

UNIVERSIDAD TECNOLÓGICA DE LA SELVA

INGENIERÍA EN DESARROLLO Y GESTIÓN DE SOFTWARE

✓ ASIGNATURA:

SEGURIDAD EN EL DESARROLLO DE APLICACIONES

✓ NOMBRE DEL DOCENTE:

MTRO. ANDRÉS DONACIANO MARTÍNEZ GUILLÉN

NOMBRE DEL LOS INTEGRANTES:	PARTICIPACIÓN
JOSÉ ALEJANDRO DÍAZ GÓMEZ	100%
HUGO RUBÉN DÍAZ CRUZ	100%
MARIANA EURICE GUILLÉN NAVARRO	100%
FROILÁN NÁJERA MORALES	100%

✓ GRADO: 8 GRUPO:” A “TURNO: VESPERTINO

✓ LUGAR Y FECHA DE ENTREGA:

OCOSINGO, CHIAPAS A 08 DE ABRIL DEL 2025

Contenido

Introducción	3
Herramienta Utilizada.	4
• Instalación.....	4
• Pruebas empleando la herramienta.....	6
• Identificación del tipo de pruebas que se realizó la herramienta.....	7
• Identificación de los elementos que evalúa la herramienta	9
• Vulnerabilidades y tipo que localizó la herramienta	13
• Interpretación de la prueba realizada con la herramienta	17
Conclusión.....	18

Introducción

En el presente trabajo se utilizó la herramienta OWASP ZAP (Zed Attack Proxy) para realizar un análisis de seguridad a una aplicación web. El objetivo principal fue identificar posibles vulnerabilidades y evaluar qué tan expuesta se encuentra la aplicación ante ataques comunes, como inyección de código, fallos en la gestión de sesiones, o configuraciones inseguras.

A través de diferentes técnicas de escaneo —como el escaneo pasivo, activo y spidering— se logró obtener un panorama general del estado de seguridad de la aplicación. Posteriormente, se analizaron las alertas generadas, enfocándose en vulnerabilidades críticas como inyección SQL o Cross Site Scripting (XSS), permitiendo así interpretar los riesgos y proponer recomendaciones para su mitigación.

Este ejercicio forma parte del proceso de aprendizaje en pruebas de seguridad web y en el uso de herramientas automatizadas para la detección de fallos en entornos reales o simulados.

Herramienta Utilizada.

OWASP ZAP (Zed Attack Proxy)

- **Instalación.**

Se utilizó la herramienta **OWASP ZAP (Zed Attack Proxy)**, una plataforma gratuita y de código abierto especializada en pruebas de seguridad para aplicaciones web.

Sitio oficial: <https://www.zaproxy.org/>

The screenshot shows the official website for OWASP ZAP. At the top, there's a navigation bar with links for Blog, Videos, Documentation, Community, a search icon, and a prominent orange 'Download' button. To the right of the download button are social media icons for GitHub and Twitter. Below the navigation, the main title 'Zed Attack Proxy (ZAP)' is displayed, followed by the text 'by CheckmarX'. A subtext describes it as 'The world's most widely used web app scanner. Free and open source. A community based GitHub Top 1000 project that anyone can contribute to.' To the right of this text is a cartoon illustration of a blue shield with a checkmark, surrounded by a network of lines and nodes. At the bottom of the main content area are two buttons: 'Quick Start Guide' and 'Download Now'.

Seleccionamos la partición en donde queramos descargar, después que se haya descargado, la instalación solo darle next, next hasta que se finalice.

