007 Clean Code for Federal Projects

SUMMARY KEYWORDS
developers, security, jason, code, software developers, people, applications, vulnerabilities, building, software development, world, pittsburgh, sneak, organization, leverage, tool, john, software, dependencies, libraries

Welcome to the federal check podcast. My name is John Gilroy and I'll be your moderator. today. Our guest is Jason gets he's a senior director of public sector at sneek s and YK and today, we're going to focus on securing software for the federal government. Now before we go on, I know my audience very well and they're going to get a hold of me and they go What the heck is this company s and we've never heard of these guys. Are they from like Southern California? Are they from Mars or who? s and y k Is this a joke sneak so so give us a background here Jason on me. maybe a little bit on you a little bit on this company called sneak. Sure.

05:03
Good morning, John. Thanks very much. So first and foremost, sneak is actually an acronym S NY K stands for. So now, you know. So now you know at least that much about sneak, and sneak is a Boston based company that is working towards enhancing what developers can do around security and application development. So, you

05:21
know, I brought you in because people have never heard of the company and they don't You don't know. And, and I think there's so many good companies active in Silicon Valley or Boston, that can really help the federal people who can't they?

05:33
Absolutely, John, absolutely. And our goal is really to make the world a better place by delivering secure code. So we enable developers around the globe to create software and applications in such a way that it doesn't introduce additional vulnerabilities into their network and into the cyber world.

05:50
Yeah. Well, when it comes to secure code, I think you got a reading and really audience here. And, and I think if you talk about secure code, the first thing people are gonna think about is maybe the software supply chain, and we can name companies that have problem that area. But I don't want to talk about companies like to talk about solutions. And so we know that it's possible that to all kinds of attack vectors. And I think the federal government addressed this sometime last year with the presidential executive order improving the nation's cybersecurity and there was probably a little JSON paragraph and there wasn't there.

06:22
There was John Absolutely. What they talked about. So the things that we're doing around supply chain, and in particularly around open source. So the beautiful part of the story is that developers have

an easier job today, because there are components and chunks of code out there available to them that they can leverage and embed within the applications that they're building. The scary part about that is there are hundreds of chunks of code out there that developers can grab and use within their application. What I mean by that is inherently within open source, there are dependencies, hundreds, sometimes 1000s of dependencies, if you grab a chunk of code that has underlying dependencies that you may not even recognize that are there. So the great part is that there's a lot of software libraries out there that software developers can leverage. But when they leverage those libraries, they have to go in eyes wide open and recognize that their dependencies and sub dependencies, dependencies, entire dependency trees that rely that lie within that code.

07:22
You know, I have a friend who, who ran a health food store, and she'd bring in someone to run the cash register. And it would take, take the person so long to learn it, that she just do it herself. And and so maybe this is the case for software developer, I'm thinking, Well, you know, I can bring in this library, or take 40 to 50 hours more, take this certified, make sure it's correct, or why don't I just write the code? Is this the dilemma? People? Haven't I just write the code myself? It's to be safer that way? Or, or how do you trust? Or who do you trust?

07:49
Well, developers are asked to do more with less like every everyone these days. And so developers are called upon to create more applications create them faster. But there's this underlying theme that as they're creating, they need to do it securely. And that's the tension that exists. sneek oftentimes breaks up the Family Feud, the classic Family Feud, the classic Family Feud within an organization are the security teams and the development teams. And we're the ones that bring those two audiences together.

08:19
Well, what if someone says, oh, yeah, well, we use organizations like Sonatype. And they have certified libraries, we just dragon from there. Is that the appropriate response for this situation?

08:31
Well, again, most people were either developing their own custom code, or they're leveraging the open source components that are already out there. And the focus needs to be on how do you go about doing that in a way that does not introduce additional vulnerabilities. John, there's a very common phrase that you use today, which is shift left, and it's a buzzword, people are saying it all the time. And what they are, what they're trying to describe, there is a movement to take security from the end of the process, and move it to the left, move it earlier in the process. The typical software development lifecycle is I'm going to build tweak, build some more, expand, add on some more functionality. And I'm building building building, I'm putting together this Lego set. And then after I assemble everything, right before it goes into production, I'm gonna scan it. The problem with that is you have this build up where the developer thinks that he or she is doing a great job right up to the point of delivery. Then the security team comes in, which by the way, is greatly outnumbered, you typically have hundreds to 1000s of developers and you have a very small security team. So they are the bottleneck. Everything needs to go through them before it goes into production. So they run the scan, and then they find

everything wrong. They find all the vulnerabilities are within or within that code was created. So again, that tension the family feud that I referred to as the developers don't like the security team, because they know they're doing a great job they're building they're being innovative. They're bringing to bear these applicants Shouldn't the organization that government entity needs for Citizen Service, other things like that, they're bringing that to bear. And then Debbie Downer comes in at the end and tells them everything that's wrong. Meanwhile, the security team knows that you have these, this huge group of developers that are creating all these applications. And they know that it's the wild, wild west, they're creating all of these applications, and they're coming their way, and they need to scan them at the end, nobody's happy, nobody's happy security team is frustrated that the wild wild west created this and introduced all these vulnerabilities. The development team is unhappy, because they just wrote a great application. And they had no idea what they created was introducing vulnerabilities. So hence, you have the scan at the end of the process. Now the shift left movement is to scan earlier, but that doesn't solve the problem, John?

