Brief guide of installing programs on Linux

Prerequisites: Basic Terminal Knowledge Article Type: Instructive

Installing a program on Linux can be a tedious and grueling task especially for a first time Linux user. Unlike Mac and Windows, you cannot always do a simple Google search and press the download button to get the program you want. This guide will clear up the confusion on installing programs on Ubuntu-based Linux system. Without further ado, here are the three main ways to install programs.

```

sudo apt install vlc

Reading package lists... Done

Building dependency tree

Reading state information... Done

The following additional packages will be installed:

fonts-freefont-ttf libaribb24-0 libbasicusageenvironment1 libdc1394-22 libdca0 libdvbpsi10 libebml4v5 libgroupsock8

libkate1 liblivemedia62 libmatroska6v5 libmicrodns0 libnfs11 libopenmpt-modplug1 libplacebo4 libprotobuf-lite10

libproxy-tools libqt5x11extras5 libresid-builder0c2a libsdl-image1.2 libsdl1.2debian libsidplay2 libupnp6

libusageenvironment3 libvlc-bin libvlc5 libvlccore9 libvulkan1 libxcb-xv0 vlc-bin vlc-data vlc-l10n vlc-plugin-base

vlc-plugin-notify vlc-plugin-qt vlc-plugin-samba vlc-plugin-skins2 vlc-plugin-video-output

vlc-plugin-video-splitter vlc-plugin-visualization

Suggested packages:

libdvdcss2

The following NEW packages will be installed:

fonts-freefont-ttf libaribb24-0 libbasicusageenvironment1 libdc1394-22 libdca0 libdvbpsi10 libebml4v5 libgroupsock8

libkate1 liblivemedia62 libmatroska6v5 libmicrodns0 libnfs11 libopenmpt-modplug1 libplacebo4 libprotobuf-lite10

libproxy-tools libqt5x11extras5 libresid-builder0c2a libsdl-image1.2 libsdl1.2debian libsidplay2 libupnp6

libusageenvironment3 libvlc-bin libvlc5 libvlccore9 libvulkan1 libxcb-xv0 vlc vlc-bin vlc-data vlc-l10n

vlc-plugin-base vlc-plugin-notify vlc-plugin-qt vlc-plugin-samba vlc-plugin-skins2 vlc-plugin-video-output

vlc-plugin-video-splitter vlc-plugin-visualization

0 upgraded, 41 newly installed, 0 to remove and 152 not upgraded.

Need to get 17.1 MB of archives.

After this operation, 76.6 MB of additional disk space will be used.

Do you want to continue? [Y/n]

```

1. Using APT

In Linux, a package manager is your main tool for installing new programs. Internally, a package manager is quite complex, thus I only list basics commands required to install a program. If you want to learn the package manager more, check out my tutorial here.

For Ubuntu, your package manager is apt. You need to execute two commands to install a program.

$ sudo apt update

$ sudo apt install <package/program name>

The first command is for updating your package manager. It is always recommended to update it before installing a program, especially if you haven’t updated it for a while. The second command is for installing your desired program. For example, to install firefox, run:

$ sudo apt update

$ sudo apt install firefox

Sometimes, on the program’s main website, they will ask you to add a repository. Adding a repository allows your package manager to access a larger variety of software. To install a program through a repository, use the commands:

$ sudo apt add-repository <software repository name>

$ sudo apt update

$ sudo apt install <package/program name>

2. Installing from tar, tar.gz, or tar.bz2 files

Tar files are bundles of files, like a ZIP file. To install programs from tar files, first unpack the tar file from terminal with the command:

tar xjzf <tar\_file>

This will create a copy of all the files in your tar file into a folder with the same name. Access the contents of the folder with the command *cd* <*folder\_name*>.

Usually there is a file with installation instructions, generally named some variation of *install*, *instructions* or *readme* in the program’s root or bin folder. Follow the given instructions

Most of the time you only need to run an executable to install the program (especially for tar.bz2 files). The proper executable file to run should be named a variation of *configure, configure.sh install, install.sh* or the program name itself and is located in the program root or bin folder. The file can be run using *./filename*. For example, if you are installing *eclipse*, the executable file is usually named *install*, *configure*, or *eclipse*. The executable files can be run using ./*install*, *./configure*, or *./eclipse*.

You can delete your tar file after installation.

1. Installing programs from deb packages

Some files you download from the internet may be packaged in a .deb file. To install a program from a .deb package, run the command:

$ sudo apt install ./filename.deb

You can delete your deb file after installation.

Note: If you see a .deb file, you will most likely encounter a .rpm file. A .rpm file is for a different flavor of Linux, not Ubuntu.

If you want to learn about how to configure your installations, click here.