

Building a virtual home lab

Allow me to show you how to set up a Free Virtual IT Lab at Home.

In the early 2000s, setting up an IT lab required significant hardware investment. Technicians often relied on physical servers or refurbished desktops to simulate enterprise environments. While effective for hands-on learning, it was time-consuming and costly to maintain.

Today, virtualization and open-source technologies have simplified the process. With sufficient computing resources, you can create a fully functional IT lab directly on your personal computer—no dedicated hardware required.

For this project, I built a complete virtual IT lab from the ground up using only open-source software. The lab is cost-free, efficient, and scalable, allowing me to replicate real-world IT and cybersecurity scenarios in a controlled environment.

The following sections outline each step of the process, including system requirements, software setup, network configuration, and testing procedures. By following these instructions, you'll be able to deploy your own virtual IT lab for training, research, or certification practice.

Index

- Virtualization Overview
- Free Virtualization Software
- Downloading and Installing VirtualBox
- Creating a Virtual Network
- Creating a Virtual Machine
- Downloading Your operating systems ISO(s)
- Installing an OS on Your Lab VMs
- Conclusion

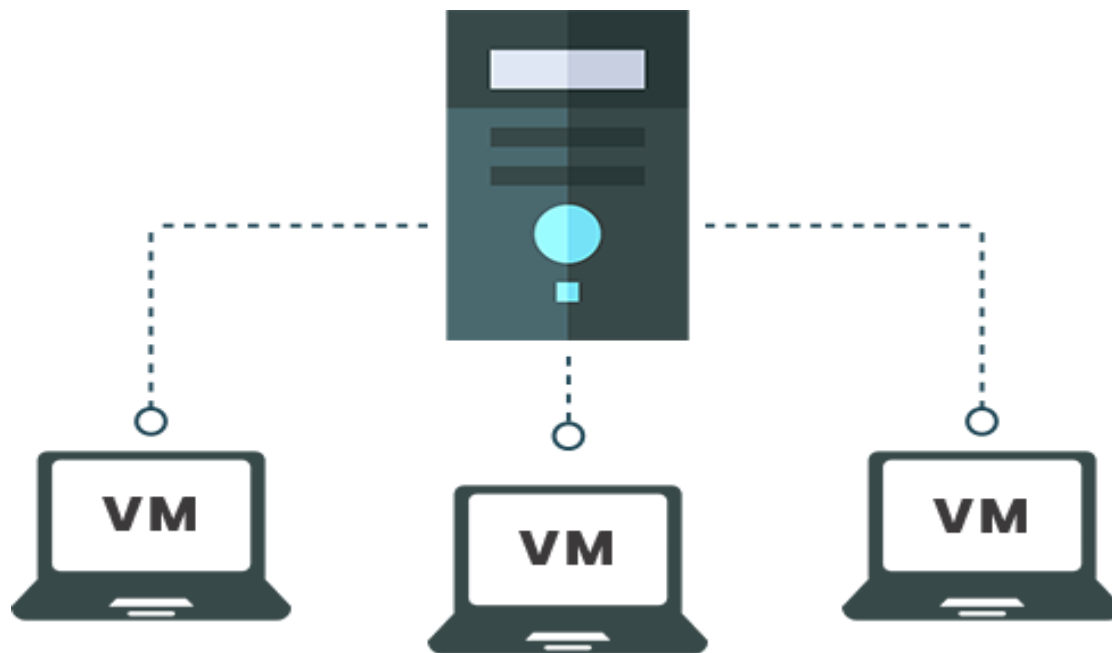
Virtualization Overview

What Is Virtualization?

Virtualization is the technology that allows you to simulate multiple computer systems on a single physical machine. Instead of relying on separate hardware for each system, you can create **virtual machines (VMs)**—software-based environments that behave like independent computers.

In this setup, your main computer acts as the **host system**, providing the physical resources such as CPU, memory, and storage. Each virtual machine, known as a **guest**, runs its own operating system and operates independently from the host and other guests.

Essentially, virtualization enables you to run several isolated computers within one physical device, making it an efficient and flexible way to test, learn, or deploy IT solutions without additional hardware.



Virtual Machines — Computers Within a Computer

A **Virtual Machine (VM)** is a software-based simulation of a physical computer. Each VM operates as an independent system, allowing you to perform the same actions you would on dedicated hardware—such as powering on or off, restarting, installing operating systems, and running applications.

This flexibility makes virtual machines an essential component of any virtual IT lab. They provide an isolated and controlled environment where you can safely test configurations, experiment with software, and develop hands-on IT or cybersecurity skills without the risk of damaging your main system or requiring additional hardware.

Free Virtualization Software

There are many virtualization tools available, and choosing the “best” one can feel overwhelming.

