Pharmaceutical Affairs Bureau Notification No. 0825-1

August 25, 2020

To the Directors of Health Departments (Bureaus) in Each Prefecture

Director, Medical Devices Review and Management Division, Pharmaceutical and Medical Devices Bureau, Ministry of Hea

(official seal omitted)

Guidelines for the Introduction and Operation of SBOM in Medical Devices (Version 1)

A Software Bill of Materials (SBOM) visualizes the software components that make up a medical device, facilitating the identification and management of vulnerabilities. It is a useful method for ensuring the cybersecurity of medical devices and is increasingly being used both domestically and internationally.

This time, in the "FY2025 Project for Promoting Cybersecurity Response for Medical Devices," we have compiled the "Guidelines for the Introduction and Operation of SBOM in Medical Devices (Version 1)" as attached. Therefore, we request that you inform relevant businesses and other entities under your jurisdiction to take necessary measures, referring to these guidelines, in order to ensure the cybersecurity of medical devices.

Please note that copies of this notice will be sent to the President of the Pharmaceuticals and Medical Devices Agency, the Chairman of the Japan Federation of Medical Devices Manufacturers Associations, the Chairman of the Medical Devices and IVD Manufacturers Association of the United States, the Chairman of the Medical Devices and IVD Committee of the European Business Association, the Chairman of the Japan Clinical Laboratory Pharmaceutical Association, and the Representative Secretary-General of the Council of Registered and Certified Bodies under the Pharmaceuticals and Medical Devices Act.

Attachment

In medical devices

SBOM Implementation and Operation Guidelines (Version 1)

Project to Promote Cybersecurity for Medical Devices in FY2025

---

table of contents

1. Background and Objectives

1.1. Background

In recent years, with the increasing sophistication of software functions in medical devices and the expansion of network connectivity, the importance of cybersecurity measures has grown even further.
On the other hand, many manufacturers and distributors do not have a sufficient understanding of software components or manage update histories adequately, and there are many cases where they are unable to respond quickly and accurately to incidents or technical inquiries from medical institutions.

In terms of regulations,Act on Securing Quality, Efficacy and Safety of Pharmaceuticals, Medical Devices, etc."41Article No.3 Standards for medical devices determined by the Minister of Health, Labour and Welfare pursuant to the provisions of the Article (Heisei17Ministry of Health, Labour and Welfare Notification No.122No. (Hereinafter referred to as "Basic Requirements Standards.")12Article No.3items and the United StatesFD&CLaw No.524BAuthorities both domestically and internationally are strengthening regulatory requirements regarding cybersecurity, such as the Bill of Materials (BOM)Software Bill of Materials,below"SBOM" he said. The use of this technology in risk management is becoming international. It is becoming a trend. Also,10Some medical devices have been in use for over a year, and open-source software (OSS) or ready-made (Off-The-Shelf,below"OTS" he said.) Software including software Neglecting the lifecycle management of wearable components raises concerns that security risks may be left unaddressed for extended periods.

Medical professionals also want to "quickly find out which medical devices are affected and in what way" when an incident occurs or a vulnerability is disclosed,There have been complaints that "the manufacturer is not providing enough information." Therefore, manufacturers and distributors, etc.SBOMThere is a need for enhanced information sharing and accountability through these channels.

Note: Originally,OTSThe definition of software includesOSSThis includes the aspects of maintenance and management. Because they are treated differently, "OTS"of,OSSExcludingOTSUsed as a term to describe software. It is.

1.2. Purpose

Based on this background, these guidelines are intended for manufacturers and distributors.SBOMThe aim is to support manufacturers and distributors in gradually transitioning to a system that can incorporate this into their work and utilize it continuously.SBOMIf you are already using or implementing the system, this does not require any changes to your procedures or methods.

in particular,

- As the first step, the U.S. Department of Commerce's Telecommunications and Information Administration (National Telecommunications and Information Administration,below"NTIA" (says.)SBOMBased on the minimum elements, etc. SBOMWe have established a system that allows us to create and update these documents.SBOMCreating and managing changes
- after that,SBOMVulnerability management, license management,End of Life (EOL)/EOSExisting quality control systems such as management (Quality Management System,below"QMS" he said.) and post-market safety management program

This outlines a step-by-step storyline: integrating with SESS and utilizing it for risk management throughout the entire medical device lifecycle.

These guidelines do not require all companies to jump straight into advanced implementation, but rather emphasize providing a roadmap for "gradual introduction" that takes into account the maturity level of each organization.

1.3. Main target audience

The main target audience of these guidelines is the following individuals in charge at manufacturers and distributors:

- SBOMCompanies that are about to start developing
- alreadySBOMAlthough some companies are implementing it on a trial basis, they are facing challenges with the operational methods and organizational structure.

As for specific readers,

- Software Development Department
- quality assurance·QMSDepartment in charge
- Pharmaceutical Affairs and Regulatory Application Division
- PSIRTInformation Security Department
- Field Service and Maintenance Department

This position is intended for practitioners involved in software configuration management and cybersecurity measures for medical devices.

These guidelines aim to provide these departments with common background knowledge and guidance, and to address cross-organizational issues.SBOMThis is intended to serve as a common guideline for utilizing [the system/technology].

Furthermore, these guidelines are intended for healthcare institutions, medical information system providers, regulatory authorities, and other stakeholders involved in healthcare, and are intended to support manufacturers and distributors.SBOMIt is also intended to be used as reference material when understanding its positioning and operational image.

2. Overview of SBOM

2.1. What is SBOM?

SBOMThis refers to ensuring transparency in medical device software by defining the components that make up the software embedded in medical devices.OSS,OTSThis document systematically records a list of software (including in-house developed modules) and their attribute information. It serves as the basis for risk-based approaches such as vulnerability management, license management, and lifecycle management.

- MDS2andSBOMRelationship

In the medical fieldMDS2(Manufacturer Disclosure Statement for Medical Device SecurityAs a result, a framework for disclosing security functions and operational requirements on a product-by-product basis to medical institutions is widely used.

MDS2teethSBOMThis is not a document that specifies the format or content,SBOMThere are sections regarding provision,SBOMThis can be used as one practical means of informing medical institutions whether or not they offer this service and how they will provide it.

- In these guidelinesSBOMPositioning

The following guidelinesSBOMThis applies to all software included in medical devices and will be presented to medical institutions.SBOMRegarding this, it is also possible to present only the software for hardware units or components used for data communication, etc. However, in this case, partialSBOMIt is necessary to explicitly state that this is the case.

2.2. Effects of Introducing SBOM

SBOMThis is an information infrastructure that visualizes the software configuration included in medical devices, supporting vulnerability management, license management, and improved development productivity. Compared to conventional operations where configuration information is tracked manually, it is expected to significantly reduce the amount of time spent on investigation and the risk of oversights.

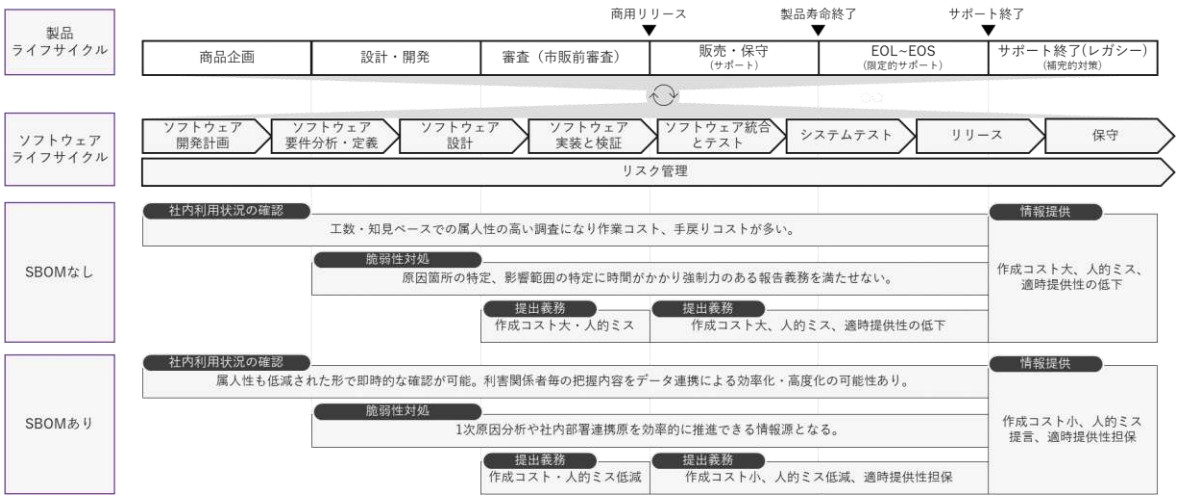


figure1:SBOMThe effects of introducing it

In line with the lifecycle of medical devicesSBOMThe following are some specific examples of its value:

- Checking internal usage: During internal evaluation and pilot implementation phases.SBOMBy using this tool, you can view a list of vulnerabilities and license violations, allowing you to identify problematic components early on.

- Vulnerability Response: New Vulnerability (CVEWhen) was made publicSBOMBy cross-referencing this information, potentially affected products and versions can be identified immediately, accelerating the development of patch plans. This also makes it easier to provide evidence-based reports to stakeholders.

- Information provided in response to inquiries from medical institutions and regulatory authorities.SBOMBased on this, it is possible to explain "which devices are affected by vulnerabilities, etc.," thereby reducing the time required for information provision and the reliance on individual personnel.MDS2This allows for the efficient delivery of customer-facing security documents, such as advisory information.

In this waySBOMThis is not merely documentation to meet regulatory requirements, but rather a mechanism to enhance both risk management and operational efficiency throughout the entire medical device lifecycle.

2.3. Minimum elements of SBOM

SBOMThe amount and type of information that should be included can vary depending on the purpose of use and the maturity level of the organization. In this guideline,1As the minimum elements of the edition,NTIAby"The Minimum Elements for a Software Bill of Materials (SBOM)Based on the "Guidelines for the Introduction of Cybersecurity for Medical Devices (No. 1)" issued by the Ministry of Health, Labour and Welfare2version) "Appendix to the (hereinafter referred to as the "Manual")AShown The smallest element (Table 1) is adopted.

Specifically, this includes the name of the software component supplier, the component name, the version, and other unique identifiers (e.g.,purl,CPE(etc.), relationships between components,SBOMThe creator, the timestamp, and the final recipient of the information provided to at least medical institutions, etc.SBOMThe basic principle is to include it. Although the component hash is listed as an optional element in the manual, it should be included whenever possible from the perspective of vulnerability management and integrity verification, which will be discussed later.

table1:SBOMThe minimum elements

element	Content
Software component supplier - Name	Create, define, or identify components. Entity
Name of software component	The supplier defines the software unit Name assigned to the net
Software component version	I identified the changes from the previous version. Identifiers used by suppliers
Other unique identifiers	Used to identify components or related database looka Identifier that functions as a key
Component hash	To identify the component binary Cryptographic hash used (optional)
relationship	Upstream componentsXis softwareY The software architecture that is included Information that characterizes the relationships on the texture
SBOMThe entity that created it	SBOMEntry creator
Timestamp	SBOMRecord of the date and time the data was aggregated. (※1)

*1: The timestamp is,ISO 8601Expressed in this format

Furthermore, the United StatesFDAThe guidance states that in pre-market applications for medical devicesNTIAMachine-readable according to baseline attributesSBOMIt is recommended to submit the following, as well as the support status of each software component (e.g., maintenance continuation/end of maintenance) and component support information (EOSWe understand lifecycle information such as support levels, etc., and cybersecurity throughout the lifecycle of medical devices.

—It is required to be used for security risk management.

Meanwhile, in the United StatesSBOMConsideration is underway regarding the items and standards, and in addition to the basic elements mentioned above, license information andSBOMInformation about the tools used to generate it, and at what development stage it was generated. SBOMThere is discussion about positioning information indicating whether something is happening (e.g., at build time/release time) as an element that should be added or enhanced.SBOMIt is desirable that this be as complete a list of components and dependencies as possible, and for parts where dependencies cannot be expressed or information cannot be obtained, this should be noted as "Known Unknowns (Whon UnknownsIt is recommended to explicitly state this as follows:(For example, if there are components provided externally in binary format and whose internal dependencies cannot be understood, this fact and the reason must be clearly stated.))

Thus, the first of these guidelines1The "minimum element" in the edition is,NTIABased on the minimum elements and the guidelines issued by the Ministry of Health, Labour and Welfare, and taking into account international trends, component support information (EOS This approach is based on the premise of gradually incorporating elements such as support levels and license information.SBOMWhile the required attributes and level of detail are expected to continue evolving, manufacturers and distributors should use the minimum elements outlined in this section as a starting point and expand on them with additional information as needed, depending on their own maturity level and product portfolio.

2.4. SBOM Format

SBOMData must be created and shared using machine-readable and interoperable formats. Using a common format streamlines management within an organization and facilitates cross-organizational communication. SBOMThis improves interoperability when sharing resources and contributes to increased transparency in the software supply chain.
Furthermore, the guidelines for implementing cybersecurity for medical devices state thatSWID(Software Identification Tag)tooSBOMAlthough it is mentioned as one of the formats used for the purpose, this guideline does not recommend it as the format for new implementations.SPDXandCycloneDXBased on this, with existing assetsSWIDIf using [a specific method], consider using it in combination or converting it as needed.

