Hands-on Lab Description

CS-SYS-00101 –
Virtual Machine in VirtualBox

Category:

CS-SYS: Computer System

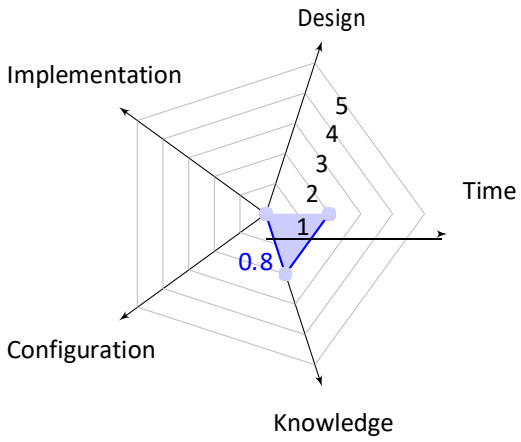
Objectives:

1. Learn how to install VirtualBox
2. Learn how to create virtual machine in VirtualBox
3. Learn how to configure virtual machine in VirtualBox

Estimated Lab Duration:

1. Expert: 20 minutes
2. Novice: 100 minutes

Difficulty Diagram:



Difficulty Table.

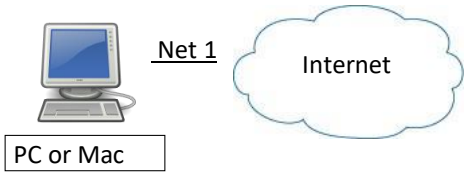
Measurements	Values (0-5)
Time	2
Design	0
Implementation	0
Configuration	0
Knowledge	2
Score (Average)	0.8

Required OS:

Linux: Ubuntu 18.04 LTS

Lab Running Environment:

VirtualBox https://www.virtualbox.org/



1 Computer OS: Windows 10, Mac OS 10 or Linux 64-bit verison
2 Network Setup:
Internet: Accessible

Lab Preparations:

Initial setup: At least 4GB of RAM and 20GB of hard disk space is required for this lab

Task 1 VirtualBox

Oracle VM VirtualBox is a cross-platform virtualization application. What does that mean? For one thing, it installs on your existing Intel or AMD-based computers, whether they are running Windows, Mac OS X, Linux, or Oracle Solaris operating systems (OSes). Secondly, it extends the capabilities of your existing computer so that it can run multiple OSes, inside multiple virtual machines, at the same time. As an example, you can run Windows and Linux on your Mac, run Windows 10 on your Linux server, and so on, all alongside your existing applications. You can install and run as many virtual machines as you like. The only practical limits are disk space and memory.

Oracle VM VirtualBox is deceptively simple yet also very powerful. It can run everywhere from small embedded systems or desktop class machines all the way up to datacenter deployments and even Cloud environments.

When dealing with virtualization, and also for understanding the following chapters of this documentation, it helps to acquaint oneself with a bit of crucial terminology, especially the following terms:

Host operating system (host OS). This is the OS of the physical computer on which Oracle VM VirtualBox was installed. There are versions of Oracle VM VirtualBox for Windows, Mac OS X, Linux, and Oracle Solaris hosts.

This manual discusses all Oracle VM VirtualBox versions together. There may be platform-specific differences which we will point out where appropriate.

Guest operating system (guest OS). This is the OS that is running inside the virtual machine. Theoretically, Oracle VM VirtualBox can run any x86 OS such as DOS, Windows, OS/2, FreeBSD, and OpenBSD. But to achieve near-native performance of the guest code on your machine, we had to go through a lot of optimizations that are specific to certain OSes. So while your favorite OS may run as a guest, we officially support and optimize for a select few, which include the most common OSes.

Virtual machine (VM). This is the special environment that Oracle VM VirtualBox creates for your guest OS while it is running. In other words, you run your guest OS in a VM. Normally, a VM is shown as a window on your computer's desktop. Depending on which of the various frontends of Oracle VM VirtualBox you use, the VM might be shown in full screen mode or remotely on another computer.

Oracle VM VirtualBox comes in many different packages, and installation depends on your host OS. If you have installed software before, installation should be straightforward. On each host platform, Oracle VM VirtualBox uses the installation method that is most common and easy to use.

Go to Downloads – Oracle VM VirtualBox to download the newest version of VirtualBox, which is 6.1.22 as of this manual is being created. Using the link for Windows hosts if your host OS is Windows, and OS X link if your host OS is Mac OS, so on so forth.

