wizer

Secure Code Training
Content List

Level 1

Topic	Title	Category	Languages
Inside the attacker's mind	Inside the Attacker's Mindset	Culture - Intro	Agnostic
XSS (Cross-Site Scripting)	What is XSS (Cross-Site Scripting)	OWASP10	Agnostic
	XSS Mitigation	OWASP10	Agnostic
IDOR (Insecure Direct Object Reference)	What is IDOR (Insecure Direct Object Reference)	OWASP10	JS, Java, Python, C#, Rust, Go, PHP, C++
	IDOR Mitigation	OWASP10	JS, Java, Python, C#, Rust, Go, PHP, C++
SQL Injection	What is SQL Injection	OWASP10	Agnostic
	SQL Injection Mitigation	OWASP10	JS, Java, Python, C#, Rust, Go, PHP, C++
SSRF (Server-Side Request Forgery)	What is SSRF (Server-Side Request Forgery)	OWASP10	JS, Java, Python, C#, Rust, Go, PHP, C++
	SSRF Mitigation	OWASP10	JS, Java, Python, C#, Rust, Go, PHP, C++
SSTI (Server-Side Template Injection)	What is SSTI (Server-Side Template Injection)	OWASP10	Agnostic
	SSTI Mitigation	OWASP10	JS, Java

Topic	Title	Category	Languages
Insecure Deserialization	What is Insecure Deserialization	OWASP10	Agnostic
	Insecure Deserialization Mitigation	OWASP10	Java
Prototype Pollution	What is Prototype Pollution	OWASP10	JavaScript
	Prototype Pollution Mitigation	OWASP10	JavaScript
Cryptographic Failure	What is Password Hashing?	OWASP10	Agnostic
	How to Securely Hash Password?	OWASP10	JS, Java, Python
CSRF (Cross-Site Request Forgery)	What is a CSRF (Cross Site Request Forgery)	OWASP10	Agnostic
	CSRF Mitigation	OWASP10	JS, Java, Python, C#, Rust, Go, PHP, C++
Authentication Forms	What is broken authentication	OWASP10	Agnostic
	How to avoid broken authentication	OWASP10	JS, Java, Python, C#, Rust, Go, PHP, C++
OS Command Injection & Code Injection	What are OS Command and Code Injections	OWASP10	Agnostic
	OS Command and Code Injections Mitigation	OWASP10	Agnostic
CSP	What is Content-Security-Policy	Best Practices	Agnostic
	How to use Content-Security-Policy	Best Practices	Agnostic
Race Conditions	What's the Impact of Race Conditions	Best Practices	Agnostic
	Mitigating Race Condition Risks	Best Practices	JS, Java, Python, C#, Rust, Go, PHP, C++
Security Logging Failures	Security Logging Failures	OWASP10	Agnostic

Topic	Title	Category	Languages
Security Misconfiguration (Level 1)	Security Misconfiguration: Headers	OWASP10	Agnostic
	Security Misconfiguration: Cookies	OWASP10	Agnostic
Vulnerable and Outdated Components	Vulnerable and Outdated Components	OWASP10	Agnostic

Level 2

Topic	Title	Category	Languages
JWT Tokens Risks	JWT Tokens' Common Risks	Best Practices	JS, Java, Python, C#
	Mitigating JWT Tokens Risks	Best Practices	JS, Java, Python, C#
File Upload Vulnerability	What are File Uploads Vulnerabilities	Best Practices	Agnostic
	Mitigating File Uploads Vulnerabilities	Best Practices	JS, Java, Python, C#, Rust
TerraForm Threats	Terraform Part 1	Dev & DevOps	Agnostic
	Terraform Part 2	Dev & DevOps	Agnostic
Using AI Tools to Boost Development	Using AI Tools to Boost Development	Best Practices	JS, Java, Python, C#, Rust, Go, PHP, C++
AI Common Risks - Prompt Injection	Prompt Injection Part 1	OWASP10	Agnostic
	Prompt Injection Part 2	OWASP10	Agnostic
Secure Software Development Lifecycle	Secure Software Development Lifecycle	OWASP10	Agnostic
Powershell Common Vulnerabilities	Powershell Common Vulnerabilities	Best Practices	Agnostic