Recommended in the medical fieldSBOMThe format (machine-readable) is as follows:
SPDX(Software Package Data Exchange)2.2That's all.CycloneDX 1.6That's all.

however,SBOMIf it consists of multiple files,SBOMAttach a document explaining the structure (including folders).

SPDXThis allows for detailed recording of the software included in the product, including information necessary for legal and audit purposes, such as licenses, copyrights, sources, and dependencies.ISO/IEC 5962:2021It is an internationally standardized format. Its distinguishing feature is that it not only provides a simple list of components, but also systematically organizes elements necessary for compliance, such as copyright information and license conditions for each file, and the source of the components.

This makes it clear "where the software that makes up the product is obtained from and under what license it is being used," making it easier to meet the accountability and quality assurance requirements for medical devices. Furthermore, because it allows for detailed description, including the hierarchical structure of dependencies, it is also suitable for visualizing complex software configurations and for consistent management from development to maintenance.

CycloneDXIt is a security-focused tool designed to streamline vulnerability management and dependency assessment.SBOMIt is a format. The point that describes the software configuration isSPDXSimilar to other methods, but its distinguishing feature is its ability to handle elements directly related to cybersecurity practice, such as vulnerability information, exploitability information, and information on the operational and public access methods of services and external interfaces provided and used by the software, in a lightweight structure.
In particular, it indicates whether the vulnerability affects the product in question.VEX(Vulnerability Exploitability eXchangeThis facilitates collaboration with other organizations and allows for faster decision-making in the cybersecurity operations of medical devices.

The two have different objectives and strengths.SPDXIt focuses on legal compliance,CycloneDXThis approach specializes in visualizing and managing security risks. The choice between the two should be determined based on the priorities and usage scenarios of the manufacturer or distributor.SBOMWhen converting between different formats,SBOMIt is desirable to ensure that the minimum necessary elements are maintained, as well as that the elements required individually based on the regulatory requirements of each country and region are not omitted.

2.5. Misconceptions about SBOM

SBOMIn the initial stages of implementation, various misunderstandings often exist, which frequently hinder the decision to implement and the successful establishment of operational systems. This chapter will organize representative examples of misunderstandings particularly common in the medical device field and the factual circumstances surrounding them.SBOMThis section presents a framework for understanding and utilizing the information correctly.

・"SBOM"Publicizing this information increases the risk of attacks."

SBOMIt has been pointed out that this could "serve as a roadmap for attackers," but how it is managed and shared is crucial.SBOMThis information is primarily for the defense side, and plays a role in accelerating vulnerability response and identification of the scope of impact by increasing transparency. On the other hand,SBOMThe information contained therein requires security considerations depending on the scope of disclosure and recipients, and therefore should not be made public without restriction. AttackersSBOMEven if it does not exist, attacks can be carried out using techniques such as reverse engineering and known vulnerability information.SBOMConfidentiality does not directly translate to attack deterrence. In the case of medical devices,SBOMThe defensive benefits of properly managing and sharing this information are considered to outweigh the risks.

・"SBOM"Therefore, you won't get any useful or practical information."

SBOMThis is not merely a list of parts, but a practical tool that supports suppliers, manufacturers, and stakeholders responsible for the operation, maintenance, and security of medical devices through the visualization of software configurations.

It is an information infrastructure. For example, when cyberattacks or vulnerability information is made public,SBOMBy using this method, it is possible to quickly determine whether or not a company's products will be affected and to what extent the impact will extend. This is practically useful in the medical device field, where post-market safety measures and incident response are highly valued.

・"SBOMThis is used only by the security department and has nothing to do with development, quality assurance, or regulatory affairs."

SBOMThis information is not exclusive to the security department. For the development department, it is useful for understanding the components used, organizing dependencies, and visualizing factors that reduce maintainability and elements that will require future updates or replacements. For the quality assurance department, it can be used as supporting information for change management, impact analysis, and regulatory compliance. For field service and maintenance personnel, it is also useful as basic information for verifying the configuration of individual devices and planning updates.SBOMIt should be positioned as a common language for development, quality, and security to work together.

・"SBOM"You only need to create it; maintenance and management are unnecessary."

SBOMIt is assumed that this is not a one-time creation, but rather a system that will be continuously updated and managed in response to software revisions and library updates. This is especially true for medical devices, as products are used for long periods even after they are marketed.SBOMObsolescence of this technology leads to the weakening of vulnerability management.SBOMIt is important to position this as part of software lifecycle management and operate it as a configuration management and change management process.

・"SBOM"We need to address all vulnerabilities identified by the tool."

SBOMThe results detected by vulnerability scanning tools and vulnerability search functions within the tool may include vulnerabilities that are not actually exploitable or that do not affect the product configuration in question. In medical devices, prioritization based on risk assessment, scope of impact, and response costs is essential, taking into account patient safety and stable operation of the device.SBOMThis information is intended to support decision-making and does not require a mechanical response to every detection result.

・"SBOMThe implementation costs are too high, making it unsuitable for small and medium-sized manufacturers and distributors."

SBOMImplementation involves learning costs related to understanding the concepts and operating the tools, as well as initial costs for tool implementation. However, freeOSSBy utilizing tools and phased implementation, it is possible to avoid excessive burden. Furthermore, considering the medium- to long-term benefits such as more efficient vulnerability response, simplified regulatory compliance, and faster response in the event of an incident,SBOMRegardless of scale, it can be considered a rational investment. The important thing is to consider it within the scope appropriate to your company's implementation objectives.SBOMThe key is to make use of it.

・"SBOMteethOSS"Only that needs to be managed; in-house developed code and commercial components are excluded."

SBOMThis is included in the productOSSThis does not apply only to our own developed software, commercial software components (OTSThis information visualizes the entire software configuration, including the software itself, third-party libraries, etc. In the case of medical devices, it takes into consideration the long-term use after market release and the impact on patient safety.OSSNot limited to comprehensiveSBOMManagement is essential.

It is essential.

・"SBOMis oneExcelThat's all you need; tools and normalization aren't necessary."

SBOMThis is not merely a list of software components, but an information infrastructure based on the premise of describing configuration information in a standardized format and managing it in a way that allows for mechanical interpretation and utilization. In the medical device field as well, taking into account international trends,SBOMAs a formatSPDXorCycloneDXIts use is required.SBOMImplementation doesn't need to be done all at once; in the initial stages of implementation...Excel To visualize the main components, Aligning the understanding of the structure among all parties involved is effective. However, ExcelUsing only this method helps to suppress variations in notation, prevent updates from being missed, and enable automatic matching using software identifiers. There are limitations to consistently managing change history. In practical useSBOMIn order to continue using it, Depending on the operational maturity level, this may include support for standard formats, centralized management using tools, and multiple options. SBOM It is necessary to gradually introduce normalization to absorb inconsistencies in notation.

・"SBOMSimply installing the tool itself becomes a complete security measure.SBOMThe process is now complete.

SBOMTools are merely means to streamline the collection, organization, and updating of configuration information; they do not automatically eliminate vulnerabilities or prevent incidents.SBOMTo translate this into effective security improvements, first, existing processes within the organization, such as software lifecycle management, change management, and corrective action response, must be addressed.SBOMIt is essential to incorporate operational procedures and establish operational procedures.

Furthermore,SBOMEffective security improvements can only be achieved by continuously implementing vulnerability management and incident response based on this foundation. It's important to note that simply introducing tools without a proper strategy often results in a superficial response.

・"With the tool"SBOM"As long as you generate it, the content will automatically be correct, so no verification is necessary."

Output by the toolSBOMUnconditionally accepting this as correct poses a quality assurance risk for medical device software.SBOMThe accuracy and completeness of the analysis depend on the definition of the target of analysis, the build procedure, the tool settings and detection capabilities, and the operational process, as well as static linking.OSSCare must be taken when handling modified components and generated products.

Therefore, what is important isSBOMThis approach involves ensuring that the tools and operational processes used to generate the data are appropriate in light of their intended purpose.SBOMRegarding the generation tools, we will also consider the concept of computerized system validation and, after confirming that they function appropriately for their intended purpose, treat the resulting output as valid information.

SBOMIt is not a finished product at the time of creation; it is necessary to continuously ensure reliability that contributes to cybersecurity risk management and regulatory compliance through operation based on verification, evaluation, and improvement.

3. Basic Policy Regarding SBOM Implementation and Operation

3.1. Basic Policy for SBOM Implementation and Operation

The implementation of SBOM in medical devices is not merely about documentation or regulatory compliance; it aims to improve safety, quality, and maintainability throughout the entire software lifecycle. In particular, when vulnerability information is disclosed or an incident occurs, it is necessary for stakeholders, including software development departments, quality assurance departments, and PSIRTs, to quickly identify the scope of impact and take appropriate initial responses. For this reason, the practical usefulness of developing SBOM as information for understanding software configuration and scope of impact is increasing in the operational phase, including post-market.

The software incorporated into a product consists of a diverse range of components, including not only in-house developed software, but also open-source software (OSS), purchased libraries, and out-of-the-box software (OTS). All of these software items must be classified according to their impact on safety and security, and managed as part of risk management.

While manufacturers and distributors bear ultimate responsibility for the safety and security risk management of the product as a whole, the correction and maintenance of OTS software are generally handled by each party based on contractual relationships with suppliers and other parties, with each party sharing responsibilities. SBOM (Software-Based Memory) is positioned as common information to visualize software components based on this division of responsibility, and to support the rapid assessment of the scope of impact and decision-making in the event of a vulnerability.

While the automatic generation of SBOMs (Software-Based Models) alone does not achieve its objectives, it is considered an effective means of detecting discrepancies between developers' perceptions and actual build artifacts, and of objectively understanding the factual relationships of the software configuration. On the other hand, since product scale, software configuration, and organizational structure vary greatly from company to company, demanding comprehensive and advanced SBOM operation from the initial stages may actually lead to it becoming a mere formality.

Therefore, this guideline adopts the basic policy of establishing effective operation in the field by gradually introducing and expanding SBOM. First, it is important to clarify the issues that your organization wants to solve with SBOM and the purpose of its introduction, and then start with the minimum scope and items corresponding to those, and gradually increase the level of sophistication according to the organization's maturity and operational performance.

Phase 1: Establishing a system for creating and updating SBOMs (Implementation Phase)

The first stage aims to establish a system for creating and updating SBOMs, based on the minimum elements of an SBOM, and to create SBOMs for target products and establish a state where changes can be managed. At this stage, it is important to base the SBOM format on internationally recognized standard specifications such as SPDX or CycloneDX. Adopting an SBOM that is confined to a proprietary specification from the initial stage may hinder future compliance with regulatory requirements and information sharing with suppliers and business partners, so it is necessary to design it with connectivity to international standards in mind. This will enable smooth future expansion of the SBOM, introduction of tools, and response to information requests from regulatory authorities.

The information to be included in the SBOM will initially consist of configuration information focusing on the minimum elements of the SBOM.

The focus is on accurately managing basic items such as software component identification, name, version, and supplier information. Furthermore, since SBOM is not something that is created once and then forgotten, but rather an information asset that is updated in accordance with software revisions and configuration changes, it is necessary to establish an operation that links it to the change management process.

In addition, it is important to clarify the roles and responsibilities regarding the creation, updating, storage, and provision of SBOMs internally and externally, and to establish a common understanding among relevant departments, including not only the development department but also quality assurance, regulatory affairs, PSIRT, and security departments. In the initial phase, it is practical to start operations on specific products or within a limited scope, understand the operational burden and challenges, and then gradually expand the scope.

Phase 2: Integration with QMS and post-market safety management (Advanced Phase)

In the second phase, the goal is to integrate the SBOM creation and management system established in the first phase with existing QMS and post-market safety management processes such as vulnerability management, license management, and EOL/EOS management, thereby evolving the SBOM into risk management for the entire medical device lifecycle.

Since medical devices are intended for long-term use and continuous maintenance after market release, the SBOM (Software-Based Memory) should not be considered merely information referenced during the development phase, but rather as an information asset utilized throughout the entire lifecycle, including post-market support. Therefore, it is necessary to set a retention period for the SBOM that takes into account the expected product lifespan and maintenance contract period, and to manage it in conjunction with software revision history, corrective actions, security response history, etc.

At this stage, it is desirable to gradually expand the information recorded in the SBOM beyond the minimum elements initially included in the initial implementation, to include information that is highly practically useful in vulnerability impact assessment and maintenance decisions, such as inter-component dependencies, license information, and EOL/EOS information. This will enable the rapid and systematic evaluation of the impact of publicly disclosed vulnerability information on your products, and to implement corrective actions and customer support based on consistent decision-making criteria.

Furthermore, by strengthening coordination with PSIRT activities, post-market safety management, and quality management through information sharing centered on SBOM, SBOM will function not merely as a list of configuration information, but as an effective risk management tool that supports the safety and reliability of medical devices.

3.2. SBOM Implementation and Operation System and Processes

In medical devicesSBOMImplementation and operation should not be merely about creating lists, but rather positioned as a core management tool supporting software configuration management and cybersecurity measures throughout the entire product lifecycle. Therefore,SBOMThis should not be treated as a limited part of development activities, but rather established as a consistent process under a cross-organizational operational structure.

