METI
Ministry of Economy, Trade and Industry

Guidance on Introduction of Software Bill of Materials (SBOM) for Software Management ver. 2.0

Summary

August 29, 2024

Cybersecurity Division

Commerce and Information Policy Bureau

Ministry of Economy, Trade and Industry

Summary of Revision

- To promote the distribution of secure software, the Ministry of Economy, Trade and Industry (METI) is **promoting the use of Software Bill of Materials (SBOM), an inventory of software components, for use by companies**.

- On July 28, 2023, METI published a guidance titled "**Guidance on Introduction of (SBOM) for Software Management ver. 1.0**," which outlines **the benefits of adopting SBOM for companies and key points to consider during implementation**.

- Recently, **this guidance has been revised to demonstrate ways for all companies, to utilize SBOM more effectively**, including small and medium-sized enterprises.

[Key Points of Revision]

➢ **Specification of vulnerability management process** which outlines specific procedures and concepts for effectively using SBOM to manage software vulnerabilities.

➢ **"SBOM Compliance Model"** is a framework to determine scope of SBOM implementation considering effectiveness and cost.

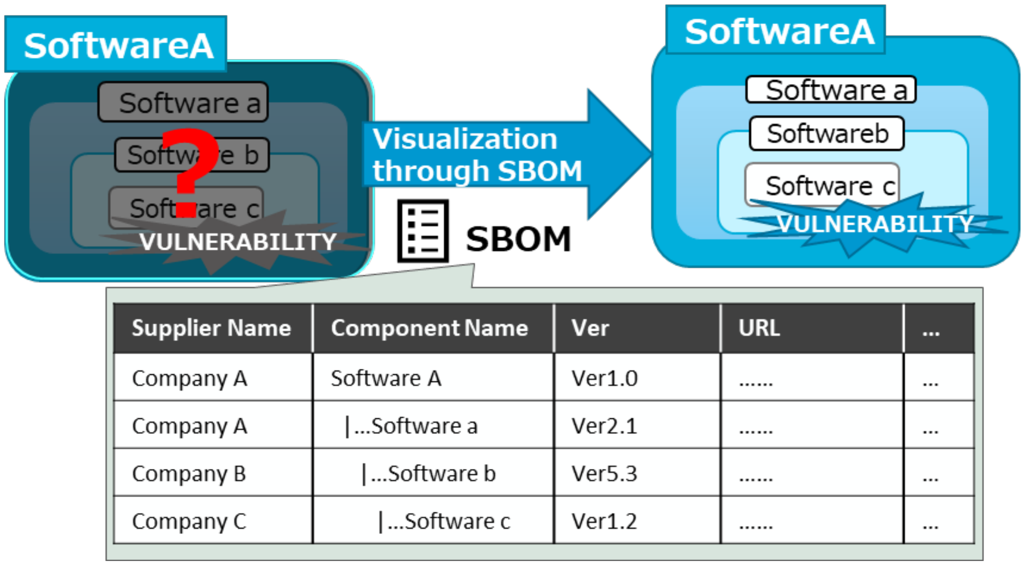
➢ **"SBOM Contract Model"** specifies items (requirements, responsibilities, cost burdens, rights, etc.) regarding SBOM that should be stipulated in contracts with contractors, etc.