10:50
Well, I let's toss around some phrases. What about baked in security? Well, yeah, is that what you're talking about here is baked in security, what you can provide?

10:58
Yeah, great question. So I'm gonna I'm gonna draw a line have to differentiate here. So when we typically talk about shifting left, what we mean is scan earlier. But that's not the problem. It's like, Would you like to have 100 lashes at the end, or 10 lashes every day building up to that neither one of them are good, right? Neither one are good. So just shifting left and moving the scan earlier in the process isn't going to fundamentally fix what's wrong. What needs to happen is developers need to embrace and be involved in the security process. And right now, they typically aren't. And they aren't, because all they have are the tools that will run a scan and tell them everything that they're doing is wrong. A developer doesn't want to hear that a developer wants to do their job. And they just like all of us, they want to do their job, and they want to do it, well. They have timeframes, they have budgets, and they need to deliver a product on time. And just to tell them what's wrong doesn't solve the problem, what we need to do is empower them to do what's right. And what I mean by that is, as a developer is going through the process, number one going to them, so not giving them a new tool to use a new interface to use, but embedding ourselves within the applications and the tools that they're already using for development, number one, and then when we find something, not telling them what they're doing is bad, telling them that there's a better way to do it so that they aren't introducing vulnerabilities into the network. And so you go to them within the tools that they're using. And then when you do find a vulnerability for them, allow them give them the tools embedded in with what they're already doing to fix that vulnerability right there in real time in the development cycle. In fact, we're, we have a fundamental belief that developers can actually develop just as fast, sometimes even faster, and securely.

12:51
Speaking of buzzwords, I love the buzzword orchestration, I used it yesterday is kind of proud of myself. And my question was, well, I've been down to the National Symphony, and, you know, Kennedy Center there, and, and they orchestrate things, right? They're all on the same page, right? But guess what? That piano doesn't move. Now, you J, you're in a situation where the pianos moving. In other

words, we can get an oh, by the way, is a new vulnerability found oh, by the way, and then there's the Jason vulnerability and so and so it seems all well and good if you say, but this is a dynamic situation, isn't it?

13:23
Very much so. And you're spot on. So not only does a developer have to keep an eye on what they're building today, that they're about to deliver. But they also need to inventory what they've already built, because the piano is moving. Later on down the road, we're going to find an issue that wasn't an issue at the time. But today, it's an issue. So let's talk about solar winds. Let's talk about log for Shell, or spring, those are open source projects that a lot of people use 1000s of people around the globe use embedded that code. And then later on, there was a breach, there was a there was a way that they were able to penetrate into that code and leverage that code for malicious purposes. But the problem is those applications already deployed. So the so what we need to do is give the the development and the security teams a way to inventory what they have out in production and still be able to respond to that. We refer to that as technical debt.

14:16
Mm hmm. Yeah, we know the old the old software development, you know, Sprint's in depth thoughts? Well, I'm an east coast here. And there's a team up in Boston called the Red Sox, and we've got the Yankees, they hate each other. And so you talked about the dev people and the security people. And so we're talking about the New York Red Sox here. We're talking about people who don't want to be in the same team? Ah,

14:36
absolutely, absolutely. And the one team greatly outnumbers the other. Like I mentioned, there's hundreds 1000s of developers that their job day in and day out, is to create and deliver applications that are useful to the government organization, and the security team is much much smaller, but they have the job of keeping everything secure.

14:54
Okay, Jason, wait a Cotton Pickin minute way to cotton make it so let's say we're sitting having a coffee and Tyson's in As I said, we'll look at their big building here. That's Appian A P P I N. I know one of the founders, Michael Beckley, one of the four founders. And well, they say, Well, hey, Jason, the answer here is low code, no code, then that's really the best way to handle this. So how do you work with low code? No code, or is that are that's like the New York Red Sox as well?