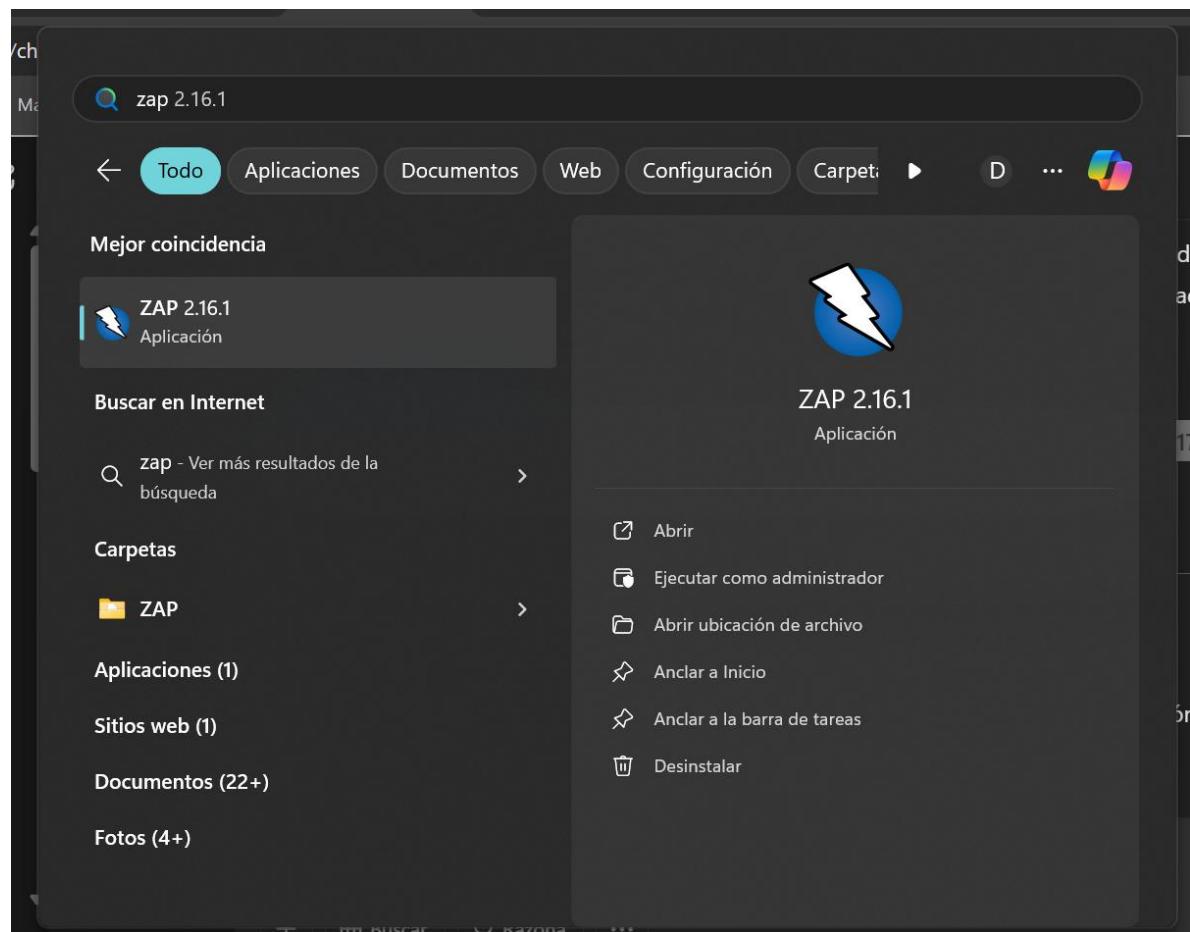
The screenshot shows the download page for ZAP 2.16.1. The top section has a large blue header with the text 'Download ZAP'. Below this, there are two bullet points:

- Checksums for all of the ZAP downloads are maintained on the [2.16.1 Release Page](#) and in the relevant [version files](#).
- As with all software we strongly recommend that ZAP is only installed and used on operating systems and JREs that are fully patched and actively maintained.

Below these instructions, the version 'ZAP 2.16.1' is displayed. Underneath it, there are three download links:

- Windows (64) Installer** (234 MB) with a 'Download' button.
- Windows (32) Installer** (234 MB) with a 'Download' button.
- Linux Installer** (226 MB) with a 'Download' button.

Verificamos la instalación en Windows.

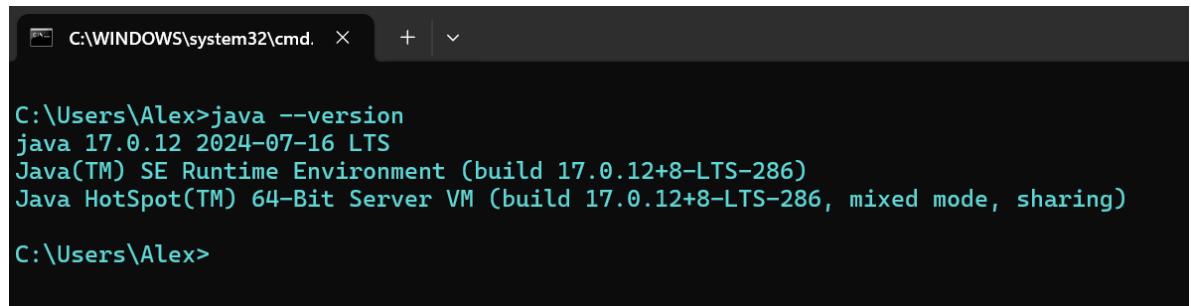


Durante la instalación, fue necesario contar con Java Runtime Environment (JRE) versión 17 o superior. Se instaló Java JDK 17, lo cual permitió ejecutar el instalador de ZAP sin inconvenientes.

Sitio oficial: <https://www.oracle.com/java/technologies/javase/jdk17-archive-downloads.html>

ORACLE			
Products	Industries	Resources	Customers
Partners	Developers	Company	
Windows x64 Compressed Archive	172.87 MB	https://download.oracle.com/java/17/archive/jdk-17.0.12_windows-x64_bin.zip (sha256)	
Windows x64 Installer	153.92 MB	https://download.oracle.com/java/17/archive/jdk-17.0.12_windows-x64_bin.exe (sha256)	
Windows x64 MSI Installer	152.67 MB	https://download.oracle.com/java/17/archive/jdk-17.0.12_windows-x64_bin.msi (sha256)	

Verificar la instalación de java, abrimos la terminal y escribimos java –version.

