Assignment No : 1

Install, Configure 64 bit Linux Operating System, study basic architecture, memory system & learn basic administration.

➢ Steps to Dual-boot Ubuntu 16.04, Windows 10 on a PC with UEFI firmware

Ubuntu 16.04 LTS code name '**Xenial Xerus**' has been released recently on 21st April 2016. As this release is under LTS(Long Term Support) so its Desktop support will be for next 5 years and Server support will be for next 3 years. Some of new improved features of Ubuntu 16.04 LTS are listed below :

- New Linux Kernel 4.4
- Snap – New application Package format
- Introduction of LXD – new Container hypervisor on Linux, In Ubuntu 16.04 LTS docker containers can run inside LXD.
- Latest version of Openstack Mitaka included in this release.
- Ubuntu 16.04 will support IBM Z and LinuxONE Servers
- Python 3
- PHP 7
- Gnome Desktop 3.18
- Chromium 48
- LibreOffice 5.1

In this is article i will demonstrate how Dual-boot Ubuntu 16.10, Windows 10 on a PC with UEFI firmware.

Step1 : Get Your Windows 10 Computer Ready

For a pain-free and successful operation of the sort that you're about to undertake, it is recommended that you disable Secure Boot (Restricted Boot). How this is done depends on your computer, but if you have the same Lenovo G50 laptop should consult this guide.

How to disable Secure Boot on a Lenovo G50 laptop

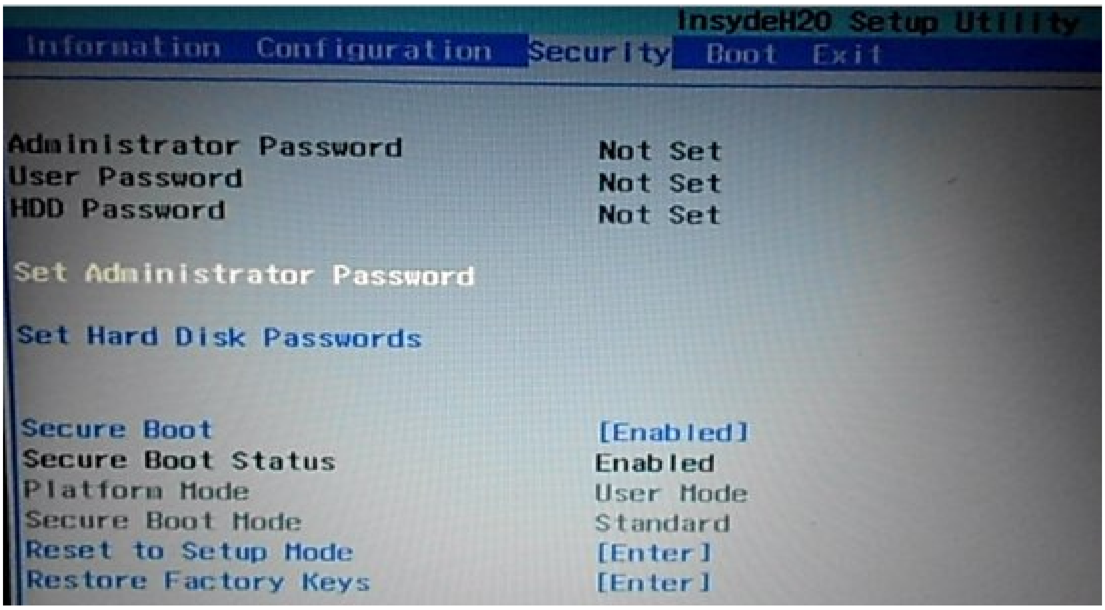
Like virtually all OEM computers that ship with Windows 8, it has Secure Boot enabled. The real terminology is Restricted Boot.

Leaving Restricted Boot enabled is fine if you intend to run just Windows 8, or the next version of Windows, only, or a Linux distribution that can tolerate Restricted Boot. However, if you intend to install a distribution like Ubuntu, Linux Mint alongside Windows 8, you'll have to disable Restricted Boot. That's because:

Ubuntu, Linux Mint does not use digital signatures and does not register to be certified by Microsoft as being a "secure" OS. As such, it will not boot with Secure Boot. If your system is using secure Boot, turn it off.

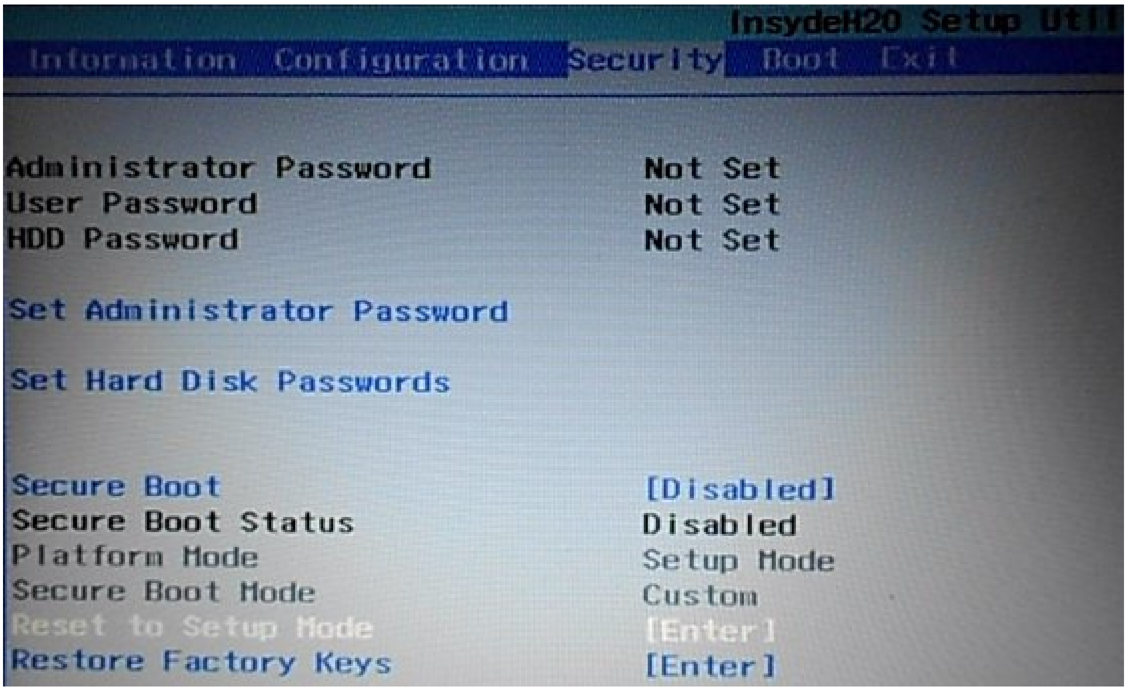
So here's how to turn off Restricted Boot on a Lenovo G50. The steps should be the same on other (Lenovo) computers, but perhaps with minor variations, if any. On a Lenovo G50, you access the **InsydeH20 Setup Utility** (BIOS/UEFI setup utility) by pressing the **Fn+F2** key just as the system

starts booting. You may also access it by pressing the **Novo** button on the left side of the unit, just beside the power connector, and selecting the appropriate option from the screen that opens.
Then scroll to the **Security** tab by using → . Then use ↓ to scroll to the Secure Boot entry. Press the **Enter** key and select **Disabled**. Follow that by navigating to the **Reset to Setup Mode** entry using the ↓ again. Press Enter, then select Yes on the popup to clear the PK, disable secure boot and enter setup mode. You'll notice that the **Platform Mode** changes to Setup Mode and **Secure Boot Mode** becomes Custom. Press the Fn+F10 key to reboot.



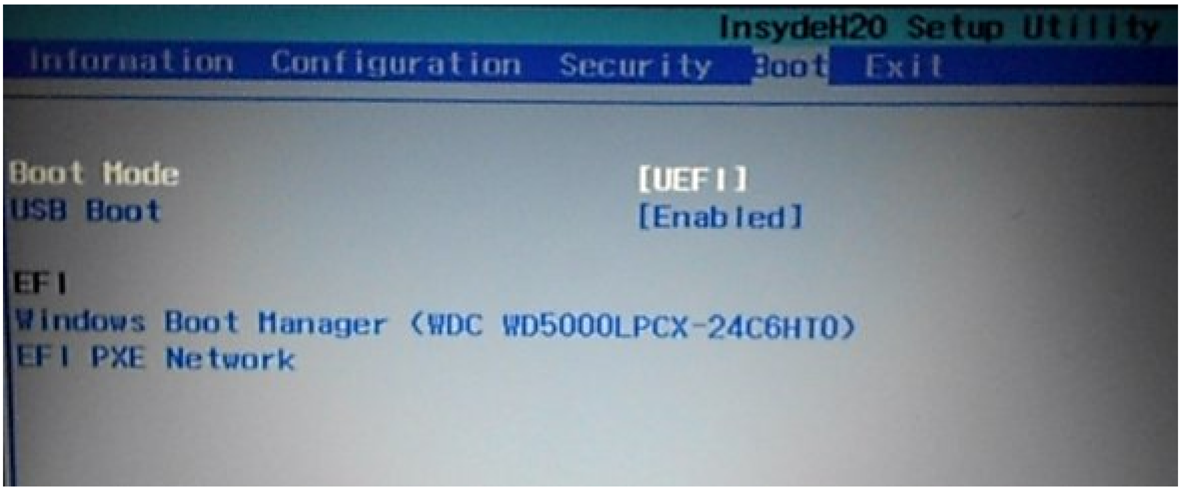
```
Administrator Password          Not Set
User Password                   Not Set
HDD Password                    Not Set

Set Administrator Password

Set Hard Disk Passwords

Secure Boot                     [Enabled]
Secure Boot Status              Enabled
Platform Mode                   User Mode
Secure Boot Mode                Standard
Reset to Setup Mode             [Enter]
Restore Factory Keys            [Enter]
```

When next you access the InsydeH20 Setup Utility > Security tab, you'll see that the **Secure Boot Status** will be disabled. And if you ever wish to restore the settings, access the Security tab again, navigate to the **Restore Factory Keys** entry, and press Enter. Restricted Boot will be enabled on the reboot after that, provided you save the settings using the Fn+F10 keys.



```
Administrator Password          Not Set
User Password                   Not Set
HDD Password                    Not Set

Set Administrator Password

Set Hard Disk Passwords

Secure Boot                     [Disabled]
Secure Boot Status              Disabled
Platform Mode                   Setup Mode
Secure Boot Mode                Custom
Reset to Setup Mode             [Enter]
Restore Factory Keys            [Enter]
```

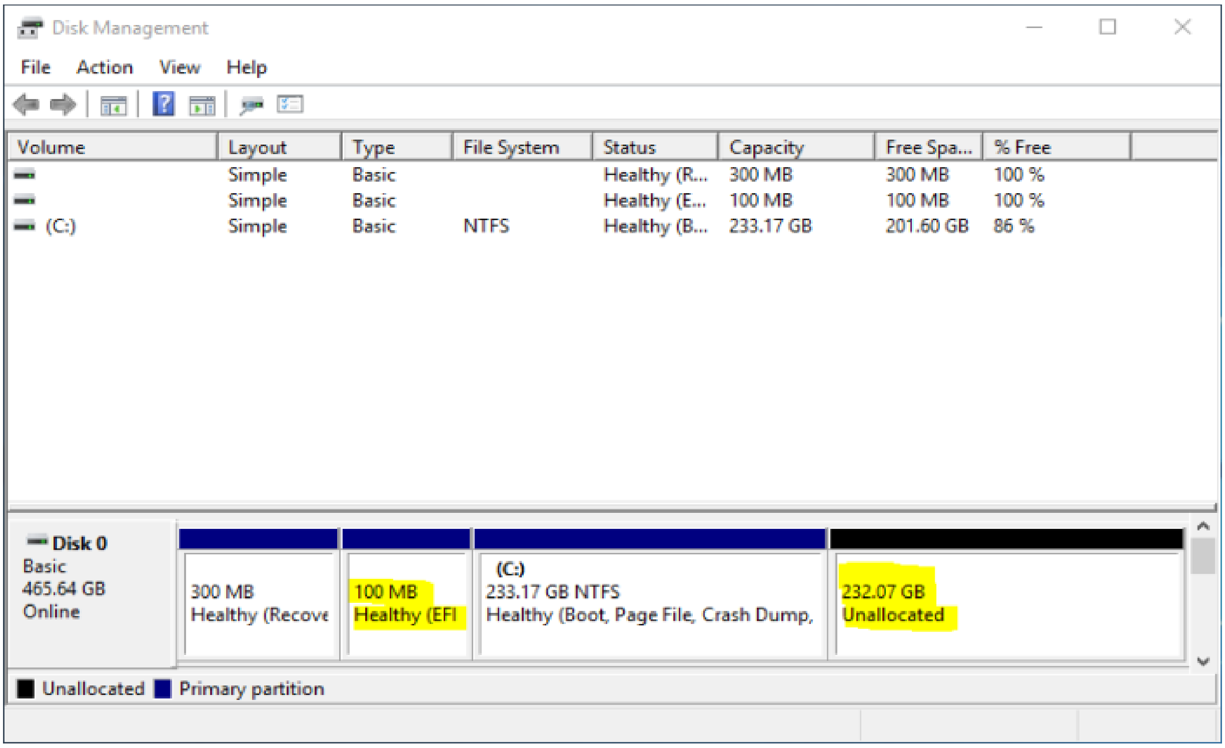
Although not necessary for using or not using Restricted Boot, a setting you can also enable/disable, is UEFI/Legacy BIOS mode. I don't see any reason to mess with this setting, but it's under the Boot tab if you ever wish to change it.



Boot Mode [UEFI]
USB Boot [Enabled]

EFI
Windows Boot Manager (WDC WD5000LPCX-24C6HT0)
EFI PXE Network

From the BIOS or UEFI setup utility, configure the computer to boot from external media.
After that, you need to free up disk space from your computer by shrinking the C drive or other partition with enough free space on it. Getting that done is fairly simple by using Windows 10 partition manager. When completed, the partition should show partitions and free disk space.

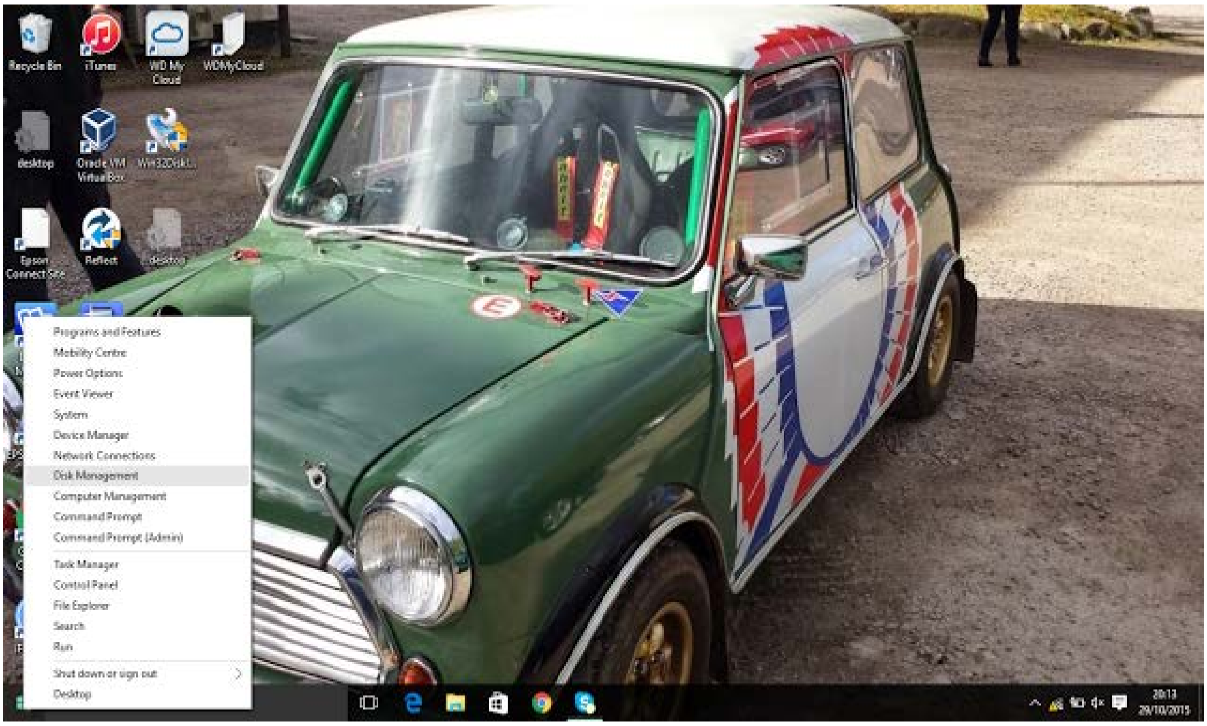


Volume	Layout	Type	File System	Status	Capacity	Free Spa...	% Free
	Simple	Basic		Healthy (R...	300 MB	300 MB	100 %
	Simple	Basic		Healthy (E...	100 MB	100 MB	100 %
(C:)	Simple	Basic	NTFS	Healthy (B...	233.17 GB	201.60 GB	86 %

Disk 0 Basic 465.64 GB Online	300 MB Healthy (Recove	100 MB Healthy (EFI	(C:) 233.17 GB NTFS Healthy (Boot, Page File, Crash Dump,	232.07 GB Unallocated

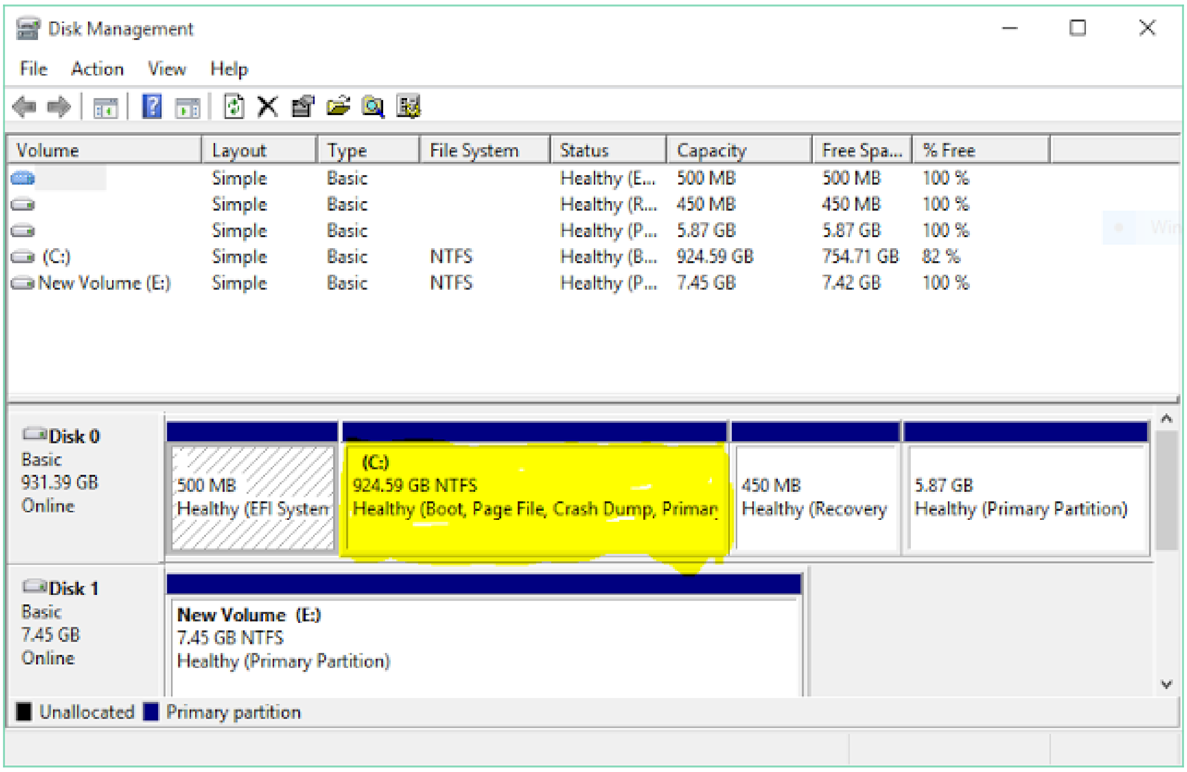
■ Unallocated ■ Primary partition

The tool used for shrinking Windows partitions is "Disk Management".
To start "Disk Management", right click on the Windows start button in the bottom left corner of the screen and click on the "Disk Management" option.



A screen will appear listing all the disks available to you and the partitions on the disk that is selected within the window.

The hard drive is usually disk 0. It is usually easy to spot because it is larger than the other drives and already has a number of partitions on it. As you can see from the image above there are 4 partitions. The amount of partitions on your hard drive may differ.

