

# PORTSreport

BY THOMAS ABTHORPE

Ports Report is a summary of recent activity in the Ports infrastructure, news in the world of Ports, tips and tricks.

## NEW PORTS COMMITTERS

**A**s a dedicated contributor to the Ports Tree, other committers sit up, take notice and opt to propose you for a commit bit so they won't have to do your work for you anymore [sic]. Our most recent addition to the ranks of Ports Committers is Jost Meixner. One of Jost's many interests is maintaining Linux ports for FreeBSD.

## NEWEST MEMBERS TO portmgr-lurkers

**W**ith the ongoing success of the portmgr-lurker project, we have started the second intake of lurkers to learn, observe, and contribute to the Ports Management

Team. In March, Alexy (danfe@) Dokuchaev and Frederic (culot@) Culot commenced their duties. Frederic is also doing double duty, shadowing the portmgr-secretary and helping with the duties of that position.

## THE SECOND BRANCH OF THE PORTS TREE

**B**ecause the first—2014Q1—branch was experimental, you might not have heard of it as yet. January 2014 saw the release of the first quarterly branch, intended to provide a stable and high-quality ports tree. Those stable branches are snapshots of the head ports tree taken every three months and currently supported for three months,

## DOING YOUR PART TO IMPROVE THE PORTS TREE

**C**ompanion tools to the build systems of ports-mgmt/tinderbox and ports-mgmt/poudriere mentioned above are ports-mgmt/porttools. With these tools, you can use it to create a new port, file a PR for an update via send-pr(1), or even use it as a poor man's build system by issuing the command port test. You can read more at <http://www.freebsd.org/doc/en/books/porters-handbook/testing-porttools.html>.

When you install the tools, you get another great porter's tool called portlint(1). Just as lint(1) aids you in getting the fluff out of your C program, portlint uses heuristics to assist you in finding errant whitespace, incorrectly placed directives, along with a host of other tips and suggestions to improve your port. For instance issuing

```
portlint -C /usr/ports/devel/fakeport # this is an example only
```

might yield something like

```
WARN: Makefile: [14]: possible direct use of command "env" found. use ${SETENV} instead.
```

```
WARN: Makefile: only one MASTER_SITE configured. Consider adding additional mirrors.
```

```
WARN: Makefile: "RUN_DEPENDS" has to appear earlier.
```

```
WARN: Consider to set DEVELOPER=yes in /etc/make.conf
```

```
0 fatal errors and 5 warnings found.
```

So I take the advice and add DEVELOPER=yes to /etc/make.conf. This, in itself, does not aid portlint, but gives some developer-directed verbose output that is useful when you build a port.

The other warnings need to be scrutinized by you, the porter, to see if additional action needs to be taken. Just remember we control portlint, it does not control us. Its warnings are intended as a guideline only.

**T**he port tree is a collection of value-added software for FreeBSD.

It is a collaborative effort by contributors around the world, and each one has done a little something to make it just a little better. A big "Thank You" to all who have helped out.

- <http://fb.me/portmgr> — "Like" us
- [http://twitter.com/freebsd\\_portmgr](http://twitter.com/freebsd_portmgr) — Follow us
- <http://blogs.freebsdish.org/portmgr/> — Our blog
- <https://plus.google.com/u/0/communities/108335846196454338383> — G+1 us

during which they receive security fixes as well as build and runtime fixes. Packages are built on a regular basis on that branch (weekly) and published as usual via [pkg.FreeBSD.org](http://pkg.FreeBSD.org) (/quarterly, instead of the usual, /latest). On April 1 (no joke), the 2014Q2 branch was created and the first builds from it began shortly thereafter.

## TIPS FOR PROSPECTIVE PORTERS

Use a (semi)-automated build system to test your ports. In a previous issue, we recommend ed subscribing to <http://red-ports.org> to test your ports. It is available for all to use for the common good. Some people may have the necessary hardware resources to build their own build system. If you are one of them, then there are some tools in the trees you can try. The original build system is the venerable Tinderbox, found in the tree as `ports-mgmt/tinderbox`, or its bleeding edge counterpart, `ports-mgmt/tinderbox-devel`. You can read more about it at their website <http://tinderbox.marcuscom.com/>. In recent years, a newer build system came into being called Poudriere, which loosely translated is simply French for tinderbox. It can be found in the ports tree as `ports-mgmt/poudriere` along with its bleeding-edge version `ports-mgmt/poudriere-devel`. You can read more about it at their website, <http://fossil.etoilebsd.net/poudriere>. The poudriere build system is now the foundation for the Ports Management Team to do -exp runs and package building.

Chose a build system that works for you, use it to your advantage to test build ports, verify that they install and uninstall cleanly, and even set up your own private packaging system.

Let **FreeBSD Journal** connect you with a targeted audience!

**Advertise Here**

**CLIMB  
WITH US!**

→ **LOOKING  
for qualified  
job applicants?**

→ **SELLING  
products  
or services?**

**Email**

**[walter@freebsdjournal.com](mailto:walter@freebsdjournal.com)**

**OR CALL**

**888/290-9469**

