cs-sa07-24019
John_Mbithi_Mutave

SQL Injection Fundamentals

Introduction

Inlanefreight 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.

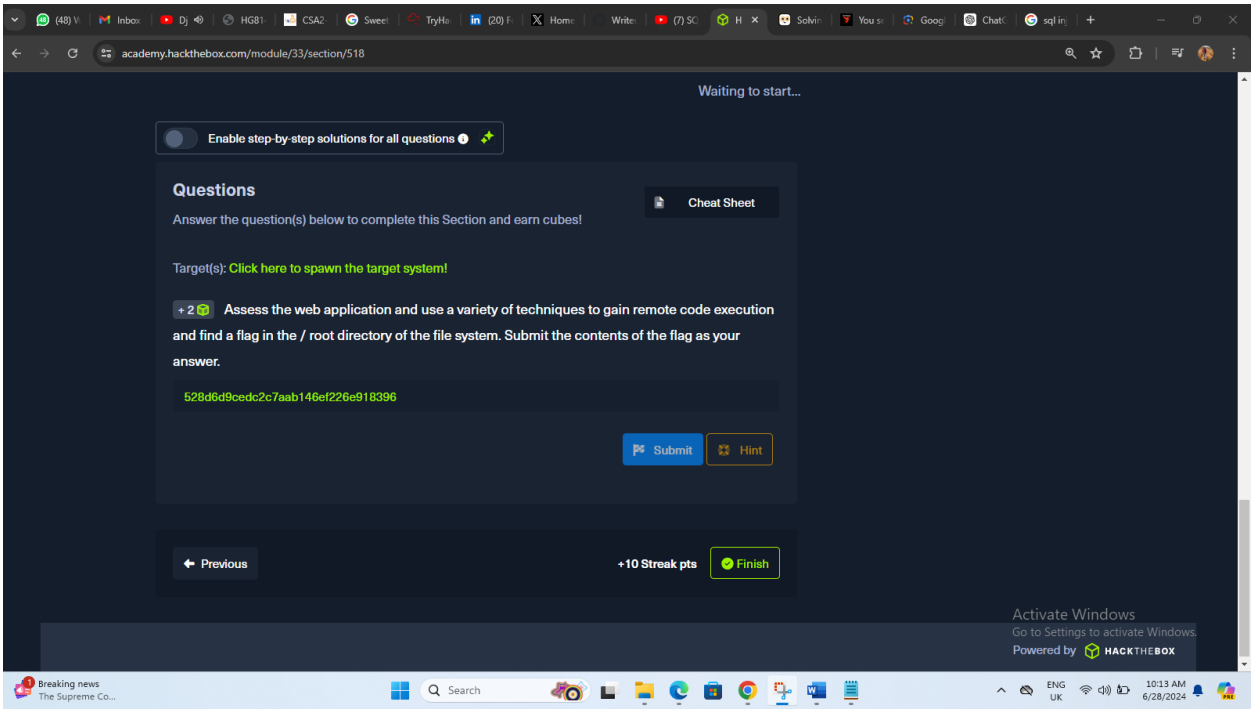
Step 3: Retrieving the Flag

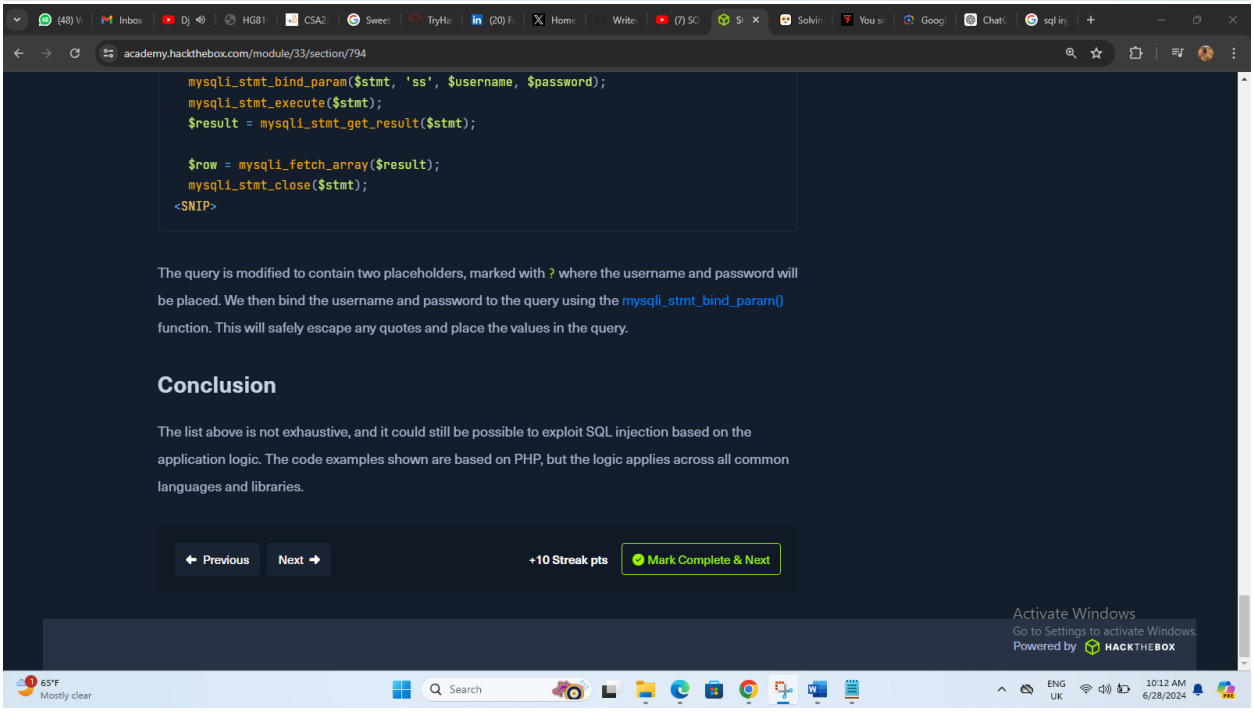
The final phase involved exploiting the identified SQL injection vulnerability to gain access to the underlying system and retrieve the flag.

