**Practical 6a Web Application Security - SQLmap**

SQLmap is an automated SQL Injection tool. In the previous exercises on SQL Injection, you crafted the SQL queries in order to exploit the SQL Injection vulnerabilities. With SQLmap, these steps could be automated.

**Exercise Start XAMPP and set up DVWA**

We will use the SQL Injection vulnerability in DVWA. You have already installed XAMPP and DVWA in the previous exercises.

In Win10 VM

1. In the Search textbox, type “XAMPP”.
2. Run XAMPP Control Panel.
3. Start the Apache Web Server and MySQL.
4. Check that you can browse to http://127.0.0.1/dvwa-master.

**Exercise Using SQLmap**

SQLmap is already installed on Kali

In Kali :

1. Browse to your DVWA.
2. Login with username “admin” and default password “password”. (If you have changed the DVWA admin password, use the new password)
3. In the left hand menu, click on DVWA Security. Set the security level to low.
4. Click on SQL Injection.
5. In a terminal, run the following commands to see the options for SQLmap, or view the manual page for SQLmap.

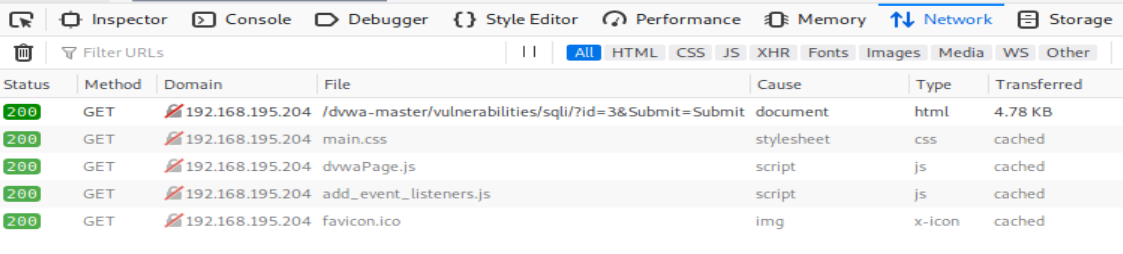
sqlmap -h

man sqlmap

1. We will run SQLmap with a target URL. The target URL is the webpage that has the SQL Injection vulnerability. We will also provide any cookie values if required.
2. To see the HTTP request that is sent to the web application, we can use the F12 Developer Tools of the web browser.

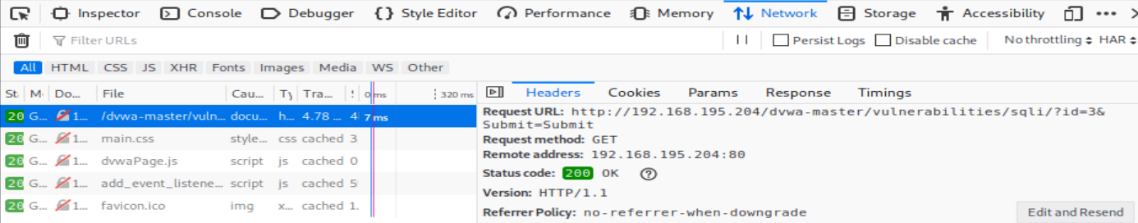
In the Web Browser, press F12 to bring up the Developer Tools.

1. In the Web Browser Developer Tools, click on the Network tab.
2. In the Web Browser, enter a number for the User ID (eg 3) and click Submit.
3. In the Web Browser Developer Tools, under the Network tab, look for the HTTP packet that contains the request and select it.



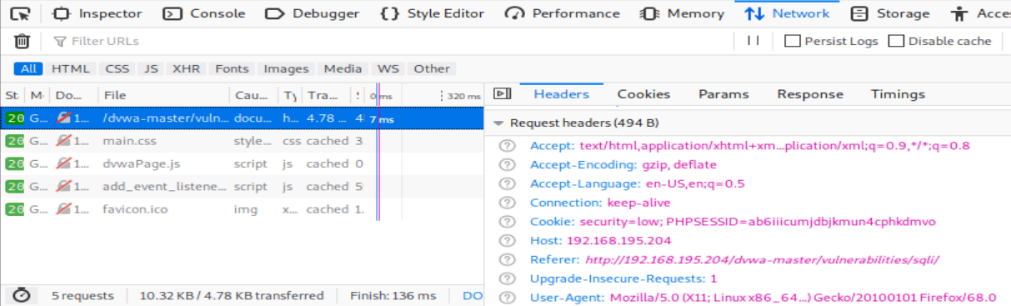
Under the Network tab, select the packet containing the SQL Injection request.