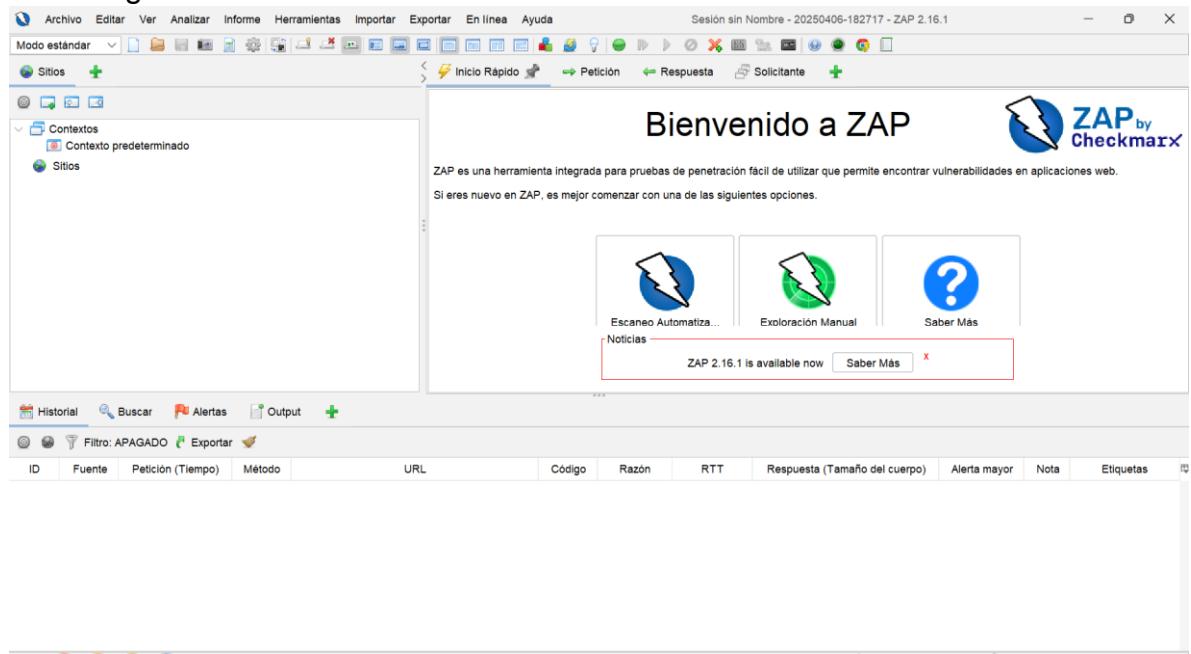


```
C:\Users\Alex>java --version
java 17.0.12 2024-07-16 LTS
Java(TM) SE Runtime Environment (build 17.0.12+8-LTS-286)
Java HotSpot(TM) 64-Bit Server VM (build 17.0.12+8-LTS-286, mixed mode, sharing)

C:\Users\Alex>
```

- **Pruebas empleando la herramienta.**

Interfaz gráfica de la herramienta



Tiene dos opciones de escaneo uno que es automática que es lo que se va a utilizar y el otro que es manual.

Bienvenido a ZAP



ZAP es una herramienta integrada para pruebas de penetración fácil de utilizar que permite encontrar vulnerabilidades en aplicaciones web.

Si eres nuevo en ZAP, es mejor comenzar con una de las siguientes opciones.



- **Identificación del tipo de pruebas que se realizó la herramienta.**

Para empezar el escaneo de la aplicación web, le damos clic en escaneo automático, abre esta ventana en donde solicita la URL.

The screenshot shows the "Escaneo Automatizado" (Automated Scan) configuration window. It includes fields for entering the URL (http://), selecting a traditional spider (checked), choosing an AJAX spider (Modern, Firefox Headless), and starting the attack (Atacar button). The progress bar at the bottom indicates "No iniciado" (Not started).

Escaneo Automatizado

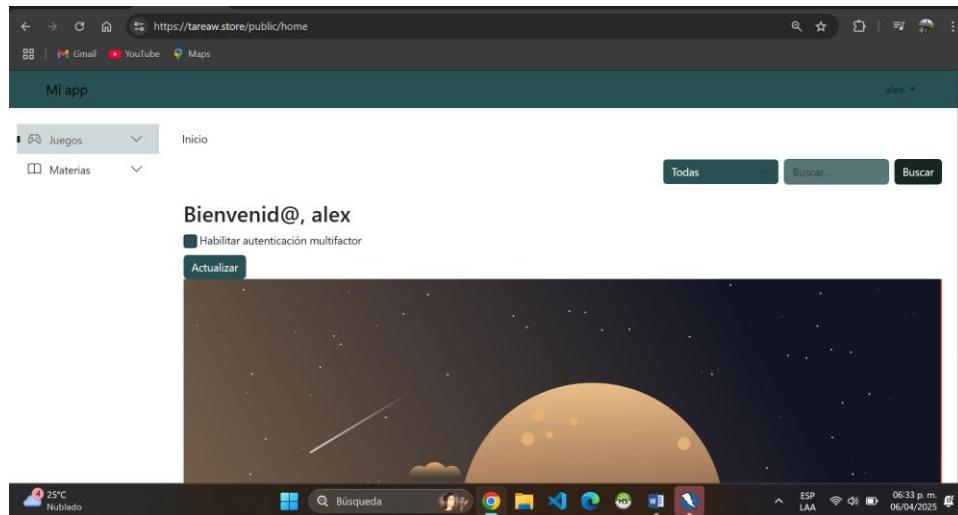
URL a atacar: http://

Usar el spider tradicional:

Usar el spider ajax: Si es Moderno con Firefox Headless

Progreso: No iniciado

Agarramos la URL de esta aplicación alojada en un hosting que fue desarrollado por nosotros mismos para una actividad en clase.



