

# Introduction to Linux

## What is Linux?

Linux is a **free and open-source operating system** based on Unix principles. It acts as an interface between **computer hardware** and **user applications**.

👉 In simple words: **Linux controls how hardware and software communicate.**

---

## Why Linux?

Linux is widely used because it is:

- **Free & Open Source** – No license cost, source code is available
  - **Stable** – Can run for years without reboot
  - **Secure** – Strong permission and user control
  - **Multi-user** – Multiple users can work at the same time
  - **Multitasking** – Can run many processes simultaneously
  - **Portable** – Runs on servers, desktops, mobiles, cloud, embedded systems
- 

## History of Linux

- Developed in **1991** by **Linus Torvalds**
- Inspired by **UNIX**
- Initially a personal project, later supported by a global developer community
- Today Linux powers:

- Servers
  - Cloud platforms (AWS, Azure, GCP)
  - Android devices
  - Supercomputers
- 

## Linux vs Windows (Quick Comparison)

Feature	Linux	Windows
Cost	Free	Paid
Source Code	Open Source	Closed Source
Security	Very High	Moderate
Customization	High	Limited
Stability	Excellent	Moderate
Usage	Servers, Cloud, DevOps	Desktop, Gaming

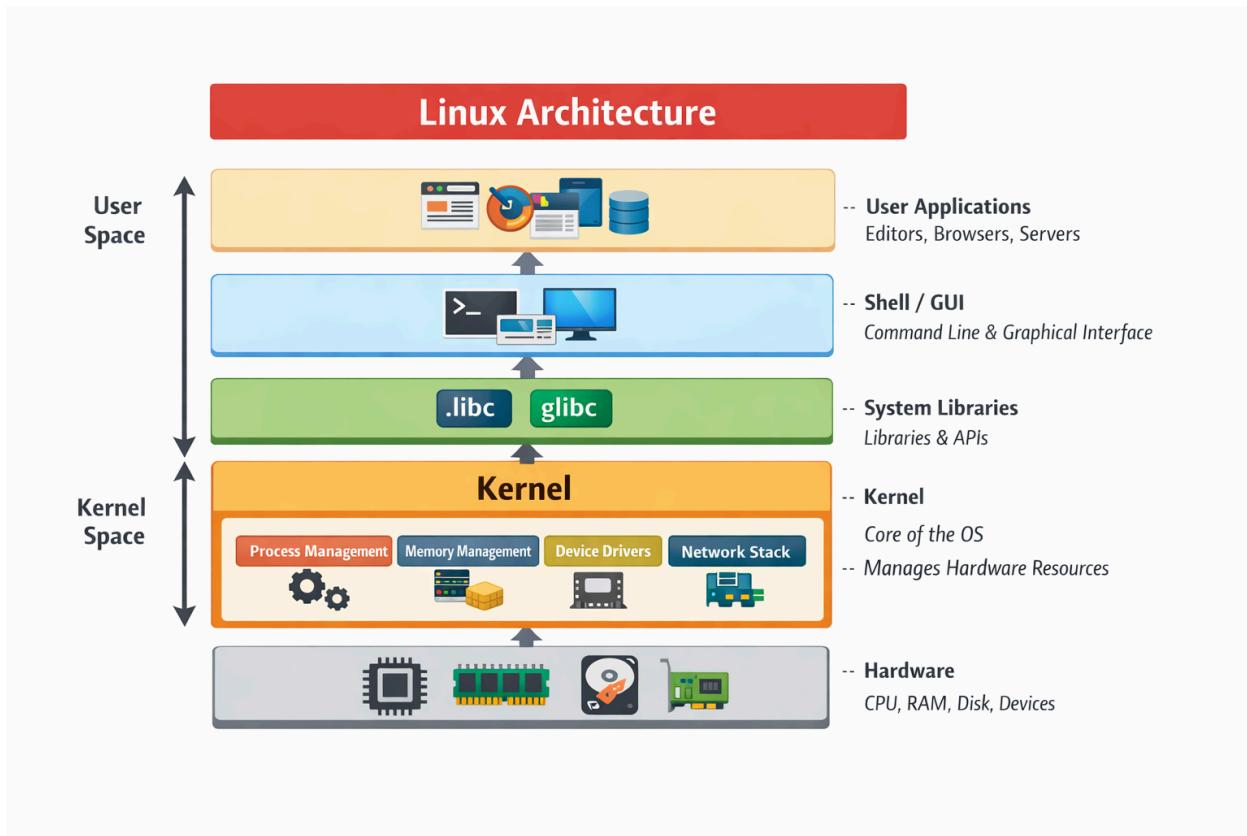
---

## Linux Architecture

Linux has a layered architecture:

1. **Hardware** – CPU, RAM, Disk
2. **Kernel** – Core of Linux (manages CPU, memory, devices)
3. **Shell** – Command interpreter (CLI)
4. **Applications** – Editors, browsers, servers

👉 **Kernel = Brain of Linux**



## 1. Hardware Layer

This is the **lowest layer** of Linux.

### Includes:

- CPU
- RAM
- Hard Disk / SSD
- Network Cards
- Keyboard, Mouse, Display

👉 Hardware **cannot be accessed directly** by users or applications.

## 2. Kernel (Core of Linux)

The **Kernel** is the heart of Linux.

It directly communicates with **hardware** and manages system resources.

### Responsibilities of the Kernel:

#### a) Process Management

- Creates, schedules, and terminates processes
- Uses **process IDs (PID)**
- Decides which process gets CPU time

Example:

```
ps  
top
```

---

#### b) Memory Management

- Allocates and deallocates RAM
- Manages virtual memory
- Uses **paging and swapping**

Example:

```
free -m  
vmstat
```

---

#### c) File System Management

- Manages files and directories

- Controls access permissions
- Supports multiple file systems:
  - ext4
  - xfs
  - nfs

Example:

```
mount  
df -h
```

---

#### d) Device Management

- Controls hardware using **device drivers**
- Treats hardware as files

Examples:

```
/dev/sda  
/dev/null
```

---

#### e) Network Management

- Handles network protocols
- Manages IP, TCP, UDP
- Supports firewall and routing

Example:

```
ip addr
```

`netstat`

---

## f) System Call Interface

- Acts as a **bridge between user space and kernel**
- Applications request services via system calls

Examples of system calls:

- `read()`
- `write()`
- `fork()`
- `exec()`

👉 Users never directly access hardware—everything goes through the kernel.

---

## 3. System Libraries

System libraries provide **standard functions** used by applications.

### Purpose:

- Simplify application development
- Translate application requests into kernel system calls

### Examples:

- GNU C Library (glibc)
- `libc`

Example:

```
ls
```

The `ls` command internally uses system libraries to interact with the kernel.

---

## 4. Shell (Command Interpreter)

The shell is the **interface between user and kernel**.

### Functions:

- Accepts user commands
- Interprets commands
- Sends requests to kernel

### Types of Shells:

- bash (Most common)
- sh
- zsh
- csh

Example:

```
echo $SHELL
```

👉 Shell is not part of the kernel.

---

## 5. GUI (Graphical User Interface)

Optional layer on top of Linux.

### **Examples:**

- GNOME
- KDE
- XFCE

### **Features:**

- Mouse-based interaction
- Windows, icons, menus
- User-friendly

👉 Servers usually **do not use GUI** for better performance.

---

## **6. User Applications**

These are programs used by users.

### **Examples:**

- Editors (vi, nano)
- Browsers
- Databases (MySQL)
- Servers (Apache, Nginx)
- DevOps tools

Applications **do not access hardware directly**.

---

# Description of Linux Architecture Diagram

The Linux architecture diagram represents the **layered structure of the Linux operating system**, showing how users and applications interact with hardware in a secure and organized manner.

At the **bottom layer** is the **Hardware**, which includes physical components such as **CPU, RAM, storage devices, and network interfaces**. Hardware cannot be accessed directly by users or applications.

Above the hardware layer lies the **Kernel**, which is the **core of the Linux operating system**. The kernel manages system resources and acts as a bridge between hardware and user-level programs. It is responsible for **process management, memory management, device drivers, file system management, and network communication**. All hardware access is controlled by the kernel.

On top of the kernel are the **System Libraries**, such as **glibc**, which provide standard functions and APIs required by applications. These libraries convert application requests into **system calls** that the kernel can understand.

Above the system libraries is the **Shell / GUI layer**, which acts as the **user interface**. The shell allows users to interact with the system using commands, while the GUI provides a graphical, mouse-based interface. The shell interprets user commands and passes them to the kernel for execution.

At the **topmost layer** are the **User Applications**, which include editors, browsers, servers, and utilities. These applications run in **user space** and rely on system libraries and the kernel to perform operations.

The diagram also highlights the separation between **User Space** and **Kernel Space**, which enhances **system security and stability** by preventing direct hardware access from user applications.

---

## Linux Distributions (Distros)

A Linux distribution = **Linux Kernel + Software + Package Manager**

Popular Linux distributions:

- Ubuntu
- CentOS / Rocky Linux

- Red Hat Enterprise Linux (RHEL)
- Debian
- Amazon Linux

👉 Different distros are designed for different purposes.

---

## Where is Linux Used?

Linux is used everywhere:

- Web servers
- Cloud & DevOps
- Cybersecurity
- Android OS
- IoT devices
- Supercomputers
- Database servers

---

## Linux Login Types

### 1. GUI (Graphical User Interface)

- Mouse based
- User-friendly

### 2. CLI (Command Line Interface)

- Terminal based

- Faster and powerful
  - Preferred by system administrators
- 

## Desktop Vs Server Edition

Linux operating systems are generally available in **Desktop** and **Server** editions, each designed for different use cases.

---

## Desktop Edition

The **Desktop edition** is designed for **end users** who require a graphical environment for daily tasks.

### Key Characteristics:

- Comes with a **Graphical User Interface (GUI)** by default
- Includes desktop applications such as:
  - Web browsers
  - Media players
  - Office tools
- Easier to use for beginners
- Higher resource usage (RAM, CPU, Disk)
- Suitable for:
  - Personal use
  - Learning basics
  - Development work

### **Examples of Use:**

- Personal computers
  - Laptops
  - Training systems
- 

## **Server Edition**

The **Server edition** is designed to run **server-side applications** and provide services to multiple users.

### **Key Characteristics:**

- Usually **no GUI** (CLI-based)
- Lightweight and optimized for performance
- Lower resource consumption
- More secure due to minimal packages
- Designed for **24/7 uptime**
- Supports server services like:
  - Web servers
  - Database servers
  - File servers
  - Application servers

### **Examples of Use:**

- Data centers

- Cloud environments
  - Production servers
- 

## Desktop vs Server – Comparison Table

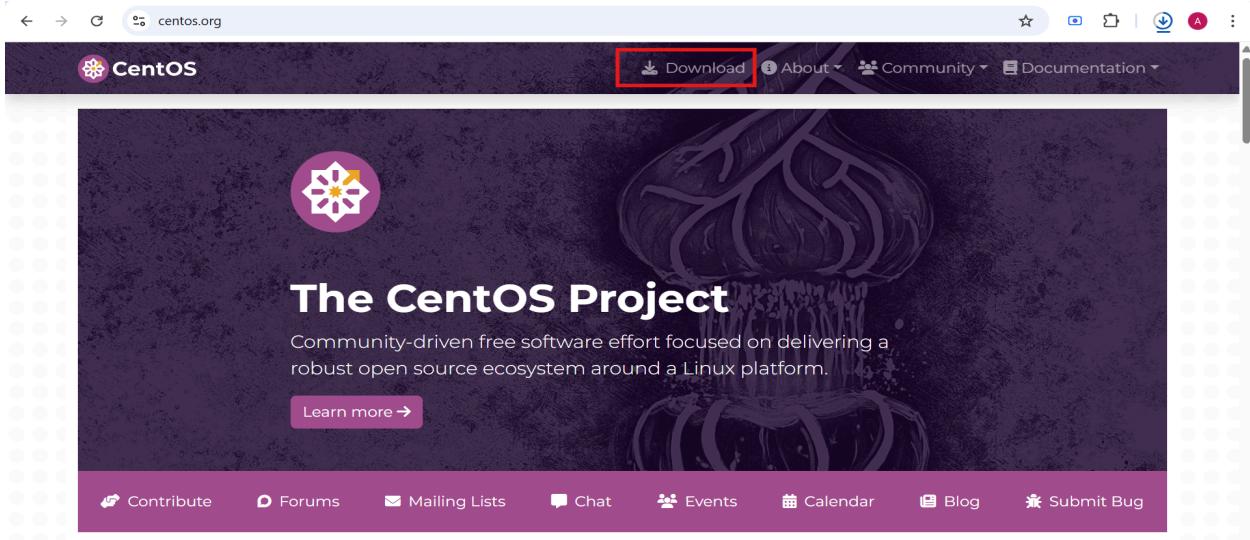
Desktop vs Server – Comparison Table

Feature	Desktop Edition	Server Edition
Interface	GUI + CLI	CLI only
Resource Usage	High	Low
Performance	Moderate	High
Security	Standard	High
Applications	User apps	Server services
Target Users	End users	System administrators
Uptime	Limited	24/7

---

## Steps to Download CentOS Stream 9 ISO

1. Open a web browser and visit the official CentOS website:  
<https://www.centos.org/>
2. On the homepage, click on the **Download** option.



