[MECHANICAL VOICE]

**SPEAKER:** Welcome to season five of the *Prodcast,* Google's podcast about site reliability engineering and production software. This season, we are continuing our theme of friends and trends. It's all about what's coming up in the SRE space, from new technology to modernizing processes. And of course, the most important part is the friends we made along the way, so happy listening and may all your incidents be novel.

[MECHANICAL VOICE]

**JORDAN GREENBERG:** Welcome to this episode of season five of the *Prodcast,* Google's podcast on site reliability engineering. My name is Jordan Greenberg, and I am here with my co-host Florian. Say hi, Florian.

**FLORIAN RATHGEBER:** Hey, folks.

**JORDAN GREENBERG:** Long time listeners may remember our guest today from season two, Life of An SRE, where she told us about her path into SRE. Now, she's not an SRE anymore. Who are you and what happened?

**SHANNON BRADY:** I'm Shannon from the gLinux team here at Google. What happened is I'm still on the same team doing Linux at Google, but I just have a new job title. Essentially, our team structure changed, and so while my team aren't technically SREs anymore, we are still SREs very much at heart, and we are still driven by the same principles that are the backbone of SRE that we've always had.

**JORDAN GREENBERG:** And if you can remind us just a little bit about the gLinux platform team and what that's like for you in your day to day, so our listeners have some context, if they haven't listened to the last episode.

**SHANNON BRADY:** Of course. So Google actually has their own version of a Linux distribution called gLinux. Previously, it was called gBuntu, but we've switched from an Ubuntu based distribution to Debian a few years ago.

gLinux is kind of a combination of internal packages, R configurations, and things that are available on Debian upstream that a lot of Googlers and engineers use in their day to day work.

**FLORIAN RATHGEBER:** Thanks, Shannon. That's interesting context. So as far as I'm aware, gLinux is, of course, used on a bunch of devices that Googlers use in their day to day. And your team is managing that, in some description. So how do you manage a large fleet of devices efficiently? And also, security presumably plays quite a big important role here, how do you keep them secure?

**SHANNON BRADY:** It's no easy or small task, but the backbone is testing, canarying, and monitoring with a little bit of puppet thrown in there.

**JORDAN GREENBERG:** You have birds and puppets?

**SHANNON BRADY:** Yeah.

**JORDAN GREENBERG:** Oh my gosh.

**SHANNON BRADY:** Yeah.

**JORDAN GREENBERG:** gLinux sounds kind of cool.

**SHANNON BRADY:** It's awesome, actually. I think it's awesome. So essentially, our philosophy is that testing is super important because you want to catch as many issues as possible when you're testing a potential release rather than when users already start to experience these issues.

Obviously, testing can't catch everything. But the second best option is to impact a small number of people who have opted into being testers of gLinux and who might give you a heads up when something is failing. And so the best problem is a problem users never know about. The second best problem is a problem that the vast majority of your users will never know about.

**FLORIAN RATHGEBER:** So you're applying the classic SRE principles of you shift left and you do stage rollouts and whatnot.

**SHANNON BRADY:** Yep, we have that classic stage rollout philosophy for pushing new releases every week to our users. And so once something has hit testing and we're relatively sure that it's in a good state, then we can marry it out slowly to the fleet using, of course, monitoring as a part of our rollouts to help prevent issues and then recognize them as soon as possible. Monitoring is super important.

**FLORIAN RATHGEBER:** Yeah, that makes a lot of sense. So presumably, the kind of changes that you test in this way would be like, if there's a new upstream kernel, things like that?

**SHANNON BRADY:** Yep. So that could be anything from a new kernel to a new package to anything, really.

**FLORIAN RATHGEBER:** Even like a config change you make to it.

**SHANNON BRADY:** Yeah, so it's everything from security tools to a kernel to just configuration changes that we want to push out to our users. Really, anything can go wrong, so we want to anticipate whatever can go wrong.

**JORDAN GREENBERG:** OK, so you talked about the canaries, which are you have opted into testing unstable or beta builds, whatever you might call them. Can you tell us about the puppets? Everybody loves puppets.

**SHANNON BRADY:** Puppet is a really powerful tool that we use on gLinux. And it's used to enforce both configuration or just set defaults that our users can change on our fleet. Puppet is a configuration language. And the best thing about it is that it's very flexible, allowing us to target different configurations on our fleets or just subsets of it.

