Deloitte.

Architecting the Cloud, part of the On Cloud Podcast

Mike Kavis, Managing Director, Deloitte Consulting LLP

Title:	**Struggling with DevOps? Remove silos and embrace change**
Description:	DevOps can help make software delivery faster and more flexible. However, many organizations still struggle to effectively implement DevOps at scale. The reasons are myriad, but they may include a combination of siloed operations, too much "toil" in operations, and trying to apply old processes to new capabilities. In this episode of the podcast, Mike Kavis and guest, Puppet's Alanna Brown, discuss how organizations implement DevOps successfully and get more value from their software delivery cycles. According to Alanna, success strategies include improving collaboration to remove silos, embracing change, automating what you can, and improving communication between IT and the business. *Disclaimer: As referenced in this podcast, "Amazon" refers to AWS (Amazon Web Services) and "Google" refers to GCP (Google Cloud Platform).*
Duration:	**00:28:44**

Operator:
This podcast is produced by Deloitte. The views and opinions expressed by podcast speakers and guests are solely their own and do not reflect the opinions of Deloitte. This podcast provides general information only and is not intended to constitute advice or services of any kind. For additional information about Deloitte, go to Deloitte.com/About. Welcome to Architecting the Cloud, part of the On Cloud Podcast, where we get real about Cloud Technology what works, what doesn't and why. Now here is your host Mike Kavis.

Mike Kavis:
Hey, everyone. Welcome back to the Architecting the Cloud Podcast where we get real about cloud technology. We discuss cloud with the people in the field who do the work every day. I'm your host Mike Kavis, chief cloud architect over at Deloitte. Today I am joined by Alanna Brown, senior director of community and development relations at Puppet. Alanna, we talked in 2016 about one of these State of DevOps reports, so it's great to have you back on the show to talk about it again. But tell us a little bit about yourself and what you do today at Puppet.

Alanna Brown:

Sure, Mike, and thank you so much for having me on your podcast and it's great to talk to you again. 2016 feels like a really long time ago. But I'm at Puppet and I'm responsible for a number of things including community and developer relations, strategic partnerships, and thought leadership. And I started at Puppet back in 2011 as DevOps was starting to become a thing. And I didn't even really know anything about DevOps then, but I was really drawn to Puppet, the company, and Luke Kanies, the founder, who you know as well, because of the huge potential I saw to change how IT operations works. And our core principle is all about making infrastructure actionable, scalable, and intelligent, so that the teams that operate that infrastructure are more powerful and valued in their organizations, and the teams that consume that infrastructure are also more agile and resilient.

So one of the things I'm most proud of in my career is creating the State of DevOps report, which I've been an author of every year since its inception back in 2012. And if you're not familiar with it, it's actually on its ninth year, and it is the longest-standing and most widely-referenced study on DevOps today. In the past eight years we've surveyed over 33,000 technical professionals from all around the world. Our report has been downloaded hundreds of thousands of times and it's referenced everywhere. So if you've been to a technical conference recently, you've probably seen the stats referenced there. We've been talked about in analyst reports and the whole "Accelerate" book is based on this data.

Mike Kavis:
Cool stuff, and I've got every single one except 2012 here on my drive. And one of the things I did in preparation from this – and I've been writing about this for a long time, how one of the things DevOps focuses on is removing bottlenecks, right, and the bottlenecks are changing as companies mature. And I kind of said, like, at the beginning it was CI, right? How do we get consistent build and how do we get consistent deployment, and then a few years later DevSecOps, and now we're doing a lot of NewOps, and there's all these things. So what I did is I went, and I just took a couple highlights of the executive summaries over the years. I'm going to go through year after year and then I'm going to ask you what are the themes for 2020.

Alanna Brown:
Great.

Mike Kavis:
So there are some consistent themes each year, which is basically validating that DevOps done right helps companies become high performers, but there are specific things – I think the first couple years it was more validating DevOps and validating performance. And then 2014 it started talking about culture a little bit. 2015 there's lean management, a heavy focus on continuous delivery, the first couple whispers of burnout, right, and IT managers are critical in moving the change. And then I looked at 2016; we started talking about the entire software lifecycle, started talking about security and QA and end to end and it's not just dev and ops, right, which was pretty cool.

