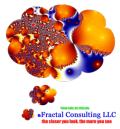
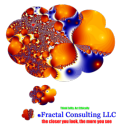


This position paper is being sent in response to the National Telecommunications and Information Administration (NTIA) Request for Public Comment on "Software Bill of Materials (SBOM) Elements and Considerations" [1] to help NTIA fulfill Executive Order (EO) on Improving the Cybersecurity of the Federal Government (14028) [2]. I would like NTIA to consider my views as an individual citizen with expertise in this field. The views expressed are my own and do not represent any organizations I participate in, nor any of my clients, nor my previous employers.

I applaud the NTIA SBOM efforts and have great confidence in NTIA's ability to fulfill its mission in the area of supply chain security. I was first exposed to the concepts behind SBOM decades ago when I was technical lead on the first instantiation of the US Government's Managed Trusted IP Service [3]. Like most, I initially considered the problem intractable and resisted implementation - but the benefits greatly outweighed the costs and I have publicly been a strong SBOM advocate for many, many years. I have been a member of Iamthecavalry.org since very near its inception and I believe the work the NTIA is doing is important for public safety [4] beyond the 'critical infrastructure' of the EO. I volunteer a significant portion of my time pro bono to the NTIA Open and Transparent Process on Software Component Transparency [5] and have co-authored some of the NTIA SBOM material [6].

For the TL;DR version, just read the headings. My 18 recommendations are:

1. Go beyond EO use cases – Q2
2. Document Use Cases - Q2
3. Support CycloneDX and SPDX - Q1
4. 'Minimum' Data Elements are different for different use cases - Q1
5. Define the 'Beyond Baseline' Data Elements – Q1
6. Add Know-Unknown as a Data Element – Q1, Q 3h
7. Add SBOM Metadata – Q1
8. Create Best Practices beyond SBOM creation/analysis - Q1, Q2
9. Use OpenC2 – Q2, Q3g
10. Define Hash Data Element better - Q1
11. Best Practice on More Depth – Q 3h
12. Create Frequency Best Practices - Q1, Q 3c,e
13. Don't forget vanilla use cases - Q1, Q3d,e,f
14. Create, but don't prioritize, 'legacy' use cases – Q3c
15. Ignore rate-of-change fallacy - Q5
16. Continue with Proof-of-Concepts - Q5
17. Create and Track Metrics - Q5
18. Participate Internationally – Q5



1. Go beyond EO use cases – Q2

Question 2 asks for additional use cases. It should be noted that the EO use cases (cybersecurity of supply chain for critical infrastructure) are only a subset of many SBOM use cases of value to the US public. I.e. there are security use cases beyond critical infrastructure and there are non-security use cases such as licensing and software development efficiency/effectiveness. Although not part of the EO tasking, NTIA should continue to work the entire spectrum of use cases.

2. Document Use Cases Q2

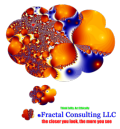
Question 2 asks for additional use cases. I recommend NTIA curate/develop/publish more use cases, including examples to aid in understanding the differences between the use cases and the impact of those differences on the content/creation/use of the relevant SBOMs. The Phase 1 (2019) document, "Roles and Benefits from SBOM Across the Supply Chain"[7], is a great start but more specific use cases are needed. This should also include use cases beyond those necessary to meet the tasks in the EO, and should include the differences in necessary data elements (see next section). Use cases should be created for each part of the SBOM lifecycle. Use cases should be created for both 'during build' and 'after built' SBOM creation, including ripple into analysis/decision/act use cases. Use cases should be created for creating/using 'complete' SBOMs, and for creating/using SBOMs with 'known-unknowns'. Use cases are needed for 'beyond creation/analysis' (see later section). Use cases are needed for 'single file executable' components and use cases are needed for 'collection of files' components. Use cases are needed for 'source' SBOMs, for 'built' SBOMs, and for 'process to build' SBOMs. Most importantly, use cases are needed for current state (i.e. not many SBOMs, and many 'known-unknowns') and future state (most products with complete SBOMs).

3. Support CycloneDX and SPDX – Q1

NTIA Question 1 asks for comments on the proposed automation support. The automation support material prior to Q1 mentions SPDX, CycloneDX, and SWID tags as SBOM data formats. I recommend encouraging/allowing the use of CycloneDX[8] and SPDX[9] and not 'picking a winner'. Significant innovation has occurred in the last 3 years on SBOM formats, at least partially spurred by competition between these formats. As they are easily converted between, IMHO all 3 formats should be allowed and encouraged.

