A logo of university of new haven

Description automatically generated

Performing SQL Injection to Manipulate Tables in a Database

Ethical Hacking & Lab 11

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name: Nishchal Sreevathsa |  |  |  |  |

Table of Contents

1. [Executive Summary 2](#_Toc409378035)

[Highlights](#_Toc409378036)

[Objectives](#_Toc409378037)

1. [Lab Description Details 2](#_Toc409378038)

[Include Steps Taken, Notes, & Screen Shots demonstrating completion of lab objectives](#_Toc409378039)

1. [Supporting Evidence 14](#_Toc409378040)
2. [Conclusion & Wrap-Up 16](#_Toc409378041)

[Summary with observations, Success & Failures, Challenges](#_Toc409378042)

# Executive Summary

## Highlights

|  |  |
| --- | --- |
|  | In this lab, you will use Kali Linux tools such as nmap and Metasploit to attack a MySQL database from the outside network. To locate the MySQL database running on port 3306, you must first search for open ports. Next, you will attempt to crack the admin password using the mysql\_login module in Metasploit. After gaining access with the admin credentials, you will look through the various tables and databases and take account and credit card information. In order to keep access to the system, you will lastly create a backdoor user account called "hacker" with admin privileges. Learning offensive security tools and techniques, such as port scanning, SQL injection, brute forcing logins, and building backdoors, is the aim. |

## Objectives

|  |  |
| --- | --- |
|  | This lab's goal is to teach you how to use nmap to scan networks, brute force the password and username of a MySQL database, and then use the database to your advantage to get in, steal information, and create a backdoor account. |

# Lab Description Details

## Include Steps Taken, Notes, & Screen Shots demonstrating completion of lab objectives

**Scanning**

**Step 1:** Accessed the external Kali Linux2 machine with IP Address 175.45.176.199 on the topology and entered **root** as username and **toor** as password.

**Step 2:** Use **Nmap** to find the open ports on the **www.campus.edu network**.

A computer screen shot of a program

Description automatically generated

**Step 3:** We now know that the SQL port is **3306**; to find the target port, run a service script scan on it.

A computer screen shot of a black and white screen

Description automatically generated

**Step 4:** Launch the Metasploit framework's **msfconsole** and start the **Postgresql service**.

A screenshot of a computer

Description automatically generated

**Challenge #1:** Using the msfconsole command line, type the **banner** command until the flag2 value appears. **Flag2: 776554**.

Screens screenshot of a computer screen

Description automatically generated

**Challenge #2:** Using the msfconsole command line, type the **banner** command until the flag3 value appears. **Flag3: 223444**.

A screenshot of a computer

Description automatically generated

**Step 5:** Look for the MySQL login tool.

A screen shot of a computer

Description automatically generated

**Step 6:** Next, make use of the MySQL Login Tool. Find out more about the MySQL Login Tool.

A screenshot of a computer

Description automatically generated

**Step 7:** Give the scanner permission to use blank passwords, and configure the USERNAME to be root and the remote host to 203.0.113.100.

A screen shot of a computer

Description automatically generated

**Step 8:** Once the password has been found, set the Password file to stop and view the options you have selected.

A screenshot of a computer

Description automatically generated

**Step 9:** After using the auxiliary module, shut it down.

A screenshot of a computer

Description automatically generated

**Connecting to the Database and Viewing the Tables**

**Step 10:** View the switches that are available for the mysql command.

A screenshot of a computer

Description automatically generated

**Step 11:** Check for open ports on the second firewall.

A screenshot of a computer program

Description automatically generated

**Step 12:** Use the show databases command to view the data in the databases.

**Challenge #3:** Captured the **flag: 334422** when entered the command show databases.

A screenshot of a computer

Description automatically generated

**Step 13:** Choose the database called **information\_schema**. Display the information\_schema database's tables.

A screenshot of a computer

Description automatically generated

**Step 14:** Look through every database. Choose the database for **DVWA**.

A screenshot of a computer

Description automatically generated

**Step 15:** Examine the dwwa database's tables.

A screenshot of a computer

Description automatically generated

**Step 16:** The **Metasploit database** should be selected. Examine the Metasploit database's tables.

A screenshot of a computer

Description automatically generated

**Step 17:** The **MySQL database** should be chosen. Examine the MySQL database's tables.

A screenshot of a computer

Description automatically generated

**Step 18:** Opt to use the **owasp10** database. Examine the owasp10 database's tables.

A screenshot of a computer

Description automatically generated

**Step 19:** Choose the database for **tikiwiki.** Examine the tikiwiki database's tables.

A screenshot of a computer

Description automatically generated

**Step 20:** Choose the database named **tikiwiki195**. Examine the tikiwiki195 database's tables.

A screenshot of a computer

Description automatically generated

**Step 21:** After looking through the contents of the different databases and the tables that went with them, it seemed that the information in the tables of the **owasp10 database**—which included information about accounts and credit cards—was especially interesting. Choose the **credit card** table using the owasp10.

A screenshot of a computer program

Description automatically generated

**Step 22:** View the columns and data in the **account table** by selecting it.

**Challenge #4:** Captured the **Flag5: 335553** when viewed the account database.

A screenshot of a computer

Description automatically generated

**Challenge #5:** Captured Flag6**: 223311** when viewed the account database.

A screenshot of a computer

Description automatically generated

**Step 23:** Make a User Hacker and give it complete access. Attempt to log in using the password **mypass123** and the newly created user "**Hacker**".

A screenshot of a computer

Description automatically generated

# Supporting Evidence

**Screenshots, Research, Etc.**

**A screenshot of a computer

Description automatically generated**

**A screenshot of a computer

Description automatically generated**

**A screenshot of a computer

Description automatically generated**

# Conclusion & Wrap-Up

## Summary with observations, Success & Failures, Challenges

Using offensive security tools and techniques to attack a MySQL database, participants in this lab gained practical experience. The lab offered useful experience in breaching a database with standard penetration testing instruments and methods. Identifying and correctly scanning services, exploring database contents, brute forcing login credentials, and building backdoors for persistent access are some of the most important lessons that have been learned. Enhanced defensive measures and surveillance could aid in thwarting such incursions.

**Observations:**

* An open port 3306, identified by Nmap scanning, indicated a MySQL database service.
* The admin password was successfully brute-forced by Metasploit's mysql\_login module.
* Investigating the databases revealed private client information in the tables.

**Success:**

* Used nmap scanning to find the MySQL port and service.
* Used Metasploit's password brute forcing feature to crack the admin password.
* Obtained admin credentials and used them to access the MySQL server.
* Discovered private client information in the accounts and credit card tables.
* Admin-level user account backdoor was created.

**Challenges:**

* Finding the admin password required a lot of brute forcing through a big password file.
* It was necessary to enumerate the databases and tables that held sensitive data.
* The attack might have been prevented with more detection and preventive measures in place.

**Risks:**

* Credential compromise is a risk associated with brute forcing passwords.
* SQL injection may enable unwanted access to and alteration of data.
* Exposure to sensitive customer data increases the possibility of identity theft or fraud.
* Recurring unapproved access is possible through backdoor accounts.

**Remediations:**

* Put multi-factor authentication and strong password policies into practice.
* To stop SQL injection, use prepared statements and input validation.
* Limit database access and encrypt critical fields.
* Identify and prevent frequent attempts at SQL injection.
* Review user access on a regular basis to look for backdoors.
* Close unused ports, such as 3306, and toughen up database servers.
* Configure attack log monitoring and intrusion detection.
* Educate people on security awareness regarding phishing and social engineering.
* To identify and resolve vulnerabilities, conduct routine penetration tests.