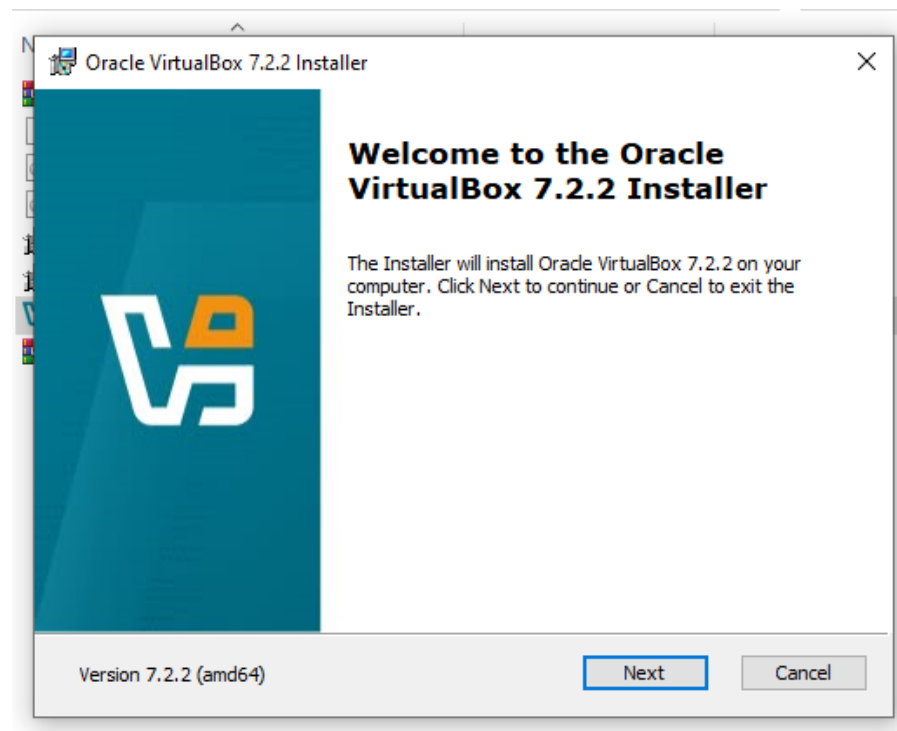
The reality is, there’s no one-size-fits-all solution—it all comes down to your operating system and specific requirements.

For this project, I’m using Oracle VM VirtualBox because it runs smoothly on both Windows and Linux.

That said, you’re free to use any virtualization software you prefer—most of the setup steps will be quite similar across different platforms.

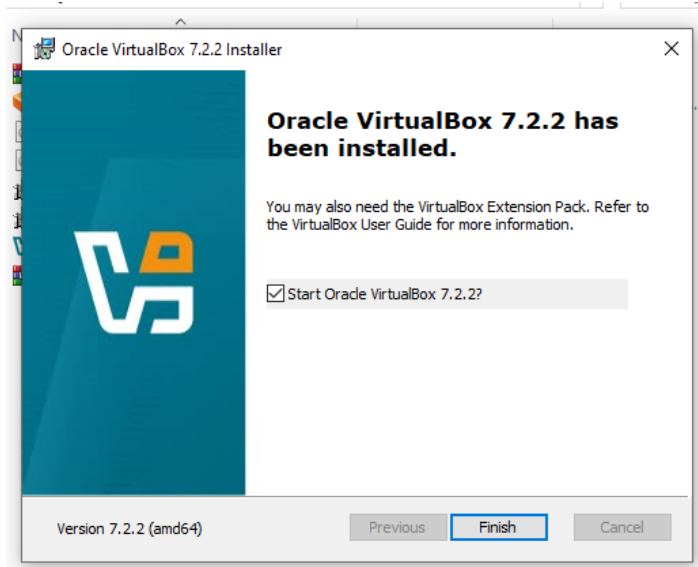
Downloading and Installing VirtualBox

Download Oracle VM VirtualBox by [clicking here](#) or from a quick google search. Once the download is completed, I launched the installation

