# #160 - Secure Developer Training Programs P1 (with Scott Russo)

[00:00:00]

[00:00:00] **G Mark Hardy:** Hello, and welcome to another episode of CISO Tradecraft, the podcast that provides you with the information, knowledge, and wisdom to be a more effective cybersecurity leader. I'm your host, G Mark Hardy, and today we're going to do a deep dive into the world of creating secure developer training programs.

And to assist with that today, we're bringing on a special guest who's a true expert in the field of secure developer training programs. If you're on YouTube, you can see I'm wearing my festive hat and I've got my Marian Platzmug here from my Glühwein. I'm in Munich, Germany, and it's snowing to beat the band out there.

And so, anyway, I'm kind of glad that I'm inside. So I'll risk hat here, take off the funny hat, and, and we'll get going. So, by the way, if you like listening or watching CISO Tradecraft, you're going to love earning CPEs. We've teamed up with the ISACA, Central Maryland [00:01:00] chapter, to bring you live online their 20th annual year.

Or day, actually, with G Mark on Wednesday, the 10th of January, 2024. Starts at 8 a. m. Eastern Time, goes till 4 p. m. And you'll earn seven CPEs, which are good for ISACA certs, ISC squared certs, SANS certs, pretty much every major cert issue out there. So they will document that and you've got that in your record.

What a great way to start the new year. So check out the link at CISOTradecraft.com/ISACA, I S A C A. It goes to our page and you click on that link, we'll take you to the registration. It's very inexpensive relative to what you'd normally pay for a whole day worth of security training. And there's a limited number of seats, so don't wait, do it right now and get your year started well.

So let's go ahead and welcome Scott Russo to the show. Scott, welcome, glad to have you on board. Thank you for being part of CISO Tradecraft. Could you tell us a little bit about your background and how you got on into building out a secure developer training program?

[00:01:55] **Scott Russo:** Absolutely. And thank you, G Mark, for having me on the show. So a little bit about myself. [00:02:00] I've been working in engineering and cybersecurity for over 15 years now, and it's been quite a varied amount of roles. So I actually started working on refrigerator electronics and then went to writing firmware for nuclear power plant control systems.

[00:02:14] **G Mark Hardy:** Good sense, you know, it's Homer H. Simpson, right? You know, refrigerators on one day

[00:02:19] **Scott Russo:** mAkes perfect sense, but I will say it keeps it interesting. , did a little bit of pen testing in there. , we got into learning and development, and that is where I created the program that we'll talk about today. I also done software engineering, spent some years in cloud security, and right now I am a director in cyber risk and analysis.

So how did I get into this, into doing this program with trading? , I actually throughout have always had a passion for teaching and mentoring others. I love to see people grow in their careers and succeed. And you can see that, , the program I'll talk about today really is the culmination of, of all my thoughts and feelings there.

[00:02:53] **G Mark Hardy:** Well, appreciate your passion. I feel the same way, which is why we do these podcasts all the time. And it's really interesting on your [00:03:00] background. And so you can provide an overview of the secure developer training program that you built.

[00:03:05] **Scott Russo:** Absolutely. So the name of the program was the Certified Secure Software Engineer, or CSSE for short, and I built that program for Capital One about six years ago that, that we launched that program and I had a fairly simple goal. So I wanted to change the nature of how developers engage with secure coding training.

A lot of times it's, they're assigned it, they have to take it, they're compelled to take it. I wanted to flip that, and it's, I want them to come to me. Let's make it. Fun and engaging. They'll actually want it. So kind of a wild idea., and I also wanted to have a community that helped build it, engage with it, help others learn.

, that's, so I was striving to have that, that interaction, not just me building and it's my program, but it actually is the community's program. , and there were a few things I did to, to make that program and meet those objectives. So we had things like hands on training with hacking exercises. That's, that's very key.

I had multiple tiers that got progressively more challenging. there were cert, there were [00:04:00] certifications that started with certified, right? and then other ways to celebrate achievement for, for those as they engage with the program in different ways. the other thing is, it did start voluntary, so that was, that was a cool part.

I, I did achieve the goal initially, got to a thousand folks who actually engaged without being told you must do it. however, it did flip a few years in, where the CIO declared, this is a great program, I'd like everybody to do it, should be part of the new hire training. So, yep, at the end of the day, it now is required, but I still like to think the spirit is there.

[00:04:30] **G Mark Hardy:** So it went from everybody wanting to do it to everybody having to do it, which means they don't want it anymore, right? Not really, but it seems that way, right?

[00:04:37] **Scott Russo:** It changes the charm. But again, the spirit of, I just, I want people to still feel like it was fun and worthwhile.

[00:04:43] **G Mark Hardy:** Yeah, now that's a lot to do, but that's a huge, huge impact. What would motivate you to go ahead and create this type of a training program?

[00:04:51] **Scott Russo:** Funny enough, what I was seeing back at that time, and I think it was in most companies, a lot of the [00:05:00] secure development training, it was kind of came as a CBT. It's your compliance check the box exercise. You know, all your typical content that you get with those computer based trainings, the here's your slide show, click through that, at the end here's some multiple choice questions, and hopefully that's good enough for you, right?