And actually that was the year we last talked about this topic. QA is everyone's job was one of the key findings in there and do experimentation, another – talk about lean, right? Go do experiments, stuff like that. Well, to do that, you have to have lean infrastructure. Then in 2017 there was a lot of talk about leadership, right, that the bottlenecks – we have teams that do this stuff good, but we can't scale this across the organization, so leadership. 2018 was a big focus on DevSecOps, cross-team sharing, the key practices for evolving, so again, that's more on the leadership; how do we evolve and grow DevOps throughout the whole company? And then last year it was a lot about work-life balance and burnout again. The change management process itself is a bottleneck, right? You can automate everything, but if you're automating and old legacy processes you still have these huge bottlenecks. And then communities of practice, right, again about how do we scale this, how do we get the knowledge out of those first couple groups and share that?

So that's kind of the history as I summed it up really quick. What can we expect in 2020? What are some of those themes?

Alanna Brown:
Well, I'm so glad you asked, and thank you for that brief history of the State of DevOps report. You have really done your homework and there is a lot. I mean, I've written hundreds and hundreds of words about DevOps over the years, and even I have to go back through the reports to remember what we've talked about. But you're right. There are a lot of big barriers to DevOps adoption, and we've been at this for over a decade now. Like, think about that. It's been a decade of DevOps, right? The term was first coined, what, in 2008? And so I think a lot of organizations frankly are still struggling. We see a lot of organizations that have pockets of success with DevOps, but they haven't been able to realize the full potential of DevOps and scale that out more broadly across all of their teams so that it just becomes the de facto way that everyone works together.

And I think part of the problem is that there are so many organizational silos, and one of the things that has been so illuminating for me is doing what I call field work with these teams who are trying to adopt DevOps and getting together with these massive IT organizations, all the leaders from the different silos in IT. And the amazing thing to me is that they don't talk to each other. They don't share their practices. They have no idea what other teams are doing, because they're so focused on their own functional silo. And so that has been really amazing for me to see.

And I think there – I have a whole laundry list of barriers, everything from leadership support, which we've found from previous reports is hugely important because this cannot just be a bottoms-up, grassroots effort. At some point leadership has to be able to meet in the middle and unblock things for these technical teams. Measurement – we know that IT sucks at measurement. And DevOps is inherently hard to measure. And so we do have a lot of statistics and a lot of data that prove the value of DevOps and kind of justify it from a business perspective. The next step, though, is empowering people with the right language. And you really have to be able to speak the language of the other teams in order to get buy-in for this initiative.

Silos – again, as I mentioned these teams just don't talk to each other, and it's a big miss for a lot of companies. I think there's also a perceived skills gap, and software engineering practices are becoming more and more relevant. Now that doesn't mean that everyone has to be a developer, but it does mean that they have to know software engineering practices in order to be relevant and to continue their careers. Process inertia is another big issue. I think one of the biggest mistakes that organizations make and the biggest trap they fall into is as they're trying to implement these DevOps practices, they're also trying to apply old processes to new capabilities. And it's an effort to govern and control something that feels unknown and scary, but sometimes you just have to break the rules so you can make progress.

And I think this requires really strong leadership to give teams the permission to break the rules, to take some calculated risks and to allow for failure. And we always hear fail fast, and I believe that is so true. If you want teams to move fast, you have to decentralize decision making, you have to put the decision making power back into the hands of the people actually doing the work, and you have to expect failure sometimes. And finally I think toil is a big issue and something that is a major roadblock to success. If you're not familiar with Google's SRE book and how they define toil, it's basically any work that is directly tied to running a service that is both manual, repetitive, automatable, and there's no enduring value in this toil. And so the SREs at Google are tasked with spending 50 percent of their time on engineering work, and that's to automate all of these manual time-consuming processes. So I think that's a big barrier.

So looking across all these big barriers, one of the things that we're most interested in tackling for the 2020 State of DevOps survey and report is looking into change management practices. Like ITSM and ITIL are pretty much ubiquitous in large enterprises, and I think they are perceived as a large barrier to adopting DevOps practices. And then we also want to understand how that ties into self-service platforms. So we see a movement towards operations teams developing these self-service platforms that enable agile delivery teams to build and deploy their applications on top of, and we're just interested in investigating that a little bit more.

