

SQL Injection Fundamentals

Introduction

Inlane freight has contracted our services to perform a security assessment of their public-facing web application, focusing on SQL injection vulnerabilities. The assessment was prompted by a recent breach experienced by one of their main competitors. This report details the findings from our grey box assessment, including the identification and exploitation of SQL injection vulnerabilities and the retrieval of a sensitive flag from the target system.

Methodology

The assessment was conducted using a structured approach, following the skills and techniques covered in the Hack The Box SQL Injection Fundamentals module. The steps included:

Enumeration and Information Gathering

SQL Injection Testing and Exploitation

Retrieving the Flag

Step 1: Enumeration and Information Gathering

The initial phase involved identifying the target web application's structure and potential points of SQL injection vulnerability.

Accessing the Web Application: Using a web browser, we navigated to the target IP address and explored the web application's functionalities.

Identifying Input Fields: We identified various input fields, such as login forms, search bars, and URL parameters, that could be potential vectors for SQL injection attacks.

Step 2: SQL Injection Testing and Exploitation

We conducted SQL injection tests on the identified input fields to determine if they were vulnerable to SQL injection attacks.

Manual Testing: Inputting common SQL injection payloads, such as ' OR '1'='1 and '; DROP TABLE users; --, into the fields to observe the web application's response.

Automated Testing: Using tools like SQLmap to automate the detection of SQL injection vulnerabilities.

Vulnerable Input Field: The login form was found to be vulnerable to SQL injection.

Database Information: Extracted information about the database, including the names of databases, tables, and columns.

Step 3: Retrieving the Flag

The final phase involved exploiting the identified SQL injection vulnerability to gain access to the underlying system and retrieve the flag.

Remote Code Execution: Using SQL injection to execute arbitrary commands on the server.

Navigating the File System: Once a shell was obtained, we navigated the file system to locate the flag in the /root directory.

Waiting to start...

Enable step-by-step solutions for all questions

Questions

Answer the question(s) below to complete this Section and earn cubes!

Target(s): [Click here to spawn the target system!](#)

+2 Assess the web application and use a variety of techniques to gain remote code execution and find a flag in the / root directory of the file system. Submit the contents of the flag as your answer.

528d6d9cedc2c7aab146ef226e918396

Submit

Hint

Previous

+10 Streak pts

Finish

Activate Windows

Go to Settings to activate Windows.

Powered by HACKTHEBOX

Breaking news
The Supreme Co...

Search

academy.hackthebox.com/module/33/section/794

```
mysql_stmt_bind_param($stmt, 'ss', $username, $password);
mysql_stmt_execute($stmt);
$result = mysql_stmt_get_result($stmt);

$row = mysql_fetch_array($result);
mysql_stmt_close($stmt);
<SNIP>
```

The query is modified to contain two placeholders, marked with ? where the username and password will be placed. We then bind the username and password to the query using the `mysql_stmt_bind_param()` function. This will safely escape any quotes and place the values in the query.

Conclusion

The list above is not exhaustive, and it could still be possible to exploit SQL injection based on the application logic. The code examples shown are based on PHP, but the logic applies across all common languages and libraries.

Previous

Next

+10 Streak pts

Mark Complete & Next

Activate Windows

Go to Settings to activate Windows.

Powered by HACKTHEBOX

63°F
Mostly clear

Search

ENG
UK

10:13 AM
6/28/2024

Waiting to start...

Success

Congratulations! You earned 1 cubes!

Enable step-by-step solutions for all questions

Questions

Answer the question(s) below to complete this Section and earn cubes!

Target(s): [Click here to spawn the target system!](#)

+1

Find the flag by using a webshell.

d2b5b27ae688b6a0f1d21b7d3a0798cd

Submit

Hint

Previous

Next

+10 Streak pts

Mark Complete & Next

Activate Windows

Go to Settings to activate Windows.

Powered by HACKTHEBOX

65°F

Mostly clear

Search

ENG UK

10:12 AM

6/28/2024

Waiting to start...

Enable step-by-step solutions for all questions

Questions

Answer the question(s) below to complete this Section and earn cubes!

Target(s): [Click here to spawn the target system!](#)

+1

We see in the above PHP code that '\$conn' is not defined, so it must be imported using the PHP include command. Check the imported page to obtain the database password.

dB_pAasw0rd_IS_flag!

Submit

Hint

Previous

Next

+10 Streak pts

Mark Complete & Next

Activate Windows

Go to Settings to activate Windows.

Powered by HACKTHEBOX

66°F

Mostly clear

Search

ENG UK

10:11 AM

6/28/2024

cs-sa07-24019
John_Mbithi_Mutave

Waiting to start...

Success
Congratulations! You earned 1 cubes!

