Quick Start Guide
To Install Ubuntu Linux
With Support For Steam Games

Created by

Aaron Franke

and

James Black

Table of Contents

Introduction

This quick start guide helps you install Ubuntu Linux and helps you set up Steam games on it. Ubuntu is a fast and efficient operating system based on Linux. Ubuntu is a great alternative to Windows or MacOS. It is free and open, and designed to work for you, not against you.

This guide expects you to have the following things:

- Access to the Internet
- Steam account already created (for gaming)
- Flash drive or DVD to use as installation media
- A working computer to create installation media
- A computer that you wish to install Linux on (can be the same as above)

You can install Linux on pretty much any computer, laptop or desktop, PC or Mac. However, it is worth noting that Nvidia GPUs tend to work better than AMD GPUs on Linux.

Of course, for gaming, you want it to be a powerful system. If you are not planning on gaming, or you want to play retro games, you don't need a powerful computer, and you can still follow this guide to set up Ubuntu for non-gaming purposes if you wish.

Step 1. Prepare your computer

Before you install Ubuntu Linux, you have two options. You can either wipe your existing OS, having only Ubuntu installed, or dual-boot, so you have both installed at once.

If you have Windows installed and wish to dual-boot, press "Win+X," then click "Power Options" -> "Choose what the power buttons do" -> "Change settings that are currently unavailable" -> Disable "Fast startup (recommended)" and Hibernation, as shown in Figure 1.

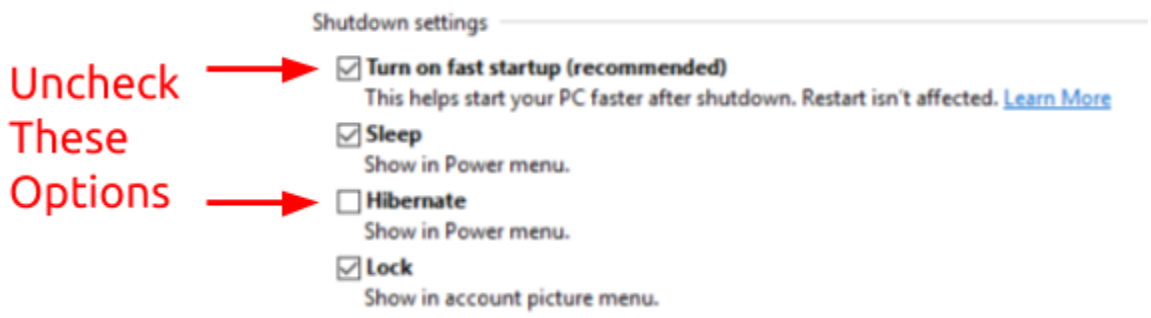


Figure 1 - Startup and shutdown settings

Step 2. Create installation media

You will need to download Ubuntu, which comes as an ISO file. Click here to download Ubuntu 18.04 LTS Bionic Beaver. The steps are the same for flavors such as Xubuntu and Kubuntu, if you wish to use those instead.

Once downloaded, you will either need to burn this image to a DVD or you will need to put it on a large flash drive. For flash drives, you will need to use the Rufus or dd tools if you are creating the installer from Windows or Mac/Linux respectively. Follow the steps below to put the installer on a flash drive:

Windows:

1. Download Rufus and run it. It will look similarly to Figure 2 on the right.
2. Select the flash drive you wish to use.
3. Select the ISO image using “SELECT.”
4. Click “START” to begin writing the ISO.
5. Wait for it to finish. Be patient, this may take a long time.

Mac or Linux:

1. Open a Terminal.
2. Get a list of disks using “diskutil list” on Mac or using “lsblk” on Linux.
3. Look for the “/dev/disk[number]” or “/dev/sd[letter]” of your disk.
4. Write the ISO file using this command: sudo dd if=ubuntu.iso of=/dev/[whatever]
5. Wait for it to finish. Be patient, this may take a long time. The dd tool does not display progress while running.

