COMP 4511

Using Virtual Machine & Compiling Linux Kernel

Goals

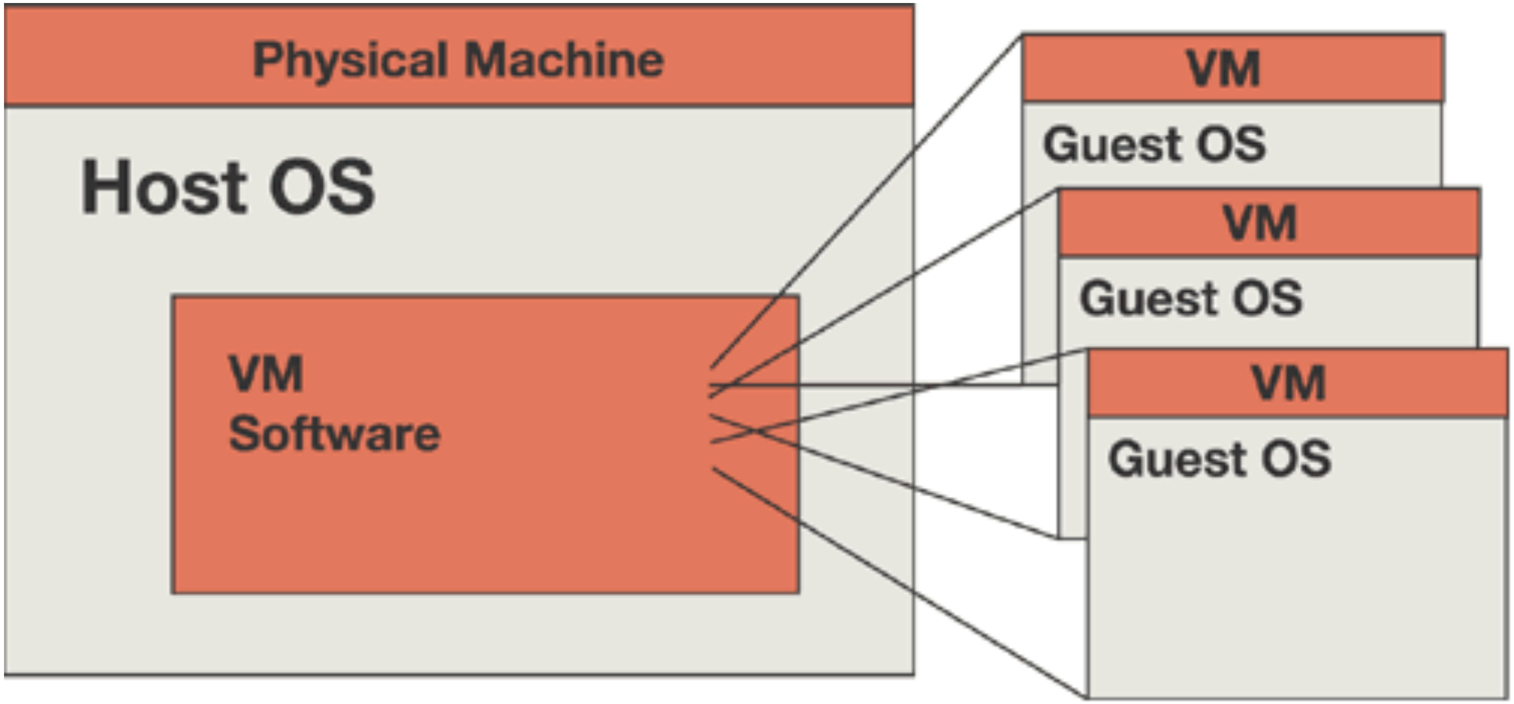
- Create a virtual machine from an ISO

- Connect the guest OS (i.e. Ubuntu VM) with the host OS (i.e. your physical OS)

- Compile a new kernel and replace the existing kernel

Host and Guest Operating Systems

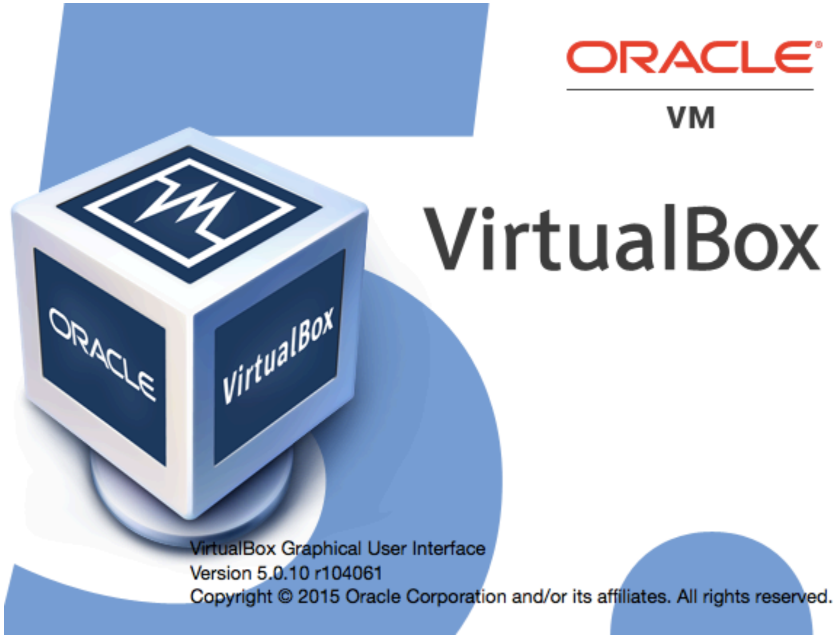
- Relationship among Virtual Machine (VM) software, host operating system, and guest operating system(s)



Reference: http://windowsitpro.com/site-files/windowsitpro.com/files/archive/windowsitpro.com/content/content/42269/figure_01.gif

Prerequisites

- Virtual Machine Software
  - Oracle Virtualbox (https://www.virtualbox.org/)
  - It is cross-platform
- Files
  - Ubuntu ISO (ubuntu-server)
    - If you don't like Ubuntu, you may use another Linux distro
  - Linux Kernel 3.10.94 source files (linux-3.10.94.tar.xz)



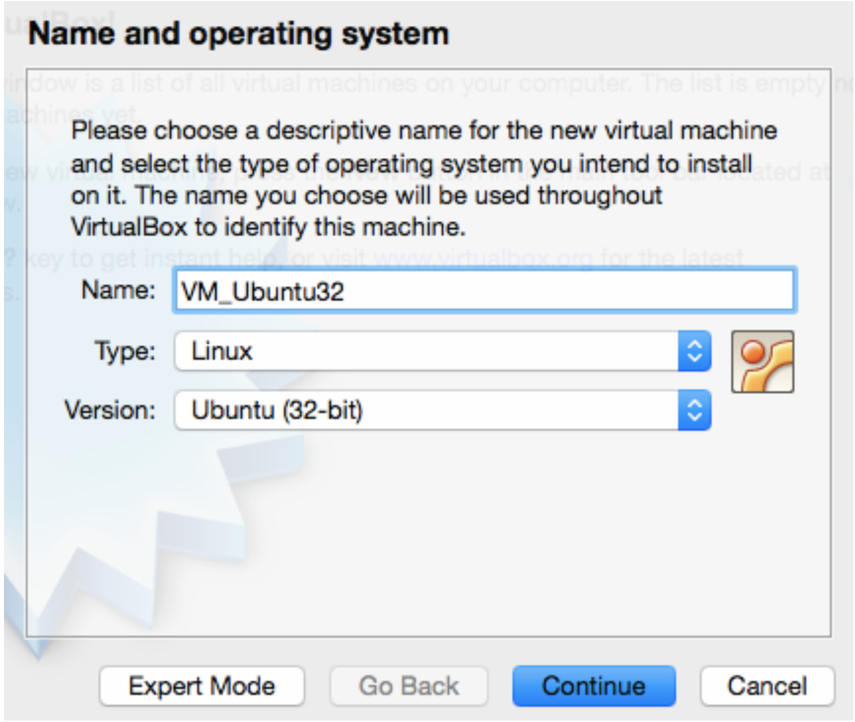
VirtualBox Graphical User Interface
Version 5.0.10 r104061
Copyright © 2015 Oracle Corporation and/or its affiliates. All rights reserved.

Lab task 1

- Create a virtual machine instance from an ISO

Step 1 – Name your Virtual Machine

- Give a name of your VM
- Select the type as "Linux"
- Select the version as "Ubuntu (32-bit)"



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.

