Sharing and Exchanging SBOMs

NTIA Multistakeholder Process on Software Component Transparency
Framing Working Group
2021-02-10

Overview

Transparency in the supply chain enables better risk decision-making for both suppliers and users of software. This means that information about the underlying software components in a piece of software—a Software Bill of Material (SBOM)—should be accessible to the right entities at the right time. An SBOM identifies and lists software components used in building a piece of software, as well as information about those components and the relationships between them. An SBOM is effectively a nested inventory, or a list of ingredients that make up software components.[1]

Sharing SBOM data across the supply chain will involve a combination of technical platforms, predictable data formats, and operational processes. There will not be a single one-size-fits-all solution across the diverse needs of the entire software ecosystem, but modeling SBOM processes on existing approaches and methods will simplify this process and minimize the amount of new tools and processes needed for better supply chain management.

For more background on SBOM, please see the NTIA Software Component Transparency document library.[2]

The terms in this document reflect the definitions in Framing Software Transparency[3] unless otherwise noted.

Goal

The goal of this document is to provide a small set of SBOM discovery and access options as architectural building blocks to allow flexibility in different use cases and applications. An important element of this goal is to minimize the burden on diverse authors and consumers of SBOM data. SBOMs may be delivered in any number of ways based on standardized automated exchange mechanisms or through mechanisms specified by prearranged agreement such as a contract. This overview focuses primarily on standardized approaches.

This document deals with getting the SBOM from the upstream author to the downstream consumer of the SBOM. It does not deal with creating the SBOM (see *Framing Software*

[1] For more information about the basics of a Software Bill of Materials, see www.ntia.gov/SBOM
[2] https://www.ntia.gov/sbom
[3] https://www.ntia.doc.gov/files/ntia/publications/ntia_naming_use_cases_-_framing_2020-04-11.pdf

Component Transparency: Establishing a Common Software Bill of Material (SBOM)[4]) nor does it deal with how the receiver uses the SBOM or the integrity of the SBOM data (see *Roles and Benefits for SBOM Across the Supply Chain).*[5]

[4] https://www.ntia.gov/files/ntia/publications/framingsbom_20191112.pdf
[5] https://www.ntia.gov/files/ntia/publications/ntia_sbom_use_cases_roles_benefits-nov2019.pdf

Terminology

To best understand how to interpret the information provided in this document, it is important to understand the terms and how they are used in context. See Section 5 of *Framing Software Component Transparency: Establishing a Common Software Bill of Material (SBOM)*[2] for a full list of terms and their definitions. Table 1 defines some additional terms.

Term	Meaning
Author	entity that creates an SBOM The Author is the creator of the SBOM, ideally but not necessarily the supplier, even if there are several intermediaries (e.g., mid-chain suppliers) involved.
Consumer	entity that receives the transferred SBOM This role is sometimes called a “receiver”, “operator” or “leaf entity.”
Upstream	towards the supplier For example, an image processing library is upstream of a photo editing application.
Downstream	towards the consumer From the previous example, the photo editing application is downstream of the image processing library.
Intermediate supplier	suppliers who make use of upstream components that the suppliers combine into new components delivered downstream Many suppliers are intermediate suppliers.
Discovery	the mechanism used by the consumer to know the SBOM exists and how to access it
Access	transfer of the SBOM using the method derived from discovery

Table 1: General terminology

Getting SBOM data to the right people at the right time consists first of knowing the SBOM exists and how to access it. “Discovery” is used to describe the mechanism used by the consumer to know the SBOM exists and how to access it. One key goal is automated SBOM discovery. The term “advertisement” is used to mean the mechanism by which the author makes known how the consumer may access the SBOM, e.g., either through a well-known location or through an announcement of some form.

Advertisement and Discovery

Advertisement is how software or a device informs consumers that an SBOM is available and discovery is how the consumer learns of the location of the SBOM. These are commonly used, but not required.

