Issue 110 | January 2026

NTT DATA

Radar

The Cybersecurity Magazine

Cybersecurity as a business enabler

By Maria Pilar Torres Bruna

Welcome, 2026! We're kicking off a new cycle shaped by a technological landscape that's as exciting as it is challenging. Businesses are increasingly dependent on the technology that supports them, and the rapid evolution of artificial intelligence—from predictive to generative and now into an agentic era—is driving the creation of digital agents capable of amplifying human work.

Quantum computing, which is being talked about more and more each day, will no longer be a distant concept but will become a reality in just a few years. Many organizations are already exploring new business models and efficiency opportunities based on this technology, while also preparing for the **new cyber threats** it will bring.

Those of us working in cybersecurity share the excitement of the entire tech sector regarding these transformations. However, we also understand that cybersecurity is no longer just a business accelerator: today, it is a **fundamental enabler**. Without cybersecurity, there simply is no business.

At the start of this year, I want to share the **three key areas** on which a CISO focuses—or should focus—their efforts. These directly address the main pain points that any organization faces regarding security, and they are the pillars we promote at NTT DATA as a global company. The services we will offer in the coming years are built around them:

1. Risk management and proactive compliance

Every cybersecurity initiative must start from a clearly identified risk and contribute to mitigating it. Likewise, it is essential to maintain continuous review to anticipate new risks that emerge as attack techniques become increasingly sophisticated.

2. Enabling a secure business

The CISO must act as another strategist within the C-level, ensuring that cybersecurity objectives are fully aligned with corporate goals. Their role also involves building **value stories** that reinforce the trust of customers, employees, and stakeholders in the organization.

3. Cyber resilience

Resilience has become an essential concept. Any disruption results in economic losses, but the real difference lies in an organization's ability to **recover quickly** from an incident and minimize its impact on the business.

In this 2026, we remain committed to helping organizations strengthen themselves around these three pillars. I am convinced that those of us working in cybersecurity play a key role in the evolution of companies in the coming years, and that's why we do such a fascinating job.

The NTT DATA team wishes you a happy and cyber-secure 2026!

Maria Pilar Torres Bruna
Head of Cybersecurity IBIOL

Shai-Hulud 2.0: the day the worm understood the entire chain

Cyber chronicle by Marlon Nivia Devia

During September and November of 2025, the software industry discovered that the era of supply chain attacks had not yet reached its peak—it was only just beginning. What had previously been considered a risk associated with using third-party components evolved into a structural threat capable of compromising not only the code, but also the people who write it, package it, automate it, and deploy it.

The appearance of the Shai-Hulud worm and, a few months later, its more advanced variant, Shai-Hulud 2.0, marked a turning point for the technology ecosystem and cloud development platforms. The industry realized that the problem extended far beyond npm; it was about the trust placed in an ecosystem of automated processes that no one questioned.

The first wave arrived in September 2025, when researchers revealed a self-propagating package poisoning attack in npm. It was classified as a massive infection—dozens of packages modified, thousands of compromised downloads, and a fast-moving domino effect whose impact seemed, at first, confined to the repository. That assessment turned out to be, in hindsight, an optimistic mirage. In November 2025, Shai-Hulud 2.0 emerged: a stealthier worm, far more aware of its surroundings, and with ambitions that went well beyond code distribution. The attack no longer affected only published versions but also digital identities directly linked to GitHub, AWS, Google Cloud Platform, and Azure. The attack surface had expanded from a package.json to the very infrastructure where every piece of software is developed, integrated, and deployed.

The shift in strategy was reflected in its new capabilities. Shai-Hulud 2.0 stole npm tokens, GitHub credentials, and API keys, but it also used those secrets to infiltrate the native credential-management services of the three major cloud providers: AWS Secrets Manager, Google Secret Manager, and Azure Key Vault. It even targeted legacy systems like Azure Pod Identity, still present in many Kubernetes clusters.

The worm didn't just capture what was static; it understood what was dynamic: the narrowly scoped permission, the pipeline variable, the key that opens production from an environment that should never have had it.