SBOMThe roles related to introduction and operation are summarized below, along with the main tasks and outlines for each role.SBOMThe responsibility for creating such documents rests, in principle, with the manufacturer and distributor.OSS,OTS Regarding software, external vendor software, etc., from third partiesSBOMEven when receiving information, the manufacturer or distributor is responsible for verifying its validity, integrating it, and managing its contents.

			社内関係者					
フェーズ	ステップ	実施概要	開発部門	品質保証部門	薬事部門	法務部門	サービス部門	PSIRT
環境構築・体制整備	適用範囲の明確化	対象ソフトウェアを精査し適用範囲を明確化する。	✓					✓
	ツールの選定	選定観点を整理し、評価・選定を行い潜在課題整理を行う。	✓					✓
	ツールの導入・設定	選定評価時の顕在課題を解決し、導入・設定を行う。	✓					✓
作成・共有	コンポーネント定義	対象ソフトウェアの構成要素を識別・整理し根拠立てを行う。	✓	✓				✓
	SBOMの作成	要件(記載要素、フォーマットなど)に沿った作成を行う。	✓					✓
	SBOMの管理	統合、編集などのプロセス、アクセス権限等のルールを定める。	✓	✓				✓
運用	脆弱性情報の収集	脆弱性情報との効果的・効率的な照会方法を定める。		✓				✓
	ライセンスの管理	ライセンス評価ポリシーを定める。	✓	✓		✓		✓
	影響調査と対応要否	優先度付け(トリアージ)定義したプロセスを定める。	✓	✓				✓
	対処	製品・事業の対外的関係者を考慮したプロセスを定める。	✓	✓			✓	✓
	開示	決定方針に沿った開示・情報提供を行う。		✓	✓		✓	✓

figure2:SBOMExamples of implementation and operation processes and the involvement of stakeholders

役割	主管部署	所掌範囲例
SBOMの導入を主管する	PSIRT	› 企業内でのSBOM導入推進（啓蒙・構想・利害関係者明確化など） › 外部ベンダー調査・情報収集 › 外環境情報の収集
SBOMを作成する	開発	› ツール利用による解析・特定フォーマットでの出力
申請・開示を行う	薬事/法務	› SBOM内容から申請・開示内容に沿った形式へ変換
SBOMを運用・管理する	品質管理	› SBOMの運用・管理方針の策定（アクセス制御・保管ポリシーなど）
脆弱性を管理する	セキュリティ統括/PSIRT	› 管理下にある脆弱性情報との照合 › 自社インシデント対応プロセス効率化
ライセンスを管理する	開発/品質管理/法務	› SBOM内容からライセンス利用状況・種別を確認 › ライセンスの監査（ライセンス違反の確認）

figure3:SBOMExamples of implementation and operation processes and the roles of stakeholders.

SBOMTo maximize its effectiveness after implementation, a collaborative system involving multiple departments within the organization is essential. Development, quality assurance, Pharmaceutical and legal affairs departments, Service department andPSIRT/ In addition to the security department, external vendor software and development contractors,SBOMStakeholders with different roles in the creation, evaluation, and utilization of vulnerabilities are required to share information with each other and continuously collaborate throughout the entire process. Establishing this collaborative system will enable the rapid identification of vulnerabilities and

This enables a significant improvement in security levels and quality throughout the entire product lifecycle. Furthermore, it allows for consistent explanations in response to external inquiries and regulatory requirements, contributing to increased organizational credibility.SBOMIts implementation should not be seen merely as a technical initiative, but rather as a mechanism to promote strengthening governance throughout the entire organization.

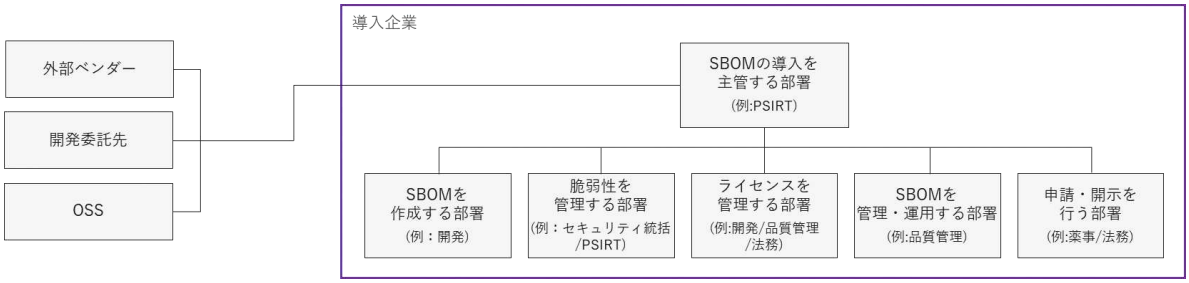


figure4Examples of internal and external related organizations

4. Actions to be taken during the planning phase (environmental and organizational development)

4.1. Clarification of the Scope of Application of SBOM

SBOMIn order to operate it effectively, the target medical device and the software running on which hardware components within the device are necessary.SBOMIt is important to clearly define the scope of application during the planning stage.SBOMCreating such a system could lead to the omission of critical components or cover an excessively broad scope in practical terms, potentially hindering risk management and maintenance operations.

This section summarizes the configuration and characteristics of the target software.SBOMThis document outlines considerations for systematically visualizing the scope of application.

In medical devicesSBOMThe scope of application varies greatly depending on the software configuration and architecture. In this guideline, a typical medical device software configuration is described below.4pattern(A~D) and for eachSBOMWe will consider the scope of application.

table2:Software Configuration Example

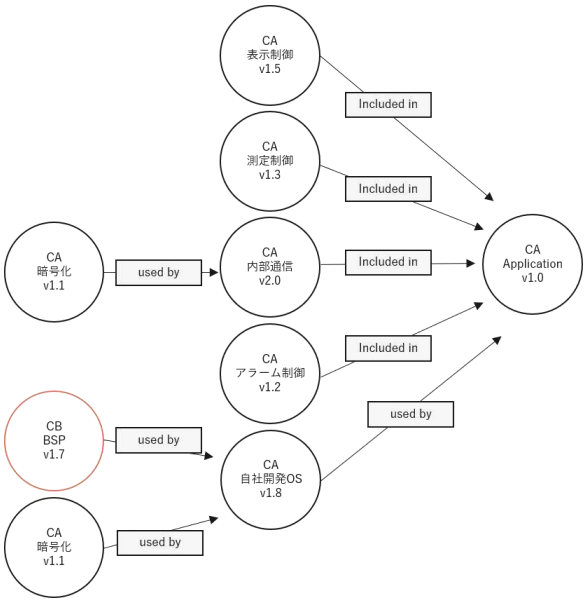
pattern	Software Configuration	Product Examples
A	Embedded software primarily based on in-house developed code	➢ Electronic blood pressure monitor ➢ pulse oximeter ➢ Portable electrocardiograph ➢ Small infusion pump
B	In-house developed code and commercialOSDevices that operate on the above Installed software	➢ Bedside monitor ➢ respirator ➢ dialysis machine ➢ Specimen testing equipment
C	SaMD(Software as a medical deviceSoftware provided as	➢ Image analysis program ➢ AIDiagnostic support program ➢ Patient management support app
D	A~CSoftware composed of a combination of elements	➢ MRIsystem ➢ CTsystem ➢ Biological information monitoring system Mu

these areSBOMThis is a model case for considering the scope of application, and actual products may contain a mixture of multiple patterns. In such cases, apply this pattern to each software component of the target product and consider each one.SBOMClarify the scope of application.

Each individual componentSBOMIn addition, the entire system that shows the relationships between those components.SBOM Alternatively, by preparing a diagram, it becomes easier to use for risk management throughout the entire lifecycle.

It gets worse.5This is organized once the software configuration for the medical device has been finalized. SBOMThis shows the final configuration image. Specifically, it shows the software elements that operate for each hardware component that makes up the device, visualizes their dependencies and hierarchical structure as a component tree, and organizes information such as the name, version, provider, and dependencies of each component as a component table.

ActualSBOMDuring the creation process, the software configuration is gradually clarified as the design and implementation progresses, but ultimately it is not represented by a diagram.5The structural relationships shown are organized in the following way.SBOMThis is intended to be generated. This diagram shows an example of the completed form, and the hierarchical structure and granularity of the components will be adjusted as appropriate according to the architecture of each medical device.



ID	Component Name	Supplier	Version	Author	Hash	UID	Relationship	Relationship Completeness
1	Application	CA	1.0	CA	0x111	111	Primary	Known
2	表示制御	CA	1.5	CA	0x222	222	Included in ID1	None
3	測定制御	CA	1.3	CA	0x333	333	Included in ID1	None
4	内部通信	CA	2.0	CA	0x444	444	Included in ID1	Known
5	暗号化	CA	1.1	CA	0x555	555	Used by ID4,ID7	None
6	アラーム制御	CA	1.2	CA	0x666	666	Included in ID1	None
7	自社開発OS	CA	1.8	CA	0x777	777	Used by ID1	Known
8	BSP	CB	1.7	CA	0x888	888	Used by ID7	Unknown

*CA...Company A, CB...Company Babbreviated expression

figure5:Between componentsSBOMConceptual diagram

4.1.1. A. Embedded software primarily based on in-house developed code

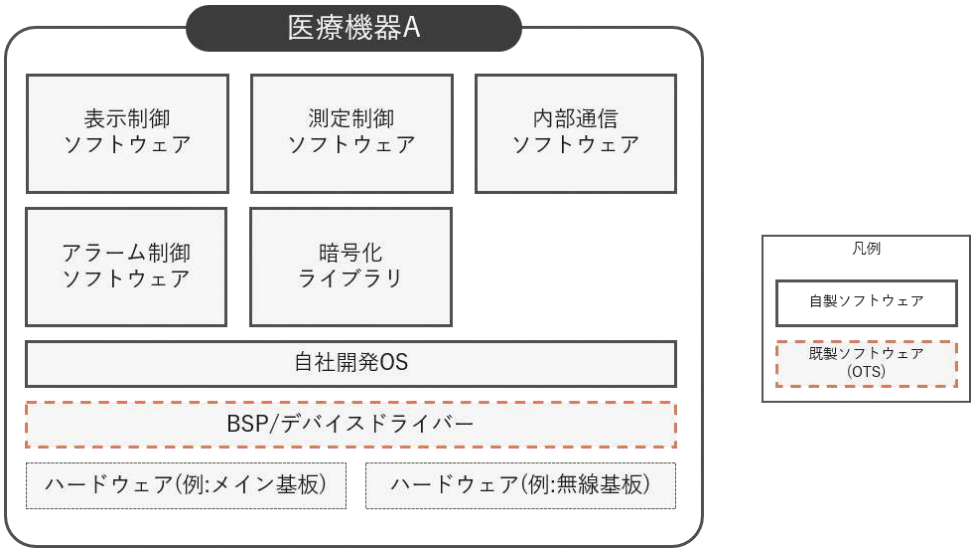


figure6:Medical devicesASoftware configuration (example)

◆ Prerequisites regarding software configuration

The focus is on firmware that runs on dedicated hardware, and commercialOSThis assumes a configuration where the custom-developed code is implemented on bare metal or a simplified runtime, without using any external libraries. In some cases, a lightweight custom-developed system with only basic functions such as task scheduling and memory management may be used.OSThis also includes configurations that utilize [specific technologies/libraries]. In some cases, software components other than those developed in-house may be used, such as standard libraries included with the compiler or external libraries provided by vendors for numerical computation, encryption, and communication.

◆ Features

Because the majority of the software configuration is under our internal control, understanding components and managing changes is relatively easy. On the other hand, external libraries, toolchains, and board support packages (BSP), components included in vendor-provided embedded libraries, etc., may be implicitly used, andSBOMAutomatic detection by tools is difficult, and the third-party components included in these areSBOMIt leaks easily.

Furthermore, due to limited network connectivity, updates are infrequent, and there is a risk that vulnerabilities originating from older compilers/libraries and deprecated algorithms will persist for extended periods.

◆ SBOMScope of application

In this pattern, at least the following components are requiredSBOMInclude it in the target.

- In-house developed firmware (individual component units within the identifiable range)
- Linked external libraries (compiler-provided libraries, mathematical, cryptographic, and communication libraries, etc.)
- If you are using vendor-provided middleware equivalent components (such as evaluation board libraries), please provide their names and versions.

- SoC/FPGAUpdateable firmware for peripheral devices (communication modules, etc.)
- Vendor-provided board support package (BSP) , startup code etc.
  Party Components
- Runtime libraries that originate from the build toolchain (compiler, linker, etc.) and are embedded within medical devices.

On the other hand, components that are not incorporated into the final medical device product, such as build-specific tools and test code for testing, are typicallySBOMIt can be excluded from the scope.

