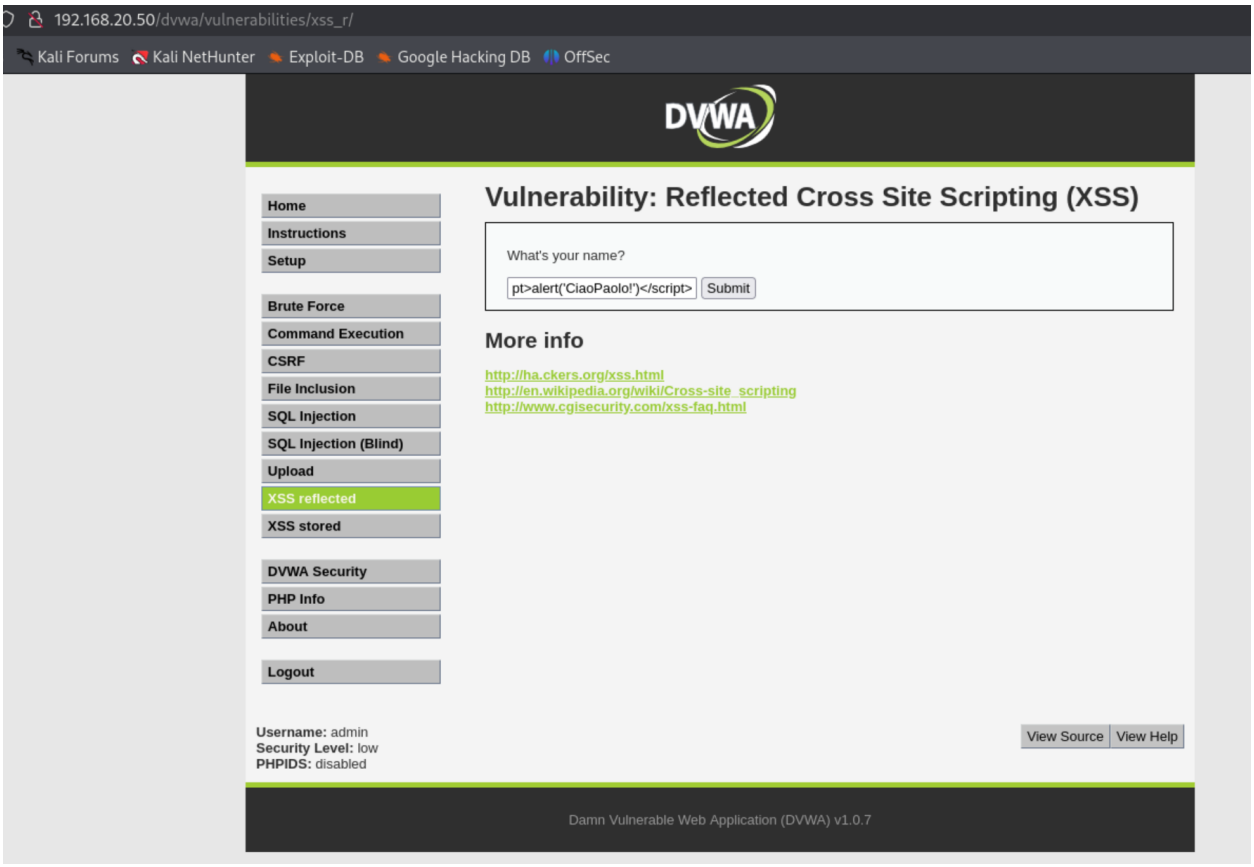
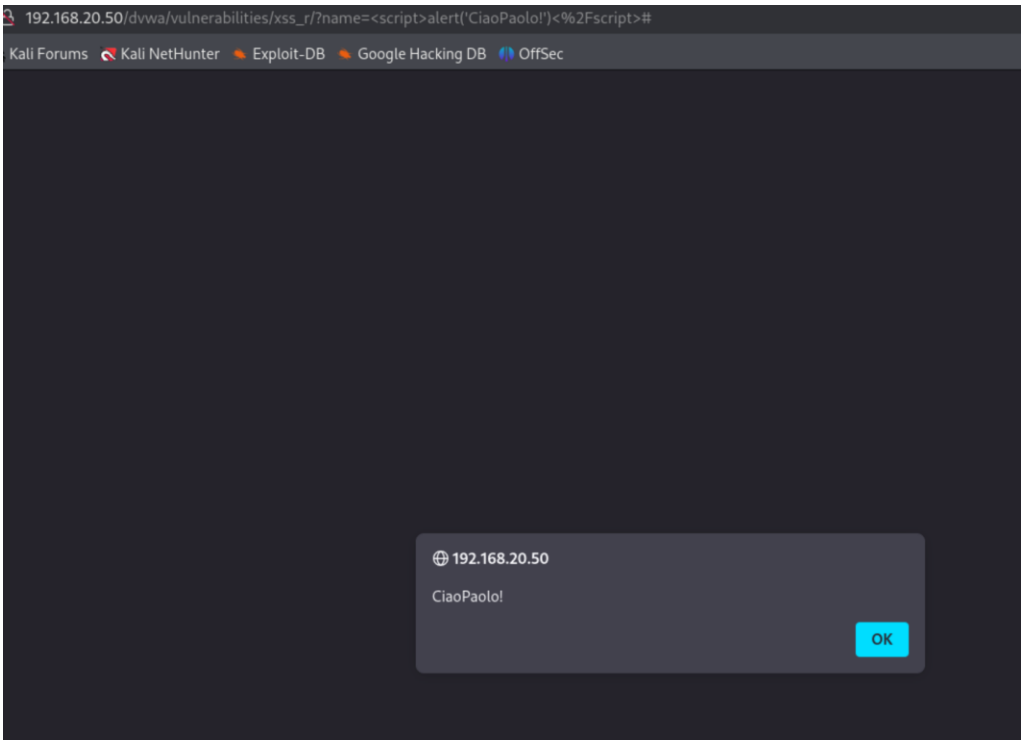
ESERCIZIO



