June 17, 2021

*Via email SBOM_RFC@ntia.gov*

Evelyn L. Remaley
Acting Administrator
National Telecommunications and Information Administration
Department of Commerce
Washington, DC 20230

**Subject: Software Bill of Materials Elements and Considerations (NTIA–2021–0001)**

Dear Acting Administrator Remaley:

The U.S. Chamber of Commerce welcomes the opportunity to provide the National Telecommunications and Information Administration (NTIA) with feedback on its June 2, 2021, request for public comment on minimum elements for a Software Bill of Materials (SBOM), as well as other factors that should be considered in the request, production, distribution, and consumption of SBOMs.[1]

## NTIA SBOM EFFORT IS PROGRESSING WELL

NTIA's SBOM initiative, which has been in progress for three years, was incorporated into the Biden administration's May 12, 2021, Executive Order (EO) 14028, *Improving the Nation's Cybersecurity*.[2] Section 4 of the EO calls for enhancements to the federal government's software supply chain security and directs NTIA to "publish minimum elements for an SBOM" within 60 days (July 11, 2021) of the EO's release.[3]

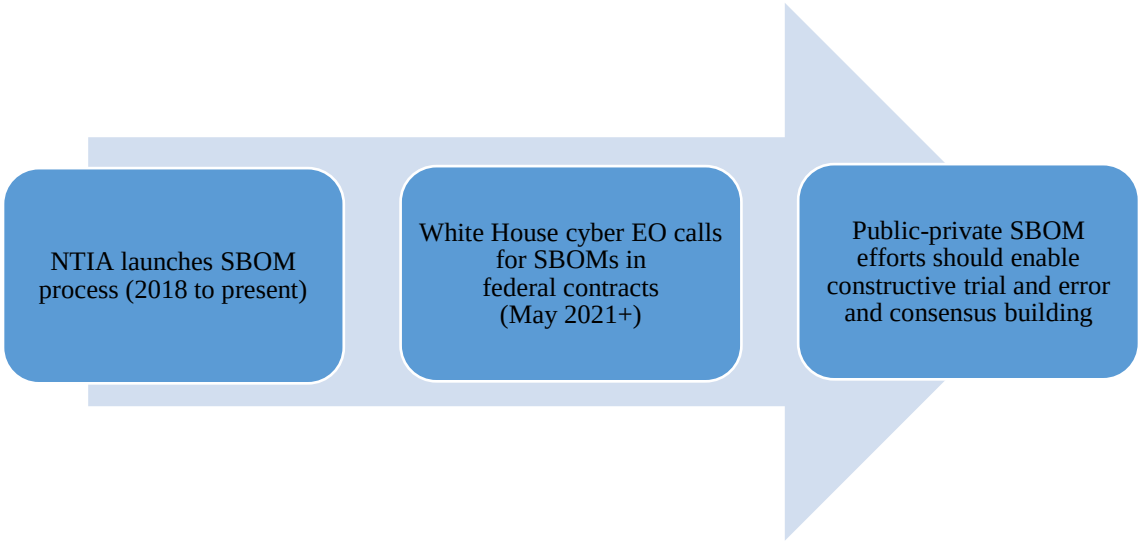
The Chamber has generally supported NTIA's multistakeholder SBOM process. Since 2018, Allan Friedman has led an open and a constructive effort enabling public and private parties to engage the agency and provide feedback. He has also joined our Cybersecurity Working Group on several occasions to inform businesses about SBOM developments and solicit private-sector input.

The Department of Commerce has a rich history of conducting public-private initiatives to identify tough policy and technical issues and make advances toward stronger cybersecurity. The Internet of Things (IoT) is a leading example. The Chamber shares many of the administration's goals regarding an SBOM. Nonetheless, the EO's complexity and aggressive timelines may hamper the methodical consensus building that is still needed to develop an SBOM, particularly for federal contracting, that achieves mutual industry-government goals, such as increased software transparency, trust, innovation, and security.

## THE EO SHOULD FOSTER SBOM EXPERIMENTATION AND CONSENSUS, NOT RIGID REQUIREMENTS

Chamber member opinion on an SBOM runs the gamut from strong support to robust skepticism.[4] The Chamber agrees with NTIA's view that a no one-size-fits-all approach should apply to SBOM creation and deployment whether in government or commercial markets. Policymakers in the executive branch and Congress should understand that while an SBOM is advancing well in some areas of the economy (e.g., the medical device and energy industries), which the Chamber is pleased to see, it needs more time to mature from a more macro standpoint. The EO could disrupt this progress. We believe that both the business community and NTIA want policymakers to ensure that an SBOM works for both businesses and agencies rather than see it become, unintentionally, an unproductive procurement and/or regulatory instrument.

