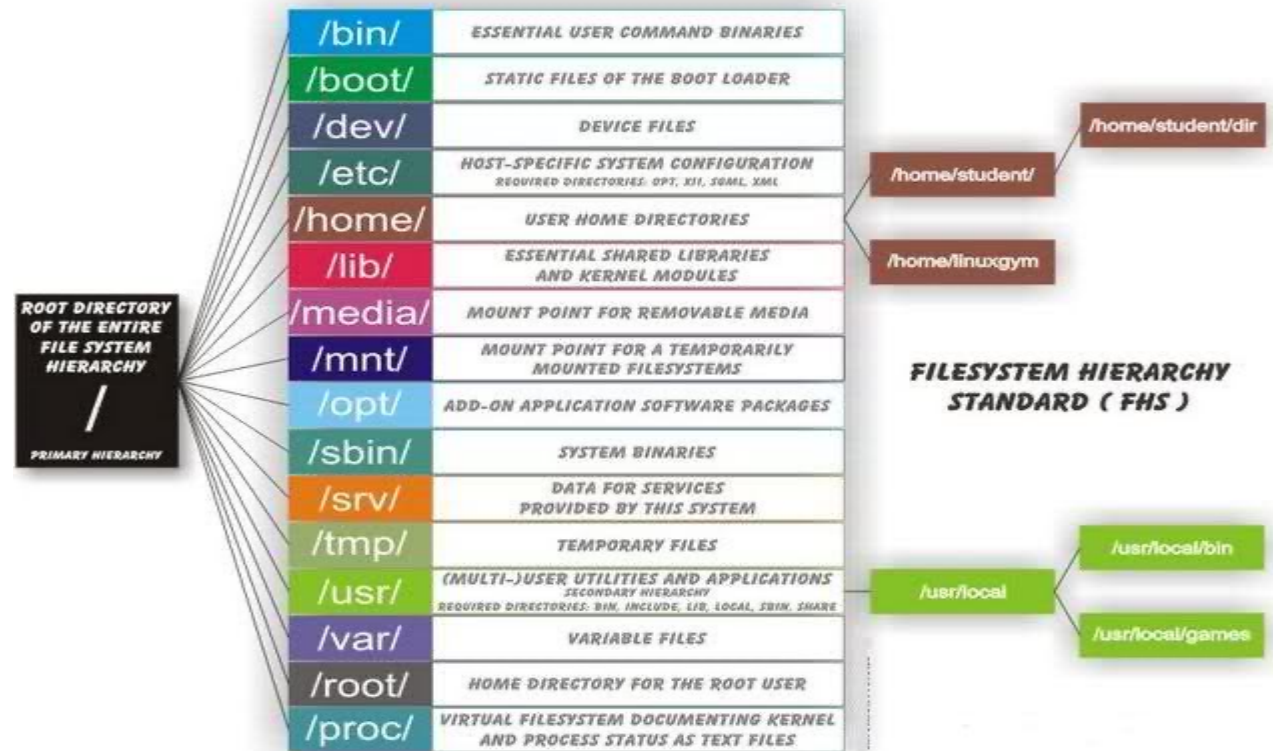


BASH – the golden rule

- When you do not know what a command does:
 - **man** – stands for manual
 - man ls
 - man cd
 - man grep
 - etc,

Linux directory hierarchy

- In Windows we call them folders, in Linux the term used is **directory/directories**.



Linux directory hierarchy (II)

- The equivalent of the “C:\” partition in Windows is referred in Linux as “/” – also called “root directory”, or “slash”.
- The Linux filesystem has the root directory at the top of the directory tree.
- The following list of directories are subdirectories of the root directory. This directory is denoted by the / (pronounced "slash") symbol.
 - **/bin:**
Contains executable programs such as ls (“dir” in Windows) and cp (“copy” in Windows). These programs are designed to make the system usable.
 - **/etc**
Contains configuration files which are local to the machine. Programs store configuration files in this directory and these files are referenced when programs are run.
 - **/home**
Contains user account directories. Each user created by the system administrator will have a subdirectory under /home with the name of the account. This is the default behaviour of Linux systems. E.g. User account for Anna is created, her home directory will be located in /home/anna.