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

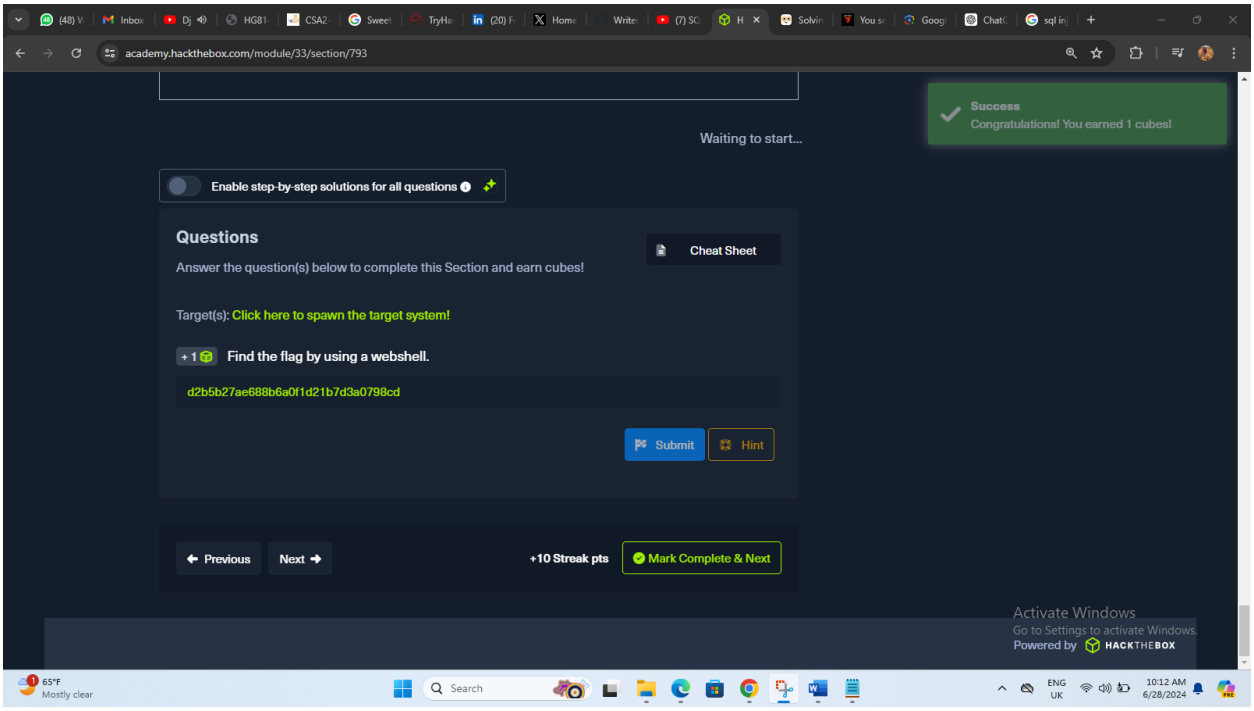
Navigating the File System: Once a shell was obtained, we navigated the file system to locate the flag in the /root directory.

