

CS 332/532 Systems Programming

Lecture 14

Linux Crash Course

Professor : Kai Zhao – UAB CS

Agenda

- Command line arguments
 - argc
 - argv
- Linux Crash Course

Command-Line Arguments

- In C, command-line arguments allow users to pass inputs to a program when it is executed.
- Two parameters are used in the main function to handle these inputs:
- **int argc** (argument count): It represents the number of command-line arguments.
- **char *argv[]** (argument vector): It is an array of strings (pointers to characters) representing each argument.

- **argc:** Holds the number of arguments passed. It always includes the program's name as the first argument, so argc is at least 1.
- **argv[]:** An array of strings where:
 - **argv[0]:** The name of the program or the path to the executable.
 - **argv[1], argv[2], ..., argv[argc-1]:** The additional arguments passed by the user.

C hello.c X

C hello.c

```
1  #include <stdio.h>
2
3  int main(int argc, char *argv[]) {
4      printf("Number of arguments: %d\n", argc);
5      for (int i = 0; i < argc; i++) {
6          printf("Argument %d: %s\n", i, argv[i]);
7      }
8      return 0;
9  }
```

- (base) mahmutunan@Mahmuts-MacBook-Pro c_practice % gcc hello.c -o hello
- (base) mahmutunan@Mahmuts-MacBook-Pro c_practice % ls
hello hello.c
- (base) mahmutunan@Mahmuts-MacBook-Pro c_practice % ./hello
Number of arguments: 1
Argument 0: ./hello
- (base) mahmutunan@Mahmuts-MacBook-Pro c_practice % ./hello somefilename someparameter somethingelse
Number of arguments: 4
Argument 0: ./hello
Argument 1: somefilename
Argument 2: someparameter
Argument 3: somethingelse
- (base) mahmutunan@Mahmuts-MacBook-Pro c_practice %

C additionProgram.c

```
1  #include <stdio.h>
2  #include <stdlib.h>
3
4  int main(int argc, char *argv[]) {
5      if (argc != 3) {
6          printf("Usage: %s num1 num2\n", argv[0]);
7          return 1;
8      }
9      int num1 = atoi(argv[1]);
10     int num2 = atoi(argv[2]);
11     printf("Sum: %d\n", num1 + num2);
12     return 0;
13 }
14
```

```
(base) mahmutunan@Mahmuts-MacBook-Pro c_practice % gcc -o additionProgram additionProgram.c
(base) mahmutunan@Mahmuts-MacBook-Pro c_practice % ls
additionProgram      additionProgram.c
(base) mahmutunan@Mahmuts-MacBook-Pro c_practice % ./additionProgram 35 55
Sum: 90
(base) mahmutunan@Mahmuts-MacBook-Pro c_practice %
```

C additionProgram.c

```
1  #include <stdio.h>
2  #include <stdlib.h>
3
4  int main(void) {
5      if (argc != 3) {
6          printf("Usage: %s num1 num2\n", argv[0]);
7          return 1;
8      }
9      int num1 = atoi(argv[1]);
10     int num2 = atoi(argv[2]);
11     printf("Sum: %d\n", num1 + num2);
12     return 0;
13 }
```

```
(base) mahmutunan@Mahmuts-MacBook-Pro c_practice % gcc -o additionProgram addit
additionProgram.c:5:9: error: use of undeclared identifier 'argc'
    if (argc != 3) {
        ^
additionProgram.c:6:41: error: use of undeclared identifier 'argv'
    printf("Usage: %s num1 num2\n", argv[0]);
                                    ^
additionProgram.c:9:21: error: use of undeclared identifier 'argv'
    int num1 = atoi(argv[1]);
                    ^
additionProgram.c:10:21: error: use of undeclared identifier 'argv'
    int num2 = atoi(argv[2]);
                    ^
4 errors generated.
```

Linux Certifications

- Most of the hiring manager are looking to recruit Linux professionals.
- The emergence of open cloud platforms is creating increasing demand for Linux professionals who have the right expertise
- Linux-certified professionals always be a better position in the job market
- Employers are looking for more Linux talent.
- Better salary increments for Linux certified professionals
- Some famous certificates
 - Red Hat Linux SuSE Linux
 - Linux Professional Institute (LPIC)
 - CompTIA
 - Linux Foundation
 - Oracle

Linux

- Linux and Unix are different, but they do have a relationship with each other
 - Linux is derived from Unix.
- Linux is just the kernel and not the complete OS. This Linux kernel is generally packaged in Linux distributions which thereby makes it a complete OS
- There are many different versions of Linux, and the core of Linux is free to distribute and use.
- Normally, distributions are made for specific reasons and have been tailored to address a series of concerns

Linux

- Popular Linux distributions
 - Ubuntu
 - Debian
 - CentOS
 - Red Hat

Installation

- Fresh Install
- Dual Booting
- Live CD/DVD
- Linux as a VM

VirtualBox

. Download VirtualBox from

<https://www.virtualbox.org/wiki/Downloads> .

(win, osx, or linux)

2. Install VirtualBox with default setting.

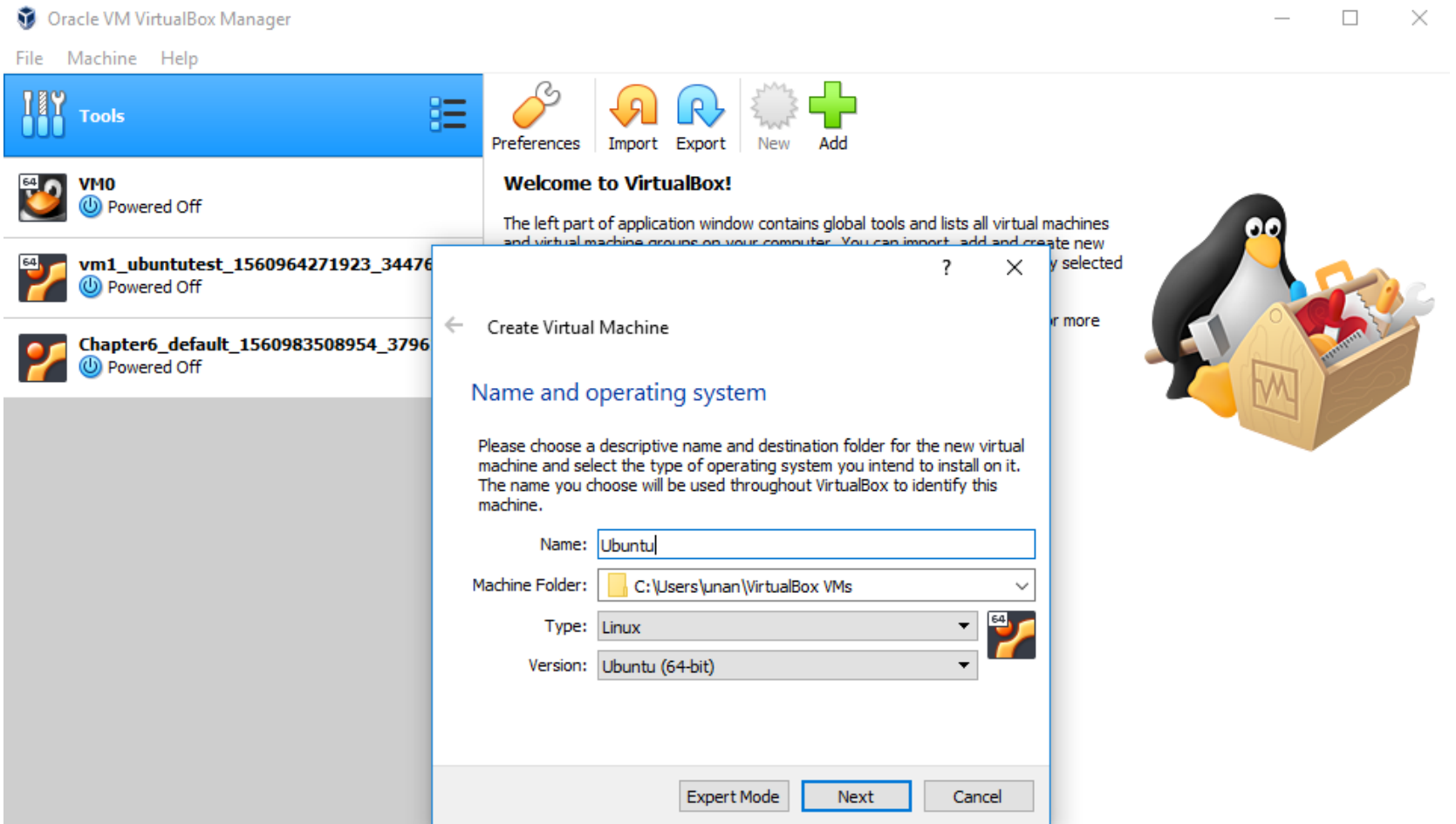


Ubuntu - Desktop

- We will download the ISO file of the Linux distribution.
- Download the Ubuntu for Desktop;
- <https://ubuntu.com/desktop>

