December 4, 2018 Community Telecon

Architecture and 10/24 Reprise

Dave Waltermire started with raising issues from SCAP v1 that need to be addressed:

1. There is a need for a mix of "standards activities" and "community engagement". SCAP v1 ended up shifting almost exclusively to standards activities and the effort suffered. We need both. This involves work in standards bodies, but also work on "creating specifications".
2. Standards organizations might help with generalized solutions, but are not ideal for maintaining specific solutions. He cited SWID tags as a good target for standards organizations, while OVAL worked better with a small group. There is a need to better understand what the best fit is.
3. There is a need to support diverse endpoint types. SCAP v1 focused on traditional endpoints. There was some support for other types (e.g., network devices) but those were second-class citizens. We need to bring other endpoint types (network devices, mobile, IoT) in as equal members.
4. Posture needs to be collected in a relevant timeframe. Today, the timeframe is days. We need to bring the timeframe to minutes or seconds.
5. Today's SCAP tools tend to be data islands and don't expose their data to the enterprise. We need greater interoperability and orchestration
6. Content has been hard to produce and has required significant expertise. The content is also costly to test. Also, the demand for SCAP content is much greater than the ability to support that demand. This needs to be solved.
7. CPE is not scalable for software inventory and patch management. Today most CPEs are created by vulnerability analysts to document their vulnerability reports. Also, CPEs don't support patches, so cannot be used to identify when a vulnerability has been remediated. Also, with an increased reliance on open source, there is an increased need to understand software dependencies. Vulnerabilities in a module could be inherited by all software that uses that module. We need a federated model to support software management.

Dave clarified that the whitepaper articulates a vision, rather than a final design. We need community input to move forward. Silence, even happy silence, won't work – we need to hear opinions, even if just to say things work well.

Dave presented the core use cases. He noted that we are open to adding more use cases.

Dave explained the iterative approach to SCAP development. There will first be a core to establish an architecture and standardized interfaces between components. Then, to deal with the use cases, extensions will be developed. The plan is to start with software inventory and vulnerability management, both because of their traditional role in SCAP and because they are foundational to other use cases. He noted that this does not preclude immediate work on other use cases if people are willing to step up and do so.

Vijay Sarvepalli asked how this addressed acquisition issues, such as supplier selection. Dave responded that this was something to discuss. He noted that SCAP might have a role in addressing supply chain issues.

Dave polled the community. 4-5 participants identified themselves as vendors of a security automation product. 8-11 identified themselves as end user organizations. Some parties identified themselves as belonging to both groups.

# Question: Which use cases are most challenging

* David Kennel – Configuration assessment
* Jessica Fitzgerald McKay – Software and hardware management, both because they are challenging and because they are foundational to other use cases.
* Gary Gapinsky – Sufficiently accurate software inventory
* Vijay Sarvepalli – Vulnerability management across multi-vendor/multi-version scenarios. Specifically, being able to tell which products are impacted by a given vulnerability. Also, support for enterprises in tool selection – currently there is no visibility into what software dependencies a given product has, and those dependencies might have vulnerabilities.
* Jarret Lu– Timeliness of endpoint posture information
* Adam Montvale - Software and hardware management. Also, creation of standardized content (as opposed to custom content).

# Question: Vendors – what is most challenging for your customers

* John Field (Pivotal) – The cycle time of how frequently people want to replace software. We are seeing that shortening. We need faster inventory checks to support this.
* Adam Montvale – Agree with John. Also, we are trying to accelerate how quickly content reaches end users. We want a "zero-day release".
* Bill Munyan (CIS) – There is a challenge (particularly in OVAL) of getting new schemas and then corresponding tooling developed to the point that it is usable. This can take a long time.
* Gary Gapinsky – There appears to be a long-standing resignation that an OS cannot attest to its own posture. It would be good if standards could be adopted by device manufacturers rather than relying on third party agents. I'm not claiming that every product needs to adopt the same, standardized API, but if products had some API to expose necessary information that would be great.

# Question: Are there other use cases to standardize?

* John Field – I'll have to think about that. I'm sure there are cases where, if we could shorten the cycle time, there would be security benefits. People are deploying new software stacks in hours. They often don't have the ability to see if it is correctly configured or if there are unintended consequences. Better visibility could automate some of this. It is all about getting rapid feedback to operators.
* Gary Gapinsky – Reduction of false positives is important – there is a lot of that in our current software products.
* Vijay Sarvepalli - Possible use case for vendors who are bundling other software into a product and for CERT incident response teams.
* Jessica Fitzgerald McKay – Supply chain could build on hardware/software use cases to help validate that the software/hardware arrived via the proper path.
* Dave Waltermire – Yes, that could be very useful by DevOps teams by helping parties using third party dependencies to better understand what vulnerabilities might be present in those dependencies.

Jessica presented on the SCAP conceptual architecture. She showed the architecture as having two key frameworks: information collection (Endpoints, PCS, CMDB, and SCAP Content Repository) and posture evaluation (PES, CMDB, and SCAP Content Repository). She then provided an overview of the interfaces.

John Field asked if it would be reasonable for endpoint information to be known in a management server responsible for provisioning an endpoint. That provisioning agent would know the initial deployment of potentially many devices, allowing a single point for this information rather than needing to individually poll many identical devices. Jessica said that she could see this, but also saw reasons to allow individual assessment.