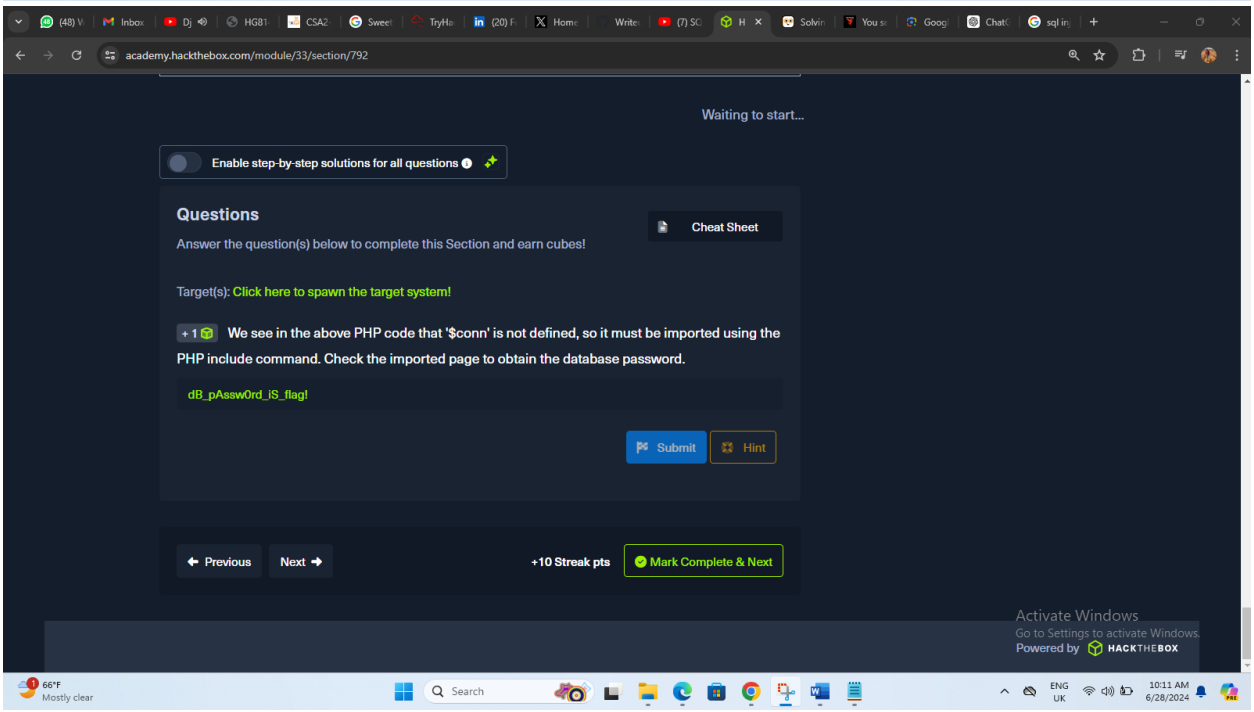
cs-sa07-24019
John_Mbithi_Mutave



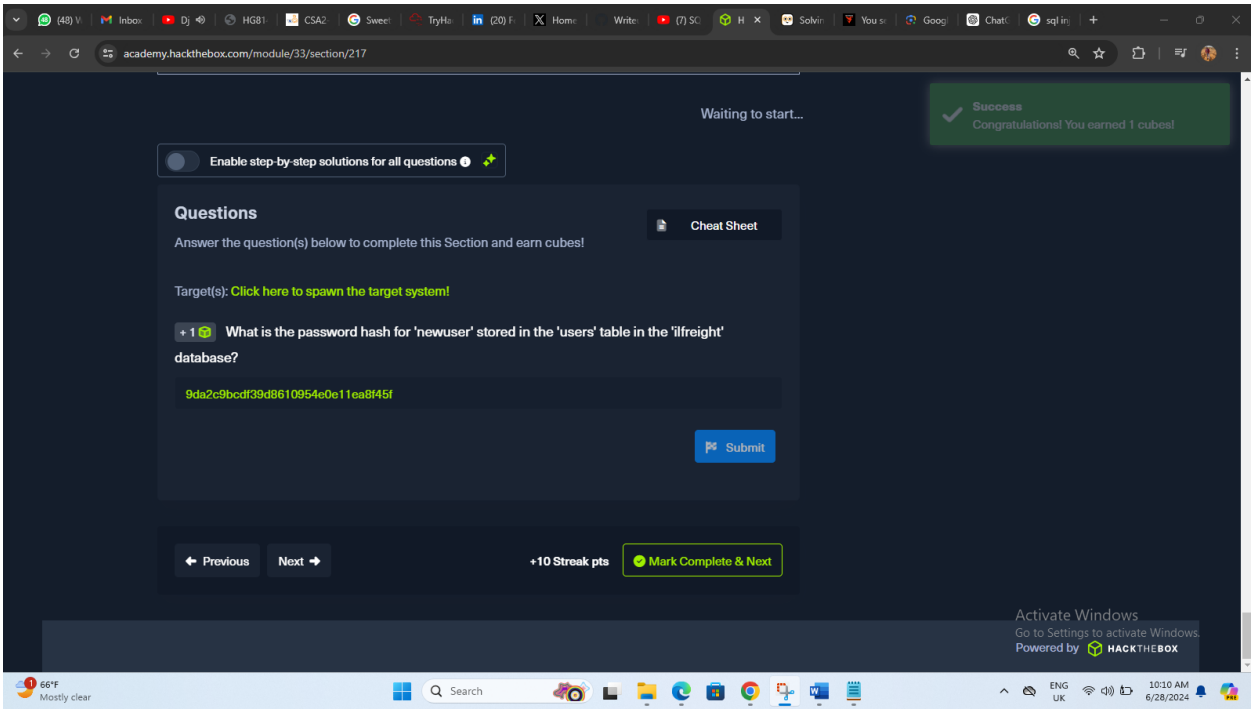


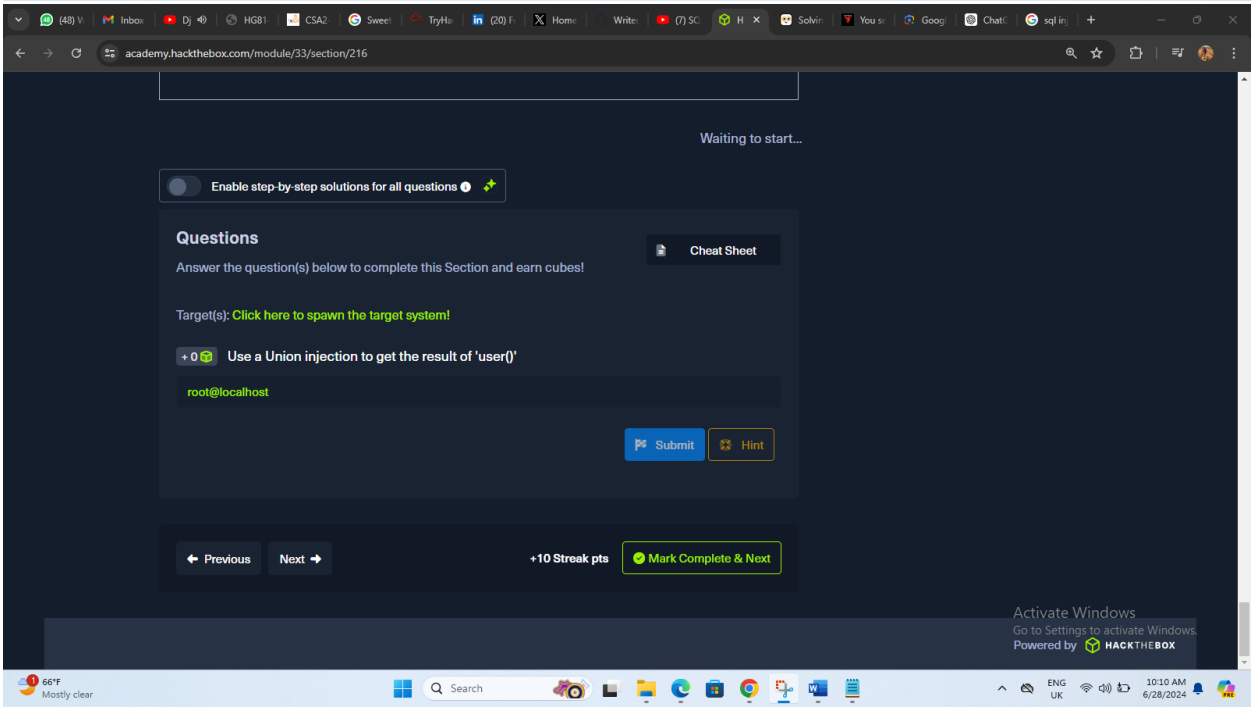
cs-sa07-24019