And, and the goal is, yeah, you gotta get it done, we gotta report this to a regulator. And, and I'm not knocking that, I mean, sometimes that's necessary. but it just kinda didn't make sense to me of, there's no code in here, right? Or you're not actually doing any coding. Maybe there's a snippet shown here or there, but there's no actual coding.

It was also clear that a lot of people that took that, the real goal they had in mind was, how quickly can I get to the end and be done with this thing? so Really, what I was trying to do is, can I, can we make something better than this that's just so much more fun? And moreover, I always questioned, are people really learning if they're just trying to get to the end as fast as possible?

So that's, that's what it was about. That's what motivated me. Was that part of, I want them to come willingly and I want to make it fun and rewarding.

[00:05:59] **G Mark Hardy:** Now that [00:06:00] does seem like it would be a motivating factor, the fun and the rewarding part. Now, what do you think are the primary objectives have been of secure developer training program as you created that?

[00:06:10] **Scott Russo:** So the most obvious one, but I'm not gonna settle on it, was We want the developers to write secure code and we kind of have to keep that in mind. So if I said it was something else, I think folks would have a hard time buying it. and also I will point out for the more business minded folk, should I have this program?

we are of course looking for ways to save costs. we want to get our developers to the point where they write secure code, feel comfortable with that. there are lots of ways to do that using external vendors and they can get rather pricey. So part of it is cost savings, of course. but really the bigger thing was it's more like a culture change that we're trying to drive.

And training alone, of course, that's not going to change culture, but it does something very specific that we need to help build culture. It empowers people. It reduces their anxiety. So if you're a developer and you're being told you need to write secure code, you need to fix all these [00:07:00] vulnerabilities, the question is, okay, so how am I supposed to do that?

Like, do I have the skills? Do I know what I need? So, if anything, I wanted them to come out of that The contribution to the culture, feeling empowered, I can overcome these issues, I can write secure code, it is not a hopeless lost cause of endless issues going to be assigned. So that's what I saw the rule, it's empower the developer.

[00:07:22] **G Mark Hardy:** Yeah, I like that. Now, now one of the things that people sometimes get stuck on is the term secure developer, because obviously 100 percent secure development seems like an impossible task, because even large corporations that are well funded been around for years, like Microsoft spent a ton of money on this, and they haven't quite achieved it.

Is there still vulnerabilities in Microsoft Windows? No, I'm not knocking them because it's what, 50 million plus lines of code and things like that. But in this context, Of your program, how do you define the term secure developer?

[00:07:52] **Scott Russo:** Yeah, I'm glad you asked that. So, first off, there is no silver bullet, right? The, I want to be 100 percent secure, what does that mean, and is that [00:08:00] achievable? The answer is absolutely not. And funny enough, part of the training that we gave for it even went over that point that there is no silver bullet.

You, we, you're gonna do your best, but obviously people make mistakes. so how do I really define it then? So our developers have this constant onslaught of you need to fix this, you need to patch this, here's a thing that a tool found in your code, here's something that the pen test found. They have a constant onslaught coming their way of fix this, fix that.

and that can be quite overwhelming. I'm not expecting them to get to the point of perfection. You never make a flaw, everything is patched in advance. How would you even get there in reality? But what I want to do is get them to the point where they feel the burden on them has been lessened. They're making less mistakes.

When they get assigned something, it's really, Oh yeah, I know what to do with that. Here, fix this fast, done, no problem. That's what it's all about. And creating that community of volunteer experts, that's also important, right? Because it's the concept of many hands make light work. If it's [00:09:00] always the cybersecurity department will tell me what to fix and how, now you've got this whole, I have to go ask them, they have to explain it, and we're going to go back and forth, did I do it right?

But if we all have this knowledge of what, what to do and how to do it, we're all working towards that same goal. That burden becomes lesser and lesser. So it's not someone who makes no mistakes. And don't make that your metric for how you measure your program. it's more like we get them to the right skills so that it's so much less a burden to them.

[00:09:31] **G Mark Hardy:** Well, that sounds good. Now that's really good definition. I think of what a secure developer program entails. You're not perfection, but you're able to go ahead and make things happen faster. If you find some issues, you could avoid major errors and fix them quickly when they do occur. So, and also, I think, as you said, they're going to guide others as they go through their career.

For your program, what skills and knowledge areas does it cover that would enable that type of a developer?

[00:09:57] **Scott Russo:** Yeah, so It's, there are multiple tiers, [00:10:00] and where I wanted to start was, let's get the baseline in place. Straight to the secure coding, align to the OWASP top 10, that's the first tier, right? Like, we just, we all need to speak the same language, have a general understanding. How does a SQL injection look like?

