Working with Virtual Machines

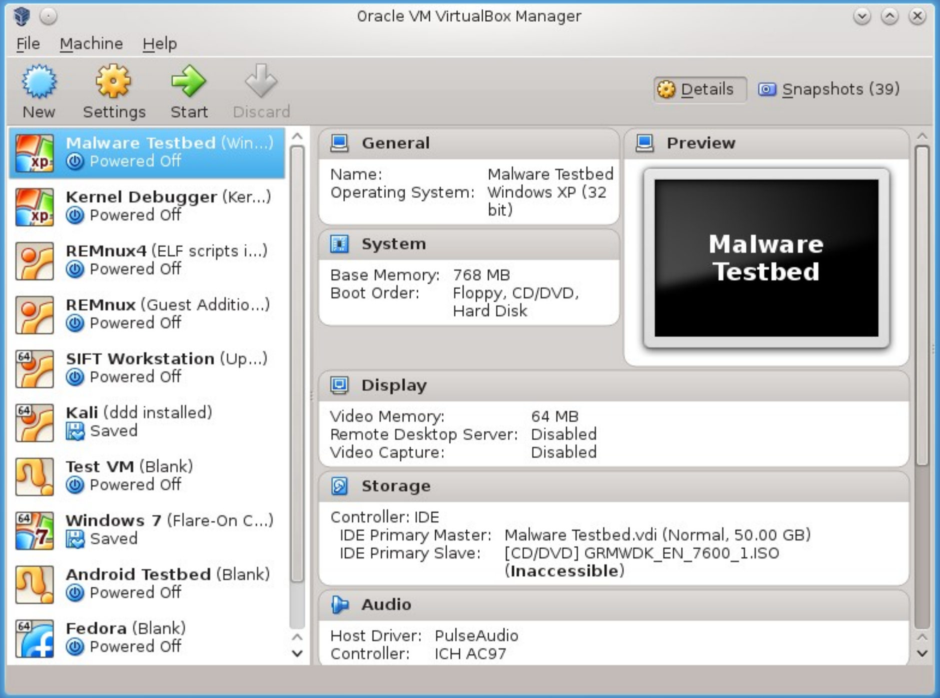
Why Virtual Machines?

- Experimentation: try, break, restore.
- Allows use of multiple OSs on one system.
  - Mac running Windows, Linux, vice-versa, etc.
- Testing malware.
  - Let it run loose in a closed environment, restore when finished.
  - Take full memory dumps, analyze.

Setting up VirtualBox

Starting VirtualBox

- Main window displays VMs, settings, options.



Malware Testbed (Win...)
Powered Off

Kernel Debugger (Ker...)
Powered Off

REMnux4 (ELF scripts i...)
Powered Off

REMnux (Guest Additio...)
Powered Off

SIFT Workstation (Up...)
Powered Off

Kali (ddd installed)
Saved

Test VM (Blank)
Powered Off

Windows 7 (Flare-On C...)
Saved

Android Testbed (Blank)
Powered Off

Fedora (Blank)
Powered Off

General

```
Name:               Malware Testbed
Operating System:   Windows XP (32
                    bit)
```

System

```
Base Memory:   768 MB
Boot Order:    Floppy, CD/DVD,
               Hard Disk
```

Preview

Malware Testbed

Display

```
Video Memory:            64 MB
Remote Desktop Server:   Disabled
Video Capture:           Disabled
```

Storage

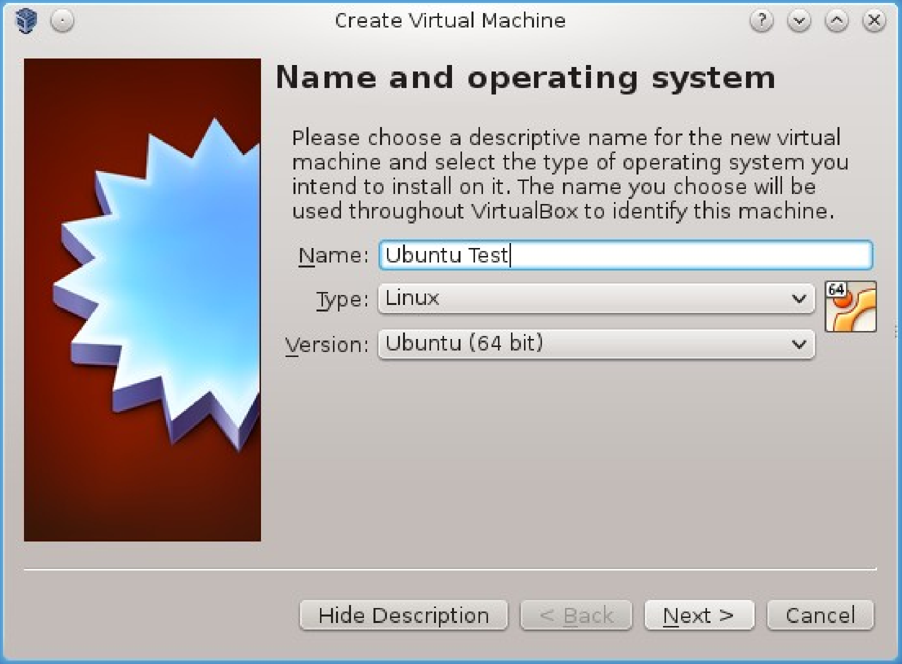
```
Controller: IDE
 IDE Primary Master:   Malware Testbed.vdi (Normal, 50.00 GB)
 IDE Primary Slave:    [CD/DVD] GRMWDK_EN_7600_1.ISO
                       (Inaccessible)
```

Audio

```
Host Driver:   PulseAudio
Controller:    ICH AC97
```

Setting Up a VM

- Click “New” to start wizard.
- The specific choice of OS isn't important – just make sure 64/32-bit setting is correct.