Because the gLinux user base is extremely diverse. And we have lots of different hardware and lots of different setups. And we want to make sure that the right people are having the right configurations.

**JORDAN GREENBERG:** That makes total sense, because if I have my config set up, I don't necessarily want to have it changed every week if some update happens.

**SHANNON BRADY:** Oh, definitely.

**JORDAN GREENBERG:** Helpful.

**FLORIAN RATHGEBER:** Surely, Linux users are among the most opinionated people you can find anywhere. So yeah, I'm sure that is super important.

**SHANNON BRADY:** Yeah, but that's a blessing as someone on the gLinux team. The great thing about Googlers is that they're Googley. And so we get a lot of really constructive feedback from our users about what doesn't work in their setups or what they would like to see. And that can really help us drive innovation on our platform and really understand what we can do to make gLinux better for everybody.

**JORDAN GREENBERG:** Nice.

**FLORIAN RATHGEBER:** Win-win all around. That sounds great.

**SHANNON BRADY:** Oh, definitely.

**FLORIAN RATHGEBER:** So do you think, then, we have already passed the year of the Linux on the desktop/laptop, because that has been a thing at Google, certainly, for a while. So are we now in the year of Linux in the cloud.

**SHANNON BRADY:** No. I think, in general, we are really in the year of Linux everything. And that's both at Google and outside of Google. I've been using Linux since I was a child, which makes sense because my dad was actually a Unix engineer. So it runs in the family. And it's been a really long time since my first Linux installation CD.

And it's been just so amazing to see how over the years Linux has evolved and grown and how every year, every change, it continues to get better and more user friendly. And the fantastic thing about the year of the Linux everything is that we have a lot of Linux users that don't even know they're Linux users.

**JORDAN GREENBERG:** Yes.

**SHANNON BRADY:** So did you know that things like the Steam Deck or Chromebooks are actually powered by Linux, so they are Linux at their core.

**JORDAN GREENBERG:** Absolutely. I think this is so special for people to learn and understand. Linux used to be kind of scary, and you used to send your friend into a bad time by telling them, hey, you should try installing Arch, right?

Now, instead of it being year of the desktop, I think year of the Linux everything does apply. Form factors for things have changed. So example, many PCs are coming out. They're this big. Steam just made an announcement about their new hardware lineup, definitely will be running on Linux.

**SHANNON BRADY:** Oh, I did hear about that.

**JORDAN GREENBERG:** Yes. So that just makes Linux approachable for people. And hopefully, they can realize that it's the coat of paint that they put over a powerful operating system, is actually something that's really helpful in their day to day.

**FLORIAN RATHGEBER:** Yeah, we shouldn't forget, in fact, about the huge fleet of mobile devices running Linux under the hood in form of Android, which is-- still runs the Linux kernel, even though the rest is maybe not as recognizable as the Linux system anymore.

**JORDAN GREENBERG:** Absolutely.

**FLORIAN RATHGEBER:** If we take that into account, we are certainly in the year of Linux everywhere, which is great.

**SHANNON BRADY:** That is also the beauty of Linux is that some people may not be ready for a [? gen2, ?] and that's OK. And other people, that is exactly the kind of customization that they want that drives them. And the beauty of Linux is that it can be anything for anyone. If you want something, make it. That, for the most part, open source code is just workable.

If you want to make your own Ubuntu flavor, go for it.

**JORDAN GREENBERG:** Absolutely. You want to have your home server so that you can share your pictures to grandma, or if you want to have it so that you can host your team's D&D session in your own foundry instance or something like that, do it, because Linux is easy. It is made for you.

**SHANNON BRADY:** Yeah, it's made for everybody.

**JORDAN GREENBERG:** And now, post-Linux commercial break. I have a question to you about how AI has impacted your work or device requirements because it's not like we're going to be running Steam Deck at the corporate.

**SHANNON BRADY:** Yeah, that would be nice, though.

**JORDAN GREENBERG:** It would be nice. Has the work or the requirements for devices changed in the age of AI, and how does the way that you secure them change?

**SHANNON BRADY:** Users having computers that enabled them to do their best work and have all of the different hardware requirements that they need has been something that my team has been involved in for a long time, even before the rise of AI.

AI engineers, who are working in these fields, definitely have unique hardware requirements and unique software requirements in order to be able to effectively do their jobs. And who better to help us understand those needs than the people who work with those engineers every day.