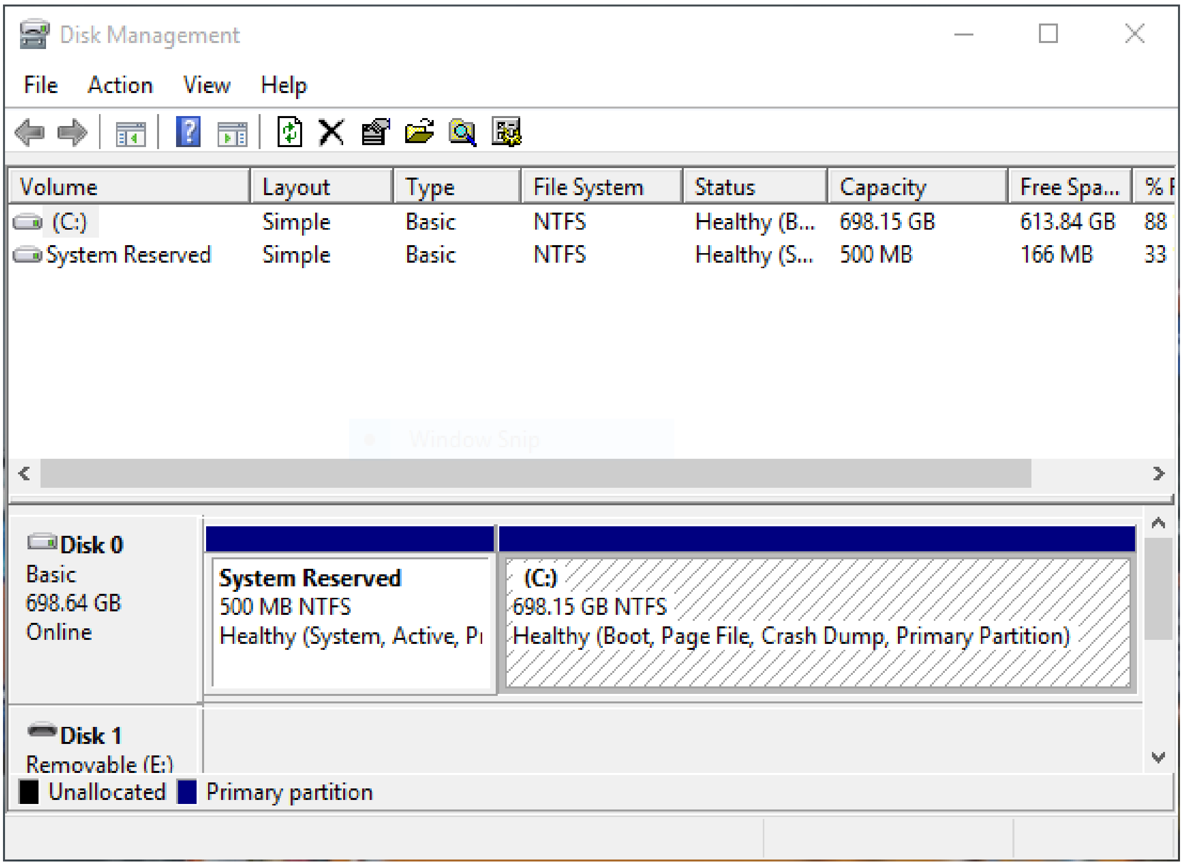


Volume	Layout	Type	File System	Status	Capacity	Free Spa...	% Free
	Simple	Basic		Healthy (E...	500 MB	500 MB	100 %
	Simple	Basic		Healthy (R...	450 MB	450 MB	100 %
	Simple	Basic		Healthy (P...	5.87 GB	5.87 GB	100 %
(C:)	Simple	Basic	NTFS	Healthy (B...	924.59 GB	754.71 GB	82 %
New Volume (E:)	Simple	Basic	NTFS	Healthy (P...	7.45 GB	7.42 GB	100 %

Disk	Partitions			
Disk 0 Basic 931.39 GB Online	500 MB Healthy (EFI System	(C:) 924.59 GB NTFS Healthy (Boot, Page File, Crash Dump, Primar	450 MB Healthy (Recovery	5.87 GB Healthy (Primary Partition)
Disk 1 Basic 7.45 GB Online	**New Volume (E:)** 7.45 GB NTFS Healthy (Primary Partition)			

■ Unallocated ■ Primary partition

For example the screenshot above shows the partitions on my computer which used to run Windows 8 (UEFI) and the screenshot below shows the partitions on my computer which used to run Windows 7 (BIOS).



Volume	Layout	Type	File System	Status	Capacity	Free Spa...	% F
(C:)	Simple	Basic	NTFS	Healthy (B...	698.15 GB	613.84 GB	88
System Reserved	Simple	Basic	NTFS	Healthy (S...	500 MB	166 MB	33

Disk 0
Basic
698.64 GB
Online

System Reserved
500 MB NTFS
Healthy (System, Active, Pı

(C:)
698.15 GB NTFS
Healthy (Boot, Page File, Crash Dump, Primary Partition)

Disk 1
Removable (E:)

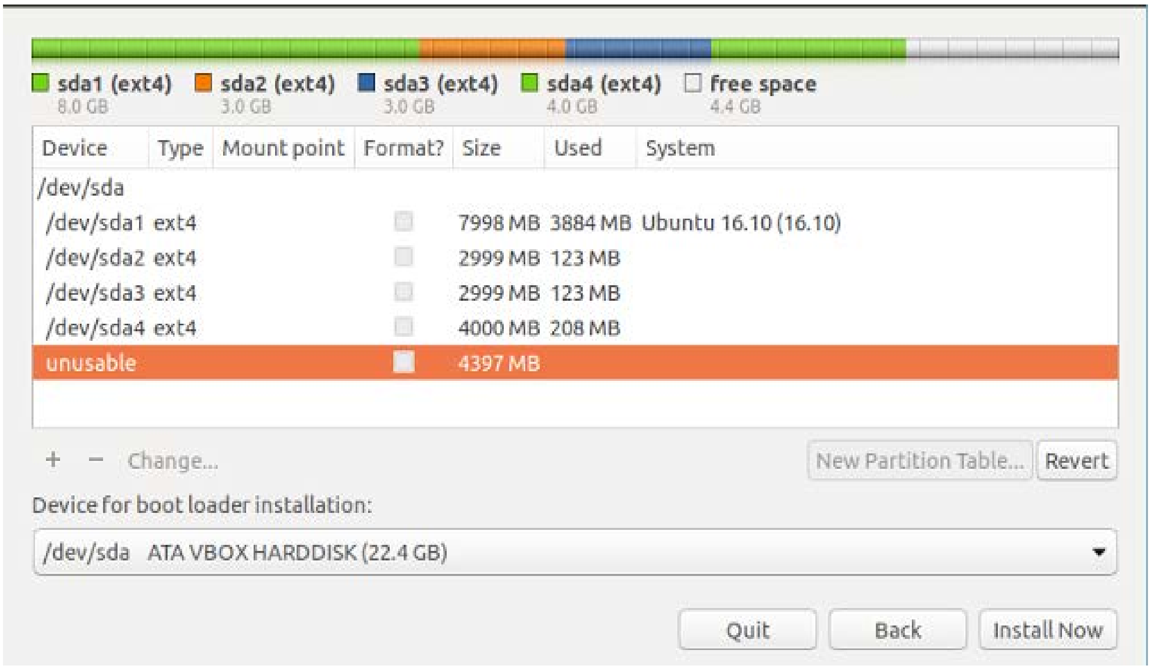
■ Unallocated ■ Primary partition

What you are looking for is the NTFS partition which is both active and primary. In most cases it is drive C. It is also likely to be the largest partition on the drive.

NOTE: If your computer does not have an EFI partition and there are 4 partitions on the drive which say primary then you should read this guide before continuing as you cannot have more that 4 primary partitions on a non-EFI drive.

If there is no EFI partition then you could have issues with partitioning when installing Ubuntu.

The older style BIOS only allows a user to create 4 primary partitions on a disk whereas the newer GPT partitioning system allows more than you will ever need.
The problem with having just 4 partitions available is that Windows 10 often takes up a number of partitions itself. There is one for Windows and at least one for a recovery partition. The manufacturer of the computer quite often has a partition for its own recovery partition and then another partition may have been created for another reason.
As there can only be 4 primary partitions if you have shrunk the Windows partition the free space you have created cannot be placed in a partition and therefore cannot be used.
When you try to install Ubuntu you will not see an option to install alongside Windows and when you choose something else as an option you will see unusable space as shown below.



Device	Type	Mount point	Format?	Size	Used	System
/dev/sda						
/dev/sda1	ext4		□	7998 MB	3884 MB	Ubuntu 16.10 (16.10)
/dev/sda2	ext4		□	2999 MB	123 MB	
/dev/sda3	ext4		□	2999 MB	123 MB	
/dev/sda4	ext4		□	4000 MB	208 MB	
unusable			□	4397 MB		

Device for boot loader installation:

/dev/sda ATA VBOX HARDDISK (22.4 GB)

The Solution

I can't give you a step by step solution to fix this as it depends entirely on what partitions are currently used on your system. I can however tell you that all is not lost.

Whilst you can only have 4 primary partitions you can split a single partition into a number of extended partitions. If you can free up one of the 4 primary partitions you can then create a number of logical/extended partitions on that single partition for installing Ubuntu.

How To Remove A Partition

Freeing up a single partition is the difficult bit.

If you have shrunk the Windows partition to free up space then you will want to delete the partition next to it (no, not the Windows one, probably the one to the right).

The issue is however, what are those partitions used for. If the partition is a Windows recovery partition then you could elect to move that partition to an external hard drive.

The partition might also be the recovery partition created by your computer's manufacturer. In this case you should find the software used by the manufacturer as this may allow you to back up the system to an external hard drive or USB drive which means you can delete the manufacturer's recovery partition and use it with the free space you created by shrinking Windows.

Another option is of course to use Macrium Reflect which I recommended as the backup tool as part of the process for dual booting Ubuntu and Windows 10. You can use Macrium to create recovery media on DVDs, USB drives and external hard drives. With the recovery media safely stored externally you can safely delete the Windows recovery and manufacturer's recovery media.

If you have another partition called data then you might wish to move the data from there onto the Windows partition or indeed another drive such as an external hard drive and delete that partition.

You can delete a partition within the disk management tool by right clicking on it and choosing "delete volume".

Windows recovery partitions cannot be deleted using the disk management tool because the partition will be protected. It may be the case that the manufacturer's partition or Windows recovery partition is quite large and therefore you don't need the free space created by shrinking Windows any more. You can give the disk space you gained from shrinking Windows back to

Windows by right clicking on the Windows partition in the disk management tool and choosing "Extend Volume".

Given the choice between removing the Windows recovery partition and the manufacturers recovery partition I recommend removing the manufacturer's partition. I would however make sure that I had a viable recovery option available via Macrium reflect or another such tool.

The upshot is that you want to get to a position where you have only 3 primary partitions and then a section of free space on your computer large enough to install Ubuntu.

This guide shows how to delete a protected partition.

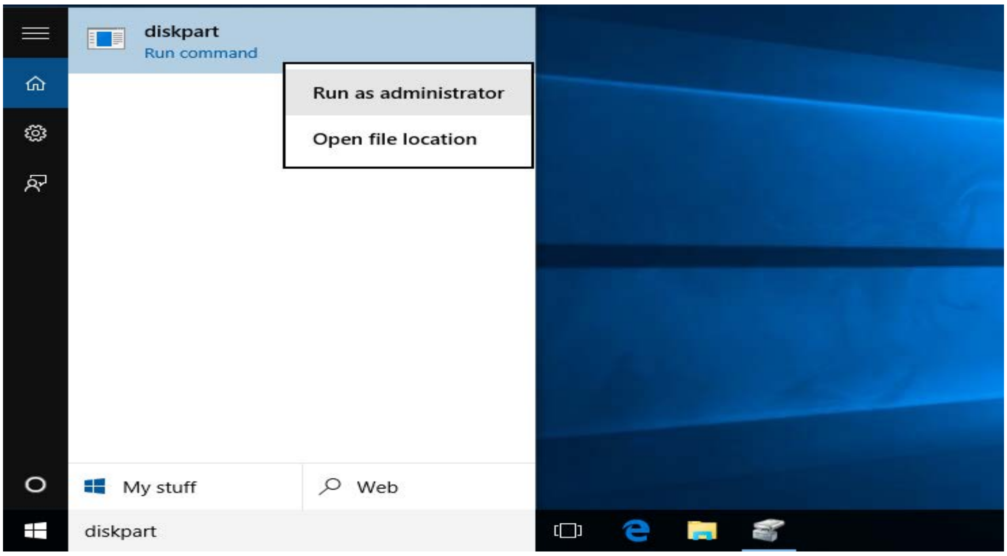
If you see a few gigabytes recovery partition in Disk Management, it is definitely included by computer manufacturers such as Dell, HP and Lenovo. This recovery partition contains an image of everything pre-installed along with operating system. On some computers, it is labelled as OEM reserved partition. You may also see a recovery partition with several gigabytes space and an OEM partition that is less than 100 MB.

The Windows recovery partition consumes much less storage space. For example, Windows 10 recovery partition only takes about 450MB, and even less in the Windows 8 or 7. Therefore, it is recommended to keep the Windows recovery partition. However, if you really need to get rid of the recovery partition, you can create Windows 10 recovery disk with a USB drive, and then Windows 10 will give you an option to delete the current recovery partition for Windows 10.

Remove recovery partition with Diskpart

As you know, you are not allowed to remove recovery partition in Disk Management, but you use Diskpart.exe to delete partitions from command prompt. You can use the following steps to remove recovery partition with diskpart:

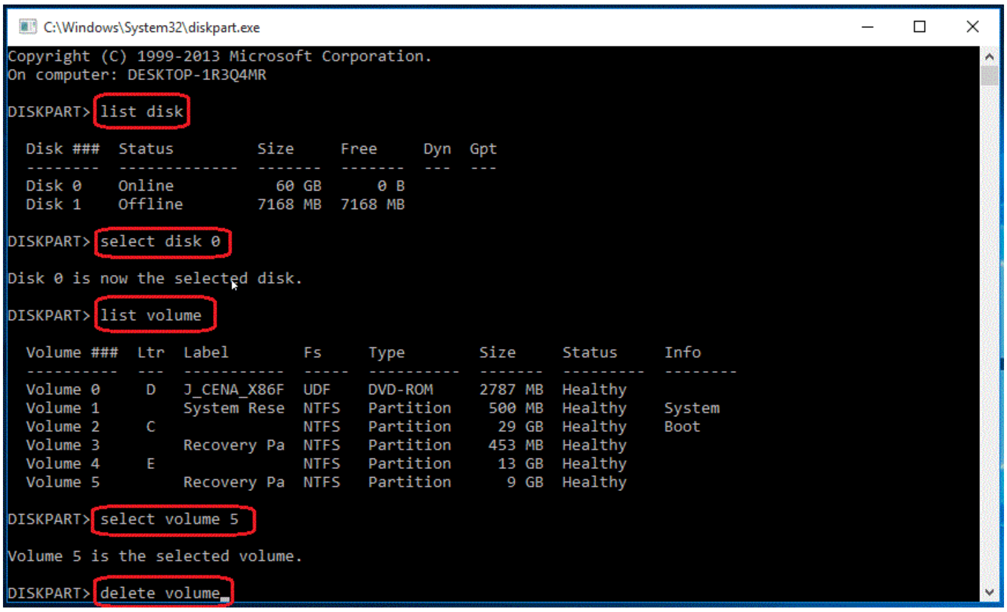
1. Type "diskpart.exe" in the search box, in the listed results, right-click on Diskpart and select Run as administrator.



2. At a Diskpart prompt, type in "list disk" to display all the disks. Each disk on your computer will be listed with a disk number.

3. Specify the disk that contains the recovery partition by typing in "select disk n", in which "n" should be replaced with the disk number listed before.

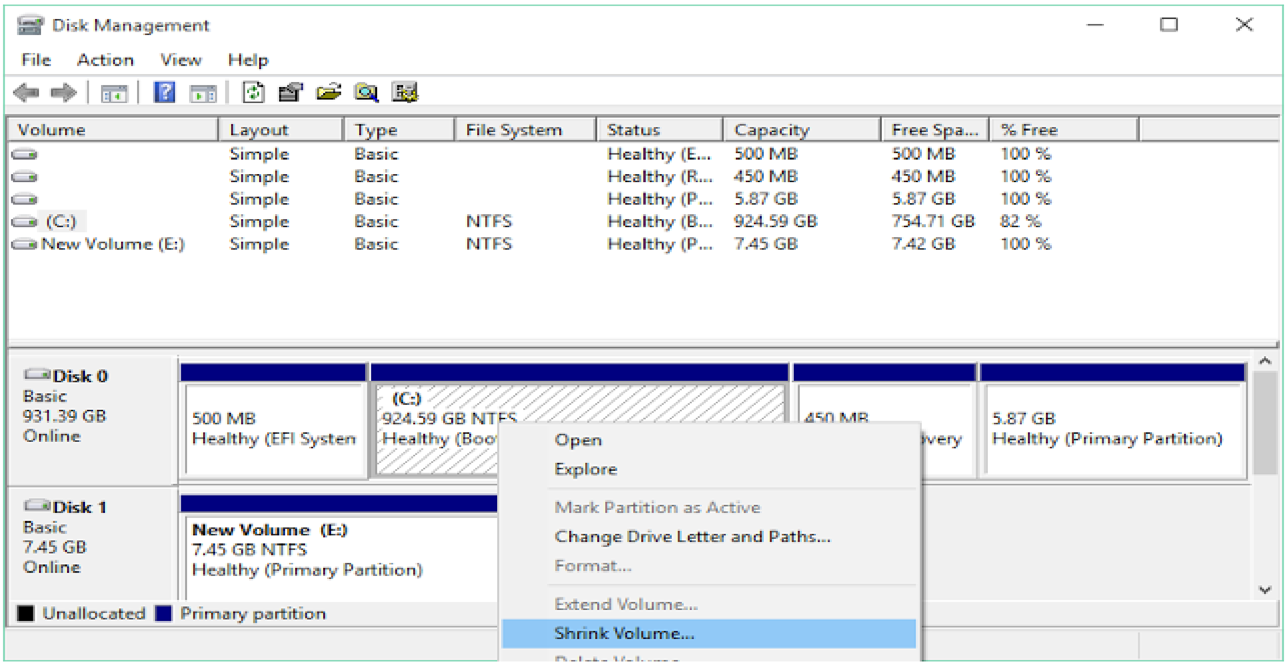
4. Type in "list volume" to list all the partitions on the selected disk with all the partition labels, so you can determine which one is the recovery partition to be deleted. If you want to see the partition type, use "list partition" command instead.
5. Specify the partition you want to delete using the volume number, like this: "select volume 3".
6. Type in: "delete volume".



```
Copyright (C) 1999-2013 Microsoft Corporation.
On computer: DESKTOP-1R3Q4MR

DISKPART> list disk

  Disk ###  Status         Size     Free     Dyn  Gpt
  --------  -------------  -------  -------  ---  ---
  Disk 0    Online           60 GB      0 B
  Disk 1    Offline        7168 MB  7168 MB

DISKPART> select disk 0

Disk 0 is now the selected disk.

DISKPART> list volume

  Volume ###  Ltr  Label        Fs     Type        Size     Status     Info
  ----------  ---  -----------  -----  ----------  -------  ---------  --------
  Volume 0     D   J_CENA_X86F  UDF    DVD-ROM     2787 MB  Healthy
  Volume 1         System Rese  NTFS   Partition    500 MB  Healthy    System
  Volume 2     C                NTFS   Partition     29 GB  Healthy    Boot
  Volume 3         Recovery Pa  NTFS   Partition    453 MB  Healthy
  Volume 4     E                NTFS   Partition     13 GB  Healthy
  Volume 5         Recovery Pa  NTFS   Partition      9 GB  Healthy

DISKPART> select volume 5

Volume 5 is the selected volume.

DISKPART> delete volume
```

Without a clear interface to show you display the details of each partition, there are chances that you may delete inappropriate partitions. What’s worse, this operation cannot be cancelled or undo. For those who are not familiar with Diskpart, the best choice to get rid of recovery partition should be using third party free partition manager software.

