DESARROLLO DE APLICACIONES WEB

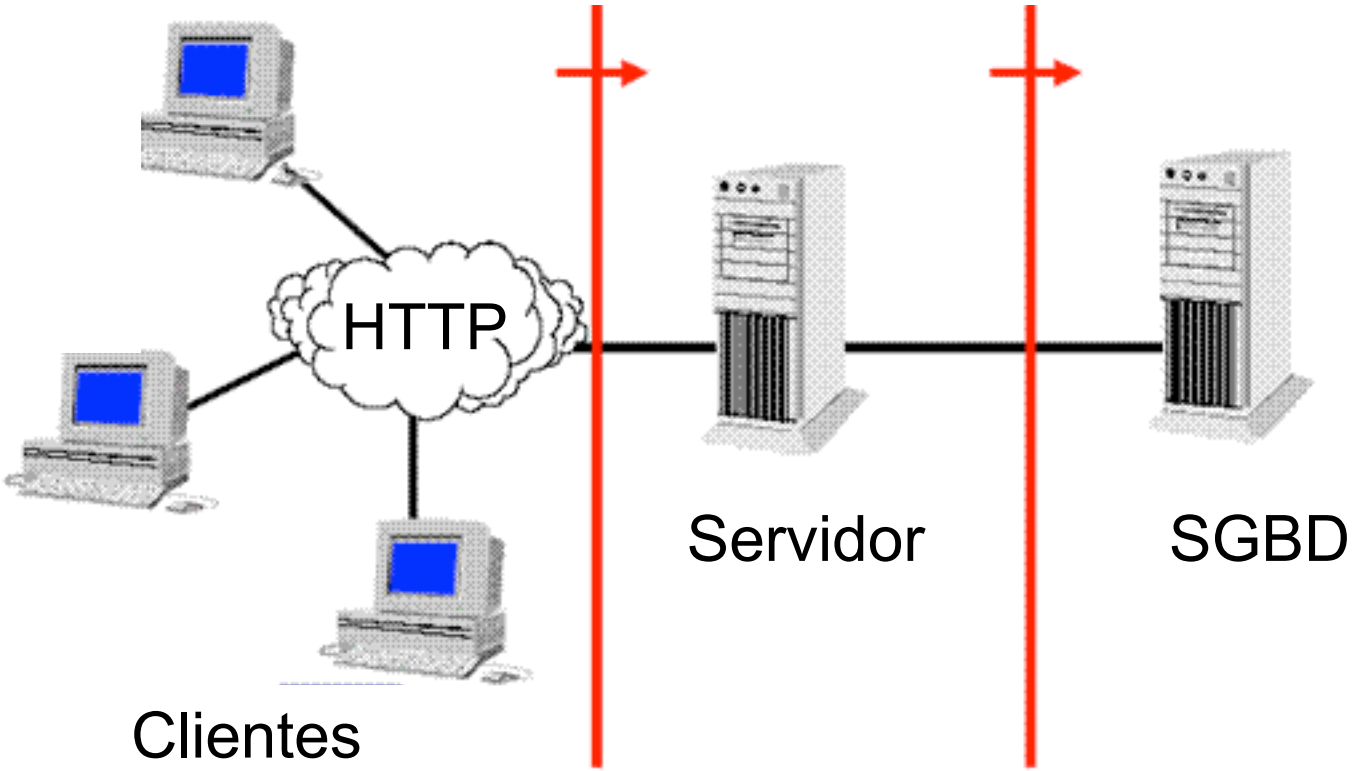
Tema 6.- Otras Tecnologías

Seguridad Web

Contenido

- Introducción
- Protocolo Seguro
- Seguridad en el cliente
- Seguridad en el servidor y en el SGBD
- Seguridad en la aplicación
- Principales amenazas: OWASP
- Librerías y utilidades
- Referencias

