June 17, 2021

Jennings R. Aske, JD, CISSP, CIPP/US
SVP, Chief Information Security Officer
NewYork-Presbyterian Hospital
jraske@nyp.org

National Telecommunications and Information Administration
U.S. Department of Commerce
1401 Constitution Avenue NW,

To Whom it May Concern:

Please find comments from NewYork-Presbyterian Hospital regarding the *Notice and Request for Comments on Software Bill of Materials Elements and Considerations*[1], published in the Federal Register on June 2, 2021. NewYork-Presbyterian Hospital is pleased to provide comments as the hospital has been a strong advocate for the use of software bills of material to provide software transparency to critical industry sectors including the healthcare industry.

Thank you for the opportunity to provide comments. NYP looks forward to continuing to engage with the community supporting software transparency, and the use of software bills of material to enhance the information security of the software consumed by the hospital.

Thank you,

Jennings R. Aske

[1] https://www.ntia.gov/federal-register-notice/2021/notice-rfc-software-bill-materials-elements-considerations

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

NYP Response: The seven elements described in the Notice, including data fields, are sufficient for operational considerations in relation to SBOM, including support for automation. As one of the organizations participating the in the NTIA-led Software Transparency efforts, NewYork-Presbyterian (NYP) has partnered with medical device manufacturers and other community stakeholders to demonstrate the SBOMs are practical, and provide value to both manufacturers and their customers.

NYP has added SBOM generation to its Security DevOps development pipeline, and is creating SBOMs aligned with this initial proposal. Further, the hospital has developed a software platform that is currently ingesting SBOMs utilizing the seven elements.

Further, while other fields could be added, NYP contends that the seven elements provide a scalable, first-generation baseline for SBOM generation to become a standard practice across different supply chains. Additional elements, including those associated with high assurance use cases (see response to question #4 below) can be added similar to how functionality is added to other technical standards over time, through consensus driven processes.

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

NYP response: NYP believes that many of the data elements that initially will be the focus of "high-assurance" use cases should in fact be added to the baseline, minimum standard. This issue is further discussed in questions # 3.d and #3.f.

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:

a. Software Identity: There is no single namespace to easily identify and name every software component. The challenge is not the lack of standards, but multiple standards and practices in different communities.

NYP response: NYP is confident that this will be resolved over time through consensus efforts, including the possible federation and interrelation of different hierarchical namespaces. Further, the component hash provides an alternative for identifying software components, and is currently supported by existing identification systems and SBOM formats like SWID, SPDX and CylconeDX. The VersionEye API at https://www.versioneye.com is one example of currently available tooling to support identifying software hashed software components included in an SBOM. All of this adds up to the fact that the current lack of single namespace is not a barrier to SBOM creation or utilization.

b. Software-as-a-Service and online services: While current, cloud-based software has the advantage of more modern tool chains, the use cases for SBOM may be different for software that is not running on customer premises or maintained by the customer.

NYP response: NYP believes there is a significant overlap in the use cases for cloud-based software consumed by healthcare organizations when compared to the software deployed in our data centers, on medical devices, etc. The use cases include procurement, on-going vulnerability management and security operations, incident response, and supplier service-level management.

The details of how these cases operationalize SBOM may be different, but the basic goals would be the same. As an example, NYP is typically responsible for patching the software running in our data centers. This would most often not be true for cloud services. However, the ultimate goal of timely and accurate software patching remains even if the responsibility is with the software supplier.

c. Legacy and binary-only software: Older software often has greater risks, especially if it is not maintained. In some cases, the source may not even be obtainable, with only the object code available for SBOM generation.

NYP response: These should not be issues affecting the success of SBOM. Tooling currently exists for scanning legacy and binary-only software to identify software components. Overtime additional tooling vendors and approaches will come to market making SBOM creation for legacy products even easier and more accurate. NYP does agree that software suppliers should be provided time for implementing capabilities for generating the SBOMs for legacy products.

d. Integrity and authenticity: An SBOM consumer may be concerned about verifying the source of the SBOM data and confirming that it was not tampered with. Some existing measures for integrity and authenticity of both software and metadata can be leveraged.

