apk Alpine package manager

CONCEPTS

apk keeps a world file (/etc/apk/world) of what you asked for; everything else is a dependency and goes away with it.

Repositories /etc/apk/repositories
↓ `apk update`
Local index package list
↓ `apk add`
World file wanted packages
↓ `resolve`
Installed world + deps

INSTALL AND REMOVE

apk add *<pkg>*
Install a package

apk add *<pkg1> <pkg2>*
Install several packages

apk add --no-cache *<pkg>*
Install, skip the local index cache

apk del *<pkg>*
Remove a package and unused deps

apk del -r *<pkg>*
Also remove packages depending on it

apk add -t *<name> <pkg>...*
Group packages as a virtual package

apk del *<name>*
Remove a virtual package and its deps

apk add *<pkg>=<version>*
Install a specific version

apk add *<pkg>@<tag>*
Install from a tagged repository

apk add -u *<pkg>*
Upgrade a package if installed

apk add cmd:*<command>*
Install whichever package provides a command

apk add -s *<pkg>*
Simulate an install

UPDATE AND UPGRADE

apk update
Refresh the package index

apk upgrade
Upgrade all installed packages

UPDATE AND UPGRADE (cont.)

apk update && apk upgrade
Refresh, then upgrade

apk upgrade -s
Preview an upgrade

apk upgrade --available
Sync to repos after a release switch

apk upgrade *<pkg>*
Upgrade only this package

apk list -u
List upgradable packages

apk fix
Repair or reinstall broken packages

SEARCH AND INSPECT

apk search *<name>*
Search package names

apk search -e *<name>*
Search for an exact name

apk search -d *<text>*
Search descriptions

apk search -v *<name>*
Show versions and descriptions

apk search cmd:*<command>*
Find the package providing a command

apk info *<pkg>*
Show package details

apk info
List installed packages

apk list -I
List installed packages with versions

apk info -L *<pkg>*
List files in a package

apk info -W *<path>*
Find the package owning a file

apk info -e *<pkg>*
Check whether a package is installed

apk info -R *<pkg>*
List a package's dependencies

apk info -r *<pkg>*
List packages depending on it

apk policy *<pkg>*
Show available versions and repos

REPOSITORIES

cat /etc/apk/repositories
Show configured repositories

apk add -X *<url> <pkg>*
Add a repository for one command

Repositories are main, community and testing; testing is edge-only. Use a @tag prefix to opt in per package.

repositories line

Part	Example
`Plain`	https://dl-cdn.alpinelinux.org/alpine/v3.20/main
`Community`	https://dl-cdn.alpinelinux.org/alpine/v3.20/community
`Edge testing`	@testing https://dl-cdn.alpinelinux.org/alpine/edge/testing
`Local path`	/var/apk/my-packages

CACHE AND WORLD

apk cache clean
Delete outdated cached packages

apk fetch *<pkg>*
Download a package without installing

cat /etc/apk/world
Show explicitly installed packages

update-conf
Merge .apk-new config files

RECIPES

Dockerfile install

```
FROM alpine:3.20
RUN apk add --no-cache curl ca-certificates
```

Temporary build dependencies

```
apk add --no-cache -t .build-deps build-base
# compile something here
apk del .build-deps
```

Pin a testing package

```
echo '@testing https://dl-cdn.alpinelinux.org/alpine/edge/testing' \
  >> /etc/apk/repositories
apk add <pkg>@testing
```

Who provides this file?

```
apk info -W /usr/bin/<cmd>
apk search cmd:<cmd>
```