Introducción

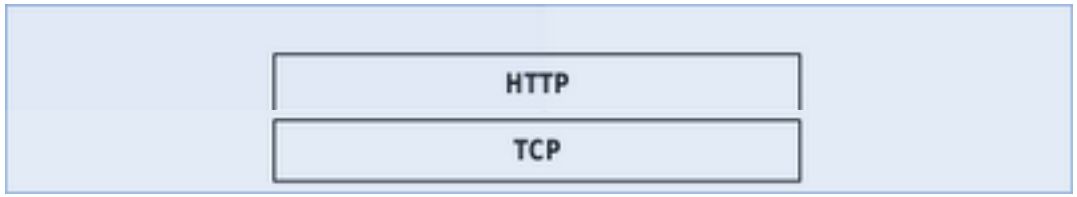


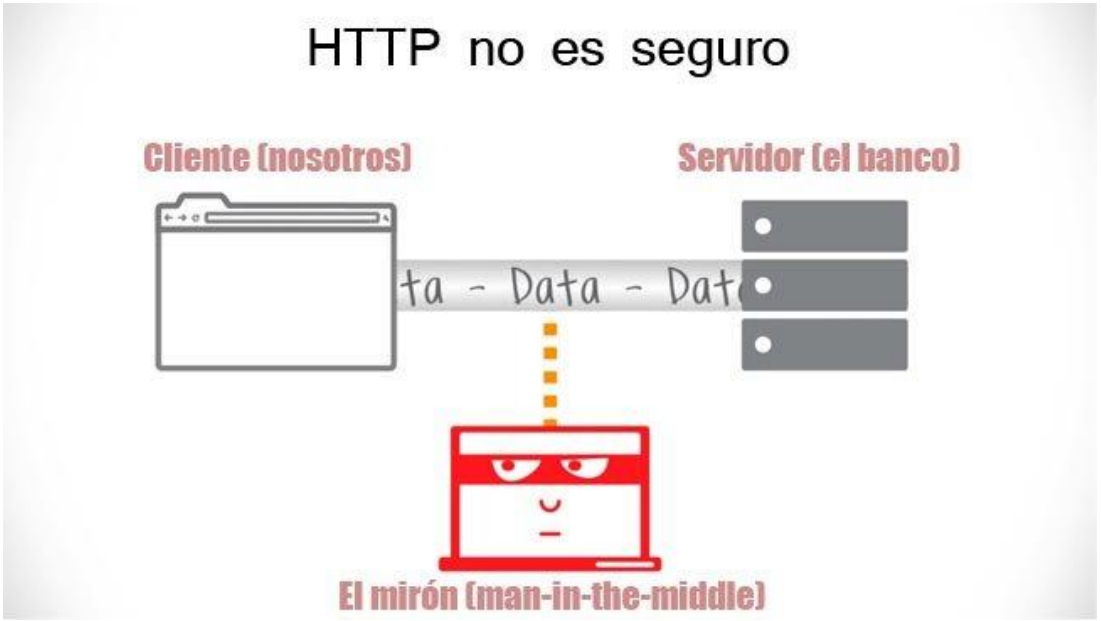
Introducción

- Seguridad de Comunicación
  - HTTP
- Seguridad en las tecnologías del Cliente
  - Navegador
  - Lenguajes
- Seguridad en las tecnologías del Servidor
  - Servidor Web / Servidor de Aplicaciones
  - SGBD
  - Lenguajes / Frameworks
- Seguridad en la Aplicación (top amenazas)
  - Autenticación, Autorización, confidencialidad, etc
  - Validación de datos
  - Programación segura

Protocolo Seguro

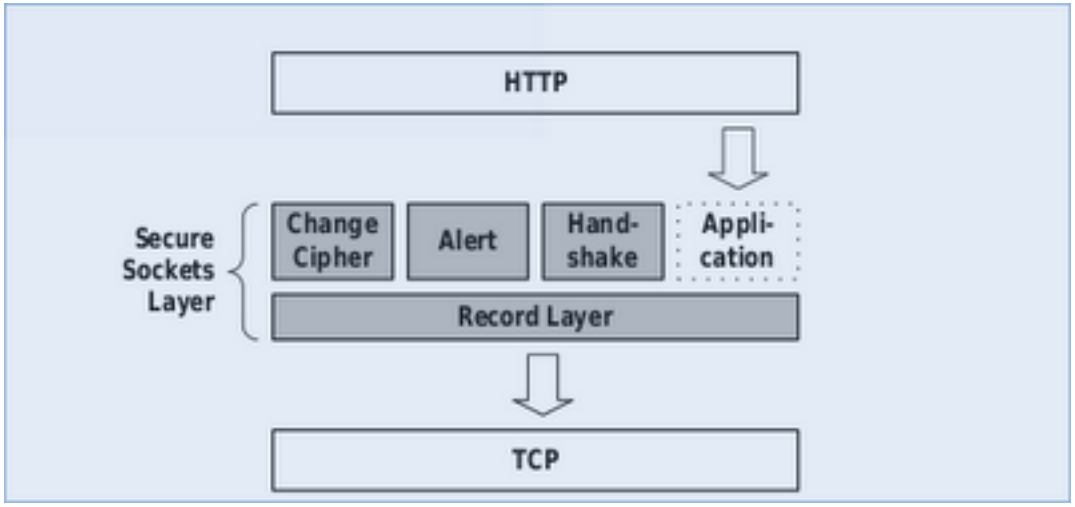
- Seguridad de Comunicación
  - HTTP: Protocolo no seguro





