CS-057 · Application Security

OWASP Top 10

A community consensus list of the most critical web application security risks.

Executive Summary

The OWASP Top 10 is a widely referenced, community-driven awareness document that ranks the most critical security risks facing web applications. It is not a formal standard or a checklist of every possible flaw, but a prioritized starting point that helps teams focus effort where the greatest risk usually sits. Security teams, developers, and auditors use it as a common language for discussing and reducing application risk.

What It Is

The OWASP Top 10 is a periodically updated list published by the Open Worldwide Application Security Project, a nonprofit that produces free security resources. Each edition groups the most impactful classes of web application weakness into a small number of broad categories, ordered roughly by prevalence and severity. The categories are conceptual buckets rather than single bugs. For example, broken access control covers many ways authorization can fail, and injection covers many ways untrusted input can be interpreted as code or a command. The list is drawn from real-world data contributed by many organizations combined with a survey of practitioners, which keeps it grounded in what is actually being exploited.

Why It Matters

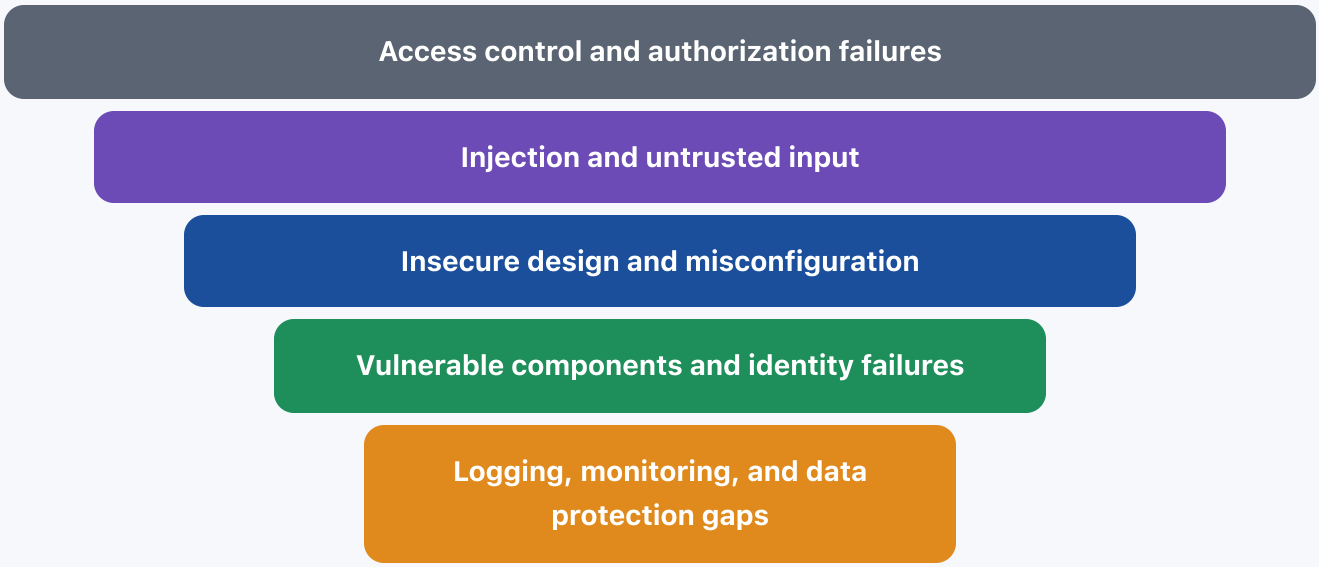
Web applications are one of the most exposed parts of any organization because they are reachable from the internet and often handle sensitive data and money. The OWASP Top 10 matters because it turns a huge, intimidating problem space into a focused set of priorities that a team can reasonably act on. It gives developers, testers, and leaders a shared vocabulary, so a finding can be discussed and tracked consistently. Many security requirements, contracts, and

compliance efforts reference it as a baseline expectation, which means fluency in the Top 10 is expected of anyone who builds, tests, or governs web software.

