OPSWAT.

WHITEPAPER

SBOM in 2025: A Strategic Asset, Not Just a List

A Practical Guide to Implementing Software Bills of Materials for Enhanced Security and Regulatory Compliance

The Growing Need for SBOMs

The need for SBOMs (Software Bills of Materials) has been intensifying over the past years due to several high-profile supply chain attacks that exploited vulnerabilities in software supply chains.

Examples include the Log4j critical flaw in a popular Java library that has impacted millions of applications (2021); ongoing attacks targeting open-source packages like NPM and PyPI ecosystems; or the high-profile attacks on software suppliers Codecov and Kaseya that were followed by U.S. Executive Order 14028 mandating SBOMs in the same year (2021).

Software ecosystems only continue to grow even more complex, which means more organizations will be required to prioritize software transparency, integrity, and visibility – whether it's the software they produce or purchase.

This whitepaper aims to:

1. Provide a clear explanation of what SBOM is, its benefits, and relevant concepts around the topic.
2. Help organizations stay up to date on the most recent SBOM recommendations and regulations for 2025.
3. Offer guidance for incorporating SBOM into organizational security processes.

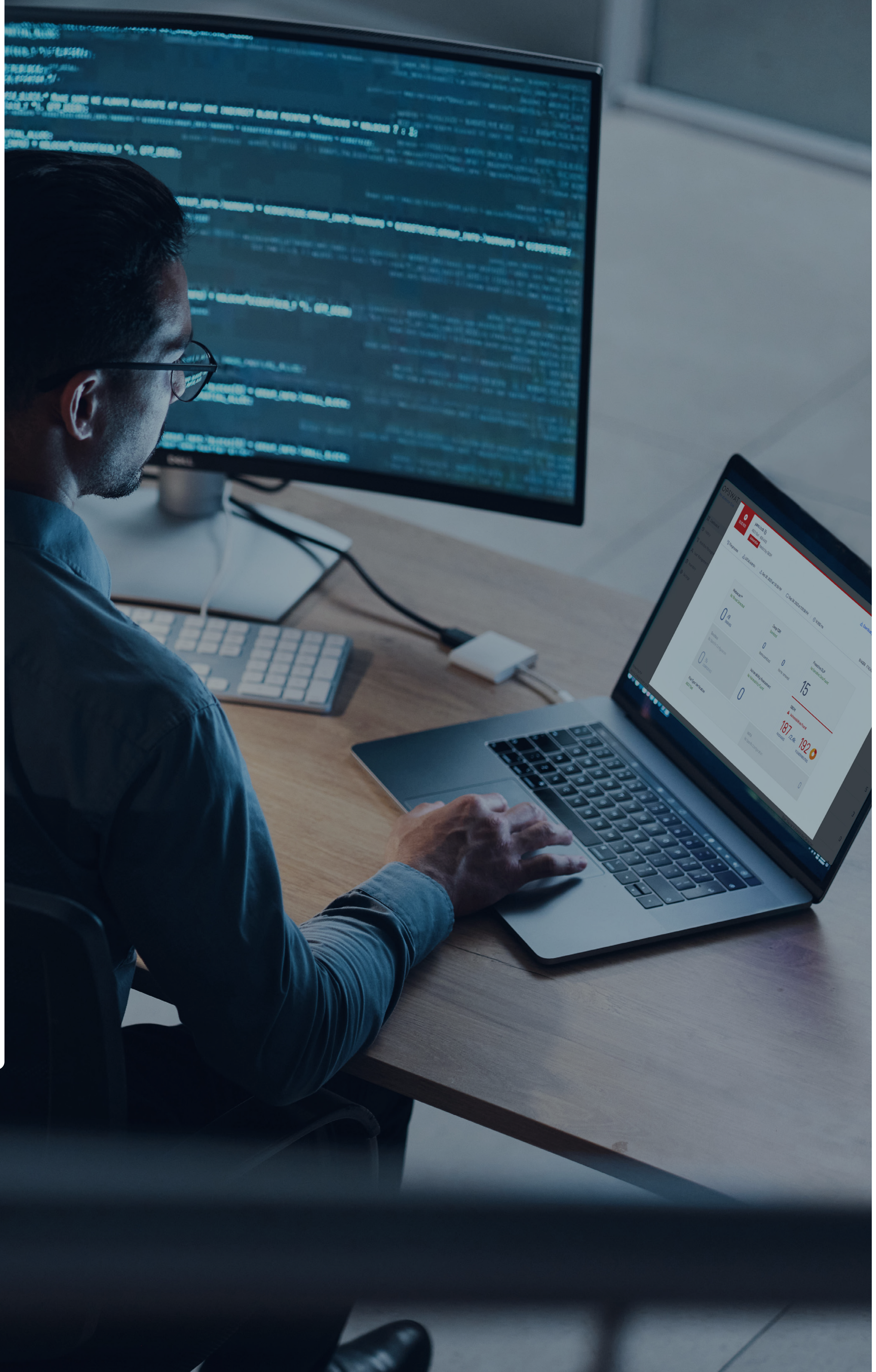


Table of Contents

01 SBOM Concepts

02 SBOM: More Than Just a List

03 SBOM Generation

04 Choosing an SBOM Tool

05 The Future of SBOM and What's Next

01

What is an SBOM?