1. In the Web Browser Developer Tools, look for the Request URL of the selected packet. This will be the Target URL we will provide to SQLmap.



Request URL

1. In the Web Browser Developer Tools, look for the Request Headers of the selected packet. Look for the Cookie value. We will provide this Cookie value to SQLmap.



Cookie value

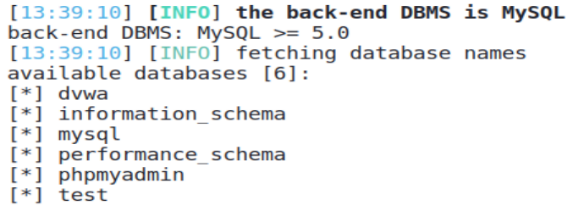
1. In the terminal, run SQLmap with the Target URL and the Cookie. The --dbs option means to list out any databases found.

Press Enter to use the default options when asked.

sudo sqlmap –u "http://192.168.13.100/dvwa-master/vulnerabilities /sqli/?id=3&Submit=Submit" --cookie="security=low; PHPSESSID=ab6iiicumjdbjkmun4cphkdmvo" --dbs

Run in a single line and replace with your Win10 IP and your cookie value

1. When SQLmap completes, you will see information about the databases found.



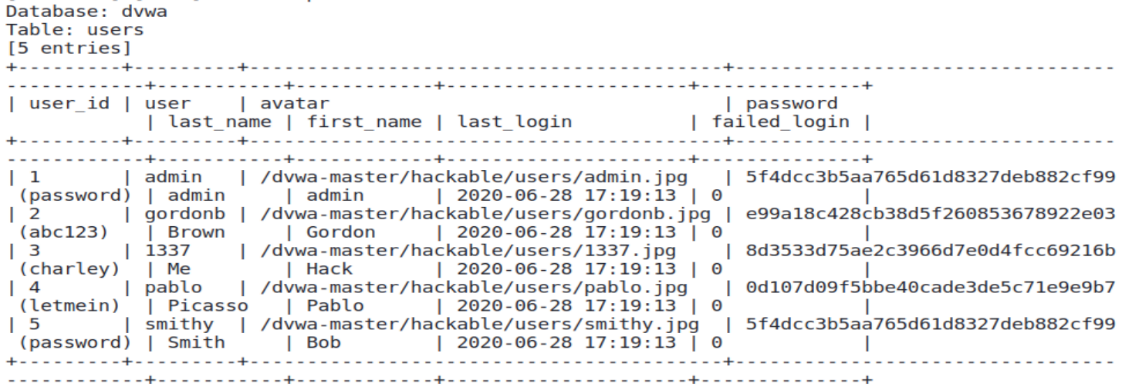
1. In the terminal, run SQLmap with the Target URL and the Cookie again, this time with the –D and --dump-all options to specify the dvwa database and to dump the contents of every table.

Press Enter to use the default options when asked.

sudo sqlmap –u "http://192.168.13.100/dvwa-master/vulnerabilities /sqli/?id=3&Submit=Submit" --cookie="security=low; PHPSESSID=ab6iiicumjdbjkmun4cphkdmvo" –D dvwa --dump-all

Run in a single line and replace with your Win10 IP and your cookie value

1. When SQLmap completes, it will display the contents of the tables in the dvwa database. It even tries to crack the password hashes in the users table.



**Exercise Using SQLmap with Blind SQL Injection**

An extract of the SQLmap output showing the contents of the users table plus the cracked passwords

In Kali :

1. Browse to DVWA and check the Security Level is low.
2. Click on SQL Injection (Blind).
3. For User ID, enter “1”. The message “User ID exists in the database” is displayed.
4. For User ID, enter “6”. The message “User ID is MISSING from the database” is displayed.
5. If you try SQL Injection by entering the following for the User ID (see diagram).

ppp' or '0' = '0



You only see the message “User ID exists in the database”, indicating the SQL Injection is successful, and the SQL query was able to return records, though the web page will not display any of the retrieved records.

