OWASP Top 10 (2021) with Spring Security Tips

What is OWASP?
OWASP (Open Worldwide Application Security Project) is a non-profit organization focused on improving the security of software. It provides free tools, documentation, and standards to help developers, testers, and organizations build secure applications. One of its most well-known projects is the **OWASP Top 10**, a regularly updated report that highlights the most critical web application security risks. Knowing these risks helps developers prevent common vulnerabilities early in the development process.

Broken Access Control (A01:2021)

When users can access data or perform actions they shouldn’t be allowed to.
Example: A user can change another user’s profile by modifying the URL (`/user/123` → `/user/456`).
Spring Security Tip: Use method-level security (@PreAuthorize, @PostAuthorize), validate user roles, avoid client-side checks.

Cryptographic Failures (A02:2021)

Weak or no encryption, exposing sensitive data.
Example: Storing passwords in plain text.
Spring Security Tip: Use BCryptPasswordEncoder for hashing, enable HTTPS, avoid hardcoding secrets.

Injection (A03:2021)

Untrusted data is interpreted as code or command.
Example: SQL Injection (' OR 1=1 --).
Spring Security Tip: Use parameterized queries (JPA/Spring Data), avoid concatenating raw input.

Insecure Design (A04:2021)

Flaws in application architecture that make it insecure.
Example: No rate limiting on login attempts.
Spring Security Tip: Apply defense-in-depth, secure at every layer.

Security Misconfiguration (A05:2021)

Default, insecure, or misconfigured settings.
Example: Default admin credentials, unsecured actuator endpoints.
Spring Security Tip: Disable unused endpoints, secure Actuator with auth, use CSP headers.

Vulnerable and Outdated Components (A06:2021)

Using outdated libraries/frameworks with known vulnerabilities.
Example: Old version of Spring with RCE vulnerabilities.
Spring Security Tip: Keep dependencies updated, use OWASP Dependency-Check.

Identification and Authentication Failures (A07:2021)

Broken login, session management, or password handling.
Example: Predictable session IDs, missing logout.
Spring Security Tip: Use Spring Security’s session management, enforce password policy, implement logout.

Software and Data Integrity Failures (A08:2021)

Code or data integrity is not verified.
Example: Using plugins from untrusted sources.
Spring Security Tip: Use signed dependencies, verify file checksums, use trusted CI/CD pipelines.

Security Logging and Monitoring Failures (A09:2021)

Insufficient logging/monitoring allows attacks to go undetected.
Example: No logs for failed logins.
Spring Security Tip: Enable Actuator + centralized logging, log suspicious activities.

Server-Side Request Forgery (SSRF) (A10:2021)

Server is tricked into making requests to internal services.
Example: User submits a URL that fetches localhost admin endpoint.
Spring Security Tip: Whitelist outbound requests, validate URLs, disable unnecessary network access.