A Software Bill of Materials (SBOM) is a formal, machine-readable inventory that details the components, dependencies, and relationships within a software product. By providing a comprehensive record of these elements, SBOMs serve as a critical tool for organizations seeking to enhance software security, compliance, and operational efficiency. They offer visibility into the software supply chain to ensure that every component—whether open-source or proprietary—is accounted for and traceable.

Beyond serving as a simple list, SBOMs play a foundational role in key cybersecurity and risk management activities. They enable organizations to proactively manage vulnerabilities, enforce license compliance, and improve software asset tracking. By documenting component relationships, capturing licensing details, and preserving modification histories, SBOMs empower businesses to reduce risks and maintain the integrity of their software ecosystems.

> A Software Bill of Materials (SBOM) is a formal record containing the details and supply chain relationships of various components used in building software. These components, including libraries and modules, can be open source or proprietary, free or paid, and the data can be widely available or access-restricted.

CISA (Cybersecurity and Infrastructure Security Agency)[1]

History of SBOM

The concept of an SBOM dates back to the 1990s and early OSS (open-source software) development efforts. Over time, SBOMs have become a standardized, essential part of regulatory frameworks related to secure software development.

1990-1999

The use of binary-based SBOMs grows as various industry standards and regulations incorporate SBOM requirements.

2000-2009

As OSS adoption expands, standardized practices for managing dependencies and licenses are established.

2010-2015

SBOMs gain prominence in the context of software supply chain security.

The Linux Foundation introduces the SPDX SBOM format in 2010, originally named "Packaged Facts."[2]

2016

NIST emphasizes the importance of SBOMs in its report, Improving Software Supply Chain Security.

2018

The NTIA (National Telecommunications and Information Administration) develops a foundational framework for SBOMs.

OWASP (Open Web Application Security Project) introduces CycloneDX, an alternative SBOM format, in March.[3]

2020

The use of binary-based SBOMs grows as various industry standards and regulations incorporate SBOM requirements.

2021

Executive Order 14028 directs the Secretary of Commerce to define minimum SBOM elements.

The U.S. Department of Commerce and NIST issue the Minimum Essential Elements for an SBOM.

NTIA publishes the Software Suppliers Playbook: SBOM Production and Provision.

SPDX publishes the ISO/IEC 5962:2021 specifications, which define a standard data format for software components.[4]

Google introduces SLSA (Supply Chain Levels for Software Artifacts), a framework for ensuring software artifact integrity throughout the supply chain, drawing inspiration from its internal "Binary Authorization for Borg" framework.[5]

2022

The OMB (Office of Management and Budget) incorporates SBOMs into its memorandum Enhancing the Security of the Software Supply Chain through Secure Software Development Practices (OMB M-22-18).

The Secure Software Development Framework (SSDF) V1.110 is released with SBOM requirements, aligned with Executive Order 14028.

Gartner publishes its research report, *Innovation Insight for SBOMs.[6]

2023

Carnegie Mellon University's Software Engineering Institute introduces the Software Bill of Materials Framework: Leveraging SBOMs for Risk Reduction.[7]

Gartner publishes the report "Mitigate Enterprise Software Supply Chain Security Risks"[8]

The U.S. Food and Drug Administration requires medical device manufacturers to provide SBOMs as part of cybersecurity attestations under the Consolidated Appropriations Act.