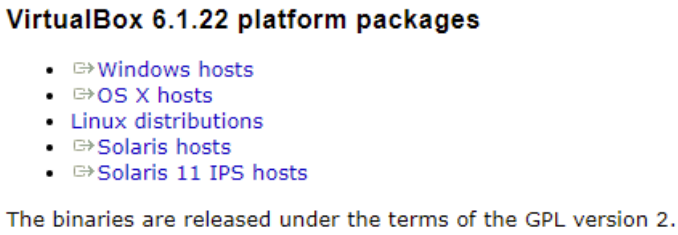


Figure CS-SYS-00101.1 Downloads link of VirtualBox.

Click on the link to download VirtualBox. After the download process is finished, you should get a .exe file for Windows or a .dmg file for Mac OS, double click it to start the installation process.
The installer have a step-by-step interface for you to follow, keep everything by default if possible, and only change settings when your understand what's you're change and what you want.
(If you're using Ubuntu or other type of Linux OS as Host OS, please use apt install or other command line to install Virtualbox.)

Task 2 Install provided VM on VirtualBox

We provide several pre-built Ubuntu 18.04 VirtualBox image, which can be downloaded from Project Overview pages：

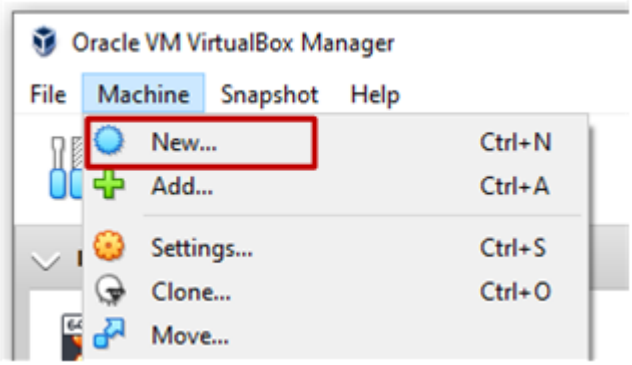
Project 1 image: https://www.coursera.org/learn/cse548-advanced-computer-network-security/supplement/GmMTZ/project-1-packet-filter-firewall-iptables
Project 2 and 3 image: https://www.coursera.org/learn/cse548-advanced-computer-network-security/supplement/W7shJ/course-project-2-sdn-based-stateless-firewall
Project 4 image: https://www.coursera.org/learn/cse548-advanced-computer-network-security/supplement/eyCEM/project-4-machine-learning-based-anomaly-detection-solutions

Account Information of this VM：

User name: ubuntu Password: 123456

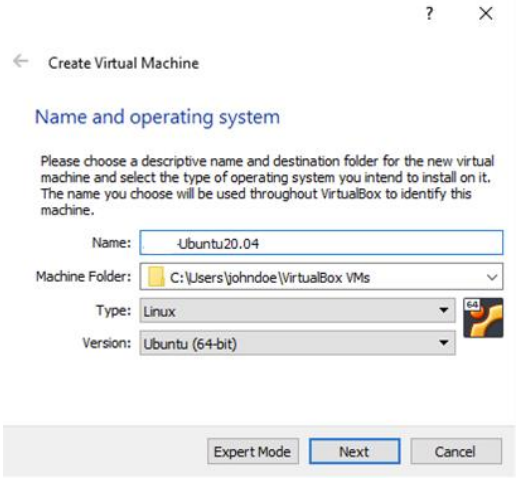
Task 2.1 Create a New VM in VirtualBox

We need to first use New to create a new virtual machine



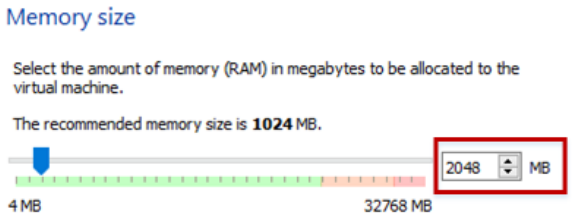
Task 2.2 Provide a Name and Select the OS Type and Version

Our prebuilt Ubuntu 18.04 VM is 64-bit, so pick Ubuntu (64-bit).