SBOM as a Means of Ensuring Software Security

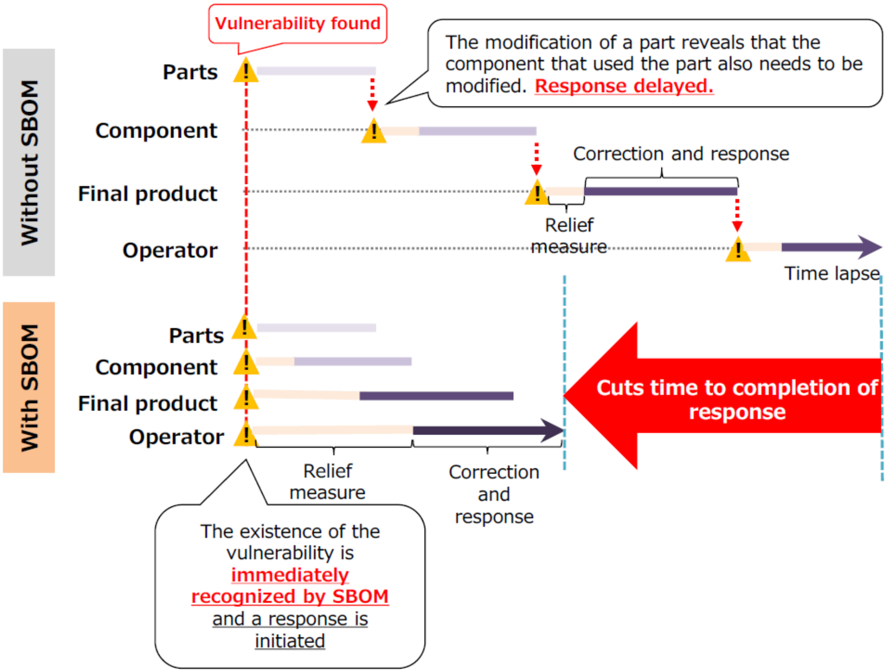
- SBOM (Software Bill of Materials) refers to **a list of software components** that indicates **who created each component, what is included, and how the components are structured**.

- By utilizing SBOM, it is possible to gain a comprehensive understanding of the software's composition, enabling the immediate identification of vulnerability information and **facilitating responses to such vulnerabilities**. However, there are challenges related to the effectiveness and cost of creating SBOM. To verify the feasibility of SBOM, METI conducted practical proof-of-concepts (PoC).

- In July 2023, METI published **“Guidance on Introduction of Software Bill of Materials (SBOM) for Software Management ver. 1.0,**“ which provides basic information about SBOM and outlines key points for its implementation.



Supplier Name	Component Name	Ver	URL	...
Company A	Software A	Ver1.0	......	...
Company A	|…Software a	Ver2.1	......	...
Company B	|…Software b	Ver5.3	......	...
Company C	|…Software c	Ver1.2	......	...



The modification of a part reveals that the component that used the part also needs to be modified. Response delayed.

The existence of the vulnerability is immediately recognized by SBOM and a response is initiated

Guidance on Introduction of Software Bill of Materials (SBOM) for Software Management

Background and Purpose

- As software supply chains become more complex and the use of open source software (OSS) becomes more common, vulnerability management and license management in software are becoming increasingly important.
- As a method of software management, a management method using the Software Bill of Materials (SBOM) has been attracting attention.
- Through the Proof-of-Concept (PoC) in several industrial fields, it was confirmed that SBOMs can be used for efficient software management, but it also became clear that various hurdles exist in the actual implementation of SBOMs.
- This guidance provides basic information about SBOM, as well as **myths and facts about SBOMs**. In order to assist companies in introducing SBOM, it also presents **actions for the SBOM introduction and key points to be aware of when introducing an SBOM**. (ver. 1.0)
- In addition, the following items are provided as reference examples: **specific procedures and concepts for effectively utilizing SBOM in** a series of processes to manage software vulnerabilities; a **framework for considering the appropriate scope of SBOM implementation in** light of its effectiveness and cost; and items that should be stipulated in the contract between **a procurer and a supplier regarding SBOM (requirements, responsibilities, cost burden, rights**, etc.) when placing an order for and receiving software. **Reference examples are given for items (requirements, responsibilities, cost burdens, rights, etc.) that should be stipulated in contracts regarding SBOM between a procurer and a supplier in the receipt and placement of orders for software**. (ver. 2.0)