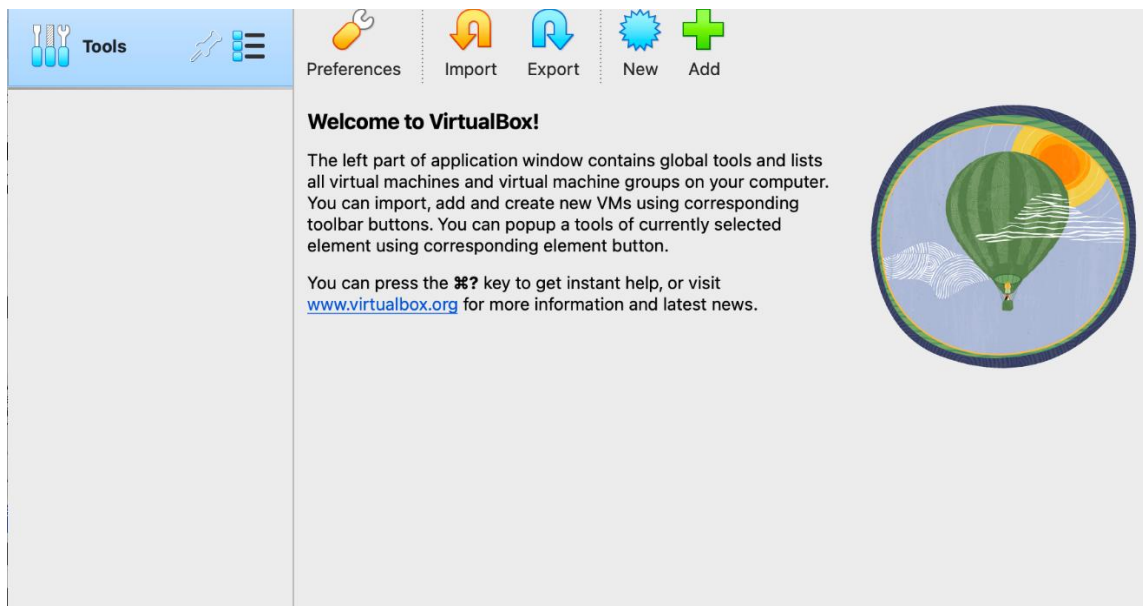


I went through the installation using **all the default options**.

Whenever you're prompted with a **yes/no question**, simply choose **"Yes"** to continue.

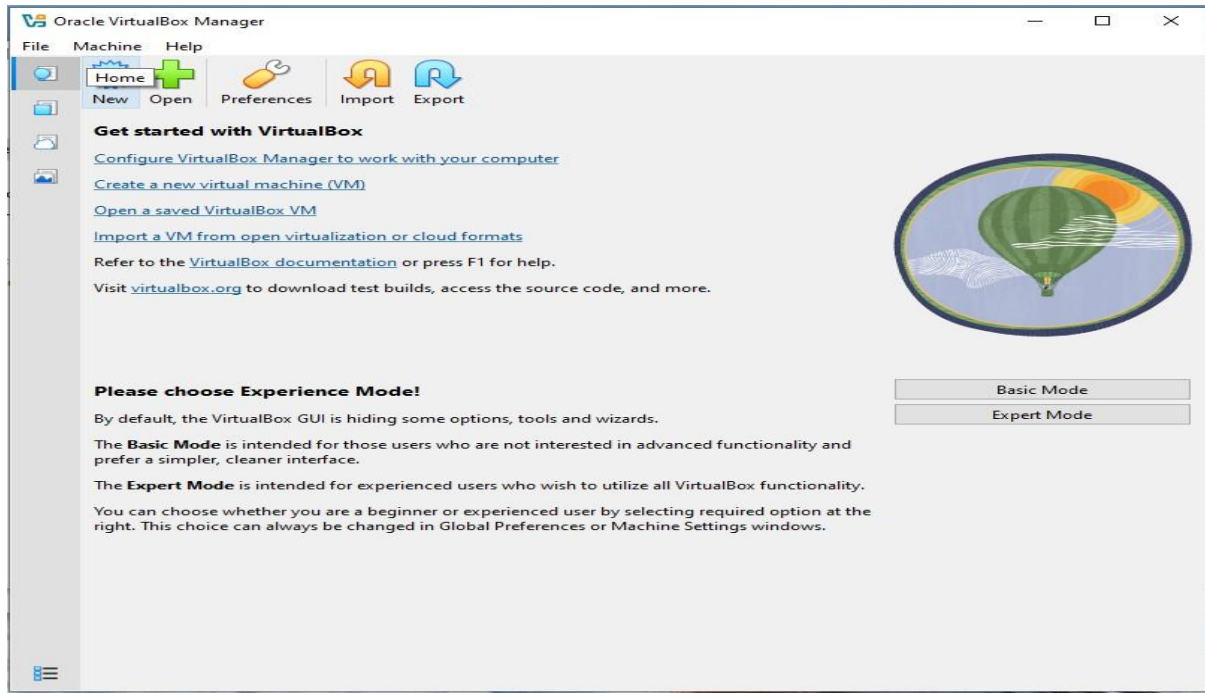


Click Finish and launch VirtualBox.