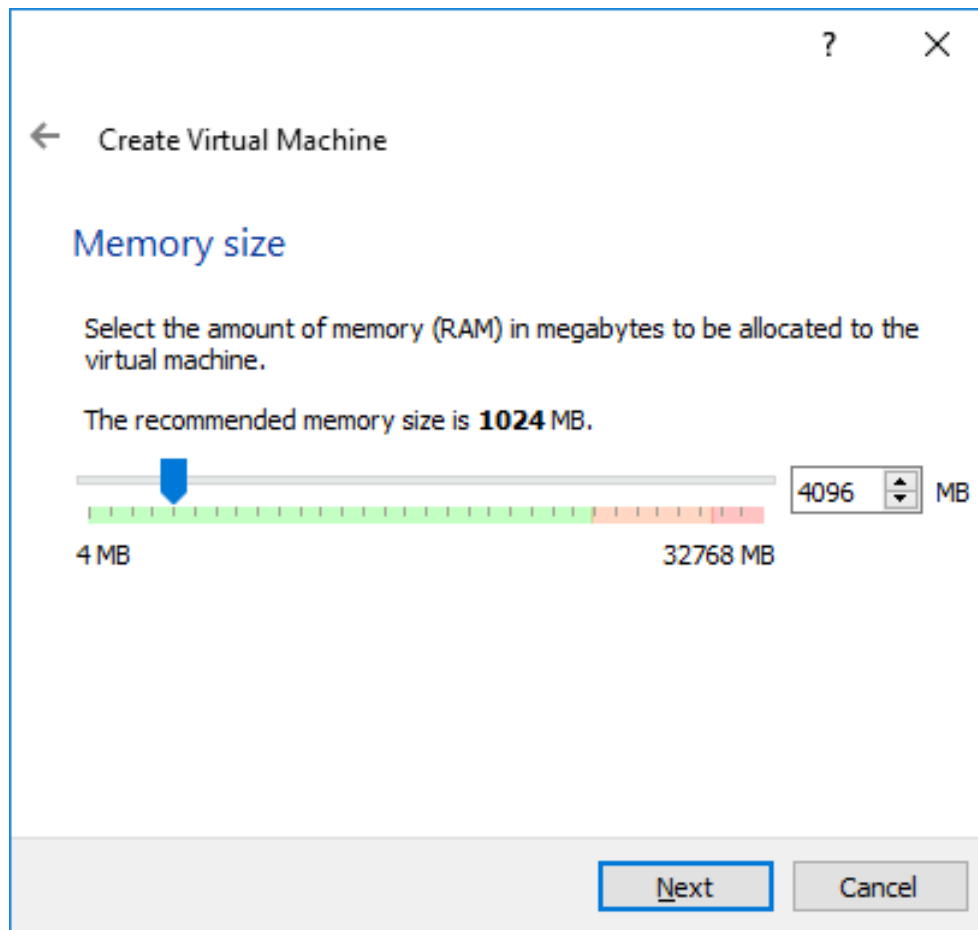
Install Ubuntu on VM

- Start the VirtualBox and click on the New symbol. Give the virtual OS a relevant name.



Allocate RAM

Allocate RAM to the virtual OS. My system has 32GB of RAM and I decided to allocate 4GB of RAM to it. You can use more RAM if your system has enough extra RAM.



← 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 **1024 MB**.

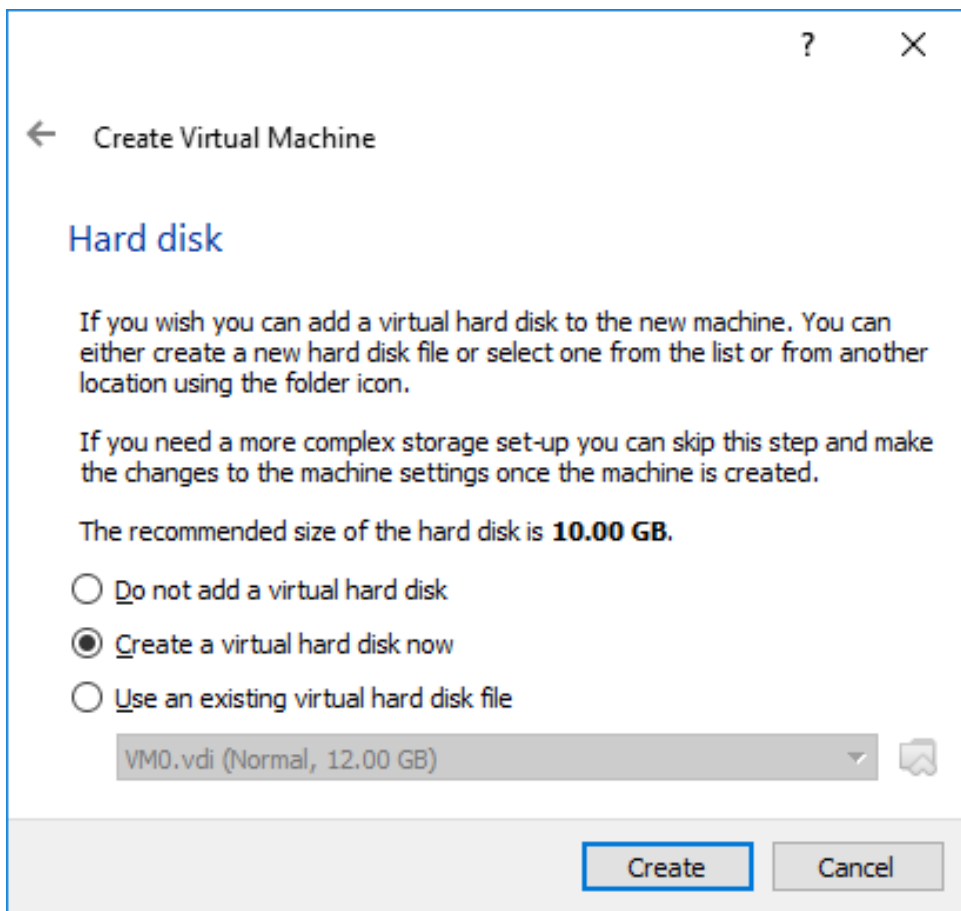
4 MB 32768 MB

4096 MB

Next Cancel

Virtual Disk

- Create a virtual disk. This works as the hard disk of the virtual Linux system. This is where the virtual system will store its files.



← Create Virtual Machine

Hard disk

If you wish you can add a virtual hard disk to the new machine. You can either create a new hard disk 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 disk is **10.00 GB**.

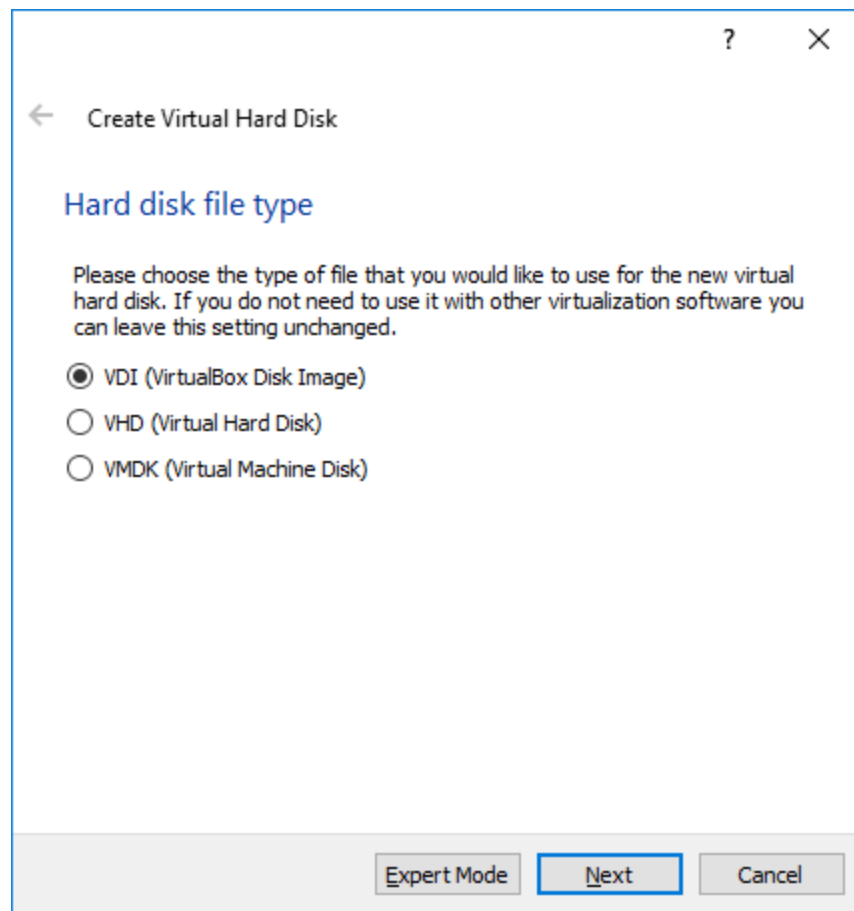
☐ Do not add a virtual hard disk

☒ Create a virtual hard disk now

☐ Use an existing virtual hard disk file

VM0.vdi (Normal, 12.00 GB) 

Create Cancel



← Create Virtual Hard Disk

Hard disk file type

Please choose the type of file that you would like to use for the new virtual hard disk. If you do not need to use it with other virtualization software you can leave this setting unchanged.

☒ VDI (VirtualBox Disk Image)

☐ VHD (Virtual Hard Disk)

☐ VMDK (Virtual Machine Disk)

Expert Mode Next Cancel

Dynamically Allocate / Size

?

×

← Create Virtual Hard Disk

Storage on physical hard disk

Please choose whether the new virtual hard disk file should grow as it is used (dynamically allocated) or if it should be created at its maximum size (fixed size).

A **dynamically allocated** hard disk file will only use space on your physical hard disk as it fills up (up to a maximum **fixed size**), although it will not shrink again automatically when space on it is freed.

A **fixed size** hard disk file may take longer to create on some systems but is often faster to use.

☒ Dynamically allocated

☐ Fixed size

Next

Cancel

?

×

← Create Virtual Hard Disk

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.

C:\Users\jnan\VirtualBox VMs\Ubuntu\Ubuntu.vdi

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.

