

Web Application Security Resources – an Interview with Joe White – Imperva Customer and Application Security Practitioner

Listen to Podcast [here](#).

Brian Contos: Welcome to the Imperva Security podcast. I'm Brian Contos, chief security strategist for Imperva. If you would like to learn more about this subject and Imperva, visit Imperva.com, check out our [blog](#), follow us on [Twitter](#), or send us an e-mail at blog@imperva.com.

Joining me again today is Joe White, Security Practitioner and Imperva customer. Welcome back to the show, Joe.

Joe White: Thank you Brian.

Brian Contos: So, for those of you who caught our last podcast with Joe White, Joe is an Imperva customer that works for a SaaS company. Joe, what are we going to talk about today?

Joe White: I was hoping to take some time to maybe help folks who aren't as confident in their web application security abilities or awareness. Often times it seems that there are a lot of people talking about what is going on, but we sometimes forget what it was like to be new to the whole area. So if we could maybe step back, I think there's audience that wants to know a bit more about how they can become more security aware and how they could go about learning more about their application and kind of be more, work towards being expert in the web application security field.

Brian Contos: Yeah, I think it is going to be really interesting to hear your experiences about how you sort of went from a traditional network system security type person to somebody who had to start taking a look at applications and learning more about application security. Probably when you made the change, there might not have been as many resources out there as there are today. Currently now there's a lot of great info that people can latch onto. And if they know the right place to look, they can get up to speed much more quickly.

Joe White: That's exactly the point, because I do come from, I come to web application security from more of a traditional operations and systems security side doing traditional ping tests a long time ago before web applications focused on assessments were really as popular as they are now. So, I mean, to a certain extent, I am a testament to the fact that you can actually learn new things.

Brian Contos: Joe, before we even get into the tools and resources available, why are we even talking about this? Why is it important for operational security people to understand app security and database security, these things that were typically, you know, in some other group? You know, they're programmers, they're different individuals. They're not security folks. Why is it important for security guys to even care about this?

Joe White: The reality is as information security professionals, we all have to adapt and learn to adapt to the threats as they change. And with web applications being so pervasive, and more and more applications making their way to the web, it only makes sense that as a security professional you become more aware of your existing security and is it adequate to address web applications or not? There's the notion of a traditional perimeter security, which kind of doesn't really work anymore. So, maybe some of the discussions or decisions that you have made in the past at your organization just really aren't the best to be securing a web application.

Hopefully we can take some steps toward making you more aware of that, and offer you the tools so that you can start learning more about that on your own.

Brian Contos: So, let's just go ahead and jump into it. You know, there's some folks out there, some listeners, they're saying how is it that I get involved in web application security side? Where should I start? What are some of the resources? What kind of tools and resources are out there? Maybe you can share some of your experiences, too; what you used, what you liked, what you didn't like, et cetera.

Joe White: The key objective here, I think, for anyone who is either new in web application security or interested in getting into it, or maybe you just kind of want, you're looking at maybe broadening what you know. The key point here is that you ultimately want to make yourself basically the internal subject matter expert on both your applications and the whole area of web application security. So, ultimately what that means is you really have to know your application almost inside and out. You have to look at the traffic, look at the web application traffic and be able to figure out what is good traffic and what is bad traffic; where do you get into the application, the ingress and egress points. So, the whole point here ultimately is that you have to know your application very, very well because there are some clever guys and clever tools that are being used against your application so you need to know what those tools are and how you can see what they are seeing on your application.

The first place to start I think would be basic HTTP proxies. When I got started in web application security, I don't know, four, five or six years ago; I'd have to think about when it was exactly. I started with Paros Proxy. Most people that I know, maybe because they started around the same time tend to start with Paros Proxy.

It's just basically a Java app and it listens to all of the web traffic as it's leaving your browser and allows you to manipulate it. This would be one of the similar tools that maybe you'll see internally or from inquisitive users or from basically people trying to attack your application.

So, it is one of the tools that everyone has in their tool chest. When I started out, I started with Paros. And then I moved on to Burp Suite which is another HTTP proxy suite by Port Swigger. A lot of folks I know tend to start with Paros and move on to Burp Suite. I don't know why.