3. From the available options, select **CentOS Stream 9**.

A screenshot of the 'Download' page for CentOS Stream 9. The URL in the address bar shows 'Home &gt; Download'. The main heading is 'Download'. Below it is a 'Table of Contents' section. The 'CentOS Stream' section is highlighted with a red box. It contains a subtext: 'Continuously delivered distro that tracks just ahead of Red Hat Enterprise Linux (RHEL) development, positioned as a midstream between Fedora Linux and RHEL. For anyone interested in participating and collaborating in the RHEL ecosystem, CentOS Stream is your reliable platform for innovation.' A navigation bar at the top includes 'Download' (highlighted with a red box), 'About', 'Community', and 'Documentation'. Below the navigation is a table for selecting system architecture. The table has columns for 'Architecture', 'ISOs', 'RPMS', 'Cloud', 'Containers', and 'Vagrant'. The first row, 'x86\_64', has its 'ISOs' link highlighted with a red box. The second row, 'ARM64 (aarch64)', also has its 'ISOs' link highlighted with a red box. The table header is circled with a red box.

4. Choose the system **Architecture** as **x86\_64**.
5. Click on the **ISO** image link to start the download

#### Direct CentOS 9 Latest Download Link:

[https://mirrors.tuna.tsinghua.edu.cn/centos-stream/9-stream/BaseOS/x86\\_64/iso/CentOS-Stream-9-latest-x86\\_64-dvd1.iso](https://mirrors.tuna.tsinghua.edu.cn/centos-stream/9-stream/BaseOS/x86_64/iso/CentOS-Stream-9-latest-x86_64-dvd1.iso)

The ISO file download will begin automatically.

# Steps to Download & Install Oracle VirtualBox on Windows

1. Open a web browser and visit the official Oracle VirtualBox download page:  
<https://www.virtualbox.org/wiki/Downloads>
2. Under **VirtualBox Platform Packages**, select **Windows hosts**.
3. The installer download will begin automatically.



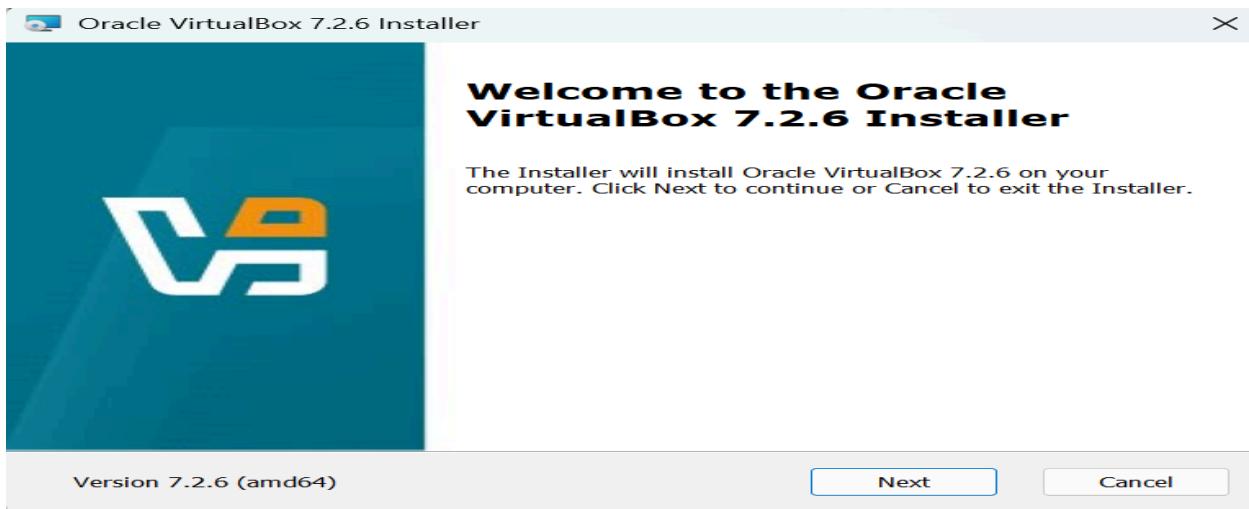
Once the download is complete, proceed with the installation.

4. Navigate to the **download location** on your system.
5. Select the **Oracle VirtualBox installer file** and **double-click** it to begin the installation.

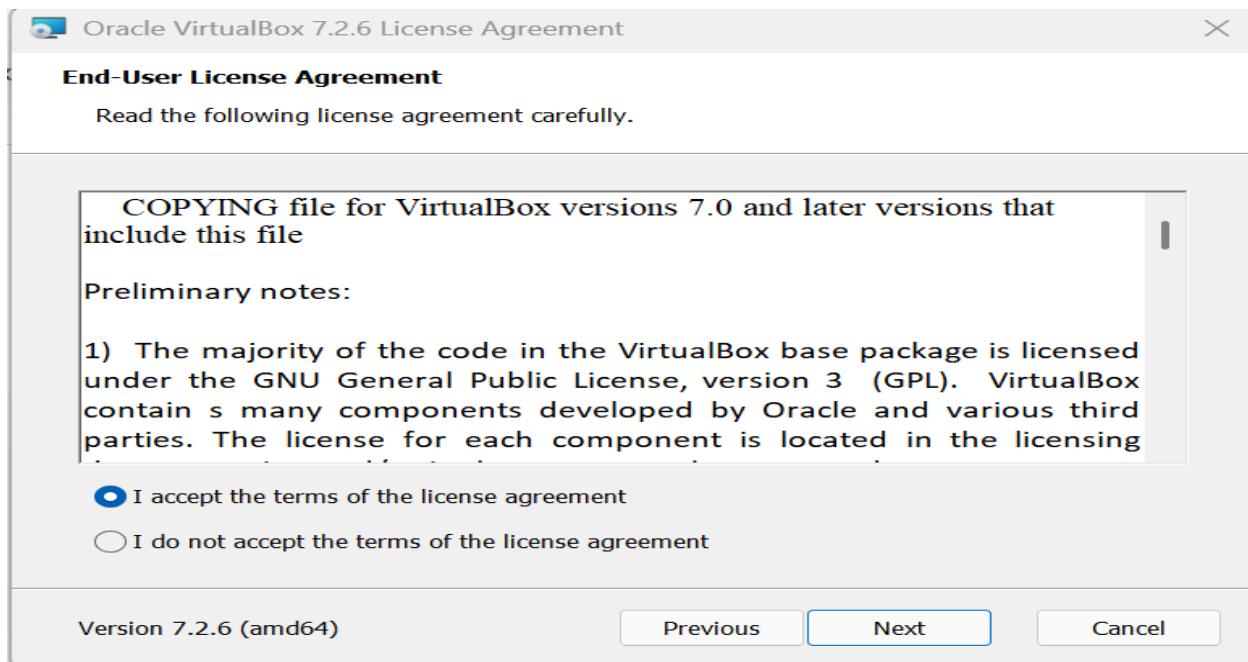
**Direct Oracle VirtualBox Download Link:**

<https://download.virtualbox.org/virtualbox/7.2.6/VirtualBox-7.2.6a-172322-Win.exe>

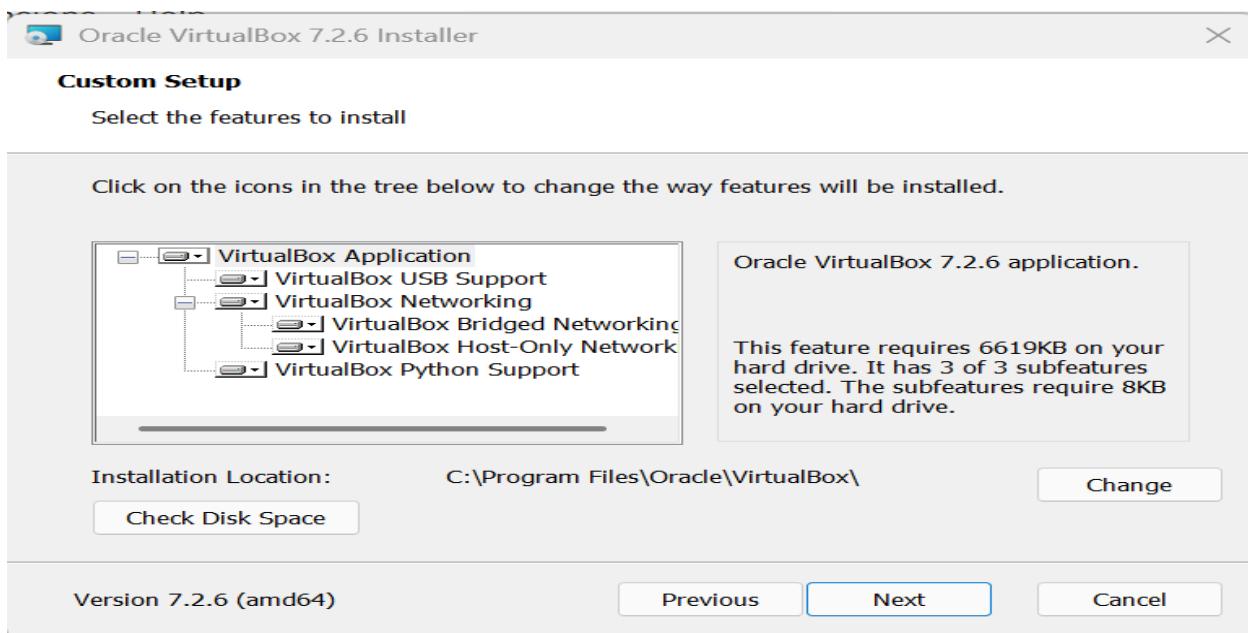
**Click Next:**



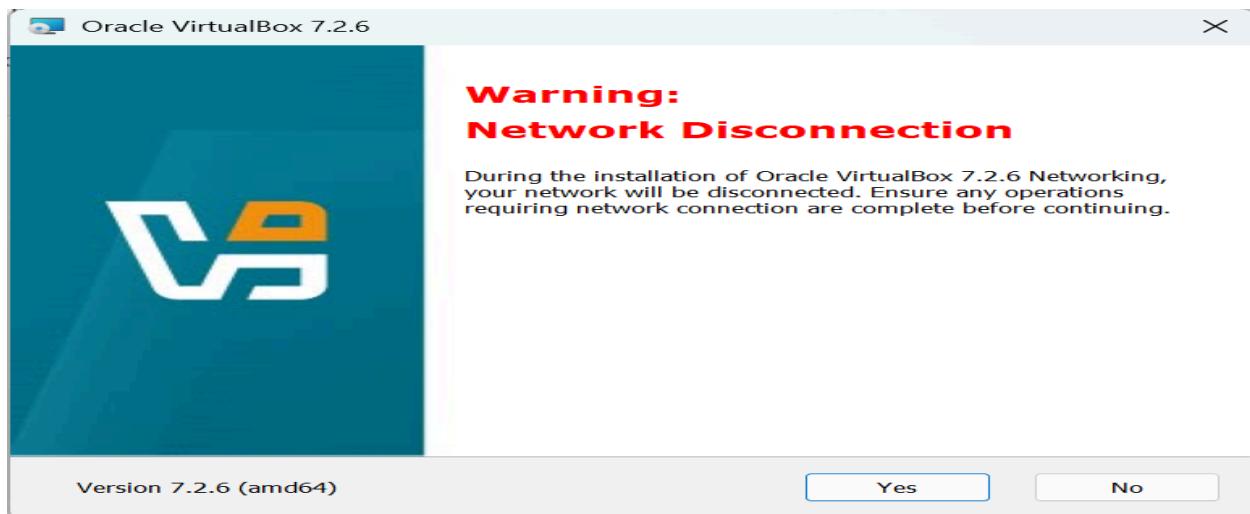
**Accept the terms & license:**



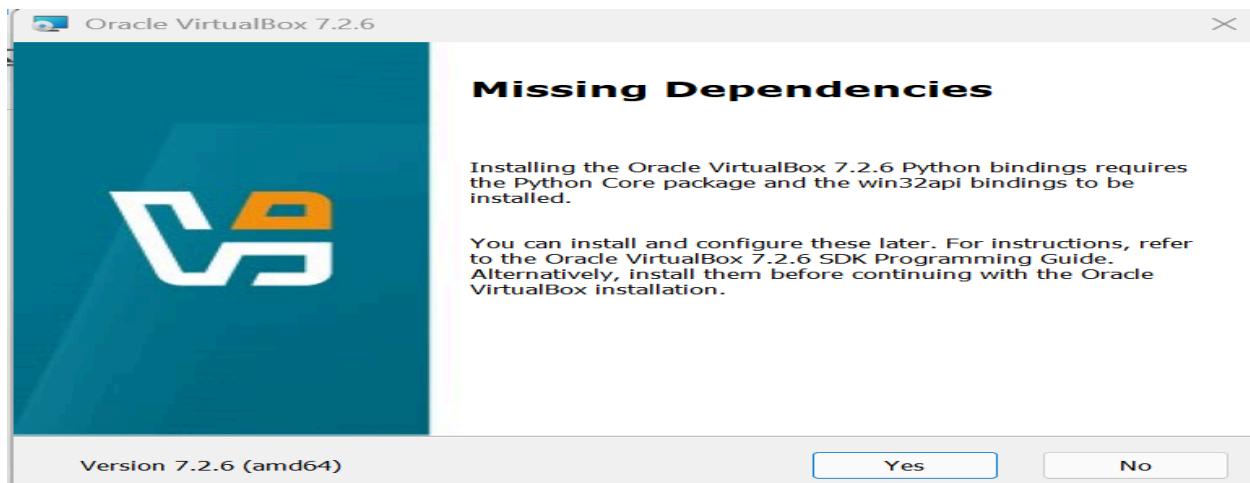
If you wanted to change the installation location click on change and select the desired Drive and Folder [Here we kept the default]:



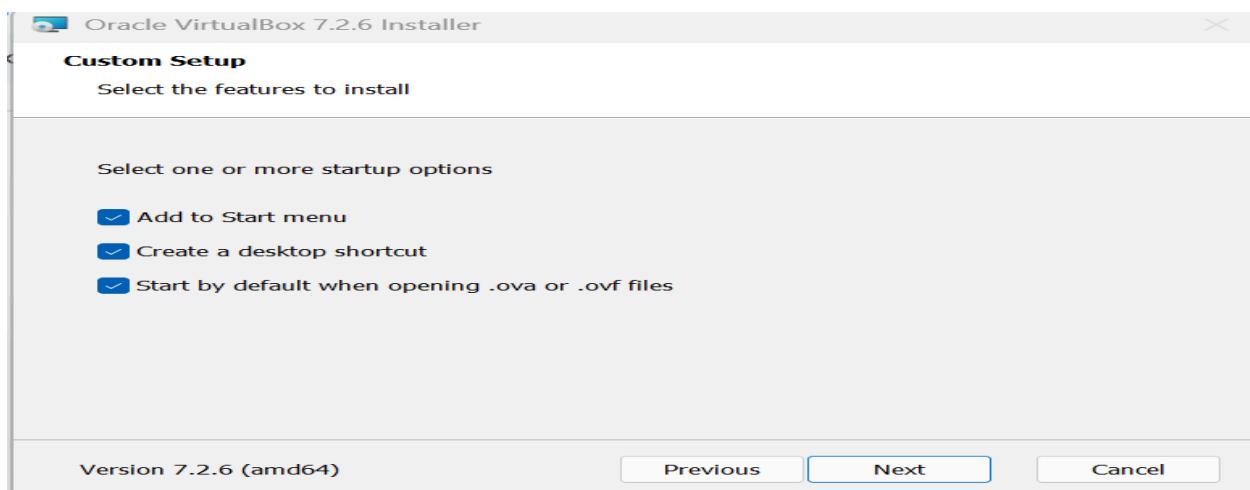
Click Yes:



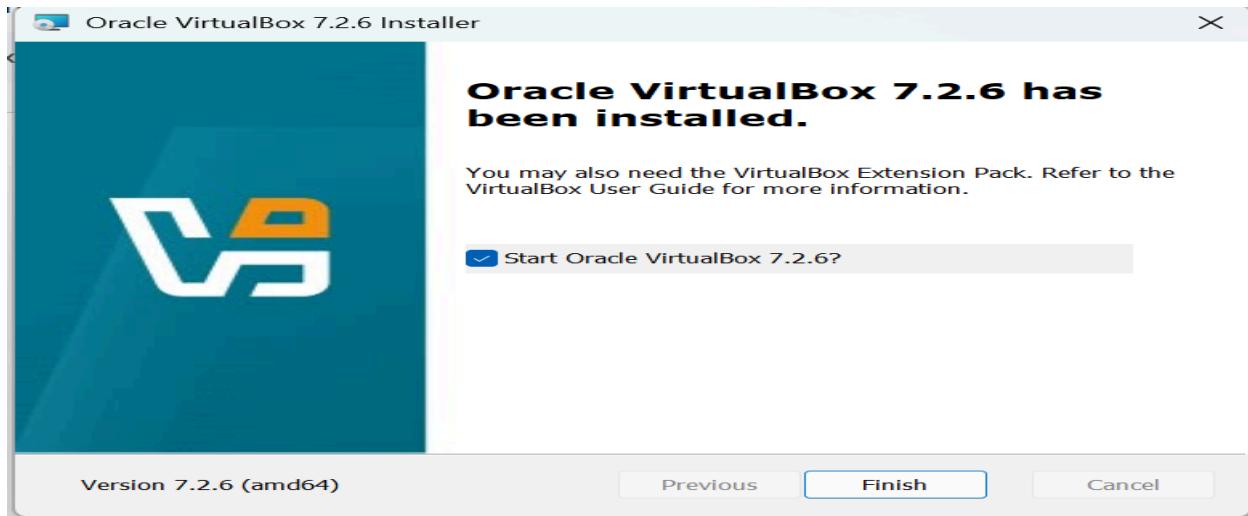
**Click Yes:**



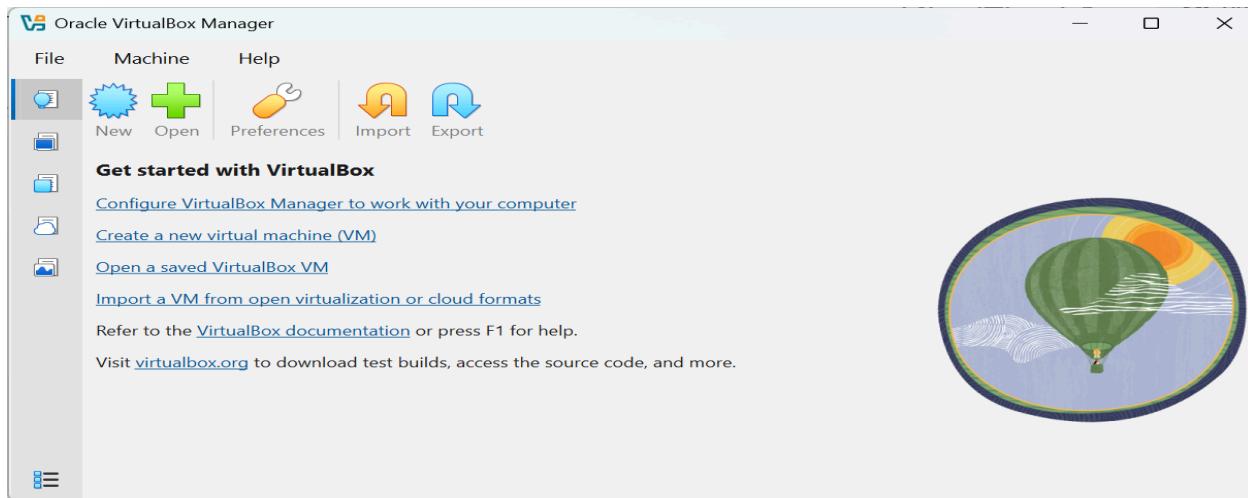
**Click Next -> Install**



**Click Finish:**



**Oracle VirtualBox:**



## **Creating Your First CentOS Stream 9 Virtual Machine Using Oracle VirtualBox**

This document explains how to create and install your first **CentOS Stream 9 Linux virtual machine** on a Windows system using Oracle VirtualBox.

## ◆ Prerequisites

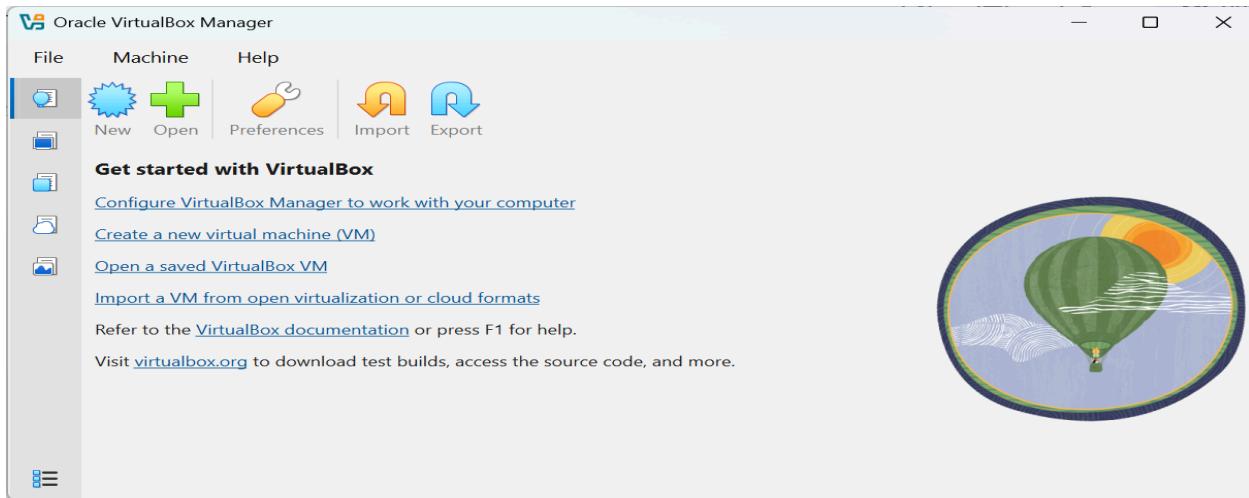
- Oracle VirtualBox installed on your system
  - CentOS Stream 9 ISO file downloaded
  - Minimum system RAM: 8 GB (recommended)
- 

## ★ Step-by-Step VM Creation & Installation

---

### Step 1: Open VirtualBox

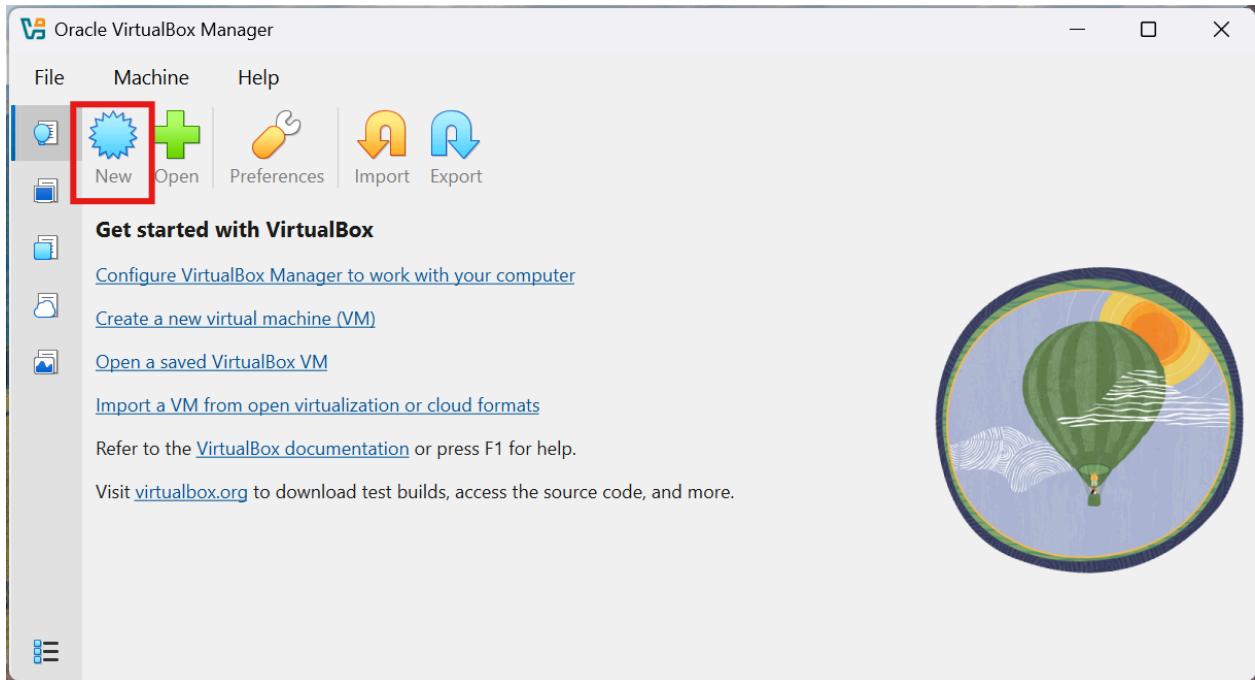
- Launch the **Oracle VirtualBox** application.



---

### Step 2: Create a New Virtual Machine

- Click on **New**.



