SANS | Research Program

Spotlight

ASPM: Understanding the New Application Security Landscape

Written by **Chris Edmundson** and **Jack Poller**

March 2025

seemplicity

©2025 SANS™ Institute

Introduction

Security teams need a smarter, scalable, and more proactive approach to address modern software security requirements. Malicious actors continue to prey on the challenges of rapid software development cycles and cloud computing adoption, adding to existing pressure to establish visibility and streamlined remediation across the entire application life cycle. This is where an application security posture management (ASPM) solution comes in (see Figure 1).

Every day, organizations face a barrage of security threats targeting their applications. These range from sophisticated zero-day exploits to injection attacks such as SQL injection, which can manipulate databases. Cross-site scripting (XSS) allows attackers to inject malicious scripts into web pages. Broken access control can lead to unauthorized data access. Cryptographic failures can expose sensitive information. The rise of cloud-native applications, microservices architectures, and rapid deployment cycles has created an incredibly complex security landscape that traditional approaches struggle to handle.

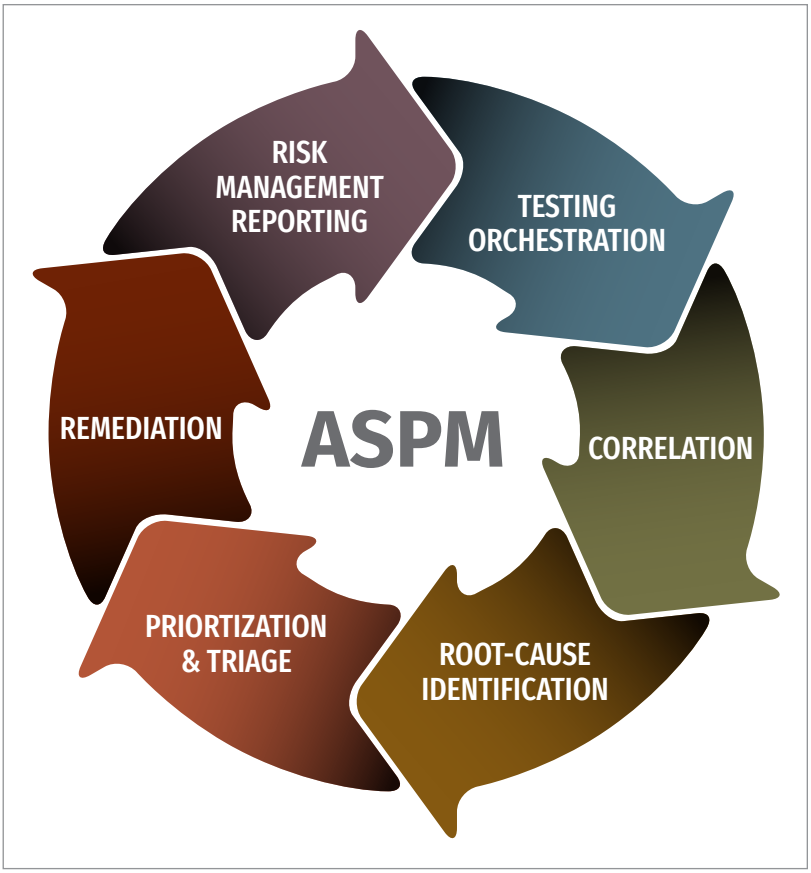


Figure 1. ASPM Life Cycle

Consider some of the most common threats we're seeing today. Bad actors are constantly exploiting insecure design practices, probing for vulnerable dependencies, and exploiting misconfigurations in cloud services. Supply chain attacks have become particularly concerning—remember the SolarWinds incident, where hackers compromised SolarWinds' application build system to insert malicious code that was then automatically distributed to roughly 18,000, leading to wide-scale compromises that went undetected for months? That showed us how compromised dependencies can have devastating downstream effects.

Known threats aren't the only challenge. Zero-day vulnerabilities—previously unknown security flaws that attackers discover and exploit before developers learn about and fix them—can emerge at any time. And the increasing use of open source components means that a vulnerability in a single popular package can potentially affect thousands of applications. Add to this the growing sophistication of attackers who are using automated tools and AI to find and exploit vulnerabilities, and you have a perfect storm of security challenges.

Building Applications: The Security vs. Speed Dilemma

Building secure applications in today's environment is like trying to build a fortress while simultaneously renovating it and keeping it open for business. Development teams are under pressure to deliver new features quickly, while security teams need to ensure that every change doesn't introduce new vulnerabilities. This tension often leads to what we call the "security vs. speed" dilemma.

