Week 3 Assignment 2

Course: Cloud and Network Security - C1-2026

Student Name: Bussllus Bertrand

Student Number: CS-CNS11-26004

Tuesday, February 3, 2026

Class Exercise: Web Requests

Contents

Week 3 Assignment 2

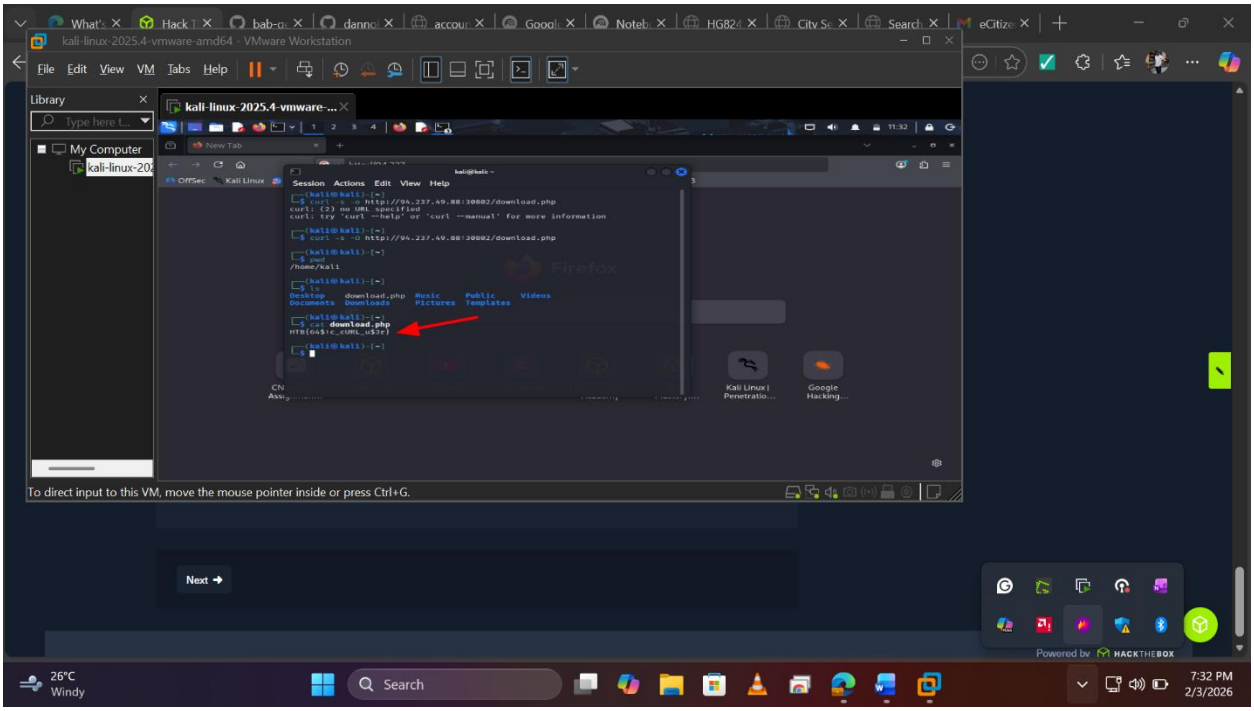
Introduction

This report provides a detailed overview of the Web Requests module, which is designed to cover the fundamental concepts necessary for understanding web application interactions. The following sections examine the critical tools used to analyze these interactions, specifically curl and the Browser DevTools. Additionally, the report explores the essential protocols of the web, including the HyperText Transfer Protocol (HTTP) and Hypertext Transfer Protocol Secure (HTTPS), while breaking down the mechanics of HTTP requests, responses, and headers

Questions and Answers

HyperText Transfer Protocol (HTTP)

To get the flag, start the above exercise, then use cURL to download the file returned by '/download.php' in the server shown above. HTB{64$!c_cURL_u$3r}