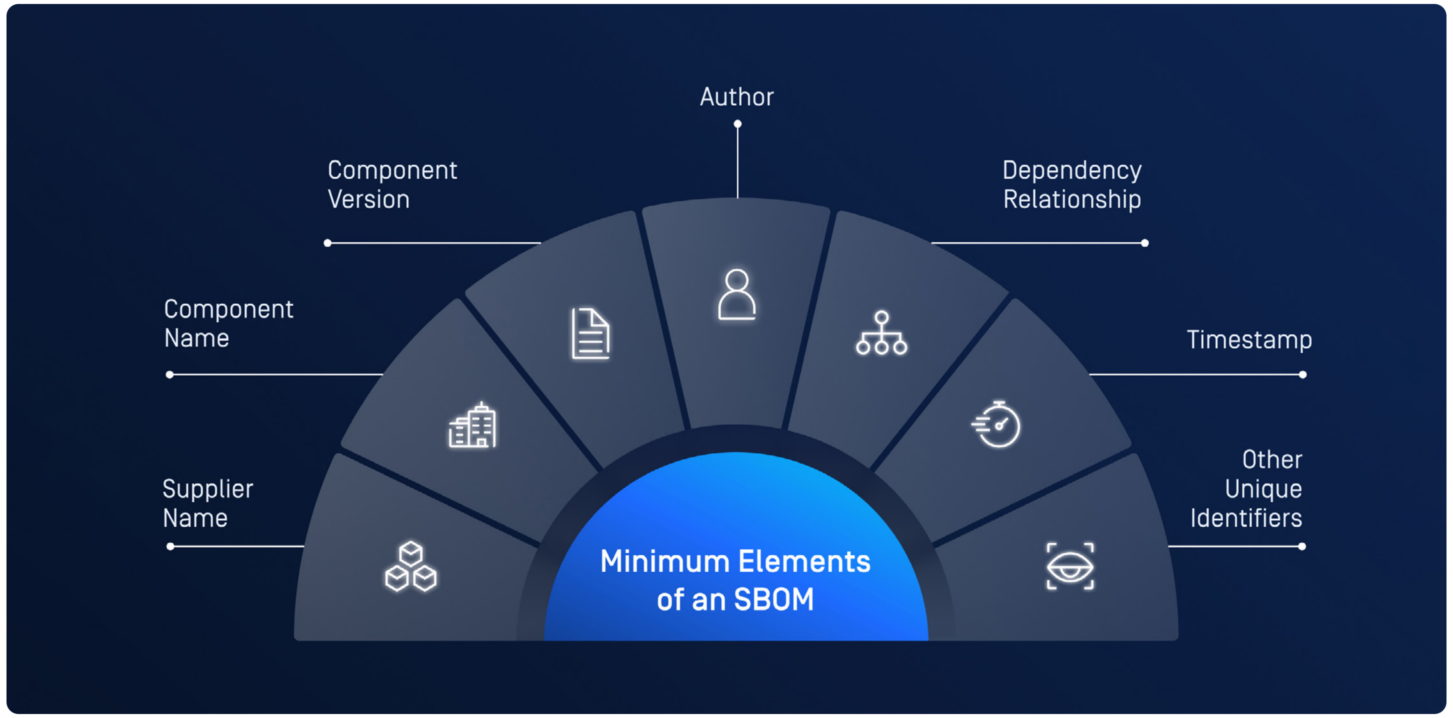
2024-present

Gartner publishes the research "Leader's Guide to Software Supply Chain Security"[9]

The demand for software supply chain security continues to grow. More tools in the market have emerged.

SBOM Elements

An SBOM consists of essential attributes that identify software components, including the author, timestamp, supplier name, component name, version string, unique identifier, and relationships. Additional metadata can be included to support specific security, compliance, and operational needs.

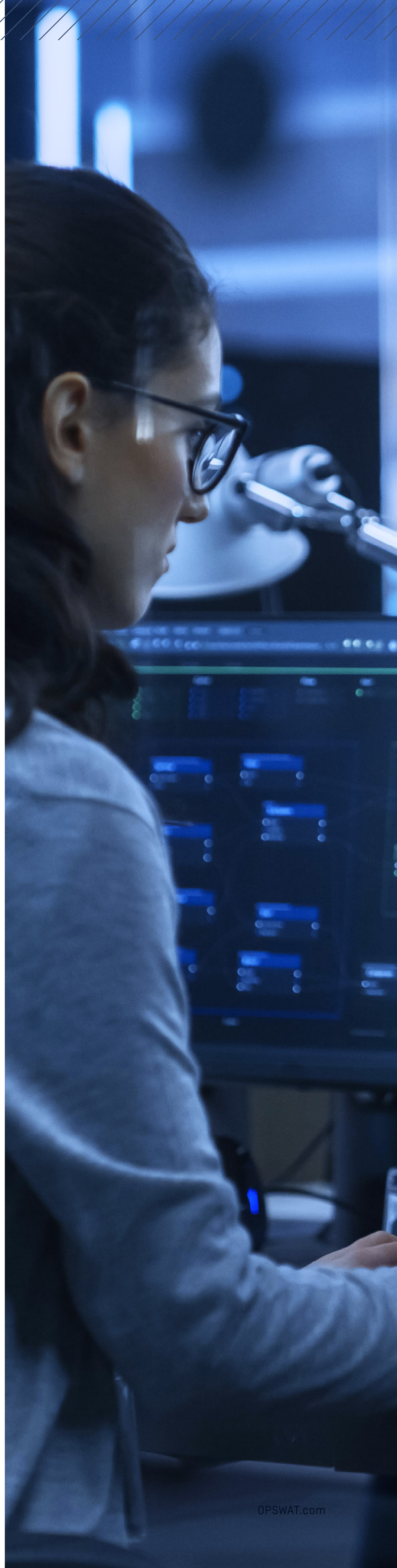


The U.S. Department of Commerce defines a set of minimum data elements required for SBOM.[10]
These elements ensure consistency, traceability, and interoperability across software supply chains:

- **Supplier Name:** Identifies the entity responsible for creating and defining the software component.
- **Component Name:** The official designation assigned by the original supplier.
- **Version of the Component:** Specifies changes between different software releases.
- **Other Unique Identifiers:** Additional references used to identify a component or link it to relevant databases.
- **Dependency Relationship:** Describes how an upstream component (X) is included in another software (Y).
- **Author of SBOM Data:** The entity responsible for generating and maintaining the SBOM.
- **Timestamp:** A record of the date and time when the SBOM data was assembled.

SBOM Use Cases

Product Security, Architectural, and Licensing Risks Malware	SBOMs help identify vulnerabilities in software components, assess architectural dependencies, and ensure compliance with open-source and third-party licensing requirements.
Procurement and M&A	When evaluating software for purchase or acquisition, SBOMs provide insight into component origins, security risks, and licensing obligations, reducing potential legal and operational liabilities.
Software Component Transparency	By documenting all software components and dependencies, SBOMs offer a clear view of what is inside an application, helping teams track updates, patches, and potential security risks.
Supply Chain Transparency	SBOMs provide visibility into the origins and integrity of software components, enabling organizations to mitigate risks associated with compromised or unverified third-party code.
Vendor Risk Management	By requiring SBOMs from software vendors, organizations can assess security practices, ensure compliance with internal policies, and reduce exposure to supply chain threats.



Why SBOM?

