WeGet letters

by Michael W Lucas

> Dear Letters Columnist,
>
> The boss says "everything must be secure." I started making a list of things to check for security, and there's no way I can do all this. What do I do?
>
> Thank you,
>
> —They'll Blame Me
> for Ransomware

Oh, TBMfR, my sweet summer child.

You hit the "rant" button. Buckle in.

I've previously written in this very column about how the word "firewall" means nothing. The term is void, without clarity or purpose. The F-word should be removed from your vocabulary immediately, by armed force if necessary, and replaced by a more specific term that means… something. Anything.

"Security?" It's like that, but even more appalling.

I will readily concede that out in meatspace, these eight distasteful letters have a role. How would you know which people to avoid without the phrase *security guard*? Yes, yes, *authoritarian goon* could serve in its place, but it doesn't precisely roll off the tongue. Social Security? That's a thing. But how do these relate to computing?

As always in these cases, I reach for the Single Source of Linguistic Truth: my Oxford English Dictionary, from that delightful Edwardian era best known as World War Intermission. Computers were people then and understood instructions like "lock up the cipher's secret keys at the end of your shift." We didn't have to define locks, or secrets, or ciphers in sufficient detail that a machine designed to the highest standards of malicious obedience couldn't misunderstand them. Security meant "do it right or the authoritarian goons will smack you until you do." So, let's go to the official definition of this word.

Wait. The official definition fills most of page 370 and spills over onto 371. There's no way I'll quote all that. I'll skim and cherry-pick some definitions that conveniently support my argument.

1. "The condition of being secure."

Defining a word with its own root? That's nearly as helpful as the documentation helpdesk staff give users. Moving on.

2. "The condition of being protected from or not exposed to danger; safety."

Here's my question: does the boss want the staff computers protected from danger, or the staff protected from the dangers of computers? Don't you dare try to tell me that computers don't threaten people; I've seen YouTube, and don't get me started on Myspace or Facestagram or whatever they call it these days.

3. **"Freedom from doubt; confidence, assurance. Now chiefly, well-founded confidence, certainty."**

Computers are not only doubt incarnate, they are unapologetic doubt factories, spewing digital uncertainty every millisecond they're running. We call ourselves engineers, but civil engineers get really upset when a suspension bridge unexpectedly dumps core. It's the sort of thing that makes the news and gets unpopular employees exiled to Farawayistan to maintain that oh-so-vital sham of accountability. In computing, when a server crashes, we check the logs and see there's nothing, so we wait to see if it happens again, all the while desperately hoping it doesn't. Maybe we turn on extra monitoring if we have it. This isn't a matter of laziness. The tools to identify many problems do not exist, and the ones that do exist are beyond the comprehension of the average sysadmin. You can learn the tools, yes, but when you master them, you have to figure out how to fix them and then it's too late, you've become a developer and your life is essentially concluded. Civil engineers at least have the benefit of being able to go look at their bridges and say helpful things like "This critical bolt is starting to bend, maybe we should stop sending trains full of lead across it while I check it out." They'll be told no, of course, but the engineers know to save the memos so that when the bridge dumps core the blame flows uphill.

If you don't want doubt, get a different job. Try something with feral hamsters. They're more rational than computers.

4. **"Freedom from care, anxiety, or apprehension; a feeling of safety or freedom from or absence of danger."**

Oh *heck*, I know—I KNOW—that you did not just try to apply "freedom from care" to anything in systems administration. Anxiety and apprehension are the soul of technology management. Confidence is for the organization's Chief Scapegoat Officer, someone so far removed from the day-to-day operation that they have no idea what's really going on down in the cubicle sewer. People who do the real work understand that computers are untrustworthy. Your test environment is exactly like your production environment, except that the database server has a slightly older CPU lacking two instructions present in production? Guess what's going to bite you? Hint: it's not that. You know about that. Rule 44 of systems administration clearly declares that "a perfect deployment means only that you haven't yet noticed the catastrophe." If you think this rule isn't true, you haven't been paying attention.

Your job description should read "go surfing in a blender."

And that's the real problem. You don't want to get chopped into Sysadmin Smoothie. Especially not for something as daft as the Chief Scapegoat Officer being unwilling to perform his one duty.

I recommend ignoring your boss' instruction in favor of building your own professional reputation.

The word "security" is thrown like a blanket over a bunch of other stuff. Experts who get fancy certifications like the CISSP will tell you that the Security Blanket covers a combination of confidentiality, integrity, and availability. I'm not an expert, because I let my CISSP lapse rather than risk ambush from unscrupulous recruiters armed with trank guns and nets. (If I ever awaken in a cubicle and discover I have a dart in my back, a sedative-induced hangover, and my feet forcibly jammed into—ugh—shoes, the world. Will. Pay.)

The good news is anyone can chant those three words. Confidentiality—the stuff that should be secret, remains secret. Integrity—data isn't mucked with except by authorized muckers. Availability—the computer more-or-less keeps running. The better news is you can take this mantra back to the boss and request clarification. Every organization has its own threats. Nobody knows what they are. Leverage this ocean of ignorance to accomplish three things.

First, address whatever your boss thinks the biggest threat is. That's probably whatever's been in the news most recently. At this exact moment, that's ransomware. This gives you every excuse to deploy a mammoth ZFS-backed fileserver and a snapshot regimen and declare that anything on the server is safe from ransomware.

Second, take advantage of the mandate to choose an interesting project that can be reasonably stuffed beneath the Security Blanket. Learn about proxy servers, or netflow, or DTrace, or tcpdump. Use an article about security flaws in old processors to get everything older than ten years old replaced.

Third, do the best you can with everything else.

Be sure to save all the emails where the boss refuses to let you do things. They might force the Chief Scapegoat Officer to go out and fall on his sword.

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

MICHAEL W LUCAS's has written over 40 books and resists all efforts to make him stop. One title, *Only Footnotes*, was a collection of all the footnotes from all his books, but he immediately invalidated it by adding more footnotes, so that doesn't count. His most recent titles include *TLS Mastery* and *$ git sync murder*. He's currently writing a second edition of *DNSSEC Mastery*, doubling down on the harm he previously inflicted on civilization.