So when we're looking into choosing a device or seeing what's going to work on gLinux, we talk to people at DeepMind, we talk to people who are going to be using these devices and say, hey, what do you need? What is going to enable you to do your best work?

Additionally, something that we look for is choosing a device that is very flexible. So choosing a device that is inherently very flexible allows us to have a base unit that we could build on, and that we have a good base unit of specs that will, by and large, cover most of our users, but with the flexibility of changing things for specific teams, or specific needs, including our AI engineers.

**FLORIAN RATHGEBER:** Speaking of AI, maybe, in a slightly different context, if you want to tell us about your own experience. What would you call the single most important use of AI in your day-to-day SRE work?

**SHANNON BRADY:** Well, that's a very hard question. In my opinion, AI is really such a powerful and multifaceted tool, and there isn't one single most important use of it. Different teams and even different individual engineers will find different things useful. Like, something that I need to do very frequently is analyzing signals and analyzing exported logs from our internal fleet.

Naturally, these are logged to different databases, and they're structured in entirely different ways. And in order to be able to detect issues and determine the impact of our users, we really need to put all of these signals together in an easy-to-use way.

And AI allows us to be able to more quickly and easily build queries that we need across all of these different sources to get the answers that we need and respond to potential problems. Because the last thing you want to do when there's an outage is be messing around with an SQL query trying to get your join correct.

**JORDAN GREENBERG:** Oh, yeah.

**FLORIAN RATHGEBER:** Yeah. But presumably, you don't want to give all that control to the AI. Do you see any risks that AI may pose in the context of decision making in SRE?

**SHANNON BRADY:** Of course. The biggest risk for AI, in this context, is having an overreliance on what it's telling us. And that can be an overreliance on the queries it makes, the alerts it's flagging to us, and the recommendations for potential workarounds that it's giving.

AI won't always be correct, and it's up to us as engineers to ensure that we're checking our results and we're maintaining the institutional knowledge of our teams and our products in order to be able to correctly interpret what we're seeing from the AI. AI is at its most powerful and most useful when it's used by a skilled engineer as a part of their toolkit, not as a whole toolkit.

**FLORIAN RATHGEBER:** Yeah, that is very good advice.

**JORDAN GREENBERG:** Very smart. And just so we touch on it briefly, too, the tasks that SREs need to do are pretty similar to tasks that people in other fields might need to do, too. So example we use AI in my team for data formatting and building out spreadsheets. All of these things can leverage AI to make them a little bit more efficiently done and take some of the burden off of us.

I did want to switch gears a little bit just because I was thinking about your title change and how SREs, they have a few different identities, or hats, we might call them, that they may wear. What would you say is the hat that you wear most frequently in your current role? And do you think that it's best if an SRE is very specific, or if they are able to adapt and context switch, and wear a different hat one day? What kind of mix do you think is effective?

**SHANNON BRADY:** I think a lot of people end up wearing multiple hats over the course of their career. That's not something that's unique to SRE. And being able to grow in your role and grow as an engineer frequently means changing out your hats. And the best part about that is that, when you get a new hat, you don't just throw away the old one. It's still there for when you need it. It's still there kind of as an old reliable.

I'm wearing mostly the systems engineer's hat, so a lot of what I do is way more Linux focused than coding focused. But I've also had other hats. So for example, I used to work in tech support, and that was a hat I wore for a long time.

**JORDAN GREENBERG:** Yeah. And there's different flavors of that, too. There's software engineering, systems administration, systems engineering, like you were saying. And we also just had Jen on, security SRE, so we know that there are very strange types SREs. And in concert, we work together to make such a beautiful situation happen.

**SHANNON BRADY:** Oh, definitely.

**JORDAN GREENBERG:** So in the context of that, what is a typical week looking like for you, outside of an on call?

**SHANNON BRADY:** A typical day or a typical week is different week by week for me. But overall, I work tons with our internal partners, especially with our support staff, to really make sure that they're both able to give the best support possible to our internal Linux users, but also to hear their feedback so that we can know and we can listen to what they're telling us about what they're seeing, what our users are seeing that we may have not otherwise known about.

It comes back to the mantra of, not everything is caught in testing.