And if the theft failed, the malware resorted to its final move—a destructive behavior capable of wiping entire directories, as if it understood that in a broken supply chain, destruction can be just as profitable as theft.

The real impact wasn't measured in infected packages, but in exposed secrets. According to later investigations, approximately 400,000 raw credentials were collected and spread across tens of thousands of public repositories, left openly accessible to anyone who found, analyzed, or reused them. What was most alarming was discovering that many of these tokens were still active when the campaign came to light.

The industry was still debating how many versions had been affected when the real question surfaced: who now had access to this data, and for how long? What began as a technical incident in npm ended up impacting continuous integration platforms such as GitHub Actions, Jenkins, GitLab CI, and AWS CodeBuild, compromising Linux container–based pipelines that automated publishing and deployment processes. The attack proved that it wasn't necessary to breach production if you could control the entire factory that produced the software.

For the development community, Shai-Hulud 2.0 represented more than a worm: it was a signal that the trust model of modern software needed to be rethought. Open source is built on collaboration, but extreme automation is built on faith—on the assumption that every package is safe, every token is protected, and every script executed during installation was placed there with good intentions. The worm demonstrated that in a world where installing is equivalent to executing, every line of downloaded code is a security decision.

One year, two variants, and an obvious message: the threat no longer enters through the application—it enters through the people who build it.

Shai-Hulud didn’t just steal secrets; it exposed an uncomfortable truth: the software development chain is only as strong as its weakest dependency and only as secure as the most forgotten token hiding in an environment variable. Protecting it requires looking beyond the repository and accepting that every automation—no matter how useful—can become an unsupervised execution serving the attacker.

Marlon Nivia Devia
Cybersecurity Engineer

OWASP TOP 10 2025: better programmers or better frameworks?

Article by Martín Bedoya Rodriguez

In recent years, modern frameworks for developing web applications, mobile apps, and APIs have improved significantly—not only in speed and usability, but also in security. Technologies like Spring Boot, .NET, FastAPI, React, and Flutter have incorporated protection mechanisms that help prevent common vulnerabilities. However, despite these improvements, there is still a gap between the tools available and the level of awareness development teams have regarding secure programming practices.

The shift has also been reflected in how software is structured. Instead of monolithic applications, modular architectures are now preferred, promoting separation of responsibilities. This makes it easier to implement security best practices from the design phase. However, using a modern architecture does not in itself guarantee that software is secure. Security still depends largely on the knowledge and judgment of those who design and implement the solutions.

With the OWASP TOP 10, it’s possible to understand how software threats have evolved. It’s an ordered list that groups the main categories of vulnerabilities most exploited over the last four years, making it a perfect guide for development teams to prioritize security activities throughout the software lifecycle. The latest version, OWASP TOP 10:2025, includes major changes that reflect how cyberattacks have evolved and the impact they have had on organizations.

One of the most striking changes is the drop of injection vulnerabilities to fifth place. In 2017, this category held the top spot; in 2021, it fell to third place. This shows that the controls implemented by frameworks have had a positive effect. Most frameworks include sanitization mechanisms that prevent this type of vulnerability without requiring direct intervention from the programmer. Even so, legacy applications or the incorrect use of framework capabilities still pose a risk.

In contrast, the top spot continues to be held by access control vulnerabilities—failures that allow users to access information or functions they shouldn’t.

This type of vulnerability is harder to mitigate automatically because it depends on how permissions and roles are defined within the software. It requires deliberate decisions from developers, who must thoroughly understand how the business works in order to implement effective controls.