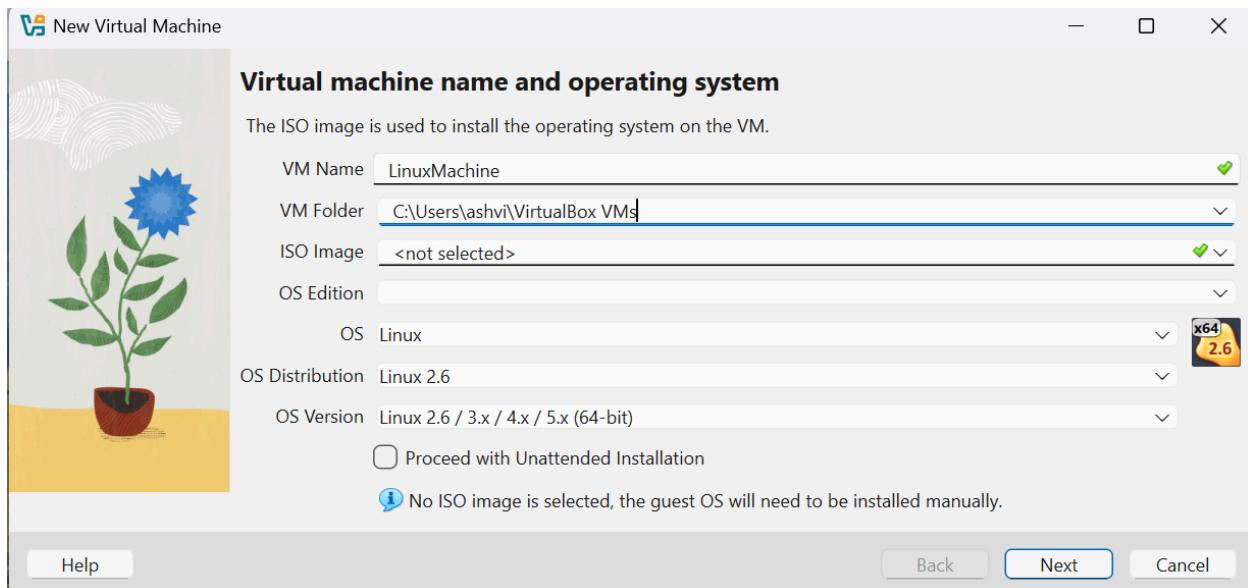
---

### Step 3: Enter Virtual Machine Details

Fill in the following information:

- **VM Name:** `LinuxMachine`
- **VM Folder:** Leave unchanged (default)
- **ISO Image:** Leave unchanged
- **Type (OS):** `Linux`
- **Version (OS Distribution):**  
`Linux 2.6 / 3.x / 4.x / 5.x (64-bit)`

Click **Next**.

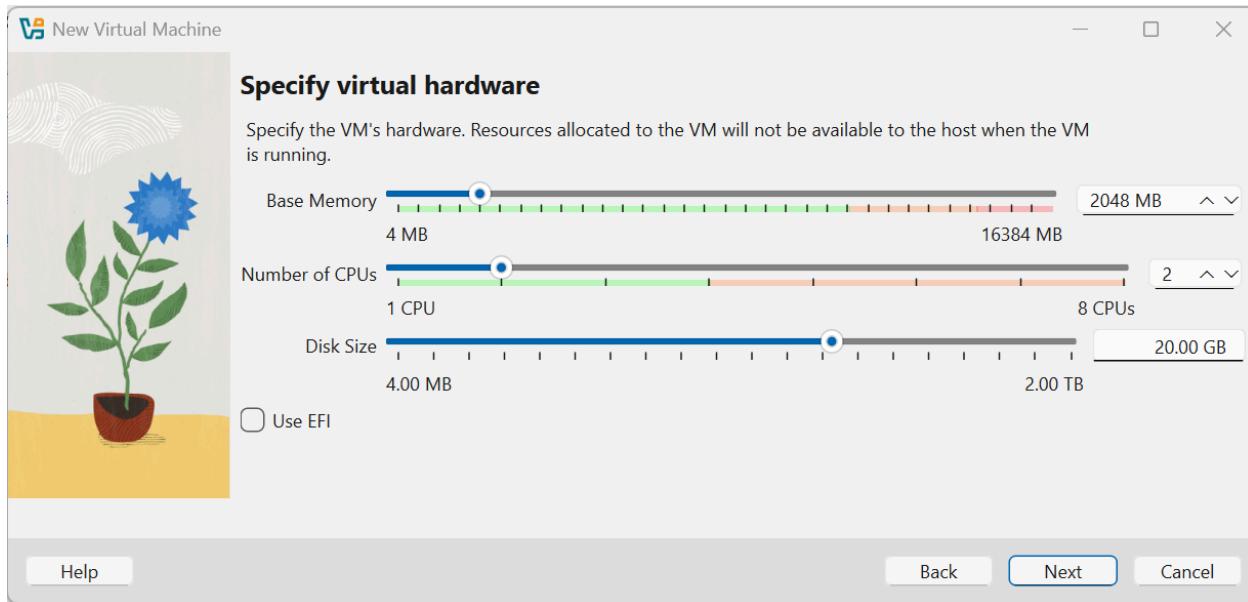


## Step 4: Configure Virtual Hardware

Set the hardware resources as follows:

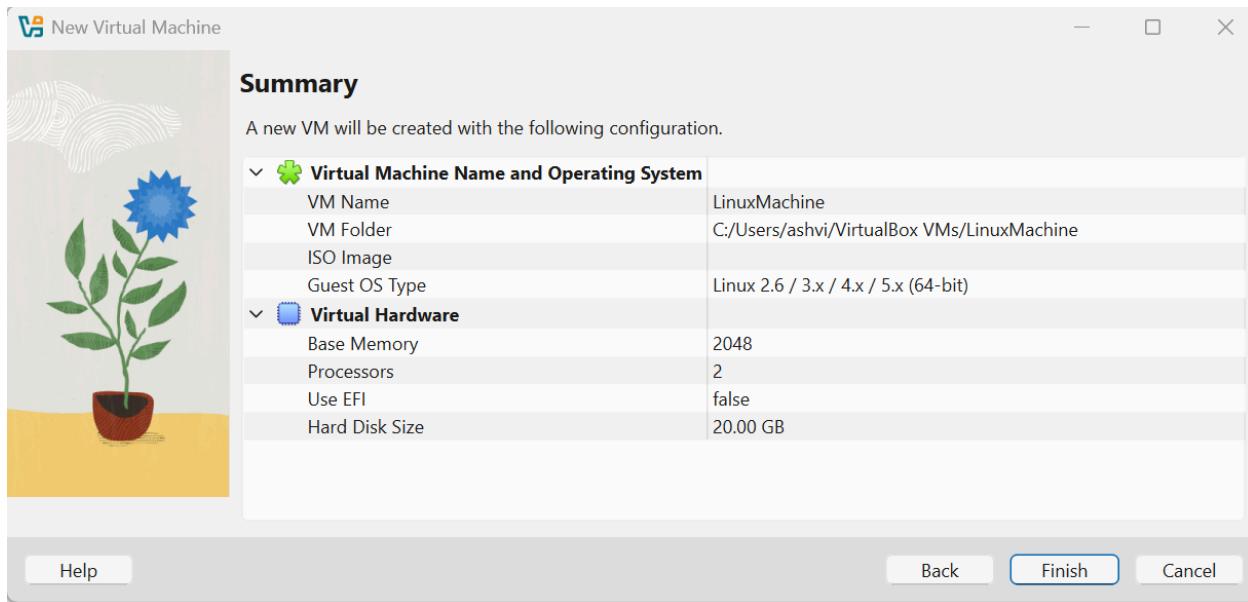
- **Base Memory (RAM):** 2048 MB
- **Processors (CPU):** 2
- **Disk Size:** 20 GB

Click **Next**.



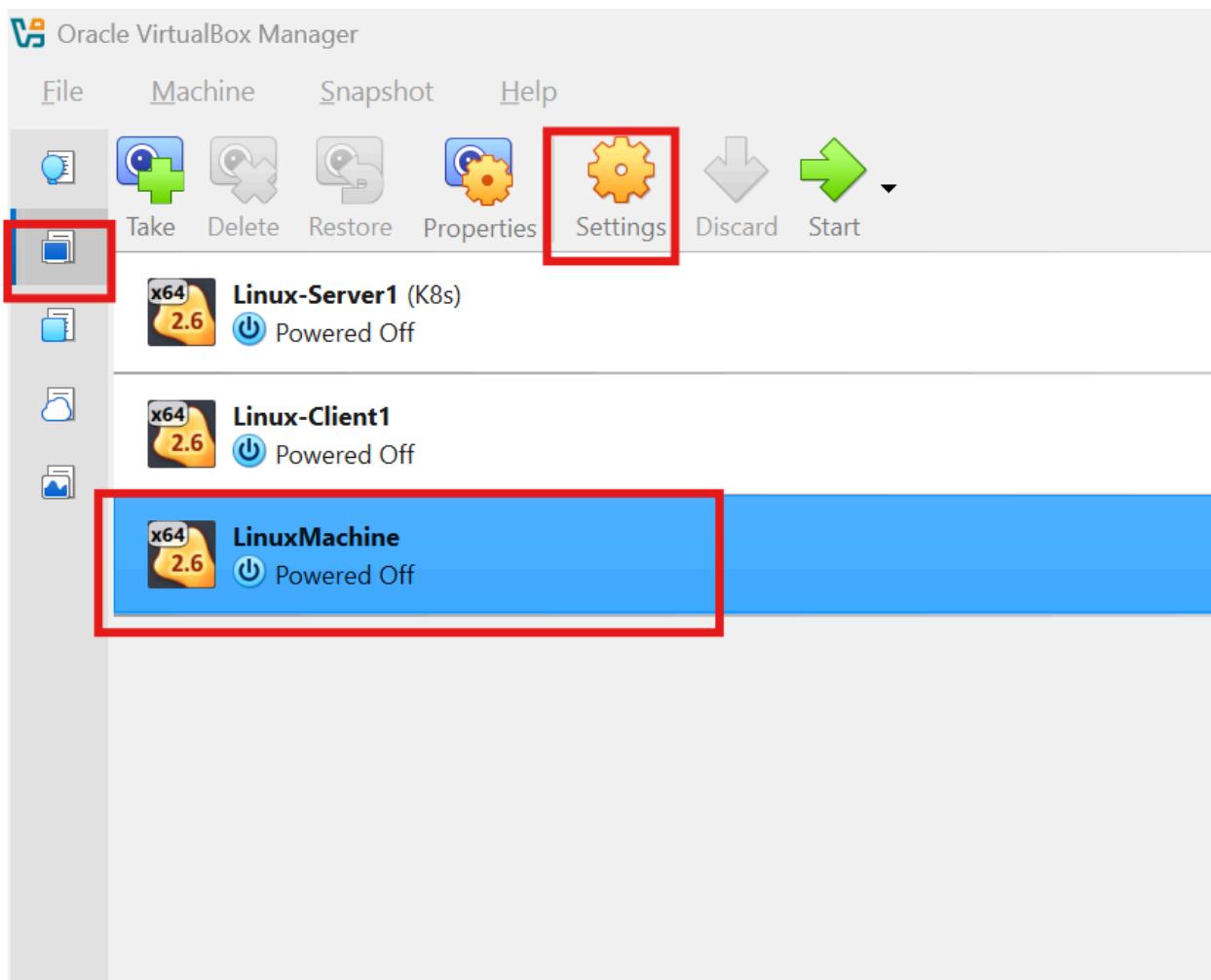
## Step 5: Review and Create

- Verify all the settings.
- Click **Finish** to create the virtual machine.