4.1.2. B. Onboard software running on proprietary code and commercial operating systems

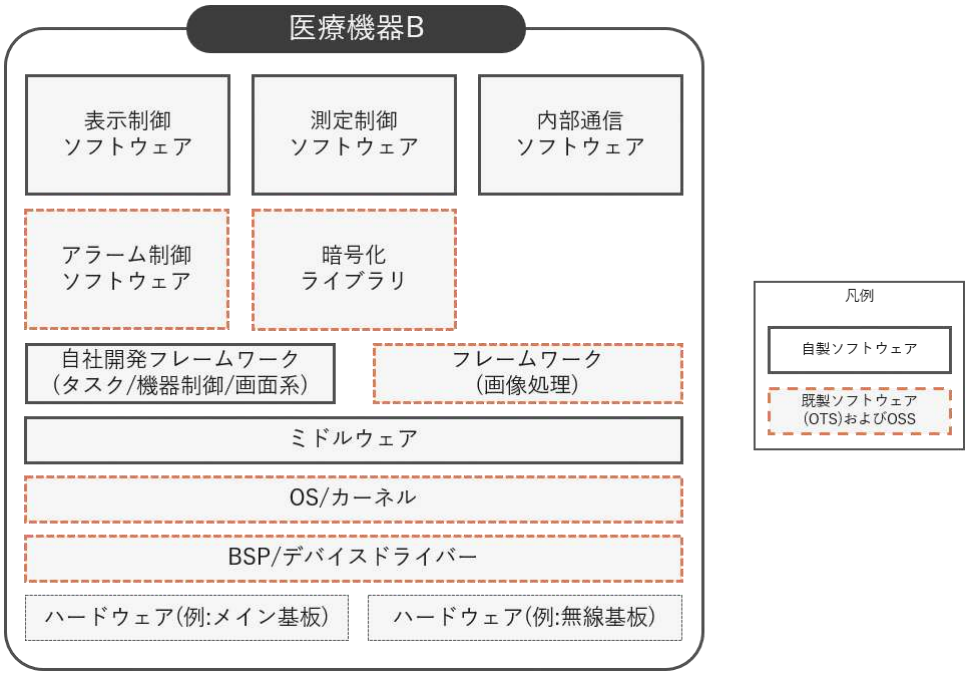


figure7Medical devicesBSoftware examples

◆ Prerequisites regarding software configuration

Medical devices that fall under this pattern are:Windows,Linux,RTOSCommercial use such asOSIt assumes a configuration where multiple software layers operate on top of it, with each layer being interdependent. Typically,OSIt has a hierarchical structure consisting of device drivers, middleware, runtime, and application software, with each layer being coupled by explicit and implicit dependencies.

Furthermore, the software that runs within the product is not solely composed of in-house developed code, but also includes software provided by external vendors.OTSSoftware, Open Source Software (OSS), and the resulting complex configuration includes transitive dependency components contained within them.SBOMWhen defining the scope of the project, it is necessary to take a comprehensive view of the entire software execution environment, rather than limiting it to a single deliverable or superficial configuration.

◆ Features

The most distinctive feature of this pattern is that a portion of the software configuration is not under the complete control of the company.

ru.

The product is commercialOSIn addition to that,OTSsoftware,OSSThese are some of the components that are incorporated, and some of these have source code that is not publicly available, or whose detailed internal structure is difficult to understand.

Furthermore, a multi-layered dependency structure is commonly formed, where libraries used by an application depend on other libraries, and those libraries, in turn, depend on yet another module. While such a structure is effective in terms of extensibility and development efficiency, if a vulnerability is discovered, the scope of impact is wide.OSIt has the risk characteristic of having a wide-ranging impact, from the level to the application layer. Therefore, SBOMTo effectively perform vulnerability management and impact analysis using this method, it is essential to accurately visualize the overall software configuration, including dependencies.

◆ SBOMScope of application

In this patternSBOMThe scope of application, in principle, covers all software components operating within the product. It is not limited to applications and modules developed by the company itself, but must be comprehensively categorized to include the following: CommercialOSand its components, externally provided software incorporated into the product, open-source software, tools introduced without a development contract, and all software elements on which they internally depend.SBOM It will be subject to the following.

In particular, runtimes, frameworks, utility libraries, and components introduced via package management systems that do not appear to directly constitute product functionality are also considered part of the execution environment.SBOMIt is important to include this. This will enable the establishment of a system that allows for the rapid and objective determination of whether or not there will be an impact on the medical device when vulnerability information or licensing information is made public.

In this pattern, at least the following components are requiredSBOMInclude it in the target.

- The device's built-in application (including proprietary software, to the extent that it can be identified as a main functional unit or module)
- External libraries used directly by the application (numerical computation, image processing, communication, etc.) and their dependent libraries.
- Open source software incorporated into the product (OSS)component
- CommercialOSAdditional middleware, frameworks, and runtimes that are installed or included on top of the above (e.g., ..NET Framework,Java,Pythonetc.)
- Directly involved in realizing product functions, safety, and basic performance.OSThe above services and additional components (specificWebserver,DB(Servers, communication stacks, device drivers, etc.)
- Provided by an external vendorOTSSoftware and tools that are implemented without development contracts.

In this patternSBOMThe scope of application, in principle, covers all software components operating within the medical device product. However, it is not possible to cover all layers at once.

Because this is often not practical, in the initial stages of implementation, the applications installed on the device, the middleware and runtime embedded in the product,OSA phased implementation targeting top-level components such as the services and additional components mentioned above is effective. Furthermore, dependency information from the package manager and information provided by suppliers are also useful.SBOMBy utilizing these methods, the scope of application is gradually extended to lower-level dependent components, starting with the parts that can be obtained.

4.1.3. C. Software provided as SaMD (Software as a Medical Device)

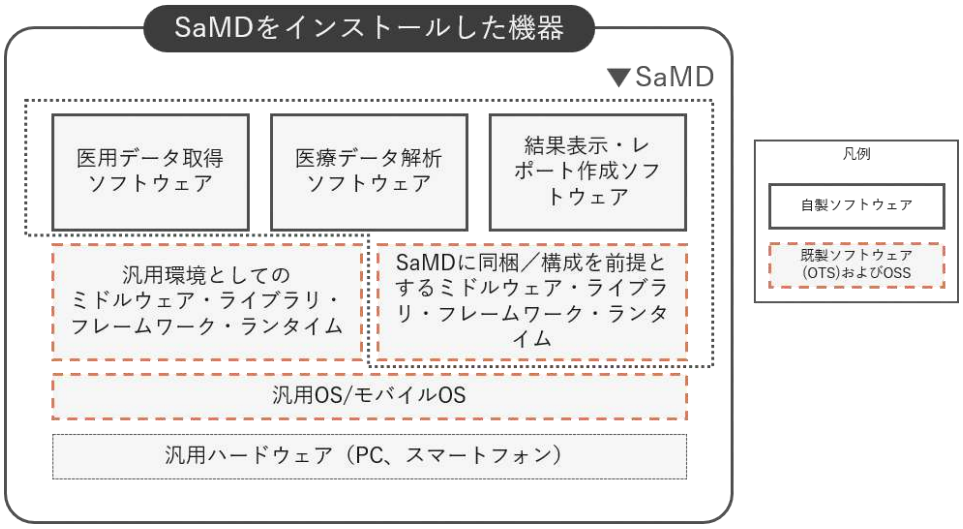


figure8:Medical devicesCSoftware configuration (example)

◆ Prerequisites regarding software configuration

General purposeOS(example:Windows,Linux) and mobileOS(example:iOS,Android) Program medical devices that operate on (SaMD) will be the target.CT/MRITypical examples include image analysis software that detects potential lesions from images, diagnostic support software that analyzes electrocardiogram data, and diabetes management apps for smartphones. In addition to the application itself, runtime components are also included..NET Framework,JVM,Pythonetc.),Web /GUIFrameworks, external libraries (image processing, numerical computation, cryptography/communication, etc.) database orWebThis assumes a configuration that combines middleware such as servers.

◆ Features

OSSorOTSIt depends on numerous components, including software, and the software configuration forms a layered structure with multiple levels of dependencies. In addition, the runtime and framework automatically retrieve and load dynamic link libraries and plugins, and externalAPIBecause the configuration may change during execution due to reliance on cloud services, it is difficult for the development team alone to fully grasp the actual components being used.SaMDContinuing to operate without clearly identifying which components are being used in a particular version makes it difficult to later identify vulnerabilities or the scope of impact from end-of-support.

◆ SBOMScope of application

In this pattern, at least the following components are requiredSBOMInclude it in the target.

- SaMDThe main application module (including frontend/backend, etc.)
- The framework and runtime being used (e.g., .).NET Framework,Java,Pythonetc.)
- External libraries (image processing, numerical computation, cryptography, communication, etc.) , and embedded databases and messengers middleware such as saging
- SaMDDirectly related to the safety and performance ofOSThe additional components above (specificDB server, Web(Servers, drivers, etc.)

General purposeOSWhile it's not necessarily required to list every single component included as standard, SaMDFocusing on components directly related to the basic performance and safety, as well as components that are installed or included in addition to the standard configuration,SaMDTo enable management that reduces cybersecurity risks that could impair functionality or raise security concerns. SBOMWe need to define the scope.

In addition,SaMDCloud services andSaaSEven when relying on a specific service, these guidelines stipulate that manufacturers and distributors must have access to all component information within that cloud service. SBOMI don't require it to be included.

however,

- Cloud components that directly affect the safety and basic performance of medical devices (e.g., managed systems that store medical data)DB(Image analysis engine, etc.)
- Components whose versions and support policies are managed by manufacturers and distributors through contracts and specifications.

Regarding this, to the extent possibleSBOMAlternatively, record the configuration information in the relevant documents.SaMD It is necessary to utilize this in risk management so that it is possible to manage and reduce cybersecurity risks that may impair functionality or raise security concerns.

4.1.4. D. Software composed of a combination of A and B

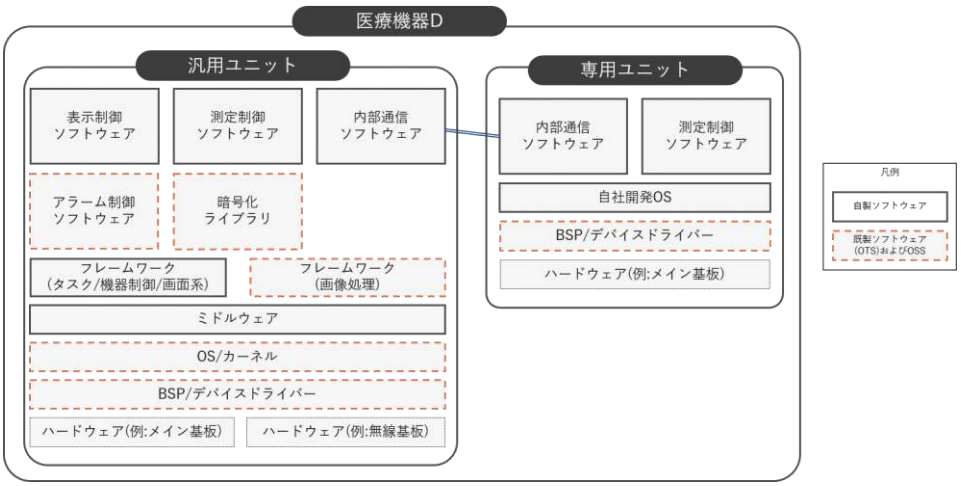


figure9:Medical devicesDSoftware configuration (example)

◆ Prerequisites regarding software configuration

This pattern is one of several software configuration patterns (A,B) combined,1We envision medical devices being provided as a single system.

MRI/CTTypical examples include systems, biological information monitoring systems, and specimen testing systems, which include components such as the following:

- Measurement units, etc., embedded software (patterns) primarily based on proprietary code.A)
- Imaging devices, bedside terminals, control consoles, etc.OSOnboard software that operates on the above (pattern)B)
- Analysis software and viewer software that operate on a terminal or cloud separately from the device itself.

These are interconnected in a network, as a whole system1This is based on the assumption that the system is designed and provided as a single medical device, or as a group of medical devices used together.

◆ Features

Each component of the system has a different lifecycle and update cycle (updates to equipment software, additions/replacements of peripheral devices, etc.), and partial updates and configuration changes are often performed. The software of each component is

- The part that the manufacturer completely manages (pattern)A)and,
- CommercialOSmiddleware,OSS/OTSParts that depend on third-party components such as (patterns)B)

A mixture of these exists.

When a system is implemented in a medical institution, site-specific configurations (number of connected terminals, presence or absence of modules, network configuration, etc.) are often set. "The whole product" and "each individual component" Understanding the configuration from both perspectives is crucial for vulnerability management and legacy medical device management.

◆ SBOMScope of application

In this pattern, "By componentSBOM"and "to have an overview of the entire system"SBOMThe two-layer structure of "

Let's think about it.

- Each componentSBOM
  - ➢ Each software component of the system (e.g., imaging device, control console, analysis workstation, server application, etc.) is handled by the medical device.A,BFollowing this way of thinkingSBOM Create (see the respective sections for details on each pattern). .
- To get an overview of the entire systemSBOM/Configuration Information
  - ➢ The product model number and configuration pattern as a system, and each element that constitutes it (A,BWe will organize the correspondence with each medical device. "Which system configuration and installation site, which component?" To enable you to determine whether it contains "
  - ➢ At this time, each componentSBOMReferences to (file names and documents)IDBy having such features, it becomes possible to quickly and efficiently identify potentially affected system configurations when vulnerabilities or end-of-support issues arise in specific components.

