A Platform-Independent Model for DevSecOps

featuring Tim Chick and Joseph Yankel as Interviewed by Suzanne Miller

---

Welcome to the SEI Podcast Series, a production of the Carnegie Mellon University Software Engineering Institute. The SEI is a federally funded research and development center sponsored by the U.S. Department of Defense. A transcript of today's podcast is posted on the SEI website at sei.cmu.edu/podcasts.

Suzanne Miller: Welcome to the SEI podcast series. My name is Suzanne Miller. I am a principal researcher in the SEI's Software Solutions Division. Today, I am joined by my friends and colleagues, Tim Chick and Joseph Yankel. Tim is the Security Automation System team's technical manager in the SEI's CERT Division. Joe is the team lead of the DevSecOps Innovations team in the Software Solutions Division.

Today we are here to talk about their work on a platform-independent model for DevSecOps. Welcome to both of you.

Tim Chick: Hi, Suz. Thanks.

Joseph Yankel: Thank you.

Suzanne: You have both been guests on our show before, but some of our audience may not know you yet. Let's start by just telling us a little bit about yourself, what brought you to the SEI, and the work that you do here. Let's start that with Tim.

Tim Chick: I have been here about 10-15 years now. I actually started working for SSD, the group you work for still, really focusing on process improvement, software project management activities. Last 10 years or so, I have moved over to CERT. There I really started focusing more on software assurance and applying my background to how do I move good cyber engineering practices more to the left of the cycle where traditional cyber stuff has been really operational focused. That is really where I have been transitioned the last several years.

Suzanne: And Joe, what about you? What brought you here?

Joe: Oh, so many different things I would say. I guess the opportunity to work on many areas in software engineering. So I found myself a bit of a, we call it Swiss army knife, a little bit, and that skill set lends itself really well with the kind of work we do at the Software Engineering Institute, which as you know, Sue, we get to work on a lot of different things. We are always learning, love that environment and I think in this field to continue to learn is what we have to do and this, you know, we get to do that here.

Suzanne: Yes, we do, and sometimes you and I get to get together, which is even more fun. All right, let's start about, before you tell us about the model, tell us about why you started building this platform independent model. You have a recent blog post and in that, you talked about how many of the organizations and highly regulated environments, our DoD environments, banking, healthcare, they are facing a lot of challenges implementing DevSecOps, and so tell us a little bit about the challenges that the model is intended to solve, and, Joe, why don't we start with you on that one?

Joe: That is right. A lot of this was little bit of an idea kicked around. Tim is a primary lead on developing this model. But we quickly collaborated on the idea that we need a source of truth on how to go about really understanding what DevSecOps means per organization.

Suzanne: I note you say for organization because I think what I have read and what I have seen is one of the challenges is it's not cookie cutter, right? DevSecOps, is not a cookie cutter; you can't just say, *Oh, it worked over in this hospital. I can now put this approach into that bank*, and so That is…

Joe: Right.

Suzanne: One of the challenges. Go, Tim.

Tim: Yes, that is a huge challenge, right? And the reason we built this platform independent model is that there are two extremes for DevSecOps. One is more of the academic, the theoretical and principle, what the DevSecOps should be, and then you have the solutions, right? There are lots of organizations to advertise or show their toolset or how they are doing DevSecOps. But every organization is different; every business is different. So how do you get beyond, what is that middle ground between those two extremes? That is really what led us to this model.

Suzanne: So let's talk about the model. What is it? How did you build it, and how significant is it that it is platform independent? You just kind of started us on that but let's continue a little more on that importance.

Tim: Right, so what is it? It is basically applying model-based systems engineering (MBSE) and enterprise architecture principles and concepts to DevSecOps. It is, *How do I caption, and how do I articulate at that next level abstraction from the academic piece to what does it mean to DevSecOps*? DevSecOps is a really complicated socio-technical system. We need to apply good system engineering principles to it. The derivation of the platform independent model and the platform-specific model is that we are not dictating solution. We are not saying you have to use this tool or that tool or the interactions. We are simply saying that a good DevSecOps, the ideal DevSecOps, has these elements in terms of activities, in terms of meeting certain requirements, in terms of certain types of resources, the person or software or equipment, and how do they all interact together. It really forms that basis for someone to make an informed decision on what they need to do for themselves.