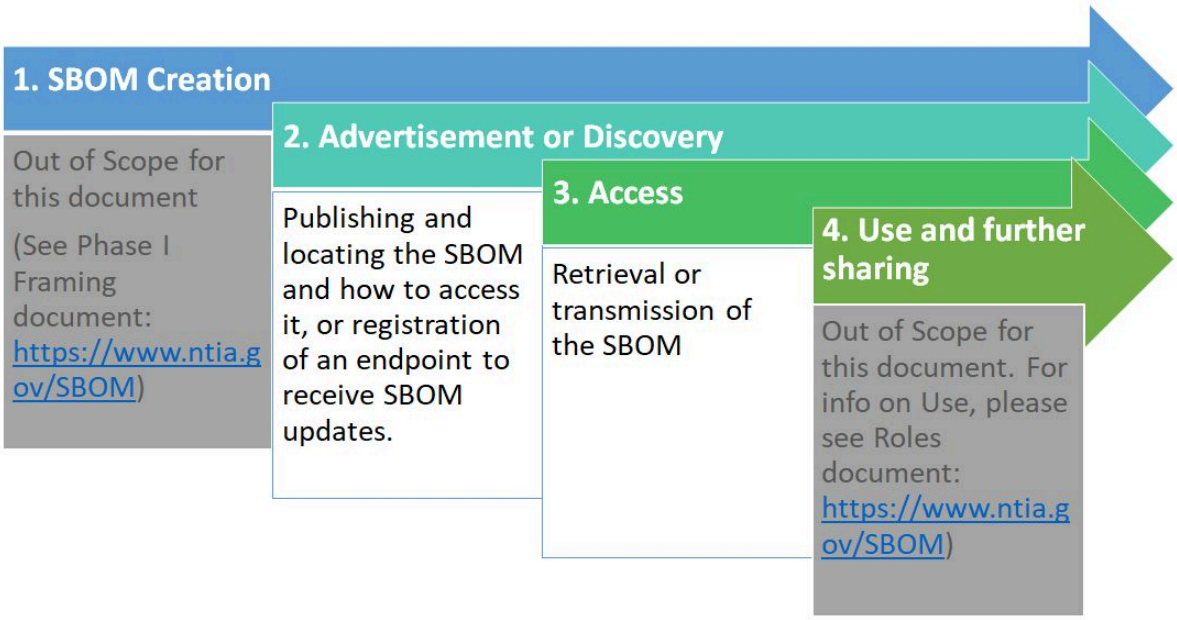

Figure 1: Conceptual SBOM exchange

Example 1: URL

When an SBOM is to be used in an operational deployment, it might be included as a URL in product literature or packaging, or as part of a Manufacturer Usage Description (MUD) [RFC 8520].[6] MUD provides a means for devices to describe their capabilities and needs for deployments. A MUD extension for SBOM provides a choice for one or more ways to retrieve the SBOM. An SBOM could also be searched for in a search engine.

[6] https://datatracker.ietf.org/doc/draft-ietf-opsawg-sbom-sharing/

An SBOM could also be retrieved from the device itself, via a ".well-known" URL [RFC 8615]. This would require a short RFC to define the desired URL, and possibly to register MIME types for any SBOM formats that do not yet have a definition.

Example 2: Manifest

When an SBOM is to be used by downstream developers, the software package could include a manifest in a well-known location. For instance, there are tools that can create an SBOM document in a top-level directory of a software repository[7]. Package management tools and containers have guidelines for the location of manifest information placement as well. This form is well suited for distributions for use by parties in the middle of the supply chain to indicate licensing requirements and package contents.

Example 3: Publish/Subscribe System

Another means to share SBOMs is via a publish/subscribe system. In this case, a consumer would subscribe to a supplier service for updates that would be published. An example of this would be a shared channel established between supply chain partners where SBOMs are published by upstream authors.

Access

Access is the transfer of the SBOM using the method derived from discovery. The input to this process is the location and access method to retrieve the SBOM. Several transfer mechanisms will be discussed under different scenarios depending on where the SBOM resides (note these are not mutually exclusive, and there are other ways to receive an SBOM):

- Method 1: SBOM is provided directly to the receiver through email or similar informal communication mechanisms that are determined by pre-arrangement between the SBOM supplier and downstream consumers.
- Method 2: SBOM is resident on the device executing the software the SBOM describes.
- Method 3: SBOM resides on a repository available to software consumers.

Method 1: SBOM is provided directly to the receiver using email or similar "out of band" mechanism

This would be used in cases where the manufacturer has the SBOM but no automated infrastructure for sharing it. This is a less preferred method because it is not easily amenable to automation (a goal of this project).

Method 2: SBOM is resident on the device executing the software the SBOM describes

[7] https://reuse.software/faq/#bill-of-materials

When the SBOM co-resides on a device, it can be retrieved using one of a number of protocols, such as HTTP, Constrained Application Protocol (CoAP) or an OpenC2 binding. This is useful in cases of highly tailored systems that have the ability to expose an API. In the case of HTTP, CoAP and their secure variants, the SBOM would be found at a well-known location, such as /.well-known/sbom (as described above). Such names are unique to each origin HTTP or COAP service. OpenC2 has a number of protocol bindings, such as HTTP/S, Message Queuing Telemetry Transport (MQTT), OpenDxl, and OpenDDS. OpenC2 super-imposes its security model on those bindings. See the OpenC2 FAQ for more information.

Method 3: SBOM resides on a repository available to software consumers

An SBOM could be published to a public or private website, database, or other shared repository available to consumers of the associated software. One method is to publish SBOMs on Digital Bill of Materials (DBoM) Channel repositories. An author may publish an SBOM on a DBoM Channel that they or others create to make it available to all or only some parties based on the policy of that channel. See the DBoM Project GitHub for more info.

When the SBOM is on a web site (be it published on the web or through a customer portal), the SBOM is retrieved using HTTP over Transport Layer Security (HTTPS). This is useful in the case of small or legacy systems that have no APIs to transmit SBOMs, or when a software package is being included by a developer, and the corresponding SBOM is to be included downstream. Authors may leverage the security model of HTTPS to test for entitlement or otherwise limit distribution as they see fit. Automated tooling must take care to identify portal requirements, and may need to alert the administrator of any registration requirements.

The repository could be an offering or a function of a supply chain consortium. Some complex supply chains require quantified trustworthiness from their participants, including specific qualification, scoring, and tracking over time. To achieve these goals, the participants contribute their trust data to a common database that is used for trust verification by other supply chain partners. A wide variety of data types contribute to a supplier's trustworthiness, and almost anything could be included: quality test results, regulatory certification, financial health, etc. SBOM is a natural extension to both the database and the requirements for trust.

Status Of This and Future Work

While SBOMs can be shared now, the NTIA Software Component Transparency Framing Working Group continues to work to improve their applicability to various market segments, associated processes, and tooling.