SOFTWARE BILL OF MATERIALS FOR AI

Minimum Elements

Federal Office
for Information Security

G7 FRANCE ÉVIAN 2026

AGENZIA PER LA CYBERSICUREZZA NAZIONALE
ACN
SAPIENTIA ADIUVAT

Communications Security Establishment Canada

Centre de la sécurité des télécommunications Canada

CYBERSECURITY & INFRASTRUCTURE SECURITY AGENCY
CISA

RÉPUBLIQUE FRANÇAISE
Liberté
Égalité
Fraternité

AGENCE NATIONALE DE LA SÉCURITÉ DES SYSTÈMES D'INFORMATION
ANSSI

国家サイバー統括室
National Cybersecurity Office

National Cyber Security Centre
a part of GCHQ

With the participation of the European Commission

EXECUTIVE SUMMARY

Accessing information on the supply chain of an artificial intelligence (AI) system, as well as its individual components and dependencies, is critical to strengthen cybersecurity of AI. Transparency and knowledge about AI system composition fosters vulnerability management and supports cybersecurity risk management.

This document provides actionable guidelines for public and private sector stakeholders on what is reasonable to expect in a Software Bill of Materials (SBOM) for AI, and to improve transparency and cybersecurity along the AI supply chain. It builds on the shared vision of SBOM for AI published by the G7 Cybersecurity Working Group in June 2025 and provides useful practical recommendations on which minimum elements SBOMs for AI should include. As such, this document is meant to cover a minimum set of elements identified and agreed on by experts within the G7 Cybersecurity Working Group. These minimum elements are not mandatory; do not create requirements, standards, or legislation; and are open to further refinements to keep pace with technological development and evolution of legal or policy frameworks within G7 members. Additionally, in some jurisdictions, certain elements proposed in this document may already be, or may be expected to be, addressed through legal requirements and obligations, or through existing or forthcoming standards.

This document is jointly published by Germany's Federal Office for Information Security (BSI), Italy's National Cybersecurity Agency (ACN), France's National Cybersecurity Agency (ANSSI), Canada's Communications Security Establishment (CSE), the US Cybersecurity and Infrastructure Security Agency (CISA), UK's National Cyber Security Centre (NCSC) and Japan's National Cybersecurity Office (NCO), in collaboration with the EU Commission.

This document has been written thanks to the support provided by the G7 Presidencies of Canada (2025) and France (2026), under the work stream "Artificial Intelligence: SBOM for AI" co-led by Italy (ACN) and Germany (BSI).

Table of Contents

1. Introduction

In June 2025, the G7 Cybersecurity Working Group published a shared vision for Software Bill of Materials for artificial intelligence (SBOM for AI)[1] defining the notion, goals, benefits, and properties of an SBOM for AI, with a view of supporting cybersecurity along the AI supply chain by increasing transparency about AI system components. The document states that an SBOM for AI should define the type of information to capture, providing clarity to stakeholders on how to achieve transparency across the AI supply chain. It recommended a set of high-level, illustrative minimum elements, listing them as clusters.

The document also recommended additional work by G7 experts to further define the clusters and specify which detailed information about AI system components should be included in each cluster.

The guidelines listed below are the result of that work. The minimum elements complement the shared vision of SBOM for AI with shared practical recommendations on which minimum elements an SBOM for AI should include to facilitate adoption and implementation by public and private sector stakeholders. This is the first G7 guidance document on SBOM for AI, and the outcome of joint work completed by cybersecurity and AI experts within the G7 Cybersecurity Working Group between August 2025 and February 2026.

AI systems are also software systems. Therefore, SBOMs still remain valid for AI systems. The minimum elements in an SBOM for AI are in addition to the general SBOM minimum elements.

Why SBOM for AI

Drawing from the existing Software Bill of Materials (SBOM) concept, an SBOM for AI consists of a structured record, or inventory of details and supply chain relationships for the various components used in building an AI system. This structured record is divided into different clusters. Each cluster contains "elements", or information that captures the distinctive features of AI system components.