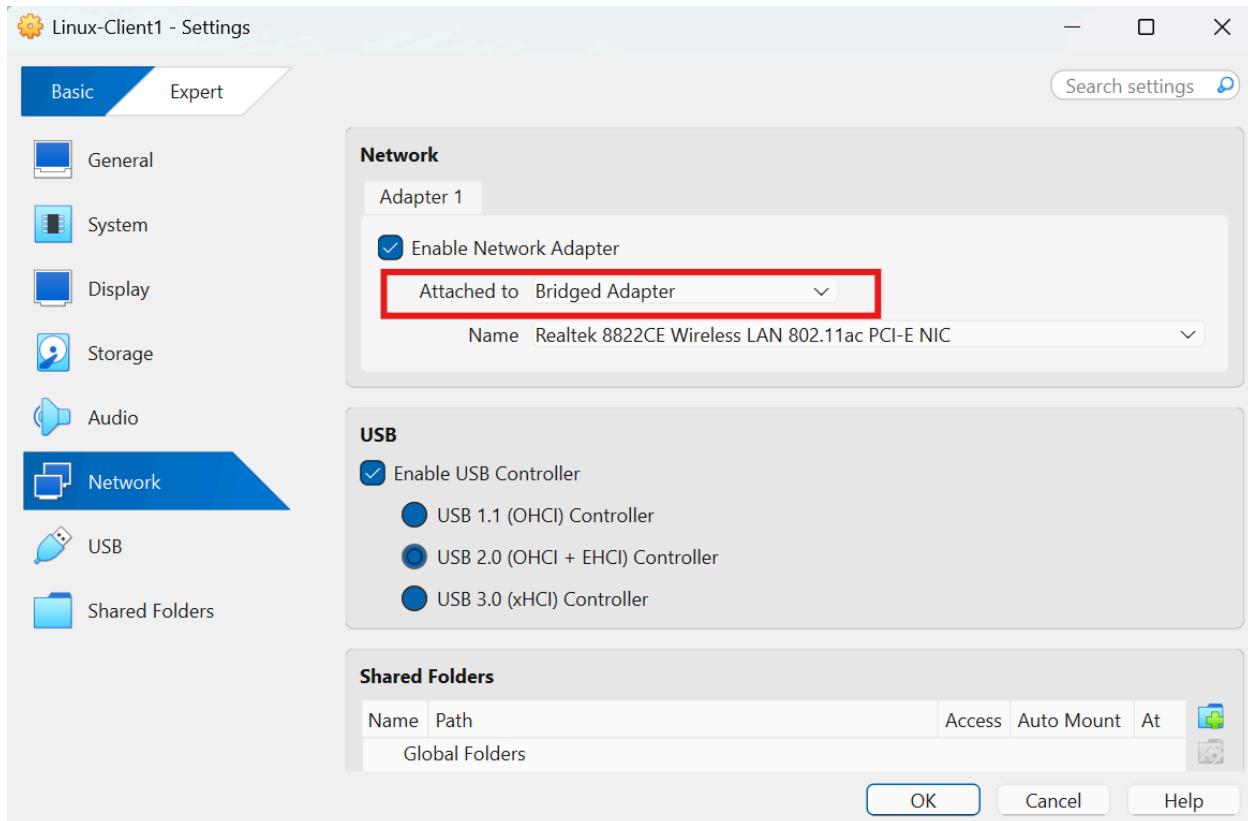


## Step 6: Configure Network Settings

1. In the **Machines** list, select the newly created VM.
2. Click **Settings** → **Network**.

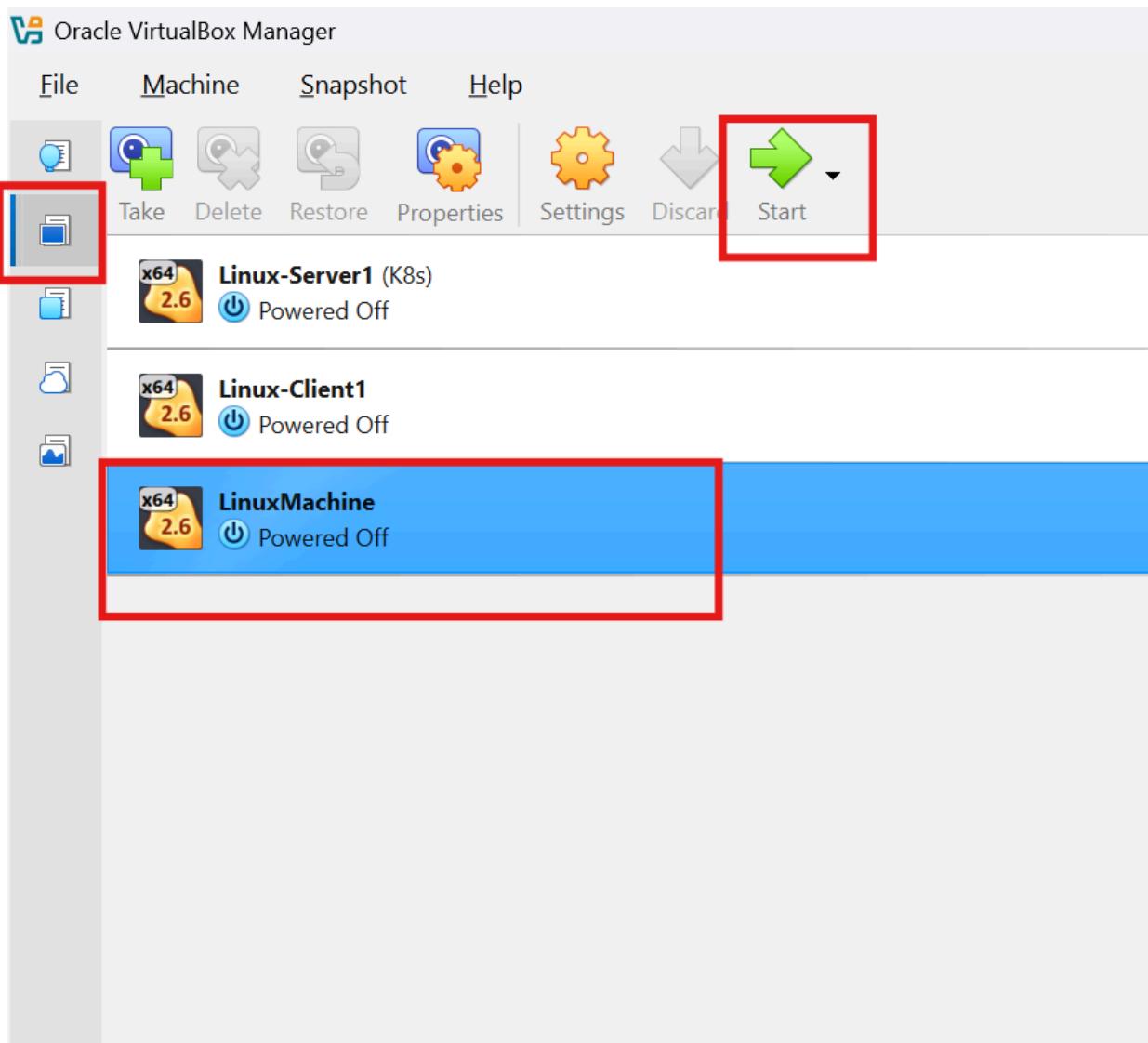


3. Under **Adapter 1**:
  - Change **Attached to** from **NAT** to **Bridge Adapter**.
4. Click **OK**.

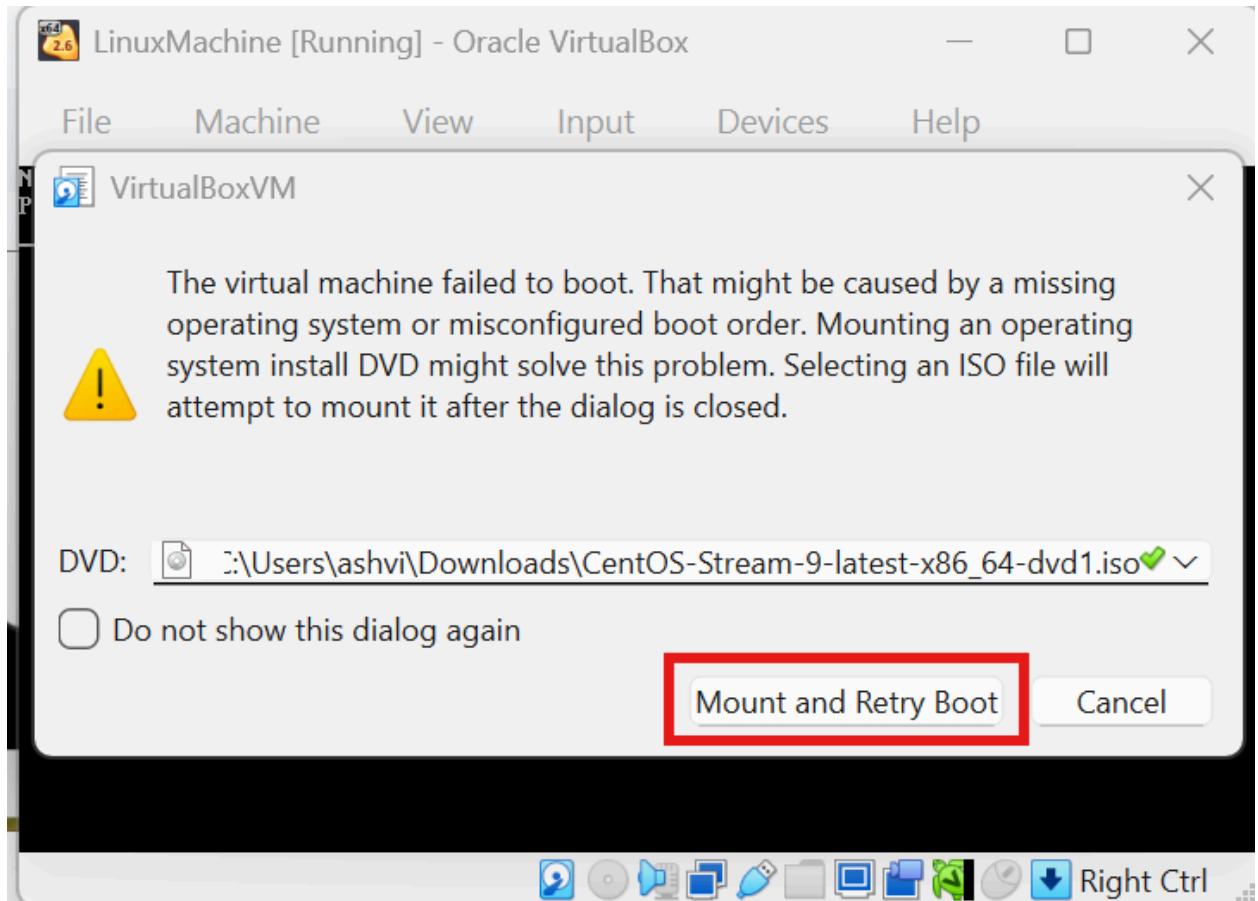


## Step 7: Attach CentOS Stream 9 ISO

1. Select the VM and click **Start**.



2. A new window will appear asking to select a startup disk.
3. Click the **dropdown arrow**.
4. Browse to the location of your downloaded **CentOS Stream 9 ISO** file.
5. Select the ISO file.
6. Click **Mount and Retry Boot**.



## Step 8: Start CentOS Installation

1. The CentOS installation menu appears.
2. Click **Capture** when prompted.



VirtualBox - Information

X



You have **clicked the mouse** inside the Virtual Machine display or pressed the **host key combo**. This will cause the Virtual Machine to **capture** the host mouse pointer (only if the mouse pointer integration is not currently supported by the guest OS) and the keyboard, which will make them unavailable to other applications running on your host machine.

You can press the **host key combo** at any time to **uncapture** the keyboard and mouse (if it is captured) and return them to normal operation. The currently assigned host key combo is shown on the status bar at the bottom of the Virtual Machine window, next to the  icon. This icon, together with the mouse icon placed nearby, indicate the current keyboard and mouse capture state.

The host key combo is currently defined as **Right Ctrl**.