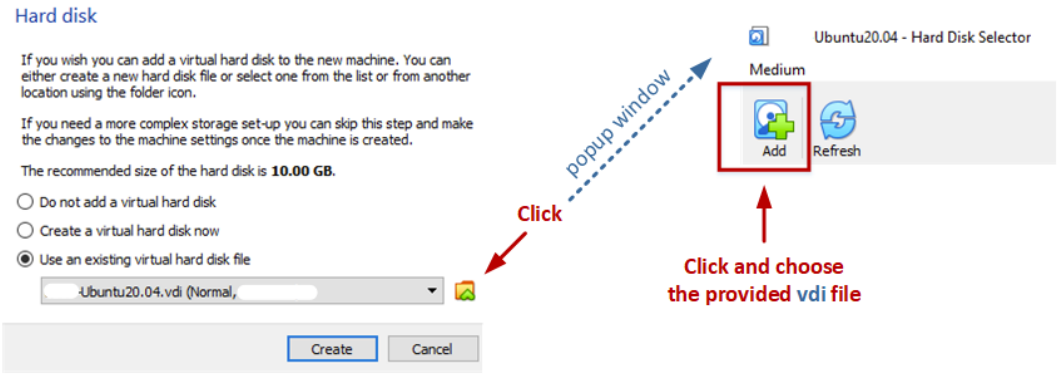
Task 2.3 Set the Memory Size

We need to allocate dedicated memory for the VM. We recommend at least 2GB. If your computer has more RAM, you can increase accordingly. The more memory you give to the VM, the better the performance you will get. But notice that you need keep some memory for your Host OS, so, say if your total memory is 8GB, then you should allocate not more than 6GB memory to the VM.



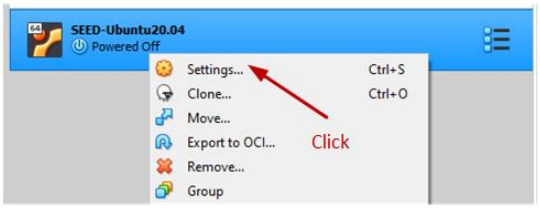
Task 2.4 Select the Pre-built VM File Provided by Us

Pick "Using existing virtual disk file", then Click the folder image. On the popup window, use the Add button to search and select the .vdi file downloaded from the google drive link.



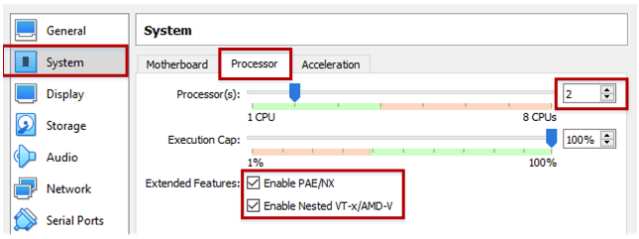
Task 3 Configure the VM

After the previous step, your VM will be created, and you will see it on VirtualBox's VM panel. We need to do some further configuration. Right-click the VM, click the Settings option, and we will see the Settings window.



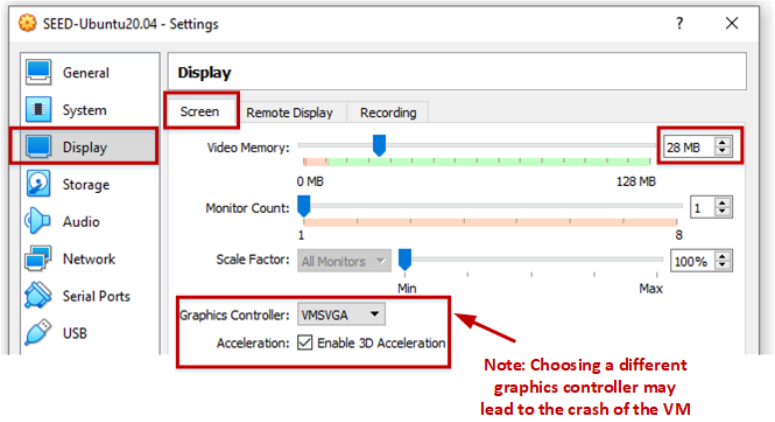
Task 3.1 CPUs

Go to the System category and select the Processor tab. Assign number of CPUs to this VM if you prefer. Although 2 may be sufficient, if the performance seems to be an issue, increase the number. And remember to keep some CPU for you Host OS!



Task 3.2 Display

Go to the Display category, and select the Screen tab. If the display does not seem to work properly, try to increase the amount of video memory. In our testing, 28 MB seems to be sufficient.