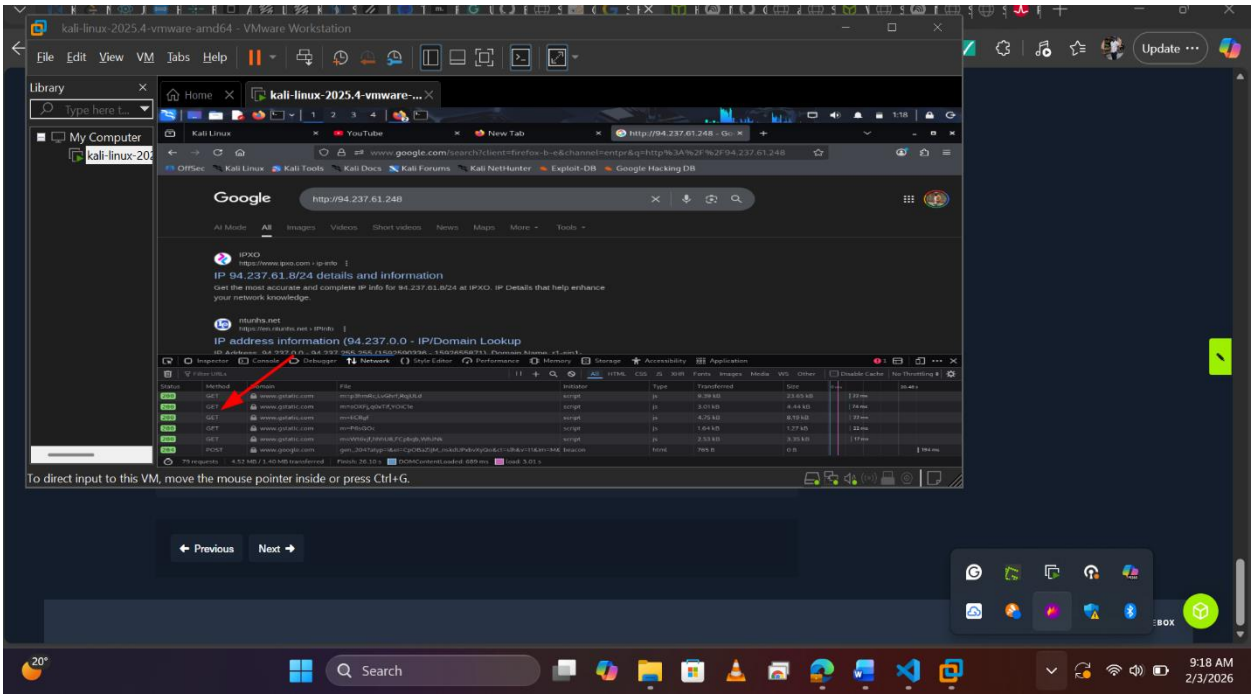


Week 3 Assignment 2

HTTP Requests and Responses

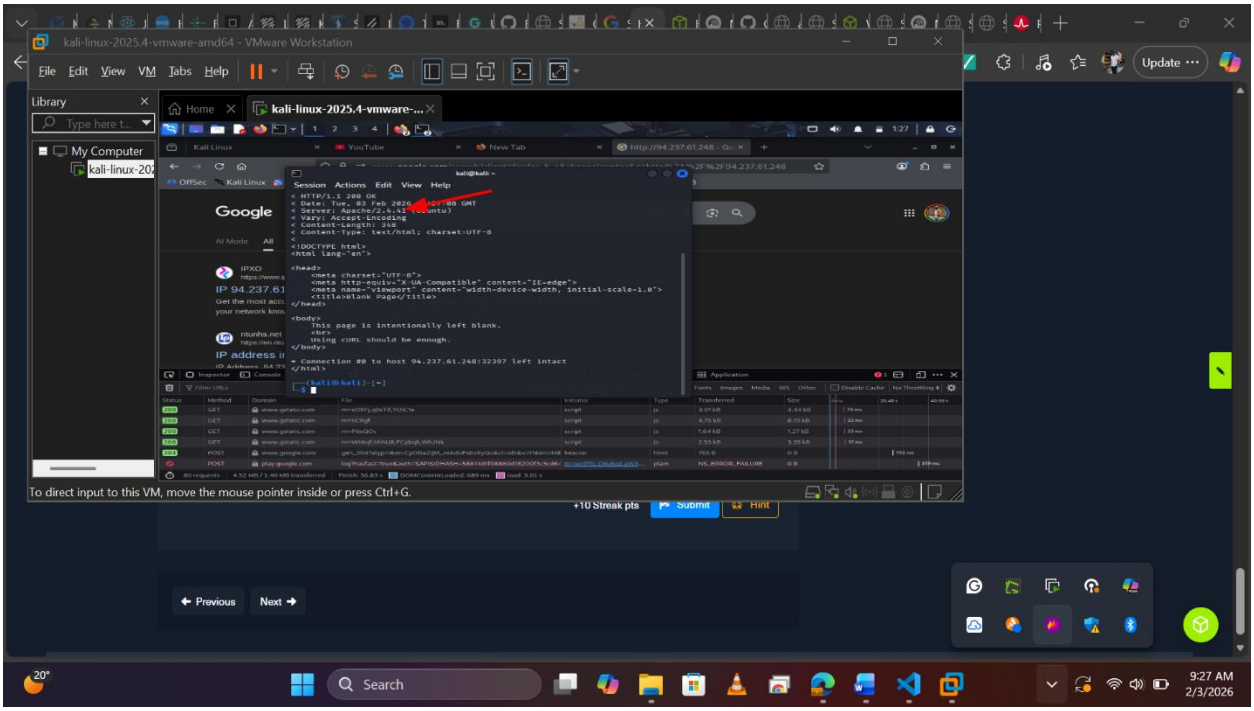
What is the HTTP method used while intercepting the request? (case-sensitive)

