WeGetletters

by Michael W Lucas

Dear Last Desperate Chance,

I've been round and round with the boss, explaining over and over that systems administration is an art as much as a craft and I can't write a complete procedure for every last thing I do. He's got a copy of the Policies and Procedures manual from his previous job at the StarBux Coffee Hole and says that figuring out bad ARP caches can't be nearly as bad as being a barista, and that's documented down to exactly how to make the foam in the top look like the corporate logo and how covering up the cup's copyright notice is a termination offense. I've tried everything else, so now I'm trying you. Please, give me an Argument From Authority that declares documenting systems administration is doomed to fail.

—Sick of Unreasonable Requests

Dear SOUR,

"Documenting systems administration is doomed to fail." See, I can lie with the best of them.

You did quite well in waiting to contact me until the very end. Unfortunately, you desecrated that immaculate record by contacting me. It's this failure your coworkers, family, and the random strangers reading this column will remember you for. But today, it's either correct your ignorance or finish writing up the apology letter my settlement with the Avocado Liberation Front demands, so I'll give it a stab.

I'm fairly certain that you don't even know what your job is. Yes, you received a farcical document when you started that said things like "install the software" and "debug PHPython" and that oh so precious "other duties as events warrant" meaning that the boss can drop a mountain of what he's been told is web server load balancer droppings on your desk and tell you to "grope" it for salvageable SMPT headers. You'll probably react by declaring to those entities unlucky enough to live with you that your boss is an idiot who doesn't even know how to spell SMTP rather than the far more productive process of determining exactly who on the network team dared displease you and how to best demonstrate the distinctly discomforting consequences of doing so upon them and anyone within smelling distance of their cubicle.

None of this is your job, mind you. It's simply a prerequisite to doing your job. Your job? Your real job? The thing you're paid to do? It has nothing to do with system administration.

Your job is to make your boss happy.

Not your employer. Not the company.

Your boss. Your immediate supervisor.

That's it. That's the whole job. You were hired to make him happy--in a computery way.

Sure, he'll disguise it behind fancy lingo like *stockholder value* and *delight customers* and *FIPS compliance*, but it's all about making him happy. He exists to make his boss happy, and so on. A business is a tree of boot-kissing, like a TLS Chain of Trust but even more malignant.

Your boss doesn't truly want a manual on how to use ls(1). If he insists he does, make him get out his crowbar and pry open his wallet to pay for a copy of *Nemeth's Unix and Linux System Administration Handbook*. What he wants is a great big teddy binder that he can cuddle and show off to his boss. He exists to make his boss happy, after all.

So, give him what he wants, not what he asked for.

Start with a wiki. You young punks like wikis. I don't know why you can't be bothered to learn Docbook and SGML and just pretend to be a competent worthwhile person, but if I concern myself with your lack of character, this column will go on far too long, and if I don't get that inane apology letter in the post, the judge will hold me in contempt again. My attorney insists that reaching an even dozen citations will not make me go up a level when that's clearly untrue.

So. A wiki. Or a Markdown. One of them.

Pick the most tedious task you perform—say, installing software on a server. The first time you make a server reach across the Internet and grab software and install it all on its own you might feel a frisson of wonder, but as a professional sysadmin, you're too aware of all the times a simple install plunged you into the infernal abyss. Today you type pkg install fubar and watch as the package tools update the repository and search for incompatibilities and meticulously trash your LDAP database. Cast back your mind to the days when you cared about your job—yes, I know it's difficult to dig that far back, and recalling that chipper youth who was going to change the world threatens your carefully maintained shell of indifference, but that brittle shell needs substantial reinforcement and you won't develop such without fierce practice. It won't be sufficiently robust until anyone who dares poke it by asking you an innocent question gets drenched in bitter torrents of bile.

If you cared, you'd back up the host before installing anything on it. Maybe not the whole host. User home directories can burn and die, of course, because the peasantry has been told not to trust computers with anything important, but the software configuration files and data files and all those things that you're responsible for, sure, they should get backed up. Or snapshotted, or tarsnapped, or microengraved onto mysterious three-sided steel monoliths and erected in the Utah desert as a monument to all the disaster recovery plans that never got acted on because the hurricanes and avalanches were so inconsiderate as to skip the party.

So, scribble "backup" on a piece of scrap paper.

Not legibly, mind you. Just clear enough that the sight of the scrawl makes you think backup, but not plainly enough to make the housekeeper emptying your trash bin think you're considering backups. Let's say that's all you can think of. Thinking is a skill like any other, and you can improve if you keep practicing.

Maybe all the backup you need is a boot environment. Boot environments are free so long as you don't churn your data. Everybody likes free. So, write on your wiki.

Installing Software

1. Create a boot environment

2. Run pkg install whatever

Now comes the selfish bit. Your job is to make your boss happy, but that's not your *goal*. The true goal of system administration is to minimize sysadmin suffering (Sysadmin Rule #5). Minimizing sysadmin suffering demands consistency. Consistency means scripting. Sysadmins like to script.

So, write a script for installing software the way you want it installed.

Add a note at the bottom of your wiki that says, *This procedure is implemented as breakeverything.sh.*

As you slog through the muck of making your boss happy and crafting an unusual web page that will have amusing affects on the load balancer and give the network folks their due cardiac tremors, maybe you'll hit a problem that leads to a troublesome library on a host. Was that library there last week, before you ran the software install? You trudge through boot environments and find out. A list of what software was installed on a host before you installed a package would reduce your suffering, though. Add that to your procedure, and your script. Yes, this bears a suspicious resemblance to programmers having to document their code. Procedures are programs.

The next time your boss brings up the documentation thing, print out your wiki and hand it to him.

You need more procedures? Well, what other scripts have you written to make your life easier?

Bleed out documentation quickly enough to content the boss, but not so quickly as to make him jaded. He'll be happiest if he sincerely believes you work really hard on the tasks he assigns.

Keep it up long enough, and you'll be able to hand your job to some optimistic newcomer and get a new job, where you get an entire team of people who don't yet understand that their job is to make you happy.

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

MICHAEL W LUCAS's most recent books include *SNMP Mastery*, *Cash Flow for Creators*, and *Drinking Heavy Water*, plus a bunch more at https://mwl.io. Under no circumstances is he allowed near users.