

Arch Linux Package Management

Prof. Rossano Pablo Pinto

v0.9 – Feb 2019
(under construction)

Agenda

- TODO

Introduction

- TODO

Important dirs / files

- `/var/lib/pacman/`
 - **DATABASE** with information about packages
 - `local/$package-$version/files` : **list of files contained in package** `$package-$version`
- `/var/cache/pacman/pkg/`
 - **PACKAGES** themselves
- `/etc/pacman.conf`
 - **main pacman configuration file**

Common operations

- `pacman -S package`
 - Installs the *package*. Example: `pacman -S vim`
- `pacman -Syu` (upgrade all packages)
- `pacman -Syyu` (Force pacman to refresh the package lists before upgrade)
- `pacman -Q`
 - Lists all the installed packages (`-Qe` explicit installed packages - omits dependencies, `-Qqe` omits version, etc..)
- `pacman -Ss expression`
 - Finds packages that contains the expression in their names or description.
Example: `pacman -Ss apache`

Common operations

- `pacman -Ql package`
 - Lists all files that are part of an installed *package*
- `pkgfile file`
 - Lists the package that owns file “*file*”
- `pkgfile -l package`
 - Lists all files provided by package “*package*”
- `pacman -Sgg`
 - Lists group and all files that is part of that group

Common operations

- `pacman -Qo file-path`
 - Finds the package that owns that file. Example:
`pacman -Qo /etc/fstab`
- `pacman -R package`
 - Removes the *package*. Example: `pacman -R vim`
- `pacman -Rc package`
 - Removes the *package* and all its dependencies (DANGEROUS). Example: `pacman -Rc vim`

Common operations

- Omitting some packages when upgrading
 - I will avoid the upgrade of the linux kernel as an example:

```
sed -i 's/#IgnorePkg    =/IgnorePkg    = linux/g'  
/etc/pacman.conf
```

- See pacman.conf documentation:

```
man pacman.conf
```


Using a local (private) mirror

- Server (192.168.200.198)
- Client
 - put the following line before any other server at `/etc/pacman.d/mirrorlist`

```
Server = http://192.168.200.198/repositorio/archlinux/$repo/os/$arch
```

Sharing a cache dir as a repo

- **Server (192.168.90.43)**

```
pacman -S darkhttpd
```

```
darkhttpd /var/cache/pacman/pkg
```

- **Client**

- put the following line before any other server at
`/etc/pacman.d/mirrorlist`

```
Server = http://192.168.90.43
```

Replicating a install

- At the source machine

```
pacman -Qqe > pkglist.txt
```

- transfer the file `pkglist.txt` to the destination machine

- At the destination machine

```
pacman -S - < pkglist.txt
```

Speeding up

- Reflector
 - script that can retrieve the latest (and fastest) list of mirrors from MirrorStatus page.
- Rank mirrorlist by speed
 - `cp /etc/pacman.d/mirrorlist \`
`/etc/pacman.d/mirrorlist.backup`
 - `rankmirror -n 10 \`
`/etc/pacman.d/mirrorlist.backup > \`
`/etc/pacman.d/mirrorlist`

Using AUR

- AUR (Arch User Repository) - Example: chrome
 - `git clone https://aur.archlinux.org/google-chrome.git`
 - `cd google-chrome`
 - `makepkg -s`
 - `sudo pacman -U google-72.0.3626.81-1x86_64.pkg.tar.xz`

Refreshing keys

- This is used to correct invalid GPG keys when GPG signatures appears as corrupt or invalid...
 - `pacman -S archlinux-keyring`
OU
 - `pacman-key --refresh-keys`