

General Settings:

1. Basic

These are fundamental settings for identifying and configuring your VM's base operating system.

- **Name:**
 - This is simply the name you assign to your virtual machine. It's how you'll recognize your VM within VirtualBox.
 - **Type:**
 - Refers to the type of operating system you're installing on the VM (e.g., **Windows**, **Linux**, **macOS**).
 - **Subtype:**
 - It specifies the **specific variant** or environment for the chosen OS. For example:
 - If you choose **macOS** as the Type, you might see options for different macOS releases like **Big Sur**, **Catalina**, or **Mojave**.
 - For Linux, there might not always be a "subtype" listed explicitly, but the **version** would often fulfill this role, specifying the particular distribution and version (e.g., Ubuntu 20.04, CentOS 7).
 - **Version:**
 - Specifies the version of the selected OS type. For example, if you chose "Linux" as the type, you would specify whether it's **Ubuntu 64-bit**, **Red Hat 32-bit**, etc.
-

2. Advanced

These settings allow you to customize how VirtualBox handles specific features and save states.

- **Snapshot Folder:**
 - A **snapshot** is a saved state of the virtual machine at a particular point in time. This feature allows you to **freeze** your VM at a certain state and return to that state later, like a backup.
 - The **snapshot folder** is where VirtualBox stores these saved states. You can take multiple snapshots if you want to be able to roll back to earlier versions of your VM.
- **Shared Clipboard:**
 - Controls whether you can copy and paste text or files between the **host** (your physical computer) and the **guest** (your VM). The options are:
 - **Disabled:** No clipboard sharing.
 - **Host to Guest:** You can copy content from the host machine and paste it into the VM.
 - **Guest to Host:** You can copy from the VM and paste to the host.
 - **Bidirectional:** Clipboard works both ways, between the host and the guest.

- **Drag 'n' Drop:**
 - Allows you to drag and drop files and folders between the host and the VM. Options are:
 - **Disabled:** No drag-and-drop sharing.
 - **Host to Guest:** You can drag files from the host system to the guest.
 - **Guest to Host:** Drag files from the VM to the host.
 - **Bidirectional:** Drag and drop in both directions.
-

3. Description

- You can write a custom **description** for your virtual machine. This is just a **note** for you or other users to describe the purpose of the VM, configurations, or any important details. It doesn't affect the VM's functionality.
-

4. Disk Encryption

- VirtualBox allows you to encrypt your VM's hard disk to protect its data.
- Disk encryption in VirtualBox allows you to secure your virtual machine's **virtual hard disk**. When you enable disk encryption, all of the data stored on your VM's virtual hard drive is **encrypted** using a specific algorithm. This means that even if someone gains access to the virtual disk file on your host system, they won't be able to read its contents without the decryption key (password).
 - **Disk Encryption Cipher:** This option lets you choose the **encryption algorithm** used to encrypt the VM's virtual disk. VirtualBox supports the **AES (Advanced Encryption Standard)** algorithm, which is a widely used and secure encryption method. There are two key lengths available:
 - **AES-XTS128:** AES (Advanced Encryption Standard) with a 128-bit key.
 - **AES-XTS256:** AES with a 256-bit key, which provides stronger encryption than 128-bit.

To enable disk encryption, you'll need to set a **password** that will be required to start the VM. Encryption increases security, but it may also slightly affect performance due to the overhead of encrypting/decrypting data.

System Settings:

1. Motherboard Settings

These settings define the basic hardware configuration for your virtual machine.

- **Base Memory:**
 - This is the amount of **RAM** (memory) allocated to the VM. The more memory you assign, the better your VM will perform, but it also depends on how much RAM your

host machine has. For example, allocating 2GB of memory would be enough for light tasks, while more intensive OSes or apps might need 4GB or more.

- **Boot Order:**

- Defines the order in which devices are checked for bootable media when starting the VM. **Bootable media** refers to any storage device or medium that contains a bootable operating system or software capable of launching the boot process for a computer or virtual machine. Common options include:

- **Floppy:** The virtual floppy drive.
- **Optical:** The virtual CD/DVD drive. If you have an ISO file (a disk image) mounted in the optical drive, the VM will attempt to boot from this first. This is often used for installing operating systems from ISO images.
- **Hard Disk:** The VM's virtual hard disk (usually the primary boot device). This option represents the virtual hard disk (e.g., .vdi, .vmdk, etc.) that stores the operating system and other data. It's usually set as the primary boot device, as most operating systems are installed on the hard disk.

Common file formats for virtual hard disks include:

- **.vdi:** VirtualBox's native format (Virtual Disk Image).
- **.vmdk:** Used by VMware (Virtual Machine Disk).
- **.vhd:** Used by Microsoft (Virtual Hard Disk).
- **Network:** Allows booting from a network (used for network installations).

