Héctor Ramírez López

Introducción

Este documento describe el método y los hallazgos de un estudio de seguridad, enfocado en las etapas de recopilación de información y búsqueda de debilidades, sobre tres activos web accesibles al público. La finalidad principal de esta práctica es realizar una evaluación utilizando herramientas comunes en el campo de la ciberseguridad para detectar la superficie de ataque, las tecnologías básicas y las posibles vulnerabilidades de seguridad.

Para llevar a cabo este análisis, se ha empleado un conjunto de herramientas especializadas que cubren distintas fases del reconocimiento:

- **WHOIS:** permite obtener información pública sobre el registro del dominio (propietario, entidad registradora, fechas de creación y expiración).
- **WhatWeb:** Herramienta diseñada para identificar las tecnologías con las que está construido un sitio web, como el servidor web, el sistema de gestión de contenidos, lenguajes de programación y librerías JavaScript.
- **WPScan:** Un escáner de vulnerabilidades específico para **WordPress**. Su uso es fundamental dado que WordPress es el CMS más utilizado a nivel global, y permite detectar versiones desactualizadas, *plugins* y *temas* vulnerables, así como los usuarios expuestos.
- **Nikto:** Un escáner de vulnerabilidades de aplicaciones web que comprueba la existencia de archivos peligrosos, configuraciones obsoletas o inseguras y otros problemas de seguridad conocidos.

Alcance

La selección de los tres sitios web para esta práctica no ha sido aleatoria. Se ha buscado que abarque diferentes sectores y arquitecturas tecnológicas, permitiendo así comparar distintos tipos de seguridad:

1. **Sector Público/Institucional:** https://aytocabanillas.org

Por qué se escogió: Este sitio representa a una entidad pública (un ayuntamiento). La seguridad en el sector público es vital, ya que este tipo de portales suelen gestionar información sensible de los ciudadanos y sirven como canal oficial de comunicación.

2. **Comercio Electrónico (E-commerce):** https://webrepuesto.com
   Por qué se escogió: Se trata de una tienda *online*. Este tipo de plataforma es un objetivo valioso para los ciberdelincuentes, ya que procesa transacciones financieras, maneja datos personales y de pago de clientes.

3. **Blog/Plataforma de Contenido:** https://devcraze.com
   Por qué se escogió: Este sitio tiene la categoría de blogs técnicos o plataformas de contenido especializado (para desarrolladores). Estos sitios buscan construir una comunidad y su reputación es clave.

https://aytocabanillas.org

Vulnerabilidades más importantes (resumen y prioridad)

Información del Dominio extraída con WHOIS

Dominio	Aytocabanillas.org
Registrador	IONOS SE
Fecha de creación	29 de julio de 2010
Última actualización	12 de septiembre de 2025
Fecha de expiración	29 de julio de 2026
Servidores DNS	• ns1075.ui-dns.org • ns1075.ui-dns.biz • ns1075.ui-dns.com • ns1075.ui-dns.de
DNSSEC	No firmado (Las respuestas dns no estan firmadas permite ataques de suplantación de dns)

Estado del dominio	clientTransferProhibited

Hallazgos de Nikto (servidor / HTTP)

1. **Falta de Strict-Transport-Security (HSTS)** — *Alta prioridad*
   Riesgo: ataques de downgrade/strip de HTTPS por redes no fiables.
2. **Falta X-Frame-Options (clickjacking)** — *Media-Alta*
   Riesgo: la web puede ser embebida y manipular usuarios (clickjacking).
3. **Falta X-Content-Type-Options: nosniff** — *Media*
   Riesgo: riesgos XSS por sniffing de MIME.
4. **Uso de `Content-Encoding: deflate` — posible exposición a BREACH** — *Alta (si se devuelven datos sensibles)*
   Riesgo: extracción de datos sensibles cuando se combina compresión HTTP y respuestas que contienen secretos.
5. **Apache MultiViews (mod_negotiation) activado** — *Media*
   Riesgo: facilita enumeración de archivos (fuerza/brute forcing de nombres).
6. **Divulgación del banner/stack (Apache → nginx)** — *Baja-Media*
   Riesgo: revelación de tecnología usada, ayuda a un atacante a elegir exploits.
7. **Cabeceras inusuales divulgando información operacional** (`x-ws-ratelimit-*`, `x-redirect-by`, `Link: wp-json`) — *Informativo, pero relevante*
   Riesgo: información que facilita planificación de ataques (límites, uso de WordPress, API pública).
8. **robots.txt contiene entradas** — *Baja-Media*
   Riesgo: lista rutas que el atacante inspeccionará.