GET



Send a GET request to the above server, and read the response headers to find the version of Apache running on the server, then submit it as the answer. (answer format: X.Y.ZZ)**2.4.41**

Week 3 Assignment 2

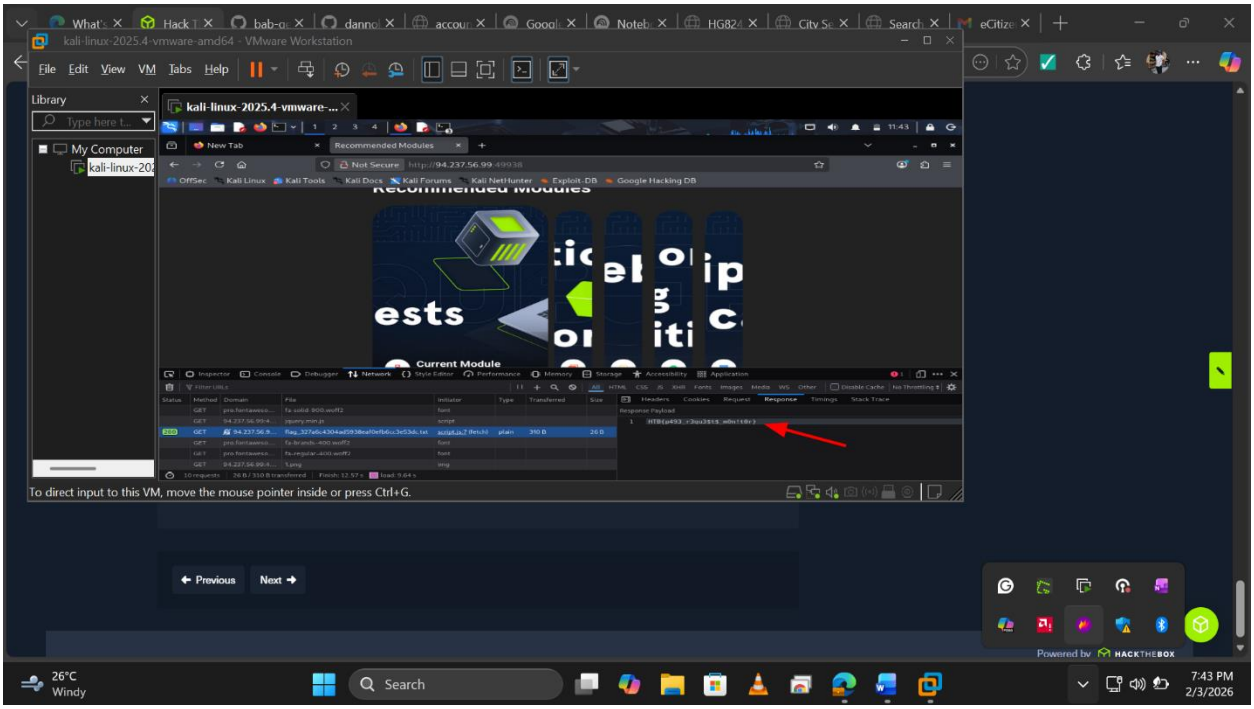


HTTP Headers

The server above loads the flag after the page is loaded. Use the Network tab in the browser devtools to see what requests are made by the page, and find the request to the flag.