- **Best Practices:**

- For installing an OS, make sure the Optical drive is at the top of the list (with your ISO mounted) so that the VM can boot from it.
- Once the OS is installed, you might want to set the Hard Disk as the primary boot device to speed up booting in the future.

- **Chipset:**

- **A chipset is a collection of electronic components in a computer or VM that manages the data flows between the processor, memory, and peripherals. Chipset** is the controller that connects the CPU with other parts of the system (e.g., memory, input/output devices).

- **Components:**

The chipset typically consists of:

- **Northbridge:** Handles communication between the CPU, RAM, and high-speed devices like graphics cards.
- **Southbridge:** Manages lower-speed peripherals, including storage devices, USB ports, and network connections.

- Common chipsets in VirtualBox include:

- **PIIX3 (Parallel I/O Controller):** Older chipset, typically used for older operating systems. It uses IDE, also known as ATA (Advanced Technology

Attachment), which is an older interface standard used for connecting hard drives, CD-ROM drives, and other storage devices to a computer's motherboard.

- **ICH9 (Intel I/O Controller Hub 9):** Newer, more modern chipset that supports up to 36 network adapters and is designed for newer operating systems. It uses SATA (Serial Advanced Technology Attachment), which replaced IDE as the primary method for hard drives and SSDs.

Facts:

- a) Data Transfer Speed of SATA is higher than IDE.
- b) SATA supports features like **hot swapping** which means replacing drives without shutting down the system.

- **TPM (Trusted Platform Module):**

- **TPM** stands for **Trusted Platform Module**. It's a specialized hardware component used in computers to enhance security, particularly in environments where data protection and secure authentication are critical.
- TPM is a dedicated microcontroller designed to secure hardware by integrating cryptographic keys into devices. It is a hardware-based security solution that provides a range of security functions.
- The TPM chip is usually soldered onto the motherboard of a computer or can be installed as a discrete chip in the system. It can also be found in some external devices.
- TPM can securely store cryptographic keys, passwords, and digital certificates. It helps protect sensitive information from unauthorized access or tampering.
- TPM can perform encryption and decryption operations using stored keys. This can be used for disk encryption solutions (like BitLocker in Windows) that protect data on hard drives by encrypting the entire disk.
- TPM can help verify the integrity of a system to a remote party. This means that if you're running a secure application on a machine, you can prove to a server or another system that your hardware and software haven't been tampered with.
- TPM can be used in conjunction with the secure boot process, ensuring that only trusted software is loaded during the startup sequence.
- When configuring a virtual machine in VirtualBox or other virtualization platforms, enabling TPM can provide additional security for the VM, particularly for operating systems (Windows 11) that utilize TPM for features like BitLocker or secure boot.

- **Pointing Device:**

- A pointing device is an input device that allows users to interact with a computer's graphical user interface (GUI) by moving a cursor on the screen. Common examples include a mouse, trackpad, and touchscreen.
- In a virtual machine environment, the pointing device setting determines how the VM interprets input from your physical mouse or trackpad and how it translates that input into actions within the virtual machine.
- Defines how the mouse is presented to the guest OS. Options include:

- **PS/2 Mouse:** This option emulates a traditional PS/2 mouse, which connects to the computer via a 6-pin connector. Common with older operating systems.



- **USB Tablet:** This emulates a USB tablet device, which is like a graphics tablet or touchscreen. It translates more precise touch or pen inputs into mouse movements on the screen. Ensures smoother mouse integration between host and VM.
- **USB Multitouch Tablet:** Supports multi-touch gestures in the guest OS (if supported). Supports gestures like pinch-to-zoom and swipe.
- **USB MT TouchScreen and TouchPad:** This option emulates a USB **multi-touch** touchscreen and **touchpad** input device. This means that the virtual machine can simulate the functionality of devices that support both touch and traditional pointer input i.e. Mouse.

- **Extended Features:**

- **Enable I/O APIC (Advanced Programmable Interrupt Controller):**

- APIC helps manage interrupts between the CPU and peripheral devices (e.g., keyboard, mouse). An **interrupt** is a signal sent by a device (such as a keyboard, mouse, or network card) to notify the CPU that it needs attention. The CPU then temporarily stops what it's doing to service the interrupt. **I/O APIC** is required for multi-core processing and 64-bit guest OSes.
- Note: APIC and Chipset both manages communication between CPU and I/O device but APIC manages Interrupt and Chipset manages Data Flow between CPU and I/O device.

- **Enable Hardware Clock in UTC Time:**

- Configures the virtual machine's hardware clock to be set to **UTC (Coordinated Universal Time)**, which is useful for some operating systems that expect the hardware clock to be in UTC rather than local time (like Linux).
- **Coordinated Universal Time (UTC)** is the global time standard that all time zones are measured from. It does not change with the seasons and is not affected by Daylight Saving Time.