Mike Kavis:
Yeah, so you hit on a couple topics I talk a lot about. One is cloud platforms, and those can be good and bad, right? They should be good, right – creating a way where developers can come, self-service, get what they want, all the goodness of the security and control policies baked in to keep them safe. What often happens, though, is they build those on their favorite IT service-desk platforms instead of building it on – if they're using AWS or Google on those platforms. So they're doing the right thing trying to create cloud platforms, but they're trying to use technologies that they're used to, kind of what you were referring to, bringing the old processes and technologies forward. And then the other thing I wanted to ask you – you mentioned SRE. I keep reading DevOps versus SRE and I kind of shake. It's like it's not – to me DevOps is end to end. You have a business problem and all the people, process, technology to get that running, and I'm going, "SRE's a part of that." But what's your take on it? I keep going LS is dead, SRE, and SRE versus DevOps, and I'm like, "That doesn't even make sense to me." So I just wanted to get your take on that.

Alanna Brown:
It doesn't make a lot of sense to me, and I think really what we have to go back to is the core principles of DevOps themselves. Those have not changed in the last decade. Culture, automation, measurement, sharing – those are still the core pillars of DevOps. No matter what you call your function, I think those core pillars are so relevant to kind of modern operations.

Mike Kavis:
I agree. The culture and the measurement part is not usually the focus early on. The focus is usually the tools, right, and a lot of times implementing those tools on top of legacy – or sometimes implementing legacy tools, but on top of legacy change management processes, right? One of the biggest bottlenecks is having these 12 review boards and you can't push anything to production till the Thursday meeting where you've got to get and talk to a bunch of people who don't know your app and get approval and all that stuff. And I kind of mentioned this the other day to some friends. This is a once-in-a-lifetime opportunity to build these processes greenfield. Don't bring the old stuff forward. But that's what happens a lot. Why do you think that is?

Alanna Brown:
Well, I think it's the familiar. It's the known. And I think there are a lot of different functions within an organization that feel like they need to have a say in what's being built and what's going on. And there's also a lot of status quo thinking frankly in these organizations. So when we talk about process inertia, we see this so much, especially in large organizations where we have this very involved change management process, but no one fully understands what it is, what the purpose of it is, how they can actually effect change in these processes to make it more agile. Typically what we see is people going a little bit rogue, right? So they will do something. They'll do it ten times, prove success with it, and they'll be like, "Hey, look, I have the proof that this will work without all of this heavy change management process." And that's the only way that they can actually effect change. So that's one of the things that we're looking at in the 2020 survey. We really want to understand how do people effect change. Do they feel empowered to actually change the processes that have been foisted upon them that they had no hand in creating?

Mike Kavis:
Yeah, and a lot of heavily regulated companies that I've been to, there is no flexibility to change that individually. I mean, some of them through campaigning and meeting there's an opportunity, but there's no way that like a developer could say, "Hey, I've done this ten times, and we're just going to implement that." Especially in like banking and healthcare, right? There are strict, strict rules. They're so conservative, right? One of the places I worked at, though we finally got into a value-stream mapping exercise, and we had a good 20-plus people from all over the place. And when we drew this thing out on the wall, I think it was the first time anyone understood what was all involved. And even the process owners of certain things looked at this and said, "Well, this is wrong." Sometimes it just takes visualizing it. As you said, people don't know how to even navigate this process. So do you see a lot of teams taking that approach? I know it's tough because you're crossing a lot of organizational boundaries, but do you see companies taking that value-stream approach and mapping things out and saying, "This is what it is and here are the bottlenecks"? Do you see any of that?

Alanna Brown:
I see a little bit of it, but to be honest I don't see a ton of it, because I think where most people start is where the pain is the greatest. So most people focus on deployment pain, and I think that is absolutely the right place to start, is to really focus on the things that are painful and prioritize them. And that's really where the dev and the ops functions meet; it's at deployment, right? And so that's why there's been such a focus on that. And I think as soon as you start to move further out from deployment, things start to get really complicated. So in our 2019 State of DevOps report, we looked at integrating security into the DevOps lifecycle.

And what we found is that in the middle stages of that integration process, things get so messy. And we've seen this in other reports as well, where the middle stages are always messy, and it's called a J curve, where things start out swimmingly, right? Everyone is kind of boosted by some of the low-hanging fruit and the quick wins and the success that they're seeing, but then things quickly get really, really hard as you start to kind of unlock and dig into some of the underlying complexity that has really been masked over by all of this duct tape, right? And so – and things do get better over time if you push through those middle stages, but it can be really hard to do that, and a lot of companies just don't come out of that trough.