Create Virtual Machine

Name and operating system

Please choose a descriptive name for the new virtual machine and select the type of operating system you intend to install on it. The name you choose will be used throughout VirtualBox to identify this machine.

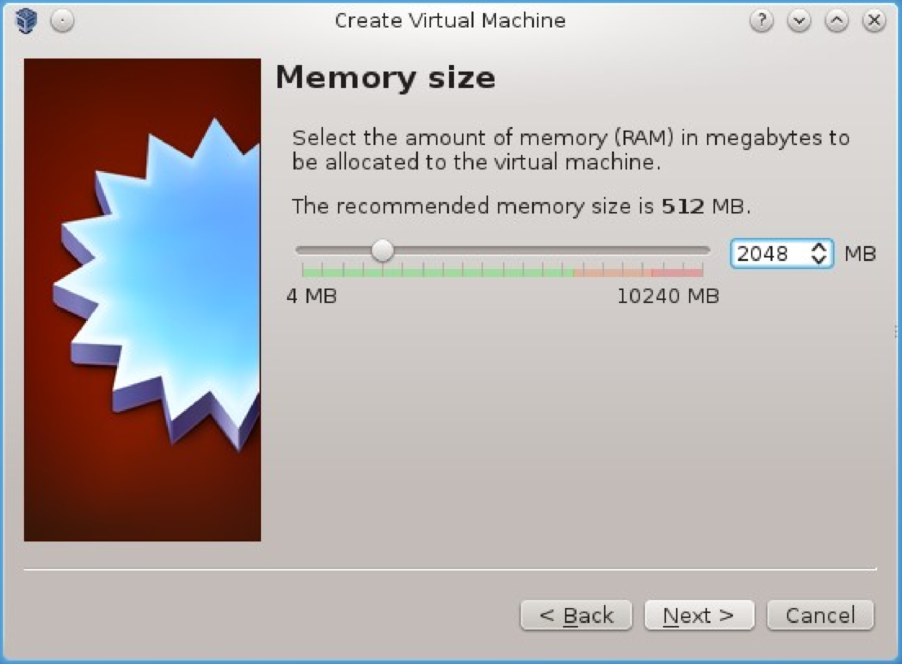
Name: Ubuntu Test

Type: Linux

Version: Ubuntu (64 bit)

Setting Up a VM (2)

- Amount of memory depends on the OS
  - Windows XP can do 384-512MB minimum
  - Linux can vary depending on the DE and programs to be run
  - More is always better



Create Virtual Machine

Memory size

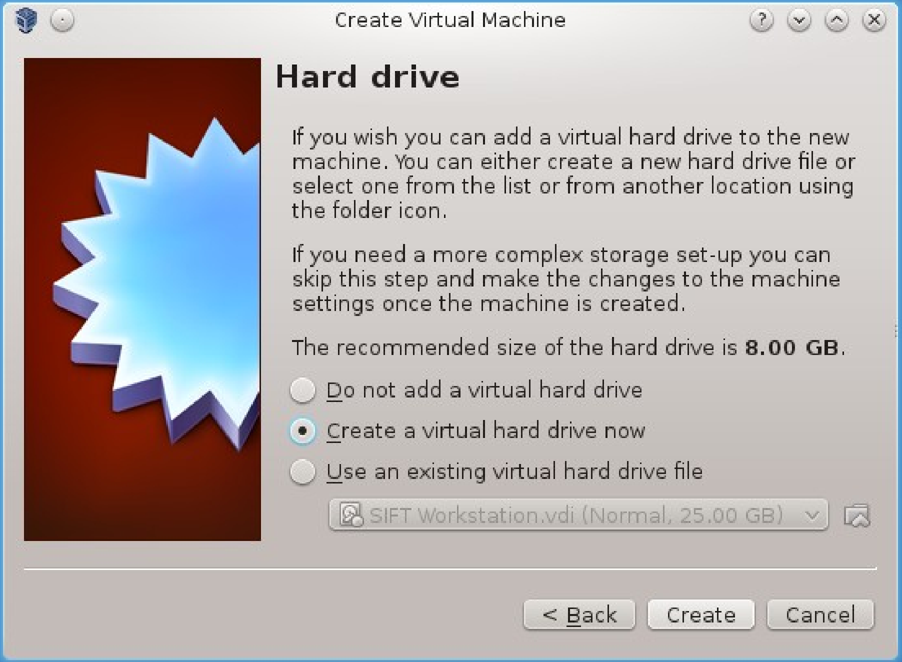
Select the amount of memory (RAM) in megabytes to be allocated to the virtual machine.

The recommended memory size is **512** MB.

2048 MB

Setting Up a VM (3)

- Create a virtual HDD
  - A file on the host that represents VM's HDD.
  - Choose “Dynamically Expanding” to start the file small and let it grow.
  - Again, size depends on OS and what the OS will run.
  - VDI format is VirtualBox standard; VMDK works and is compatible with VMWare (with/without tweaking?)



Hard drive

If you wish you can add a virtual hard drive to the new machine. You can either create a new hard drive file or select one from the list or from another location using the folder icon.

If you need a more complex storage set-up you can skip this step and make the changes to the machine settings once the machine is created.

The recommended size of the hard drive is **8.00 GB**.

