OPEN WEB APPLICATION SECURITY PROJECT (OWASP)

TOP 10

Contenidos

1. Introducción.
2. OWASP Top 10 - 2021.
    1. A1:2021 – Broken Access Control.
    2. A2:2021 – Cryptographic Failures.
    3. A3:2021 – Injection.
    4. A4:2021 – Insecure Design.
    5. A5:2021 – Security Misconfiguration.
    6. A6:2021 – Vulnerable and Outdated Components.
    7. A7:2021 – Identification and Authentication Failures.
    8. A8:2021 – Software and Data Integrity Failures.
    9. A9:2021 – Security Logging and Monitoring.
    10. A10:2021 – Server Side Request Forgery.

Introducción

OWASP

- **OWASP** (**O**pen **W**eb **A**pplication **S**ecurity **P**roject)
- **Fundación OWASP** es un organismo sin ánimo de lucro
  - Formada por empresas, organizaciones educativas y particulares de todo el mundo
  - Su objetivo es determinar y combatir las **causas que hacen que el software sea inseguro**.
  - Apoya y gestiona proyectos e infraestructura de OWAS
  - Crear artículos, metodologías, documentación, herramientas y tecnologías que se liberan y pueden ser usadas gratuitamente

Introducción

Proyectos de OWASP

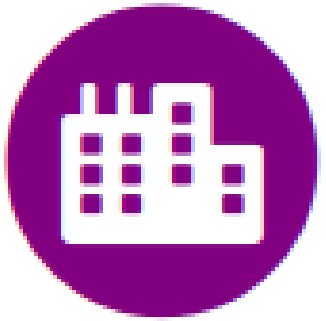
- https://owasp.org/projects/
- Los clasifican en 4 categorías:
  - **Flagship**  : proyectos de alto valor estratégico.
  - **Production**  : proyectos listos para producción.
  - **Lab**  : Proyectos con entregables revisados por OWASP.
  - **Incubator**  : Proyectos aún en fase experimental o fábrica de ideas.
- Hay de todo tipo: herramientas, librerías de código, documentación, … **Más de 300 proyectos** en el inventario.

Introducción

Flagship projects

- OWASP **Amass**
- OWASP **Top 10**
- OWASP **Web Security Testing Guide** (WSTG)
- OWASP **Mobile Security Testing Guide** (MSTG)
- OWASP **ZAP**
- OWASP **ModSecurity Core Rule Set**
- OWASP **Juice Shop**
- OWASP **Cheat Sheet Series**
- OWASP **Security Shepherd**

Introducción



Production projects

- OWASP **API Security Project**
- OWASP **Bug Logging Tool**
- OWASP **Coraza Web Application Firewall**
- OWASP **CSRFGuard**
- OWASP **WrongSecrets**

Introducción

Lab projects

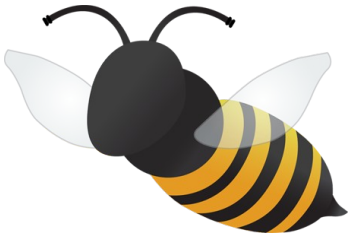
- OWASP **Mobile Top 10**
- OWASP **Internet of Things**
- OWASP **AntiSamy**
- OWASP **Attack Surface Detector**
- OWASP **Java HTML Sanitizer**
- OWASP **Secure Coding Dojo**
- OWASP **Secure Header Project**
- OWASP **Vulnerable Web Applications Directory**
- OWASP **WebGoat**
- OWASP **Benchmark**

Introducción

Incubator projects

- OWASP **AWScanner**
- OWASP **Damn Vulnerable Web Sockets**
- OWASP **Developer Guide**
- OWASP **Docker Top 10**
- OWASP **Forensics Testing Guide**
- OWASP **Honeypot**
- OWASP **Kubernetes Security Testing Guide**
- OWASP **Kubernetes Top Ten**

OWASP Top 10



- https://owasp.org/www-project-top-ten/
- **Proyecto documental** que trata de concienciar sobre **riesgos en aplicaciones Web**.
- Es un **estándar *de facto***
- Recoge los **10 principales riesgos de seguridad en aplicaciones web**
  - Basado en datos de más de 40 empresas dedicadas a la seguridad.
  - Recoge además datos de cientos de organizaciones y alrededor de 100K aplicaciones reales y APIs
- **Objetivo**: educar a desarolladores, diseñadores, arquitectos, managers y organizaciones.

