SOLUTION BRIEF

ox security

AppSec, uncomplicated: Everything you need to know about ASPM and more!

Software supply chain attacks have increased 742% in the past three years. Something's gotta give – but it doesn't need to be your security team, your SDLC, or your security posture.

Whether you're building an AppSec program for the first time, or working from a well-established base, Application Security Posture Management (ASPM) can help cut through the complexity, giving you the unity of vision you need to protect applications from code to cloud. Gartner forecasts that, by 2026, more than 40% of organizations developing proprietary applications will adopt ASPM.

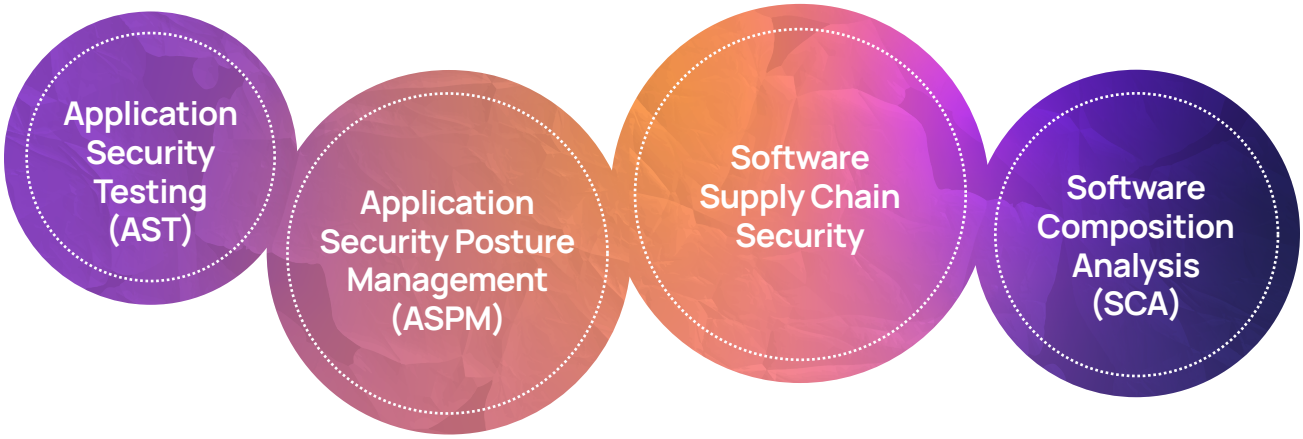
Here's what you need to know, and what to look for in an ASPM solution...

A software-defined storm

Ninety-one percent of organizations experienced at least one software supply chain security (SSCS) incident in 2023. Chances are the other 9% are riding their luck: Today's applications are complex, interconnected, and span multi-cloud, multi-access device landscapes. A single exploited vulnerability can trigger compromises that spread far beyond the initial point of weakness.

At the heart of the problem: Companies that aren't rooted in software development are building, developing,and shipping software at an accelerated pace – often without the AppSec tools and the CI/CD pipeline integration they need to make secure DevOps a reality. Under pressure to deliver, developers favor rapid release cycles enabled by third-party code libraries, tools, and processes.

More and more, developers are taking greater responsibility for application security. However, traditional AppSec tools weren't built for developers' use and can't keep pace with today's rapid development cycles. AppSec teams and developers need a solution that unifies processes and toolsets to mitigate risks – without slowing delivery and enhancing collaboration between teams. Old school can't cut it any more.



All of the tools, none of the clarity

Traditional AppSec and software security tools have largely been driven by scanning technologies.Good tools exist, but they only solve for one – maybe two – aspects of the software development lifecycle (SDLC).

The result? AppSec engineers are left manually piecing together data from disparate, fragmented solutions. Even then, the work isn't done: the data from those manual processes has to be enriched and correlated from third-party resources including the NVDD, CISA, KEV and other vulnerability trackers. More tools, more processes, more data, more processes, more gaps...Did somebody say "visibility"? How about "control"?

As shown below, whether standalone or combined, traditional AppSec tools can't provide the code-to-cloud visibility and manageability today's AppSec teams need to keep up with a radically transformed SDLC.