○ Do not add a virtual hard drive

● Create a virtual hard drive now

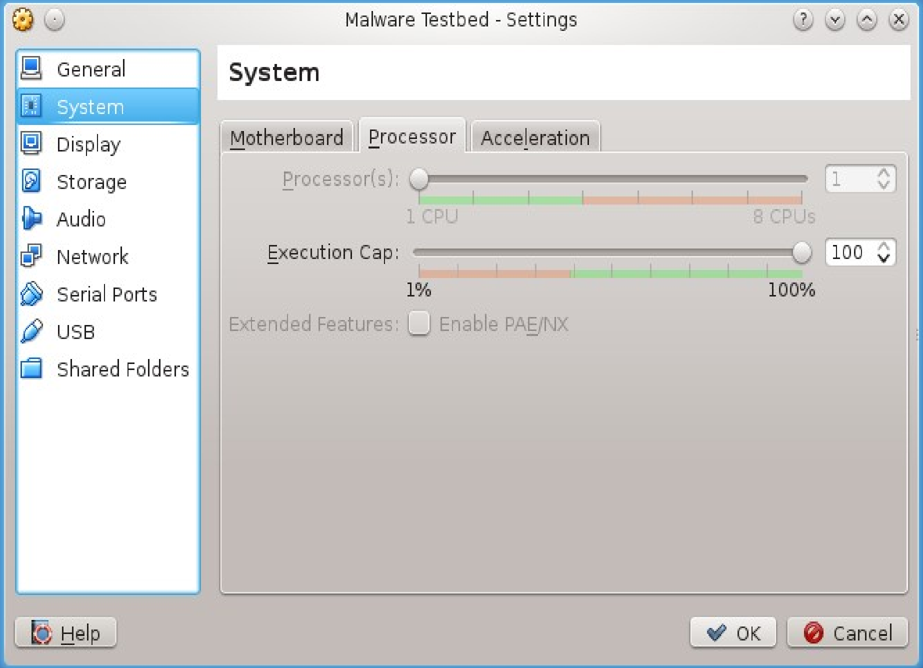
○ Use an existing virtual hard drive file

SIFT Workstation.vdi (Normal, 25.00 GB)

Configuring a VM

Configuring a VM

- Number of processors: depends on how much power is needed.
- More than the physical CPU cores on host is not recommended.



System

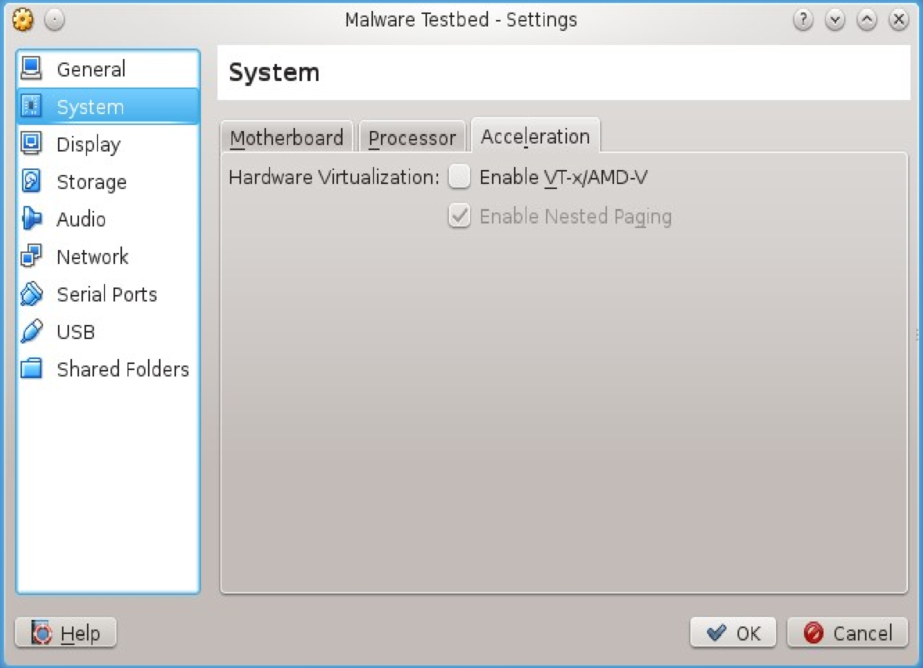
Processor(s): 1

Execution Cap: 100

Extended Features: ☐ Enable PAE/NX

Configuring a VM (2)

- VT-x is required for 64-bit VMs or VMs running multiple cores.



Malware Testbed - Settings

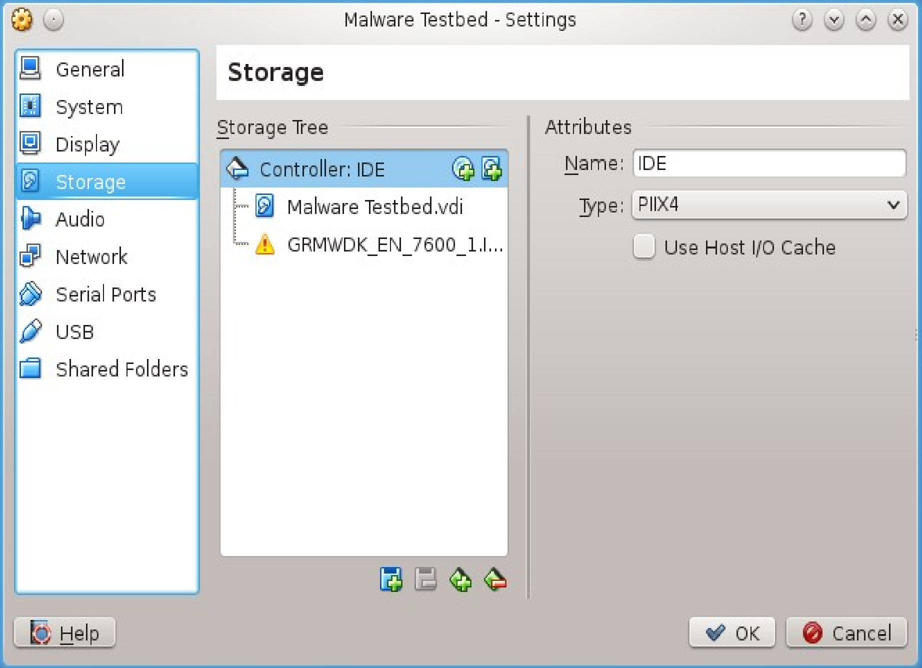
System

Hardware Virtualization: □ Enable VT-x/AMD-V

☑ Enable Nested Paging

Configuring a VM (3)

- The storage controller can be configured here (SATA, IDE, etc.)
- If an optical drive is setup here, the “disk” can be put in.
  - Virtual: .iso disk images.
  - Physical: use host's drive.
- Some OSs and filesystems recommend to use Host I/O Cache.
  - Slows performance if not necessary.