- UTC is the same worldwide and does not change with local time zone shifts. Local times around the world are expressed as an offset from UTC. For example:
 - UTC+5:30 is the time zone for India Standard Time (IST).
 - UTC-5 is the time zone for Eastern Standard Time (EST) (used in places like New York).
- In the context of virtual machines or operating systems, when you see "Enable Hardware Clock in UTC," it means the VM or system will keep the system clock set to UTC, regardless of the local time zone. The operating system then applies the local time zone offset to display the correct time for the user.
- **Enable EFI (Special OSes only):**
 - **EFI (Extensible Firmware Interface) or UEFI (Unified Extensible Firmware Interface)** – modern version of EFI) is an advanced firmware that some operating systems (like Windows 10/11, macOS , or certain Linux distributions) use to boot instead of the older BIOS. Enable this for OSes that require UEFI.
 - It is a newer type of **firmware** that controls how your computer's hardware starts up and communicates with the operating system. It replaces the older **BIOS** (Basic Input/Output System) that was used in earlier computers.
 - Older operating systems (like **Windows XP** or **Windows 7**) don't need EFI or UEFI. These systems use the older BIOS method for booting.
 - Another possible use of EFI in VirtualBox is development and testing of EFI applications, without booting any OS.
- **Enable Secure Boot:**
 - **Secure Boot** is a security feature found in UEFI (Unified Extensible Firmware Interface) firmware. It ensures that a computer only boots using software that is trusted by the manufacturer.
 - **Secure Boot** is a feature that is specific to **UEFI (Unified Extensible Firmware Interface)** and is **not available with traditional BIOS (Basic Input/Output System)**.
 - This ensures that only digitally signed boot loaders and OS kernels are loaded during startup, which is an essential security feature (especially for Windows 11 and some Linux distributions). It helps prevent malicious code from running before the OS loads.

2. Processor Settings

These settings control the virtual CPU assigned to your VM.

- **Processors (CPU):**
 - Defines how many **CPU cores** (virtual processors) you want to assign to your VM. The more cores you allocate, the faster the VM can perform (especially for CPU-intensive

tasks), but be careful not to assign too many, as your host OS still needs resources to run.

- **Execution Cap:**

- Sets the **maximum percentage** of the host CPU that the VM can use. For example, if set to **50%**, the VM will use only half of the host's processing power i.e. if provided 4 CPU cores then it will use only half that is 2 CPU cores, which can help manage performance but may limit the VM's speed.

- **Extended Features:**

- **Enable PAE/NX (Physical Address Extension / No-eXecute):**

- **PAE** allows 32-bit operating systems to access more than 4GB of memory (though this is mostly relevant to older systems).
 - In a standard 32-bit architecture, the maximum addressable memory space is **2^{32}** , which equals **4,294,967,296 bytes** (or **4 GB**). This means that a 32-bit operating system can theoretically access up to 4 GB of RAM.
 - However, in practice, the usable memory for applications is even less due to the memory used by the operating system and hardware, often leaving only around 3 to 3.5 GB available for applications.
 - This limitation is a significant drawback for applications that require a lot of memory, such as databases, large data processing applications, or certain games.
 - As software demands increased over time, the limitation of 4 GB became a bottleneck for performance and capabilities.
- **NX (No-eXecute)** is a security feature that marks certain areas of memory as non-executable, preventing malicious code from running there (required by modern operating systems for security). This means that even if an attacker manages to inject malicious code into that memory region, the CPU will refuse to execute it, effectively blocking the attack.

- **Enable Nested VT-x/AMD-V:**

- Enables **nested virtualization**, allowing you to run another virtual machine inside the guest VM. This is called **Nested Virtualization**. For example, you could run VirtualBox inside a virtual machine. This feature is useful for advanced setups like testing hypervisors.

3. Acceleration Settings

These settings optimize the performance of your VM through virtualization technologies.

- **Paravirtualization Interface:**

- **Paravirtualization** allows the guest OS to **better interact** with the host hardware, improving performance. VirtualBox offers several options depending on the guest OS:
 - **Default:** VirtualBox automatically chooses the best option based on the guest OS.

- **None:** No paravirtualization interface is exposed to the guest.
 - **Minimal:** Provides basic paravirtualization support.
 - **KVM** (Kernel-based Virtual Machine): Suitable for Linux guests.
 - **Hyper-V:** Suitable for Windows guests, improving the interaction between the VM and host.
- In paravirtualization, the hypervisor provides a modified version of the guest OS that is aware of the virtual environment. Instead of translating all hardware calls (as in full virtualization), the guest OS makes special API calls to the hypervisor for certain operations, which can improve performance.
 - In paravirtualization, the guest OS uses **hypercalls** (similar to system calls) to request services from the hypervisor. This allows for efficient resource management and reduces the overhead associated with virtualization. For example, instead of the guest OS trying to manage interrupts or memory directly, it will use hypercalls to handle those tasks through the hypervisor.
- **Hardware Virtualization:**
 - This setting leverages the host's CPU virtualization features for faster VM performance.
 - **Enable Nested Paging:**
 - **Nested Paging** is a memory management feature that improves the performance of virtual memory operations. It enables the host CPU to handle memory more efficiently for the guest VM, reducing overhead and improving speed.
 - In traditional virtualization without nested paging, the hypervisor manages the mapping of guest virtual addresses to host physical addresses, often leading to increased overhead and complexity. Each layer of virtualization requires additional address translation, which can slow down performance.
 - With nested paging, the guest VM itself is allowed to manage its virtual memory mappings directly using hardware support, reducing the number of translations required and the workload on the hypervisor.
-