Hallazgos de WPScan (plugins WordPress)

1. **GDPR Cookie Consent — Improper Access Controls (versión < 1.8.3)** — *Media-Alta*

   a. CVE reportado (CVE-2020-20633).

   b. Riesgo: usuarios no autorizados podrían acceder o modificar configuraciones del plugin o realizar operaciones que requieren privilegios.

2. **One Click Accessibility / Pojo Accessibility — Stored XSS (version < 3.2.0)** — *Alta*

    a. CVE vinculado (CVE-2025-32640).

    b. Riesgo: administrador puede almacenar payloads XSS que se ejecutan en navegadores de otros usuarios; puede conducir a comprometimiento de cuentas administrativas o persistencia de código malicioso.

3. **Usuarios encontrados —** baja

    Admin, mdiez, plopez, dbiblio

Análisis de fingerprint con Whatweb

Elemento identificado	Descripción
Servidor Apache	Ya detectado por nikto
Ubicación IP: Alemania	Útil para localización
Email expuesto en el HTML	Se puede usar para spam o phising
Uso de Google Analytics	Información no critica, pero indica rastreo externo
Versión de jQuery: 3.7.1	Aunque es moderna, conviene verificar si plugins cargan versiones diferentes más vulnerables.
WordPress versión 6.8.3	Indica que **WordPress está actualizado a la última versión**, lo cual es positivo. Sin embargo, que sea visible puede ayudar a ataques dirigidos si se descubre una vulnerabilidad 0-day.
Cabecera x-pingback habilitada	Indica que el **servicio XML-RPC está activo**, lo que puede permitir ataques como pingback DDoS o fuerza bruta. Si no se usa, se recomienda desactivarlo.
Cabecera X-UA-Compatible: ie=edge	Obsoleta, generalmente innecesaria en navegadores modernos. Se puede eliminar para reducir la exposición.
Cabeceras inusuales (x-ws-ratelimit-*, link)	También se han detectado por nikto. Se podrían eliminar si no se necesitan
Plugins y tecnologías detectadas	El hecho de que use WordPress y jQuery permite identificar todas las tecnologías

	que usa la web, y esa información puede ayudar a un atacante a planear ataques.

Relación con owasp 2024

Vulnerabilidad detectada	Categoría owasp	Explicación
GDPR Cookie Consent - Controles de acceso inadecuados (versión < 1.8.3)	Control de acceso roto	El complemento permite que usuarios no autorizados accedan o modifiquen configuraciones restringidas. Esto rompe los controles de acceso haciendo posible que se realicen acciones fuera del rol asignado
One Click Accessibility / Pojo Accessibility - XSS almacenado (versión < 3.2.0)	Inyección	permite que un atacante inyecte código JavaScript persistente en el sistema que se ejecuta en los navegadores de otros usuarios
Usuarios encontrados: admin, mdiez, plopez, dbiblio	Fallos en identificación y autenticación	La enumeración de usuarios expone información sensible sobre cuentas válidas esto facilita ataques de fuerza bruta

https://webrepuesto.com

Vulnerabilidades más importantes (resumen y prioridad)

Información del Dominio extraída con WHOIS

Dominio	Webrepuesto.com
Registrador	Cloudflare, Inc. (IANA ID:1910)
Fecha de creación	5 de diciembre de 2008
Última actualización	7 de diciembre de 2024
Fecha de expiración	5 de diciembre de 2025
Servidores DNS	• carol.ns.cloudflare.com • sean.ns.cloudflare.com
DNSSEC	No firmado (Las respuestas dns no estan firmadas permite ataques de suplantación de dns)
Estado del dominio	clientTransferProhibited

Hallazgos de Nikto (servidor / HTTP)

1. **Falta el encabezado X-Frame-Options** — Media
   Riesgo: Permite ataques de clickjacking
2. **Versión de PHP desactualizada: 7.2.34** — *Alta*
   Riesgo: PHP 7.2 está sin soporte desde 2020 → muchas vulnerabilidades conocidas.
3. **robots.txt con rutas sensibles** — *Media*
   Riesgo: Puede revelar contenido oculto.
4. **Exposición de endpoints REST API de WordPress** — Media
   Riesgo: permite enumeración de usuarios.

