OCTOBER 6, 2021

Eclypses Interviews with Industy Experts: Scott Foote

FEATURING:

Scott Foote, CISO, DPO, Managing Director, Founder at Phenomenati

Can you briefly tell our audience a little bit about your cyber security background?

As a career perspective, I started off as a software engineer and rose through the ranks to be a product executive for a number of companies. I've been a market analyst, a board member, and more recently I'm a CISO, chief information security officer, and a chief privacy officer. The point in bringing up the product background is that it blends well for our clients. In my consulting firm, when we engage, we're not just securing their company, but many of them are product providers. So, I will work with the vice president of engineering or the vice president of product to assess the security of what they bring to market and to completely enumerate, frankly, the risks to that product- to that platform. So, the blend there helps to make sure that whatever technology controls they deploy align well to the business objectives.

In your opinion what is the impact of a weak mobile application security strategy?

CISOs talk a lot about what keeps us up at night, and that is our infrastructure, and the portions of it that are insecure. It's no surprise to anyone that we've seen a massive movement to mobile platforms. Not just phones, but tablets and even IoT devices should be considered within that. The bottom line is that's opened an attack surface for the corporations we defend. For our organizations now, the technology has been progressing, but the dependence of the end users, whether it's a consumer or an employee and the dependence we have on our endpoints, specifically our mobile phones, is just unprecedented.

> At a recent event, I was doing a keynote and we were talking about just how much trust people have in their phones. I challenged the audience - it was on wall street - to just have one person step up, trust me, and hand me their phone. Could have been locked but trust me and just hand me your phone. Not a single person would hand over their phone. The point was well made. We have such an incredible dependency and innate trust on this new platform, that we put a lot at risk in terms of the information that's being managed in that new infrastructure.

Mobile apps are not just convenience anymore. Corporations deploy them for the vast majority of what they do for engagement with their employees. Whether it's just basic collaboration and communication, or it's the core services they offer to their employees. That innate dependency puts us at risk and not just in terms of confidentiality breaches. If the mobile apps we trust spill the data, could be from the endpoint or in the communication, that's the confidentiality breach. There's a lot of issues there, from privacy to corporate espionage, but also the integrity of the endpoint because adversaries understand our dependency. They've watched this migration over decades.

Today, mobile endpoints are attacked more than laptops. This is now the second or third year in a row that most analyst firms will report that the mobile endpoint device is being attacked far more. That's not going away. That's going to continue to be the case because that's our primary point of interface to pretty much any information system. So, we've got both confidentiality challenges and integrity challenges, in terms of what the potential loss is, presented by these mobile devices.

In your opinion what are the top mobile application threats out there today?

Probably the best source for this content is the OWASP, the mobile top 10. There are many issues on that, they do revolve periodically from year to year, but almost always in the top three is insecure communication. The communication between the mobile app, the actual servers that support that, and the services that support that. Specifically insufficient protection of that transport layer. We do have protocols like TLS, which we've introduced and continue to fortify, running over other protocols like IPsec. They do bring significant levels of protection, but that doesn't mean they're always configured properly. Adversaries experiment with that very protocol all the time to look for weaknesses. Whether it's a zero-day or even a well-known weakness, because someone hasn't properly configured it in terms of how the client speaks to the server. Then the response that is always in the top three is that insecure protocol.

The second of those is something that many app developers take for granted when they get started, and that is the protocol where we do a handshake between the client and the server, TLS is one example of this. Often in that handshake, we wind up with a negotiation of what cipher suite are we going to use. I can only encrypt to this cipher suite, so they negotiate down to a common means of encryption, in many cases, even servers.

