

Distributions

- [Arch](#)
- [Debian](#)

Arch

Package Management

Pacman

First, you should check to verify your `pacman-mirrors` are configured to the nearest location. Do this manually by editing `/etc/pacman.d/mirrorlist`, or run `sudo pacman-mirrors -g` -

```
[kapper@kanjaro ~]$ sudo pacman-mirrors -g
INFO Downloading mirrors from repo.manjaro.org
::INFO User generated mirror list
::-----
::INFO Custom mirror file saved: /var/lib/pacman-mirrors/custom-mirrors.json
::INFO Using default mirror file
::INFO Querying mirrors - This may take some time
0.772 United_States : https://repo.ialab.dsu.edu/manjaro/
0.756 United_States : http://repo.ialab.dsu.edu/manjaro/
::INFO Writing mirror list
::United_States : https://repo.ialab.dsu.edu/manjaro/testing
::INFO Mirror list generated and saved to: /etc/pacman.d/mirrorlist
```

Now, you should have much faster download speeds when updating or grabbing packages.

To install a package, run `sudo pacman -Syu <package>`. For example, to install `htop`, run `sudo pacman -Syu htop`. This will not only install `htop`, but first it will check that your package list and installed packages are up to date to ensure you get the latest version.

If you are used to the `apt` package manager, this is basically like running `sudo apt update && sudo apt upgrade`, `pacman` can run these updates alongside every new package installation with the `-Syu` parameters.

Partial Upgrade Cleanup

Sometimes a run of `pacman -Syu` will complete normally, but later you may notice that certain packages were either upgraded incorrectly or not upgraded at all. One reason this may happen is a hiccup in PGP key validation by pacman during the upgrade. The commands below may help in fixing such a problem -

```
# Refresh all PGP keys installed on the system
sudo pacman-key --refresh-keys

# Reinstall all packages on the system
sudo pacman -Qqn | sudo pacman -S
```

These two commands will either print errors providing further information on the broken packages or complete and fix the broken packages. After running, you may need to reboot.

AUR Packages

AUR = arch user repository

Sometimes a package may exist within the community but not in any official repository. To manage these, we have AUR helpers.

This [list of AUR helpers](#), AKA community / AUR packages, is useful in selecting the best tool to suit your needs.

Using [yay](#), some basic commands are seen below -

```
# Search foreign package db for package
yay -q pycharm

# Will prompt for install with list of results and descriptions

# To upgrade yay alongside pacman, run the following
yay -S yay-bin
sudo pacman -Syu
yay -S yay
```

After installation, the `/opt/<PackageName>` will contain the new files created for the installed package.

Debian

Release cycles

The Ubuntu release cycle is at a glance pretty straight forward, but when on the `18.04` release and running `sudo do-release-upgrade` produces unexpected results like the below, it raises some questions.

```
Checking for a new Ubuntu release
There is no development version of an LTS available.
To upgrade to the latest non-LTS development release
set Prompt=normal in /etc/update-manager/release-upgrades.
```

Below, running `lsb_release -a` verifies our version, and looking at the release cycles on the Ubuntu website we appear to be behind on the LTS release upgrade..

```
No LSB modules are available.
Distributor ID: Ubuntu
Description:   Ubuntu 18.04.4 LTS
Release:      18.04
Codename:     bionic
```

Why wouldn't Ubuntu pick the 20.04 upgrade to install on our system? This is not by mistake, but due to the planning of Ubuntu releases. While `20.04` is a LTS release, and we *are* on the previous LTS release, `do-release-upgrade` will not detect an upgrade until Ubuntu `20.04.1` is released. This is by design, but can be overridden with `sudo do-release-upgrade -d`, which switches you to the next development release.

As the output states from `do-release-upgrade` above, we can specify within `/etc/update-manager/release-upgrades` how we want to handle the upgrades on our system, and this setting should always be considered before attempting a system upgrade. An example of the file's contents can be seen below

```
# Default behavior for the release upgrader.

[DEFAULT]

# Default prompting and upgrade behavior, valid options:
```

```
#
# never - Never check for, or allow upgrading to, a new release.
# normal - Check to see if a new release is available. If more than one new
#          release is found, the release upgrader will attempt to upgrade to
#          the supported release that immediately succeeds the
#          currently-running release.
# lts - Check to see if a new LTS release is available. The upgrader
#       will attempt to upgrade to the first LTS release available after
#       the currently-running one. Note that if this option is used and
#       the currently-running release is not itself an LTS release the
#       upgrader will assume prompt was meant to be normal.
Prompt=lts
```

Apt

The apt package manager is fairly straightforward to work with in terms of its usage and help text, so I'll leave the basics up to `apt -h` -

apt 1.6.12 (amd64)

Usage: apt [options] command

apt is a commandline package manager and provides commands for searching and managing as well as querying information about packages.

