Introducción

Este laboratorio tuvo como objetivo practicar técnicas de explotación de vulnerabilidades web utilizando Burp Suite como herramienta principal. Se enfocó en dos ataques clave:

-Cross-Site Scripting (XSS)

-Command Injection

El entorno consistió en una aplicación web vulnerable (simulada) configurada para pruebas éticas, aunque no se especificó si se empleó DVWA u otra plataforma.

Objetivos:

-Capturar y manipular tráfico HTTP/HTTPS.

-Demostrar explotación de vulnerabilidades con evidencias claras.

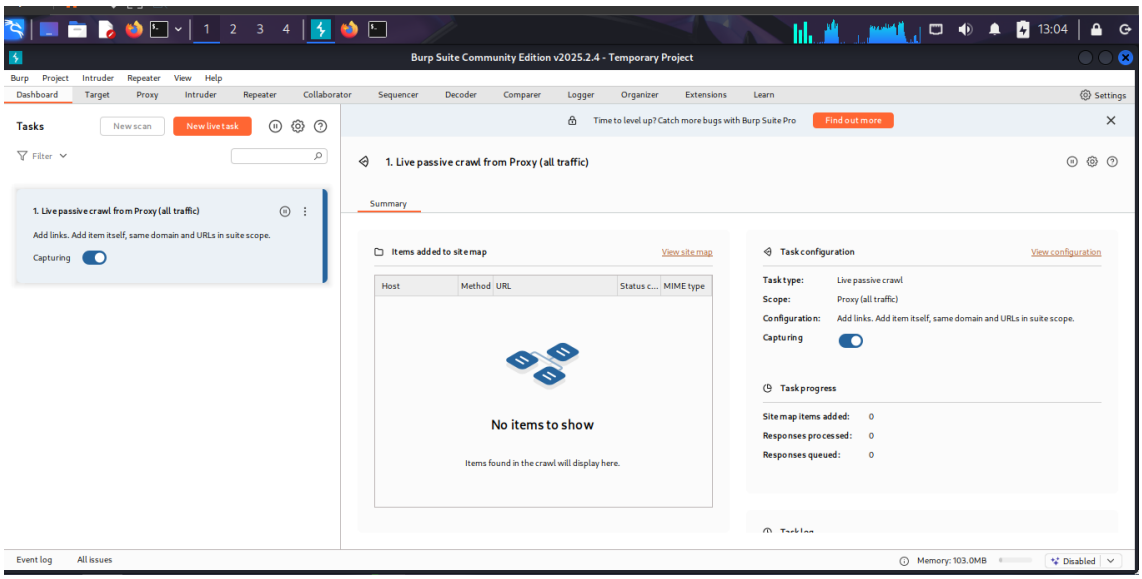
Entorno Técnico

Herramientas:

-Burp Suite Community

-Firefox configurado como proxy (127.0.0.1:8080).

-Aplicación vulnerable local (DVWA).



Configuración del Entorno

Preparación de Burp Suite:

-Iniciar Burp Suite y activar el proxy en el puerto 8080.

-Se exportó e instaló el certificado de Burp en Firefox para interceptar tráfico HTTPS sin errores de certificación

