

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.

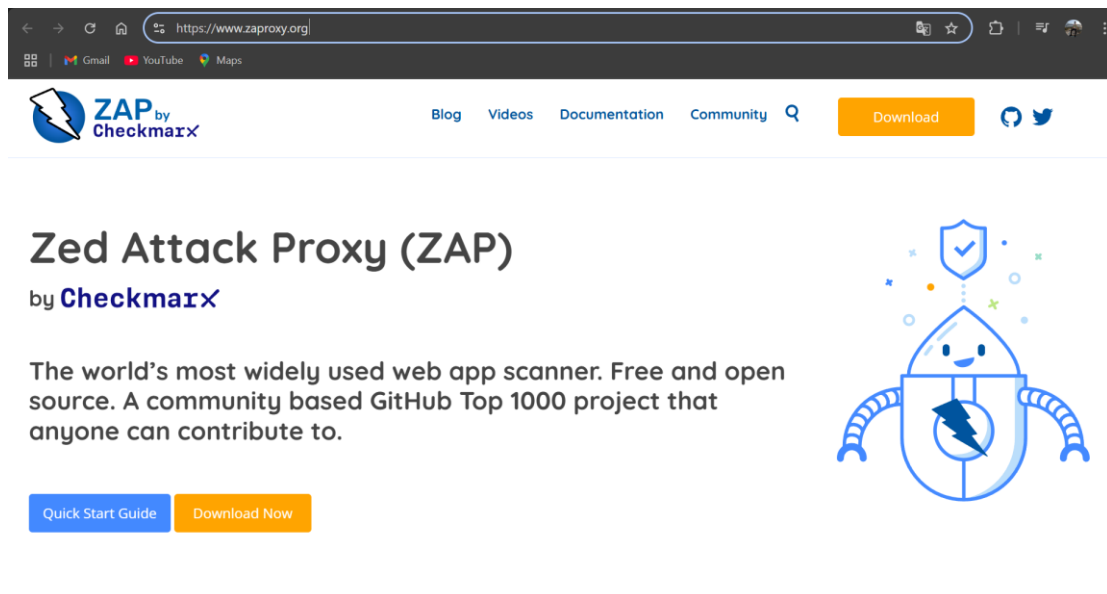
Herramienta Utilizada.