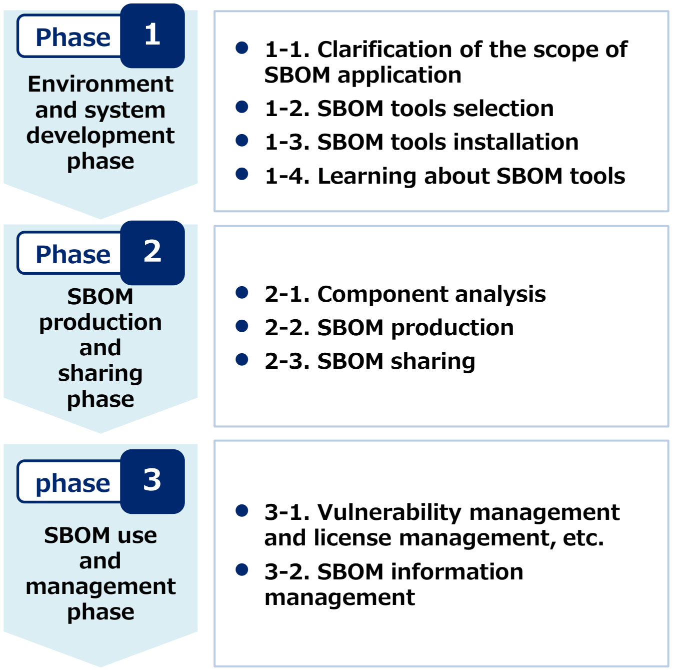
Target Audience

- Mainly software suppliers dealing with packaged software and embedded software.
  - ✓ Software development departments
  - ✓ Product security departments (PSIRT)
  - ✓ Executives/Management
  - ✓ Legal and intellectual property departments

Benefits of SBOM

- **Vulnerability management**
  - ✓ Reduce residual vulnerability risk
  - ✓ Reduce vulnerability response time
  - ✓ Reduce cost of vulnerability management
- **License management**
  - ✓ Reduce risk of license violations
  - ✓ Reduce cost of license management
- **Development productivity**
  - ✓ Prevent development delays
  - ✓ Reduce development costs

Process for SBOM implementation (ver. 1.0)



Phase 1 Environment and system development phase

- **1-1. Clarification of the scope of SBOM application**
- **1-2. SBOM tools selection**
- **1-3. SBOM tools installation**
- **1-4. Learning about SBOM tools**

Phase 2 SBOM production and sharing phase

- **2-1. Component analysis**
- **2-2. SBOM production**
- **2-3. SBOM sharing**

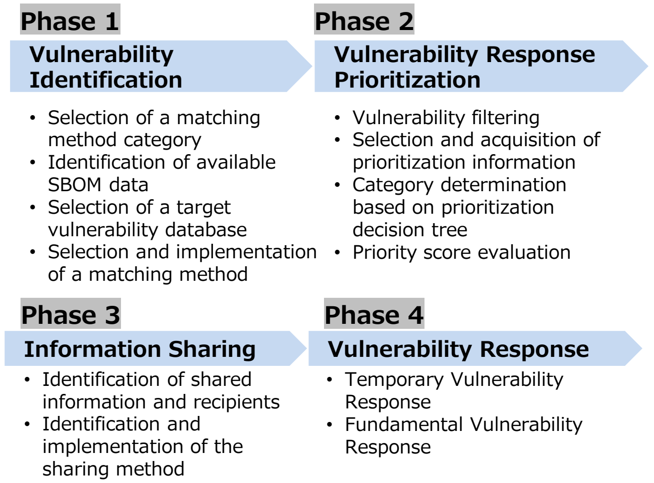
phase 3 SBOM use and management phase

- **3-1. Vulnerability management and license management, etc.**
- **3-2. SBOM information management**

Embodying vulnerability management process (ver. 2.0)

- **The phases related to vulnerability management are particularly important in the process of using SBOM,** as the effect of using SBOM is expected to be the reduction of vulnerability risk through software vulnerability management.
- **By summarizing specific procedures and concepts for** effectively utilizing SBOM in a series of vulnerability management processes, the guidance provides **reference** information to **increase the effectiveness of SBOM utilization**.

Vulnerability management process using SBOM



Phase 1
Vulnerability Identification

- Selection of a matching method category
- Identification of available SBOM data
- Selection of a target vulnerability database
- Selection and implementation of a matching method

Phase 2
Vulnerability Response Prioritization

- Vulnerability filtering
- Selection and acquisition of prioritization information
- Category determination based on prioritization decision tree
- Priority score evaluation

Phase 3
Information Sharing

- Identification of shared information and recipients
- Identification and implementation of the sharing method

Phase 4
Vulnerability Response

- Temporary Vulnerability Response
- Fundamental Vulnerability Response

SBOM Compliance Model (ver. 2.0)

- A **framework (systematized to cover the 5W1H (Five Ws and How)) for considering the scope of appropriate** SBOM implementation**,** taking into account the effectiveness and cost of SBOM implementation.
- Through the PoC, reference examples of a reasonable scope of response in the fields of **medical devices, automobiles, software products, etc.,** in consideration of cost and effectiveness, will be presented.
- By using this framework, it is expected that software that can be highly managed, i.e., secure software, will be appropriately evaluated by the market and its distribution will be promoted.