John Field paraphrased by saying that, as long as an implementation complied with the interface requirements (and assuming that there was some authentication to ensure everyone was talking with the party they through they were) that this would be acceptable. Jessica agreed.

Joe Wolfkiel asked if these interfaces were literal. He noted that he had scenarios where agents moved around and a collection server would not necessarily know the IP or MAC addresses of those devices in order to initiate a connection. Jessica clarified that there is no mandate that the PCS initiate communications and that SCAP would need to be able to support cases where endpoints initiate the communication.

Vijay Sarvepalli asked if an endpoint could simply respond with "no change from previous assessment". Jessica said that this could be done in a query, and that the subscribe-post exchange in I1 is expected to only trigger if/when there is a change on the endpoint.

Vijay Sarvepalli asked if the PCS was tracking endpoint history. Jessica suggested that understanding how much state the PCS needed to retain was a topic for conversation. She noted its role is to manage collection, but potentially there might be a reason to have it retain some state.

Gary Gapinsky noted that there are already products and standards that support many of the interactions identified in SCAP. He asked if SCAP's interfaces were a response to flaws in those products/standards. Jessica clarified that anything that met the SCAP requirements could be part of the solution and that we are trying to minimize the creation of completely new elements. Gary asked if the conversation between the endpoint and PCS would be constrained to an SCAP standard. Jessica clarified that the I1 is intended to represent a minimally viable set of requirements and that any suggestions for inclusion in the ecosystem will be considered. Gary mentioned RedHat Satellite as a possible implementation of I1, but also noted that this solution was proprietary and didn't necessarily fit with other elements of the architecture. Jessica encouraged such suggestions to be sent to the mailing list.

Vijay Sarvepalli asked if some of the roles in the SCAP architecture could be federated. Jessica and David both noted that many of the roles would probably need to be federated, at least in larger organizations, and that having multiple devices supporting a single role (e.g., multiple CMDB instances) could be beneficial for scaling, resiliency, and data organization reasons.

# Question: What reactions do you have to the proposed architecture?

* David Kennel – The biggest weakness seems to be hardware detection. The architecture seems to rely heavily on agents on the endpoints. Jessica agreed that agents are used in some solutions, although SCAP doesn't require this. She noted that some solutions purport to be agentless and those should be discussed. David Kennel clarified that some device types (e.g., smart refrigerators) might not be capable of hosting a third party agent. Jessica agreed and noted that SCAP's broadened support for endpoint types would likely need to address this. Dave agreed and added that we are looking for experts to help contribute to these discussions.
* Adam Montvale - How do SCAP and SACM relate? Jessica noted that our proposed solutions for some interfaces include SACM standards. She clarified that we want SCAP to be able to adopt anything that meets requirements. Jessca noted that we currently focus on SACM's endpoint data collection and don't do anything with SACM's data orchestration work. She asked if this is something that should be considered. Adam felt that orchestration of some kind would likely be necessary in the architecture. Jessica felt that it might be necessary for bigger enterprises, but that it might just be an unnecessary burden on smaller enterprises and, as such, shouldn't be required. David noted that orchestration can mean management of instructions and it can mean data brokering. He noted that both are probably part of the discussion.
* Vijay Sarvepalli – Are cloud endpoints treated like regular endpoints? Jessica responded that each type of endpoint requires consideration and that we would need a cloud expert to help answer this. She felt that some aspects of cloud would be similar to traditional endpoints, but others (such as remediation) would be very different. John Field added that this harkened back to his earlier comment about provisioning servers – the server provisioning cloud VMs might be a good source of information, especially for short-lived images.
* Dave Waltermire – There might be a role for a managed security provider in SCAP. In this case, the endpoints would be on-premises, but the PCE and PES components might be run at a third party. How this looked and how it was supported is an open question.

Dave provided an overview of proposed solutions within the SCAP architecture, including SWID, NEA, and ROLIE.

# Question: Would it be useful to standardize anything else?

* John Kennel – It would probably be useful to standardize CMDB interactions in a cloud setting.
* Joe Wolfkiel – If the cloud provider had a CMDB, it would be useful to have standards to allow replication of relevant data to a local CMDB.
* Joe G – Many cloud solutions are API-driven, but none of those use the same standard. Joe Wolfkiel noted it would be good to at least be able to specify what one could ask and what one wanted back, even if there wasn't a standardized interface. Gary Gapinsky noted that this was effectively asking for a lingua franca masking the differences between APIs.
* Leland Steinke – OSCAL would at least be a start. Dave noted that what is being discussed goes beyond OSCAL.
* Joe Wolfkiel – We have a set of web services on our CMDB sour accreditation systems can talk to them. These are being build with JSON web interfaces rather than dedicated languages, but they could be a start. Joe G noted that JSON could readily be translated and re-visualized.

# Question: Is there value in standardizing the PCS-PES communications (I4)?

* Joe G – It would probably be a good idea.
* Joe Wolfkiel – We have UIS that allow people to define combinations of queries. I'm not sure if that is the same kind of interface.
* Dave Waltermire – The intent of the interface is to make the system more dynamic so that the types of collection can be changed and adapted. Joe Wolfkiel responded that their system did that. For example, it could collect all devices with a given piece of software, and then query those devices to see if they had a patch installed.

Vijay Sarvepalli asked how widespread SWID use was. Joe Wolfkiel noted that there is federal direction to report SWID tags. However, due to the large number of legacy systems without tags in his environment, he has not yet been able to use tags operationally. Dave noted that NIST is working to increase SWID adoption and support.