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/>



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

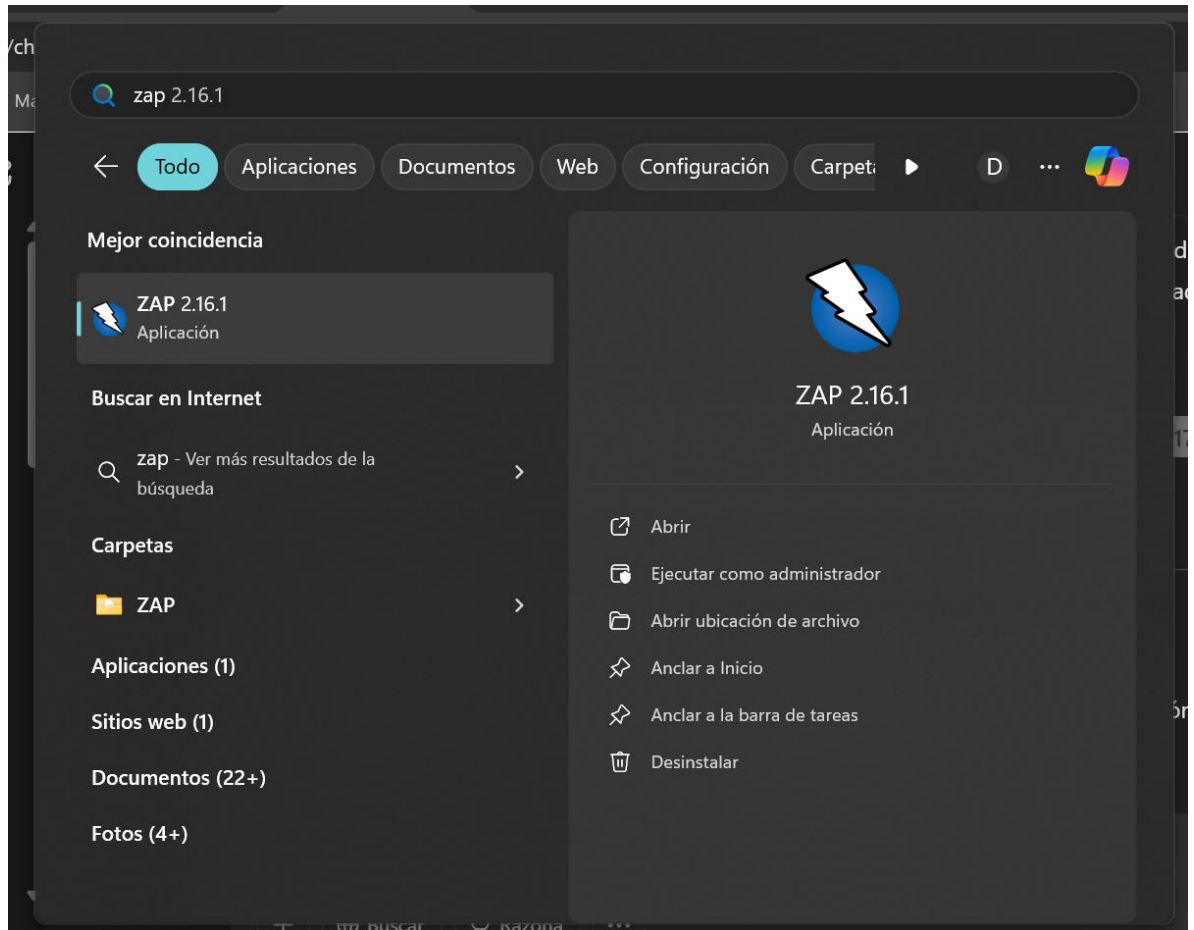
Download ZAP

- 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.

ZAP 2.16.1

Windows (64) Installer	234 MB	Download
Windows (32) Installer	234 MB	Download
Linux Installer	226 MB	Download

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

Q

🇺🇸

View Accounts

Contact Sales

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.

