sonatype

The ULTIMATE SBOM GUIDE

Table of Contents

Congratulations on downloading this comprehensive guide to one of the most important elements of your software supply chain security strategy—the software bill of materials (SBOM). Sonatype has been committed to secure software for nearly two decades, and was recognized as a Leader in ***The Forrester Wave: Software Composition Analysis, 2024***. This independent report identified the "10 providers that matter most" and recognized Sonatype with the highest possible scores in the following areas: malicious package detection, software bill of materials (SBOM, policy anagement, and AI component analysis.

As modern software supply chains become more complex, **SBOMs** have emerged as a key tool for managing, sharing, and securing open source components. It's the only way to provide a detailed inventory of every component within an application, and it's an organization's best defense against potential legal culpability by showing that proper security measures were in place and that first-party and third-party components were managed effectively. Faster development, easier collaboration, and visibility into potential vulnerability or licensing risks are just some of the benefits SBOMs provide.

The resources compiled in this volume represent the latest industry research, compliance guidance, and insight from our customers and partners on SBOM best practices. This workbook has been specifically designed to help you and your organization apply best practices in SBOM management. Each section includes actionable insight and practical steps, and where applicable, we've included checklists and workbooks to help you gauge your organization's readiness. We hope you find this information useful, and we welcome the opportunity to **discuss any additional questions** about how you can use SBOMs to consistently deliver high-quality solutions to your customers.

FORRESTER®
WAVE LEADER 2024
Software Composition Analysis Software

Recognized as a Leader in *The Forrester Wave: Software Composition Analysis, 2024*

Sonatype was awarded with the highest possible scores in the following areas:

Malicious package detection

Policy management

Software bill of materials (SBOM):

- Ingestion and analysis
- Generation, export, and sharing

AI component analysis

INTRODUCTION:

The state of modern cybersecurity

With the explosion in the use of open source, the industry has rallied around the value of SBOMs. They offer an easy, scalable way to understand all of the components and dependencies contained in a piece of software. SBOMs provide a comprehensive inventory of an application's packages and libraries, enabling organizations to meet increasingly rigid regulatory requirements, address security vulnerabilities, and improve collaboration.

Just as open source changed how developers could accelerate innovation with third-party tools, the rise of artificial intelligence (AI) and machine learning (ML) models increases dependency and complexity with external components. SBOMs are essential for tracking the origin of components to facilitate efficient audits and compliance verification.

As software supply chain risks grow, SBOMs are becoming essential for:

- **Reduce open source risk** by evaluating and managing first-party and third-party risks
- **Prevent open source license risk** issues through improved observability
- **Reduce the risk of non-compliance** by keeping track of applied annotations and release status
- **Spend less time on incident investigation**, required remediation, and monitoring efforts
- **Sharing SBOMs at scale** with traceable and transparent vulnerability annotations (VEX)
- **Staying ahead of regulatory requirements** with creation, storage, and monitoring in one place

Using this guide, you'll learn about:

- Minimum requirements for SBOMs according to current regulations and industry best practices
- The immediate steps to take when getting started with SBOMs
- How to effectively integrate SBOMs into your workflows
- The specific rules and regulations for SBOMs as outlined in various legislation and cybersecurity guidance around the world
- How to evaluate your level of compliance/readiness
- We'll also share the most pressing SBOM questions we hear from customers and answers to those questions from Sonatype and industry experts

PART 1

SBOM S.O.S. 6 things to do right now

If you're just starting the SBOM journey, these initial steps can help you stay compliant, reduce vulnerabilities, and gain immediate insight across your software portfolio.

Grasp the regulatory landscape

Familiarize yourself with **applicable industry standards** and government-mandated requirements for SBOMs. For example, in the US, when the White House issued Executive Order 14028 on Improving the Nation's Cybersecurity, multiple agencies were tasked with defining best practices for ensuring the security of software supply chains. As a result, the Department of Commerce, in coordination with the National Telecommunications and Information Administration (NTIA), defined the "**minimum elements**" necessary for an effective SBOM. The National Cybersecurity Strategy also promotes the adoption of SBOMs as vital to a secure software development lifecycle. Similar legislation is being adopted around the world. In Europe, the **Network and Information Systems Directive (NIS2), the Cyber Resiliency Act (CRA)**, and the **Digital Operational Resilience Act (DORA)** all recognize the importance of having a full understanding of the software components being used in an application.