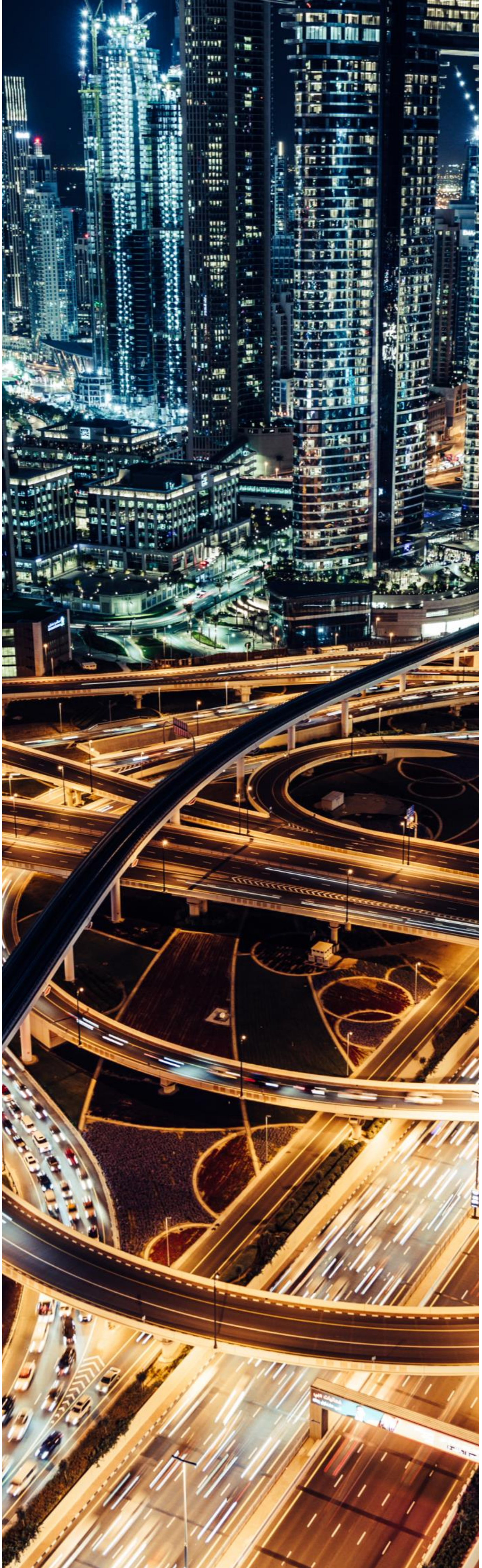
Another important addition in the 2025 edition is the inclusion of the category “software supply chain failures,” which reflects the risks of relying on external libraries without validation. Today, it’s common for an application to depend on dozens of components developed by third parties, and a single vulnerable dependency is enough to compromise the entire system. This category highlights typosquatting, a recent technique involving the renaming of public libraries and infecting them with malware in hopes that unsuspecting developers will import them.

OWASP TOP 10:2025 shows significant progress in the security offered by development frameworks, which has helped reduce certain historic vulnerabilities. However, it also makes clear that vulnerabilities directly tied to human decision-making still persist. Authorization logic mistakes, poor dependency management practices, or failures due to lack of knowledge demonstrate that the developer remains a key actor in securing the software lifecycle.

In short, although frameworks, tools, platforms, and software development methodologies have evolved to provide greater security, responsibility remains shared. Development teams must incorporate best practices from the very beginning of the software process and understand that security is not just a technical matter—it is a strategic business necessity.

OWASP TOP 10:2025 not only highlights the most exploited vulnerabilities of the past four years, but also invites a rethinking of how software is built in an increasingly complex environment. Instead of reacting to vulnerabilities, the trend is clear: prevent them from the start through secure design, well-trained teams, and a culture of responsible development.

Martín Bedoya Rodriguez
Cybersecurity Expert Engineer



OWASP Top Ten 2025: from secure code to a secure ecosystem

Article by Evelyn Terrones Romero

La seguridad en el desarrollo de aplicaciones ha evolucionado a gran velocidad. Lo que antes era una disciplina centrada en corregir errores en el código, hoy se ha convertido en una gestión integral que abarca todo el ecosistema de *software*. Este artículo muestra la evolución del OWASP Top Ten, analiza las principales novedades en su edición 2025 y compara los cambios más relevantes respecto a la versión anterior (2021), con el objetivo de entender mejor los riesgos actuales y cómo prepararnos para mitigarlos.

OWASP Top Ten

The OWASP Top Ten is an open, global project that identifies the main security vulnerabilities in applications and has become a de facto standard in risk management for secure development. Its approach prioritizes the risks with the highest impact and most frequent exploitation, making it one of the most influential references in application security.