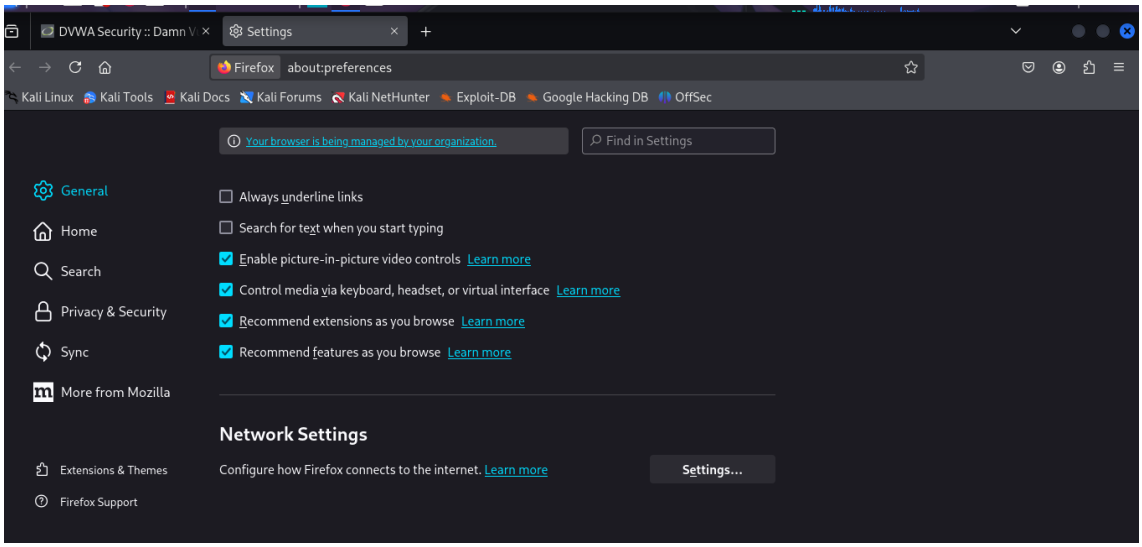
-Configuración de Firefox:

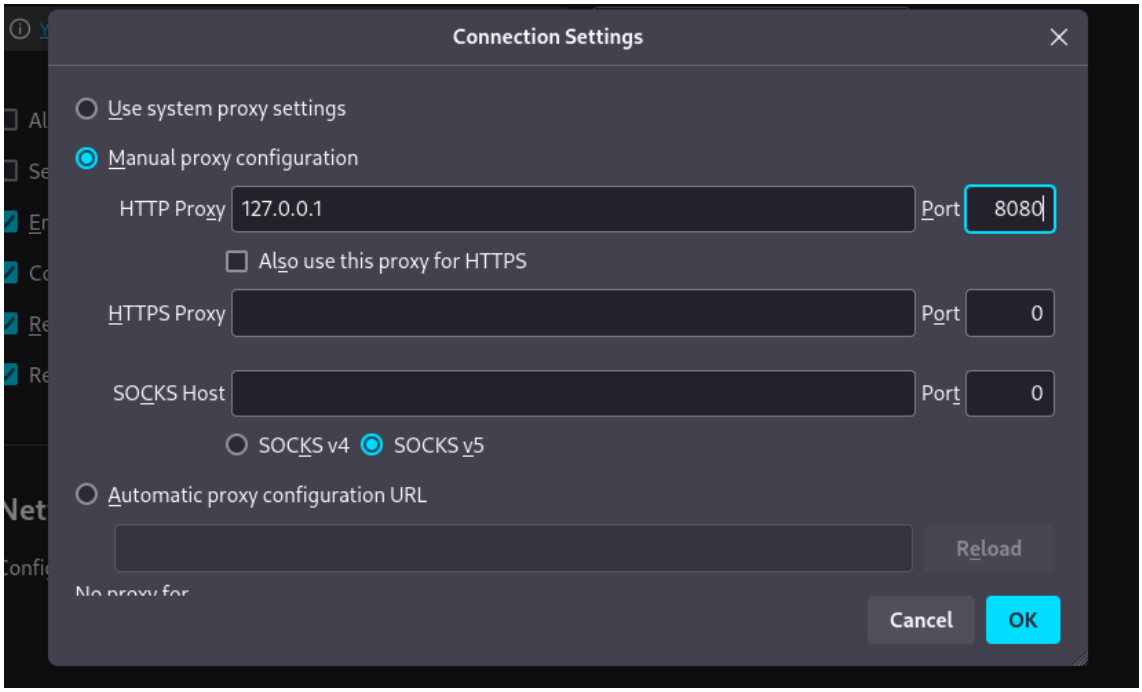
-Ir a Configuración > Red > Configuración manual del proxy.

