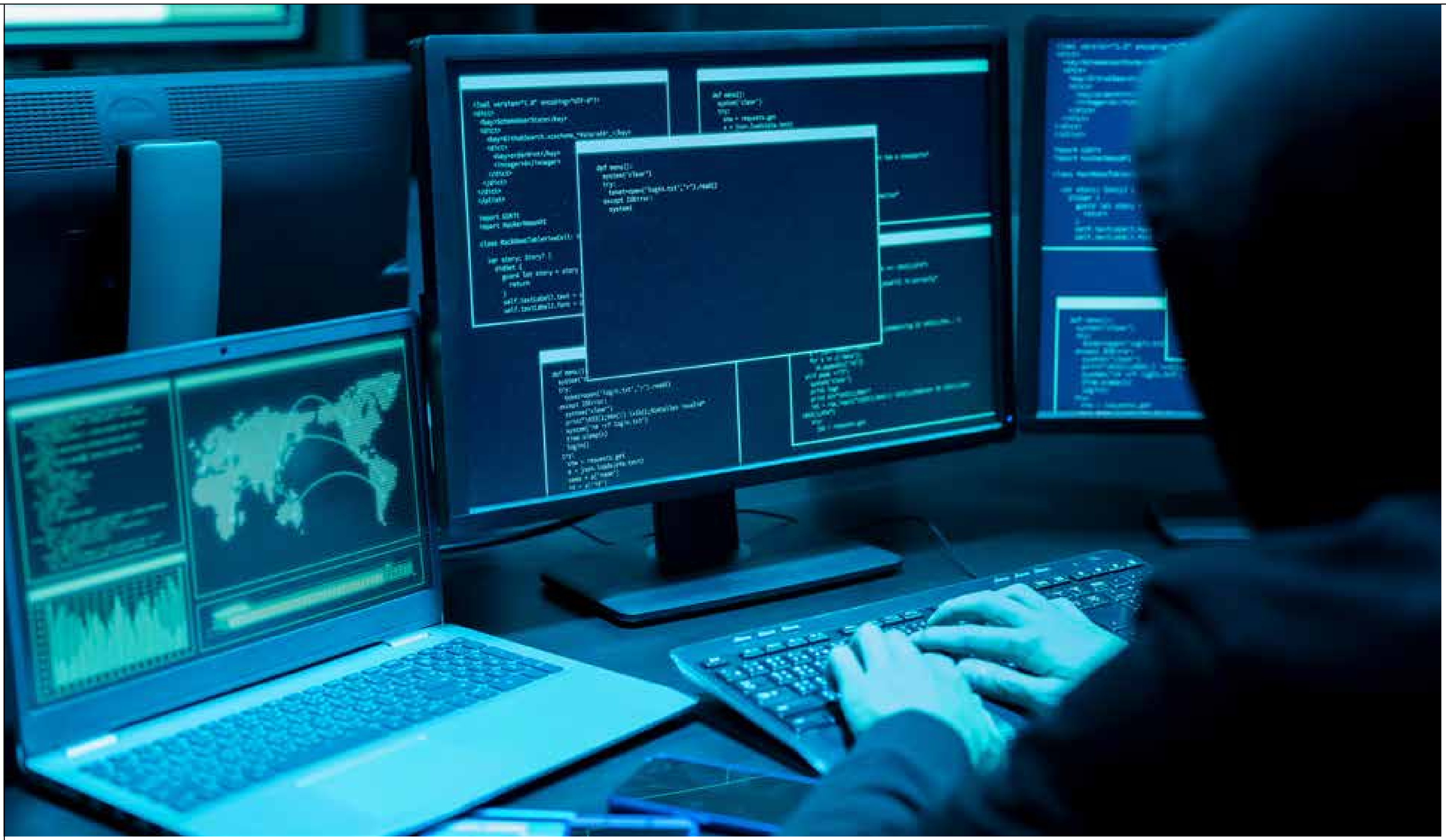

Picture credit: AdobeStock

HACKER'S EYE VIEW

Neil Roseman *outlines the difference checkbox vulnerability management and real AppSec risk mitigation can make*

In enterprise applications, systems, and infrastructure, vulnerabilities are an unavoidable reality. The problem is not that they exist, but whether organisations have the right tools and strategies to identify, prioritise and remediate the vulnerabilities that truly matter.

Too often, vulnerability management is reduced to a compliance exercise – checking off boxes to meet regulatory or customer demands. But a 'checkbox approach' creates overwhelming noise for developers and security teams, draining resources while leaving exploitable risks untouched. True application security (AppSec) risk mitigation requires shifting from a volume-based mindset to a risk-first, offensive, attacker-informed approach.

Modern digital infrastructure runs on software, from business-critical systems to consumer apps and cloud services. As demand accelerates for faster releases and more interconnected solutions, development teams are under pressure to deliver at speed. Even when security testing is integrated into pipelines, time pressures often force teams to prioritise productivity over thoroughness – and that means vulnerabilities slip through.