Asignamos la URL y le damos clic en atacar.

Escaneo Automatizado

ZAP by Checkmarx

Esta pantalla le permite iniciar un escaneo automático contra una aplicación: simplemente ingrese su URL a continuación y presione 'Atacar'.

Tenga en cuenta que solo debe atacar aplicaciones para las cuales ha recibido previamente una clara autorización.

URL a atacar:

Usar el spider tradicional:

Usar el spider ajax: con

Progreso: No iniciado

Empieza el escaneo en donde muestra opciones como la petición, método, URL, tamaño en bytes y respuestas etc..

ID	Petición (Tiempo)	Marca de tiempo Respuesta	Método	URL	Código	Razón	RTT	Tamaño de la Cabecera de Respuesta	Respuesta (Tamaño del cuerpo)
1.101	06/04/25 18:43:18	06/04/25 18:43:18	POST	https://tareaw.store/public/register	302	Found	176millseg...	1.335bytes	390bytes
1.102	06/04/25 18:43:18	06/04/25 18:43:18	POST	https://tareaw.store/public/login	302	Found	161millseg...	1.332bytes	378bytes
1.103	06/04/25 18:43:18	06/04/25 18:43:18	POST	https://tareaw.store/public/password/security	200	OK	134millseg...	1.256bytes	6.126bytes
1.105	06/04/25 18:43:18	06/04/25 18:43:18	POST	https://tareaw.store/public/password/security	302	Found	132millseg...	1.344bytes	426bytes
1.106	06/04/25 18:43:18	06/04/25 18:43:18	POST	https://tareaw.store/public/login	200	OK	158millseg...	1.256bytes	6.509bytes
1.108	06/04/25 18:43:18	06/04/25 18:43:18	POST	https://tareaw.store/public/login	302	Found	137millseg...	1.332bytes	378bytes
1.109	06/04/25 18:43:18	06/04/25 18:43:18	POST	https://tareaw.store/public/login	200	OK	157millseg...	1.256bytes	6.658bytes
1.111	06/04/25 18:43:18	06/04/25 18:43:18	POST	https://tareaw.store/public/password/security	200	OK	138millseg...	1.256bytes	6.126bytes
1.113	06/04/25 18:43:18	06/04/25 18:43:18	POST	https://tareaw.store/public/password/security	302	Found	212millseg...	1.344bytes	426bytes
1.114	06/04/25 18:43:18	06/04/25 18:43:18	POST	https://tareaw.store/public/login	200	OK	180millseg...	1.256bytes	6.724bytes
1.116	06/04/25 18:43:18	06/04/25 18:43:19	POST	https://tareaw.store/public/register	302	Found	309millseg...	1.335bytes	390bytes
1.117	06/04/25 18:43:18	06/04/25 18:43:19	POST	https://tareaw.store/public/login	302	Found	144millseg...	1.332bytes	378bytes
1.118	06/04/25 18:43:18	06/04/25 18:43:19	POST	https://tareaw.store/public/password/security	200	OK	149millseg...	1.256bytes	6.126bytes
1.120	06/04/25 18:43:19	06/04/25 18:43:19	POST	https://tareaw.store/public/password/security	302	Found	217millseg...	1.344bytes	426bytes
1.121	06/04/25 18:43:19	06/04/25 18:43:19	POST	https://tareaw.store/public/login	200	OK	278millseg...	1.256bytes	6.724bytes
1.122	06/04/25 18:43:19	06/04/25 18:43:20	POST	https://tareaw.store/public/login	200	OK	932millseg...	1.256bytes	6.724bytes

Termina el escaneo con el ID 9,578 es decir que escaneo 9,578 veces por todo en la aplicación.

ID	Petición (Tiempo)	Marca de tiempo Respuesta	Método	URL	Código	Razón	RTT	Tamaño de la Cabecera de Respuesta	Respuesta (Tamaño del cuerpo)
9.557	06/04/25 19:18:36	06/04/25 19:18:36	POST	https://tareaw.store/public/send	200	OK	154millseg...	1.256bytes	5.513bytes
9.559	06/04/25 19:18:36	06/04/25 19:18:36	POST	https://tareaw.store/public/password/email	200	OK	122millseg...	1.256bytes	6.274bytes
9.561	06/04/25 19:18:36	06/04/25 19:18:37	POST	https://tareaw.store/public/login	419	proxy rea...	619millseg...	835bytes	6.609bytes
9.562	06/04/25 19:18:37	06/04/25 19:18:37	POST	https://tareaw.store/public/login	419	proxy rea...	144millseg...	835bytes	6.609bytes
9.563	06/04/25 19:18:36	06/04/25 19:18:36	POST	https://tareaw.store/public/login	200	OK	123millseg...	1.256bytes	6.509bytes
9.565	06/04/25 19:18:36	06/04/25 19:18:37	POST	https://tareaw.store/public/password/email	200	OK	155millseg...	1.256bytes	6.274bytes
9.567	06/04/25 19:18:37	06/04/25 19:18:37	POST	https://tareaw.store/public/login	419	proxy rea...	130millseg...	835bytes	6.609bytes
9.568	06/04/25 19:18:37	06/04/25 19:18:37	POST	https://tareaw.store/public/password/email	200	OK	129millseg...	1.256bytes	5.978bytes
9.569	06/04/25 19:18:36	06/04/25 19:18:37	POST	https://tareaw.store/public/login	419	proxy rea...	1segundo...	835bytes	6.609bytes
9.571	06/04/25 19:18:37	06/04/25 19:18:38	POST	https://tareaw.store/public/sms/send	200	OK	134millseg...	1.256bytes	5.513bytes
9.573	06/04/25 19:18:37	06/04/25 19:18:37	POST	https://tareaw.store/public/register	200	OK	654millseg...	1.257bytes	10.698bytes
9.575	06/04/25 19:18:38	06/04/25 19:18:39	GET	https://tareaw.store/public/password	404	Not Found	1.16seg...	348bytes	6.603bytes
9.576	06/04/25 19:18:39	06/04/25 19:18:39	GET	https://tareaw.store/public/sms	404	Not Found	379millseg...	348bytes	6.603bytes
9.577	06/04/25 19:18:39	06/04/25 19:18:40	GET	https://tareaw.store/public/password	404	Not Found	337millseg...	348bytes	6.603bytes
9.578	06/04/25 19:18:40	06/04/25 19:18:40	GET	https://tareaw.store/public/sms	404	Not Found	426millseg...	348bytes	6.603bytes