4.2. Selection of SBOM Tools

SBOMThe implementing organization was clearly defined in advance.SBOMBased on the scope of application, appropriateSBOMIt is necessary to establish an environment and system that allows for the creation and management of such documents.SBOMThe central role in creating and managing it isSBOMAs a tool, selecting the right tool is extremely important, taking into account the scale and configuration of the target software, as well as the organization's development structure.

SBOMWhile it's not always necessary to use tools to create them, and it's theoretically possible to create and manage them manually, in environments where the configuration is complex and sophisticated, such as medical device software, it's necessary to understand the configuration, including indirect dependencies that developers may not be explicitly aware of, and components that are automatically incorporated during the build process.

In response to such requests,SBOMBy utilizing these tools, the effort required for component management can be significantly reduced, dependencies between components and reusable parts can be efficiently identified, and the lead time from the disclosure of a new vulnerability to identifying affected products and scope can be shortened, among other benefits.SBOMThere are significant advantages to using the tool.

In these guidelines,SBOMUtilizing the powerSBOMAssuming the creation and management of [the system], this document outlines the points to consider and precautions to take when selecting tools.

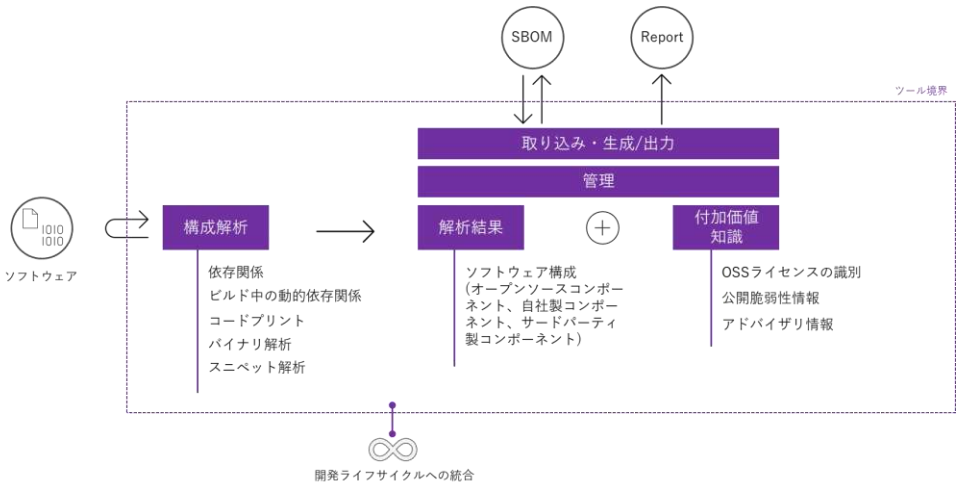


figure10:SBOMExample of an overview to consider when selecting a tool

4.2.1. Criteria for selecting SBOM tools

SBOMWhen evaluating and selecting tools,SBOMBased on the purpose and scope of implementation, the following points should be considered in advance.

table3:SBOMExample of tool selection criteria

function	The features supported by each tool vary.
	Select the necessary functions based on the purpose (SBOM creation, vulnerability management, development support, regulatory compliance, etc.) and scope of application.
performance	The degree of false positives (accuracy of matching OSS detection and vulnerability license information) and missed detections.
	How quickly are new vulnerabilities reflected?
Analyzable information	Vulnerabilities and licenses can be analyzed automatically.
	Does it have strengths such as additional information or accuracy?
Analyzable Data format	Supported file formats (support status by file extension), package manager types, etc.
cost	Definition of licensing and fee structures
	Billing method (amount of analysis code and number of developers)
	Whether or not there are economies of scale
Compatible Format	Output formats
	Types of importable formats (SPDX, CycloneDX)
Components Result analysis method law	Is the analysis method one that meets the predetermined analysis criteria (code analysis, build information analysis, binary analysis, etc.)?
	The accuracy may decrease depending on the analysis method, but is it acceptable?
	Code matching, snippet matching, binary analysis, etc.
	Some tools allow for multiple analyses.
Support body system	Plans that allow inquiries other than about the tool, and support for free tools.
	Support contract based on the knowledge level of the person in charge
Other tools and Collaboration	Integration to streamline the entire development lifecycle (such as integration of development environments, build tools, and container scan tools).
Provided form	We will also consider the maintenance costs for both on-premise and cloud-based servers.
	Cloud versions require careful attention to ensuring confidentiality.
Japanese language support	Few developers offer Japanese language support.
	In some cases, the sales agent may have translated the instruction manual into Japanese.
User Interface - Face	CLI or GUI provided
	GUI-enabled tools allow for intuitive operation and visualization of results.
Operation method	Reduce the burden on developers by integrating with the development environment.
	The ease of automation is expected to improve analysis accuracy and operational efficiency.
Software Development Language	While many tools support languages like C and C++, support for some languages is limited.

4.2.2. Points to Note When Selecting an SBOM Tool

SBOMWhen selecting a tool, it is important not to judge solely on the number of features or price, but to comprehensively evaluate its suitability to the purpose of implementation and operational style. First, multipleSBOMUsing different tools for different purposes can complicate operations and potentially increase management costs and human resources. Therefore, unless there is a specific reason otherwise, consider whether the system can be operated with the minimum necessary tools for the given purpose.

SBOMWhile tools exist in both paid and free versions, the distinction between paid and free is merely a difference in the form of provision and does not allow for a uniform evaluation of functionality or practicality. The following should be understood as general trends, and ultimately, individual decisions should be made based on the requirements of your own organization.

- Generally, paidSBOMThe tool,Web UIIn many cases, an administration screen and intuitive operation are provided.SBOM Generation, viewing, and managementGUIThere are several things that can be done above. Also, in many cases, you can consult with the vendor or sales agent regarding any questions you may have during implementation and operation. Furthermore, there are mechanisms to automatically perform software builds, tests, and releases.CI/CD) and the ability to easily integrate with existing development environments, such as by linking with systems that manage the change history of source code and deliverables (version control systems), can be an advantage.

- FreeSBOMTools can be implemented with low initial costs and are often suitable for small-scale trials and proof-of-concepts. Start by using free tools.SBOMCreate a document to understand the configuration of your company's software and SBOMBy understanding the operational flow through practical experience, it becomes possible to organize operational challenges and necessary functional requirements. Based on these results, a step-by-step approach is effective in determining whether or not to migrate to paid tools or implement additional ones.OSS Some tools are continuously improved through community-driven efforts, and future functional enhancements can be expected. However, there are points to keep in mind when using free tools. Information regarding environment setup and configuration may not be adequately available. Also, many free tools are developed overseas, and reference documentation is often only available in English. Therefore, when installing, it is necessary to use sample code to ensure that the expected results are achieved.SBOMSetting up the system may require trial and error, such as checking whether the output is correct. To alleviate this burden, freeSBOMReceiving support from companies that provide implementation assistance and support services for the tools is also an option.

The method of providing the tools also requires careful consideration. On-premise tools are available for implementation. OS While the network environment may be limited,SaaSWith type tools, it is necessary to verify in advance whether the structure ensures that highly sensitive technical information, such as source code, is securely managed. Furthermore,SBOMTo ensure that the introduction does not lead to a decrease in development efficiency, it is important to select tools that can be seamlessly integrated into existing development processes and to design operations that minimize the burden on developers. SBOMTools come in a variety of types depending on their output format, analysis method, vulnerability integration capabilities, etc.SBOMIn terms of format,OWASPFormulated byCycloneDXAnd,The

Linux FoundationFormulated bySPDXThere are also toolsets and converters available for each format, which can be used to compare and consider options, for example, by referring to the following information.

https://cyclonedx.org/tool-center/

https://spdx.dev/use/spdx-tools/

4.2.3. Evaluation and Selection Process

SBOMWhen evaluating and selecting tools, it is desirable to compare and consider multiple tools based on organized criteria. By utilizing free trials and conducting experimental analyses using the source code of your company's representative products or projects, you can concretely evaluate usability, learning curve, and the validity of the analysis results.

Furthermore, if selection is difficult, multipleSBOMConsulting with a distributor that handles the tools and comparing their features, advantages, and disadvantages is also an effective approach.

4.3. Installation and configuration of SBOM tools

This document outlines practical considerations for the introduction and initial setup of SBOM tools necessary to begin SBOM operation in medical devices. Since SBOM tools do not immediately become effective simply by being installed, it is essential to prepare and configure an environment that is suitable for the target software configuration and operational objectives.

4.3.1. Confirmation and preparation of environmental requirements before implementation

The environment requirements for SBOM tools vary significantly depending on the product and vendor. Therefore, before implementation, it is necessary to thoroughly check the operating environment required by the target tool and prepare the necessary infrastructure. Specifically, requirements may include the presence or absence of internet connectivity, machine specifications such as CPU, memory, and storage, the type and version of the supported OS, the specified supported browser, and the software build environment for the PC or server on which the tool will run.

Furthermore, some SBOM tools only support Linux environments, and using them on Windows terminals may require the creation of a virtual machine or container environment. In medical device development, alignment with internal IT policies and security rules is also important, so it is necessary to understand the gap between tool requirements and the company's environment in advance and to systematically prepare the deployment environment.

4.3.2. Installation and Initial Setup of SBOM Tools

After the environment is set up for deployment, the SBOM tool will be installed and initially configured. The basic procedure will follow the official documentation provided by each tool, such as the instruction manual and README file. Initial configuration involves setting the SBOM output format (SPDX, CycloneDX), the range of source code and binaries to be targeted, analysis options, and the location to save the output file, which are directly related to SBOM creation.

The items that are affected need to be appropriately defined.

When using paid SBOM tools, support systems are often in place by sales agents or tool vendors, allowing for efficient and reliable initial setup by providing support and technical assistance for installation and configuration inquiries.

Regarding the introduction of SBOM tools, please note that the sales agents and tool vendors to be used are as follows: "Medical Ministerial Ordinance Concerning Standards for Manufacturing Control and Quality Control of Medical Devices and In Vitro Diagnostic Reagents (Ministry of Health, Labour and Welfare, 2004) Ministry of Labor Ordinance No. 169 (hereinafter referred to as the "QMS Ordinance")) Note that it may be subject to purchasing management based on Articles 37 to 39. Also note that, in accordance with Article 5-6 of the QMS Ordinance, it may be necessary to implement validation, record management, and change management related to the SBOM tool to be introduced.

4.3.3. Operational Considerations for Vulnerability Management Use

When using the SBOM tool for vulnerability management, Known vulnerability information (CVE, etc.) is included when generating the SBOM. It is crucial to refer to and cross-reference data to identify vulnerabilities in the target software. This makes it possible to visualize security risks stemming from the open-source software (OSS) and over-the-counter (OTS) software being used, both during the development phase and after commercialization.

Furthermore, if vulnerabilities are identified through SBOM, it is necessary to integrate that information into the problem-solving process (impact assessment, risk analysis, corrective and preventive actions, change management, etc.) and establish a system to resolve them through technical measures and operational responses. In the medical device field, post-market vulnerability management is directly linked to patient safety and business continuity, so it is important to position SBOM tools as a means to support vulnerability detection and utilize them within the organization's established quality and risk management process.

4.3.4. Information and characteristics for identifying components

SBOMThe accuracy and comprehensiveness of component identification in this context depend on the underlying information used to identify and organize the constituent elements included in the target software.SBOMThe tool provides a mechanism for identifying and organizing components using multiple sources of information, such as dependency information, characteristic information contained in source code and binaries, and metadata associated with build artifacts.

Dependency information is a method that identifies direct and indirect dependencies by analyzing metadata from package managers and build tools. While it has a low rate of false positives, it cannot detect components outside of package management. Identification based on source code and binary characteristics is knownOSSThis method involves matching the code, and while it has the advantage of being able to detect copied and pasted code, it also has the potential for false positives and missed detections. Furthermore, it may also detect components that are not explicitly managed by tools, etc.SBOMWhile these elements may be definable as described above, it is necessary to pre-establish organizational interpretation rules regarding the definition units and the determination of authenticity.

4.4. Acquisition of SBOM information and supplier management for third-party software

4.4.1. Basic Policy

This section outlines the basic policy regarding obtaining information from suppliers and managing suppliers as a response to supply chain risks caused by third-party software. This policy is: 3.1. Basics of SBOM Implementation and Operation

This policy is concretized from the perspective of third-party software and supplier management. The purpose of supplier management is not simply to obtain the SBOM (Software-Based Manufacturing) deliverable, but to appropriately manage supply chain risks such as vulnerabilities and end-of-support issues that affect the security and safety of medical devices. Therefore, it is important to maintain relationships and systems that allow for the continuous acquisition of necessary information from suppliers, such as vulnerability information and support information, in addition to the SBOM.

Regarding the requirements for concretizing these into contract clauses and procurement specifications, 6.3. In procurement and contracting

Refer to the "SBOM requirements".

4.4.2. Information Sharing System with Suppliers

When manufacturers and distributors consider adopting third-party software or external services, they should evaluate whether the supplier can continuously provide information on software configuration and vulnerabilities, in addition to functionality, performance, and quality.