In 2024 alone, over 40,000 new CVEs were published, a nearly 40 percent jump over the previous year. But not all vulnerabilities carry the same risk. Treating them as equal creates a false sense of security and wastes precious time.

Not all vulnerabilities carry the same risk and treating them as equal can waste precious time

Neil Roseman is the CEO of Invicti Security. During his decade at Amazon.com as Vice President of Technology, he led the development of foundational systems including the Marketplace Platform, Digital Media Technologies and Worldwide Retail Software systems.

CODING TOOLS

The adoption of AI-powered coding tools has surged, offering efficiency gains but also introducing new security challenges. Existing research has shown that, under experimental conditions, AI code generation models frequently output insecure code. Additionally, as noted by security and academic researchers, AI code assistants can invent nonexistent package names. In a recent study, researchers found that about 5.2 percent of package suggestions from commercial models didn't exist, compared with 21.7 percent from open-source or openly available models. These hallucinated packages can open doors to supply chain attacks if attackers create malicious packages matching those invented names.

DAST plays a crucial role in mitigating these risks by dynamically testing the actual behaviour of AI-generated code in a live environment, ensuring that vulnerabilities are identified and addressed before deployment.

AVOIDING THE TRAPS

Organisations today face two competing pressures: deliver software faster to stay competitive and meet growing demands for robust vulnerability management to pass audits, satisfy customers and comply with regulations. Many fall into the trap of focusing on what auditors want to see – collecting reports, scanning for the sake of scanning and fixing every finding without context. This checkbox approach can mislead teams into thinking they are secure. In reality, it often results in wasting resources on low-risk, theoretical issues; missing the real, exploitable vulnerabilities that attackers prioritise; and burning out development teams with alert fatigue.

Imagine comparing two vulnerabilities: a firewall misconfiguration that only an attacker with system-level access can exploit and an SQL injection vulnerability on a public login page that allows unauthenticated access to sensitive data. Both are important, but the SQL injection represents a far higher and more immediate risk. Yet without a system to validate exploitability and prioritise based on attacker pathways, many organisations treat these findings with equal urgency.

This approach undervalues real threats and overvalues minor issues. Worse, it squanders developer attention on false positives and non-critical findings, delaying the remediation of truly dangerous vulnerabilities.

Many vulnerability management programmes rely heavily on static application security testing (SAST) and software composition analysis (SCA). While essential for identifying code-level and dependency risks, these tools often produce long lists of issues without context or prioritisation. Without a filtering layer, teams are left sorting through theoretical risks without knowing which can actually be attacked.

Advanced dynamic application security testing (DAST) tools fill this gap. By interacting with running applications, DAST identifies vulnerabilities from an attacker's perspective, showing which issues are exploitable in real-world conditions. This 'outside-in' testing provides the crucial validation layer needed to cut through the noise.

Traditional penetration testing offers valuable, targeted insights by using human ingenuity to mimic attacker behaviour. However, pen tests are usually periodic, expensive and time-consuming.

A 'CHECKBOX APPROACH' CAN BE OVERWHELMING FOR SECURITY TEAMS AND DRAIN RESOURCES

For most enterprises undergoing constant digital transformation, continuous security visibility is essential. Modern DAST tools provide automated, scalable and continuous 'hacker's-eye' perspectives, uncovering real risks on demand.

To move beyond checkbox compliance, organisations need to combine the right tools with the right priorities. A DAST-first approach means:

Focusing on real risk:
Scanning live applications to identify vulnerabilities that attackers can actually exploit, not just theoretical code or component issues

Validating findings with proof:
Using proof-based scanning to eliminate false positives and confirm exploitability

Prioritising remediation efficiently:
Helping development and security teams concentrate on what matters most, reducing wasted time on non-critical issues

This doesn't mean discarding SAST or SCA – they remain crucial for comprehensive coverage. But DAST provides the frontline gauge of application security posture, ensuring that teams address the vulnerabilities posing the most immediate and tangible risk.

Ultimately, modern AppSec must go beyond meeting compliance requirements. In Verizon's 2024 Data Breach Investigations Report, vulnerability exploitation accounted for 20 percent of all breaches, a year-over-year increase of more than one-third. Attackers aren't looking for checkbox compliance; they're hunting for oversights and exploitable gaps.

By adopting a DAST-first, risk-focused mindset, organisations can reduce security debt and cut through alert fatigue; prioritise resources where they matter most; and build a scalable, proactive security programme that protects against real-world threats.

Checkbox vulnerability management may fulfil minimum requirements, but it won't stop determined attackers. To mitigate real AppSec risk, organisations need the right tools, the right context and a risk-first approach that turns vulnerability management from a compliance burden into a competitive advantage ●