I guess the dilemma here is that you want to start with, do you want to start with where people end up, or do you want to kind of start with Paros and then kind of make the, it's up to you. But, you need to ultimately know your proxy and know it inside and out, knowing what it offers and how you can actually use it, because this is going to be the same types of tools as well that someone who is interested in attacking your application are going to use as well.

Brian Contos: What kind of things might I be able to figure out by using a proxy?

Joe White: It's really eye opening if you haven't use done before. Basically, what you do is you're having your browser yourself, before the traffic leaves your box, basically you're stepping through another application, which is the proxy, which allows you either to manipulate the data before it goes into your SSL tunnel, before it goes into the actual, before it leaves your laptop or your desktop to go to the web application. You can manipulate the parameters; you can change things around that you wouldn't ordinarily expect that you can do unless you've actually seen the power of an HTTP proxy.

Brian Contos: So, not only can you see the flow of data across layer seven like you might expect to see with traditional TCP Dump or Snoop or Ethereal, or something like that.

Joe White: Exactly.

Brian Contos: But you can also manipulate it as it's going through and make modifications and GREP certain bits of data.

Joe White: Right. And this is not something, I don't think, that maybe a web application developer would have expected someone to be able to do, but it's certainly available to any user that has one of these tools available. Something as simple as changing some characters on your session identifier might allow you to get someone else's session, or changing a value of a particular parameter may actually start giving you the data that's associated with a different user. You need to be able to know what can and can't be done to your application using these tools.

So, I mentioned Paros, I mentioned Burp. Another tool, Fiddler, which is excellent if you like to use IE, lets you find vulnerabilities in your web application basically while you're browsing the web app. So, if you have Fiddler on and you are looking at your web app and just kind of driving through, then the Watcher plug in with Fiddler would alert you to things that it thought you should be concerned about.

Proxy Strike is a similar proxy that is a free standing proxy that will look for vulnerabilities basically in the web application while you're just driving through the web application.

Another proxy which you'll see commonly is from OWASP called WebScarab. I don't see as many folks using WebScarab, but it looks incredibly powerful. You just have to decide which one you that you want to be very, very good at.

Brian Contos: So, it's two fold, right? Find a tool that you're comfortable with for your initial analysis, when you get going, but as you start requiring more advanced features maybe give all of them a test run and see which one of them fits your specific needs best.

Joe White: Yeah, I don't want to influence anyone to start or to do something outside of what they are going to do on their own. I can just tell you from my experience, certain ones are more popular and certain ones aren't, and I'm not really sure why that is. But, definitely try them all out because you may find that one of them you like better than the others.

Brian Contos: So proxy is definitely an important piece to have in your toolkit. What else do we want to add?

Joe White: As you are looking at the web application traffic itself, you'll need to have some reference to the types of manipulation or exploits that someone may, or your application may be vulnerable to. And in that, I think that de facto standard is R-Snake's Cross Site Scripting Cheat Sheet. I know he does a very good job of keeping it up to date, so there's many different types of Cross Site scripting injection streams that could be used

for your application. Some of them are specific to a particular browser. But it is an excellent resource to kind of get a feel for the possibilities.

Brian Contos: In addition to R-Snake's Cross Site Scripting Cheat Sheet, there is also another cheat sheet that is pretty interesting on OWASP if you can find it. It is the Cross Site Scripting Prevention Cheat Sheet. So, kind of back to the notion of breakers versus builders, even though these are both white hat focused. The prevention cheat sheet is very interesting because it sort of walks you through in detail how you can prevent cross site scripting and by mixing that with what you see from R-Snake's Cross Site Scripting Cheat Sheet, it can be very compelling. You can put the fixes in place, and you can test them to make sure that they were implemented properly. It's a great way to go back and forth between what the bad guys might do and what the good guys are suggesting you try.

Joe White: That's actually a good point because I can't even tell you how utterly spectacular the OWASP Cross Site Scripting Prevention Cheat Sheet actually is. It's the most amazing reference. If I had had it three or four years ago, it would have been phenomenal. What that is going to help you do is as you become more familiar with the types of attacks or types of vulnerabilities that your application is susceptible to, you are going to have to communicate those changes back to a developing environment, to the developers. And the way you are going to do that is by using the prevention cheat sheet because it is going to tell and it is going to be able to communicate the changes that need to be made to the application to make it to where it is not vulnerable.