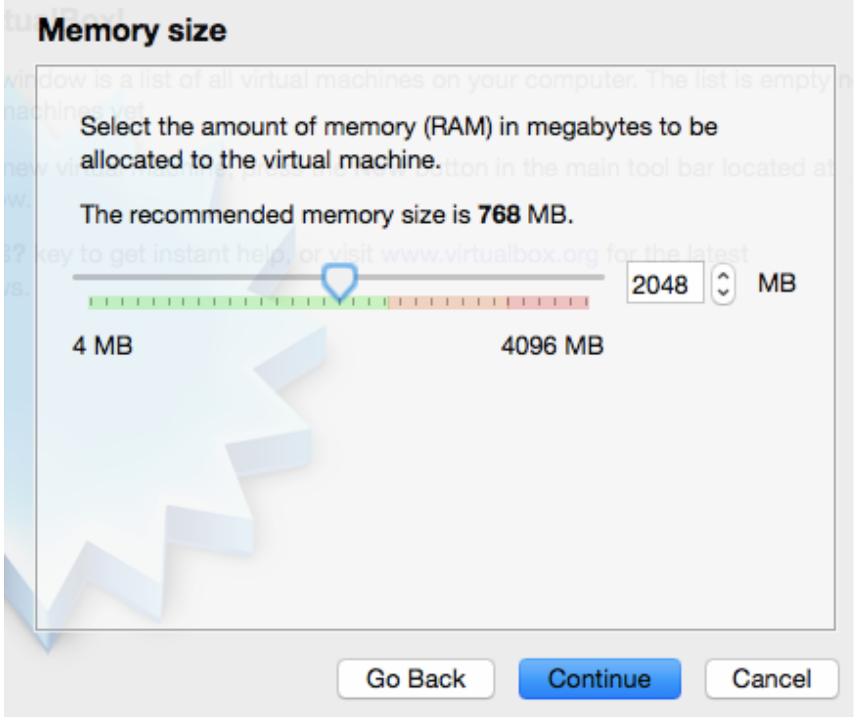
Name: VM_Ubuntu32

Type: Linux

Version: Ubuntu (32-bit)

Step 2 – Configure Memory Size

- The default RAM is less than 1GB
- It is recommended to have 2GB (i.e. 2048MB) RAM for the VM if the hosting platform has 4GB of physical RAM



Memory size

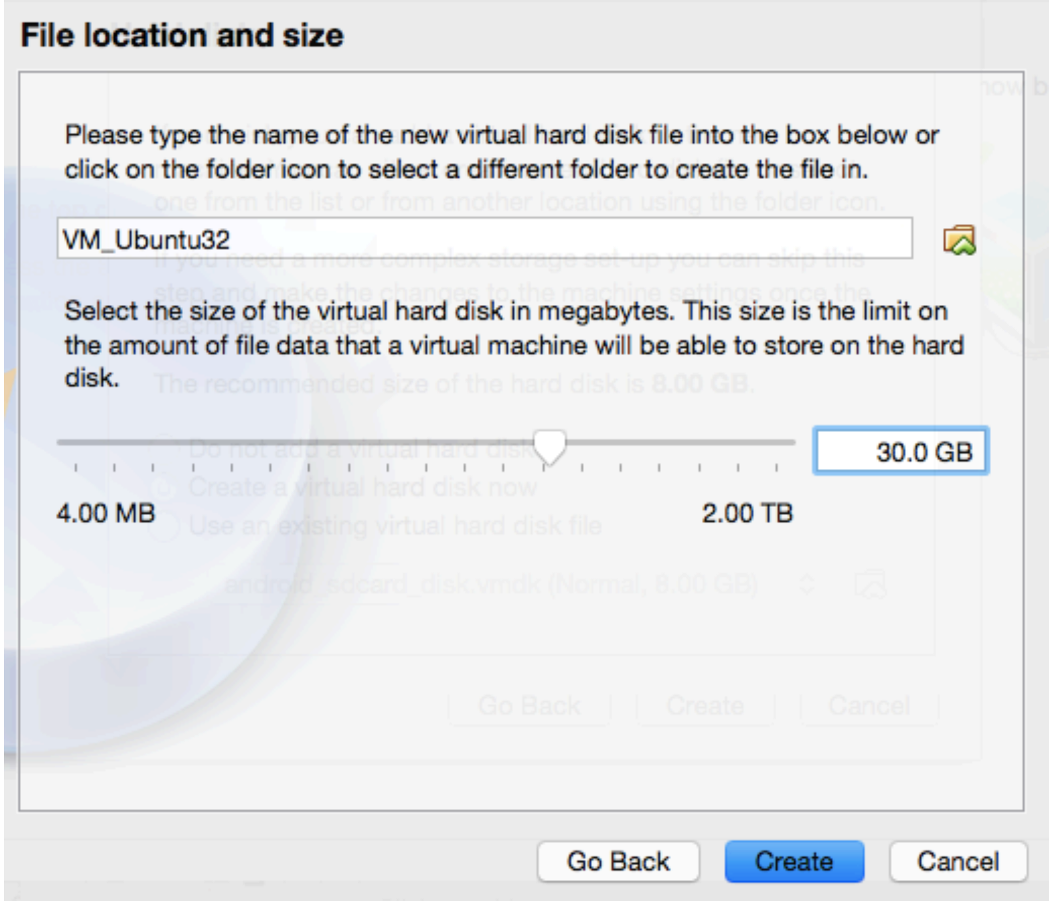
Select the amount of memory (RAM) in megabytes to be allocated to the virtual machine.

The recommended memory size is **768** MB.

2048 MB

Step 3 – Create a Virtual Hard Disk Drive

- The default value is 8GB
- We may need to increase the amount to <u>30GB</u>
- Use the default virtual hard disk format: <u>VDI, dynamically allocated</u>



File location and size

Please type the name of the new virtual hard disk file into the box below or click on the folder icon to select a different folder to create the file in.

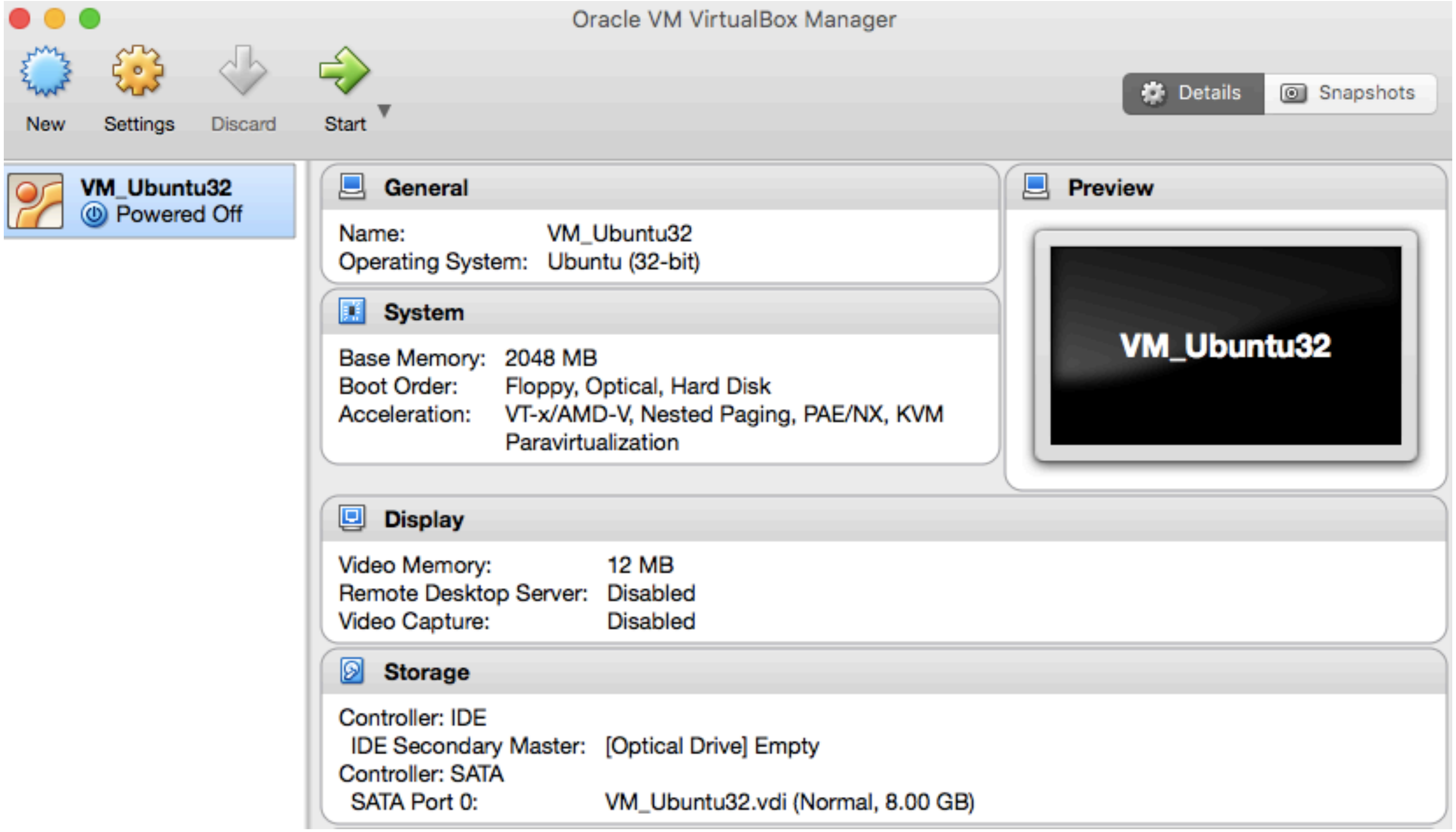
VM_Ubuntu32

Select the size of the virtual hard disk in megabytes. This size is the limit on the amount of file data that a virtual machine will be able to store on the hard disk.

