



MODELO D+i
FASE II: Diseño e implementación

Integrantes del equipo

Litzy Alexandra García Richaud 10

Tadeo Uriel González Cervantes 10

Octavio Augusto Novelo Martínez 10

Aymerik Aili Ocampo González 10

Nombre del proyecto:

Proyecto: Auditor de seguridad “HERA”

Objetivo:

Desarrollar una aplicación para verificar la integridad de una red de cualquier tamaño, su seguridad y su eficiencia.

Diseño del proyecto

Para el diseño de HERA se optó por la realización de un *Dashboard*, empleando la misma gama de colores que se visualiza en el logo. Esta página demuestra los resultados principales del escaneo, organizados de forma clara y precisa para que el usuario pueda comprenderlos rápidamente. Asimismo, se enlistan sugerencias de acciones para mejorar el estado de red.

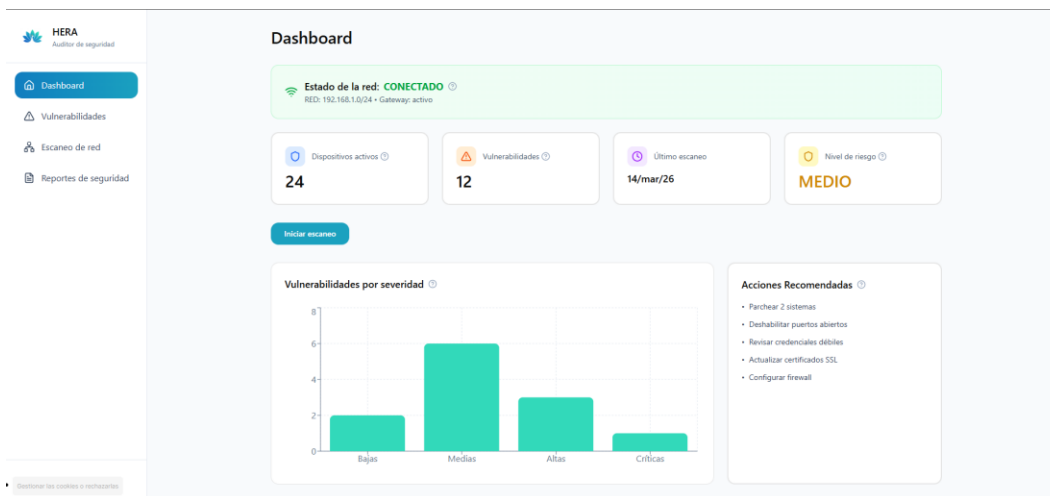


Figura 1. Página principal

En la Figura 2 se encuentra toda la información relacionada a vulnerabilidades. Aquí los resultados se explican por medio de gráficas y otros elementos visuales; sin embargo, en la parte inferior se encuentra una tabla con más detalles respecto a la severidad, dispositivos y estados de estos. Cada apartado cuenta con una breve explicación por medio de tarjetas desplegadas.