OWASP 2025: Notable updates

The OWASP Top Ten 2025 is the eighth edition since its launch in 2003 and remains the global reference document on the ten most critical risks in web applications. This version introduces a shift in perspective: while it continues to focus on structural issues, it expands its view toward risks arising from the operational environment, the software supply chain, and the handling of exceptional conditions.

What does the new 2025 version include?

- Two new categories.
- Name and scope changes in several existing categories.
- Risk consolidation to group them by root cause, not just by how they manifest.

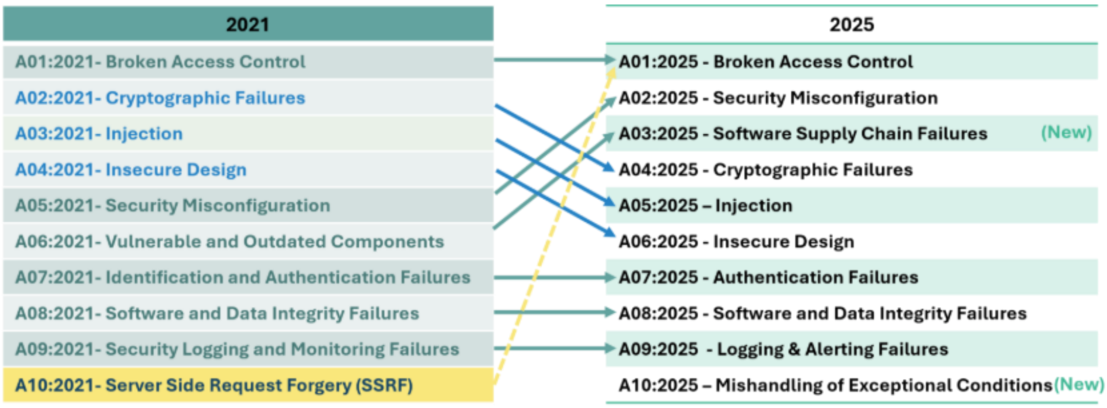
Below are the main changes featured in this new edition:

A03: Software supply chain failures

This new risk ranks third and replaces the former “Vulnerable and Outdated Components” category from 2021. It now encompasses not only the use of insecure libraries but the entire dependency ecosystem: malicious packages, contaminated scripts, compromised pipelines, and errors in secret management. The emphasis is on strengthening third-party management.

A10: Improper handling of exceptional conditions

This new category addresses failures in error handling, uncontrolled time-outs, logical errors in abnormal states, leakage of sensitive information through error messages, and poorly managed exceptions that can open critical security gaps. These issues, once considered operational, are now real attack vectors that allow the exposure of confidential data or the execution of malicious logic. In 2025, OWASP makes it clear that an application can be technically correct and still vulnerable if it does not respond securely to the unexpected.



2021	2025
A01:2021- Broken Access Control	A01:2025 - Broken Access Control
A02:2021- Cryptographic Failures	A02:2025 - Security Misconfiguration
A03:2021- Injection	A03:2025 - Software Supply Chain Failures (New)
A04:2021- Insecure Design	A04:2025 - Cryptographic Failures
A05:2021- Security Misconfiguration	A05:2025 – Injection
A06:2021- Vulnerable and Outdated Components	A06:2025 - Insecure Design
A07:2021- Identification and Authentication Failures	A07:2025 - Authentication Failures
A08:2021- Software and Data Integrity Failures	A08:2025 - Software and Data Integrity Failures
A09:2021- Security Logging and Monitoring Failures	A09:2025 - Logging & Alerting Failures
A10:2021- Server Side Request Forgery (SSRF)	A10:2025 – Mishandling of Exceptional Conditions(New)

Merging of SSRF with Loss of Access Control

Server-Side Request Forgery (SSRF), previously found in category A10:2021, is now incorporated into A01:2025 "Loss of Access Control." This change reflects that SSRF should be understood as an access control issue rather than an isolated vulnerability. It reinforces the focus on access to internal resources, APIs, and backend services, which today represent one of the main attack vectors in cloud environments.