The inclusion of Application Security Orchestration and Correlation (ASOC) here underlines the expanding scope and near-constant evolution of AppSec: While it marked a significant advance on the road to a more unified, streamlined approach, ASOC lacks the proactive, comprehensive approach needed to handle the complexity of distributed environments and human processes. As we'll see, that's where ASPM steps in, building on ASOC's foundations to deliver a more comprehensive, risk-based approach to modern AppSec.

	AST	ASOC	SCA	Software Supply Chain Security
Focus	Static or dynamic assessment of applications for vulnerabilities or security flaws.	Continuously discover and mitigate application vulnerabilities and ensure compliance.	Continuously scrutinize and manage open-source components of applications.	Securing the code throughout the SDLC, all components, the development and build environments, and the distribution mechanisms.
Practices and Tools	Secure code practices, audits, reviews, and pen tests automated by SAST/DAST/IAST tools.	Orchestration and correlation of security and configuration issues. Includes automated remediation.	Specialized tools to automate open-source components vulnerability management, compliance, and licensing conflicts.	Ensure secure SDLC practices, use of SBOM for complete transparency into software components, vulnerability management, risk assessment, and secure handling and storage of code.
Gaps	▪ Testing is point-in-time. ▪ Automating requires code-specific scanners and more tools in the security stack.	Narrowly focused on centralized vulnerability management.	Concentrated only on open-source software components.	▪ Implementing controls can introduce friction and require renegotiation of processes. ▪ Inadequate handling of 3rd-party and proprietary code ▪ Heavy reliance on manual processes for changes or component scanning, slowing down development.

ASPM is here, and the revolution will not be siloed

Take everything we've just looked at, add the fact that the average security team now monitors 129 applications and sixty-eight or more technologies, and try not to black out under the strain of coverage and visibility gaps, excessive alerts, and laundry lists of "to dos." Something had to give, and it did...

For AppSec defenders, Application Security Posture Management (ASPM) is an emerging approach that unifies the separate AST, SCA, and software supply chain capabilities, providing more context and giving AppSec practitioners the ability to prioritize, fix, and track issues throughout the SDLC.

The evolution of ASOC into ASPM unifies application security practices across the SDLC, providing end-to-end visibility from code to cloud and cloud to code. Moving beyond the limitations of historical ASOC, AST and supply chain security tools include contextualized prioritization, automated response, and remediation capabilities, and (in Ox Security's case), a proprietary Pipeline Bill of Materials (PBOM) to provide a real-time list of software lineage, from the first line of code to release.

So far, so great. But many ASPM platforms continue to be out-foxed by the needs of fast-paced AppSec environments. They can't:

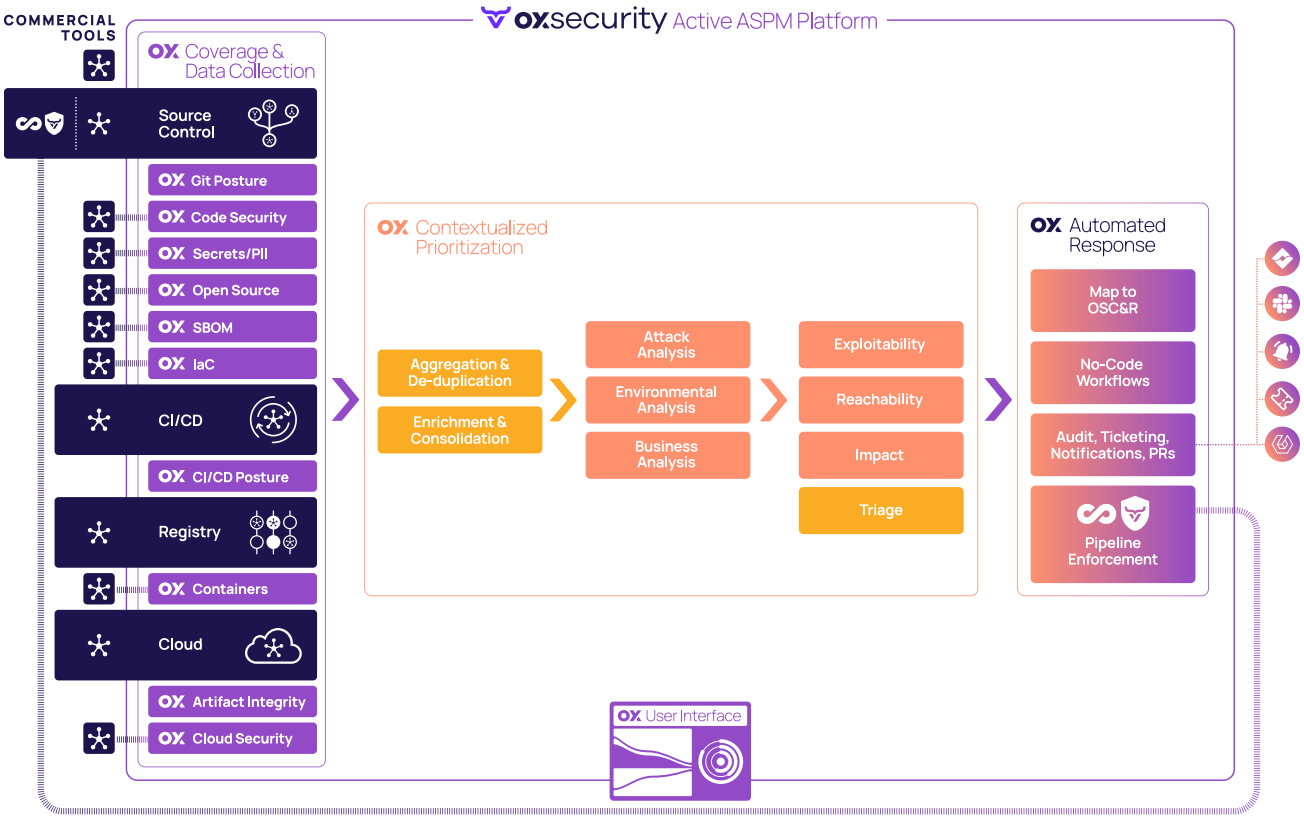
- Address the complexity of application environments
- Keep pace with application lifecycles
- Integrate seamlessly with DevOps practices
- Provide standardization
- Ensure complete visibility into third-party components
- Swiftly adjust to the dynamic threat landscape
- Reduce the strain on AppSec resources
- Adequately align with regulatory compliance
- Effectively support incident response and remediation

Introducing AppSec data fabric: the unifying layer for ASPM

A data fabric approach is at the heart of Ox Security's Active ASPM. Put simply, a data fabric is an architectural model that automatically aggregates disparate sources of data, normalizes it all, and then allows for correlation and deep analysis. This unique, consolidated approach ensures that AppSec and DevOps teams can stay on the same page, improve communication and collaboration, and increase velocity.

At its best, an AppSec data fabric enables more than data aggregation from a network of connectors; it generates actionable insights from data processing, ensuring alerts are actionable, prioritized, and contextualized – reducing manual effort and speeding up issue resolution. Data fabric brings flexibility and depth to AppSec risk management, enabling native scanning capabilities to be augmented by third-party integrations without generating additional alerts – the same normalization, correlation and deep analysis can be applied.

While many traditional AppSec tools leave defenders manually correlating third-party enrichment and struggling for visibility into the SDLC, Ox Security's data fabric approach provides end-to-end visibility from code to cloud. It allows for accelerated, more granular insights into your AppSec data, throughout the entire CI/CD pipeline. This innovative data fabric approach transforms our ASPM platform into a driver of faster, better, more informed risk decisions.



oxsecurity

The Magnificent Seven: key components you should look for in an ASPM platform

Taking data fabric as the base layer, ASPM platforms should give defenders the unified tools they need to secure their software supply chain and eliminate manual AppSec. The key benefits and capabilities for any ASPM platform should include:

1. What you can't see can hurt you: Enhanced visibility and eliminate blind spots

Let's start with the problem: Managing security postures across hybrid environments is tricky. Organizations with diverse and complex software supply chains are often in the dark when it comes to establishing a complete software inventory – including third-party components and dependencies, which are often the weak links in the supply chain.

ASPM platforms should provide defenders with the visibility and deep insights they need to make informed decisions and take quick action. However, some piece together so many third-party tools to scan for risks and vulnerabilities, they generate more data than teams can handle, making prioritization and follow-up difficult. Add poor data, unreliable normalization and correlation engines to the mix, and coverage gaps become an issue.