Risk Management

SBOMs provide critical insights for making informed risk management decisions when acquiring or deploying software. By maintaining a complete and accurate inventory of both first-party and third-party components, organizations can identify risks more efficiently. Additionally, SBOMs enhance overall risk management by ensuring vital security and compliance information is consistently captured and exchanged.

Vulnerability Management

SBOMs play a crucial role in vulnerability management, helping organizations make better decisions about software deployment and ongoing operations. When combined with VEX (Vulnerability Exploitability eXchange) data, SBOMs allow teams to assess risks from vulnerable components more quickly and accurately. Having an SBOM for an operating system enables developers, IT administrators, and end-users to gain visibility into software components, detect vulnerabilities, and apply necessary patches efficiently.

Incident Management

SBOMs help organizations strengthen their incident management capabilities by enabling proactive detection and response to newly discovered software vulnerabilities. By offering a clear record of software components, they support continuous monitoring, improve risk identification, and enhance overall cybersecurity posture.

Compliance Management

Organizations can use SBOMs to demonstrate compliance with statutory regulations and industry standards, ensuring transparency into software dependencies. SBOMs also support license compliance by tracking open-source and proprietary software components, reducing legal risks associated with improper software usage.

Supply Chain Transparency

SBOMs provide a detailed breakdown of software supply chains, offering visibility into component origins and potential risks. This transparency ensures accountability in software development and allows organizations to make informed decisions about the security and integrity of the software they rely on.

Software Asset Management

By maintaining a comprehensive view of software assets, SBOMs help organizations optimize their IT and OT (operational technology) infrastructure. They enable better tracking of software versions, dependencies, and potential security risks, improving overall efficiency in asset management.

Informed Decision-Making

SBOMs empower organizations and end-users to make well-informed decisions about the software they use. With a complete understanding of software components, organizations can evaluate security, compliance, and operational risks before adopting or deploying software solutions.

Enhanced Security and Privacy

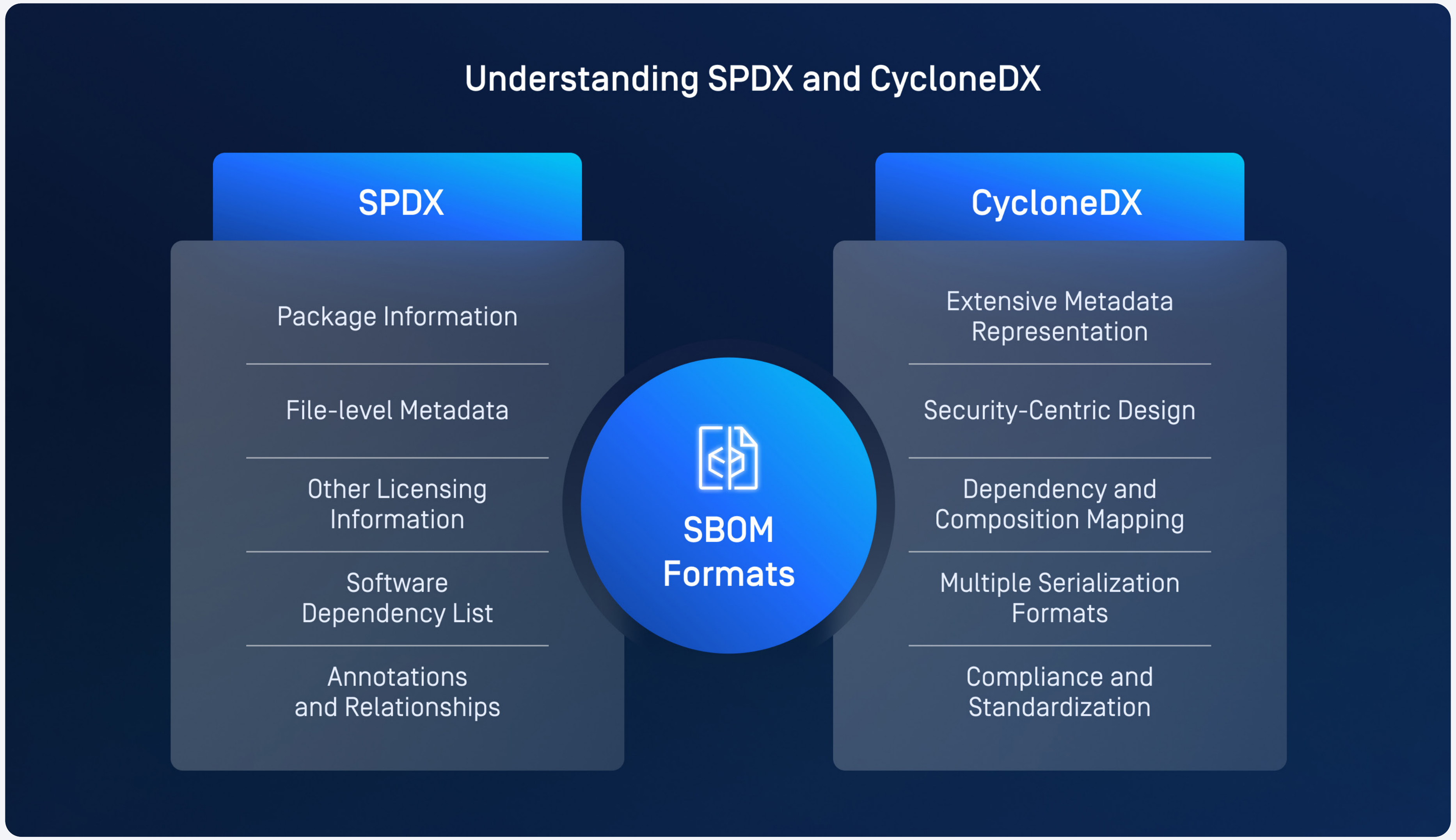
SBOMs contribute to improved security, privacy, and software maintenance by offering a structured way to track software components over time. By facilitating automation and continuous monitoring, they support proactive vulnerability management, risk reduction, and overall software safety.

SBOM Formats

SBOMs can be represented in different standardized formats, with SPDX (Software Package Data Exchange) and CycloneDX being the most widely used.

SPDX®

An ISO-certified, open-source standard designed to communicate software components, licenses, and copyrights. It is commonly used for compliance and legal documentation, providing a structured format for sharing software package details.



Understanding SPDX and CycloneDX

SPDX

- Package Information
- File-level Metadata
- Other Licensing Information
- Software Dependency List
- Annotations and Relationships

SBOM Formats

CycloneDX

- Extensive Metadata Representation
- Security-Centric Design
- Dependency and Composition Mapping
- Multiple Serialization Formats
- Compliance and Standardization

CycloneDX

A lightweight, open-source standard designed for software security and risk analysis. Defined using JSON Schema, XML Schema, and Protocol Buffers, it captures metadata, components, dependencies, vulnerabilities, and complex relationships, making it highly extensible for specialized and future use cases.

```
    "string-width": {
          "version": "4.2.0",
          "resolved": "https://registry.npmjs.org/string-width/-/string-
    width-4.2.0.tgz",
          "integrity": "sha512-zUz5JD+tgqtuDjMhwIg5uFVV3dtqZ9yQJlZVfq4I01/K5Paj5UHj
          "requires": {
            "emoji-regex": "^8.0.0",
            "is-fullwidth-code-point": "^3.0.0"
            "strip-ansi
          },
          "dependencies
            "ansi-regex
              "version"
              "reso
              "inte
            },
            "emoji
              "ver
              "res
              "inte
```

Software Bill of Materials

⚠ Vulnerabilities Found

34 /78

PACKAGES

47

VULNERABILITIES

```
    "string-width": {
          "version": "4.2.0",
          "resolved": "https://registry.npmjs.org/string-width/-/string-
    width-4.2.0.tgz",
          "integrity": "sha512-zUz5JD+tgqtuDjMhwIg5uFVV3dtqZ9yQJlZVfq4I01/K5Paj5UHj7VyrQOJvzawSVlKpObApb
          "requires": {
            "emoji-regex": "^8.0.0",
            "is-fullwidth-code-point": "^3.0.0",
            "strip-ansi": "^6.0.0"
          },
          "dependencies": {
            "ansi-regex": {
              "version": "5.0.0",
              "resolved": "https://registry.npmjs.org/ansi-regex/-/ansi-regex-5.0.0.tgz",
              "integrity": "sha512-bY6fj56OUQ0hU1KjFNDQuJFezqKdrAyFdIevADiqrWHwSlbmBNMHp5ak2f40Pm8JTFyM2mq
            },
            "emoji-regex": {
              "version": "8.0.0",
              "resolved": "https://registry.npmjs.org/emoji-regex/-/emoji-regex-8.0.0.tgz",
              "integrity": "sha512-MSjYzcWNOA0ewAHpz0MxpYFvwg6yjy1NG3xteoqz644VCo/RPgnr1/GGt+ic3iJTzQ8Eu3T
            },
            "is-fullwidth-code-point": {
              "version": "3.0.0",
              "resolved": "https://registry.npmjs.org/is-fullwidth-code-point/-/is-fullwidth-code-point-3
              "integrity": "sha512-zymm5+u+sCsSWyD9qNaejV3DFvhCKclKdizYaJUuHA83RLjb7nSuGnddCHGv0hk+KY7BM
```

02

SBOM: More Than Just a List

SBOM is not merely a list of components. In 2025, it is a central pillar of cybersecurity protocols, essential for maintaining compliance and keeping up with changes in the software development lifecycle.

Software Provenance

Software provenance refers to the complete history of a software product, tracking its origin, development, ownership, and modifications throughout its lifecycle. According to the National Institute of Standards and Technology (NIST), provenance includes “the chronology of the origin, development, ownership, location, and changes to a system or system component and associated data, as well as the personnel and processes involved in modifications.” (NIST Definition)

The Multi-Faceted Values of SBOM

Vulnerability Detection
Public known vulnerabilities
Vulnerable components and dependencies

Vulnerability Management
Compromised software components
Supply chain risk mitigation

Incident Response
Continuous security throughout SDLC
Software supply chain threats visibility

Risk Assessment
Verify third-party components
Increase software transparency and cyber-resilience

SBOM for Security

Vulnerability Detection

SBOMs enable automated vulnerability detection by creating an automated link between public vulnerabilities and vulnerable products, accelerating the process of searching for vulnerable components and dependencies. This allows security teams to focus on prioritizing and mitigating vulnerabilities.

Vulnerability Management

Integration with VEX allows for more effective vulnerability management by indicating the actual impact of a vulnerability on a product, and can also clarify when software is not exploitable despite having a vulnerability. Using SBOMs and VEX together enhances the vulnerability management processes.