30.0 GB

Step 4 – An overview of VM Hardware

- Double-check the virtual hardware settings
- Click the "Start Button" to start preparing a guest operating system



General

Name: VM_Ubuntu32
Operating System: Ubuntu (32-bit)

System

Base Memory: 2048 MB
Boot Order: Floppy, Optical, Hard Disk
Acceleration: VT-x/AMD-V, Nested Paging, PAE/NX, KVM Paravirtualization

Display

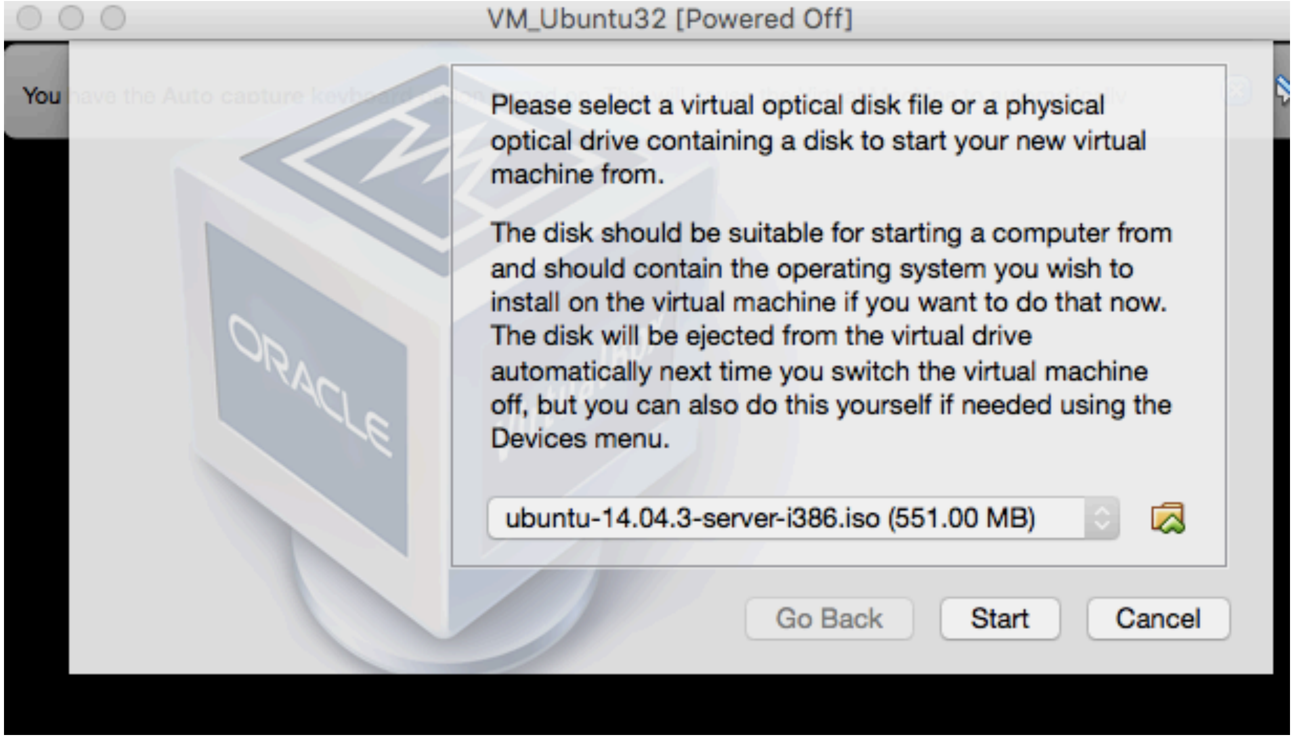
Video Memory: 12 MB
Remote Desktop Server: Disabled
Video Capture: Disabled

Storage

Controller: IDE
IDE Secondary Master: [Optical Drive] Empty
Controller: SATA
SATA Port 0: VM_Ubuntu32.vdi (Normal, 8.00 GB)

Step 5 – Preparing a Guest OS

- In the previous steps, we configured the virtual hardware
- Next, we need to install a guest operating system (Ubuntu) to the VM hardware
- The guest OS consists of a default kernel together with a number of software application
- Once we boot up, we will re-compile a new kernel from source, and then replace the existing kernel of the guest OS



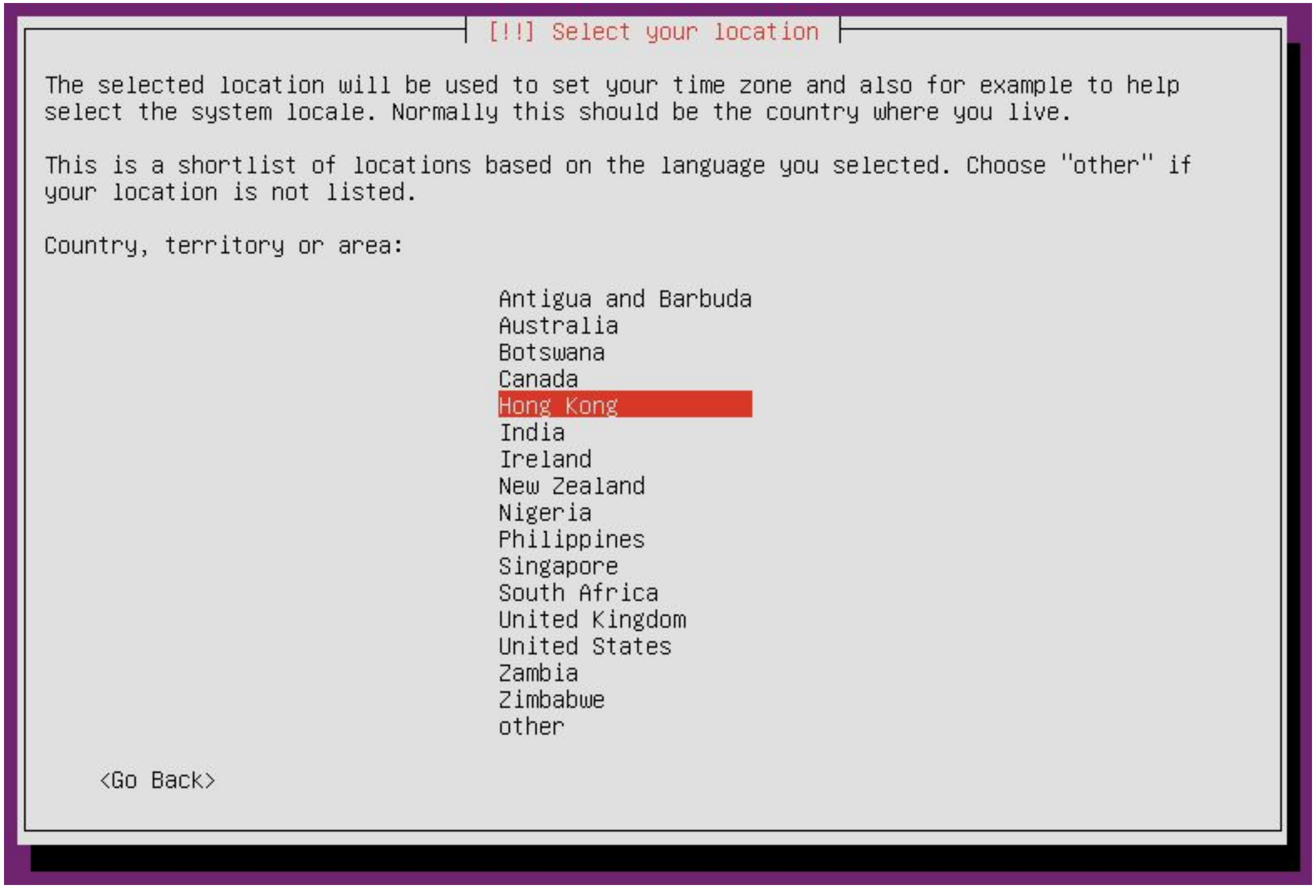
Please select a virtual optical disk file or a physical optical drive containing a disk to start your new virtual machine from.