Display Settings:

1. Screen Settings

- **Video Memory**
 - **Definition:** Video memory (or **VRAM**, Video Random Access Memory) in VirtualBox refers to the amount of memory allocated to the virtual graphics card of the virtual machine (VM). This memory is used by the VM to store graphical data, such as textures, frames, and other graphical information needed to display the guest operating system's interface, applications, or games on the screen.

- **Importance:** Increasing video memory allows the VM to handle higher resolutions and better graphics performance, which is particularly important for graphic-intensive applications or operating systems.
 - **Monitor Count**
 - **Definition:** This option allows you to specify how many monitors the virtual machine can use.
 - **Importance:** Useful for setups requiring multiple displays, it enables users to extend their desktop environment across multiple screens, enhancing productivity and multitasking.
 - **Scale Factor**
 - **Definition:** The scale factor adjusts the size of the VM's display. It can be set to scale the display up or down, affecting how it appears on the host's screen.
 - **Importance:** This is especially useful when working with high-resolution displays or for users who need larger or smaller text and interface elements within the VM. For example, a 100% scale factor shows the VM at its native resolution. Increasing it to 150% or 200% enlarges the VM's display, making it easier to read or work with on high-resolution screens.
 - The scale factor changes the size of the display, not the resolution of the VM. So, if your guest OS is running at 1920x1080 resolution, changing the scale factor to 150% won't alter the resolution; it will only enlarge the display without reducing image quality.
 - **Graphics Controller**

The **Graphics Controller** in VirtualBox refers to the virtual graphics card that the virtual machine (VM) uses to display output, like the desktop interface of the guest operating system. A graphics controller in a virtual machine environment (like VirtualBox) simulates the functionality of a GPU within a virtual machine. It allows the VM to render graphics and display them on the host system. The options include:

 - **VMSVGA:** A modern graphics controller that supports 3D acceleration and is compatible with many guest operating systems, including Linux and Windows. It's suitable for most modern applications and environments.
 - **VBoxVGA:** An older graphics controller used primarily for compatibility with older guest OSes. It does not support 3D acceleration and is generally not recommended for modern systems.
 - **VBoxSVGA:** A hybrid controller that supports 3D acceleration while maintaining compatibility with older guest systems. It's a good option for Windows guests and provides decent performance.
 - **Extended Features: Enable 3D Acceleration**
 - **Definition:** This option enables 3D graphics support in the VM, allowing guest operating systems to use hardware acceleration for rendering graphics.
 - **Importance:** Enabling this feature can significantly improve graphics performance for applications that rely on 3D rendering, such as games or graphic design software.
-

2. Remote Display

- **Enable Server**

This option allows you to enable remote access to the virtual machine's display, which can be useful for remote management or accessing the VM from another device.

Options:

- **Server Port:** Specifies the port number on which the remote display server listens for incoming connections.
- **Security Method:** Allows you to choose the security protocol used for the remote connection, ensuring that data transmitted over the network is protected. The Security Method we have in our VirtualBox:
 - **TLS (Transport Layer Security):** TLS is a cryptographic protocol designed to provide secure communication over a computer network. When this option is selected, the connection between the remote client and the VM is encrypted using TLS, ensuring that data (such as session information, keystrokes, etc.) is protected from interception.
 - **RDP (Remote Desktop Protocol):** RDP is a proprietary protocol developed by Microsoft that allows users to connect to another computer over a network connection. It also has built-in security mechanisms. In this mode, the connection will rely on the security features provided by RDP itself.
 - **Negotiate:** The "Negotiate" option allows the remote display server and client to negotiate the best available security method. The client and server will automatically choose between TLS and RDP, depending on what both sides support.

NOTE: TLS vs RDP

- a) **TLS** is a **general-purpose** encryption protocol to secure data transmission over any type of connection.

RDP is designed **specifically** for remote access to a desktop, with built-in remote access and some encryption features.

- b) **TLS** can be used in a wide variety of network applications (e.g., web, email, file transfer).

RDP is primarily used for remote desktop connections.

- **Authentication Method:** Specifies how users are authenticated when they attempt to connect to a virtual machine's display or the remote display server.

Options:

- **External:** This method allows the use of external authentication mechanisms, such as operating system user accounts or third-party authentication services.
- **Guest:** This method allows users to authenticate using credentials that exist within the guest operating system of the virtual machine.
- **Null:** This option means that no authentication is required for remote display connections. Anyone can connect without needing to provide credentials.

