Ke, Jessica - Intern

From: Sharma, Saurabh <SaurabhSharma2@eaton.com>
Sent: Friday, June 18, 2021 12:01 PM
To: SBOM_RFC
Cc: Krzeszewski, John T
Subject: Eaton - Comments on the minimum elements for an SBOM

Hello

After careful review of the proposed requirements of the SBOM, we have the following comments.

1. Are the elements described above, including data fields, operational considerations, and support for automation, sufficient? What other elements should be considered and why?

   Other fields, that can be considered are:

   - License names, license families
   - Usage – [e.g] whether the components are statically or dynamically linked, merely aggregated, prerequisite for another component etc.
   - Commit activity – whether new changes to the existing components increasing, decreasing or stable. This is can help identify components that are end-of-life.

2. Are there additional use cases that can further inform the elements of SBOM?

- Software Bill of Material usually consists of Open-source and commercially acquired libraries. The data fields related to both these types of components needs to be consistent.
- Depth. The ideal SBOM should track dependencies, dependencies of those dependencies, and so on down to the complete graph of the assembled software. Complete depth may not always be feasible, especially as SBOM practices are still novel in some communities. When an SBOM cannot convey the full set of transitive dependencies, it should explicitly acknowledge the "known unknowns," so that the SBOM consumer can easily determine the difference between a component with no further dependencies and a component with unknown or partial dependencies.

3. SBOM creation and use touches on a number of related areas in IT management, cybersecurity, and public policy. We seek comment on how these issues described below should be considered in defining SBOM elements today and in the future.

h. Depth. As noted above, while ideal SBOMs have the complete graph of the assembled software, not every software producer will be able or ready to share the entire graph.

Identifying the dependencies is a hard task for any existing Software composition analysis software available in the market. This requirement should not be put as a mandatory data field in the SBOM.

i. Vulnerabilities. Many of the use cases around SBOMs focus on known vulnerabilities. Some build on this by including vulnerability data in the SBOM itself. Others note that the existence and status of vulnerabilities can change over time, and there is no general guarantee or signal about whether the SBOM data is up-to-date relative to all relevant and applicable vulnerability data sources.

All the vulnerabilities related data for components needs to be considered as point-in-time instead of cumulative. Adding vulnerability data in SBOM will add more complexity to the existing solution.

j. Risk Management. Not all vulnerabilities in software code put operators or users at real risk from software built using those vulnerable components, as the risk could be mitigated elsewhere or deemed to be negligible. One approach to managing this might be to communicate that software is "not affected" by a specific vulnerability through a Vulnerability Exploitability eXchange (or "VEX"),[14] but other solutions may exist.

Triaging information of a particular vulnerability can provide some context as to which vulnerabilities are actually deemed real vs the ones that have been mitigated.

4. Flexibility of implementation and potential requirements. If there are legitimate reasons why the above elements might be difficult to adopt or use for certain technologies, industries, or communities, how might the goals and use cases described above be fulfilled through alternate means? What accommodations and alternate approaches can deliver benefits while allowing for flexibility?

There may be cases that there are certain data fields that will be difficult to obtain. Manual changes to SBOM might not be the most user-friendly solution to incorporate in case of a inconsistency. Accommodations need to be thought for on a case-by-case basis.

Regards
Saurabh Sharma
Sr. Cybersecurity engineer, Product security
Cybersecurity Center of Excellence
Eaton Corporation
1000 Cherrington Pkwy, Moon township
SaurabhSharma2@eaton.com