

# FACULTAD DE INGENIERIA DEPARTAMENTO DE INGENIERIA DE SISTEMAS

Nombre de la asignatura	SEGURIDAD OFENSIVA			
Código de la asignatura	24369			
Fecha de creación	19 de marzo de 2024			
Taller	Hacking de Aplicaciones Web			
Docente	Ing. Yesith Alexander Alvarez Matta.			

# Presentación

Damn Vulnerable Web App (DVWA), es una aplicación web la cual viene con diferentes vulnerabilidades y problemas de seguridad, el objetivo principal de esta aplicación es ayudar a los profesionales de la seguridad, desarrolladores y estudiantes del campo a identificar, explotar y mitigar las vulnerabilidades que pueda presentar una aplicación sobre un entorno real. Esta herramienta proporciona un ambiente de entrenamiento robusto y a la medida, sin la preocupación de realizar un daño a un activo productivo [1].

# Objetivo

El objetivo de este taller es entender, identificar y replicar el camino utilizado por los Ciberdelincuentes, profesionales de Ciberseguridad, Auditores, Ethical Hackers y Pentesters al momento de comprometer una aplicación Web vulnerable expuesta a internet.

# Habilidades que contribuye a desarrollar

- Identificación de las vulnerabilidades más comunes sobre las aplicaciones Web.
- Reconocimiento de las herramientas y procedimientos más utilizados para la detección y explotación de vulnerabilidades sobre las aplicaciones Web.
- Ejecución y explotación a una aplicación web con vulnerabilidades críticas y medias.
- Reporte y documentación del proceso realizado.

#### **Taller**

1. Sobre una máquina virtual con Kali Linux previamente instalada, descargue y configure el siguiente Docker: <a href="https://hub.docker.com/r/vulnerables/web-dvwa">https://hub.docker.com/r/vulnerables/web-dvwa</a>.

Comando 1: sudo docker pull vulnerables/web-dvwa

Comando 2: sudo docker run --rm -it -p 80:80 vulnerables/web-dvwa

Figure 1. Ejecución de docker.

Nota: Si el comando "docker" no es reconocido, realice la instalación con el comando:

Comando 0: sudo apt install docker.io

2. Ejecute sobre su navegador la siguiente dirección: http://127.0.0.1



Figure 2. Acceso a la aplicación.

3. Con las credenciales que a continuación se relacionan ingrese al aplicativo de prueba:

Username: admin Password: password



Figure 3. Configuración de la aplicación.

4. Una vez dentro de la aplicación dé clic en el botón "Create/Reset Database" para crear la base de datos:

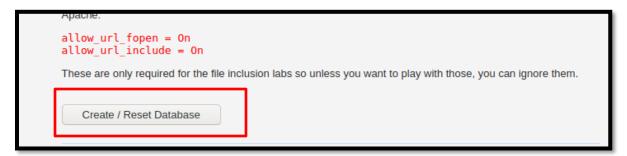


Figure 4. Creación de la base de datos.

**5.** La aplicación cerrará sesión automáticamente y usted deberá acceder nuevamente con el usuario y la contraseña del punto 3.

6. Luego diríjase a la sección "DVWA Security", realice el cambio de Low a High y presione el botón "Submit" ☑ DVWA Security :: Damn V(× + O 🗅 127.0.0.1/security.php **■** 140% ☆ DVWA Security 🎉 Home Instructions **Security Level** Setup / Reset DB Security level is currently: low. **Brute Force** You can set the security level to low, medium, high or impossible. The security level changes the vulnerability **Command Injection** level of DVWA: CSRF 1. Low - This security level is completely vulnerable and has no security measures at all. It's use is to be as an example of how web application vulnerabilities manifest through bad coding practices and to serve File Inclusion as a platform to teach or learn basic exploitation techniques. File Upload 2. Medium - This setting is mainly to give an example to the user of bad security practices, where the developer has tried but failed to secure an application. It also acts as a challenge to users to refine their Insecure CAPTCHA exploitation techniques. 3. High - This option is an extension to the medium difficulty, with a mixture of harder or alternative bad **SQL** Injection practices to attempt to secure the code. The vulnerability may not allow the same extent of the SQL Injection (Blind) exploitation, similar in various Capture The Flags (CTFs) competitions. 4. Impossible - This level should be secure against all vulnerabilities. It is used to compare the vulnerab Weak Session IDs source code to the secure source code.

Prior to DVWA v1.9. this level was known as 'high'. XSS (DOM) XSS (Reflected) Low ∨ Submit XSS (Stored) **CSP Bypass PHPIDS** JavaScript PHPIDS v0.6 (PHP-Intrusion Detection System) is a security layer for PHP based web applications. PHPIDS works by filtering any user supplied input against a blacklist of potentially malicious code. It is used in Figure 5. Sección de seguridad. High Submit



Figure 6. Cambio de nivel de seguridad

Nota: Para validar que la aplicación se encuentra configurada en modo High, acceda a las herramientas de desarrollador (F12), luego a "Storage" y finalmente en la sección de Cookies encontrara una variable con el valor configurado.



Figure 7. Validación del Cambio del nivel de seguridad

#### **Ejercicio**

**a.** Realice una prueba de intrusión para la empresa DAMN INC evaluando la seguridad de la aplicación, pruebe cada una de las opciones que se evidencian a continuación:



Figure 8. Objetivo de la aplicación.

- **b.** Realice un documento detallado sobre las pruebas realizadas, teniendo en cuenta (Portada, Introducción, Tabla de Contenido, Informe Ejecutivo, Informe técnico, Glosario).
- **c.** En la sección Informe Ejecutivo del documento, este debe incluir:
  - Graficas con las vulnerabilidades encontradas.
  - Criticidad de las vulnerabilidades en un mapa de calor.
  - Recomendaciones generales por parte del auditor.
- d. En la sección Informe técnico del documento, este debe incluir:
  - **Severidad:** Alta, Media o Baja.
  - CVSS: https://chandanbn.github.io/cvss/
  - Clasificación: CWE-### (Código CWE la vulnerabilidad)
  - Sistema Vulnerable: http://127.0.0.1/ (URL completa donde se encontró el hallazgo)
  - Vulnerabilidad: (Resumen de la vulnerabilidad).
  - Impacto de la vulnerabilidad: (Explicación del impacto de la vulnerabilidad).
  - Solución: Posible solución con el Link de soporte.
  - Evidencia: (Imágenes y descripciones del paso a paso para explotar la vulnerabilidad).

**Nota:** Documente cada acción realizada con pantallazos, no olvide nombrar las herramientas que utiliza y los pasos detallados para llegar a la vulnerabilidad.

#### Referencias

- [1]. https://dvwa.co.uk/
- [2]. https://cwe.mitre.org/
- [3]. https://owasp.org/