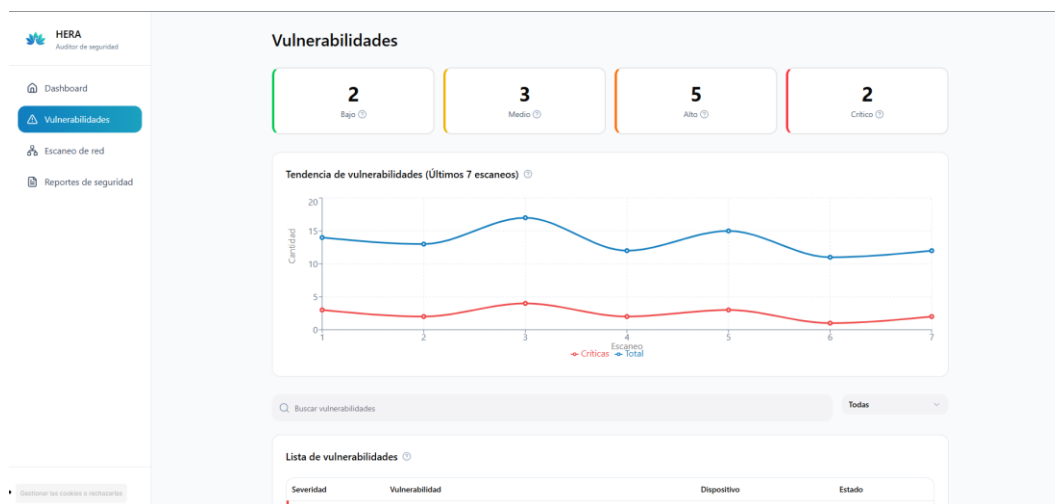


Figura 2. Página de vulnerabilidades



Figura 2.1. Continuación de Vulnerabilidades

Posteriormente, se encuentra el apartado para realizar un escaneo de red, permitiendo visualizar los resultados recientes. La tabla inferior es un resumen que incluye los datos de IP, puertos y servicios detectados.

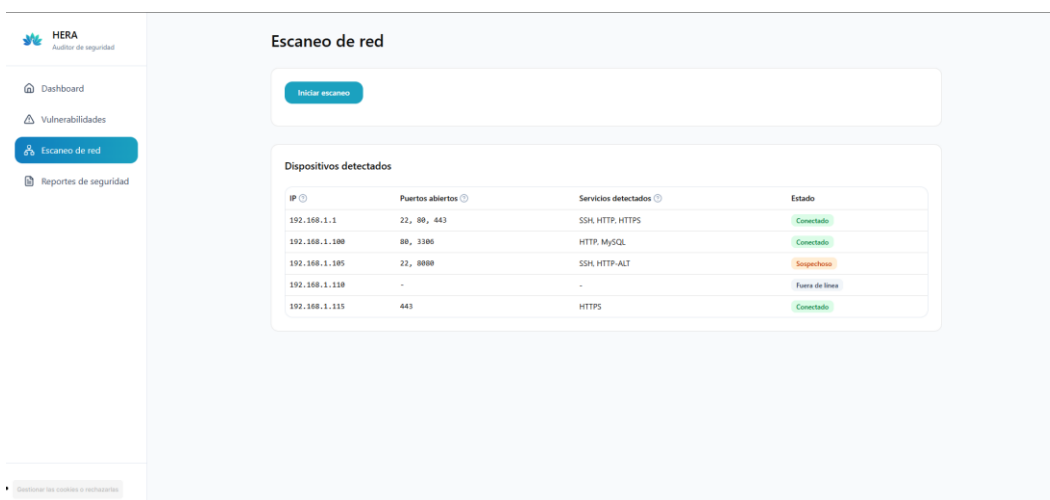


Figura 3. Escaneo de red

El historial de análisis, con la opción de descargar un reporte detallado o revisar todos los escaneos realizados, se encuentra en la cuarta página de la aplicación, marcada en la Figura 4.



Figura 4. Reportes de seguridad

En el siguiente video se describe detalladamente el funcionamiento de HERA por medio del CLI, explicando cada apartado del código y las funcionalidades abarcadas.

Enlace: [Descripcion_HERA.mov](#)