The disk should be suitable for starting a computer from and should contain the operating system you wish to install on the virtual machine if you want to do that now. The disk will be ejected from the virtual drive automatically next time you switch the virtual machine off, but you can also do this yourself if needed using the Devices menu.

ubuntu-14.04.3-server-i386.iso (551.00 MB)

Step 6 – Select your location

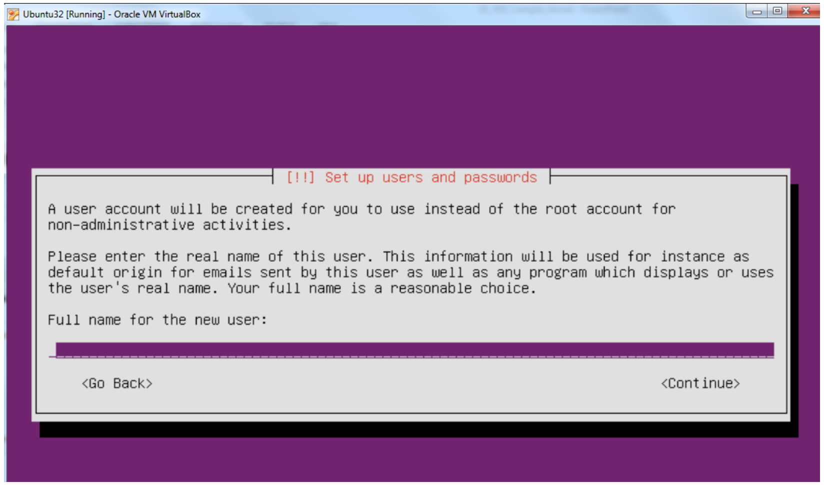
- Accept the default settings until the page of selecting your country
- In this page, please specify "Hong Kong"



```
[!!] Select your location

The selected location will be used to set your time zone and also for example to help
select the system locale. Normally this should be the country where you live.

This is a shortlist of locations based on the language you selected. Choose "other" if
your location is not listed.

Country, territory or area:

                                  Antigua and Barbuda
                                  Australia
                                  Botswana
                                  Canada
                                  Hong Kong
                                  India
                                  Ireland
                                  New Zealand
                                  Nigeria
                                  Philippines
                                  Singapore
                                  South Africa
                                  United Kingdom
                                  United States
                                  Zambia
                                  Zimbabwe
                                  other

    <Go Back>
```

Step 7 – Setup the default administrative account

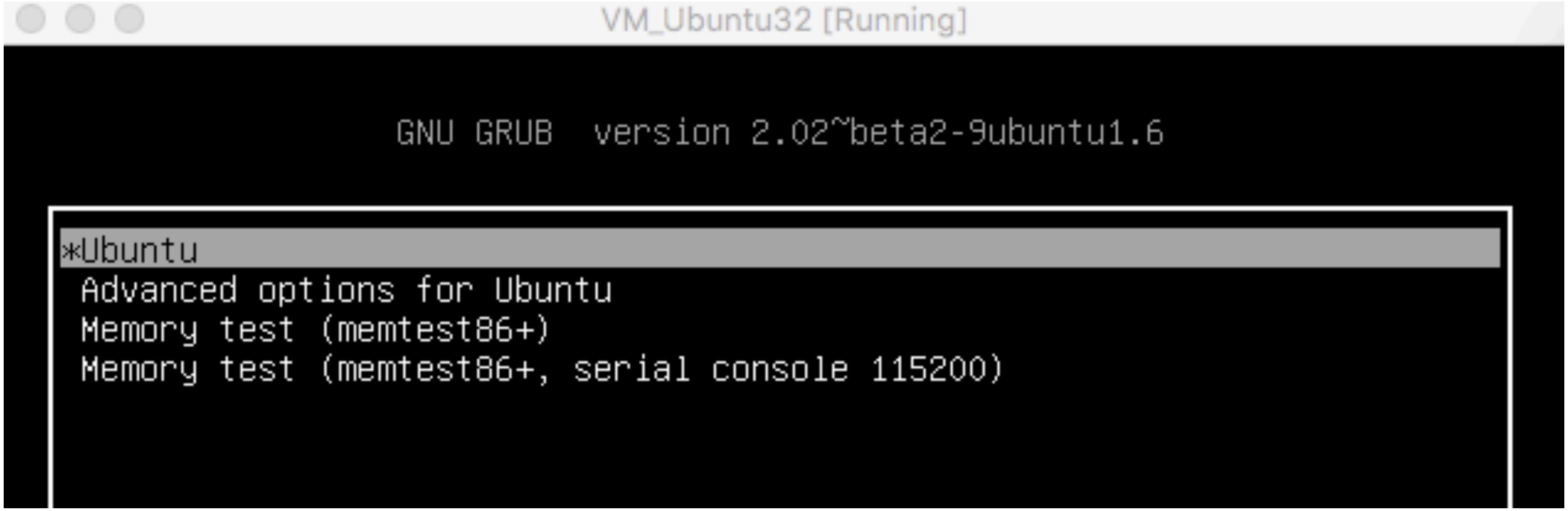
- Accept the default settings until the page to set up users and passwords
- Create the default administrative account by inputting your own username and password
- Note:
  - By default, the **root account password** is locked in Ubuntu
  - We need to create an account with administrative rights, and later on using **sudo** to do administrative tasks
  - We will re-visit how to use "sudo" command later



```
[!!] Set up users and passwords

A user account will be created for you to use instead of the root account for
non-administrative activities.

Please enter the real name of this user. This information will be used for instance as
default origin for emails sent by this user as well as any program which displays or uses
the user's real name. Your full name is a reasonable choice.

Full name for the new user:

_______________________________________________________________________________

    <Go Back>                                                           <Continue>
```

Step 8 – Boot up the Guest OS

- Accept the default settings until the guest OS is installed
- When the system is re-booted, the GNU GRUB screen will be shown
- GNU GRUB is the default boot loader of Ubuntu Linux distro



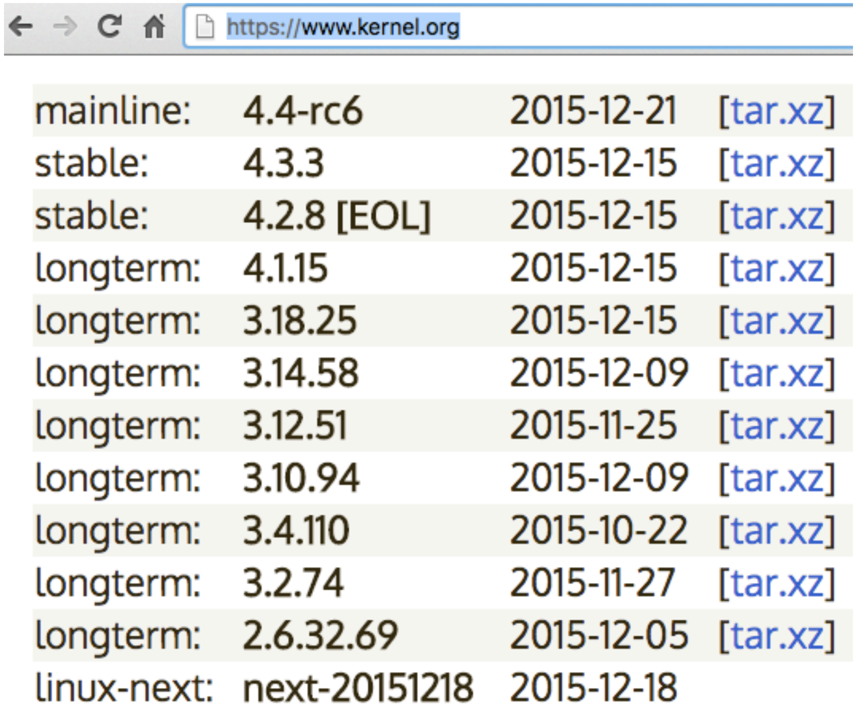
```
GNU GRUB  version 2.02~beta2-9ubuntu1.6

*Ubuntu
 Advanced options for Ubuntu
 Memory test (memtest86+)
 Memory test (memtest86+, serial console 115200)
```

Step 9 – Login to the Guest OS

- Enter the username and password to login to the guest OS
- Check the default Linux Kernel
- Command: uname –r
  - The default Linux Kernel of the guest OS will be shown (i.e. Kernel version 3.19.0-25-generic)

Note: Ubuntu LTS ≠ Linux Kernel LTS

- LTS = Long-term Support
- The Linux Kernel development is managed by Linux Foundation, while Ubuntu development is managed by Canonical Ltd.
- Ubuntu 14.04 is LTS Linux distro, but it ships with a non-LTS Kernel (3.19.0)
- To download and check information related to Linux kernel, visit https://www.kernel.org/

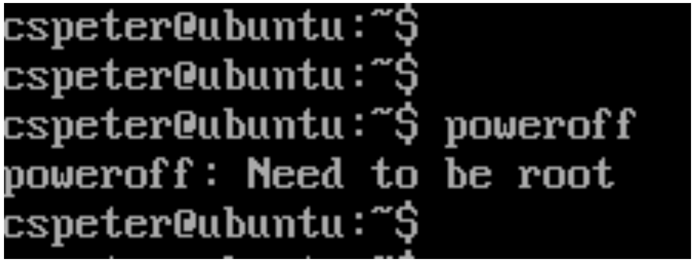


https://www.kernel.org

mainline:	4.4-rc6	2015-12-21	[tar.xz]
stable:	4.3.3	2015-12-15	[tar.xz]
stable:	4.2.8 [EOL]	2015-12-15	[tar.xz]
longterm:	4.1.15	2015-12-15	[tar.xz]
longterm:	3.18.25	2015-12-15	[tar.xz]
longterm:	3.14.58	2015-12-09	[tar.xz]
longterm:	3.12.51	2015-11-25	[tar.xz]
longterm:	3.10.94	2015-12-09	[tar.xz]
longterm:	3.4.110	2015-10-22	[tar.xz]
longterm:	3.2.74	2015-11-27	[tar.xz]
longterm:	2.6.32.69	2015-12-05	[tar.xz]
linux-next:	next-20151218	2015-12-18	