☐ Enable step-by-step solutions for all questions

Questions

Answer the question(s) below to complete this Section and earn cubes!

Target(s): [Click here to spawn the target system!](#)

+1 🧊 What is the password hash for 'newuser' stored in the 'users' table in the 'ilfreight' database?

9da2c9bcd139d8610954e0e11ea8f45f

Submit

PreviousNext

+10 Streak pts

Mark Complete & Next

Activate Windows
Go to Settings to activate Windows.
Powered by HACKTHEBOX

Waiting to start...

☐ Enable step-by-step solutions for all questions

Questions

Answer the question(s) below to complete this Section and earn cubes!

Target(s): [Click here to spawn the target system!](#)

+0 🧊 Use a Union injection to get the result of 'user()'

root@localhost

SubmitHint

PreviousNext

+10 Streak pts

Mark Complete & Next

Activate Windows
Go to Settings to activate Windows.
Powered by HACKTHEBOX

cs-sa07-24019
John_Mbithi_Mutave

academy.hackthebox.com/module/33/section/806

Enable step-by-step solutions for all questions

Success

Congratulations! You earned 1 cubes!

Questions

Answer the question(s) below to complete this Section and earn cubes!

Cheat Sheet

Target(s): [Click here to spawn the target system!](#)

Authenticate with user "root" and password "password"

+ 1

Connect to the above MySQL server with the 'mysql' tool, and find the number of records returned when doing a 'Union' of all records in the 'employees' table and all records in the 'departments' table.

663

Submit

Hint

Previous

Next

+10 Streak pts

Mark Complete & Next

Activate Windows

Go to Settings to activate Windows.

Powered by HACKTHEBOX

66°F

Mostly clear

Search

ENG UK

10:09 AM

6/28/2024

academy.hackthebox.com/module/33/section/799

Waiting to start...

Success

Congratulations! You earned 1 cubes!

Enable step-by-step solutions for all questions

Questions

Answer the question(s) below to complete this Section and earn cubes!

Cheat Sheet

Target(s): [Click here to spawn the target system!](#)

+ 1

Login as the user with the id 5 to get the flag.

cdad9ecd6f14b45ff5c4de32909caec

Submit

Hint

Previous

Next

+10 Streak pts

Mark Complete & Next

Activate Windows

Go to Settings to activate Windows.

Powered by HACKTHEBOX

66°F

Mostly clear

Search

ENG UK

10:08 AM

6/28/2024

Waiting to start...

Success

Congratulations! You earned 1 cubes!

Enable step-by-step solutions for all questions

Questions

Cheat Sheet

Answer the question(s) below to complete this Section and earn cubes!

Target(s): [Click here to spawn the target system!](#)

+1

Try to log in as the user 'tom'. What is the flag value shown after you successfully log in?

202a1d1a8b195d5e9a57e434cc16000c

SubmitHint

PreviousNext

+10 Streak ptsMark Complete & Next

Activate Windows

Go to Settings to activate Windows.

Powered by HACKTHEBOX

66°F Mostly clear

Search

ENG UK10:07 AM6/28/2024

academy.hackthebox.com/module/33/section/193

when we can get the **PHP** or **SQL** errors in the front-end, and so we may intentionally cause an SQL error that returns the output of our query.

In more complicated cases, we may not get the output printed, so we may utilize SQL logic to retrieve the output character by character. This is known as **Blind SQL injection**, and it also has two types: **Boolean Based** and **Time Based**.

With **Boolean Based** SQL injection, we can use SQL conditional statements to control whether the page returns any output at all, 'i.e., original query response,' if our conditional statement returns **true**. As for **Time Based** SQL injections, we use SQL conditional statements that delay the page response if the conditional statement returns **true** using the **Sleep()** function.

Finally, in some cases, we may not have direct access to the output whatsoever, so we may have to direct the output to a remote location, 'i.e., DNS record,' and then attempt to retrieve it from there. This is known as **Out-of-band** SQL injection.

In this module, we will only be focusing on introducing SQL injections through learning about **Union Based** SQL injection.

PreviousNext

+10 Streak ptsMark Complete & Next

Activate Windows

Go to Settings to activate Windows.

Powered by HACKTHEBOX

66°F Mostly clear

Search

ENG UK10:07 AM6/28/2024

cs-sa07-24019
John_Mbithi_Mutave

Waiting to start...

☐ Enable step-by-step solutions for all questions

Questions

Answer the question(s) below to complete this Section and earn cubes!

Target(s): [Click here to spawn the target system!](#)

Authenticate with user "root" and password "password"

+1 In the 'titles' table, what is the number of records WHERE the employee number is greater than 10000 OR their title does NOT contain 'engineer'?

