Before the
NATIONAL TELECOMMUNICATIONS AND INFORMATION ADMINISTRATION
Washington, DC 20230

In the Matter of	)	
	)	
Software Bill of Materials Elements and Considerations	)	Docket No. 210527-0117
	)	

COMMENTS OF ERICSSON

I. INTRODUCTION

Ericsson supports the goals espoused in the President's Executive Order on Improving the Nation's Cybersecurity[1] which defines the term "Software Bill of Materials" or "SBOM" as a formal record containing the details and supply-chain relationships of various components used in building software. Ericsson has vast experience in secure software development, following industry best practices which serve as a model for using proprietary, third-party, and open source software in development projects. As part of this process, Ericsson maintains internal SBOMs for its products and make these available to customers on a contractual basis. Following industry best practices, Ericsson evaluates its software to identify vulnerabilities, including SAST/DAST, scans, code reviews, and pen testing.

Ericsson urges NTIA to consider the following SBOM requirements:

1. SBOM should be delivered in a controlled manner with limited distribution to only trusted third parties according to contractual agreements which should include, confidentiality and integrity protection and mutual authentication during the transfer process. SBOMs should not be publicly disclosed.

2. Third parties in receipt of SBOM must securely store the SBOM data with modern ciphers for confidentiality and integrity protection.

[1] Exec. Order No. 14,028, 86 Fed. Reg. 26,633 at Section 4(f) (May 17, 2021).

3. Third parties in receipt of SBOM must maintain secure, controlled access to it according to contractual agreement.

4. Apply SBOM requirements based upon criticality rating of the software and a commonly accepted definition of 'critical software.' Ericsson cautions that a one-size-fits all approach for software could place an undue burden on software vendors and their customers. If all software is critical then no software is critical. Criticality ratings and the definition of 'critical software' may vary between industries and use cases. NTIA should only proceed with SBOM requirements when 'critical software' is appropriately defined.

5. Vendors should be responsible to assess the impact of a known vulnerability in the form of a Common Vulnerabilities and Disclosures ("CVE") system based upon software analysis, operational environment, and use case. The vendor assigns the impact rating (Critical/High/Med/Low).

II. DISCUSSION

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

   Data fields:

   As noted in the SBOMs should include "baseline component information" that includes: 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 can contain commercially sensitive content that can be used by competitors to gain business advantages and by malicious hackers to reverse engineer and exploit known vulnerabilities, especially in instances where they may not normally have access or visibility to the software. Widely available consumer software has a much broader level of availability and exposure as compared to the unique solutions generally deployed within critical infrastructure networks. SBOMs should be treated as confidential information and documented in a consistent, repeatable way. For this reason, Ericsson recommends that the set of required baseline

components information fields be limited to the following set of fields to provide the proper level of transparency without introducing unnecessary risk:

1. Component name
2. Supplier name
3. Version of the component
4. Origin (could be different than Supplier)

Operational considerations

SBOMs should include a set of operational and business decisions and actions that establish the practice of requesting, generating, sharing, and consuming SBOMs, including consideration for:

Frequency:

Operational considerations should touch on when and where the SBOM data is generated and tracked.

The process of generating, verifying, signing, tracking and updating SBOM data should be part of a mature Software Development Lifecycle ("SDLC") process. Ericsson follows best practice for secure software development and internally maintains a detailed SBOM during the entire SDLC such that vulnerability notifications and updates can be properly and safely handled across the installed customer base. This includes a cataloging or inventory of software in use as well as check-in and check-out of code to confirm file integrity and authorization/identity of the developer. There should be a distinction between the level of detail collected at an internal level versus what is shared externally (where there could also be variations, depending upon context and criticality level of the software). Ericsson recommends an attestation that such best practices are followed.

Depth:

SBOMs should track dependencies, dependencies of those dependencies, and so on down to the complete graph of the assembled software. When an SBOM cannot convey the full set of dependencies, it should explicitly acknowledge the "known unknowns," so that the SBOM consumer can determine the difference

between a component with no further dependencies and a component with unknown or partial dependencies.

The complexity of generating a SBOM full set of dependencies needs to be addressed and resolved by the government and the industry before making it a compulsory deliverable within the SBOM. Listing the "known unknowns" when SBOM cannot convey the full set of dependencies, it could give an inaccurate picture to the SBOM consumer which might lead to false assessment about the vulnerabilities of the product. A software supplier should be able to list third-party software components at the level of the software module delivered from the 3rd-party to the software supplier.