The goal of SBOM for AI is to help secure AI systems and supply chains through transparency and traceability of components and dependencies. Like SBOM, SBOMs for AI serve as an ingredient list, providing organizations with data they can use to ensure effective IT security processes.

What to include in an SBOM for AI: the minimum elements

The proposed minimum elements listed in this document are the product of G7 expert consensus and provide **actionable guidelines on how AI developers and deployers should implement an SBOM for AI** to improve transparency and cybersecurity along the supply chain, thereby contributing to AI governance.

An SBOM for AI structures information that is useful to **track vulnerabilities and weaknesses and reduce cybersecurity risks**.

---

[1] "A shared G7 vision on Software Bill of Materials for AI. Transparency and Cybersecurity along the AI Supply Chain" hereafter referred to as "Shared G7 Vision".

While the proposed minimum elements are not exhaustive, and additional clusters or elements may be necessary, depending on the industry, sector, or jurisdiction, it is suggested that an SBOM for AI includes these minimum elements. In this regard, the purpose of this document is to offer useful guidance, without creating new requirements, standards, legislation, or implementation details of an SBOM for AI, which are outside the scope of this work.

The list of proposed minimum elements is open for further expansion to keep pace with the rapid development of AI technology.

AI software supply chain security is becoming increasingly important, with several international initiatives underway. As one component of this effort, SBOM for AI contributes to strengthening AI supply chain security.

2. Clusters and Cluster Elements

This section depicts a set of high-level minimum elements for an SBOM for AI, which extends the information used for traditional SBOMs, listed as clusters. The clusters initially presented served as a basis for discussing the outline of the set of minimum elements (see Shared G7 Vision).

The G7 Cybersecurity Working Group has been carefully evaluating this initial clusters proposal. The resulting guidelines represents the outcome of a year-long collaborative effort among partners. It led to a co-created list of minimum elements through consensual decision-making achieved during virtual and hybrid meetings. Eventually, some clusters have been adjusted, some added and some removed.

Figure 1 provides an overview of the seven clusters described in the following sub-sections. Besides the names, descriptions and examples are given for each cluster element. Examples—where listed—are to be considered as illustrative and not exhaustive. Examples are meant to help the reader understand what type of information could be included in elements that are not self-explanatory. The Metadata cluster is shown first, as it contains information about the SBOM for AI itself. The remaining clusters follow in no particular order.

