February 27, 2019 SCAP Community Telecon

SCAP v2 Integrating Other Efforts

The following is a collection of the key points brought forth in the two-hour telecon on February 27, 2019. The questions used to drive the conversation are highlighted in blue.

Danny Haynes stated that the purpose of the telecon was to talk about other efforts going on that may be able to leverage SCAP data to inform decision making and how SCAP architecture might be able to leverage data from other sources to improve how scans might be triggered.

# OpenC2 Presentation (Joe Brule – NSA)

Joe Brule, NSA, gave the opening presentation. He stated that we’re spending 1/3 of our global defense budget being victims of cybercrime. Human-controlled point defenses just aren’t going to work. Integration in the absence of standards is a mess. OpenC2 is part of a bigger picture, part of a suite of OASIS standards. The high-level language and basic idea are there, but there is the need to get the actuator profile out to get the architecture in place.

Jessica Fitzgerald-McKay, NSA, said that OpenC2, in trying to develop automated command and control languages, is relying on other standardization efforts to support them and is waiting for SCAP to provide situational awareness. She asked for more detail on what NSA is hoping OpenC2 will be able to leverage from an SCAP standard.

Joe Brule felt that there was focus on the “verb”, or what they’re trying to do and what they’re trying to protect themselves from. He doesn’t feel there is a solid handle on the “nouns”. There is a need for a good grip on what exists on the network today and how one is going to interface it and how does one define what the scope of an actuator profile is. He stated if there is some decision analytic that says “deny this range of IPs, look for this file type, and move it to a honeynet, do scanner analysis, etc.”, what’s missing is knowing what is going on the current network. NSA thinks SCAP will augment what they’re trying to do.

Joe Brule stated that he knows products do a lot of different things, but he really wants to interface with products in a standard manner. He asked for insight from the SCAP community.

## Do you have ideas on what that type of information might be? Any certain things you’re looking for? Any use cases you care about? (Danny Haynes)

Joe Brule commented that they did get a good idea on what information is needed, the most concise way to do it, and what shiny options that some companies have. From the perspective of use cases, the only cases OpenC2 focuses on are the DoD specific ones – the unattended sensors, unmanned platforms, satellites, RF domains/links, etc.

## Can you walk through an example of how you envision this working – OpenC2 says do a scan that does what? And what does it provide back?

Joe Brule said that from an OpenC2 point of view, he assumes decision is already made and knows the action to take. He is looking at it from the perspective of “we need to do these three steps, tell the routers to look at a block, check for malware, tell the tool to isolate nodes”. Within OpenC2, what information does your widget need to execute that desired command.

A participant asked if it made sense to say that OpenC2 says I need SCAP to do something for me and it expects Y and Z back. Joe Brule said yes but to be clear, they’re on the “act” portion of the loop. The scope may be expanded to include some of the decision analytics. The focus is on the southbound command going from mission manager to that end device.

Jessica Fitzgerald-McKay didn’t think they were talking the same language and put it another way. She asked what would you use SCAP results for? Task actuators take action, but who are those actuators? We need SCAP to give an inventory of actuators on the network.

Joe Brule responded with yes, there is a need to know the status of the actuator and what is currently loaded on it, so the capability is known. The orchestrator will send a query to SCAP looking for the location of IPs on the firewall, which version of OpenC2 is supported and which actuator profiles are loaded. In terms of sending query to perimeter device, a command is sent but the external analytic does it. In a cloud environment, better distribution is needed.

Gary Gapinski asked if a conversation between the orchestrator and actuator has to have a minimum enough to be able to allow communication to take place. He presumed OpenC2 mediates that. What’s the minimum for an actuator to present a novel type of action to be taken?

Joe Brule said that OpenC2 needs both devices communicating the same language. The focus is on an atomic action – one action. A complete course of action is not being addressed. Business process management languages out there can probably be repurposed to accommodate the courses of action. If one has a shiny new box that doesn’t fit OpenC2, the language specification spells out how to import a custom actuator profile. AT&T has a demo where an actuator profile is being work on but AT&Ts profile goes one step further. The actuator profile has to be decoupled from language specification. If a company builds routers, they don’t need to concern themselves with other things.

## What are the advantages? What is the role in an ecosystem with BPEL? (David Solin)

Joe Brule said that, to the extent of his understanding of BPEL, BPEL can be used to create courses of action using atomic OpenC2 commands. He needed this investigated further.

David Solin said it seems BPEL is designed to be a catch-all. If something does “this” then I should do “that”, as a language for orchestration framework. He asked what is the motivation for this? It seems there’s limited security vocabulary with less flexibility.

Joe Brule said OpenC2 is limited to commands as it’s built at the orchestrator and consumed at the actuator. It’s not as trivial as one would think. There’s no mechanism to do conditional logic. They don’t build the “do this, do that” because they want the orchestrator to send commands to infer the state from the actuator.

David Solin thinks there are disadvantages, but they could help define the space that is being addressed. Joe said to not assume they’re addressing the analysis piece because they’re not.