-Establecer:

Proxy HTTP: 127.0.0.1

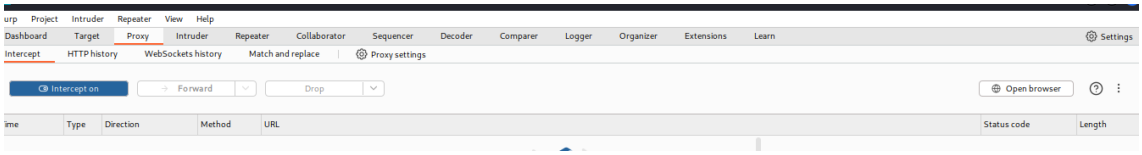
Puerto: 8080





-Activación del Intercept:

-En Burp Suite, navegar a la pestaña Proxy > Intercept y activar "Intercept is on".



-Explotación de Vulnerabilidades

-Cross-Site Scripting (XSS)

Objetivo: Ejecutar código JavaScript en el contexto del usuario.

Pasos realizados:

Interceptar una solicitud HTTP que incluya un campo de entrada.

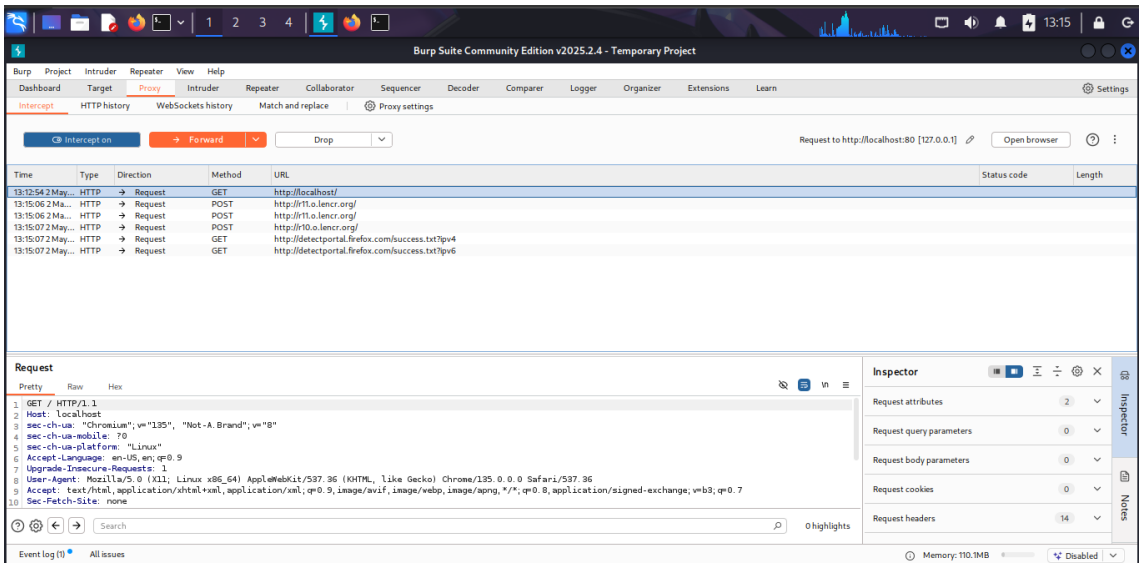
Modificar el parámetro con el payload html