- **Identificación de los elementos que evalúa la herramienta.**

Durante el escaneo, OWASP ZAP evaluó diferentes elementos clave de la aplicación, tales como.

Formularios y campos de entrada: Para detectar inyecciones, XSS, o ausencia de validaciones.

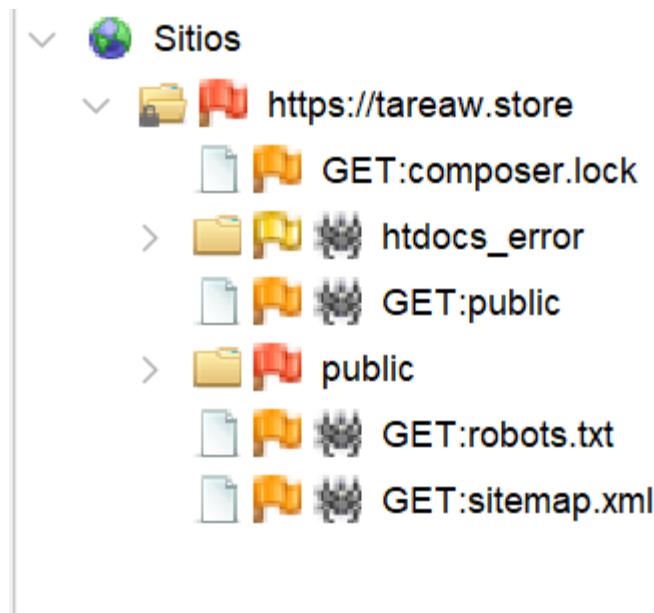
Cookies y encabezados HTTP: Para validar si están configuradas con atributos de seguridad como HttpOnly, Secure, etc.

Archivos JavaScript, HTML y recursos externos: Para detectar enlaces inseguros o vulnerabilidades en librerías.

Parámetros en URLs (GET y POST): Para encontrar puntos donde puedan injectarse datos maliciosos.

Endpoints de la aplicación: Identificando los puntos de entrada del sistema como formularios de login, registros, APIs, etc.

En las banderas rojas son alertas con alta prioridad, el color naranja son alertas con media prioridad y el amarillo son alertas con baja prioridad.



También se muestra la petición que hizo la herramienta, mostrando información como la referencia, cookie, URI entre otros.

```
⚡ Inicio Rápido ➔ Petición ➜ Respuesta 📄 Solicitante +  
Cabecera: Vista Raw Cuerpo: Vista Raw |    
POST https://tareaw.store/public/password/security HTTP/1.1  
host: tareaw.store  
user-agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/131.0.0.0  
Safari/537.36  
pragma: no-cache  
cache-control: no-cache  
content-type: application/x-www-form-urlencoded  
referer: https://tareaw.store/public/password/security  
content-length: 139  
Cookie: XSRF-TOKEN=  
eyJpdiI6Imx5MmR00Wl5ZUSSXJNUEhNNEdpSlE9PSIsInZhbHVLIjoieY1VNaWlyODcvQld0R3hZY1V0VEJK0GkraWJqa3M3RmVXK1k5UC93Y0I  
pGaHNMbFvPMFFpSkg4b1pkclBmWTBjSz15TG14VEFoUVNKcFVyMC9MZFrRMEQ3RVzXQ29aSVFzMzJRCWhxNLBENnB2a2MvRnBYSlhEMjNrazzJJc  
LCJtYWMi0iJizGVimTk5yWRkMTg30Tk2MGQyMjQ1NjY4ZmVmYjYz0DLiN2Q1ZjUzZDE4MDkwYzMwYzgxZWm3MGQ2MWFkNjFiiwidGFnIjoIn  
0%3D; laravel_session=  
eyJpdiI6IkluQnJ5ZdpWx2UE82UlllWC9jWxC9PSIsInZhbHVLIjoieEJHeWZleE95Mmd0Zmg3K09PWGE4UWtSaVlwV1QvdzRDSFpzWkJaWH  
hpTFp1c29snItIajl3QjZRak5rcFhkVi92aE1vZ3NFTzBTNFVPNjF4bEs3ekR1Wll5L0c1NLJrSUvobmdsbmF0b0txYmlPdkZzK2pEbXIxaXlxI  
LCJtYWMi0iI2MDZiMGIyY2Q2ZmFizGM1NTBjYTQ5MzkzNjg3N2fLMThjMGViMWIyZtg2NmNlyzQ0YTI3NzQ3YzMx0TM2NGQ0IiwidGFnIjoIn  
oen  
_token=case+randomblob%28100000%29+when+not+null+then+1+else+1+end+&email=zaproxy%40example.com&  
security_answer_1=ZAP&security_answer_2=ZAP
```