NYP response: As SBOM creation scales over time, issues related to SBOM integrity will be paramount. As noted in the response to question #3.f below, there are well-established cryptographic techniques that can be leveraged for validating the identity of an SBOM publisher that can be leveraged. The minimum seven elements in the proposal does include hashing of software components, which NYP believes is the most precise way to identify a software binary, providing assurance that the binary is in fact what is claimed in the SBOM.

e. Threat model: While many anticipated use cases may rely on the SBOM as an authoritative reference when evaluating external information (such as vulnerability reports), other use cases may rely on the SBOM as a foundation in detecting more sophisticated supply chain attacks. These attacks could include compromising the integrity of not only the systems used to build the software component, but also the systems used to create the SBOM or even the SBOM itself. How can SBOM position itself to support the detection of internal compromise? How can these more advanced data collection and management efforts best be integrated into the basic SBOM structure? What further costs and complexities would this impose?

NYP: NYP believes that it is too early in the SBOM journey to answer authoritatively.

f. High assurance use cases: Some SBOM use cases require additional data about aspects of the software development and build environment, including those aspects that are enumerated in Executive Order 14028. How can SBOM data be integrated with this additional data in a modular fashion?

NYP Response: NYP contends that the Software Component Verification Standard published by OWASP offers a practical model for expanding SBOM requirements in a modular format for high assurance use cases, including SBOM signature thorough signature authorities. The Standard can be found at: https://owasp-scvs.gitbook.io/scvs/

g. Delivery. As noted above, multiple mechanisms exist to aid in SBOM discovery, as well as to enable access by??? to SBOMs. Further mechanisms and standards may be needed, yet too many options may impose higher costs on either SBOM producers or consumers.

NYP response: NYP anticipates that a single standard for SBOM discovery and access will be not be defined. Rather, multiple methods will become standards, including publication via web services, on a device, etc. Consensus-driven processes currently underway, along with industry-specific standards based on supplier and customer need will naturally emerge. NYP agrees that too many options will impose higher costs, but feels

confident that as long as SBOMs include the minimum necessary fields and are machine readable, that support a number of methods for discovery and access are supportable. The SBOM ingestion platform NYP developed supports direct upload of SBOMs, but could easily ingest SBOMs published directly via a medical device or a web service.

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.

NYP response: More information about the software NYP consumes is always preferable, and a complete graph is ideal. NYP anticipates that as our suppliers begin providing SBOMs there will limits to the graph due to a lack of complete software inventories for software components in "legacy" products. As our supply chain builds SBOMs for new products we anticipate this will be less of an issue. And, as SBOM becomes standard practice, we anticipate that the fear, uncertainty and doubt (FUD) about SBOM will dissipate, leading suppliers to accept this as a common and helpful practice.

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.

NYP response: NYP agrees with this concern, and has no expectation that SBOMs mitigate the need for the hospital to stop performing other activities associated with vulnerability management. Simply put, an SBOM is another tool in the arsenal, one which provides data points to assist with our overall "defense-in-depth" approach to information security.

j. Risk Management. Not all vulnerabilities in software code put operators or users at 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''), but other solutions may exist.

NYP response: The Healthcare SBOM proof-of-concept (PoC) will explore consuming VEX documents in the next phase of the PoC. There is significant interest in the healthcare community for VEX as it will facilitate security-related use cases at healthcare delivery organizations that often have tens of thousands of medical devices from hundreds of manufacturers. NYP's SBOM platform will ingest the initial VEX documents as they are provided to PoC participants.

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?

NYP response: The proposal to-date aligns with the needs of the healthcare industry. As noted above, flexibility related to SBOM discovery and access will be necessary to support the different technologies and software deployed in a healthcare setting.