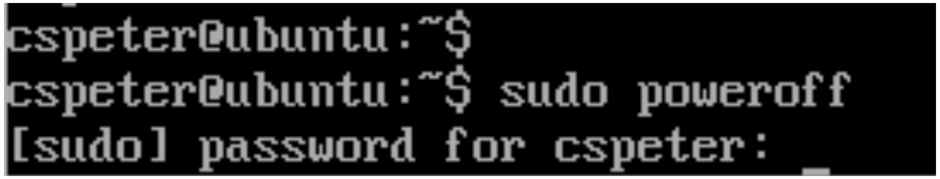
Step 10 – Command to shutdown the guest OS & introduction of "sudo" command

- In this course, we are going to use a guest OS without any graphical user interface
- Command to shutdown
  - poweroff
  - By default, it is not allowed



```
cspeter@ubuntu:~$
cspeter@ubuntu:~$
cspeter@ubuntu:~$ poweroff
poweroff: Need to be root
cspeter@ubuntu:~$
```

- To carry out administrative tasks:
  - Use "sudo" and type in the password everytime (safer) **OR**
  - Enable the root password
    - □ "sudo passwd root" and
    - □ Change to root by "su –"
- Example:



```
cspeter@ubuntu:~$
cspeter@ubuntu:~$ sudo poweroff
[sudo] password for cspeter: _
```

Lab task 2

- Connect your guest OS (i.e. Ubuntu VM) and your host OS (i.e. your physical OS)

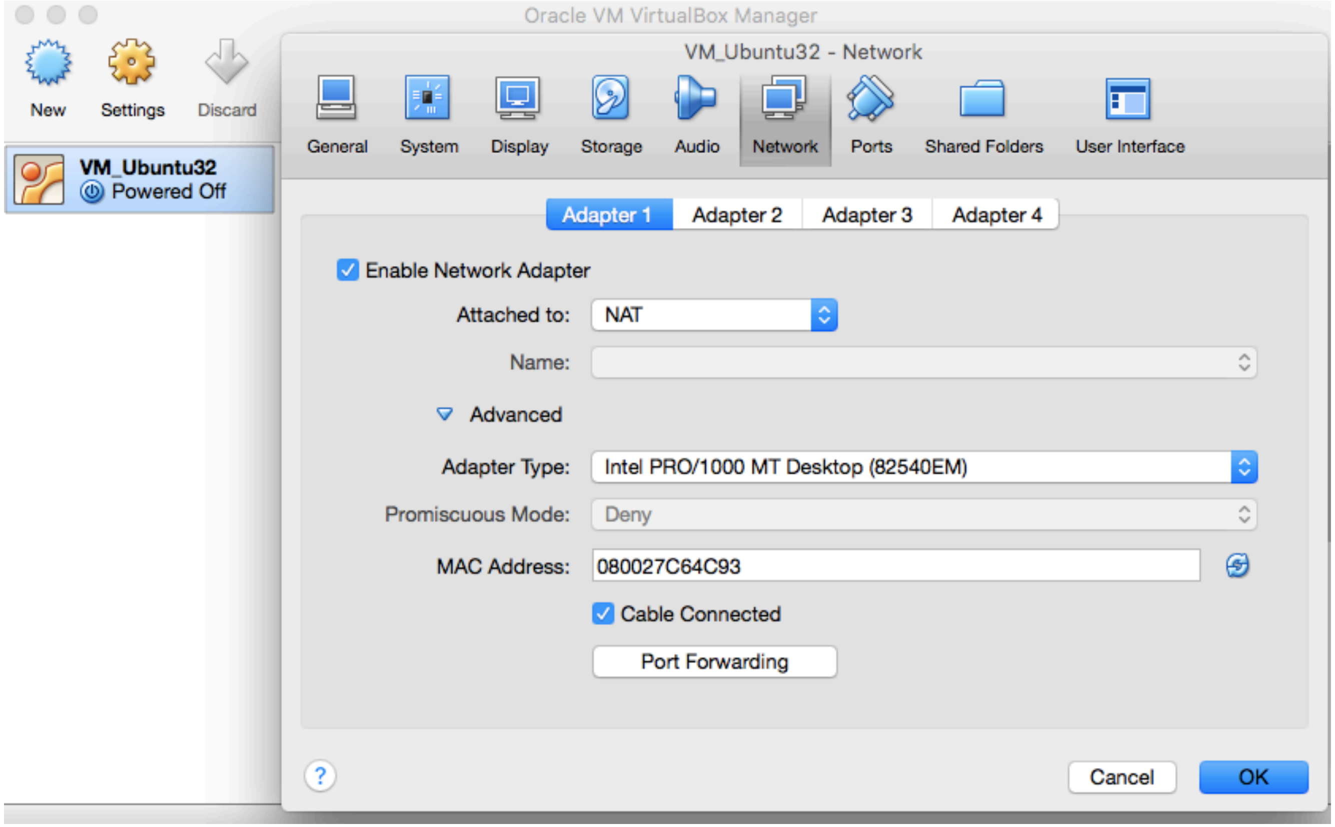
Reference: http://stackoverflow.com/questions/5906441/how-to-ssh-to-a-virtualbox-guest-externally-through-a-host

Connecting the Guest OS and the Host OS

- We have 2 operating systems
  - The Guest OS (Ubuntu Linux VM server)
  - The Host OS (Windows/Mac OS X/Linux)
- We may need to transfer files between them
  - For example, the guest OS don't have graphical user interface. We may need to download the Linux kernel source code zip file from the host OS and transfer it to the guest OS
- Solution: Port forwarding
  - We are going to setup port forwarding in Oracle Virtualbox

Step 1 – Configure the Network Settings

- Select your VM instance (i.e.VM_Ubuntu32)
- Click "Settings" button, a dialog will be prompted
- Click "Network" button, select "Adapter 1"
- Click "Port Forwarding" button at the bottom



Oracle VM VirtualBox Manager

VM_Ubuntu32 - Network

Adapter 1 | Adapter 2 | Adapter 3 | Adapter 4

☑ Enable Network Adapter

Attached to: NAT

Name:

Advanced

Adapter Type: Intel PRO/1000 MT Desktop (82540EM)

Promiscuous Mode: Deny

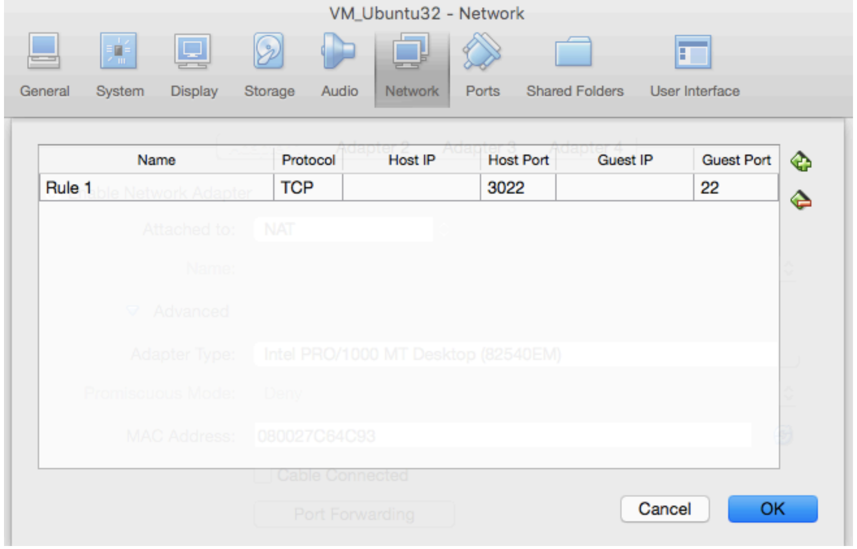
MAC Address: 080027C64C93

☑ Cable Connected

Port Forwarding

Step 2 – Configure Port Forwarding

- Port forwarding setup
  - SSH (Secure Shell) runs on TCP and the default port is 22
  - For file transfer, SFTP is used. SFTP is a subsystem of SSH, so it also runs on TCP port 22
  - For the host port, you are free to use any reasonable port above 3000 (e.g. 3022)
  - Leave the host IP and guest IP as blank (i.e. Any IP)
  - Remember to click "Ok" at the end
- Meaning of this port forwarding rule
  - In your host computer (i.e. the computer running Virtualbox), connecting to localhost with port 3022 is equivalent to connecting the VM port 22



Name	Protocol	Host IP	Host Port	Guest IP	Guest Port
Rule 1	TCP		3022		22

Attached to: NAT

Advanced

Adapter Type: Intel PRO/1000 MT Desktop (82540EM)