10.00 GB

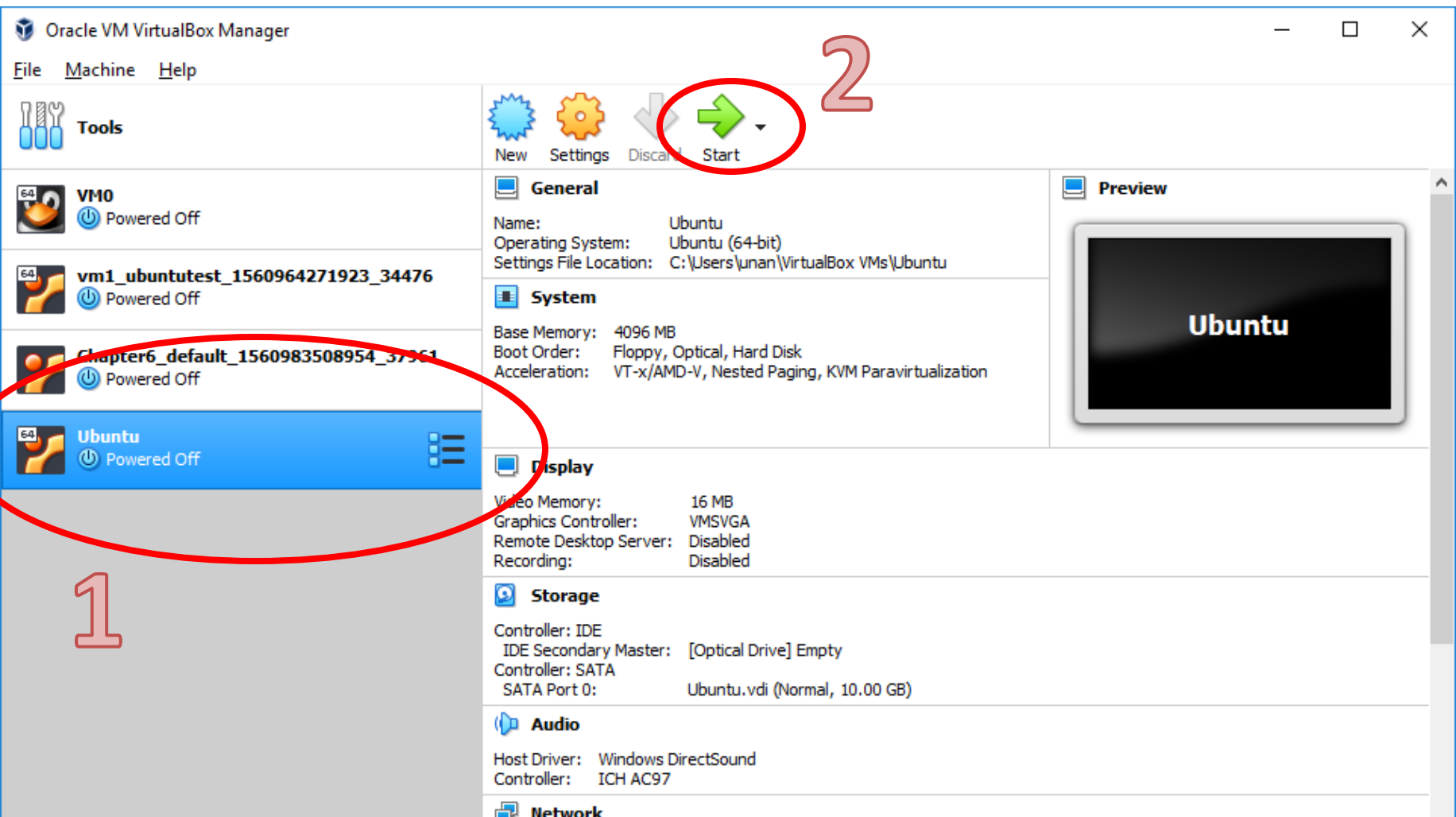
4.00 MB2.00 TB

Create

Cancel

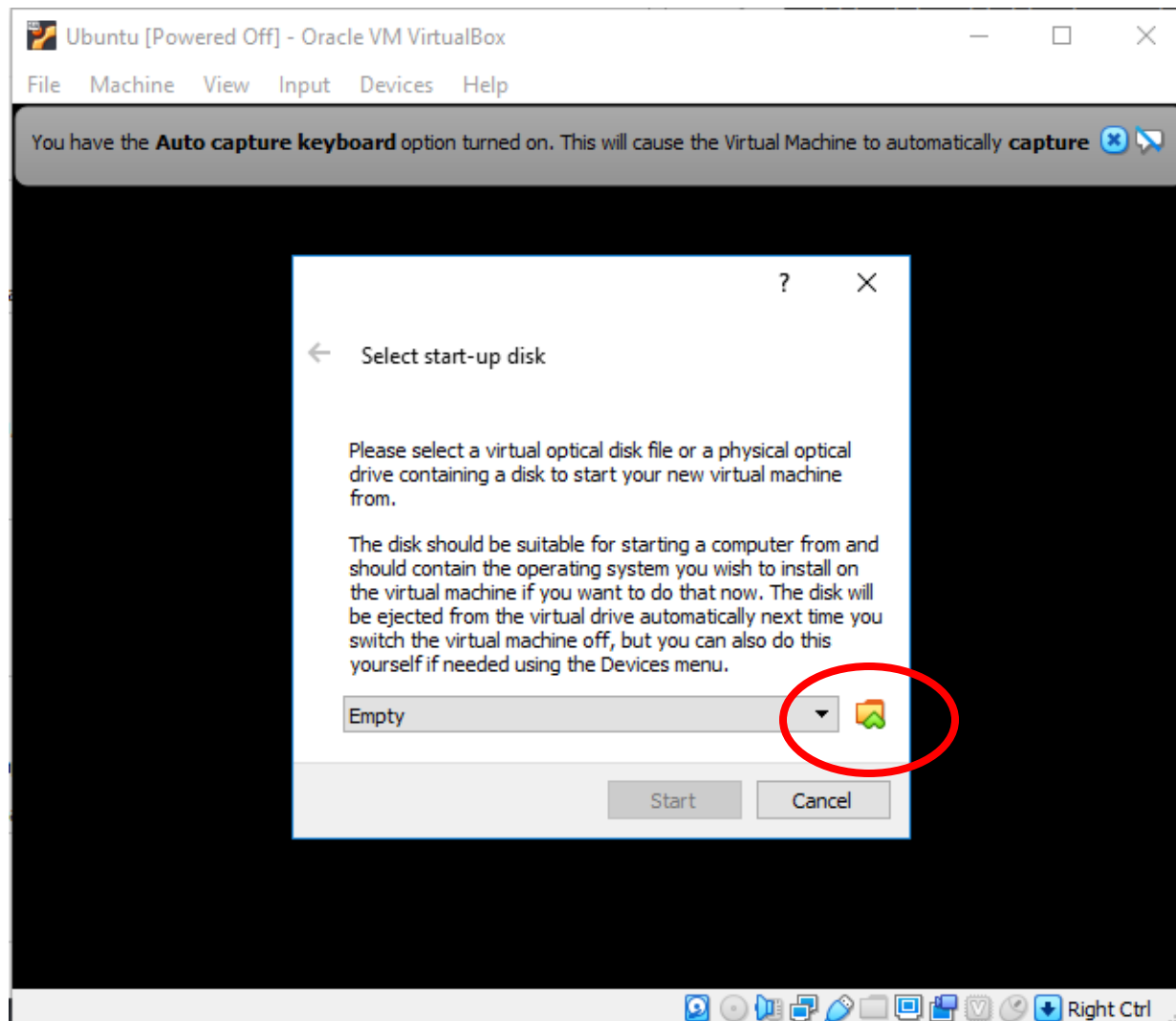
Boot the ISO

- Once everything is in place, it's time to boot that ISO and install Linux as a virtual operating system.