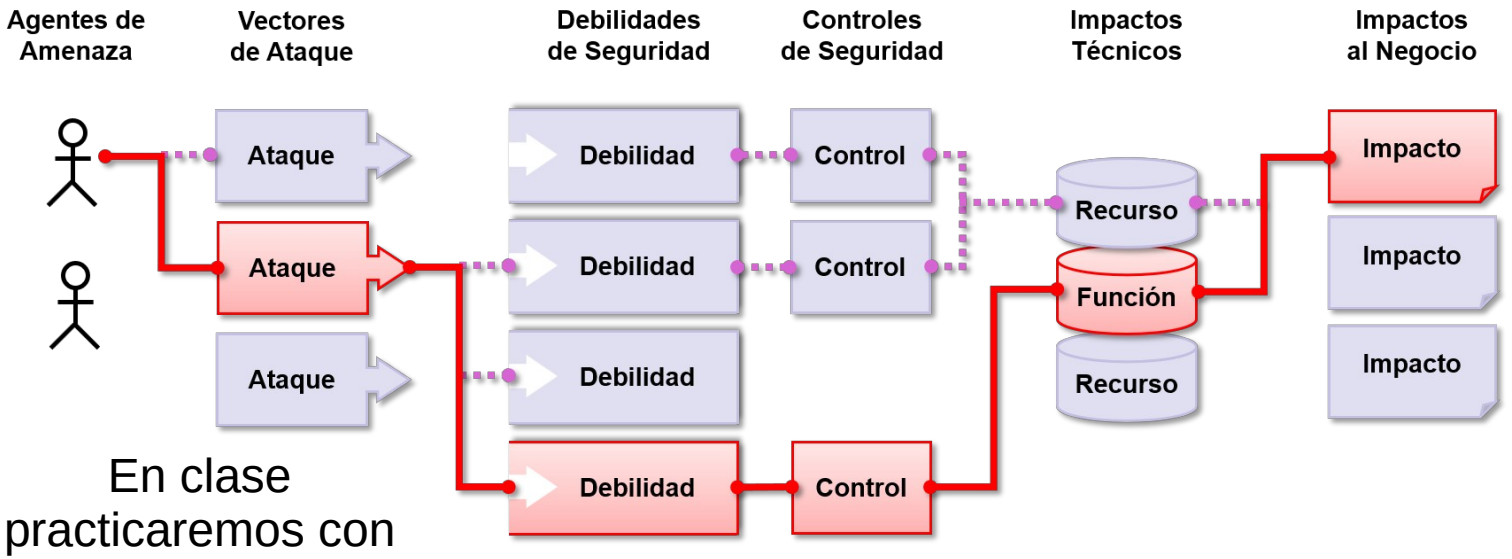
OWASP Top 10

- Para elaborar el listado se hace un recuento de cada **CWE** (**Common Weakness Enumeration**) presente en una aplicación.
- Con la lista de CWE (casi 400 en la versión 2021) se agrupan en categorías y se ordenan en función de su explotabilidad e impacto. De este modo se construyen las **8 primeras categorías del Top 10**.
- Las **2 últimas categorías** se elaboran a través de encuestas realizadas por la comunidad.
  - Los datos recabados del análisis de aplicaciones es mirar al pasado por lo que se podrían estar perdiendo las últimas tendencias.
  - Así se recogen datos de vulnerabilidades que pueden tener un impacto alto en la actualidad.

OWASP Top 10

¿Qué es un riesgo?

- Los atacantes pueden usar múltiples caminos para atacar una organización.
- Para determinar cada riesgo se tiene en cuenta:
  - probabilidad asociada a cada **agente de amenaza** + **vector de ataque** + **debilidades de seguridad** en combinación con el **impacto técnico** e **impacto al negocio**



En clase practicaremos con **vectores de ataque**

Cada uno de estos **caminos** representa un **riesgo**

OWASP Top 10

- Los 10 **riesgos** de OWASP Top 10 se **priorizan** en función de: su **frecuencia** (prevalencia), su **explotabilidad** (facilidad de explotación), grado de **detectabilidad**, impacto.
- Para cada **riesgo** se tiene en cuenta lo siguiente:

Agente de Amenaza	Explotabilidad	Prevalencia de Vulnerabilidad	Detección de Vulnerabilidad	Impacto Técnico	Impacto de Negocio
Específico de la Aplicación	Fácil 3	Difundido 3	Fácil 3	Severo 3	Específico del Negocio
	Promedio 2	Común 2	Promedio 2	Moderado 2	
	Difícil 1	Poco Común 1	Difícil 1	Mínimo 1	

- Aunque se trata de generalizaciones, permite clasificar cada riesgo por orden de importancia.

