WeGet letters

by Michael W Lucas

Dear Last Worst Hope,

It's all too much. I've cleaned up the servers and dealt with the outages and stabilized the environment, but the boss keeps piling on more work and more work and there's no way to complete it all. I can't quit, but how to I manage all this?

—Overwhelmed

Dear Overwhelmed,

Helping you begins with rewriting your letter to put the blame where it belongs.

I have performed my duties with the bare minimum of competence, but I don't understand that the reward for work well-done is more work. My efforts to educate my manager about the amount of work required for further tasks have failed, either because my communication skills are inadequate or my manager honestly does not see how the amount of effort is relevant because he is a sociopath rocketing to the C-level. Probably the latter. I arranged my life with insufficient prescience or flexibility, and now I'm trapped. What should I do?

There. That's better.

The problem with computing professionals is that they think of themselves as "problem solvers." After all, they make these super complicated machines do complicated things, like add really big numbers together. Each processor core contains a billion trillion gazillion transistors, and you command them all! When this horribly complex machine implodes, you fix it! You don't work with software—you are a professional problem solver.

Problem solver. That's what the sociopath rocketing to the C-level wants you to think.

Computers are simple. The complexity of the most advance computer is wholly inadequate to grasp the intricacies of the simplest virus, and those pale next to the horrors of corporate politics. Computing professionals are skilled at solving problems in a tightly constrained environment where the Four Sacred Resources—Processor, I/O, Storage, and Memory—reign, but those skills cannot model the innumerable resources of reality.

To succeed in the outside world, you must accept the limitations of your skills and reject the constraint of being a *problem solver.* As long as you dream of yourself that way, you'll lose against the illogic of meatspace. You have metrics. You have measurements. You have all the data that says you're working hard. Abandon the labels others have slapped on you. Liberate yourself. Abandon solving problems in favor of *strategic failure*.

Strategic failure isn't about bringing the whole system down. Any sysadmin can do that. It's not even about timing, although timing is important. It's about choosing failures that will embarrass the right people at the right time, and being able to declare with a straight face, "I only had time to maintain one system, and I chose the mission-critical one."

Yes, your manager will be angry. So what? If you can't quit, they can't replace you. Everyone in a position like yours, in any organization, possesses a unique combination of skills ranging from the bizarre to the obscene, a brew which is entirely impossible to replicate in any other single person. They only way to develop those skills is to *be* you, and nobody will sign up for that. Don't be arrogant—after all, would anyone competent outside the tiny, specialized cell of computing let themselves be maneuvered into this situation? Showing anger and frustration will only get you sent to Human Resources for counseling about your attitude. A shrug and an indifferent "I allocated my resources in accordance with guidance from management" will serve you well. Use those words, *in accordance with guidance from management*, like roasted garlic. A bit sprinkled here and there will give your businesslike attitude credibility.

Yes, your manager will be angry. So what?

After two or three incidents of properly chosen, high visibility, irritating but non-devastating strategic failures, you'll wind up in meetings with your manager and assorted outsiders who want to know why you suck so much. Your manager would prefer to throw you out the nearest airlock, so ignore them. Concentrate on the others. Be calm. Present everyone but your manager with the documentation on how you work. Almost always, merely having the documentation will suffice. Outsiders won't ask too many questions, out of a well-reasoned fear that they might learn something about computers and thus be forcibly transferred to your department.

People will present solutions. You should also offer solutions. One of them should be your preference, the others, acceptable. If the company wants you to, say, stop aggregating syslog and netflow data and shut down those systems, that's fine. You can always answer trouble tickets with, "Management has declared that I cannot help you with this problem."

Eventually, they'll settle on hiring someone. As previously established, whoever they hire will not be qualified to help you or to manage your systems. Remember Sysadmin Rule #27: "Competent coworkers are not hired. They are forged. By you." If you offer to mentor a junior sysadmin and save the firm tens of thousands of dollars, you improve the odds of getting help. If you make that offer in front of other people, you improve your image in the company. Be sure to say that you have specific questions you want to ask applicants.

Talking to job seekers? Ugh. Yes, I know, it's painful, but you're going to have to talk to the survivor daily so you best discard everyone who'll be painful to work with. No, don't ask about binary trees or bubble sorts or any of that other garbage. You want to discard applicants as quickly as possible, so set up a puzzle. Now that technology has advanced and CRT monitors are no longer standard, I can finally share my Secret Helpdesk Hiring Puzzle.

I would bring the applicant to an isolated room lit with the worst fluorescent tube in the building. If I had a chance beforehand, I would establish mood by starting my CD of "Great Horror Movie Screams" at a nearly subliminal volume and lighting a sample of incense from the Despair Collection. The room contained a desk, a computer, and a CRT monitor with a twisted and distorted image. I would say "If you worked for me, how would you fix this? Talk me through it."

Simple, right?

Swapping out the monitor didn't work. Neither did swapping out the video card, or the whole computer. At that point, most applicants said they would ship the whole computer back to the manufacturer.

The people who realized that the problem involved the computer's location and shifted it two feet from the magnet I'd taped to the bottom of the desk? They got the job.

You need a puzzle like this, with modern technology and just a whiff of malice. Something even your boss can understand. The convenient thing about the *problem solver* label is that people outside the computing department also believe it. "I set up this typical problem I have to solve, and only these applicants could solve it," is instantly credible.

Yes, you'll spend time training your new flunky—but the reward for work well-done is more work. Plus, you'll be training them to handle the work you don't want to do, and you already know they have problem solving skills sufficient for the tightly constrained environment of computers. This will give you time to solve your real problem and rearrange your life to be more flexible.

Be careful practicing strategic failure, however. Do it too much, and you'll find yourself rocketed to the C-level.

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

MICHAEL W LUCAS is the author of *Absolute FreeBSD*, *$ git sync murder*, and fifty-odd other books. *Letters to ed(1)* collects his FreeBSD Journal columns. Learn more at https://mwl.io.