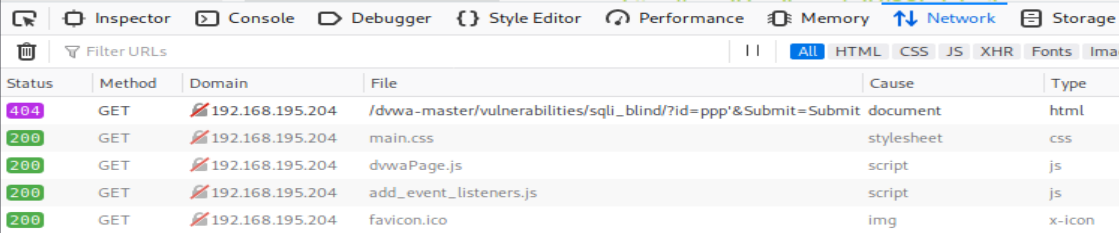
So this is considered Blind SQL Injection, as there are only two different messages displayed as output to your SQL Injection attempts.

Blind SQL Injection attack is more advanced, but SQLmap can help.

1. You can use the Web Browser Developer Tools of the web browser to find the Request URL of the web page that is vulnerable to the SQL Injection and the cookies.

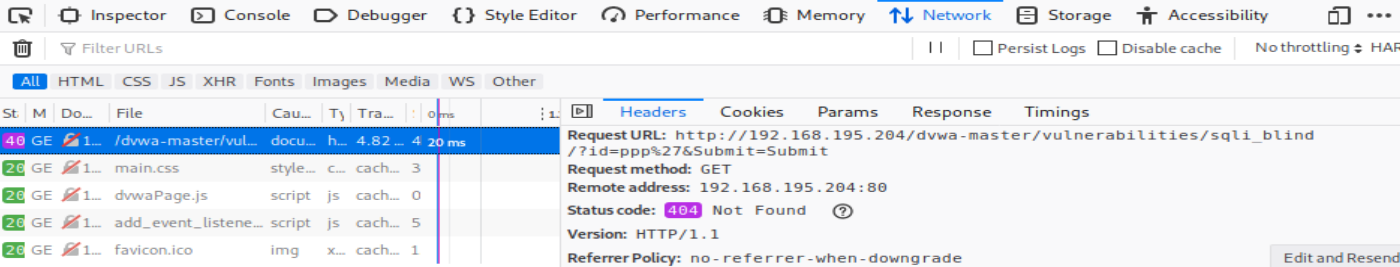
In the Web Browser, press F12 to bring up the Developer Tools.

1. In the Web Browser Developer Tools, click on the Network tab.
2. In the Web Browser, enter a number for the User ID (eg 3) and click Submit.
3. In the Web Browser Developer Tools, under the Network tab, look for the HTTP packet that contains the request and select it.



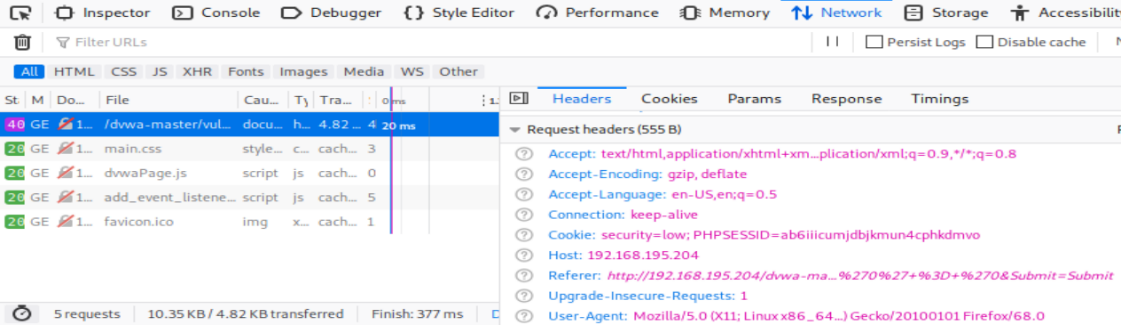
Under the Network tab, select the packet containing the SQL Injection request.