Understand the scale of your SBOM management needs

The number of SBOMs an enterprise can expect to manage is substantial, driven by the volume of both internal and external applications, their release/update frequencies, and regulatory retention requirements. Managing SBOMs encompasses first-party software released internally, third-party and commercial off-the-shelf (COTS) software that provides SBOMs, and archived binaries or legacy systems that might be uncovered during an SBOM implementation. This can add up fast. For example, we know the average enterprise manages more than 6,000 applications. Updated or released 12 times a year, this would generate 72,000 SBOMs annually. It's likely that you also have specific retention periods to comply with. For example, in the US, for certain federal agencies, this period is 7 years, bringing the number of SBOMs required to more than 500,000. This means you need to plan to store and manage these documents at scale, and to plan for the amount of total SBOMs to grow over time.

The average enterprise manages more than 6,000 applications, which would **generate 72,000 SBOMs each year if updates occur on a regular basis**. SBOM management at this scale requires world-class features and support.

Prioritize automation

SBOMs are not meant to be created, stored, and forgotten. But without automation, it's impossible to do anything with the sheer number of SBOMs created. Automation makes it possible to update your SBOMs regularly or after significant updates or changes, including adding new components, updating existing ones, and removing depreciating elements. Instead, consider an SBOM as a snapshot in time, continuously monitored to catch new vulnerabilities and intentionally malicious information. This approach is especially crucial for addressing zero-day scenarios and maintaining a secure and efficient software supply chain. Automating SBOM generation not only simplifies the development processes but also provides a holistic view of software supply chain health.

Integrate SBOMs into your software development lifecycle

Make SBOMs part of your standard SDLC process. Development teams need to see security as part of the solution instead of a gating factor, and getting their visibility into vulnerabilities earlier in the process will help. Building SBOM creation into the CI/CD pipeline and automatically generating a list of all open source or third-party dependencies makes the SBOM an artifact of each release, providing an immutable, historical record of all components and risks present at the time of release. Make this so much a part of your SDLC process that software only gets deployed if it has an SBOM.

Use SBOMs to your advantage

SBOMs aren't just another bureaucratic requirement; by providing a detailed breakdown of the packages and libraries included in an application, they provide an opportunity to build better software and identify vendors with better software. SBOMs make it easier to track issues and cross-reference those against a vulnerability database (for example, **NIST's National Vulnerability Database)** and take action to update or replace components that are flagged for policy violations or security risks. Vulnerabilities can be present in every level of your software stack, so regularly verifying the accuracy and completeness of the SBOM by comparing it against the actual software components deployed will help investigate and resolve discrepancies, resulting in better, more secure applications.

Protect your SBOMs, but get comfortable with the idea of sharing

Being subjected to SBOM regulations is unavoidable, which means sharing SBOMs with customers, partners, or regulatory bodies. This can be intimidating, but the right tools can make the process not just manageable but also give you the peace of mind that your software is in the best shape possible. For projects that use proprietary source code, consider limiting SBOM access to customers or qualified leads. But for open source projects, dependency information is already publicly available. Store SBOMs in a secure yet centralized and accessible location for compliance and vulnerability management and make a plan to limit access to authorized personnel or use access controls to ensure that only those who need access can view or modify it. You can also require digital signatures that provide a unique identifier to authenticate that the SBOM originates with you. Any changes to the SBOM will generate a new signature. Providing training for developers and stakeholders on the importance of security is another way to increase the sensitive mindset when it comes to SBOM management.

PART 2

Best practices for SBOMs in DevSecOps

Achieve comprehensive component visibility in your software supply chain and ensure compliance with software development and security regulations.

Automated SBOM generation

- **Automate for precision:** Leverage automation tools for each software build, ensuring your SBOM is always accurate and current.
- **Separate build and release:** Incorporate SBOMs within your software development life cycle (SDLC) to enable monitoring. Also ensure SBOM data is meticulously captured and securely retained for versions that are released, deployed, or shipped.

Integration with DevSecOps workflow

- **CI/CD pipeline embedding:** Incorporate SBOM generation and management tools within CI/CD workflows for automatic security assessments.
- **In-depth component scanning:** Ensure SBOMs are created with accurate identification tied to deep, timely, accurate data to ensure a proper view of risk.

Strategic utilization

- **Rapid vulnerability response:** Quickly identify and remediate vulnerabilities identified via SBOMs to ensure compliance and secure software components.
- **Assurance:** Maintain audit-ready compliance by importing and retaining every SBOM unlocking rapid response to incidents, audits, and compliance requests.