Malware Testbed - Settings

Storage

Storage Tree

Controller: IDE
Malware Testbed.vdi
GRMWDK_EN_7600_1.I...

Attributes

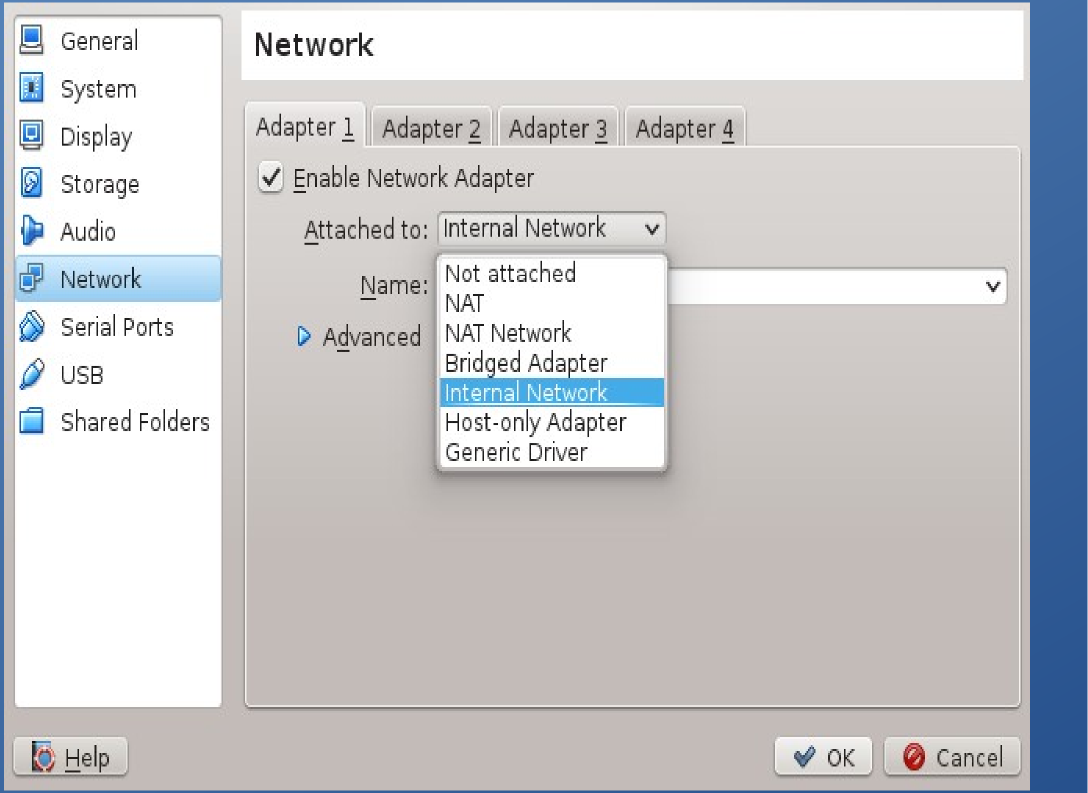
Name: IDE

Type: PIIX4

□ Use Host I/O Cache

Configuring a VM (4)

- NAT: host acts as the NAT router to the VM.
- Bridged: VM uses host NIC as its own – both get an IP.
- Internal Network: only VMs can talk to each other in their own virtual network. Many virtual switches can be created.
- Host-only Adapter: Like “Internal Network,” but allows communication to the host OS, as well.



Network

Adapter 1 | Adapter 2 | Adapter 3 | Adapter 4

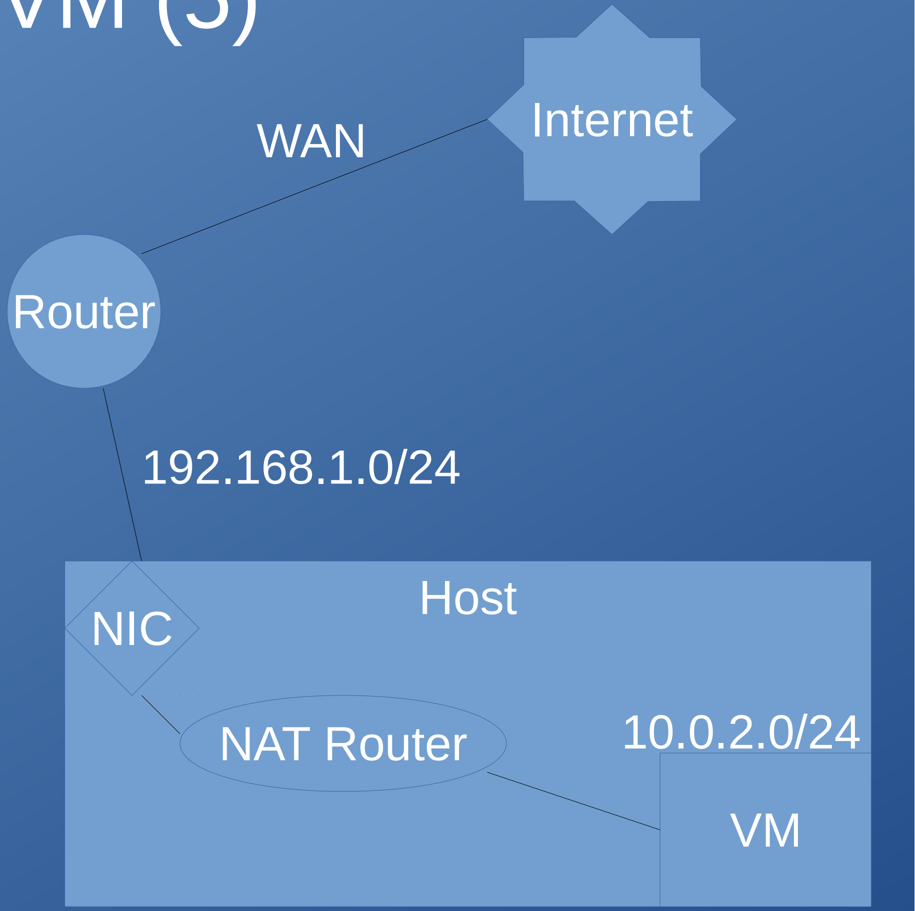
☑ Enable Network Adapter

Attached to: Internal Network

Name:

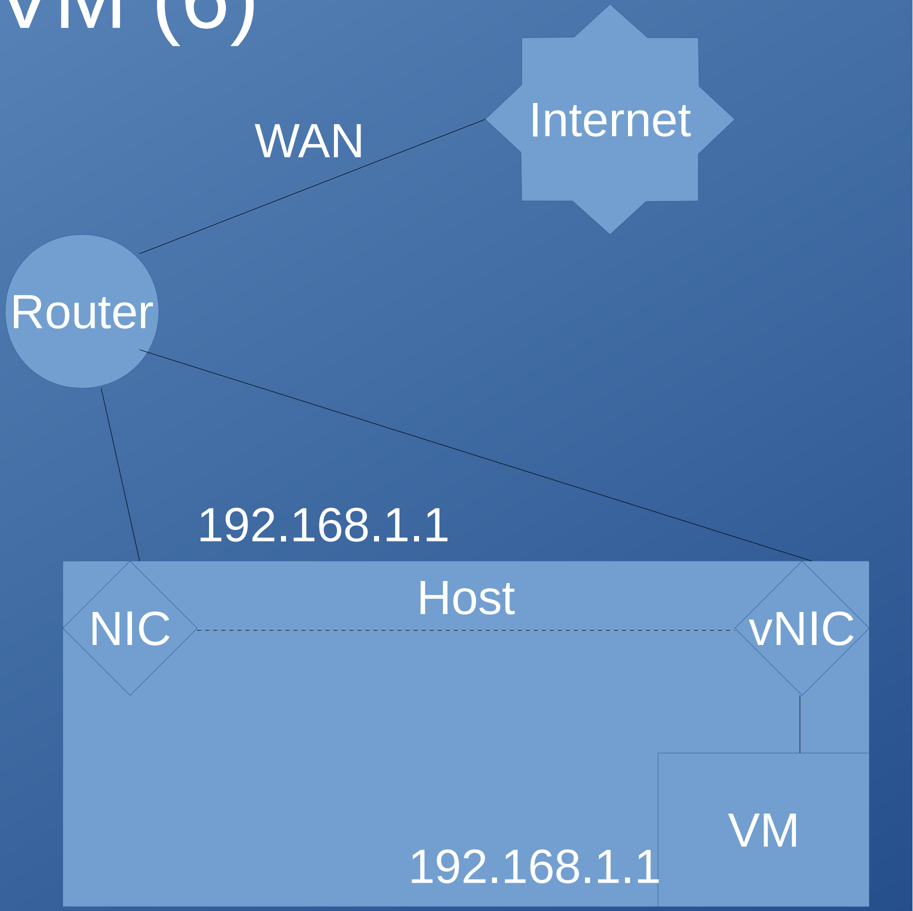
Configuring a VM (5)

- NAT: host acts as the NAT router to the VM.
  - No way to get to the VM without port forwarding.
  - VM can connect to machines on the Router network



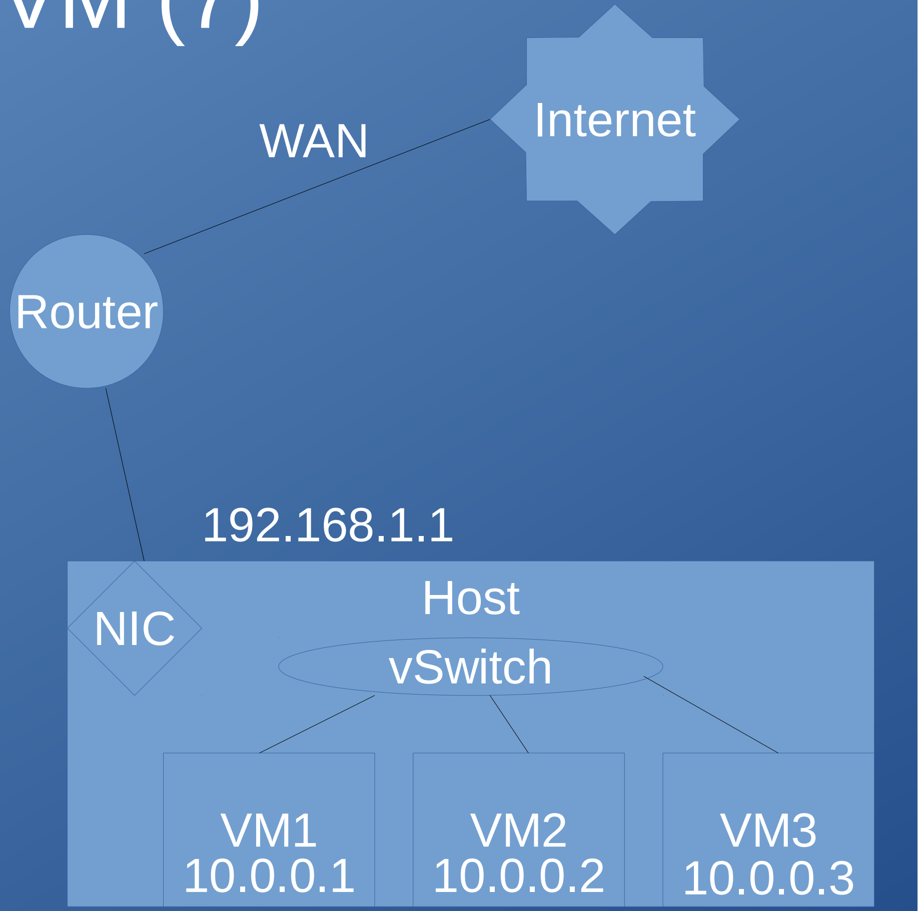
Configuring a VM (6)