# Collection and Evaluation of Content (Kathleen Moriarty – Dell EMC)

Kathleen Moriarty briefly discussed the challenges associated with the current state of endpoint assessment. Specifically, enterprises leveraging proprietary products that may or may not leverage standardized data models and protocols and how the lack of standardization makes integration of data from different sources difficult. All of which overburdens IT and security staff - negatively affecting the security of the enterprise. Next, Kathleen described how the use of standardized data models, such as those used in the network management space, can increase an enterprise’s ability to manage configuration information across products and vendors. Furthermore, Kathleen described how standardized data models have improved the ability of enterprises to share threat intelligence data and mitigate cyber attacks. Lastly, Kathleen gave a quick overview of using IETF NEA to support endpoint data collection as well as other efforts to collect information from different endpoint types and store it in a centralized server where it could be used by vendor products. No questions were asked after the presentation.

# Separating Collection and Evaluation (Danny Haynes – MITRE)

The closing presentation focused on separating collection and evaluation. Danny Haynes talked about how the current model ties it together but doesn’t make the distinction clear. The idea behind the concept of SCAP v2 architecture is that there is a set of endpoints with PCS that will query or have endpoints self-report.

## Are there other types of information that are relevant that might be useful to determine applicability?

Several made suggestions of the need to check arbitrary endpoint information for applicability (e.g., software configuration and hardware types, in addition to software type/version) because enterprises provision endpoints differently and use cases may require a wide variety of information. Thus, having the ability to add to the applicability attributes over time is likely necessary. The applicability check could be in the form of SWID tags, CPE, an OVAL check, etc. Several agreed that the current “applicability\_check” attribute, in OVAL, isn’t specified in a useful way.

Danny Haynes said identifying and profiling an endpoint would be another thing to consider if the community cared about that.

Jessica Fitzgerald-McKay mentioned device component information and software and configuration information. She feels there are lots of way to express software configuration information but asked what else do we want to be able to express?

Danny Haynes indicated it sounds like whatever is supported or leveraged as data in the next version of SCAP, there’s a need to support a way of analyzing data with checks and configuration.

Leland Steinke (DISA), mentioned that one thing that would be useful would be if we could take output of OCIL questionnaire and populate the OVAL check through an external variable type mechanism – IP firewall, netblock router, bastion hosts, or someplace else in the network topology. Allow it to be automated and learn what is going on in a local system. He said DISA benchmarks expect a specific language syntax so how do we feed that in?

David Ries said if software inventory meets criteria, it’s not likely to change often. If the software inventory data is cached, it can be expressed efficiently. There needs to be a mechanism that notes that content is comprised of something that meets those characteristics (e.g., role, architecture, environment, etc.). Implementations should be enabled to use event driven figures as a separate collection of evaluation. It would be good if users could identify data they care about and tag when there is an update.

Danny Haynes said to simplify we want to have a way to indicate in content what we care about is meaningful to an author, won’t change frequently, and collected easily. We don’t want to express what those pieces of information are, but we want a way for an organization to specify it for themselves. It’s important to determine applicability.

## Do people have ideas on what specifications need to be updated to support this capability? Are there other specifications that need to change?

David mentioned things like metadata lives outside of the specifications right now. There is nothing telling us what we’re supposed to be targeting. Danny Haynes agreed and thought it would be a good idea to look into this more at the workshop.

## How do we expose previously collected data? What would be the requirement for tools to trust previously collected data?

Danny Haynes thinks that CMDB helps but there are other challenges. There are trust issues with other tools collecting data. Tools may have agreements with other tools they’re sharing data with.

David (Unsure of last name) doesn’t think it’s a provenance type of issue but a supportability issue. Who takes ownership if a customer is giving you data collected from other vendors? At the end of the day, there’s not much that can be done about it. The current standardized formats seem to be as good as it gets. This will be a challenge.

## How do we tie pre-collected data to evaluator needs to data, especially when separate? What type of identifiers? Are there any pieces of data that stand out that would be useful?

Hardware component identities were identified as possibly being pre-collected but there’s concern over them not being changed frequently enough. When the community was asked if there are interface and protocol considerations to think about as we look for solutions of those interfaces that would be affected, there was no response.

## How and where should issues such as data freshness be handled? Are there other qualities that evaluators need to consider beyond data freshness?

David Solin suggested that evaluators need to consider if the data was collected by a privileged account.

David Ries gave an example of a simple OVAL Definition, where you may determine whether an application has a vulnerability solely based on its version. Unfortunately, while this example is simple and straight forward, David explained that there are other checks where that expression becomes complicated when considering different variations of the software across platforms as well as if it requires configuration information. For example, a control might specify that no home folder of a user that belongs to group ABC has properties XYZ. How do you pre-collect the data needed for this check divorced from a fair amount of logic? David’s concern is if we don’t consider these more complex checks, we’re only capable of simple data collection and may not able to get more data.

Jessica Fitzgerald-McKay’s concern is that there are some things we just won’t be able to do, like event driven collection. Can we quantify how often an organization struggles with that?