I see this all the time when I do assessments on vendors. I evaluate their servers to see whether it is capable of being forced to negotiate to a weak cipher protocol. That negotiation happens and it's completely invisible to the end-user. Even the app developer only knows that TLS will eventually get to an agreement on what encryption we're going to use, but attackers understand this, and they force that protocol down to its lowest possible agreement. Often, weak cipher suites can be broken, and too often people don't check the SSL certificate. Many app developers will simply say "oh the server's surrendering an SSL certificate, it must be valid so I'm going to go ahead and use that certificate, and if it looks okay, I'm going to continue to set up the communication with the server," but here's the problem, adversaries know this too. Adversaries can very carefully forge certificates that can stand up a server that is in the middle of the protocol from the mobile device to the server, it's called a man-in-the-middle attack. They very politely negotiate a relationship. Their malware server picks up the outbound communication from the mobile device or app which states "I'm the server you think I am, let's negotiate." Then on the other side of the man-in-the-middle is a negotiation with the server itself. So, now you have a very effective third party hiding in the middle of what is thought to be a confidential communication path. That happens far more often than people acknowledge, especially with early-stage mobile apps where folks haven't really built in the hardening to prevent it.

So, those are the top three that I see regularly in OWASP, and whether we really understand what's going on, we need to fortify our development organizations. As a VP of product, I've done this with my engineers to teach them better programming techniques to avoid these assumptions. Don't allow these types of adversarial manipulation of the communication path to exploit our confidentiality. All of them wind up being fundamental, but it isn't just the confidentiality, the man-in-the-middle can also push malware back to the endpoint and violate the integrity of that endpoint. All of it appearing to be done legitimately based upon what we think is a secure conversation.

If you were talking to a mobile app developer, what would you tell them to do for their mobile security strategy?

I think from the first perspective, don't rely on just one technique. Layer your controls, layer your defenses. Yes, you're going to use TLS, we all do. Yes, we might use lower-level protocols if the network between them supports that. But, if there are additional controls that you could layer on top of that to handle things like confidentiality, further encrypt the data in the stream, further obfuscate the data in the stream. If there's additional integrity checks, allow the protocol between the mobile app and the server to communicate and exchange pieces of information where they can further authenticate each other and, if the response isn't appropriate, now you don't trust that communication path. Build-in these additional layers of controls, because relying on TLS, we've seen adversaries can exploit that dependency, so have due diligence and do care in mind as you're bringing your products to market and recognize that one lock on the door is not always sufficient to keep the adversary from exploiting what is their primary target, your data.

How is 5G going to create new security attack surfaces? Since 5G is going to lead to a greater risk on devices, what do we do about it?

It's an emerging platform. The first challenge there is we're trying to fortify that (5G). It's got great bandwidth and great configurability, but as people bring their technologies and services to market the standards and the protocols are still emerging. So, organizations like 3GPP or the internet engineering task force IETF or ITU are the bodies that continue to refine the standards, but we're still in the early days of those standards. They're built on mature predecessors, but we're still in the early days as the service providers attempt to adopt the standards and adhere to them. It's a moving target that presents a nascent set of new vulnerabilities, just like don't buy the first model car when that car comes out. You don't want to buy the first model because the manufacturer probably has not worked out all the unique details where it hasn't been troubleshot yet, so we are in that early adopter phase of 5G.

Another challenge in the 5G space is this concept of slicing and how the actual networks can be manipulated themselves. Now, we all have the right intent when we set up our network through a slice, we try to protect it from a cryptographic point of view, but the 5G networks have to support layered 4G or legacy protocols on top of that backbone for 4G clients that may be riding on the new 5G network. That means that many of these protocols allow you to negotiate down to what the legacy device may be able to support. So, a man-in-the-middle attacker, someone who's actually in that slice, could potentially manipulate the communication, the handshake between the two endpoints, and force the negotiation down to a weaker legacy protocol. The vulnerabilities there are very well known, and the adversaries continue to attempt to exploit that. So, just because you get onto that 5G backbone, that doesn't mean 5G won't very conveniently negotiate things down to the most convenient level of communication for you. It's invisible to you as either the app provider or the end-user sitting on that mobile device. So, 5G itself is nascent and it has great promise, but all of these issues exist and don't think for a minute that the adversaries aren't setting up test environments to test exactly these vulnerabilities to look for places where they can exploit the dependency. Once the communication exists, it must be secure because it's now over the 5G pipe. Make no assumptions, let's actually build in additional controls.

As you get to know Eclypses a little bit more, what are your thoughts on Eclypses' MTE technology?

I think about this both from a defender and an attacker perspective. At points in my past and my career, I've run red teams whose job it is to exploit weaknesses in a target set of infrastructure. Looking at MTE technology as I've seen it emerge, I've thought about it as a product person between the defender and the attacker. I want to be able to do a couple of things fundamentally. First, is I want to protect my data. The best way to protect the data is if it's never there. Quite bluntly, I won't put my real data in the pipe. If I can create an obfuscation or an additional set of encoding, such that when the data gets exploited it's useless to the attacker. They may get it, but it won't be anything that they can monetize. It won't be anything that they can actually make use of. That additional set of controls for confidentiality is invaluable to me as the product provider because of the due diligence factor. I can demonstrate I'm building additional controls. Even if the exploit happens, whether it's likely or not, I'm minimizing the impact.

The second piece, as the product provider, is I'd really love the ability to know when my components are speaking to each other. That they are in fact who they say they are. Think about zero trust, it's kind of a buzzword in the space, but it's a legitimate approach to communication between remote pieces of software. Never trust. Never assume that once you see yourself on the other side of a client-server connection, it must be who it was intended to be. Let's add an additional layer of authentication that will expose a lack of integrity. So, if we get into the MTE environment between a mobile app and its respective server and the mobile app is not providing the right responses to the server and vice versa, that exposes the fact that maybe one of them has been compromised. The integrity of one of these endpoints has been compromised. Now, I can, as the product provider, build in an appropriate set of controls to back out of that communication, to terminate the communication, but also to send alerts to whatever the appropriate reporting infrastructure is. To say, "I don't know that it's been compromised, but it's not behaving as expected." That gives me an additional set of controls, not just for confidentiality, but to identify breaches of integrity that can be handled appropriately. Again, drive down risk, minimize the risk to my user.

Who in your opinion should be paying attention to MTE technology?

I think about problems like this from the buy-side and the sell-side. Let's start with the sell-side. Having been a product person, developers must think about the liability that their mobile app brings to their corporation. As the provider of the mobile app, they could be exposing personal data. Now you wind up with GDPR and other types of privacy laws. Think about the potential downstream impact of your exposing corporate data. The mobile app developers should be thinking about this as an additional control that is easy to adopt and it's non-intrusive in the environment in terms of the deployment. This gives me an additional set of control. Nothing guarantees a hundred percent, but this certainly raises the bar for the adversary. As a vendor, I have stronger confidence in what I'm bringing to market and the fact it isn't introducing more risk to my clients.

Now on the buy-side of course, I put on my CISO hat or my chief privacy officer hat. I do vendor risk management every day. Every day we evaluate new vendors, and we evaluate the potential risk those vendors bring into our corporate environments. We need to look at risk scenarios like this: "what if we have a data spill?" We're bringing in a new HR app. They have a mobile endpoint for the app. Well, what if we have a data spill between that mobile endpoint and the corporate server? What's the liability to us as stewards of that data? When we look at these vendors, we should be asking them for more than the basics. We should be looking for where they make implicit dependencies on the underlying infrastructure, and we should be asking them to raise their game. Show me an additional set of controls. Show me that you're going beyond the basics to ensure the integrity of my information when it's running over your application. That gives the vendor a leg up as I start to evaluate vendors side by side. The criteria is heavily weighted in terms of which of these vendors has demonstrated a better program in terms of being a solid data steward of my corporate information. So, both buy-side and sell-side should have a significant interest in technologies, in apps that depend upon additional controls like what MTE brings to market.