- **Authentication Timeout:** Sets a time limit for authentication attempts, enhancing security by preventing unauthorized access.
 - **Extended Features: Allow Multiple Connections:** This option permits multiple clients to connect to the VM's display simultaneously, which can be useful for collaborative work or demonstrations.
-

3. Recording

- **Enable Recording**

- **Definition:** This setting allows you to record the VM's display output, which can be useful for creating tutorials, presentations, or documentation.

Options:

- **Recording Mode:** Specifies how the recording will be performed (e.g., full-screen or windowed). The options we have here:
 - Video Only
 - Audio Only
 - Video/Audio
 - **File Path:** Determines where the recorded video file will be saved on the host machine.
 - **Frame Rate:** Sets the number of frames captured per second in the recording, affecting the smoothness and quality of the video.
 - **Frame Size:** Defines the resolution of the recorded video, influencing the clarity and file size.
 - **Video Quality:** Adjusts the quality of the recorded video, impacting both visual fidelity and file size.
 - **Audio Quality:** Specifies the quality of audio recording, which can enhance the overall viewing experience when sound is included in the video.
 - **Screens:** This option allows you to choose which monitor's output to record if you are using multiple monitors.
-

Storage Settings:

1. Devices

- **Controllers:** In VirtualBox, a controller is a virtual device that manages storage devices attached to a virtual machine (VM). Two common types of controllers are:
 - **IDE (Integrated Drive Electronics):** A traditional controller that connects hard drives and optical drives. IDE is older and less commonly used with modern operating systems but is still supported for compatibility.
 - **SATA (Serial ATA):** A newer standard for connecting hard drives and solid-state drives (SSDs). SATA is faster than IDE and supports advanced features like hot-swapping (connecting and disconnecting drives while the system is running).

2. Attributes

This section allows you to configure the properties of the storage devices connected to your VM.

- **Name:** The name you assign to the storage device. It's used for identification within the VirtualBox interface.
- **Type:** This specifies the type of storage controller used. Each type has its own characteristics:
 - **AHCI (Advanced Host Controller Interface):** This is a newer standard for SATA controllers that supports features like native command queuing and hot-swapping.
 - **PIIX3 and PIIX4:** Older IDE controllers. PIIX4 supports additional features compared to PIIX3, such as better support for DMA (Direct Memory Access).
 - **ICH6:** An Intel chipset that provides additional features and improved performance over PIIX controllers.
 - **LsiLogic and BusLogic:** These are used primarily for SCSI (Small Computer System Interface) devices. They are suitable for enterprise-level virtual machines requiring SCSI support.
 - **I82078:** A controller type for floppy disk drives.
 - **LsiLogic SAS:** A SCSI controller specifically designed for Serial Attached SCSI (SAS) devices, offering advanced performance and scalability features.
 - **USB:** Allows you to connect USB storage devices directly to the VM.
 - **NVMe (Non-Volatile Memory Express):** A high-performance interface for SSDs that provides faster data transfer rates compared to SATA. Fastest among all but after selecting it whether you will see performance improvement or not depend upon various factors like, guest OS, Host Storage Type, etc.
 - **virtio-scsi:** A paravirtualized SCSI controller that improves performance in guest operating systems by reducing overhead.
- **Port Count:** This indicates the number of ports available for connecting storage devices to the controller. For example, an IDE controller typically has 2 ports (primary and secondary), while a SATA controller can have multiple ports depending on its configuration.
- **Use Host I/O Cache:** This option allows the VM to use the host's I/O cache, which can improve disk performance. It enables the VM to cache disk I/O operations in the host memory, reducing latency for disk access. However, *it may also lead to data inconsistency if the VM is not shut down properly, as cached data may not be written back to the disk immediately.*

Audio Settings:

1. Enable Audio:

- This option allows you to enable or disable audio support for the virtual machine. If enabled, the VM can play sound through the host's audio system.
-

2. Host Audio Driver:

- This setting allows you to choose the audio driver used by the host machine for the VM's audio output. The options include:
 - **Default:** Uses the default audio driver of the host operating system.
 - **Windows Audio Session:** Uses the Windows Audio Session API (WASAPI) for more advanced audio management on Windows hosts, allowing for better audio quality and performance.
 - **Windows DirectSound:** Uses the older DirectSound API for audio output on Windows. It may be less efficient than WASAPI.
 - **Null Audio:** Disables audio output, effectively muting the VM. This is useful if you don't need sound and want to save resources.
-

3. Audio Controller:

- This setting defines the type of virtual audio hardware i.e., sound card that the VM will emulate. The options include:
 - **ICH AC97:** An older audio standard; provides basic audio support suitable for many applications but may not support advanced features.
 - **Sound Blaster 16:** An emulation of a classic sound card widely used in older systems. It offers decent compatibility for older applications that require legacy audio support.
 - **Intel HD Audio:** A modern audio controller that supports high-definition audio. It's the recommended option for most modern operating systems and applications, providing better audio quality and features.
-