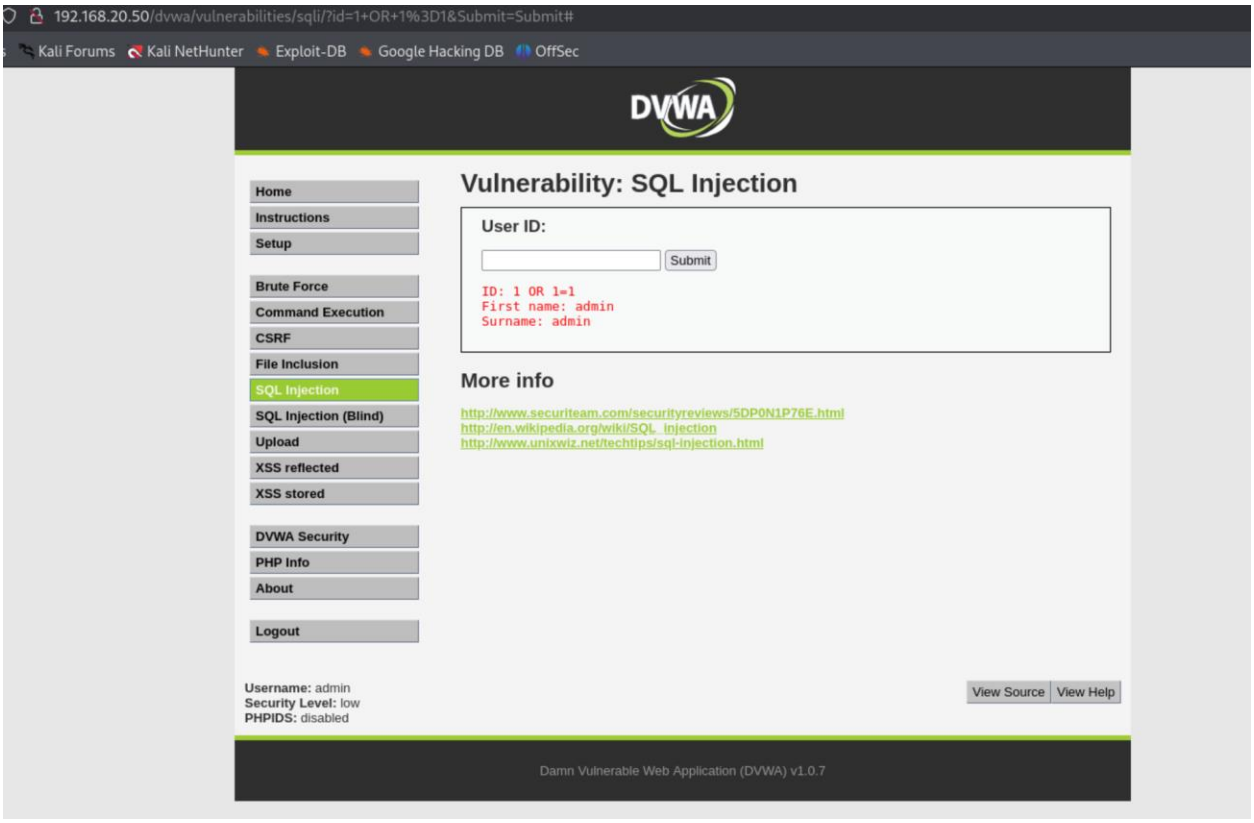
Nella pagina XSS reflected della DVWA andiamo a testare il nostro payload:

<script>alert('CiaoPaolo!')</script>



Dato che la pagina è vulnerabile, quando inviamo il form o visitiamo l'URL con il payload, vediamo apparire una finestra di allarme con il testo "CiaoPaolo!". Questo dimostra che il codice JavaScript è stato eseguito nel contesto del browser dell'utente.

Ovviamente noi ci siamo limitati a salutare il nostro professore, ma è intuibile come un utente malintenzionato possa sfruttare un attacco XSS riflesso, generando un link con payload malevolo e distribuendolo ad utenti ignari che lo attiveranno semplicemente cliccandoci sopra.



Qui, invece, vediamo un esempio di SQL injection. Per sfruttare la vulnerabilità SQL Injection dobbiamo manipolare l'input dell'ID utente in modo da interferire con la query SQL che viene eseguita sul database.

1 OR 1=1 Questa semplice payload, che si basa su una condizione sempre vera, ha funzionato e ci conferma la vulnerabilità a SQL injection.

La nostra breccia è aperta, da qui non resta che allargarla utilizzando ad esempio la tecnica UNION SELECT.