When contracting with suppliers, placing orders, or implementing solutions, the format and items to be provided, the timing and frequency of updates, the provision of lifecycle information, and vulnerability notifications should be considered.SBOMCheck for updates and other updates in advance whenever possible.

Furthermore, it is important to ensure that information obtained from suppliers is continuously updated throughout the product lifecycle, reflecting this in contract terms or operational procedures, and to maintain a system that allows for prompt acquisition of information when security incidents or critical vulnerabilities are disclosed. It is desirable to maintain records in a format that can be verified later regarding requests for information from suppliers, the content of their responses, and whether or not information was provided.

4.4.3. Alternative management when information cannot be obtained

Depending on the supplier's policies, contract terms, or product characteristics,SBOM, vulnerability information orEOL/EOS Information may not be provided in all cases. In particular, for general-purpose operating systems, cloud services, large-scale software platforms, etc., individual information may not be available.SBOMIt is possible that such information may not be available. Therefore, it is important for manufacturers and distributors to conduct a reasonable impact assessment using available information and to continue risk management based on the results. Alternative information could include publicly available security advisories, product specifications, release notes, vulnerability databases, supplier support information, contract information, and other publicly available information.

Furthermore, if necessary,Software Composition Analysis(SCAIt is also effective to supplement understanding the software structure and known vulnerabilities contained in the target product by using binary analysis, container image analysis, vulnerability scanning, and other technical methods. However, these results are prepared by the supplier.

The officialSBOMIt is not a substitute for other methods, and its use should be considered in light of its limitations in detection accuracy, comprehensiveness, and authenticity.

4.4.4. Continuous Review and Documentation

From the supplierSBOMIf information such as the above could not be obtained, the reason, the content of inquiries to suppliers and their responses, the alternative controls implemented, the publicly available information referenced, the results of the impact assessment, the risk controls adopted, and the results of the residual risk assessment will be documented.

Furthermore, instead of considering the inability to obtain information as permanent, we will reconfirm with the supplier at opportunities such as contract renewal, software updates, maintenance contract reviews, or periodic reviews, and implement new measures. SBOMIf vulnerability information is provided, it is desirable to promptly review the impact assessment and risk assessment.

5. Actions to be taken during the SBOM preparation and disclosure phase

5.1. Concepts and Procedures for Component Analysis

This section discusses medical device software.SBOMRegarding "component analysis," which is central to the creation and disclosure stages,SBOM This section outlines the implementation details and points to note based on the use of the tools.SBOMThis is a crucial process that affects the comprehensiveness and accuracy of the vulnerability assessment, and serves as a basis for decisions in subsequent vulnerability management, licensing management, and explanations to regulatory authorities and customers.

5.1.1. [Procedure] Flowchart for Component Analysis

Component analysis in these guidelines should be conducted primarily according to the following procedure.

1.SBOMClarification of the target and scope of creation (clarification of target software and deliverables)

SBOMBefore creation, clearly define the target software and its scope of application. For specific countermeasures, see below.4.1.SBOMRefer to "Clarification of Scope" and implement accordingly. And. Third partySBOMIf there is [5.2.4.Provided by a third partySBOMSee [Notes on importing].

2. Determining the working environment to be used for component analysis

SBOMTo perform the creation, decide whether to use a development environment, build environment, or production-equivalent environment. The components and version information detected may differ depending on the environment selected.SBOMBased on the purpose of use (pre-market submission, post-market management, internal management, etc.), an appropriate environment will be selected from the perspectives of reproducibility and validity.

3.SBOMTool configuration and component analysis execution

ImplementedSBOMBefore running the scan of the target software, you must first confirm and configure the tool's component analysis method, target directories, exclusion conditions, and package manager integration settings.GUIorCLIBecause the execution method differs depending on the method, please refer to the instruction manual orREADMERefer to the relevant documents and define the information according to the standardized procedures within the organization.

4. Checking the results

After component analysis,SBOMDon't judge solely on the fact that a file was generated; check the tool's analysis logs and warning/error outputs. Also, verify that the tool was not interrupted or skipped midway, and that internal processing was completed successfully. Even if it appears to have completed successfully on the surface, check whether any processing was skipped due to errors.

5.1.2. [Confirmation] How to check the analysis results

It is necessary to implement a systematic verification process, assuming that component analysis results may contain false positives or missed detections. Symbolic links, runtime libraries, components located in deep hierarchies, and control system-specific components used only in specific fields are particularly prone to being missed. Furthermore, it should be noted that even if a component itself is detected, its version information may be incorrectly identified.
When verifying the results, it is effective to evaluate the validity of the detection results by comparing them with configuration information managed by the package manager, detailed design documents, build settings, etc. If a package manager is used, a dependency tree can be used to verify components within the scope of management with relatively high reliability.

5.1.3. [Correction] Manual correction and handling of unspecified components

SBOMNot found in the tool's databaseOSSFor components that cannot be identified, it may be necessary to manually correct or add information using the tool's console, etc. In such cases, if any unclear information remains, organize that fact,SBOMIt needs to be explicitly managed above. SBOMRegulatory authorities, healthcare institutions,OEM/ODMWhen disclosing information to joint development partners, etc., sharing information that has not been fully identified as unspecified components ensures transparency and prevents excessive misunderstandings.

5.1.4. [Operation] Considerations for verification effort and accuracy

The thorough examination of component analysis results is essentially a manual process, and verifying sub-tier components and third-party components, in particular, can require considerable effort. On the other hand, guaranteeing that no components will be overlooked is not always practical. Therefore, it is important to define in advance the scope of the verification process, taking into account the importance of the software, security risks, complexity of dependencies, provider, and update frequency.SBOM The tool's output is compared with dependency information, design documents, build settings, and a list of libraries used, all managed by the package manager, to evaluate whether the analysis results accurately reflect the actual software configuration.

5.1.5. [Operation] Impact of analysis environment and settings

The analysis results will vary depending on the environment in which they are performed (development environment, build environment, production-equivalent environment, etc.),SBOMThis may vary depending on the tool's settings. For example, if analysis is performed in a development environment, libraries not included in the product may be detected. Therefore, it is important to consider which environment to use.SBOMIt is necessary to clearly define the criteria for creation before conducting the analysis. Furthermore, depending on the tool's repository settings and analysis parameters, SBOMThe above representation of component configuration and dependencies may differ from the actual software configuration.

Therefore, the validity of the settings should be checked during the initial setup.

5.1.6. [Operation] Operation based on the use of SBOM tools

SBOMBy using the tool, manual component definition andSBOMCompared to creation, a reduction in workload can be expected.OSSIn products containing, SBOMComponent analysis based on the tool andSBOMCreating and managing it is a practical and effective method. Therefore, this guidelineSBOMAssuming the use of toolsSBOMThe creation and management of the system shall be carried out.

5.2. Creating an SBOM

SBOMThis is not merely a deliverable, but important management information aimed at regulatory compliance, vulnerability management, and ensuring supply chain transparency. Therefore, during the creation phase, requirements definitions are made based on the recipients and regulatory demands, and accuracy and comprehensiveness are fully ensured.SBOMIt is required to generate [something].

5.2.1. Basic Concepts for Creating an SBOM

SBOMIt is created based on the information of the analyzed software components. Prior to creation, SBOMThe requirements, such as the items to be included, the format to be used, the version of the format, and the output file format, must be determined in advance. In regulations and requirements,SBOMBecause the format and required items may be specified, the format should be determined by comprehensively considering regulations and guidance, customer and supplier requirements, and the company's own operational policies.

5.2.2. Points to note regarding names and additional information within SBOM

SBOMIn addition to component information, it also includes project name and product identification information,SBOMThis may include information set in the tool.SBOMIt is important to consider whether the content is easy for users to understand and can be used effectively.
In particular, from the development stageSBOMIf you are using a tool for component management, project names and version notations that have been used only within the company may remain unchanged.SBOMIt may be shared externally as such.SBOMIn order to prevent rework and inquiries after sharing,SBOMIt is desirable to set names and identification information from the user's perspective.

5.2.3. Actions to be taken when creating an SBOM

SBOMThe main steps involved in the creation process are as follows: 1.

Determining the creation policy based on the requirements.

Based on the requirements set by the service provider or our own company,SPDXorCycloneDXformat, format

Determine the version and output file format.

2.SBOMUse the tool to satisfy the requirements.SBOMCreate

According to the determined format, version, and output file format,SBOMUsing the tool SBOMGenerates.

5.2.4. Precautions when importing SBOM provided by a third party

Third-party software andOSSIf you are using components provided by third parties such as the community,SBOMIt may be made public or made available. Provided by a third party.SBOMBy utilizing this,SBOMThis not only streamlines the creation process, but also allows for improved accuracy and refinement of component analysis results conducted in-house.

On the other hand, provided by a third partySBOMImportSBOMWhen creating a system, inconsistencies may arise between the system and the actual product configuration due to differences in code modifications and build conditions within your organization. As a result, incorrect information may be generated.SBOMBecause of the risks described, it is essential to verify the consistency with your company's actual implementation after importing and make corrections as necessary.

5.3. Disclosure of SBOM

5.3.1. Baseline in Disclosure

SBOMThis information enhances transparency regarding the cybersecurity of medical devices and provides fundamental information for medical institutions to understand vulnerabilities in their own environments and the risks of devices becoming legacy medical devices. Therefore, manufacturers and distributors should:SBOMWe will position this as one of our customer-facing security documents and establish a system to provide it to medical institutions, etc., primarily during the introduction of medical devices and when important software changes are made. At the very least, throughout the product's support period, In response to requests from medical institutions, etc.SBOMThe ability to present, This will serve as the baseline for these guidelines.

Furthermore, it will be provided to medical institutions, etc.SBOMThis document is primarily intended to support vulnerability management and security impact assessment, and does not represent a definitive legal opinion regarding the terms of use, redistribution conditions, or licensing obligations of the software.SBOMThe license information included is for reference purposes only for license management and risk assessment.OSSThe formal terms of use and obligations for each component, including those mentioned above, shall be determined based on the license documents, agreements, and other official documents accompanying the software.

5.3.2. Disclosure Level

Used for applications to regulatory authorities, etc.SBOMThe principle is to include all component information that is known. For healthcare institutions, based on an agreement with the healthcare institution, the information included is limited to what is necessary for the risk management of that institution.SBOMTo provide.

For medical institutionsSBOMThis includes at least the mainOSS/OTSRegarding the component, supply

Yard name, component name, version, and other unique identifiers,NTIAThe basic principle is to include configuration management information equivalent to the minimum essential elements. Furthermore, to the extent that there are no technical or contractual obstacles, the internal structure of the software and general-purpose information will be included according to the needs and agreements of the medical institution.OSIt is desirable to disclose as broadly as possible the components that are included as standard.

5.3.3. How to provide

SBOMFor the time being, the method of providing this will be:MDS2Manual operations are also permitted, such as attaching them to customer security documents or providing electronic files upon individual request. Manufacturers and distributors are permitted to:SBOMThe basic principle is to provide the relevant information (binary or source) by the time of delivery of the software, and at the very least, to make it available to customers by the time of product release. In the medium to long term, this will be provided via a website or an authenticated portal.APIBy utilizing a standardized distribution mechanism using such methods, it is possible to automatically detect and acquire data in a manner that enables automated detection and acquisition.SBOMIt is desirable to provide this.

In the future,SBOMagainstVEX(Vulnerability Exploitability eXchangeBy combining formats such as those mentioned above, and providing information on whether or not each vulnerability has an impact on the medical device, it is believed that the prioritization and response decisions of medical institutions can be made even more efficient.SBOMSince this information details the configuration of medical devices, it must be appropriately protected as confidential information based on memorandums of understanding, contracts, etc., and measures must be taken to mitigate the increased cyber risk due to leakage to third parties, such as using communication methods equipped with encryption and authentication.

6. Actions to be taken during the SBOM operation and management phase

6.1. SBOM Management

6.1.1. Administrative positioning of SBOM

SBOMThis is not a static document created once and then forgotten, but rather information that needs to be managed as it is updated in accordance with the release and updates of medical device software. Manufacturers and distributors must:JIS T 2304 Based on the software configuration management process, as neededJIS T 81001-5-1Based on the requirements of the software configuration management process,SBOMThe creation, updating, and storage of these files need to be integrated into the existing configuration management and change management framework.

6.1.2. SBOM Update and Change Management

Manufacturers and distributors, for each productSBOMIn addition to managing them individually,SBOMThe component information included in the "SBOMComponent repository (master database) "to construct, It is desirable to incorporate this into internal configuration management. Also, development contractors andOEM/ODMI previously obtained it from a third-party vendor.SBOMIt is important not only to store them individually, but also to integrate them into this repository so that components commonly used across all of the company's products can be understood across the board. This will allow for the identification of vulnerabilities and other issues related to components used in common across multiple products.End of Life (EOL)/EOSIt is now possible to update license information in bulk, and for each product SBOM Compared to individually inspecting each item, this method allows for efficient risk assessment while minimizing omissions and redundancies. Furthermore,SBOMWhen operating a component repository, it is necessary to absorb variations in the notation of component names and supplier names, and multipleSBOMIt is effective to establish a "normalization" mechanism that allows for consistent treatment across different systems. For example, in the United StatesMITREIn a white paper on medical devices, severalSBOMA central alias database for handling names included in a unified manner (Central Alias DatabaseThe use of such technologies has been proposed.