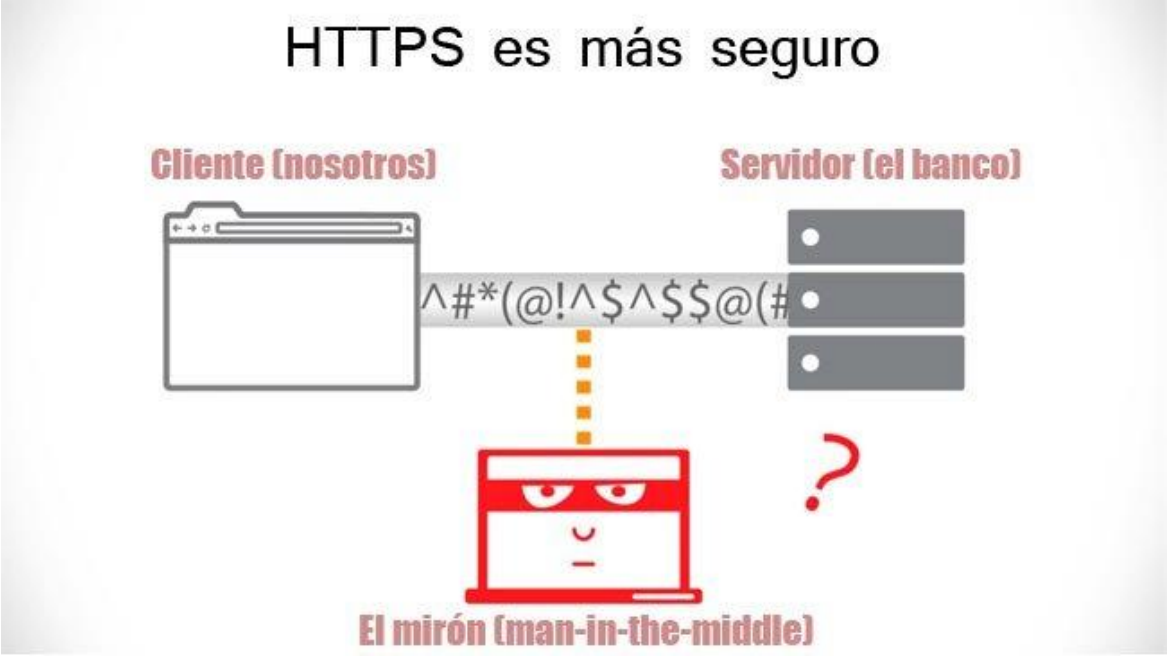
Protocolo Seguro

- Seguridad de Comunicación
  - SSL (Secure Sockets Layer) y TLS (Transport Layer Security)
  - HTTPS: Protocolo seguro



Protocolo Seguro

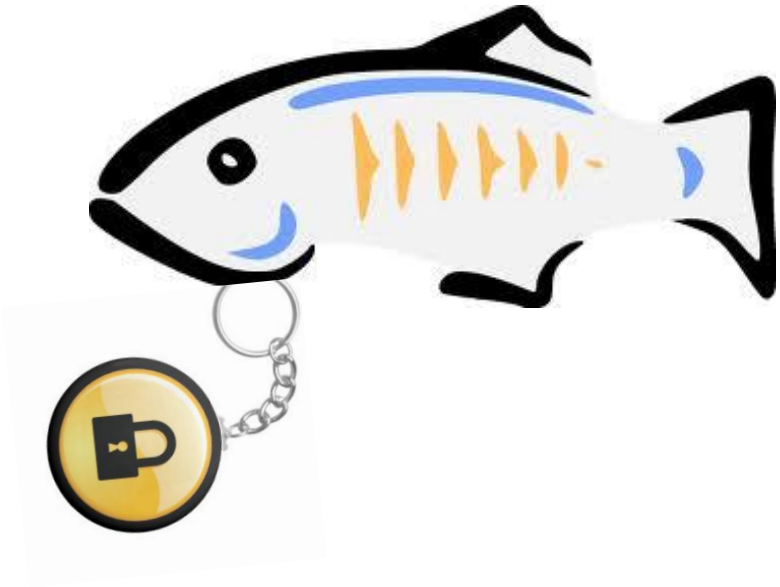
- Seguridad de Comunicación
  - SSL (Secure Sockets Layer) y TLS (Transport Layer Security)
  - HTTPS: Protocolo seguro



  - Más información sobre SSL y TSL (ver vídeo)