How do I avoid that? so, so that's where we start, but then we have more tiers. So, you'll get into more complex coding issues, some of the security, vulnerabilities that maybe are less, more esoteric, more challenging to address, and then even begin introducing at the advanced tiers, some of those broader topics that we might have, something like threat modeling or maybe, how do you deal with infrastructure as code and what's the security look like there? So we can branch out a little bit as it goes on, but where I wanted to start was what's the very base? What's the the new college hire that we just brought in? Like what is the very baseline? I just first, first things first, what is the basic secure coding issues that come up and how to address them?

[00:10:54] **G Mark Hardy:** So who would have to take this type of training program? Is it just the software developers? Is it gonna be the administrators of [00:11:00] applications? Is it anybody who builds an Excel macro? What would you recommend? How do you scope this out?

[00:11:05] **Scott Russo:** Now, for the program I built, my focus was primarily meant for the software developers and particularly those building web apps. And why was that? That's. because it was the most common form of software being developed at the company. So let's get the maximum bang for the buck for the program we're going to build.

I don't want to discount though, there are others building different things, similar things, supporting the software that likely need some form of training. but definitely want to make sure I was giving the right value and needed to have the right target audience for that. now it does get interesting because big company, lots of different programming languages, frameworks.

So how do you get coverage? And I know we'll, we'll chat about that as we go on. but yeah, it can get quite complicated how to scope that and who are the right people. but I also had to consider a few other roles to include that really would drive value, right? So I mentioned maybe there's some interaction with the cyber folks.

We need to speak the same language. We have to [00:12:00] have the same understanding. so we did target them as well. Thought about how they might engage, look at things like, hey, a lot of them write Python, so maybe we can show them how it might manifest in things they know, maybe even give them pseudocode. We want to hit a little bit more than just the developers, but think about their whole experience, and what will it take to lessen that burden, and the interactions.

but like, like I said at the beginning, I think the first thing you gotta do is look at Who, who makes sense to train in your company? I, I can't imagine it's the same everywhere. where your biggest risks lie, then you can start to hone that in and, and, and get the right group.

[00:12:34] **G Mark Hardy:** one of the things I've found is we kind of need to understand the effectiveness of any training program. For example, take a quiz on the material, you teach it in the class, you teach it just before the class is taught. And then you get a baseline understanding. This is what people know. Then you teach the materials to the people in the class, and then afterwards you get the results.

Quiz at the end. And if you can show the scores went up by contrasting the before and after quizzes, then you can say that this improved understanding and it was [00:13:00] a direct result from the training. Now, how would you assess the current security knowledge of participants before they start in the program?

[00:13:08] **Scott Russo:** You actually, you hit the nail right on the head with my favorite way to do it. The pre test and the post test, it really gives you a lot of data to show you how they progressed. it lets me gauge effectiveness. Is the course actually working? we'll talk about more metrics of how to delve and did it actually deliver some result to lessen the time it takes to reduce code and how do I go about that?

But ultimately, pre and post test was key. the other benefit of doing that is it lets some people test out right away, right? So if I take the pretest and like, wow, I actually got a hundred percent, maybe, or ninety percent, maybe I don't need to go take a four hour course to learn the OWASP top ten to get that first tier.

so it did have some benefits and, and on the, on the flip side for the folks who maybe they're new to this, They're struggling a little bit. We could even give them some specifics out of the pretest of like, these are things you should focus on. Here's some additional training if you're struggling with the concept.

so that was very helpful for us to set up folks for [00:14:00] success and actually see how they progressed. I will say there is something you must do. So if you want to do pre post tests and you're building a program like this, you got to have a big enough question bank. because what I don't want to be doing is measuring did you memorize the test, the test questions from the pretest and then just regurgitate them.

So you've got to think the bank has to be big enough to use that mechanism. if not, you may look at other, other ways to measure where they are and where they are after, but ultimately, got to make sure you get that. That big enough test question bank to support doing that.

[00:14:30] **G Mark Hardy:** And that, and that's a really good point because otherwise, as you said, someone's just going to do it one or two times. Like, okay, I've seen all of them. Let's just try one more time. And it's a little bit like Super Mario. First time through, you make a mistake. Second time through, you get it right. And you keep doing the exact same scenario over and over again.

How about different learning techniques that you've used in your training program? Workshops, online courses, maybe hands on labs. Is there one that you found more than the others that's particularly effective?

[00:14:58] **Scott Russo:** Yeah, I'll kind of take that as [00:15:00] in two parts for that question. So you asked about, workshops online, maybe offline. That's one of my first considerations. And when I launched the program, I actually started with self paced options. Why did I do that? Because I had to prove the program value, build a team, and doing a lot of in person instruction was going to be very challenging for, for what was essentially a three person team.

the self paced option is actually critical, even in the long term, because there are folks who, hey, I, I would just like to go through this myself. I, I don't need to sit in a class. Let me, let me take my time, put it into my own bite sized chunks. So I started there. very helpful to get it launched and off the ground and allow me to scale.

but then after that, of course, some folks need structure, right? Like, it's hard for them to say, okay, I need to study for four to eight ish hours to be able to pass this test. And when am I going to do that, right? Like, how do I carve out the time? So you've got to have those, training options. And I, I know I started with primarily they were in person instructor led.