What is more, an SBOM is often likened to a list of ingredients on a food package. But to those unfamiliar with an SBOM, such analogies can overly simplify the vast and complex nature of formats, procedures, uniformity, and protections (especially against foreign attackers) that are needed to make SBOMS manageable at scale across a growing cyber ecosystem.




## U.S. CHAMBER'S VIEWS ON SBOM BASICS ARE PRELIMINARY

Commenters have been afforded only 12 business days to review NTIA's notice and prepare replies. The Chamber does not attempt to answer every question in the notice. Instead, we have pulled together a number of organizations' views that track with topics NTIA is interested in vetting. Notably, Chamber thinking on some of the finer points on an SBOM are still in development given, among other things, the novel and technical nature of an SBOM.

## 1. Data Fields

NTIA proposes a definition of the "minimum elements" of an SBOM that builds on three broad, interrelated areas: data fields, operational considerations, and support for automation. Focusing on these three elements should "enable an evolving approach to software transparency and serve to ensure that subsequent efforts will incorporate more detail or technical advances," the agency notes. NTIA says that certain data about third-party components that make up software should be tracked. This "baseline component information" could include the items below:

- Supplier name
- Component name
- Version of the component
- Cryptograph hash of the component
- Any other unique identifier
- Dependency relationship
- Author of the SBOM data

SBOMs should include information about a supplier and component version. SBOMs should also enable users to identify software vulnerabilities based on a supplier's name, the component name, and version of the component. Thus, these three data fields that NTIA lists—supplier name, component name, and version of the component—should be included in the baseline elements of the SBOM.

NTIA is interested in including the cryptograph hash of software in an SBOM and should clarify the following:

- The purpose of the hash. A hash can be used to connect a digital artifact to a particular SBOM, which may be useful in investigating a security incident. A hash can also provide a method for a downstream supplier or consumer of an SBOM to verify that its multicomponent software artifact includes the component referenced by an upstream SBOM. Both use cases have value, but hashes do not need to be a baseline component.

- The method for creating a hash and whether the hash should be derived from binary or source code. A hash of the binary can have advantages for these reasons:

    - Source code is not always available to the SBOM author. This is especially true the further downstream in the supply chain that SBOM authorship occurs.
    - From a cybersecurity standpoint, the binary code, as opposed to the source code, is the definitive artifact of interest for determining vulnerability and exploitability of deployed software.
    - A binary hash will sometimes be created by a supplier when it provides a cryptographic signature of an executable file that is being provided to a supplier or consumer.

While there is some value in a hash of the component, it is not essential. Hashes should not be a mandatory minimum element of an SBOM. Where secondary SBOM authorship occurs, it should be possible to produce SBOMs that are considered in alignment with the requirements of the EO with respect to federal purchases. However, it will not always be possible to create a hash in the case where the SBOM author is not the software supplier. Hash of binary cannot be done for statically linked code.

---

NTIA should clarify what "Any other unique identifier" means. As with a hash, it seems like an "Any other unique identifier" should be an optional SBOM element. A unique identifier generated by a software supplier will not always be available where legacy software is involved. And it is unclear how a secondary SBOM could typically be able to obtain or create a unique identifier in a way that would be beneficial from a risk management point of view.

---

Based on several organizations' efforts in SBOM tools and methods, they advise against including a vulnerability list in the data fields. The EO states that "buyers can use an SBOM to perform vulnerability or license analysis, both of which can be used to evaluate risk in a product," and a vulnerability list is the product of one or more vulnerability analysis efforts, not an SBOM. This is a further security step compared with what it seemingly required in the SBOM. If the SBOM is equivalent to ingredients in food packaging, then including a vulnerability list would equate to adding health risks for each ingredient to food packaging. The ability to exchange with a standard format for currently associated vulnerabilities against the SBOM is advisable, though the analysis and reporting of such concerns should be held in a separate location.

---

Wrapper data refers to a data structure or software that contains—or "wraps around"—other data or software. Such information (e.g., the entity doing the wrapping, the date of wrapping, and a code signing certificate) could be considered in baseline SBOM data.[5]

## 2. Operational Considerations

### *2.1 Frequency*

One operational consideration that NTIA points to is the frequency of SBOM creation and tracking. The agency says, "Operational considerations touch on when and where the SBOM data is generated and tracked. SBOM data could be created and stored in the repository of the source. For built software, it can be tracked and assembled at the time of build. A new build or an update to the underlying source should, in turn, create a new SBOM."