Modern applications are incredibly complex, often comprising hundreds of microservices, multiple cloud services, and countless third-party dependencies. Each component introduces its own set of security considerations. A single application might use containers, serverless functions, APIs, and traditional server components, each requiring different security approaches and tools.

The challenges are multifaceted and interconnected. Development teams need to move fast and deliver features continuously, but the traditional approach of security reviews at the end of development simply doesn't work anymore. By the time you discover the security issues, the cost of fixing them has increased dramatically.

Dependencies are constantly changing and updating, with a typical Node.js application potentially having hundreds or even thousands of nested dependencies. Keeping track of vulnerabilities in all these components and ensuring timely updates is a massive undertaking.

Furthermore, configuration mistakes can be catastrophic—a single misconfigured S3 bucket or an overly permissive API endpoint can expose sensitive data. With cloud-native applications, the attack surface has expanded dramatically.

How Does DevSecOps Change the Security Landscape?

This is where DevSecOps comes in—it's an approach that aims to integrate security practices throughout the software development lifecycle (SDLC) rather than treating it as an afterthought. The idea is to make security a shared responsibility between development, operations, and security teams.

DevSecOps promotes a fundamental shift in how organizations approach security. It's an integrated approach that coordinates development, security, and operations within the SDLC. The methodology emphasizes embedding security practices from the inception of the project rather than treating security as a final step. By involving security practices early and throughout the development process, DevSecOps aims to reduce vulnerabilities, accelerate secure software delivery, and foster a culture where security is a shared responsibility among all team members. DevSecOps uses automated tooling to continuously monitor, test, and mitigate risks, ensuring security measures are consistent and scalable. This approach breaks down traditional silos, promoting collaboration and enabling faster, more secure software releases.

Although DevSecOps is a crucial step forward, implementing it effectively across multiple projects and environments presents its own challenges. Organizations often struggle with tool sprawl, where different teams use different security tools, creating silos of security information and inconsistent security practices.

What Makes Multi-Project Security So Challenging?

One of the biggest headaches for security teams today is managing security across multiple projects and environments. Large organizations might have hundreds of applications running across development, staging, and production environments. Each environment might use different cloud providers, different technology stacks, and different security tools.

Managing security in this context is like trying to play three-dimensional chess. Security teams need to maintain comprehensive tracking of vulnerabilities across all applications and environments while ensuring they consistently apply security policies everywhere. They must maintain visibility into the security posture of each application and prioritize security issues based on real risk to the business. Perhaps their most challenging task is to coordinate remediation efforts across multiple teams, each with their own priorities and constraints.

After integrating its vulnerability scanners with the Seemplicity platform, one SaaS organization was able to reduce its backlog of over 250,000 findings by 98% before the triage process even began.

Traditional security tools, which often focus on single aspects of security or single environments, struggle to provide the comprehensive visibility needed to manage this complexity effectively. This fragmented approach leaves organizations vulnerable to security gaps and makes it difficult to maintain a consistent security posture across the entire application portfolio.

How Does ASPM Address Modern Security Challenges?

ASPM offers a holistic approach to managing application security that provides comprehensive visibility and control across your entire application portfolio.

Think of ASPM as your mission control center. It brings together security data from multiple sources—vulnerability scanners, cloud security tools, code analysis tools, threat intelligence, and more—to provide a unified view of your application security posture. But it's more than just a dashboard—it's an intelligent platform that helps you understand and improve your security posture.

What Are the Key Features and Benefits of ASPM?

ASPM platforms are designed to address the complex challenges of modern application security through several key capabilities. At their core, they provide unified visibility by aggregating security data—everything from code security issues and open source vulnerabilities to cloud misconfigurations, API security issues, and runtime security events—from multiple sources to provide an integrated and centralized, context-rich view of application security. ASPM platforms normalize and deduplicate this data, increasing your review and triage efficiency.

Using ASPM, development, security, and operations teams can see everything in one place and understand how different security issues relate to each other, developing a comprehensive understanding of your application security posture. The benefit is clear: There is no more jumping between different tools and dashboards, no more reviewing duplicate issues, and no disconnects because two different tools describe the same issue differently.

Risk-based prioritization is another crucial feature of ASPM platforms. Not all vulnerabilities are created equal, and ASPM uses context-aware risk scoring to help you focus on what matters most. ASPM platforms analyze vulnerabilities, identify root causes, and prioritize risks.

A global beverage producer deployed Seemplicity and realized a 60% reduction in findings.

Advanced platforms will integrate with threat intelligence feeds to detect emerging risks in your application portfolio. This feature identifies potential exposure to newly discovered vulnerabilities and assists in identifying likely attack vectors and implementing preventive measures. Integrating threat intelligence helps to provide real-time updates and assists in threat modeling.