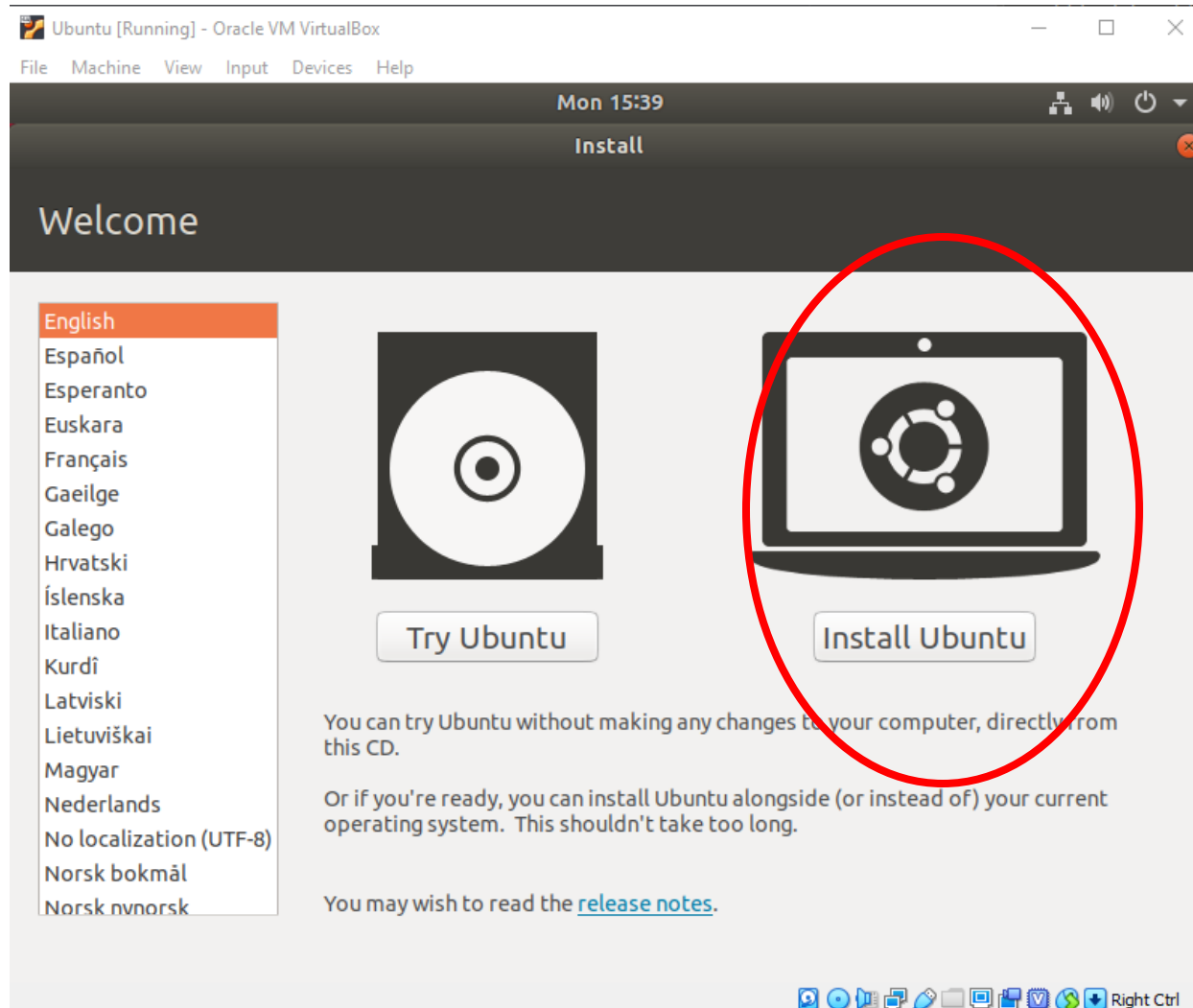
Select Start-up Disk

- If VirtualBox doesn't detect the Linux ISO, browse to its location by clicking the folder icon as shown in the picture below:



Install Ubuntu

- Soon you'll find yourself inside Linux. You should be presented with the option to install it.



Select Language

Install

Keyboard layout

Choose your keyboard layout:

English (Ghana)	English (US)
English (Nigeria)	English (US) - Cherokee
English (South Africa)	English (US) - English (Colemak)
English (UK)	English (US) - English (Dvorak)
English (US)	English (US) - English (Dvorak, alt. intl.)
Esperanto	English (US) - English (Dvorak, intl., with dead keys)
Estonian	English (US) - English (Dvorak, left-handed)
Faroese	English (US) - English (Dvorak, right-handed)
Filipino	English (US) - English (Macintosh)

Type here to test your keyboard

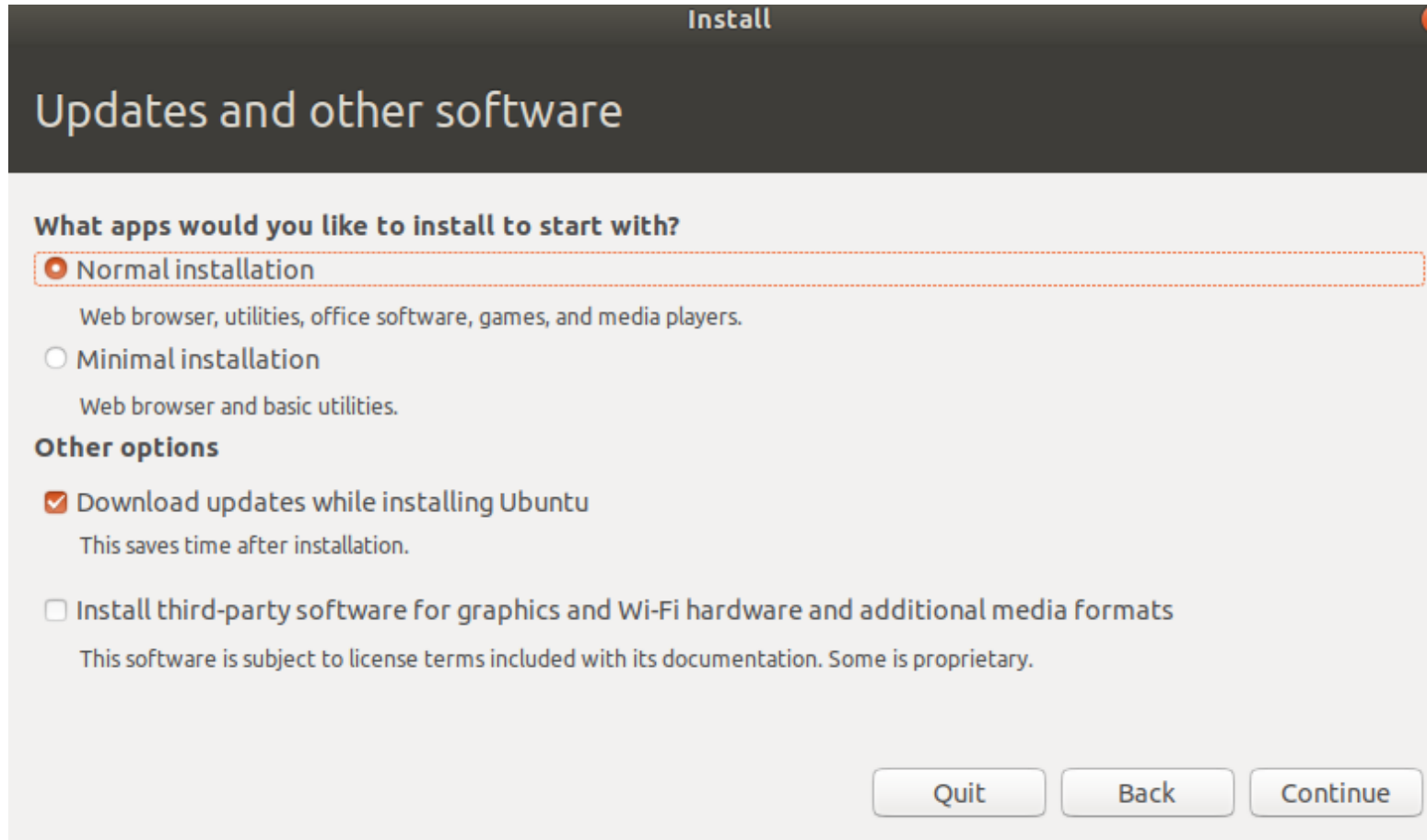
Detect Keyboard Layout

Quit

Back

Continue

Normal Installation / Download Updates



The image shows a screenshot of the Ubuntu installer window. The window has a dark grey title bar with the word "Install" in white. Below the title bar, the main window has a dark grey header with the text "Updates and other software" in white. The main content area is light grey and contains the following text and options:

What apps would you like to install to start with?

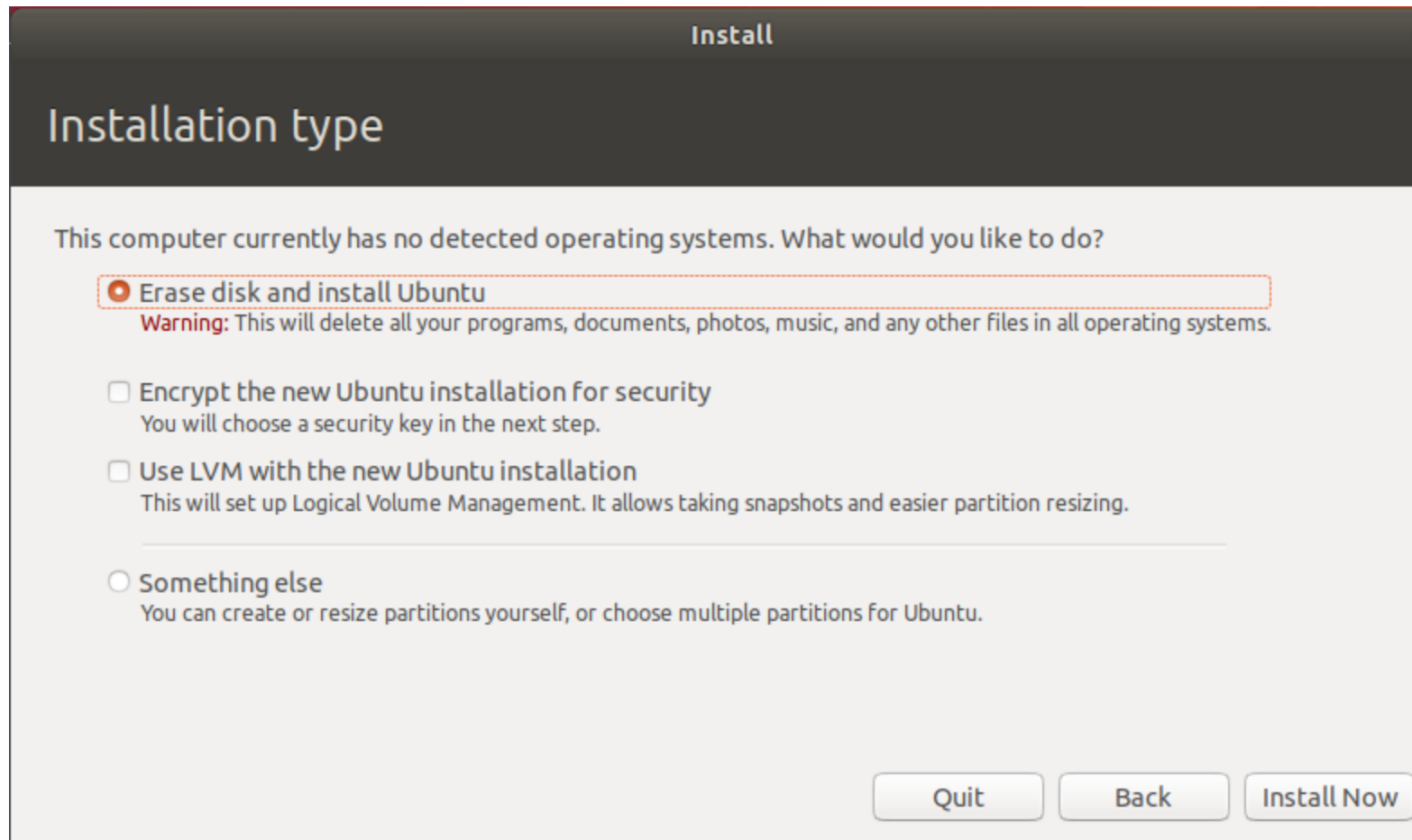
- ☒ **Normal installation**
Web browser, utilities, office software, games, and media players.
- ☐ **Minimal installation**
Web browser and basic utilities.