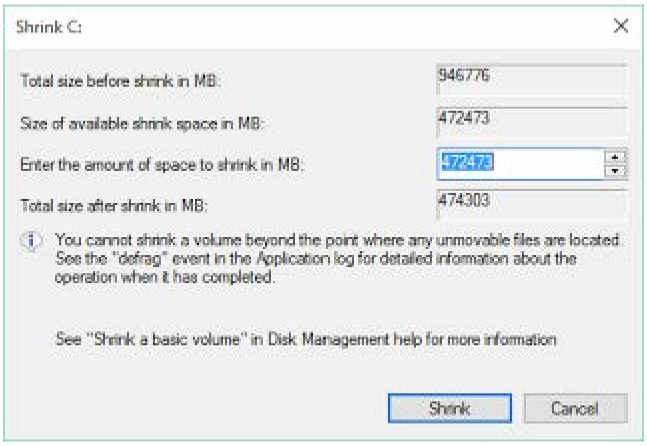
If there is EFI partition then you could follow following steps to shrink partition.



Volume	Layout	Type	File System	Status	Capacity	Free Spa...	% Free
	Simple	Basic		Healthy (E...	500 MB	500 MB	100 %
	Simple	Basic		Healthy (R...	450 MB	450 MB	100 %
	Simple	Basic		Healthy (P...	5.87 GB	5.87 GB	100 %
(C:)	Simple	Basic	NTFS	Healthy (B...	924.59 GB	754.71 GB	82 %
New Volume (E:)	Simple	Basic	NTFS	Healthy (P...	7.45 GB	7.42 GB	100 %

Right click on the partition which contains Windows (as described above, probably the C drive, NTFS file format, primary and active).

Click on the “shrink volume” option.



Shrink C:

Total size before shrink in MB: 946776

Size of available shrink space in MB: 472473

Enter the amount of space to shrink in MB: 472473

Total size after shrink in MB: 474303

You cannot shrink a volume beyond the point where any unmovable files are located. See the "defrag" event in the Application log for detailed information about the operation when it has completed.

See "Shrink a basic volume" in Disk Management help for more information

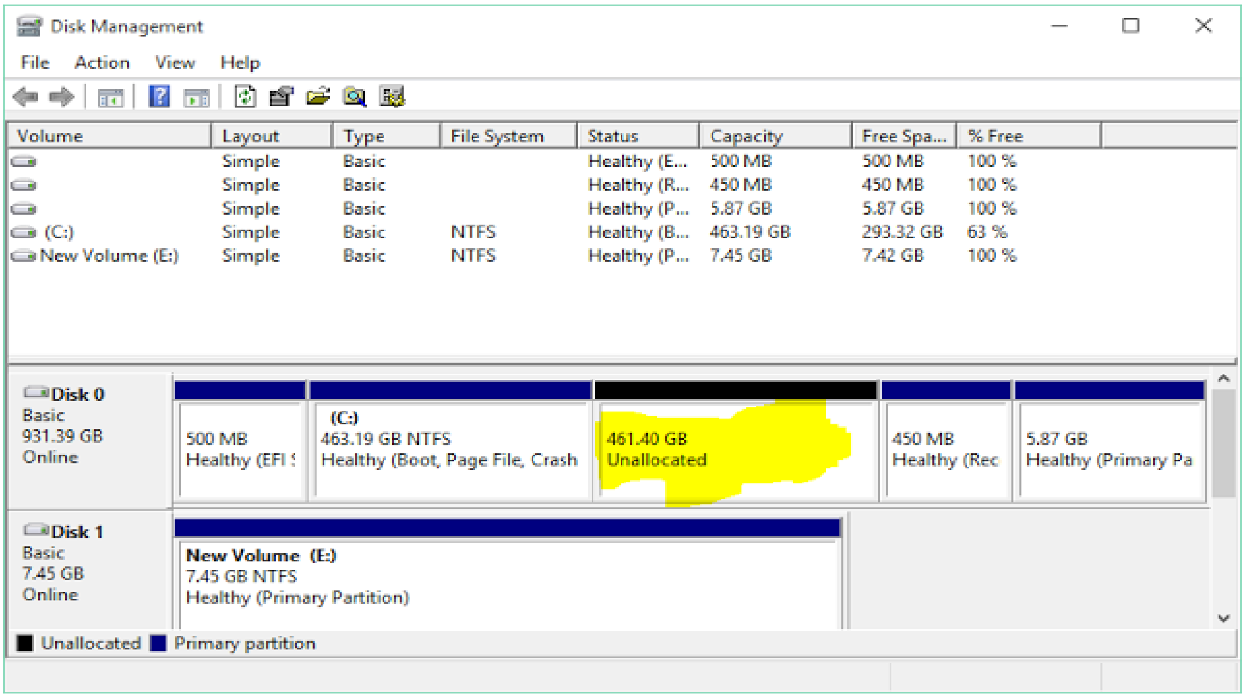
When the shrink dialogue box appears it will have the maximum amount of space available to shrink listed with an up and down arrow to increase or decrease the amount of space made available to Ubuntu.

Do not make the number if the “Enter the amount of space to shrink in MB” (box 3) larger than the “Size of available shrink space in MB” (box 2). This is the equivalent to ghostbusters crossing the streams.

You can make the amount of space available to Ubuntu smaller though. The minimum required by Ubuntu is 7 gigabytes which is 7000 megabytes. In reality you should be looking to give at least 20 gigabytes (20000 megabytes) to Ubuntu for a decent experience and if you have it I would go for 50 gigabytes (50000 megabytes).

Of course if you so wish you can leave the recommended amounts as they are.

When you have decided on an amount enter it into the box and click “Shrink”.



Volume	Layout	Type	File System	Status	Capacity	Free Spa...	% Free
	Simple	Basic		Healthy (E...	500 MB	500 MB	100 %
	Simple	Basic		Healthy (R...	450 MB	450 MB	100 %
	Simple	Basic		Healthy (P...	5.87 GB	5.87 GB	100 %
(C:)	Simple	Basic	NTFS	Healthy (B...	463.19 GB	293.32 GB	63 %
New Volume (E:)	Simple	Basic	NTFS	Healthy (P...	7.45 GB	7.42 GB	100 %

Disk 0
Basic
931.39 GB
Online

500 MB
Healthy (EFI S

(C:)
463.19 GB NTFS
Healthy (Boot, Page File, Crash

461.40 GB
Unallocated

450 MB
Healthy (Rec

5.87 GB
Healthy (Primary Pa

Disk 1
Basic
7.45 GB
Online

New Volume (E:)
7.45 GB NTFS
Healthy (Primary Partition)

■ Unallocated ■ Primary partition

When the process has finished you should see that the C drive has become much smaller (or smaller by the amount you entered in the shrink dialogue box and that a new unallocated portion of disk space has appeared.

You now have space to fit at least one Linux distribution alongside Windows 10.

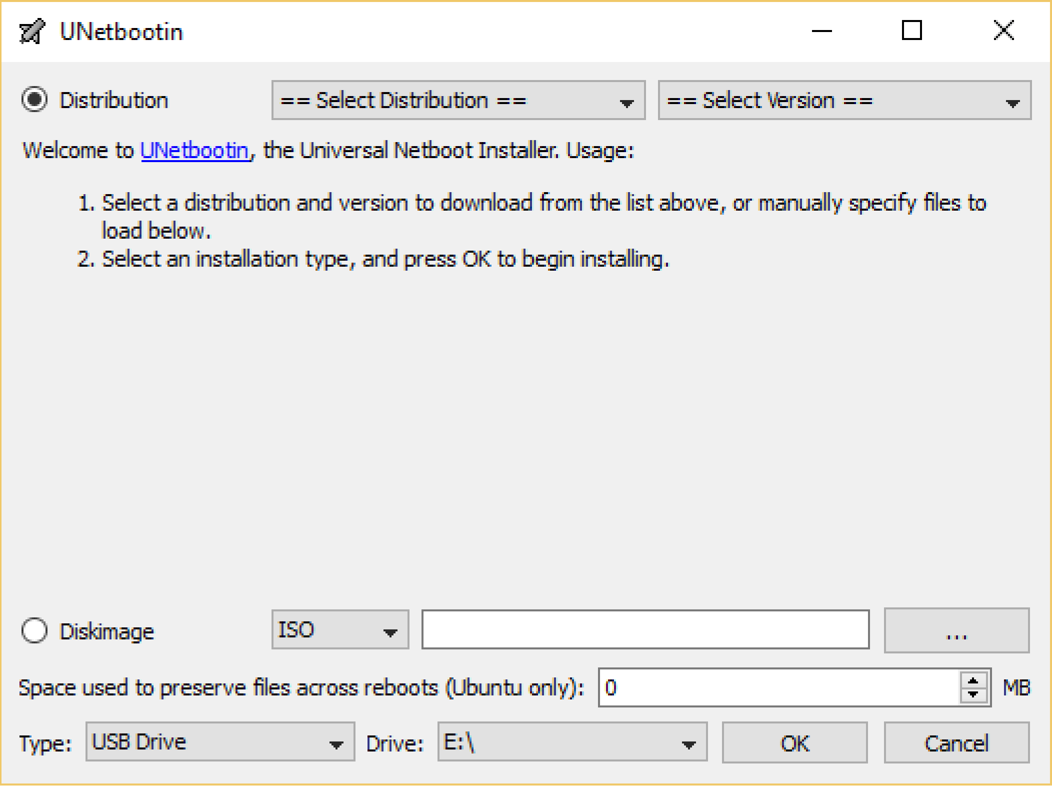
Step2 : Download Ubuntu 16.04 and Create a Bootable USB Stick

The next step in this operation is to download an installation image of Ubuntu 16.10 desktop, then transfer it to a USB stick.

Create bootable USB stick of Ubuntu 16.04

To use Unetbootin, you, of course, need a USB stick inserted into one of the computer's USB port. Unetbootin can download the distribution you want to use, but for this post, I downloaded an image of Ubuntu 16.04 before starting Unetbootin.

Figure shows the main interface. By default, it's set to auto-download a distribution from the Internet, but if you've downloaded one already, click on the **Diskimage** radio button.



UNetbootin

● Distribution == Select Distribution == == Select Version ==

Welcome to UNetbootin, the Universal Netboot Installer. Usage:

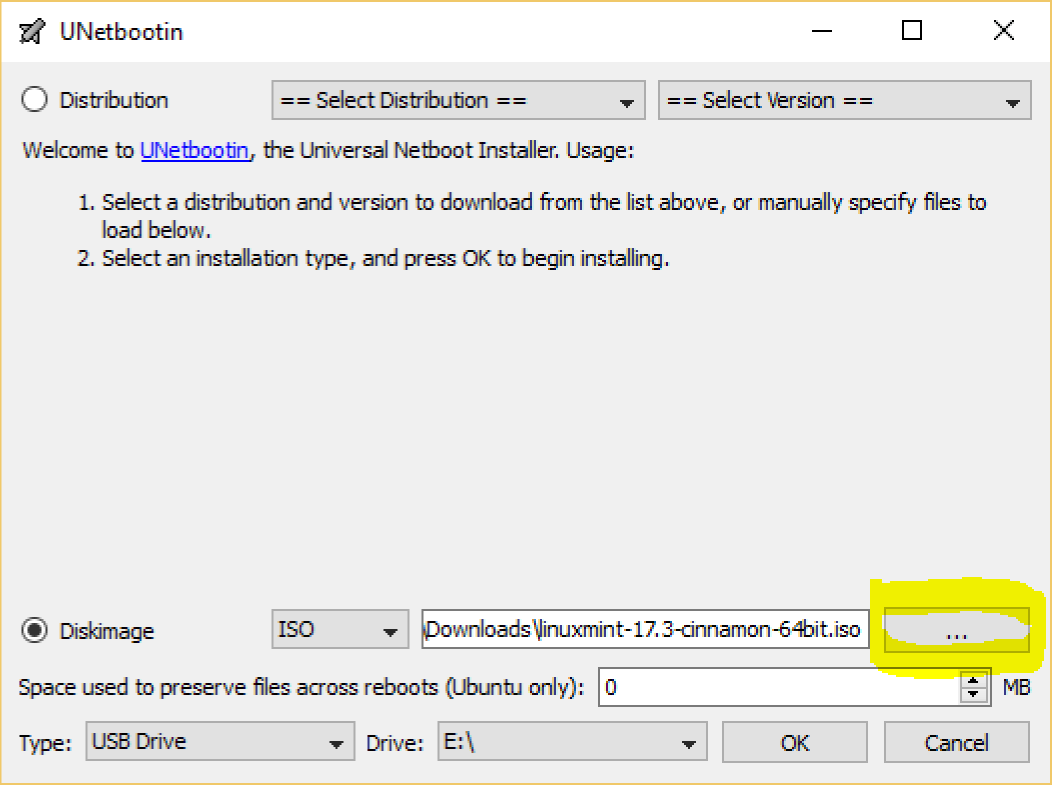
1. Select a distribution and version to download from the list above, or manually specify files to load below.
2. Select an installation type, and press OK to begin installing.

○ Diskimage ISO

Space used to preserve files across reboots (Ubuntu only): 0 MB

Type: USB Drive Drive: E:\

Then use the highlighted button to select the ISO image you downloaded. In the **Type** drop menu, select USB, then the USB drive letter, if it's not auto-selected or if you have more than one USB stick connected to the computer.



○ Distribution == Select Distribution == == Select Version ==

Welcome to UNetbootin, the Universal Netboot Installer. Usage:

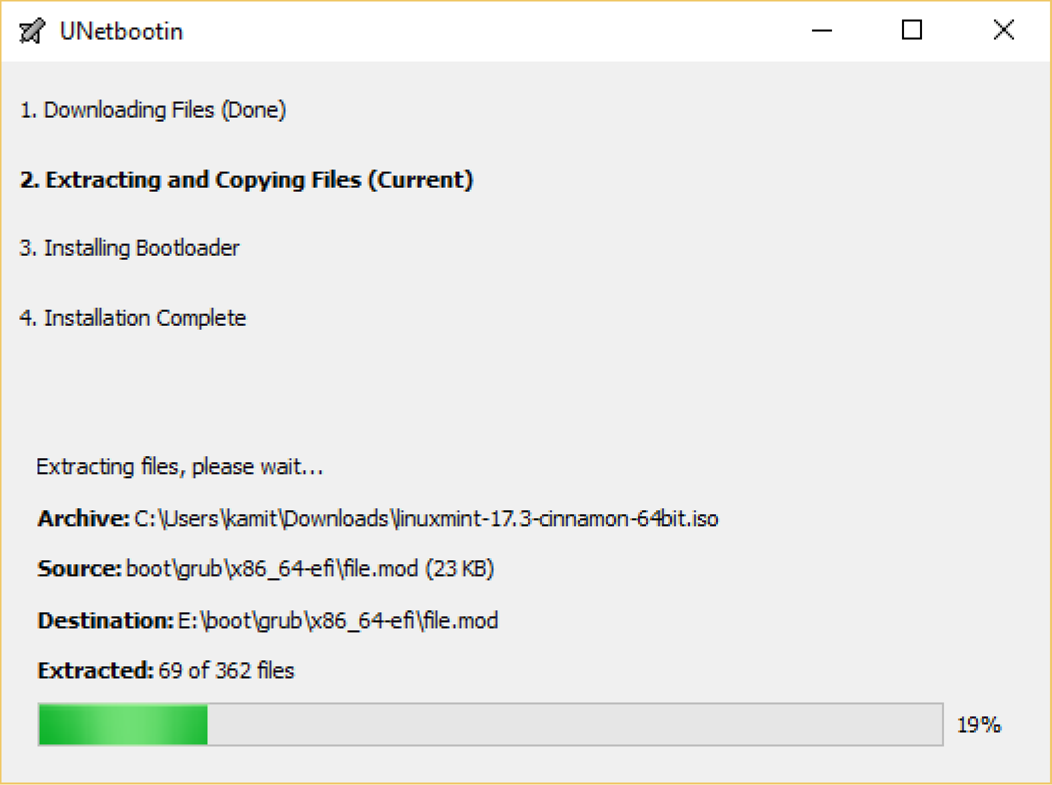
1. Select a distribution and version to download from the list above, or manually specify files to load below.
2. Select an installation type, and press OK to begin installing.

● Diskimage ISO Downloads\linuxmint-17.3-cinnamon-64bit.iso ...

Space used to preserve files across reboots (Ubuntu only): 0 MB

Type: USB Drive Drive: E:\

Wait awhile as Unetbootin does its magic.



1. Downloading Files (Done)

2. Extracting and Copying Files (Current)

3. Installing Bootloader

4. Installation Complete

Extracting files, please wait...

Archive: C:\Users\kamit\Downloads\linuxmint-17.3-cinnamon-64bit.iso

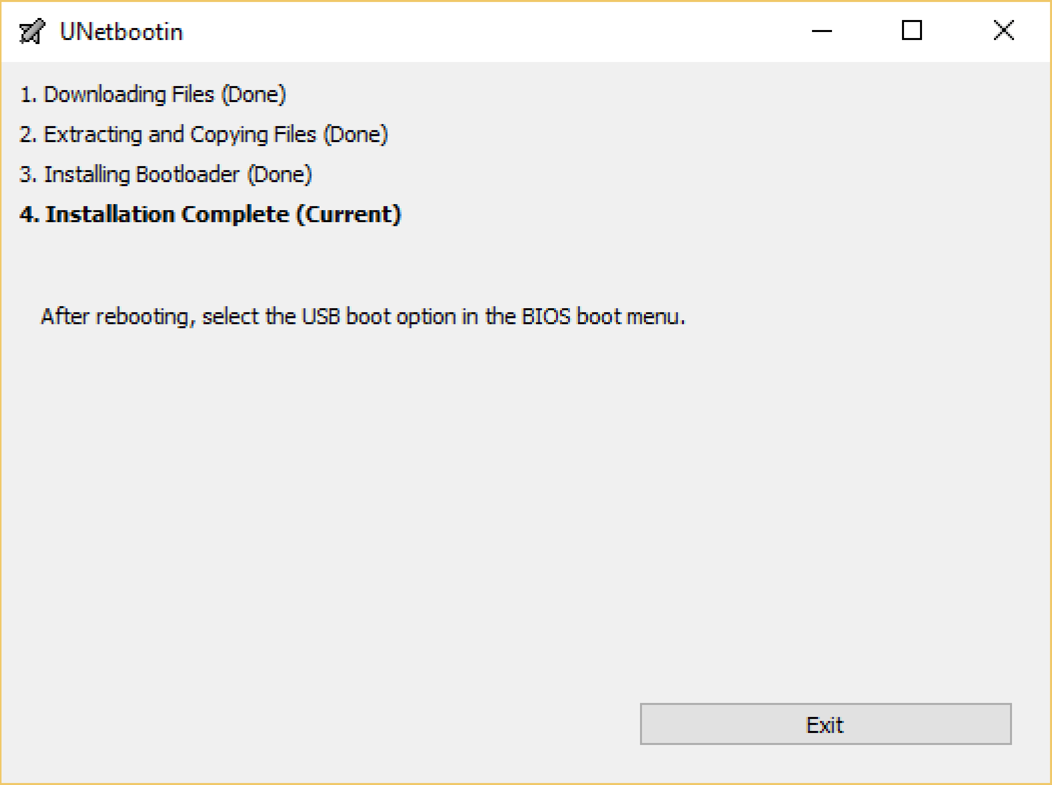
Source: boot\grub\x86_64-efi\file.mod (23 KB)

Destination: E:\boot\grub\x86_64-efi\file.mod

Extracted: 69 of 362 files

19%

When the transfer has completed successfully, exit the application. With the USB stick still connected to the computer, reboot. If the computer has been set up to boot from external media, it should reboot into the boot menu of Ubuntu 16.04.