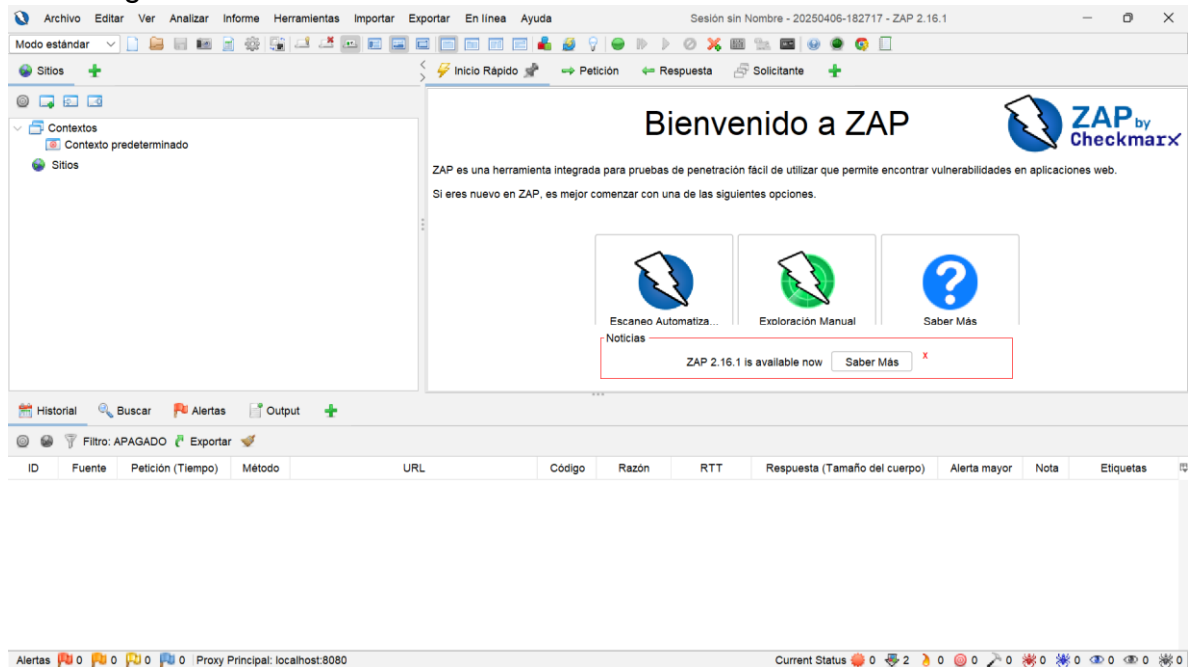
```
C:\WINDOWS\system32\cmd. X + v

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



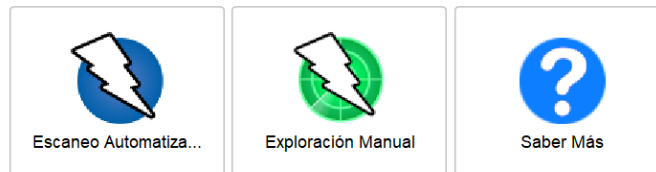
Tiene dos opciones de escaneo uno que es automática que es lo que se va a utilizar y el otro que es manual.

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.



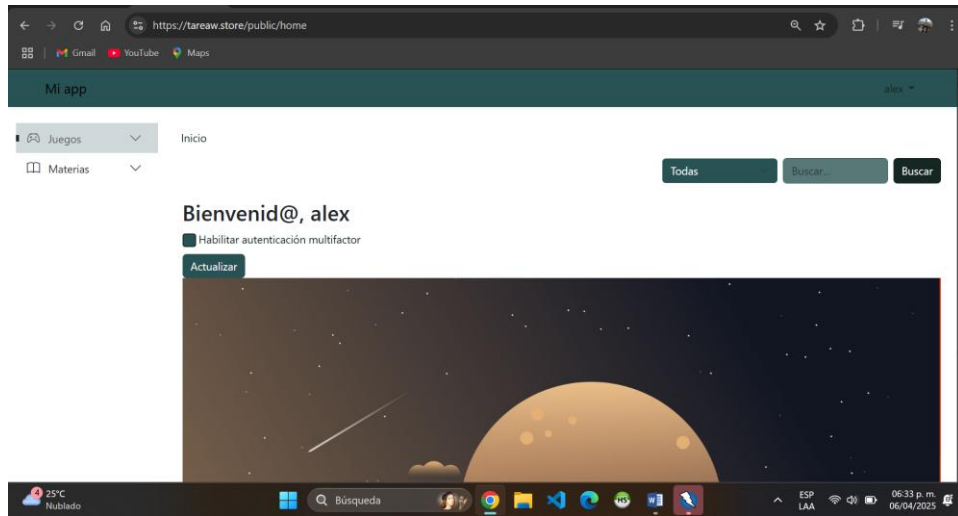
- **Identificación del tipo de pruebas que se realizó la herramienta.**

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 Scanning) window in ZAP. It features a title bar with a back button, a ZAP logo, and the title 'Escaneo Automatizado'. Below the title bar, there is a ZAP by Checkmarx logo. The main content area contains instructions: 'Esta pantalla le permite iniciar un escaneo automático contra una aplicación: simplemente ingrese su URL a continuación y presione 'Atacar'.' and 'Tenga en cuenta que solo debe atacar aplicaciones para las cuales ha recibido previamente una clara autorización.' Below the instructions, there is a form with the following fields and controls:

- URL a atacar:** A text input field containing 'http://', a dropdown arrow, and a 'Seleccionar...' button with a globe icon.