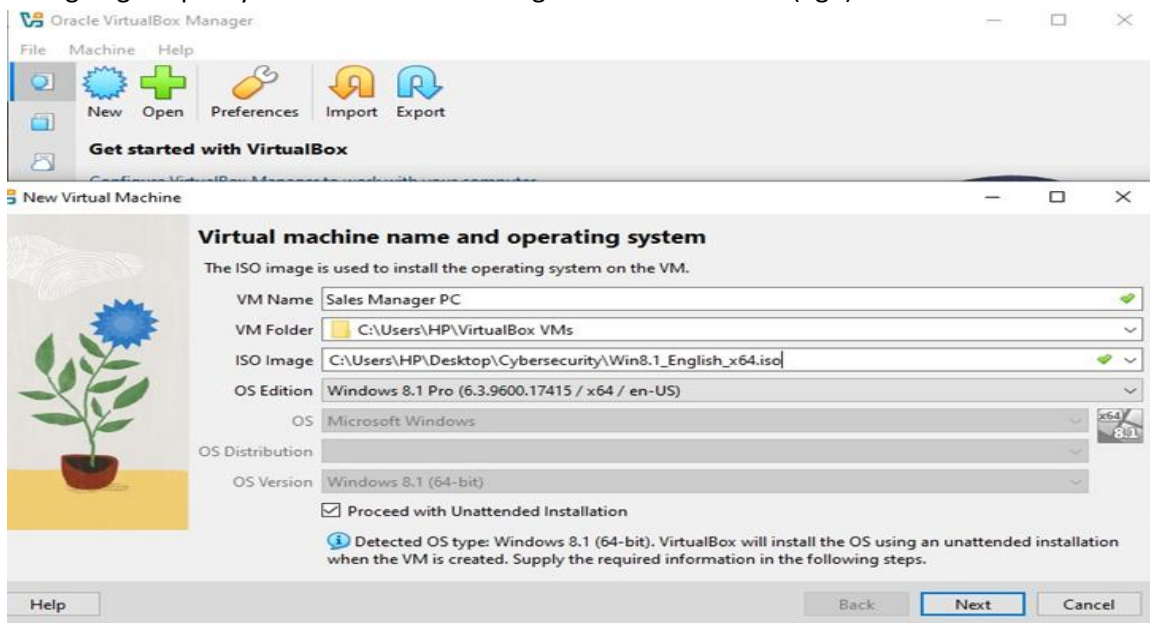
Creating a Virtual Machine

To create a Virtual Machine (VM). You can do this by clicking the **"New"** button in the VirtualBox Manager.



This will launch the new VM window. Click the *Expert Mode* button to proceed. This streamlines the setup by reducing the number of steps.

I am going to specify a name of "Sales Manager PC" and 2048 MB (2gb) of RAM then I will click Finish.



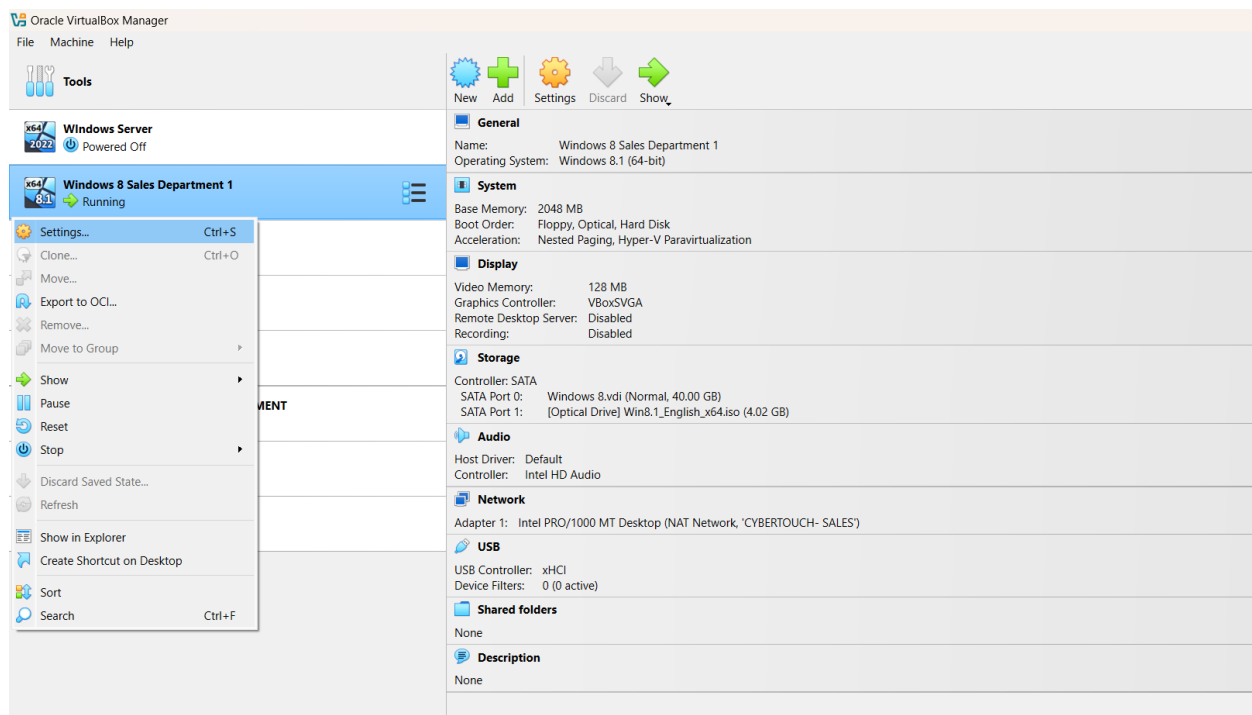
This will open another window for creating the Virtual Hard Disk. The default options here are all fine except for the File Size. I am going to set my new HDD to 40GB and then choose Finish.

Since this hard disk drive (HDD) is set to be *dynamically allocated* (see the option on the middle-right of the screenshot below), the HDD file will only take up as much space as the data stored on it—up to a maximum of **40GB**.

