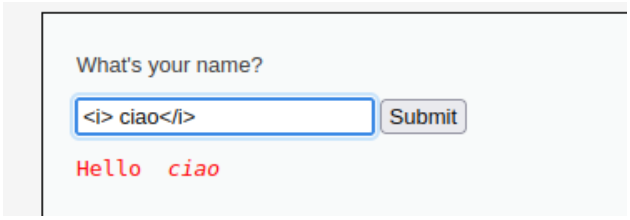
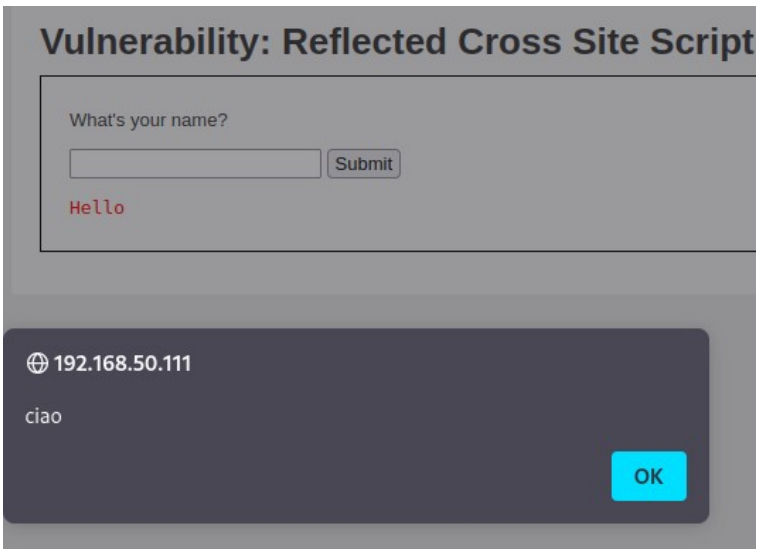
REPORT 29/11/2022 LORENZO SCORBAIOLI

TASK: EXPLOIT DVWA – XSS E SQL INJECTION

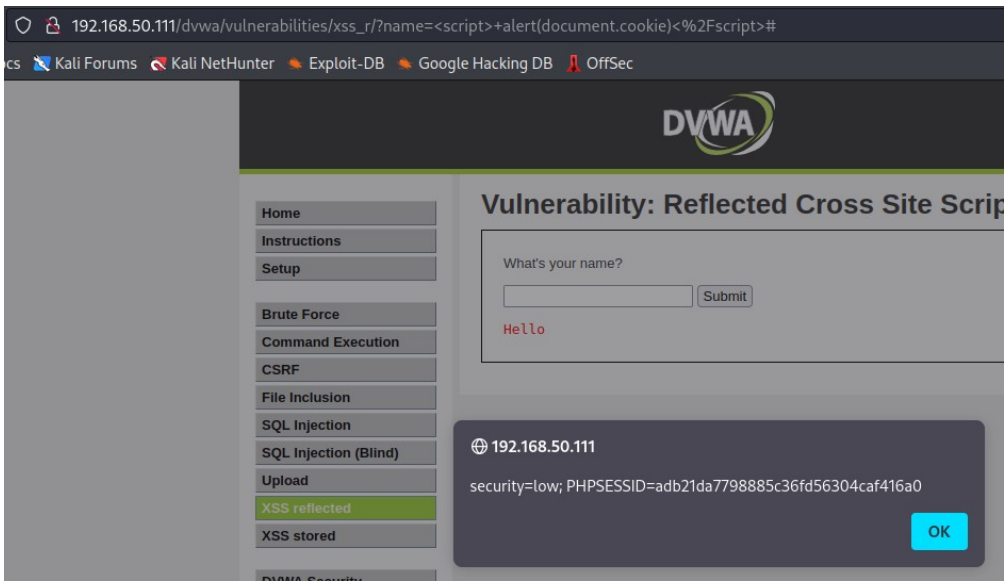
Per verificare che il sito http://192.168.50.111/dvwa/vulnerabilities/xss_r/ fosse effettivamente suscettibile ad attacchi XSS riflessi sono andato inizialmente a inserire ciao nello spazio del submit ottenendo questo risultato:



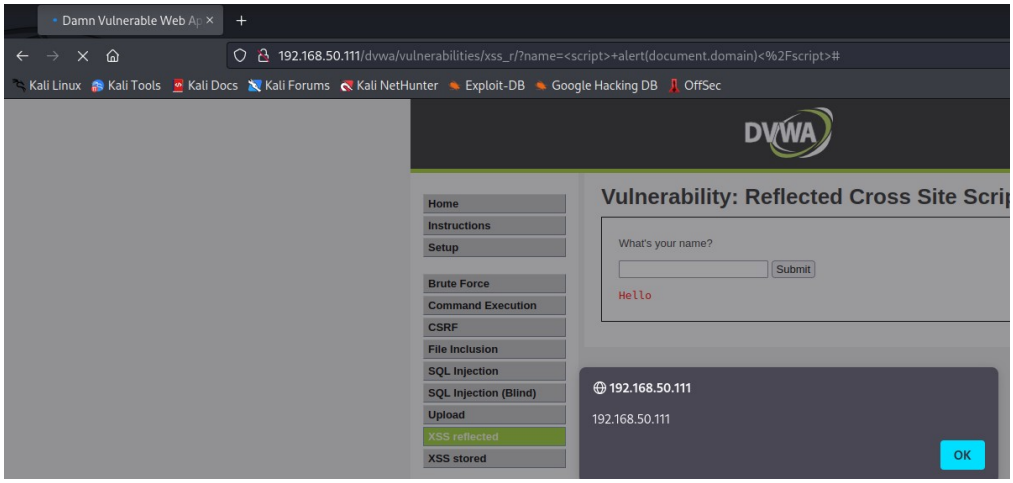
Come si può vedere dall'immagine qui sopra sono andato a modificre l'output della pagina HTML facendo uscire in output ciao in corsivo.
Successivamente ho provato a lanciare un piccolo script per fare uscire una finestra popup sempre con la scritta ciao, come si può vedere a sinistra, mentre a destra possiamo vedere il risultato:



Dopo sono andato a lanciare uno script (<script>alert(document.cookie)</script>) per ottenere sempre tramite un popup che ci mostrasse il cookie di sessione:



Ho provato anche lo script <script>alert(document.domain)</script> cche come si può vedere nell'immagine qui sotto mi ha restituito in un popup l'indirizzo IP che usciva anche nell'intestazione dei popup precedenti:

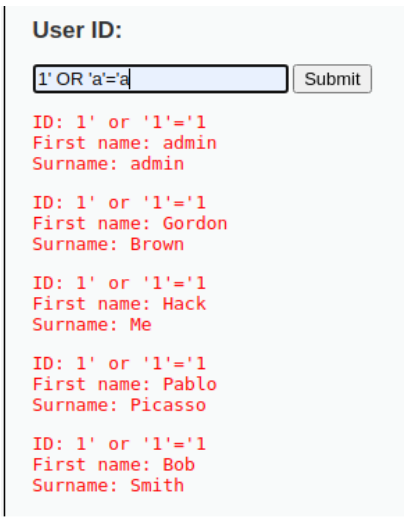


A questo punto sono passato a fare SQL injection. Sono andato a controllare il source code della pagina:

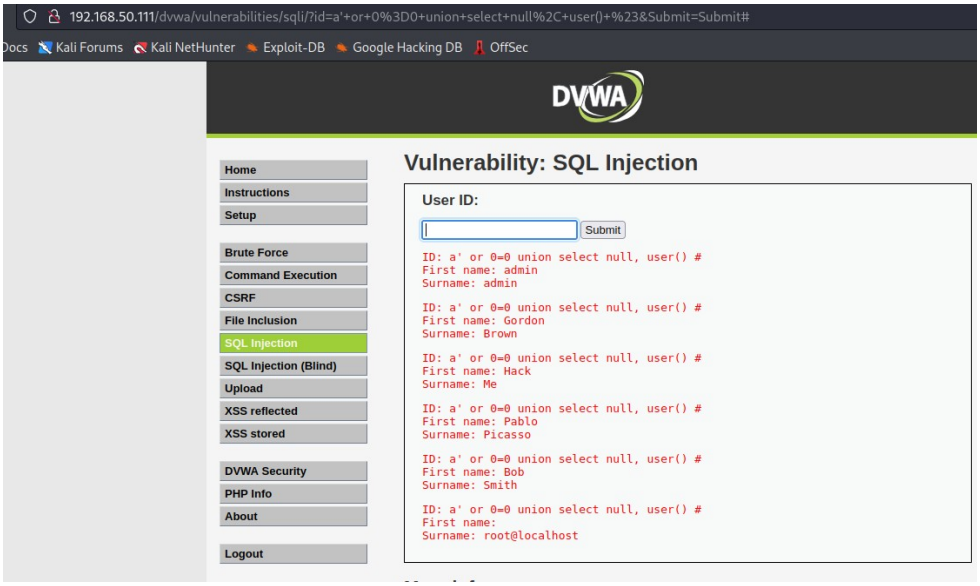
SQL Injection Source

```php
<?php

if(isset($_GET['Submit'])){

    // Retrieve data

    $id = $_GET['id'];

    $getid = "SELECT first_name, last_name FROM users WHERE user_id = '$id'";
    $result = mysql_query($getid) or die('<pre>' . mysql_error() . '</pre>' );

    $num = mysql_numrows($result);

    $i = 0;

    while ($i < $num) {

        $first = mysql_result($result,$i,"first_name");
        $last = mysql_result($result,$i,"last_name");

        echo '<pre>';
        echo 'ID: ' . $id . '<br>First name: ' . $first . '<br>Surname: ' . $last;
        echo '</pre>';

        $i++;
    }
}
?>
```

Ho notato che l'input è per una query dinamica. Ho quindi provato a vedere se nella casella submit si può inserire una query invece di un singolo parametro, con il seguente risultato, riuscendo a trovare tutte le entry del database:



Infine ho provato a usare il comando UNION ottenendo questo risultato:



Infine ho provato a vedere se si potesse fare l'Injection con sqlmap lanciando il comando sqlmap -u "http://192.168.50.111/dvwa/vulnerabilities/sqli/?id=1&Submit=Submit" -cookie="security=low; PHPSESSID=634539b620963f3da02e914c34857df4" –dump dopo avere recuperato il cookie di sessione con Burpsuite. Come si può vedere dalle immagini qui sotto possiamo vedere che il parametro "id" può essere soggetto a Injection e abbiamo ottenuto tutto il contenuto del database.