- Usar el spider tradicional:** A checkbox that is checked.
- Usar el spider ajax:** A dropdown menu with 'Sí es Moderno' selected, followed by the word 'con' and another dropdown menu with 'Firefox Headless' selected.
- Buttons:** Two buttons, 'Atacar' (with a lightning bolt icon) and 'Detener' (with a square icon).
- Progreso:** A label followed by the text '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



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:

 Seleccionar...

Usar el spider tradicional:


☒

Usar el spider ajax:

Si es Moderno

con

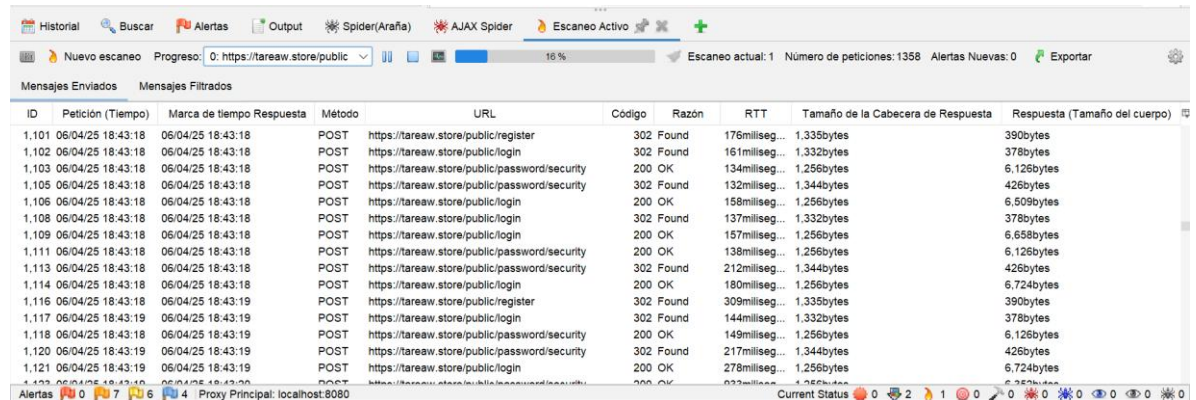
Firefox Headless

 Atacar

Progreso:

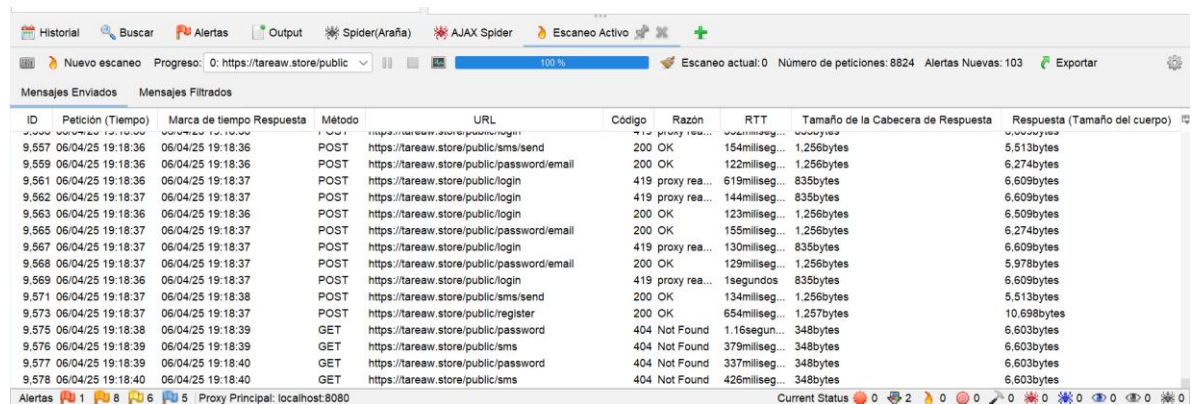
No iniciado

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	176miliseg...	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	161miliseg...	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	134miliseg...	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	132miliseg...	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	158miliseg...	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	137miliseg...	