4. 'Minimum' Data Elements are different for different use cases - Q1

NTIA Question 1 asks if the data fields, operational considerations, and support for automation are sufficient. In a previous section, the document proposes a definition for 'minimum elements' and a 'baseline component identification'. Care should be taken not to confound the work on 'commonality' across the use cases with the work on defining the data fields needed for a given use case. It is clear that component information is necessary by definition, and therefore it is clear that some means to identify those components is necessary. But for many use cases additional information is necessary. For example, licensing use cases require licensing information for the component either in directly in the SBOM or a separate relationship of the SBOM component to its licensing information. That licensing information is not necessary for the EO use cases



(cybersecurity of supply chain for critical infrastructure). However other information is necessary (e.g. information about vulnerabilities, the exploitability of those vulnerabilities, provenance/pedigree information, and other information to assist in risk/trust evaluations) either in the SBOM or a separate relationship of the SBOM component to the relevant security information (e.g. the exploitability information in CSAF for a VEX document associated with an SBOM). This context-sensitive 'minimum' is more analogous to an analog scale than a binary decision of 'minimum' vs 'beyond baseline'. The minimum for nuclear launch software might have more 'minimum' data elements than for other critical infrastructure which still may have more data elements than in a typical business or government environment.

5. Define the 'Beyond Baseline' Data Elements – Q1

The key to solving use-case-dependent-minimum (see precious section) is the 'beyond baseline' (NTIA Framing term) elements need to be specified and agreed to so that each use case (or even each particular ecosystem) can make a risk-informed decision to what is necessary for that particular use case. By specifying the 'beyond baseline' data elements, the NTIA would greatly increase the likelihood of tool interoperability and commonality of tools across many diverse ecosystems. This is necessary because of the commonality of the underlying components in the SBOMs in these different ecoystems.

6. Add Know-Unknown as a Data Element – Q1, Q 3h

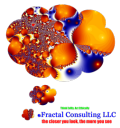
NTIA Question 1 asks if the proposed "minimum" data fields are sufficient. NTIA question 3h asks for comments on the depth of the SBOM and on the issue of 'known-unknowns'. I recommend each component in an SBOM have a data element associated with the completeness of the depth of the dependencies. In all the use cases I can think of, a user needs to be able to distinguish whether there are known-unknowns. SBOMs with known-unknowns (e.g. "one-hop" SBOMs) are still valuable, but complete SBOMs are even more valuable. For the EO use case (cybersecurity of supply chain for critical infrastructure), I recommend this 'completeness' data element be required and best practice would be it's value is 'complete' (ie no known-unknowns).

7. Add SBOM Metadata – Q1

NTIA Question 1 asks if the proposed "minimum" data fields are sufficient. The RFC does not mention metadata about the SBOM itself beyond 'author of the SBOM'. The SBOM itself should include metadata such as version, creation timestamp, hash of the SBOM, etc.

8. Create Best Practices beyond SBOM creation/analysis Q1, Q2

NTIA Question 1 asks what to consider beyond the proposed data-fields/operations/automation approach, and Question 2 asks for additional use cases. To meet the EO objectives will require vendor-agnostic, machine-speed cyber-defense automation as proposed by Integrated Adaptive Cyber Defense (IACD)[10], and it will require that automation through the entire supply chain. The NTIA Open and Transparent Process on Software Component Transparency is in a unique position to meet these needs and I recommend the group work beyond SBOM creation/analysis to include best practices for decision making and taking action as a result of SBOM analysis. Note these



actions involve both short-term mitigation (e.g. do not let the device on the network in a 'comply to connect' use case[11]. where the SBOM is not available or analysis of the SBOM results in a decision that the risk of the vulnerabilities is too great) as well as reporting/disclosure process, auto-remediation, triage, investigation, analysis, reproduction, fix development, and communication management. I think 'beyond SBOM analysis' is particularly important for the critical infrastructure of the EO, but it is also very important in the rest of the US economy as well.

9. Use OpenC2 – Q2, Q3g

Question 2 asks for additional use cases. Question 3g asks for comments on delivery mechanisms. I recommend NTIA continue to support the use of OpenC2[12] for sharing and transferring SBOMs[13]. In the operational considerations section of the request for comment, NTIA quotes the EO on the benefits of automation. As noted in a previous section, meeting the EO objectives will require vendor-agnostic, machine-speed cyber-defense automation. OpenC2 is a standardized language for the command and control of technologies that provide or support cyber defense and is ideally suited to meet NTIA needs in this area. I recommend NTIA also consider OpenC2 in use cases where actions are necessary (e.g. sending commands as result of SBOM analysis).