It provides the same functionality as the specialized APT tools, like apt-get and apt-cache, but enables options more suitable for interactive use by default.

Most used commands:

list - list packages based on package names

search - search in package descriptions

show - show package details

install - install packages

remove - remove packages

autoremove - Remove automatically all unused packages

update - update list of available packages

upgrade - upgrade the system by installing/upgrading packages

full-upgrade - upgrade the system by removing/installing/upgrading packages

edit-sources - edit the source information file

See `apt(8)` for more information about the available commands.
Configuration options and syntax is detailed in `apt.conf(5)`.
Information about how to configure sources can be found in `sources.list(5)`.
Package and version choices can be expressed via `apt_preferences(5)`.
Security details are available in `apt-secure(8)`.

This APT has Super Cow Powers.

If any of the above confuses you, see `man apt`

For most, the default repositories that come with ubuntu or the distro of your choice would be enough, but some may choose to add more trusted sources who may have packages or drivers that would otherwise be unsupported. These sources are generally stored in `/etc/apt/sources.list.d/` and we'll see how to back them up later.

Adding PPAs

Managing adding and removing ppas to your sources is seen below

```
# Add ppa
sudo add-apt-repository -y ppa:user/ppa

# Remove ppa
sudo add-apt-repository -r ppa:user/ppa
```

If you want to remove a ppa and all its related packages to ensure you don't create a conflict between dependencies, run the below commands

```
# Remove a ppa and its associated software
sudo ppa-purge user/ppa

# Alternatively we can use -o and -p to specify owner and ppa respectively
sudo ppa-purge -o user -p ppa
```

PPA Release Discrepancies

Sometimes, you may add a ppa and realize it is not using the same release as you, so to continue using it we will need to make some changes, below we see a 404 from adding a bionic ppa using Ubuntu focal

```
Kapper@kubuntu:~$ sudo add-apt-repository ppa:kgilmer/speed-ricer
Vanilla packages for fast ricing.

More info: https://launchpad.net/~kgilmer/+archive/ubuntu/speed-ricer

Press [ENTER] to continue or Ctrl-c to cancel adding it.
```

```
Hit:1 http://us.archive.ubuntu.com/ubuntu focal InRelease
Ign:2 http://dl.google.com/linux/chrome/deb stable InRelease
Get:3 http://us.archive.ubuntu.com/ubuntu focal-updates InRelease [107 kB]
Get:8 http://security.ubuntu.com/ubuntu focal-security InRelease [107 kB]
Hit:9 http://archive.canonical.com/ubuntu focal InRelease
Get:10 http://us.archive.ubuntu.com/ubuntu focal-backports InRelease [98.3 kB]
Ign:11 http://ppa.launchpad.net/kgilmer/speed-ricer/ubuntu focal InRelease
Err:14 http://ppa.launchpad.net/kgilmer/speed-ricer/ubuntu focal Release
      404 Not Found [IP: 91.189.95.83 80]
Reading package lists... Done
E: The repository 'http://ppa.launchpad.net/kgilmer/speed-ricer/ubuntu focal Release' does not have a Release file.
N: Updating from such a repository can't be done securely, and is therefore disabled by default.
N: See apt-secure(8) manpage for repository creation and user configuration details.
```

To fix this, we simply run `sudo vim /etc/apt/sources.list.d/kgilmer-ubuntu-speed-ricer-focal.list` and change `focal` to `bionic` in the line below

```
# Commented line below is what Ubuntu created using add-apt-repository
#deb http://ppa.launchpad.net/kgilmer/speed-ricer/ubuntu bionic main
# Change it to our release (focal) to fix the release file error described above
deb http://ppa.launchpad.net/kgilmer/speed-ricer/ubuntu focal main
```

Now, running `sudo apt-get update` should result in no 404's and you'll be able to grab and packages you were after within the ppa with `sudo apt install`.

