[THEME MUSIC]

JORDAN GREENBERG: Welcome to season 6 of the *Prodcast,* Google's podcast about site reliability engineering and production software. This season, we met with SREs in person to hear what's on their minds, to explore the importance of psychological safety, and to learn what's coming next for SRE. And of course, the most important part is the friends we made along the way. Happy listening, and may all your incidents be novel.
[THEME MUSIC]

[MUSIC PLAYING] STEVE: Hello, everyone. I'm Steve.

MATT: I'm Matt.

STEVE: He's Matt.

ADAM: Adam.

STEVE: He's Adam. I hope you can hear this because we're trying this kind of on the fly. This is the podcast. This is the podcast about SRE and production software here at Google. We're on site.

MATT: On site. First time.

STEVE: Where are we? What city are we in?

ADAM: Seattle.

STEVE: Seattle. Yeah. None of us live here except for this guy.

ADAM: I don't live here either. I live about 25 miles away.

STEVE: OK, he doesn't even live here. It's great. So today we're going to be talking with a few different SREs. Adam is the first one, so we're kind of figuring out what it is that we're going to talk about. So I'm guessing we're going to talk about the cloud. This is Cloud City, after all, right?

MATT: No.

ADAM: It's cloudy.

STEVE: Bespin, is that what we call this place? Something like that Adam has been working on, what, cloud compute and things. How would you describe what you work on?

ADAM: Google Compute Engine.

STEVE: Google Compute Engine. That's the one with VMs, right?

ADAM: Yes. For 13 years now.

STEVE: 13 years.

ADAM: Yeah.

MATT: On clouds.

STEVE: You want to?

MATT: Yeah, so tell us about the engine and what computes do in it and stuff.

ADAM: So it is Google's attempt at selling virtual machines to the rest of the world. So taking the infrastructure we have and virtualizing it so that we can sell little bits of computers to people or whole computers, depending on what you want to buy.

MATT: I take it's gone through a few iterations and we're at the latest one?

ADAM: Yeah, it's an incremental upgrades of things over time. It's a ship of Theseus. It has not-- There's never been a version increment, really. It's pieces are tacked on over time and bits are rewritten.

There's been multiple iterations of the control plane, multiple iterations of virtualization software. But it's still the same product, as far as people are concerned.

MATT: Tell us a bit about your role in that right now.

ADAM: I'm one of the tech leads on the compute SRE team. So we manage GCE as best we can, from an SRE perspective.

MATT: That sounds challenging. I bet the use cases are pretty diverse for computers.

ADAM: Yes, it is more than just computers. I guess it's worth pointing out. It's a data center in a box. So you end up with virtual networks, firewalls, computers, disks, load balancers, all of the things you would want.

MATT: As an SRE myself, I usually have to focus on one of those. Tell us about supporting them all.

ADAM: You don't end up with a deep knowledge of each individual piece. There's-- one of the teams manages the API, and generally the customer experience on the API itself doesn't

necessarily know how the business logic works. Other teams deal with individual networking components, like load balancers or the virtual network, or things like the virtual machines or disks.

There's multiple teams for all of these things. And that's just SRE.

STEVE: Yeah, so there's these product teams that build the systems or adapt the systems over time, I suppose. And I think-- so one piece of context that's important is that the people who listen to this podcast or watch the video, or whatever it is, are like SREs who don't work at Google. That's the idea. Or SRE adjacent folks, production teams or DevOps, things like that.

So I think a lot of them will have read things about Google internally. Like, They know about this idea of this thing called borg. So is Compute Engine just like a different borg, or is it like just an exposure of borg? Like, that doesn't sound that hard, Adam. Surely, it's just a matter of opening a firewall to the computers or something, right? He says.

ADAM: If only it was that simple. Yeah. I mean, yes, Compute Engine runs on borg, but trying to expose virtual machines on a cluster scheduling system that was designed for machines to be completely replaceable and dispensable is part of the complexity.

STEVE: And then what about, like, we want to offer just a raw computer to end customers.

ADAM: We do offer that as a feature, actually.

STEVE: I mean, generally customers don't necessarily want exactly that.

ADAM: They don't want--

STEVE: They may ask for it, but it's not actually, not actually the best plan.

ADAM: Yes.

STEVE: Could you elaborate on why that is not a good plan and why customers might--

ADAM: One of the significant benefits of the cloud is you don't have to think about hardware. You give us tell us you want the virtual machine, and if the hardware it's running on fails, we will run it somewhere else. You get redundant computers out of one computer. Obviously, if you want true redundancy, run more than one virtual machine. But on the whole, your VMs will stay running even if the underlying hardware fails.

STEVE: So you've been working on this product for a while now, I would say.

ADAM: It's been a while.

STEVE: It's been a while, has it, I don't know, gotten bigger at all?

ADAM: Just a little.

STEVE: Has demand grown?

ADAM: Yes. So I've been on the project since before it was GA, so I remember when it was not really usable.