From an update management perspective, with software revisions (new releases, patches, feature additions, etc.), the configuration of the release must be reflected.SBOMMaintainWhich product/version corresponds to whichSBOMThis corresponds The term "ka" needs to be managed in a traceable manner.SBOMBy making the completion of updates one of the conditions for software release approval, it is possible to prevent omissions and discrepancies in configuration information.SBOMThe component information described therein is vulnerability information andEnd of Life (EOL)/EOSBecause this information is used in conjunction with other data to support risk assessments and upgrade plans, it is crucial that it is consistently updated as part of the daily change management process.

Furthermore, due to limitations of the generation tools and deficiencies in the information obtained from suppliers,SBOMIf any errors or omissions are found in the description, regardless of whether the software has been revised or not.SBOMCorrect the error itself and record the circumstances of the correction in the change management record.

6.1.3. Storage and Information Provision of SBOMs

Regarding storage,SBOMThis includes design records (design history, etc.) for medical devices, risk management files, and post-market safety management.GVPRecords of sales launch (commercial release) and other related information must be retained for the duration of the product's support period and as required by applicable laws and regulations. End of product life (End of Life (EOL)), End of support (EOSAt lifecycle milestones such as those mentioned above, we will implement our own development in accordance with the Ministry of Health, Labour and Welfare notifications "Regarding the Revision of the Guidelines for the Introduction of Cybersecurity for Medical Devices" and "Regarding the Provision of Information Related to Cybersecurity Measures for Medical Devices."OSS-Components of commercially available software, etc.SBOMIt is necessary to build a system that can be presented in this way and that can also be used for managing legacy medical devices.

For practical use,SBOMThis manages configuration management information such as product model number and software version in correspondence with the product.Quickly track which components are included in which product/version.
It is essential to be able to track this information. This will allow us to identify potentially affected products and model numbers in the event of vulnerability disclosure or end of support for a component, and to provide necessary updates and information.

In addition,SBOMRecords regarding the creation, updating, and use of documents (creation/revision history, review/approval status, etc.) are managed within existing document and record management processes and change management processes, and are subject to internal audits and regulatory inspections. QMS Maintain a state that can be explained during conformity assessments, etc.

moreover,SBOMTo prevent information tampering and impersonation, and to allow recipients to verify its origin and authenticity, to the extent possible,SBOMIt is desirable to establish a system that allows for the assignment of hash values and digital signatures, and verification using existing code signing and public key infrastructure. CISAPublished by SBOMEven in framing documents,SBOMIn the ecosystem, the authorSBOMIt is considered important to ensure authenticity and integrity by having users digitally sign documents and verify those signatures.

6.2. Vulnerability Management Based on SBOM

SBOMThis is an information infrastructure for checking whether libraries and components included in software have known vulnerabilities, and for quickly identifying the scope of impact if problems are found. Medical device software includes numerous components, and vulnerability information may be made public for each of them. SBOMThis helps identify the scope of impact in the vulnerability management process by understanding which components are included in which versions of which products.

In these guidelines,SBOMVulnerability management using [this method] will generally be implemented following the steps outlined below.

1.Collection of vulnerability information andSBOMVerification

2.Impact assessment and prioritization (triage)

3.Determination and implementation of response policies and provision of information

4.Record of results

6.2.1. Collection of vulnerability information and matching with SBOM

Manufacturers and distributors, etc.JVNorNVDVulnerability information databases such as software vendor security advisories and information sharing and analysis organizations (ISAO/ISACWe continuously collect information from sources such as [sources] to identify potential vulnerabilities related to our products. The component names, versions, and other unique identifiers included in the collected information are used to identify vulnerabilities in our products.SBOMThis is then cross-referenced to identify potentially affected products and versions.SBOMAdding other unique identifiers and hash values as much as possible during the creation phase is useful for improving the accuracy and efficiency of automatic matching.

moreover,SBOMThere are platforms and tools that automatically cross-reference (import) vulnerability information with vulnerability databases and continuously provide relevant vulnerability information and alerts. Therefore, it is effective to appropriately utilize these to automate and streamline the cross-referencing process.

6.2.2. Vulnerability Impact Assessment and Prioritization (Triage)

SBOMThe number of vulnerabilities identified through analysis tools can range from a small number to a large number, so it is important to prioritize responses from the perspective of medical device risk management. When prioritizing (triage), the following points should be considered, for example:

- The publicly available severity levels (e.g., Common Vulnerability Scoring System)CVSS)Score)
- Potential exploitation (whether attacks have already been observed, or if they are listed in the known exploit vulnerability catalog)PoC(Presence or absence, etc.)
- The role of the component in the medical device (its impact on safety and basic performance)
- Exposure status in actual usage environments (e.g., whether it is reachable via network)
- The presence or absence of workarounds and alternatives (e.g., whether it can be mitigated by changing settings or network isolation) is considered. Based on these perspectives, we evaluate whether the vulnerability can actually be exploited in the medical device and to what extent it would affect safety, basic performance, and the clinical environment if it were exploited. (for example,CVSS (For example, scores such as these should be re-evaluated in accordance with the medical device usage environment, aligned with each company's risk management, and used to determine acceptability.)

SBOMRegarding the publicly disclosed vulnerabilities in the components listed, in principle, all will be subject to evaluation. However, if, based on the risk assessment from the above perspectives, it is determined that the overall risk to the device is sufficiently low, it may be decided to accept the risk (monitoring only) without taking proactive corrective action. Based on these considerations, things that require immediate attention" "We will address this in the next scheduled update." Prioritize at a realistic level, such as "things that need to be monitored" or "things that will only be monitored as a form of risk acceptance."SBOMRather than assuming a uniform response to all vulnerabilities output by the tool, a reasonable scope of response will be determined in accordance with the risk management framework for medical devices.

6.2.3. Determination and implementation of response policies and provision of information

For vulnerabilities that require attention, we will implement software updates, patching, and component updates.

We will consider and implement specific measures such as replacing components, changing settings, and reviewing the network configuration.OSFor vulnerabilities arising from components managed by third parties, such as cloud services, it is effective to address them by organizing policies such as fixes by the manufacturer/distributor, operational measures by the healthcare institution, risk acceptance, and continuous monitoring.

For critical vulnerabilities that could affect patient safety or the continuity of medical care, coordinated vulnerability disclosure (CVDIn accordance with the framework of), it is necessary to provide a security advisory that includes the affected products and versions, an overview and impact, recommended interim and permanent measures, etc.SBOMBy using this method to clearly identify the affected products and versions, it is possible to support healthcare institutions in prioritizing and deciding on appropriate responses. The vulnerability information that formed the basis of the response, the results of the risk assessment, and the selected countermeasures shall be recorded, and the handling and storage methods as quality records shall be in accordance with the following section.

6.2.4. Recording of Results and Reflection in SBOM

If any changes occur to the software configuration, the software will be updated accordingly each time.SBOMCreate and maintain the latest version. For example, if vulnerability fixes such as replacing vulnerable components are implemented, the result will beSBOMThis will also be reflected in the following. At the same time, a series of information will be recorded, from the receipt of vulnerability information to evaluation, the content and basis of the risk assessment, the decision on whether or not to take action and the content of the approval, to the content and results of the measures taken, and this information will be used in design records, risk management files, and post-market safety management (GVPLike the records mentioned above, these records need to be stored and managed.

These records will be retained for the period required by relevant laws and standards, and will be used in internal audits and regulatory inspections to identify "which products or devices were affected" by a particular vulnerability. "How to evaluate, and what kind of response It is necessary to maintain a system that allows for tracking, in chronological order, whether or not measures were taken.SBOMBy linking and managing configuration management information such as product model numbers and software versions, as well as vulnerability response records, efficient and consistent explanations can be provided for future inquiries and additional countermeasures, including for legacy medical devices. Furthermore, even if the evaluation results indicate "no impact" or "impact only under specific conditions," the reasons for this decision are recorded as quality records and, if necessary, can be used in the future.VEXIn formats such asSBOMIt is desirable to be able to manage and provide this information in a linked manner.

6.2.5. License and Lifecycle Management Based on SBOM

In license management and lifecycle management,SBOMThis will be used to identify the software components included in the medical device, and this information will be shared with legal and quality assurance departments (license information, contract terms, etc.).End of Life (EOL)/EOSThis information will be used in conjunction with other relevant data. Medical devices are often used over long periods, and component license violations or end-of-support issues, in addition to legal risks, also impact cybersecurity risks and the continuity of product supply.

This guideline1This version will be provided to medical institutions, etc.SBOMAs the smallest elementNTIAIt employs minimal elements, and license information is not a required element.SBOMThis is internal license management.

It is useful as "component inventory information" in this context, and therefore for internal use.SBOM(or SBOMIt is desirable to manage the component repository in conjunction with license information and evidence (source of acquisition, license document, whether modifications were made, distribution method, etc.). The determination of license compliance is as follows:OSSThis will be carried out based on a separate license management process consisting of management regulations, license review records, individual license documents and agreements, etc.

In addition,SBOMWhen including license information, simplifications such as exception clauses and dual licenses may lead to misunderstandings, so please be mindful of whether or not it is provided to external parties and the level of detail (e.g.,SPDXHandling of identifiers) ,review· approval,SBOMMain unit andOSSIt is desirable for organizations to establish a policy in advance regarding the separation of license statements and other related documents.

From a lifecycle management perspective, manufacturers and distributors, based on contracts and licenses with suppliers, should determine the end of the product lifecycle for each component.End of Life (EOL)) and end of support (EOS) Gather information about our companySBOMIt is required to incorporate these components into the component repository. This will allow for a comprehensive understanding of which products and versions may be affected by the end of support or discontinuation of development of which components, enabling early consideration of whether upgrades, replacements, or complementary measures are necessary.

The results of these license management and lifecycle management (evaluation of license compliance, content of corrective actions,End of Life (EOL)/EOSThe response policies, etc., based on the above, are based on risk management files, design records, and post-market safety management.GVPWhile ensuring consistency with the records of,QMSThis will be kept as a quality record.SBOM If component information is systematically organized, the insights gained can be fed back into component selection for future products and software updates. T-Planet(Total Product Life Cycle) This contributes to maintaining and improving overall legal compliance and cybersecurity standards.

6.3. SBOM Requirements in Procurement and Contracting

SBOMIn order to continuously and consistently improve our own products,SBOMNot only creating it, but also software suppliers and development contractors,OEM/ODMIn the contract and procurement specifications with the previous party, SBOMIt is important to include the provision and updating of software as a clear requirement. Even for software implemented by a third party through development outsourcing, etc., the finalSBOMThe manufacturer/distributor is responsible for the accuracy and comprehensiveness of the information, therefore, at the contract stage...SBOMIt is necessary to share specific expectations regarding this matter.

In practice,RFPIn basic agreements, individual transaction agreements, etc., at least the followingSBOMIt would be a good idea to review the relevant clauses.

- **Provided format and items:**NTIAMachine-readable based on minimal elementsSBOM(SPDXor CycloneDXProvide the following information: In addition to basic items such as supplier name, component name, version, other unique identifiers, relationship, creator, and timestamp, include component hash and license type where possible.
- **Availability and update frequency:**By the time of delivery of the software or componentsSBOMTo provide: When version upgrades, patches are released, or significant configuration changes are made, further support will be provided.SBOMTo provide it within a reasonable timeframe (e.g., within X days of release).

- **Providing lifecycle information:**End of support dates for each component (EOS) and end of product life (End of Life (EOL)If the information is publicly available,SBOMAlternatively, provide it as a separate document. End of Life (EOL)/EOSIf there are any changes to the information, notification will be given through the previously agreed-upon means (email notification, portal posting, etc.).

- **Vulnerability notification andSBOMupdate:**Critical vulnerabilities (e.g.:CVSS v3.1inHighIf any of the above is discovered in a component, the recipient of that component (including the client of this agreement) will be provided with information on the vulnerability, the scope of impact, recommended countermeasures, and any necessary actions.SBOMThey are responsible for notifying updates.

By specifying these conditions in the contract, the entire software supply chain, including third-party components and outsourced development portions,SBOMIt becomes easier to obtain information without any omissions. Also, information obtained from multiple suppliers...SBOMofSBOMBy integrating into a component repository and normalizing names and version notations, we can achieve cross-product license management, vulnerability management, and more.End of Life (EOL)/EOSManagement can be carried out efficiently.