Delivery:

SBOMs should be available in a timely fashion to those who need them and have proper access permissions and roles in place. Anyone offering SBOMs must have some mechanism to deliver them.

The commercial sensitivity of SBOM dictates that the information that is shared externally is limited to authorized users only to prevent business compromise and security exploitation. Access to SBOM data should be authenticated and authorized through secure access control only on a need-to-know basis.

Automation support:

SBOMs should include support for automation, including automatic generation and machine-readability. SBOMs should be machine-readable and allow for greater benefits through automation and tool integration. The SBOM community has identified three existing data standards/formats that can convey the data fields and be used to support the operations described in this RFC: SPDX, CycloneDX, and SWID tags.

Automatic generation and machine-readability support is important. But in the early stages of the SBOM adoption, the different software components that contain specific SBOM data might not be ready to automatically provide this data to the entity that produces the SBOM or support the SPDX, CycloneDX and SWID tags standards. Pilot programs might be required to

demonstrate and inform these capabilities before the wider adoption of these standards and tools Using the output from these programs, NTIA could consider general adherence to data standards and formats for future products. SBOM automatic generation and machine-readability support for legacy products should be avoided at this point.

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

The usage scenarios of the SBOM might differ from one piece of software to another. There might be cases where a software component that is used by a product and included in its SBOM is identified as vulnerable, but this vulnerability might not make the overall product vulnerable, In such scenarios, SBOM might give a false signal about the vulnerability of the product. Vulnerability ratings can be use case dependent. A vulnerability must be exploitable to be fully relevant, if not exploitable it is only to be seen as a weakness. Thus, the level of details related to the security criticality of software components in specific products and the control of SBOM data needs to be identified and managed by the organization developing the final product.

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.***

Modern software development frequently employs agile and continuous integration/ continuous development ("CI/CD") processes. This constant evolution is not necessarily practical to align with a "patch/update" mentality, where a software version is significantly changed. Consideration for "as a service" models should also factor in ownership of software versus use of software, as an SBOM is a representation at a point in time of that build, not a "real-time" attestation of what had been purchased.

SBOM data should be protected by minimizing and controlling access. SBOM material should be stored in a secure location and accessible only to customers through a secure portal authorized for verified customer use.

Assurance refers to the activities conducted to ensure that the final product is secure when it is running in its target environment. These activities include risk assessments, privacy impact assessments, secure coding, vulnerability analysis and hardening. A typical software development framework specifies the relevant assurance activities for each category in every phase of the product lifecycle: Source, Develop and Deliver. Depending on the characteristics of the product, the appropriate level of assurance activities – basic, advanced or tailored – is set for each category. Ericsson's Security Reliability Model (SRM) ensures that product security and privacy gaps are identified as early as possible – ideally during the conception phase of a new feature, product, service and solution. This allows us to control the direction of product development towards secure implementation. Considering relevant security and privacy requirements during the product design phase enables a secure implementation. Ericsson utilizes industry best practices to product security and privacy by design.[2]

Published CVEs will be the acceptable format for providing known vulnerabilities within the SBOM publication.

III. CONCLUSION

As detailed above, Ericsson is a strong proponent of secure software assurance practices and looks forward to continuing the discussion with NTIA. To that end, we suggest that NTIA hold further industry workshops, launch exemplary pilot programs and continue its collaborative

[2] *See* THE ERICSSON SECURITY RELIABILITY MODEL, https://www.ericsson.com/495435/assets/local/security/the-ericsson-security-reliability-model.pdf.

dialogue with industry. SBOM should be implemented in a way that will ultimately reduce organizational risk in a consistent and measurable way. There are multiple topics which remain open for discussion and further analysis, which could ultimately impact the development of effective and implementable software assurance guidelines, such as:

- Definition of "critical software;"
- Standards for ensuring secure data life cycle including file transfer, storage, access, and destruction;
- Ciphers for confidentiality and integrity protection of data-in-transit, data-in-rest, and data-in-use; and
- SBOM Depth for third-party and open-source software.

Respectfully submitted,

Mohammad Khaled
Security Solutions Director

Scott Poretsky
Director of Security

ERICSSON
6300 Legacy Drive
Plano, TX 75024

June 17, 2021