**JORDAN GREENBERG:** Right. Exactly.

**SHANNON BRADY:** Outside of working with our internal partners and being a people person, I'm pretty involved in working on improving and maintaining the testing infrastructure that gLinux has. gLinux does weekly releases, as we've touched on before. And that includes new and updated packages from Debian, internal packages from Google, configurations, et cetera, et cetera.

And so having a reliable test infrastructure is really super important because it allows us to have more confidence in the release. Having unreliable tests and having unreliable infrastructure means that, not only does an engineer need to potentially spend more time investigating a failure, but it also creates a kind of alert fatigue. And that could lead us to easily mistaking a real failure for just another flaky test.

Because the first 10 times, it could have just been the-- it wasn't really a failure. And then, you'll pass in, oh my god, all of our machines are broken.

**JORDAN GREENBERG:** Yeah, let's avoid that if we can.

**FLORIAN RATHGEBER:** Yeah, that is all super important work, but I'm presuming that some companies-- someone might argue that, hey, we don't need to hire SREs for this job. So in your opinion, what is or when is the right time to add SREs to your company, and what might influence someone to hire an SRE or put someone in a similar job function like yourself?

**SHANNON BRADY:** The right time is going to be different for every company. For some companies, it's going to be best to bring an SRE from the ground up. And for others, it's going to be the right time to bring in an SRE when the company really starts to scale up or starts introducing more complexity into their operations.

I think having SRE roles, or at the very least, practicing SRE fundamentals, like monitoring, disaster resilience testing, incident response, from the very beginning, can really set companies up for success, whether or not they have the SRE job title officially or not. Stability and reliability go hand in hand with development and innovation.

**JORDAN GREENBERG:** That's a great way to put it. Some people don't know that they're SREs, and some people don't have the title of SRE. But hopefully, hearing from you, they know that their work is exactly what an SRE does.

**SHANNON BRADY:** I wouldn't get too hung up on the title. It's just another hat that they're wearing. Anybody can be an SRE if they're engineering like an SRE. When you follow these best practices, when you follow these SRE principles, it doesn't really matter if your job title is DevOps or sys end or system administration or IT support, you are practicing SRE fundamentals, you're practicing as an SRE.

**JORDAN GREENBERG:** Love that.

**FLORIAN RATHGEBER:** Exactly. Yeah, you can be the SRE at heart even if you don't have that job title.

**SHANNON BRADY:** [INAUDIBLE] I'm proof of that.

**FLORIAN RATHGEBER:** Totally. So one thing that we've noticed over the last years is that we've increasingly been using services more online rather than on our devices and whatnot. Do you think that as society in general, do you think we should depend on online services more or less, or is it fine the way it is today? What do you think?

**SHANNON BRADY:** Both of those things can be true in different ways. There are lots of everyday services that really do and many that would actually benefit from digitalization and from increased connectivity. Lots of people lead very busy lives, and being able to interact with the services that they need on their own schedule from their own house is incredible.

But I think, at the same time, it's really important to think about the digital divide here and people's personal preferences as well. We need to ensure that the people that may not regularly use or have consistent access to the internet can still get what they need to have done done and also have a good experience without needing the internet.

Likewise, it's super important when designing connected products to also think about the offline experience. So will it still work and function without the internet? And how would I use this product if it was never connected to the internet in the first place?

**JORDAN GREENBERG:** Really good point. I appreciate this so much, Shannon. You're helping us think about SRE in the ways that only a person who works with the platform and thinks about things so holistically, from the user perspective, from the hardware perspective, from the supplier perspective, the reliability liability perspective molded into one can tell us.

Thank you, again, for being an iconic guest. And that's it for this episode of the*Prodcast.* Thank you to our listeners for hanging on with us, and we'll be back with some more soon. Have a great day.

**SHANNON BRADY:** Thank you so much for having me.

**JORDAN GREENBERG:** Thank you.

**FLORIAN RATHGEBER:** Thanks, Shannon. That was great.

**SPEAKER:** You've been listening to the *Prodcast,* Google's podcast on site reliability engineering. Visit us on the web at sre.google, where you can find books, papers, workshops, videos, and more about SRE.

This season is brought to you by our hosts, Jordan Greenberg, Steve McGhee, Florian Rathgeber, and Matt Siegler, with contributions from many SREs behind the scenes. The *Prodcast* is produced by Paul Guglielmino and Salim Virji. The *Prodcast* theme is "Telebot" by Javi Beltran and Jordan Greenberg.

[MECHANICAL VOICE]