These coverage gaps create blind spots in security assessments, leaving applications vulnerable to exploitation. It gets worse: some ASPM solutions depend on static information feeds from individual tools – fundamentally limiting their ability to provide a dynamic, comprehensive view of security in the development environment.

What's the answer?

ASPM should provide comprehensive visibility

The most effective ASPM platforms provide comprehensive visibility – they identify and allow operators to manage vulnerabilities, dependencies, and compliance issues across the SDLC, which encompasses code, tooling, processes, and data from operational environments such as cloud platforms, containers, and physical infrastructure.

AppSec is a continuous process; your ASPM platform should integrate organizational source control, CI/CD pipelines, registries, and cloud infrastructures through APIs. This architecture allows the ASPM provider to collect and verify 3rd party data, correlate it, enrich it, then provide operators with the centralized visibility that allows them to manage applications and build environments from app design through deployment.

BOTTOM LINE: Many ASPM platforms provide good AppSec security postures, but lack capability to adequately cover CI/CD issues. This leads to a lack of visibility into code, configurations and dependencies as they pass through pipelines. An emphasis on static rather than dynamic analysis can also be a miss for the automated, accelerated pace of CI/CD processes. Finally, many platforms lack their own scanning capability, are limited to one type of AST tool and/or can't perform remediation – all of which add to exposure risk, difficulty in prioritizing, and inconsistent fix quality.

Ask your vendor...

- What support do you offer for development, test, and runtime environments, including cloud, containers and physical infrastructure?
- Can your product support integration with third-party IT and security tools?
- What types of native scanning does your product provide?
- Can your product support the full breadth of AppSec-focused capabilities, including SAST, DAST, IAST, secrets, container security, CI/CD configuration, and more?
- How comprehensive is visibility into the software supply chain?
- What types of intelligence enrichment does your offering provide?

2. Container exposure: Improve cloud-to-code traceability

ASPM operates at the application layer, overseeing both on-premises and cloud-based environments. Look for tools that improve visibility and traceability within container environments and across hybrid cloud environments. This coverage will improve visibility and reduce manual triage and response efforts. ASPM platforms should substantially shorten response times by providing advanced, automatic triage and prioritizations capabilities, including severity criteria for container exposure.

BOTTOM LINE: If an ASPM platform doesn't substantially shorten resource times and link container security issues directly to code, keep looking...

Ask your vendor...

How do you link container security issues back to code, and assess the risk of the issues discovered?

What specific features do you offer that improve the visibility and traceability of deployed applications back to code?

What evidence do you provide that helps security teams collaborate with developers, and substantially shorten response times?

3. Bring in the BOM squad: Strengthen software supply chain security

Between 40-80% of code in new software projects comes from third parties, much of it from open-source projects. Understanding the nature of weakness and vulnerability in code is crucial for AppSec teams looking to develop a proactive security approach.

Despite widespread awareness, threats like cross-site scripting (XSS) are being introduced during development processes regularly. Managing security in the accelerated development environment is tricky, the likelihood of vulnerabilities slipping through the cracks or being introduced through recycles or third-party code is high. And if your AppSec team is wading through 100,000+ alerts, triage gets overwhelming quickly. That's where software bill of materials (SBOM) and software composition analysis (SCA) come in...

Software security and compliance, sorted

To identify all vulnerabilities, AppSec teams need a dynamic list of everything a piece of software includes and everything it has gone through – from the first line of code to release. A simple inventory of components in production apps can't cut it: effective ASPM platforms support SBOM and provide dynamic capability to track the entire software lifecycle,

including all version lineage, SLSA dev (supply chain levels for software artifacts), SaaSBOM, security tool results and build hashes. The most effective ASPM platforms reflect the realities of SaaS, APIs, and cloud components in modern development environments: they enable SaasBOM. These represent a strategic framework to provide organizations with a detailed, actionable understanding of their SaaS ecosystem, enabling rapid identification and remediation of vulnerabilities in software components. API BOM capability brings the same strategic approach to maintaining a secure API environment.

A further drawback for many ASPM platforms is that they don't integrate SCA, relying instead on external tools that, between them, produce excessive, irrelevant alerts or false positives. Others take a one-size-fits-all approach, which can't take into account the unique risk profile of each organization – from prioritization to the tools and resources available.