Incident Response

SBOMs support better incident response by facilitating detection of and response to new software vulnerabilities. SBOMs also help in quickly identifying specific networks or endpoints containing affected software.

Risk Assessment

SBOMs contribute to supply chain security by increasing transparency and helping to verify third-party components. They aid in identifying suspicious or counterfeit software, improve software supply chain resilience, and help identify and mitigate risks associated with third party software.

SBOM for Compliance

SBOMs facilitate the demonstration of regulatory compliance with statutory and other requirements, offering insight into software dependencies. They can help organizations meet requirements like the US Executive Order on Improving the Nation's Cybersecurity, the EU Cyber Resilience Act (CRA), and NIST SP 800-21832.

SBOMs support license management by tracking license obligations and ensuring compliance, and they can identify and track open-source software licenses. This helps ensure that software can be used without legal repercussions.

SBOMs also promote data standardization, reducing duplication of effort across multiple sectors and helping to manage different risk tolerance rules and policies within an enterprise.

Key Frameworks and Regulations

NIST NIST SP 800-218: Secure Software Development Framework (SSDF) which includes SBOM requirements	ISO ISO/IEC 27036-3
ISO ISO 27001/27002	OWASP SCVS (Software Component Verification Standard)
AICPA SOC 2 Service Organization Control Type 2 (SOC 2)	PCI Security Standards Council PCI DSS
IMDRF International Medical Device Regulators Forum (IMDRF)	OWASP OWASP Authoritative Guide to SBOM

U.S.	Executive Order 14028 on Improving the Nation's Cybersecurity
	Department of Homeland Security (DHS) Software Supply Chain Risk Management Act
	Food and Drug Administration (FDA) Medical device cybersecurity requirements
	National Institute of Standards and Technology NIST SP 800-161r1
	National Security Agency/Central Security Service
	National Highway Traffic Safety Administration
	US Securities and Exchange Commission Cybersecurity Risk Management Rules
EU	The EU Cyber Resilience Act (CRA)
Australia	Australian Cyber Security Centre Guidelines for Software Development
	Australian Signals Directorate Guidelines for Software Development
Canada	Canadian Forum for Digital Infrastructure Resilience
	Canadian Center for Cyber Security ITSM.10.071 Supply Chain Recommendations
	Canada SCAWG Digital Supply Chain Security Recommendations
Germany	German Federal Office for Information Security (BSI) TR -03183 Cyber Resilience Requirements
Japan	Japan Economic Security Protection Act
	Japan METI SBOM Guide
India	India CERT-in Technical Guidelines on SBOM
UK	UK GCHQ National Cybersecurity Centre Guidance

Note: this list is not exhaustive.

SBOMs for Different Software Types and Scenarios

Microservices

Since microservices operate as independent units, each service can have its own SBOM. This helps provide clear visibility into dependencies and vulnerabilities across distributed architectures.

Single Applications

Whether for desktop, mobile, or web applications, SBOMs help document software components, track dependencies, and enhance security and compliance.

Multi-Product Solutions

When multiple software products are integrated into a larger solution, SBOMs help maintain visibility into each component, ensuring better management of security risks and compliance requirements.

Multi-Module Products

Complex software products often consist of multiple interdependent modules. SBOMs provide transparency into how these modules interact and evolve over time.

Legacy Software

Even when source code is unavailable, SBOMs can help organizations manage risks associated with outdated or unsupported software by identifying components and potential vulnerabilities.

Cloud-Native Applications

Cloud-based services often consist of dynamically orchestrated components. CycloneDX provides a structured way to document and manage these services, ensuring security and compliance in cloud environments.

03

SBOM Generation

How SBOMs are Created

SBOMs provide a detailed inventory of software components, including attributes such as name, version, and unique identifiers. The generation process is typically automated and integrated into build and packaging workflows.

The responsibility for SBOM creation falls on vendors, internal development teams, and system integrators. If upstream suppliers provide SBOMs, they should be included; otherwise, a "best effort" SBOM should be created to indicate missing data. Additional attributes, such as known vulnerabilities, can be included, leveraging sources like the NVD (National Vulnerability Database).

When to Create an SBOM

SBOMs should be generated at key software lifecycle stages:

- **Initial release:** When a software product or component is first built.
- **Updates and patches:** Any time modifications are made.
- **New versions:** To reflect changes in dependencies or configurations.

Changes can be documented by:

- Listing modified components as separate, new entries.
- Updating version identifiers for existing components.

Sharing SBOMs

SBOMs should be distributed alongside software components using standardized methods:

- Embedded within the software package for seamless access.
- Metadata links (e.g., URLs) providing online access.

Different ecosystems may require varying exchange methods based on format compatibility. The IETF (Internet Engineering Task Force) has developed standardized discovery protocols for SBOMs across SPDX, CycloneDX, and other formats.

Tools for SBOM Generation

Organizations can leverage various tools to generate and manage SBOMs:

- **NTIA SwiftBOM Web Tool:** A web-based tool for generating and managing SBOMs.
- **NTIA Excel Tool:** Provides a spreadsheet-based SBOM creation method.
- **SCA (Software Composition Analysis) Tools:** Automate SBOM generation by analyzing software dependencies.
- **CycloneDX Tool Center:** A set of open-source tools for generating and validating CycloneDX SBOMs.
- **SPDX Tools:** Open-source tools supporting SPDX-compliant SBOM generation and validation.