654

Submit

Hint

Previous

Next

+10 Streak pts

Mark Complete & Next

Activate Windows
Go to Settings to activate Windows.
Powered by HACKTHEBOX

Waiting to start...

☐ Enable step-by-step solutions for all questions

Questions

Answer the question(s) below to complete this Section and earn cubes!

Target(s): [Click here to spawn the target system!](#)

Authenticate with user "root" and password "password"

+1 What is the last name of the employee whose first name starts with "Bar" AND who was hired on 1990-01-01?

Mitchem

Submit

Hint

Previous

Next

+10 Streak pts

Mark Complete & Next

Activate Windows
Go to Settings to activate Windows.
Powered by HACKTHEBOX

cs-sa07-24019
John_Mbithi_Mutave

Waiting to start...

☐ Enable step-by-step solutions for all questions

Questions

Answer the question(s) below to complete this Section and earn cubes!

Target(s): [Click here to spawn the target system!](#)

Authenticate with user "root" and password "password"

+0 What is the department number for the 'Development' department?

d005

Submit Hint

Previous Next

+10 Streak pts Mark Complete & Next

Activate Windows
Go to Settings to activate Windows.
Powered by HACKTHEBOX

66°F Mostly clear

Search

ENG UK 10:01 AM 6/28/2024

Waiting to start...

☐ Enable step-by-step solutions for all questions

Questions

Answer the question(s) below to complete this Section and earn cubes!

Target(s): [Click here to spawn the target system!](#)

Authenticate with user "root" and password "password"

+0 What is the department number for the 'Development' department?

d005

Submit Hint

Previous Next

+10 Streak pts Mark Complete & Next

Activate Windows
Go to Settings to activate Windows.
Powered by HACKTHEBOX

66°F Mostly clear

Search

ENG UK 10:00 AM 6/28/2024

Waiting to start...

Enable step-by-step solutions for all questions

Questions

Answer the question(s) below to complete this Section and earn cubes!

Target(s): [Click here to spawn the target system!](#)

Authenticate with user `"root"` and password `"password"`

+0 Connect to the database using the MySQL client from the command line. Use the `'show databases;'` command to list databases in the DBMS. What is the name of the first database?

`employees`

Submit

Hint

Previous

Next

+10 Streak pts

Mark Complete & Next

Activate Windows

Go to Settings to activate Windows.

Powered by HACKTHEBOX

66°F Mostly clear

Search

ENG UK 9:59 AM 6/28/2024

academy.hackthebox.com/module/33/section/182

```
"date": "02-01-2021",
"content": "This is the first post on this web app."
},
"100003": {
  "date": "02-01-2021",
  "content": "Reminder: Tomorrow is the ..."
}
}
```

It looks similar to a dictionary item in languages like `Python` or `PHP` (i.e. `{'key': 'value'}`), where the `key` is usually a string, and the `value` can be a string, dictionary, or any class object.

The most common example of a NoSQL database is `MongoDB`.

Non-relational Databases have a different method for injection, known as NoSQL injections. SQL injections are completely different than NoSQL injections. NoSQL injections will be covered in a later module.

Previous

Next

+10 Streak pts

Mark Complete & Next

Almost There! BETA

10 Streak pts left to meet The Weekly Goal.

20 / 30 Streak pts earned

Activate Windows

Go to Settings to activate Windows.

Powered by HACKTHEBOX

Finance headline IMF Warns: U.S...

Search

ENG UK 9:48 AM 6/28/2024

cs-sa07-24019
John_Mbithi_Mutave

The screenshot displays a web browser window with the URL `academy.hackthebox.com/module/33/section/178`. The page content includes a diagram of a three-tier architecture and a streak challenge.

Diagram: Client Application

```
graph LR
    User --> TierI[Tier I]
    TierI --> TierII[Tier II]
    TierII --> DB[Database Administrator]
```

Tier I usually consists of client-side applications such as websites or GUI programs. These applications consist of high-level interactions such as user login or commenting. The data from these interactions is passed to **Tier II** through API calls or other requests.

The second tier is the middleware, which interprets these events and puts them in a form required by the DBMS. Finally, the application layer uses specific libraries and drivers based on the type of DBMS to interact with them. The DBMS receives queries from the second tier and performs the requested operations. These operations could include insertion, retrieval, deletion, or updating of data. After processing, the DBMS returns any requested data or error codes in the event of invalid queries.

It is possible to host the application server as well as the DBMS on the same host. However, databases with large amounts of data supporting many users are typically hosted separately to improve performance and scalability.

Streak Challenge:

- 7 streaks in danger (BETA)
- Get 20 Streak pts more before next Monday to keep your Streak alive.
- 10 / 30 Streak pts earned
- +10 Streak pts
- Mark Complete & Next

Section 177:

injections cause many password and data breaches against websites, which are then re-used to steal user accounts, access other services, or perform other nefarious actions.