NOTE: After installation, the name given to the VM can be changed. For example you'll see Windows 8 Sales Department 1 listed on the VirtualBox dashboard and not Sales manager PC anymore.

Before starting the virtual machine (VM), you can modify its settings if needed. Typically, you might want to:

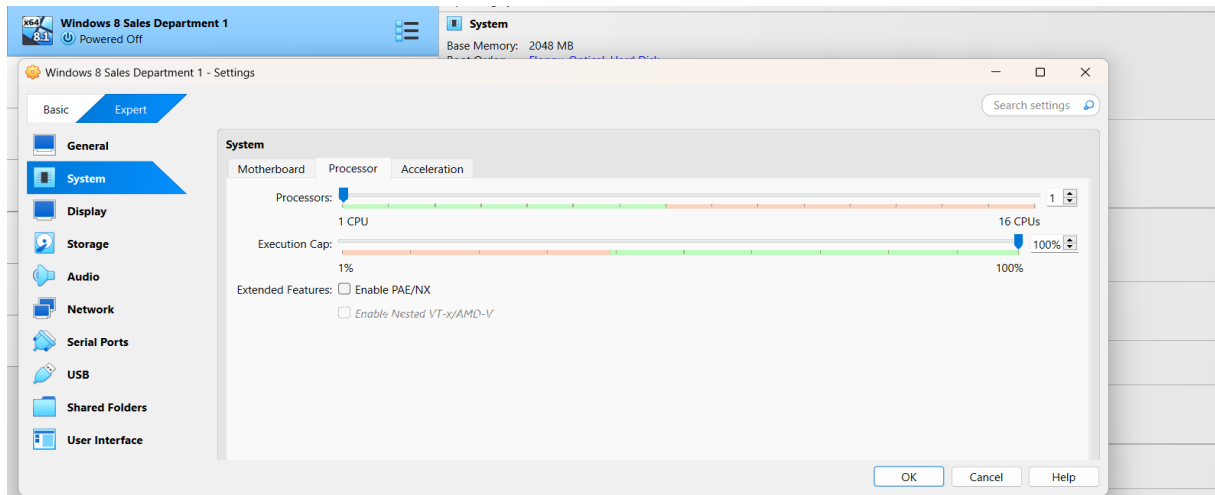
- Assign more processors
- Change the networking adapter
- Mount an ISO image



Increasing the number of CPU cores can significantly enhance your virtual machine's (VM) performance. If possible, set the number of CPUs to **2**.

Pro tip: Storing the VM's virtual hard disk on a **solid-state drive (SSD)** can further improve performance, especially when running disk-intensive tasks.

Base memory (RAM) should not be large enough for the VM to run. If the VM doesn't boot, try increasing the RAM

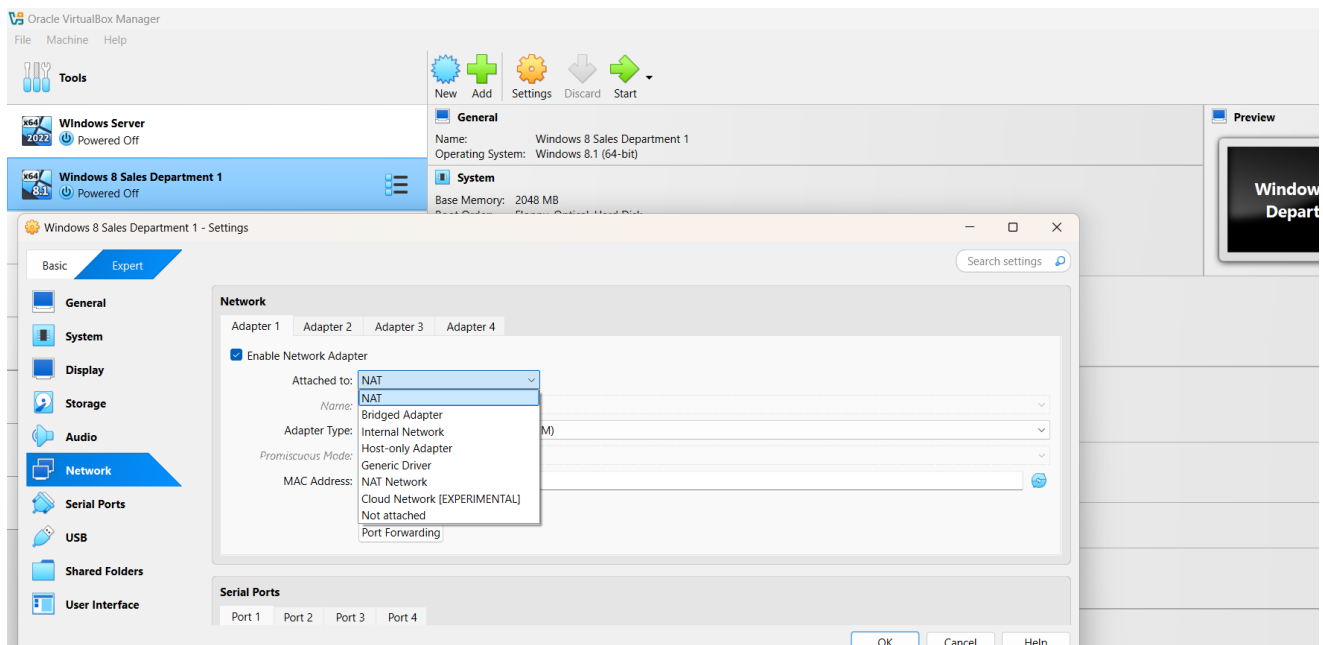


And that's all we need to do to create the virtual machine (VM). We need to download and attach an operating system ISO. This will be covered that in the next section.