Rise of the Security Misconfiguration category

Category A05:2021 "Security Misconfiguration" rises to A02:2025, acknowledging that many current failures aren't found in the code itself, but in how environments are implemented: default credentials, misconfigured permissions, exposed services, insecure policies, missing security headers, and more.

Injection and Cryptographic Failures drop a few positions

These categories fall in the OWASP Top Ten 2025 not because they have lost severity, but because the threat landscape has changed. Other risks now appear more frequently and represent more widely exploited attack vectors.

What this new edition teaches us:

- **Security is no longer only the responsibility of the development team:**

The new categories reflect that DevOps, infrastructure, and security teams must work together. Good code deployed in a poorly configured environment is still vulnerable.

- **Dependencies are your responsibility:**

It is essential to scan and manage dependencies, monitor vulnerabilities, and apply zero-trust principles from the design phase.

- **Errors are also gateways for attacks:**

The new emphasis on handling exceptions and unexpected conditions shows that system resilience is not merely operational. A poorly managed error can easily become a vulnerability, which is why anticipating and controlling these failures is an essential part of an effective security strategy.

The **OWASP Top Ten 2025** marks a new stage in application security maturity. Today, risk lies not only in the source code, but also in how we integrate, deploy, configure, and operate our software. This more holistic approach requires a true DevSecOps culture based on collaboration across different teams.

In an era of rapid development, external dependencies, and continuous deployments, knowing and applying the OWASP Top Ten is not just a best practice—**it is a necessity.**

Evelyn Terrones Romero
Cybersecurity Expert Analyst

The evolution of the OWASP ecosystem

Trends by Diego Carreño

OWASP has ceased to be just that list of vulnerabilities we see in every pentest report and has become the silent operating system that orchestrates modern software security. It's no longer just about the Top 10, but an entire ecosystem of standards (ASVS, MASVS, SAMM, API Security, LLM Top 10, AI Testing Guide, among others) that shapes how we design, develop, test, and govern applications, APIs, and AI systems.

Today, talking about OWASP means talking about an "operating system" for software security: a set of standards, controls, practices, and methodologies that shape everything from backlog planning to technical compliance audits. It's not just about protecting code—it's about designing organizations that think, build, and secure software holistically.

From checklist to AppSec "operating system"

The first major sign of this shift was OWASP ASVS (for web applications and services) and MASVS (for mobile applications), which define security levels (L1, L2, L3) and clear requirements that translate into policies, user stories, acceptance criteria, and testing scopes. Around them, a dense and growing ecosystem has formed:

- SAMM, as a maturity model for software security programs with an evolutionary approach.
- The Web Security Testing Guide (WSTG) and the Mobile Application Security Testing Guide (MASTG), serving as security testing catalogs that become regression test suites for web and mobile.
- Guides such as Cheat Sheet Series and Proactive Controls, which translate defensive coding into concrete, actionable practices.
- The Threat Modeling Project, along with tools like Threat Dragon and approaches like Cornucopia, bringing threat analysis into the backlog from the design phase.
- Intentionally vulnerable labs such as Juice Shop and WebGoat, used to train teams and validate static and dynamic analysis rules.
- And multiple specialized Top 10 lists (API Security Top 10, Top 10 for LLM Applications, etc.) that are already shaping how we test systems based on generative AI.

The underlying trend is clear: organizations are no longer consuming these projects independently. They are assembling them as modules of a single, coherent system that cuts across the entire business.

The practical result is that OWASP stops being a list consulted at the end to check for vulnerabilities and becomes the layer that structures everything from the initial idea to production.

OWASP as a "common language" between business, development and risk

One of the most visible advances is the use of OWASP as a common language among areas that historically spoke different dialects. Business operates in terms of risks and KPIs, development talks about bugs and technical debt, and risk and compliance focus on controls and regulations. OWASP is beginning to act as a translator between all of them. Here are some examples we've seen in 2025:

- Product owners, together with security analysts, set the target security level for each initiative in terms of ASVS or MASVS and incorporate it as a non-functional requirement in the backlog.
- Risk and compliance map frameworks such as PCI-DSS, NIS2, or local regulations to OWASP requirement families (authentication, logging, cryptography, etc.).
- Internal audit uses SAMM to assess capabilities, roadmaps, and evidence of continuous improvement—going beyond isolated controls.
- Development and testing factories standardize secure user story templates, acceptance criteria, and test cases based on ASVS, WSTG, MASTG, and the Cheat Sheets.