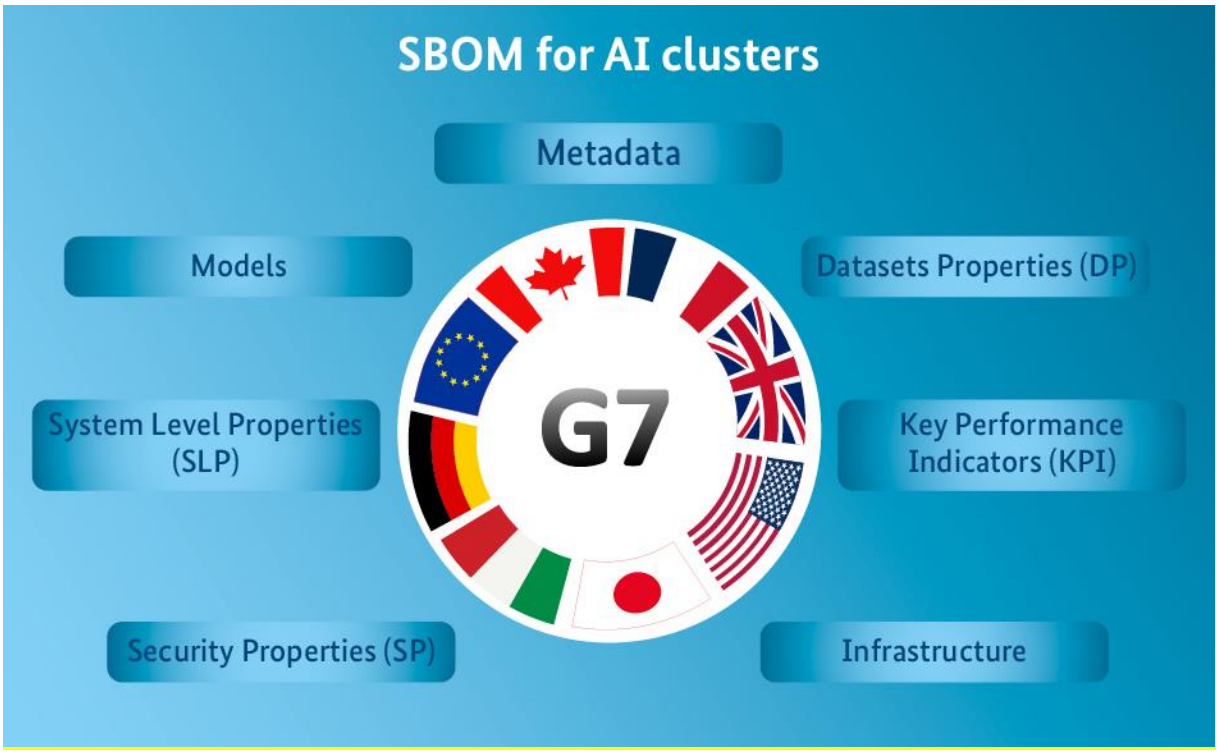

Figure 1: *Overview of the seven clusters presented in this section. Apart from the Metadata cluster, which contains information about the SBOM for AI itself, all clusters are equally important.*

2.1 Metadata Cluster Elements

This cluster is used to represent information related to the SBOM for AI itself, and not the individual components or sub-elements. Information related to individual SBOM for AI components or sub-elements will be detailed separately in the dedicated clusters.

The Metadata cluster contains the following elements:

- SBOM author
- SBOM version
- SBOM data format name
- SBOM data format version
- SBOM author signature
- SBOM tool name
- SBOM tool version
- SBOM generation context
- SBOM timestamp
- SBOM dependency relationship

Element	Description
SBOM author	The name of the entity that creates the SBOM data for the target component. The *SBOM author* element contains a string that identifies the entity that generated the SBOM for the target component. This element captures the entity operating the tool to generate the SBOM, not the tool itself. Inputs for this element should use full names and should not use acronyms (unless the tool's official name includes an acronym). The *SBOM author* element is distinct from the *Producer* element, which identifies the entity that created the target component. The *SBOM author* and the *Producer* element contents may be identical if the same entity created the target component and generated the SBOM. In cases where an entity that did not create the target component generates the SBOM for the target component, the element contents will be different.
SBOM version	Identifier designated by the *SBOM author* to specify a change in the SBOM document from a previously identified version or to indicate that it is the first version. The *SBOM version* element indicates a relationship with earlier iterations of an SBOM and signals that the *SBOM author* made changes to the previous iteration. The *SBOM version* tracks the SBOM component data for a component-name/component-version pair. The *SBOM version* element may use Semantic Versioning;[2] if it uses Semantic Versioning, the major version of a published SBOM following these minimum elements should be 1. If using an identifier such as a serial number to differentiate between versions of an SBOM, use of the identifier should conform to relevant standards (for example, RFC 9562 for serial numbers).[3] *SBOM authors* should use minor version and patch version to indicate changes to the content of the SBOM elements as appropriate. *SBOM authors* should update the version for an SBOM for a given component name-version pair when editing data about the target component.

[2] Preston-Werner, T. and The SemVer Team. Semantic Versioning Specification Version 2.0.0. June 18, 2013.
[3] Davis, K., Peabody, B., and P. Leach. Universally Unique IDentifiers (UUIDs), RFC 9562. DOI 10.17487/RFC9562. May 2024.