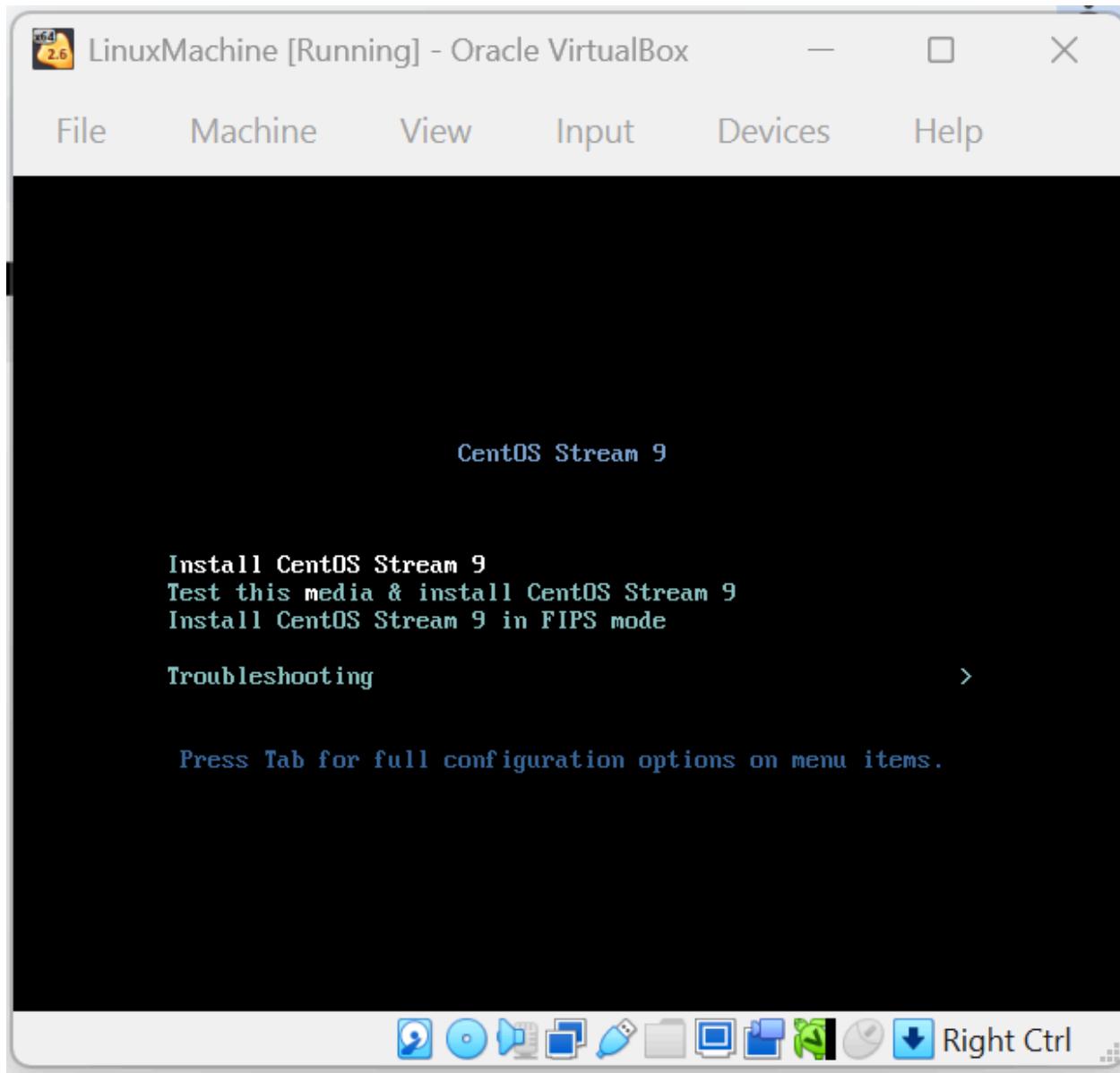
Do not show this message again

Capture

Cancel

3. Using the **keyboard arrow keys**, select:

Install CentOS Stream 9

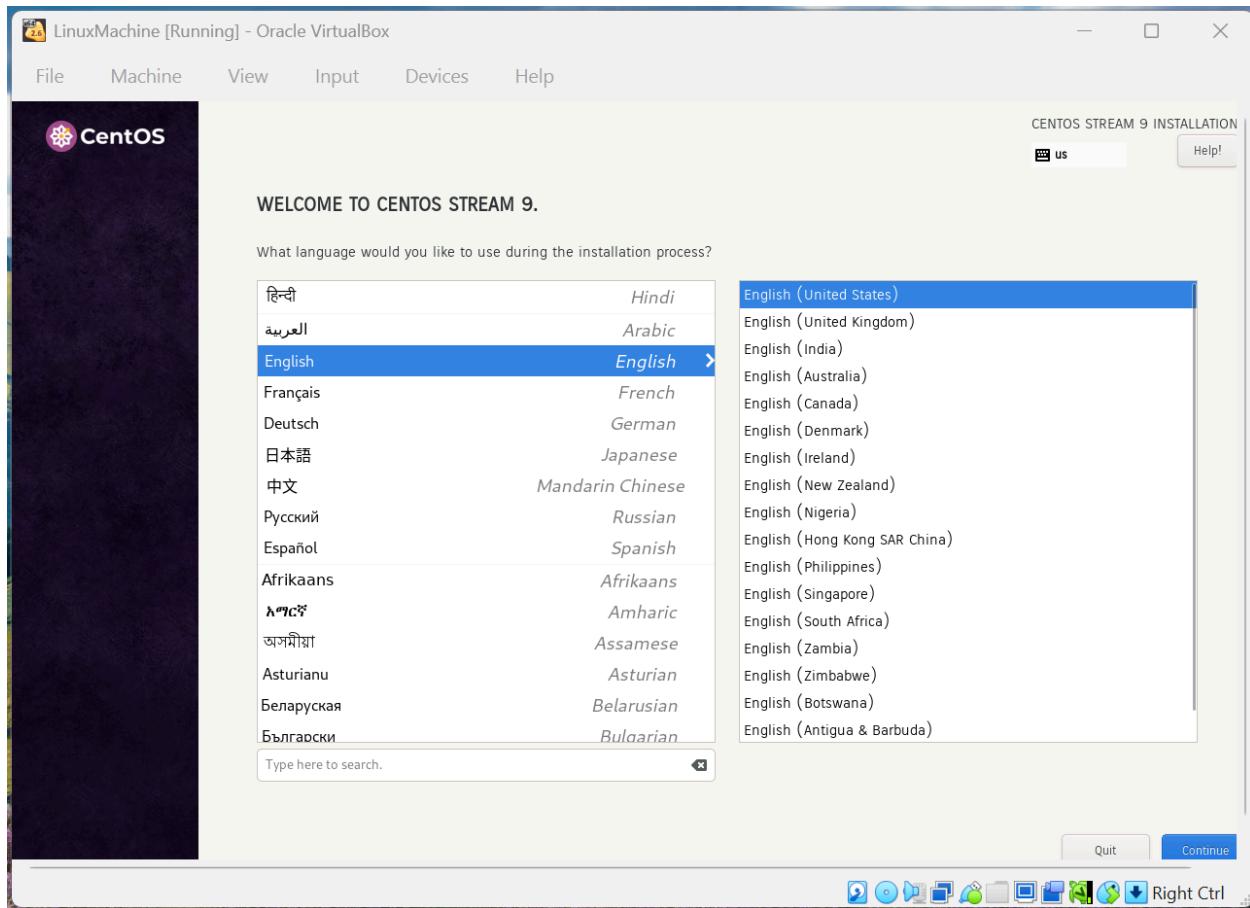


4. Press **Enter**.

---

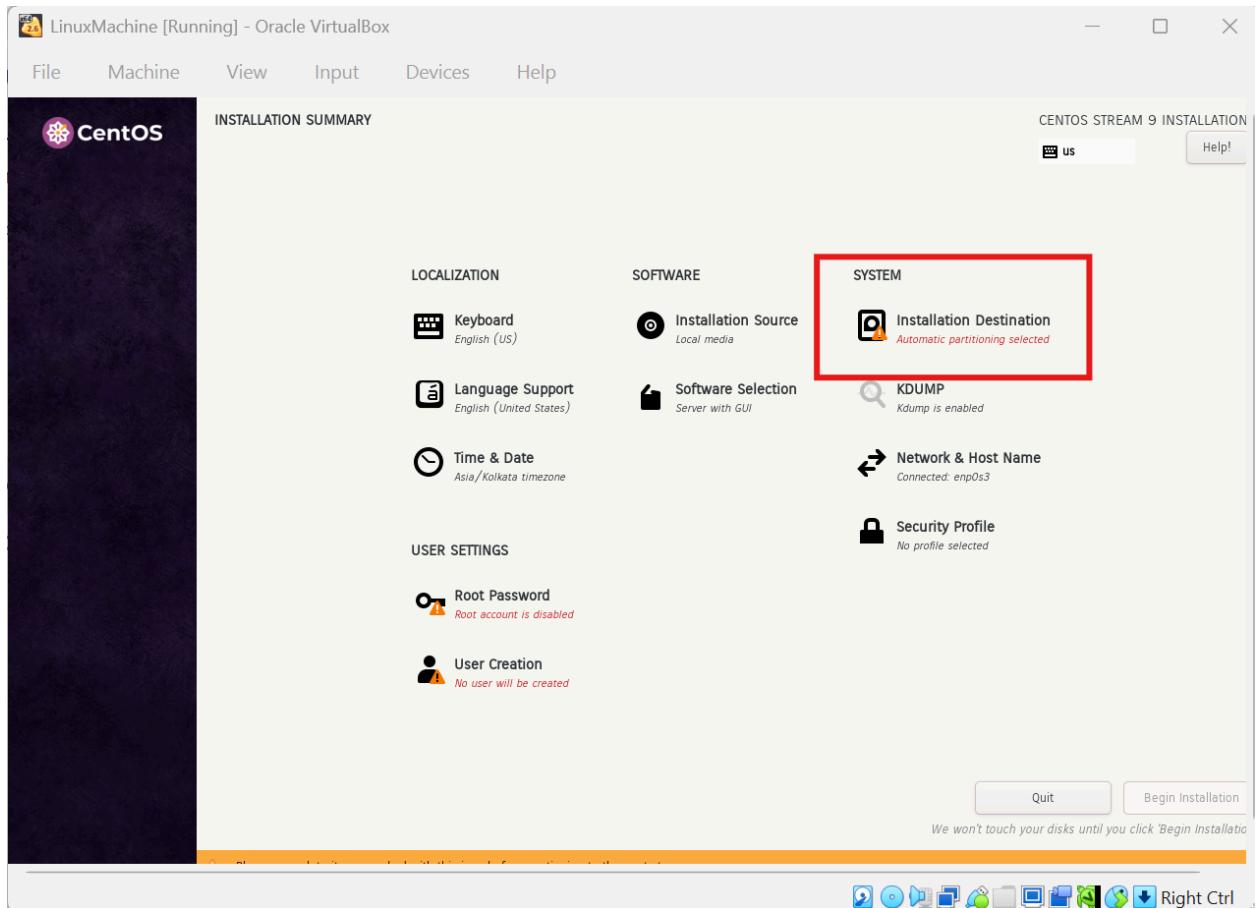
### Step 9: Select Language

- Choose **English**.
- Click **Continue**.