Manufacturers and distributors should collaborate with their internal procurement and legal departments.SBOMIt is necessary to reflect the requirements in the company's standard contract forms and procurement policies.

literature

1) FDA, Cybersecurity in Medical Devices: Quality Management System Considerations and Content of Premarket Submissions ｜2026

2) NTIA, Framing Software Component Transparency: Establishing a Common Software Bill of Materials (SBOM) Second Edition ｜2021

3) NTIA, The Minimum Elements for a Software Bill of Materials (SBOM) ｜2021

4) ANSI/NEMA, HN 1-2019 Manufacturer Disclosure Statement for Medical Device Security ｜ 2019

5) CISA, 2025 Minimum Elements for a Software Bill of Materials (SBOM) Public Comment Draft ｜2025

6) IMDRF, N70 Principles and Practices for the Cybersecurity of Legacy Medical Devices ｜2023

7) IMDRF, N73 Principles and Practices for Software Bill of Materials for Medical Device Cybersecurity ｜ 2023

8) Japanese Standards Association, JIS T 2304:2017 Medical device software - Software lifecycle process | 2017

IEC, IEC 62304:2006+A1:2015 Medical device software — Software life cycle processes

9) Japanese Standards Association, JIS T 81001-5-1:2023 Safety, effectiveness and security of health software and health IT systems - Part 5-1 | 2023: Security - Activities in the product lifecycle

IEC, IEC 81001-5-1:2021 Health software and health IT systems safety, effectiveness and security Part 5-1: Security — Activities in the product life cycle ｜2021

10) ISO/IEC, IEC/TS 81001-2-2:2025: Health software and health IT systems safety, effectiveness and security Part 2-2: Guidance for the implementation, disclosure and communication of security needs, risks and controls | 2025

11) ISO/IEC, ISO/IEC 5962:2021 Information technology - SPDX® Specification V2.2.1 ｜ 2021

12) MITRE, DATA NORMALIZATION CHALLENGES AND MITIGATIONS IN SOFTWARE BILL OF MATERIALS (SBOM) PROCESSING ｜2024

13) ENISA SBOM LANDSCAPE ANALYSIS ｜2025

14) CISA, Framing Software Component Transparency: Establishing a Common Software Bill of Materials (SBOM) Third Edition ｜2024

15) Guidelines for the Secure Management of Medical Information Systems (Security Management Guidelines), Version 6.0

16) Ministry of Health, Labour and Welfare, Guidelines for the Introduction of Cybersecurity for Medical Devices (2nd Edition) | 2023

17) Ministry of Health, Labour and Welfare, Guidelines for Ensuring Cybersecurity of Medical Devices in Medical Institutions | 2023

18) Ministry of Health, Labour and Welfare administrative notice Regarding the "Guidelines for the Secure Management of Medical Information Systems" "Cybersecurity Measures Checklist for Medical Institutions" and "Incidents in Medical Information Systems, etc."

"About the Time Response Flowchart" | 2021/10

19) Ministry of Health, Labour and Welfare Notification "Regarding the Strengthening of Cybersecurity Measures in Medical Institutions, etc." | 2018/10/29

20) Medical Care Act / Enforcement Regulations of the Medical Care Act...Safe management of medical information and medical devices

21) Ministry of Health, Labour and Welfare Notification "Regarding the Ensuring of Cybersecurity in Medical Devices" | 2015/04/28

22) Ministry of Health, Labour and Welfare Notification "Guidance on Ensuring Cybersecurity for Medical Devices" | 2018/07/24

23) Ministry of Health, Labour and Welfare Notice: "Regarding the Publication of IMDRF Guidance" | 2020/05/13

24) Ministry of Health, Labour and Welfare Notification: "Guidelines for Ensuring and Thoroughly Implementing Cybersecurity for Medical Devices" | 2021/12/24

25) IMDRF, Principles and Practices for Medical Device Cybersecurity｜2020/03/18

26) Ministry of Health, Labour and Welfare Administrative Notice: "Regarding Cyberattacks Using Ransomware Targeting Medical Institutions (Warning)" | 2021/06/28

27) Ministerial Ordinance concerning standards for post-market safety management of pharmaceuticals, quasi-drugs, cosmetics, medical devices, and regenerative medicine products

28) Ministry of Economy, Trade and Industry, Guidelines for the Introduction of SBOM (Software Bill of Materials) for Software Management, ver 2.0 | 2024/08/29

Terminology (in alphabetical order)

fifty Alphabetical order	term	source
a	**update** Changes related to modifications, prevention, adaptations, or improvements to medical device software. Note 1: This stems from software maintenance activities as defined in ISO/IEC Note 2: 14764:2006. Updates include patches and configuration changes. Note 3: Changes related to adaptation and refinement were not present in the design specifications; these were software modifications. It is good.	IMDRF Guidance Translation (Japanese) twist (Guide for Manufacturers and Distributors) (From the book)
tree	**Common Vulnerability Scoring System (CVSS)** CVSS is an open and general-purpose evaluation method for information system vulnerabilities, providing a common evaluation method that is independent of the service provider. Using CVSS, the severity of vulnerabilities can be quantitatively compared under the same criteria. IPA Common Vulnerability Scoring System (CVSS) v3 Overview https://www.ipa.go.jp/security/vuln/scap/cvssv3.html FIRST Common Vulnerability Scoring System Version 4.0 https://www.first.org/cvss/v4-0/	(Guide for Manufacturers and Distributors) (From the book)
child	**attack** Destruction, exposure, alteration, invalidation, theft, unauthorized access to or attempt to use assets	JIS Q 27000:2019
	Computerized System Validation This refers to the validation of a "computerized system" that integrates "computer systems" and "business processes."	Quality control and supervision system (QMS) related computers Application of the software Validation regarding Also, electronic documents and Guide regarding record management Dance 2020
	component These are individual components or elements that make up the functionality of a system or software. These components function independently while also working in conjunction with other components to realize the overall system or application.	
difference	**Cybersecurity** Information and systems are protected from unauthorized activities (unauthorized access, use, disclosure, interruption, alteration, destruction, etc.), and risks related to confidentiality, integrity, and availability are maintained at an acceptable level throughout their lifecycle.	JIS T 81001-1:2022,
	End of Support (EOS) The point in the product lifecycle when the manufacturer ceases all support activities. Service support does not extend beyond this point.	IMDRF Guidance Translation (Japanese) More (for manufacturers and distributors) (From the manual)

fifty Alphabetical order	term	source
death	**Information Sharing and Analysis Organizations (ISAOs)** An organization established for the collection, analysis, sharing, and dissemination of cybersecurity-related information. By actively participating in ISAO, manufacturers and distributors can proactively address cybersecurity vulnerabilities and minimize their misuse through deployments that include communication and coordination with patients and users, thereby supporting businesses, the medical device community, and the medical and public health sectors. There is also an organization called Information Sharing and Analysis Centers (ISAC). As an international organization, H-ISAC (Health Information Sharing and Analysis Center) https://h-isac.org/ MedISAO (https://www.medisao.com/) exists. In Japan, One of the information-sharing organizations established by NISC (National Center of Incident Readiness and Strategy for Cybersecurity) is the Medical Scepter (Secretariat: Information Systems Division, Japan Medical Association). The Japan Federation of Medical Devices Manufacturers Associations (JFMDA) and the Japan Association of Healthcare and Welfare Information Systems Industries (JAHIS) participate as observers, and these member organizations and companies can utilize the cybersecurity information of the Medical Scepter.	(Guide for Manufacturers and Distributors) (Partially revised from the original text)
height	**vulnerability** Flaws or weaknesses in the design, implementation, operation, or management of a system that could be exploited to circumvent the system's security policies. Weaknesses in assets or controls that could be exploited by one or more threats.	JIS T 81001-1:2022 is based on JIS Q 27000:2019. (Guide for Manufacturers and Distributors) (From the book)
	Security Advisory The following information will be provided: - Vulnerabilities in other companies' products or general technologies that have a significant impact on our own products. - Capture and add information regarding vulnerabilities related to our company. - Information about vulnerabilities for which a patch module has not yet been created.	(Guide for Manufacturers and Distributors) (From the book)
	End of Life (EOL) This stage in a product's lifecycle involves ending sales of a product that has exceeded its specified end-of-life (EOL) period and implementing a formal EOL process (such as notifying users).	IMDRF Guidance Translation (Japanese) More (for manufacturers and distributors) (From the manual)
fire	**PSIRT (Product Security Incident Response Team)** An internal organizational function for addressing risks arising from vulnerabilities in products provided by the organization. An organization whose purpose is to address vulnerabilities in its own products and to manage and improve the security quality of its products. JPCERT/CChttps://www.jpcert.or.jp/research/psirtSF.html (Japan Computer Software Association, JPCERT/CC) Product Developer's Guide (IPA) https://www.ipa.go.jp/security/guide/vuln/rcu1hd000000a5enatt/guide_for_dev.pdf PSIRT Services Framework 1.0 Japanese Version https://www.first.org/standards/frameworks/psirts/FIRST_PSIRT_Services_Fra mework_v1.0_jp.pdf	(Guide for Manufacturers and Distributors) (From the book)
eye	**metadata** Data used to explain certain data	
the law of nature	**Reverse Engineering** This is a technical method that involves disassembling and analyzing existing products and software to reveal their structure, operating principles, design information, and manufacturing methods. Unlike conventional engineering, which is a "forward" process of creating a product from a design drawing, this method employs a "reverse" approach, extracting design information by working backward from the finished product, hence the name.	
re	**Legacy medical devices** Medical devices that cannot be protected against current cybersecurity threats by reasonable means such as updates or supplementary measures, regardless of the number of years since they were first sold.	IMDRF Guidance Translation (Japanese) Further modifications (manufacturing and sales (From the sales manual)

fifty Alphabetical order	term	source
C	**CI/CD** In the concept of automation processes in software development, "Continuous Integration (Continuing) (Continuous Integration) " and "Continuous Delivery/Continuous Deployment (Continuous (Continuous delivery/Continuous deployment) " Continuous Integration (CI): A system that frequently and automatically integrates (merges) changes made by developers into software, and then runs builds and automated tests. This enables early detection of bugs and reduces rework during integration. Continuous Delivery (CD): A process that automatically delivers software integrated by CI to staging and production environments according to quality standards. It minimizes manual intervention and supports safe and rapid releases. In medical device development, CI/CD can streamline software change tracking, quality assurance, and the automatic generation and updating of SBOMs.	
	CPE (Common Platform Enumeration: List of Common Platforms) A standardized naming convention for uniquely identifying IT products such as hardware, operating systems, and applications in information systems. It is a component of SCAP (Security Content Automation Protocol), a standard managed by NIST (National Institute of Standards and Technology) for automating vulnerability management, measurement, and evaluation. Described in URI format, it is widely used in vulnerability databases such as NVD to identify products affected by vulnerabilities.	SCAP (Security Content) (Automation Protocol)
	CVE (Common Vulnerabilities and Exposures)) This is an international system that assigns a unique identification number (CVE-ID) to each individual vulnerability in software and hardware. It is primarily managed and operated by the US non-profit organization MITRE. The format is "CVE-year-number" (e.g., CVE-2021-44228). A CVE-ID is assigned to a vulnerability. CVE-IDs allow for accurate reference and matching of the same vulnerability across different vendors, tools, and databases. CVE-IDs are issued by an organization with the authority to assign them, known as the CNA (CVE Numbering Authority).	
J	**JVN (Japan Vulnerability Notes)** This is a vulnerability countermeasure information portal site for Japan, jointly operated by JPCERT/CC (JPCERT Coordination Center) and IPA (Information-technology Promotion Agency, Japan). It publishes security vulnerability information and countermeasures for software and hardware both domestically and internationally. Related services such as JVN iPedia (vulnerability countermeasure information database) and MyJVN (vulnerability countermeasure information collection tool) are also provided.	
M	**MDS2 (Manufacturer's Security Disclosure Statement for Medical Devices)**Manufacturer Disclosure Statement for Medical Device Security (MDS2) is a security declaration document that provides a format for medical device manufacturers to disclose security-related information to healthcare providers (medical institutions). It is a template document created and published by HIMSS in December 2004 in the United States to comply with security regulations under the HIPAA Act, with the latest version released in 2019. MDS2 has become established as an information sharing tool between medical institutions and manufacturers/distributors and is widely used. A Japanese translation is available on the website of the Japan Medical Imaging Systems Industry Association (JIRA): https://www.jira-net.or.jp/publishing/security.html	ANSI/NEMA HN 1-2019 (Guide for Manufacturers and Distributors) (From the book)
N	**NVD (National Vulnerability Database))** This is the world's largest vulnerability information database, managed and operated by NIST (National Institute of Standards and Technology). Based on CVE-IDs, it collects and organizes vulnerability information, and publishes it with detailed technical information such as severity scores from CVSS (Common Vulnerability Scoring System), affected product information from CPE (Common Product Exposure), and vulnerability classifications from CWE (Common Weapon Exposure Classification). As a SCAP-compliant repository, it serves as the foundation for automated vulnerability management and compliance verification.	

fifty Alphabetical order	term	source
P	**purl (Package URL)** A standardized URL-based identifier used to uniquely identify packages and libraries across various software package management ecosystems (such as npm, Maven, PyPI, deb, rpm, and Docker). While CPE is well-suited for product and platform-level identification, purl specializes in OSS package (component) level identification and is widely used in SBOM (Software Bill of Materials) and vulnerability management.	
S	**SWID (Software Identification Tag))** This is XML-formatted metadata uniquely generated for each software product, and is an international standard defined by ISO/IEC 19770-2 for the purpose of improving the efficiency of software asset management (SAM). It structures and describes information such as the software name, vendor, version, and edition, enabling the automation of the identification, detection, and management of installed software. In Japan, an equivalent standard is adopted as JIS X 0164-2:2018. It is also used as a component of SBOM.	JIS X 0164-2:2018