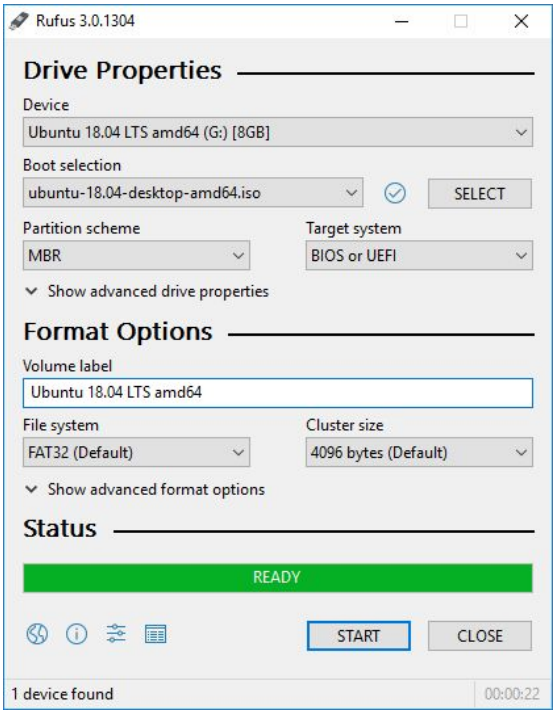


Figure 2 - Rufus on Windows

Step 3. Configure motherboard

In order to ensure Ubuntu works correctly, you should change some of your motherboard firmware settings.

- To open up your firmware settings, check your motherboard's manual (if available) to see what button to press when booting. Sometimes the key to press will show up on the screen as it boots.
- With some newer computers, the procedure is more complex:
    - In Windows, hold Shift and click Restart. You should get a blue colored screen.
    - Click "Troubleshoot" -> "Advanced" -> "UEFI Firmware Settings" -> "Restart."

The motherboard firmware settings will look different on each computer. You will need to change the following settings:

- Disable "Secure Boot." The correct setting may look like "Off" or "Other OS."
- Disable "Fast Boot" if the option exists. If you can't find it, don't worry.
- Change the boot order to make flash drives the top priority.

Figure 3 - UEFI Secure Boot settings

Step 4. Boot the Installer

Now that you have configured your motherboard, it should be easy to boot up the installer flash drive or DVD. This is different for each computer and each motherboard.

1. Look for something like “Select boot device” or “Boot device options / override”
2. Once you are in that menu, select “UEFI: [Your flash drive or DVD name here]”. Make sure you are clicking the option with the word “UEFI” in it.

If you get an error booting the installer, try returning to Step 2 or 3.

