# WeGet letters

by Michael W Lucas


**Hello, illustrious master sysadmin,**

**Maybe something's wrong with my servers, but I'm not sure. I get complaints, but when I look, everything's fine. And the complaints are pretty weak—"it's slow," "it doesn't work," or "why do we even pay you people?" I don't really know if there is anything wrong, but my gut says there might be. How do you track down such hazy problems?**

**Thanks,**
**Befuddled**

Dear Befuddled,

The annoying thing about asymptomatic system failures is that they're asymptomatic—but no less real than the kind with noticeable symptoms. Some user makes a call, an actual voice call where they're spewing random words in some language from their food-hole and you're expected to parse that babble with your ears, when even Hollywood knows that sysadmins are artisanally optimized to receive information via their eyes and extrude alloyed sarcasm and results from their keyboard-callused fingertips. Any one of these users can at any time disrupt your meticulously assembled hallucination of whatever problem you're working on and demand that you turn your three pounds of skull-pudding to the fact that their web browser jittered, actually jittered, when they played a cat video off the fileserver or they got a "File not found" error when they know durn well that they saved their proposal under that name just last night on their son's computer.

The obvious solution—assigning every user who makes a voice call one of those nifty PDP-11 emulators as a desktop system until they learn enough about computing to be allowed near a machine with a monitor, like a Sinclair ZX-81—won't work. They'll only call more.

But some errors are more mysterious. There's nothing you can point to. No spewing volcano of log messages, no cryptic PHP errors screeching about missing files or database timeouts, not even any ping failures. Something simply feels… wrong.

You're descended from a long line of monkeys that survived the brutal savannah long enough to become parents before starving capybaras devoured them. (New archaeological evidence declares

that "torn apart by hyenas" was reserved for nobility.) Maybe your software understands shell script, but that aforementioned lump of skull-pudding hasn't been upgraded since the Paleolithic Era and doesn't have many tools to work with.

The result is that when you arrive at your ergonomically hostile cubicle every morning, a part of your brain screams that you need to climb a tree right now.

The only way to shut up that voice is data.

Every computing organization has monitoring software, probably something like Icinga or whatever: boring, reliable, and consciously and deliberately limited. Whole meetings get wasted discussing what this monitoring should check, how often it should check, when it should alarm and when it should ignore detected problems.

It's not that those problems don't occur. Maybe the monitoring checks the company's ERP system every minute. Every hour or so, it misses one check. Your organization decides to ignore those, because by the time some feeble human perceives the message and logs into the system the problem has evaporated.

That intermittent alert happened. Maybe it only lasted five milliseconds, but something failed. That lost check acts on your brain the way a rustling in the tall grass acted on your 200th-great-grandma.

Something lurks out there. Your subconscious knows it. Is it a death by noble tigers? Or shall archaeologists digging up your remains declare "Wow. The marks on these bones resemble flamingo teeth" before scurrying off to write a widely ridiculed thesis?

You need to spend time on that intermittent alert.

The world is full of monitoring tools. Each is limited in its own infuriating way. Even your complex, all-inclusive, carefully tuned Icinga with all the trimmings and extra gravy has gaps, plus Sysadmin Rule #25 declares "All monitoring reduces to 'send an email to warn you email is broken.'" Look in those gaps. And this issue of the *Journal* has a whole bunch of information on monitoring and assessment tools. For anything involving the network, you should have netflow and SmokePing. If your network administrators don't have these tools, I've written a whole bunch of books about such topics and recommend you trebuchet copies into the network department until they sense danger and install something useful.

*"You don't have time to learn every new tool? Please. If my eyes rolled any harder, they'd fall out of my head and bounce off the table."*

But which tools should you use?

All of them.

But not simultaneously.

Every tool exposes something different. That's why we have so many. Explore DTrace—yes, I know, you're merely a puny sysadmin and you

don't deal with system calls, but if your ancestors hadn't explored new things, you'd be capybara bait by the time you hit 20 so suck it up and dig in.

You don't have time to learn every new tool?

Please. If my eyes rolled any harder, they'd fall out of my head and bounce off the table.

None of us have time. Look at me. I'm taking my time away from writing stuff that people will pay me for to answer your silly letter about the nebulous worries your tree-oriented subconscious is pushing upstairs, aren't I? The only reward I got for writing this column was the annual *FreeBSD Journal* board meeting held the night before BSDCan, and while previous years' programs included words like "opulent," "bacchanalia," and "sybaritic" the *Journal's* gone free and I suspect that descriptions of future meetings will feature the phrases "gruel," "alley," and "precautionary vaccination"—but you don't see me abusing my public platform in this very journal to gripe about these abusive changes, do you? No, this isn't griping. I'm better than that. This is merely a detailed example of what I'm not griping about.

We never have time to learn new things. But learning new things is why we're in this nightmarish profession. The alternatives require wasting your dwindling supply of days deciphering the noises coming out of random people's food-holes.

Pick tools that many people love. It's not that they're good tools; none of them are good, but these tools have a better pain/reward quotient than the rest. Play with them. Run each for a few hours, or a week. See what data each provides. Download some recommended DTrace scripts for your application software and see what it spends most of its time doing. Sure, some of what you learn will disturb anyone close enough to hear the screams with which you'll wrench yourself out of your midnight nightmares, but you'll slowly assemble an awareness toolkit that lets you see the capybaras in the weeds.

Develop your skills highly enough, and you too might rate death by hyena.

PS: I commend the manners displayed in your salutation. Your parents raised you well. Not well enough to choose a better career, of course, but well. ●

**Michael W Lucas** (https://mwl.io)'s newest book is *FreeBSD Mastery: Jails*, as well as 30-odd other titles like *Absolute FreeBSD*, *PAM Mastery*, and *git commit murder*. He's shooting for death by tardigrade. Send your questions to letters@freebsdjournal.com to be complimented in the most backhanded manner he can arrange.