SBOM Contract Model (ver. 2.0)

- Reference examples of **items (requirements, responsibilities, cost burdens, rights, etc.) that should be stipulated in the contract** between the procurer and the supplier with respect to SBOM when placing and receiving orders for software components are shown below.
- The model agreement is to be combined with an existing model agreement for software to **provide suggested items for drafting an SBOM-compliant agreement**.

Reference

Guidance on Introduction of Software Bill of Materials (SBOM) for Software Management

ver. 1.0

Background and Purpose

- As software supply chains become more complex and the use of open source software (OSS) becomes more common, vulnerability management and license management in software are becoming increasingly important.
- As a method of software management, a management method using the Software Bill of Materials (SBOM) has been attracting attention.
- Through the Proof-of-Concept (PoC) in several industrial fields, it was confirmed that SBOMs can be used for efficient software management, but it also became clear that various hurdles exist in the actual implementation of SBOMs.
- This Guidance **provides basic information about SBOM, as well as myths and facts about SBOMs**. In order to assist companies in introducing SBOM, it also **presents actions for the SBOM introduction and key points to be aware of when introducing an SBOM.**

Target Audience

- Mainly software suppliers dealing with packaged software and embedded software*.
  - ✓ Software development departments
  - ✓ Product security departments (PSIRT)
  - ✓ Executives/Management
  - ✓ Legal and intellectual property departments

※ In particular,

- Organizations facing challenges in software vulnerability management
- Organizations that have heard of SBOM but do not understand the specifics and benefits of SBOM
- Organizations that understand the need for SBOM but do not know what they need to do to implement it.

Benefits of SBOM

- **Vulnerability management**
  - ✓ Reduce residual vulnerability risk
  - ✓ Reduce vulnerability response time
  - ✓ Reduce cost of vulnerability management
- **License management**
  - ✓ Reduce risk of license violations
  - ✓ Reduce cost of license management
- **Development productivity**
  - ✓ Prevent development delays
  - ✓ Reduce development costs

SBOM Introduction Process

Phase 1 Environment and system development phase

- **1-1. Clarification the scope of the SBOM application**
  - ✓ Organize information (languages, development tools, configuration diagrams, etc.) related to the target software.
  - ✓ Clarify the scope of the SBOM application based on the organized information.
- **1-2. SBOM tools selection**
  - ✓ Organize the viewpoints for the selection of SBOM tools and evaluate and select tools based on the viewpoints. (Examples of viewpoints: functions, performance, analyzable information, analyzable data format, cost, supported formats, component analysis method, support systems, coordination with other tools, etc.)
- **1-3. SBOM tools installation**
  - ✓ Install and configure the SBOM tool by reviewing the tool's instruction manual, README file, etc.
- **1-4. Learning about SBOM tools**
  - ✓ Learn how to use the SBOM tool by reviewing the tool's instruction manual, README file, etc.
  - ✓ Record know-how on using the tool and an overview of each function and share them within the organization.

Phase 2 SBOM production and sharing phase

- **2-1. Component analysis**
  - ✓ Scan the target software by using SBOM tools and analyze component information. Based on the result, check if there are any false positives and false negatives.
  - ✓ By using SBOM tools, component analysis and SBOM production can be performed more efficiently than would be possible manually.
  - ✓ Using a package manager may enable the identification of granular components that cannot be identified by SBOM tools.
- **2-2. SBOM production**
  - ✓ Determine the requirements such as items, format, and output file format to be created and create an SBOM that fulfills the requirements.
- **2-3. SBOM sharing**
  - ✓ Consider the most appropriate means of sharing the SBOM with users and/or suppliers and share the SBOM according to this method if necessary.

phase 3 SBOM use and management phase

- **3-1. Vulnerability management and license management, etc.**
  - ✓ Based on the SBOM tool output on vulnerabilities, conduct vulnerability management such as evaluating severity, assessing impact, fixing vulnerabilities, confirming residual risks, and providing information to relevant organizations.
  - ✓ Based on the SBOM tool output on licenses, confirm whether any OSS license violations have occurred.
- **3-2. SBOM information management**
  - ✓ Appropriately manage information contained in SBOM along with SBOM itself.
    *It is an effective practice to have a department equivalent to a PSIRT to manage SBOM.
  - ✓ SBOM tools that automatically update and provide notification on vulnerability information enable immediate identification of information about new vulnerabilities. If automatic management using the tool is not possible, the operation should be covered by other methods, including designating a person in charge to manage this.

ver. 2.0

Specification of vulnerability management process

Background and Objectives

- Provide specific process-based examples of methods and procedures for vulnerability management using SBOM.
- There are **currently unresolved** issues in vulnerability management using SBOM. New **technology development, standardization, and tool environment development** will be necessary to fully resolve these issues.
- In this chapter, we present the concept and the best practices that are currently possible in order to avoid the issues by operations on the SBOM user side, including those issues.