Linux directory hierarchy (III)

- **/mnt**
Used for mounting temporary filesystems. When mounting a CD-ROM for instance, the standard mount point location is /mnt/cdrom.
- **/opt**
Used for storing random data that has no other logical destination.
- **/proc**
Provides information about running processes and the kernel. A directory is provided for each running process. Useful system information such as the amount of Random Access Memory (RAM) available on the system as well as Central Processing Unit (CPU) speed in Megahertz (MHz) can be found within the /proc directory.
- **/root**
This is the home directory for the super user (root). This directory is not viewable from user accounts. The /root directory usually contains system administration files.
- **/sbin**
Similar to /bin, this directory contains executable programs needed to boot the system, however the programs within /sbin are executed by the root user.
- **/tmp**
This directory is used for temporary storage space. Files within this directory are often cleaned out either at boot time or by a regular job process.

Linux directory hierarchy (IV)

- **/usr**

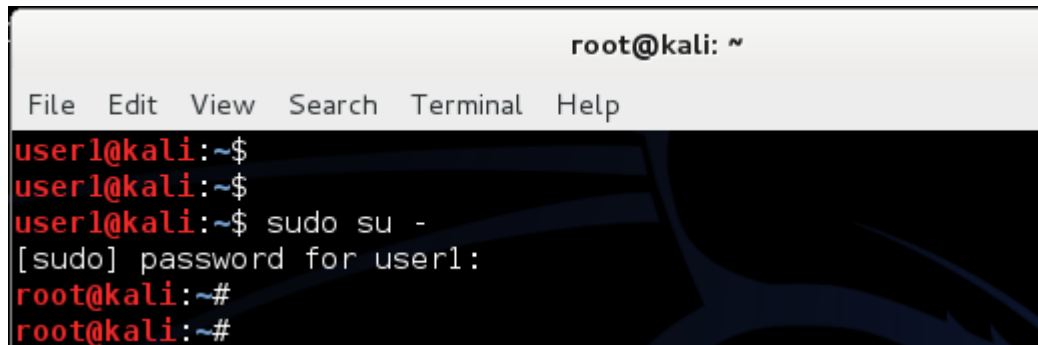
Used to store applications. When installing an application on a Debian GNU/Linux machine, the typical path to install would be /usr/local. You will notice the directory structure within /usr appears similar to the root directory structure.

- **/var**

This directory contains files of variable file storage. Files in /var are dynamic and are constantly being written to or changed. This is the directory where websites are usually stored in.

Users and groups

- Similar to Windows:
 - Linux has limited access users and, by default, one administrator (called “**root**”)
 - **root** is the user name or account that by default has access to all commands and files on Linux.
 - It is also referred to as the root account, root user and the superuser.
 - You can grant root like access to limited users using **sudo** (see “Run as Administrator in Windows”)

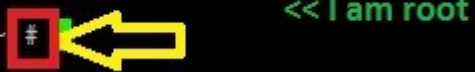
A terminal window titled 'root@kali: ~' with a menu bar containing 'File', 'Edit', 'View', 'Search', 'Terminal', and 'Help'. The terminal shows a sequence of commands: 'user1@kali:~\$' (twice), 'user1@kali:~\$ sudo su -', '[sudo] password for user1:', 'root@kali:~#', and 'root@kali:~#'. The background of the terminal has a dark blue abstract pattern.

```
root@kali: ~
File Edit View Search Terminal Help
user1@kali:~$
user1@kali:~$
user1@kali:~$ sudo su -
[sudo] password for user1:
root@kali:~#
root@kali:~#
```

Users and groups (II)

- With **sudo**, as a limited permissions user, you can be granted, temporarily, administrator/root access to execute commands usually restricted to only the root user.
- **sudo** is used in Linux Debian derivatives distros (Ubuntu, Kali Linux, etc) – but not limited to only Debian
- **sudo** can be installed on any Linux system
- Not every user can use sudo. That user must be present in the `/etc/sudoers` file
- In the BASH environment/the linux shell, the root user can be recognized by
 - the pound sign (**#**). Limited users can be recognized by the “\$” sign after their name.
 - When not sure about the user you are currently logged in, issue the **whoami** command

```
[root@fcsteaua ~]#  
[root@fcsteaua ~]# whoami  
root  
[root@fcsteaua ~]#
```



```
[ericom@fcsteaua ~]$ id ericom  
uid=10004(ericom) gid=10004(ericom)  
[ericom@fcsteaua ~]$
```

I am not root.
No # sign after my
name

Users and groups (III)

- All users have:
 - user IDs (**uid**), group IDs (**gid**).
 - The **uid** and **gid** are always decimal numbers and start from 1000 or 10000
 - The root superuser usually has **uid** and **gid** 0 (zero)
 - A specific user can be member of multiple groups.
- The **id** command show all the information you need to know about a user
- Try issuing the **id root** command and see what happens

```
[root@fcsteaua ~]# id cristi
uid=10003(cristi) gid=10003(cristi) groups=10003(cristi)
[root@fcsteaua ~]#
```

Users and groups (IV)

- How do I **add a new user** via the linux shell?
 - `useradd test -p 12345678`
 - The command above created a new user called `ericom` with the password `12345678`
- How do I **assign a user to another group**?
- `usermod -G root cristian`
- I added the **user test** to the **root group**.