Element	Description
SBOM data format name	The name of the data format used to represent the SBOM data. The *SBOM data format name* element identifies the data format that the SBOM is in (in machine processable data format).
SBOM data format version	Identifier designated by the SBOM data format to specify the version of the data format. The *SBOM data format version* element identifies the version of the data format indicated in *SBOM data format name*. Versions declared to be deprecated by the organization(s) maintaining the data format should not be used.
SBOM author signature	A digital signature attributable to the *SBOM author*. The *SBOM author signature* element provides assurance that the claimed signatory signed the information and that the information was not modified after signature generation[4]. Digital signatures should use a signature algorithm approved for secure use, according to regulations or recommendations from relevant authorities, such as the National Institute of Standards and Technology (NIST) Digital Signature Standard (DSS), the International Organization for Standardization (ISO)/International Electrotechnical Commission (IEC) 14888-4:2024, or the European Union Agency for Cybersecurity (ENISA) Agreed Cryptographic Mechanisms.
SBOM tool name	The name of the tool used by the *SBOM author* to generate or amend the SBOM. The *SBOM tool name* element contains a string that identifies the software tool that the *SBOM author* used to generate the SBOM. Inputs for this element should use full names and should not use acronyms (unless the tool's official name includes an acronym).
SBOM tool version	Identifier for the version of the tool identified in the *SBOM tool name* element. The *SBOM tool version* element captures the tool version and allows an organization to identify a specific code delivery. If

[4] National Institute of Standards and Technology, Computer Security Resource Center. Projects: Digital Signatures.

Element	Description
	no version identifier is available, the *SBOM author* should indicate that the information is unknown.
SBOM generation context	The relative software lifecycle phase and data available at the time the *SBOM author* generated the SBOM. The *SBOM generation context* element conveys information about the component data documented in the SBOM. Component data may differ based on the phase of the software lifecycle during which the SBOM was generated. General software lifecycle references such as "before build," "build," and "after build," as well as more specific identifiers, can satisfy this element.[5] For example, an SBOM generated from source code could be identified as "before build," and binary analysis tools can generate an SBOM "after build."
SBOM timestamp	Record of the date and time of the most recent update to the SBOM data. The *SBOM timestamp* element identifies when the *SBOM author* last changed the SBOM data, either manually or using a software tool. Each version of an SBOM will reflect a new timestamp. The content of this element should adhere to RFC 9557.[6]
SBOM dependency relationship	The relationship between two software components, specifically noting that Software X includes Component Y or that Component A is largely derived from Component B. The *SBOM dependency relationship* element reflects how a given software component was included in the target software. The inclusion relationship ("includes" or "included in") supports the capability to build a dependency graph. In addition to inclusion, the dependency relationship should reflect how a given component may be largely derived from another piece of software, or that it is a descendant of another piece of software. This allows an SBOM to explicitly document backported or forked software.

[5] "Build" refers to compilation, interpretation, or another process that makes source code executable.

[6] Sharma, U. and C. Bormann. Date and Time on the Internet: Timestamps with Additional Information, RFC 9557. DOI 10.17487/RFC9557. April 2024.

2.2 System Level Properties (SLP) Cluster Elements

The SLP cluster contains elements that refer **to information on the AI system as a whole** (e.g., capturing system level information and AI system inner workings that are relevant to AI systems composed of multiple AI elements, such as classifiers, large language models (LLMs) or AI agents). This cluster also includes all software dependencies and frameworks used in the AI system as well as information about how AI system components interact and process user data. Elements that are in support or that have been used to deploy the system are included in the *cluster Infrastructure* below.