To see your current release, run `lsb_release -a` -

```
kapper@kubuntu:~$ lsb_release -a
No LSB modules are available.
Distributor ID: Ubuntu
Description:   Ubuntu 20.04 LTS
Release:      20.04
Codename:     focal
```

Installing from another release

Even more, you could face an issue like the below, where I upgraded to focal, which at the time was a very new and fresh lts.

```
kapper@kubuntu:~$ sudo apt install polybar
```

[0/0]

```
[sudo] password for kapper:
```

```
Reading package lists... Done
```

```
Building dependency tree
```

```
Reading state information... Done
```

```
Some packages could not be installed. This may mean that you have  
requested an impossible situation or if you are using the unstable  
distribution that some required packages have not yet been created  
or been moved out of Incoming.
```

```
The following information may help to resolve the situation:
```

```
The following packages have unmet dependencies:
```

```
polybar : Depends: python-xcbgen but it is not installable
```

```
E: Unable to correct problems, you have held broken packages.
```

```
kapper@kubuntu:~$ sudo apt install python-xcbgen
```

```
Reading package lists... Done
```

```
Building dependency tree
```

```
Reading state information... Done
```

```
Package python-xcbgen is not available, but is referred to by another package.
```

```
This may mean that the package is missing, has been obsoleted, or  
is only available from another source
```

```
E: Package 'python-xcbgen' has no installation candidate
```

```
kapper@kubuntu:~$
```

A package I use, `polybar`, required some dependency that no longer exists on focal. So, since I've used this package just fine previously on bionic, I simply add the following line to my `/etc/apt/sources.list`

```
deb http://cz.archive.ubuntu.com/ubuntu bionic main universe
```

Then, we run the following

```
sudo apt update
```

```
sudo apt install -t bionic python-xcbgen
```

```
# Just to be sure, I don't want to install anything outside of focal if I don't have to. I'd rather not use polybar
```

```
sudo apt install -t focal polybar
```

To backup all current sources

To restore a backup of previous sources

Mess something up or lose your `sources.list`? See below for the default settings on various ubuntu releases

Ubuntu bionic 18.04

```
#deb cdrom:[Ubuntu 18.04 _Bionic_ - Build amd64 LIVE Binary 20190418-12:10]/ bionic main
```

```
# See http://help.ubuntu.com/community/UpgradeNotes for how to upgrade to  
# newer versions of the distribution.
```

```
deb http://us.archive.ubuntu.com/ubuntu/ bionic main restricted
```

```
# deb-src http://us.archive.ubuntu.com/ubuntu/ bionic main restricted
```

```
## Major bug fix updates produced after the final release of the  
## distribution.
```

```
deb http://us.archive.ubuntu.com/ubuntu/ bionic-updates main restricted
```

```
# deb-src http://us.archive.ubuntu.com/ubuntu/ bionic-updates main restricted
```

```
## N.B. software from this repository is ENTIRELY UNSUPPORTED by the Ubuntu  
## team. Also, please note that software in universe WILL NOT receive any  
## review or updates from the Ubuntu security team.
```

```
deb http://us.archive.ubuntu.com/ubuntu/ bionic universe
```

```
# deb-src http://us.archive.ubuntu.com/ubuntu/ bionic universe
```

```
deb http://us.archive.ubuntu.com/ubuntu/ bionic-updates universe
```

```
# deb-src http://us.archive.ubuntu.com/ubuntu/ bionic-updates universe
```

```
## N.B. software from this repository is ENTIRELY UNSUPPORTED by the Ubuntu  
## team, and may not be under a free licence. Please satisfy yourself as to  
## your rights to use the software. Also, please note that software in  
## multiverse WILL NOT receive any review or updates from the Ubuntu  
## security team.
```

```
deb http://us.archive.ubuntu.com/ubuntu/ bionic multiverse
```

```
# deb-src http://us.archive.ubuntu.com/ubuntu/ bionic multiverse
```

```
deb http://us.archive.ubuntu.com/ubuntu/ bionic-updates multiverse
```

```
# deb-src http://us.archive.ubuntu.com/ubuntu/ bionic-updates multiverse
```

```
## N.B. software from this repository may not have been tested as
## extensively as that contained in the main release, although it includes
## newer versions of some applications which may provide useful features.
## Also, please note that software in backports WILL NOT receive any review
## or updates from the Ubuntu security team.
deb http://us.archive.ubuntu.com/ubuntu/ bionic-backports main restricted universe multiverse
# deb-src http://us.archive.ubuntu.com/ubuntu/ bionic-backports main restricted universe multiverse

## Uncomment the following two lines to add software from Canonical's
## 'partner' repository.
## This software is not part of Ubuntu, but is offered by Canonical and the
## respective vendors as a service to Ubuntu users.
deb http://archive.canonical.com/ubuntu bionic partner
# deb-src http://archive.canonical.com/ubuntu bionic partner

deb http://security.ubuntu.com/ubuntu bionic-security main restricted
# deb-src http://security.ubuntu.com/ubuntu bionic-security main restricted
deb http://security.ubuntu.com/ubuntu bionic-security universe
# deb-src http://security.ubuntu.com/ubuntu bionic-security universe
deb http://security.ubuntu.com/ubuntu bionic-security multiverse
# deb-src http://security.ubuntu.com/ubuntu bionic-security multiverse
```

