OWASP - Open Web Application Security Project

.NET 5

OWASP – Overview

https://owasp.org

The Open Web Application Security Project (OWASP) is a nonprofit foundation that was formed on December 1, 2001. It works to improve the security of software.

Through community-led, open-source software projects with hundreds of local chapters worldwide and tens of thousands of members. The OWASP Foundation uses leading educational and training conferences to become the primary source for developers and technologists to secure the web.

OWASP Core Values are:

- Open: Everything at OWASP is radically transparent from their finances to their code.
- Innovative: OWASP encourages and supports innovation and experiments for solutions to software security challenges.
- Global: Anyone around the world is encouraged to participate in the OWASP community.
- Integrity: The OWASP community is respectful, supportive, truthful, and vendor neutral.

OWASP Risk Rating Methodology

https://owasp.org/www-community/OWASP_Risk_Rating_Methodology

Being able to estimate the associated risk to a business is important. Early in the life cycle, one may identify security concerns in an architecture by using ***threat modeling***. Security issues can be found using code review or penetration testing, while other problems may not be discovered until the application is in production and/or is actively compromised.

Likelihood and Impact Levels ranges	
0 to <3	Low
3 to <6	Medium
6 to 9	High

Having a defined system in place for rating risks makes it possible to estimate the severity of risks and make an informed decision about what to do about those risks.

A Risk Assessment System helps to ensure that a business doesn't get distracted by minor risks while ignoring more serious, less well understood, risks.

A vulnerability that is critical to one organization may not be very important to another, so OWASP uses a basic framework that should be customized for other organizations.

Overall Risk Severity (Likelihood+Impact)				
Impact	High	Medium	High	Critical
	Medium	Low	Medium	High
	Low	Note	Low	Medium
		Low	Medium	High
	Likelihood			

OWASP Risk Rating Steps

https://owasp.org/www-community/OWASP_Risk_Rating_Methodology

The standard risk model defines the severity of a risk as:

- The ***likelihood*** of the event happening multiplied by the potential harmful ***impact*** of the event.

These are the 6 steps OWASP uses to determine risk severity. Each step is assigned a number that goes into the overall risk assessment.

Always assume the worst-case scenario.

1. **Identify the Risk:** Gather information about the threat agent, attack type, vulnerability, and potential impact.
2. **Estimate Attack Likelihood:** Estimate the Threat Agents' 1) skill, 2) motive, 3)opportunity and 4) size. Assess the likelihood of the vulnerability being exploited by it's 1) ease of discovery and 2) ease of exploit, 3) how well it's known to the attackers, and 4) the victim organizations intrusion detection capabilities.

Likelihood and Impact Levels ranges	
0 to <3	Low
3 to <6	Medium
6 to 9	High

Overall Risk Severity (Likelihood+Impact)				
Impact	High	Medium	High	Critical
	Medium	Low	Medium	High
	Low	Note	Low	Medium
		Low	Medium	High
	Likelihood			

OWASP Risk Rating Steps

https://owasp.org/www-community/OWASP_Risk_Rating_Methodology

3. **Estimate Technical and Business impact:** *Technical impact* deals with confidentiality, integrity, availability, and accountability. The business risk is what justifies monetary investment in fixing security vulnerabilities. It stems from the Technical Impact but encompasses what is important to each business individually. Assess each of the following four factors, Financial Damage, Damage to business reputation, Non-compliance, and Privacy violation.

Likelihood and Impact Levels ranges	
0 to <3	Low
3 to <6	Medium
6 to 9	High

4. **Determine the Severity of the Risk:** The *Likelihood Estimate* and the *Impact Estimate* are combined to calculate the overall severity for the risk.

Overall Risk Severity (Likelihood+Impact)				
Impact	High	Medium	High	Critical
	Medium	Low	Medium	High
	Low	Note	Low	Medium
		Low	Medium	High
	Likelihood			

OWASP Risk Rating Steps

https://owasp.org/www-community/OWASP_Risk_Rating_Methodology

5. **Decide What to Fix:** Create a prioritized list of what to fix first. Often, risks on this list will never be completed. Some risks are minimal enough that the cost to fix them isn't justified.

6. **Customize the Risk Rating Model:** A customized Risk Assessment Model is much more likely to produce results that match evaluators perceptions about which of multiple risks is serious and save time evaluating those risks. There are several ways to tailor this model for an organization like 1) adding factors specific to an organization, 2) customizing options like changing names to match the business' team names or changing number equivalents, or 3) weighting factors to emphasize which ones are more important to a specific business.

Likelihood and Impact Levels ranges	
0 to <3	Low
3 to <6	Medium
6 to 9	High

Overall Risk Severity (Likelihood+Impact)				
Impact	High	Medium	High	Critical
	Medium	Low	Medium	High
	Low	Note	Low	Medium
		Low	Medium	High
	Likelihood			

OWASP – Top 10 Web Security Risks

https://owasp.org/www-project-top-ten/

1. Injection – Injection flaws (SQL, NoSQL, OS, and LDAP injection) occur when untrusted data is sent to an interpreter as part of a command or query. The hostile data can trick the interpreter into executing harmful commands or accessing data without proper authorization.
2. Broken Authentication – Authentication and session management are often implemented incorrectly. This allows passwords, keys, or session tokens to be compromised so attackers can assume another users' identity.
3. Sensitive Data Exposure – When data is improperly protected. Attackers may steal or modify the data to conduct credit card fraud, identity theft, or other crimes. Sensitive data may be compromised without extra protection, such as encryption (at rest or in transit). This data requires special precautions when exchanged with the browser.
4. XML External Entities (XXE) – Many older or poorly configured XML processors evaluate external entity references within XML documents. External entities can be used to disclose internal files using the file URI handler, internal file shares, internal port scanning, remote code execution, and denial of service attacks.

OWASP – Top 10 Web Security Risks

https://owasp.org/www-project-top-ten/

5. Broken Access Control – When restrictions on what authenticated users may do are not properly enforced, attackers can access unauthorized functionality and data like access other users' accounts, view sensitive files, modify other users' data, change access rights, etc.
6. Security Misconfiguration – This is the most commonly seen issue and is often the result of insecure default configurations, incomplete or ad hoc configurations, open cloud storage, misconfigured HTTP headers, or verbose error messages containing sensitive information.
7. Cross-Site Scripting (XSS) – XSS flaws occur whenever an application 1) includes untrusted data in a new web page without proper validation or escaping or 2) updates an existing web page with user-supplied data using a browser API that can create HTML or JavaScript. XSS allows attackers to execute scripts in the victim's browser which can hijack user sessions, deface web sites, or redirect the user to malicious sites.

OWASP – Top 10 Web Security Risks

https://owasp.org/www-project-top-ten/

8. Insecure Deserialization – Insecure deserialization can allow remote code execution. They can also be used to perform attacks, including replay attacks, injection attacks, and privilege escalation attacks.
9. Using Components with Known Vulnerabilities – Components such as libraries, frameworks, and other software modules, run with the same privileges as the application. If a vulnerable component is exploited, such an attack can facilitate serious data loss or server takeover. Applications and APIs using components with known vulnerabilities may undermine application defenses and enable various attacks.
10. Insufficient Logging & Monitoring – Most breach studies show time to detect a breach is over 200 days, typically detected by external parties rather than internal processes or monitoring. Insufficient logging and monitoring, coupled with missing or ineffective integration with incident response, allows attackers to continue attacking systems, maintain persistence, pivot to more systems, and tamper, extract, or destroy data.