The SLP cluster contains the following elements:

- System name
- System components
- System producer
- System version
- System timestamp
- System data flow
- System data usage
- System input/output properties
- Intended application area

Element	Description	Examples
System name	The name assigned by the *System producer* to an AI system. The *System name* element identifies the AI system in a human-readable way. The *System producer* determines the AI system name. Data formats implementing the *System name* element should allow for multiple entries to capture alternate names (full names and not acronyms – unless the *System*'s official name includes an acronym).	
System components	This element captures the components that are included in the AI system.	• AI models included in the system; • databases; • other software tools and components.

Element	Description	Examples
System producer	Identifier used for the name of the entity that creates, defines, and identifies an AI system.	
System version	Identifier used by the *System producer* to specify a change in a software component from a previously identified version or to indicate that it is the first version. The *System version* element captures the target component's version and allows an organization to identify a specific code delivery. If the *System producer* does not provide a version, then the *SBOM author* should indicate that the information is unknown.	
System timestamp	Record of the date and time of the most recent update to the AI system. Updates can include model updates or software updates of other components that make up the system.	
System data flow	This element describes the *Data flow* (link, reference, or description) between the different components of an AI system.	• input/output endpoints; • description of the data information flow (source → destination); • Application Programming Interfaces (APIs) of external services the AI system uses; • multi-agent communication protocols; • bidirectional data flow towards external services (web grounding).
System data usage	This element describes how data in the AI system is processed and consumed.	Link/URL to technical documentation (such documentation may contain information on whether data is

Element	Description	Examples
		used for improving the model performance, information on data logging from the APIs used to invoke the system and information on whether metadata is derived from the user input data).
System input/output properties	Element used to describe the properties of the input and the output of the AI system.	• type of input and output data handled, clearly specifying the differences between input and output data types and characteristics; • the modality (text, audio, image, video, etc.); • the input preprocessing (such as the type of tokenizer used in the case of LLMs); • link to URL/documentation describing impacts on system decision making.
Intended application area	This element describes the type of application the AI system is deployed in.	• real time or near real time application; • cybersecurity, medical/healthcare, finance, etc.

2.3 Models Cluster Elements

The Models cluster includes basic information for identifying the **models used by the AI system**, describes for each model how its weights were produced, and outlines their properties and limitations.

The Models cluster contains the following elements:

- Model name
- Model identifier
- Model version
- Model timestamp
- Model producer
- Model description
- Model hash value
- Model hash algorithm
- Model properties
- Model input-output properties
- Model training properties
- Model license
- Model external references

Element	Description	Example
Model name	The name assigned by the *Model producer* to a software component. The *Model name* element identifies the software component in a human-readable way. The *Model producer* determines the component name. This element is distinct from the *Model identifier* element. Data formats implementing the *Model name* element should allow for multiple entries to capture alternate names. Inputs for this element should use full names and should not use acronyms (unless the component's official name includes an acronym).	
Model identifier	Identifier(s) used to identify a component or serve as a look-up key for relevant databases. The *Model identifier* element contains at least one software identifier associated	

Element	Description	Example
	with the target component.[7] Machine-processable, unique software identifiers support automated analysis. This element should use common software identifiers, such as Common Platform Enumeration (CPE)[8] and Package-URL (PURL).[9] This element may also include universally unique identifiers (UUID), organization-specific identifiers, commit hashes, and intrinsic identifiers such as OmniBOR[10] and SWHID.[11] If there are multiple software identifiers, either in the same identifier format or different one(s), the *SBOM author* should include all of them.	
Model version	Identifier used by the *AI model producer* to specify a change in a software component from a previously identified version or to indicate that it is the first version. The *AI model version* element captures the target component's version and allows an organization to identify a specific code delivery. If the *AI model producer* does not provide a version, then the *SBOM author* should indicate that the information is unknown.	
Model timestamp	Record of the date and time of the most recent update to the AI model.	This element can also be used for official production release date of the AI model.