Users and groups (V)

- Create a new group:

```
[root@fcsteaua ~]# groupadd connect-group  
[root@fcsteaua ~]# cat /etc/group | grep connect  
connect-group:x:10005:  
[root@fcsteaua ~]#
```

- Delete a group:

```
[root@fcsteaua ~]# groupdel connect-group  
[root@fcsteaua ~]# cat /etc/group | grep connect  
[root@fcsteaua ~]#
```

Users and groups (VI)

- Change the password of a user with the **passwd** command:

```
[root@fcsteaua ~]# passwd ericom
Changing password for user ericom.
New password:
Retype new password:
passwd: all authentication tokens updated successfully.
[root@fcsteaua ~]#
```

- Login as root if you are changing a password for an account different than yours
 - If you are logged in with a limited user account, use the **su** command or **sudo su** to login as root

```
bash-4.1$ id cristi
uid=10003(cristi) gid=10003(cristi) groups=10003(cristi)
bash-4.1$ su root
Password:
[root@fcsteaua ~]# id
uid=0(root) gid=10004(ericom) groups=10004(ericom),0(root),1(s),4(adm),6(disk),10(wheel)
[root@fcsteaua ~]#
```

<< Limited user

<< become root

<< Confirming that I am the root superuser

Known Linux server applications

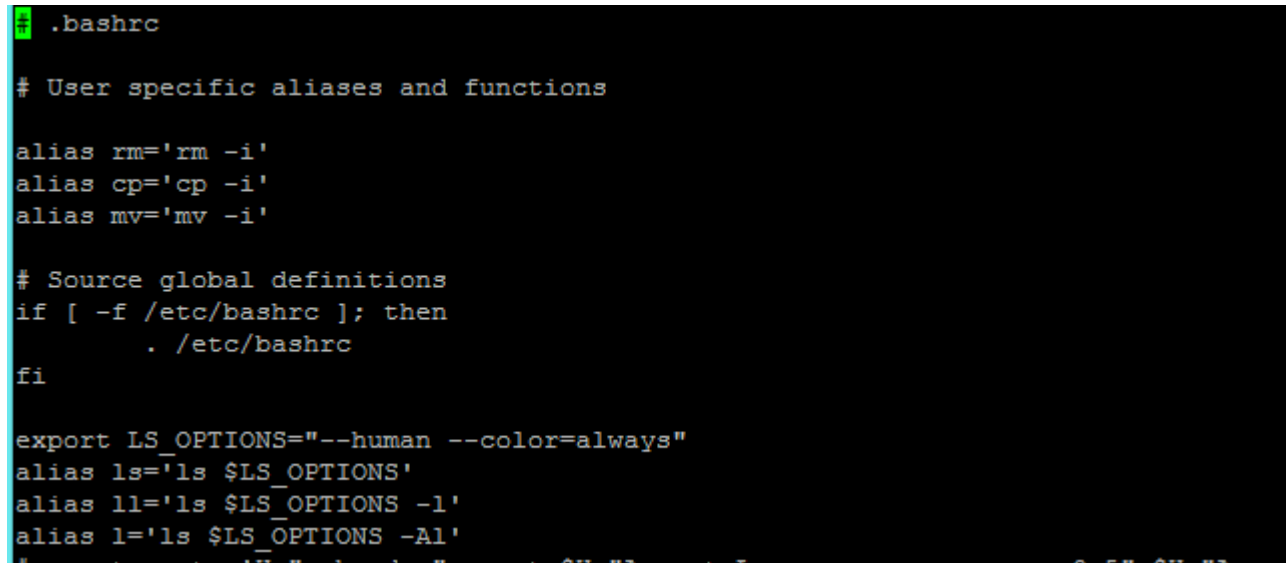
- HTTP server:
 - Apache (httpd), nginx
- SQL:
 - Mysql (mysqld), SQLite, postgresql
- FTP servers:
 - Proftpd, Pure-FTPd, vsFTPd, Filezilla
- DNS servers (Bind),
- Firewall (iptables, ipchains),
- SMTP servers (postfix, qmail, sendmail),
- POP3 / IMAP servers (Dovecot, Courier)
- Remote access server (OpenSSH)

Known Linux applications (I)

- **Text editors**

- vi

Vi is a powerful text editor included with most Linux systems, even embedded ones. Sometimes you'll have to edit a text file on a system that doesn't include a friendlier text editor, so knowing Vi is essential.