How It Works

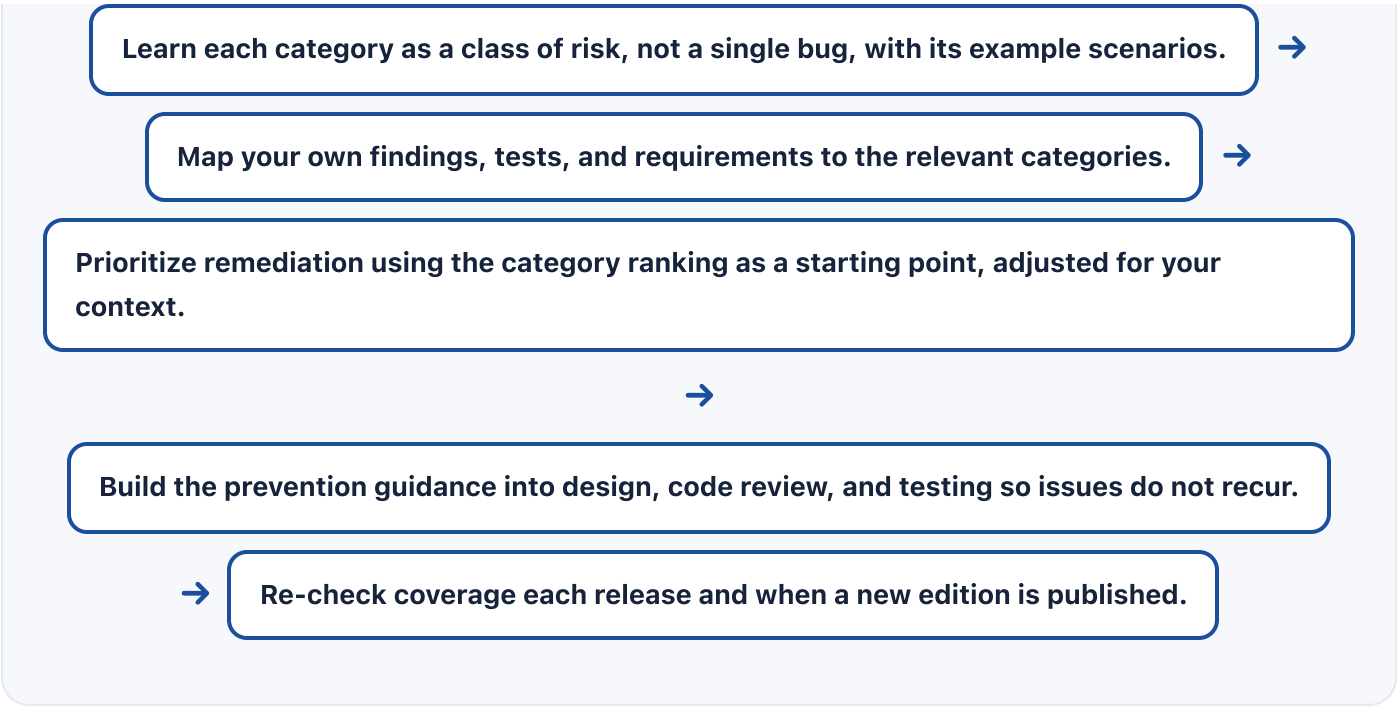
OWASP compiles data on how often each weakness class is found and how much harm it causes, then organizes the findings into ranked categories with descriptions, example scenarios, and prevention guidance. Teams use the list in several ways. Developers learn the categories so they can avoid the underlying mistakes as they build. Security testers map their findings to the categories to communicate results clearly. Risk and compliance teams use it as a baseline to measure coverage and gaps. Because it is updated over time, the exact category names, ordering, and groupings change between editions, so teams should always confirm they are working from the current edition rather than assuming a fixed order.

Architecture Diagram



The OWASP Top 10 groups many specific flaws into a small set of prioritized risk categories teams can act on.