Suzanne: OK. Joe, did you want to add anything to that?

Joe: I think Tim said it best. It is very challenging to not just throw a tool at a solution, to really understand what that capability is, so we talked about requirements. We also broke this down into capabilities.

We looked at a typical infinity loop of DevSecOps, and what we realized is that every step in that loop is another infinity loop. Even realizing that via model is a huge benefit to actually understand that, *Hey, I have a team that is going to do, let's say static analysis*. *It looks like a single step*. But as you really look at it, you see there is planning. There is personnel that have to be involved, there are activities that it has to produce inputs and outputs. When we can apply a model-based system engineering approach to it, we really help organizations understand what it entails versus *I have a solution That is going to do this for me*.

Tim: I want to go back to your original one, your original question is why the model, what it is bringing is that it allows organizations, the highly regulated organizations that are very complicated systems. They have safety requirements. They have regulatory requirements. They have to think about how are they going to meet those regulations and how these tools are going to be configured and what are the right resources needed to do those things. One of the major problems I think Joe and I both have seen is that they are not resourcing or understanding the complexity of this half of the equation. They focus on just building the product. They don't realize that that infrastructure that enables the building of their product needs a lot more of their attention, right? How do they articulate it and how do they plan that? The model should help inform that and help feed that planning process.

Suzanne: What I am hearing is two things. One is that this provides an explicit visualization of many of the factors that get ignored when people first say, *We are going to do DevSecOps*. They think of the tools, they think of the networks, but they don't think about all of these other things

that come into play. The other thing I am hearing is that where DevSecOps is enacted, essentially, in the process of building a product, *I want to build a healthcare system*, what we are really talking about with this model is guiding people on how to think about the ecosystem that that product is going to be developed in and eventually sustained in. Did I get that right?

Joe: Very much.

Tim: You got it.

Suzanne: All right.

Joe: Not so much the product but that the supporting environment equals processes, planning that is required.

Suzanne: Yes, yes.

Suzanne: OK. Have you piloted this, and what kinds of things did you see that you were really happy about? The other thing I know about modeling from some of my own background is when you go out and pilot a model you find, *Oh, you said that, oh, that is not a good idea. We need to say that a different way*. Have you had any of those kinds of experiences with the model yet?

Tim: We have done a little bit. One of things we did was is, we used the set of requirements we developed which go beyond that DevSecOps principles. What does it mean to actually do DevSecOps? What should the system or the pipeline do, right? We use that as a basis for going out and doing some assessments of different organizations and say, *Hey, we think we are doing DevSecOps, how are we doing*? What I found very interesting is that it mapped pretty well. I was able to say, *You are really good at x. You are missing y. The reason this other thing z is so hard for you is because you are missing some of those fundamental constructs that enables that other thing to happen*. It really helped frame it for them and give them an understanding of it and material to visualize and understand what we were trying to communicate. Joe and I have done these types of assessments many, many times and using the model as a base, I think, just really helped improve the understanding and actually helped them build a path to achieve what their ultimate goals are.

Suzanne: Right. That is where this becomes organization-specific because you are going to want to make…Those recommendations are specific to the organization. They are not just a canned set of things that the model spits out. It's not a machine learning model that says, *Oh, you are not there yet, so, therefore, you are going to want to know these things*, so you are tailoring. You are using the model as a source but you are tailoring it to the needs of that particular organization. Are there any interesting insights so far that it's like, *Wow, every organization we talk to has this*

problem without fail. To me, those are places that inform SEI research opportunities, right? Anything like that that has come up?

Tim: I will let you take that one Joe.

Joe: I think that the first thing we have recognized from our own work before the model—and this should, you know, ring a bell with most of us—we have run into great heroics. These are individuals that are performing so many different duties. What we broke it down to is there are roles and responsibilities attached to capabilities and activities.

Suzanne: Right.

Joe: As we did manual assessments, paper-and-pencil type of assessments, we found the same name listed as technical point of contact for this activity, for this activity. *Developer A is doing testing. Developer A is helping with infrastructure*. These people get burned out. They are doing great work, no doubt, but it's not…

Suzanne: It's not sustainable.