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	157miliseg...	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	138miliseg...	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	212miliseg...	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	180miliseg...	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	309miliseg...	1,335bytes	390bytes
1,117	06/04/25 18:43:19	06/04/25 18:43:19	POST	https://tareaw.store/public/login	302	Found	144miliseg...	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	149miliseg...	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	217miliseg...	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	278miliseg...	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/sms/send	200	OK	154miliseg...	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	122miliseg...	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...	619miliseg...	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...	144miliseg...	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	123miliseg...	1,256bytes	6,509bytes
9,565	06/04/25 19:18:37	06/04/25 19:18:37	POST	https://tareaw.store/public/password/email	200	OK	155miliseg...	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...	130miliseg...	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	129miliseg...	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...	1segundos	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	134miliseg...	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	654miliseg...	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	1.16segun...	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	379miliseg...	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	337miliseg...	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	426miliseg...	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.

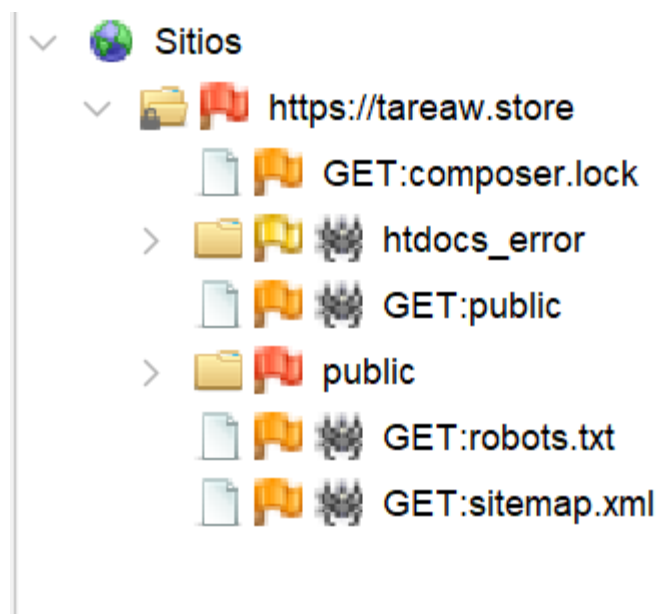
Formularios y campos de entrada: Para detectar inyecciones, XSS, o ausencia de validaciones.

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

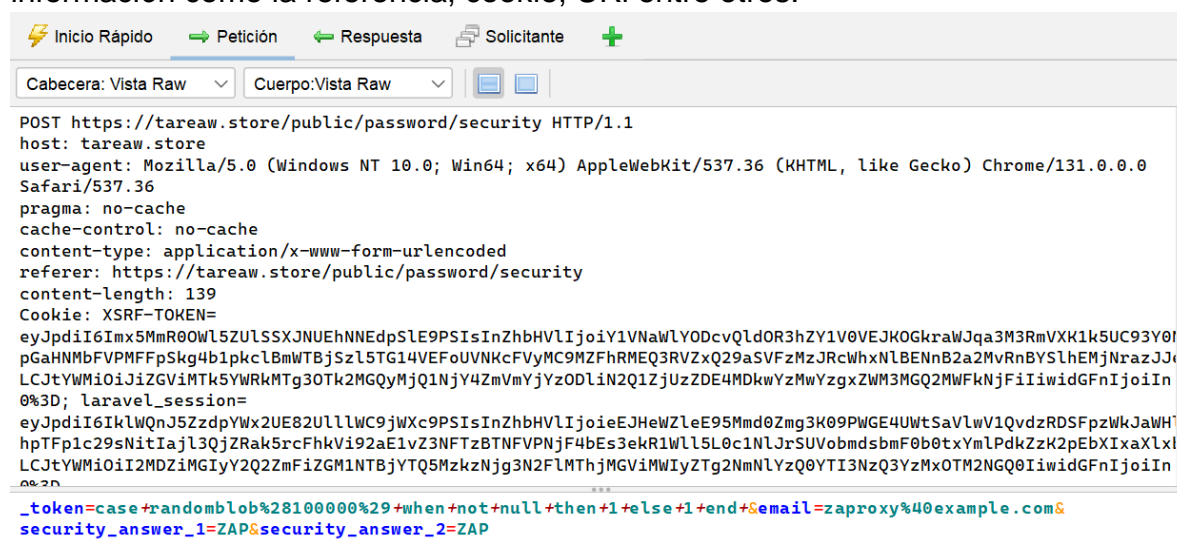
Archivos JavaScript, HTML y recursos externos: Para detectar enlaces inseguros o vulnerabilidades en librerías.

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, URL entre otros.



La respuesta de la petición que se hizo anteriormente.

Inicio Rápido

Petición

Respuesta

Solicitante

+