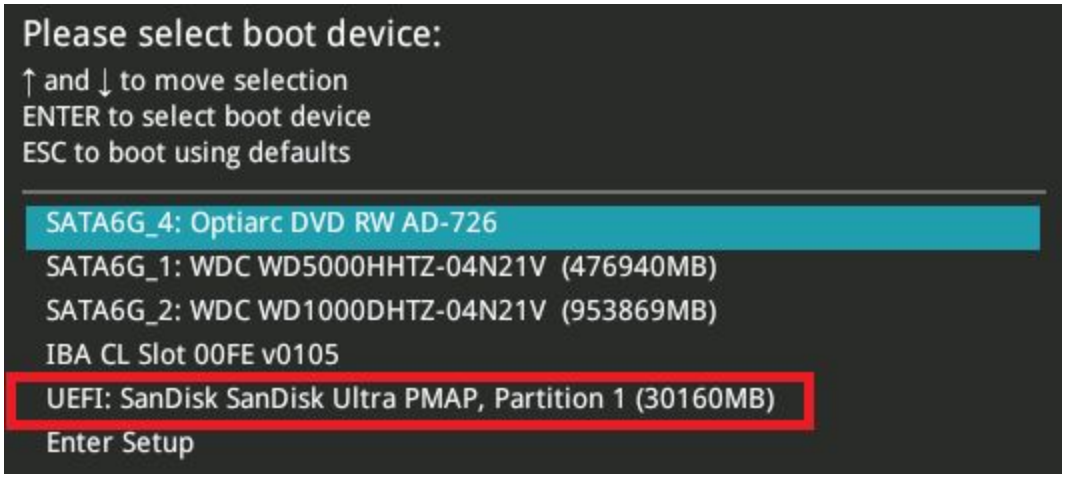


Figure 4 - An example of a boot menu

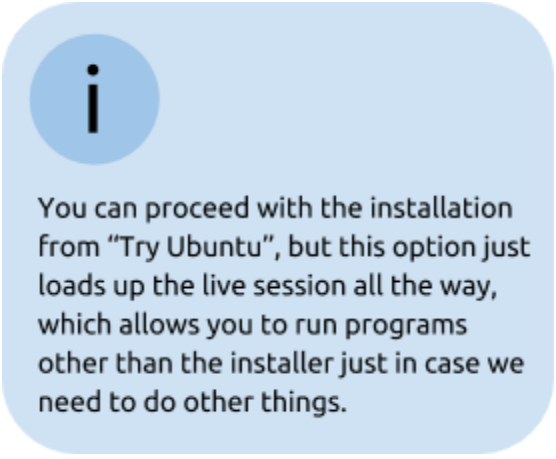
Step 5. Start the Installation Process

When the installer boots, select “Try Ubuntu” and not “Install Ubuntu” (see note).

Next, connect to WiFi, if applicable, by clicking the top right corner of the screen.

If dual-booting, try to open your storage drive. If you get a very long error, repeat Step 1.

Once you have done that, click on the “Install Ubuntu” icon on the desktop to start the installation process.



Step 6. Install the OS

First, you will be asked to select your language and keyboard layout. The default is English.

You will then be presented with several options of what software you want:

- Ensure that “Install third-party software for graphics [...]” is checked.
- We recommend using a “Normal installation” which includes useful utilities.
- Optionally, feel free to select “Download updates while installing”, which makes the installation take longer, but saves time after installing.

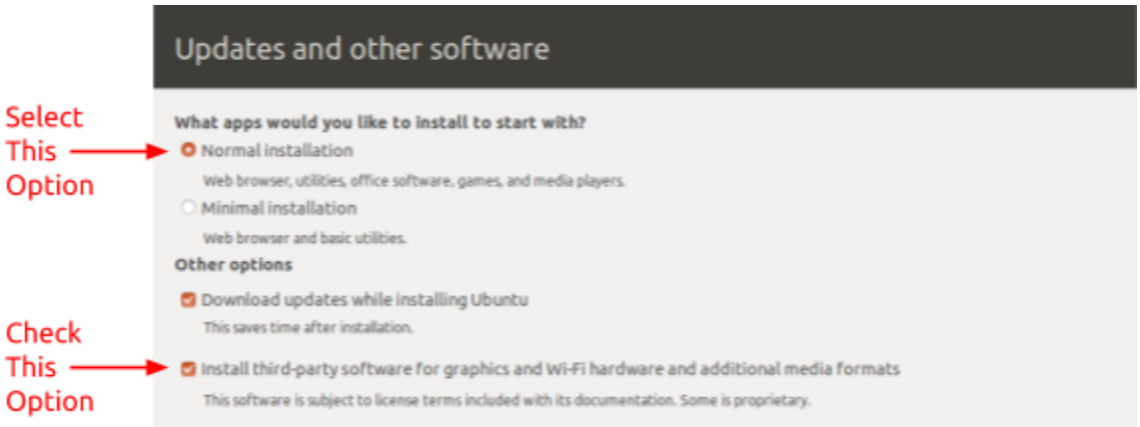


Figure 6.1 - Updates and software selection menu

Next, you will need to select how and where you want to install Ubuntu.

- If you are dual-booting, select “Install Ubuntu alongside Windows”
- If you are not dual-booting, select “Erase disk and install Ubuntu”
- If you have a complex setup, you can use“Something else”, but we do not recommend using this most of the time and we will not be covering how to use this option.