Joe: It's not sustainable. The model really helped us. This visualization of the activities we are performing and who is responsible, who interacts with the data, and who needs to either be informed by that data or who needs to produce that data, or who uses that data. These are all things that we can model very well. These things really help us understand, *Wow, we need to hire.* Or, *we have got do something different here*. *Maybe we can't do this activity*. This brings us into, we did use an idea of a little bit of a maturity level in terms of there are some really advanced things you could do with DevSecOps. The maturity level allows us to [say], *Hey, we can develop software without doing x, but we might not have that capability. But we want to know about…This may be a more advanced type of security analysis that we might want to do. We can develop software without doing DevSecOps. It has been done, but we think you might be at a different level of maturity*. That helps an organization to understand, *Here is on our roadmap where we want to go,, and here is where we are at today*.

Suzanne: Maturity levels have a history of being economic benchmarks in the past, have you run into any resistance from organizations when you say *maturity level*? And they go [screams] and *No*.

Joe: Certainly, even from ourselves.

Suzanne: Is that something that people seem to want, or is it something that is a convenience, but it doesn't really affect their ability to use what we are offering. The model is much richer than that.

Tim: The levels like the facial expression you are making, right? Joe and I really tried to resist it, but the problem is every organization's different. DevSecOps is this really complicated thing. It's really overwhelming to someone new to DevSecOps or someone who hasn't really matured their pipelines yet. So where do you start, right? So we had to make choices, *Do we model the ideal, or do we model the least common denominator, right?* The least kind of denominator is *I am building software without tools, the traditional waterfall type approach*. So, we felt the levels probably was the only way to go about it. Level 1 is the minimal core competencies one would do just develop software and then you build into, *Hey, got lots of automation and you have got like you need some machine learning going on at the real high maturity thoughts of it*. But it is not designed to meet a benchmark. It is really helping you to know where to focus in our 10 core capabilities we have identified as a part of the model. So it was really just so, *How do I not get not too overwhelmed by the complexity of it all?*

Suzanne: Well, the attraction of those kinds of ideas was always this sort of apparent simplicity with a lot of underneath-the-covers complexity. I am not surprised that there is attraction to that concept.

Joe: We hate to ever see it turn into a requirement of meeting a level for regulation, but we think it's great to shoot for, *Hey, can we use advanced machine learning and artificial intelligence to monitor our solutions and reconfigure our environment?* This is possible today, but it is very advanced.

Suzanne: It is state of the art versus state of the practice, yes. All right, so any other information about the model or using the model if someone were to come to you and say, *Hey, I want to use this model to help improve my own DevSecOps*. Any other information about the model that you would want them to have?

Tim: One is it is freely available. It is public on the SEI's GitHub site. Just link off the main SEI website. It is an interactive model. It is not traditional where it is just, *Read this Word document, or read this published book or whatever*. It really is interactive, right? We tried to use modern model bases of engineering to allow you to click here and really dig down. You get lost, and then you want to come back to the top and dig back down. It is basically interactive. The power of that is all the concepts, all the interactions, it is stated once and referenced where it makes sense. That is the power of models. It is definitely different than a lot of the other stuff out there, and it is free. It is interactive. It is an HTML-interactive model. If someone wants the core stuff, they can reach out to the SEI. We can maybe make arrangements to do that.

Suzanne: What about your goals for transitioning this into practice? So you have made it freely available. That is one of the ways that we enable transition. Do you have other plans for training people on how to use this? Do you have other job aids and things? I know you have got an

assessment method that people can ask you to perform. What are some of the other transition things you are either working on or planning to work on for this model?

Tim: Right. A good bit of it currently is we are still developing it. The current release is the first iteration of it. We are doing some activities to expand just the current definitions that are in there. The other aspect of it is one of the original reasons we have even started going down this path is, *From a cyber assurance perspective, how do I assure the pipeline?* There have been several, SolarWinds being an example, where someone attacked the pipeline to get to the product. So, *How do I begin reasoning through that?* We are struggling about that aspect of it, and we are thinking about what are some of the threat scenarios one has to worry about for the model? Then, how do we capture that and put that in the model as well?

