Prove di attacchi

SQL Injection

abbiamo verificato che l'input della DVWA fosse vulnerabile grazie al carattere " ' "(apice).



Se comparirà il seguente messaggio, avremo la conferma che il codice è vulnerabile:



Una volta confermata la vulnerabilità, ecco che abbiamo ricercato lo script SQL più adatto al nostro scopo, in questo caso quello di ottenere tutti gli utenti e le loro informazioni bypassando la password e rendendo la condizione obbligatoriamente vera:



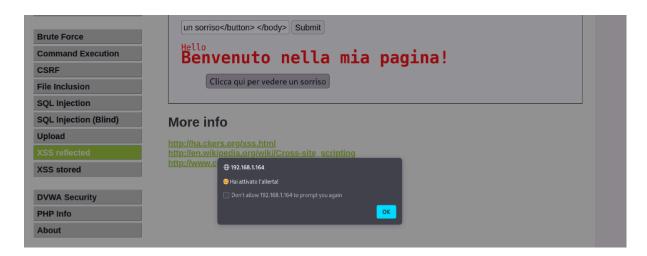
XSS Reflected

Provando con un attacco XSS Reflected invece abbiamo inserito questo codice nella casella di input-text:

<head> <meta charset="UTF-8"> <meta name="viewport" content="width=device-width,
initial-scale=1.0"> <title>Allerta Smiley</title> <script> function mostraAllerta() { alert(": Hai
attivato l'allerta!"); } </script> </head>

<body> <h1>Benvenuto nella mia pagina!</h1> <button onclick="mostraAllerta()">Clicca qui
per vedere un sorriso</button> </body>.

e questo è il risultato:



XSS STORED

Qui abbiamo inserito un codice, per far apparire un'allerta pop-up:

Home	Vulnerability: Stored Cross Site Scripting (XSS)
Instructions	
Setup	Name * Diego
	<pre><script>alert('XSS Stored Attack!');</script></pre>
Brute Force	Message *
Command Execution	Sign Guestbook
CSRF	⊕ 192.168.1.164
File Inclusion	XSS Stored Attack!
SQL Injection	Name: test Massage: This is a lest comment