The result is that a conversation that used to happen in three different languages now begins to share a common dictionary. When someone says "we're going to bring this API to an ASVS Level 2 and cover the API Security Top 10," everyone understands what that implies—and, more importantly, they can measure whether it's being met.

OWASP as the backbone of automation

The other major driver of change is automation. OWASP has become the taxonomy that many organizations use to orchestrate their DevSecOps pipelines, correlate findings, and prioritize remediation.

AST, DAST, and IAST scanners tag vulnerabilities with references to OWASP (ASVS, API Top 10 2023, LLM Top 10, etc.). In many organizations, all those results are consolidated into a single application security dashboard that groups them under the same schema and, from there, CI/CD pipelines apply different "OWASP profiles" depending on the application type and its criticality.

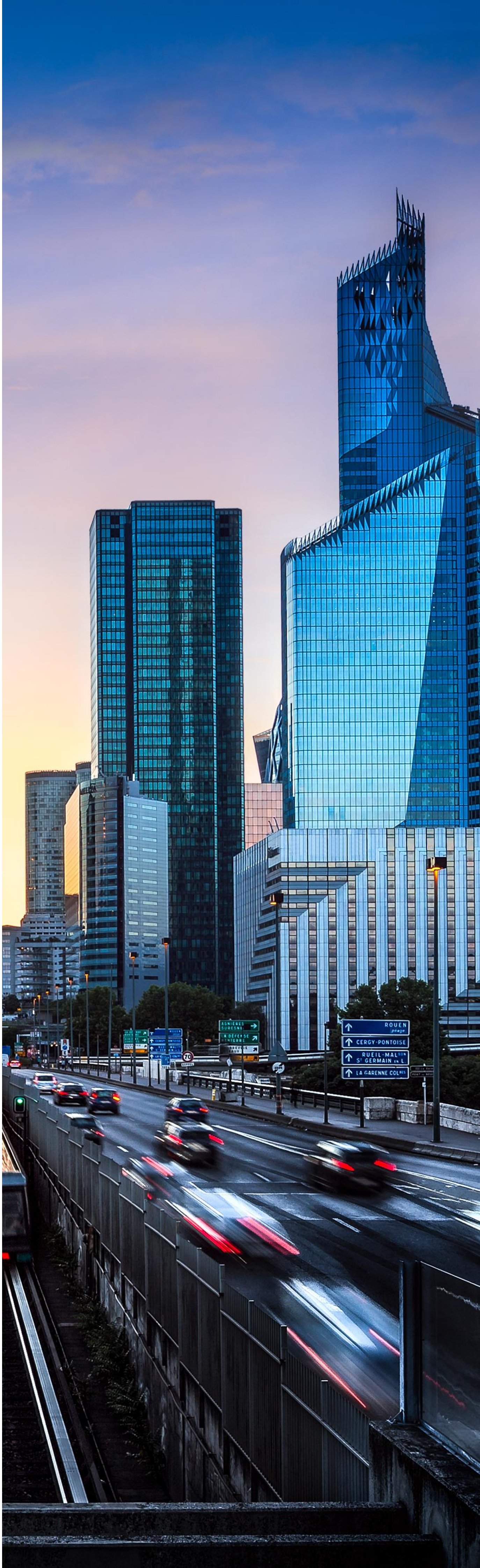
Even generative AI assistants for secure development are being trained with OWASP controls and guides: ASVS, MASVS, Cheat Sheets, WSTG, MASTG, and the newly released AI Testing Guide. This ensures that design recommendations, defensive coding tips, and testing guidelines are aligned with recognized standards from the very beginning.

Conclusion: the next competitive advantage will be speaking OWASP fluently

Organizations that embrace this movement as a strategic decision (and not "just another reference") will be the ones able to align business, development, risk, and audit under a single language; industrialize controls and testing without losing traceability; and adopt new technologies without reinventing their security model from scratch each time.