HTB{p493_r3qu3t_m0n!t0r}

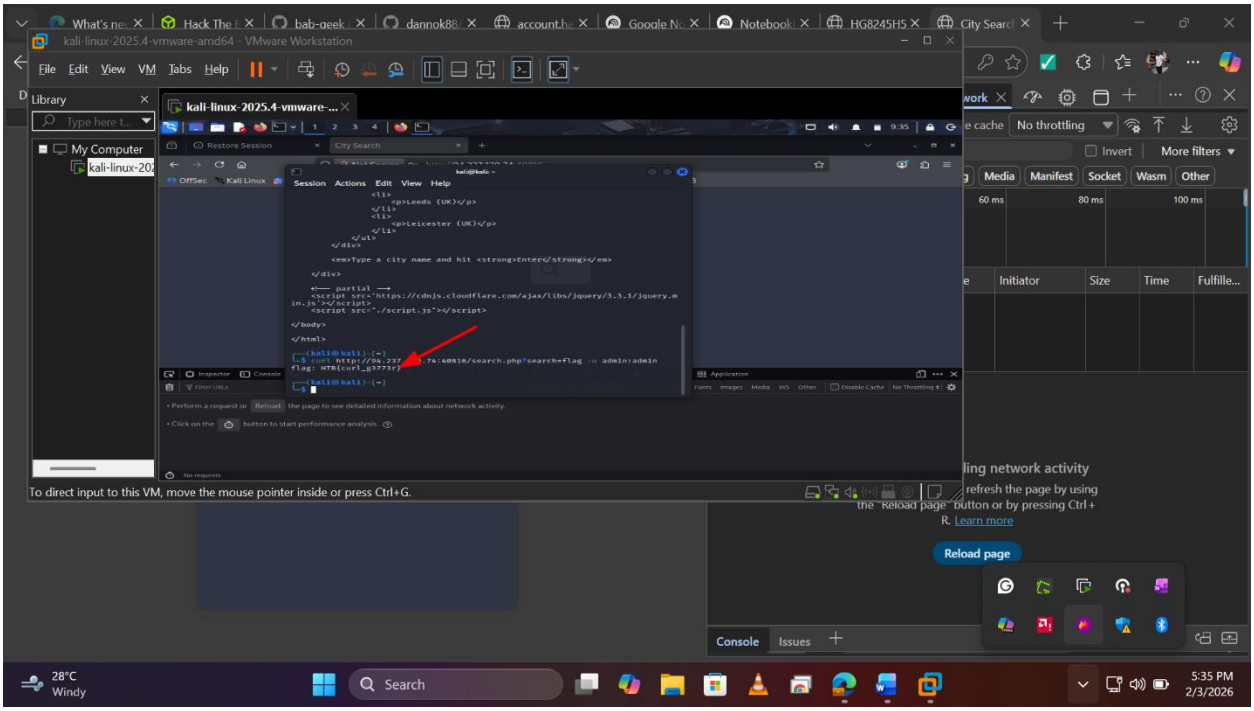
Week 3 Assignment 2



GET

The exercise above seems to be broken, as it returns incorrect results. Use the browser devtools to see what is the request it is sending when we search, and use cURL to search for 'flag' and obtain the flag. HTB{curl_g3773r}