BOTTOM LINE: Effective ASPM platforms optimize the vulnerability management process, aggregating issues to simplify remediation, and prioritize vulnerabilities based on exploitability, reachability and unique impact, resulting in significantly faster response times. By integrating SBOM, SCA, SSCS and SAST into ASPM, organizations get a coherent, context-specific baseline for security assessments. The ability to trace source code to all its sources — along with accompanying vulnerabilities — allows teams to move beyond simple identification and into holistic risk management.

Ask your vendor...

- Can your product produce an SBOM for all software in your environments, from design to release, and keep the SBOM current through material code changes?
- If so, how does your tool determine the SBOM (i.e., through and integration or native assessment)?
- How does your SBOM reporting support collaboration and communication with partners, vendors, buyers, and executive teams?
- What types of export features for reporting does your technology provide?
- Can you guarantee that your SBOM outputs are defensible?

4. Get critical insight into application risks: Enhance prioritization and provide richer context

Many ASPM vendors provide data deduplication and aggregation. This should be table stakes. But ASPM platforms that bring an additional layer of enrichment and triage can give AppSec teams enhanced, critical insight into application risk. With software vulnerabilities, context is everything, and so it's important that the vendor's solution can answer the following questions:

- Can an attacker reach a specified vulnerability?
- If it's reachable, can an attacker exploit it?
- What could an attacker achieve by exploiting this vulnerability?
- What is the business impact of an exploit, data leak, or other compromise

BOTTOM LINE: With enriched ASPM, AppSec and developer teams can evolve from simply identifying software with certain CVEs to understanding and flagging libraries that are badly maintained/ have poor hygiene/are out of date, learning which applications are affected by what vulnerabilities, following dependencies and downstream impacts, uncovering attack paths, and much more. This comprehensive approach adds the contextual component missing from siloed, traditional AppSec and DevOps processes.

Ask your vendor...

What intelligence sources do you use beyond simple data feeds?

Does the tool enrich with relevant information about the threat, application, and code component?

Can the tool facilitate root cause identification?

5. Meantime... Rapid response and remediation workflows

Many of today's ASPM platforms cannot detect software vulnerabilities early in the development cycle. They can't identify vulnerabilities or exploits quickly, and lack the ability to assess code from initial design through runtime. In other words: They create friction, slow down release cycles, and add to the pressure while DevOps and AppSec teams negotiate over release deadlines and vulnerability remediation.

Reducing mean-time-to-respond (MTTR) is crucial for reducing risk. Choose an ASPM platform that supports this goal through intuitive response and workflow automation – facilitating vulnerability identification and rapid, effective response strategies that don't slow down the release cycle.

BOTTOM LINE: **No-code workflow automation simplifies the creation of intuitive, customizable responses. ASPM platforms with automated actions reduce resolution times and accelerate release cycles by scaling security efficiently across any environment.**

Ask your vendor...

- How sophisticated is automation and orchestration?
- Can you build custom policies and workflows? How easily and quickly can you create custom workflows?
- Can the tool facilitate root cause identification?

6. Get out of the way: Seamlessly integrate with DevOps practices

AppSec teams have to align with the speed and agility of development cycles, centering software security without impacting the continuous delivery pipeline. Today's ASPM tools integrate with developer processes and workflows, so engineers can follow secure development best practices within their working environments.Not all of them make it easy, but...

Some ASPM solutions provide no-code workflow automation with drag-and-drop interfaces to simplify custom workflow creation, automate ticketing and enable granular policy enforcement. Automated response helps to streamline communication about fixes, reducing friction in remediation and response processes.

BOTTOM LINE: **Look for a platform that automates remediation, and reduces the need for extensive teams by providing easy-to-build, customizable workflows, automating certain risks while maintaining hands-on oversight for others.**

Ask your vendor...

- How does your ASPM solution integrate with existing DevOps tools (e.g., CI/CD pipelines, Git, Jenkins, etc.)?
- Does your platform offer no-code or low-code workflow automation?
- How does your solution automate remediation and response for common security issues? What level of granularity is available in policy enforcement, and how customizable is it?

7. Join the dots: Connect to commonly understood frameworks for AppSec

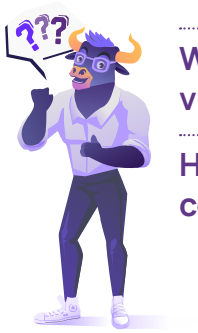
By understanding how attackers view and target the supply chain attack surface – and by using a common language to describe threats – AppSec, DevOps and security teams can align more effectively to mitigate risk at every stage of the SDLC, sometimes avoiding its introduction in the first place.