Other options

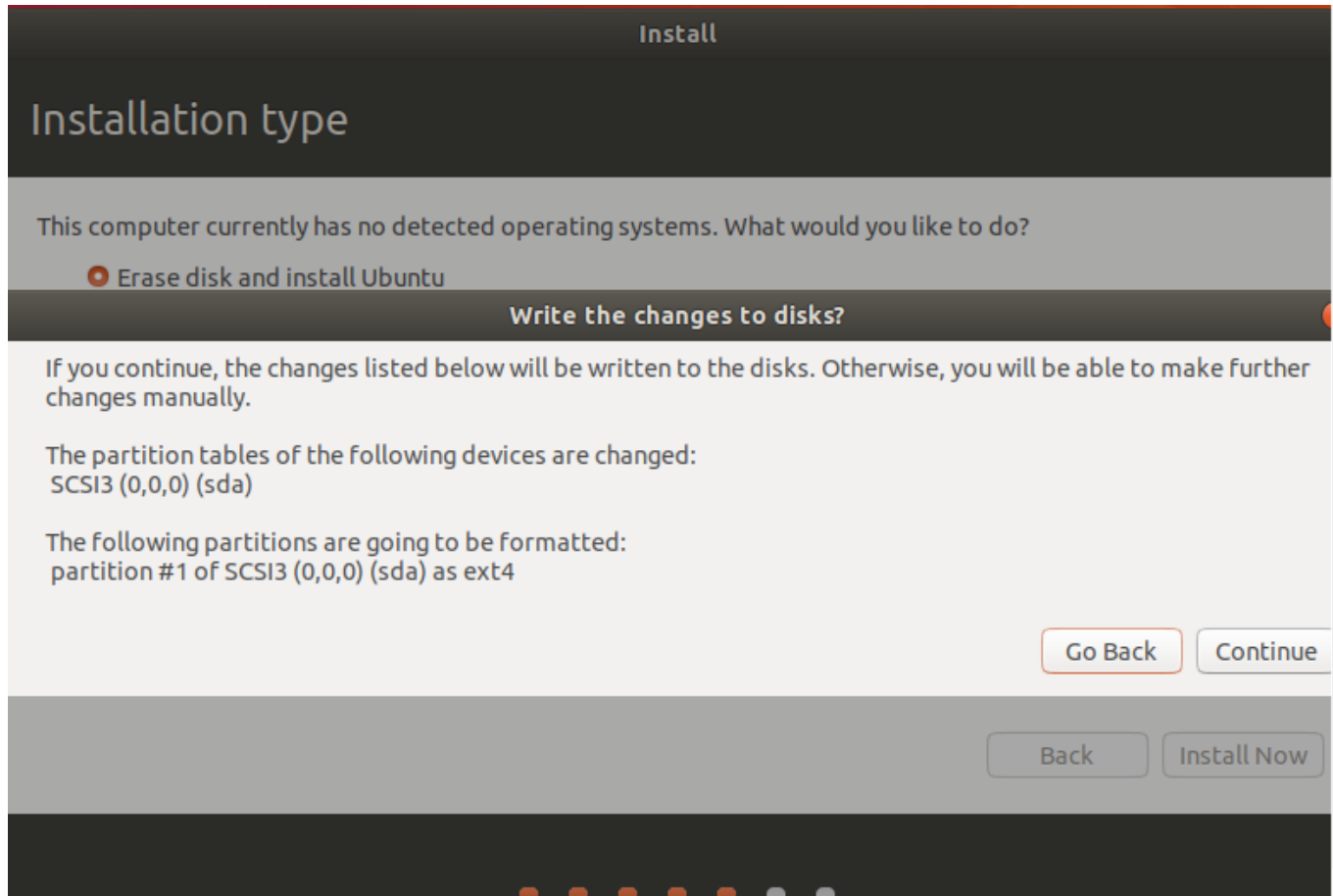
- ☒ **Download updates while installing Ubuntu**
This saves time after installation.
- ☐ **Install third-party software for graphics and Wi-Fi hardware and additional media formats**
This software is subject to license terms included with its documentation. Some is proprietary.

At the bottom right of the window, there are three buttons: "Quit", "Back", and "Continue".

Select 'Erase disk and install Ubuntu'. Don't worry. It won't delete anything on your Windows operating system. You are using the virtual disk space of 15-20GB that we created in previous steps. It won't impact the real operating system.



Continue



where are you ?

Install

Where are you?



Birmingham

Back Continue

Choose a password you can remember (Yes, I choose 1234 😊)

Install

Who are you?

Your name: ✓

Your computer's name: ✓
The name it uses when it talks to other computers.