Collaboration and accessibility

- **Universal access:** Grant all relevant teams access to an SBOM application or interface to foster a collaborative security culture.
- **Targeted training:** Provide education on the advantages and interpretations of SBOMs, emphasizing security implementations.

Tools and services

- **Focus on integration and automation:** Opt for tools that offer seamless workflow integration, automate SBOM generation, and provide comprehensive scanning for security and compliance.
- **Choose dual-purpose tools:** Ensure your tools support both integrated SBOM generation during the SDLC and efficient management of 1st- and 3rd-party applications, enabling risk and compliance oversight across your software ecosystem.

Continuous monitoring and feedback

- **Alert system:** Implement an alert mechanism for newly discovered vulnerabilities in existing SBOMs that could be affecting your 1st- and 3rd-party software components.
- **Iterative improvement:** Establish feedback loops for continuous refinement of your SBOM strategy, adapting to emerging security challenges and tech advancements.

Your Checklist: Best practices for integrating SBOMs into DevSecOps

Complete the implementation steps below to effectively integrate your SBOMs into your DevSecOps workflow.

- [ ] **Evaluate current processes:**
Assess how existing development and security workflows align with SBOM capabilities.

- [ ] **Select appropriate tools:**
Choose SBOM and DevSecOps tools that offer easy integration, scanning capabilities, and support for SBOM management.

- [ ] **Automate SBOM integration:**
Automate SBOM generation within your CI/CD pipeline for consistent updates and scanning.

- [ ] **Procurement and 3rd-party SBOM management:**
Implement processes for ingestion and management of SBOMs from 3rd-party vendors, ensuring they meet your security and compliance standards.

- [ ] **Establish Governance, Risk, and Compliance [GRC] protocols:**
Integrate SBOM insights into your governance, risk management, and compliance (GRC) framework to enhance decision-making and regulatory adherence.

- [ ] **Enable teams:**
Educate development, security, and operations teams on utilizing SBOMs for enhanced security.

- [ ] **Make available to customers:**
Develop and implement a process to share verified SBOMs with customers, enhancing transparency and trust in your software's security and compliance.

- [ ] **Implement monitoring and risk assessment:**
Establish continuous monitoring of SBOM data for real-time threat and vulnerability assessment, ensuring immediate response and mitigation.

- [ ] **Monitor performance:**
Regularly assess the effectiveness of SBOM integration on security posture and make necessary optimizations.

PART 3

SBOM best practices for development teams

Strengthen your security posture with visibility into the software supply chain and comply with regulatory and vendor requirements for composition transparency.

Integrate SBOM management into your development lifecycle

- **Make SBOMs part of your standard SDLC process:** Building SBOM creation into the CI/CD pipeline and automatically generating a list of all open source or third-party dependencies and code moves vulnerability detection left and allows DevOps teams to detect vulnerable components before deploying the application.
- **Educate yourself on the relevant regulatory requirements:** Software supply chain security is getting a lot of attention these days, and SBOMs are taking center stage in this effort. Familiarize yourself with current federal and industry-specific regulations and adopt policies that reflect your needs.
- **Understand your toolchain:** Know what build tools and environments your project uses and which ones can generate SBOMs. Not every tool has a ready-made generator and you may need to create custom scripts
- **Check out community and support:** Since SBOM technologies are new, it's worth assessing the related (if any) communities and available vendor support when choosing SBOM tools.

Automate SBOM generation and updates

- **Plan to automate:** Where possible, integrate SBOM generation into your CI/CD pipelines to ensure updates with every build. Your build process will produce many SBOMs, so be prepared for scale and management
- **Choose the right tools:** Select an SBOM generation tool that integrates well with your specific development environment. Standardizing on one SBOM format will reduce complexity but may not always be feasible.
- **Automate policy support:** Equip development teams to know upfront during the design stage what open source or third-party products are approved and which ones to avoid.

Prioritize vulnerability tracking and management at all levels

- **Cross-reference with vulnerability databases:** be proactive in updating or replacing vulnerable components.
- **Plan for comprehensive coverage:** Remember that vulnerabilities can exist within every level of the software stack, so provide full coverage for top-level components as well as all nested dependents.

Treat SBOMs as sensitive information

- **Secure storage:** Many of these SBOMs may be legal or contractual documents. Treat them accordingly and store SBOMs in a secure yet accessible location for compliance and vulnerability management.
- **Sign your SBOMs:** Providing a digital signature to your SBOM provides a way to authenticate that the SBOM originates with you. These signatures provide a unique identifier, so any changes to the SBOM will generate a new signature.

Regularly review SBOMs for licensing and compliance issues

- **Include your legal team in the process:** Work with legal counsel to establish a policy on approved and disallowed licenses. Lawsuits for unintended license violations are a hidden risk, so familiarizing development teams with which licenses are acceptable is essential in the project planning process.
- **Automate reviews and keep a schedule:** Use automated tools to analyze SBOMs, reduce the time required, and eliminate errors.
- **Plan to self-check:** SBOMs that are inaccurate are worthless and can lead to legal or contractual problems. Plan to apply a self-assessment process that can check that software components can be traced through the build.

Use an exchangeable standard format as part of your SDLC

- **Adopt standard formats to simplify the process and minimize mistakes:** These formats allow SBOMs to be automatically generated during the development process.
- **CycloneDX** is an open-source standard developed by the Open Web Application Security Project (OWASP) community. It was designed specifically to bolster security across software supply chains and is known for its lightweight nature. It fosters an environment where adoption and integration into build pipelines are seamless and efficient.
  - Notably, CycloneDX is engineered for cyber risk mitigation, gaining the trust of critical sectors such as government and defense. It boasts compatibility with over 200 tools and extends its reach across more than 20 programming languages.
- **SPDX** is an open-source blueprint for SBOMs that simplifies the conveyance of essential details such as software names, versions, components, licenses, copyrights, and security references. It excels at reducing redundancies and streamlining distribution and compliance processes, backed by its ISO/IEC recognition as an international standard.
- **Key attributes of SPDX include:**
  - Precision in documenting software components, licenses, and other critical data, making it indispensable for compliance and legal frameworks.
  - Enables linking of artifacts to global reference systems like CPE, Package URL (purl), SWHID, enhancing security and management of software components.

Understand SBOM limitations

The objective of an SBOM is to provide a clear and comprehensive description of the contents of the software you deliver. Given the complexity of modern build processes and the relative newness of SBOM tooling, it's important to recognize that not every SBOM will be as comprehensive as possible and that most SBOM generators focus on one type of content. A considerable number of SBOMs will be required to cover all the content of a software application. The goal of SBOMs being aggregators of SBOMs from previous build steps has yet to be achieved.

PART 4

Preparing SBOMs for audits

Enhance your audit readiness to comply with federal and industry-specific SBOMs cybersecurity regulations.

Internal policy requirements

- **Set expectations for OSS components:** Define risk tolerance and rules for utilizing OSS and communicate these expectations across development teams.
- **Continuously monitor for violations:** Address violations faster by continuously monitoring SBOMs for potential risks for real-time response and mitigation.
- **Provide controls:** Document and enforce policies around what components are allowed into your supply chain and which are not.

Understand applicable cybersecurity requirements

- **NIST and CISA:** If you supply software to the US government, you must comply with NIST SP 800-218 and Cybersecurity and Infrastructure Security Agency (CISA) attestation attestation mandates.
- **PCI DSS:** The PCI Software Security Framework (SSF) applies globally to any organization that handles, processes, or stores payment card data. It ensures the security of payment software, with an emphasis on security integration in the development process.
- **EU CRA:** The Cyber Resilience Act applies to any company that sells physical products containing software in the European Union.
- **EU NIS 2:** The NIS2 directive applies to any company operating a digital service or serving a critical industry in the European Union.
- **DORA:** The Digital Operational Resilience Act (DORA) is a European Union-wide act that will require EU financial entities to implement operational and resilience strategies.
- **FD&C Act:** The United States Federal Food, Drug, and Cosmetic (FD&C) Act applies to any company selling medical devices.
- **FAR:** The Federal Acquisition Regulation (FAR), applies to any US company that develops software under contract with the US federal government.

Terms and Conditions

- **Anticipate updates to Terms and Conditions:** As awareness around cybersecurity requirements grows, terms and conditions will reflect the requirement for suppliers to provide SBOMs. SBOMs are also becoming increasingly common requirements for vendor contract renewals.

Operate at scale

- **Establish processes for regular SBOM generation:** To comply with DORA, FD&C Act, and FAR
- **Deliver secure software at scale:** Manage libraries and store components in a central repository and easily share them across the SDLC.
- **Produce a machine-readable SBOM:** An SBOM that can be automatically generated, updated, and analyzed makes identifying and mitigating potential risks faster and more comprehensive across different tools.
- **Separate build and release:** Incorporate SBOMs within your **software development life cycle (SDLC)** to enable monitoring. Also, ensure SBOM data is meticulously captured and securely retained for versions that are released, deployed, or shipped.