Hallazgos de WPScan (plugins WordPress)

1. ***Rehub Theme — Authenticated (Subscriber+) SQL Injection (versión < 19.6.2)*** ***—*** *Alta*

a. CVE reportado (CVE-2024-31233).

b. Riesgo: un usuario con cuenta de suscriptor podría enviar parámetros manipulados que terminan en consultas SQL, permitiendo leer o modificar datos de la base de datos

2. ***Rehub Theme — Authenticated (Editor+) Local File Inclusion (versión < 19.6.2) —*** *Alta*

    a. CVE vinculado (CVE-2024-31232).

    b. Riesgo: un usuario con rol editor podría forzar la inclusión de archivos locales del servidor (LFI), leyendo ficheros sensibles (p. ej. /wp-config.php

3. ***Rehub Theme — Unauthenticated Arbitrary Shortcode Execution via re_filterpost (versión < 19.9.8) —*** *Alta*

    a. CVE vinculado (CVE-2025-7366).

    b. Riesgo: un atacante **no autenticado** puede inyectar o forzar ejecución de shortcodes arbitrarios que el tema procesa; esto puede llevar a ejecución de código en contexto WordPress, XSS persistente o manipulación de contenido visible públicamente.

4. ***Rehub Theme — Unauthenticated Password Protected Post Disclosure (versión < 19.9.8) —*** *Media-Alta*
    a. CVE vinculado (CVE-2025-7368).
    b. Riesgo: un atacante sin autenticar puede acceder o descubrir el contenido de posts protegidos por contraseña, filtrando información que debería estar restringida.
5. **Elementor — Admin+ SQL Injection (varias versiones < 3.12.2) —** Muy Alta

    a. CVE vinculado (CVE-2023-0329).

    b. Riesgo: un atacante puede inyectar consultas SQL a través de parámetros mal validados en la interfaz de Elementor, pudiendo extraer o alterar datos de la BD.

6. **Elementor — Missing Authorization / Privilege Escalation (varias versiones < 3.13.2) —** Muy Alta

    a. (sin CVE único listado)

    b. Riesgo: funciones que deberían requerir permisos no los verifican; usuarios con roles bajos pueden realizar acciones administrativas (leer adjuntos, modificar plantillas), facilitando escalada de privilegios.

7. **Elementor — Authenticated (Contributor+) Stored XSS via get_inline_svg() (versión < 3.16.5) —** Alta

    a. CVE vinculado (CVE-2023-47505).

    b. Riesgo: puede subir o inyectar SVG/markup malicioso que queda almacenado; cuando un administrador o editor visualice la página, el script malicioso se ejecuta (robo de cookies, CSRF contra admin).

8. **Elementor — Missing Authorization to Arbitrary Attachment Read (versión < 3.16.5) —** Alta

    a. CVE vinculado (CVE-2023-47504).

    b. Riesgo: usuarios con permisos bajos pueden leer archivos adjuntos privados (imágenes, documentos), exponiendo datos internos o ficheros sensibles.

9. **Elementor — Contributor+ Arbitrary File Upload to RCE via Template Import (versión < 3.18.2) —** Crítica

    a. CVE vinculado (CVE-2023-48777).

    b. Riesgo: un contributor puede subir plantillas que contienen archivos maliciosos; mediante ciertas importaciones esa subida puede derivar en ejecución de código remoto (RCE) en el servidor.

10. **GTranslate — Admin+ Stored XSS (versión < 3.0.4) —** Media-Alta

    a. CVE vinculado (CVE-2023-4502).

    b. Riesgo: un administrador puede insertar contenido malicioso que queda almacenado y se ejecuta en sesión de otros administradores, comprometiendo paneles de control o robando credenciales.

11. **WooCommerce — Shop Manager+ & Contributor+ issues (varias versiones < 8.x → 10.x) —** Alta / Crítica

    a. CVEs vinculados (CVE-2023-52222, CVE-2024-1310, CVE-2024-9944, CVE-2025-26762, CVE-2024-35777, CVE-2024-22155).

b. Riesgo: incluyen CSRF, XSS, HTML injection, creación de pedidos no autorizada, acceso a productos privados, SQLi en roles shop_manager — permiten manipular catálogo, crear pedidos falsos, filtrar datos PII o escalar privilegios.