[7] Cybersecurity and Infrastructure Security Agency. Software Identification Ecosystem Option Analysis. October 26, 2023.
[8] National Institute of Standards and Technology. National Vulnerability Database, Official Common Platform Enumeration (CPE) Dictionary. August 20, 2025.
[9] Ecma International. ECMA-427: Package-URL (PURL) Specification, 1st Edition. December 2025.
[10] OmniBOR Specification. Version 0.1.
[11] SWHID Specification. Version 1.2; International Organization for Standardization. ISO/IEC 18670:2025, Information Technology—SoftWare Hash Identifier (SWHID) Specification V1.2, Edition 1. April 2025.

Element	Description	Example
Model producer	Identifier used for the name of the entity that creates, defines, and identifies an AI model. The *Model producer* element contains a human-readable string that identifies the entity that produced the model. *Model producer* refers to the originator or manufacturer of the model. Organizations can use the *Model producer* element to learn more about a model's producer. It can also enable the identification of a point of contact for software security concerns. The *Model producer* element should allow for multiple entries.	The producer(s) can be defined as companies that were involved in the pre-training, post-training, or fine tuning of the model.
Model description	Element used to describe the model's capabilities, known limitations, and lineage.	• intended application area the model was designed or developed for as well as known weaknesses and limitations; • lineage contains creation information of the model, such as information about the predecessor model on which fine tuning was done (i.e., models used for distillation or finetuning) or known derivative models; • model dependencies such as internal or third-party packages, libraries, modules, or development frameworks.
Model hash value	The output generated from applying a cryptographic hash algorithm to an executable component artifact. The *Model hash value* element contains an American Standard Code for Information Interchange (ASCII)-	The hash of the weights or model file or other model related artifacts.

Element	Description	Example
	formatted value that is the result of providing the executable component artifact as input to a cryptographic hash algorithm. If the *SBOM author* does not have access to the executable component artifact, then the *SBOM author* should indicate the value is unknown.	
Model hash algorithm	The cryptographic algorithm used to compute the *Model hash value* of the software component. The *Model hash algorithm* element documents the algorithm that produced the *Model hash value* to allow validation of the integrity of the target component. The *SBOM author* should identify the algorithm using Internet Assigned Numbers Authority (IANA) Hash Function Textual Names.[12] The algorithm should be approved by a relevant authority, such as the U.S. National Institute of Standards and Technology (NIST).[13]	
Model properties	Element used to describe the characteristics of the specific model considered, such as the architecture of the model in case of parametric models. This element can contain several different sub-elements with more specific and detailed model information, such as those listed in the example column.	• parametric as Neural Networks, non-parametric such as Clustering or KNN; • the number of parameters/model size; • the architecture of the model (encoder-decoder, encoder only, decoder only, and the underlying network type, for example, transformer, Convoluted Neural Networks,, decision tree, Recurrent Neural

[12] Internet Assigned Numbers Authority. Hash Function Textual Names. December 16, 2019.
[13] National Institute of Science and Technology, Computer Security Resource Center. Projects: Hash Functions. September 9, 2024.

Element	Description	Example
		Networks , or Long Short-Term Memory; • popular architectures such as ResNet, Visual Geometry Group, or Generative Pre-trained Transformer; • information on hyper-parameters of the model.
Model input-output properties	Element used to describe the properties of the input and the output of the AI model.	• type of input and output data handled, clearly specifying the differences between input and output data types and characteristics; • the context length; • the modality (text, audio, image, video, etc.); • input preprocessing (such as the type of tokenizer used in the case of LLMs, Vision Language Models or multimodal models).
Model training properties	This element describes the different training techniques that have been used for the AI system models (this includes all types of training, including what is generally referred to as pre-training and post-training/fine tuning/continual learning).	Link/URL to documentation (for example models cards if such information is present) describing the type of learning (unsupervised learning, supervised learning, self-supervised learning) as well as the post-training characteristics (continual learning, fine tuning stages, reinforcement learning from human feedback, instruction tuning, and Reinforcement Learning Optimization type such as Direct Preference Optimization, Proximal Policy

Element	Description	Example
		Optimization, and Group Relative Policy Optimization).
Model license	This element describes the license type of the AI model.	• link/URL to the model licensing document (the document can contain references of open source licensing such as EU Public License, General Public License, and Apache) or to the corresponding fields in SPDX/CDX files; • specifies whether the AI model is open weight, open architecture, open data or open training.
Model external references	Link/URL to external references related to the model.	• link/URL to any existing framework related to the model or system card from editors; • JSON schema of the model card itself, such as is provided by existing tools; • public documentation; • research paper.

