



Dear Wise Person Who Somehow Got Shackled
Into Answering Letters For the Journal,

People keep telling me how amazing the ports and packages system is, but I keep finding myself having to build my own software instead of using packages. It's a pain. How do they get away with this? Can't you show people that ports and packages aren't as great as people think?

Thank you,
Annoyed Compiler

Dear Annoyed,

Everything is terrible. That's the core principle of systems administration. Well, maybe not *the* core principle, but it's certainly sysadmin rule number three. And it applies doubly so to packaging systems.

We wander between packaging systems toting a long list of requirements, like automatic dependency installation and upgrades and support for our environment's LDAP, YP, KerberosIV and KerberosV authentication system, not to mention that some sysadmin back in 1989 declared that the organization's official filesystem was a FAT16 release candidate that he bootlegged out of the Redmond development lab by clenching the backup tape between his mighty buttocks, and you've endured that decision ever since because that sysadmin got promoted to Chief Albatross Officer and never updated his skills again. And that doesn't even go into the boss's fondness for writing monstrously bloated checks to outside vendors while whittling employee paychecks anytime she gets bored.

Or perhaps you have a technological green field. You get to architect everything from the ground up and have temporarily deluded yourself into believing it will be glorious, flawless, and a joy both to deploy and operate. Of course, you can't make it quite dead-standard. You want to use some special feature everywhere. A feature that will make your environment perfect for you and intolerable to anyone else, as is your privilege as a founder.

In short: your requirements are unique in precisely the same way that no two gangrenous spleens are identical.

Regardless of your operating system, any packaging system is gleefully and maliciously guilty of catering to the Least Common Denominator. Package system designers have the same goal as any other person in history, which is to make people leave them alone so they can get on with what they want to do with the minimum of fuss. This devolves to solving

as many problems as possible for as many people as possible. Those of us with special requirements—and I'm one of them, you have no *idea* just how special my requirements are—are left banging on the door trying to get in. At least fancy nightclubs have bouncers to snootily inform you you're not suitable for their packaging system, but they've also shared around my photo captioned NOPE, and if that isn't discrimination I don't know what is, but still, I much prefer packaging systems that bluntly strong-arm you out of the queue and into the conveniently accessible and by some strange chance unusually dark alley, behind the dumpster, to use their professionally honed skills of targeted violence to explain that you're not welcome. Again.

Worse, none of this software was ever meant to work together. We've got MariaDB and Postgres and Apache and nginx and lighttpd and Rail Road Ruby or whatever they call it, where someone had an idea and just *had* to go implement it, never thinking of the innumerate man-hours of collateral damage they'd be inflicting on society by trying to make software better. It's not that any of these ideas were bad (sure, okay, except systemd and the entirety of Oracle, granted), but people keep getting this half-witted hope that maybe software doesn't have to be terrible.

Hope exists to teach young people that there is no hope.

You, of course, can't help but make it worse.

You've been handed a screaming mess of an environment built around an original 286 running Antediluvian Netware and held together with a glue composed of used tea bags and pureed slugs. Every bad decision any of your predecessors have ever made haunts you. So you look at this and decide what you need is another layer of sun-dried wombat leavings over it all. You can call it "rationalizing." That sounds good. And you made your decision rationally. Because you're a rational person—

Ahem. Excuse me. I seem to have coughed up a kidney laughing there.

Because you've come up with self-justifications for your prejudices and called them rationality. That's better.

So you need packages that support multiple versions of protocols, and a filesystem nobody else uses, and probably SNMP, because your bed of nails has gotten dull from use and you don't want to escalate to autotrepanation for your agony fix.

You want to take stuff that was never meant to work together, and have it behave transparently.

The shred of good news is, people have done the work before you.

Other people have decided that MariaDB and Postgres should simultaneously integrate with nginx, or Perl, or who knows what. Certain websites that shall remain nameless but that rhyme with "Whack Derange" overflow with dubious advice toward achieving your nebulously visualized totally-not-a-nightmare-I-promise dream.

And again, I'm just as guilty. I just finished writing a book about the Shoggothic Nightmare Misery Protocol, SNMP. My dire research demanded I

build net-snmp with features that I was fairly certain no human being had ever before attempted. By that sad point I could already see into dimension πr^2 and I knew the secret name of the so-called "squirrel" who occasionally dangles on my office window screen and shrieks his Ode of Eternal Enmity, so I went to the ports system determined to carve my unspeakable path. It turns out that Eternal Madness was already a build option. (They call it "embedded Perl" and "MFD rewrites," but it's eternal madness. You don't have to trust me. Those of us who Know Too Much To Ever Be Happy Again would welcome your company.)

No matter what combination of features you need, there's a really good chance someone has done it before.

And that's why the ports and packages system is so valuable. It allows you to easily repeat mistakes, both others' and your own. At scale. If you've decided that your organization is going to glue universal authentication to that Netware server via RADIUS, you can build your own package repository that globally enables Radius in everything that supports it. You use the packaging system to distribute your mistakes throughout your little slice of the world. And nobody can stop you.

So, Dear Letter Writer, you are absolutely correct. The ports and packages system is terrible. But only because everything is terrible. I would encourage you to spend some time learning how it works, only so that you can most quickly deploy your innovative new layer over your organization's infrastructure and earn your successor's undying and well-justified loathing. ●

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

Michael W Lucas (<https://mwl.io>)'s newest books are *Sudo Mastery, 2nd Edition* and *Terrapin Sky Tango*.

Write For Us!

Contact Jim Maurer
with your article ideas.

(jmaurer@freebsdjournal.com)