12. **WooZone (WooCommerce Amazon Affiliates) — Arbitrary File Upload / Missing Authorization / SQLi / Privilege Escalation (Varias versiones < 14.1.0)** — Alta / Crítica

    a. CVEs vinculados (CVE-2024-33544, CVE-2024-33545, CVE-2024-33546, CVE-2024-33547, CVE-2024-33549).

    b. Riesgo: sin la autorización adecuada, un atacante puede subir archivos maliciosos, inyectar SQL, provocar escalada de privilegios o ejecutar scripts en el servidor, comprometiendo la instalación completa.

13. **Usuarios encontrados —** baja

    admwebrep

Análisis de fingerprint con Whatweb

Elemento identificado	Descripción
Servidor Cloudflare	Ya detectado por nikto
Ubicación IP: Unión Europea	Útil para localización
Email no expuesto	Menor riesgo para spam o phising
Uso de Google Analytics	Información no critica, pero indica rastreo externo
Versión de jQuery: 3.7.1	Aunque es moderna, conviene verificar si plugins cargan versiones diferentes más vulnerables.
WordPress versión 6.8.3	Indica que **WordPress está actualizado a la última versión**, lo cual es positivo. Sin embargo, que sea visible puede ayudar a ataques dirigidos si se descubre una vulnerabilidad 0-day.
Cabecera: X-Powered-By	facilita elección de exploits
XML-RPC (/xmlrpc.php)	puede usarse para fuerza bruta, pingback DDoS o amplificación
Rehub (v detectada 17.9.2 — desactualizado)	También se ha detectado por nikto. múltiples CVE (LFI, SQLi, ejecución shortcode)

Elementor 3.11.5, WooCommerce 7.5.1, WooZone, GTranslate, Rehub-framework	También detectado por nikto. múltiples CVEs (SQLi, XSS, RCE, file upload)

Relación con owasp 2024

Vulnerabilidad detectada	Categoría OWASP 2024	Explicación
PHP 7.2 desactualizado	Uso de componentes vulnerables u obsoletos	Utiliza software viejo con fallos conocidos que pueden ser explotados.
Falta de cabeceras de seguridad (X-Frame-Options, etc.)	Configuración de seguridad incorrecta	No se aplican las configuraciones básicas de protección del navegador.
API REST expuesta	Control de acceso roto	Muestra información interna que debería mantenerse privada.
Rehub Theme – Inyección SQL / Inclusión de archivos locales (LFI)	Inyección / Configuración de seguridad incorrecta	No valida bien la información que se envía desde el usuario.
Elementor – Subida de archivos → Ejecución de código (RCE)	Diseño inseguro / Configuración de seguridad incorrecta	No controla qué tipo de archivos se pueden subir.
Elementor – Escalada de privilegios	Control de acceso roto	Falla el sistema de permisos y roles de usuario.
WooCommerce – múltiples XSS y SQLi	Inyección / Control de acceso roto	No se validan bien los datos ni los permisos.
WooZone – Subida de archivos / SQLi / Escalada de privilegios	Inyección / Control de acceso roto / Configuración de seguridad incorrecta	Combina varios fallos graves que pueden dar control total del servidor.

https://devcraze.com

Vulnerabilidades más importantes (resumen y prioridad)

Información del Dominio extraída con WHOIS

Dominio	DEVCRAZE.COM
Registrador	HOSTINGER operations (IANA ID:1636)
Fecha de creación	24 de enero de 2019
Última actualización	27 de diciembre de 2023
Fecha de expiración	24 de enero de 2026
Servidores DNS	• eric.ns.cloudflare.com • jocelyn.ns.cloudflare.com
DNSSEC	No firmado (Las respuestas dns no estan firmadas permite ataques de suplantación de dns)

Hallazgos de Nikto (servidor / HTTP)

1. **Falta el encabezado X-Frame-Options** -Media
   Riesgo: Permite ataques de clickjacking
2. **Falta Strict-Transport-Security (HSTS)** - *Alta*
   Riesgo: aumenta la posibilidad de ataques m*an-in-the-middle* y downgrade TLS.
3. **robots.txt con rutas sensibles** — *Media*
   Riesgo: Revela rutas que contienen información de plugins/estructura.