A public ledger could be well suited to managing SBOM creation and tracking, where each new version and release of an SBOM by a supplier adds a new item to the ledger. These

ledgers could be made publicly available, cryptographically authenticated, and wrapped in smart-contract terms and payments to reduce potential abuse of the ledger, while also protecting the value of the developed technology by the core industries of the U.S.

*2.2 Depth*

NTIA notes that the ideal SBOM should "track dependencies, dependencies of those dependencies, and so on down to the complete graph of the assembled software," which is very ambitious. However, organizations should have some freedom from rigid dependency mapping, particularly in the early stages of SBOM development, which NTIA recognizes is still new in many communities. Conveying dependencies is a code functionality element, not just an inventory.

Dependencies share intellectual property (IP) relevant information and algorithms that can be too easily exposed by such declarations. Dependency concepts should be limited to public SBOMs. In many cases, SBOMs should be maintained as confidential. Making them public would increase attacks, and the built software is considered proprietary competitive information. Hence, the dependency relationship(s) data field should be reconsidered. Relying on this data can be fraught with challenges. Software components can be broken down into increasingly smaller components and levels of complexity. The value of a dependency relationship is debatable from a supply chain management standpoint. A dependency relationship, for instance, is usually meaningful (e.g., functional) to the author of the software but not to the consumer or the entity that receives an SBOM.

NTIA should reconsider including dependency relationship(s) in the basic SBOM structure. A minimum depth cannot be explicitly mandated because not every software package will be accompanied with complete data. Empty or suboptimal tables can indicate that gaps and risks in relevant information and mitigation plans need to be identified depending on the context of how the package or dependent element is used.

*2.3 Delivery*

NTIA calls for SBOMs to be "available in a timely fashion to those who need them and have proper access permissions and roles in place." The access points and delivery mechanisms are loosely defined in the agency's notice such that there could be varying interpretations, including overlapping and/or conflicting government requirements. Access protocols need to be clearly defined based on industry consensus.

## ADDITIONAL POINTS TO CONSIDER

### Software Identity

NTIA notes that 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."

Mandating a universal or standard software identification method across a domain (e.g., an agency or a department) is feasible. Nevertheless, there will continue to be multiple standards and practices used at a product's origin due to its dependencies on commercial and open source software components, modules, and libraries. Any product or program that utilizes components from open sources or proprietary sources is likely operating across namespaces in which there can be no assumption of standardization and no imposition of standardization.

One potential solution, at least with respect to the Department of Defense, is to use the Platform One approach of Iron Bank,[6] which containerizes many widely used open source and commercially available software products and then uniquely identifies them and applies a version of the component. These packages have been scanned, catalogued, and vetted for use by Platform One and its user community. Reinventing this methodology, at least in this case, is unnecessary.

### Threat Models

Some narratives around an SBOM attempt to make it the pinnacle of software assurance. Yet an SBOM will neither describe who, how, and when a particular component was compromised nor provide the precise exploit code, including in a standardized format, that the attacker is embedding in the corrupted component. The questions that should be considered are Who benefits from more information, and what is the cost to obtain it?

### Vulnerabilities ("Not Vulnerable as Used")

A missing element in the notice is the ability to pinpoint components that are "not vulnerable as used" components. One difficulty with an SBOM is that it tells part of a story regarding software assurance but not the entire story. This issue speaks to mapping to vulnerability databases, but it will only tell you that there is a known vulnerability (e.g., CVE)[7] in a certain component. However, if a component is not vulnerable, then such information can be misleading at worst and incomplete at best. It could lead to customers erroneously pushing vendors to upgrade libraries that do not need upgrading. And vendors may produce patches that do not need to be produced, amounting to expending scarce resources on low-priority issues.

Veracode has done research that indicates for reviewed products less than 5% of them contain a library with a vulnerability are vulnerable. Even allowing for a library having multiple CVEs, some of which are exploitable and some of which are not, there are still many cases in

which an upgrade to a newer library version is not only unnecessary but is a misallocation of scarce resources.[8]

**Open Software and Commercial Software Balance**

Open source software is a good example of quality and innovative code that can be created consistent with the principles of an SBOM and the EO. This type of code, though, can be overweighted in SBOM formats and processes because it is created openly and available to be connected to many types of software. Makers of open source software believe that it would be prudent to balance stakeholders' expectations between open source software and commercial software (aka commercial-off-the-shelf software, or COTS) to help achieve the goals of an SBOM and the EO. Specifically, if software components are key to creating an SBOM, then such thinking should apply to the COTS community. To achieve a certain level of granularity at the component level, there needs to be workable harmony in the SBOM system across open source and commercial code bases.