I think in, [00:16:00] after the world of COVID, we've got to a lot more virtual classes. That's more so than the instructor led. different techniques you're going to want to use to keep people engaged, depending on how, how you're interacting with them. But ultimately, kind of part two of answering that question, so have different ways that people can engage on their own, in person, virtual.

But part two of that is, for all of them, no matter what, I just, hands on is so, so key for engagement. They need to be interacting with something, doing exercises. If it's Because the early goal I said was, I don't want it to be like a boring CBT, click the slides, then take the multi choice thing, and then you're done, right?

So there's gotta be, the content has to be broken up. So in a class it might be fun where, you know, everybody's energized, we're gonna do a hacking exercise, now we're gonna see how to fix this. But even for the folks who are self paced, give them challenges to do. Give them a place, like a Slack, or whatever your company chat is, to come and ask questions about the exercise and interact with you.

You gotta keep that energy and that momentum going. So, highly [00:17:00] recommend. Less lecture, more hands on, more workshop style. That's what you want to be aiming for. And it does not matter the delivery mechanism you use. They gotta have fun.

[00:17:11] **G Mark Hardy:** Yeah, and I like those recommendations. I mean, I could see how that might catch a recent college graduate's attention as a new hire. I mean, not just to say that the only people who do games and things like that like to have fun, but it kind of lines up to say, hey, this might be a nice place to work. And oh, by the way, it can be brought up to speed in a way that that works well for people.

Now, one of the things that's always a challenge with training is keeping everything up to date. And so, particularly in this environment, where we have evolving security threats and technologies, how do you keep that material current?

[00:17:42] **Scott Russo:** Yeah, that's a challenge, right? So it's funny, it mirrors what happens in the world of the software development too, right? Because they're constantly having to patch, fix bugs, add new features, change features. Same thing for the training program. I will emphasize, it is certainly not set or forget, right?

[00:18:00] Maybe you build it and launch it, but know that forever after you need to keep it current. so there's a few different things that I look at. depending on where, where's the best investment of the time for our internal team versus vendors we use. so what I'd like to do with the internal team, our goal is make sure we know what needs updating.

Like we have to constantly be on top of Is the content, did, did the OWASP top 10 change? Will we better be updating the content so that it, it's not the, I'm using the 2013 version. Oh, great. Wonderful. Right? I gotta be using what's current. our central team should keep on top of that. The nice thing about once we had a volunteer community, that was part of the reward, right?

Like, if I build something and contributed, that's a great thing for me to say. I, I have some clout now and recognition. so we can, we can give those things out like, Hey, I need. Someone to add some new content on insecure deserialization or something like that. So that was one of the ways we did that.

And think of it like open source software intersourcing, right? Like we definitely leveraged the greater [00:19:00] community as it grew. But there's one other thing to think about, right? There's certain materials that Maybe it doesn't make sense to maintain on one's own, right? Like we don't need to reinvent the wheel on how do you explain the OWASP top 10?

There's plenty of plenty of material that's out there. So I'd like to use I like to be judicious, right? Like some of it what's the secret sauce the stuff that's for my company that would be specific or that Kind of aligns with the culture that we have here. That's the stuff I should be focused on internally.

And then what's the like, I don't have to redo an entire OWASP top 10 explanation course. There are plenty out there to source from. So that's the other area that I looked at. As well, a lot of stuff from OWASP in general to use. So finding that balance. really helps as well to not have to put the entire burden of maintaining the currentness of it.

The other one thing, like tangibly, there are some very key things to keep up to date and keep a pulse on. I [00:20:00] mentioned the company has a lot of different languages and frameworks they use, right? So that does in fact change over time. The preferences there, like yearly there's a new hot thing, right?

People want to know about that. They're engaging. There might be traction that's gained there. You have got to keep that content relevant, right? Because the second you get, like, nobody writes in that language anymore, and you don't cover this language, you're starting to lose learners and, and the allure of your program.

So that one you absolutely must keep on top of.

[00:20:29] **G Mark Hardy:** And I agree, that's really important to do so. Now, because there's some things that change and some things that really don't. I mean, for example, validating user input. That's got to be a key protection mechanism for really any app, because, that never really changes. You can't trust users to only put in valid input.

They're going to screw around if they can. If they're malicious, they'll really screw around. But sometimes, just errors can go ahead and be interpreted incorrectly. So we have to sanitize it. We have to ensure they're not putting in single quotes to create a SQL injection attack [00:21:00] or something else like that, or directory tripper.

I mean, all kinds of stuff we can come up with, but can you describe any real world security incidents that your training program has actually helped to prevent or mitigate?

[00:21:10] **Scott Russo:** Yeah, I love that question because it's a really difficult one to answer, right?

[00:21:15] **G Mark Hardy:** Yeah. Don't give away any proprietary stuff like

[00:21:18] **Scott Russo:** there's

[00:21:19] **G Mark Hardy:** almost had a nuclear power plant blow up, but it didn't because of the code we wrote.