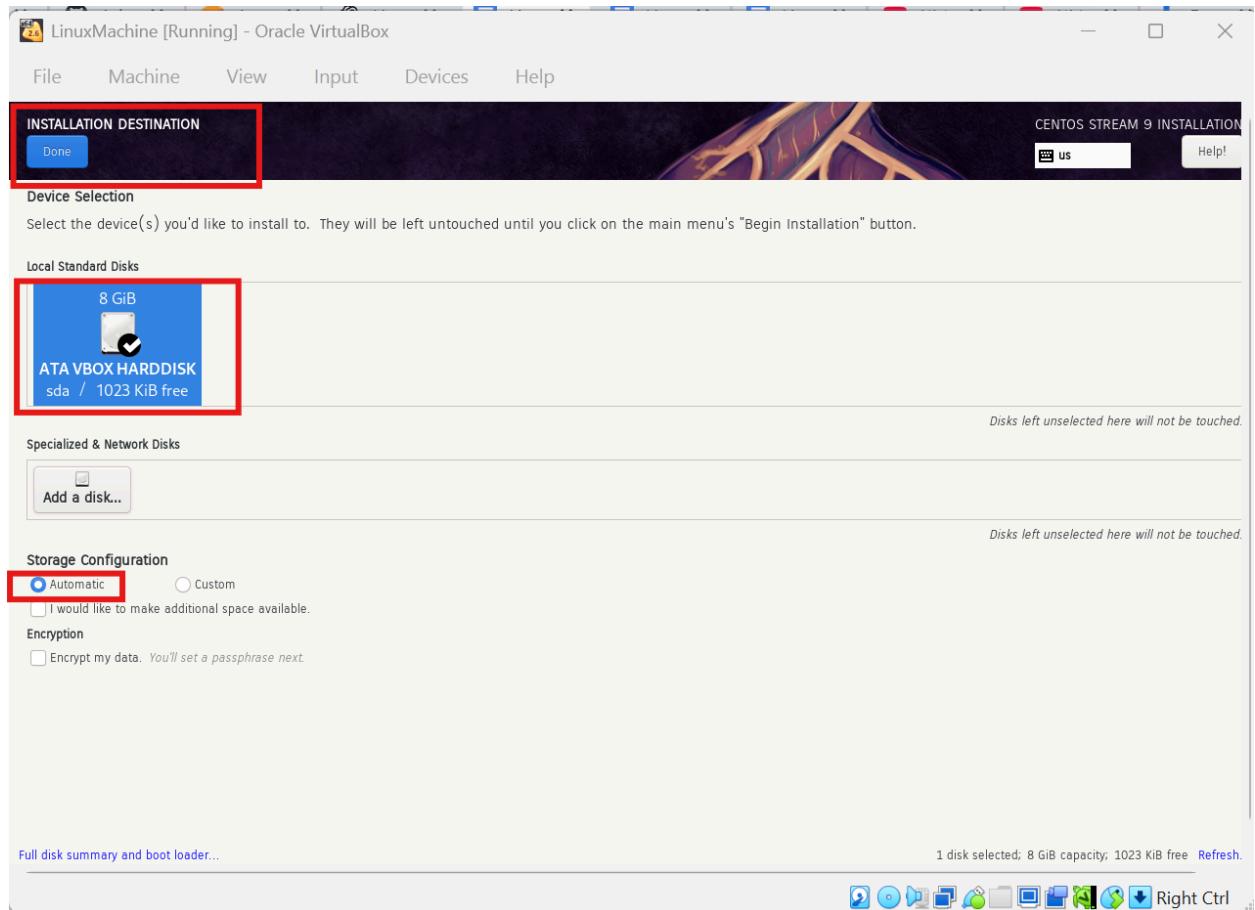


## Step 10: Configure Installation Destination

1. On the **Installation Summary** screen, click **Installation Destination**.



2. Select the available storage device.



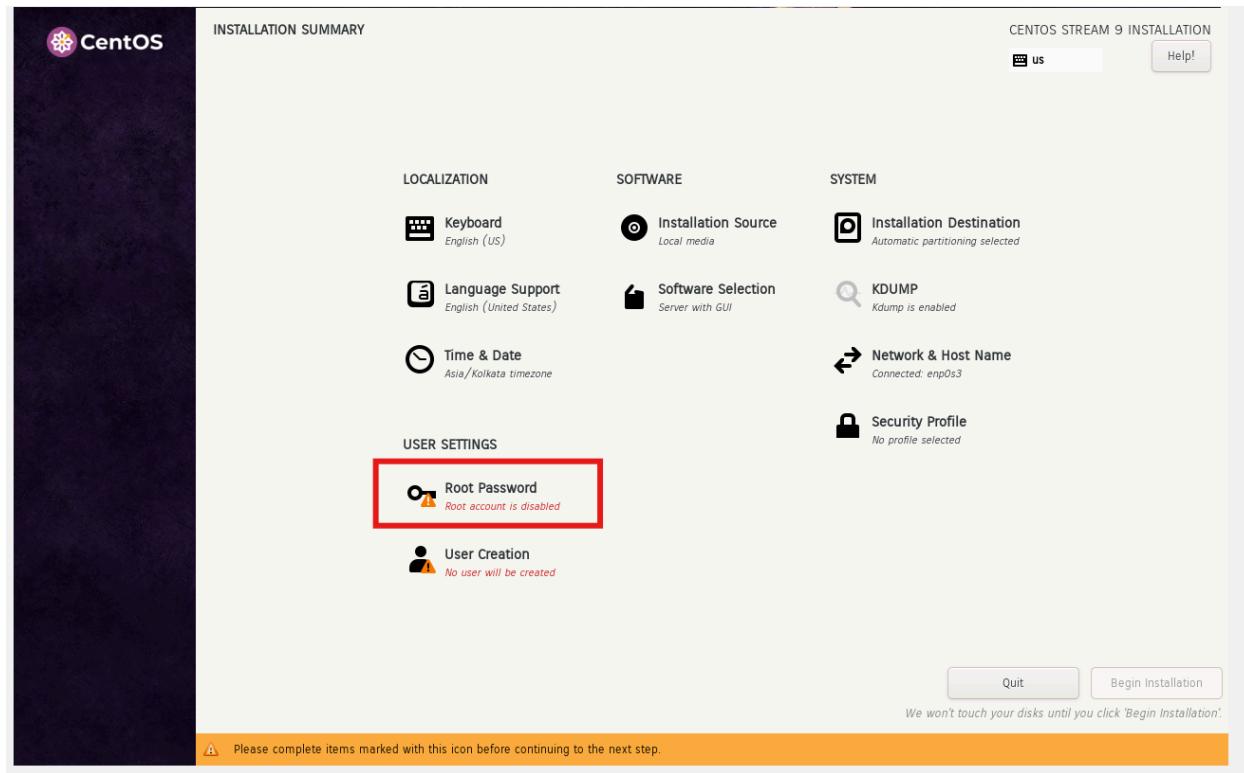
3. Keep **Storage Configuration** as **Automatic** (default).

4. Click **Done**.

---

## Step 11: Set Root Password

1. Click **Root Password**.

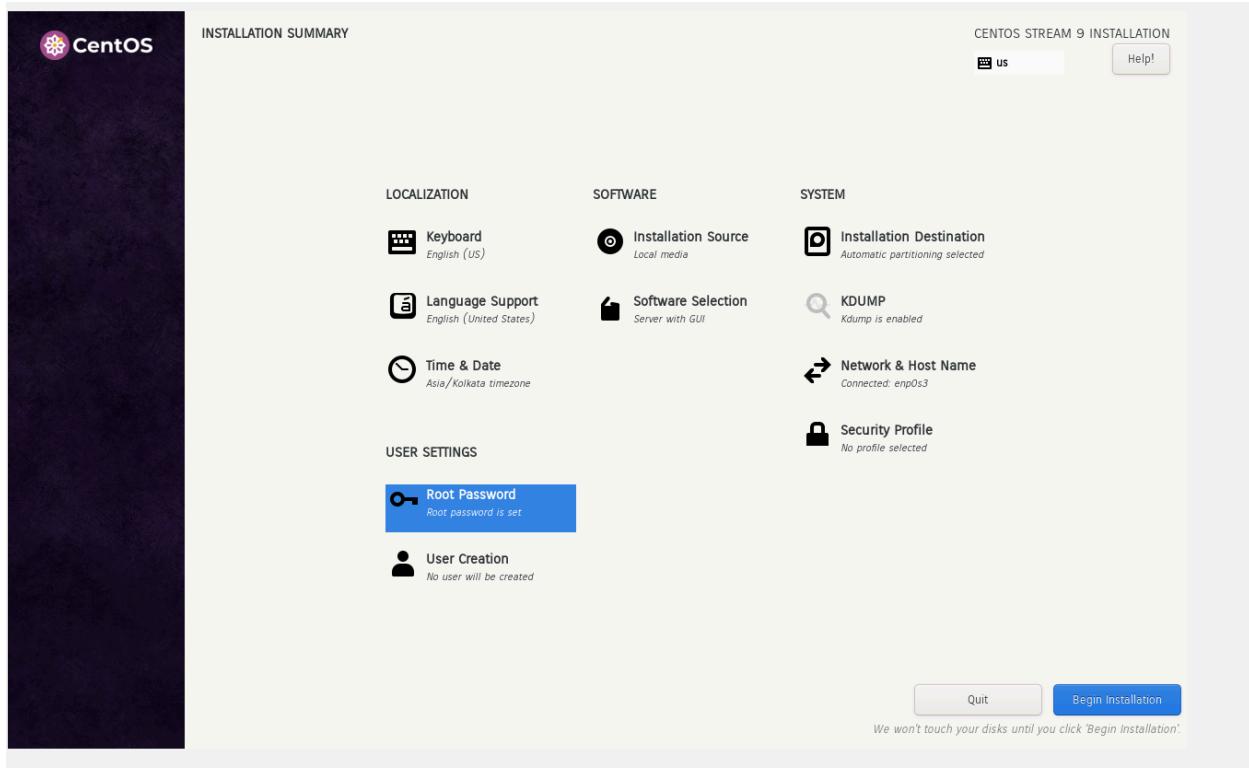


2. Set a strong password of your choice.
3. **Note the password safely.**
4. Click **Done**.

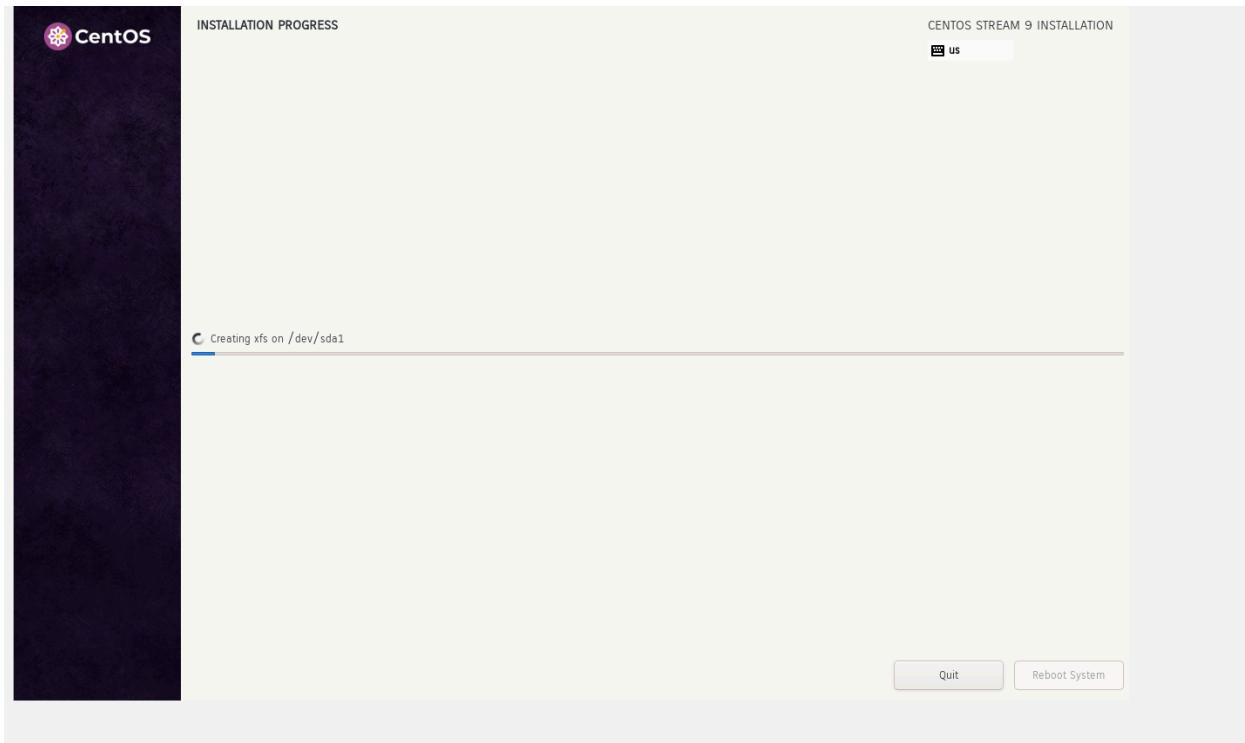
---

## Step 12: Begin Installation

- Once all required sections are completed, the **Begin Installation** button becomes active.



- Click **Begin Installation**.
- Installation will start (this may take **30–45 minutes**).



---

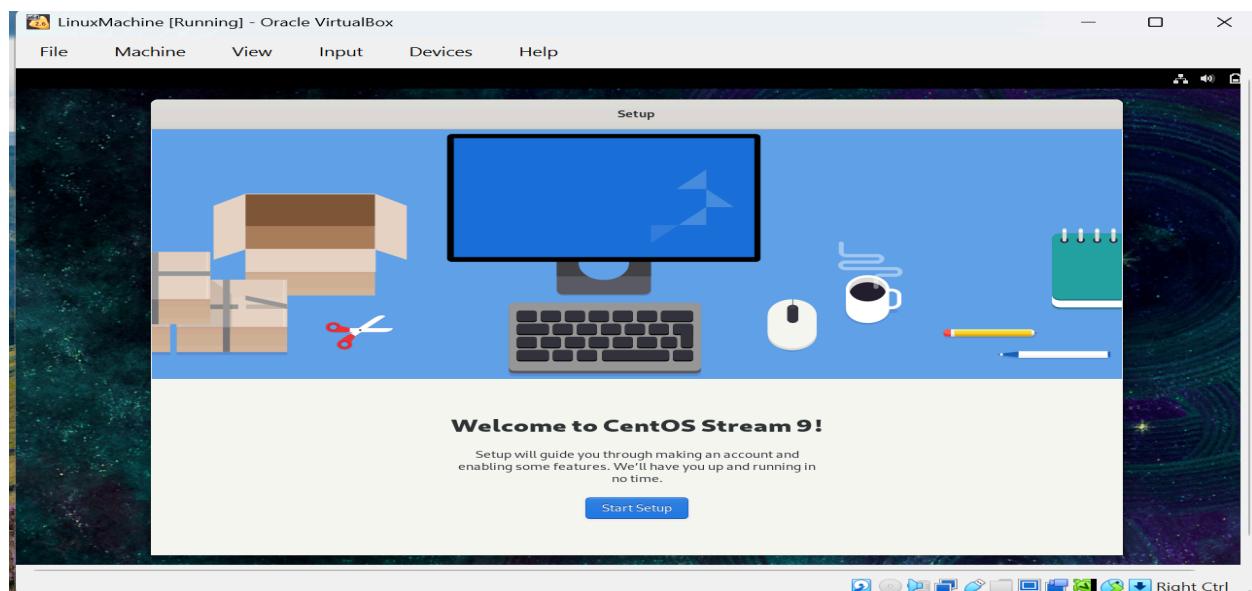
## Step 13: Reboot the System

- After installation completes, click **Reboot System**.