OWASP therefore stops being something we look at only when something goes wrong and becomes the security operating system that defines how we build what we want to go right. And that—far from being a trend—is likely to shape the next decade of secure development.

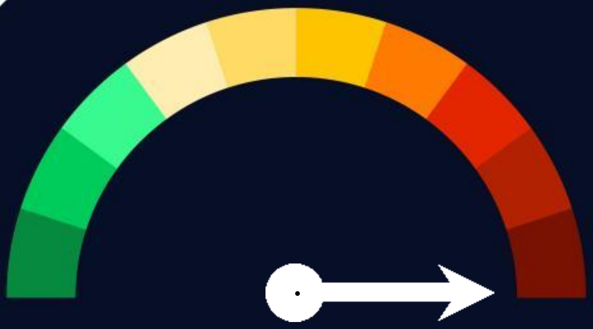
Diego Carreño
Cybersecurity Lead Analyst



Vulnerabilities

Critical vulnerability in React Server Components

Date: December 3, 2025
CVE: CVE-2025-55182



CVSS: 10

CRITICAL

Description

A critical-severity vulnerability has been reported in React server functions.

React provides tools and integrations that bundlers and frameworks use to execute code on both the client and the server. React translates client requests into HTTP requests that are forwarded to the server, which in turn translates them into function calls and returns the results.

An unauthenticated attacker could craft a malicious HTTP request targeting a React server so that, once translated, it results in code execution on the system.

Solution

It is strongly recommended to immediately update to the patched versions:

- React Server Components versions 19.0.1, 19.1.2, and 19.2.1.
- If your application uses the @vitejs/plugin-rsc framework, update to @vitejs/plugin-rsc@0.5.3 or later.
- For Next.js, versions 15.x and 16.x must be updated to the following patched releases:15.0.5, 15.1.9, 15.2.6, 15.3.6, 15.4.8, 15.5.7, and 16.0.7.
- For 14.3.0-canary.77 or later, downgrade to the stable 14.x version or to 14.3.0-canary.76.

Affected Products

Some of the affected products include:

- react-server-dom-webpack (React Server DOM Webpack package)
- react-server-dom-parcel (React Server DOM Parcel package)
- react-server-dom-turbopack.paquete dom del servidor react (react-server-dom-parcel)

References

- nvd.nist.gov
- www.incibe.es

Vulnerabilities

Critical vulnerability in Apache Tika

Date: December 4, 2025
CVE: CVE-2025-66516

CVSS: 10

CRITICAL

Description

A critical XML External Entity (XXE) injection vulnerability has been identified in multiple Apache Tika components.

This flaw allows a specially crafted PDF file containing malicious XFA content to trigger the loading of external XML entities during processing. As a result, system files may be exposed, which in certain environments can enable more severe attacks.

The vulnerability extends the scope of a previously identified issue (CVE-2025-54988), as it also affects additional modules and Apache Tika versions where the vulnerable code resided in different internal packages, making detection more difficult.

Solution

It is strongly advised to immediately update the affected versions to those in which the fixes have been applied:

- Tika-core: version 3.2.2
- Tika-parser-pdf-module: version 3.2.2
- Tika-parsers: version 2.0.0

Affected Products

The affected packages include the following versions:

- Tika-core: versions from 1.13 to 3.2.1
- Tika-parser-pdf-module: versions from 2.0.0 to 3.2.1
- Tika-parsers: versions from 1.13 up to versions prior 2.0.0

References

- [thehachernews.com]
- [incibe.es]

Android fixes 107 vulnerabilities in its December security patch

Date: December 1, 2025
CVE: CVE-2025-48631 and 106 more

Critical

Description

Android has released its December security patch, addressing a total of 107 vulnerabilities. Among them are 7 critical vulnerabilities and 98 high-severity vulnerabilities.

The vendor has reported indications of active exploitation of vulnerabilities CVE-2025-48572 and CVE-2025-48633. The former could allow an attacker to perform a privilege escalation, while the latter corresponds to an information disclosure vulnerability.