The Mitre ATT&CK framework transformed how cybersecurity teams describe and understand tactics and techniques. Inspired by its success, the OSC&R framework gives defenders a common language and attackers' view that is specific to the software supply chain, taking tools to the next level and helping both AppSec and AppDev teams keep up with the latest attack trends. But not everyone gets it…

Model the behavior you want

ASPM solutions must provide a structured approach to identifying, categorizing, and analyzing software development, operations, and supply chain threats. Accurate risk assessment extends beyond consideration of your defenses: You must also consider how attackers operate, and where they can breach your applications. ASPM solutions that integrate the OSC&R framework can help you to converge application detection and response (ADR) into the ASPM platform, enabling complete threat modeling that exposes the gaps in your defenses and the most vulnerable areas in your software supply chain. By focusing on these low-hanging fruits, you immediately craft and action plan that tackles the highest-risk areas, while allowing for continued improvement and advanced prioritization, including attacker tactics and techniques.

BOTTOM LINE: Like the MITRE framework before it, OSC&R marks a significant transformation in how AppSec teams address software security challenges. Coupled with ASPM, it brings a new approach, incorporating the attacker point of view and typical attack stages into the ASPM, giving teams an additional tool through which they can understand tactics, techniques and traditional processes attackers use to exploit an organization.



Ask your vendor…

What methodologies does your solution use to prioritize threats and vulnerabilities within our software supply chain?

How do you ensure continuous monitoring of application security posture, correlating vulnerabilities and attacker tactics and techniques?

Eliminate the chaos of managing siloed data from disparate sources

The OX Security Active ASPM platform unifies application security across the SDLC. Unlike the patchwork of features in other tools, OX delivers a tightly integrated set of capabilities that empower development and AppSec teams to deliver more secure applications at the scale and speed of today's business environment.

The OX AppSec Data Fabric is the key to our platform and the reason OX excels against other ASPM, AST, and AppSec tools. Unlike other ASPM tools that stitch technologies together through connectors, OX was purpose-built for comprehensive AppSec posture management, combining 10 native scanning solutions with source data from 3rd-party integrations.

The OX Platform intertwines deep insights from SCA, AST, Secrets, IaC, CI/CD, SBOM, cloud, and posture to reduce AppSec alert noise by 90%, provide detailed insights about each application and its vulnerabilities, and offers step-by-step recommendations and auto remediation that lower AppSec risk.

The "secret" to OX's efficacy is reliable, prioritized, and contextualized evidence-based data that incorporates reachability, exploitability, and business impact — specifically for your business.

- The OX AppSec Data Fabric delivers complete visibility and reduces manual AppSec.
- OX's proprietary OSC&R framework helps practitioners understand the software supply chain and is focused on the attacker TTPs that pose the greatest risk.
- OX prioritizes remediation over problem identification by automatically enriching and contextualizing collected data to streamline remediation. AppSec efforts are focused on effective vulnerability management and targeted root cause analysis.
- OX offers no-code workflow automation to reduce resolution times, eliminate manual efforts, and accelerate release cycles.

Learn more about how OX is going beyond traditional ASPM technologies and addressing the problems that are most critical for AppSec and DevOps teams:

WWW.OX.SECURITY/BOOK-A-DEMO/

About OX Security

At OX Security, we're simplifying application security (AppSec) with the first-ever Active ASPM platform offering seamless visibility and traceability from code to cloud and cloud to code. Leveraging our proprietary AppSec Data Fabric, OX delivers comprehensive security coverage, contextualized prioritization, and automated response and remediation throughout the software development lifecycle. Recently recognized as a Gartner Cool Vendor and a SINET 16 Innovator, OX is trusted by dozens of global enterprises and tech-forward companies. Founded by industry leaders Neatsun Ziv, former VP of CheckPoint's Cyber Security business unit, and Lior Arzi from Check Point's Security Division, OX's Active ASPM platform is more than a platform; it empowers organizations to take the first step toward eliminating manual AppSec practices while enabling scalable and secure development.

oxsecurity