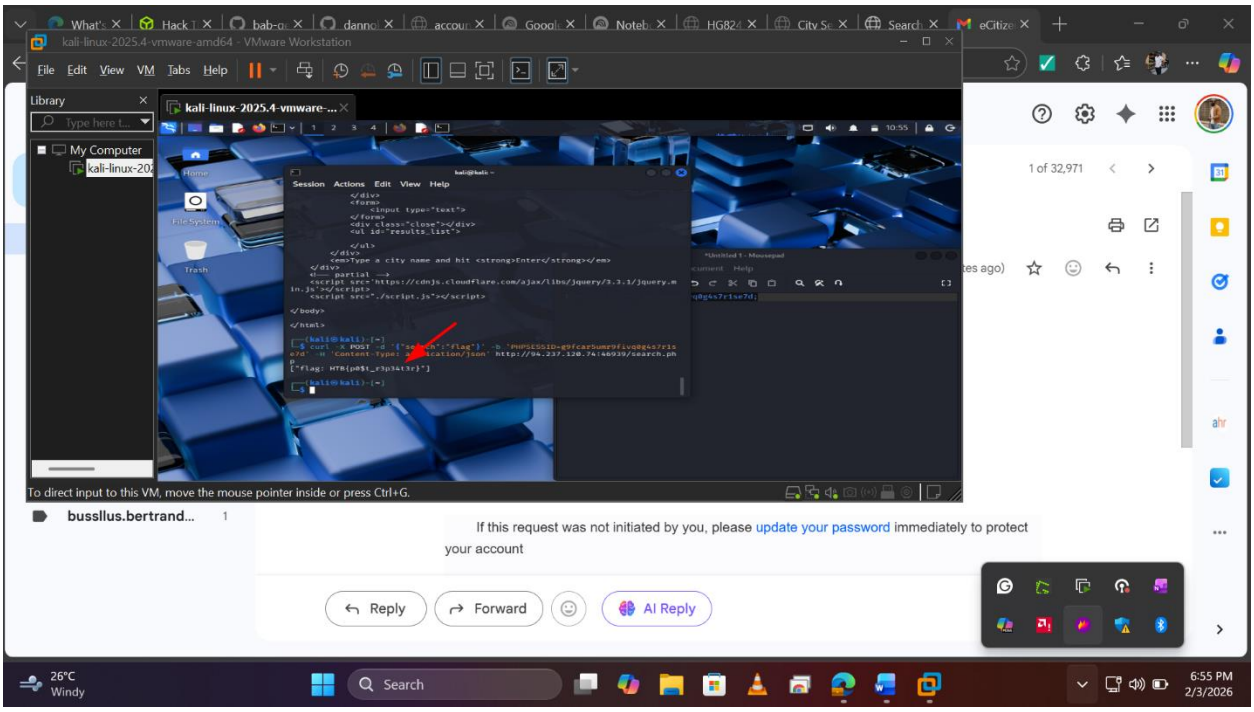
Week 3 Assignment 2



POST

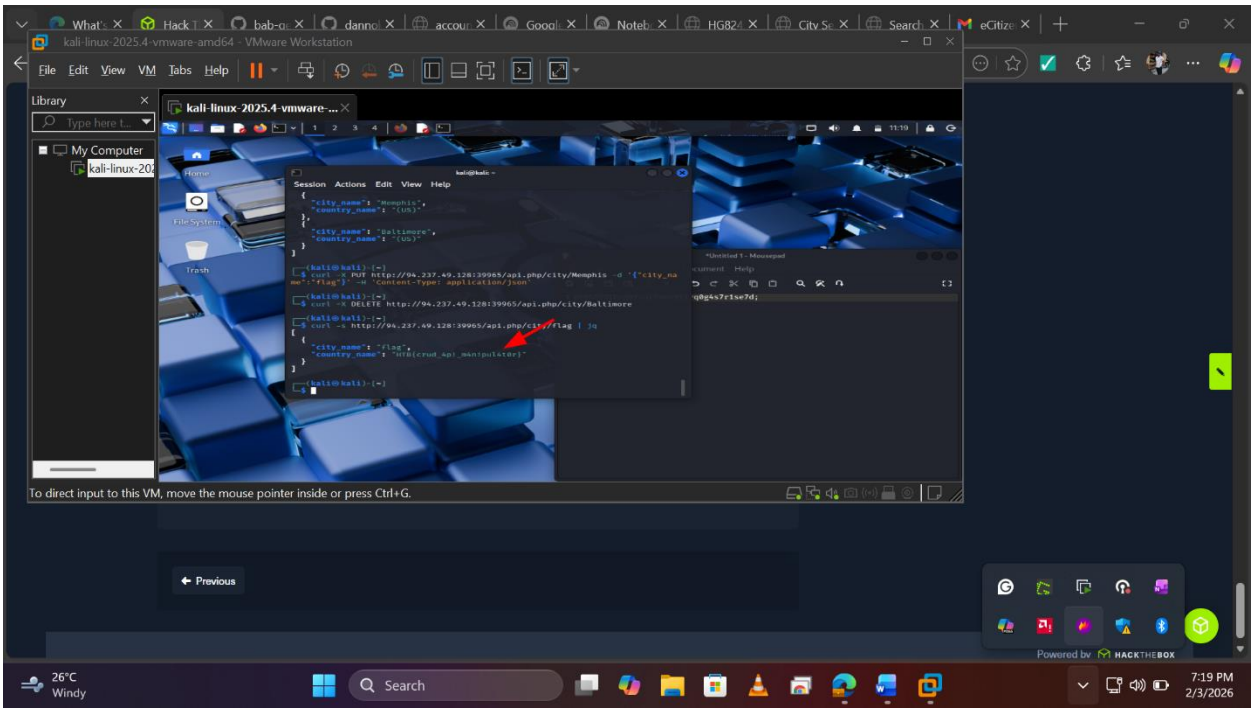
Obtain a session cookie through a valid login, and then use the cookie with cURL to search for the flag through a JSON POST request to '/search.php'