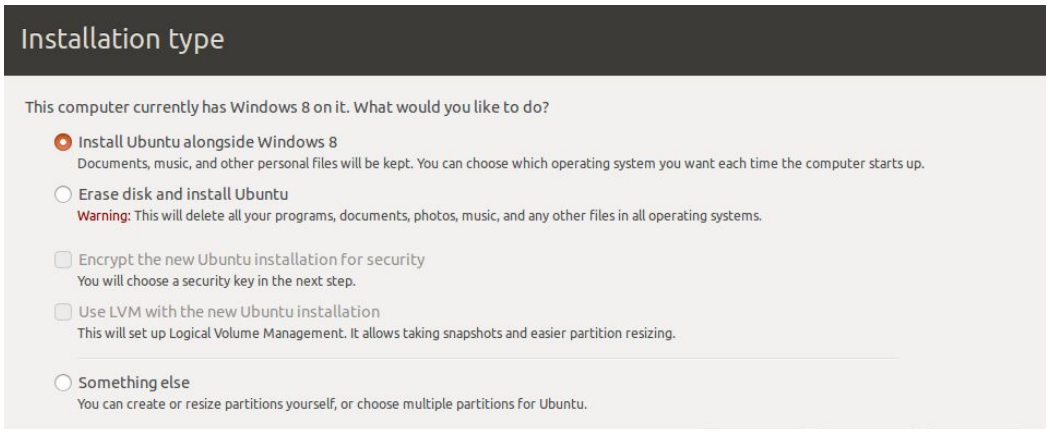


Figure 6.2 - Installation type menu

The installer will then ask you to configure your time zone. Just click on the map. Do not worry if the location is listed as “New York”, that is simply the time zone for the east coast.

Next, you will be asked for your name, your computer’s name, and your username. Your name can be as fancy as you want, and it is only used for displaying. Try to keep the computer’s name and username short, descriptive, and using simple characters. For example, the computer name could be “john-hp-ub18”, which is a descriptive name because it tells me that it’s John’s HP computer running Ubuntu 18.04.

Figure 6.3 - Account setup menu

When you have finished with all of the above, you simply need to wait until the installation finishes.

Once finished, reboot the computer, unplugging the installation media when asked, as in Figure 6.4. Then, press Enter to finish rebooting.

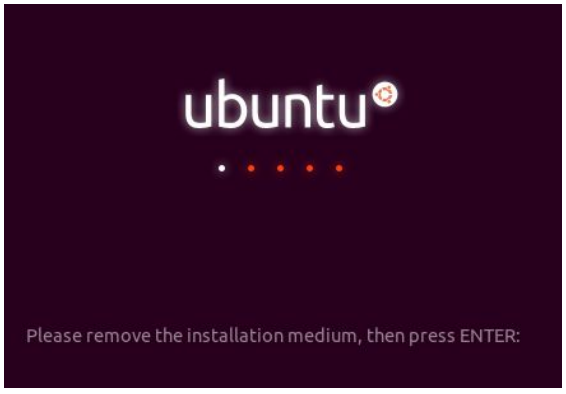


Figure 6.4 - Installation media removal prompt

Step 7. Boot the OS for the first time

After you complete the installation, turn on or reboot the computer and boot up Ubuntu.

If you are dual-booting, make note of GRUB, the tool that allows you to select which OS you want to boot. This will appear every time you turn on your computer. If you are not dual-booting, this screen will not appear, and instead you will automatically boot into Ubuntu.

When you log in for the first time, you will be presented with a “Welcome to Ubuntu” screen. While it is not required, we encourage you to read through this slideshow.

It starts on a “What’s new in Ubuntu” page, which is intended for people who have used previous versions of Ubuntu, but also provides a nice quick reference for navigating the UI. Most of the interface is obvious and intuitive, and similar to Windows and Mac, but if anything here isn’t obvious to you, it might be a good idea to memorize it.