1. In the Web Browser Developer Tools, look for the Request URL of the selected packet. This will be the Target URL we will provide to SQLmap.



Request URL

1. In the Web Browser Developer Tools, look for the Request Headers of the selected packet. Look for the Cookie value. We will provide this Cookie value to SQLmap.



Cookie value

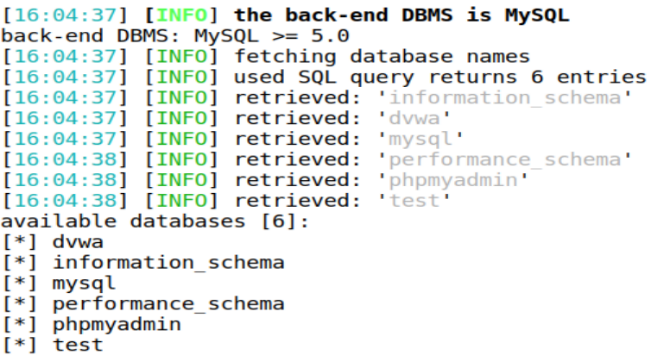
1. In the terminal, run SQLmap with the Target URL and the Cookie. The --dbs option means to list out any databases found. The command is similar to the command in the previous exercise against the normal SQL Injection webpage.

Press Enter to use the default options when asked.

sudo sqlmap –u "http://192.168.13.100/dvwa-master/vulnerabilities /sqli\_blind/?id=3&Submit=Submit" --cookie="security=low; PHPSESSID=ab6iiicumjdbjkmun4cphkdmvo" --dbs

Run in a single line and replace with your Win10 IP and your cookie value

1. When SQLmap completes, like in the previous exercise, you will see information about the databases found.



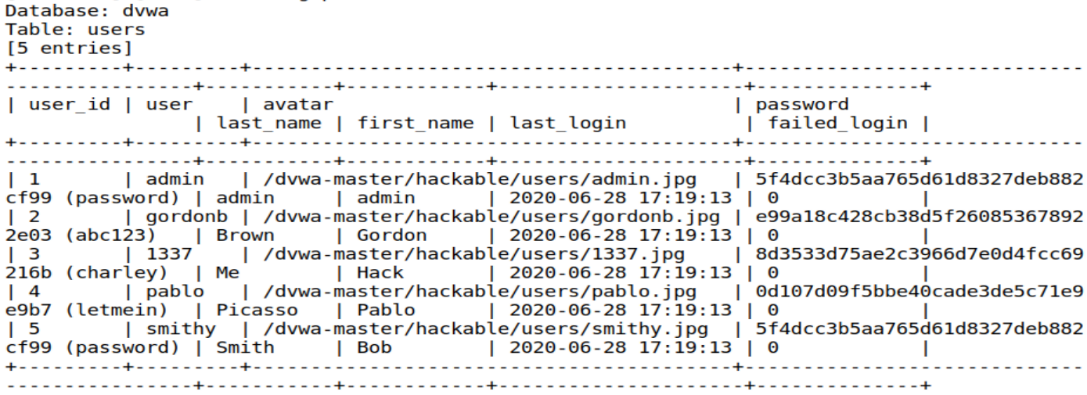
1. In the terminal, run SQLmap with the Target URL and the Cookie again, this time with the –D and --dump-all options to specify the dvwa database and to dump the contents of every table. The command is similar to the command in the previous exercise against the normal SQL Injection webpage.

Press Enter to use the default options when asked.

sudo sqlmap –u "http://192.168.13.100/dvwa-master/vulnerabilities /sqli\_blind/?id=3&Submit=Submit" --cookie="security=low; PHPSESSID=ab6iiicumjdbjkmun4cphkdmvo" –D dvwa --dump-all

Run in a single line and replace with your Win10 IP and your cookie value

1. When SQLmap completes, like the previous exercise, it will display the contents of the tables in the dvwa database. It even tries to crack the password hashes in the users table.



An extract of the SQLmap output showing the contents of the users table plus the cracked passwords

*End of Practical*