Ubuntu Focal Fossa 20.04

Following a `sudo do-release-upgrade -d -f DistUpgradeViewGtk3` on Ubuntu 18.04 with the option `Prompt=Its` set within `/etc/update-manager/release-upgrades` the sources are the following

```
# deb cdrom:[Ubuntu 18.04 _Bionic_ - Build amd64 LIVE Binary 20190418-12:10]/ bionic main

# See http://help.ubuntu.com/community/UpgradeNotes for how to upgrade to
# newer versions of the distribution.
deb http://us.archive.ubuntu.com/ubuntu/ focal main restricted
# deb-src http://us.archive.ubuntu.com/ubuntu/ bionic main restricted

## Major bug fix updates produced after the final release of the
## distribution.
deb http://us.archive.ubuntu.com/ubuntu/ focal-updates main restricted
# deb-src http://us.archive.ubuntu.com/ubuntu/ bionic-updates main restricted
```

N.B. software from this repository is ENTIRELY UNSUPPORTED by the Ubuntu
team. Also, please note that software in universe WILL NOT receive any
review or updates from the Ubuntu security team.

deb http://us.archive.ubuntu.com/ubuntu/ focal universe

deb-src http://us.archive.ubuntu.com/ubuntu/ bionic universe

deb http://us.archive.ubuntu.com/ubuntu/ focal-updates universe

deb-src http://us.archive.ubuntu.com/ubuntu/ bionic-updates universe

N.B. software from this repository is ENTIRELY UNSUPPORTED by the Ubuntu
team, and may not be under a free licence. Please satisfy yourself as to
your rights to use the software. Also, please note that software in
multiverse WILL NOT receive any review or updates from the Ubuntu
security team.

deb http://us.archive.ubuntu.com/ubuntu/ focal multiverse

deb-src http://us.archive.ubuntu.com/ubuntu/ bionic multiverse

deb http://us.archive.ubuntu.com/ubuntu/ focal-updates multiverse

deb-src http://us.archive.ubuntu.com/ubuntu/ bionic-updates multiverse

N.B. software from this repository may not have been tested as
extensively as that contained in the main release, although it includes
newer versions of some applications which may provide useful features.
Also, please note that software in backports WILL NOT receive any review
or updates from the Ubuntu security team.

deb http://us.archive.ubuntu.com/ubuntu/ focal-backports main restricted universe multiverse

deb-src http://us.archive.ubuntu.com/ubuntu/ bionic-backports main restricted universe multiverse

Uncomment the following two lines to add software from Canonical's
'partner' repository.

This software is not part of Ubuntu, but is offered by Canonical and the
respective vendors as a service to Ubuntu users.

deb http://archive.canonical.com/ubuntu focal partner

deb-src http://archive.canonical.com/ubuntu bionic partner

deb http://security.ubuntu.com/ubuntu focal-security main restricted

deb-src http://security.ubuntu.com/ubuntu bionic-security main restricted

deb http://security.ubuntu.com/ubuntu focal-security universe

deb-src http://security.ubuntu.com/ubuntu bionic-security universe

deb http://security.ubuntu.com/ubuntu focal-security multiverse

deb-src http://security.ubuntu.com/ubuntu bionic-security multiverse

To change default terminal emulator

```
sudo update-alternatives --config x-terminal-emulator
```

```
# Backup gnome tweaks and settings
```

```
cd ~
```

```
dconf dump / > saved_settings.dconf
```

```
# Restore your gnome settings
```

```
cd ~
```

```
dconf load / < saved_settings.dconf
```