Mike Kavis:
Yeah, and that's why I'm kind of a champion of re-evaluating operating models, right? Because as long as all security people are in a security silo, and all dev people are in a dev silo and all ops people are in an ops silo, it's really hard to change some of those things. So are you seeing full-stack teams being stood up that may have embedded a security architect, embedded a quality person? What do you see out there?

Alanna Brown:
Yeah, and we actually studied this in our 2019 State of DevOps report. And so we looked at security integration, team structure, that kind of thing, and what we found is that teams were embedding security experts, or they have a developer who's very security-minded and is offering guidance on security practices. But I think what we also found that is more important is that this isn't just about – I think there's a big misconception that DevSecOps is just about shifting some security tests to the left. That's not it. This is about fundamentally changing how all of these teams work together and how they collaborate. And so it's necessary to shift those practices to the left, shift security testing to the left, but it is not sufficient to be doing DevSecOps.

So one of the really interesting things is we looked at some of the key practices that improved security posture, and you know, we threw in a bunch of different practices in there, and there were five that emerged as kind of the top five that really improved security posture. And they were all – they all required a very high degree of collaboration early in the delivery cycle. So let me give you a couple of examples. The very first one was security and development teams collaborating on threat models. And if you think about this, this happens in the planning and design phases. This is very early on. And this is really like – it's a verbal exercise where you're modeling out all of these different security threats and figuring out what your risk level would be, how you're going to react to those threats, and really working as a team. And that's one of those things that brings teams together, is having that mutual shared understanding of what's at play.

The second practice was security tools are integrated in the development integration pipeline. So that makes engineers really confident that they're not inadvertently introducing known security problems into their code bases, and it puts that decision making power at – it's very timely, and it's also within the tool that the developers are already using or within the pipeline that developers are already using. So those are just a couple of examples of those practices, but I just wanted to point out that collaboration really is key, and it really does lead to better outcomes.

Mike Kavis:
One question on that is usually there's like a 20 to 1 developer to security person ratio. How can we scale that, right? Obviously security can't meet with every single development team, especially in a larger organization. I know some of that is security in the platforms, right? So it's easier to work with a platform team and embed security. But on the application side, how do we scale that? How do we get enough security collaboration with developers when there's such a different ratio?

Alanna Brown:
Yeah. I mean, I think the first thing is embedding those tools and those practices within the pipeline itself and making sure that security teams – I think training is important, but it alone is not sufficient, because who's going to go back and remember the security training they took in 2019 and be able to apply it to whatever new feature they're building today, you know? So I think the ongoing training and education is really key, but everything comes back to collaboration and sharing. So if you have a centralized security team, which most large enterprises have, their job should be really automating what they can, making sure that they are giving the feedback to the delivery teams on what tests can be automated, so that they can really focus on some of the higher-level stuff, the higher-risk areas of code where manual reviews do need to happen.

And I think the other thing is we always hear that application security developers are one of the hardest skills to hire for. I would argue that the perceived skills gap may not be as big as you think it is. There are always application developers out there who are highly security-minded. I can think of several within Puppet who have had such a big influence on our security practices. And so I think it's really important to make sure that those people are recognized and embedded within the teams.

Mike Kavis:
So last question here. We talked a lot about the barriers, but what are some of the characteristics you keep seeing in the companies that are far along in their journey and are having lots of success? What are those two or three things that you keep seeing over and over in those success stories?

Alanna Brown:
Well, as I said earlier, the DevOps principles haven't really changed over the past decade, even though our understanding of DevOps has changed quite a bit through the research we've done and through seeing how teams have actually implemented DevOps over time. And so the things that haven't really changed is that we see high performers, they are able to move faster. They're more agile. They're able to deploy more frequently. And on the flipside of that they also fail significantly less often than their lower-performing counterparts. And so they have greater system stability, they treat infrastructure as code, and they have good software engineering practices around their infrastructure as code.

A lot of the highest performers have been doing DevOps for a while. And though it's sad to say, a lot of companies have been at this for many, many years, and part of it is because they haven't had a lot of prescriptive, pragmatic guidance, because DevOps really is a set of principles. But I think there are some pretty common adoption patterns that we've seen across a lot of companies, and I think where you start is really, really important. So that's why in our 2018 State of DevOps report we built out a DevOps evolution model which is all about looking at the common adoption practices. And the first two stages are about rationalization and standardization, so it's all about rationalizing what you have today, because as you know, Mike, enterprises are very complex, and they have one of everything.