We are still growing, and we are still building the model. In terms of use cases of it, we definitely really want to pilot, we want to try to use it. We do have a few other blogs and other publications coming out to start explaining the models and stuff. I am hoping to eventually draft a blog with you to talk about the levels and the appraisal stuff. I know you shared some material with me. [We] just haven't gotten there yet. That is kind of where I think we are at. Joe, do you have something you want to add?

Joe: Just, yes, we are exploring the space. We are looking for collaborators that are interested in this approach. We believe it is the right approach. Versus a sea of documents that describes processes, this is a really easy-to-understand approach of, *Here are these activities we are doing*. I believe you could offshoot this lots of ways. I think organizations might say, *We are going to show how you train these certain activities*. Or, I think it could go a lot of directions. I don't know if that is up to us. I think that is up to the community to say, *Here is how we would like to use this*.

Suzanne: So there is a concept of reference architectures in various parts of the DoD, and I imagine other domains as well. We didn't talk about this ahead of time, but is this one of the possible uses of this to create a version of the model that essentially expresses a reference architecture? Where the reference architecture documents typically don't talk about any of the activities or processes, you could actually have a reference architecture that included some, *And here is the required resources for this, and here is the kind of competencies those resources need to have*. Is that something you could do with this model if you had time?

Tim: That is exactly what a platform-independent model is. It's a reference architecture at a high level, and what you would do is, is you build that platform-specific model. That platform-specific model would be that program or that company's specific instantiation, but they could actually reference the independent model and show how they have met the requirements or the

activities or the intent of the PIM in their platform-specific model. Or, if they choose not to, they can rationalize why they consciously chose not to do something, which is OK as well.

Suzanne: Yes, we do allow people to do things differently from one program to another. That is always going to be the case. You talked about [how] you are both working on expanding this model and also starting to build some transition mechanisms for it. What is next? What are the areas of interest for you two in terms of your research in this area?

Joe: I'll speak to that. I would like for some more of our experts in the SEI to incorporate their expertise into this model. Tim and I quite often talk about software engineering processes. They are not different for any of our organization. We have got expertise in malware. We have got expertise in insider threat. We are collaborating with those groups, but we'd like to see a little bit more collaboration, even amongst our own experts in how to represent their expertise in the model.

Suzanne: Right. OK.

Tim: Yes. Like, we just scratch the surface on metrics, for example. We have a place where you collect data, but what are really good DevSecOps metrics, right? It really is a…

Suzanne: That is an emerging field.

Tim: It is really a weak area of defined material. That is, actually really good metrics and it's a really, good questions from a DevSecOps pipeline perspective. There is some stuff out there, but definitely room for growth and research there. We have folks who are very focused on that. So how do I get that knowledge into the model to begin expanding the reference material?

Suzanne: OK, so Joe, Tim, if people want to work with you to use the model, either an appraisal or other context, how do they get a hold of you and make that happen?

Joe: An easy way is just to email us at info@sei.cmu.edu. Those messages will get to Tim and me, and we will get back to you.

Suzanne: All right, I want to thank you for joining us. This is exciting stuff to have something new like this at the very beginning of its use. I appreciate you sharing this with all of our viewers, and I can see where there is going to be a lot more activity in this area. So yes, dealing with all the operational aspects of this is what you have ahead of you, congratulations. I want to make sure our viewers know that we will include links to the GitHub and other resources that we have talked about in our transcripts so they will have that. I also want to remind our viewers that you can find this podcast wherever you find your podcasts. We are everywhere. My favorite is, of course, the SEI YouTube channel, but you will have your own favorites.

I want to thank you, Joe and Tim, and thank all of our viewers for joining us today. Thank you very much.

Joe: Thanks Suze.

Tim: Thanks.

Thanks for joining us. This episode is available where you download podcasts, including SoundCloud, Stitcher, TuneIn Radio, Google Podcasts, and Apple Podcasts. It is also available on the SEI website at sei.cmu.edu/podcasts and the SEI's YouTube channel. This copyrighted work is made available through the Software Engineering Institute, a federally-funded research and development center sponsored by the U.S. Department of Defense. For more information about the SEI and this work, please visit www.sei.cmu.edu. As always, if you have any questions, please don't hesitate to email us at info@sei.cmu.edu. Thank you.