- Bridged: VM uses host NIC as its own – both get an IP.
  - VNIC is really NIC, but it appears to Router as a separate device (unique MAC, gets its own IP.)
  - Host will not “see” traffic to/from VM.
  - Communication to and from VM is possible.
  - Good for simulating devices on a network.



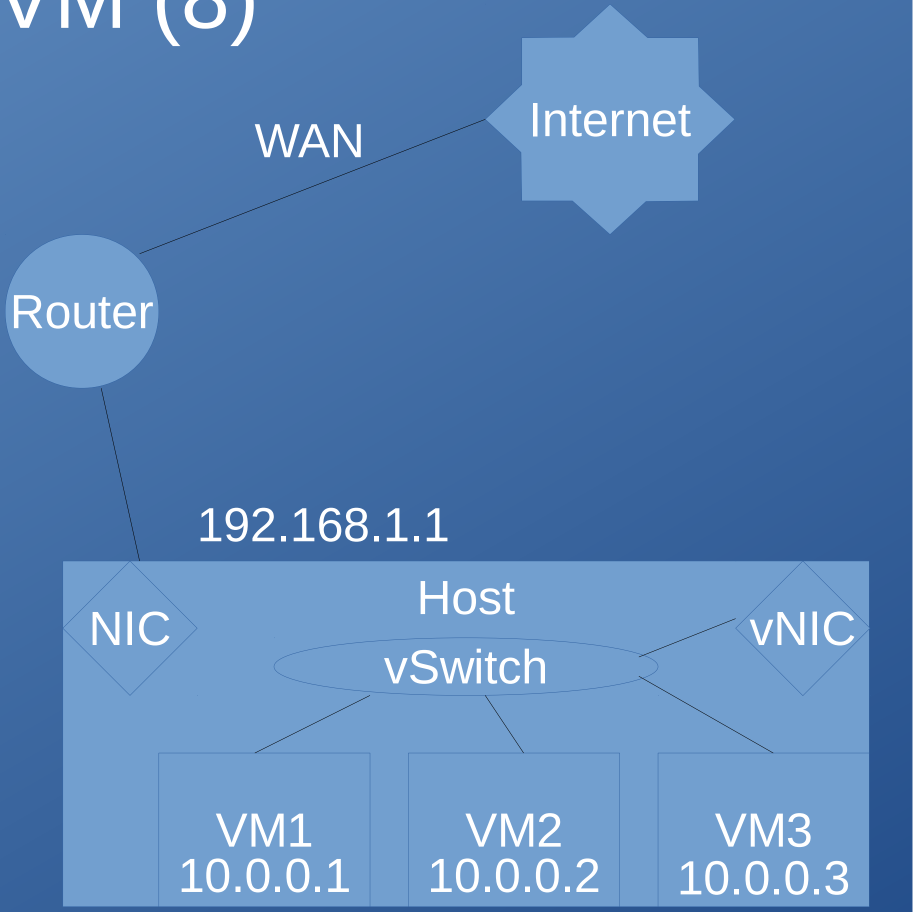
Configuring a VM (7)

- Internal Network: only VMs can talk to each other in their own virtual network. Many virtual switches can be created.
  - You name a virtual switch.
  - VMs can only talk to each other – no internet access.



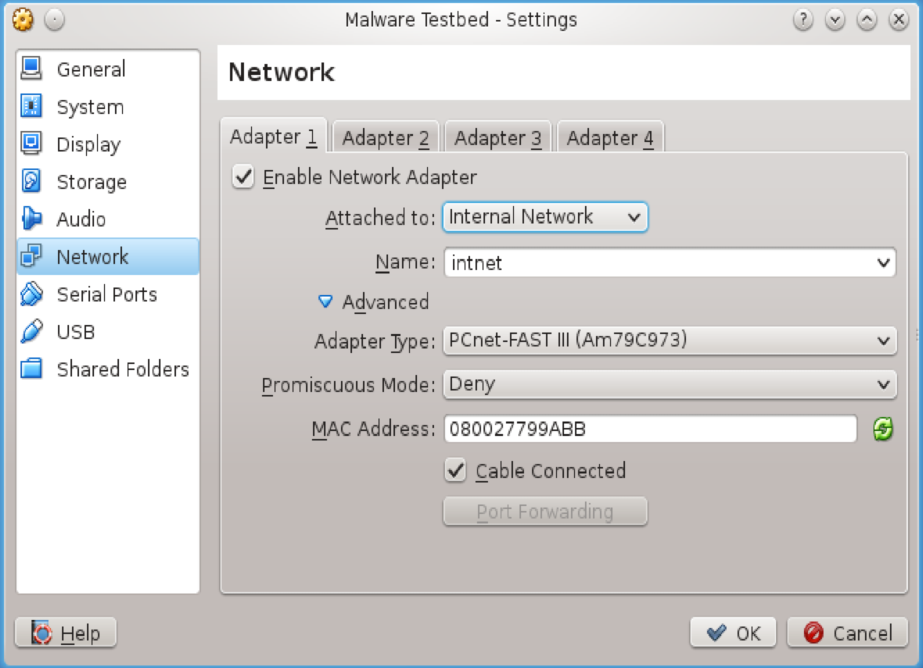
Configuring a VM (8)

- Host-only Adapter: Like “Internal Network,” but allows communication to the host OS, as well.
  - A virtual NIC is created on the host OS (can verify with ifconfig or Windows control panel)
  - vNIC has no internet access.
  - Host is on the vSwitch via vNIC and can talk to the VMS.