4. **Versión de PHP 8.0.30 detectada (sin soporte activo)** - alta
   Riesgo: no recibe parches; hay vulnerabilidades conocidas que pueden explotarse.
5. **Exposición de endpoints REST API de WordPress-** media

   Riesgo: Permite enumeración de usuarios y recursos públicos que pueden facilitar ataques dirigidos (fuerza bruta, spear-phishing, identificación de plugins/temas).

Hallazgos de WPScan (plugins WordPress)

1. ***Autoptimize < 2.7.7 - Authenticated Arbitrary File Upload) -*** *Baja*
   a. CVE reportado (CVE-2020-24948).
   b. Riesgo: un usuario con acceso administrativo puede subir archivos arbitrarios (por ejemplo .php) y ejecutar código en el servidor.
2. ***Autoptimize < 2.8.4 - Authenticated Stored Cross-Site Scripting (XSS)-*** *Baja*
   a. CVE vinculado (CVE-2021-24332).
   b. Riesgo: un usuario autenticado puede inyectar scripts persistentes que afecten a administradores/usuarios, robando sesiones o manipulando contenidos.
3. ***Autoptimize < 2.7.8 - Race Condition leading to RCE-*** *Alta*
   a. CVE vinculado (CVE-2021-24377).
   b. Riesgo: puede conducir a ejecución remota de código
4. ***Download Monitor < 1.7.1 - Reflected Cross-Site Scripting (XSS)-****Media*
   a. CVE vinculado (CVE-2015-9296).
   b. Riesgo: entrada manipulada devuelve scripts no persistentes que pueden atacar visitantes o administradores.
5. **Download Monitor < 1.9.7 - Unauthenticated Downloading of Logs -**Muy Alta
   a. CWE vinculado (CWE-287).
   b. Riesgo: cualquiera puede descargar ficheros de logs que contienen información sensible (IPs, rutas, errores).
6. **Sassy Social Share 3.3.23 - Missing Access Controls to PHP Object Injection-** Muy Alta

a. CVE vinculado (CVE-2021-39321)

b. Riesgo: inyección de objetos PHP puede llevar a ejecución remota o corrupción del estado del plugin/servidor.

7 **Sassy Social Share < 3.3.57 - Contributor+ Stored XSS -** media

a. CVE vinculado (CVE-2024-1448).

b. Riesgo: scripts por usuarios con rol contributor

8. **Social Sharing Plugin – Sassy Social Share < 3.3.59 - Authenticated (Contributor+) Stored Cross-Site Scripting via Shortcode-** media

a. CVE vinculado (CVE-2024-1989).

b. Riesgo: Alguien puede introducir código malicioso dentro de un *shortcode* (una pequeña etiqueta que WordPress procesa y convierte en contenido

9. **Urvanov Syntax Highlighter < 2.8.34 - Highlighting Blocks Mgt via CSRF-**media

a. CVE vinculado (CVE-2023-45106).

b. Riesgo: Alguien podría enviar un email o un enlace malicioso que parece legítimo. Si el administrador hace clic o abre ese enlace, el atacante puede provocar que el administrador realice acciones en la web sin saberlo, como cambiar opciones de configuración

10. **Yoast SEO < 22.6 - Reflected Cross-Site Scripting-** Media

a. CVE vinculado (CVE-2024-4041).

b. Riesgo: XSS reflejado explotable desde urls manipuladas.

11. **Usuarios encontrados-**baja

cedcraftscodes

Análisis de fingerprint con Whatweb

Elemento identificado	Descripción

Servidor Cloudflare	Ya detectado por nikto
Ubicación IP: Estados Unidos	Útil para localización
Email no expuesto	Menor riesgo para spam o phising
Uso de Google Analytics	Información no critica, pero indica rastreo externo
Versión de jQuery: 1.12.4	Aunque es moderna, conviene verificar si plugins cargan versiones diferentes más vulnerables.
WordPress versión 6.2.8	Ya detectado por wpscan

Relación con owasp 2024

Vulnerabilidad detectada	Categoría OWASP 2024	Explicación
PHP 8.0.30 (versión fuera de mantenimiento)	Uso de componentes vulnerables u obsoletos	facilita explotación de fallos conocidos a nivel de lenguaje/servidor.
Falta de cabeceras de seguridad (X-Frame-Options, etc.)	Configuración de seguridad incorrecta	Permite *clickjacking*: otra web puede cargar tu sitio en un iframe y engañar a usuarios/administradores
Download Monitor — SQLi, LFI, SSRF, descargas arbitrarias y fallos de autorización (varios CVE recientes)	Control de acceso roto / Inyección / Configuración de seguridad incorrecta	Combina inyección SQL, inclusión local de archivos y falta de autorizaciones: permite leer/modificar BD, leer ficheros sensibles (ej. wp-config) o pivotar internamente mediante SSRF.
Autoptimize — Subida arbitraria de archivos / RCE / XSS (varias CVE: p. ej. CVE-2020-24948, CVE-2021-2437x, CVE-2023-2113)	Inyección / Configuración de seguridad incorrecta	El plugin permite subir o importar contenido sin validar, provocar ejecución de código o almacenar XSS — riesgo de control remoto del servidor o secuestro de cuentas.