Promiscuous Mode: Deny

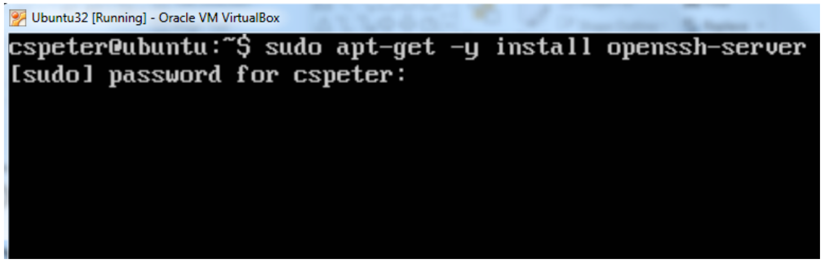
MAC Address: 080027C64C93

Cable Connected

Port Forwarding

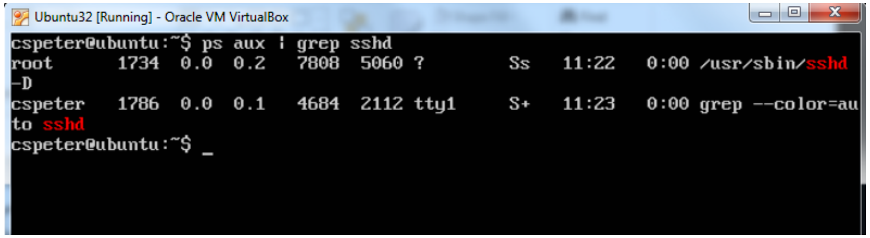
Step 3 – Install Open SSH Server on the Guest OS (i.e. Ubuntu VM)

- In Ubuntu, **apt-get** will be used to install/remove software packages
  - yum will be used for other Redhat-based Linux distributions
  - You can treat it as a Windows update / Mac App Store without GUI
- As software installation is an administrative task, remember to use "sudo" command
- The exact command to install Open SSH Server is:
  - sudo apt-get –y install openssh-server
  - -y means saying "yes" for all options
  - You can also check whether the SSH daemon is running or not



Ubuntu32 [Running] - Oracle VM VirtualBox

```
cspeter@ubuntu:~$ sudo apt-get -y install openssh-server
[sudo] password for cspeter:
```

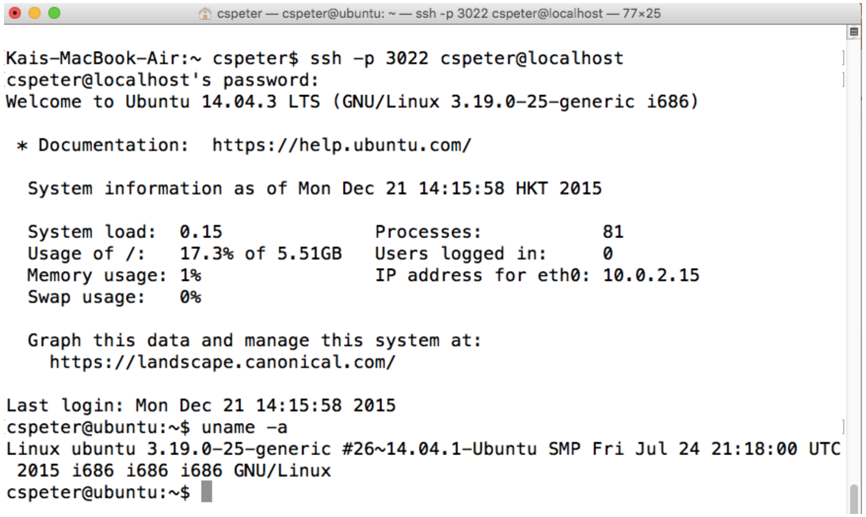


Ubuntu32 [Running] - Oracle VM VirtualBox

```
cspeter@ubuntu:~$ ps aux | grep sshd
root      1734  0.0  0.2   7808  5060 ?        Ss   11:22   0:00 /usr/sbin/sshd
-D
cspeter   1786  0.0  0.1   4684  2112 tty1     S+   11:23   0:00 grep --color=au
to sshd
cspeter@ubuntu:~$ _
```

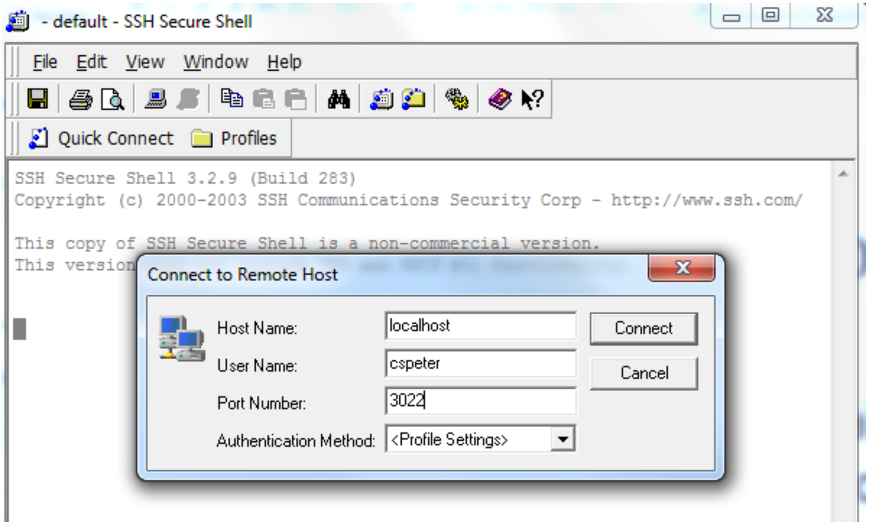
Step 4 – Connect your guest OS via SSH

- Open a terminal / SSH Secure Shell client at your host OS
- Connecting to the guest OS by specifying
  - the client port as “3022” and
  - the server computer as “[your user name]@localhost


cspeter — cspeter@ubuntu: ~ — ssh -p 3022 cspeter@localhost — 77×25

```
Kais-MacBook-Air:~ cspeter$ ssh -p 3022 cspeter@localhost
cspeter@localhost's password:
Welcome to Ubuntu 14.04.3 LTS (GNU/Linux 3.19.0-25-generic i686)

 * Documentation:  https://help.ubuntu.com/

  System information as of Mon Dec 21 14:15:58 HKT 2015

  System load:  0.15              Processes:           81
  Usage of /:   17.3% of 5.51GB   Users logged in:     0
  Memory usage: 1%                IP address for eth0: 10.0.2.15
  Swap usage:   0%

  Graph this data and manage this system at:
    https://landscape.canonical.com/

Last login: Mon Dec 21 14:15:58 2015
cspeter@ubuntu:~$ uname -a
Linux ubuntu 3.19.0-25-generic #26~14.04.1-Ubuntu SMP Fri Jul 24 21:18:00 UTC
 2015 i686 i686 i686 GNU/Linux
cspeter@ubuntu:~$ 
```

Example: Connecting via Terminal (Mac OS X)



```
SSH Secure Shell 3.2.9 (Build 283)
Copyright (c) 2000-2003 SSH Communications Security Corp - http://www.ssh.com/

This copy of SSH Secure Shell is a non-commercial version.
This version
```

Host Name: localhost

User Name: cspeter

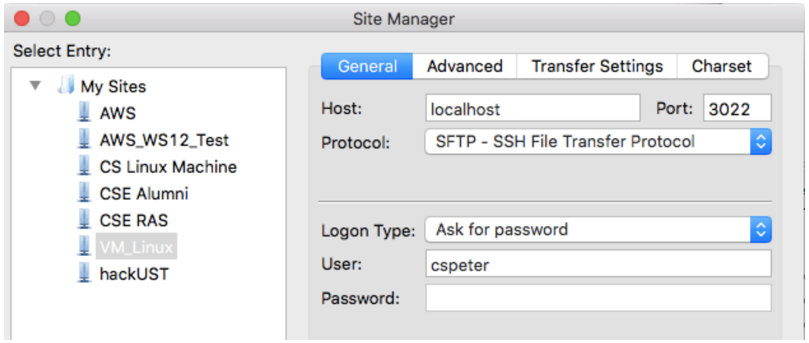
Port Number: 3022

Authentication Method: <Profile Settings>

Example: Connecting via SSH Client (Windows)

Step 5 – Transfer files via SFTP using Filezilla

- It is possible to configure a FTP client and transfer files via SFTP
- Open Site Manager, Create a new site
- Parameters
  - Host: localhost
  - Port: 3022
  - Protocol: SFTP
  - Logon Type: Ask for password
  - User: [Your username]
- Click "Connect"



Site Manager

Select Entry:

- My Sites
  - AWS
  - AWS_WS12_Test
  - CS Linux Machine
  - CSE Alumni
  - CSE RAS
  - VM_Linux
  - hackUST

General | Advanced | Transfer Settings | Charset

Host: localhost Port: 3022

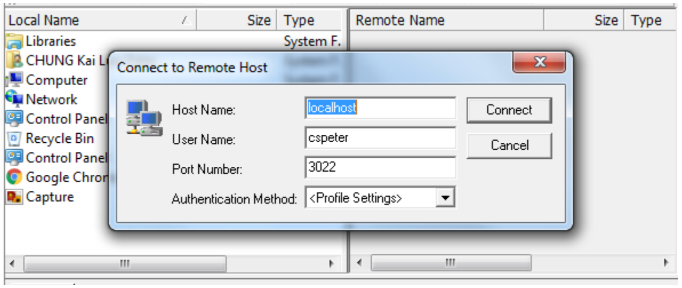
Protocol: SFTP - SSH File Transfer Protocol