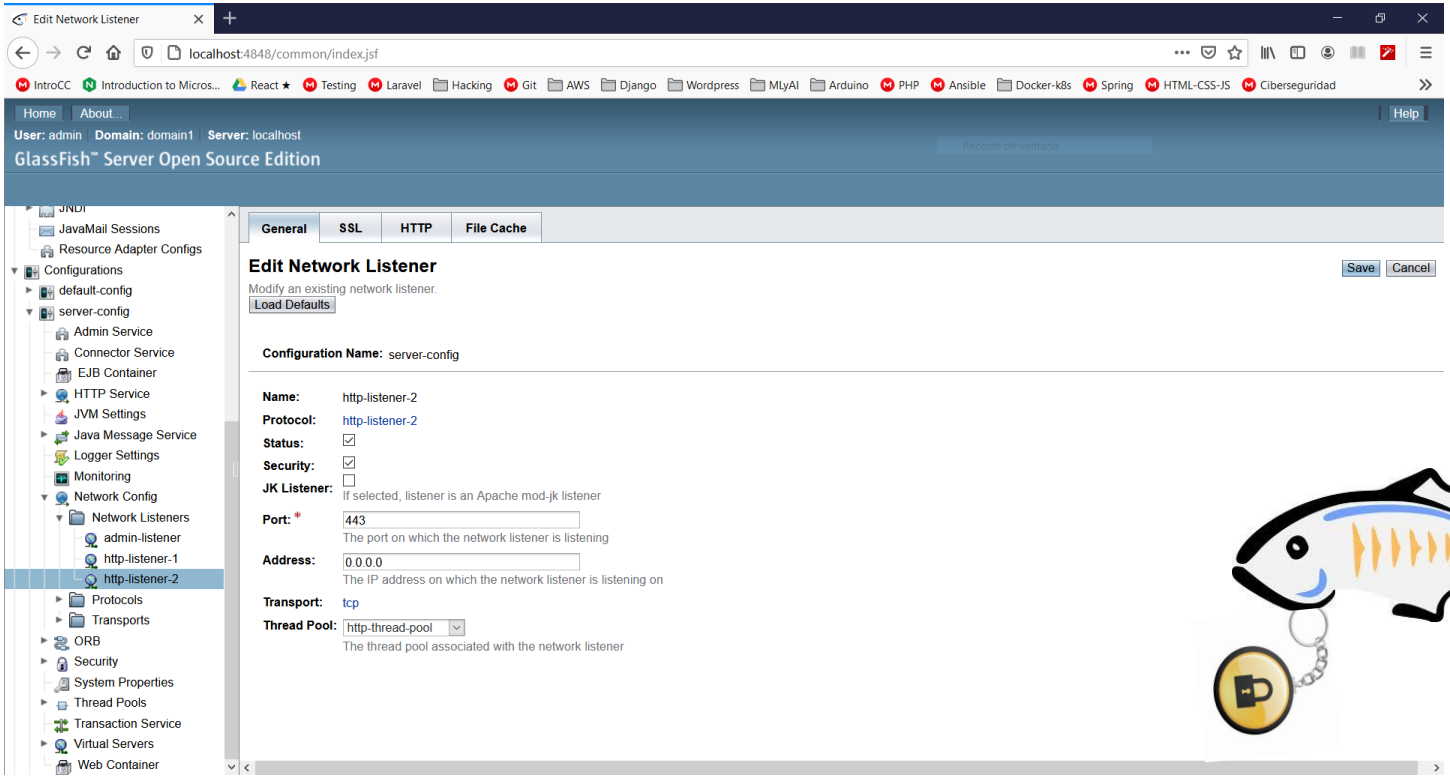
Protocolo Seguro

- Seguridad de Comunicación
  - SSL (Secure Sockets Layer) y TLS (Transport Layer Security)
  - HTTPS: Protocolo seguro



Protocolo Seguro

- Seguridad de Comunicación
  - SSL (Secure Sockets Layer) y TLS (Transport Layer Security)
  - HTTPS: Protocolo seguro



Seguridad en el Cliente

- Seguridad en las tecnologías del Cliente
  - Navegador
  - Lenguajes

JavaScript
Applet
Plugins
etc.

Seguridad en el Servidor

- Seguridad en las tecnologías del Servidor
  - Servidor Web
  - SGBD
  - Lenguajes

    - **Problemas**

    - Vulnerabilidades debidas a versiones no actualizadas
    - Uso de configuraciones por defecto o inadecuadas
    - Permisos sobre dimensionados

Seguridad en el Servidor

- Seguridad en las tecnologías del Servidor
  - Servidor Web
  - SGBD
  - Lenguajes

    - **Soluciones**
    - Actualizar a versiones mejoradas
    - Personalizar la configuración
    - Deshabilitar funcionalidad no utilizada
    - Registrar actividad en registros de logs y revisarlos periódicamente
    - Definir usuarios con los permisos necesarios
    - Poner cortafuegos y limitaciones al acceso remoto
    - Encriptar datos sensibles

Seguridad en la Aplicación

- Seguridad en la Aplicación (top amenazas)
  - Aquí es donde tenemos que hacer el mayor trabajo
    - Identificación y Autorización de usuarios
    - Formularios con tokens CFRS y Captchas
    - Tamaño y contenido de passwords seguras
    - No almacenar passwords, si no su huella
    - Definir procedimientos seguros de recuperación
    - Validación de los datos tanto en el cliente como en el servidor antes de hacer uso de ellos
    - Definir tiempo de sesiones seguros