Creating a Virtual Network with VirtualBox

Virtual networks allow your VMs to connect to each other, your host machine, or the internet depending on how you configure them. VirtualBox supports several types of virtual network configurations, and the one you choose will depend on your specific use case.

Note: Some network types can only be configured *after* a VM has been created, as they're managed within the VM's settings.



Network Types in VirtualBox

1. NAT (Network Address Translation)

This network type allows your VM to access the internet using the host computer's connection. However, it does not allow communication between the VM and the host or between multiple VMs.

Use this if: Your lab setup only includes one VM and it just needs internet access.

To enable NAT:

- Right-click your VM
- Select Settings
- Go to the Network tab
- Choose Attached to: NAT (see image above)

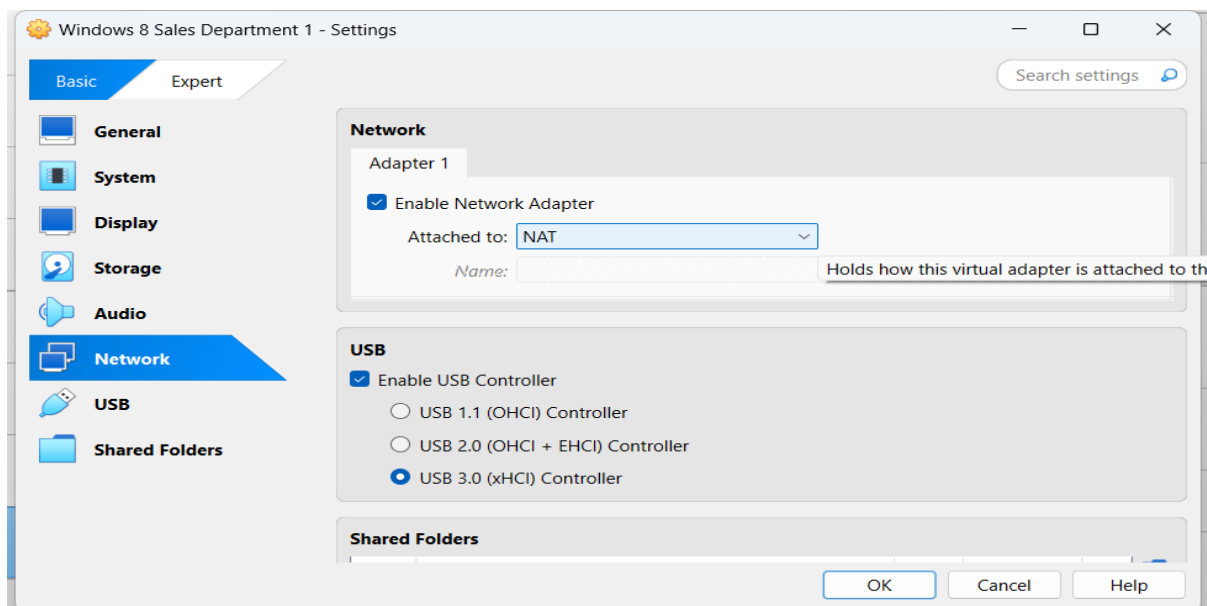
2. NAT Network

This option is similar to NAT but also allows VMs connected to the same NAT Network to communicate with each other, while still maintaining internet access.

Use this if: Your lab setup includes multiple VMs that need to talk to each other and access the internet.

To create a NAT Network:

- Go to File > Preferences
- Select the Network tab
- Click the plus (+) button to create a new NAT Network
- Then, in your VM's settings, attach it to the NAT Network you just created



3. Bridged Adapter

This network type makes your VM appear as a separate physical device on your local network. Your router will assign it an IP address, just like it would with any other device.

Use this if: You need your VM to be accessible from **other devices** on your local network (e.g., your host, other VMs, or even other physical computers).

To enable a Bridged Adapter:

- Right-click your VM
- Select **Settings**
- Go to the **Network** tab
- Choose **Attached to: Bridged Adapter**

4. Internal Network

This creates a completely **isolated** network shared only between VMs attached to the same internal network. There's **no internet access**, and your host computer cannot communicate with these VMs.

Use this if: You want a fully isolated lab environment for testing, malware analysis, or network simulation.