4. Extended Features:

- **Enable Audio Output:** Allows audio output from the VM. If disabled, no sound will be played.
 - **Enable Audio Input:** Allows the VM to capture audio input from the host system's microphone or other audio devices. This is useful for applications that require voice input or audio recording.
-

Network Settings:

1. Enable Network Adapter:

- This option allows you to enable or disable the network adapter for the virtual machine. When enabled, the VM can connect to a network and communicate with other devices.
-

2. Attached to:

- This setting specifies the type of network connection for the VM. Each option determines how the VM interacts with the network:

- **Bridged Adapter:** Connects the VM directly to the physical network through the host's network adapter. The VM gets its own IP address on the network, allowing it to communicate with other devices as if it were a physical machine.
 - **NAT (Network Address Translation):** Allows the VM to share the host's IP address. The VM can access external networks (like the internet), but incoming connections to the VM are not allowed by default.
 - **Internal Network:** Creates a private network shared only between VMs on the same host that are configured to use this option. These VMs can communicate with each other but not with the outside world or the host.
 - **Host-Only Adapter:** Creates a network that allows communication between the VM and the host only. The VM cannot access the external network or the internet, but it can interact with the host machine.
 - **Generic Driver:** The Generic Driver option in VirtualBox allows you to use a custom network interface for your virtual machine (VM). This option is designed for advanced users who need specific network configurations that are not covered by the default network options.
 - **NAT Network:** Similar to NAT but allows multiple VMs to share a common network. VMs can communicate with each other and the outside network.
 - **Cloud Network:** Connects the VM to a cloud provider, enabling cloud-based services.
 - **Not Attached:** Disables the network adapter, preventing the VM from connecting to any network.
-

3. Name:

- This field allows you to specify or view the name of the network adapter. The name typically reflects the underlying host network adapter being used (e.g., Ethernet, Wi-Fi).
-

4. Adapter Type:

- This setting specifies the type of network adapter emulation used by the VM. Options include different virtual network interface cards (NICs) that may offer various levels of performance and compatibility. Common types include Intel PRO/1000, Paravirtualized Network (virtio-net), etc.
-

5. Promiscuous Mode:

- Promiscuous Mode is a network interface mode that allows a network device (like a virtual network adapter) to intercept and read all traffic on the network segment it's connected to, regardless of the destination address
- This mode is often used for network monitoring and packet capturing. It allows a device to see all packets flowing through the network, making it useful for troubleshooting, network analysis, or security monitoring.
- In VirtualBox, you typically have three options for Promiscuous Mode:

- **Deny:** The virtual network adapter will ignore all network traffic that is not specifically addressed to it. This is the default setting, providing normal operation where the VM can only see its own traffic.
 - **Allow VMs:** The virtual network adapter can see traffic that is addressed to other VMs on the same host. This is useful if you want to enable communication between multiple VMs on the same network segment while still limiting visibility of traffic from outside.
 - **Allow All:** The virtual network adapter will see all traffic on the network segment, including traffic not addressed to it. This is useful for network monitoring tools or when you want the VM to capture all network traffic for analysis.
-

6. MAC Address:

- This is the unique identifier for the virtual network adapter. You can either use the default generated MAC address or specify a custom one. The MAC address is important for network communication and must be unique on the network.
-

7. Cable Connected:

- This option simulates whether a network cable is connected to the virtual adapter. If enabled, the VM can send and receive network traffic. If disabled, the VM cannot communicate over the network, regardless of other settings.
-

Serial Ports Settings:

1. Enable Serial Port:

- The Serial Port settings in VirtualBox allow you to configure a virtual serial port for your virtual machine (VM). Serial ports can be used for various purposes, such as connecting to external devices, debugging, or managing network communications.

Options:

- **Port Number:**
 - This defines which serial port the VM will use. Commonly used port numbers include COM1, COM2, etc., depending on your host system configuration.
- **IRQ (Interrupt Request Line):**
 - This specifies the interrupt request line for the serial port. It's used by the CPU to communicate with peripheral devices.
 - This is a number that tells the CPU which device needs attention. Each serial port uses a specific IRQ number.
 - **Example:** The default IRQ for COM1 is usually **4**. You don't need to change it unless you know what you're doing.
- **I/O Port:**

