Seguridad en Aplicaciones

Prof. Sebastián Martínez

Open Web Application Security Project

OWASP

OWASP

Open Web Application Security Project

Prof. Sebastián Martínez

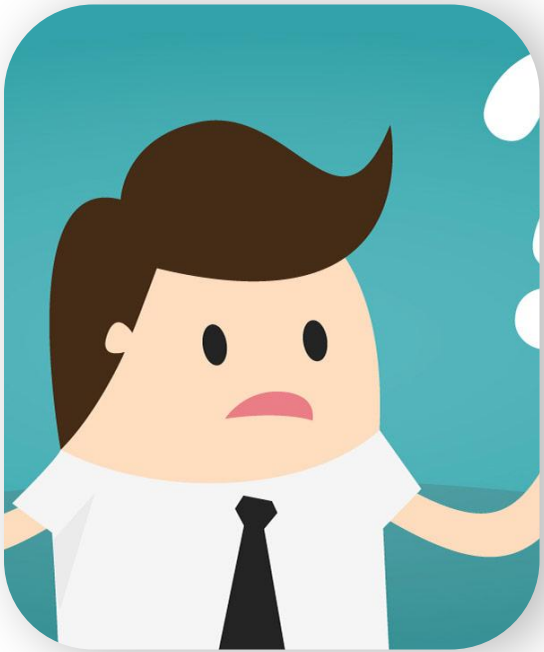
Inter
Escuela Internacional de Estudios Superiores

¿Qué es?
¿Cuál es su finalidad?
¿TOP 10?

Prof. Sebastián Martínez

Inter
Escuela Internacional de Estudios Superiores

¿Qué es?



OWASP es una organización sin fines de lucro que se dedica a mejorar la seguridad del software.

Misión: Hacer que la seguridad de las aplicaciones sea visible para que las organizaciones puedan tomar decisiones informadas.

Estándar de facto en la industria para la seguridad de aplicaciones.

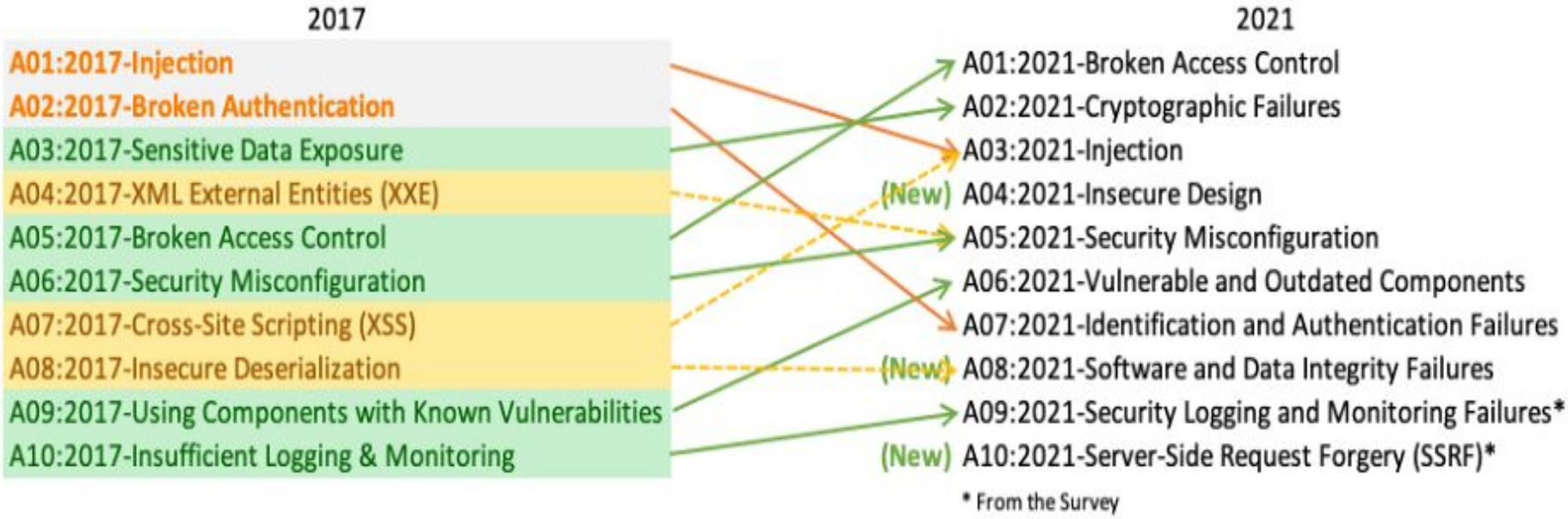
¿Qué significa "OWASP Top 10"?

El OWASP Top 10 es una lista de los 10 riesgos de seguridad más críticos que afectan a las aplicaciones web.

- Es una referencia global para identificar y mitigar vulnerabilidades comunes.
- Está basado en datos reales de vulnerabilidades reportadas y analizadas por la comunidad de OWASP.

¿Qué significa "OWASP Top 10"?

- Concientizar a desarrolladores, empresas y profesionales de seguridad sobre los riesgos más frecuentes.
- Proporcionar una guía práctica para priorizar esfuerzos en la mejora de la seguridad.
- La lista se revisa y actualiza cada 3-4 años para reflejar los cambios en el panorama de amenazas.



A01:2021 – Control de acceso roto

Los atacantes pueden acceder a recursos o funcionalidades no autorizadas.

Ejemplo: Acceso a archivos o datos sensibles sin autenticación.

Mitigación: Implementar controles de acceso estrictos y pruebas regulares.

Prof. Sebastián Martínez

A02:2021 – Fallas criptográficas

Uso incorrecto de algoritmos criptográficos o manejo inseguro de claves.

Ejemplo: Almacenamiento de contraseñas en texto plano.

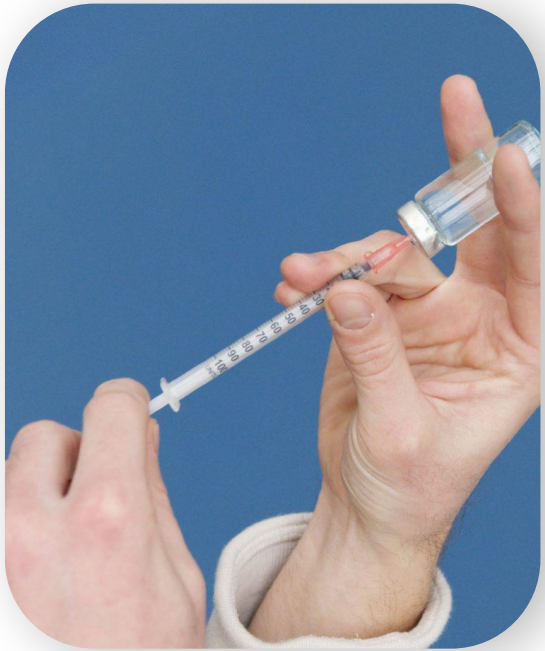
Mitigación: Usar algoritmos robustos y gestionar claves de forma segura.

A03:2021 – Inyección

Inserción de código malicioso en entradas no validadas.