Configuring a VM (9)

- If using Internal Network, choose the vSwitch by typing the name. VMs with the same “Name” in this field are automatically on the same switch.
- MAC address can be chosen here if needed.
- Up to 4 adapters can be configured on 1 VM.
- Multiple adapters can be used to combine network schemes.
  - Have many VMs in an Internal Network, but 1 also has a Bridged Adapter: use 1 VM as a router to give Internal VMs internet access.



Malware Testbed - Settings

Network

Adapter 1 | Adapter 2 | Adapter 3 | Adapter 4

☑ Enable Network Adapter

Attached to: Internal Network

Name: intnet

Advanced

Adapter Type: PCnet-FAST III (Am79C973)

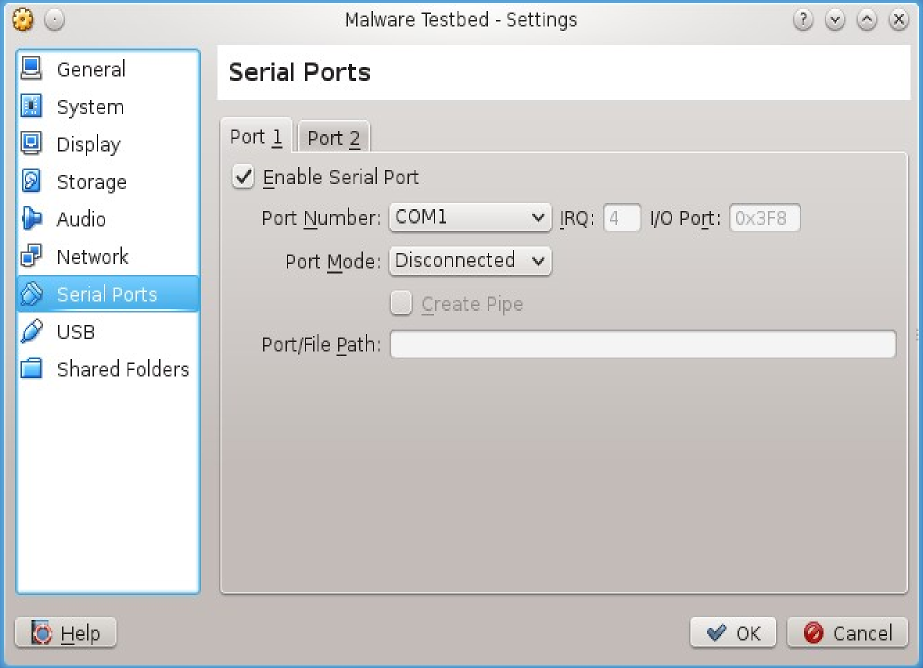
Promiscuous Mode: Deny

MAC Address: 080027799ABB

☑ Cable Connected

Configuring a VM (10)

- Serial ports can be used for kernel debugging.
- Useful for analyzing rootkits or driver issues.
- Also useful for other serial devices.



Malware Testbed - Settings

Serial Ports

☑ Enable Serial Port

Port Number: COM1 IRQ: 4 I/O Port: 0x3F8

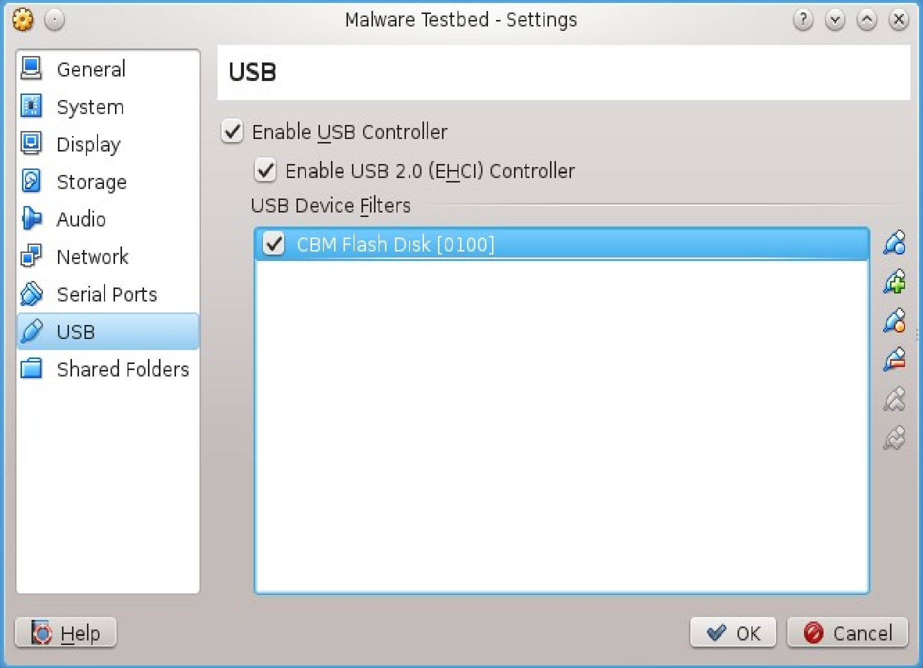
Port Mode: Disconnected

□ Create Pipe

Port/File Path:

Configuring a VM (11)

- USB devices can be passed through to VMs.
- “Expansion Pack” required for USB 2.0 (download available at virtualbox.org.)
  - Otherwise stuck at USB 1.1
- Set a filter so that the device is sent straight to the VM when it is plugged in.