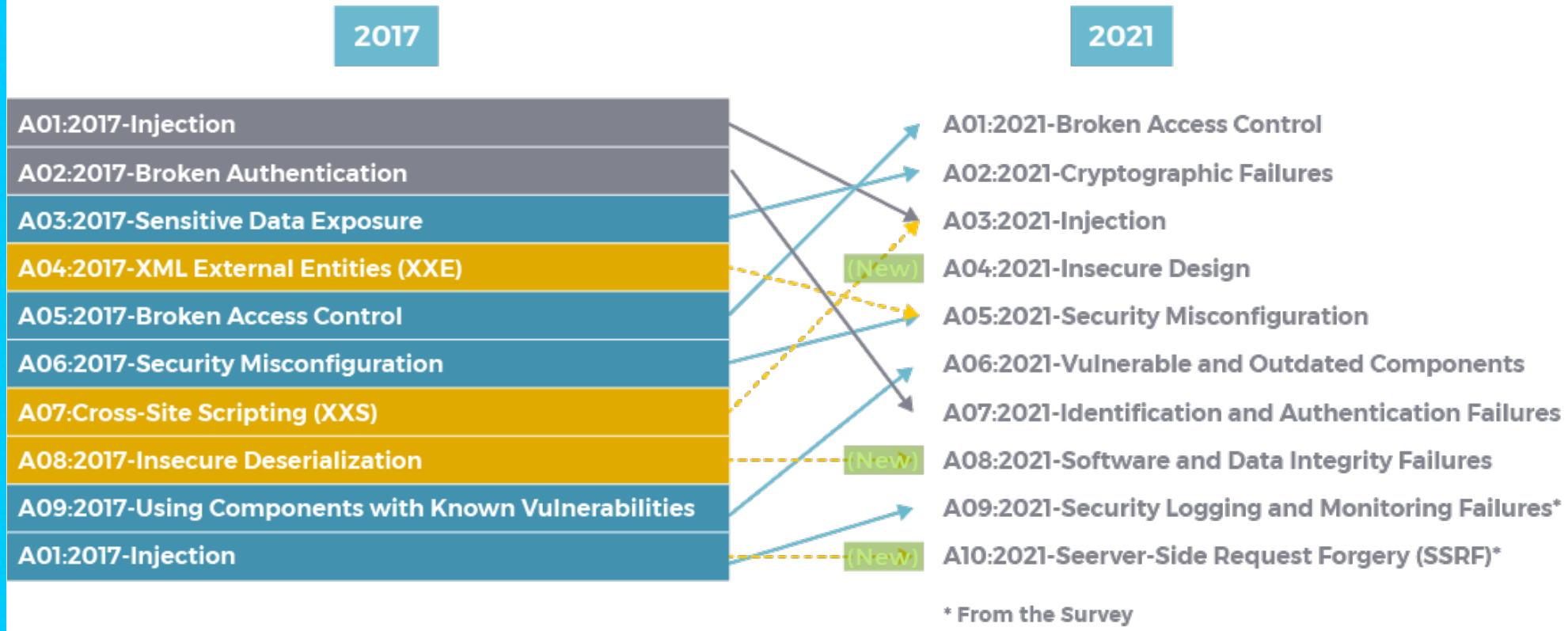
OWASP Top 10

Cambios de OWASP Top 10 – 2013 a OWASP Top 10 - 2017

OWASP Top 10 - 2013	→	OWASP Top 10 - 2017
A1 – Injection	→	A1:2017-Injection
A2 – Broken Authentication and Session Management	→	A2:2017-Broken Authentication
A3 – Cross-Site Scripting (XSS)	↘	A3:2017-Sensitive Data Exposure
A4 – Insecure Direct Object References [Merged+A7]	U	A4:2017-XML External Entities (XXE) [NEW]
A5 – Security Misconfiguration	↘	A5:2017-Broken Access Control [Merged]
A6 – Sensitive Data Exposure	↗	A6:2017-Security Misconfiguration
A7 – Missing Function Level Access Contr [Merged+A4]	U	A7:2017-Cross-Site Scripting (XSS)
A8 – Cross-Site Request Forgery (CSRF)	☒	A8:2017-Insecure Deserialization [NEW, Community]
A9 – Using Components with Known Vulnerabilities	→	A9:2017-Using Components with Known Vulnerabilities
A10 – Unvalidated Redirects and Forwards	☒	A10:2017-Insufficient Logging&Monitoring [NEW,Comm.]

