Cybersecurity Analyst Challenge 1: Penetration Test V2

An important job for cybersecurity analysts is determining potential vulnerabilities within a network. By performing a penetration test, analysts can take the perspective of a potential intruder and find these weaknesses before they can be exploited. These tests may or may not involve technical controls and are an important part of preventing those with malicious intent from accessing your data.

Challenge Objective

Evaluate server security by testing and analyzing SSL-enforced website vulnerabilities.

Challenge Description

Complete a penetration test. Using one of the recommended penetration testing tools, run a test on a web platform of your choice. (NOTE: Make sure you follow ethical hacking guidelines.) Follow the steps outlined below to find the weak points and vulnerabilities in your chosen platform. Synthesize this information into a presentation. Be sure to include all necessary information from the below instructions in your submission.

1. Install package.
2. Run the test on the test domain.
3. Extract the report.
4. Synthesize information.
5. Create a presentation highlighting key areas.
6. Record or screenshot your test completion notification.
7. Upload your test results to files.
8. Think and answer (within the Project):
   a. What did you learn?
   b. How can you improve your skills?
   c. What would you have done differently?

Instructions

1. Start the Challenge.
2. Select the Project Pitch Template.
3. Follow the Task Checklist and make sure to do each step - the Resources provided have all the information needed to learn each task.
4. Track your work for each step.
5. Submit your Project and move on to the next Challenge.

Challenge Tasks

- Review Challenge Description
- Review resources associated to the Challenge
- Research other sources if needed (ie: Google)
- Follow outlined steps in the Challenge
- Answer Challenge Questions
- Document your process and upload necessary photos and/or videos to the 'Project Gallery'
- Finalize and upload your solution to files
- Add any additional links related to your project

Challenge Questions

- How did you approach the Challenge?
- What did you learn?
- What skills did you need to complete the Challenge?
- How can you improve your skills?
- What tools and processes did you use?
- What roadblocks did you have, if any? Or, what would you have done differently?

Solution

1. Server Information:

- **Target Hostname**: fantasy.premierleague.com
- **Target IP**: 151.101.126.133
- **Target Port**: 80 (HTTP)
- **Server**: Varnish (A caching proxy used to speed up web traffic)
- **CDN**: Fastly (Content Delivery Network) is being used, as indicated by the x-timer header.

2. Security Issues Identified:

- **Missing X-Frame-Options Header**: This header prevents clickjacking attacks. Its absence means the website could be vulnerable to this type of attack.
- **Missing X-Content-Type-Options Header**: This header is meant to prevent browsers from interpreting files in a way that is inconsistent with their declared content types. Its absence could lead to content type mismatches, possibly exposing the website to attacks.

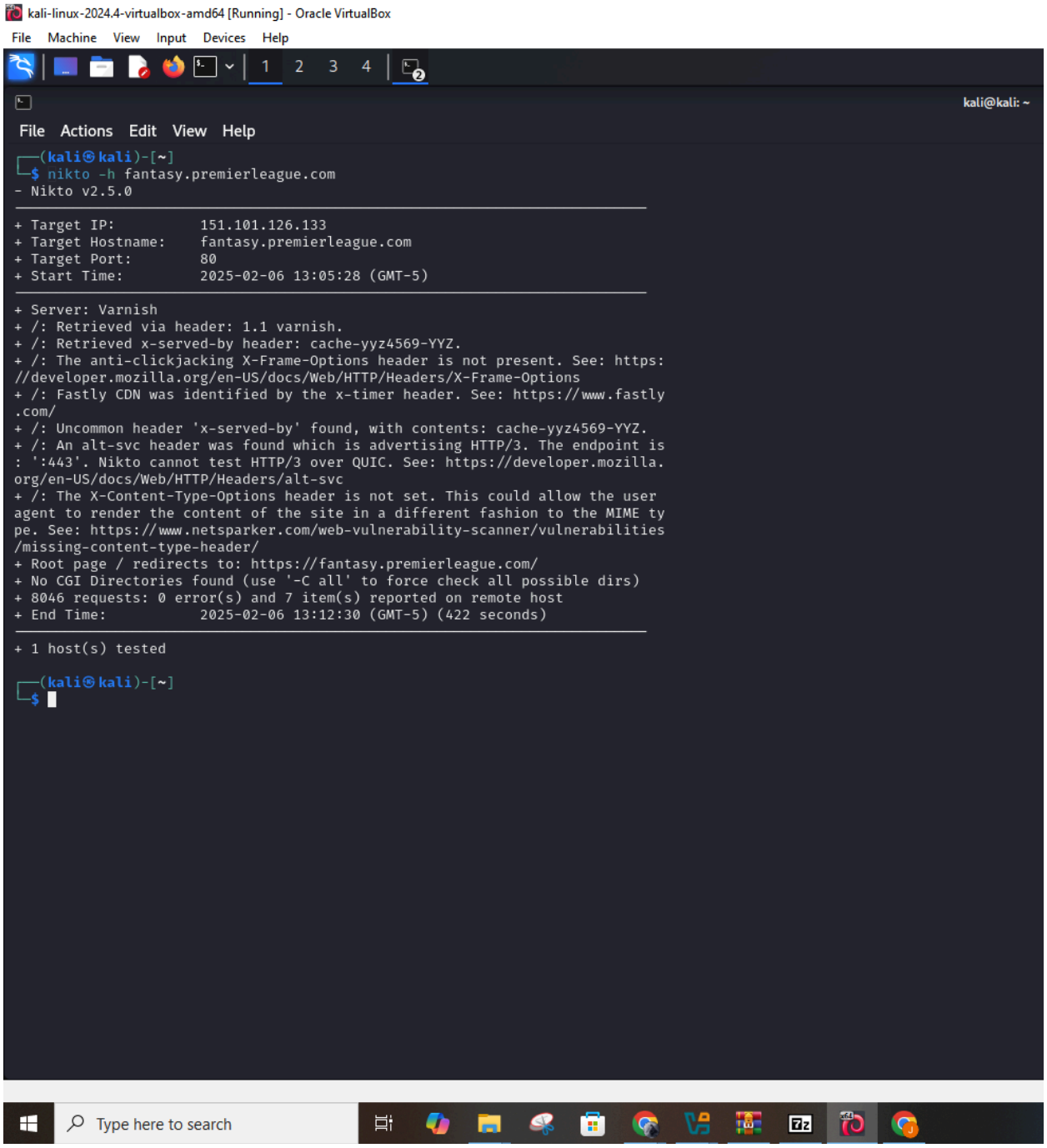
3. Additional Observations:

- **Alt-SVC Header Found**: The website supports HTTP/3 over QUIC, which Nikto cannot test, but it's useful to note as it indicates the use of modern protocols.
- **Redirect to HTTPS**: The website's root page redirects to https://fantasy.premierleague.com/, meaning it's likely configured to force HTTPS usage.

4. No CGI Directories Found: This means the scanner didn't find any CGI scripts that are commonly vulnerable to exploits, but you can manually verify this with a more thorough scan if needed.

Potential Recommendations:

- **Add an X-Frame-Options Header**: This would prevent the site from being embedded within iframes, mitigating clickjacking vulnerabilities.
- **Implement X-Content-Type-Options Header**: This will ensure that browsers respect the declared content type, preventing potential attacks that exploit content type mismatches.
- **Review the use of HTTP/3 and QUIC**: While this isn't directly exploitable, monitoring its usage for any vulnerabilities related to QUIC would be beneficial.


kali-linux-2024.4-virtualbox-amd64 [Running] - Oracle VirtualBox
File Machine View Input Devices Help
1 2 3 4
kali@kali: ~
File Actions Edit View Help
(kali㉿kali)-[~]
$ nikto -h fantasy.premierleague.com
- Nikto v2.5.0
+ Target IP: 151.101.126.133
+ Target Hostname: fantasy.premierleague.com
+ Target Port: 80
+ Start Time: 2025-02-06 13:05:28 (GMT-5)
+ Server: Varnish
+ /: Retrieved via header: 1.1 varnish.
+ /: Retrieved x-served-by header: cache-yyz4569-YYZ.
+ /: The anti-clickjacking X-Frame-Options header is not present. See: https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/X-Frame-Options
+ /: Fastly CDN was identified by the x-timer header. See: https://www.fastly.com/
+ /: Uncommon header 'x-served-by' found, with contents: cache-yyz4569-YYZ.
+ /: An alt-svc header was found which is advertising HTTP/3. The endpoint is : ':443'. Nikto cannot test HTTP/3 over QUIC. See: https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/alt-svc
+ /: The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site in a different fashion to the MIME type. See: https://www.netsparker.com/web-vulnerability-scanner/vulnerabilities/missing-content-type-header/
+ Root page / redirects to: https://fantasy.premierleague.com/
+ No CGI Directories found (use '-C all' to force check all possible dirs)
+ 8046 requests: 0 error(s) and 7 item(s) reported on remote host
+ End Time: 2025-02-06 13:12:30 (GMT-5) (422 seconds)
+ 1 host(s) tested
(kali㉿kali)-[~]
$
Type here to search

1. How did you approach the Challenge?

I started by learning how to set up **Kali Linux** in a virtual box, which was a first-time experience for me. Once I had Kali Linux up and running, I focused on learning the essential tools that would help me perform the penetration test, including how to navigate the environment and use specific tools.

2. What skills did you need to complete the Challenge?

By completing the penetration test, I learned how to use **Nikto**, a web vulnerability scanner, to identify potential security weaknesses on the website. I also became more familiar with the features of **Kali Linux** and how it integrates tools to help with penetration testing tasks.

3. What tools and processes did you use?

I needed a combination of technical skills to complete the challenge, including:

- **Operating System Setup**: Understanding how to set up and configure Kali Linux in a virtual environment.
- **Penetration Testing Tools**: Learning to use tools like **Nikto** to scan for vulnerabilities in websites.
- **Web Security Concepts**: Understanding common web vulnerabilities like clickjacking and the importance of headers such as **X-Frame-Options** and **X-Content-Type-Options**.

4. What roadblocks did you have, if any? Or, what would you have done differently?

To improve my skills, I can dive deeper into using the tools in **Kali Linux**, especially more advanced features of scanners and testing tools. I could also focus on scripting within the environment to automate tasks and become more efficient in conducting penetration tests.

5. What did you learn?

I used **Nikto** for scanning the website. The process involved:

- Running **Nikto** to perform a vulnerability scan on the target website (fantasy.premierleague.com).
- Analyzing the headers, server information, and any issues identified by the scanner.

6. How can you improve your skills?

One challenge I faced was getting Kali Linux set up in a virtual box, as this was my first time working in that environment. I also had to familiarize myself with Nikto's output and interpretation of the results. In hindsight, I could have spent more time learning how to manually exploit vulnerabilities or perform additional scans with other tools like Burp Suite to get a broader perspective on the website's security.