Simulación

1. Pantalla de inicio

Acción: El usuario abre HERA

Resultado: Accede al panel principal

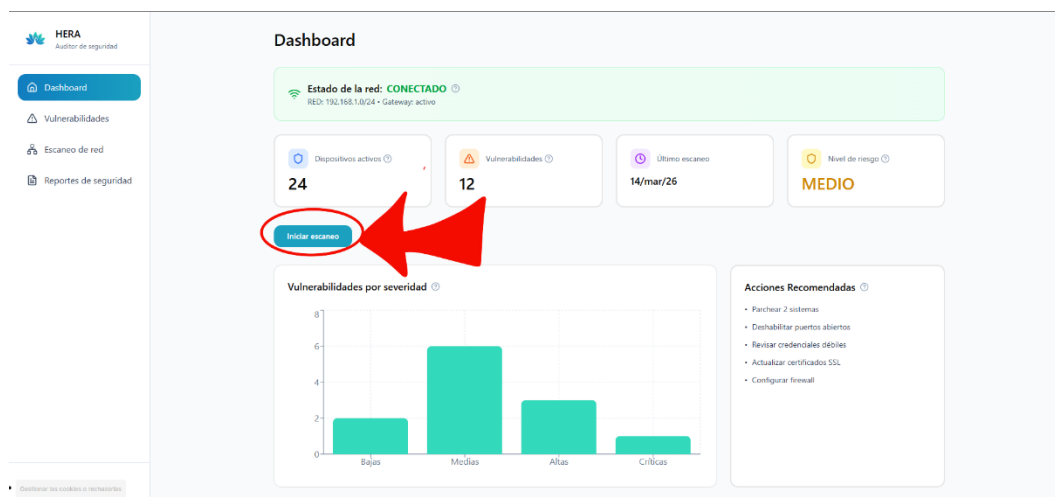


Figura 5. Realización de escaneo desde el Dashboard

2. Panel principal (Dashboard)

Visualiza: Estado general de seguridad

Clic: Botón “Iniciar escaneo”

Resultado: El sistema comienza a analizar la red



Figura 5. Seguimiento de resultados en Vulnerabilidades

3. Proceso de escaneo

Acción automática:

- HERA analiza puertos
- Detecta vulnerabilidades
- Revisa configuraciones

Indicador visual: Barra de progreso (0% → 100%)

Resultado: Se genera un reporte

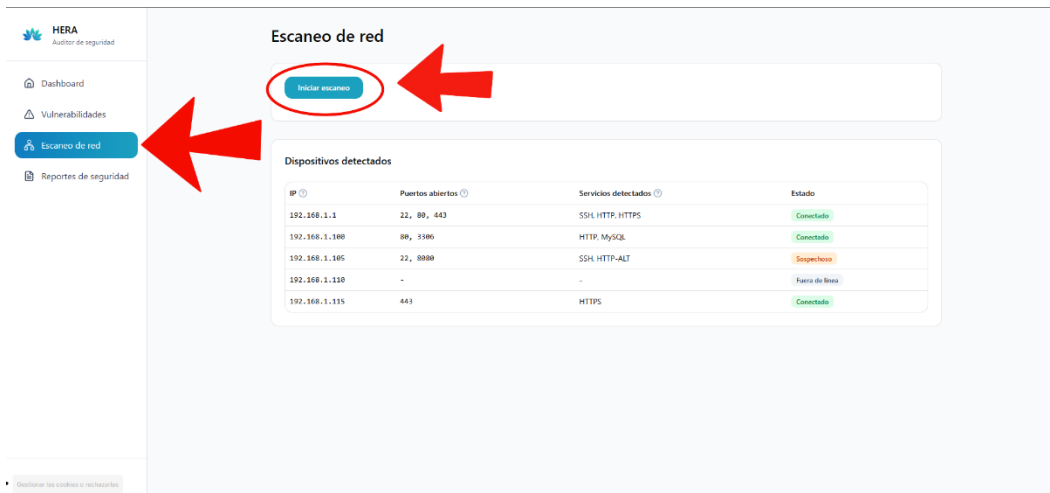


Figura 6. Visualización del proceso de escaneo

4. Resultados del análisis

Pantalla muestra:

- Nivel de riesgo (Alto / Medio / Bajo)
- Lista de problemas detectados

Clic: “Ver detalles”