[00:21:23] **Scott Russo:** Yeah, we're not, we're not gonna talk about that. But anyway, so, yeah, it, what I would have loved and always strived for but was just seemingly out of touch to be able to really share widely is connecting the train directly to Someone didn't make this mistake and thus we prevented this attack, but it's kind of difficult because of that.

Well, how did you know that they didn't make the mistake because of your training or that they were gonna make it in the

[00:21:46] **G Mark Hardy:** And that's the entire cybersecurity budget argument in about one

[00:21:49] **Scott Russo:** every time, right? But actually there was some good ways to do it and bring in the real world Even if I'm not gonna say I specifically stopped this attack, right? So there's a couple ways I thought about it [00:22:00] First was there are plenty of news articles always floating around.

This breach happened, that breach happened. sometimes we're lucky and we get a little more detail on how it happened and then that just works beautifully to let's make sure we get a new example into the training. When we do the hands on exercise, we can reference what happened in the real world and and like here's how you're seeing it manifest and why you need to do the security.

That why part becomes so much more. Compelling when you make it real. Now on an even closer and personal note. So I, I, I count myself lucky when I teach these courses. I had those years when I was doing pen testing. I have plenty of stories, some I can tell, some I can't tell. but lots of ways to connect.

Hey, by the way, I did actually see this. Here's how it really manifests. and I have one that, that's so much fun for me to tell. It's CVE 2016 6127. It was a very interesting stored cross site scripting vulnerability that I found that involved uploading a file that I'd specifically tampered with. Very, very interesting stuff.

kind of [00:23:00] novel. And they, the learners, they love it. When you tell those, those real stories, and then you can show them, like, I know this sounds far fetched, but here's exactly how it happens. you just gain so much more trust from them, so much more engagement, and it is fun, and they'll ask you lots of fun questions.

Now, I, I know probably not everybody who's endeavoring to make a program like this is gonna have their own CVEs that they can go tell their own stories. but what I would suggest, and what I actually even did in my own program, talk to the pen testers, the offensive security folk in your own company, right?

They have got to have some examples. And again, they'll probably have some, I can't talk about this one, but I can, here's a general one I can talk about. they just make the greatest guest speakers. And then the whole audience is like, wow, what else did you hack? And you can see that sudden interest. And the most important thing for that, so it doesn't teach them how to secure code better, but it convinces them why they should.

And that in itself is very important.

[00:23:54] **G Mark Hardy:** Yeah, that makes good sense. I mean, having good stories on how training will actually stop an attack is really kind of what

[00:24:00] it's all about, and that'll keep your program funded, too, because you're able to tie that cause effect back as best we can. Now, how about industry best practices and standards? Are there any that you'd recommend so the program aligns with it, like an OWASP Top 10, this cybersecurity framework?

What's out there that you would definitely try to align with?

[00:24:17] **Scott Russo:** Yeah, that's a great question too. So, OWASP top 10 for sure. And that's going to be the one you want to make the most, like, facing to your learners. How is this directly tied to the top 10? And I say that because one, it, it makes it simple for the learners. OWASP does an absolutely fantastic job, explaining the different vulnerabilities, providing examples, cheat sheets for prevention, just.

Everything you could ever want, frankly, to support your learners and reference. So why not make your life easy? They also have a great, tool called OWASP Juice Shop. So if you want to get your learners hands on, hack a thing, do a hacking demonstration for them, you got Juice Shop. You can do it right there, have them play along with you.

so I [00:25:00] love it because it just makes it so simple for me to explain the concepts to my learners and have all the tools I need to support. Now, behind the scenes, there's a little more to it, because eventually If your training program takes off, suddenly it's going to link back up to compliance, whether you like it or not, like, so how does, show me how this maps to, and then NIST is going to come in, MITRE is going to come in, and it's great content.

I know I have read and enjoyed NIST 800-53, very long document, a lot of great content in there. I've done mappings to it, I've done mappings to MITRE ATT&CK, all good stuff. Stuff you should be internally understanding and how it relates to your training. but probably not the thing you want to put in front of your learner.

I would absolutely not ask my learners to go read NIST 800 53. Because that would go right back to the oops, I've lost you because I just made you delve into more into my world than you probably wanted to. So, balance. I guess what I'm saying is use OWASP Top 10, but anticipate the fact you're going to be asked about the rest of the standards and where they align and do use [00:26:00] them.

They're, they're quite useful.

[00:26:02] **G Mark Hardy:** Yeah. And I think you can show that to your auditor or maybe your regulator, but saying, Hey, look, our standards all map out. But if you'd said, don't say, boom, here's a couple of mispubs. Have fun with that.

[00:26:12] **Scott Russo:** Yeah, the learners don't really want to read that stuff,

[00:26:14] **G Mark Hardy:** Most people don't. Well, it's great, great material. Now, Ron Ross, I'm going to try to get him on the show.