Logon Type: Ask for password

User: cspeter

Password:

Example: FileZilla file transfer setup (Mac OS X)



Connect to Remote Host

Host Name: localhost

User Name: cspeter

Port Number: 3022

Authentication Method: <Profile Settings>

Connect

Cancel

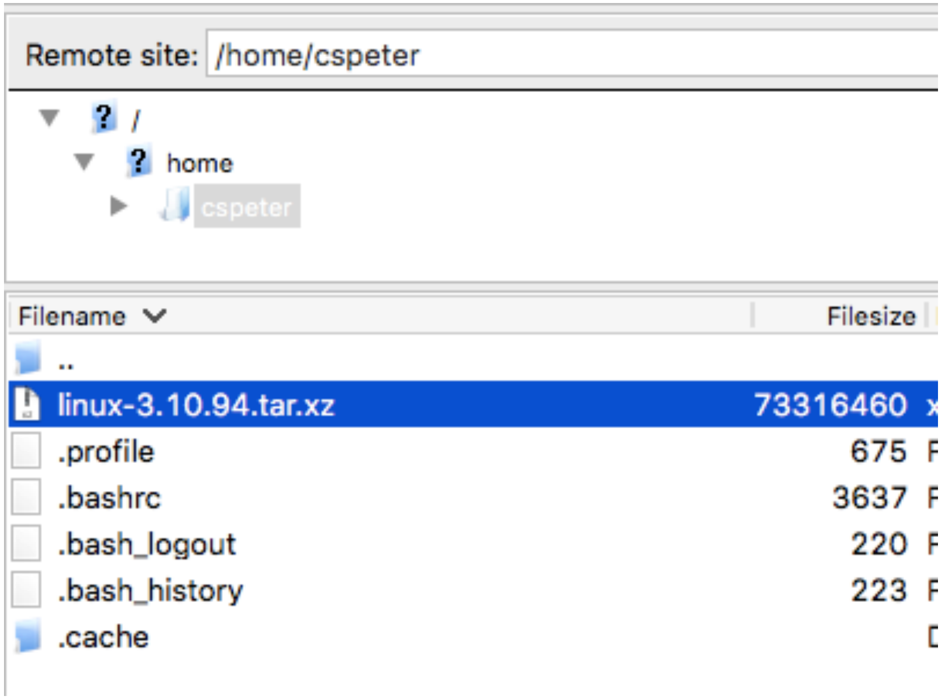
Example: SSH client file transfer setup (Windows)

Lab task 3

- Compile a new Linux kernel and replace the existing kernel

Step 1 – Preparation for Linux Kernel Compilation

- Upload the Linux Kernel source code zip file (linux-3.10.94.tar.xz) to the home directory of Ubuntu VM

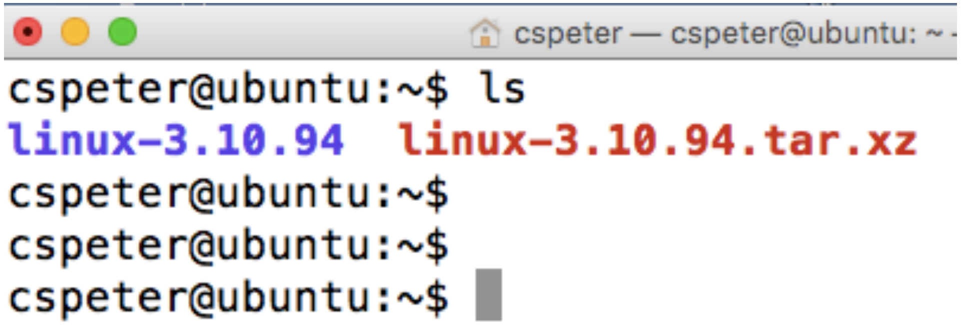


Remote site: /home/cspeter

Filename	Filesize
..	
linux-3.10.94.tar.xz	73316460
.profile	675
.bashrc	3637
.bash_logout	220
.bash_history	223
.cache	

Step 1 – Preparation for Linux Kernel Compilation

- Downloads the package lists from the repositories
  - sudo apt-get update
- Download the missing packages for kernel compilation
  - sudo apt-get install -y gcc make wget libncurses5-dev
- Unzip the source code package using the "tar" command
  - tar xJvf linux-3.10.94.tar.xz
- Expected result:

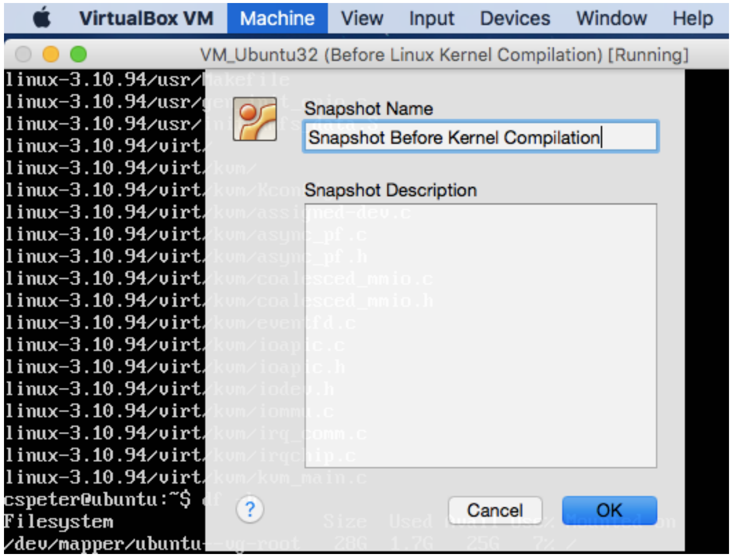


```
cspeter@ubuntu:~$ ls
linux-3.10.94  linux-3.10.94.tar.xz
cspeter@ubuntu:~$
cspeter@ubuntu:~$
cspeter@ubuntu:~$
```

Suggestion: Take a snapshot before Linux Kernel Compilation

- A VMware snapshot is a copy of the virtual machine's disk file at a given point in time
- Snapshots provide a change log for the virtual disk and are used to restore a VM to a particular point in time when a failure or system error occurs
- Let's take a snapshot before Kernel compilation

- Machine > Take Snapshot



VM_Ubuntu32 (Before Linux Kernel Compilation) [Running]

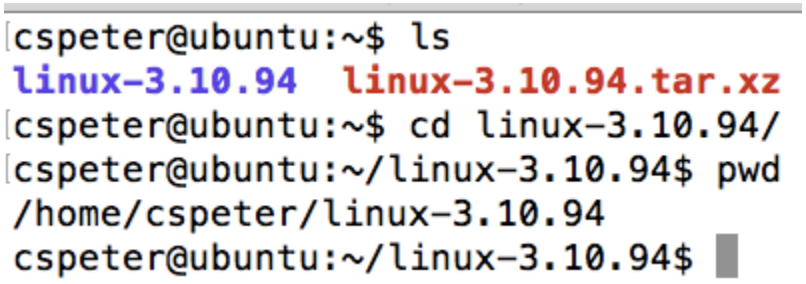
```
linux-3.10.94/usr/
linux-3.10.94/usr/
linux-3.10.94/usr/
linux-3.10.94/virt/
linux-3.10.94/virt/
linux-3.10.94/virt/
linux-3.10.94/virt/
linux-3.10.94/virt/
linux-3.10.94/virt/
linux-3.10.94/virt/
linux-3.10.94/virt/
linux-3.10.94/virt/
linux-3.10.94/virt/
linux-3.10.94/virt/
linux-3.10.94/virt/
linux-3.10.94/virt/
linux-3.10.94/virt/
linux-3.10.94/virt/
linux-3.10.94/virt/
cspeter@ubuntu:~$
Filesystem
/dev/mapper/ubuntu-
```

Snapshot Name: Snapshot Before Kernel Compilation

Snapshot Description

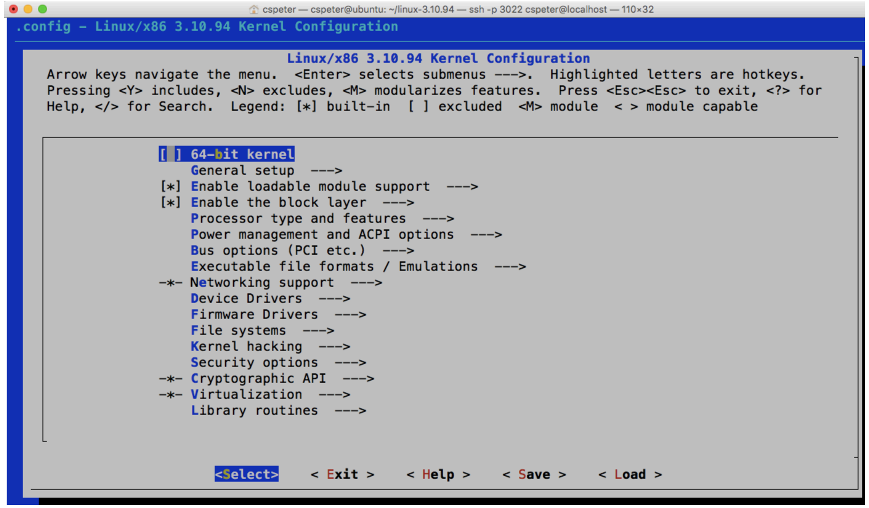
Step 2 – Creating .config for kernel compilation