A screenshot of a terminal window with a black background and green text. The terminal shows the contents of the .bashrc file. The text is as follows:

```
.bashrc

# User specific aliases and functions

alias rm='rm -i'
alias cp='cp -i'
alias mv='mv -i'

# Source global definitions
if [ -f /etc/bashrc ]; then
    . /etc/bashrc
fi

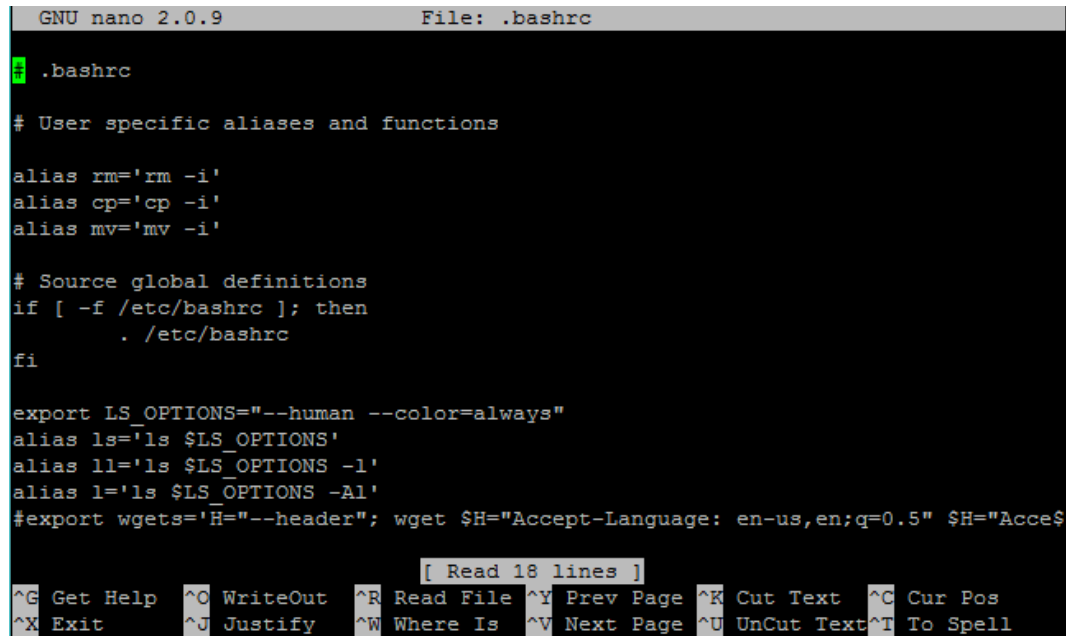
export LS_OPTIONS="--human --color=always"
alias ls='ls $LS_OPTIONS'
alias ll='ls $LS_OPTIONS -l'
alias l='ls $LS_OPTIONS -Al'
```

Known Linux applications (II)

■ Text editors

■ nano

nano is a small and friendly text editor. Besides basic text editing, nano offers many extra features like an interactive search and replace, go to line and column number.

A screenshot of the nano text editor interface. The title bar at the top shows "GNU nano 2.0.9" and "File: .bashrc". The main editing area has a black background with white text. It shows the beginning of the .bashrc file, including comments and aliases for rm, cp, mv, and ls. The bottom of the screen features a status bar with a line count "[Read 18 lines]" and a row of keyboard shortcuts: ^G Get Help, ^O WriteOut, ^R Read File, ^Y Prev Page, ^K Cut Text, ^C Cur Pos, ^X Exit, ^J Justify, ^W Where Is, ^V Next Page, ^U UnCut Text, and ^T To Spell.

```
GNU nano 2.0.9          File: .bashrc

.bashrc

# User specific aliases and functions

alias rm='rm -i'
alias cp='cp -i'
alias mv='mv -i'

# Source global definitions
if [ -f /etc/bashrc ]; then
    . /etc/bashrc
fi

export LS_OPTIONS="--human --color=always"
alias ls='ls $LS_OPTIONS'
alias ll='ls $LS_OPTIONS -l'
alias l='ls $LS_OPTIONS -Al'
#export wget='H="--header"; wget $H="Accept-Language: en-us,en;q=0.5" $H="Acce$

[ Read 18 lines ]
^G Get Help  ^O WriteOut  ^R Read File ^Y Prev Page ^K Cut Text  ^C Cur Pos
^X Exit      ^J Justify   ^W Where Is  ^V Next Page ^U UnCut Text ^T To Spell
```

END