

# Lab 1

## Lab set-up

### [Virtual Box Set-up]

1. Through the lab class, you will use Ubuntu virtual machine which is operated in Virtual Box. This lab will guide you how to install VirtualBox and Ubuntu VM in order to make everyone start from the same page.
2. **(Install Virtual Box)** To start the installation, from <https://www.virtualbox.org/wiki/Downloads> you have to download the followings:

- a. *VirtualBox 6.1.34 platform packages*
- b. *VirtualBox 6.1.34 Oracle VM VirtualBox Extension Pack*

Particularly, installing *VirtualBox 6.1.34 Oracle VM VirtualBox Extension Pack* is not mandatory, but it enables you to use the advanced and more convenient functions in VirtualBox.

*VirtualBox 6.1.34 platform packages* is a typical installation file. You can install it by just executing it (Double-click the file). After you installed VirtualBox packages, you can install extension pack by double-clicking the downloaded file (Oracle\_VM\_VirtualBox\_Extension\_Pack-6.1.34.vbox-extpack). Then, the VirtualBox (VB) will be automatically started and ask you to confirm whether you want to install this extension pack. Select install.

3. **(Install Ubuntu)** In this lab, you are going to use, Ubuntu 18.04 LTS. You can download this version of Ubuntu (ubuntu-18.04.6-desktop-amd64.iso) from <https://releases.ubuntu.com/18.04/>.

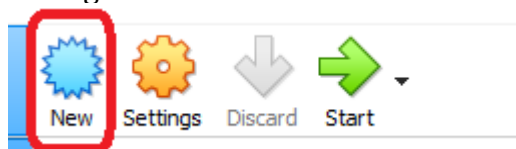
### Desktop image

The desktop image allows you to try Ubuntu without changing your computer at all, and at your option to install it permanently later. This type of image is what most people will want to use. You will need at least 1024MiB of RAM to install from this image.


### 64-bit PC (AMD64) desktop image

Choose this if you have a computer based on the AMD64 or EM64T architecture (e.g., Athlon64, Opteron, EM64T Xeon, Core 2). Choose this if you are at all unsure.

- a. To create a new virtual machine, select the “New” icon in the VB Manager.



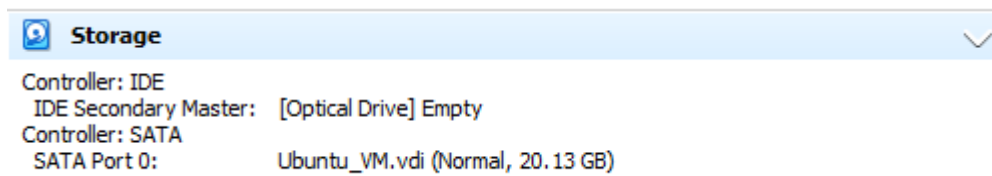
- b. Input the Virtual Machine Name (e.g., Ubuntu\_VM) and select “Linux” and “Ubuntu (64bit)” as Type and Version, respectively as shown below

Name:   
 Machine Folder:   
 Type:    
 Version:

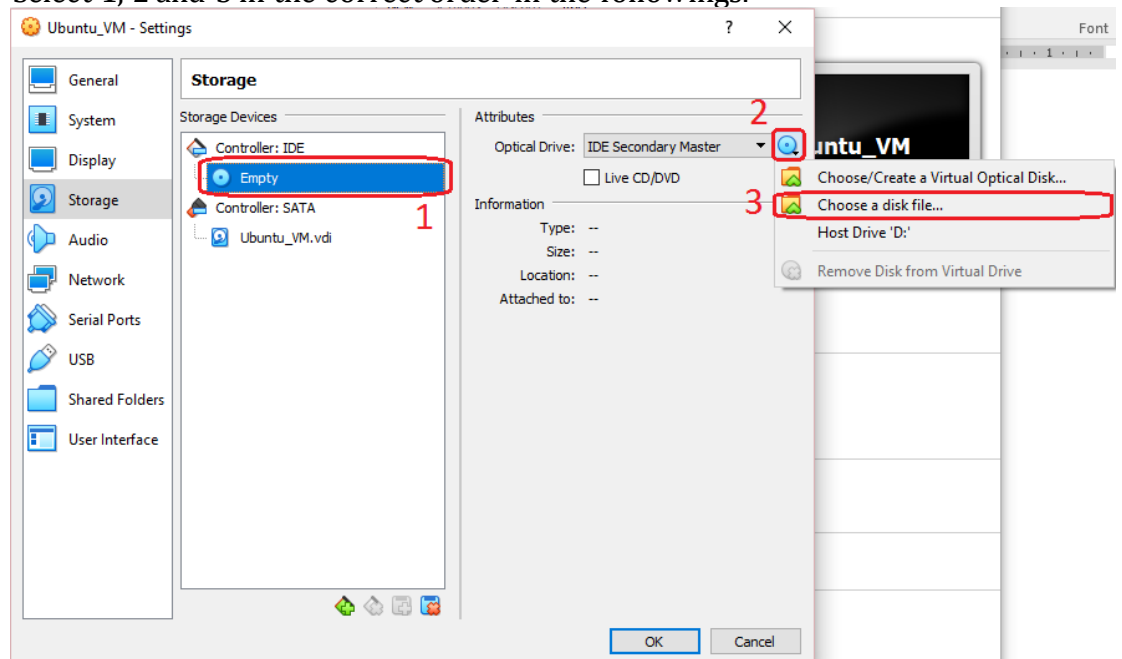
- c. You then,
- 1) allocate base memory (at least 1GB)
  - 2) select Create a virtual hard disk now
  - 3) select VDI (VirtualBox Disk Image) as Hard disk file type
  - 4) select Dynamically allocated as Storage on physical hard disk (this will save your hard disk storage)
  - 5) check you file location and you must allocate at least (20GB) for this subject.
- Finally, you will see that new Ubuntu\_VM virtual machine is created in your VM Manager as follows:



- d. You still have to install Ubuntu. Select Ubuntu\_VM and in the left panel of the VM Manager. There will be Storage property in the right panel as



- e. Click the text "Storage". You can see storage devices in the pop-up window. By default, you can see that your IDE controller is empty. Select 1, 2 and 3 in the correct order in the followings:



If a file selection window is open, go to the directory where the Ubuntu image file are downloaded and select the image file, *ubuntu-18.04.6-desktop-amd64.iso*. Select “OK” button after you load Ubuntu image file on your IDE controller.




- f. Start Ubuntu\_VM by clicking Start button. If you request to select start-up disk, select the Ubuntu image file you just virtually inserted. Follow the steps to set-up Ubuntu as usual.
4. Running VMs on Virtual Box/Configuring your VirtualBox setting
    - The following setting MUST be set while Ubuntu is not operating.
      - Now select “Ubuntu\_VM” (On the list in your main VB window) → Click the right button of Mouse → Select “Settings” → On the pop-up window.
      - Select “Network” in the left menu. In the “Adapter 1” tab, Check “Enable Adapter Network”. → Select “NAT” in the drop-down list for “Attached to”. This will connect your Ubuntu to the network in the lab.
      - Select “General” in the left menu. In the “Advanced” tab → Set “Shared Clipboard” and “Drag’n’Drop” to Bidirectional.
      - The last important step is to install “Guest Additions” to UbuntuVM. In your VM Windows, Select “Devices” → “Insert Guest Additions CD image” and run the program.
      - Reboot Ubuntu to make the network setting change take effect; Please check the network connection by accessing any website after rebooting.

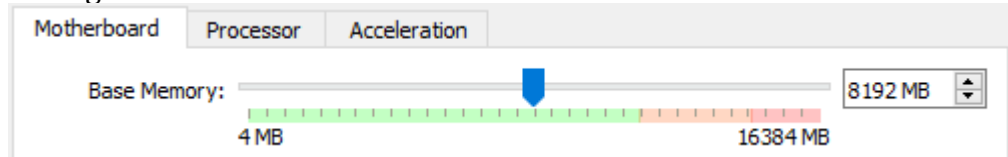
#### [Other Miscellaneous Settings]