Principales Amenazas: Top 10

2021 / 2025

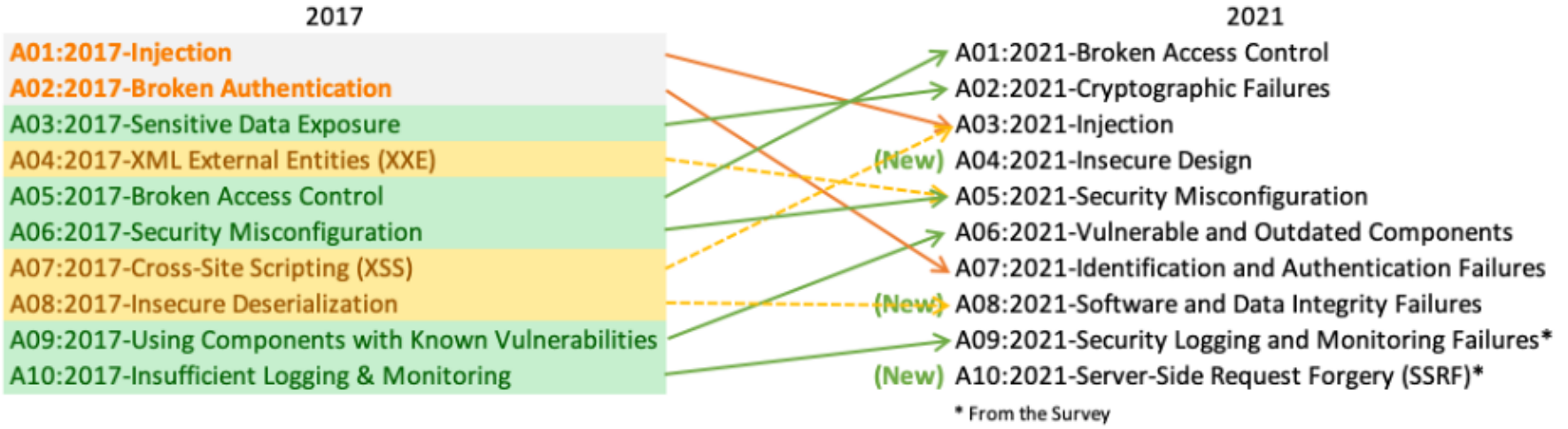
https://owasp.org/www-project-top-ten/

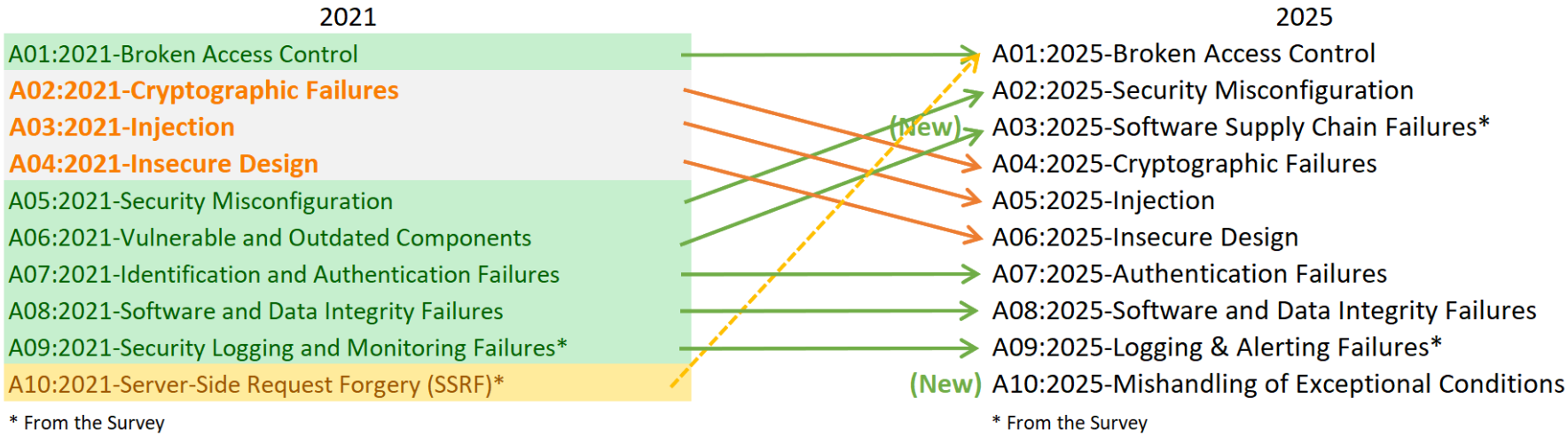
https://owasp.org/Top10/

Principales Amenazas: Top 10 - 2017

https://www.owasp.org/images/5/5e/OWASP-Top-10-2017-es.pdf

Principales Amenazas: 2017-2021-2025





Principales Amenazas: 2013-2017

OWASP Top 10 2013	±	OWASP Top 10 2017
A1 – Inyección	→	A1:2017 – Inyección
A2 – Pérdida de Autenticación y Gestión de Sesiones	→	A2:2017 – Pérdida de Autenticación y Gestión de Sesiones
A3 – Secuencia de Comandos en Sitios Cruzados (XSS)	↘	A3:2017 – Exposición de Datos Sensibles
A4 – Referencia Directa Insegura a Objetos [Unido+A7]	U	A4:2017 – Entidad Externa de XML (XXE) [NUEVO]
A5 – Configuración de Seguridad Incorrecta	↘	A5:2017 – Pérdida de Control de Acceso [Unido]
A6 – Exposición de Datos Sensibles	↗	A6:2017 – Configuración de Seguridad Incorrecta
A7 – Ausencia de Control de Acceso a las Funciones [Unido+A4]	U	A7:2017 – Secuencia de Comandos en Sitios Cruzados (XSS)
A8 – Falsificación de Peticiones en Sitios Cruzados (CSRF)	☒	A8:2017 – Deserialización Insegura [NUEVO, Comunidad]
A9 – Uso de Componentes con Vulnerabilidades Conocidas	→	A9:2017 – Uso de Componentes con Vulnerabilidades Conocidas
A10 – Redirecciones y reenvíos no validados	☒	A10:2017 – Registro y Monitoreo Insuficientes [NUEVO, Comunidad]

