

What is pkg on FreeBSD?

REVISION HISTORY			
-------------------------	--	--	--

NUMBER	DATE	DESCRIPTION	NAME
	2014-12-26		WB

© 2014 Warren Block

Last updated 2014-12-26

Available in [HTML](#) or [PDF](#). Links to all my articles [here](#). Created with [AsciiDoc](#).

1. *What is pkg or pkgng?*

The new package management system on FreeBSD. pkg is the preferred name now.

2. *Should I use pkg?*

Yes. In fact, if you want to keep using binary packages or ports built after September 2014, it is required.

3. *How can I tell which package management system is installed?*

FreeBSD 10.1 and later do not include the old package management system. On earlier systems, running `pkg info` shows a list of packages if pkg has been installed, or offers to install the new package management system if it is not present.

4. *If I switch, can ports still be used?*

Of course. When a port is compiled, it actually creates a package. The package is then installed and managed by the package management system. A binary package is the same, it has just been compiled on a different computer.

5. *How can I switch to pkg?*

- `portmaster ports-mgmt/pkg` to install the port.
- `pkg2ng` to read the old package database and write the new one. Do that conversion only once.

6. *After switching to pkg, can the old pkg_* commands still be used?*

No, do not use the old `pkg_*` commands any more. The database they use is likely to be outdated or not even present. The new pkg commands are similar, but the pkg command is followed by a space and a subcommand, like `pkg info`.

7. *Can the old package database files be removed?*

The pkg database is a SQLite file called `local.sqlite` in `/var/db/pkg/`. The old package database consisted of a subdirectory for each package in this same directory. All of those old subdirectories can be deleted, although `portmaster` uses them to keep track of things and will recreate them.

8. *What is Poudriere?*

An automated system that builds binary packages out of ports. This is what the FreeBSD cluster systems use to build binary packages. It can also be used by individuals to build their own customized binary packages. If several machines need the same applications, one system can be used to build those binary packages from ports, and then all of them can quickly install the resulting binary packages.

9. *Is Poudriere required to build from ports?*

No. The standard `make build/install/clean` ports commands work fine. Poudriere just makes it possible to build binary packages from ports in bulk.

10. *Should I use ports or binary packages?*

It depends. Binary packages are convenient because they require no compiling. However, they do not allow customized options. For an example, consider the many options of the `www/nginx` port. In a binary package, these are set to defaults and cannot be changed. Ports must be compiled, but can be customized and optimized for an individual system or use. If you support multiple machines, Poudriere makes it possible to build customized binary packages for them, requiring no build time on the target system.

11. *When will FreeBSD drop ports altogether?*

Binary packages are created from ports. So the answer to this question is "probably never". However, a computer that uses only binary packages does not need a copy of the ports tree.