La respuesta de la petición que se hizo anteriormente.

```

HTTP/1.1 419 proxy reauthentication required
Connection: Keep-Alive
Keep-Alive: timeout=5, max=100
x-powered-by: PHP/8.2.27
cache-control: no-cache, private
content-type: text/html; charset=UTF-8
set-cookie: laravel_session=eyJpdiI6IlJuY2RlajZ1V215bkxPWE1iWHLNLwLE9PSIsInZhbkHVLijoiS1VaM3YrTWI0d3lBOUpoSEJTS0I0VFhtZEUyeUd6T3I2ZW1rbVzvTG/Y4Ykg5dTfxVUzaVC9DNTdweWtTV2hKMEJibmtwS2s1NVNiYzlPVXBsy2RuSELQaGp0eXVL2ZC0dk0U2VqMmRXRkkzYnYwNkl6SmRtUVRPQjVhLCJtYWMi0i2ZTNhYWRhZDk2NWZkODE3NDQxZGFmMzI0zM3NjdjMzNiNTJiZGM0NWE4NDmNjNkNDiZNzA3MGQ1NWmxNTNhIwidGFniJoIn0%3D; expires=Mon, 07 Apr 2025 02:56:53 GMT; Max-Age=7200; path=/; secure; samesite=lax
content-length: 6609
date: Mon, 07 Apr 2025 00:56:53 GMT

<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <title>Page Expired</title>
  
```

Identificación de URLs Vulnerables, mostrando el proceso y el tipo de método.

Procesado	Método	URI
●	GET	https://tareaw.store/public/login
●	GET	https://tareaw.store/robots.txt
●	GET	https://tareaw.store/sitemap.xml
●	GET	https://tareaw.store/htdocs_error/style.css
●	GET	https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css
●	GET	https://fonts.googleapis.com/css?family=Open+Sans:300,300i,400,400i,600,600i
●	GET	https://tareaw.store/htdocs_error/something-lost.png
●	GET	https://tareaw.store/public
●	GET	https://tareaw.store/public/register
●	GET	https://tareaw.store/public/password/reset
●	GET	https://fonts.bunny.net/
●	GET	https://fonts.bunny.net/css?family=Nunito
●	GET	https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha1/dist/css/bootstrap.min.css
●	GET	http://[::1]:5173/resources/sass/app.scss
●	GET	http://[::1]:5173/@vite/client
●	GET	http://[::1]:5173/resources/sass/app.js

Alertas: 1 | 8 | 6 | 5 | Proxy Principal: localhost:8080 | Current:

Historial completo del escaneo, es decir que muestra todas las opciones que se realizó y que se pueda visualizar en uno solo.

ID	Fuente	Petición (Tiempo)	Método	URL	Código	Razón	RTT	Respuesta (Tamaño del cuerpo)	Alerta mayor	Nota	Etiquetas
1	Proxy	06/04/25 18:37:52	GET	https://tareaw.store/public/login	200	OK	1.12segundos	6.509bytes	Medio		AntiCSRF, Form, Pa...

En el cuadro no se refleja todo, pero esto sería todas las opciones que se podrá visualizar en el historial.

- ID
 - Fuente
 - Petición (Tiempo)
 - Marca de tiempo Respuesta
 - Método
 - URL
 - Nombre de Host
 - Ruta y Consulta
 - Código
 - Razón
 - RTT
 - Tamaño de la Cabecera de Pregunta
 - Tamaño Cuerpo de Petición
 - Tamaño de la Cabecera de Respuesta
 - Respuesta (Tamaño del cuerpo)
 - Alerta mayor
 - Nota
 - Etiquetas
-
- Desplazamiento Horizontal
 - Compactar Todas las Columnas
 - Compactar la Columna Seleccionada
-
- Desplazamiento automático
 -  Exportar
 - Restablecer Columnas
 - Mostrar solo los bytes

- **Vulnerabilidades y tipo que localizó la herramienta.**

Durante el escaneo realizado con OWASP ZAP, se detectaron las siguientes vulnerabilidades.

En total fueron 20 alertas o vulnerabilidades, lo que representa riesgos potenciales para los usuarios de la aplicación y pueden ser explotadas por atacantes si no se corrigen a tiempo.