John_Mbithi_Mutave



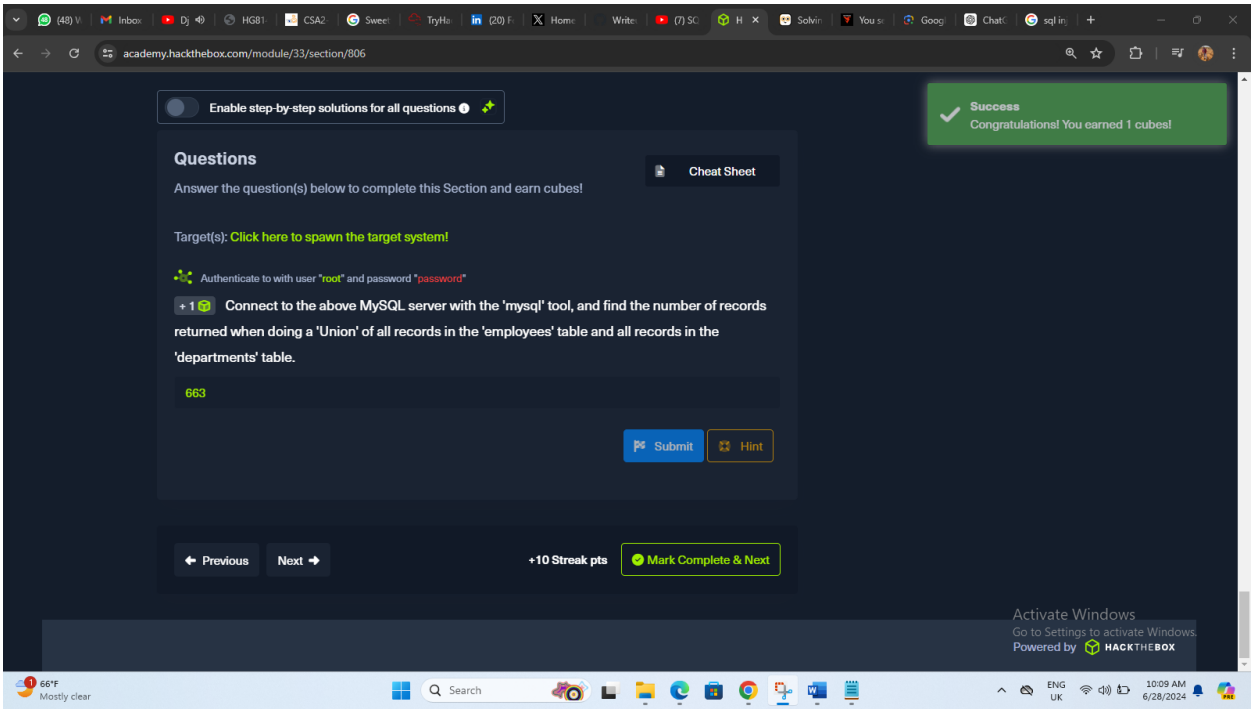


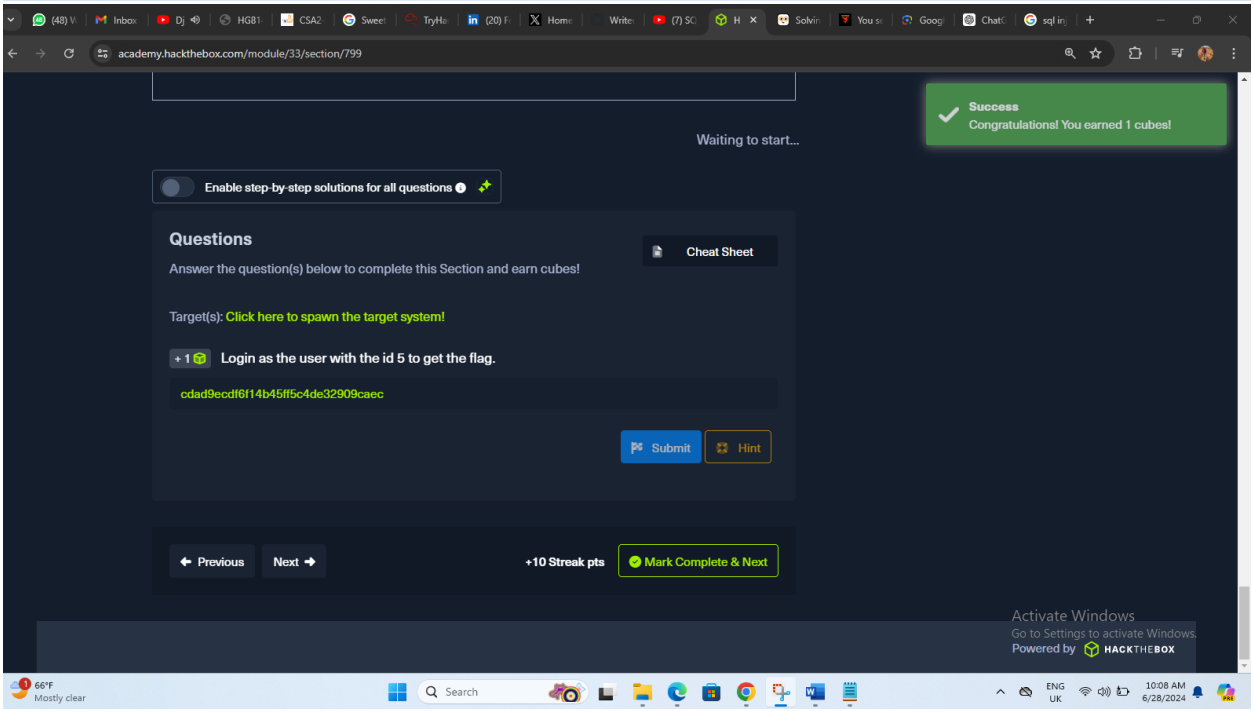
cs-sa07-24019
John_Mbithi_Mutave



