WeGet letters

by Michael W Lucas

> Dear Why Do They Let You Write This Column,
>
> FreeBSD 15 is now out, and it includes the base system as packages. In the January/February 2022 issue of the *FreeBSD Journal*, you spent your entire column ranting about pkgbase. You were wrong. Obviously wrong. Is that enough to finally make you shut up and quit answering our letters?
>
> —Wants Useful Answers

Dear Wanna-Play-Gotcha,

Getting me to shut up is a solved problem. It costs $200 per hour (https://givebutter.com/c/Penguicon24/auction/items/258146). And why do they let me write this? Because John Baldwin, *FreeBSD Journal* editor, was desperate for a letters column. Now that he has one, he's even more desperate, but for completely different reasons. A change is as good as a rest.

Now on to your real question. Packaged base. What's up? Why?

I did not spend the "entire column ranting about pkgbase." Yes, I said the only true way to upgrade BSD was to build it from source and install it yourself. I also said that packaged base was "the dread dragon of FreeBSD, devouring every developer who sets out to conquer it."

Dragons fill vital ecological space. They're warnings. They're limits: *You shall go this far, and no further.* We call them evil only because people hate limits. When we encounter a limit or a dragon, we try to slay it. We still remember Saint George slaying an innocent dragon to save a princess. We remember because royalty keeps spreading the tale in the hope that we'll rescue them from the next dragon, all while neglecting to remind us that the king thought that letting dragons devour people was regrettable but acceptable until his own daughter landed on the menu.

Intelligence has very little connection to the real world and provides zero protection against peer pressure.

Anyway. Slay the dragon, and you'll go down in history.

The real problem here? Developers, like most "technical" people, are quite intelligent. Intelligence has very little connection to the real world and provides zero protection against peer pressure.

"Self-hosting," the ability to build a system from code and tools on the system, was integral to primordial BSD and remains critical today. A BSD system should ship with all the source code and tools needed to rebuild itself. The natural state of a BSD system is a single, whole entity. Breaking it up into pieces is like selling apples by the slice. Packaging sure *feels* convenient, though. The cool penguin kids have base system packaging! I surrendered any hope of being cool shortly after being born, so peer pressure triggers mere ennui.

Some folks found building FreeBSD from source code "slow" and "annoying." It might appear so to the unenlightened, but it's all about technical correctness. Correctness is a primary goal of every BSD, and watching the compiler churn as it recursively rebuilds the compiler for the eleventy-ninth time is a minor price to pay to achieve such pinnacles of purity. Technical correctness is not the worst kind of correctness. It merely feels that way.

In 2006, Colin Percival released freebsd-update(8) to allow the impatient to bypass this rite of passage, leading to wider adoption and deployment of FreeBSD.

FreeBSD also has the ports system. Ports allow the sysadmin to build add-on software exactly as desired and bundle it up in convenient tarballs so you can install them anywhere. The thirty-four thousand plus add-on programs in the ports collection supports a tangle of options and dependencies that nicely illustrates exactly why we build BSD as a monolithic whole, and the package management software needed to cope with all these interactions. The modern package management tools in pkg(8) were first included in FreeBSD 9.1 at the end of 2012.

In 2006, Colin Percival released freebsd-update(8) to allow the impatient to bypass this rite of passage, leading to wider adoption and deployment of FreeBSD.

No sooner had the new package tools been launched than folks obsessed with mere "usability" ignored the Unix philosophy of "many small tools that each handle one task" and started babbling about the convenience of using a single tool to manage both the base system and add-on software. How hard could it be? After all, bloated text editors like Emacs and vim are also built into a whole bunch of files. They could just tar up the whole base system, give it a packing list, and call it done?

Well, no.

Innumerable developers succumbed to the temptation to wrangle FreeBSD into friendly packages until a public call for tests went out in 2016. As you might expect, that's when everything went wrong. If you damaged your installation and wanted to remove all packages and start over, you would lose your operating system. That detail got promptly fixed, but it turned out there are dozens and dozens of edges and hundreds of interactions between them.

That original call for testing claimed that FreeBSD 11 would ship with a packaged base system. That slipped release after release, as lingering bugs were found and fixed.

Like all volunteer software projects, pkgbase hung at 99% complete for years.

FreeBSD prides itself on its open management, but one person in the project wields phenomenal cosmic power: the release engineer. The FreeBSD 15 release engineer put his foot down and said, "We are not only packaging the base system for 15, I will delay the release until the base system is packaged." People flinched but got to work. The pkgbaseify tool converts FreeBSD 14 hosts to use pkgbase so they can be easily upgraded to FreeBSD 15 with the packaging tools.

After all this testing, pkgbase should work for most people. I have no doubt that somebody with an excess of cleverness will use it to render their hosts unbootable or install Rails as their new kernel, but users are responsible for their own ideas, and I have a big bucket of popcorn ready.

A clean install of FreeBSD 15 on my crashbox shows 311 packages, all but one with a name beginning with "FreeBSD." The exception is, of course, "pkg." Frankly, if you delete the FreeBSD-kernel-generic package on a production server, you deserve what you get, which is a warning that you can't delete the kernel. Or libc, or the linker, or any of the other obvious candidates. You can remove the compiler but, as I said in the original pkgbase discussion, "real operating systems ship with fully functional compilers in the default install." I also offered choice words for folks who suffer from a morbid fear of compilers.

A transition such as this merits a deeper look and a consideration of historical context. Nineteen years ago, Colin released freebsd-update out of a misguided impulse to simplify upgrading and patching. The release engineer who held FreeBSD 15 hostage until it included a packaged base system?

Colin Percival.

I think I see the *real* problem…

Have a question for Michael?
Send it to letters@freebsdjournal.org

MICHAEL W LUCAS iis the author of *Absolute FreeBSD*, *Dear Abyss*, *SSH Mastery*, and too many other books. Learn more at https://mwl.io.