**OWASP Top 10**

The **OWASP Top 10** is a standard awareness document for developers and web application security. It represents a broad consensus about the most critical security risks to web applications. The following list of the OWASP top 10:

A01:2021-**Broken Access Control** moves up from the fifth position; 94% of applications were tested for some form of broken access control. The 34 Common Weakness Enumerations (CWEs) mapped to Broken Access Control had more occurrences in applications than any other category.

A02:2021-**Cryptographic Failures** shifts up one position to #2, previously known as Sensitive Data Exposure, which was broad symptom rather than a root cause. The renewed focus here is on failures related to cryptography which often leads to sensitive data exposure or system compromise.

A03:2021-**Injection**slides down to the third position. 94% of the applications were tested for some form of injection, and the 33 CWEs mapped into this category have the second most occurrences in applications. Cross-site Scripting is now part of this category in this edition.

A04:2021-**Insecure Design** is a new category for 2021, with a focus on risks related to design flaws. If we genuinely want to “move left” as an industry, it calls for more use of threat modelling, secure design patterns and principles, and reference architectures.

A05:2021-**Security Misconfiguration** moves up from #6 in the previous edition; 90% of applications were tested for some form of misconfiguration. With more shifts into highly configurable software, it’s not surprising to see this category move up. The former category for XML External Entities (XXE) is now part of this category.

A06:2021-**Vulnerable and Outdated Components** was previously titled Using Components with Known Vulnerabilities and is #2 in the Top 10 community survey, but also had enough data to make the Top 10 via data analysis. This category moves up from #9 in 2017 and is a known issue that we struggle to test and assess risk. It is the only category not to have any Common Vulnerability and Exposures (CVEs) mapped to the included CWEs, so a default exploit and impact weights of 5.0 are factored into their scores.

A07:2021-**Identification and Authentication Failures** was previously Broken Authentication and is sliding down from the second position, and now includes CWEs that are more related to identification failures. This category is still an integral part of the Top 10, but the increased availability of standardized frameworks seems to be helping.

A08:2021-**Software and Data Integrity Failures** is a new category for 2021, focusing on making assumptions related to software updates, critical data, and CI/CD pipelines without verifying integrity. One of the highest weighted impacts from Common Vulnerability and Exposures/Common Vulnerability Scoring System (CVE/CVSS) data mapped to the 10 CWEs in this category. Insecure Deserialization from 2017 is now a part of this larger category.

A09:2021-**Security Logging and Monitoring Failures** was previously Insufficient Logging & Monitoring and is added from the industry survey (#3), moving up from #10 previously. This category is expanded to include more types of failures, is challenging to test for, and isn’t well represented in the CVE/CVSS data. However, failures in this category can directly impact visibility, incident alerting, and forensics.

A10:2021-**Server-Side Request Forgery** is added from the Top 10 community survey (#1). The data shows a relatively low incidence rate with above average testing coverage, along with above-average ratings for Exploit and Impact potential. This category represents the scenario where the security community members are telling us this is important, even though it’s not illustrated in the data at this time.

**The following are the CWE for the top 5 vulnerabilities:**

**Broken Access Control: CWE-285**

List of Mapped CWEs

CWE-22 Improper Limitation of a Pathname to a Restricted Directory ('Path Traversal')

CWE-23 Relative Path Traversal

CWE-35 Path Traversal: '.../...//'

CWE-59 Improper Link Resolution Before File Access ('Link Following')

CWE-200 Exposure of Sensitive Information to an Unauthorized Actor

CWE-201 Exposure of Sensitive Information Through Sent Data

CWE-219 Storage of File with Sensitive Data Under Web Root

CWE-264 Permissions, Privileges, and Access Controls (should no longer be used)

CWE-275 Permission Issues

CWE-276 Incorrect Default Permissions

CWE-284 Improper Access Control

CWE-285 Improper Authorization

CWE-352 Cross-Site Request Forgery (CSRF)

CWE-359 Exposure of Private Personal Information to an Unauthorized Actor

CWE-377 Insecure Temporary File

CWE-402 Transmission of Private Resources into a New Sphere ('Resource Leak')

