Application security, in concrete terms.

Application security is the practice of preventing the failures attackers use to take over web applications, steal data, or pivot to internal systems. Most modern web breaches still come from a small set of well-understood flaw classes: data input that the app trusts too much, server-side logic that fetches the wrong thing, authorization checks in the wrong place, and token systems that were configured permissively.

HOW IT WORKS

01 Topics

- What is SQL Injection?: the oldest and still one of the most damaging web vulnerabilities. How it works, why parameterized queries fix it, when modern frameworks still leave it open.
- What is Cross-Site Scripting (XSS)?: when attacker-controlled content executes as code in another user's browser. Stored, reflected, and DOM-based variants explained.
- What is Server-Side Request Forgery (SSRF)?-: when an application fetches a URL the attacker chose, often reaching internal services that were never meant to be public.
- What is Insecure Direct Object Reference (IDOR)?: when a user can access another user's data by changing an ID in the URL or request body. The single most common authorization flaw in production.
- JWT Security: Common Attacks and Defenses-: the standard token format for modern APIs, and the configuration mistakes that turn it into a backdoor.
- What is Local File Inclusion (LFI)?: when an application includes a file the user names, exposing source and secrets and often reaching code execution.
- What is Command Injection?: running operating-system commands on the server by smuggling them into input passed to a shell.
- What is XXE?: abusing XML external entities to read server files, reach internal systems, and exfiltrate data.

SOURCES

[1] OWASP Top 10 (2021)

[2] OWASP Application Security Verification Standard

[3] MITRE ATT&CK for Enterprise

- What is SSTI?: when user input is rendered as template code, frequently reaching remote code execution.
- What is Insecure Deserialization?: rebuilding objects from untrusted data, where loading the data runs the attacker's code.
- What are File Upload Vulnerabilities?: from a weak upload control to a web shell and full server compromise.
- What is Path Traversal?: using ../ sequences to escape the intended directory and read files anywhere the app can reach.
- What is CSRF?: tricking a logged-in user's browser into sending a state-changing request without their consent.
- What is an Open Redirect?: abusing a trusted site's own link to send users to a malicious destination.
- What is an Authentication Bypass?: gaining authenticated access without valid credentials via logic flaws or token tampering.
- What is a Race Condition?: exploiting the timing window between a check and an action with parallel requests.
- What is HTTP Request Smuggling?: desyncing a front-end and back-end so a smuggled request poisons the next user's.
- What is Prototype Pollution?: a JavaScript flaw injecting properties into the object every object inherits from.

Scope an application penetration test.
securelayer7.net/learn/application-security

Open online