The screenshot shows the OWASP ZAP interface with the 'Alertas' tab selected. A tree view displays 20 alerts under the 'Alertas (20)' folder. The alerts are categorized by type, with some expanded to show specific details. The categories include:

- Inyección SQL - SQLite (2)
- CSP: Directiva Wildcard (12)
- CSP: Failure to Define Directive with No Fallback (12)
- CSP: script-src unsafe-inline (12)
- CSP: style-src unsafe-inline (12)
- Cabecera Content Security Policy (CSP) no configurada (2)
- Falta de cabecera Anti-Clickjacking (5)
- Hidden File Found (Archivo Oculto Encontrado)
- Las Páginas Seguras Incluyen Contenido Mixto (Incluyendo So
- Cookie Sin Flag HttpOnly (11)
- El servidor divulga información mediante un campo(s) de enca
- Falta encabezado X-Content-Type-Options (7)
- Gran redirección detectada (posible fuga de información confid
- Inclusión de archivos fuente JavaScript entre dominios (15)
- Strict-Transport-Security Header No Establecido (9)
- Aplicación Web Moderna (2)
- Petición de Autenticación Identificada

At the bottom, there are navigation buttons for 'Alertas' and numerical links (1, 8, 6, 5), along with a 'Proxy Principal: localhost:8080' status bar.

En esta captura se observa una alerta generada por OWASP ZAP correspondiente a una vulnerabilidad de tipo SQL Injection, específicamente en una petición con método POST. Este tipo de vulnerabilidad permite a un atacante injectar sentencias SQL maliciosas en campos de entrada de la aplicación, las cuales son ejecutadas por el servidor sin validación adecuada.

This screenshot provides a detailed look at the 'Alertas (20)' section, specifically focusing on the 'Inyección SQL - SQLite (2)' category. Two specific POST requests are highlighted in blue, indicating they are the focus of the analysis:

- POST: https://tareaw.store/public/password/security
- POST: https://tareaw.store/public/password/security

Al hacer doble clic en la alerta, se abrió una ventana con la información detallada de la petición y la respuesta del servidor. Esta ventana es clave para entender el contexto de la vulnerabilidad, lo más importante es que brinda una solución.

 Editar Alerta X

Inyección SQL - SQLite

URL: `https://tareaw.store/public/password/security`

Riesgo: High

Confianza: Medium

Parámetro: `_token`

Ataque: `case randomblob(100000) when not null then 1 else 1 end`

Evidencia: `wop3Ndg6HcQIHG9AOF22oon49EMrXT] tardó [125] milisegundos.`

CWE ID: 89

WASC ID: 19

Descripción:
Inyección SQL puede ser posible.

Otra información:
El tiempo de consulta es controlable mediante el valor del parámetro [case randomblob(100000) when not null then 1 else 1 end], que hizo que la petición tardara [388] milisegundos, el valor del parámetro [case randomblob(1000000) when not null then 1 else 1 end 1, que ha provocado

Solución:
No confie en los datos de entrada del lado del cliente, incluso si existe una validación del lado del cliente.
Como norma general, escriba la verificación de los datos en el lado del servidor.

Referencias:
https://cheatsheetseries.owasp.org/cheatsheets/SQL_Injection_Prevention_Cheat_Sheet.html

Etiquetas de Alerta:

Cancelar Guardar

Ejemplos con alertas de prioridad media, básicamente se repite el proceso anterior para poder visualizar la información o la posible solución que nos brinda la herramienta.



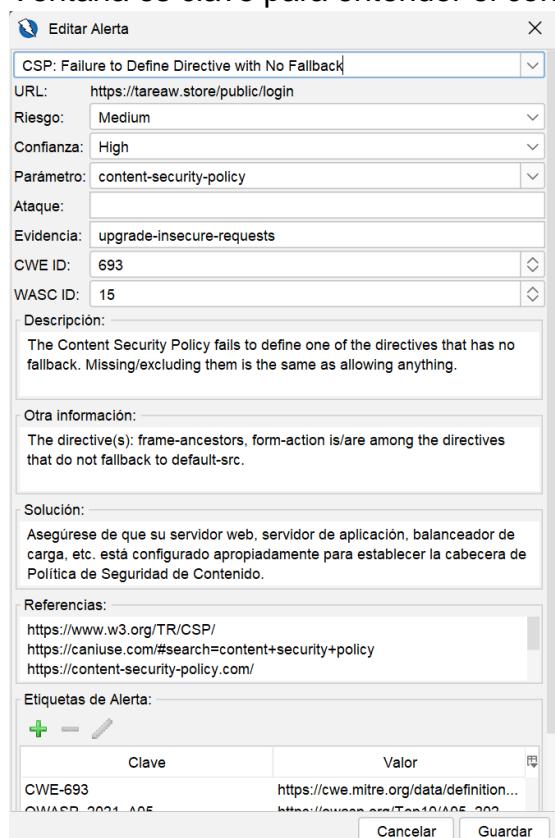
The screenshot shows a tree view of alerts. The 'Alertas (20)' folder is expanded, revealing three main types of vulnerabilities:

- Inyección SQL - SQLite (2)
- CSP: Directiva Wildcard (12)
- CSP: Failure to Define Directive with No Fallback (12)

Under the third category, there are five specific GET requests listed:

- GET: https://tareaw.store/public
- GET: https://tareaw.store/public/
- GET: https://tareaw.store/public/login (This item is highlighted with a gray background)
- GET: https://tareaw.store/public/password/reset
- GET: https://tareaw.store/public/password/security

Ventana es clave para entender el contexto de la vulnerabilidad.



