To All the SREs I’ve Loved

Felix Glaser

Shopify

We’ve all come across an application pushed out of the door with questionable reliability. It makes our lives tougher when, inevitably, the application experiences its first outage and we are called in to help make the application more reliable. We aren’t surprised by this but wish we would have been involved in the design and planning of the app and made reliability a first-class citizen. It’s so avoidable—and it can wear us down.

I am not an SRE—well, not anymore. I returned to my first passion, security, drawn to strengthening the security of systems, protecting our users’ data, and keeping the bad guys out. The change in perspective from switching teams led to an insight: security is to SRE what SRE is to product teams. We are here to support you to keep your systems secure, up and running, and ultimately reliable. In a way, we are *your* SRE team. Yet, you so often treat us the same way the single-minded product team treats you.

I’ve heard the complaints. You feel like we’re slowing you down by being paranoid and always thinking of the worst-case scenario. What makes it worse is that we don’t even have good data to convince you it’s worthwhile. It’s not like we can say, “You need to update this library or else the servers will be hacked.”

Scaling up servers for the next DDoS (distributed denial of service) or flash sale is very tangible. See how many connections a single instance can handle, see how many people tried to connect the last time, and you just scale past that. As a security engineer, I have no way of pointing at that one CVE (Common Vulnerabilities and Exposures) piece that you need to fix or else. As long as you haven’t been breached, there is no way to tell whether you’re secure. There is no way to prove that a system is secure.

It’s a thankless job; security teams aren’t perceived as bringing the same value to a company as the SRE team, mostly because our work is nebulous. We have some great tools and techniques, including keeping operating systems, VMs (virtual machines), and containers updated; installing security patches; scanning the company’s networks and IP (internet protocol) to detect all the software someone took online without consulting us; encouraging developers to update their dependencies; fuzzing their code before it goes out; and making sure permissions aren't over-granted. So we do the best we can, although if we do it well, no one will notice.

So what would make a production security engineer's life easier? Giving security the same room and attention you wish product teams would give SRE. Keeping us in the loop from the start. Communicating changes in infrastructure as early as possible. Involving us in the decision making. Not running an old operating system on a VM somewhere that you forgot about. Clicking Merge on that Dependabot PR (pull request) in a timely fashion. Trusting our recommendations. Ultimately, making us part of your organization by integrating us into your daily work and decision making. This has obvious benefits for you, because a hacked system might go down and cause a lot of downtime—and I would hardly call that reliable. So the next time we come to you with a seemingly paranoid recommendation, remember that we care deeply about keeping our customers safe and everything up and running. Don't let security become an afterthought!

And always remember: treat your security team the same way you wish that one product team treated you.