Resultado: Se despliega información específica

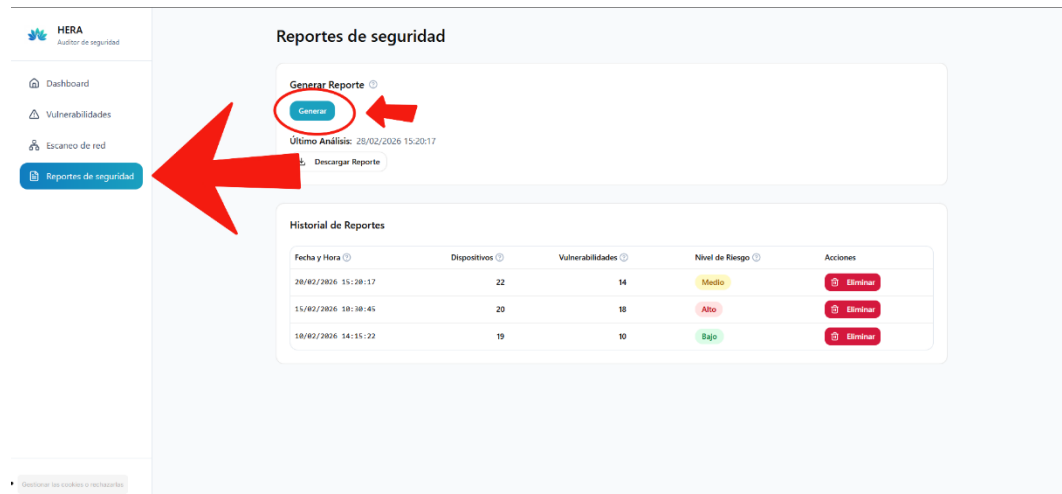


Figura 7. Visualización de resultados

Características

Para el desarrollo de HERA se emplearon las siguientes herramientas

- Nmap: De código abierto para determinar qué hosts están disponibles en la red, qué servicios ofrecen, qué sistemas ejecutan, qué tipo de filtros de paquetes se utilizan, etc. De ahí se obtuvieron las principales funciones a través del código binario y se adaptaron para las opciones del software, como el análisis de red para verificar la rapidez del internet, encontrar vulnerabilidades en los puertos, escaneos TCP, etc.
- Github: Plataforma basada en la nube para almacenar, compartir y trabajar junto con otros usuarios para escribir código. Fue el principal medio donde el desarrollo del software se realizó, pues los desarrolladores actualizaban sus cambios y todos los integrantes del equipo podían participar en su mejora.
- Canva: Herramienta de diseño gráfico que se caracteriza por proveer cientos de plantillas para que puedas crear tus diseños sin tener conocimientos en el área. Por medio de esta plataforma se diseñaron las presentaciones para las exposiciones en parciales y pitch, pues también permite trabajar de forma colaborativa y aplicar la identidad de marca de forma sencilla.
- Figma: Sirve para crear, compartir y probar diseños de sitios web, aplicaciones móviles y otros productos y experiencias digitales. Fue utilizado para la creación del prototipo final, haciendo

de este interactivo y más profesional al mejorar los bocetos iniciales.

- Visual Studio Code: Es un editor de código para programadores gratuito, de código abierto y multiplataforma. Aquí se realizó todo el desarrollo del proyecto, desde el backend hasta el avance del frontend.

A continuación se presenta el cálculo de costos del proyecto, realizado a través de Excel (Figura). En esta se encuentran los apartados necesarios para justificar su viabilidad. Se manejan comparaciones mensuales y anuales que abarcan desde el primer año de desarrollo, y el siguiente, donde habrá una mayor estabilidad de gastos. La primera sección engloba los gastos que involucran los costos de infraestructura por usuario, así como la publicidad necesaria para cumplir con las proyecciones. En *Ingresos* se enlistan las suscripciones individuales, los créditos en Azure involucran el uso de herramientas de Microsoft sin costo alguno. *Proyecciones* detalla la cantidad de suscripciones mínimas para el cumplimiento de las metas con base a las ganancias esperadas, siendo que para el primer año se recauden 60 usuarios, 5 suscripciones nuevas por mes, y para el año siguiente 120, duplicando las suscripciones mensuales. Finalmente, *Resultados anuales* recaba los anteriores resultados para el cálculo de ganancia bruta y ganancia neta tras impuestos.