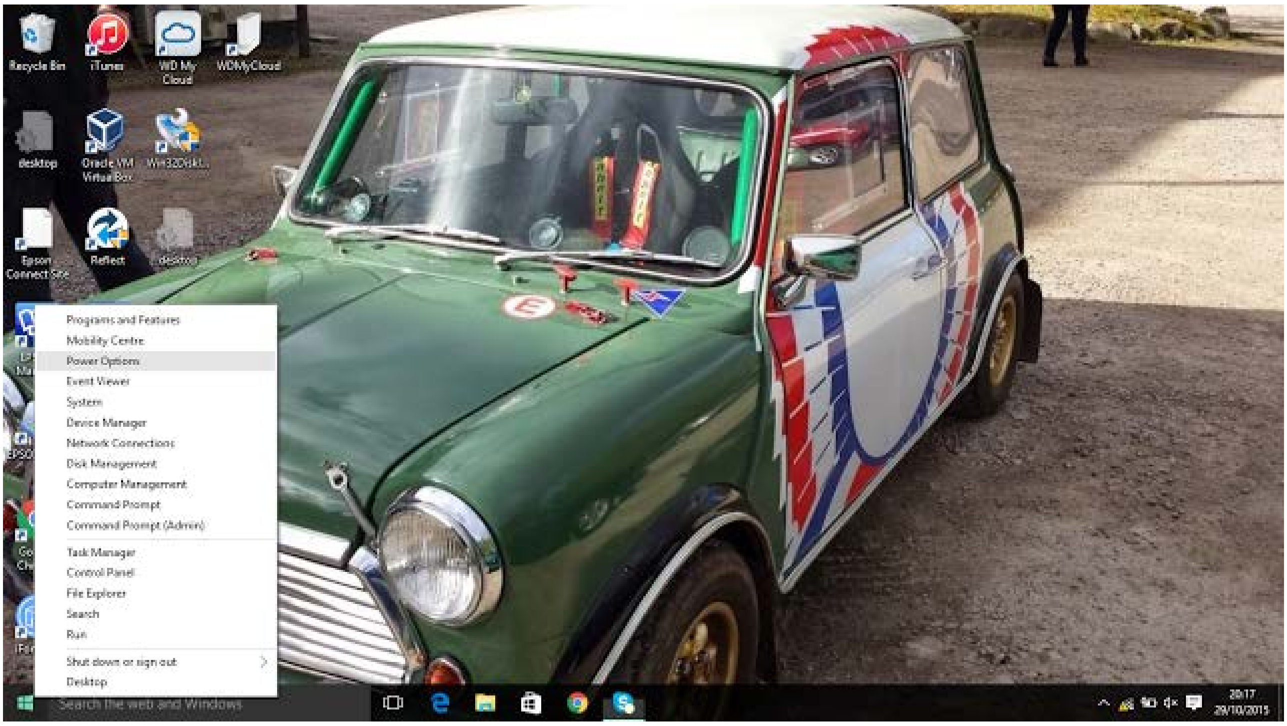
UNetbootin

1. Downloading Files (Done)
2. Extracting and Copying Files (Done)
3. Installing Bootloader (Done)
4. **Installation Complete (Current)**

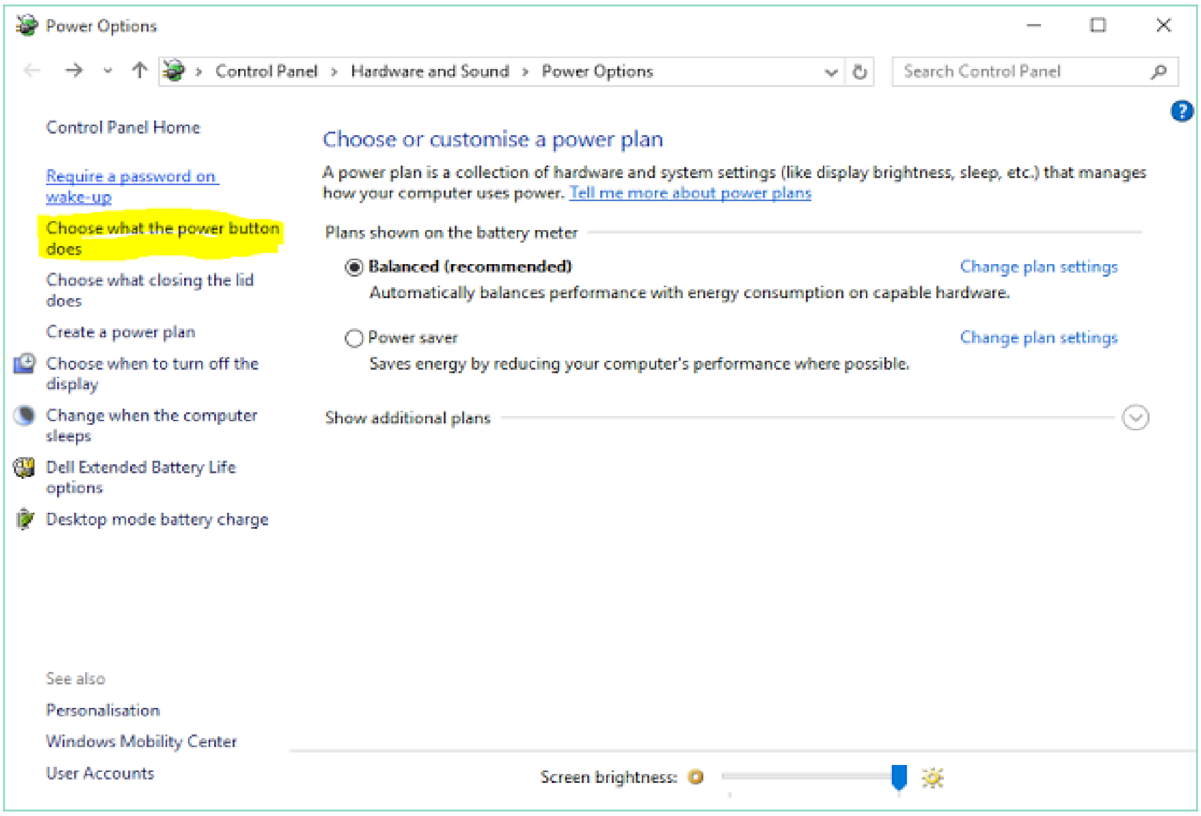
After rebooting, select the USB boot option in the BIOS boot menu.

When you've finished creating the installation media, insert it into the appropriate port in your computer and reboot. In the next step, you'll use that bootable USB stick to install Ubuntu 16.04 alongside Windows 10.

Turn Off Fast Boot (You only have to do this on computers with a UEFI bootloader.)
In order to speed up the boot time, Windows prevents booting from a USB until it has fully loaded.
This obviously prevents Ubuntu booting from the USB drive which is of course a major issue.
You can turn off the fast boot option which resolves the problem.



Right click on the start menu in the bottom left corner and when the menu appears click on “Power Options”.


Power Options
Control Panel > Hardware and Sound > Power Options
Search Control Panel
Control Panel Home
Require a password on wake-up
Choose what the power button does
Choose what closing the lid does
Create a power plan
Choose when to turn off the display
Change when the computer sleeps
Dell Extended Battery Life options
Desktop mode battery charge
See also
Personalisation
Windows Mobility Center
User Accounts

Choose or customise a power plan

A power plan is a collection of hardware and system settings (like display brightness, sleep, etc.) that manages how your computer uses power. Tell me more about power plans

Plans shown on the battery meter

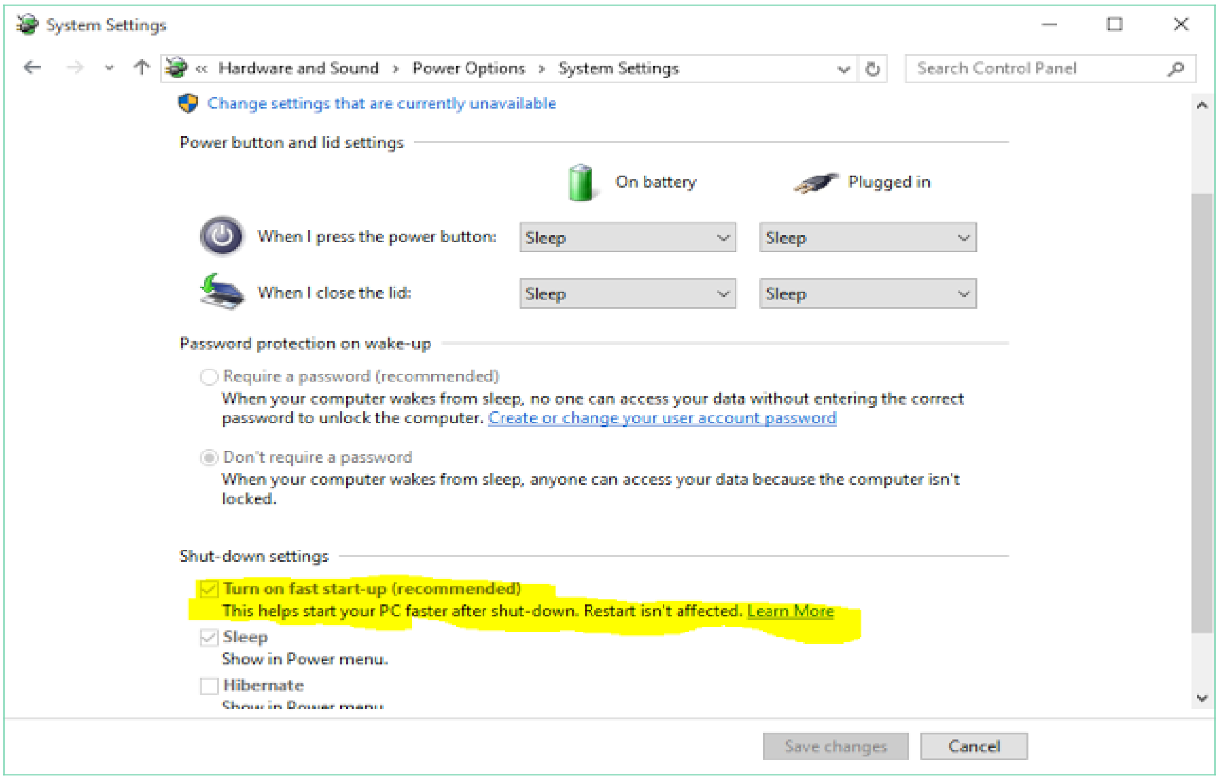
● Balanced (recommended) — Change plan settings
Automatically balances performance with energy consumption on capable hardware.

○ Power saver — Change plan settings
Saves energy by reducing your computer's performance where possible.

Show additional plans

Screen brightness:

There are lots of power options within Windows 10.
The left hand side of the screen has a list of categories. Click on the option “Choose what the power button does” as shown in the image above.



Change settings that are currently unavailable

Power button and lid settings

	On battery	Plugged in
When I press the power button:	Sleep	Sleep
When I close the lid:	Sleep	Sleep

Password protection on wake-up

○ Require a password (recommended)
When your computer wakes from sleep, no one can access your data without entering the correct password to unlock the computer. Create or change your user account password

● Don't require a password
When your computer wakes from sleep, anyone can access your data because the computer isn't locked.

Shut-down settings

☑ Turn on fast start-up (recommended)
This helps start your PC faster after shut-down. Restart isn't affected. Learn More

☑ Sleep
Show in Power menu.

□ Hibernate
Show in Power menu

Click the link which says "Change settings that are currently unavailable".

Scroll down the page until you see the option “Turn on fast start-up (recommended)” and turn it off by unchecking the box.

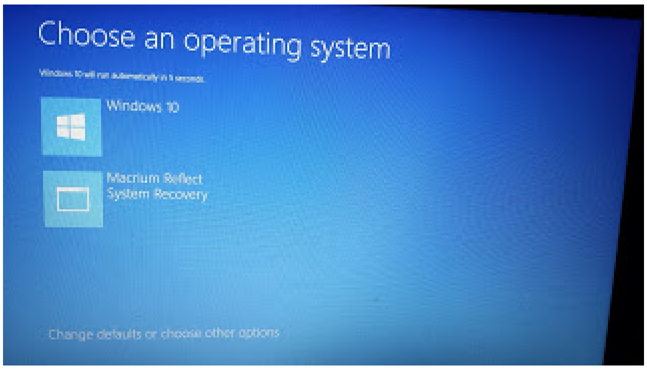
Click “Save Changes”.

Now you might be wondering why I am recommending turning off a recommended option.As previously mentioned you can’t boot from a USB drive whilst this option is checked. In addition to this I feel that the option is a bit cloak and dagger. What it says is that if you have the option checked your computer will load faster.

The trouble is that once your computer has finished booting it will still go and do all of the things that it missed out on whilst booting to make it load faster but it will do it quietly in the background. This will use up resources until it has completed. In reality your start screen might appear more quickly but it hasn’t really booted and quicker.

How To Boot Into Live Ubuntu USB Environment (non-UEFI)
Make sure that the Ubuntu USB drive that your created previously is plugged into the computer.
If your computer has a standard BIOS simply reboot your computer and a menu should appear with an option to try Ubuntu.
A large dialogue window will appear with options to install Ubuntu and to Try Ubuntu. Click on the “Try Ubuntu” option.
Ubuntu will now be loaded as a live session. You can try out all of the features of Ubuntu but if you reboot all the changes will be lost.

How To Boot Into Live Ubuntu USB Environment (UEFI)
If your computer has EFI then read on.



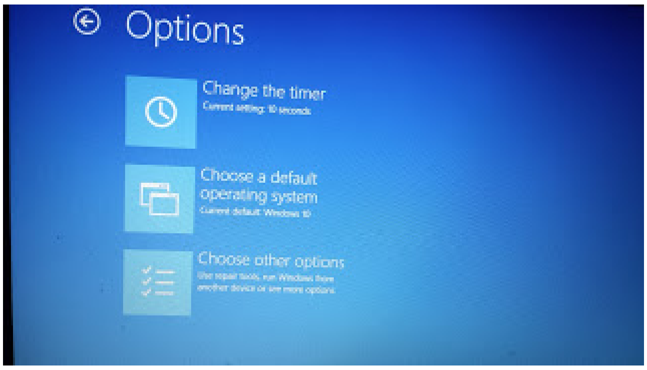
Choose an operating system

Windows 10 will run automatically in 5 seconds.

Windows 10
Macrium Reflect System Recovery
Change defaults or choose other options

When that screen appears click on the “Change defaults or choose other options” link at the bottom of the screen.

If you chose not to create the Macrium boot menu option hold down the shift key and reboot your computer. (Keep the shift key held down until a screen similar to the one below appears).



Options

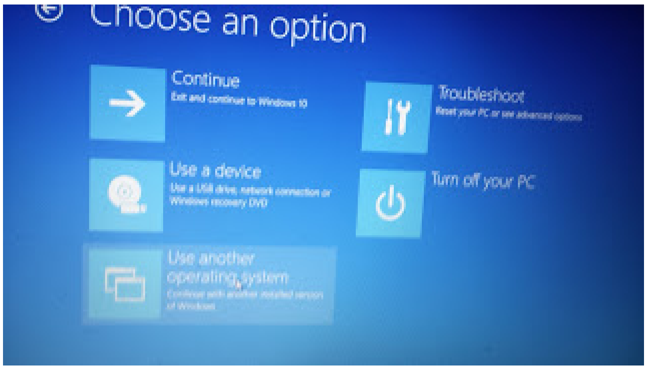
Change the timer
Current setting: 10 seconds

Choose a default operating system
Current default: Windows 10

Choose other options
Use repair tools, run Windows from another device or see more options

Each manufacturer has a different version of UEFI and so the menu options may be different. The important thing is that a blue screen with white writing appears.

You are basically looking for the option to boot from the USB drive and this may take some finding. From the image above I chose the “Choose other options” menu item which produced the screen below.



Choose an option

Continue
Exit and continue to Windows 10

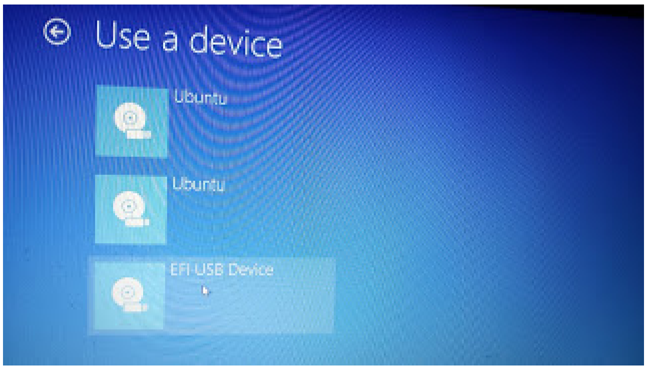
Troubleshoot
Reset your PC or see advanced options

Use a device
Use a USB drive, network connection or Windows recovery DVD

Turn off your PC

Use another operating system
Continue with another installed version of Windows

I then clicked on the “Use a device” option which as you can see has the subtext “Use a USB drive, network connection or Windows recovery DVD”.



Use a device

Ubuntu

Ubuntu

EFI USB Device

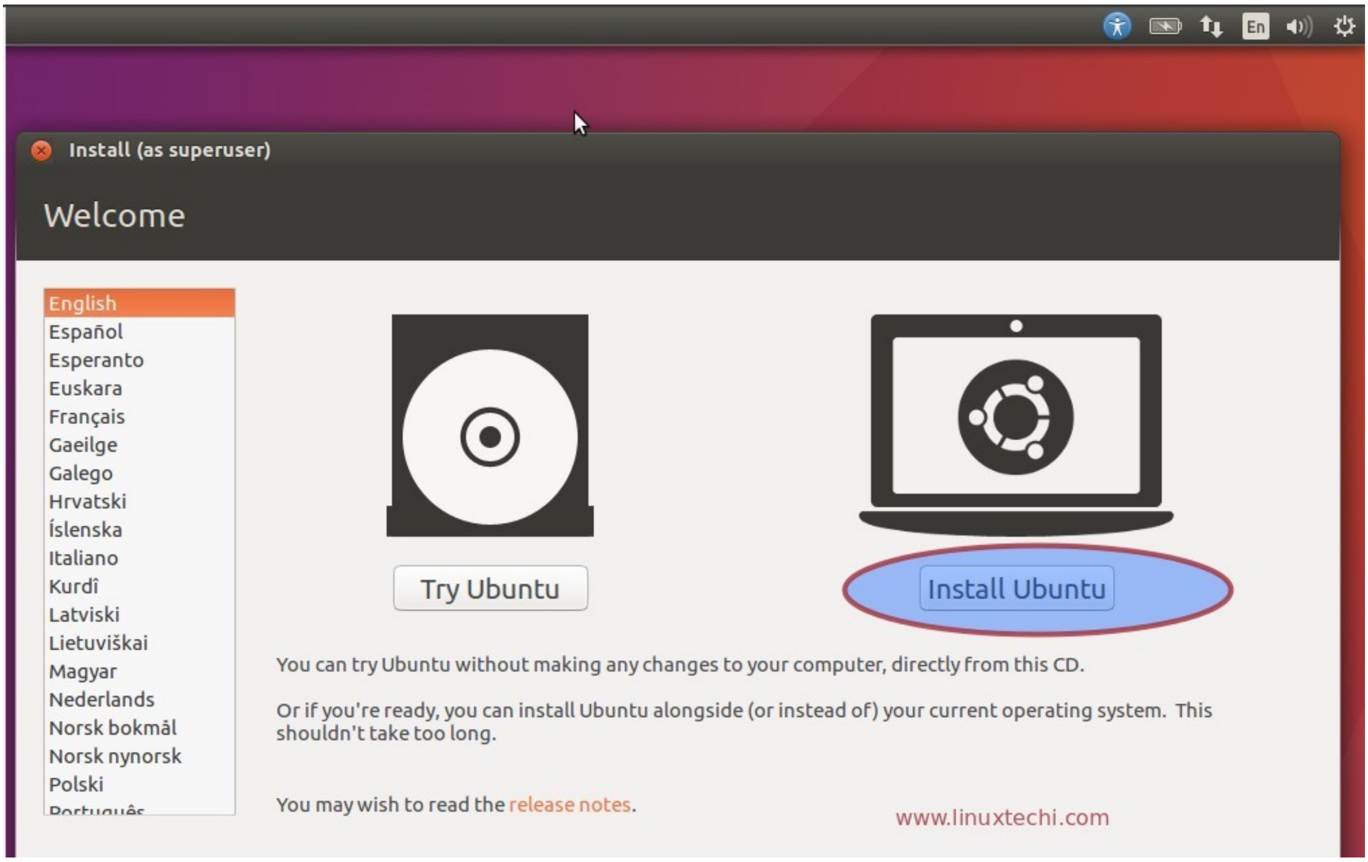
A list of devices will now appear. This isn’t the first time I have installed things on this computer and my EFI partition still has links to old Ubuntu versions. The important link on this screen is the “EFI USB Device” option.

Choose the EFI USB Device option.
An option should appear with an option to try Ubuntu.
Ubuntu should now boot from the USB drive.

A boot menu will appear. Choose the first menu option to try Ubuntu.
A large dialogue window will appear with options to install Ubuntu or to Try Ubuntu. Click on the "Try Ubuntu" option.
Ubuntu will now be loaded as a live session. You can try out all of the features of Ubuntu but if you reboot all the changes will be lost.

Step3 : Select 'Install Ubuntu' to start installation.

If the computer has been set up to boot from external media, you should see the Ubuntu 16.04 boot menu. One of the entries will be to boot into the Live desktop. I suggest you select that option, play around with the Live desktop, and if you're satisfied, launch the installer by clicking on its icons on the dock or on the desktop.
After the installer starts, click through the first few steps until you get to the one shown in Figure. The installer should detect that Windows 10 is already installed and offer to install Ubuntu 16.04 alongside it. That's great, but it will not create a separate partition for your files and folders.