To use an Internal Network:

- Right-click your VM
- Select **Settings**
- Go to the **Network** tab
- Choose **Attached to: Internal Network**

5. Host-only Adapter

This setup is similar to an internal network but also gives the **host computer** a direct IP connection to the VM. The VM still doesn't have internet access unless combined with another adapter (e.g., NAT).

Use this if: You need to access the VM **directly from your host** (e.g., using RDP, SSH, ping, or file transfer). This is ideal for web development labs or local server testing.

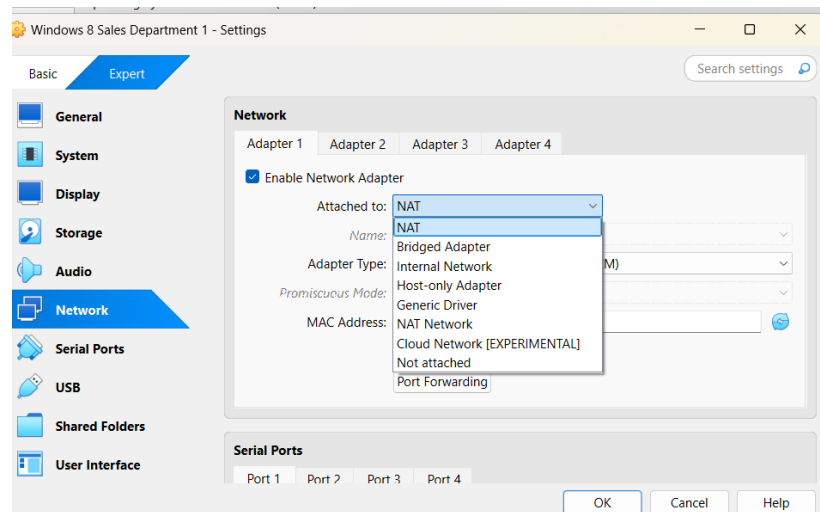
To set up a Host-only Adapter:

- Right-click your VM
- Select **Settings**
- Go to the **Network** tab
- Choose **Attached to: Host-only Adapter**

You can create a host-only network by select File > Host Network Manager

I'm going to leave all the NAT Network settings at their defaults so that is it! The **Virtual Network** is now set up and ready to use.

Now we configure the **Networking** tab of your VM to match your chosen network type. The **easiest and most flexible option** is to create and use a **NAT Network**, especially for multi-VM labs that require internet access.



Downloading Your Operating System ISO(s)

Now that your virtual network is ready, the next step is to download an **operating system (OS)** to install on your virtual machine.

While it's technically possible to install from a physical disc, the most common and convenient method is to use an **ISO file** a digital copy of the OS installation media.

To find an ISO file, the easiest approach is to search online. For example, if you're looking to install **Window Server**, simply search:

"Window Server ISO Download"

Make sure to download ISOs from official or trusted sources to avoid corrupted or unsafe files. Some operating systems like Ubuntu, Fedora, or Windows Evaluation Editions provide free, official ISOs on their websites.

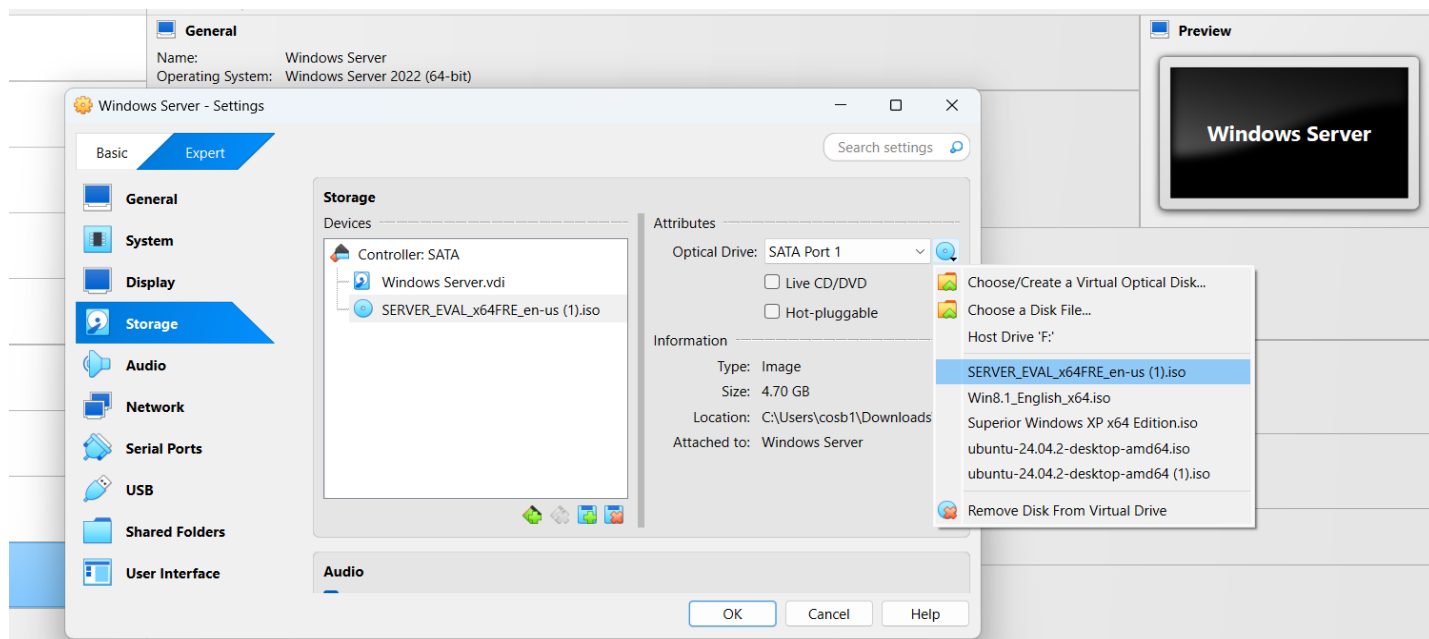
You should **ONLY** download ISOs from official websites.