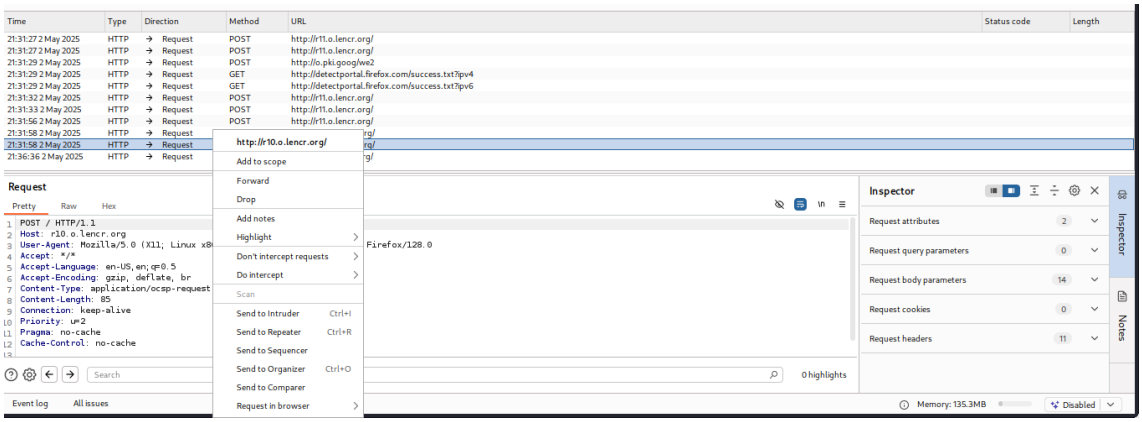
<script>alert("Hacked")</script>

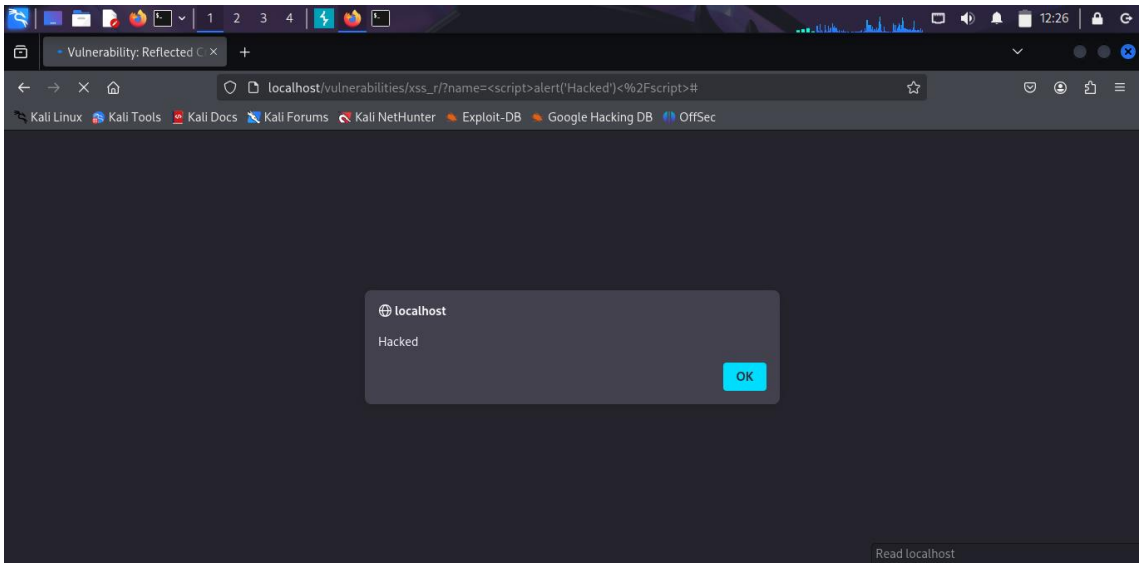
Reenviar la solicitud con Burp Repeater.

Resultado:

El navegador ejecutó el script, mostrando una alerta con el mensaje "Hacked".







Command Injection

Objetivo:

-Ejecutar comandos en el sistema operativo del servidor.

Pasos realizados:

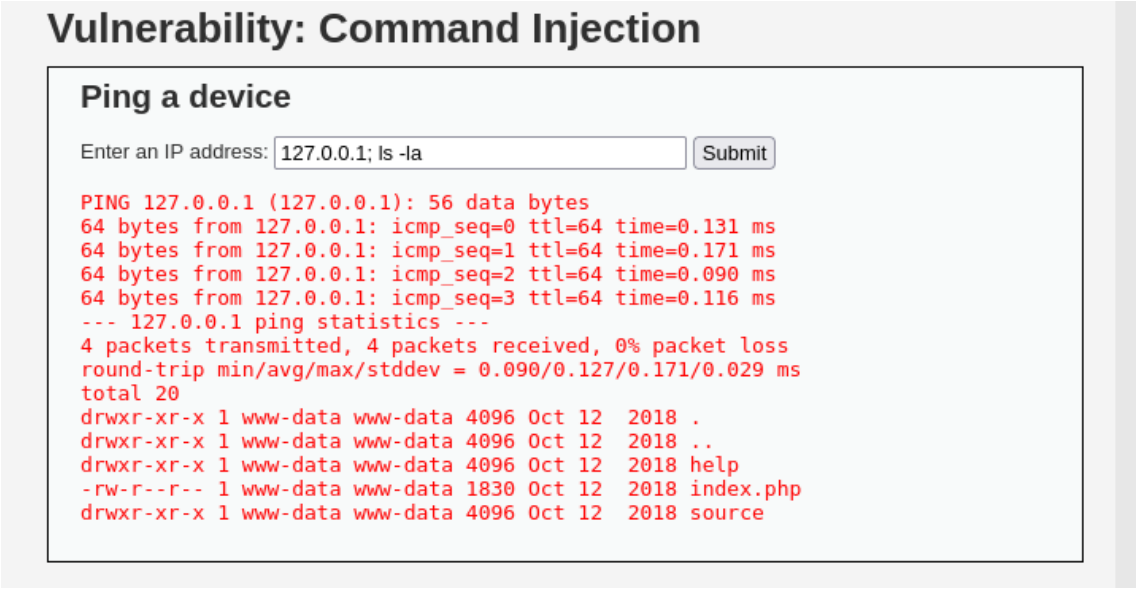
-Interceptar una solicitud que acepte entradas (ej: formulario de ping).