- This is the input/output port address for the serial port.
- This is the address that the operating system uses to communicate with the serial port. Different ports have different addresses.
- **Example:** COM1 typically uses the I/O port address **0x3F8**.
- **Port Mode:**
 - This setting determines how the serial port will operate. The options include:
 - **Disconnected:** The serial port is not connected to any device or resource. It's like turning the serial port off.
 - **Host Pipe:**
 - A pipe is a method used for inter-process communication (IPC) that allows data to flow from one process (program) to another.
 - When one process writes data to the pipe, another process can read that data. Pipes can be either **anonymous** (temporary) or **named** (persistent).
 - A **named pipe** is a specific type of pipe that has a name in the file system, allowing processes to communicate with each other using that name.
 - Unlike anonymous pipes, named pipes remain available in the file system until explicitly deleted.
 - **Cross-Process Communication:** Multiple processes can read from and write to the **named pipe**, which makes it suitable for more complex communications.
 - In the context of virtualization (like in VirtualBox), a **host pipe** refers to a named pipe that is used for communication between a virtual machine and the host machine (your computer).
 - **Host Pipe** allows a virtual machine to send data to a program running on the host or receive data from the host.
 - **Host Device:** The serial port connects directly to a physical serial port on the host machine.
 - **Raw File:** The serial port can send data to a raw file on the host. Data written to the serial port will be saved in this file.
 - Imagine you have a VM running a program that outputs diagnostic data to a serial port. By connecting that port to a raw file, you can:
 - Specify a file path (e.g., C:\logs\serial_output.txt).
 - As the program runs, all output sent to the serial port will be captured in this text file.
 - You can open the text file later to review the output and diagnose issues.
 - **TCP:** The serial port can connect over a TCP/IP network to another machine. This is useful for network-based communication.
- **Connect to Existing Pipe/Socket:**
 - If you select Host Pipe or TCP, this option allows you to specify an existing named pipe or network socket that the virtual serial port will connect to.

- **Path/Address:**
 - This field specifies the path or address for the connection, such as the named pipe path on the host or the IP address and port number for a TCP connection.
-

USB Settings:

1. Enable USB Controller

This setting allows you to activate USB support for your virtual machine, enabling it to recognize and interact with USB devices connected to your host machine.

Options

- **USB 1.1 (OHCI) Controller:**
 - **OHCI** stands for **Open Host Controller Interface**.
 - This option enables support for USB 1.1 devices, which operate at speeds of up to 12 Mbps.
 - It's useful for older USB devices that do not require higher speeds.
 - **USB 2.0 (OHCI + EHCI) Controller:**
 - **EHCI** stands for **Enhanced Host Controller Interface**.
 - This option enables support for USB 2.0 devices, which can operate at speeds of up to 480 Mbps.
 - USB 2.0 is backward compatible with USB 1.1, meaning that you can use USB 1.1 devices with this controller.
 - This is the recommended option for most modern USB devices, as it provides much higher data transfer rates than USB 1.1.
 - **USB 3.0 (xHCI) Controller:**
 - **xHCI** stands for **eXtensible Host Controller Interface**.
 - This option enables support for USB 3.0 devices, which offer significantly improved performance compared to USB 2.0, allowing data transfer rates of up to 5 Gbps (gigabits per second).
 - **Backward Compatibility:** USB 3.0 is backward compatible with USB 2.0 and USB 1.1 devices. This means you can still use older USB devices with the USB 3.0 controller, but they will operate at their respective speeds.
-

2. USB Device Filters

- **Definition:** USB device filters allow you to specify certain USB devices that the virtual machine should recognize and connect to. When a device that matches the filter is plugged into the host machine, VirtualBox can automatically connect it to the VM.
- **How It Works:**

- You can create filters based on various attributes of USB devices, such as vendor ID and product ID.
 - This way, you can ensure that specific devices are always redirected to the virtual machine, preventing the host from using them.
 - **Usage Example:** If you have a USB printer and you always want it to be available to your VM, you can create a filter for that printer. Whenever you connect it to your host machine, it will automatically be directed to the VM.
-

Shared Folder Settings:

Shared folders allow you to create a link between a folder on your host machine and the virtual machine (VM), enabling easier file transfer and access between the two environments.

1. Name

- **Definition:** This is the name you assign to the shared folder within VirtualBox. It will be used to reference the folder from within the guest operating system.
 - **Example:** If you name your shared folder "SharedDocs," you will access it in the VM under this name.
-

2. Path

- **Definition:** This is the path on the host machine where the folder resides. It points to the actual location of the folder you want to share.
 - **Example:** If your shared folder is located at C:\Users\YourUser\Documents\SharedFolder, that is the path you would enter.
-

3. Access

- **Definition:** This setting determines the type of access the VM has to the shared folder.
 - **Read-Only:** The VM can only view files in the shared folder but cannot modify or add new files.
 - **Read/Write:** The VM can view, modify, add, or delete files in the shared folder.
 - **Importance:** Choose the access level based on how you plan to use the shared folder. If you need to save files from the VM to the host, you should select Read/Write.
-

4. Auto Mount

- **Definition:** If this option is enabled, the shared folder will automatically be mounted when the VM starts.
- **Usage:** Enabling auto-mount is convenient because you won't need to manually mount the folder each time you start the VM. It will appear in the VM's file system automatically.