Principales Amenazas: Top 10 - 2021

A01: 2021 **Perdida de Control de Acceso**	A01:2021 – **Broken Access Control**. No se aplican correctamente las restricciones sobre lo que los usuarios autenticados pueden hacer. Los atacantes pueden explotar estos defectos para acceder, de forma no autorizada, a funcionalidades, cuentas de otros usuarios, datos sensibles, etc.
A02: 2021 **Fallos Criptográficos**	A02:2021 – **Cryptographic Failures**. (antes referenciado como Exposición de datos sensibles) se deben a la ausencia de métodos criptográficos para proteger la información sensible (como información financiera, de salud, etc), no cumpliendo leyes como la GDPR Europea y donde los atacantes pueden usar esa información para fraudes u otros delitos.
A03: 2021 **Inyección**	A03:2021 – **Injection**. Se producen cuando la aplicación no valida, filtra ni "desinfecta" los datos proporcionados por el usuario y se envían directamente como parte de un comando o consulta. Las más comunes son SQL, NoSQL, comando OS, asignación relacional de objetos (ORM), LDAP y lenguaje de expresión (EL).
A04: 2021 **Diseño Inseguro**	A04:2021 – **Insecure Design**. Nueva categoría en 2021, se centra en los riesgos relacionados con errores en el diseño y la arquitectura. Algunas debilidades son Generación de mensajes de error con información confidencial, credenciales insuficientemente protegidas, Infracción de límites de confianza.
A05: 2021 **Configuración de Seguridad Incorrecta**	A04:2021 – **Security Misconfiguration**. Es un problema muy común que se debe en parte a establecer la configuración de forma manual, por omisión, o directamente por la falta de configuración, o bien por habilitar funcionalidad innecesaria o sobredimensionar los permisos de los usuarios.

Principales Amenazas: Top 10 - 2021

A06: 2021
Componentes Obsoletos o Vulnerables

A01:2021 – **Vulnerable and Outdated Components**. Provocados por el uso de bibliotecas o frameworks obsoletos con vulnerabilidades conocidas con los que se pueden llevar a cabo diferentes tipos de ataques que tengan un impacto variado

A07: 2021
Fallos de Identificación y Autenticación

A02:2021 –**Identification and Authentication Failures**. Se centran en problemas relacionados con la confirmación de la identidad del usuario, la autenticación y la administración de sesiones

A08: 2021
Fallos de integridad de datos y software

A03:2021 – **Software and Data Integrity Failures**. Nueva categoría, que se centra en hacer suposiciones relacionadas con actualizaciones e integraciones de la aplicación y de los datos críticos sin verificar la integridad

A09: 2021
Fallos de Seguridad en Monitorización y Registro

A04:2021 –**Security Logging and Monitoring Failures**. El registro y monitoreo inexisten o insuficiente, junto a la falta de respuesta ante incidentes permiten a los atacantes mantener el ataque en el tiempo, pivotear a otros sistemas y manipular, extraer o destruir datos

A10: 2021
Falsificación de solicitudes del lado del servidor

A04:2021 – **Server-Side Request Forgery (SSRF)**. Los ataques SSRF ocurren cuando una aplicación está obteniendo un recurso remoto sin validar la URL proporcionada por el usuario. Permite que un atacante “coaccione” a la aplicación para que envíe una solicitud diseñada a un destino inesperado

Principales Amenazas: Top 10

- Inyección SQL
  - Sentencia SQL tradicional para identificar usuarios

```
sql = "SELECT id FROM usuarios
WHERE login = '" + user + "' AND password = '" + pass + "'";
```

  - Datos enviados:

user: ' OR 1=1 OR (login = '

pass: ') OR password = '

sql = "SELECT * FROM usuarios

WHERE login = ' 'OR 1=1 OR (login= ' ' AND password = ' ‘)

OR password = ' '

Principales Amenazas: Top 10

- Inyección SQL
  - Sentencia SQL tradicional para identificar usuarios

```
sql = "SELECT id FROM usuarios
WHERE login = '" + user + "' AND password = '" + pass + "'";
```

  - Insertando user: pepe'-- (NOTA: -- comentario SQL)

sql = "SELECT id FROM usuarios
WHERE login= 'pepe' -- AND password=' '

  - Insertando user: pepe y pass: ' OR ' '= '

sql = "SELECT id FROM usuarios
WHERE login='pepe' AND password=' ' OR ' '= ' '

Principales Amenazas: Top 10

- Inyección SQL
  - Sentencia SQL tradicional para identificar usuarios

sql = “SELECT id FROM usuarios WHERE id=“+ id;

  - Pasando en id: '; drop table usuarios;

SELECT id FROM usuarios WHERE id=‘ '; drop table usuarios

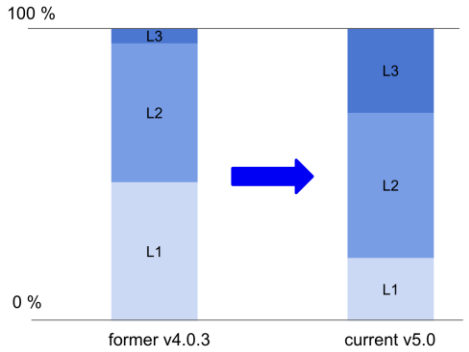
ASVS: Verificar Seguridad Web

OWASP ASVS Estándar de Verificación de Seguridad en Aplicaciones

ASVS es un marco de requisitos para garantizar la seguridad de las aplicaciones web y servicios web modernos

350 controles
3 niveles con una filosofía diferente al 4.x
17 requisitos o controles

ASVS



- Niveles de ASVS 4.x (286 controles en 14 requisitos)
  - ASVS Nivel 1: Mínimo nivel de garantía, comprobable con pentesting.
  - ASVS Nivel 2: Nivel recomendado para aplicaciones que contienen datos confidenciales que requieran.
  - ASVS Nivel 3: Nivel para aplicaciones críticas, como aquellas que realizan transacciones de alto valor o requieran un gran nivel de confianza.