ADAM: Yeah. No, so in the beginning, you couldn't even boot off of a persistent disk. So persistent disk is the reliable storage that virtual machines have. At the beginning, we just exposed the disks that machines were on as well, and that was what you had to boot off of. That's an example of the way it's changed over time.

That is not an option anymore. Now you have to boot off of a persistent disk so that if the underlying host crashes, you still have your virtual machine.

STEVE: You mentioned something a little while ago, which is a term that I think people have heard about but may not know deeply. Like, you mentioned the control plane. So you said that there is a control plane for GCE or just for this data center product. How would you describe a control plane and let alone this control plane to our listeners?

ADAM: Yeah, there's-- of course, there's more than one. There's layers of control planes.

STEVE: Perfect.

ADAM: So when I say control plane, I'm talking about the API endpoint that people talk to mutate their Compute Engine resources. So this is compute.googleapis.com is the host name. That is the HTTP server that people talk to and change things. That turns into a very large amount of business logic and a bunch of different systems to actuate what people ask.

But the control plane, as we think of it, is the thing that holds the knowledge of the resources that exist and allows you to mutate them.

STEVE: So it's like it has a database, it knows what's going on in the world. You can tell it, I want one more of those things, please, or one less of those things, et cetera. And--

ADAM: It makes that happen.

STEVE: And it makes that happen. But the interesting part, I think, around it is that it's not just about the VMs themselves. Like you said, it's firewalls and network and storage and all these other things as well.

ADAM: What data center in a box means is still evolving. This is changing with AI drastically as well, with the large amounts of storage, bandwidth, memory, interconnect, everything that people want.

STEVE: Accelerators.

ADAM: Yes. And locality matters a lot more all of a sudden. There's-- the product is changing constantly. The requirements are changing. The old requirements never really go away either.

MATT: Speaking of scale, imagine I'm trying to draw by analogies. Something you've done, we've done a massive planet scale. I'm trying to think of what I do in my tiny little company, which I'm doing it at small scale. What's something that actually isn't any different, even though it's planet scale compared to my small business?

I'm cloudifying my thing, or not even that, but I have to do something a little larger. What's a support story you still have to do that looks like what you would have done at a millionth of the scale?

ADAM: Actually understand what your customers need. It's very easy, with a big company, with a lot of customers and enormous scale to forget that when someone is saying they have a problem, they probably are actually having a problem that can be traced back through the system into what's actually happening. Kind of breaking through all those layers and figuring that out is hard. But I think the same, regardless of the size of a company.

MATT: And do you think that this the same this is the same story over and over again, which is getting good at understanding what is needed, how to prioritize it, how to make it actionable, responding to it. Are these the tools that we're still using the same muscles in our brain and going a little bit faster with some of the tooling? How is that-- something that's actually changed in the last year, if I'm not mistaken.

ADAM: I feel like you're trying to lead me somewhere, but I'm not sure where.

MATT: Well, he just asked a question about AI's influence, like anomaly detection.

ADAM: Oh, yeah. It feels like this stuff is changing faster than I could keep up with it. But there is-- over the course of the last year, I've seen computers go from things where if a human had not suggested all of the relevant dashboards, all of the relevant information, there was no additional-- there was no added information when something went wrong. But there's a lot. Like the aggregation and correlation abilities of these systems is improving.

MATT: From what I could tell in my small business, I need a lot of the automated anomaly detection because I don't have the time to dedicate one person to look at all the dashboards.

Because they're probably swapping disks around and also responding to customer phone calls and all these things at the same time. So that tooling is fantastic for me. And at scale, yeah, I can devote a team of people to respond to things manually, but probably they want that automation too.

And that's kind of happening all at the same time. What are you seeing in triage ticketing right now? Like, this is clearly happening right now.

ADAM: Yeah. So, I mean, agents are pretty good at triaging bugs. So the summarizing, triaging and routing of things.

MATT: That's pretty interesting to me.

STEVE: I want to turn the corner a little bit into something adjacent. So you're on a couple teams, we call them IRT teams. They're Incident Response Teams. They're kind of like an escalated incident response team. So when people are in trouble, they can, like, ah, it's bigger than I thought or I'm not sure what to, exactly how to, they can basically page this team or engage this team in some way. Can you tell us about why did you choose to do this or what is it that-- how would you summarize this type of work?

ADAM: So it's my favorite part of the job, honestly, which feels a little weird because you're only summoned when there's a disaster going on. So IRT teams are-- incident response teams deal with incident management, incident command, and in general, just being around and helping to make things better when everything is going wrong.

Generally, they're comprised of the people who can't help but try to help when things are broken. And it's not a matter of wanting to rubberneck or of watch the fire burn. But the thought, maybe I can make it better in some way if I sit on the sidelines and watch and learn what's going on. That turns into, once you get enough skill and you stop, you don't get cortisol spikes when you're nervous about that.

STEVE: It helps.

ADAM: You can start taking command of these incidents.