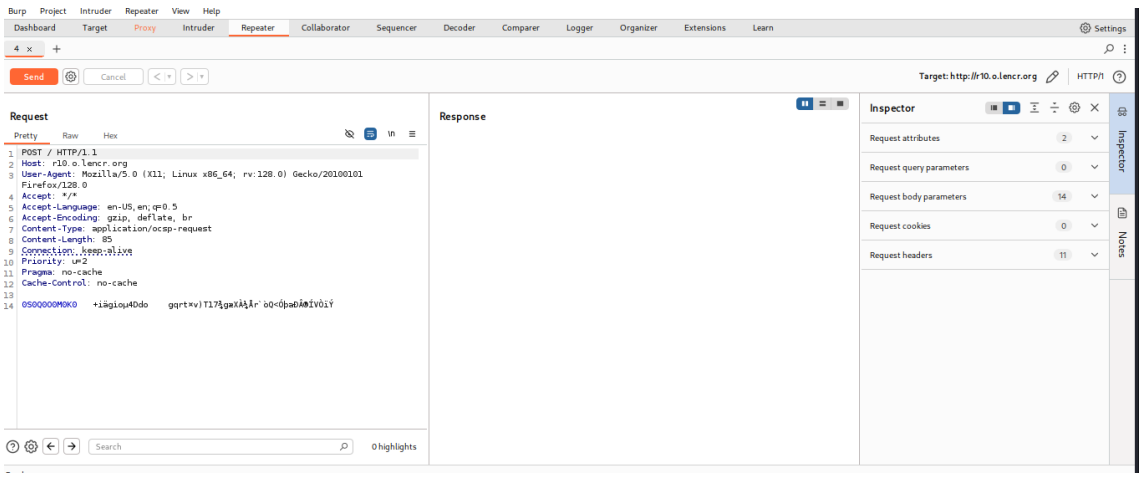
127.0.0.1; ls -la

Reenviar la solicitud y analizar la respuesta.

Resultado:

Al inyectar el comando 127.0.0.1; ls -la, el servidor respondió con el listado completo del directorio raíz, confirmando que la aplicación no sanitizaba correctamente las entradas del usuario.





-Errores y Aprendizajes

Problemas comunes:

Configuración incorrecta del proxy (ej: olvidar desactivar extensiones como VPNs).

Certificado HTTPS no instalado, bloqueando tráfico seguro.

Lecciones técnicas:

Burp Repeater es esencial para pruebas iterativas.

La falta de sanitización de entradas permite ataques simples pero críticos.

-Conclusiones

Durante las pruebas, Burp Suite permitió identificar y explotar eficientemente vulnerabilidades como XSS y Command Injection, evidenciando la importancia de una configuración segura en aplicaciones web.

Los resultados obtenidos destacan dos riesgos críticos: la ejecución de scripts arbitrarios (XSS) y la inyección de comandos del sistema. Ambos casos podrían mitigarse implementando filtros de entrada y adoptando prácticas como el principio de mínimo privilegio.

GitHub: https://github.com/Lucased12