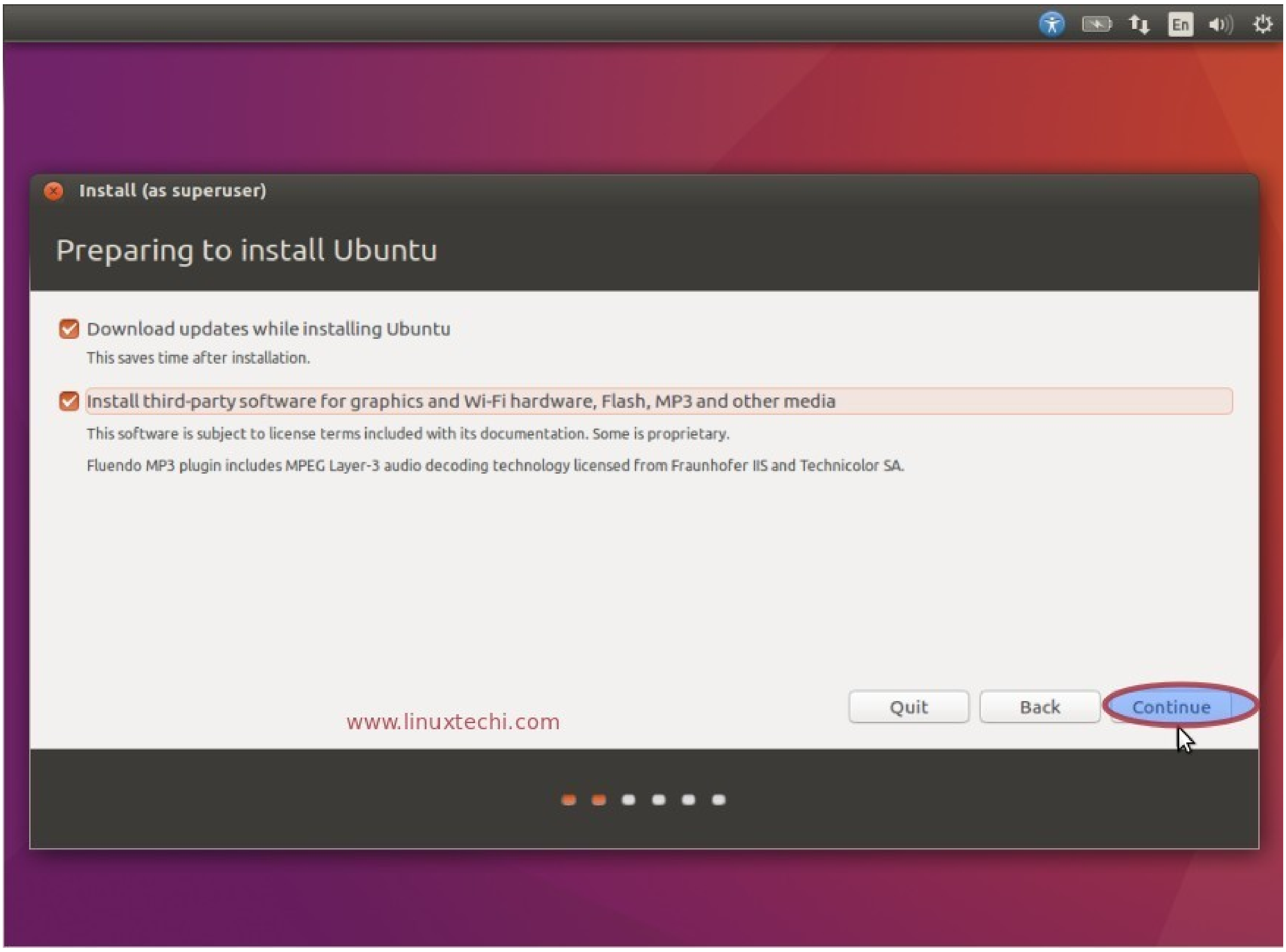


You can try Ubuntu without making any changes to your computer, directly from this CD.

Or if you're ready, you can install Ubuntu alongside (or instead of) your current operating system. This shouldn't take too long.

You may wish to read the release notes.

In case your system is connected to the Internet and wants to install third party tools during installation, you can select both the options as shown in below snap otherwise leave the options uncheck.



Install (as superuser)

Preparing to install Ubuntu

☑ Download updates while installing Ubuntu

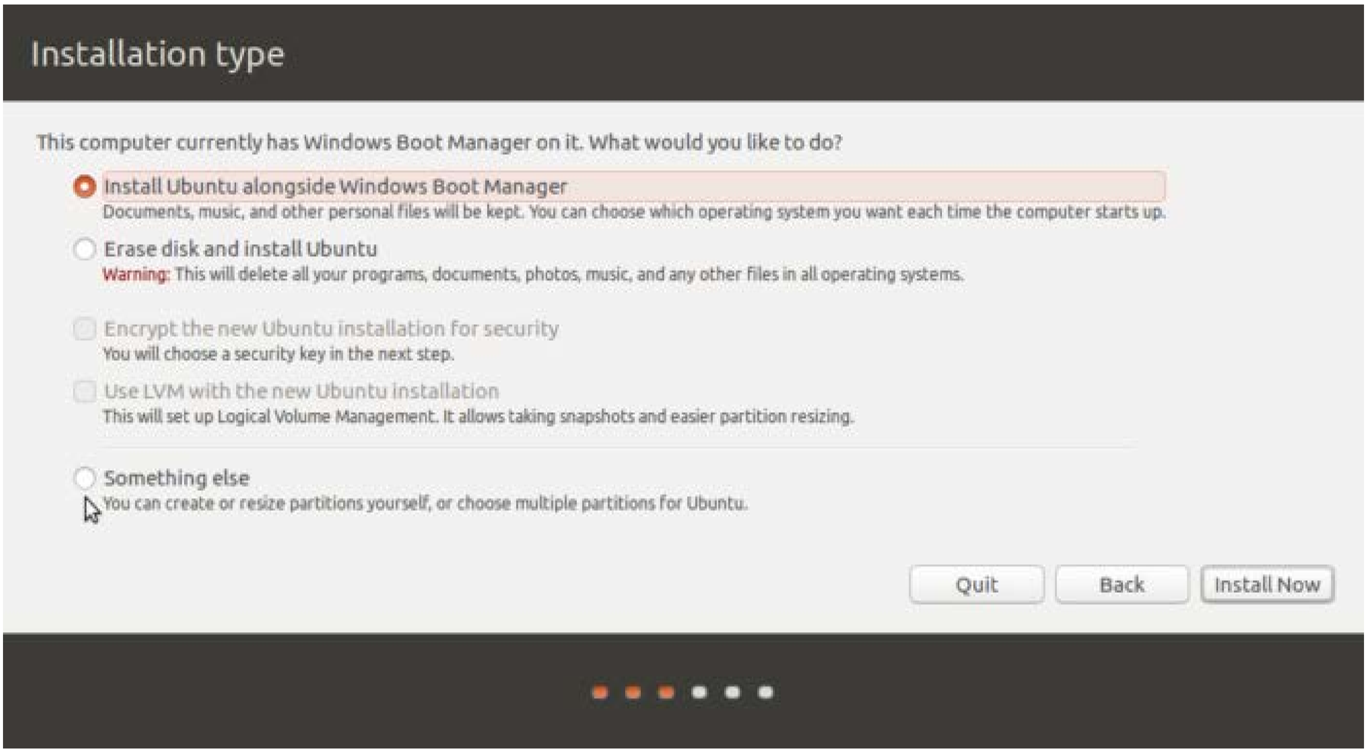
This saves time after installation.

☑ Install third-party software for graphics and Wi-Fi hardware, Flash, MP3 and other media

This software is subject to license terms included with its documentation. Some is proprietary.

Fluendo MP3 plugin includes MPEG Layer-3 audio decoding technology licensed from Fraunhofer IIS and Technicolor SA.

Click on **Continue** to proceed further.



Installation type

This computer currently has Windows Boot Manager on it. What would you like to do?

● Install Ubuntu alongside Windows Boot Manager

Documents, music, and other personal files will be kept. You can choose which operating system you want each time the computer starts up.

○ Erase disk and install Ubuntu

Warning: This will delete all your programs, documents, photos, music, and any other files in all operating systems.

□ Encrypt the new Ubuntu installation for security

You will choose a security key in the next step.

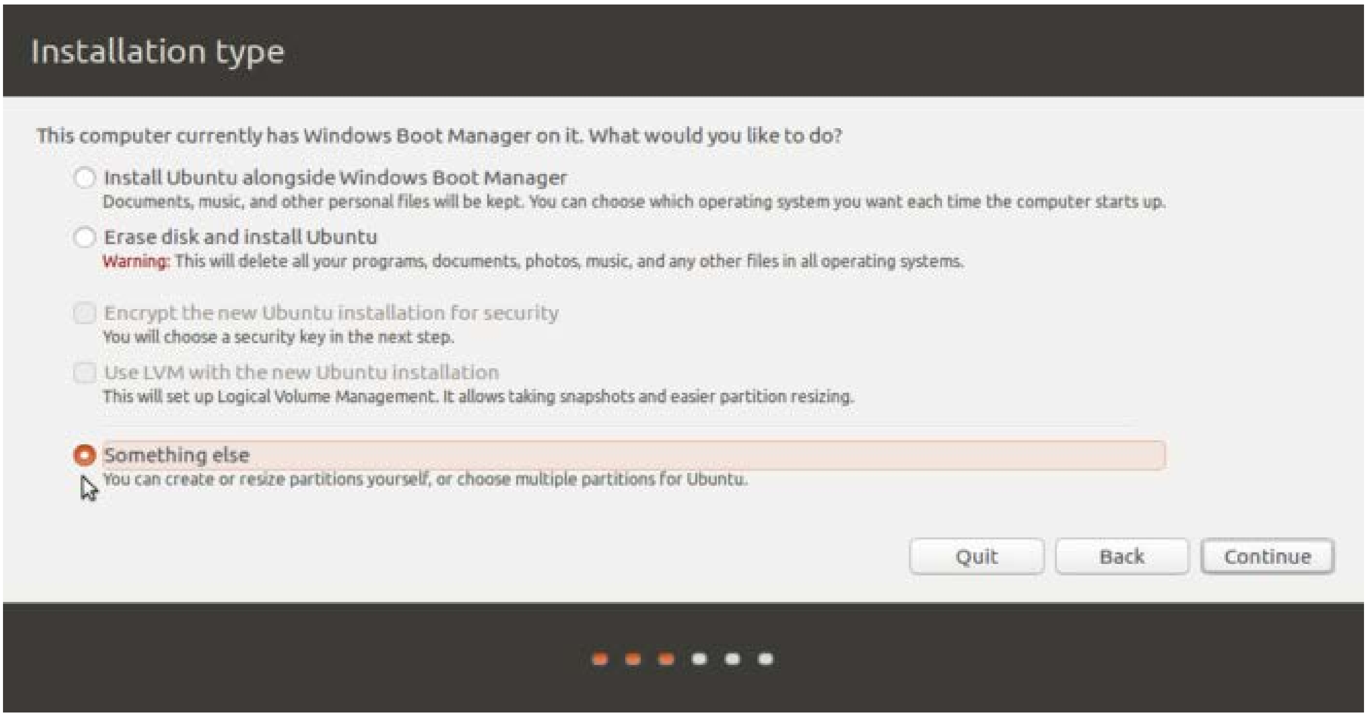
□ Use LVM with the new Ubuntu installation

This will set up Logical Volume Management. It allows taking snapshots and easier partition resizing.

○ Something else

You can create or resize partitions yourself, or choose multiple partitions for Ubuntu.

For that to happen, you'll have to select the **Something else** option, then click on the **Continue** button to move to the next step.



Installation type

This computer currently has Windows Boot Manager on it. What would you like to do?

○ Install Ubuntu alongside Windows Boot Manager
Documents, music, and other personal files will be kept. You can choose which operating system you want each time the computer starts up.

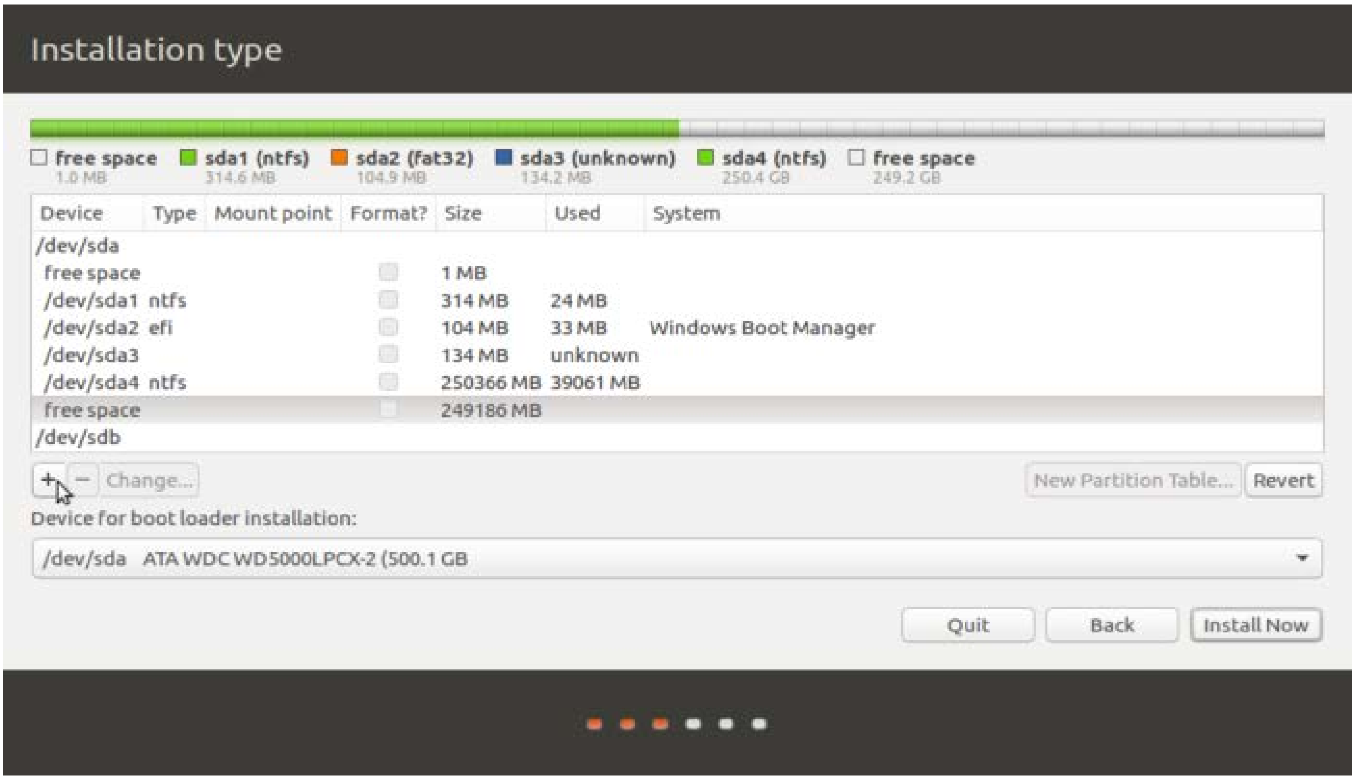
○ Erase disk and install Ubuntu
Warning: This will delete all your programs, documents, photos, music, and any other files in all operating systems.

□ Encrypt the new Ubuntu installation for security
You will choose a security key in the next step.

□ Use LVM with the new Ubuntu installation
This will set up Logical Volume Management. It allows taking snapshots and easier partition resizing.

● Something else
You can create or resize partitions yourself, or choose multiple partitions for Ubuntu.

That next step shows the so-called advanced partitioning tool, where you'll see all the partitions detected by the installer. Of the existing Windows 10 partitions, the one of interest here is the one that has **efi** in the Type column and **Windows Boot Manager** in the System column. That is the EFI System Partition, where the boot loader files will be installed. To start creating partitions, select the free space, then click the **+** button.



Installation type

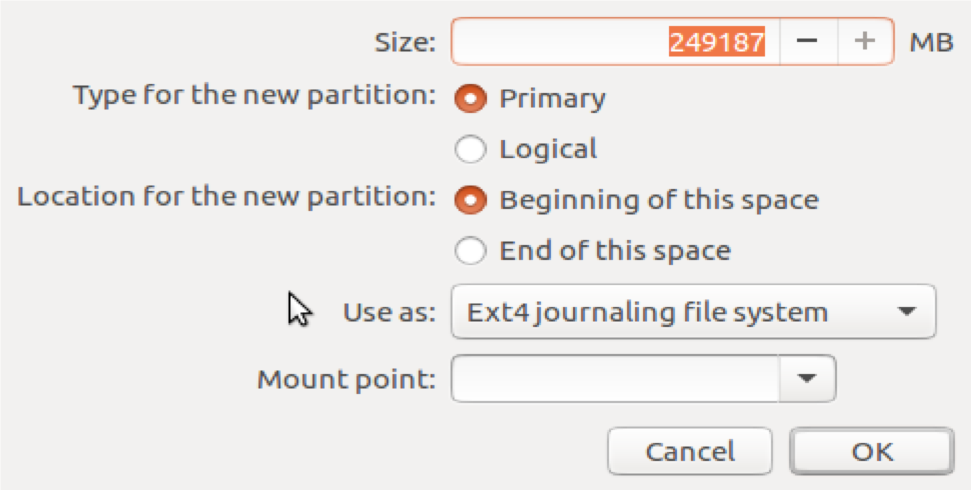
□ free space 1.0 MB | 🟩 sda1 (ntfs) 314.6 MB | 🟧 sda2 (fat32) 104.9 MB | 🟦 sda3 (unknown) 134.2 MB | 🟩 sda4 (ntfs) 250.4 GB | □ free space 249.2 GB

Device	Type	Mount point	Format?	Size	Used	System
/dev/sda						
free space			□	1 MB		
/dev/sda1	ntfs		□	314 MB	24 MB	
/dev/sda2	efi		□	104 MB	33 MB	Windows Boot Manager
/dev/sda3			□	134 MB	unknown	
/dev/sda4	ntfs		□	250366 MB	39061 MB	
free space			□	249186 MB		
/dev/sdb						

Device for boot loader installation:

/dev/sda ATA WDC WD5000LPCX-2 (500.1 GB

That should bring up the installer's partition editor. For each partition, you only need to specify the size, the file system (from the **Use as** menu) and the mount point. For partitions other than the one that will be used as swap, stick with the default file system – **Ext4 journaling file system**.



Size: 249187 MB

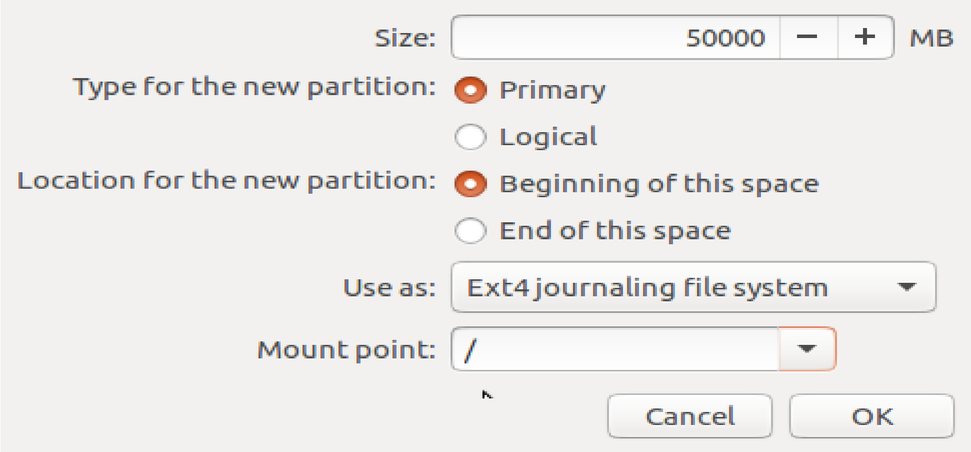
Type for the new partition: ● Primary ○ Logical

Location for the new partition: ● Beginning of this space ○ End of this space

Use as: Ext4 journaling file system

Mount point:

So let's start by creating the root partition. For size, any value higher than 20 GB will do. From the **Mount point**, select /. Click **OK**.