He's been at NIST for over 50 years. The guy is just sort of a living legend and I'm looking forward to having him on the show in the near future. But he's been over there, but you know, let me, let's not digress. Let's, let's focus with your program. Okay, because those are great standards. And overall then, if somebody is going to get into your training program, how long is it going to take from when they start to when they're completed?

[00:26:41] **Scott Russo:** So, well, let me, let me flip that one around a little bit, for someone thinking about building one. how long do you think the learners, managers and leaders are going to give you is the first question you should ask. So, this was kind of a funny thing for me when I started to launch the program. I did have to wrestle with that question, how [00:27:00] long can the training be?

And where I started was, well, how much content do I have, at least for the first tier to get the, through the whole OWASP top 10? I had about three days worth of content. and then I did some, like, experiments, proof of concept, let's launch it with a small focus group, get feedback, let's see if I do it with different amount of time, what's the likelihood that people, how many people are actually going to sign up for this, versus say that's too much.

So. Four hours, that's what I found out. If they're going to engage with my program at the first tier, and this was when I'm launching, right? So I don't have all the name brand recognition yet and it's not well known. Most of, the most that they're going to give me was four hours. So then the question became, well how do I fit it into four hours?

Because I have three days of content. And that is actually what drove me to make that tiering that I mentioned earlier. I had to split it out. So, okay, I absolutely cannot hit the whole OWASP top 10 in four hours, but what can I get? Can I get some basics down, the injection, the cross site scripting, that kind of stuff.

And then my goal [00:28:00] was. Make it like, well, that was great. I actually want to keep going. so that's actually what I did was the hook had to be four hours because that is literally the amount that I was able to get for people to commit voluntarily to come to my program. but don't get me wrong. I very well was aware.

That's like the first milestone of the accomplishments. We got to tear this out because if I stop there, I won't have fully addressed the, the needs here.

[00:28:25] **G Mark Hardy:** Yeah, so progressive learning, instead of playing it at 6x, going really, really fast. make it sound like a used car salesman. But when I think about this type of training, there's probably some things if you're going to be a trainer, you're going to need to buy, right? And so what type of resources do you buy and then provide to the participants during training?

You have to buy tools, you have to buy software, you have to buy manuals and things like that. What do you, what do you need?

[00:28:48] **Scott Russo:** So you can really go. Quite low cost initially, in the launch and in the training and getting people hands on. I mentioned OWASP Juice Shop, so that was by far my [00:29:00] favorite way to get people to interact and to actually do the hacking exercises. It costs you nothing. what I would do is have all my learners set that up on their laptops in a container and then they can follow along with me, do exercises, we can peer into that code.

so you can actually go fairly Cost, let's just call it very cost effective if you need to, and maybe that's important in an early day. Prove the value before you get the budget sometimes will happen, right? So, there is actually a lot that you can do. the other thing that I would highly recommend, So I mentioned like don't go reinvent basics of OWASP top 10 beginners course, right?

Don't reinvent that. That's, you will spend more time, you'll spend more money in your time doing that than you would to just find one, but what I would imagine is for a lot of the learners endeavoring to make a program like this, your company probably has some learnings that they offer, like a generic learning environment, maybe it's Udemy or something, I don't know.

But, they probably have something, right? [00:30:00] More than likely, in there is already a course on the OWASP Top 10. Start there as well. So, because that's something, maybe you're not using that in the course, but maybe that's like the, hey, if you need more before you start. Here's a good resource that can explain it to you. It's not free, right? Like the company's paying for it. But from my point of view in my program's budget, that was a free resource that I had available to me. So a lot of it started there. Now the thing that I really needed to buy, I mentioned you've got to have an updated bank of test questions. And it has to have enough questions that your folks aren't just memorizing from the pre test.

that is Surprisingly expensive to develop on one's own. Having a good, effective question text bank, keeping it updated and rotating enough, could take a lot of time to build and again time translating into money. So that one is what I would recommend sourced from a vendor. It is worthwhile at the end of the day.

The costs are going to vary depending on [00:31:00] what what vendor you use, but to me that was a no brainer. It's economies of scale thing. I'm never going to, alone with my small team, am I going to keep a test bank across multiple languages and frameworks that's up to date? Or is it worth it just to pay the per seat cost?

And that's where we're going to go to do our exercises. So that's, that's one thing to think about. because truly, if you, you could punt that and put that part in the slides, but then I would question, did you actually make it better than the CBT that probably did the same thing? so that's, that's worthwhile to me to where to make the investment, but don't get me wrong.

There is a lot one can do without having to spend from your budget first.

[00:31:35] **G Mark Hardy:** Yeah, I mean, as you were talking there, I was, I was thinking to myself, well, hey, get every, each student at the end of the course to give you two or three questions with some answers. Things that they thought would be good, but you mentioned the maintain part and that's a hard part. You could probably collect a lot of this stuff and then you got to sift the good ones, but it becomes too much work.

I can see that. And so.