- Before kernel compilation, a configuration file (.config) is necessary in the top Linux kernel source directory
  - /home/[your username]/linux-3.10.94



```
cspeter@ubuntu:~$ ls
linux-3.10.94  linux-3.10.94.tar.xz
cspeter@ubuntu:~$ cd linux-3.10.94/
cspeter@ubuntu:~/linux-3.10.94$ pwd
/home/cspeter/linux-3.10.94
cspeter@ubuntu:~/linux-3.10.94$ 
```

- Download the config file provided on the course web and put it at the top directory of Linux Kernel source code
- After that, start the Linux Kernel configuration
  - sudo make menuconfig

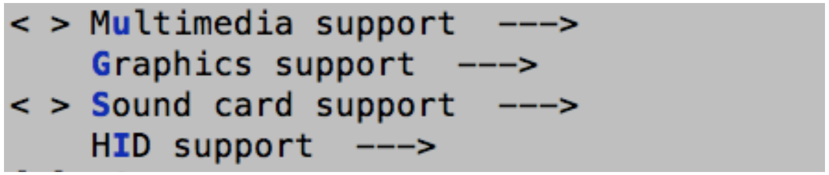


```
cspeter — cspeter@ubuntu: ~/linux-3.10.94 — ssh -p 3022 cspeter@localhost — 110×32
.config - Linux/x86 3.10.94 Kernel Configuration
                         Linux/x86 3.10.94 Kernel Configuration
  Arrow keys navigate the menu.  <Enter> selects submenus --->.  Highlighted letters are hotkeys.
  Pressing <Y> includes, <N> excludes, <M> modularizes features.  Press <Esc><Esc> to exit, <?> for
  Help, </> for Search.  Legend: [*] built-in  [ ] excluded  <M> module  < > module capable

                  [ ] 64-bit kernel
                      General setup  --->
                  [*] Enable loadable module support  --->
                  [*] Enable the block layer  --->
                      Processor type and features  --->
                      Power management and ACPI options  --->
                      Bus options (PCI etc.)  --->
                      Executable file formats / Emulations  --->
                  -*- Networking support  --->
                      Device Drivers  --->
                      Firmware Drivers  --->
                      File systems  --->
                      Kernel hacking  --->
                      Security options  --->
                  -*- Cryptographic API  --->
                  -*- Virtualization  --->
                      Library routines  --->

                 <Select>    < Exit >    < Help >    < Save >    < Load >
```

Step 3 – Configuration of .config

- Linux is an open-source kernel, it is possible to customize it to optimize your desired performance
- In this course, we don't need device drivers such as USB support, sound card support
- Menu Control
  - Arrow keys to navigate the menu
  - Enter to select submenus
  - Pressing <Y> to include the feature
  - Pressing <N> to exclude
  - Pressing <M> to include the feature as a module (Modules will be explained later in this course)
- Note: Remember to choose "Save" at the end of configuration

- Example: Sound card support can be disabled if needed



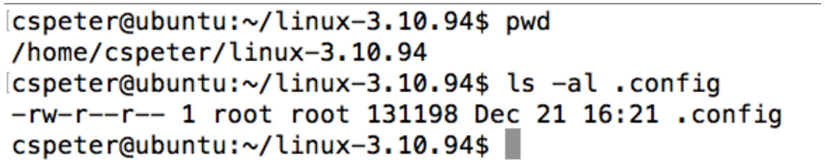
```
< > Multimedia support  --->
    Graphics support  --->
< > Sound card support  --->
    HID support  --->
```

If you make some mistakes, the new kernel may not be bootable

Don't worry!
If we make any mistake, we can restore from the snapshot and start over!

Step 4 – Linux Kernel Compilation

- Make sure that the current directory is located at the top Linux kernel source directory
- Make sure that .config is created



```
[cspeter@ubuntu:~/linux-3.10.94$ pwd
/home/cspeter/linux-3.10.94
[cspeter@ubuntu:~/linux-3.10.94$ ls -al .config
-rw-r--r-- 1 root root 131198 Dec 21 16:21 .config
cspeter@ubuntu:~/linux-3.10.94$ █
```

- Command to start Linux kernel compilation
  - <u>Command:</u> sudo make –j4
    - -jX is a speed up option if we have multiple processors
  - It will take around 25 minutes in a lab machine
- Tips
  - You can press Ctrl-C to stop the build process and then poweroff,
  - Re-compile later by "sudo make" again

Step 5 – Install the Compiled Kernel Image

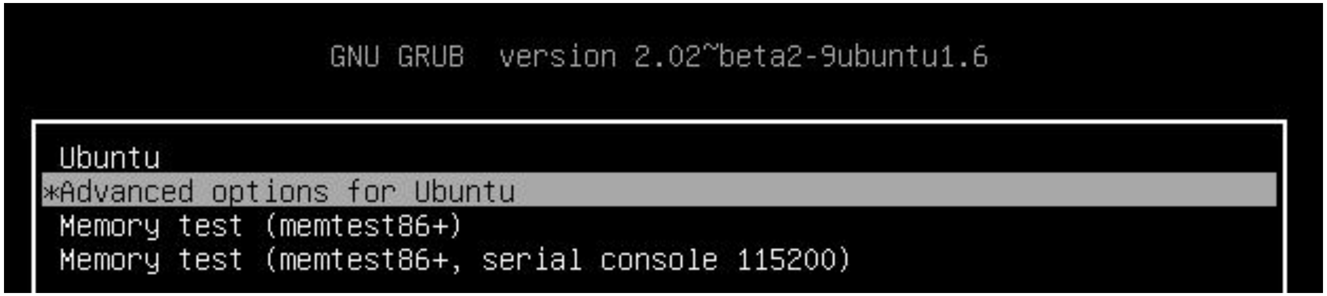
- After kernel compilation, we need to install modules and the kernel image
  - sudo make modules
  - sudo make modules_install
  - sudo make install
  - It is important to make and install the modules first, and make install second
- Installed modules:
  - /lib/modules
- Installed kernel images:
  - /boot

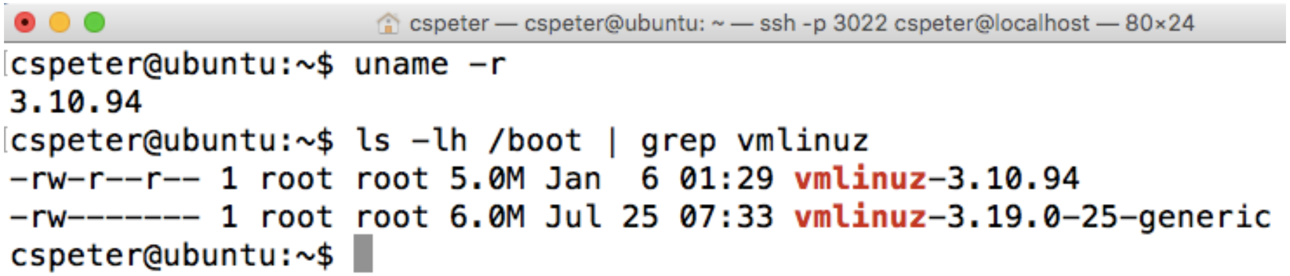
Step 6 – Configure the boot loader

- The final step is to configure the boot loader
- Edit the file by the following command:
  - sudo nano /etc/default/grub
- As we need to select our new kernel during boot time, we need to increase the timeout of the boot loader
  - GRUB_TIMEOUT=10
- After that, save the file and type in the following command to update the boot loader
  - sudo update-grub
- Reboot the machine by typing the following command:
  - sudo shutdown –r now

Step 6 – Reboot and Check

- Select the new kernel in the boot loader (GRUB) menu
- Once the system is booted, login and check if you have the new kernel being used
  - uname –r
  - The expected kernel version is 3.10.94
  - Congratulation! Now, we have a customized Kernel, smaller in size and boot much faster than the bundled kernel


GNU GRUB version 2.02~beta2-9ubuntu1.6
Ubuntu
*Advanced options for Ubuntu
Memory test (memtest86+)
Memory test (memtest86+, serial console 115200)


cspeter — cspeter@ubuntu: ~ — ssh -p 3022 cspeter@localhost — 80×24

```
cspeter@ubuntu:~$ uname -r
3.10.94
cspeter@ubuntu:~$ ls -lh /boot | grep vmlinuz
-rw-r--r-- 1 root root 5.0M Jan  6 01:29 vmlinuz-3.10.94
-rw------- 1 root root 6.0M Jul 25 07:33 vmlinuz-3.19.0-25-generic
cspeter@ubuntu:~$ 
```