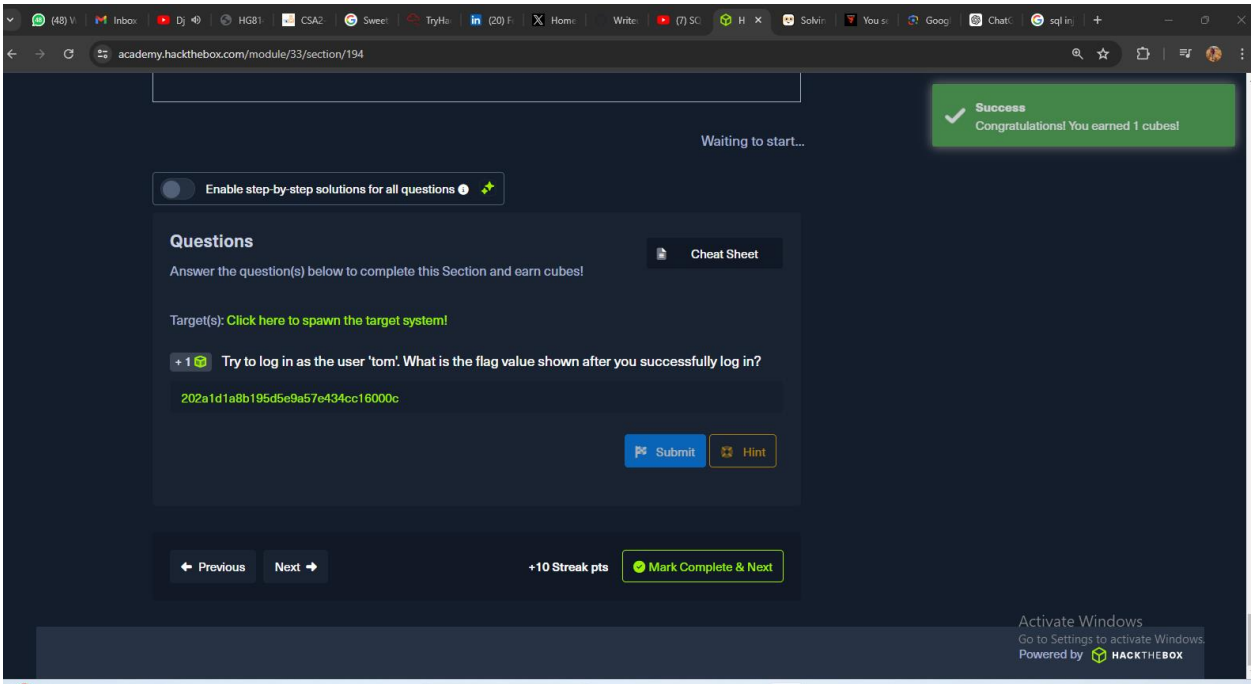


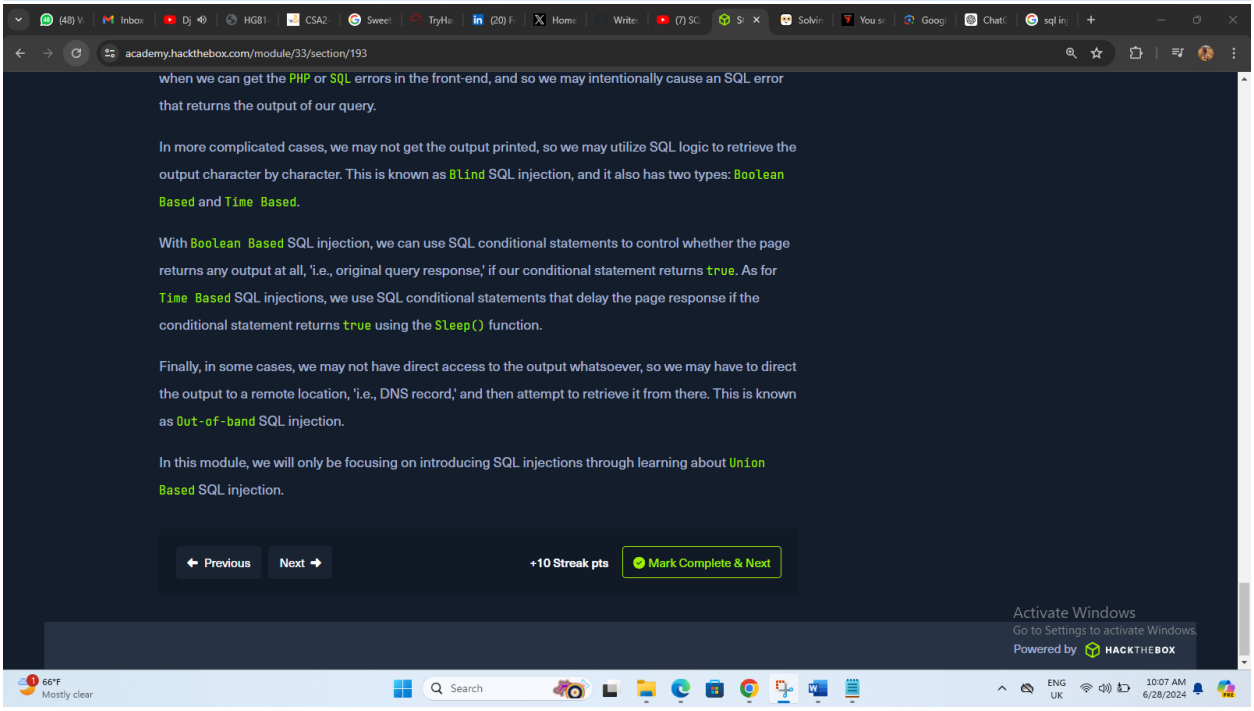
cs-sa07-24019
John_Mbithi_Mutave



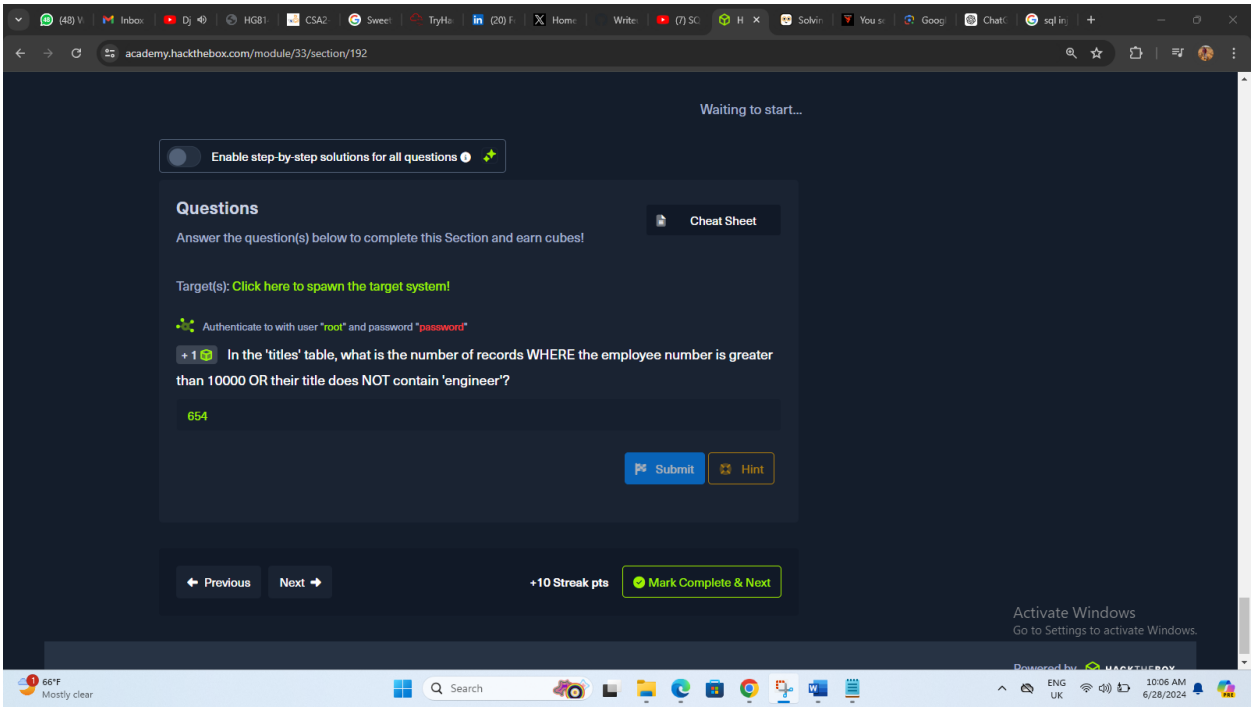


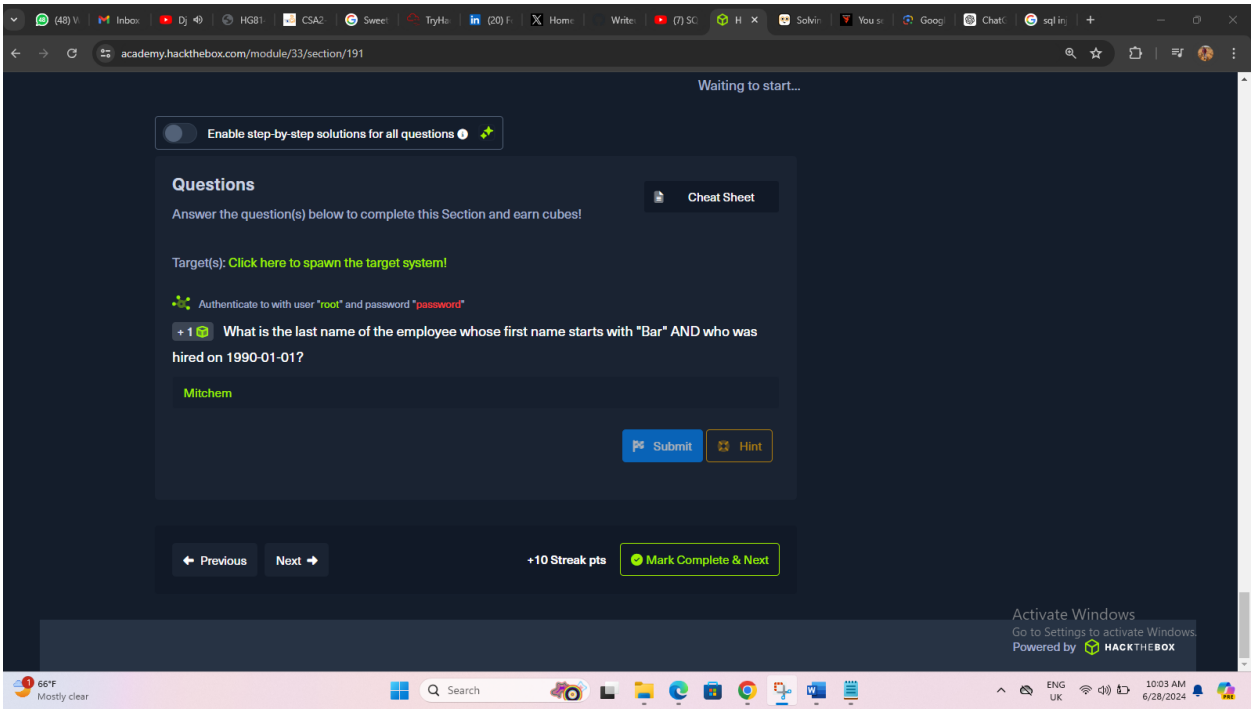