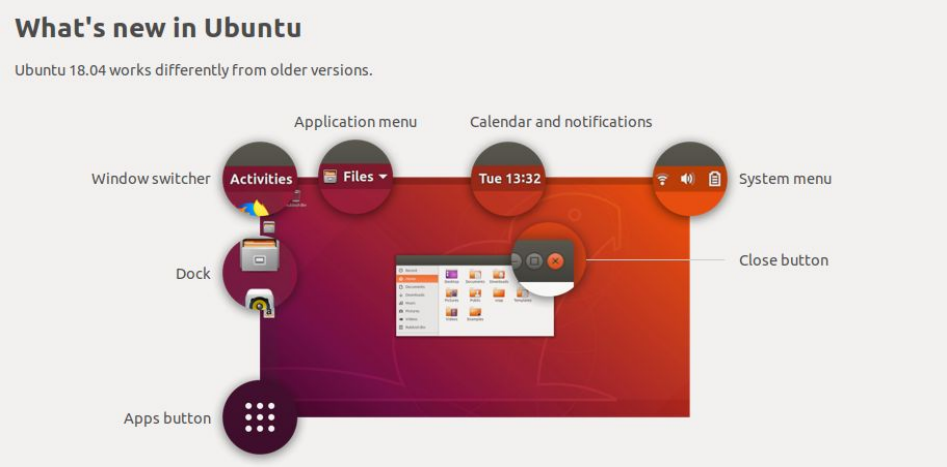


Figure 7 - Welcome to Ubuntu screen

The next page allows you to set up a live patching service, so that you don’t have to restart your computer to install updates. It’s not necessary, you can just skip this page.

The next page asks if you wish to send some system information to Canonical to help improve Ubuntu. We recommend keeping this enabled, as the data collected is extremely minimal.

The last page of the welcome screen encourages you to try out “Ubuntu Software”, Ubuntu’s graphical package manager program. Feel free to look around!

Step 8. Configure the installation

First, open the “Software Updater” menu, and use it to update your computer. This tool updates everything on your system - not just the OS, but also programs installed via the package manager. Run this tool frequently to keep your system updated and secure.

Figure 8 - Software Updater

Ubuntu, like most other Linux distros, are preconfigured to work for you, and not for a company. There is no candy crush or spyware pre-installed, so unlike with Windows, there are few things that should be changed or disabled.

However, Ubuntu is preconfigured for general use, not gaming. We highly recommend downloading and running this setup script which sets up many gaming-related things for you, including Steam, Wine, Java, Itch, Discord, and more.

- Download the ZIP archive linked above.
- Extract the ZIP file using the built-in archive manager.
- Open a Terminal through the “Applications” menu. Don’t be afraid, it’s just text.
- Run the command “cd Downloads/Linux-tools-master/ubuntu-only”.
- Run the command “chmod +x *” to mark the scripts as executable.
- Run the command “./setup.sh” and enter your password to run the script.
- Wait for the script to finish, it may take half an hour, or more if your Internet is slow.

When typing your password into the terminal, it does not display anything, not even asterisks. Rest assured, your password is indeed being typed.

Step 9. Install graphics drivers

Open “Software & Updates” and click the “Additional Drivers” tab. Note: “Software & Updates” is a different tool from “Software Updater”.

In most cases, you’ll see your GPU and any other devices that need extra drivers. Select the latest drivers for each and then click “Apply Changes”. Then, reboot your computer.