COSTOS DE PROYECTO - Start Up

Gastos				
	Gastos iniciales		Gastos después del primer año	
	Gasto mensual	Gasto anual	Gasto mensual	Gasto anual
Registro de marca		\$ 4,500		
Publicidad	\$ 20,000	\$ 240,000	\$ 20,000	\$ 240,000
Infraestructura				
Internet			\$ 550	\$ 6,600
Nómina			\$ 19,200	\$ 230,400
Renta			\$ 12,500	\$ 150,000
Agua			\$ 250	\$ 3,000
Luz			\$ 1,000	\$ 12,000
Computadoras			\$ 5,500	\$ 66,000
Licencias	\$ -	\$ -	\$ -	\$ -
Servidores, dominios, etc.			\$ 1,000	\$ 12,000
Subtotal	\$ 20,000	\$ 244,500	\$ 60,000	\$ 720,000

Ingresos				
	Ingresos iniciales		Ingresos segundo año	
	Ingreso mensual	Ingreso anual	Ingreso mensual	Ingreso anual
Créditos en Azure	\$ 36,348	\$ 436,175		
Suscripciones premium por usuario	\$ 999	\$ 11,988	\$ 999	\$ 11,988

Proyecciones				
	Año 1: 5 usuarios nuevos por mes		Año 2: 10 usuarios nuevos por mes	
	Mes	Año	Mes	Año
Suscripción	\$ 4,995	\$ 719,280	\$ 9,990	\$ 1,438,560
Costo de infraestructura	\$ 1,695	\$ 244,500	\$ 5,000	\$ 720,000
Ganancia	\$ 3,300	\$ 474,780	\$ 4,990	\$ 718,560

Resultados anuales		
	Año 1	Año 2
Gastos	\$ 244,500	\$ 720,000
Ingresos totales	\$ 719,280	\$ 1,438,560
Ganancia bruta	\$ 474,780	\$ 718,560
Impuestos (30%)	\$ 142,434	\$ 215,568
Ganancia neta	\$ 332,346	\$ 502,992

Figura 8. Página principal

Enlace: [Costos de fabricación - HERA.xlsx](#)

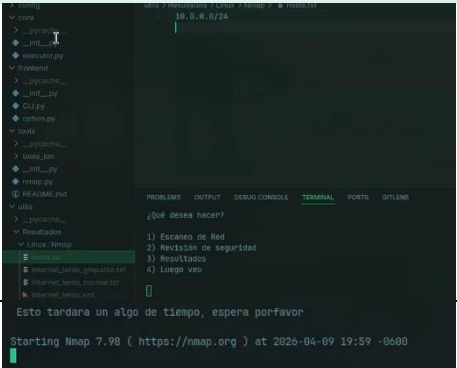


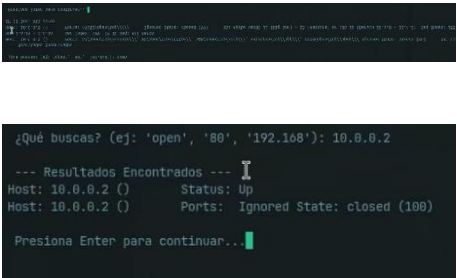


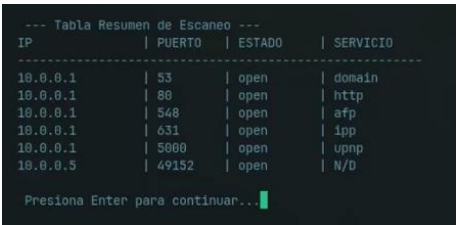
Pruebas

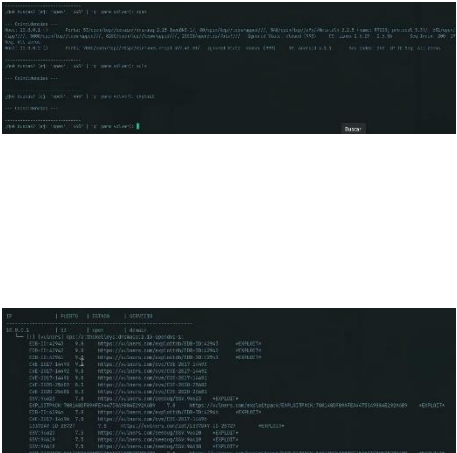
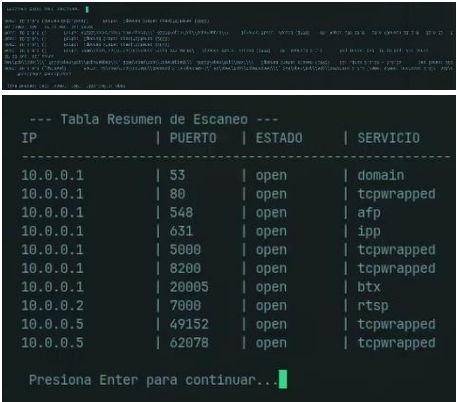
Pruebas del sistema “H.E.R.A”