Malware Testbed - Settings

USB

☑ Enable USB Controller

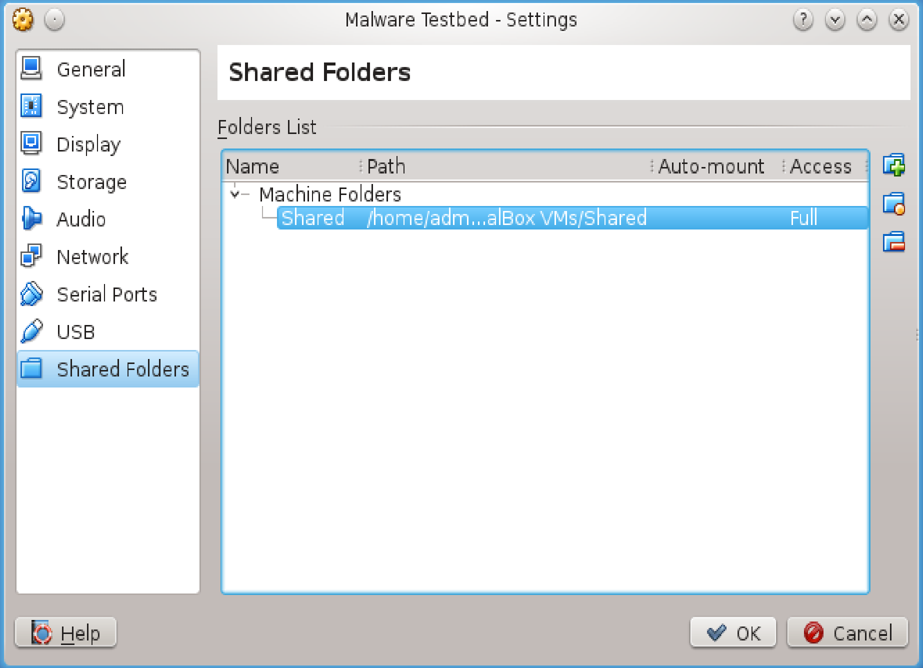
☑ Enable USB 2.0 (EHCI) Controller

USB Device Filters

☑ CBM Flash Disk [0100]

Configuring a VM (12)

- Shared folders: useful for sharing data between host, VM, and other VMs.
- Select a folder on the host to share.
- Requires Guest Additions.
- Accessed via "network."
  - Windows: map network drive.
  - Linux: *sudo mount -t vboxsf <folder name> <mount point>*
- Caution: if given Full access, guest can delete files.



Malware Testbed - Settings

Shared Folders

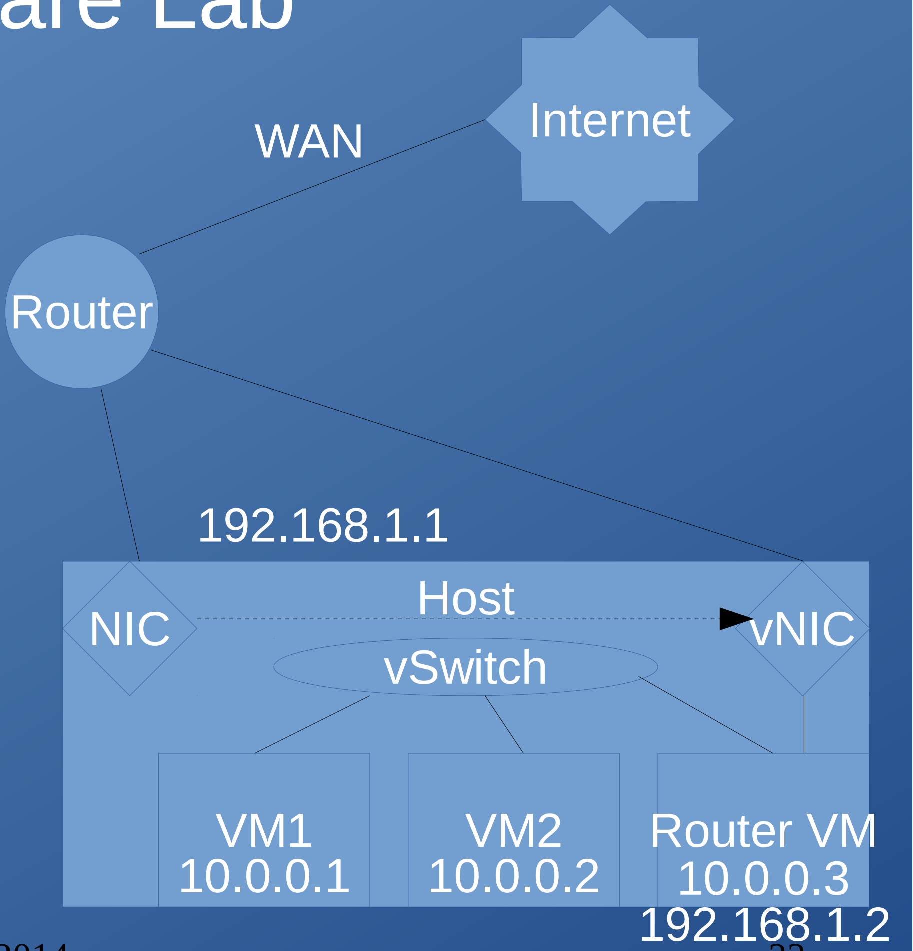
Folders List

Name	Path	Auto-mount	Access
Machine Folders			
Shared	/home/adm...alBox VMs/Shared		Full

Demos...

Setup for Malware Lab

- Linux VM set up as a router.
  - Bridged Adapter: talk to outside world.
  - Internal Network: talk to VMs in isolated network, provide.
  - Provides network access for internal VMs.
  - Sniffs/proxies traffic, applies firewall rules.
- Internal Network VMs isolated from the outside – traffic moderated by Linux VM router.



Internet
WAN
Router
192.168.1.1
Host
NIC
vNIC
vSwitch
VM1
10.0.0.1
VM2
10.0.0.2
Router VM
10.0.0.3
192.168.1.2