

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.

Revision #1

Created 20 May 2020 00:31:07 by Shaun Reed

Updated 18 December 2021 16:37:39 by Shaun Reed