2.4 Datasets Properties (DP) Cluster Elements

The DP cluster provides information on datasets used during the whole life cycle of the model, including basic information that documents the identity and provenance of data.

The DP cluster contains the following elements:

- Dataset name
- Dataset description
- Dataset content
- Dataset identifier
- Dataset hash
- Dataset provenance
- Dataset statistical properties
- Dataset sensitivity
- Dataset dependency relationship
- Dataset license

Element	Description	Example
Dataset name	The name assigned to the dataset by the dataset creator.	• string to publicly identify the dataset; • equivalent to component name.
Dataset description	Element used to describe the dataset, such as its primary intended use with regards to AI model training (for example, fine-tuning, benchmark, context extension).	• Dataset used for pre-training, fine tuning, or model/system evaluation; • can include whether the dataset is private or public.
Dataset content	Element used to describe the content of the dataset.	• description of the content of the dataset, such as financial data, medical records data, etc.; • the format of the dataset (data structure, encoding like JSON or XML); • the data format (image, audio, video, 3D).
Dataset identifier	Information that allows univocal identification of the dataset.	Dataset URL, URI, etc.

Element	Description	Example
Dataset hash	The cryptographic value generated from taking the hash of the dataset.	The hash over the dataset file.
Dataset provenance	Element used to track dataset provenance.	Element used to capture information regarding: • the origin (sources or organizations contributing to the dataset elements), the data collection methods (for example, web crawling or commercial agreements), the data post processing steps, the data preprocessing/curation steps, and the data labelling steps; • the creator; • in case of synthetic data, the methods used to create it.
Dataset statistical properties	Element used to capture statistical characteristics associated with the dataset throughout its lifecycle.	Statistical metrics, when relevant, of the dataset (e.g., mean, variance, median, mode, range, skewness).
Dataset sensitivity	Element used to express the type of data present inside the dataset, where type is related to the sensitivity level of the data.	This element covers whether the dataset includes PII data, freely accessible data, copyright protected data, sensitive data (like financial or medical records) or national security related data.
Dataset dependency relationship	List of software used to create, modify and maintain the dataset during its lifecycle. The dependency relationship should reflect how a given dataset may be largely derived from another dataset.	E.g., management of datasets, labeling tools, filtering tools.

Element	Description	Example
Dataset license	This element describes the license type of the datasets.	link/URL to the dataset licensing document.

2.5 Infrastructure Cluster Elements

The Infrastructure cluster contains physical and virtual infrastructure that is critical to proper operation and support of the AI system. If existing, it also includes a link to a Hardware Bill of Materials (HBOM), to also cover specialized AI hardware.

The Infrastructure cluster contains the following elements:

- Infrastructure software
- Infrastructure hardware

Element	Description	Example
Infrastructure software	This element lists dependencies to software components specifically required to deliver and run an AI system.	List of dependencies to • firmware; • package managers; • third party libraries; • frameworks; • runtime environments; • tools used by the AI system.
Infrastructure hardware	This element links to an existing HBOM as dependencies, which contains the hardware components on which the AI system is deployed.	Link to an existing HBOM.

2.6 Security Properties (SP) Cluster Elements

The SP cluster focuses on the **cybersecurity measures** that apply to AI models and systems.