Continuous monitoring and feedback

- **Alert system:** Implement an alert mechanism for newly discovered vulnerabilities in existing SBOMs that could be affecting your first- and third-party applications.
- **Iterative improvement:** Establish feedback loops for continuous refinement of your SBOM strategy, adapting to emerging security challenges and tech advancements.
- **Internal audits:** Build an expectation with customers of proactive communication when critical vulnerabilities or license issues are discovered.

Your Checklist: Preparing SBOMs for audits

Complete the implementation steps below to effectively integrate your SBOMs into your DevSecOps workflow.

- [ ] **Create SBOMs throughout the release process:**
  Creating SBOMs for every application provides visibility into each version.
- [ ] **Automate SBOM creation:**
  Ensures each build has a corresponding SBOM for compliance or auditing purposes.
- [ ] **Centralize your SBOMs:**
  Provides a central location for access across your organization.
- [ ] **Include scan results with your SBOM:**
  Provides transparency and helps customers assess threat levels of specific components.
- [ ] **Establish Governance, Risk, and Compliance (GRC) protocols:**
  Enhances decision-making and governance, risk management, and compliance.

PART 5

The 10 most frequently asked SBOM questions answered

As a pioneer in software supply chain management, here are some of the most asked questions we hear from developers, organizations, and cybersecurity professionals around the world.

What are the minimum requirements for an SBOM?

In order to validate the authenticity of a particular software component, an SBOM has to provide identification, including the supplier and name, the component name, the version, dependency relationships, the author of the SBOM, and the time the data was added to the SBOM. Also required is support for automatically generating and parsing, with the goal of interoperability across organizations. **System integrators and software vendors** that supply software to the US government must also attest to compliance using instructions outlined in the Secure Software Development Attestation Form provided by the Cybersecurity and Infrastructure Agency (CISA) and the Office of Management and Budget (OMB).

What impact does EO 14028 have on supply chain regulations?

Recent legislation, including Executive Order 14028 and NIST SP 800-218 in the US and NIS2 and the Cyber Resiliency Act (CRA) in the European Union, are driving important awareness of the effectiveness of SBOMs. Other countries, including Germany, Japan, and Korea, are all looking at ways to legislate the use of SBOMs. In 2023, CISA launched its Secure by Design initiative with the participation of cybersecurity authorities from around the world to underscore the role security has throughout the software development lifecycle.

Are you seeing industries standardizing around SBOM formats?

Yes, standardization is one of the key pillars of SBOM management. The National Telecommunications and Information Administration (NTIA) recognizes three main formats for SBOMs: CycloneDX, SPDX, and SWID. These formats allow SBOMs to be automatically generated during the development process and ensure compatibility with other tools and systems so vendors and customers work with compatible tools.

Could you explain the VEX process and how it applies to SBOM vulnerabilities?

An SBOM provides a comprehensive list of software components, while VEX (Vulnerability Exploitability eXchange) documentation offers crucial insights into known vulnerabilities and exposures within these components. This integration of VEX data with SBOMs is not just a technical detail, but a significant step towards enhancing software security. By pinpointing the components that are truly at risk, you can effectively prioritize vulnerabilities that demand immediate attention. VEX-enriched SBOMs play a pivotal role in identifying and mitigating the most critical vulnerabilities within a specific application, thereby aiding organizations in adhering to regional regulations.

My organization has a wide range of security tools, how does Sonatype help?

Yes. The danger of having a scattershot collection of tools is that organizations often rely on providing reports to developers. But without guidance or policy control, developers just get inundated. The number of dependencies in a software application is overwhelming. A system that can provide prioritized, actionable guidance is necessary. Having the right tools in place is a good start, and Sonatype can help make sure the data those tools provide is actionable.

How does SBOM management change in a SaaS development model vs. a COTS?

It's important to build the SBOM generation and evaluation into your normal build testing workflow. From a COTS perspective, the ability to consume and evaluate SBOMs for policy guidelines is critical. Sonatype makes it possible to merge these two use cases using the same tool to evaluate both types of SBOMs for policy and potential security issues. Remediation is mostly done by the vendor In the COTS model, so it's important to set standards for vendor management.

As the use of AI in software development increases, how will SBOM management evolve?