Challenges in SBOM Generation

Despite its benefits, SBOM generation presents several challenges:

- **Completeness, accuracy, and currency:** Ensuring SBOMs contain up-to-date and precise information.
- **Handling legacy and cloud-based systems:** Legacy software may lack metadata, and cloud-native applications introduce complexity.
- **Parsing binaries without package managers:** Difficulties arise when analyzing compiled software lacking clear dependency records.
- **Multi-language repositories:** SBOM tools must support diverse programming languages and build environments.
- **Process maturity:** Ensuring SBOM generation is consistent, scalable, and reliable within development workflows.

The Importance of Automation

Automating SBOM generation helps organizations:

- Reduce human errors and ensure consistent documentation.
- Integrate seamlessly into the SDLC (software development lifecycle).
- Make SBOM generation a continuous, repeatable process.

By embedding SBOM automation into DevSecOps pipelines, organizations can enhance software security, improve compliance, and respond to vulnerabilities more efficiently.

04

Choosing an SBOM Tool

The NSA (National Security Agency) outlines the essential functionalities that an SBOM management tool should cover.[11] According to their guide, the tool should be able to:

SBOM Input

- Support multiple SBOM formats. Import and manage the latest CycloneDX and SPDX versions, ideally supporting JSON, XML, and CSV file types.
- Check structure and compliance. Ensure SBOMs meet format and version specifications upon import, with visual indicators of compliance quality and options for correction.

SBOM Output

- Export SBOMs in CDX or SPDX formats, ideally supporting multiple file types like JSON, XML, and CSV.
- Convert between SBOM formats and file types, and aggregate multiple SBOMs into one.

SBOM Generation

- Generate SBOMs from various sources such as build environments, binary files, system queries, etc.

Components Handling

- Show the NTIA-minimum required fields (e.g., supplier name, component version, dependency relationships) for each component.
- Enrich information by adding external references to the component data and providing visual cues for source references. Graphical representations to visualize component dependencies and enrichments.

SBOM Validation and Integrity

- Capture and display hash information for each component, ideally with digital signatures and provenance data.
- Link to external sources for component provenance verification.

Vulnerability Tracking and Analysis

- Offer daily vulnerability updates, ideally with continuous analysis and threat intelligence integration.
- Notify users of new vulnerabilities, integrate customizable policies, and track vulnerability remediation across SBOMs.

Identifying Exploitable Vulnerabilities

- Indicate whether a vulnerability is exploitable, ideally using VEX format for justification.

User Interface

- Follow HCI standards, incorporate accessibility features, and provide easy-to-understand graphical representations.
- Allow users to view more details on vulnerabilities, components, and risk status with multiple filtering options.

Output Forms & Methods

- Generate standardized reports on component attributes, vulnerabilities, licenses, and suppliers.
- Represent dependencies in graphic/text formats for analysis and communication.

SBOM Versioning and Configuration

- Organize SBOMs, track versions, and compare differences between SBOM versions.
- Provide graphical trends showing vulnerability severity and component version changes.

Integration and Workflow

- Facilitate import/export via APIs and integrate with multiple SBOM sources for automated workflows.
- Support secure exchanges between SBOM producers and consumers.

Supporting Access to Data Sources

- Analyze SBOM components against diverse threat patterns using AI/ML engines.
- Maintain an updatable library of open-source software licenses.

