

20WASP

September 9, 2025

[]: 2021

open
web
application
security project

1 1 Broken Access Control

bypass authorization safeguards
become privileged user

4. Reducing Development Costs

- **Proactive vs. Reactive:**
 - Addressing vulnerabilities during development is far cheaper than fixing them after a breach or incident.
 - Frameworks like OWASP guide teams to "build security in" rather than retrofitting it later.

principle of least privilege

2 2 Cryptographic failure

failure of secure data transfer

https
use libraries (bcrypt?)

3 3 Injection

introducing hostile code to program
changing the program execution

example
code to steal authentication cookies

XSS

vulnerable to injection when user provided data is not sanitized/ validated by the application

4 4 Insecure design

use security design and patterns by default
for new application

5 5 Security Misconfiguration

bad configuration present in 90%
lack of security controls

4. Reducing Development Costs

- **Proactive vs. Reactive:**
 - Addressing vulnerabilities during development is far cheaper than fixing them after a breach or incident.
 - Frameworks like OWASP guide teams to "build security in" rather than retrofitting it later.

6 6 Vulnerable and outdated components

os
servers
DBMS
APIS
Runtime Environments
libraries

7 6 Broken Authentication

It has come down (standardise frameworks are helping)
doubt () (oauth wagers ??) user MFA

4. Reducing Development Costs

- **Proactive vs. Reactive:**
 - Addressing vulnerabilities during development is far cheaper than fixing them after a breach or incident.
 - Frameworks like OWASP guide teams to "build security in" rather than retrofitting it later.

8 7 Software and Data integrity failure

using untrusted app plugins (wordpress using compromised plugin)

9 8 Security login and Monitoring failures

serious impact on digital forensics

10 9 SERVER SIDE REQUEST FOREGERY

Why OWASP and other practical frameworks are important.
because they provide standardized, actionable best practices

Promoting Best Practices

Standardization

4. Reducing Development Costs

- **Proactive vs. Reactive:**
 - Addressing vulnerabilities during development is far cheaper than fixing them after a breach or incident.
 - Frameworks like OWASP guide teams to "build security in" rather than retrofitting it later.

Compliance with Industry Standards

Fostering Developer Education