In Sonatype's most recent State of the Software Supply Chain Report, three out of four DevOps leads reported concerns about the impact of generative AI on security, especially in open-source code. With the rise of AI, we need to be aware that if used incorrectly, it's going to make this problem worse, not better. SBOMs will need to accurately reflect the inclusion of AI components within software systems and emphasize transparency, providing insights into how AI components contribute to overall system functionality. SBOM tools also need to integrate with security scanning and vulnerability management systems to identify and address potential risks associated with AI components.

How can we automate the storage, management, and distribution of our numerous SBOMs?

There should be an emphasis on automation. Use tools to automate the process of collecting, managing, maintaining, and storing your SBOMs. It's not unusual for developers to publish multiple times a day, and application teams can't keep up. The more you can automate and integrate SBOMs and build them into the CI/CD pipeline, the more you can alleviate that pressure. This could include automating policy support so development teams know upfront during the design stage what open source or third-party products are approved and which ones to avoid.

How can I convince my organization about the value of SBOMs?

SBOMs are increasingly becoming necessary for doing business. For example, the FDA is no longer approving new medical devices without an associated third-party risk model. There are also emerging requirements like IEC (International Electrotechnical Commission) 63000 that define requirements for listing software of unknown origin. An SBOM is an automated, standardized answer to that requirement. Organizations that are taking SBOM management seriously recognize it as more than a compliance effort—it's about using it as a testing framework for developers to act on quickly. The faster you can alert developers to potential vulnerabilities, the quicker it can be addressed.

How can an SBOM help identify only snippets of a third-party component?

If the developer repackages dependencies, they will become apparent in the dependency hash section as they do not correspond with canonical package identities. Sonatype can highlight this and either mark it as 'similar to package-foo' or a 'completely unknown component.

Have more questions about SBOM management?

Reach out today and get advice from a Sonatype expert.

CONCLUSION

The sense of urgency around SBOMs

Legislative compliance is just part of the reason why SBOMs are important. Software liability is another growing trend, and SBOMs are the only way to provide a detailed inventory of every component within an application. An SBOM is an organization's best defense against potential legal culpability by showing that proper security measures were in place and that third-party components were managed effectively.

The increased emphasis on the value of SBOMs and the importance of good SBOM management is foundational to protecting the software supply chain, but navigating the best way forward can be daunting.

How SBOMs drive a stronger SCA strategy

One thing has become clear during this period of shoring up the security of our software supply chains—SBOMs are fundamental to an effective SCA strategy. They provide the transparency needed to identify, track, and manage open-source and third-party risk and make it possible to have comprehensive visibility of all software components, versions, and dependencies. This streamlines vulnerability management and accelerates incident response, allowing teams to quickly identify and remediate affected components when new threats emerge.

Ultimately, SBOMs transform SCA from a reactive process into a proactive, automated security practice, driving continuous improvement in software security and resilience. For more information about anything in this workbook or to learn about how Sonatype can help your organization manage SBOMs, **reach out to one of our experts**.

Simplify SBOM Compliance and Security Monitoring with Sonatype SBOM Manager

More than 70 percent of Fortune 100 companies manage their software supply chains with Sonatype, and our SBOM Manager has been developed to take the uncertainty out of SBOM collection and to monitor compliance. **Sonatype SBOM Manager** can help users assess the contents of an SBOM against regulatory or industry compliance standards. This tool provides specific details about the location of the affected file, where the file is referred to in other areas of the software, and recommendations on the next actions.

Discover how to apply these best practices to your organization — **connect with us** to learn more.

Sonatype is the software supply chain security company. We provide the world's best end-to-end software supply chain security solution, combining the only proactive protection against malicious open source, the only enterprise grade SBOM management and the leading open source dependency management platform. This empowers enterprises to create and maintain secure, quality, and innovative software at scale. As founders of Nexus Repository and stewards of Maven Central, the world's largest repository of Java open-source software, we are software pioneers and our open source expertise is unmatched. We empower innovation with an unparalleled commitment to build faster, safer software and harness AI and data intelligence to mitigate risk, maximize efficiencies, and drive powerful software development. More than 2,000 organizations, including 70% of the Fortune 100 and 15 million software developers, rely on Sonatype to optimize their software supply chains. To learn more about Sonatype, please visit www.sonatype.com.

Headquarters
8161 Maple Lawn Blvd,
Suite 250
Fulton, MD 20759
USA • 1.877.866.2836

European Office
168 Shoreditch High
St, 5th Fl
London E1 6JE
United Kingdom

APAC Office
60 Martin Place,
Level 1
Sydney 2000, NSW
Australia

Sonatype Inc.
www.sonatype.com