***

The Chamber appreciates the opportunity to provide you and your NTIA colleagues with comments on SBOM elements and considerations. If you have any questions or need more information, please do not hesitate to contact Christopher Roberti (croberti@uschamber.com, 202-463-3100) or Matthew Eggers (meggers@uschamber.com, 202-463-5619).

Sincerely,

Christopher D. Roberti
Senior Vice President,
Cyber, Intelligence,
and Supply Chain Security

Matthew J. Eggers
Vice President, Cybersecurity Policy

Notes

[1] National Telecommunications and Information Administration (NTIA), "Software Bill of Materials Elements and Considerations," *Federal Register*, June 2, 2021. https://www.federalregister.gov/documents/2021/06/02/2021-11592/software-bill-of-materials-elements-and-considerations

[2] White House, Executive Order (EO) 14028, *Improving the Nation's Cybersecurity*, May 12, 2021. https://www.whitehouse.gov/briefing-room/presidential-actions/2021/05/12/executive-order-on-improving-the-nations-cybersecurity

https://www.federalregister.gov/documents/2021/05/17/2021-10460/improving-the-nations-cybersecurity (*Federal Register*, May 17, 2021)

[3] A White House fact sheet on the EO declares:

> Improve Software Supply Chain Security. The Executive Order will improve the security of software by establishing baseline security standards for development of software sold to the government, including requiring developers to maintain greater visibility into their software and making security data publicly available. It stands up a concurrent public-private process to develop new and innovative approaches to secure software development and uses the power of Federal procurement to incentivize the market. Finally, it creates a pilot program to create an "energy star" type of label so the government—and the public at large—can quickly determine whether software was developed securely. Too much of our software, including critical software, is shipped with significant vulnerabilities that our adversaries exploit. This is a long-standing, well-known problem, but for too long we have kicked the can down the road. We need to use the purchasing power of the Federal Government to drive the market to build security into all software from the ground up.

https://www.whitehouse.gov/briefing-room/statements-releases/2021/05/12/fact-sheet-president-signs-executive-order-charting-new-course-to-improve-the-nations-cybersecurity-and-protect-federal-government-networks

Section 4(e)(vii) of the EO requires contractors to provide government purchasers with an SBOM for each product directly or by publishing one on their public websites.

[4] Some opponents of an SBOM say that it unworkable. One member told the U.S. Chamber that an SBOM "will foster a cascading effect of vulnerabilities and create a high concentration of risk and map for bad actors and adversaries to exploit."

With respect to an SBOM's utility, another member said the following: "Our organization's interest in an SBOM is as a consumer of the data to manage supply chain risk. In recent years, some organizations have added language to their standard contracts to include an SBOM when the product or element in question includes software. An SBOM can be an important tool for companies to manage their risk. At the same time, an SBOM is a relatively recent development, so we urge NTIA to maintain flexibility and encourage experimentation with an SBOM to see how it can create value for stakeholders.

"The key is that SBOM information is useful and could be a best practice for software suppliers to provide constructive information. But we don't want a rigid system where an SBOM has to be approved by government. NTIA can play a role in encouraging SBOM use and providing examples about how to make it useful. There will be some fits and starts in SBOM evolution over time."

Still a third member said, "Our organization supports NTIA's proposal to create the identified 'baseline component information.' This information should be made readily available to the purchaser of any equipment that contains software. This is critical to enabling purchasers to identify the components that make up the equipment they are purchasing and, thus, avoid or at least reduce the risk that a purchaser will unknowingly purchase equipment that contains software manufactured by a company deemed by U.S. policymakers a threat to national security. The lack of such information could put small telecommunications providers at risk of losing essential federal funding for their network infrastructure."

[5] https://www.pcmag.com/encyclopedia/term/wrapper

[6] https://software.af.mil/dsop/platform-one-resources

[7] https://cve.mitre.org

[8] See Chris Wysopal, Veracode, "How Understanding Risk Is Changing for Open Source Components," March 2019, RSA Conference. Also see "Open Source Components: Vulnerability Information Sources & Vulnerability Likelihood," July 19, 2018. "For Java, Ruby and Python, less than 5% of products that contain a library with a vulnerability are vulnerable."
https://published-prd.lanyonevents.com/published/rsaus19/sessionsFiles/12890/PDAC-R11-How-Understanding-Risk-Is-Changing-for-Open-Source-Components.pdf
https://www.ntia.doc.gov/files/ntia/publications/wysopal_swct_kickoff_perspective.pdf