Hyatt investigates malware found on payment processors

Trend Micro: Lessons learned from 2015 cyber attacks

Israeli missile defense chief dismissed for breach of security protocol

December 2015 Issue

## Editorial

Let's just call it "The era of IT security"

Subscribe


Archive

Peter Stephenson, technology editor, SC Magazine

December 16, 2015

# Hunting and STIX

Share this article:


- facebook
- twitter
- linkedin

- google
- Comments
- Email
- Print

Today I'm starting a three parter that will introduce you to STIX, a neat way of documenting and understanding your threat hunting targets.

Peter Stephenson, technology editor, SC Magazine

STIX is "Structured Threat Information eXpression." It really is a structured language for characterizing cyber threat intelligence built on XML. I am not going to take a deep dive into XML. Beyond some minor explanation I am going to assume that you either understand XML or know how to learn more about it. In that regard an excellent high-level introduction is in XML Programming success in a day by Sam Key. I would hardly call this the definitive guide to XML – I don't really think that you need to go much beyond this book, but that's up to you – but it has more than enough solid information to get you to the point of understanding STIX.

There are a couple of things that you should know about XML, however, to provide context for what comes next. XML is a markup language (XML means eXtensible Markup Language). It uses tags that, at first blush, look a bit like HTML tags. That is because both are markup languages and both have a similar genesis. XML is hardware and software independent and the big deal about it is that you can define the vocabulary any way you want. That means that you create your own tags unlike HTML which has the tags pre-created. In this regard, XML is a lot more like an actual programming language than HTML.

So, if you want to create a set of project-specific tags, how will the next person to use your application – including XML, of course – know what you had in mind? The answer is that you create a schema that defines a particular format that you want to use and then you declare your tags in the schema or the actual code for your particular project. STIX has a schema and we will talk a bit about that shortly.

Before we go too much further, let's complete the picture for you. For a lot of detail about STIX, go to the STIX project at https://stixproject.github.io/ . There is a lot of documentation and some tools that you will find necessary. In order to exchange STIX files it is helpful to have an easy way and that way is TAXII – Trusted Automated eXchange of Indicator Information. You can get what you need to build your own TAXII server at https://taxiiproject.github.io/ along with a lot of information. However, my advice is to set up a ready-to-go TAXII server. You can get that at https://soltra.com/ . That is how I set up my server and it is the one we will use for these postings. The tool is called Soltra Edge and it is free.

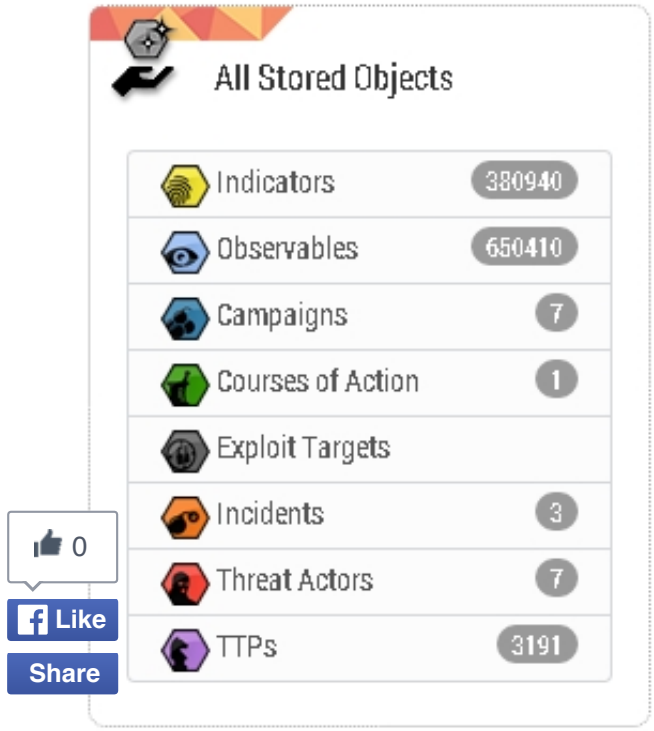
The next thing you need is to get some STIX indicators from a public TAXII server. Sadly, there are very few of those but an excellent one is Hail A Taxii at http://hailataxii.com/ . We'll dig through that in these postings and use some of the STIX files you can find there. I recommend that you analyze existing STIX files from hailataxii.com before you start building your own. We will end this short series by showing you how to turn things that you find in your systems – malware, attack attempts, phishing attempts, etc. – into useful STIX files.

Before we go too deeply into this, though, let's understand the answer to the "What's in it for me?" question. In short, why bother? The answer is pretty simple: this is how you collect, categorize and document the threat intelligence you discover as you hunt. But, as the infomercials say, "Wait! There's more!" This is how you exchange threat intelligence with other hunters and more and more devices such as threat intelligence systems, firewalls, IDS/IPSs, etc. are beginning to ingest STIX files as part of their intelligence-driven threat protection. So understanding this process is more than useful – it will, over the next year or so, become

necessary. Besides, it's fun. I completely enjoy digging into a threat picture and extracting campaigns, incidents, malware, observables, and so on.

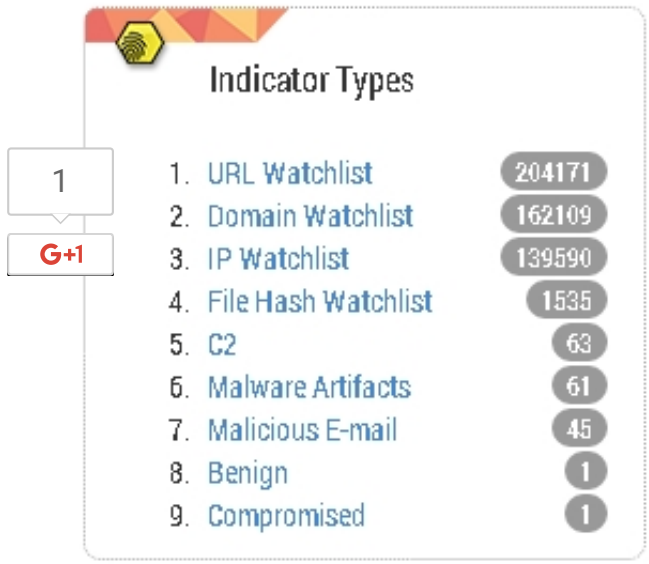
Before we leave for this posting, let's take a quick look at what to expect in a STIX characterization. We'll get into the weeds next time. For our examples and screen shots I'll use my Soltra Edge server. In STIX we tend to think, in general, of things as indicators. In fact, indicators are just one of the eight kinds of objects with which we are concerned. Soltra Edge makes it easy for us to work with these object by providing a control panel for them. See Figure 1.

Figure 1 - Types of STIX Objects




Additionally, we can break indicators down further. Think of this as a sort of taxonomy of threat indicators. See Figure 2.

Figure 2 - Indicator Types




That's enough for now. We'll take this up in more detail next time. Meanwhile, install Soltra Edge, do some reading about STIX and TAXII and, maybe, even connect to hailataxii.com and download some STIX files to study. Next time we will take a sample campaign and break it down using STIX.

So… until next time….

--Dr.S

*If you use Flipboard, you can find my pages at http://tinyurl.com/FlipThreats. Here I flip the interesting threat-related stories of the day – nothing particularly technical, but interesting stories none-the-less.*