Recomendación de mitigaciones

1. Actualización y mantenimiento del entorno

- Mantener todos los componentes actualizados:
    - Actualizar PHP a una versión soportada (mínimo 8.2 o superior).
    - Aplicar parches de seguridad en temas y plugins vulnerables (Elementor, WooCommerce, Rehub Theme, GTranslate, WooZone, Autoptimize, Download Monitor, Sassy Social Share, Yoast SEO, etc.).
    - Verificar periódicamente nuevas versiones y vulnerabilidades (CVE) en fuentes como *WPScan Vulnerability Database* .
- Eliminar plugins o temas obsoletos o sin soporte.
  Todo componente sin mantenimiento representa una puerta de entrada.
- Usar entornos de pruebas para validar compatibilidad antes de actualizar en producción.

2. Configuración del servidor y configuración HTTP

- **Implementar cabeceras de seguridad clave:**
    - Strict-Transport-Security: → fuerza HTTPS
    - X-Frame-Options: SAMEORIGIN → evita ataques de *clickjacking*.
    - X-Content-Type-Options: nosniff → previene ejecución de contenido con tipos MIME incorrectos.
    - Content-Security-Policy (CSP) → controla orígenes válidos de scripts.

    - Referrer-Policy: no-referrer-when-downgrade → reduce la filtración de URLs internas.
    - Permissions-Policy → limita acceso a APIs del navegador (cámara, micrófono, geolocalización).

- **Deshabilitar compresión HTTP sensible (deflate/gzip) cuando se envían datos confidenciales**, para prevenir ataques tipo *BREACH*.
- **Desactivar módulos no necesarios**, como mod_negotiation (MultiViews), que facilita enumeración de archivos.
- **Eliminar banners de servidor** para evitar revelar información de stack o versión.

3. Protección de la aplicación WordPress

- **Restringir la exposición de la REST API** (permite que aplicaciones se comuniquen) si no se usa.
    - Limitar acceso a /wp-json/wp/v2/users y otros endpoints.
- **configuraciones del archivo robots.txt:**
    - Evitar listar rutas sensibles (admin, uploads, backups).
- **Revisar permisos de roles y usuarios:**
    - Asegurar que roles como *contributor*, *editor* o *shop_manager* no tengan privilegios innecesarios.
    - Aplicar principio de mínimo privilegio.
- **Activar autenticación multifactor (2FA)** en cuentas administrativas.
    - Implementar limitador de intentos de login y CAPTCHA contra fuerza bruta.
- **Proteger el archivo wp-config.php y la carpeta /wp-admin/:**
    - Establecer permisos 400/440 (solo lectura).
    - Restringir acceso por IP.

4. Seguridad en la gestión de archivos y contenidos

- **Controlar las subidas de archivos:**
    - Limitar tipos permitidos (.jpg, .png, .pdf).
    - Bloquear la subida de .php, .html, .exe, .js, .svg .
    - Usar antivirus o escáner de archivos en tiempo real.

- **Evitar exposición de logs o archivos de configuración:**
    - Mover logs fuera del directorio público o protegerlos con autenticación.
    - Revisar permisos en archivos de registro.

5. Monitoreo, auditoría y respuesta

- **Activar logs de auditoría (WP Activity Log).**
    - Registrar inicios de sesión, cambios de roles y modificaciones de archivos.
- **Supervisar integridad de archivos de WordPress.**
    - Comparar hashes con versiones oficiales.
- **Configurar alertas de seguridad y escaneos automáticos semanales.**
    - Escanear con herramientas como *WPScan*.
- **Implementar un plan de respuesta a incidentes:**
    - Backups automáticos.
    - Procedimientos de restauración.
    - Implementación de un sistema de monitoreo en tiempo real.