10. Define Hash Data Element better Q1

NTIA Question 1 asks if the proposed "minimum" data fields are sufficient. It is not entirely clear from the wording whether hash is one of multiple ways to identify a component (ie it's optional), or if hash is a required field. I recommend NTIA have more specificity with respect to hash. A hash works best on a well-defined immutable object, generally a file. A hash is very appropriate for the use case of identifying an already-built single file component. Care must be taken for components with multiple parts. It is a tractable problem but involves an agreed set of rules on how to create the hash. Hash for a mutable object is much more problematic. Unfortunately many of today's software products have many optional components which make them more like a mutable object. The problem is also tractable but will require more work and may require operational changes (e.g. a customer may have recognize they are accepting additional risk when they choose only install a subset of patches). I recommend NTIA specify when hash is appropriate (eg for pre-built single-file executable) and work on the methods to increase the number of components it can be used for.

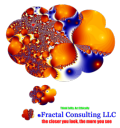
11. Best Practice on More Depth – Q 3h

NTIA question 3h asks for comments on the depth of the SBOM and on the issue of 'known-unknowns'. As stated in a previous section, I recommend 'completeness' (or conversely 'known-unknown') be one of the minimum data fields for each component. I also recommend NTIA have a best practice on completeness/known-unknowns, encouraging SBOMs to be complete while acknowledging an SBOM with known-unknowns is still valuable.

There is a secondary depth issue on 'who does the work'. The scenario is:

- Cindy's Cyrpto module has a complete SBOM
- Allan's App includes/uses/is-dependent-on Cindy's Crypto

Allan has several choices when creating his SBOM. He can



- create a one-hop SBOM including dependency on Cindy's Crypto
- create a one-hop SBOM referencing Cindy's SBOM, including Cindy's SBOM as a separate file (or link)
- create a complete SBOM incorporating Cindy's SBOM within the Allan App SBOM

I recommend case 1 be considered 'known-unknown' (albeit technically it is knowable, it is not 'included'). Case 2 is complete, but I recommend Case 3 as 'best practice'. The supplier does the inclusion once and it applies to all SBOM consumers. Case 2 requires each SBOM consumer to independently do the work.

12. Create Frequency Best Practices Q1, Q 3c,e

NTIA Question 1 asks for comments on the proposed 'frequency' operational consideration. NTIA Question 3e asks how SBOM can be used to detect internal compromise.
I recommend NTIA create best practices for:

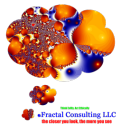
- creating SBOMs at build time (preferred)
- creating SBOMs after built (to verify build SBOMs for Q 3e, or for legacy use cases Q3c)
- completeness of the SBOM
- updating the SBOM when known-unknown dependency data becomes known (a new version of an existing SBOM),
- when software is updated/patched/changed (a new SBOM).

13. Don't forget vanilla use cases Q1, Q3d,e,f

The EO use case is cybersecurity of supply chain for critical infrastructure which is a high assurance (Q3f) use case with a very capable adversary (Q3e) requiring a high degree of integrity and authenticity (Q3d). This is very important. Today, most of software is not part of the critical infrastructure of the US, most software does not have an SBOM, and most software has unpatched vulnerabilities. Fixing this (i.e. everything not included in 'critical') is also very important. Solutions (likely different solutions) should be developed for both 'critical' and 'vanilla'.

14. Create, but don't prioritize, 'legacy' use cases – Q3c

NTIA Question 3c seeks comments on defining SBOMs after a system is built, and the source code may not even be available. As in previous sections, I recommend best practices should be developed for these use cases. The 'beyond end of (security) support' use case is a specific 'legacy' use case that should be enumerated. It is important, particularly in industries with products with long in-service lifetimes. However, this does not mean new products should be absolved from providing complete SBOMs created at build time just because it may be hard to create an SBOM for a device manufacture-discontinued years ago. I recommend focus should first be on new products and future purchases.



15. Ignore rate-of-change fallacy Q5

The NTIA Instructions for Commenters invites comments on issues not included in NTIA Questions 1-4. I recommend ignoring the 'rate of change' arguments around data elements in an SBOM. Some argue including/excluding data elements based on their rate-of-change (e.g. include licensing because they are "static", don't include vulnerabilities because they are 'dynamic').

Some claim the SBOM components for a given built version of a product are static and immutable. In a perfect world of complete SBOMs for all software, that may be true. However the state of today's software is quite different and I think our processes should allow for new versions of an SBOM as information is discovered about dependencies. I predict many SBOMs will change frequently as component SBOMs move from 'known-unknown' to 'known'.

Some claim the licensing information about software components is static and immutable. I disagree with that argument. Similar to dependencies, today's world is not perfect and licensing information may be added later. Also, licensing changes – particularly commercial licensing. It may only be on a reasonably small percentage of the components, but licensing does change.