OWASP Top 10

Cambios de OWASP Top 10 – 2017 a OWASP Top 10 - 2021



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

OWASP Top 10 - 2021

Lista de riesgos de OWASP Top 10

- **A1:2021 – Broken Access Control** (Pérdida de Control de Acceso)
- **A2:2021 – Cryptographic Failures** (Errores criptográficos)
- **A3:2021 – Injection** (Inyección)
- **A4:2021 – Insecure Design** (Diseño inseguro)
- **A5:2021 – Security Misconfiguration** (Configuración de seguridad incorrecta)
- **A6:2021 – Vulnerable and Outdated Components** (Componentes vulnerables y desactualizados)
- **A7:2021 – Identification and Authentication Failures** (Fallos en identificación y autenticación)
- **A8:2021 – Software and Data Integrity Failures** (Fallos en el software y en la integridad de los datos)
- **A9:2021 – Security Logging and Monitoring Failures** (Fallos en el registro y monitoreo)
- **A10:2021 – Server Side Request Forgery** (Falsificación de solicitudes del lado servidor)

A1:2021 – Broken Access Control

- **Control de acceso** → restricciones sobre lo que un usuario puede hacer o no en una aplicación web (admin vs usuario normal)
- Explotación de esos mecanismos puede permitir a un usuario corriente a acceder a datos sensibles, realizar funcionalidad no autorizada, …
- Alta tasa de incidencia (3.81% de las aplicaciones analizadas).
- Ejemplos:
  - ***CWE-35: Path Traversal*** que permite a un usuario salirse de la ruta web asignada y acceder a ficheros del sistema operativo local.
  - ***CWE-352: Cross-Site Request Forgery (CSRF)***.
  - ***CWE-284: Improper Access Control***.

A2:2021 – Cryptographic Failures

- **Causa raíz** → fallos criptográficos que tiene como consecuencia la exposición de datos confidenciales (integra la categoría ***A3:2017 Sensitive Data Exposure***).
- Alta tasa de incidencia (4.49% de las aplicaciones analizadas).
- Ejemplos:
  - ***CWE-328: Use of Weak Hash.***
  - ***CWE-261: Weak Encoding for Password***.
  - ***CWE-326: Inadequate Encryption Strength***.

A3:2021 – Injection

- Los ataques de inyección se producen cuando una aplicación:
  - No valida ni sanea las entradas de usuario.
  - Realiza consultas dinámicas sin codificar los parámetros.
  - Permite afectar a sentencias SQL, procedimientos almacenados o comandos ejecutados de manera subyacente por la misma.
- Ejemplos:
  - ***CWE-89: SQL injection.***
  - ***CWE-78: OS Command Injection***.
  - ***CWE-79: Cross-Site Scripting***.

A4:2021 – Insecure Design

- Categoría nueva que recoge vulnerabilidades causadas por un mal diseño de la aplicación.
- Ejemplos:
  - ***CWE-653: Improper Isolation or Compartmentalization.***
  - ***CWE-656: Reliance on Security Through Obscurity***.

A5:2021 – Security Misconfiguration

- Estos problemas de seguridad están relacionados con deficiencias en la configuración del software o del entorno: falta de hardening, funciones innecesarias habilitadas, no enviar cabeceras o directivas de seguridad cuando sea necesario, mantener cuentas predeterminadas, mostrar demasiada información a los usuarios en los errores, etc.
- Ejemplos:
  - ***CWE-260: Password in Configuration File.***
  - ***CWE-1004: Sensitive Cookie without flag 'HttpOnly'***.
  - ***CWE-611: Improper Restriction of XML External Entity Reference*** (anteriormente A3:2017).

A6:2021 – Vulnerable and Outdated Components

- Riesgos a los que se expone una aplicación web debido a vulnerabilidades en alguno de los componentes que usa debido a no estar suficientemente actualizados: sistema operativo, servidor web o de aplicaciones, SGBD, bibliotecas y dependencias, incluyendo las anidadas.
- Un ejemplo de ellos es lo ocurrido con la vulnerabilidad ***log4shell*** (CVE-2021-44228), en la librería ***log4j*** de registro de logs en Java presente en multitud de aplicaciones.
- Ejemplos:
  - ***CWE-937: Using Components with Known Vulnerabilities***.
  - ***CWE-1104: Use of Unmaintained Third Party Components***.

A7:2021 – Identification and Authentication Failures

- Vulnerabilidades relacionadas con la identidad, la autenticación y la gestión de sesiones de usuario. Algunos ejemplos:
  - Permitir ataques de fuerza bruta y/o automatizados
  - Permitir contraseñas débiles o bien conocidas (admin/admin), (admin/password), …
  - Ofrecer un proceso débil o no efectivo para el recordatorio de contraseñas.
  - Exponer el identificador de sesión en la URL.
  - Reutilizar el identificador de sesión.
- Ejemplos:
  - ***CWE-613: Insufficient Session Expiration***.
  - ***CWE-620: Unverified Password Change***.