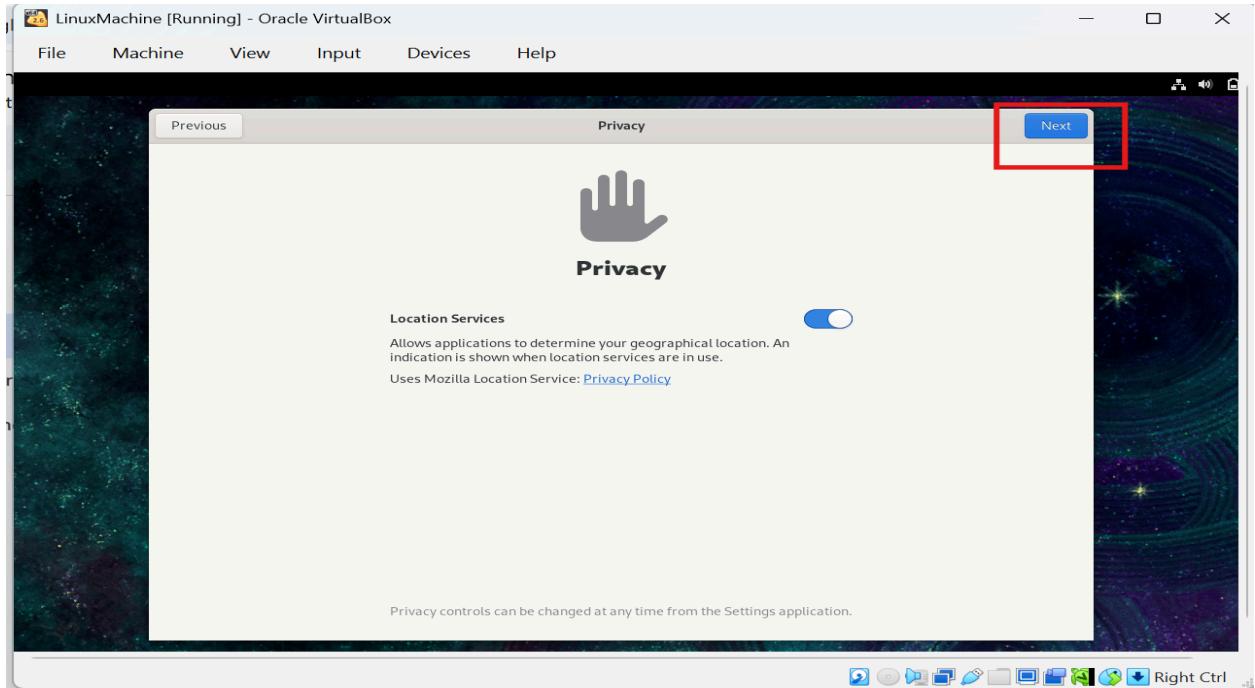


## Step 14: Complete Initial Setup After Reboot

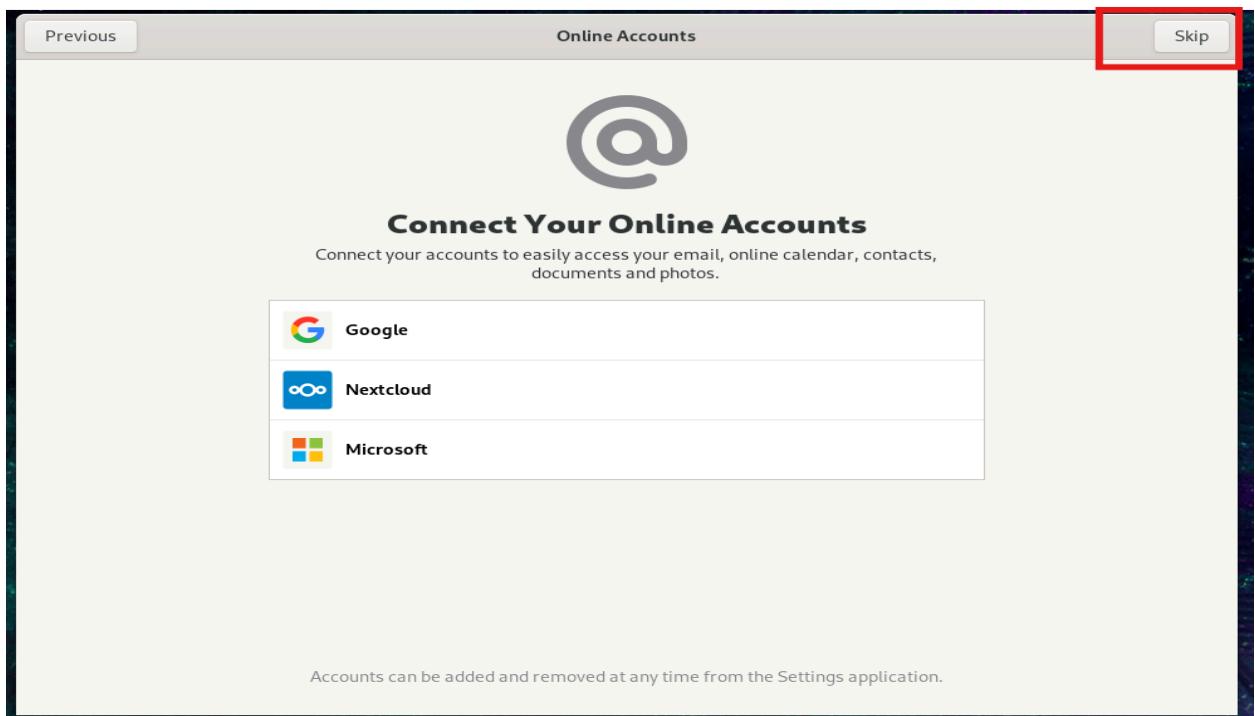
- Once the system reboots, the **Welcome Screen** will appear.



2. Click **Start Setup**.
3. On the **Privacy** screen, click **Next**.

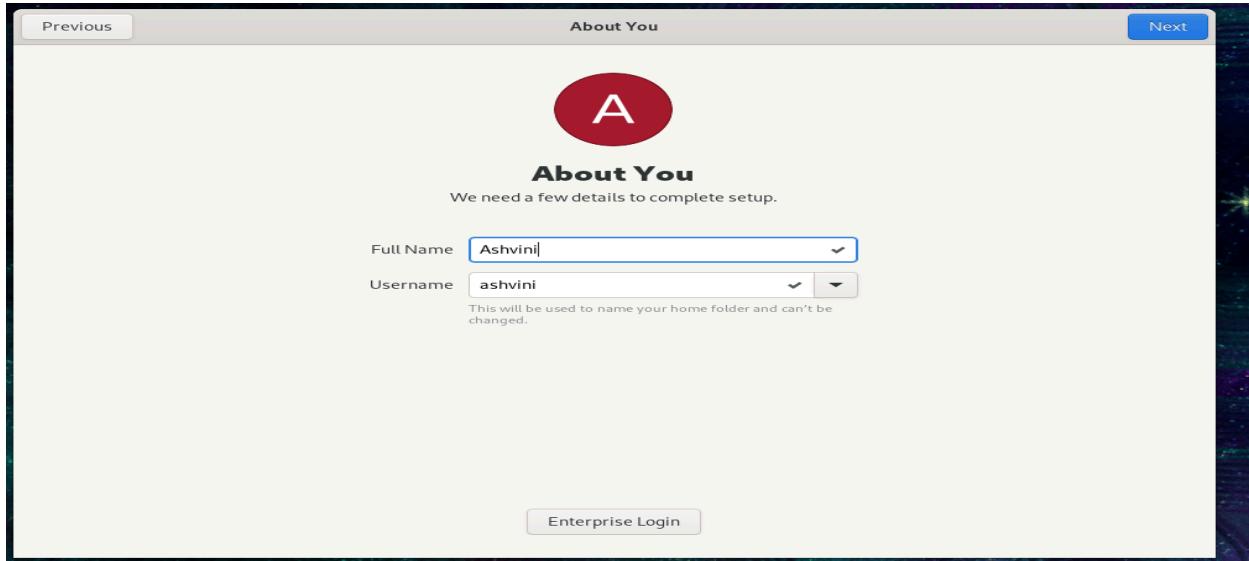


4. On the **Online Accounts** screen, click **Skip**.



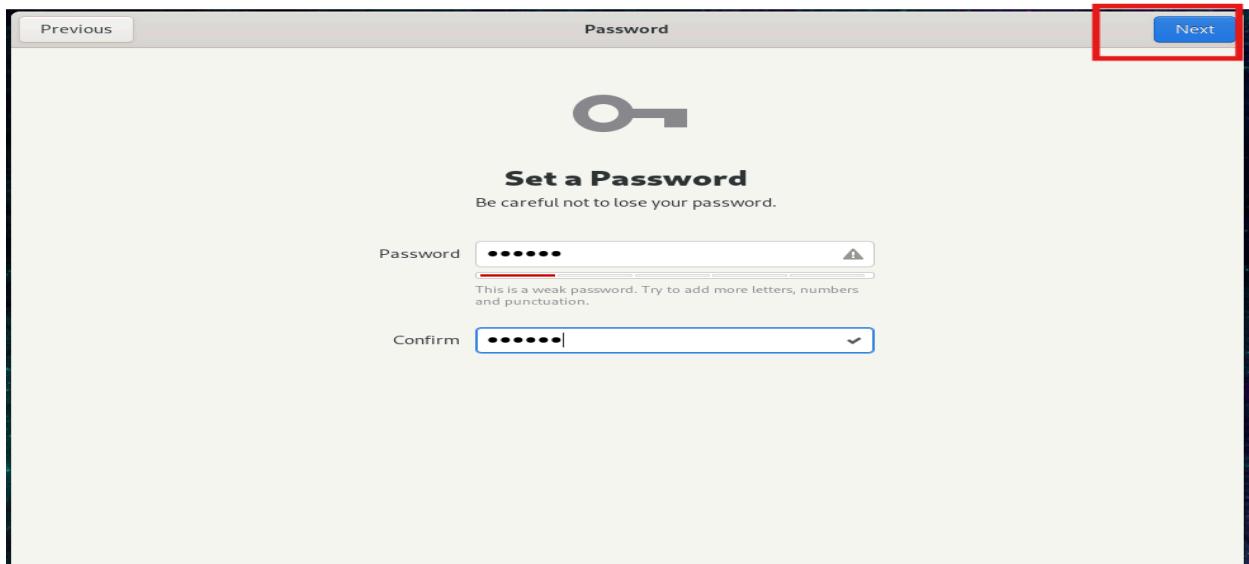
5. In the **About You** section:

- Enter your **full name** and **username**.
- Click **Next**.

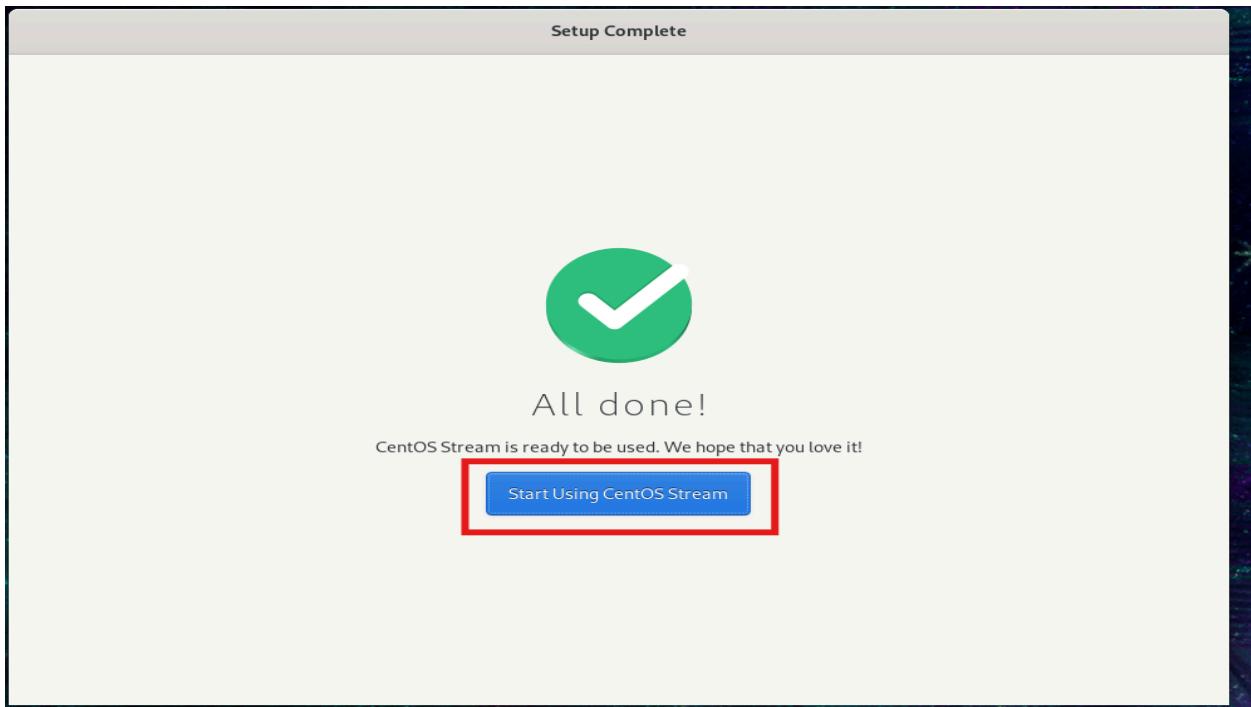


6. Set a **user password**:

- Enter and confirm the password.
- Click **Next**.



7. Click **Start Using CentOS Stream**.



8. You may choose to **Take the Tour** or click **Skip** to go directly to the desktop.

**Your Linux virtual machine is now ready to use** 🎉