15:19
No, it's not. It's not at all. But I would submit to you this, there's no such thing as no code. And low code, what they what they simply are saying is that someone else's code, so someone else is building it, and you're just leveraging it. So yes, there's less code on your part that's custom written by you. But there's still being a code being used. And so further, all of the libraries that are being used, all those dependencies that are being introduced, there still needs to be a mechanism for for assessing those and determining whether or not they introduce vulnerabilities. Now, let me be, let me be specific here, we talk about a risk score a vulnerability score. So every day you get in your car, and you assume risk

every day, you you get up out of bed and you assume risk. So there is going to be some degree of risk. The question is, the fundamental question is, what level of risk are you comfortable with. And that's what we're assessing. So when we provide a risk score, that risk score is based on our intelligence. But it's also based on you as an organization saying, we're not so worried about this, because it's an internal application, but something that we're putting out on our public facing website, we're very worried about that being breached. So that's gonna carry with it a higher, higher focus, a higher risk score.

16:36
I spent many years working with software developers, and many software developers have interesting personalities for 30 hours, and then fall asleep under their desk. I seen software developers with the craziest, craziest habits. So this seems like it's that's your target audience, that quirky, brilliant, somewhat unpredictable person that mean, that's a that's a kind of tough horse to train, I mean, put a saddle on, that ain't gonna happen, buddy. You know, I don't want to go to the software developer. And whoa, whoa, that's a tough crowd, isn't it?

17:10
100%. And, John, I'm gonna be sensitive here. My son is a software developer, Oh, I haven't written a line of code since college. But my son is a software development developer, and I understand that world. But you're absolutely right. Again, back to the point that I made earlier, you, you are going to be incredibly disruptive. If you force that developer to use something new, a new interface, again, shifting left does not solve the problem, what you need to do is deliver a tool to them that they actually want to use. So the GSA actually gave us a really great compliment. The program manager over at GSA, which is one of our flagship accounts said, this is the first time that I've given my developers a tool that has to do with security that they actually like. Further, he went on to say that they hated everything else that he had ever provided to them. And so that's the dilemma that we're faced with here is you just give them another scanning tool to tell them what they're doing is bad, that doesn't help solve the problem, what you need to do is encourage them that there's a better way a faster way to develop and deliver what they're trying to do, while staying secure. So we our emphasis is on delivering a platform that was built by developers, for developers, everything about what we do is developer focused. I want to

18:29
go from shift left to redshift. So there's an AWS conference coming up in a couple of weeks. So listeners come down there and throw tomatoes at you and stuff. Are you gonna have a booth? There?

18:38
We are, we certainly are going to show up in a big way. AWS is, is a significant partner for us. And we look forward to participating in their event.

18:47
So is that mid May? Or what is that actually?

18:50
The date for that is May 24, and 25th. And it's actually at the Ronald Reagan center.

18:54
I've been there many times s and y k face.

18:58
Let me let me back up. The dates are may 24, and 25th. And the location is the Washington DC conference center.

19:06
Yeah, face to face. Yeah, I was at an event two weeks ago, at a conference at the convention center. And it feels like you like you're going back home after 10 years, this feels like I've been away longer. But it's you fall back into the good habits of meeting people and inducing yourself. So they used to have days to fill that whole place up before COVID. I don't forget it now. But it was huge. Back in the day.

19:26
There's definitely a shift and we'll see how it goes with going back to live events. We're seeing that ramp up right now. A lot of the events or the events are coming back to live in person.

19:37
So s and yk.com. Excuse me, sny k dot I O 's. Okay, sounds like a developer site anyway. So So what approach do you use when you talk with developers you talk about in general about software development, or he goes you dive right into FedRAMP? Or do you talk about, you know, the 95 vulnerabilities that's released in March 25? I mean, there's a there's a certain approach with 10 developers, there's 10 different approaches. I mean, some really want to get in the nitty gritty and some are kind of wary.

20:06
It's a great question because we actually have, we're a little bit split brain, we have two approaches. So sometimes we talk to developers, sometimes we talk to security teams, and CISOs. And the message is very different depending on those audiences. So for a developer, again, we are telling them that we allow, we can deliver to them a platform that allows them to deliver the same products they're doing, but securely, that will be that will not be disruptive to their current process. So we go to them for the security team, we are telling them that they can actually provide the developers within their organization, a tool that they'll be willing to use and will actually enjoy using, but will help solve their problems. Because again, for the security team, there's far less people and they have a lot of work to do to scan everything that comes out. And if we don't have that built up to the scan at the end of the process, then it makes their job that much easier.

21:00
When I think of security, I think of brigadier general Greg touhill. And currently, he resides in Pittsburgh, Pennsylvania, where he was born and raised. And he's running this little organization called cert, he's teaching at Carnegie Mellon, I don't know what he's doing. He's writing books, I can't, I can't figure out this great guy can't hold them down. Maybe you can get staple them to the ground, and nobody's doing

so many things. And he should be retired, but he isn't. And so I imagine that you have a strong relationship with public private organizations like certain other organizations, is that true?