Among the vulnerabilities, the critical CVE-2025-48631, located in the framework component, stands out, as it could allow a remote denial-of-service attack without requiring additional privileges.

Affected Products

The products affected by the update are as follows:

- Android Open Source Project (AOSP): versions 13, 14, 15, and 16.
- Components from Arm, MediaTek, Unisoc, and Qualcomm.

Solution

It is recommended to apply the security patches released by the vendor.

References

- source.android.com
- incibe.es

Sneeit Framework fixes a remote code execution (RCE) vulnerability

Date: December 8, 2025
CVE: CVE-2025-6389

Critical

Description

A critical vulnerability has been identified in Sneeit Framework, a component widely used by multiple premium WordPress themes and templates.

The vulnerability allows an unauthenticated remote attacker to execute arbitrary PHP functions through a crafted request to the framework.

The flaw resides in a function that processes user-supplied input without proper validation, enabling arbitrary function execution on the server. This could lead to backdoor installation, creation of unauthorized administrator accounts, or full compromise of affected websites.

Affected Products

The products affected by the vulnerability are as follows:

- All versions of Sneeit Framework up to and including version 8.3.
- Any WordPress theme or template that incorporates this version of the framework.

Solution

The developer recommends:

- Updating to Sneeit Framework version 8.4.
- Additionally, reviewing configurations, administrative users, and any potential indicators of compromise.

References

- techradar.com
- nvd.nist.gov

Events

NIST Small Business Cybersecurity Webinar

20 January

NIST will offer a virtual webinar, via Zoom for Government, aimed at helping small and medium-sized businesses protect Controlled Unclassified Information. During the session, the new “Small Business Primer” for SP 800-171 Revision 3 will be presented, explaining its key requirements. NIST experts will provide guidance on how to begin implementing these security practices and will answer questions from attendees.

Link

II DORA Conference

21 January

The meeting is positioned as a leading forum to share experiences, assess progress, highlight key challenges, and anticipate next steps. All of this will take place through roundtable discussions bringing together the main stakeholders involved in this regulation: regulators, CISOs, and representatives from the Public Administration. Specifically, participants will include representatives from the Ministry for Digital Transformation and the Civil Service, INCIBE, the Madrid Cybersecurity Agency, the Bank of Spain, Banco Santander, BBVA, CaixaBank, Mapfre, ING Bank, Allianz, Abanca, Sabadell Digital, Bankinter Group, Unicaja, Singular Bank, Santalucía, AXA Seguros, and Triodos Bank.

Link

IA Expo Internacional 2026

31 January

The 2026 International AI Expo will take place on January 31, 2026, at The Westin Santa Fe in Mexico City. The event will bring together leaders, entrepreneurs, developers, researchers, and executives to analyze real-world cases of AI adoption, its practical applications across different sectors, and key topics such as ethics, security, automation, digital transformation, and AI-driven innovation.

Link

Resources

➢ Guidelines for Media Sanitization

NIST, through its publication “Guidelines for Media Sanitization” (NIST Special Publication 800-88 Revision 1), establishes a clear and standardized technical framework for the secure sanitization of storage media, ensuring that sensitive information is effectively removed and cannot be recovered by unauthorized actors.

The document outlines methods for logical cleaning, purging, and physical destruction applicable to different types of devices (HDDs, SSDs, USB drives, mobile devices, tapes, etc.). It provides criteria for selecting the appropriate technique based on the sensitivity level of the data and the lifecycle stage of the media, and it defines organizational responsibilities to ensure secure and traceable management.

Link

➢ NIST Investments 2025

The document “NIS Investments 2025” from ENISA provides an in-depth analysis of how the Member States of the European Union and operators of essential services are investing in cybersecurity capabilities to meet the requirements of the NIS2 Directive and strengthen their resilience against growing threats. The report presents data and trends on investment priorities, the maturity of national capabilities, the evolution of risks, as well as the regulatory and operational challenges faced by the European ecosystem.

Link

Subscribe to RADAR

Powered by the
cybersecurity
NTT DATA team

es.nttdata.com