- Niveles de ASVS 5.x (350 controles en 17 reqiusitos)
  - ASVS Nivel 1:Requisitos mínimos, críticos y básicos (20%)
  - ASVS Nivel 2: Ataques menos comunes o soluciones complejas a ataques comunes (50%)
  - ASVS Nivel 3: Mecanismos de defensa profundos (30%)

ASVS: Verificar Seguridad Web 4.x

- Requisitos
  - V1 Arquitectura, Diseño y Modelado de Amenazas
  - V2 Autenticación
  - V3 Gestión de sesiones
  - V4 Control de Acceso
  - V5 Validación, Desinfección y Codificación
  - V6 Criptografía almacenada
  - V7 Manejo y Registro de Errores
  - V8 Protección de Datos
  - V9 Comunicación
  - V10 Código Malicioso
  - V11 Lógica de Negocio
  - V12 Archivos y Recursos
  - V13 API y Servicios Web
  - V14 Configuración

ASVS 4.0.3
Spanish

ASVS: Verificar Seguridad Web 5.x

- Requisitos
  - V1 Encoding and Sanitization
  - V2 Validation and Business Logic
  - V3 Web Frontend Security
  - V4 API and Web Service
  - V5 File Handling
  - V6 Authentication
  - V7 Session Management
  - V8 Authorization
  - V9 Self-contained Tokens
  - V10 OAuth and OIDC
  - V11 Cryptography
  - V12 Secure Communication
  - V13 Configuration
  - V14 Data Protection
  - V15 Secure Coding and Architecture
  - V16 Security Logging and Error Handling
  - V17 WebRTC

ASVS 5.0

Probando vulnerabilidades

- Kali Linux
  - Distribución Linux para auditoría y seguridad informática
  - Penetration Testing, Security Research, Computer Forensics and Reverse Engineering
- Máquina Virtual WingkaL4bs
  - MV vulnerable basada en Ubuntu 14.04

    https://github.com/SVelizDonoso/wingkalabs

- OWASP Mutillidae II
  - Aplicación web deliberadamente vulnerable

    https://github.com/webpwnized/mutillidae

- Proyecto bWAPP (buggy Web APPlication)
  - Aplicación web deliberadamente vulnerable

    http://www.itsecgames.com/

Testing

- OWASP Web Security Testing Guide (WSTG) Project

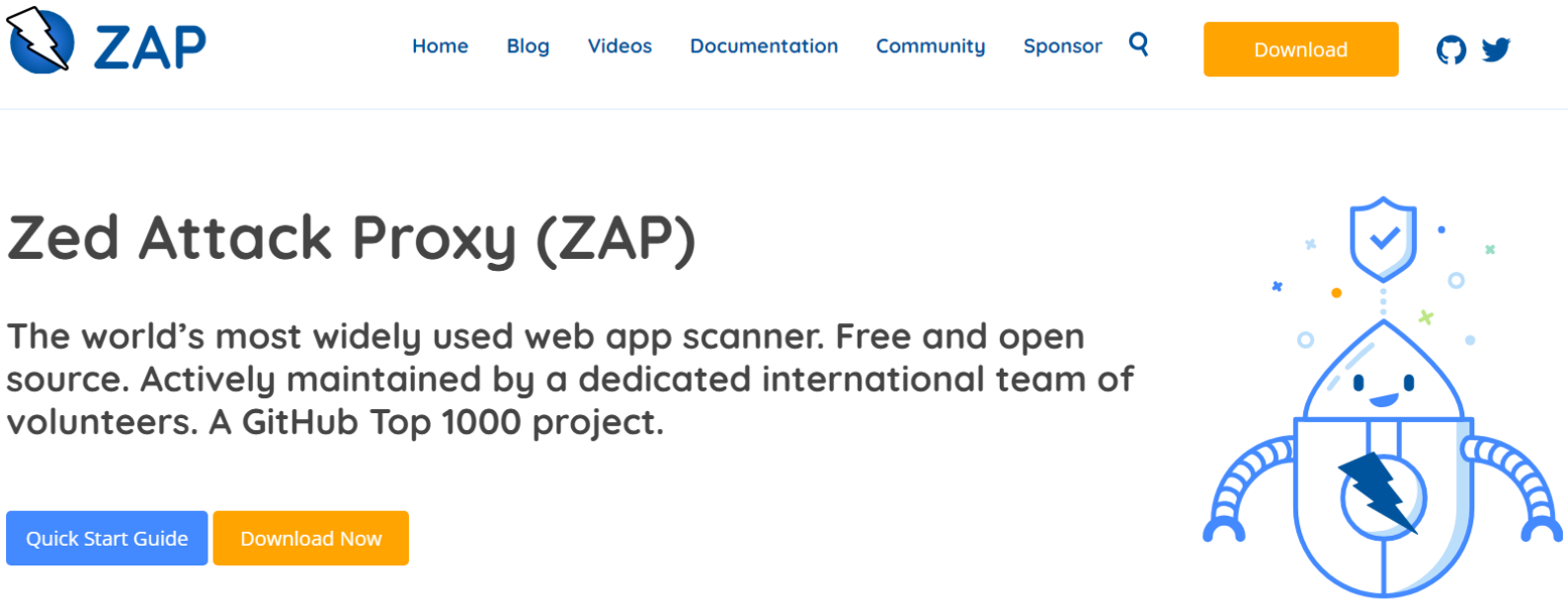
  https://owasp.org/www-project-web-security-testing-guide/

https://owasp.org/www-project-web-security-testing-guide/stable/

Recursos

- OWASP ZAP
  - Escáner de seguridad web para Pentesting

https://www.zaproxy.org/



Recursos

- Librería OWASP
  - Enterprise Security API (ESAPI)

https://owasp.org/www-project-enterprise-security-api/

OWASP Enterprise Security API (ESAPI)

