Carnegie Mellon University
Software Engineering Institute

SEI Podcasts

Conversations in Artificial Intelligence, Cybersecurity, and Software Engineering

API Security: An Emerging Concern in Zero Trust Implementations

Featuring McKinley Sconiers-Hasan as Interviewed by Tim Morrow

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.

Tim Morrow: Application programing interfaces, more commonly known as APIs, are the engines behind the majority of internet traffic, more recently API vulnerabilities due to design flaws, incorrect deployment, or business applications that made them a target for actors. Welcome to the SEI podcast series. My name is Tim Morrow, and I am the Situational Awareness Technical Manager in the CERT Division at the SEI. Joining us today to talk about APIs and zero trust is McKinley Sconiers-Hasan, an engineer also in the CERT Division who has been leading our research in this area. Welcome, McKinley.

McKinley Sconiers-Hasan: Thanks, Tim.

Tim: Although you have already published a podcast, several blog posts, and technical reports, our audience members probably aren't familiar with you. I was wondering if you would tell us a little bit about yourself and your work.

McKinley: Yes. I have been here at the SEI for about two years now. Prior to the SEI, I was a hardware engineer mostly doing hardware verification. Moving to the SEI was my first foray into cybersecurity, and I have been doing situational awareness stuff ever since.

Tim: Oh, and I appreciate that you made a difference within our team, so that is wonderful. I wanted to ask you to give us a quick refresher on APIs. I know you have written a lot about them, but I thought for today's audience, if you could tell us a little bit about what an API is and how about three of the security issues you have talked about in the past?

McKinley: Yes. So application programing interfaces, like you said, are the connectors between a lot of internet traffic. For example, one of the most common uses of them is logins. So an API will have a very specific inputs that it needs to take. In the case of login, it would be like a username and a password. Then it takes that information, and then it could access the database to then provide, in that case, something like a token. That is one of the commonly used ways APIs are used, but there are also things like administrators. There is machine to machine communication. APIs, as long as they have the information that they are supposed to have, they can access certain information, request certain information and things like that. That is the basic idea of what APIs do.

Tim: OK. Could you tell me a few of the security things that you have identified that people should be aware of?

McKinley: Yes, with security, with APIs, like you said, APIs are really, they are becoming more and more common, and so they increase the attack surface of a network. If you have more APIs, you have more entry points into the network, which can cause security issues. There are also cascading failures, which have something to do with microservices. Microservices are specific, tiny services that usually have APIs connected to them that do a very specific task, like a login or a password reset or things like that. But if those APIs are too interconnected, then it can cause failures down the line. That is also an issue, that is called cascading failures. Another common one is third-party integrations. If you are using code that is used from a third party or you are using third party APIs, and you are not aware of what goes into that code and its coming into your network, that can also cause security issues.

Tim: OK. Thanks. I appreciate the refresher. I thought that was very good. So we are wondering about what does API security have [to do] with zero trust? I think when I look at the commercial products being offered today, there is a lot of focus on identity and access management, micro-segmentation, dealing with data protection, maybe some configuration compliance. Those seems to be the heavy emphasis from commercial products. I am like, *Well, where does API security fit in there?* I know one of the pillars for zero trust strategy is the application and workload. I was wondering, can you help me better appreciate how that fits into zero trust?

McKinley: Yes, I think APIs fit well into a zero-trust architecture. Zero trust relies on the idea that the internal part of the network should be as secure as the perimeter. Unlike traditional cybersecurity, which mostly just focuses on the perimeter. The internals of a network often have a lot of APIs, APIs that are specifically working in internal networks, some that are public facing. They have a lot of different uses. I think that really connects well with the zero-trust architecture in ensuring that even the APIs that are only communicating internally are also secure. I think that is part of the reason that zero trust and APIs work well together is just making sure that no matter what an API is communicating with—whether it is public, whether it is a person internal to the network, whether it is another machine—that it should still be secured.

Tim: I get a good sense of that. It seems like maybe the API security is something that tends to be more of a mature operation or focus of organizations. I think a lot of the things I mentioned earlier you hear about identity, but when you start to get into the inside of how things work and stuff focused around APIs that seem to be something a little bit more mature. I think it is very timely that we are discussing this subject today.

Looking through some reports…Part of my job is to look throug different reports and see what things are trending. I found a couple key findings I thought were important from a Gartner report, it was identified as how to avoid the top five API adoption pitfalls. I was just going to tell you a couple quotes and see what you think about them from your research.

One is *Poor API implementation leads to low-quality APIs with inconsistent response times and availability and exposes security risks that make it hard for developers to use them. This results in a lack of trust in using enterprise APIs.*

The second one was *Insufficient focus on API security exposes the business to security risk that can have significant financial and reputational impacts.* What are your recommendations based on those two findings? How do you respond to that or what do you think?

McKinley: I think yes, they touched on some of the big issues with APIs. There are a few main components when you are making a new API. First of all, it is the design of it. It has to work well with your business purposes, and it has to be designed not only on its own but within the network. It has to fit well functionally and security-wise within your current network. Then there is the actual design of the API. With that, you can use secure coding practices. You can ensure that any third-party code you are using is secured, so the design itself of the API should be secure. Then there is also the testing, so making sure that there is thorough testing of any APIs. Like end-to-end testing, unit testing, fuzz testing, any type of testing that could possibly be done in the API that will also help ensure that the APIs are secure. One of the biggest things that is arguably the biggest issue is the documentation. Most organizations don't document their APIs well, and that also causes security issues if they are lost. Then you have an open API that is open to the public, but you haven't kept track of it. That could be a huge security issue. Yes, I think the Gartner report has an overall good way of describing all of those issues.