STEVE: Would you say that's a good approach to take? Like, if someone finds themselves in this position and they want to get better at it and they want to be-- for example, I'm on a team that's just like this and I'm trying to get better at it. And I know that you've been in this position for a long time.

How do I get better at being a member of an IRT team? What's a good path to take or things to think about?

ADAM: Everyone needs to be aware of who is actually in control of the incident and defer to them. So there's, regardless of position in a company, regardless of level, managerial relationships or the like, someone is in charge and they should be listened to and it should be treated as a temporary leadership position. I don't know what the right way to describe it is.

STEVE: Yeah, temporal, for sure. Like, it's during the course of the incident.

ADAM: Yes.

STEVE: I heard one phrase got passed around, which is if someone is encroaching, maybe, a simple phrase is would you like to take over as IC?

ADAM: I have literally done that.

STEVE: That's a good one.

ADAM: And the person stopped. Because that is another aspect of this, is you show up largely because you were asked to, and it's not a favor because it's part of one's role at a company. But you are there because it is necessary for you to be there.

STEVE: Yeah.

MATT: This goes all the way back to where the first episode of last season. Or second episode, which is an industry principle, which is clarity of role, horizontal access, and openness and clear communication globally and solving problems together, as people. And I love hearing these stories, and also know that this is a part time work for many people.

Some cloud, IRT for some people is not a full time gig. It is something you do when there's only an incident, and it doesn't need to be a full time gig. It can be, in a big organization, where there needs to be a full time response. But as far as I can tell, there are people who only do that when it's called for.

What's it like when an incident happens all of a sudden? How is it self-organized or partially organized? Describe a life of an incident. Go through a--

STEVE: A big one, though.

ADAM: A big one?

MATT: Make it more--

STEVE: Make it exciting.

MATT: Yeah, exactly.

ADAM: So I can speak-- I'll speak for tech IRT because those are even bigger. If too many incidents are opened in too short a period of time, tech IRT is alerted to say, you should probably look and see what's going on because there might be something wrong.

So there's-- often that is the start of realizing that something is broken. This results in seeing what's broken, finding the on calls that have already been paged because they've already opened incidents, assuming that that's the case, and dragging everyone together, usually into a video conference, because I found that's the highest bandwidth way of getting people on the same page and making sure that you aren't duplicating roles and you've got the right people in the room.

I think it usually very quickly turns into, do we understand what is wrong and who are the right people to be working on this and let's get them here right now, at any cost. And, ideally, you can get that all going within about 10 minutes or so. Depending on the scope of the outage, that's either good or bad. Because if-- sometimes if cooling is failing, that needs to be faster than that. But those are much smaller, more constrained incidents in terms of who needs to respond.

STEVE: This is an interesting, like the way this is spoken about in external, more academic circles is like this is a case of what we call adaptive capacity, and it's the ability for a team to be able to borrow capacity from someone else, in this case, you and your friends, to come in and provide some capability or some knowledge or just like connective tissue, almost, with other teams, things like this.

Or just simply just like, hey, how about I just take over for you. Even that level, I think, does a lot for teams like that. So can you imagine the case where incidents like you've seen before were happening and IRT did not exist? Like, besides, just the actual access control that you might have to be able to change a load balancer or change a board job or something like that. Do you think that just this ability to step in and say, hey, we're here to help, just that alone, does that help the situation?

ADAM: I've seen it repeatedly help situations. There's, there's a level of feeling of responsibility for something but not knowing what to do that can make people vapor lock or deadlock, almost, in responding to things.

MATT: This sounds like, what's it, psychological safety here is really paramount. Being safe saying you don't know the answer is the only way to make progress here. It's like, I don't know, and I need to have the following things in order to know. And then having an incident commander say, OK, well, you need to go either tell me who to go find out that for you or delegate that to someone else.

STEVE: So has anyone ever gotten in trouble for asking for help from IRT?

ADAM: No.

STEVE: Leading question. I was really hoping you were going to say no.

ADAM: The attitude is if we're paged for a completely specious thing, the attitude is please don't page us for that next time, but feel free to page us with new problems, because don't let the problem sit and simmer if you don't know what to do.

STEVE: Yeah, it's better to apologize for a page than to apologize for not a page.

ADAM: Yes.

STEVE: I would say. Like, if you just let it just blow up and you didn't ask for help, that would be real bad. Thank you for coming. This was really fun.

And do you have anything you want to add? What do you want to say to the SREs who don't work at this company? Anything. Or do you want to share any social connections or where you, formerly known as tweet, or anything like that. Do you like--

ADAM: [LAUGHS]

STEVE: Fair enough, yeah.

ADAM: Yeah. I don't know, keep playing with fun, reliable software.

STEVE: Fair enough, cool.

MATT: Thank you.

STEVE: Thanks very much.

ADAM: Thanks.

STEVE: Bye.

[MUSIC PLAYING]
JORDAN GREENBERG: You've been listening to the *Prodcast,* Google's podcast on site reliability engineering and production software. 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 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.

[THEME MUSIC]