CWE-425 Direct Request ('Forced Browsing')

CWE-441 Unintended Proxy or Intermediary ('Confused Deputy')

CWE-497 Exposure of Sensitive System Information to an Unauthorized Control Sphere

CWE-538 Insertion of Sensitive Information into Externally-Accessible File or Directory

CWE-540 Inclusion of Sensitive Information in Source Code

CWE-548 Exposure of Information Through Directory Listing

CWE-552 Files or Directories Accessible to External Parties

CWE-566 Authorization Bypass Through User-Controlled SQL Primary Key

CWE-601 URL Redirection to Untrusted Site ('Open Redirect')

CWE-639 Authorization Bypass Through User-Controlled Key

CWE-651 Exposure of WSDL File Containing Sensitive Information

CWE-668 Exposure of Resource to Wrong Sphere

CWE-706 Use of Incorrectly-Resolved Name or Reference

CWE-862 Missing Authorization

CWE-863 Incorrect Authorization

CWE-913 Improper Control of Dynamically-Managed Code Resources

CWE-922 Insecure Storage of Sensitive Information

CWE-1275 Sensitive Cookie with Improper SameSite Attribute

**Cryptographic Failures: CWE 916**

List of Mapped CWEs

CWE-261 Weak Encoding for Password

CWE-296 Improper Following of a Certificate's Chain of Trust

CWE-310 Cryptographic Issues

CWE-319 Cleartext Transmission of Sensitive Information

CWE-321 Use of Hard-coded Cryptographic Key

CWE-322 Key Exchange without Entity Authentication

CWE-323 Reusing a Nonce, Key Pair in Encryption

CWE-324 Use of a Key Past its Expiration Date

CWE-325 Missing Required Cryptographic Step

CWE-326 Inadequate Encryption Strength

CWE-327 Use of a Broken or Risky Cryptographic Algorithm

CWE-328 Reversible One-Way Hash

CWE-329 Not Using a Random IV with CBC Mode

CWE-330 Use of Insufficiently Random Values

CWE-331 Insufficient Entropy

CWE-335 Incorrect Usage of Seeds in Pseudo-Random Number Generator(PRNG)

CWE-336 Same Seed in Pseudo-Random Number Generator (PRNG)

CWE-337 Predictable Seed in Pseudo-Random Number Generator (PRNG)

CWE-338 Use of Cryptographically Weak Pseudo-Random Number Generator(PRNG)

CWE-340 Generation of Predictable Numbers or Identifiers

CWE-347 Improper Verification of Cryptographic Signature

CWE-523 Unprotected Transport of Credentials

CWE-720 OWASP Top Ten 2007 Category A9 - Insecure Communications

CWE-757 Selection of Less-Secure Algorithm During Negotiation('Algorithm Downgrade')

CWE-759 Use of a One-Way Hash without a Salt

CWE-760 Use of a One-Way Hash with a Predictable Salt

CWE-780 Use of RSA Algorithm without OAEP

CWE-818 Insufficient Transport Layer Protection

CWE-916 Use of Password Hash With Insufficient Computational Effort

**Injections: CWE 564**

List of Mapped CWEs

CWE-20 Improper Input Validation

CWE-74 Improper Neutralization of Special Elements in Output Used by a Downstream Component ('Injection')

CWE-75 Failure to Sanitize Special Elements into a Different Plane (Special Element Injection)

CWE-77 Improper Neutralization of Special Elements used in a Command ('Command Injection')

CWE-78 Improper Neutralization of Special Elements used in an OS Command ('OS Command Injection')

CWE-79 Improper Neutralization of Input During Web Page Generation ('Cross-site Scripting')

CWE-80 Improper Neutralization of Script-Related HTML Tags in a Web Page (Basic XSS)

CWE-83 Improper Neutralization of Script in Attributes in a Web Page

CWE-87 Improper Neutralization of Alternate XSS Syntax

CWE-88 Improper Neutralization of Argument Delimiters in a Command ('Argument Injection')

CWE-89 Improper Neutralization of Special Elements used in an SQL Command ('SQL Injection')