Cabecera: Vista Raw

Cuerpo: Vista Raw

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=eyJpdiI6IlJuY2RlajZ1V2I5bkcWpE1iWHNLWLE9PSIsInZhbnVlIjo1S1VaM3YrTWI0d3lB0UpoSEJTS0I0VFhtZEUyeUd6T3I2ZW1rbVZvTG; Y4Ykg5dTFxVUZaVC9DNTdweWtTV2hkMEJibmtwS2s1NVNiYzLPVXBSY2RuSElQaGp0eXVBL2ZC0Dk0U2VqMmRXRkkzYnYwNk16SmRtUVRPQjVhLCJtYWMiOiI2ZTNhYWRhZDk2NWZkODE3NDQxZGFmMzI0ZjM3NjdjMzNiNTJiZGM0NW4NDdmNjNkNDIzNzA3MGQ1NWxNTNhIiwidGFnIjo1In0%3D; expires=Mon, 07 Apr 2025 02:56:53 GMT; Max-Age=7200; path=/; secure; httponly; 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.

URLs vulnerables	Nodos ingresados	Mensajes
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/js/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 realizo y que se pueda visualizar en uno solo.

Historial

Buscar

Alertas

Output

Spider(Araña)

AJAX Spider

Escaneo Activo

+

Filtro: APAGADO

Exportar

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...

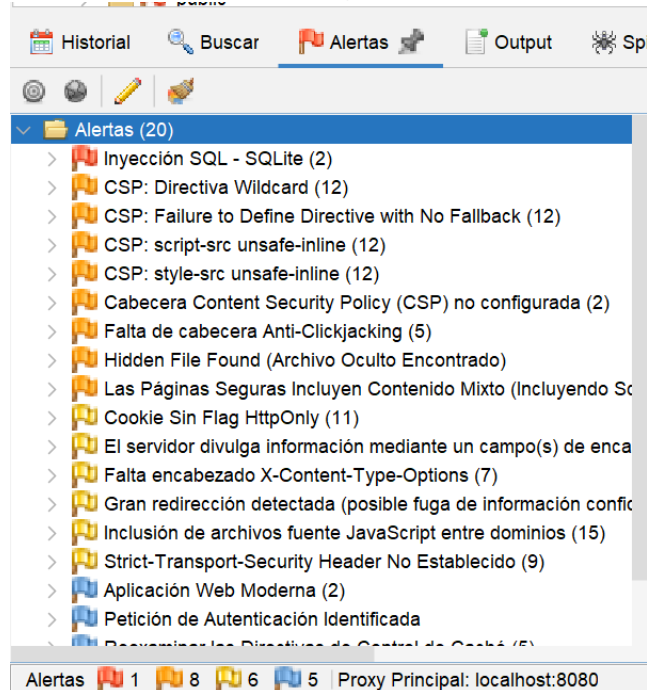
En el cuadro no se refleja todo, pero esto sería todas las opciones que se podrá visualizar en el historial.

✓ 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

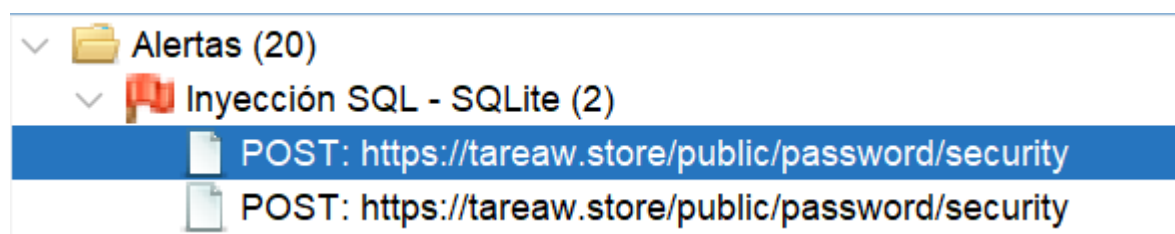