Tim: Thank you very much. I didn't get to the part, when I hear your responses to that, is that reputation is something that we don't talk a lot about with companies and organizations. This is part of their quality, their cybersecurity and aspects of things. I think all the things you identified are all things that organizations need to think about. You don't want to produce bad products that have problems like this. We try to remind people, keep reputation in mind. It is a good reason to do these things. Another area that I would like to talk about then was for the API security, looking at the area of the monitoring and analytics for APIs. I think that is something that I am not sure organizations plan for, think about. Why is that important from a zero-trust perspective, dealing with APIs?

McKinley: APIs, like I said, can access a lot of sensitive information, and a lot of them are public facing, so they should be logged and monitored. You should have a log of all of the requests, all of the inputs that are being put into the APIs, all of the responses, all of that type of stuff is important. That goes into what I was saying before about lack of documentation, because sometimes if the API is undocumented, then it may not be being properly logged or monitored. Oftentimes, like an API gateway or a separate SIEM [Security Information and Event Management] will do the logging and monitoring. But if you don't know that the API exists, then it is not being attached to the logging and monitoring structures like it should be. Having logging and monitoring in place is important for any part of a network, but especially with APIs and especially with zero trust all inputs to an API should be logged and monitored just for visibility purposes.

Tim: Right. Thank you very much. I think that is a very important emphasis to make where organizations are thinking about moving to the zero-trust strategy is considering that aspect. They tend to focus on the things we talked about earlier, so this is another key one. Thank you for talking about that. I think a lot of things you have talked about for API security tend to be more traditional-based approaches for how you secure that. I am seeing a lot more emphasis on machine learning and dealing with API security. Can you tell me what you are seeing in that area, or what are some of your thoughts?

McKinley: Yes. Machine learning is becoming a lot more common when it comes to all types of security, but especially API security. And traditional cybersecurity focuses more on static methods of protection. So things like using attack signatures that are commonly used and similar things like that. Whereas machine learning is more dynamic. It can do things like anomaly detection on the fly that the traditional means haven't been able to use. I think, yes, machine learning should advance the technology in API security in a way that traditional cybersecurity hasn't been able to.

Tim: Right. It is another facet of securing APIs that is definitely something that is growing, and it is going to make a difference, so people need to be aware of that type of effort going on. So thank you. Another thing that we are seeing is in defense acquisitions lately: there is an emphasis on more COTS-based [commercial off-the-shelf] products. They want to get the systems out there to support the warfighter much quicker, at a lower cost, without all the typical delays associated with, say, a major capability acquisition. We are starting to see a lot of commercial vendors. They tend to focus on saying that more of a real time, more active organizations that are combining, teaming together to put their products together. How they do that is through the use of APIs. Some company would say, *I am much better at doing drone technology or being out there on the tactical edge in the field*. Other ones would be focused more on, *How do I take care of my data on a data warehousing enterprise level*. These people are teaming up. APIs is what I am seeing is how that is occurring. Do you have any thoughts on this transition to this type of an acquisition, or what are your thoughts on what people should be worried about?

McKinley: With things like that, with APIs being used in that way with different agents communicating with different things, that is a new method of API-to-API communication or API-to-machine communication, which isn't how APIs were originally used and how historically they haven't been used. It is a new consideration of how to secure APIs when they are communicating with other machines. I think that also goes back to zero trust and why it is so important with all of this is helping ensure that that API communication, even if it is with another API or another machine, it is still secured. I think that is an important consideration with all of these deployments of different API usages and making sure that you are not just using traditional means, which, traditionally APIs weren't considered as a huge security risk. And now they are, and they are being used more than ever, and they are being used for very big projects and with very serious things with very sensitive data. I think securing them is more important than ever.

Tim: Yes. I think the thing, like you mentioned earlier about the logging and monitoring, you need to have more points where you are accumulating that information. You have to have better insight into the architecture to see, *OK, where is where are my APIs? How are they being used*? *What type of data is going across?* I appreciate that. I think that is a very key point that people need to consider as we are starting to see more of these systems. Another area we are seeing is the increased use of AI agents in large language models. It seems like they have a very strong reliance on APIs. Can you tell me a little bit about what is going on here?

McKinley: Yes, I think similar to what you mentioned before, there are large language models, and then there are other AI agents, and they can be communicating with each other. For example, the output of a large language model could be sent to another AI agent for analysis and things like that. I think the same principles of making sure that things are being logged and monitored properly and making sure there is good visibility into the system also applies to any communication between like an LLM and a separate AI agent.

Tim: OK. Thank you, thank you. I think people are getting the sense that APIs are very important when you think about a system and applying a zero-trust strategy. I appreciate you telling me about that. McKinley, what resources are available for our audience members who want to learn more about your work in this area?

McKinley: I have written a blog post and a paper on APIs, and also a podcast. In the future I want to do more research on APIs, specifically when it comes to zero trust and machine learning.

Tim: OK. Thank you. Please know that we will include links to all our resources mentioned today in our transcript. McKinley, I want to thank you for joining us today. We look forward to having you come back and telling us more about your future work. Thank you very much.

McKinley: Thanks, Tim.

Tim: Finally, as a reminder to our audience that our podcasts are available on *SoundCloud*, *TuneIn radio*, *Apple podcasts* and as well as the *SEI's YouTube channel*. If you like what you see and hear today, please give us a thumbs up, and thank you again for joining us.

Thanks for joining us. This episode is available where you download podcasts, including SoundCloud, TuneIn radio, 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 e-mail us at info@sei.cmu.edu. Thank you.