Major issues and solution approaches and know-how

Issue	Solution approaches and know-how
Failure in matching with vulnerability DB due to coexistence of part IDs	Use of ID conversion tools such as Purl2cpe and partial ID matching using APIs
Ensuring coverage of diverse vulnerability databases	Presentation of vulnerability DB selection method based on risk and cost reduction benefits
Prioritizing vulnerability responses for rapid response and efficiency	Presentation of a prioritization category decision method using a decision tree based on SSVC
Information sharing and role-sharing throughout the supply chain	Presentation of the steps for information sharing and implementation items by developers and users
Division of roles in vulnerability response	Presentation of items to be implemented by developers and users in provisional and full-scale response

Vulnerability Management Process and Step using SBOM (Overview)

Phase 1 Vulnerability Identification

1. **Selection of a matching method category**
   Organizations should choose a method from (i) the Use of existing SBOM tools, (ii) API utilization scripts, and (iii) Web UI based on their technical capabilities, budget, and available resources.
2. **Identification of available SBOM data**
   Determine how to obtain the SBOMs that will be utilized.
3. **Selection of a target vulnerability database**
   Choose the vulnerability databases that will be used for vulnerability identification and prioritization of responses.
4. **Selection and implementation of a matching method**
   Based on the results from Steps 1 to 3, decide on the organization’s specific method for vulnerability identification.

Phase 2 Vulnerability Response Prioritization

1. **Vulnerability filtering**
   Narrow down vulnerability information that needs to be addressed from known information prior to prioritization using external information.
2. **Selection and acquisition of prioritization information**
   Based on the components of risk, select and acquire the necessary information according to the company's policy, such as the existence of incidents, Exploit code distribution status, CVSS, and VEX information.
3. **Category determination based on prioritization decision tree**
   Prioritize category judgments according to (developer, user organization) x (high/low technical capability) according to the judgment tree organized based on SSVC.
4. **Priority score evaluation**
   In parallel with steps 1 through 3, perform detailed prioritization by prioritizing within categories by quantitative scoring as needed.

Phase 3 Information Sharing

1. **Identification of shared information and recipients**
   - Identify shared information such as vulnerability information, load information, and prioritization results.
   - Identify sharing parties and the order of sharing, such as social organizations, external parties (users, vendors, suppliers), etc.
   - Identify sharing triggers such as triggers for sharing (push type/pull type).
2. **Identification and implementation of the sharing method**
   - Identify the sharing method for file sending/receiving, SaaS, etc.
   - Specify access privileges: Specify the private and disclosure scope, privileges, etc., depending on the confidentiality.
   - Implement sharing based on the determined sharing method and access privileges.

Phase 4 Vulnerability Response

1. **Temporary Vulnerability Response**
   - Establish temporary measures, such as such as suspension of use, reduction, and workaround.
   - Apply the temporary measures, after informing stakeholders of the temporary measures decided upon.
2. **Fundamental Vulnerability Response**
   - Implement a fundamental response by identifying the developer of the software involved in the vulnerability and having the developer fix the vulnerability.
   - Update SBOM/VEX information upon fixing vulnerabilities.
   - Share updated SBOM/VEX with the supply destination and conduct SBOM history management as necessary.

Overview of the SBOM Compliance Model

ver. 2.0

Key components of SBOM Compliance Model

- The options for creating and utilizing SBOMs have been systematically organized to cover the 5W1H (Five Ws and How) aspects regarding items that significantly impact cost and effectiveness. The SBOM-related items have been organized to reflect the opinions of experts and the findings from PoC.
- Through these PoC, reference examples of reasonable scope of response have been provided, considering cost and effectiveness in fields such as medical devices, automobiles, and software products.