HTB{p0$t_r3p34t3r}



CRUD API

First, try to update any city's name to be 'flag'. Then, delete any city. Once done, search for a city named 'flag' to get the flag. HTB{crud_4p!_m4n!pul4t0r}



Anatomy of Web Traffic

To effectively secure or exploit web applications, one must dissect the traffic into its constituent parts. The training material specifies a focus on **HTTP requests and responses**, including a detailed examination of their **headers**.

HTTP Methods

The "Web Requests" module categorizes the actions a client can request from a server. The source explicitly lists the common HTTP methods that are fundamental to RESTful services and web browsing:

- **GET:** Standard retrieval of data.
- **POST:** Submitting data to be processed.
- **PUT:** Updating existing resources.
- **DELETE:** Removing resources.

Week 3 Assignment 2

Response Codes

Understanding how the server responds is just as important as the request itself. The curriculum covers **HTTP response codes**. These numerical codes (e.g., 200 OK, 404 Not Found, 500 Internal Server Error) tell the analyst whether a request succeeded, failed, or was blocked—vital information during penetration testing or incident response.

API Interaction

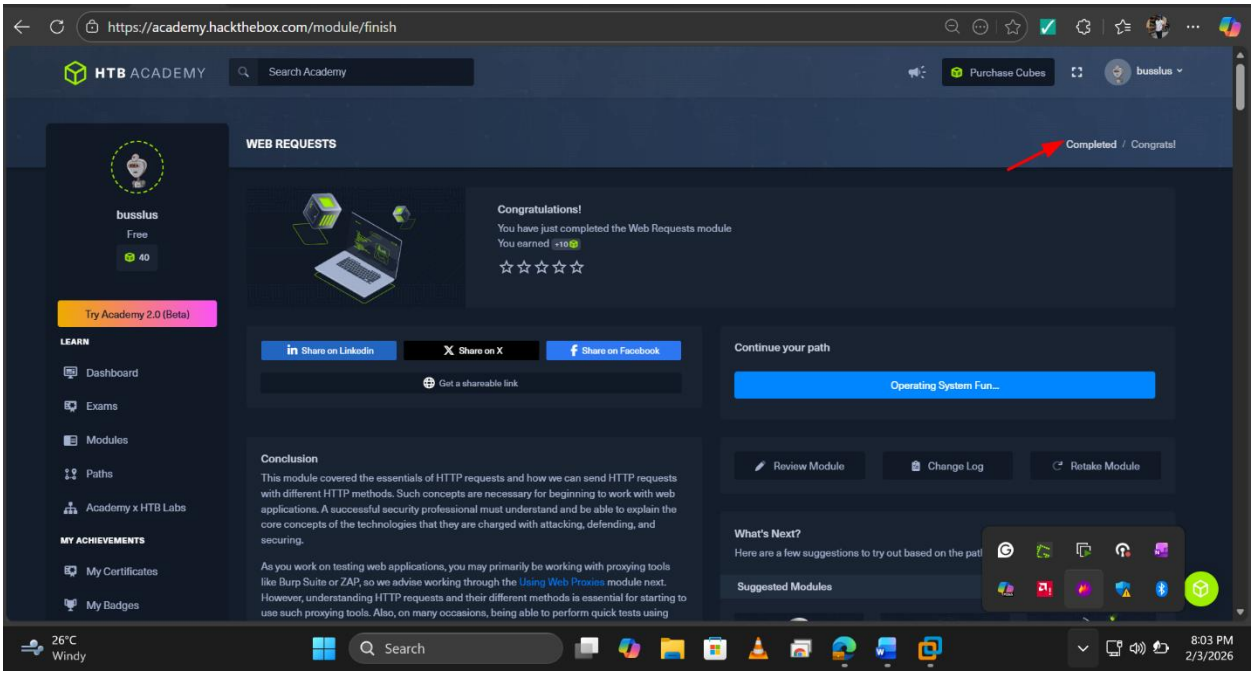
Beyond standard web browsing, the sources indicate a specific focus on **Interacting with APIs**. APIs (Application Programming Interfaces) are the connectors of the modern web, allowing different applications to talk to each other.