Topic	Title	Category	Languages
Bash Common Vulnerabilities	Bash Common Vulnerabilities	Best Practices	Agnostic
CSS and other Injection Vulnerabilities	CSS Injection Vulnerabilities	OWASP10	Agnostic
	Other Injection Vulnerabilities	OWASP10	Agnostic
GitLab/GitHub Vulnerabilities	Git Vulnerabilities	Tools	Agnostic
	GitLab/GitHub Vulnerabilities	Tools	Agnostic
SAST/DAST & Depend-bots Practices	SAST Scope & Best Practices	Best Practices	Agnostic
	DAST Scope & Best Practices	Best Practices	Agnostic
GraphQL Vulnerabilities	GraphQL Vulnerabilities - Part 1	Tools	Agnostic
	GraphQL Vulnerabilities - Part 2	Tools	JS, Java, Python, C#, Rust, Go, PHP
Security Shift Left	Security Shift Left	Best Practices	Agnostic
OAuth Risks and Best Practices	OAuth Risks & Best Practices - Part 1	Best Practices	Agnostic
	OAuth Risks & Best Practices - Part 2	Best Practices	Agnostic
DOM Clobbering Vulnerability	DOM Clobbering Vulnerability	OWASP10	Agnostic
NoSQL Injection	NoSQL Injection - Part 1	OWASP10	Agnostic
	NoSQL Injection - Part 2	OWASP10	JS, Java, Python, C#, Rust, Go, PHP, C++

↓

Level 3 (Released in phases throughout 2025)

Topic	Title	Category
OWASP 10 2025 Update #1	Expected to be available by OWASP Org by 7/1	OWASP10
	Expected to be available by OWASP Org by 7/2	OWASP10
OWASP 10 2025 Update #2	Expected to be available by OWASP Org by 7/1	OWASP10
	Expected to be available by OWASP Org by 7/1	OWASP10
Open Redirect	Part 1	OWASP10
	Part 1	OWASP10
CORs	Part 1	OWASP10
	Part 2	OWASP10

Low Level C/C++ Level 1

Topic	Description
Reversing	Introduction to binaries, assembly, decompiling, Obfuscation
	Reverse engineering a simple binary using Ghidra
Buffer Overflows	Showcasing a simple buffer overflow
	How to mitigate against buffer overflows
Integer Overflows	Showcasing an integer overflow, real life examples
	How to mitigate against integer overflows
Format Strings	Showcasing a simple format string vulnerability
	How to mitigate against format string vulnerabilities

Topic	Description
Use After Free / Double Free	Showcase a simple use after free vulnerability
	How to mitigate against use after free vulnerabilities
Uninitialized memory	Showcase a simple uninitialized memory vulnerability
	How to mitigate against uninitialized memory vulnerabilities

OWASP 10 Crash Course (< 30 total minutes playtime)

Topic	Description
A01	Broken Access Control: Unauthorized users can access sensitive data or functionality.
A02	Cryptographic Failures: Weak or misconfigured cryptographic algorithms can be exploited.
A03	Injection: Malicious code or data can be injected into applications.
A04	Insecure Design: Fundamental design flaws can lead to various vulnerabilities.
A05	Security Misconfiguration: Misconfigured security settings can create open doors for attackers.
A06	Vulnerable and Outdated Components: Using outdated or vulnerable libraries and components can introduce known security flaws.
A07	Identification and Authentication Failures: Inadequate identification and authentication mechanisms can allow unauthorized access.
A08	Software and Data Integrity Failures: Compromised software or data can lead to various attacks.
A09	Security Logging and Monitoring Failures: Insufficient or inadequate logging and monitoring can hinder incident detection and response.
A10	Server-Side Request Forgery: Attackers can manipulate a server to make requests on their behalf, potentially accessing sensitive data or resources.

OWASP 10 Deep-Dive For QA

Built for QA, PMs, DevOps, and other non-developer tech roles

- SQL Injection
- IDOR
- SSRF
- OS Command and Code Injection
- Race Conditions
- Cryptographic Failure: Hashing
- CSRF
- CSP
- SSTI
- Broken Authentication
- XSS
- Logging and Monitoring
- Security Misconfigurations: Headers
- Security Misconfigurations: Cookies
- Vulnerable and Outdated Components