The SP cluster contains the following elements:

- Security controls
- Security compliance
- Cybersecurity policy information
- Vulnerability referencing

Element	Description	Example
Security controls	This element describes the implemented AI specific and general cybersecurity controls. The element may also include a link to a specific framework or guideline used as reference for the implementation.	Technical controls implemented, such as: <u>General cybersecurity controls:</u> • encryption; • data minimization; • differential privacy techniques; • access controls; • API authentication; • ip/op anomaly detection systems and filters; • physical controls; • technical controls; • system-level controls (role-based access); • administrative measures; • link/references to specific applied security guidelines. <u>AI specific controls:</u> • adversarial robustness training; • prompt injection controls and tools for LLMs or LLM-based agents; • input/output filters;

Element	Description	Example
		• data-level controls (to curate training data).
Security compliance	This element describes the cybersecurity standards and certifications related to the AI model/system, if obtained by the producer.	The element can include cybersecurity certification schemes, standards or frameworks to which the AI model/system is compliant.
Cybersecurity policy information	This element provides a link/URL to documentation regarding the AI model/system producer's published security.txt file.	
Vulnerability referencing	This element links to databases/repositories that provide exploitability of known vulnerabilities in the AI model/system. This is to help users decide whether to deploy a model in specific contexts where there is known risk.	This element can contain a static link/URL to the security repository of the system provider of the AI model.

2.7 Key Performance Indicators (KPI) Cluster Elements

The KPI cluster contains elements that refer to information on the AI system's KPIs and its components (including AI models that are integrated in the system), focusing on their lifecycle phases.

The KPI cluster contains the following elements:

- Security metrics
- Operational performance KPIs

Element	Description	Example
Security metrics	Metrics to evaluate the security characteristics of the AI models integrated in the AI system or the system itself.	Security related benchmarks and metrics, e.g., robustness (resilience against third-party manipulation).
Operational performance KPIs	KPIs related to the operational condition or other threat indicators of the AI system.	System uptime, incident resolution time, system latency, request throughput, and load balancing.

3. Discussion

In addition to the clusters described above, the G7 Cybersecurity Working Group considered additional elements which might be useful for an SBOM for AI in the future. One example is the level of decision making or autonomy of an AI system which might become more relevant due to the fast-changing developments in technology, particularly around agentic AI. Including such an element in SBOMs for AI could help to assess the impact of a potentially damaging compromise. However, while the group recognized the importance and relevance of decision making or autonomy of an AI system to cybersecurity, it was decided to not explicitly call it out as a separate element. This element may be addressed differently across different jurisdictions, including through safety requirements.

Besides addressing single elements, the authors highlight that an SBOM for AI by itself is not sufficient for increasing cybersecurity along the supply chain. To ensure substantial protection of the AI supply chain, it is necessary to connect the SBOM for AI to cybersecurity tools, such as vulnerability scanning and management tools, security advisories and bulletins, and promoting development of adaptable and evolutionary tooling mechanisms.

4. Conclusion

This guideline jointly drafted by the cybersecurity agencies of the G7 Group is a first step towards increasing the supply chain transparency and security of AI models and systems. While there have been multiple efforts dedicated to such an endeavor, the document is meant to cover a minimum set of criteria and does not claim to be exhaustive. Rather, it presents a shared understanding on which elements foster transparency and increase cybersecurity along the AI supply chain. Eventually, an SBOM for AI will help to strengthen the security of the AI supply chain if deployed together with the right cybersecurity tools. This work also seeks to bring added value to stakeholders along the AI supply chain.

References

- A shared G7 vision on Software Bill of Materials for AI. Transparency and Cybersecurity along the AI Supply Chain https://www.acn.gov.it/portale/documents/d/guest/paper_sbom-for-ai_19may2025_-clean-2
- NCSC Guidelines for Secure AI System Development https://www.ncsc.gov.uk/collection/guidelines-secure-ai-system-development
- SBOMs and the importance of inventory https://www.ncsc.gov.uk/blog-post/sboms-and-the-importance-of-inventory