Like, every software, every platform, every language under the sun, right? And so the really hard work starts at the beginning of being able to standardize and really limit choice, and a lot of organizations don't want to do that. They think of standardization as a dirty word, but it's so important. And then as we look upstream from that, you look at the organizations that are succeeding at building out these self-service platforms, why is that? Well, part of the reason is because they did that work early on to standardize and rationalize, so now they have this limited set of offerings that they can offer and then they can scale more effectively that way.

So those are a couple of things. We do know that DevOps is a competitive advantage and all you have to do is look around and see the organizations that have been doing this for a while and hear how they talk about DevOps, how they're able to deploy multiple times a day. They are able to recover much faster. They're doing things that most normal organizations just can't even conceive of. We do know that automation is just critical. If you want to get rid of the toil, you have to automate, and you have to actual dedicate cycles for people to be able to automate. You have to give them the space to do that.

And then the final thing which I think is always so fun and interesting is that executives always have a totally different view of their DevOps progress than the teams that work for them. And so they tend to have a much rosier view of what's going on. They don't feel the deployment pain. They don't feel the pain as much of the manual work and tend to think that things are going great when in reality they can do a lot if they had – if they were equipped with the knowledge of what's really going on, on the ground. And I think that's a two-way street where the people on the ground and the execs need to have really frank conversations.

Mike Kavis:
Yeah, they just check: We've got DevOps.

Alanna Brown:
Yeah, exactly, because every executive wants to say they're doing DevOps, right?

Mike Kavis:
But some of the most successful companies I've seen, the DevOps movement may have started grassroots but when it took hold and went enterprise level, it was strong leadership from the top driving it. And in those cases I think they're more grounded in reality with what the state of DevOps is in their organization. It's when some person over here runs it and it's not leadership. That may be where they get that disconnect.

Alanna Brown:
Yeah, absolutely.

Mike Kavis:
Yeah, okay. Well, great talking with you again. Maybe we won't wait four years to do the next one, but looking forward to the 2020 report. I understand because of COVID and everything, it's going to be a little later than usual, but still we always look forward to that. All of us DevOps practitioners read that top to bottom. It does take a while.

Alanna Brown:
[Laughter] Slow.

Mike Kavis:
So thank you for the executive summary for sure.

Alanna Brown:
Every year I say, "I'm going to do 20 pages and that's it." It always ends up being like 80 pages. We have a lot of thoughts.

Mike Kavis:
Well, it's good stuff. We appreciate it. So that's our episode for today, Architecting the Cloud with Alanna Brown. Where can we find you on Twitter? And if you want to get these reports what's the best place to go to?

Alanna Brown:
Yeah, so you can find me on Twitter @AlannaPB, and – that's peanut butter. And you can find all of the reports at Puppet.com. Just search for 2019 State of DevOps report and that should take you right to it.

Mike Kavis:
Great, thanks. To learn more about Deloitte or read today's show notes, head over to www.DeloitteCloudPodcast.com. You can find more podcasts by me and my colleague Dave Linthicum just by searching for Deloitte On Cloud Podcast on iTunes or wherever you get your podcasts. I'm your host Mike Kavis. If you'd like to reach me you can always reach me at MKavis@Deloitte.com or follow me on Twitter, @MadGreek65. Thanks for listening. We'll see you next time on Architecting the Cloud.

Operator:
Thank you for listening to Architecting the Cloud, part of the On Cloud Podcast with Mike Kavis. Connect with Mike on Twitter, LinkedIn and visit the Deloitte On Cloud blog at www.deloitte.com/us/deloitte-on-cloud-blog. Be sure to rate and review the show on your favorite podcast app.

Visit the On Cloud library
www.deloitte.com/us/cloud-podcast

About Deloitte

The views, thoughts, and opinions expressed by speakers or guests on this podcast belong solely to them and do not necessarily reflect those of the hosts, the moderators, or Deloitte.

As used in this podcast, “Deloitte” means Deloitte Consulting LLP, a subsidiary of Deloitte LLP. Please see www.deloitte.com/us/about for a detailed description of our legal structure. Certain services may not be available to attest clients under the rules and regulations of public accounting.
Please see www.deloitte.com/about to learn more about our global network of member firms.