CWE-90 Improper Neutralization of Special Elements used in an LDAP Query ('LDAP Injection')

CWE-91 XML Injection (aka Blind XPath Injection)

CWE-93 Improper Neutralization of CRLF Sequences ('CRLF Injection')

CWE-94 Improper Control of Generation of Code ('Code Injection')

CWE-95 Improper Neutralization of Directives in Dynamically Evaluated Code ('Eval Injection')

CWE-96 Improper Neutralization of Directives in Statically Saved Code ('Static Code Injection')

CWE-97 Improper Neutralization of Server-Side Includes (SSI) Within a Web Page

CWE-98 Improper Control of Filename for Include/Require Statement in PHP Program ('PHP Remote File Inclusion')

CWE-99 Improper Control of Resource Identifiers ('Resource Injection')

CWE-100 Deprecated: Was catch-all for input validation issues

CWE-113 Improper Neutralization of CRLF Sequences in HTTP Headers ('HTTP Response Splitting')

CWE-116 Improper Encoding or Escaping of Output

CWE-138 Improper Neutralization of Special Elements

CWE-184 Incomplete List of Disallowed Inputs

CWE-470 Use of Externally-Controlled Input to Select Classes or Code ('Unsafe Reflection')

CWE-471 Modification of Assumed-Immutable Data (MAID)

CWE-564 SQL Injection: Hibernate

CWE-610 Externally Controlled Reference to a Resource in Another Sphere

CWE-643 Improper Neutralization of Data within XPath Expressions ('XPath Injection')

CWE-644 Improper Neutralization of HTTP Headers for Scripting Syntax

CWE-652 Improper Neutralization of Data within XQuery Expressions ('XQuery Injection')

CWE-917 Improper Neutralization of Special Elements used in an Expression Language Statement ('Expression Language Injection')

**Insecure Design: CWE 653**

List of Mapped CWEs

CWE-73 External Control of File Name or Path

CWE-183 Permissive List of Allowed Inputs

CWE-209 Generation of Error Message Containing Sensitive Information

CWE-213 Exposure of Sensitive Information Due to Incompatible Policies

CWE-235 Improper Handling of Extra Parameters

CWE-256 Unprotected Storage of Credentials

CWE-257 Storing Passwords in a Recoverable Format

CWE-266 Incorrect Privilege Assignment

CWE-269 Improper Privilege Management

CWE-280 Improper Handling of Insufficient Permissions or Privileges

CWE-311 Missing Encryption of Sensitive Data

CWE-312 Cleartext Storage of Sensitive Information

CWE-313 Cleartext Storage in a File or on Disk

CWE-316 Cleartext Storage of Sensitive Information in Memory

CWE-419 Unprotected Primary Channel

CWE-430 Deployment of Wrong Handler

CWE-434 Unrestricted Upload of File with Dangerous Type

CWE-444 Inconsistent Interpretation of HTTP Requests ('HTTP Request Smuggling')