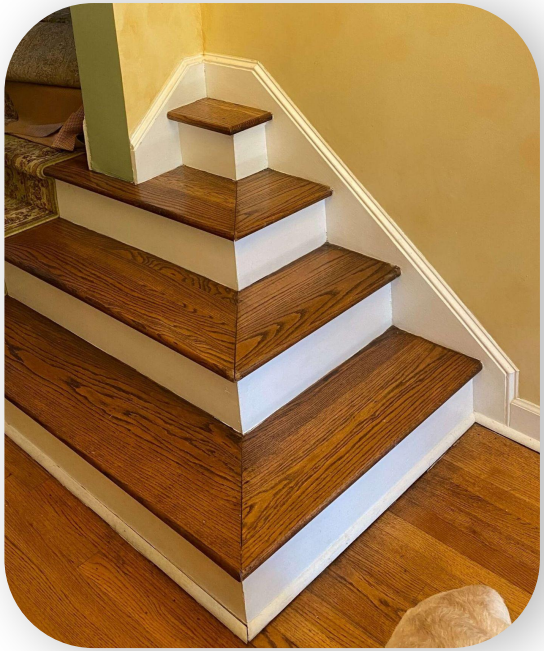
Ejemplo: SQL Injection, XSS (Cross-Site Scripting).

Mitigación: Validar y sanitizar todas las entradas, usar consultas parametrizadas.



Prof. Sebastián Martínez

A04:2021 – Diseño inseguro



Falta de consideración de la seguridad en la fase de diseño.

Ejemplo: Flujos de autenticación débiles.

Mitigación: Adoptar un enfoque de "seguridad por diseño".

A05:2021 – Configuración de seguridad incorrecta

Configuraciones predeterminadas o inseguras en servidores y aplicaciones.

Ejemplo: Uso de credenciales por defecto.

Mitigación: Revisar y ajustar configuraciones según mejores prácticas.

A06:2021 - Componentes vulnerables y desactualizados

Uso de bibliotecas o frameworks con vulnerabilidades conocidas.

Ejemplo: Explotación de vulnerabilidades en versiones antiguas de software.

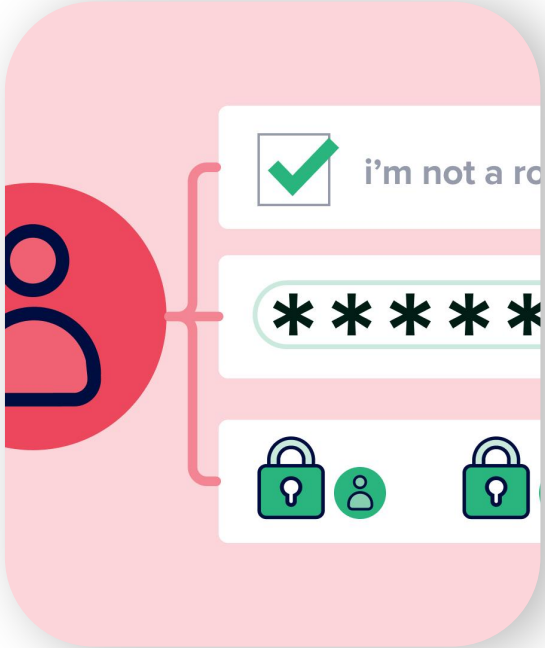
Mitigación: Mantener actualizados todos los componentes.

A07:2021 – Identificación y autenticación fallidas

Debilidades en los mecanismos de autenticación y gestión de sesiones.

Ejemplo: Ataques de fuerza bruta o sesiones no expiradas.

Mitigación: Implementar autenticación multifactor y políticas de contraseñas robustas.



Prof. Sebastián Martínez

A08:2021 – Integridad de software y datos comprometida

Manipulación no autorizada de datos o código.

Ejemplo: Ataques de inyección de dependencias.

Mitigación: Verificar la integridad de los datos y usar firmas digitales.

A09:2021 – Fallas en el monitoreo de seguridad

Falta de detección y respuesta ante incidentes de seguridad.

Ejemplo: No registrar intentos de acceso fallidos.

Mitigación: Implementar sistemas de monitoreo y registro (logging).

Prof. Sebastián Martínez

A10:2021 – Falsificación de solicitudes del lado del servidor (SSRF)

Explotación de solicitudes maliciosas desde el servidor.

Ejemplo: Acceso a recursos internos no autorizados.

Mitigación: Validar y restringir las solicitudes entrantes.