A8:2021 – Software and Data Integrity Failures

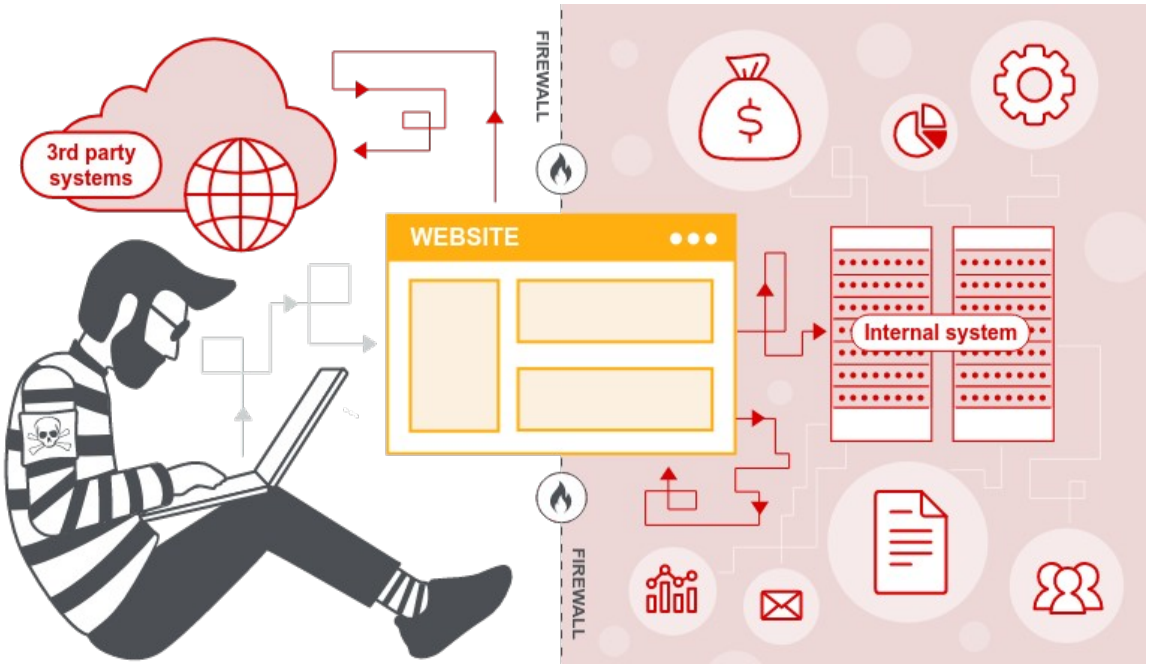
- Estas vulnerabilidades se deben a que la aplicación web depende de plugins, bibliotecas, módulos o CDNs no confiables que pueden ocasionar una vía de acceso no autorizado. Algunos ejemplos:
  - Actualizaciones no firmadas.
  - Instalación de plugins de terceros sin verificar su código.
- Ejemplos:
  - ***CWE-829: Inclusion of Functionality from Untrusted Control Sphere***.
  - ***CWE-494: Download of Code Without Integrity Check***.
  - ***CWE-502: Deserialization of Untrusted Data***.

A9:2021 – Security Logging and Monitoring

- Trata de deficiencias en el monitoreo y registro de eventos del sistema cuya ausencia o incorrecta gestión puede retrasar la respuesta a incidentes. Algunos ejemplos:
  - No registrar eventos como inicios y fines de sesión (tanto exitosos como fallidos).
  - Realizar registros de error que no proporcionan información precisa y útil al auditor.
- Ejemplos:
  - ***CWE-223: Omission of Security-relevant Information***.
  - ***CWE-532: Insertion of Sensitive Information into Log File***.
  - ***CWE-778: Insufficient Logging***.

A10:2021 – Server Side Request Forgery (SSRF)

- Esta vulnerabilidad ocurre cuando la aplicación web realiza una petición a un recurso remoto sin validar adecuadamente la URL suministrada por el usuario, lo que permite a un atacante construir una petición a un destino no esperado. Algunos ejemplos:
- Fue introducida en el Top-10 por encuestas a la comunidad y agrupa un único CWE:
  - ***CWE-918: Server side Request Forgery***.



https://portswigger.net/web-security/ssrf

FIN