Another point here which is useful is as you start looking into how to prevent the vulnerabilities, there's two notions. You can address input, or you can address on the output as the data is being rendered back to the browser. Input used to be thought as the best way to go, and I think people are now leaning more towards escaping on the output, making sure that whatever is rendered back to the browser is no longer in an executable context, but is just basically in a text based context.

You'll need to figure out what works best for your application, but there are certain cases where if you input validate or you input filter, it may actually break the functionality of the application itself. So, depending on your application, you'll need to work out what works best for you.

Brian Contos: Yeah, we're certainly seeing people who are doing both input and output validation, both on the application and database side. To Joe's point, sometimes that breaks things, right?

Joe White: Oh, definitely.

Brian Contos: There's the question of having the ultimate security, and there's also the question of making sure stuff still works. Once those of you that are new to the space start digging into it, you'll hear terms like canonicalization and encoding and things like that. I point you to a presentation called Don't Write Your Own Security Code, the Enterprise Security API Project. This was written by Jeff Williams of Aspect Security. He actually has a slide in there that actually talks about how many ways via encoding you can hide a less than sign through percent encoding, HTML Entity Encoding, Java script escaping, CSS escaping, ASCII UTF. There's just numerous ways of doing it. It's pretty amazing.

Also, if you take the word script, there's over 1.6 quadrillion ways to go ahead and encode that. So, when you are talking about how to hide things and prevent things and canonicalization and input and output validation, again it's not going to be necessary analogous to things like the network security world that said, you want to block telnet, you

know, block port 23 and it's done. Things get a little bit more sticky, and a little bit more involved when you're looking at the app side.

Joe White: If you're looking at your web application traffic, you are going to see things that are just going to be mind blowing. I mean, I can't even explain to you how your perspective and your perception is going to change. But I can guarantee you that you are going to have, I think what we talked about prior, you're going to have so many holy **** moments. It's amazing. You're going to see things going on that you didn't expect to see, but you just need to be aware of those. You are going to be looked at within your team to offer them guidance, so ultimately your goal is to be pretty much the subject matter expert in your organization for the web application. So, just knowing what the traffic should look like coming through, and the only way you can start it is to start looking at the traffic.

Brian Contos: Joe, we have just about another minute or so left here. Your perfect web application security guide toolkit. Some proxies, some great resources, what other types of things might they want to arm themselves with?

Joe White: Well, at some point you're going to have to start, I know I talked about this last time, in the Web Application Security Roadmap presentation I did a while back, I kind of talk about the approach that you need to go for addressing it, you'll need to find the vulnerabilities. And you'll need to do that with a mix of your own abilities to find them. If you use some of the tools that we're showing you, you'll be able to find these yourself. You may even want to look at getting some scanning technology. The point to remember about scanning is even though you do have a scanner that looks for web application vulnerabilities, everything is so immature in that web application security space, you're pretty much going to have to have a hybrid approach where you have a manual component and an automatic component. I think that there are few in the security world that would suggest that that is not the case.

So, as you start finding them, then you start trying to address them, and you'll use the OWASP XSS Prevention Cheat Sheet because cross site scripting is basically a stepping stone to other forms of attack so once you start addressing all of the cross site scripting vulnerabilities, that does help you.

And then towards the end you're going to just want to make sure that you are looking at the traffic through some type of web app firewall or some other mechanism that gives you more visibility into the traffic so once you start getting a baseline of expectations, you start looking for deviations from that.

Brian Contos: Well, Joe, I think that was a great primer for people that are interested in getting started in this space. We'll have you out again.

Joe White: Thank you.

Brian Contos: If you would like to learn more about this subject, and Imperva, please visit Imperva.com. For questions or comments about this podcast, please send an email to blog@imperva.com and follow us on Twitter for the latest Imperva news.



North America Headquarters
3400 Bridge Parkway
Suite 101
Redwood Shores, CA 94065
Tel: +1-650-345-9000
Fax: +1-650-345-9004

International Headquarters
125 Menachem Begin Street
Tel Aviv 67010
Israel
Tel: +972-3-684-0100
Fax: +972-3-684-0200