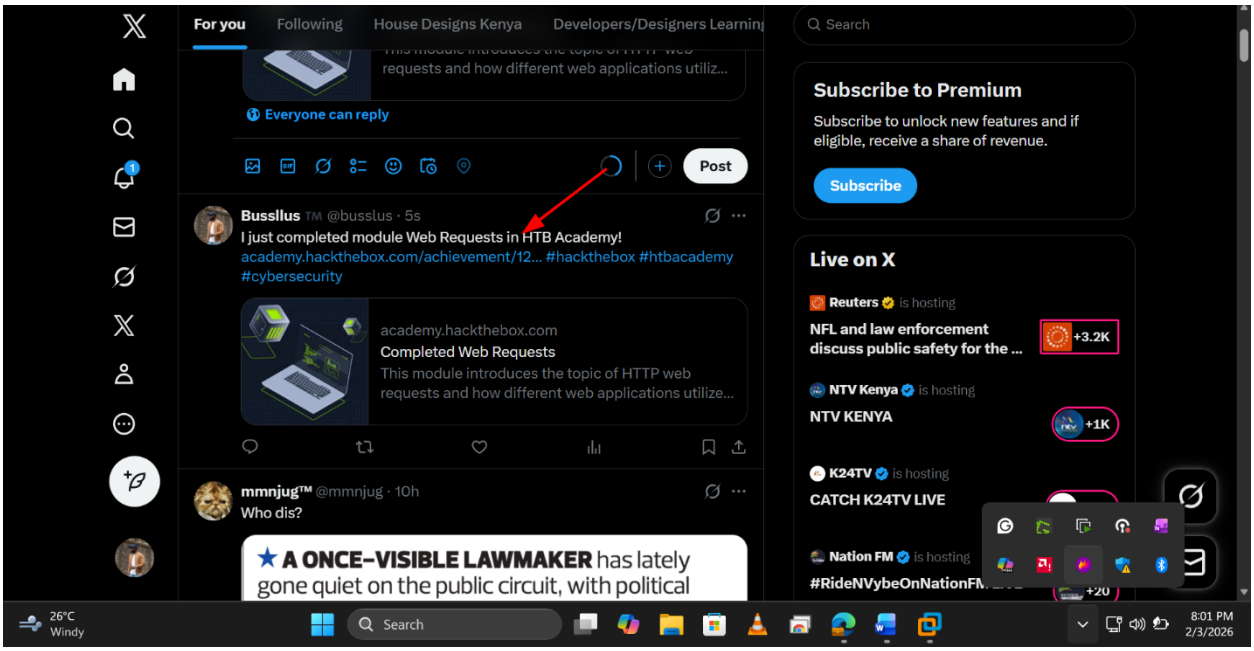
Shareable link

https://academy.hackthebox.com/achievement/1264364/35

Verification of Completion

This completion has been shared on professional networks like Twitter to publicly document my commitment to continuous learning in cybersecurity. This section provides the official verification artifacts for the "Web Requests" module.





Conclusion

In summary, mastering the flow of web traffic requires a comprehensive understanding of HTTP methods and response codes. By analyzing common methods such as **GET, POST, PUT,** and **DELETE**, and learning the techniques for interacting with APIs, this report highlights the core mechanisms that drive digital communication. Ultimately, leveraging these concepts alongside tools like curl and DevTools allows for a deeper understanding of how data is exchanged securely and efficiently across the web.