Visual Workflow



Common Attacks

- Broken access control that lets a user reach data or actions meant for someone else
- Injection where untrusted input is interpreted as a query, command, or code
- Security misconfiguration such as default settings, verbose errors, or open features
- Use of components with known vulnerabilities that were never updated
- Identification and authentication failures that allow account takeover

Common Mistakes

- Treating the Top 10 as a complete checklist rather than an awareness starting point
- Working from an outdated edition with a different category order
- Fixing only the specific reported bug and not the underlying weakness class
- Assuming a scanner that maps to the Top 10 has found every relevant issue
- Focusing on the top categories while ignoring lower ranked but relevant risks

Best Practices

- Use the current OWASP Top 10 as a shared baseline across development and security

- Pair the Top 10 with a fuller standard such as the OWASP Application Security Verification Standard for depth
- Build category-level prevention into design, coding standards, and code review
- Combine automated testing with manual review, since not every category is easy to scan for
- Track findings by category over time to see whether whole classes of risk are shrinking
- Train developers on the underlying causes, not just the names of the categories

Quick Checklist

- [x] Team is working from the current published edition
- [x] Each category mapped to concrete prevention practices in the codebase
- [x] Access control and injection defenses verified, not assumed
- [x] Dependencies inventoried and checked for known vulnerabilities
- [x] Security logging and monitoring cover authentication and access events
- [x] Findings tracked by category and trended across releases

Recommended Tools

Static application security testing (SAST)
Analyzes source code for patterns tied to several Top 10 categories

Dynamic application security testing (DAST)
Tests a running application from the outside for exploitable flaws

Software composition analysis (SCA)
Finds known vulnerabilities in third-party and open-source components

Web application firewall (WAF)
Filters malicious traffic as a compensating control while fixes are made

Industry Standards

OWASP Top 10
The awareness document itself, a baseline for web application risk

OWASP Application Security Verification Standard (ASVS)

A deeper, testable set of application security requirements beyond the Top 10

NIST SP 800-53
Control families that map to many application security expectations

Career Relevance

The OWASP Top 10 is foundational for application security engineers, penetration testers, and secure code reviewers, who use it daily to frame findings and priorities. Developers are increasingly expected to know it so they can prevent issues at the source, and GRC and audit professionals reference it when assessing application risk. For the AI-Governance-Jobs.com audience, the Top 10 is a common entry point into appsec and a frequent interview topic.

Interview Questions

- What is the OWASP Top 10, and what is it not?
- Why is broken access control such a persistent, high-impact category?
- How would you explain injection as a class of risk rather than a single bug?
- How does the OWASP Top 10 relate to a fuller standard like ASVS?
- Why should a team confirm which edition of the Top 10 they are using?

Related Certifications

GIAC Web Application Penetration Tester (GWAPT)
Offensive Security Web Assessor (OSWA)
CompTIA Security+

Further Reading

- OWASP Top 10
- OWASP Application Security Verification Standard
- CWE, Common Weakness Enumeration

Key Takeaways

- The OWASP Top 10 ranks the most critical classes of web application risk.
- It is an awareness starting point, not a complete checklist or formal standard.
- Each item is a broad category of related weaknesses, not a single bug.
- It provides a shared vocabulary across developers, testers, and auditors.
- Categories and ordering change between editions, so always use the current one.

FAQ

Is the OWASP Top 10 a security standard?

Does covering the Top 10 mean an application is secure?

How often does the list change?

Related Careers

RELATED ROLES

- Application Security Engineer
- Penetration Tester
- Secure Code Reviewer
- Security Engineer

RELATED CERTIFICATIONS

- GIAC Web Application Penetration Tester (GWAPT)
- Offensive Security Web Assessor (OSWA)
- CompTIA Security+

CURRENT OPENINGS

Live roles are on the job board.

Browse all jobs

GUIDES & SALARY

- Career guides
- Role roadmaps
- Salary data
- Career tools

SUGGESTED LEARNING PATH

1. Ground the basics with CS-001 Cybersecurity
2. Study this sheet: **OWASP Top 10**
3. Go deeper: SQL Injection
4. Go deeper: Cross-Site Scripting (XSS)
5. Validate it: work toward **GIAC Web Application Penetration Tester (GWAPT)**
6. Find the role: browse current openings

RELATED SHEETS

CS-058 SQL Injection | CS-059 Cross-Site Scripting (XSS) | CS-063 Authorization | CS-065 Secure Coding | CS-001 Cybersecurity