21:29
That is true. That is true. So again, we need to go to the developer community. So the way that we do that is through partnerships and through integrations. So if a developer is using Jenkins, or if they're using GitHub or git lab as their repository, again, to, to give them another interface to use that introduces more pain into the process, and we want to be we want to introduce less pain into the process, we want to make it easier for the developer to do security. We don't want the developer to cringe at the idea of, of being responsible for embedding security. So we work with those tools in those companies to create an integration so that we can be present right within the tool that the developers already working in.

22:12
It used to be you listen to people on the radio and podcasts and they said, Oh, that Jason guy, he jumped the shark after his last movie, you know? And this phrase is jumped dish. Oh, yeah. Oh, yeah. He was good up until he jumped the shark or that TV show or something, you know. So Jason, there's the phrase dev SEC ops, is that, is that still usable? Or is that just old Fuddy duddies? Like me use it? Are the the people developing software? Now think, oh, that that's my grandfather's phrase, or my father's Oldsmobile, you know, my father's phrase?

22:38
No, still very applicable today, John. So I would affectionately say we put the second dev SEC ops. So what we're trying to do is create a security platform for the DevOps team to embrace the security process and protocols that have to do with when they're creating applications. So that again, when they deliver that application that actually has security, a security mindset already within it.

23:02
Well, if you put the second dev SEC ops, I put the fun in fundamentals. So let's go back to fundamentals, we started this conversation talking about securing software for the federal government. And what your suggestion is, is that, don't wait until the horse is gone, you know, look in the barn first, and then shift left, which has been to the early part of the software development, and work with software developers to give them powerful tools. How's that for phases? That's something that that's not going to be really difficult to do. And, and I've seen it happen many, many times, you know, when I reach for a wrench, I have my old toolbox, and I go for nine sixteenths wrench. I got the driver use it, bang. I mean, I don't want to go to some kind of mechanical thing. 30 yards or whatnot, I want to grab it use I'm comfortable with. And so I think that's maybe Steve Jobs had, you know, maybe he had the insight many years ago. He said, No, it's got to be easy to it's got to be human friendly, I guess.

23:55
100% 100%. That's the goal to deliver secure this. The goal is to deliver security applications to them in a way that they're comfortable using it. And that's by residing in the applicant, the applications and the tools that they're already using.

24:11
I did an interview last week with a federal agency. We talked about citizen engagement. We talked about their website, and something called UI UX, user engagement, and optimizing for that. And Citizen Engagement says engagement and and it seems like you're on the other side of the coin. You know, I met the software developer user interface. Well, yeah, we do user interface for the smart people. For developers. I just guess it could I classify this Gartner have a Quadrant for that UI UX for software developers? I guess it is, I don't know. You know, I forgot to say that. One of the best ways to keep up with the fast changing world of technology is to attend webinars. You learn at your schedule and you get continuing education credits. The best webinars are from Fed insider.com Just last week, they had some federal health care professionals on. And they talked about protecting essential data. So I would go there if I wanted some free education credits. So is your company sneek participate in these conferences? Now? AWS, they offer webinars, they have a presence on YouTube. How can my listeners learn more about you, Jason?

25:26
You Okay. Back that meter, was that meter? You?

25:35
I think it was you because I think he dropped out and came back. Yeah, I just did the middle read. And we're going to transition here. So let's go 5432. So tell me, Jason, we know you're appearing at the Amazon Web Services conference coming up in a few weeks. Does your company have a presence on YouTube? How can they learn more about snick? Snow? Absolutely.

25:56
Absolutely. So we, because we have a developer community that embraces us, we're actually used by over 3 million developers around the world. That's about 10% of the Worldwide Developers on the planet. Because of that community, we're constantly putting out material for them to use and leverage to learn more. So when you sign up for a free sneak account right off of our website, you can use that tool for as long as you'd like. Embedded within that tool. There's line by line training that, that walks you through how to use the tool and how to leverage it further. But beyond that, resources on YouTube, constant video podcasts that we put out as a company, we actually have a tour that's taking place right now going to major cities around the globe. We're teaching people the importance of developer security and how to leverage the sneek platform. We have a Boston week coming up next week, in our headquarters hometown. We're going to have a bunch of developers in a room for a week talking about the importance of developer security. And we're going to take that again around to several major cities.

27:02
Good, good, good. s and y k dot i Oh, thank you very much. You've been listening to federal tech podcast with John Gilroy. I'd like to thank my guest Jason guest, Senior Director of Public Sector at sneek. Easy peasy, ha,