cs-sa07-24019
John_Mbithi_Mutave



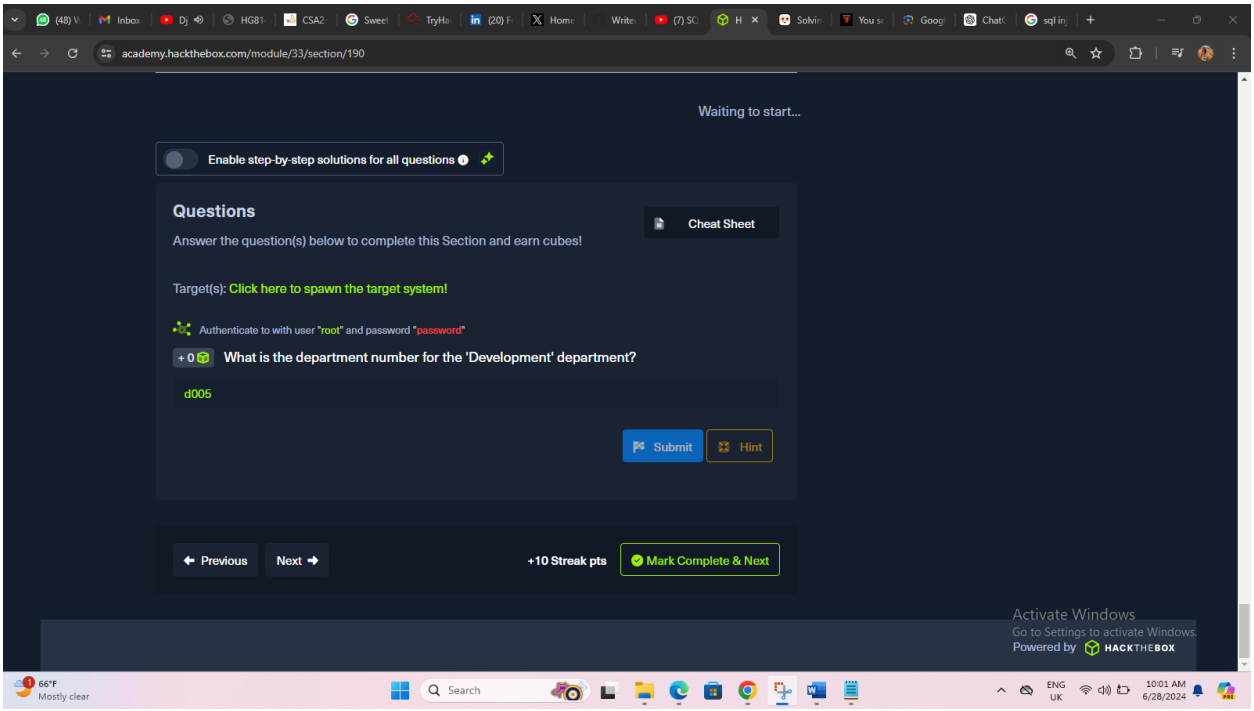


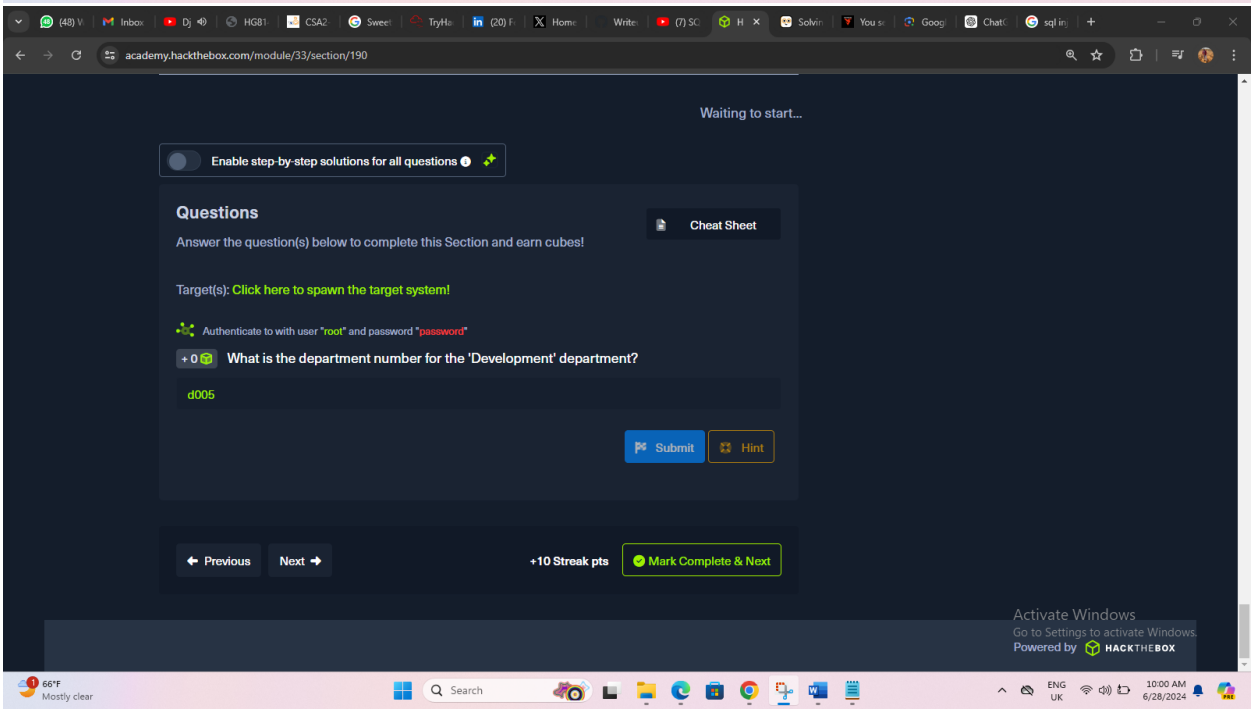
cs-sa07-24019
John_Mbithi_Mutave