Pick a username: ✓

Choose a password: **Short password**

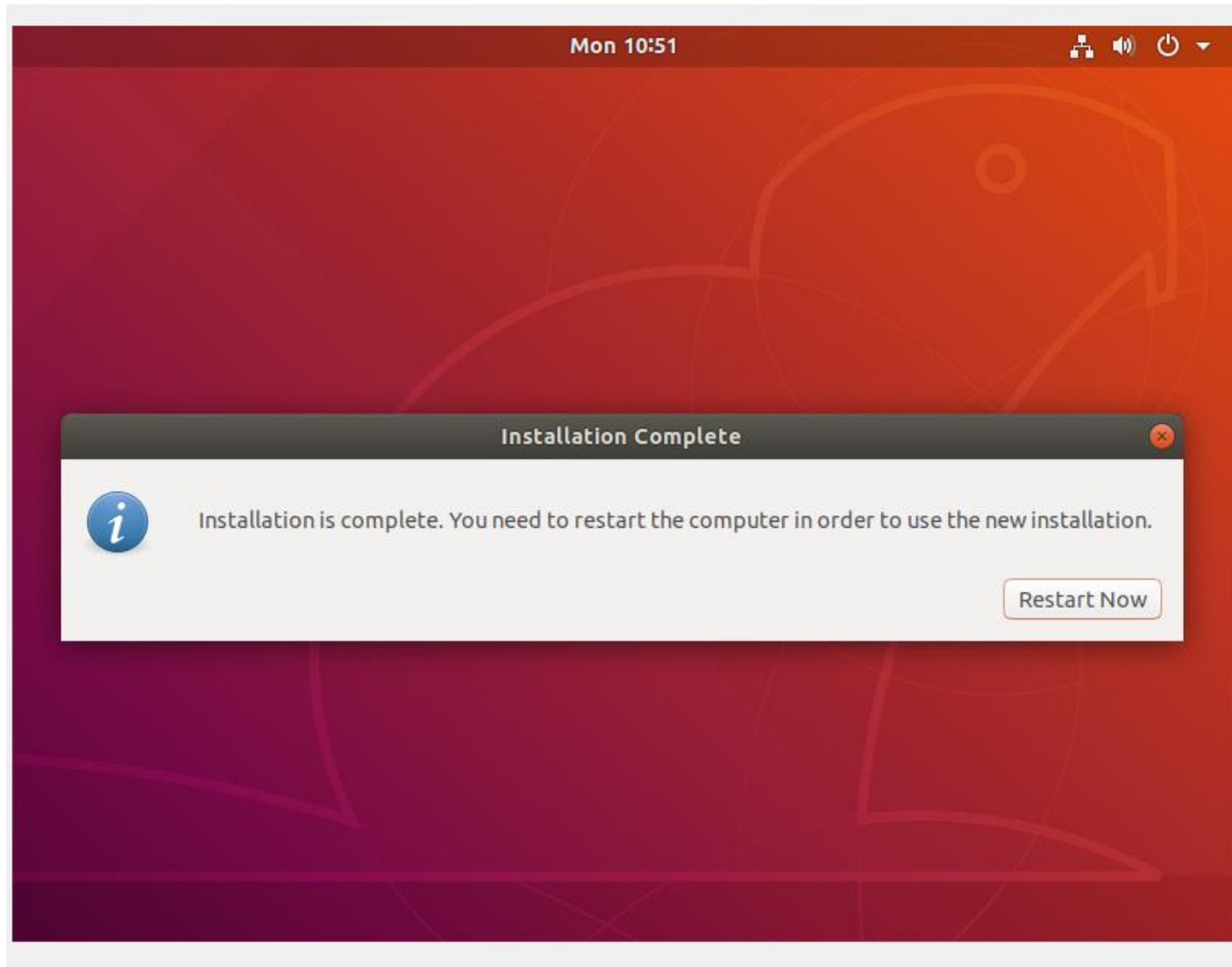
Confirm your password: ✓

☐ Log in automatically
☒ Require my password to log in

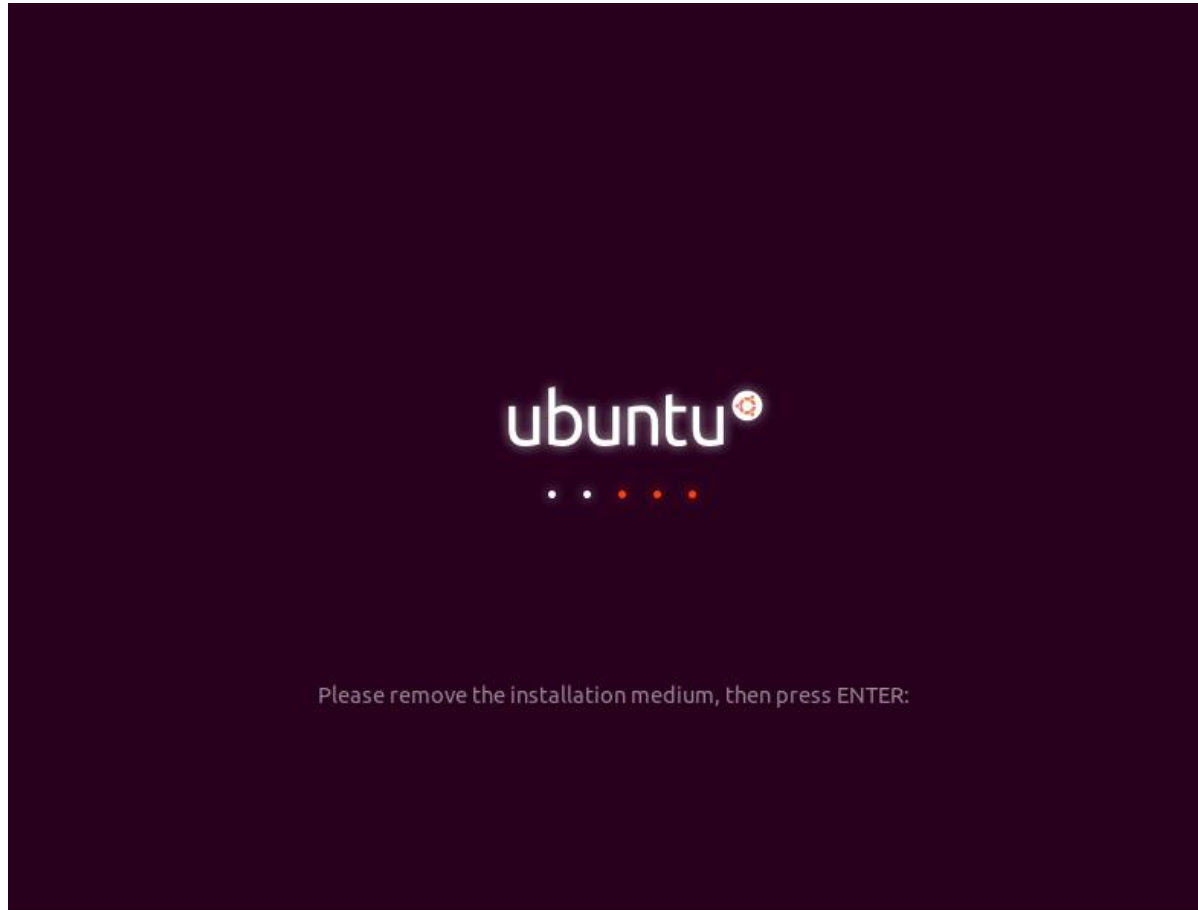
You are almost done!



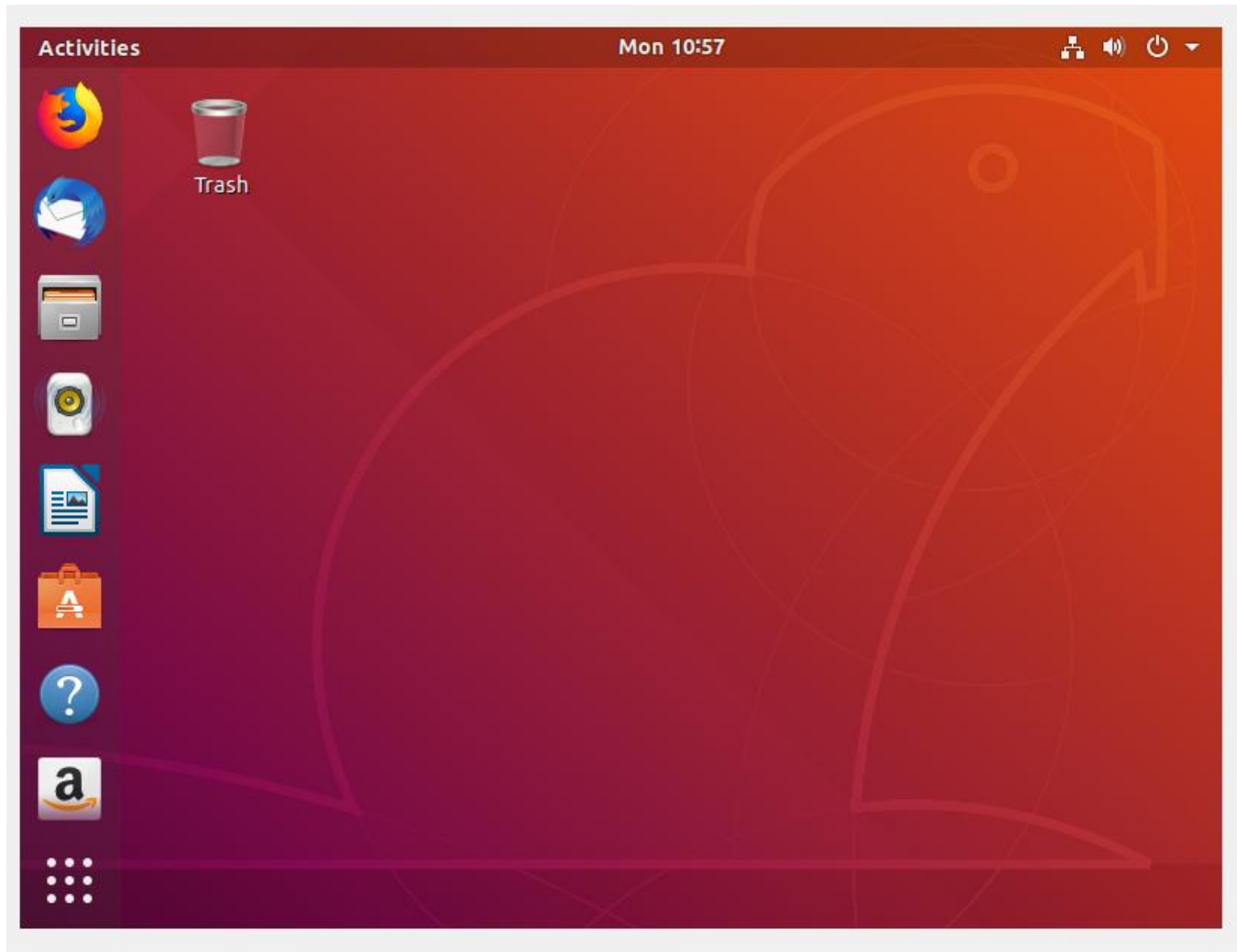
Restart Your Computer (VM)



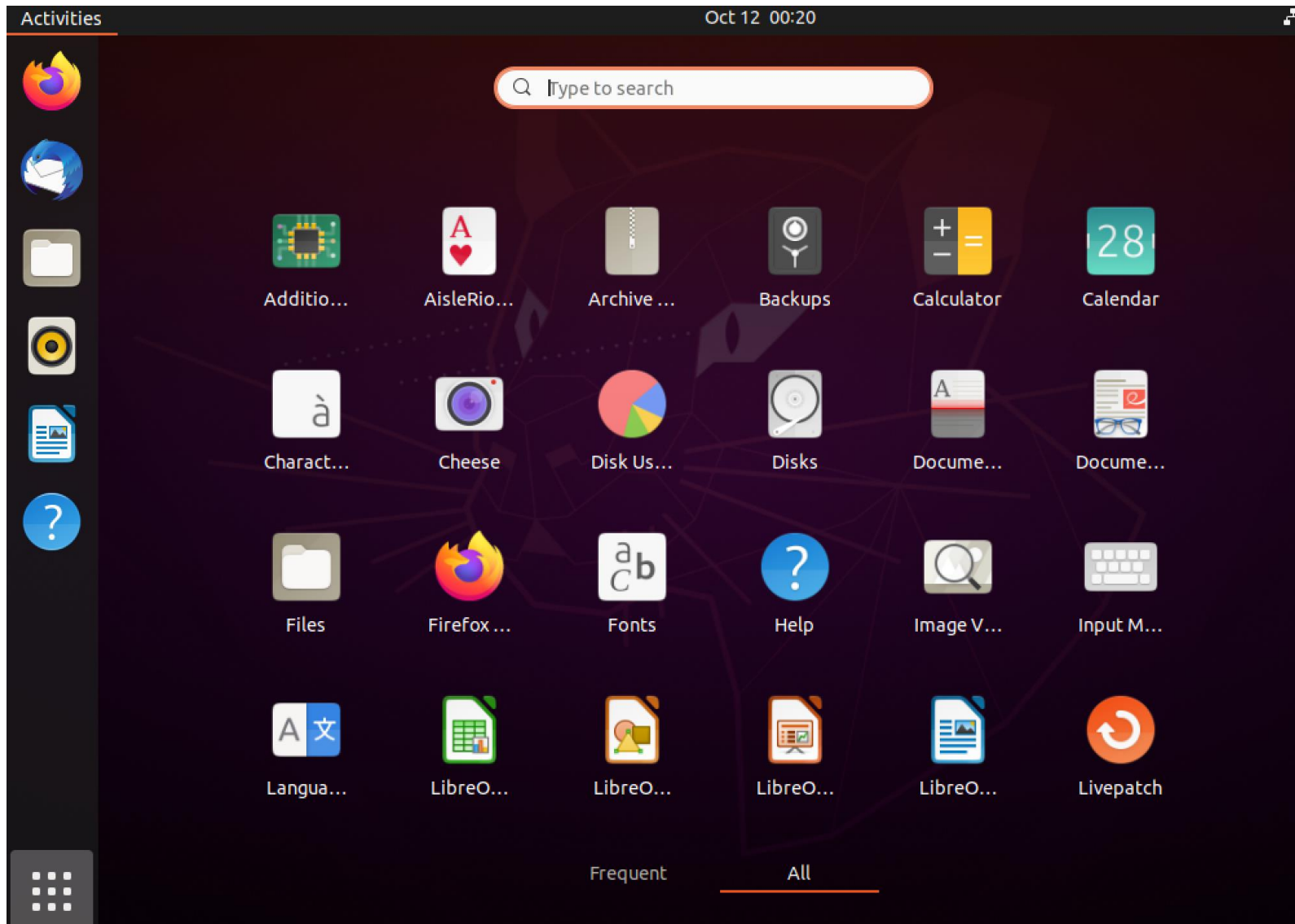
If it gets stuck on the screen below, you may close the VirtualBox.