CWE-451 User Interface (UI) Misrepresentation of Critical Information

CWE-472 External Control of Assumed-Immutable Web Parameter

CWE-501 Trust Boundary Violation

CWE-522 Insufficiently Protected Credentials

CWE-525 Use of Web Browser Cache Containing Sensitive Information

CWE-539 Use of Persistent Cookies Containing Sensitive Information

CWE-579 J2EE Bad Practices: Non-serializable Object Stored in Session

CWE-598 Use of GET Request Method With Sensitive Query Strings

CWE-602 Client-Side Enforcement of Server-Side Security

CWE-642 External Control of Critical State Data

CWE-646 Reliance on File Name or Extension of Externally-Supplied File

CWE-650 Trusting HTTP Permission Methods on the Server Side

CWE-653 Insufficient Compartmentalization

CWE-656 Reliance on Security Through Obscurity

CWE-657 Violation of Secure Design Principles

CWE-799 Improper Control of Interaction Frequency

CWE-807 Reliance on Untrusted Inputs in a Security Decision

CWE-840 Business Logic Errors

CWE-841 Improper Enforcement of Behavioral Workflow

CWE-927 Use of Implicit Intent for Sensitive Communication

CWE-1021 Improper Restriction of Rendered UI Layers or Frames

CWE-1173 Improper Use of Validation Framework

**Security Misconfiguration: CWE 614**

List of Mapped CWEs

CWE-2 7PK - Environment

CWE-11 ASP.NET Misconfiguration: Creating Debug Binary

CWE-13 ASP.NET Misconfiguration: Password in Configuration File

CWE-15 External Control of System or Configuration Setting

CWE-16 Configuration

CWE-260 Password in Configuration File

CWE-315 Cleartext Storage of Sensitive Information in a Cookie

CWE-520 .NET Misconfiguration: Use of Impersonation

CWE-526 Exposure of Sensitive Information Through Environmental Variables

CWE-537 Java Runtime Error Message Containing Sensitive Information

CWE-541 Inclusion of Sensitive Information in an Include File

CWE-547 Use of Hard-coded, Security-relevant Constants

CWE-611 Improper Restriction of XML External Entity Reference

CWE-614 Sensitive Cookie in HTTPS Session Without 'Secure' Attribute

CWE-756 Missing Custom Error Page

CWE-776 Improper Restriction of Recursive Entity References in DTDs ('XML Entity Expansion')

CWE-942 Permissive Cross-domain Policy with Untrusted Domains

CWE-1004 Sensitive Cookie Without 'HttpOnly' Flag

CWE-1032 OWASP Top Ten 2017 Category A6 - Security Misconfiguration

CWE-1174 ASP.NET Misconfiguration: Improper Model Validation

**Here are some vulnerabilities with their description and business impact:**

**Broken Access Control**

Description: Broken Access Control refers to the inadequate implementation or absence of proper access controls that should restrict users' actions and permissions. This vulnerability allows unauthorized users to access resources, data, or functionality that they shouldn't be able to access, leading to potential security breaches and data compromise.

Business Impact: Broken Access Control can have severe consequences, including unauthorized access to sensitive data, manipulation of critical settings, and exposure of confidential information. This can result in data breaches, loss of intellectual property, regulatory violations, and reputation damage for the affected organization.

**Cryptographic Failures**

Description: Cryptographic Failures encompass vulnerabilities related to the incorrect implementation or usage of cryptographic functions and protocols. When cryptographic mechanisms are flawed, attackers can potentially gain access to encrypted data or manipulate encrypted communications.

Business Impact: Cryptographic Failures can lead to the exposure of sensitive information, such as passwords, credit card numbers, or personal data. It can also result in unauthorized access, data manipulation, and system compromise. The impact can include breaches of privacy, regulatory non-compliance, financial losses, and erosion of customer trust.

**Injection**

Description: Injection vulnerabilities occur when untrusted data is inserted into an application and executed as code. This can lead to malicious code execution, data leakage, and system compromise. The injection category includes various types, such as SQL injection and cross-site scripting (XSS).

Business Impact: Injection attacks can lead to data breaches, unauthorized access to systems, and manipulation of user data. Exploiting injection vulnerabilities can allow attackers to steal sensitive information, perform actions on behalf of legitimate users, and compromise the integrity of applications, leading to financial losses and reputational damage.

**Insecure Design**

Description: Insecure Design vulnerabilities stem from flaws in the initial design and architecture of software applications. Poorly designed systems may lack proper security controls, leading to vulnerabilities that attackers can exploit.

Business Impact: Insecure Design can result in persistent vulnerabilities that are costly to fix and can potentially expose systems to various attacks. Attackers can exploit design flaws to gain unauthorized access, perform data exfiltration, and compromise the entire application. This can lead to data breaches, financial losses, and reputational harm.

**Security Misconfiguration**

Description: Security Misconfiguration occurs when applications, servers, or components are not securely configured. This includes default settings, unnecessary features, and improper access controls that can be exploited by attackers.

Business Impact: Security Misconfigurations can lead to unauthorized access, data exposure, and system compromise. Attackers can exploit misconfigured settings to gain access to sensitive data, escalate privileges, and potentially take control of systems. This can result in data breaches, regulatory violations, and damage to the organization's reputation.