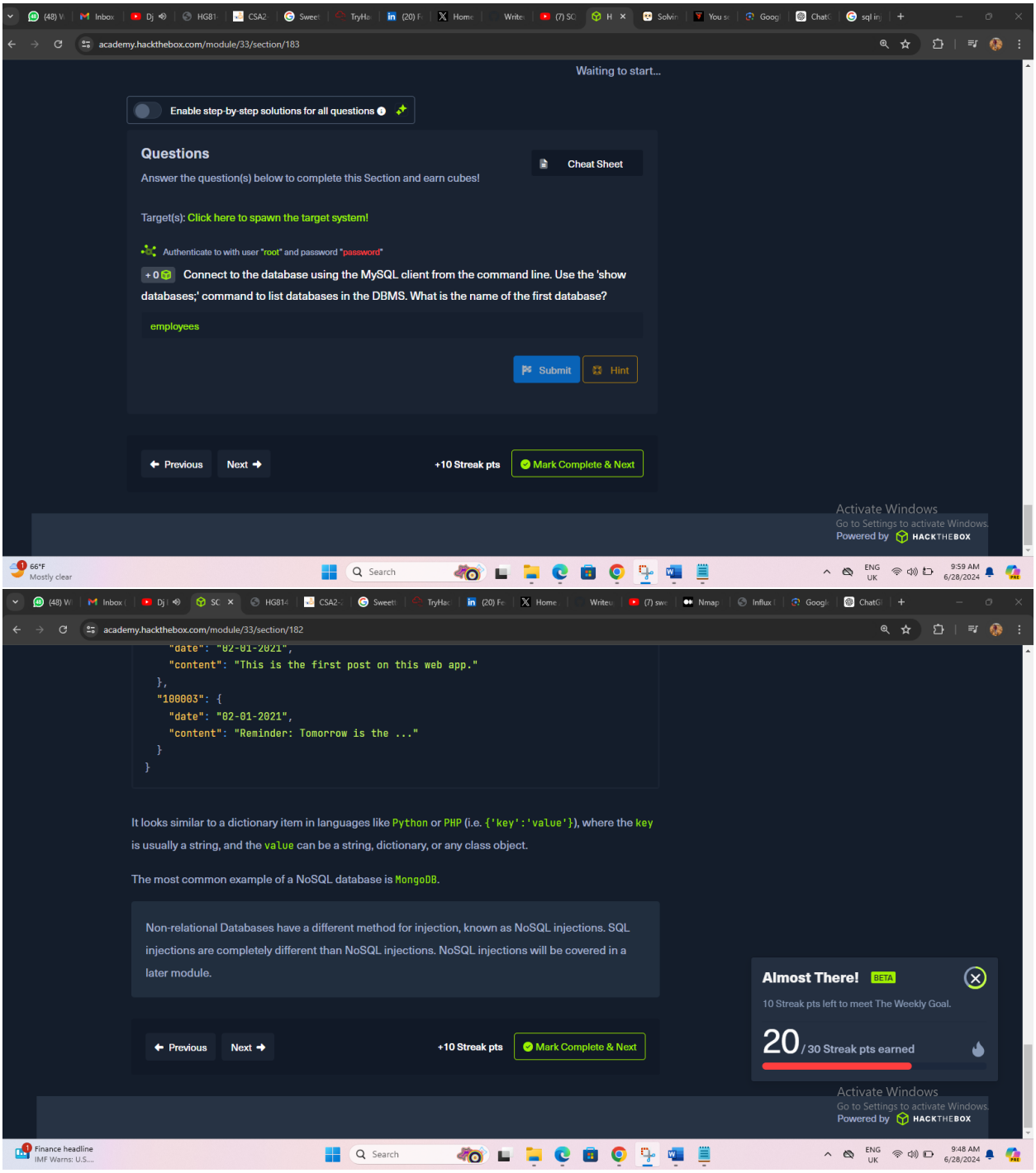


cs-sa07-24019
John_Mbithi_Mutave

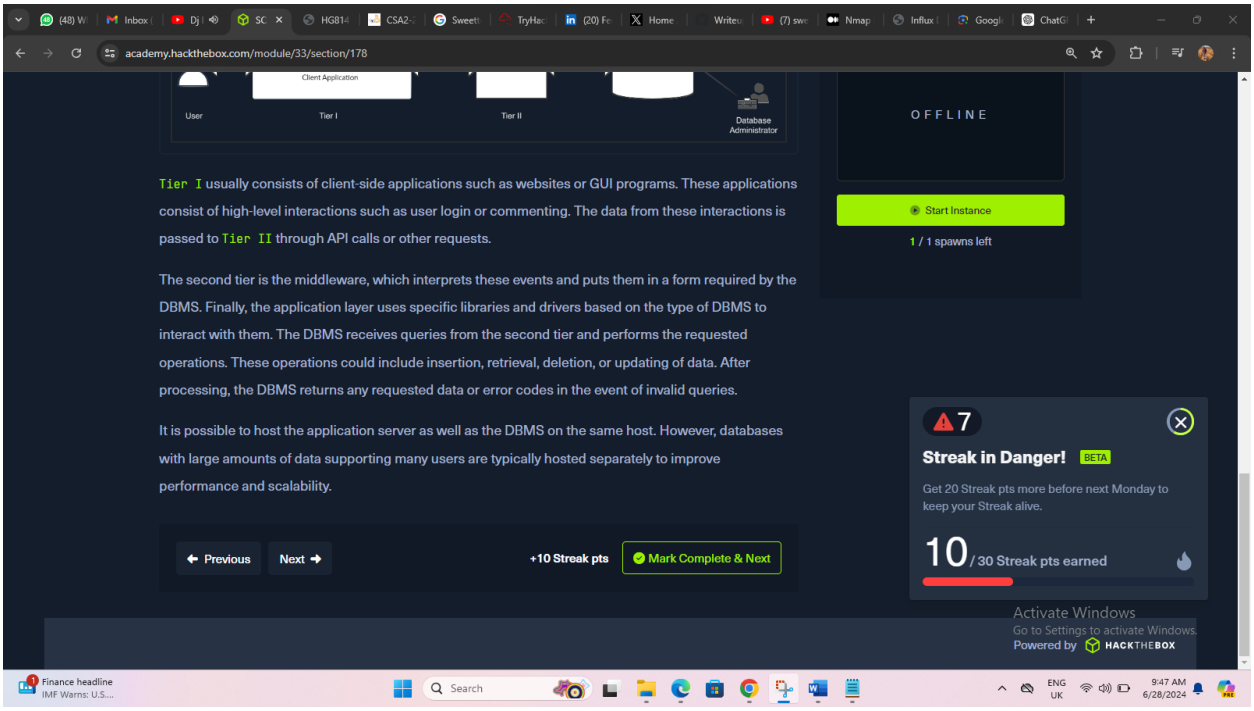


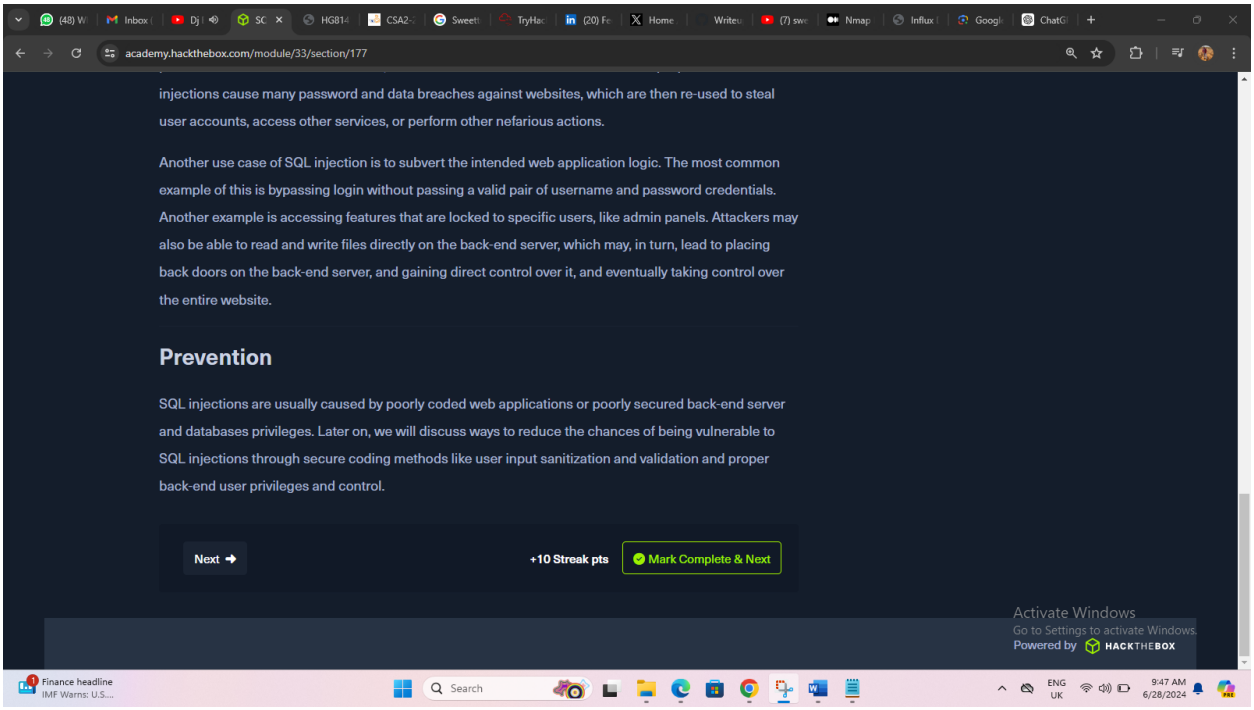