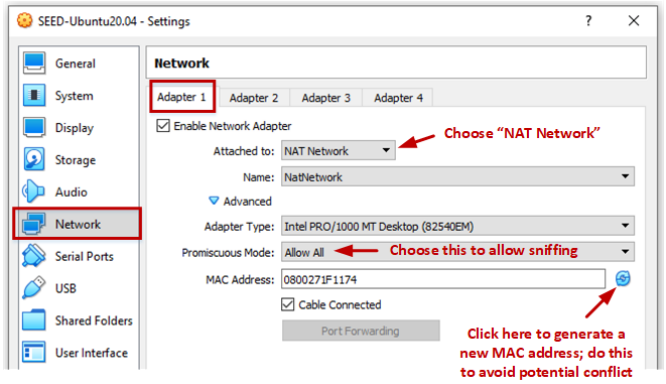


Note 1: Make sure to select VMSVGA, as choosing other graphic controllers may lead to the crash of the VM.

Note 2: If your computer's screen resolution is too high, the VM may not be able to match the high resolution. As results, your VM will be very small on your screen. To make it bigger, adjust the Scale Factor in this setting.

Task 3.3 Network

Go to the Network category, and select the Adapter 1 tab. We will choose the NAT Network adaptor. Click the Advanced drop-down menu to further configure the network adaptor. If you don't see such an adaptor, see the note below.

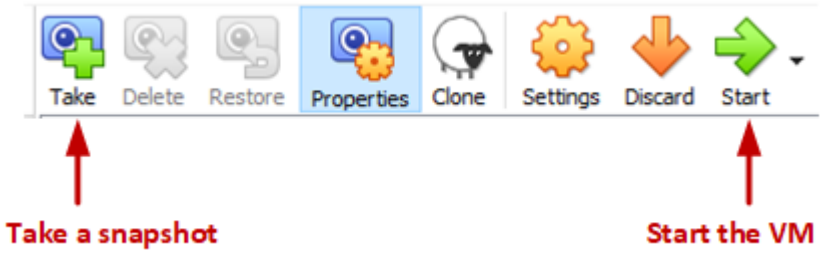


Note: If you don't see the NAT Network adaptor, you need to create one. Go to the File menu, click Preferences.... You will see a popup window. Go to the Network tab, and you can add a new Nat Network adaptor there.

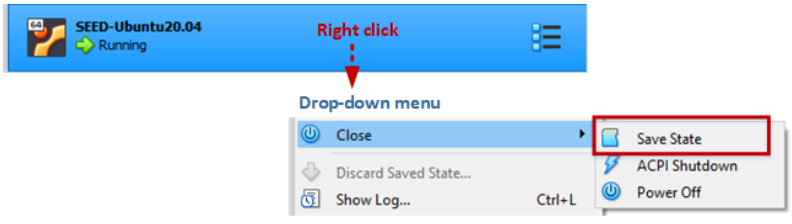
For this course, you'll need to create 2 NAT Network. First, create two NAT Network, the system will name them NatNetwork and NatNetwork1, don't change the configration of NatNetwork. Double click on NatNetwork1, and change "Network CIDR" from 10.0.2.0/24 to 10.0.1.0/24.

Task 4 Start the VM, Take Snapshot and Stop the VM

We can now start the VM. You can also use the Take button to take a snapshot of your VM. This way, if something goes wrong, you can roll back the state of your VM using the saved snapshots.



An running VM will consume a lot of computing resources from your computer so you may want to close it when it is not being used. There are many ways to stop the VM. The best way is to use the Save State. This is different from shutting down the VM. It saves the current VM state, so next time when you restart the VM, the state will be recovered. Moreover, the speed is also faster than booting up a VM.



Task 5 Requirements for Projects

For project 1, you'll need to create two VMs from the project1.img(downloaded from Google Drive with link above), by repeat Task 2 steps for twice. You can name them Client VM and Gateway/Server VM. You should configure Client VM with one network, NatNetwork, and configure Gateway/Server VM with two networks, NatNetwork and NatNetwork1.

For project 2 and 3, you only need to create one VM from the project2.img, and configure it with one network, NatNetwork.

Same for project 4, create one VM from the project4.img, and configure it with one network, NatNetwork.

Related Information and Resource

VirtualBox Download page:
https://www.virtualbox.org/wiki/Downloads
VirtualBox User Manual:
https://www.virtualbox.org/manual/UserManual.html