Selection options of SBOM compliance item (Draft)

	Application category	Main applicable items (options)	Cost	Reasons for determining cost categories (main cost elements, assumptions, etc.)
Generate & Share	(a) SBOM Creating Entity (Who)	(a1) In-house	Low	Identify components directly used in in-house development from configuration files, etc., and generate SBOM. Including code modification components.
		(a2) Supplier (development contractor) transaction contract	Medium	Generate SBOMs for parts to be used in the software of contracted development companies with whom we have business contracts.
		(a3) No supplier (third party) transaction contract	High	SBOMs are created by OSS and off-the-shelf component vendors that are unable to make SBOMs a requirement through transaction agreements. (b2)(c2)
	(b) Part Scope (What, Where)	(b1) Directly used component*1	Low	The developer identifies the components to be used directly by the developer from configuration files, etc., and generates the SBOM with tools, etc.
		(b2) Indirectly used component*2	High	For third-party parts, generate SBOMs for recursively used parts.
	(c) Means of generation (scrutiny) (How)	(c1) Manually identified (using configuration management information), generated by tool	Low	Create component information to be used directly using configuration files, etc.
		(c2) No tool to identify, generate, or scrutinize false detection	Medium	Tools will be used to generate the SBOM and scrutiny will be omitted. The use of tools is assumed to be mainly for generating SBOMs for recursive parts, so commercial tools are assumed to be used.
		(c3) Tools used to identify, generate, and scrutinize false detection	High	Generate SBOMs using commercial tools, perform source code review, and scrutinize for false positives and omissions. (including recursive use components)
		(c4) The party commissioning the development independently inspects the SBOM created by the development contractor	High	When a development contractor accepts an SBOM created by a development contractor, the reliability of the SBOM is inspected by creating the SBOM independently with a tool or other means.
	(c') Generation method (parts detection method)	Dependency analysis	Medium	Static analysis of configuration information such as package manager.
		File matching	Medium	Detect file-by-file matches of source code using hash values. This includes detection of OSS libraries.
		Snippet analysis	High	Detect by partial string matching or similarity in the source code.
		Binary analysis	High	Similarity detection based on bit patterns in binary files.
		Recursive dependency analysis within executable files	High	For libraries already linked within the executable, recursively perform dependency analysis when building the library.
		The above are not supported	High	Convert pre-recognized parts to SBOM.
	(c'') Generation method (target software type)	Components determined during development	Low	Static libraries, applications
		Components determined at runtime	Medium	Runtime libraries, services (local, external cloud), OS, middleware, execution environment (container, VM, AP server)
		Surrounding tool environment	High	Tools used in development operations (installers, updaters, distribution packages, development environments, tool chains, SBOM tools)
	(d) Data Format/Item (What)	(d1) Standard formats (SPDX, SPDXLite, etc.)	Medium	Create in a standard format such as SPDX.
		(d2) Includes the minimum elements of data fields as specified in the executive order.	Medium	Create an SBOM containing the minimum elements of the data fields in the Executive Order.
		(d3) Elements that do not satisfy the above	Low	Create your own minimum elements.
Utilization	(e) Scope of Utilization (Why)	(e1) Identification of vulnerabilities	Low	Search and identify vulnerabilities in DBs such as NVD, JVN, etc.
		(e2) Assessment of vulnerability severity	Medium	Evaluate severity based on CVSS values and set priorities for vulnerability response.
		(e3) Evaluation and mitigation of exploitability of vulnerabilities	Medium	Evaluate thepossibility of exploitation and the necessity of vulnerability countermeasuresusing VEX information, etc. Issue advisories on countermeasures, etc., as necessary.
		(e4) License identification	Medium	Identify the license and obtain the terms and conditions.
	(f) Conjugated entity (Who)	(f1) Product users	Low	If a vulnerability is identified, the use of the system is suspended and the company waits for the vendor to fix it. The damage is significant if business interruption costs are taken into account.
		(f2) Final product vendor	Medium	Notify users of vulnerabilities, request developers to correct them, and provide corrected builds and users. Report to authorities, ISAC, etc. as necessary.
		(f3) Developers of each component	High	The developer shall monitor and correct the vulnerabilities and provide the procurer with a corrected version. Report to the authorities, ISAC, etc. as necessary.

ver. 2.0

Overview of the SBOM Contract Model

Key components of SBOM Contract Model (to be specified in the contract)

- As items to be stipulated in the contract, requirements related to SBOM, responsibilities, cost burdens, and rights have been organized into distinct categories. This organization aims to encompass industry trading practices and opinions from task forces. Important requirements for vulnerability management and software quality assurance have been articulated. It is primarily assumed that the focus will be on contracts following the requirements definition phase.