[00:31:54] **Scott Russo:** I did forget one very important thing. I did have quite the budget for shirts and [00:32:00] stickers. You'd want to promote your program, you'd want other people to see your program. So, I did have a bit of spending. Things you give them at the end of the course, and maybe when they pass, you gotta give them laptop stickers and things.

It is worth spending because other people see that. What's that on your laptop? Tell me about that. Hey, this is a great training I went to. I loved it. You should go too. So, don't forget to mark it.

[00:32:21] **G Mark Hardy:** Good. Very, very good point. Now, at the end of the day, how do you measure the effectiveness of this secure developer training program? I know we talked about the before and after quizzes, but any other metrics that you could use to measure this progress?

[00:32:36] **Scott Russo:** Absolutely. So, I am a practitioner of balance scorecard. there's a great article from the 80s on that. I don't use it directly, but more so the principles of it that we've got to have metrics that look at a few different dimensions so that we're Really holistically thinking about our program. So, there's a few things.

And, where you are in the maturity of the program will matter here, right? So, I'm thinking, kind of like, from the beginning, it's voluntary. [00:33:00] What am I trying to do? I want to get engagement. I want to see, like, if someone comes, do they recommend and bring more people to my course? Or am I slowly tapering off after each course I do?

so that was one of my bigger ones at the beginning, was simply how many people are actually attending and certifying. That's telling me what the growth is looking like, that's letting me plan for, wow, I'm gonna have to have a lot more courses, or, or maybe the courses need to be in this location, so that was one of my key ones at first.

But I can't just be growing endlessly and my budget growing endlessly, right? So I have to balance this with some other things. So one of them was, of course, what is the cost of my program? As best as I could estimate it, of course, like, it's a little bit fuzzy, but, What's it costing me for each learner that I certify?

And I am incorporating things like the time that people are investing. We need to respect their time. So we've got to look at and understand that. Like, is it on average they're spending four hours engaged in the training? Eight? Twelve? Where, where, what is it really taking them? Because that's money too, right?

and then the other thing, of course, is what was their experience like? I personally like to use that promoter score question. So like, [00:34:00] would you recommend this? Maybe I'll have some other questions that are a little more detailed, especially with the focus groups, but I'd like to keep those simple.

But just would you recommend it, right? That's telling me how satisfied they are. And as the program progressed, I can even start breaking that down by, by different dimensions. Like maybe one course I've introduced something new into the content. Did we, they like it better? They like it less. So let's see what happened there.

Maybe I've got different volunteer instructors and perhaps some of them seemingly their scores are lower than I expected. Always interesting conversation with volunteers because you got to appreciate they were brave enough, but they sometimes they deserve some feedback too. Like everybody deserves honesty and

[00:34:36] **G Mark Hardy:** they're helping you save your budget, right?

[00:34:38] **Scott Russo:** Mm hmm. So they are, but you also want to make sure they do a good job, right? And so, these are all the things I could see. That's kind of how I like to balance my metrics. Now, there was a couple more things that I always wanted, but just couldn't get in a way that was useful. if I could have got two more metrics to mix in, I wanted to know, what does it look like for each [00:35:00] developer?

Like, they've taken the course, how many mistakes were we seeing in their code before, and how many after? And then the other thing I would have really loved to have was what's the average time it's taking to resolve these issues that are coming up in their code. In practice, when I attempted to get such data, I ran into a few complications, and I won't go into all of them, but I'll give like one example of it.

There's a lot of mess in the data with false positives, so how do I treat those? What if the developer ignores the false positive? What if they actually flag it? I don't, I don't know. So what I found was the data was so messy that if I tried to make any conclusions off of it, it was unlikely I would make a conclusion.

Correct conclusion. I would say it like this. We'll love to have that metric. Also, in the six years since that program launched, I'm sure that data is a lot better. There's a lot of more effort that went into false positive cleanup. If we looked again, maybe we could have it and have what we wanted. So if someone's striving to make the program, absolutely strive for those things.

But don't, also don't, if the data is not good, [00:36:00] don't make decisions based off bad data or they will also be bad decisions.

[00:36:04] **G Mark Hardy:** Yeah, clean data is always going to be useful for decision making. Now, one thing that I've seen that's really important is the gamification techniques that will recognize achievers who complete the program. Okay, there's game based learning, there's gamification, there's a couple other terms that are used out there, but what certifications or credentials can participants earn by going through your program?

[00:36:26] **Scott Russo:** Yeah, that's, that's great. So the program had multiple tiers, and each one was its own certification, and actually the advanced tier broke up because we got into specialty topics, so there was multiple little badges you could earn. so there's a lot of ways we delivered that, right? You did get a certificate, we would send that to your manager to make sure you're being recognized, we were reporting out by department, we want, and especially in the early days when people were doing it voluntarily, it really was a differentiator.

obviously a little different when you're compelled to do it, but early days, like People were proud to have that badge on their profile and get that email and show it [00:37:00] off. So, so that was helping. Now, the interesting part about gamification. I learned from this just how competitive folks can get. So, one of the things that we did in the early days was, we had some tournaments.