cs-sa07-24019
John_Mbithi_Mutave

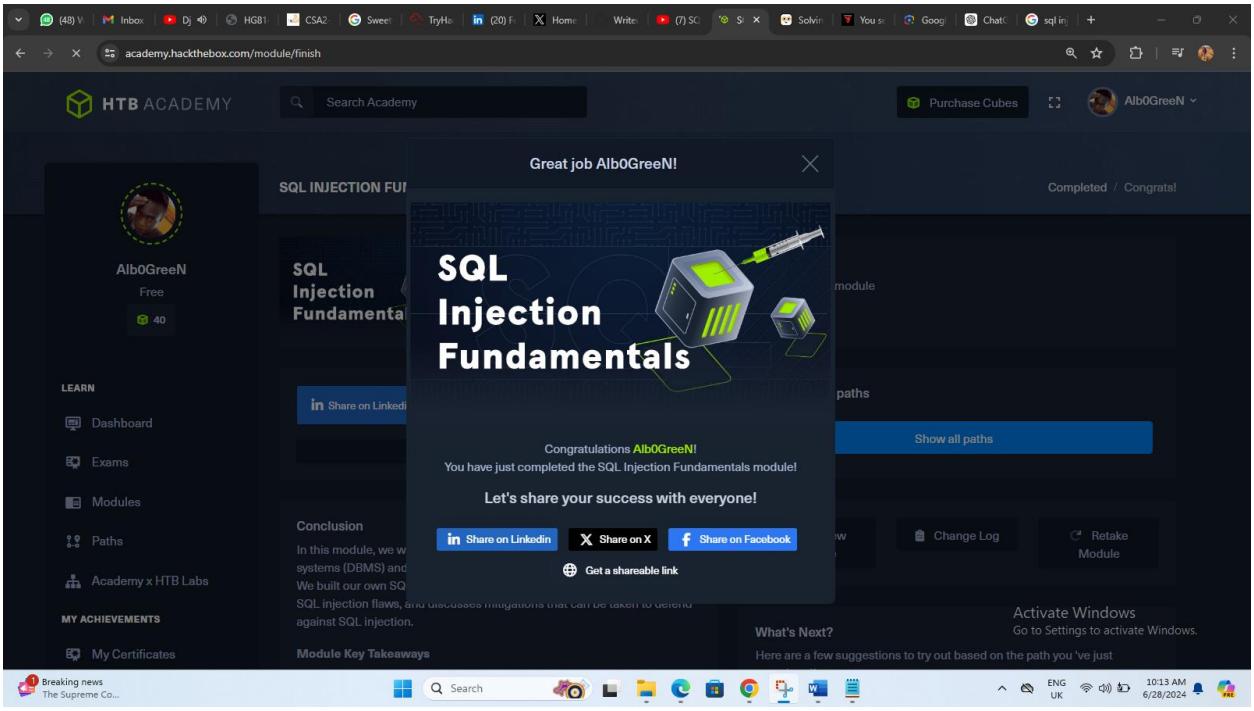


cs-sa07-24019
John_Mbithi_Mutave





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.