Another use case of SQL injection is to subvert the intended web application logic. The most common example of this is bypassing login without passing a valid pair of username and password credentials. Another example is accessing features that are locked to specific users, like admin panels. Attackers may also be able to read and write files directly on the back-end server, which may, in turn, lead to placing back doors on the back-end server, and gaining direct control over it, and eventually taking control over the entire website.

Prevention

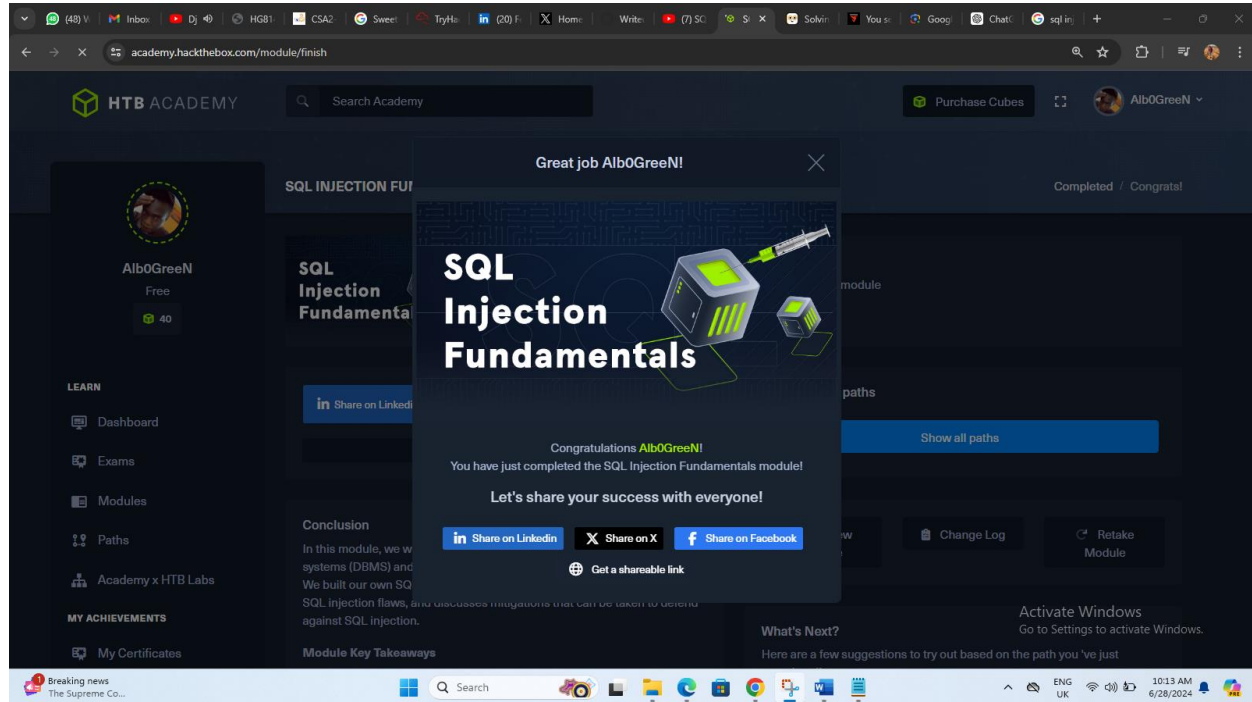
SQL injections are usually caused by poorly coded web applications or poorly secured back-end server and databases privileges. Later on, we will discuss ways to reduce the chances of being vulnerable to SQL injections through secure coding methods like user input sanitization and validation and proper back-end user privileges and control.

Next +10 Streak pts Mark Complete & Next

Activate Windows
Go to Settings to activate Windows.
Powered by HACKTHEBOX

cs-sa07-24019

John_Mbithi_Mutave



Shareable Link - <https://academy.hackthebox.com/achievement/1296187/33>

Conclusion

The security assessment of Inlanefreight's web application revealed a critical SQL injection vulnerability in the login form. This vulnerability could potentially allow attackers to:

- Extract sensitive information from the database.

- Execute arbitrary commands on the server.

- Compromise the entire web application and underlying system.

Recommendations

To mitigate the risk of SQL injection attacks, we recommend the following measures:

Input Validation: Implement strong input validation and sanitization for all user inputs.

Parameterized Queries: Use parameterized queries and prepared statements to interact with the database.

Web Application Firewall: Deploy a web application firewall (WAF) to filter and monitor incoming traffic for malicious activity.

Regular Security Audits: Conduct regular security audits and penetration tests to identify and address vulnerabilities promptly.

By implementing these measures, Inlanefreight can significantly enhance the security of their web application and protect against SQL injection attacks.