Ready To Go!!!

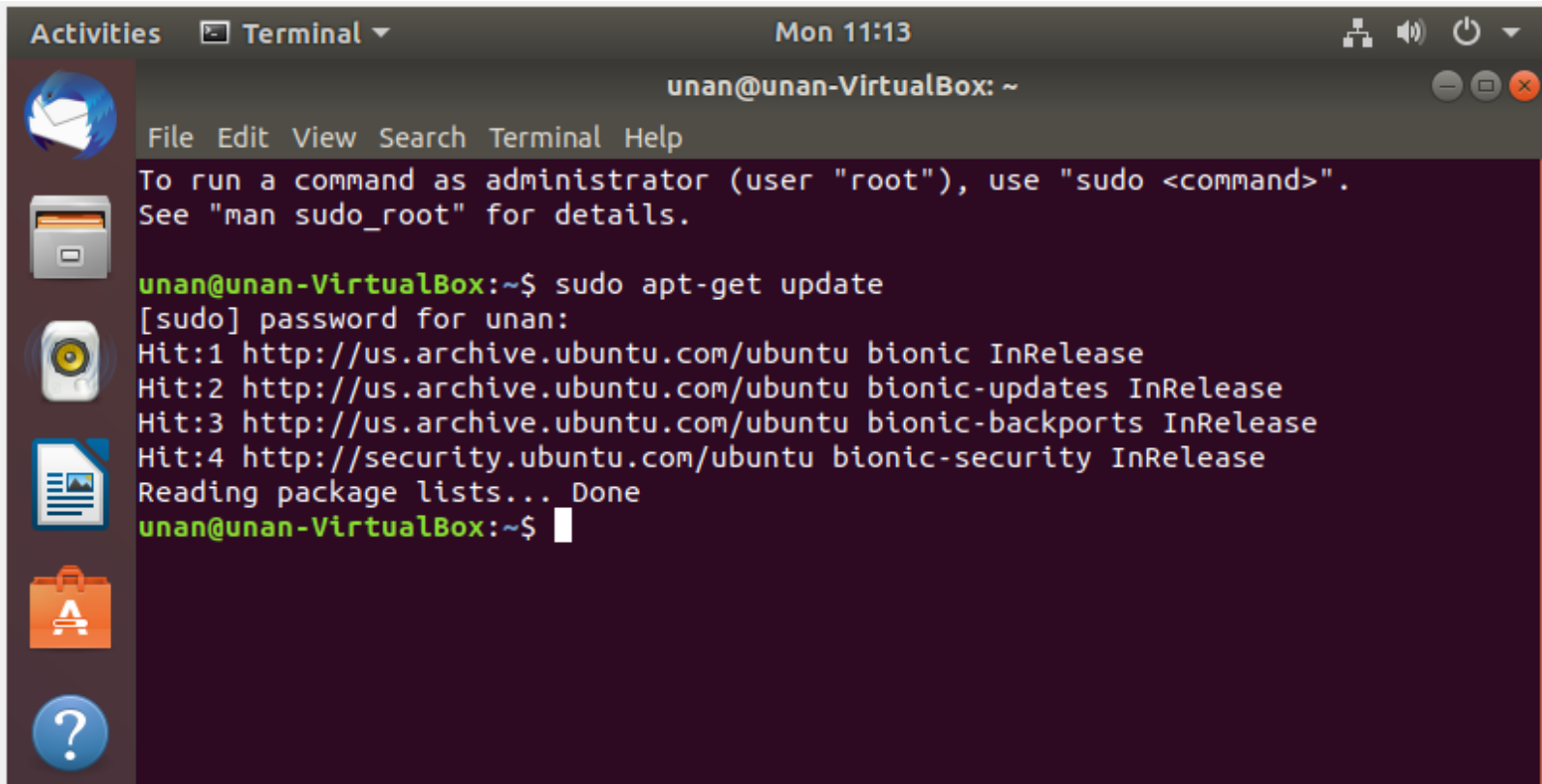


You can start terminal



Linux

- Start Linux Terminal and install the updates
 - `sudo apt-get update`



The screenshot shows a Linux terminal window titled "Terminal" with a dark theme. The window has a menu bar with "File", "Edit", "View", "Search", "Terminal", and "Help". The prompt is "unan@unan-VirtualBox: ~". The terminal displays the command "sudo apt-get update" and its output, which includes a list of package sources and their release status. The output is as follows:

```
unan@unan-VirtualBox:~$ sudo apt-get update
[sudo] password for unan:
Hit:1 http://us.archive.ubuntu.com/ubuntu bionic InRelease
Hit:2 http://us.archive.ubuntu.com/ubuntu bionic-updates InRelease
Hit:3 http://us.archive.ubuntu.com/ubuntu bionic-backports InRelease
Hit:4 http://security.ubuntu.com/ubuntu bionic-security InRelease
Reading package lists... Done
unan@unan-VirtualBox:~$
```

- `sudo apt-get upgrade`

What we have learned so far...

- ls
- cd
- rm
- ps
- cat
- touch
-
-

Copying a file or directory

- `cp [options] <source> <destination>`

```
unan@unan-VirtualBox:~/lecture36$ ls
dir1  dir2  somefile.txt
unan@unan-VirtualBox:~/lecture36$ cp somefile.txt dir1
unan@unan-VirtualBox:~/lecture36$ ls
dir1  dir2  somefile.txt
unan@unan-VirtualBox:~/lecture36$ cd dir1
unan@unan-VirtualBox:~/lecture36/dir1$ ls
file1.txt  somefile.txt
unan@unan-VirtualBox:~/lecture36/dir1$
```

Moving a File or Directory

- `mv [options] <source> <destination>`

```
unan@unan-VirtualBox:~/lecture36$ mkdir dir3
unan@unan-VirtualBox:~/lecture36$ mv dir1 dir3
unan@unan-VirtualBox:~/lecture36$ ls
dir2  dir3  somefile.txt
unan@unan-VirtualBox:~/lecture36$
```

```
unan@unan-VirtualBox:~/lecture36$ mv ./dir2 ./dir2new
unan@unan-VirtualBox:~/lecture36$ ls
dir2new  dir3  somefile.txt
unan@unan-VirtualBox:~/lecture36$
```

nl - line numbers

- `nl [-options] [path]`

```
unan@unan-VirtualBox:~/lecture36$ cat somefile.txt
Mahmut Unan 55
John Smith 45
John doe 67
Julie Doe 99
Zack Zetta 88
unan@unan-VirtualBox:~/lecture36$ nl somefile.txt
 1  Mahmut Unan 55
 2  John Smith 45
 3  John doe 67
 4  Julie Doe 99
 5  Zack Zetta 88
```

cut

- `cut [-options] [path]`

```
unan@unan-VirtualBox:~/lecture36$ cut -f 1 -d ' ' somefile.txt
Mahmut
John
John
Julie
Zack
unan@unan-VirtualBox:~/lecture36$
```

other useful data ops

- sed
 - sed stands for **Stream Editor** and it effectively allows us to do a search and replace on our data. It is quite a powerful command but we will use it here in it's basic format.
 - **sed <expression> [path]**
- uniq
 - uniq stands for **unique** and it's job is to remove duplicate lines from the data. One limitation however is that those lines must be adjacent (ie, one after the other)
 - **uniq [options] [path]**

diff

```
unan@unan-VirtualBox:~/lecture36$ cat somefile.txt
Mahmut Unan 55
John Smith 45
John doe 67
Julie Doe 99
Zack Zetta 88
unan@unan-VirtualBox:~/lecture36$ cat somefile2.txt
Mahmut Unan 55
John Smith 45
John doe 67
Julie Doe 99
Zack Zetta 88
some extra line
unan@unan-VirtualBox:~/lecture36$ diff somefile.txt somefile2.txt
5a6
> some extra line
unan@unan-VirtualBox:~/lecture36$
```


grep / egrep

- `grep [options] <pattern> [files]`
- `egrep [options] <pattern> [files]`

```
unan@unan-VirtualBox:~/lecture36$ grep 'o' somefile.txt
John Smith 45
John doe 67
Julie Doe 99
unan@unan-VirtualBox:~/lecture36$ grep -c 'o' somefile.txt
3
```