Fecha: 9/abr/26

Link: [Pruebas_HERA.mov](#)

DASHBOARD				
Prueba	Resultado esperado	¿El resultado obtenido fue satisfactorio?	Comentarios	Evidencia

		Sí	No		
1. Escaneo de red					
1.1 Internet lento	Saber qué dispositivos están conectados a la red. Para ello se crean 3 archivos con los outputs	X		El tiempo de espera para la carga de resultados fue de, aproximadamente, un minuto	
1.2 Internet fallando	Ver si hay vulnerabilidades en algún dispositivo que afecte la velocidad de red	X		El tiempo de espera fue como de 8 minutos	
1.5 Personalización					
1.5.1 TCP	1) Se realiza un TCP Xmas con los 2000 puertos más comunes, sin escanear vulnerabilidades			Se tarda aproximadamente 30 minutos	
3. Resultados					
3.1 Internet lento					
3.1.1 Información específica	1) Se prueba la opción 'open' para conocer los puertos abiertos 2) Después, se escribe para conocer más detalles del host 3) Misma prueba, pero con el 10.0.0.5	X		Los hosts fueron dados correctamente	
		X		Se muestra el estado y que los 100 puertos más comunes están cerrados	
		X	X	La pantalla se resetea al no escribir correctamente el host	
		X		Esta vez se ve el puerto, el modelo del dispositivo y sistema operativo	
3.1.2 Tabla de resumen	Mostrará los IP's junto con los puertos, estados y servicios de cada uno	X			
3.2 Internet fallando					

<p>3.2.1 Información específica (Buscador)</p> <p>3.2.2 Tabla de resumen (con vulnerabilidades)</p>	<p>1) Se prueba la opción 'open'</p> <p>2) Luego, se escribe 'vuln' para ver si hay alguna vulnerabilidad</p> <p>3) Se intenta con 'exploit'</p> <p>Aquí se presenta detalles sobre las vulnerabilidades</p>	<p>X</p> <p>X</p> <p>X</p> <p>X</p>	<p>Únicamente para visualizar los hosts</p> <p>La pantalla no muestra coincidencias</p> <p>Tampoco hay coincidencias</p> <p>Mejorar la presentación al haber mucha información</p>	
<p>3.4 Personalizado</p> <p>3.4.1 Información específica</p> <p>3.4.2 Tabla de resumen</p>	<p>1) Se escribe 'open'</p> <p>Detalles de los resultados</p>	<p>X</p> <p>X</p>		 <pre> --- Tabla Resumen de Escaneo --- IP PUERTO ESTADO SERVICIO ----- ----- ----- ----- 10.0.0.1 53 open domain 10.0.0.1 80 open tcpwrapped 10.0.0.1 548 open afp 10.0.0.1 631 open ipp 10.0.0.1 5000 open tcpwrapped 10.0.0.1 8200 open tcpwrapped 10.0.0.1 28005 open btx 10.0.0.2 7000 open rtsp 10.0.0.5 49152 open tcpwrapped 10.0.0.5 62078 open tcpwrapped Presiona Enter para continuar... </pre>

Referencias

Figma Learn. (s. f.). *¿Qué es Figma?* <https://help.figma.com/hc/es-419/articles/14563969806359--Qu%C3%A9-es-Figma>

Github Docs. (s. f.). *Acerca de GitHub y Git*. <https://docs.github.com/es/get-started/start-your-journey/about-github-and-git>

Nmap. (s. f.). *Nmap: the Network Mapper - Free Security Scanner*. <https://nmap.org/>

ThePowerMBA. (2025). *Qué es Canva y cómo usarlo para crear diseños profesionales*. The Power. <https://thepower.education/blog/business/que-es-canva-y-como-usarlo-para-crear-disenos-profesionales>