Size: 50000 MB

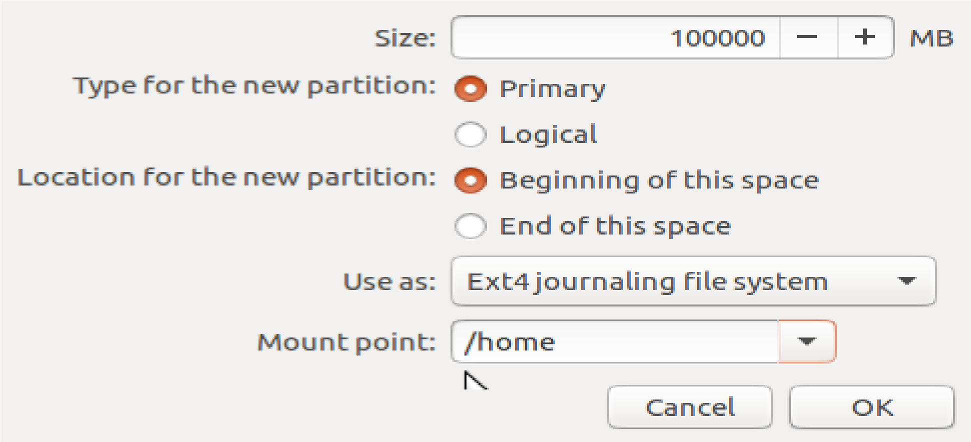
Type for the new partition: ● Primary ○ Logical

Location for the new partition: ● Beginning of this space ○ End of this space

Use as: Ext4 journaling file system

Mount point: /

The next partition will be mounted at /home. This is the partition that will hold all your files and folders, so assign most of the available disk space to it. When you're finished setting it up, click **OK**.



Size: 100000 MB

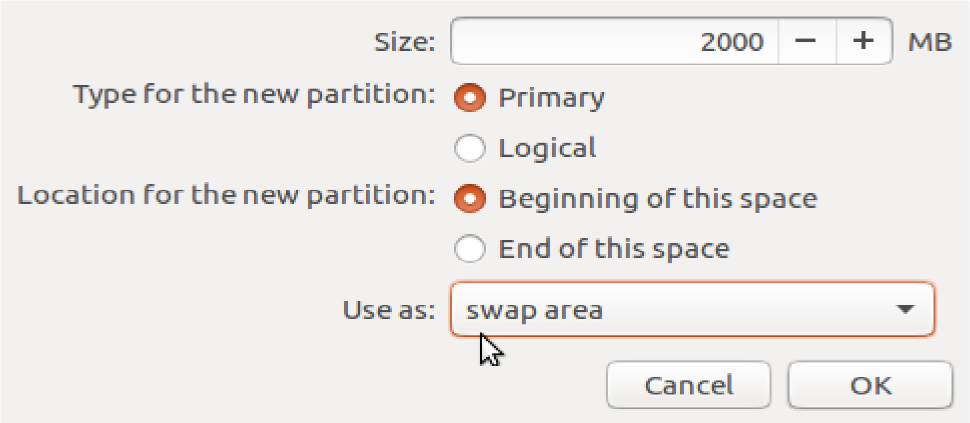
Type for the new partition: ● Primary ○ Logical

Location for the new partition: ● Beginning of this space ○ End of this space

Use as: Ext4 journaling file system

Mount point: /home

For the partition that will be used as swap, a size of 2000 MB (2 GB) to 4000 MB (4 GB) should be enough. From Use as, select **swap area. OK**.



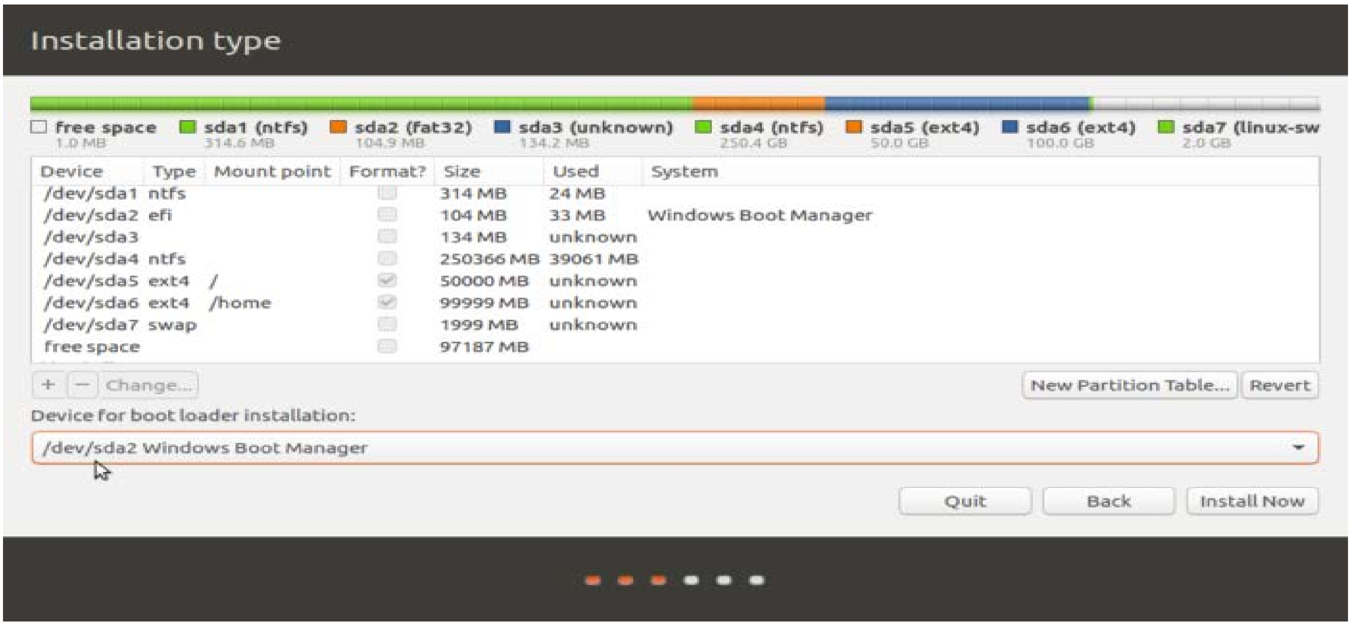
Size: 2000 MB

Type for the new partition: ● Primary ○ Logical

Location for the new partition: ● Beginning of this space ○ End of this space

Use as: swap area

After creating that last partition, you should be back in the main partitioning tool's window for the last time. From the **Device for boot loader installation**, select the efi partition. On the system used for this article, that partition corresponds to /dev/sda2, that is, the second partition on the hard drive. Then click on the **Install Now** button to continue with the rest of the installation.



Installation type

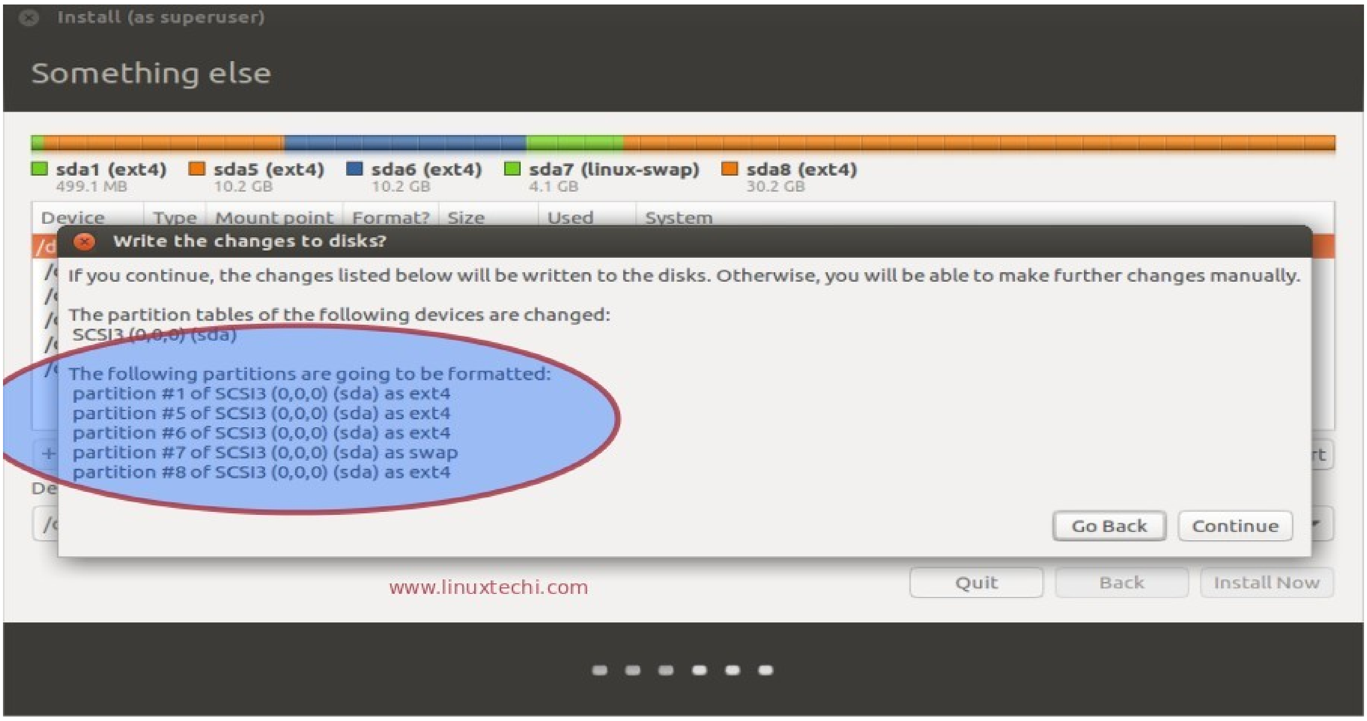
□ free space 1.0 MB 🟩 sda1 (ntfs) 314.6 MB 🟧 sda2 (fat32) 104.9 MB 🟦 sda3 (unknown) 134.2 MB 🟩 sda4 (ntfs) 250.4 GB 🟧 sda5 (ext4) 50.0 GB 🟦 sda6 (ext4) 100.0 GB 🟩 sda7 (linux-sw 2.0 GB

Device	Type	Mount point	Format?	Size	Used	System
/dev/sda1	ntfs		☐	314 MB	24 MB	
/dev/sda2	efi		☐	104 MB	33 MB	Windows Boot Manager
/dev/sda3			☐	134 MB	unknown	
/dev/sda4	ntfs		☐	250366 MB	39061 MB	
/dev/sda5	ext4	/	☑	50000 MB	unknown	
/dev/sda6	ext4	/home	☑	99999 MB	unknown	
/dev/sda7	swap		☐	1999 MB	unknown	
free space			☐	97187 MB		

Device for boot loader installation: /dev/sda2 Windows Boot Manager

After installation has completed successfully, rebooting the computer should reboot into the GRUB menu. GRUB (Grand Unified Bootloader) is the boot loader used on virtually all Linux distributions. Aside from the entries for Ubuntu 16.04, it will also have one for Windows 10, which makes it possible to boot into Windows from the GRUB menu. Unfortunately, you can't add an entry for Ubuntu 16.04 in the Windows Boot Manager menu, so it's recommended that you make GRUB the default boot manager.

Once you are done with partition table click on '**Install Now**'. It will show the below screen, click on Continue to Proceed.



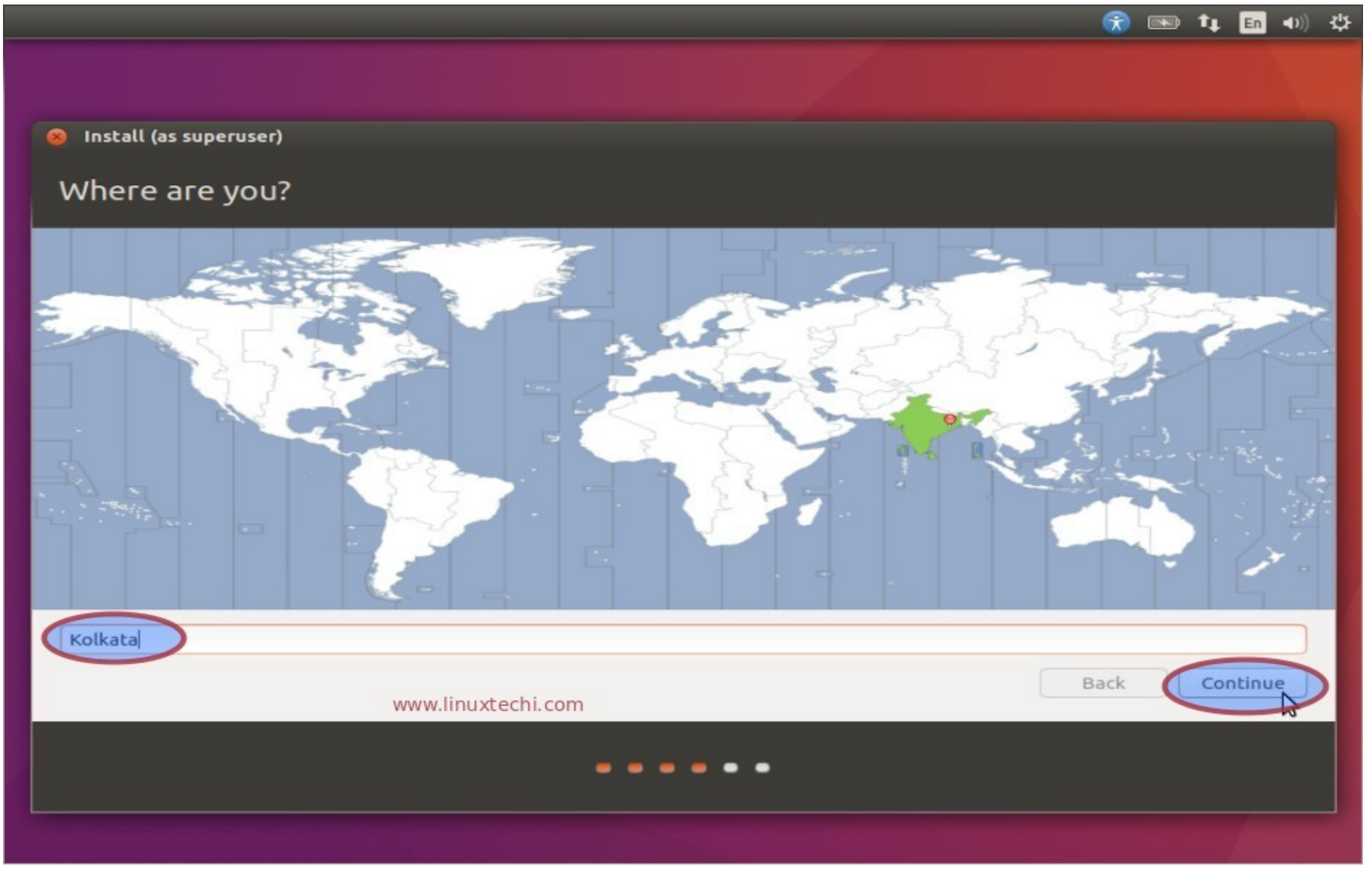
Write the changes to disks?

If you continue, the changes listed below will be written to the disks. Otherwise, you will be able to make further changes manually.

The partition tables of the following devices are changed:
SCSI3 (0,0,0) (sda)

The following partitions are going to be formatted:
partition #1 of SCSI3 (0,0,0) (sda) as ext4
partition #5 of SCSI3 (0,0,0) (sda) as ext4
partition #6 of SCSI3 (0,0,0) (sda) as ext4
partition #7 of SCSI3 (0,0,0) (sda) as swap
partition #8 of SCSI3 (0,0,0) (sda) as ext4

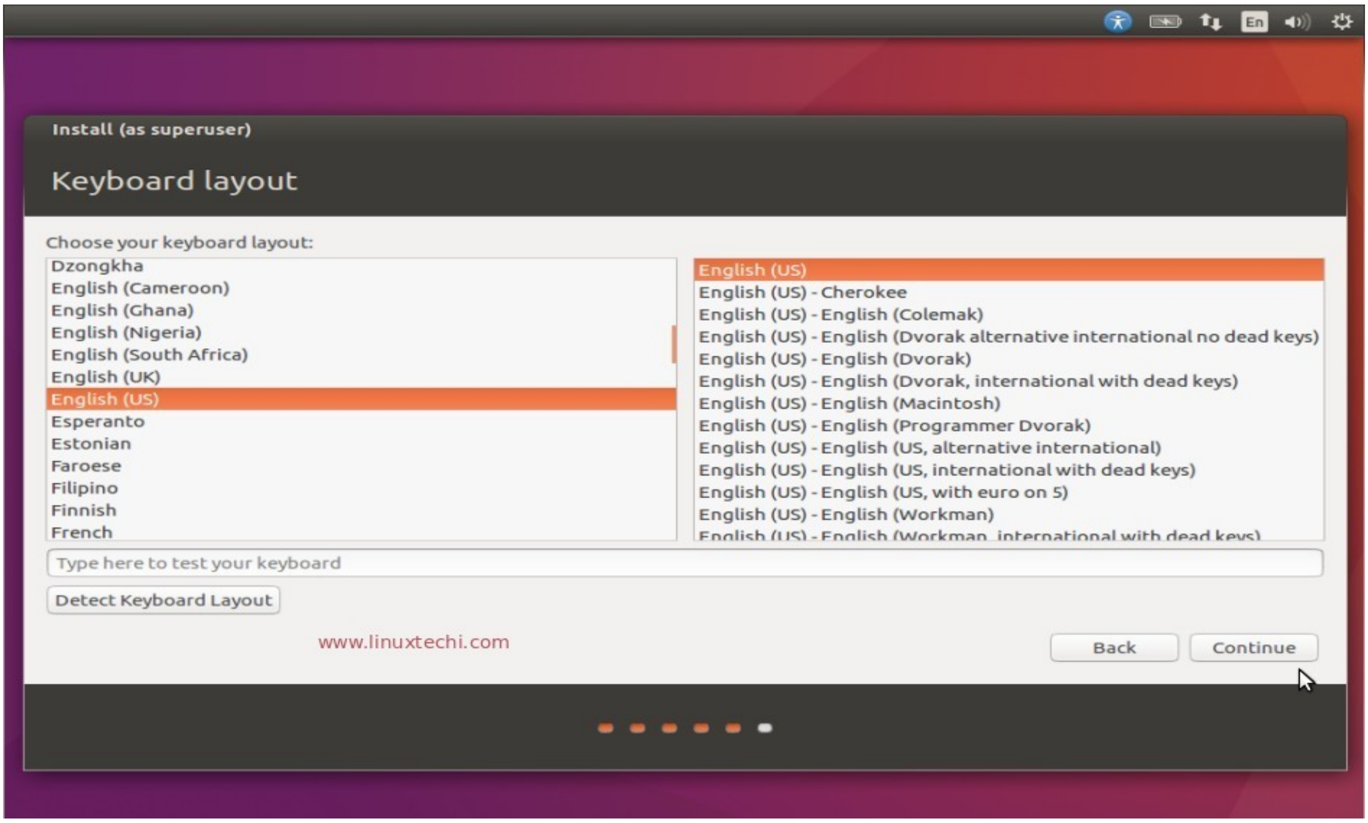
Specify the Time Zone as per your location.



Kolkata

Click on Continue.

Select the Keyboard Layout as per your setup.