5. (NAT Network) If you have more than a single VM machine and want to make them communicate each other. The current network setting does not work. In this case, you have to use “NAT Network” and allow all machines to share the same NAT Network. More specifically,
  - a. Click “File” (on the left corner of the VB Manager window) → Select “Preferences” → Click “Network” on the left panel → Click + icon on the right side of the window; “NatNetwork” will be created → Click OK (Basically NatNetwork is going to use our host machine (your PC) as a router and all the VMs are going to be clients connected to this network.)
  - b. Select “Ubuntu\_VM” (On the list in you main VB window). → Right click → Select “Settings” → On the pop-up window → select “Network”. Now, in the “Adapter 1” tab, check “Enable Adapter Network” if this is not selected. → Select “NAT Network” from the

drop-down list for “Attached to”; NatNetwork will be selected as “Name” → Click OK.

- c. Repeat step the above step (b) for all machines.
6. (Performance setting) Sometimes, you may want to allocate more memory and processors to make your VM faster. You can do this even after you installed your VM.

- a. Select “Ubuntu\_VM” in your VM Manager and select  **System** in the right panel on the VB Manager. There will be three tabs in this setting.



- b. In the “Motherboard” tab, you can adjust the base memory you allocated during the installation.
- c. In the “Processor” tab, you can allocate more or less processor cores in VM and set execution cap (100% means allocated cores works 100% for the VM). If your VM is too slow, please check those settings.
- d. It is not recommended to change the other settings in this lab.
7. (Share Drive) You also can make your Ubuntu\_VM share a folder with your host OS. Please check <https://dzone.com/articles/sharing-folders-between-a-virtualbox-host-and-gues>. Do not forget to execute `$sudo usermod -aG vboxsf <someuserID>`  
You can execute the above command in a terminal program in your Ubuntu\_VM.

## [GPG]

GPG (Gnu Privacy Guard) is widely used as a toolkit to protect the user privacy through encryption/decryption, particularly for PGP protocol. GPG is installed in Ubuntu as a default program. Let us try the following operations.

1. Symmetric Encryption with AES:  
AES is the default symmetric encryption algorithm in GPG. If not specified, GPG will encrypt your message using AES as a default.
  - a. Select any file (test.txt) to encrypt.
  - b. Encrypt the file using AES:  
`$ gpg -c test.txt.`
  - c. Open the encrypted test file (test.txt.gpg) using text editor to check whether it is not readable. Decrypt it using AES:  
`$ gpg -d test.txt.gpg`
  - d. Let's try to encrypt a folder. GPG cannot encrypt a folder. One easy way to encrypt a folder is 1) to compress the folder to a file, 2) encrypt the compressed file, 3) decrypt the compressed file and 4) extract it. More convenient way to do is to use “gpg-zip”. Try  
`$ gpg-zip -c -o file.gpg [DirName]`

Delete the subdirectory (and everything in it) and decrypt the encrypted directory by running `$ gpg-zip -d file.gpg`

## 2. Public Key Encryption/Decryption

- a. GPG provides public key encryption based on RSA. First, generate Public and Private key pair  
`$ gpg --gen-key`  
(Warning: don't forget your passphrase, which is requested while you are generating the key pair.)  
You may not have enough entropy in your random source. In this case, you have to do more tasks such as moving your mouse and open, move and close program windows. Also, execute some programs in command line.  
(Optional) Alternatively, you can quickly boost the entropy on your machine by installing *rng-tools*:  
`$ sudo apt-get install rng-tools`  
(Caution: I do not know whether random values generated by *rng-tools* is secure or not. You should use this trick only for testing purpose.)
- b. To get the list of your keys, you need to execute  
`$ gpg --list-keys`
- c. Encrypt a file with public key encryption  
`$ gpg --encrypt -r <recipient_ID> filename` (You can use `-e` instead of `--encrypt`)  
For testing, use your email address associated with your key as `<recipient_ID>`
- d. Decrypt the file.  
`$ gpg --decrypt test.txt.gpg`
- e. You also can export your public key and import other people's public key.  
`$ gpg --export <uid> > mypubkey.gpg`  
`$ gpg --import pubkey.gpg`  
where `<uid>` implies your e-mail address.
- f. Try to export the public key with `--armor` (or `-a`) option, what is different?  
`$ gpg -a --export uid > mypubkey.gpg.asc`  
`$ gpg -a --import pubkey.gpg.asc`