When prioritizing risks, ASPM platforms consider factors such as threat intelligence, the exposure level of the affected component, potential business impact, exploitation difficulty, and the presence of compensating controls. This intelligent prioritization ensures your teams can focus your limited resources on fixing the issues that pose the greatest risk to your business.

ASPM platforms provide context-driven automation and streamline workflows throughout the SDLC, replacing error-prone, manual workflows with customizable no-code automation. Automating remediation workflow processes eliminates complexity, accelerates task execution, and supports scalable operations. Using automation increases your triage efficiency, plus you can automatically route remediation tasks with precision for consistent, scalable remediation without disrupting your development timelines.

By using Seemplicity, an American EV manufacturer reduced mean time to remediation (MTTR) by 60%, returning 75% of remediation time back to the security team.

ASPM platforms integrate seamlessly with popular ticketing systems like ServiceNow and Jira, enabling real-time synchronization and effortless collaboration. Teams can update tickets, add notes, and track progress directly within their preferred tools, ensuring smooth task transitions, cross-functional alignment, and efficient workflows. This also supports bi-directional collaboration and reduces friction among teams.

The platforms provide tailored remediation plans that prioritize tasks by identifying who needs to act, what to address, and how to resolve issues. These plans dynamically align with team workflows, ensuring targeted and efficient remediation. These integrations ensure your teams can continue using the tools they always have, minimizing their learning curve and keeping the remediation process organized with clear roles and responsibilities.

ASPM platforms excel at policy enforcement, enabling you to define and automatically enforce security policies across your entire application portfolio, including minimum security standards for new applications, required security controls for different types of applications, and configuration baselines. The automation extends to security-related tasks such as creating and assigning security tickets, triggering remediation actions, and generating compliance reports.

ASPM platforms typically include robust features for managing compliance requirements. They often come with prebuilt compliance frameworks for standards like SOC 2 and ISO 27001, automated compliance reporting, and the ability to maintain comprehensive audit trails. This makes it significantly easier to demonstrate compliance and prepare for audits.

Using ASPM's tracking and reporting capabilities gives you real-time insights into critical metrics such as open vs. closed findings, SLA performance, and ticket status across teams. This streamlines your governance and compliance efforts and helps fuel process improvements and SLA compliance. You also can track other key metrics such as mean time to detection (MTTD), mean time to resolution (MTTR), and vulnerability trends, which will give you a better understanding of the improvements to your security posture.

A SaaS organization saved 400 hours in the first year by using Seemplicity ASPM to automate ticket creation.

Advanced ASPM platforms extend metric reporting with AI analysis to forecast expected workload based on vulnerability discovery rates, remediation rates, and other trends. This helps your security teams understand and dispatch tasks to the right team for action, further ensuring you meet SLA expectations.

Perhaps most importantly, ASPM platforms are designed to integrate seamlessly with modern development processes. They provide integrated development environment (IDE) plug-ins for real-time security feedback, integrate with CI/CD pipelines, and offer developer-friendly security guidance including automated fix suggestions. This integration helps shift security left and makes it easier for developers to write secure code from the start.

Why Should You Implement ASPM Now?

The security challenges facing modern applications are not going away. If anything, they're becoming more complex as we adopt new technologies and development practices. Traditional approaches to application security, with their siloed tools and manual processes, simply can't keep up.

ASPM offers a way forward. By providing comprehensive visibility, intelligent prioritization, and automated workflows, ASPM platforms help organizations reduce security risk by identifying and addressing the most critical issues first. They improve efficiency by automating security tasks and reducing tool sprawl, while enabling faster development by integrating security into development processes. Organizations can maintain compliance more easily through automated controls and reporting, and scale security practices across large application portfolios.

Most importantly, ASPM enables your shift-left, shift-right, shift-everywhere security model, where you incorporate application security considerations throughout every part of your SDLC. This gives you early risk detection, where you identify vulnerabilities during coding and testing phases to reduce remediation costs; embeds security practices into development workflows, fostering a security-first culture; and enforces secure coding standards and guidelines to prevent common vulnerabilities.

When your applications are the lifeblood of your business, you can't afford to take a piecemeal approach to application security. ASPM provides the comprehensive, intelligent approach needed to protect your applications effectively. The question isn't whether you need ASPM—it's how quickly you can implement it to start getting better visibility and control over your application security posture. The sooner you start, the sooner you can begin reducing your security risk and building more secure applications.

Sponsor

SANS would like to thank this paper's sponsor:

