What's new in the OWASP Top Ten 2025

A one-day workshop highlighting changes and trends in the OWASP Top Ten

CYDOWASP251d | 1 day | Workshop | Demonstrated labs

This workshop provides a focused, practical exploration of the OWASP Top Ten 2025, explaining how the project is structured, how the Top Ten is created, and how it has evolved since 2021. You'll examine the methodology behind the list, understand why it is not a formal standard, and review key updates such as the expanded scope of Software Supply Chain Failures (A03) and the introduction of a new category, Mishandling of Exceptional Conditions (A10).

Through deep dives, case studies, and demonstrations, the course translates theory into practice. Topics include secure configuration and secrets management, protecting sensitive data in memory, supply chain security and SBOMs, vulnerability management and CVSS scoring, as well as robust error and exception handling. The final section looks ahead to emerging risks, and tackles inappropriate trust in AI-generated code ("vibe coding"), exploring how generative AI and agentic development pose new security challenges.

By the end of the course, learners will have a clear understanding of the most important changes in OWASP Top Ten 2025 and practical guidance for applying these insights in modern development and DevSecOps environments.

Note: This course presents a concise overview of key changes and emerging trends in the OWASP Top Ten 2025

It is recommended as an update to those development groups who already attended any of our web application security courses earlier.

Skills and drills

6 LABS

5 CASE STUDIES

Audience

Managers and developers working on web application development projects

Group size

Plenary, 30 participants

Preparedness

General development

Outline

- AppSec: The weakest link in cybersecurity
- The Open Worldwide Application Security Project
- Overview of some key updates and notable changes
- Then Next Steps - beyond the Top Ten
- Wrap up

Standards and references

OWASP, SEI CERT, CWE and Fortify Taxonomy

What you'll have learned

- Managing vulnerabilities in third party components
- Understanding Web application security issues
- Detailed analysis of the OWASP Top Ten elements
- Going beyond the low hanging fruits

Table of contents

- Vulnerability management
  - Patch management
  - Vulnerability management
  - Vulnerability databases
  - Vulnerability rating – CVSS
  - CVSS – Base Metric Group
  - CVSS – Supplemental Metric Group
  - CVSS – Threat Metric Group
  - CVSS – Environmental Metric Group
  - *Lab – Finding vulnerabilities in third-party components*
  - Bug bounty programs
  - DevOps, the CI / CD build process and Software Composition Analysis
  - Dependency checking
  - *Lab – Detecting vulnerable components*

A10 - Mishandling of Exceptional Conditions

- Error and exception handling principles
- Error handling
  - Returning a misleading status code
  - Reachable assertion
  - Information exposure through error reporting
    - Information leakage via error pages
    - *Case study – Information leakage via errors in Apache Superset*
- Exception handling
  - In the catch block. And now what?
  - Catching NullPointerException
  - Empty catch block
  - *Lab – Exception handling mess*
- Control flow
  - Incorrect block delimitation
  - Dead code
  - Using if-then-else and switch defensively

› Then Next Steps - beyond the Top Ten

X03 - Inappropriate Trust in AI-Generated Code ('Vibe Coding')

- The dark side of AI code generation
  - Automated programming: from punch cards to GenAI
  - What is responsible AI?
  - XAI: what's happening in the code-generating black box?

- Security, privacy and safety: how strong is that black box?
- The two sides of the coin
- What is GenAI (not) good for?
- Dependency hallucination via generative AI

Case study – A history of GitHub Copilot weaknesses (up to mid 2025)

Demonstration – GitHub Copilot code security

- Pitfalls of AI agent-driven vibe coding
  - "Vibe coding" and its implications
  - Security concerns of agentic development
  - MCP's effect on the attack surface
  - MCP-specific attack vectors

  Case study – Database leakage via Supabase MCP
  - Hallucinations and 'agentic death spirals'

› Wrap up

Secure coding principles

- Principles of robust programming by Matt Bishop
- Secure design principles of Saltzer and Schroeder

And now what?

- Software security sources and further reading
- Resources
- Generative AI – Resources and additional guidance