Install (as superuser)

Keyboard layout

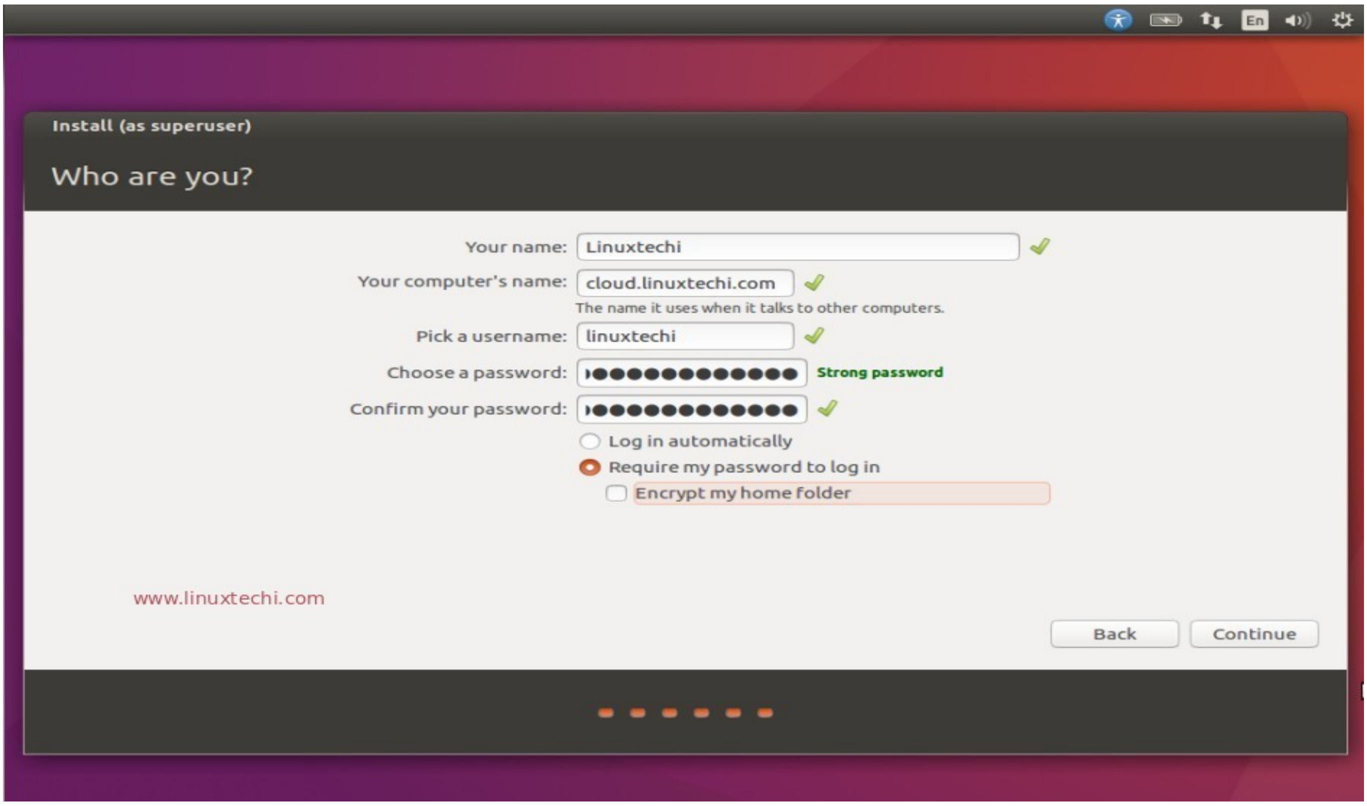
Choose your keyboard layout:

Dzongkha
English (Cameroon)
English (Ghana)
English (Nigeria)
English (South Africa)
English (UK)
English (US)
Esperanto
Estonian
Faroese
Filipino
Finnish
French

English (US)
English (US) - Cherokee
English (US) - English (Colemak)
English (US) - English (Dvorak alternative international no dead keys)
English (US) - English (Dvorak)
English (US) - English (Dvorak, international with dead keys)
English (US) - English (Macintosh)
English (US) - English (Programmer Dvorak)
English (US) - English (US, alternative international)
English (US) - English (US, international with dead keys)
English (US) - English (US, with euro on 5)
English (US) - English (Workman)
English (US) - English (Workman, international with dead keys)

Specify the Hostname, User name and its password.

In this step specify the hostname for your system, user name and its password. We will using this user to login to the system after the installation.



Install (as superuser)

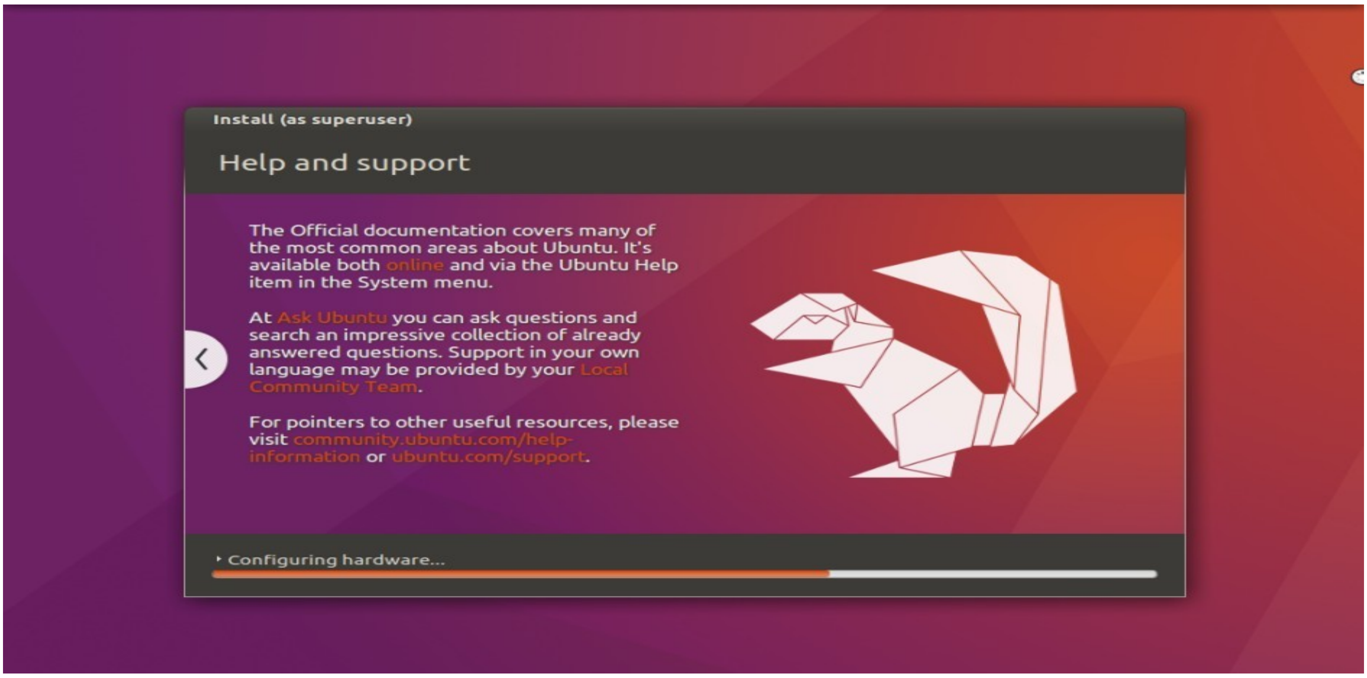
Who are you?

Your name: Linuxtechi ✓
Your computer's name: cloud.linuxtechi.com ✓
The name it uses when it talks to other computers.
Pick a username: linuxtechi ✓
Choose a password: ●●●●●●●●●●●●●● Strong password
Confirm your password: ●●●●●●●●●●●●●● ✓
○ Log in automatically
● Require my password to log in
□ Encrypt my home folder

Click on Continue to start the installation

Installation is in Progress.

As we can see below that installation is progress, once the installation is completed we will get the message to reboot the system.



Install (as superuser)

Help and support

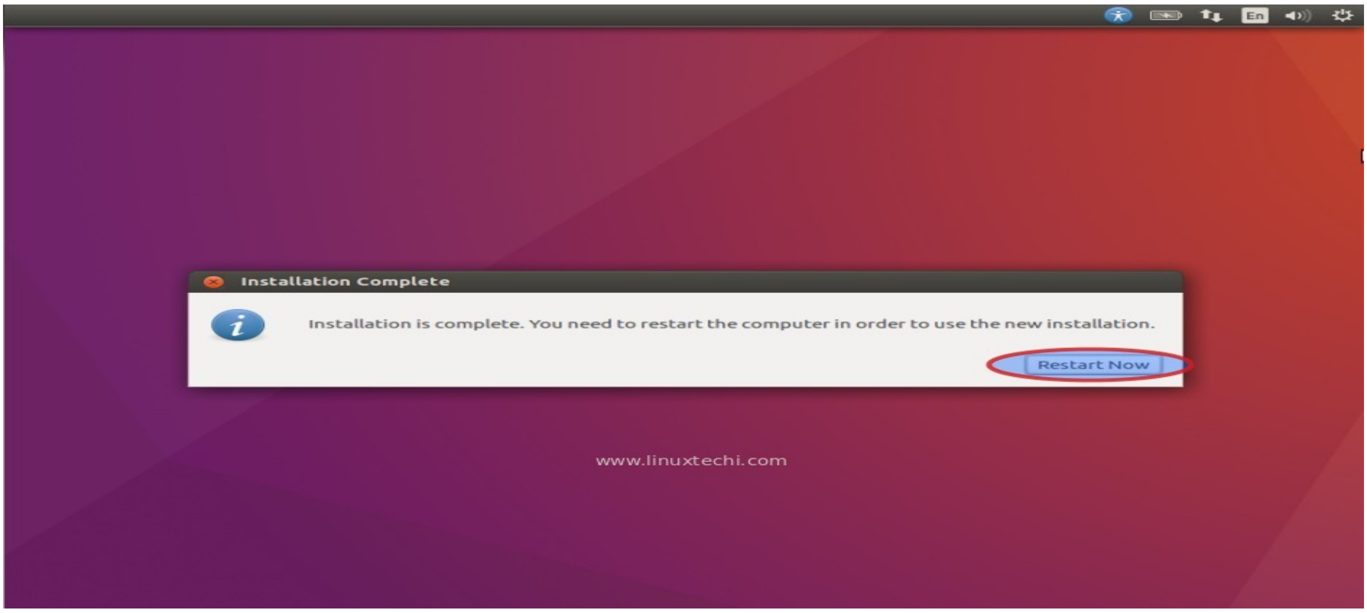
The Official documentation covers many of the most common areas about Ubuntu. It's available both online and via the Ubuntu Help item in the System menu.

At Ask Ubuntu you can ask questions and search an impressive collection of already answered questions. Support in your own language may be provided by your Local Community Team.

For pointers to other useful resources, please visit community.ubuntu.com/help-information or ubuntu.com/support.

Configuring hardware...

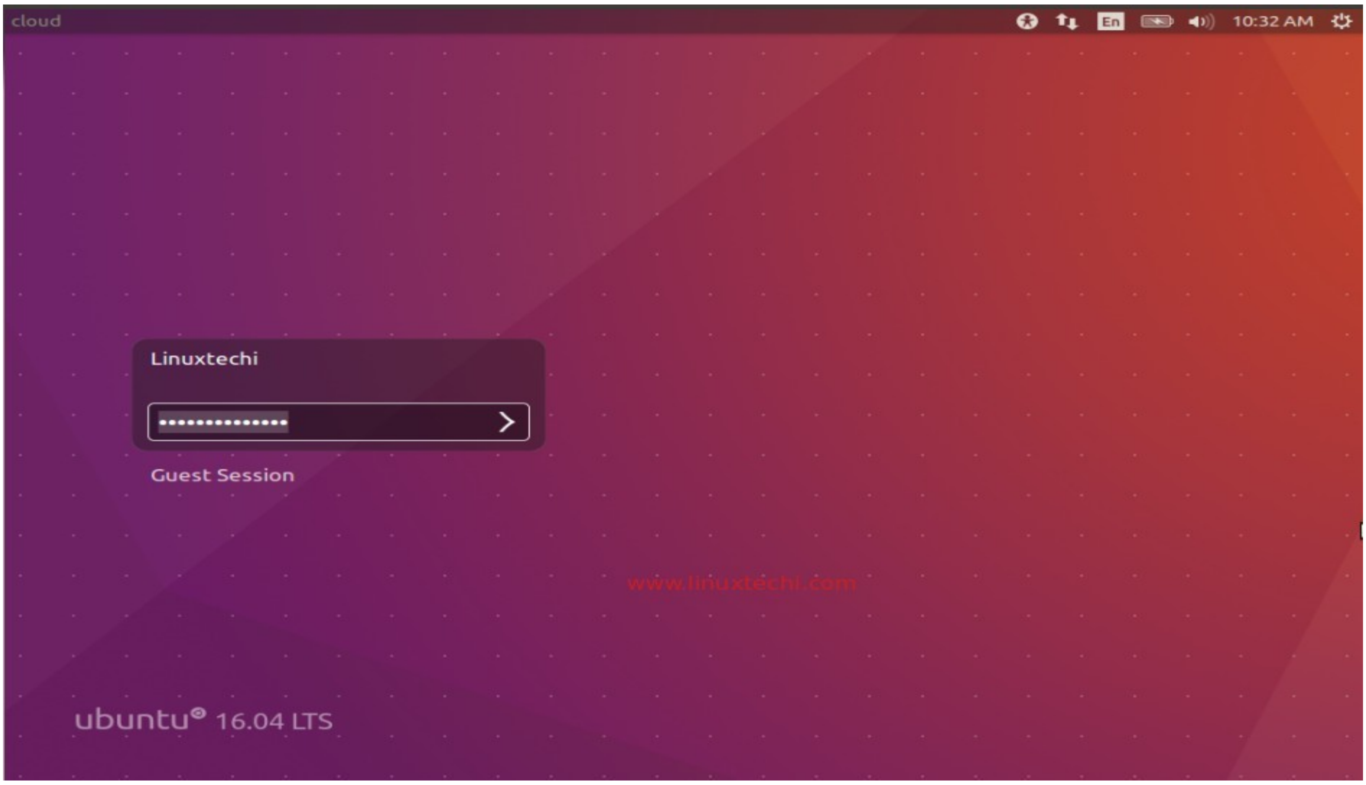
To Reboot the system click on '**Restart Now**'

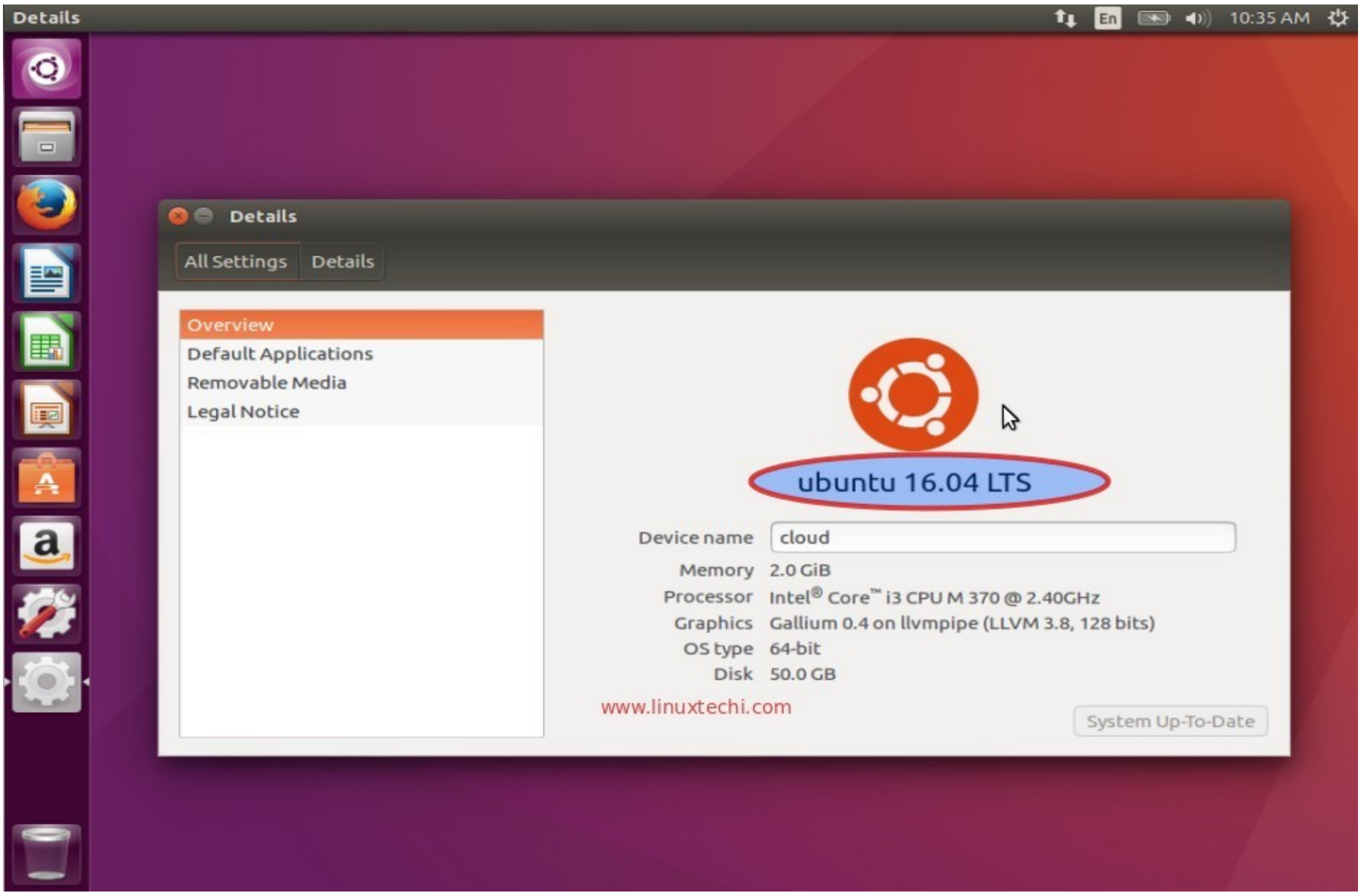


Installation Complete

Installation is complete. You need to restart the computer in order to use the new installation.

Now login to the system with User that we created during installation.





Device name cloud
Memory 2.0 GiB
Processor Intel® Core™ i3 CPU M 370 @ 2.40GHz
Graphics Gallium 0.4 on llvmpipe (LLVM 3.8, 128 bits)
OS type 64-bit
Disk 50.0 GB

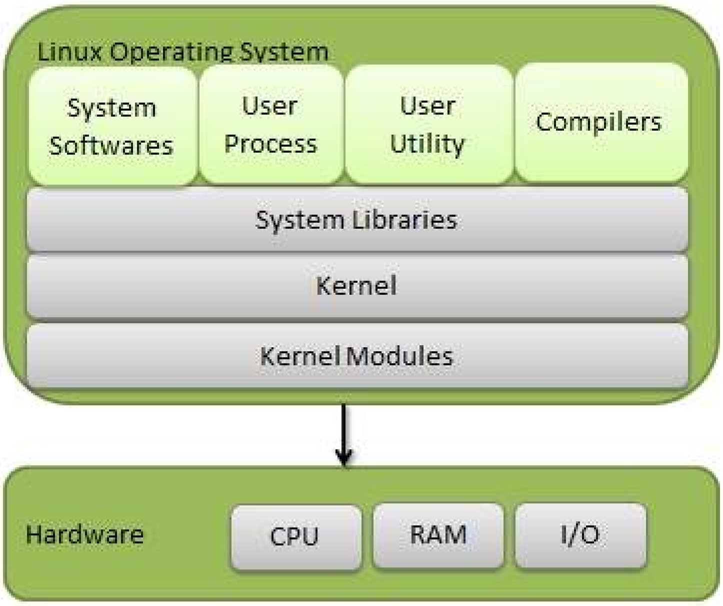
Installation of Ubuntu 16.04 LTS is completed, explore it now and have fun.

Linux is one of popular version of UNIX operating System. It is open source as its source code is freely available. It is free to use. Linux was designed considering UNIX compatibility. Its functionality list is quite similar to that of UNIX.

Components of Linux System

Linux Operating System has primarily three components

- **Kernel** − Kernel is the core part of Linux. It is responsible for all major activities of this operating system. It consists of various modules and it interacts directly with the underlying hardware. Kernel provides the required abstraction to hide low level hardware details to system or application programs.
- **System Library** − System libraries are special functions or programs using which application programs or system utilities accesses Kernel's features. These libraries implement most of the functionalities of the operating system and do not requires kernel module's code access rights.
- **System Utility** − System Utility programs are responsible to do specialized, individual level tasks.



Kernel Mode vs User Mode

Kernel component code executes in a special privileged mode called **kernel mode** with full access to all resources of the computer. This code represents a single process, executes in single address space and do not require any context switch and hence is very efficient and fast. Kernel runs each processes and provides system services to processes, provides protected access to hardware to processes.

Support code which is not required to run in kernel mode is in System Library. User programs and other system programs works in **User Mode** which has no access to system hardware and kernel code. User programs/ utilities use System libraries to access Kernel functions to get system's low level tasks.

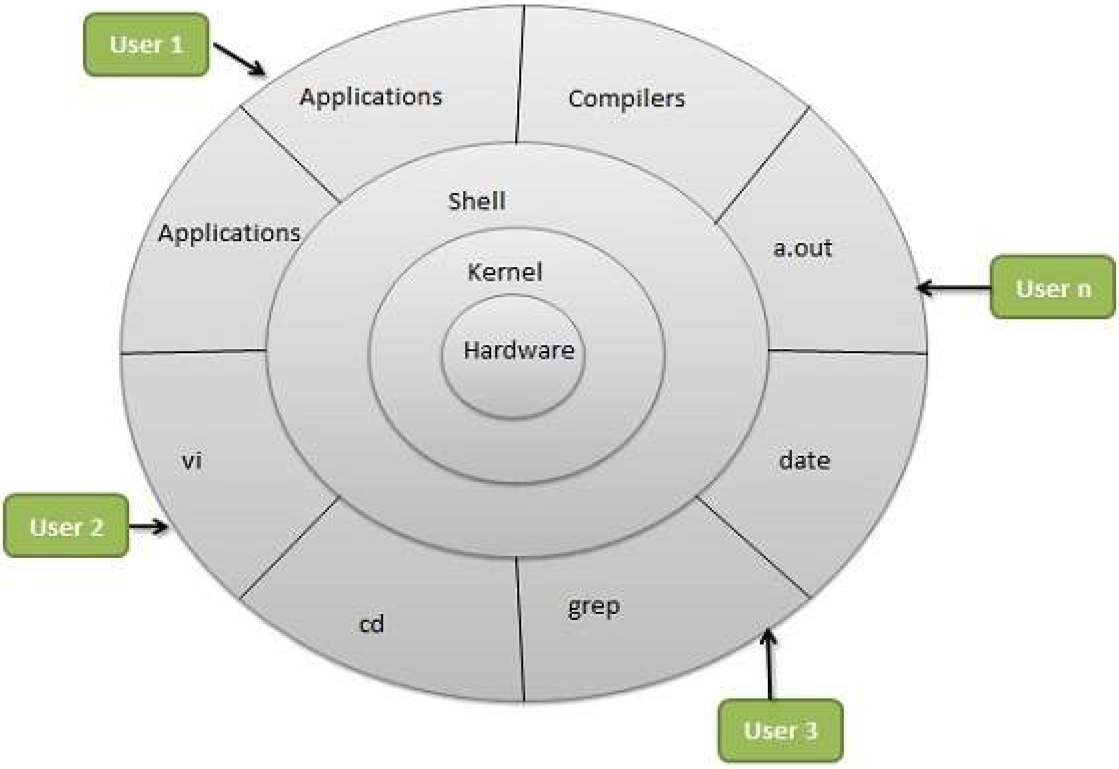
Basic Features

Following are some of the important features of Linux Operating System.