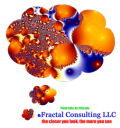
Some claim vulnerability information about software components is dynamic and 'changing all the time'. I agree it is dynamic, but approximately the same rate of change as for licensing – and both are significantly less than the rate of change for SBOMs – at least until we reach SBOM nirvana. There were ~18K CVE's in 2020. But there are 190M github repos and some estimate the 18K CVE's were out of appx 1B possible applications. I long for the time when 99% of SBOMs are 'complete' (which is what it would take to make then 'static').

16. Continue with Proof-of-Concepts – Q5

The NTIA Instructions for Commenters invites comments on issues not included in NTIA Questions 1-4. Significant value has been gained from the NTIA support of the Healthcare Proof of Concept (PoC) [14]. Keep up the good work. I believe the automotive and energy PoCs will also be successful. I recommend similar efforts in other industries – including various agencies in the US government.

17. Create and Track Metrics Q5

The NTIA Instructions for Commenters invites comments on issues not included in NTIA Questions 1-4. I recommend NTIA establish a program of metrics on SBOM availability and adoption. Admittedly there are potential rabbitholes, but metrics would allow observing improvement over time and maybe even provide feedback on what is helping and what is hindering SBOM adoption.

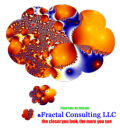


18. Participate Internationally

To meet the goals of the EO will require adoption of SBOM best practices beyond the borders of the US. I recommend NTIA take a more active role in international standards bodies to gain awareness and adoption of NTIA efforts in supply chain security. In particular, NTIA should regularly participate and provide their expertise to International Telecommunications Union Standardization Study Group 17 on Cybersecurity[15], as well as aid International Telecommunications Union Development Study Group 2, Question 3 on Cybersecurity[16] by helping with cybersecurity capacity development for developing countries (in coordination with the US State Department and the US Agency for International Development). In these efforts, I recommend NTIA propose ITU 'guidelines' and 'best practices' as opposed to de jure 'recommendations' (aka standards). We don't need to 'pick a winner', just help with awareness and adoption.

Thank you for the opportunity to input into the process.

Respectfully,
Duncan Sparrell CISSP, CSSLP, CCSK
Chief Cyber Curmudgeon
sFractal Consulting
https://www.linkedin.com/in/duncan-sparrell-cissp-csslp-ccsk-038137/



Footnote References:

[1] Federal Register https://www.federalregister.gov/documents/2021/06/02/2021-11592/software-bill-of-materials-elements-and-considerations
[2] Executive Order 14028, Improving the Nation's Cybersecurity, May 12, 2021, https://www.federalregister.gov/documents/2021/05/17/2021-10460/improving-the-nations-cybersecurity
[3] Managed Trusted Internet Protocol Service, https://web.archive.org/web/20090509183240if_/http://www.gsa.gov/gsa/cm_attachments/GSA_DOCUMENT/Managed%20Trusted%20Internet%20Protocol%20Service_REDACTED-Final_R2-z96W_0Z5RDZ-i34K-pR.pdf
[4] Sparrell, D. Cyber-safety in healthcare IOT. 2019 ITU Kaleidoscope: ICT for Health: Networks, Standards and Innovation (ITU K), Atlanta, GA, USA, https://doi.org/10.23919/ITUK48006.2019.8996148 (2019).
[5] NTIA Open and Transparent Process on Software Component Transparency, May 23, 2021 https://www.ntia.gov/SoftwareTransparency
[6] Software Bill of Materials, https://www.ntia.gov/sbom
[7] Roles and Benefits from SBOM Across the Supply Chain, https://www.ntia.gov/files/ntia/publications/ntia_sbom_use_cases_roles_benefits-nov2019.pdf
[8] CycloneDX, May 23, 2021 https://cyclonedx.org/tool-center/
[9] SPDX, May 23, 2021 https://github.com/spdx/tools
[10] Integrated Adaptive Cyber Defense https://www.iacdautomate.org/
[11] Comply to Connect, https://www.disa.mil/-/media/Files/DISA/Fact-Sheets/Comply_to_Connect_Fact_Sheet_050720.ashx
[12] OpenC2 https://openc2.org/
[13] Sharing and Exchanging SBOMs https://www.ntia.gov/files/ntia/publications/ntia_sbom_sharing_exchanging_sboms-10feb2021.pdf
[14] Experimenting with SBOM- lessons from the Healthcare sector. https://www.ntia.gov/files/ntia/publications/ntia_sbom_energy_healthcarepoc.pdf
[15] International Telecommunications Union Standardization Study Group 17 on Cybersecurity, https://www.itu.int/en/ITU-T/studygroups/2017-2020/17/Pages/default.aspx
[16] International Telecommunications Union Development Study Group 2, Question 3 on Cybersecurity, https://www.itu.int/net4/ITU-D/CDS/sg/rgqlist.asp?lg=1&sp=2018&rgq=D18-SG02-RGQ03.2&stg=2