- **Vulnerabilidades y tipo que localizó la herramienta.**

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.



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 inyectar sentencias SQL maliciosas en campos de entrada de la aplicación, las cuales son ejecutadas por el servidor sin validación adecuada.



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 ✕

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(100000) when not null then 1 else 1 end], que ha provocado

Solución:

No confíe 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:

+

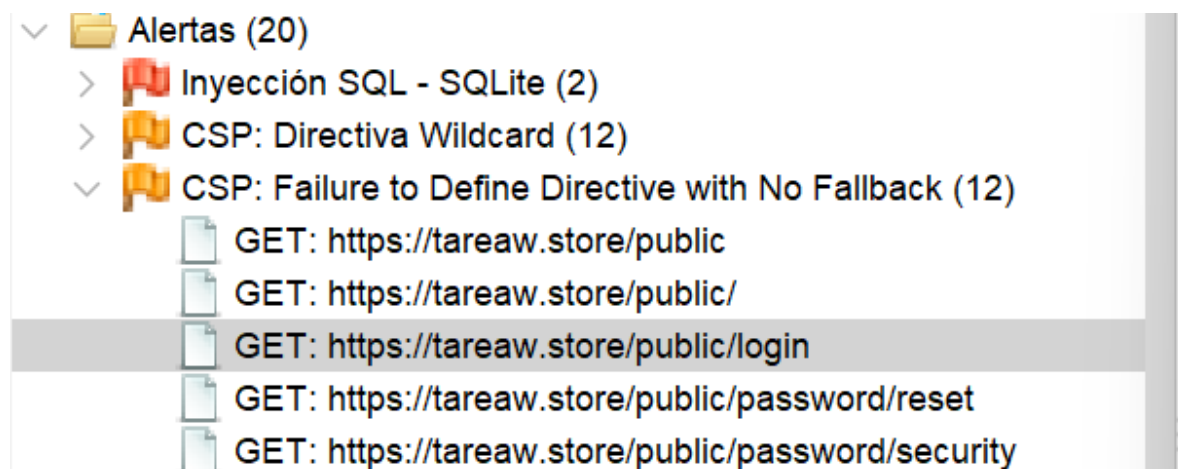
—



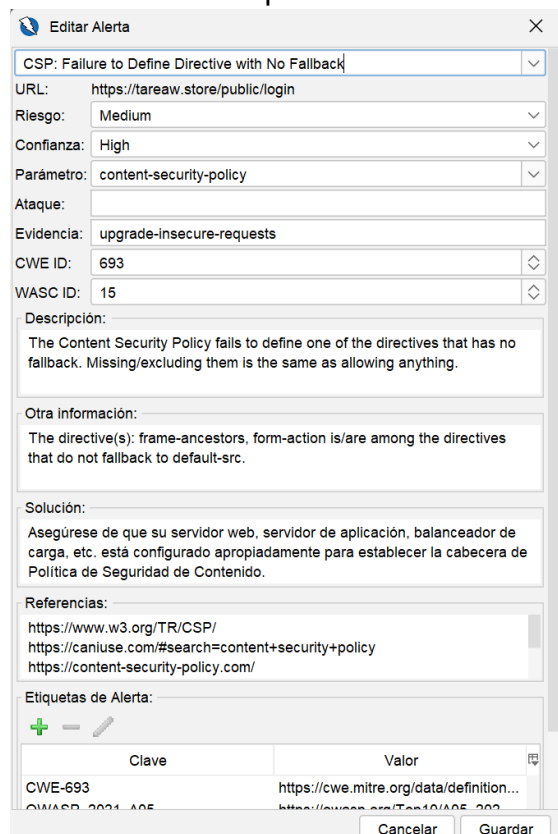
Cancelar

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.



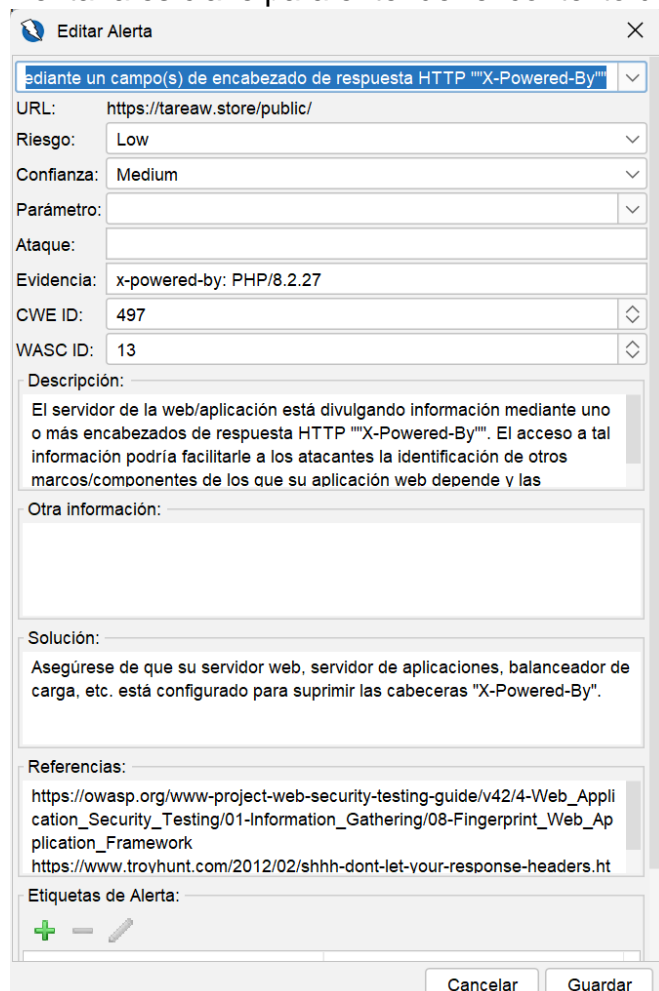
Ventana es clave para entender el contexto de la vulnerabilidad.



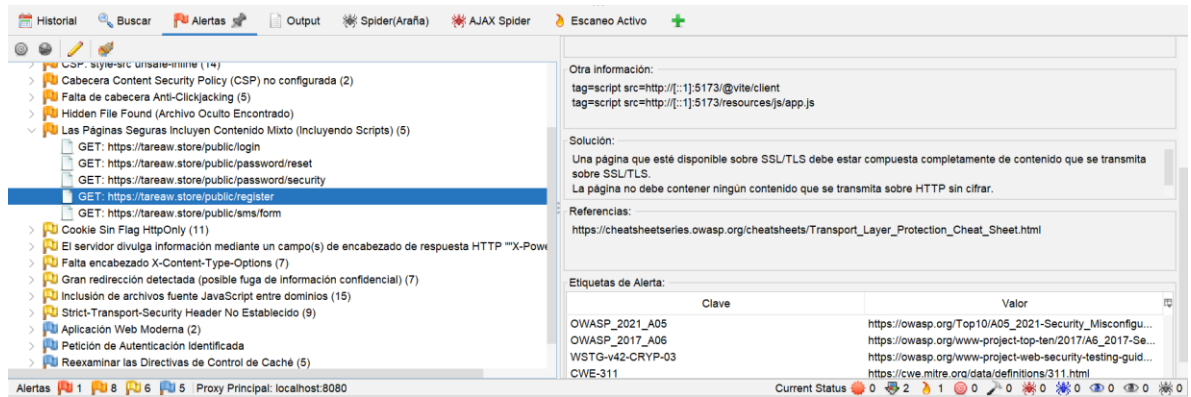
Ejemplo con una alerta prioridad baja



Ventana es clave para entender el contexto de la vulnerabilidad.



- Interpretación de la prueba realizada con la herramienta.



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 validarlo 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.