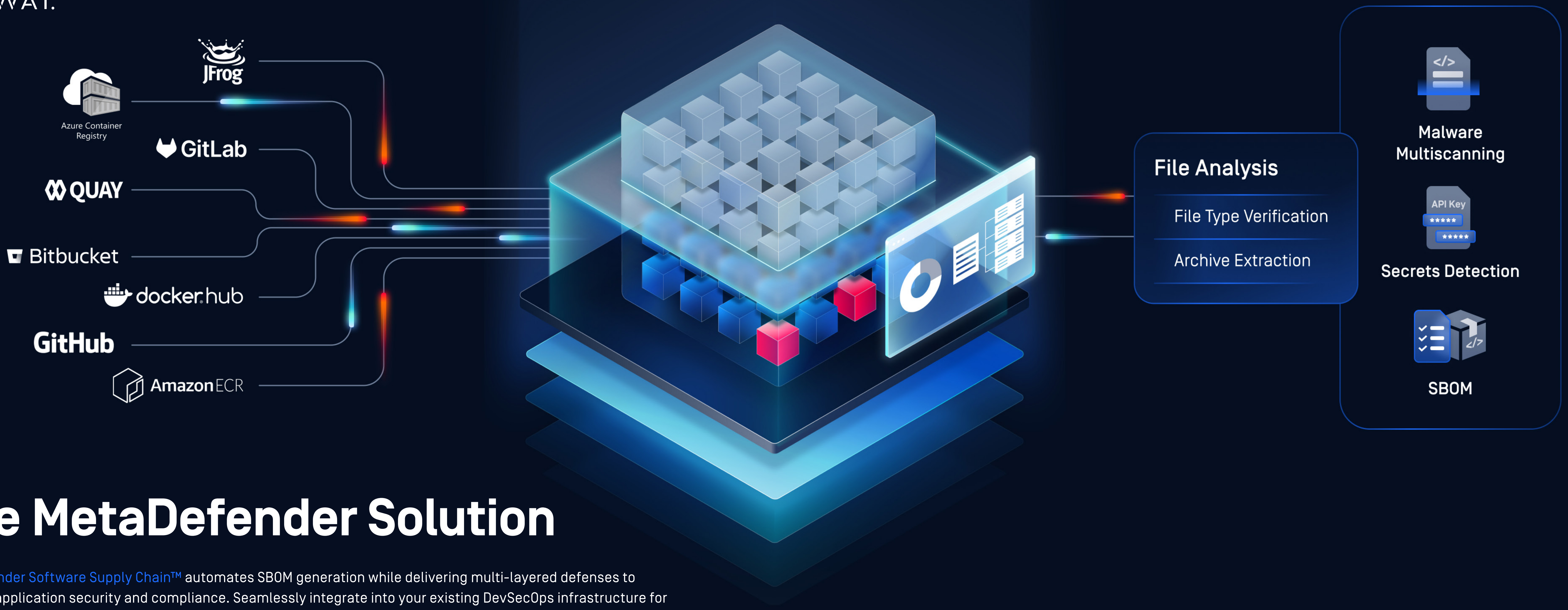
Scalable Architecture

- Support multiple sub-organizations. Enable distinct risk tolerance rules and handle various types of BOMs.
- Integrate into a suite of tools for risk, vulnerability, and incident management.

SBOM Tool Setup and Configuration

- Provide mechanisms to easily download, set up, and integrate the tool in both Linux and Microsoft environments.

OPSWAT.



The MetaDefender Solution

MetaDefender Software Supply Chain™ automates SBOM generation while delivering multi-layered defenses to enhance application security and compliance. Seamlessly integrate into your existing DevSecOps infrastructure for continuous protection throughout the SDLC. With OPSWAT's advanced threat detection and prevention technologies, your SDLC remains safeguarded against malware, vulnerabilities, and other software supply chain risks.

Multi-Layered Threat Prevention

Detect vulnerabilities and software dependencies. Protect your supply chain from malware with MetaScan™ Multiscanning. Generate SBOMs and continuously track your software components.

Software Transparency Compliance

Uncover PII and secrets in source code with Proactive DLP™. Automate the inventory of software components with SBOM to maintain software integrity, helping organizations stay compliant with regulations like SOC2 and ISO 27001.

Source Code and Container Security

Secure both source code and container images with flexible workflows that can be scheduled or triggered by specific actions. Reduce maintenance costs with an integrated solution that fits seamlessly into your CI/CD pipeline.

Integrate into CI/CD Pipeline

Easily integrate into your existing CI/CD tools, or solutions from third-party vendors. Manage roles and automate remediation steps like email alerts and pull request approvals to streamline security management.

05

The Future of SBOM and What's Next

SBOMs are rapidly becoming a fundamental part of cybersecurity and regulatory compliance. With increasing adoption by government agencies and private industries, organizations must ensure they generate, maintain, and share SBOMs to meet evolving security standards. As software supply chain attacks rise, SBOMs provide essential visibility into software components, dependencies, and potential vulnerabilities.

Integrating SBOM into DevSecOps

To maximize security benefits, SBOMs should be embedded into CI/CD pipelines, allowing for real-time monitoring and risk assessment. Best practices include:

- Automating SBOM generation at every stage of development.
- Ensuring compatibility with existing DevSecOps workflows.
- Continuously updating SBOMs to reflect new components and patches.
- Leveraging SBOM data to enhance vulnerability scanning and mitigation.

Learn more about how to automate your SBOM strategy with industry-leading solutions from OPSWAT at https://www.opswat.com/technologies/sbom.

References

1. CISA SBOM FAQ
2. Bil Bensing: History of the SBOM
3. Cyclone DX: History
4. ISO Standards: ISO/IEC 5962:2021
5. Google: Introducing SLSA
6. Gartner: Innovation Insights for SBOM
7. Carnegie Mellon University Software Engineering Institute: Leveraging SBOMs for Risk Reduction
8. Gartner: Mitigate Enterprise Software Supply Chain Security Risks
9. Gartner: Leader's Guide to Software Supply Chain Security
10. Department of Commerce: The Minimum Elements for a Software Bill of Materials (SBOM)
11. NSA: Recommendations for SBOM Management

GET STARTED

Explore how OPSWAT can secure your file workflows and elevate your security strategy.

Talk to one of our experts today.

Scan the QR code or visit us at:
opswat.com/get-started
sales@opswat.com

OPSWAT.
Protecting the World's Critical Infrastructure

For the last 20 years OPSWAT, a global leader in IT, OT, and ICS critical infrastructure cybersecurity, has continuously evolved an end-to-end solutions platform that gives public and private sector organizations and enterprises spanning Financial Services, Defense, Manufacturing, Energy, Aerospace, and Transportation Systems the critical advantage needed to protect their complex networks from cyberthreats.

Built on a "Trust no file. Trust no device.™" philosophy, OPSWAT solves customers' challenges like hardware scanning to secure the transfer of data, files, and device access with zero-trust solutions and patented technologies across every level of their infrastructure. OPSWAT is trusted globally by more than 1,800 organizations, governments, and institutions across critical infrastructure to help secure their devices, files, and networks from known and unknown threats, zero-day attacks, and malware, while ensuring compliance with industry and government-driven policies and regulations.

Discover how OPSWAT is protecting the world's critical infrastructure and securing our way of life.
Visit: www.opswat.com