Installing an OS on Your Lab VMs

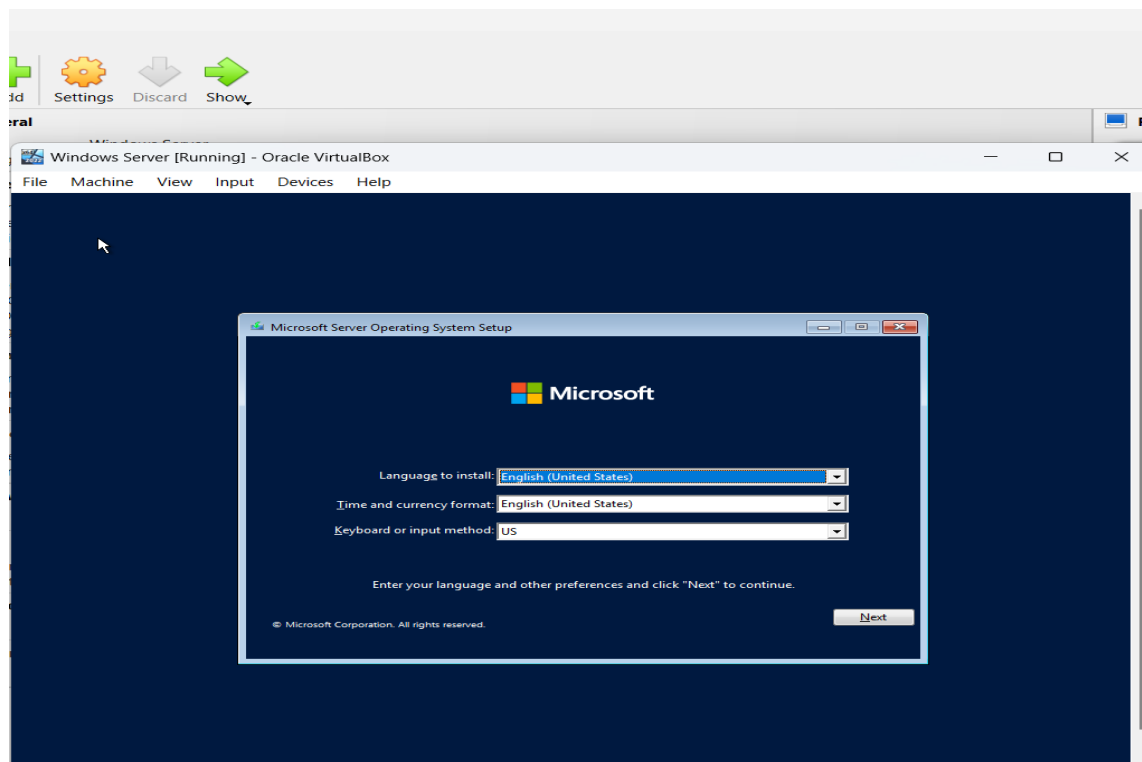
Once you have downloaded your ISO file, it's time to mount it to your VM in VirtualBox.

Mounting a VM is essentially like virtually inserting a disc into a DVD drive of a computer.

Right click on your VM and select settings then open the Storage tab. From there click the Empty disc icon, then again, the disc icon under Attributes on the right-hand side of the window. Click the Choose Virtual Optical Disk File... and browse to and open your desired ISO.



Click OK and now when we launch the VM you will be able to begin your OS installation. I mounted a Window Server ISO so when I launch the VM I will see the installation screen for this OS



That's it! You're now ready to run through your OS installation and complete your lab setup. With your virtual machine, virtual network, and operating system in place, you have everything you need to start building hands-on **IT labs** and gaining **real-world experience** — right from your own machine.

What You've Accomplished:

- Created a virtual machine
- Set up a virtual network
- Chosen and downloaded an operating system ISO
- Prepared everything for your OS installation

Now, go create something **awesome**. Whether you're chasing **certifications**, **testing software**, or designing your own **virtual network lab**, you're making great progress in sharpening your IT skills.