Category		Matters to be stipulated	Level
SBOM Requirements	Format Standard	**(SBOM format)*1** Specify the SBOM standard format to be adopted. (Specify standards and versions of SPDX, CycloneDX, SWID, etc.)	Basic
		(ID standard)*1 Specify the part ID standard to be adopted. (CPE, PURL, SWD, proprietary format, etc.)	Basic
		(SBOM minimum elements)*1 Specify the minimum element among the element items of the SBOM format to be adopted, referring to the minimum element of the SBOM of NTIA.	Basic
	Quality and Reliability (Applicable to SBOM-compliant models)	**(Supplier Contract Forms Covered)** As the scope of SBOM creation, the scope by contract form of contract development agreement and third party terms and conditions (commercial off-the-shelf products, OSS) shall be specified.	Basic
		(Recursive use parts)*1 Specify whether direct use parts or recursive indirect use parts are included in the scope of SBOM creation.	Advanced
		(Scope of application of the composition analysis method)*1 For indirect use parts, the scope of application of the composition analysis method used to identify the parts is specified. (Dependency analysis, file matching, snippet analysis, etc.)	Advanced
		(Necessity of parts scrutiny)*1 Specifies whether or not manual scrutiny of false positives and omissions is required for the results of parts identification by the tool.	Advanced
		(Target phase of the component)*1 Specify the scope of the part information, such as build time, run time, cloud services, etc.	Advanced
		(Prior Agreement for Third Party Parts) When using third-party components (commercial components, OSS), this section defines whether or not prior declaration and agreement are required.	Basic
	Maintenance and Operation	**(Sharing method)*1** This section defines real-time sharing by transfer by SBOM file or by SaaS, etc.	Basic
		(VEX support)*1 Specify whether to provide VEX information based on exploitability for vulnerability information related to SBOM.	Advanced
		(SBOM update)*1 Defines the deadline and frequency of updating the SBOM in response to software updates, SBOM defect fixes, etc.	Basic
		(Vulnerability Monitoring and Notification) During the operational phase of the software, monitor for vulnerabilities and stipulate a deadline for notification to the procurer when vulnerabilities are discovered.	Advanced
		(Vulnerability Response and Prioritization)*1 Specify whether or not information is to be provided to procurers regarding the need for vulnerability response and prioritization (triage) when vulnerabilities are discovered.	Advanced
		(EOL and EOS) This section defines the EOL and EOS for third party parts and contracted development parts and the notification of changes to their deadlines.	Advanced
Liability and Warranty		**(Submission of Evidence)** Specifies whether or not to require submission of evidence and third-party certification to prove conformity with SBOM requirements.	Advanced
		(Contract Nonconformity Liability) When nonconformity to SBOM requirements is found, it defines the necessity of defects response such as SBOM correction.	Basic
		(Compensation for damages)*2 Provide for the maximum amount of damages, etc., in the event of an accident caused by nonconformity with SBOM requirements. Includes damages for license violation.	Basic
		(Indemnification) For cases where evidence of conformance to SBOM requirements has been submitted, this section defines the limitations and disclaimers of liability for damages in the event that damages occur due to reasons attributable to technical limitations (e.g., false detection of tools).	Advanced
Cost Burden		**(Quotation)*2** Prepare an estimate based on SBOM requirements, responsibilities and warranties, and stipulate the payment of consideration based on the agreed amount.	Basic
Rights and Confidentiality		**(Attribution of Intellectual Property Rights)** This section defines the intellectual property rights of the created SBOM, the ownership of the right to use the SBOM, and whether or not the SBOM can be provided to a third party.	Advanced
		(Confidentiality) This section defines the confidentiality and management of the SBOM and the prohibition of reverse engineering using the SBOM.	Advanced

Legend:

Basic	Minimum expectations common to the field
Advanced	Expectations in specific fields and high requirement levels

*1 It is assumed that this information is included in the ordering specifications.
*2 It is assumed that the contract is common to software development contracts in general.