- **Portable** – Portability means software can works on different types of hardware in same way. Linux kernel and application programs supports their installation on any kind of hardware platform.
- **Open Source** – Linux source code is freely available and it is community based development project. Multiple teams work in collaboration to enhance the capability of Linux operating system and it is continuously evolving.
- **Multi-User** – Linux is a multiuser system means multiple users can access system resources like memory/ ram/ application programs at same time.
- **Multiprogramming** – Linux is a multiprogramming system means multiple applications can run at same time.
- **Hierarchical File System** – Linux provides a standard file structure in which system files/ user files are arranged.
- **Shell** – Linux provides a special interpreter program which can be used to execute commands of the operating system. It can be used to do various types of operations, call application programs. etc.
- **Security** – Linux provides user security using authentication features like password protection/ controlled access to specific files/ encryption of data.

Architecture

The following illustration shows the architecture of a Linux system –



The architecture of a Linux System consists of the following layers –

- **Hardware layer** – Hardware consists of all peripheral devices (RAM/ HDD/ CPU etc).
- **Kernel** – It is the core component of Operating System, interacts directly with hardware, provides low level services to upper layer components.

- **Shell** – An interface to kernel, hiding complexity of kernel's functions from users. The shell takes commands from the user and executes kernel's functions.
- **Utilities** – Utility programs that provide the user most of the functionalities of an operating systems.

Linux or ubuntu Directory structure

In the Linux operating system, all filesystems are contained within one directory hierarchy. The root directory is the top level directory, and all its subdirectories make up the directory hierarchy. This differs to other operating systems such as MS-Windows which applies a separate hierarchy for each device and partition.

- /bin -- binary applications (most of your executable files)
- /boot -- files required to boot (such as the kernel, etc)
- /dev -- your devices (everything from drives to displays)
- /etc -- just about every configuration file for your system
- /etc/profile.d -- contains scripts that are run by /etc/profile upon login.
- /etc/rc.d -- contains a number of shell scripts that are run on bootup at different run levels. There is also typically an rc.inet1 script to set up networking (in Slackwar), an rc.modules script to load modular device drivers, and an rc.local script that can be edited to run commands desired by the administrator, along the lines of autoexec.bat in DOS.
- /etc/rc.d/init.d -- contains most of the initialization scripts themselves on an rpm-based system.
- /etc/rc.d/rc*.d -- where "*" is a number corresponding to the default run level. Contains files for services to be started and stopped at that run level. On rpm-based systems, these files are symbolic links to the initialization scripts themselves, which are in /etc/rc.d/init.d.
- /etc/skel -- directory containing several example or skeleton initialization shells. Often contains subdirectories and files used to populate a new user's home directory.
- /etc/X11 -- configuration files for the X Window system
- /home -- locally stored user files and folders
- /lib -- system libraries (similar to Program Files)
- /lost+found -- lost and found for lost files
- /media -- mounted (or loaded) devices such as cdroms, digital cameras, etc.
- /mnt -- mounted file systems
- /opt -- location for “optionally” installed programs
- /proc -- dynamic directory including information about and listing of processes
- /root -- “home” folder for the root user
- /sbin -- system-only binaries (see /bin)
- /sys -- contains information about the system
- /tmp -- temporary files
- /usr -- applications mainly for regular users
- /var -- mainly logs, databases, etc.
- /usr/local/bin -- the place to put your own programs. They will not be overwritten with upgrades.
- /usr/share/doc -- documentation.

Basic Linux or ubuntu commands :

1. alias command

The **alias** command can be useful if you want to create a 'shortcut' to a command. The format is **alias name='command'**

```
> alias home='cd /home/dave/public_html'
```

This will create an alias called home which will put you in the /home/dave/public_html directory whenever you type home at the command prompt. You can alias any command you want, and include options for the command.

```
> alias list='ls -la'
```

This will create an alias called list, which will use the **ls** command to print a long-style listing of all files in the current directory (the **-l** gives a long-style list, and the **-a** shows all files - including hidden files).

(Find out more about the **ls** command)
To see a list of aliases set up on your linux box, just type **alias** at the prompt.

```
> alias
alias attrib='chmod'
alias chdir='cd'
alias copy='cp'
alias cp='cp -i'
alias d='dir'
alias del='rm'
alias deltree='rm -r'
alias dir='/bin/ls $LS_OPTIONS --format=vertical'
alias edit='pico'
alias ff='whereis'
alias ls='/bin/ls $LS_OPTIONS'
alias mem='top'
alias move='mv'
alias mv='mv -i'
alias pico='pico -w -z'
alias rm='rm -i'
alias search='grep'
alias v='vdir'
alias vdir='/bin/ls $LS_OPTIONS --format=long'
alias which='type -path'
alias wtf='watch -n 1 w -hs'
alias wth='ps -uxa | more'
>
```

2. unalias command

The **unalias** command is used to remove an alias (see alias).
The format is **unalias name**

```
> unalias home          ; This will remove the alias called home.
```

3. apt-get command

apt-get : APT is acronym for Advanced Package Tool. It supports installing packages over internet using ftp or http protocols. You can also upgrade all packages in a single operations, which makes it even more attractive.
You can find out debian package name with the following command:

```
apt-cache search {package-name}
```

```
apt-cache search apache
```

Finally, most of the actions listed in this post are written with the assumption that they will be executed by the root user running the bash or any other modern shell.

apt-get add a new package
Add a new package called samba. The syntax is:

apt-get install {package-name**}**

To install a package called samba, run:

```
# apt-get install samba
```

apt-get remove the package called samba but keep the configuration files
The syntax is:

apt-get remove {package-name**}**

Example:

```
# apt-get remove samba
```

4. dpkg command

dpkg : Debian packaging tool which can be use to install, query, uninstall packages.
dpkg command to get package information such as description of package, version etc.

The syntax is:

dpkg --info **{**.deb-package-name**}**

Example:

```
# dpkg --info sudo_1.6.7p5-2_i386.deb | less
```

List all installed packages

The syntax is:

dpkg -l

Example:

```
# dpkg –l
```

To list individual package try such as apache

```
# dpkg -l apache
```

You can also use this command to see (verify) if package sudo is install or not (note that if package is installed then it displays package name along with small description):

```
# dpkg -l | grep -i 'sudo'
```

To list packages related to the apache:

```
# dpkg -l '*apache*'
```

5. cat command (concatenate files)

The **cat** command can be used to join multiple files together and print the result on screen.
One dictionary definition of concatenate is *"to connect or link in a series or chain*"
Lets say we have a file called 01.txt which is a simple text file containing the text "this is the text in file 01", and another file called 02.txt containing the text "this is the text in file 02". If we run **cat** on file 01 we get the following result.

```
> cat 01.txt
this is the text in file 01
```

If we run the cat command on both files, we get this...

```
> cat 01.txt 02.txt
this is the text in file 01
this is the test in file 02
```

You can see the **cat** has printed the contents of both files.
If no file is named, the standard input (usually the keyboard) is printed on screen (also known as 'standard output').

6. cd command

The **cd** command, which stands for "change directory", changes the shell's current working directory.

Description

The **cd** command is one of the commands you will use the most at the command line in linux. It allows you to change your working directory. You use it to move around within the hierarchy of your file system.

About Directories

To help you organize your files, your file system contains special files called directories. Think of them like folders in a file cabinet: they have names, just like files, but their function is to "contain" other files, and other directories. In this way, you can keep the files on your system separate and sorted according to their function or purpose.

All files and directories on your system stem from one main directory: the root directory. There are no directories "above" the root directory; all other directories are "below" the root directory.

Any directory which is contained inside another directory is called a subdirectory. Subdirectories "branch" off the "root" of the directory "tree." Unlike a real tree, directory trees are upside-down: the root is at the top and the branches reach down. When you move into a subdirectory, you are moving "down" the tree; when you move into a directory's parent directory, you are moving "up" the tree.

All directories on your file system are subdirectories of the root directory.

Note: By default, when you open a terminal and begin using the command line, you are placed in your home directory.

How Directories Are Represented

Directories are separated by a forward slash ("/"). For instance, the directory name "**documents/work/accounting**" means "the directory named **accounting**, which is in the directory named **work**, which is in the directory named **documents** which is in the current directory."

To change into this directory, and make it our working directory, we would use the command:

```
cd documents/work/accounting
```

If the first character of a directory name is a slash, that denotes that the directory path begins in the root directory. So, in contrast to the example above, the directory name "**/documents/work/accounting**" (note the beginning slash) means "the directory named **accounting**, which is in the directory named **work**, which is in the directory named **documents** which is in the root directory."

To change into this directory, making it our working directory, we would use the command:

```
cd /documents/work/accounting
```

Representing The Root Directory

The root directory itself is represented by a single slash ("/").

To change into the root directory, making it our working directory, we would use the command:

```
cd /
```

Representing The Working Directory

The current directory, regardless of which directory it is, is represented by a single dot ("**.**").

So, running this command:

```
cd .
```

...would change us into the directory we're already in. In other words, it would do nothing, but it would do it successfully.

What's actually happening is the dot represents the "assumed" directory; it's a placeholder, and you can use the dot anywhere in a directory name. So, the command:

```
cd documents
```

...is the same as the command:

```
cd ./documents
```

...and also the same as:

```
cd documents/.
```

...as well as:

```
cd ./documents/.
```

In all of these examples, the dot represents "the directory assumed to be there". You can use it as a placeholder anywhere you want to tell the shell that a directory goes in that place, and to assume the appropriate value.

Representing the Parent Directory

The parent directory of the current directory — in other words, the directory one level up from the current directory, which contains the directory we're in now — is represented by two dots ("**..**").

So, If we were in the directory **/home/username/documents**, and we executed the command:

```
cd ..
```

...we would be placed in the directory **/home/username**.

The double-dot ("**..**") directory notation can be used anywhere in a directory name to represent going up one level. For instance, if we have two directories, **/home/username/documents** and **/home/username/downloads**, and we are currently in **/home/username/documents**, we could type the following:

```
cd ../downloads
```

...and we would be placed in **/home/username/downloads**.

Another "useless" command, but one that illustrates the way you can use the double-dot notation, is this one:

```
cd documents/..
```

...which will place us in the directory one level above the subdirectory **documents** — in other words, the current directory. Note that this will only work if the subdirectory **documents** already exists.

Similarly, the command:

```
cd documents/../documents
```

...is functionally the same as this command:

```
cd documents
```

Representing Your Home Directory

Your home directory is the directory you're placed in, by default, when you open a new terminal session. It's the directory that holds all your settings, your mail, your default documents and downloads folder, and many other personal items. It has a special representation: a tilde ("**~**").

So, if our username is **username**, and our home directory is **/home/username**, the command:

```
cd ~
```

...is functionally the same as the command:

```
cd /home/username
```

...and we can always access the subdirectories of our home directory by placing the tilde as the first component of the the directory name. For instance, if your documents folder is named **/home/username/documents**, you can always move into that directory using the command:

```
cd ~/documents
```

Using A Trailing Slash

Using a slash at the end of a directory name is optional. Directories are simply files, so you don't need to put it there; but if you do put it there, the system knows for sure you're expecting that file to be a directory. For example, if there is a subdirectory off the current directory named **dirname**, the command:

```
cd dirname
```

...is the same as the command:

```
cd dirname/
```

The second form of the command explicitly states that **dirname** is a directory, but both commands are equivalent.

cd syntax

cd [-L|-P] *directory*

Options

-L Force symbolic links to be followed. In other words, if you tell **cd** to move into a "directory" which is actually a symbolic link to a directory, it moves into the directory the symbolic link points to.

This is the default behavior of **cd**; normally, it will always act as if **-L** has been specified.

-P Use the physical directory structure without following symbolic links. In other words, only change into the specified directory if it actually exists as named; symbolic links will not be followed. This is the opposite of the **-L** option, and if they are both specified, this option will be ignored.

If the **-P** option is specified, and the current working directory cannot be determined, this option tells **cd** to exit with an error. If **-P** is not specified along with this option, this option has no function.

cd examples

```
cd hope
```

The above example would change the working directory to the **hope** subdirectory if it exists.

```
cd ../computerhope
```

The above example would traverse up one level to the parent directory and then down into the directory **computerhope**.

```
cd ../../
```

Traverse two directories up the directory tree. In other words, move into the directory which contains the directory which contains the current working directory.

7. chown command (change ownership)

The **chown** command is used to change ownership of files and directories.

The format is **chown newowner filename**

```
> chown david file.txt
```

This will cause file.txt to now be owned by david.

8. ls command (List directory contents)

The **ls** command will list the files and directories within the current working directory (the directory you are currently in).
There are a few options you can use with **ls**, and the format, or syntax of the command is....

ls [options] [file]

EXAMPLE
>ls -l /home/rich/www
Lets break that down....

ls is the command
-l is the option which will give you a long listing format (which shows more info than just the file names - the owner, size, date last modified etc)
/home/rich/www is the directory we want to see a listing of (if you omit this part, **ls** will print the contents of the directory you are in).

Options
Some useful options are -l, -a, -s, -h and -R
-l will give you a long listing (as explained above)
-a will show you ALL the files in the directory, including hidden files
-R will the subdirectories recursively, which means it will show all the directories and files within the specified directory.
-s will also show you the size of the files (in blocks, not bytes)
-h will show the size in "human readable format" (ie: 4K, 16M, 1G etc). Of course you must use this option in conjunction with the -s option.
You can combine as many of these options as you wish.
EXAMPLES
>ls -la /home
Lists ALL the files and directories in the /home directory, in the long listing format.
>ls -ash
Lists ALL the files in the current directory (no directory was specified so it lists the contents of the current directory), and the size of the files/directories, written in 'human readable' format.
There are many more options for **ls**, but these are a few you may want to use. To see the total list of options for the **ls** command, you can type **man ls** at the prompt.

9. grep command examples

Search for a given string in a file (case in-sensitive search).
$ grep -i "the" demo_file
Print the matched line, along with the 3 lines after it.
$ grep -A 3 -i "example" demo_text
Search for a given string in all files recursively
$ grep -r "ramesh" *

10. ssh command examples

Login to remote host

```
ssh -l jsmith remotehost.example.com
```

Debug ssh client

```
ssh -v -l jsmith remotehost.example.com
```

Display ssh client version

```
$ ssh -V
OpenSSH_3.9p1, OpenSSL 0.9.7a Feb 19 2003
```

11. vim command examples

Go to the 143rd line of file

```
$ vim +143 filename.txt
```

Go to the first match of the specified

```
$ vim +/search-term filename.txt
```

Open the file in read only mode.

```
$ vim -R /etc/passwd
```

12. shutdown command examples

Shutdown the system and turn the power off immediately.

```
# shutdown -h now
```

Shutdown the system after 10 minutes.

```
# shutdown -h +10
```

Reboot the system using shutdown command.

```
# shutdown -r now
```

Force the filesystem check during reboot.

```
# shutdown -Fr now
```

13. kill command examples

Use kill command to terminate a process. First get the process id using ps -ef command, then use kill -9 to kill the running Linux process as shown below. You can also use killall, pkill, xkill to terminate a unix process.

```
$ ps -ef | grep vim
ramesh    7243  7222  9 22:43 pts/2    00:00:00 vim

$ kill -9 7243
```

14. rm command examples

Get confirmation before removing the file.

```
$ rm -i filename.txt
```

It is very useful while giving shell metacharacters in the file name argument.

Print the filename and get confirmation before removing the file.

```
$ rm -i file*
```

Following example recursively removes all files and directories under the example directory. This also removes the example directory itself.

```
$ rm -r example
```

15. mv command examples

Rename file1 to file2. if file2 exists prompt for confirmation before overwritting it.

```
$ mv -i file1 file2
```

Note: mv -f is just the opposite, which will overwrite file2 without prompting.
mv -v will print what is happening during file rename, which is useful while specifying shell metacharacters in the file name argument.

```
$ mv -v file1 file2
```

16. cat command examples

You can view multiple files at the same time. Following example prints the content of file1 followed by file2 to stdout.

```
$ cat file1 file2
```

While displaying the file, following cat -n command will prepend the line number to each line of the output.

```
$ cat -n /etc/logrotate.conf
   1       /var/log/btmp {
   2         missingok
   3         monthly
   4         create 0660 root utmp
   5         rotate 1
   6       }
```

17. mount command examples

To mount a file system, you should first create a directory and mount it as shown below.

```
# mkdir /u01

# mount /dev/sdb1 /u01
```

You can also add this to the fstab for automatic mounting. i.e Anytime system is restarted, the filesystem will be mounted.

```
/dev/sdb1 /u01 ext2 defaults 0 2
```

18. chmod command examples

chmod command is used to change the permissions for a file or directory.
Give full access to user and group (i.e read, write and execute) on a specific file.

```
$ chmod ug+rwx file.txt
```

Revoke all access for the group (i.e read, write and execute) on a specific file.

```
$ chmod g-rwx file.txt
```

Apply the file permissions recursively to all the files in the sub-directories.

```
$ chmod -R ug+rwx file.txt
```

19. chown command examples

chown command is used to change the owner and group of a file. \
To change owner to oracle and group to db on a file. i.e Change both owner and group at the same time.

```
$ chown oracle:dba dbora.sh
```

Use -R to change the ownership recursively.

```
$ chown -R oracle:dba /home/oracle
```

20. mkdir command examples

Following example creates a directory called temp under your home directory.

```
$ mkdir ~/temp
```

Create nested directories using one mkdir command. If any of these directories exist already, it will not display any error. If any of these directories doesn't exist, it will create them.

```
$ mkdir -p dir1/dir2/dir3/dir4/
```

21. ifconfig command examples

Use ifconfig command to view or configure a network interface on the Linux system.
View all the interfaces along with status.

```
$ ifconfig -a
```

Start or stop a specific interface using up and down command as shown below.

```
$ ifconfig eth0 up

$ ifconfig eth0 down
```

22. man command examples

Display the man page of a specific command.

```
$ man crontab
```

When a man page for a command is located under more than one section, you can view the man page for that command from a specific section as shown below.

```
$ man SECTION-NUMBER commandname
```

Following 8 sections are available in the man page.

1. General commands
2. System calls
3. C library functions
4. Special files (usually devices, those found in /dev) and drivers
5. File formats and conventions
6. Games and screensavers

7. Miscellaneous
8. System administration commands and daemons

For example, when you do whatis crontab, you'll notice that crontab has two man pages (section 1 and section 5). To view section 5 of crontab man page, do the following.

```
$ whatis crontab
crontab (1)        - maintain crontab files for individual users (V3)
crontab (5)        - tables for driving cron
$ man 5 crontab
```

23. su command examples

Switch to a different user account using su command. Super user can switch to any other user without entering their password.

```
$ su - USERNAME
```

Execute a single command from a different account name. In the following example, john can execute the ls command as raj username. Once the command is executed, it will come back to john's account.

```
[john@dev-server]$ su - raj -c 'ls'
[john@dev-server]$
```

Login to a specified user account, and execute the specified shell instead of the default shell.

```
$ su -s 'SHELLNAME' USERNAME
```

24. yum command examples

To install apache using yum.

```
$ yum install httpd
```

To upgrade apache using yum.

```
$ yum update httpd
```

To uninstall/remove apache using yum.

```
$ yum remove httpd
```

25. ping command examples

Ping a remote host by sending only 5 packets.

```
$ ping -c 5 gmail.com
```

26. date command examples

Set the system date:

```
# date -s "01/31/2010 23:59:53"
```

Once you've changed the system date, you should syncronize the hardware clock with the system date as shown below.

```
# hwclock –systohc
# hwclock --systohc –utc
```