5. At

- **Definition:** This specifies where the shared folder will be mounted in the guest operating system.
 - **Common Locations:**
 - In Linux, shared folders are typically mounted under `/media/sf_<Name>` or `/mnt/hgfs/<Name>`.
 - In Windows, it may be accessible as a network drive (e.g., `\\VBOXSVR\<Name>`).
 - **Customization:** You can choose to change this location if desired, but it's generally best to use the default settings for easier access.
-

User Interface Settings:

1. Top Toolbar

The top toolbar in VirtualBox contains various menus that provide access to different functions and settings. Here's a breakdown of the options you will find:

- **File:** Contains options for managing the VirtualBox application and its settings, including:
 - **Preferences:** Opens the Preferences window, allowing you to configure general settings, input options, network settings, extension management, proxy settings, and auto-update preferences.
 - **Reset All Warnings:** Clears all warning messages, allowing you to reset the warning state and ensure that any future warnings will be displayed.
 - **Close:** Exits the VirtualBox application, prompting you to handle any running virtual machines before closing.
- **Machine:**
 - This menu allows you to manage the currently selected virtual machine. The options we have:
 - **Settings:** Opens the settings dialog for the selected virtual machine, allowing you to configure various aspects like system settings, storage, network, and more.
 - **Show Logs:** Displays logs related to the virtual machine's operations, which can be useful for troubleshooting issues.
 - **ACPI Shutdown:** Initiates a graceful shutdown of the virtual machine using the Advanced Configuration and Power Interface (ACPI), similar to shutting down a physical computer.
 - **Take Snapshot:** Captures the current state of the virtual machine, allowing you to revert back to this state later if needed. Snapshots are useful for preserving the VM's status before making changes.

- **Session Information:** Provides details about the current session, including performance statistics and VM state.
- **File Manager:** Opens the file manager for accessing files related to the virtual machine, such as disk images and configuration files.
- **Pause:** Pauses the virtual machine, freezing its current state. This allows you to temporarily halt operations without shutting down the VM.
- **Reset:** Restarts the virtual machine, effectively powering it off and back on, similar to a hard reboot.
- **View:**
 - Adjusts the display settings of the VirtualBox interface, including:
 - **Full Screen Mode:** Switches the virtual machine to full-screen view, maximizing its display to cover the entire monitor.
 - **Seamless Mode:** Integrates the virtual machine's windows with the host operating system, allowing you to use applications from the VM alongside your host applications seamlessly.
 - **Scaled Window:** Resizes the virtual machine's display to fit within the window, allowing for more flexible usage.
 - **Adjust Window Size:** Automatically adjusts the virtual machine's display size to match the size of the window.
 - **Auto-Resize Guest Display:** Automatically resizes the guest operating system's display to match the host window size when you resize it.
 - **Take Screenshot:** Captures an image of the current screen in the virtual machine, which can be saved for documentation or sharing purposes.
 - **Recording:** Starts or stops the recording of the virtual machine's screen, useful for creating tutorials or demonstrations.
 - **Remote Display:** Provides options to configure and manage remote display settings.
 - **Menu Bar:** Shows or hides the menu bar in the VM window.
 - **Status Bar:** Shows the current status of the virtual machine, including performance metrics and network status.
 - **Virtual Screen 1:** Allows you to select which virtual screen to view if multiple screens are configured.
- **Input:**
 - This menu offers controls for managing input devices.
 - You can capture the keyboard and mouse input, or release it back to the host, along with options to configure keyboard shortcuts.
 - **Keyboard:** Configures keyboard settings, including shortcuts and input options.

- **Mouse Integration:** Enables or disables mouse integration, allowing seamless control between the host and the guest operating systems.
 - **Devices:**
 - This menu allows you to attach or detach devices to the VM while it is running.
 - You can add USB devices, optical disks, or floppy disks, and manage shared folders from this menu.
 - **Debug:**
 - This menu is primarily for developers and advanced users.
 - It provides tools and options for debugging the virtual machine.
 - **Help:**
 - This menu offers access to VirtualBox documentation, the VirtualBox User Manual, and other resources.
 - You can also check for updates or view the About section.
-

2. Visual State

- The visual state refers to how the VM is presented on your screen. This includes options to view the VM in different display modes such as:
 - **Normal Window:** The VM runs in a windowed mode within your desktop environment.
 - **Full-Screen Mode:** The VM occupies the entire screen, providing an immersive experience.
 - **Seamless Mode:** This mode integrates the guest and host environments, allowing you to use applications from the VM alongside those on the host without window borders.
-

3. Mini Toolbar

- The **Mini Toolbar** is a floating toolbar that appears when you hover over the VM window.
- It provides quick access to frequently used functions without needing to navigate through the top menu.
- Common functions include:
 - **Power Options:** Start, pause, or reset the VM.
 - **View Options:** Switch between display modes (normal, full-screen, seamless).
 - **Device Attachments:** Quickly access USB and other devices.