Who could secure code the best, who could do like, we do like a capture the flag invite that you'll get, if you do really well, and people were very competitive for that. And really want to be the top of the tournament to the point where I had to turn people away at some point of like you've won the last three, it's time to let other people participate.

I was, I was actually kind of blown away by the gamification really does work. I won't say everybody, right? There's different learning styles, but some folks were like, They wanted to be the top and compelled them to go further in the program. And it was, I mean, they made great, folks as part of our community to teach others and guide others.

but yeah, it, it absolutely works and people love the competition. It's, it's amazing.

[00:37:57] **G Mark Hardy:** Yeah, I, I've seen it in other areas. I've been, going [00:38:00] through Duolingo. I've been working on, one of my language lessons and I am now This week in the finals, I guess you go 10 leagues to get to the diamond. You think he made it said, Oh, and then you got quarterfinals. Now this is the final final. So then once you win, it's like, what do you get?

Oh, you get some gems for the game. I mean, nobody comes to your house. President doesn't decorate you. You don't end up with big confetti or whatever, but it's that competitive spirit to say, Hey, what do you mean? I got. Six hours ago and this guy's a hundred points. Yeah, I could get things such as that. So you stimulate that natural Type of competitiveness as long as it doesn't become obsessive.

Okay, then you could spend too much time on almost anything But I think in general the mindset of being able to go after that competitive nature is a winner and adding that games and just in a friendly way. We don't people getting into fistfights and things like that. But it also be nice when you talk about getting this certification and getting that those little letters, you can put that on your resume and you're in performance reviews and just say, Hey, look what I've accomplished.

Now, not everybody comes in the door at the same starting point. [00:39:00] As you said, some people, you might go ahead and give them a pretest and said, Yeah, you're good. Don't worry about this. But other people are like brand new. So how do you go ahead and tailor this training experience to these different levels of beginner, maybe somebody who's intermediate?

Or maybe that advanced person who still has something to learn.

[00:39:17] **Scott Russo:** So, it's kind of funny because I mentioned why the tiering came about, which was actually to solve my, how do I crunch this into the time that I'm gonna be able to get from folks, but it thankfully also provided part of the answer to the question you just asked, right? Because Given there were multiple tiers, I could actually have different entry points for folks.

there was one other thing, so I'm also double dipping with the, we had the pre and the post test, right? So, that, these two things came together to help me solve that problem. I, so I got metrics and all of that, out of that, and that was all great, right? But what I also got was, when people scored so well on the pre test, I basically just said, okay, cool, take the real test, pass.

Let's get you to the next tier right away. Skip, skip [00:40:00] the beginner training, skip the intro training, just go right to the next tier. so I could accelerate people through it. In general, we also, we also didn't say you have to get the earlier tiers to start an advanced content. Like if you really want to go learn right now about infrastructure as code, that's your thing.

Maybe you don't, maybe you aren't actually doing software development in the traditional web app sense. You, you're allowed to do that too. So we let people enter in different ways. And, and let's not discount there, you know, there's one other path. So, I could also handle the situation, you know, there are going to be folks that like, they don't, they've never seen the OWASP Top 10 before.

It's getting rarer, it's definitely taught in college for the software engineers now. but you know, some of them it's like, I really don't have the background, I don't know, what is a, what is a CSRF? Help me here. so we could see that kind of come out during the early test, and even direct them on a separate path that was, Let's start here.

Here's a couple hour course. It's going to explain everything. Now you will have the ground and be able to move. So we had a lot of different ways and we'd guide them based off that pre test to the right entry point. and that, that really helped a lot. [00:41:00] That plus the tiers, you could go right to where was right for you.

[00:41:03] **G Mark Hardy:** That makes sense. I know the FCC does that. When I went ahead and get my, ham radio operators license, you sit down and you take the technical and if you pass that, they said, okay, fine. You want to try this one? If you do that, you go ahead, you know, you have the general and you go all the way up. to the advanced if you want and go, you know, go right through the whole series in one setting and walk away saying, Hey, there we go.

I'm at the top of the tier. and for those who people who might be interested in that I've been in several Security conferences where they've actually had those licenses and they do come from a large question bank, so there's a couple approaches you said you could try to learn all the questions and the answers You don't know anything.

You actually kind of study the fundamentals of what's behind it and it's the difference between Knowing the answer to the question and being able to provide a solution, because that says you kind of thought about that.

Well, hope you've enjoyed the show up to this point. We've got a whole lot more information. And so what we're going to do is we're going to split the show into two parts. We'll bring Scott back for part two. We thank you for listening to our show CISO Tradecraft. Have you [00:42:00] enjoyed us? Give us a thumbs up on the channel that you listen to five stars.

If you think we're worth it, follow us on YouTube or LinkedIn. We've got more information for you out there on LinkedIn and, go through our chapter notes that we put out there in terms of the podcast. You can find more information there. So this is your host, G Mark Hardy. Until next time, have a great holiday season and stay safe out there