The 'Editar Alerta' (Edit Alert) window is open, showing the details for the selected alert:

Alerta: CSP: Failure to Define Directive with No Fallback

Datos:

- URL: https://tareaw.store/public/login
- Riesgo: Medium
- Confianza: High
- Parámetro: content-security-policy
- Ataque:
- Evidencia: upgrade-insecure-requests
- CWE ID: 693
- WASC ID: 15

Descripción:

The Content Security Policy fails to define one of the directives that has no fallback. Missing/excluding them is the same as allowing anything.

Otra información:

The directive(s): frame-ancestors, form-action is/are among the directives that do not fallback to default-src.

Solución:

Asegúrese de que su servidor web, servidor de aplicación, balanceador de carga, etc. esté configurado apropiadamente para establecer la cabecera de Política de Seguridad de Contenido.

Referencias:

<https://www.w3.org/TR/CSP/>
<https://caniuse.com/#search=content+security+policy>
<https://content-security-policy.com/>

Etiquetas de Alerta:

Clave	Valor
CWE-693	https://cwe.mitre.org/data/definition/693.html
OWASP_2021_A05	https://owasp.org/Top10/A05_2021

Botones: Cancelar, Guardar

Ejemplo con una alerta prioridad baja

- >  Cookie Sin Flag HttpOnly (11)
- <  El servidor divulga información mediante un campo(s) de encabezado
-  GET: https://tareaw.store/public/
 -  GET: https://tareaw.store/public/login
 -  GET: https://tareaw.store/public/password/reset
 -  GET: https://tareaw.store/public/password/security
 -  GET: https://tareaw.store/public/register
 -  GET: https://tareaw.store/public/sms/form
 -  POST: https://tareaw.store/public/login
 -  POST: https://tareaw.store/public/password/email

Ventana es clave para entender el contexto de la vulnerabilidad.

 Editar Alerta X

mediante un campo(s) de encabezado de respuesta HTTP ""X-Powered-By""

URL: https://tareaw.store/public/

Riesgo: Low

Confianza: Medium

Parámetro:

Ataque:

Evidencia: x-powered-by: PHP/8.2.27

CWE ID: 497

WASC ID: 13

Descripción:
El servidor de la web/aplicación está divulgando información mediante uno o más encabezados de respuesta HTTP ""X-Powered-By"". El acceso a tal información podría facilitarle a los atacantes la identificación de otros marcos/componentes de los que su aplicación web depende y las

Otra información:

Solución:
Asegúrese de que su servidor web, servidor de aplicaciones, balanceador de carga, etc. está configurado para suprimir las cabeceras "X-Powered-By".

Referencias:
https://owasp.org/www-project-web-security-testing-guide/v42/4-Web_Application_Security_Testing/01-Information_Gathering/08-Fingerprint_Web_Application_Framework
<https://www.troyhunt.com/2012/02/shhh-dont-let-your-response-headers.html>

Etiquetas de Alerta:

Cancelar Guardar

- **Interpretación de la prueba realizada con la herramienta.**

The screenshot shows the Araña Spider tool's interface. The main pane displays a list of security findings for a target website. One finding, 'GET: https://tareaw.store/public/register', is highlighted with a blue background. The right pane contains detailed information about this finding, including 'Otra información' (Other information) with code snippets, 'Solución' (Solution), 'Referencias' (References), and 'Etiquetas de Alerta' (Alert tags). A table lists alert tags with their corresponding URLs. At the bottom, there are status indicators for various metrics like 'Current Status' and 'CWE-311'.

Clave	Valor
OWASP_2021_A05	https://owasp.org/Top10/A05_2021-Security_Misconfigu...
OWASP_2017_A06	https://owasp.org/www-project-top-ten/2017/A6_2017-Se...
WSTG-v42-CRYP-03	https://owasp.org/www-project-web-security-testing-guid...
CVE-311	https://cve.mitre.org/data/definitions/311.html

Se detectó una vulnerabilidad de tipo Reflected XSS en el endpoint.

GET: <https://tareaw.store/public/register>

Esto significa que el servidor refleja parte del input del usuario en la respuesta HTML sin validar o sanearlo adecuadamente, lo que permite que un atacante inyecte código malicioso (como JavaScript).

Un atacante podría engañar a un usuario para que haga clic en un enlace malicioso, y sin saberlo, este ejecutará código JavaScript en su navegador. Esto podría:

- Robar cookies de sesión.
- Redirigir a sitios fraudulentos.
- Modificar la apariencia de la página.
- Realizar acciones en nombre del usuario si está autenticado.

Conclusión

Esta práctica permitió explorar y aplicar una herramienta profesional de análisis de seguridad web como lo es OWASP ZAP. A través del escaneo activo y pasivo, se identificaron vulnerabilidades reales en una aplicación web de prueba. Gracias a esta experiencia, se comprendieron los riesgos de seguridad más comunes como XSS, CSRF, y la mala configuración de cookies y encabezados HTTP.

También se reforzó el conocimiento sobre la importancia de proteger las aplicaciones web desde el desarrollo, aplicando buenas prácticas y realizando pruebas constantes de seguridad.