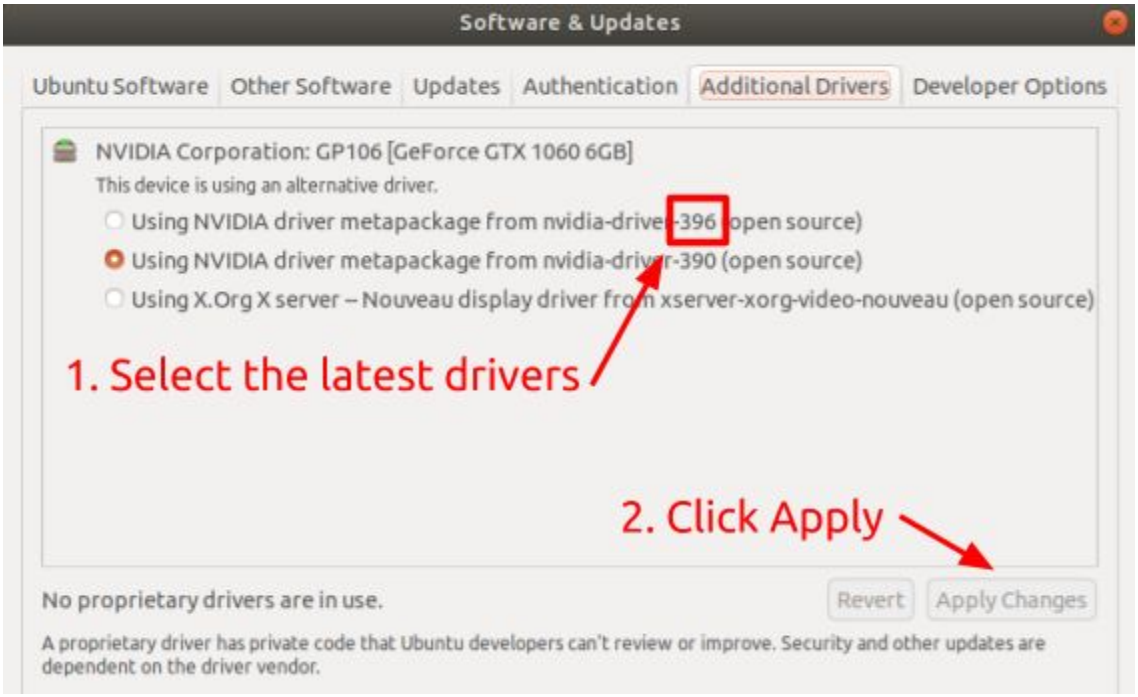


Figure 9 - Additional Drivers menu

NEVER download graphics drivers with your web browser. Drivers on Linux should always be installed with the built-in tools.

Step 10. Install Steam

Install Steam via the package manager by using the terminal.

- Open a terminal.
- Run the command “sudo apt install steam” to install Steam.
- Open Steam, and you should get the screen on the right.

If Steam does not open after installing, try repeating Step 8 and/or Step 9.

We assume that you’ve already made an account, so click “Login to an existing account” and log in. If everything works, you’ll be greeted with your Steam library!

Figure 10.1 - First time Steam login menu

You may have noticed that not every game is available to download. Only some games are made natively for Linux.

In order to play Windows games on Linux, Steam includes a tool called Steam Play. By default, only pre-approved games work. To run other games, we need to enable Steam Play for all titles.

- In Steam, click “Steam” in the top-left corner, then “Settings”.
- Click the “Steam Play” category.
- Check “Enable Steam Play for all other titles”.
- Click OK, and Steam will restart.

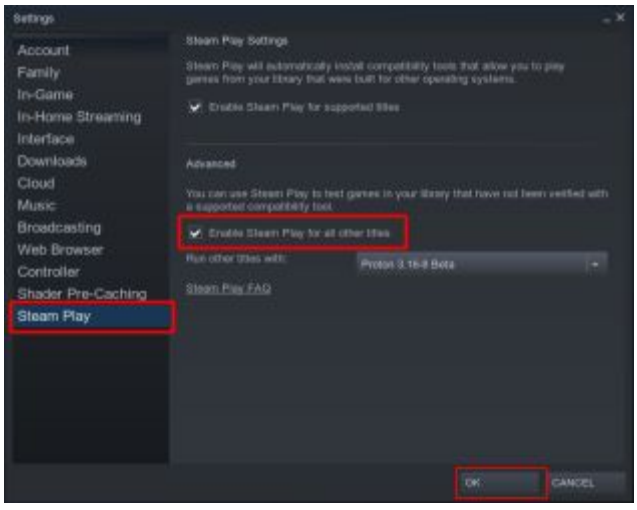


Figure 10.2 - Steam Play settings

Now, you’re done! You are now able to download a game through Steam and play it just like you would play a Steam game on Windows or Mac. We assume you know how to do this.

Conclusion

Now that you have completed this quick start guide, you have a working installation of Ubuntu that can run Steam and games.

Here are some things you should keep in mind when using Linux.

Installing software:

- On Linux, you install most software via the repositories. In most cases you will not be downloading new software from a web browser. Instead, use the “Ubuntu Software” app, or use “sudo apt install” in the terminal. For games, use Steam.
- When you do use a web browser, you should look for a .deb package.
- You can easily keep your system updated via the package manager. Simply use the built-in “Software Updater” tool to update your system.
- Use the 64-bit versions of software whenever possible. It saves disk space due to not having 32-bit versions of libraries, and 64-bit packages generally work better.

Wine and non-Linux software:

- Similarly to Macs, not all software is compatible with Linux. If there’s no Linux version, find an alternative application, or use Wine.
- Steam has a built-in custom version of Wine called Proton, preconfigured for running games. Not every game will work, but the majority do.
- To find out if a program will run in Wine, use the WineHQ AppDB.
- To find out if a game will run in Steam Play / Proton, use ProtonDB.