Wildcards

- * - zero or more characters
- ? - a single character
- [] - a range of characters

Networking / Communication

- SSH
 - SSH username@ip-address or hostname
- Ping
 - ping hostname
 - ping ipaddress

```
unan@unan-VirtualBox:~/lecture36$ ping www.google.com
PING www.google.com (64.233.185.99) 56(84) bytes of data.
64 bytes from yb-in-f99.1e100.net (64.233.185.99): icmp_seq=1 ttl=63 time=23.2 m
64 bytes from yb-in-f99.1e100.net (64.233.185.99): icmp_seq=2 ttl=63 time=24.8 m
64 bytes from yb-in-f99.1e100.net (64.233.185.99): icmp_seq=3 ttl=63 time=22.6 m
```

- ftp hostname

```
unan@unan-VirtualBox:~/lecture36$ ftp ftp.javatutorialhub.com
Connected to javatutorialhub.com.
220----- Welcome to Pure-FTPd [privsep] [TLS] -----
220-You are user number 1 of 50 allowed.
220-Local time is now 15:39. Server port: 21.
220-This is a private system - No anonymous login
220-IPv6 connections are also welcome on this server.
220 You will be disconnected after 15 minutes of inactivity.
Name (ftp.javatutorialhub.com:unan):
```

- telnet
 - telnet hostname

Linux Administrator

- Add new user
 - `sudo adduser mynewuser`
- Disable an account
 - `sudo passwd -l mynewuser`
- Delete an account
 - `sudo userdel -r mynewuser`
- Add user to usergroup
 - `sudo usermod -a -G home mynewuser`

finger

- This command is used to **procure information of the users on a Linux machine.**
- You can use it on both local & remote machines
- The syntax 'finger' gives data on all the logged users on the remote and local machine.
- The syntax 'finger username' specifies the information of the user.

date /cal / time

```
unan@unan-VirtualBox:~/lecture36$ date
Fri 20 Nov 2020 01:54:12 PM CST
unan@unan-VirtualBox:~/lecture36$ date +%a %d-%m-%y
Fri 20-11-20
unan@unan-VirtualBox:~/lecture36$ time ls
dir2new  dir3  somefile2.txt  somefile.txt

real    0m0.002s
user    0m0.001s
sys     0m0.000s
unan@unan-VirtualBox:~/lecture36$ cal
      November 2020
Su Mo Tu We Th Fr Sa
 1  2  3  4  5  6  7
 8  9 10 11 12 13 14
15 16 17 18 19 20 21
22 23 24 25 26 27 28
29 30
unan@unan-VirtualBox:~/lecture36$
```

wget / curl

- wget <https://cdn.kernel.org/pub/linux/kernel/v4.x/linux-4.17.2.tar.xz>
- curl https://www.example.com/

<https://curl.se/docs/manual.html>

<https://linuxize.com/post/wget-command-examples/>

gzip / tar/ bzip2

- gzip filename
- gzip -k filename
- gunzip
- tar [options] [archive-file] [file or directory to be archived]
- bzip2 somefilename
- bunzip2 somefilename

su / sudo

- `su <username>`
- `su root`
- `sudo apt install curl`

mount

- `mount -t type <device> <directory>`

```
unan@unan-VirtualBox:~/lecture36$ mount
sysfs on /sys type sysfs (rw,nosuid,nodev,noexec,relatime)
proc on /proc type proc (rw,nosuid,nodev,noexec,relatime)
udev on /dev type devtmpfs (rw,nosuid,noexec,relatime,size=1987276k,nr_ino
6819,mode=755)
devpts on /dev/pts type devpts (rw,nosuid,noexec,relatime,gid=5,mode=620,
e=000)
tmpfs on /run type tmpfs (rw,nosuid,nodev,noexec,relatime,size=403088k,mo
/dev/sda5 on / type ext4 (rw,relatime,errors=remount-ro)
securityfs on /sys/kernel/security type securityfs (rw,nosuid,nodev,noexe
ime)
```

shred

```

unan@unan-VirtualBox:~/lecture36$ ls
dir2new dir3 somefile2.txt somefile.txt
unan@unan-VirtualBox:~/lecture36$ shred somefile.txt
unan@unan-VirtualBox:~/lecture36$ ls
dir2new dir3 somefile2.txt somefile.txt
unan@unan-VirtualBox:~/lecture36$ cat somefile.txt
  uIoneJ\;  *' LokeQv;XovMZaa? ph  w  ;  K^W
  U  
 r    A   e  n    
] \  mB  _ RU    . ($ [  9  + fD,'    iru  @  g 44 q  i^weqR1 O  ) mP!   N i 
Z xk   0
 /( fa  k C[  3D"   \8 " 1 y T n p J   MP) a#P |R   2r i+  vet,- 
n  v f9  ~ u    |g     cZ!G :  3  UKA  ezK    F    i UW حw 
 q  ;n  %  p   s,06 I,   b
Bk  su    Kw Fpo  uB          t  p!          )T      f    ; <   GA Y      j   L  
     =E  j!  y r  x#   A    } ? q * nm= L  
    b  T(u8|  (E
  \L>,B Y; Y 2 s E06a t U    Z    }  [*  ; mqjB k    "VX  
j/ + hf    M  -          [HgN        b          d,9 
 E     
d [    j  AAAY    P   

```

verbose info / delete

- **shred -u -v <fileName>**

```

unan@unan-VirtualBox:~/lecture36$ ls
dir2new  dir3  somefile2.txt  somefile.txt
unan@unan-VirtualBox:~/lecture36$ shred -u -v somefile.txt
shred: somefile.txt: pass 1/3 (random)...
shred: somefile.txt: pass 2/3 (random)...
shred: somefile.txt: pass 3/3 (random)...
shred: somefile.txt: removing
shred: somefile.txt: renamed to 00000000000000
shred: 00000000000000: renamed to 000000000000
shred: 00000000000000: renamed to 000000000000
shred: 000000000000: renamed to 000000000000
shred: 000000000000: renamed to 0000000000
shred: 00000000000: renamed to 0000000000
shred: 0000000000: renamed to 00000000
shred: 00000000: renamed to 000000
shred: 000000: renamed to 00000
shred: 00000: renamed to 0000
shred: 0000: renamed to 000
shred: 000: renamed to 00
shred: 00: renamed to 0
shred: somefile.txt: removed
unan@unan-VirtualBox:~/lecture36$ ls
dir2new  dir3  somefile2.txt
unan@unan-VirtualBox:~/lecture36$

```

last

```
unan@unan-VirtualBox:~/lecture36$ last
unan      :0                :0                Thu Nov 19 22:47    gone - no logout
reboot    system boot    5.4.0-42-generic Thu Nov 19 22:47    still running
unan      :0                :0                Thu Nov 19 20:13    - crash (02:34)
reboot    system boot    5.4.0-42-generic Thu Nov 19 20:12    still running
unan      :0                :0                Mon Oct 12 00:19    - crash (38+20:53)
reboot    system boot    5.4.0-42-generic Mon Oct 12 00:18    still running
unan      :0                :0                Mon Sep 14 16:52    - crash (27+07:26)
reboot    system boot    5.4.0-42-generic Mon Sep 14 16:51    still running
unan      :0                :0                Tue Jul 28 11:49    - crash (48+05:02)
reboot    system boot    5.4.0-42-generic Tue Jul 28 11:49    still running

wtmp begins Tue Jul 28 11:49:20 2020
```

ifconfig

- The command ifconfig stands for interface configurator. This command enables us to initialize an interface, assign IP address, enable or disable an interface. It display route and network interface.
- You can view IP address, MAC address and MTU (Maximum Transmission Unit) with ifconfig command.
- A newer version of ifconfig is ip command. ifconfig command works for all the versions.

Syntax:

ifconfig

ip

```
unan@unan-VirtualBox:~/lecture36$ ip
Usage: ip [ OPTIONS ] OBJECT { COMMAND | help }
       ip [ -force ] -batch filename
where  OBJECT := { link | address | addrlabel | route | rule | neigh | ntable |
                  tunnel | tuntap | maddress | mroute | mrule | monitor | xfrm |
                  |
                  netns | l2tp | fou | macsec | tcp_metrics | token | netconf |
                  ila |
                  vrf | sr | nexthop }
       OPTIONS := { -V[ersion] | -s[tatistics] | -d[etails] | -r[esolve] |
                   -h[uman-readable] | -i[ec] | -j[son] | -p[retty] |
                   -f[amily] { inet | inet6 | mpls | bridge | link } |
                   -4 | -6 | -I | -D | -M | -B | -O |
                   -l[oops] { maximum-addr-flush-attempts } | -br[ief] |
                   -o[neline] | -t[imestamp] | -ts[hort] | -b[atch] [filename] |
                   |
                   -rc[vbuf] [size] | -n[etns] name | -N[umeric] | -a[ll] |
                   -c[olor]}
```


Linux Certifications

- Most of the hiring manager are looking to recruit Linux professionals.
- The emergence of open cloud platforms is creating increasing demand for Linux professionals who have the right expertise
- Linux-certified professionals always be a better position in the job market
- Employers are looking for more Linux talent.
- Better salary increments for Linux certified professionals
- Some famous certificates
 - Red Hat Linux SuSE Linux
 - Linux Professional Institute (LPIC)
 - CompTIA
 - Linux Foundation
 - Oracle

Useful resources

- <https://www.guru99.com/best-linux-books-beginners.html>
- <https://ubuntu.com/tutorials?page=2>
- <https://ubuntu.com/tutorials/create-your-first-snap#1-overview>