Main | Downloads | What I did with ESAPI | Should I use ESAPI? | Glossary | OWASP ESAPI for Java EE | Project Details

owasp lab project

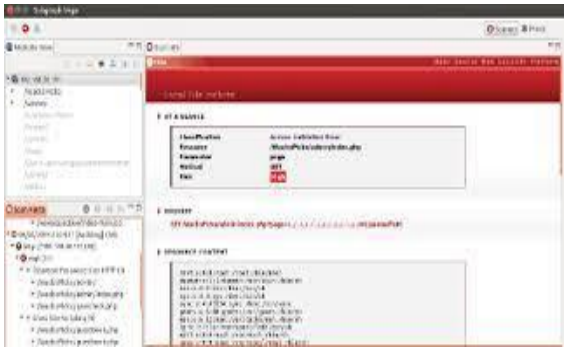
What is ESAPI?

ESAPI (The OWASP Enterprise Security API) is a free, open source, web application security control library that makes it easier for programmers to write lower-risk applications. The ESAPI libraries are designed to make it easier for programmers to retrofit security into existing applications. The ESAPI libraries also serve as a solid foundation for new development.

Allowing for language-specific differences, all OWASP ESAPI versions have the same basic design:

Recursos

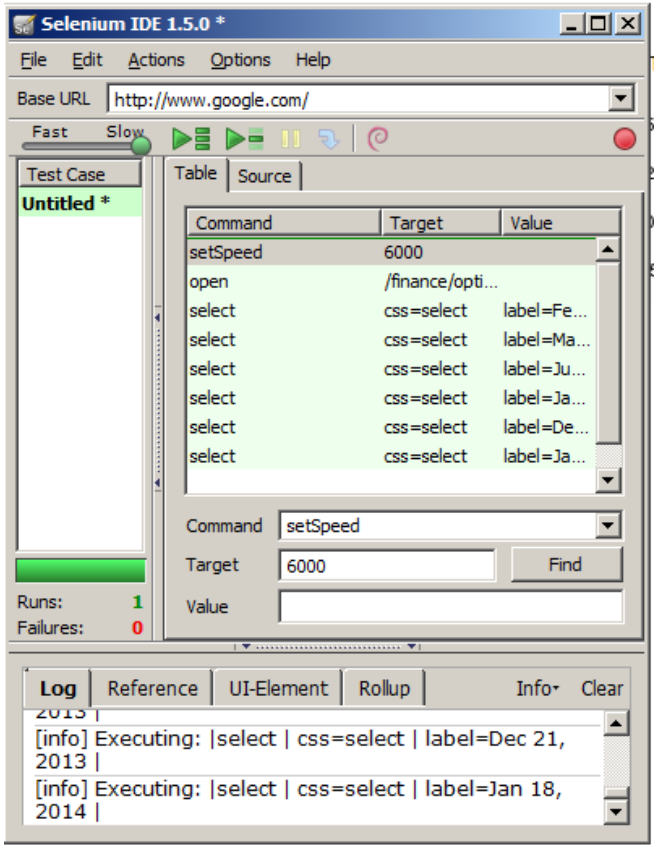
- Vega (https://subgraph.com/vega/)
  - Plataforma para escanear y probar la seguridad de Aplicaciones Web



- Sqlmap (https://sqlmap.org/)
  - Herramienta para realizar pruebas de penetración para la detección de errores en servidores de bases de datos

Recursos

- Selenium (http://www.seleniumhq.org/)
  - Entorno de pruebas de software para aplicaciones Web



Recursos

- Proyectos OWASP

  https://owasp.org/projects/

  - Amenazas Top 10

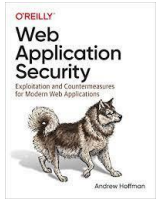
    https://owasp.org/www-project-top-ten/
  - Application Security Verification Standard

    https://owasp.org/www-project-application-security-verification-standard/
  - Web Security Testing Project

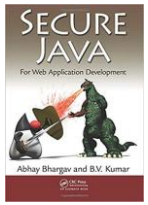
    https://owasp.org/www-project-web-security-testing-guide/

Recursos

- Libros



Web Application Security:
Exploitation and Countermeasures for Modern Web Applications
C. Kern , A. Kesavan , N. Daswani
Apress, 2020.



Secure Java: For Web Application Development
A. Bhargav, B. V. Kumar
CRC Press, 2017.

Seguridad informática:
Ethical Hacking. Conocer el ataque para una mejor defensa
ACISSI (Auditoría, Consejo, Instalación y Seguridad de Sist. de Información)
Ediciones ENI, 2022.