File path traversal, traversal sequences blocked with absolute path bypass –

Lab Report

Submitted By:

Name: Arsalan Khan
Position/Role: Internee
Date: July 25, 2025

Platform:

PortSwigger

Objective:

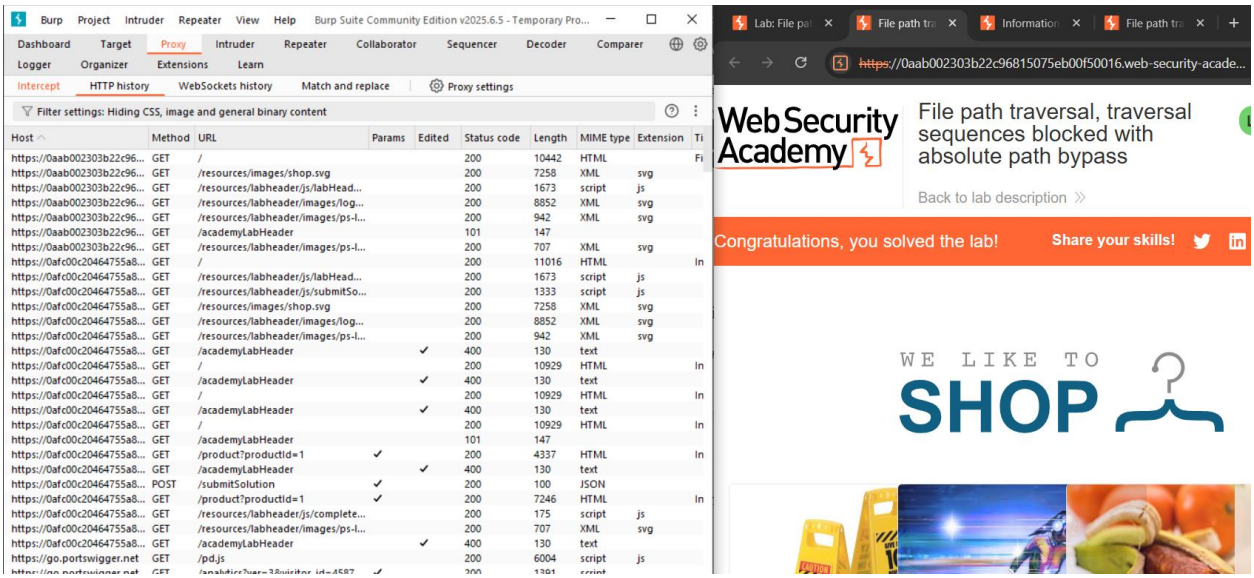
Bypass path traversal filtering and retrieve the contents of the /etc/passwd file by supplying an absolute path instead of using traversal sequences.

Tools Used:

- Burp Suite Community

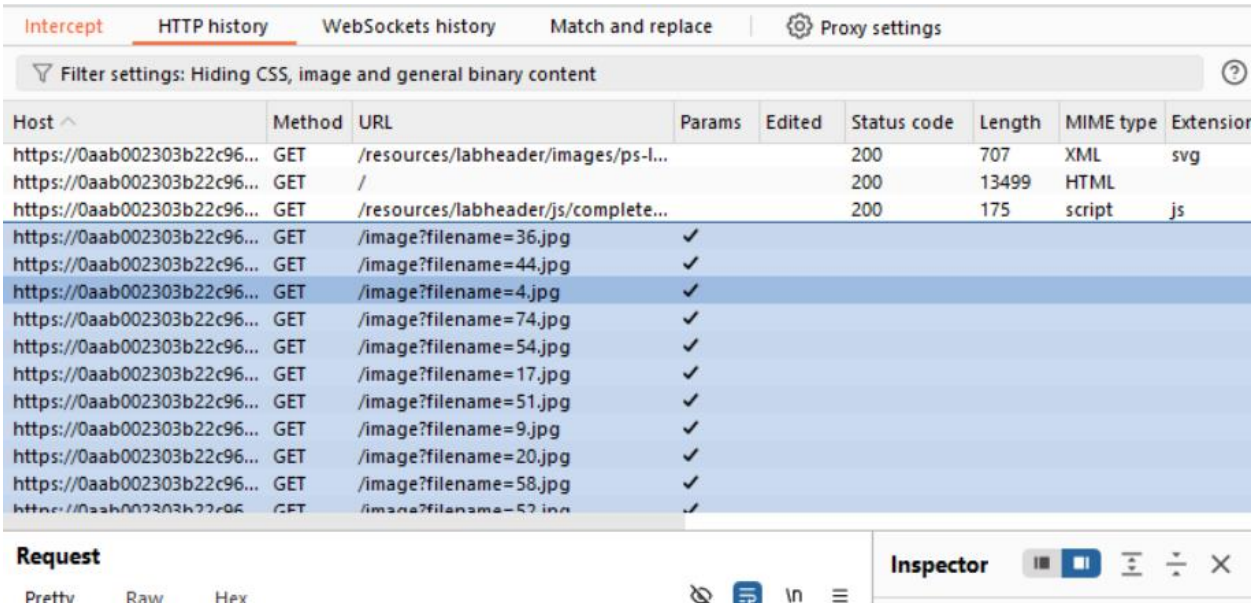
1. Access the Lab

- Clicked on one of the product images to trigger an image request.



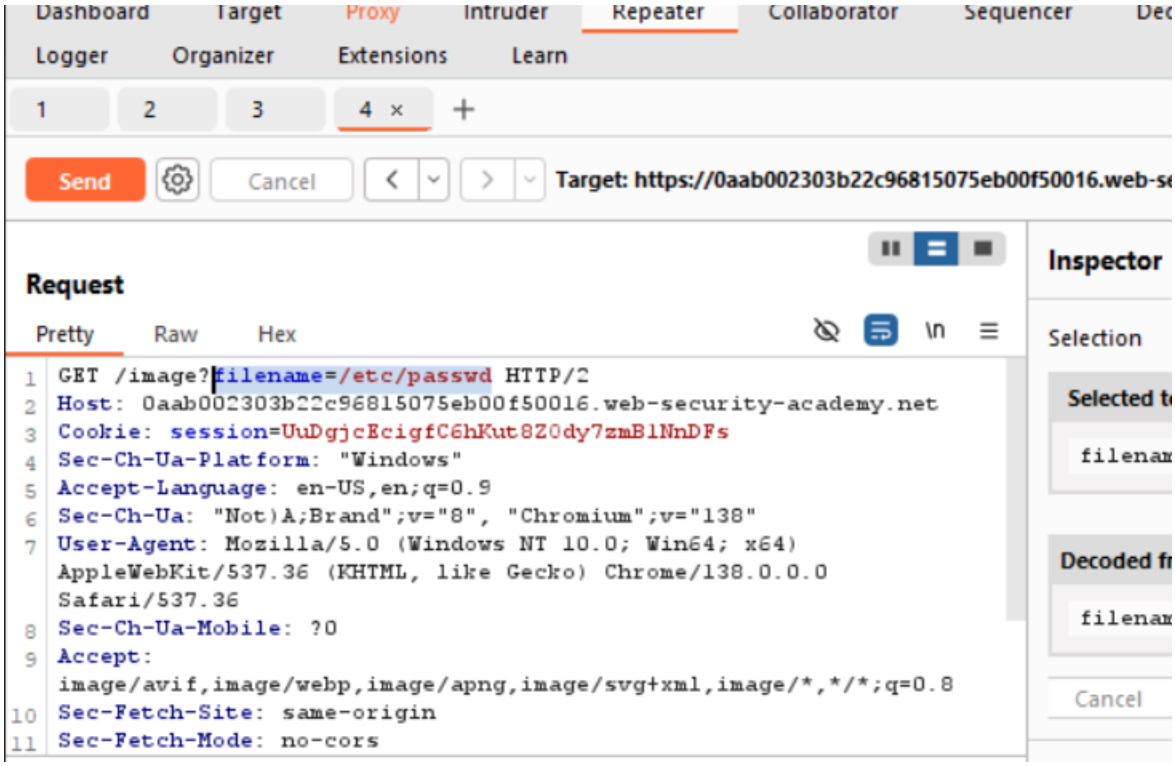
2. Intercept the Image Request.

- Enabled Intercept and clicked a product image to capture the request.



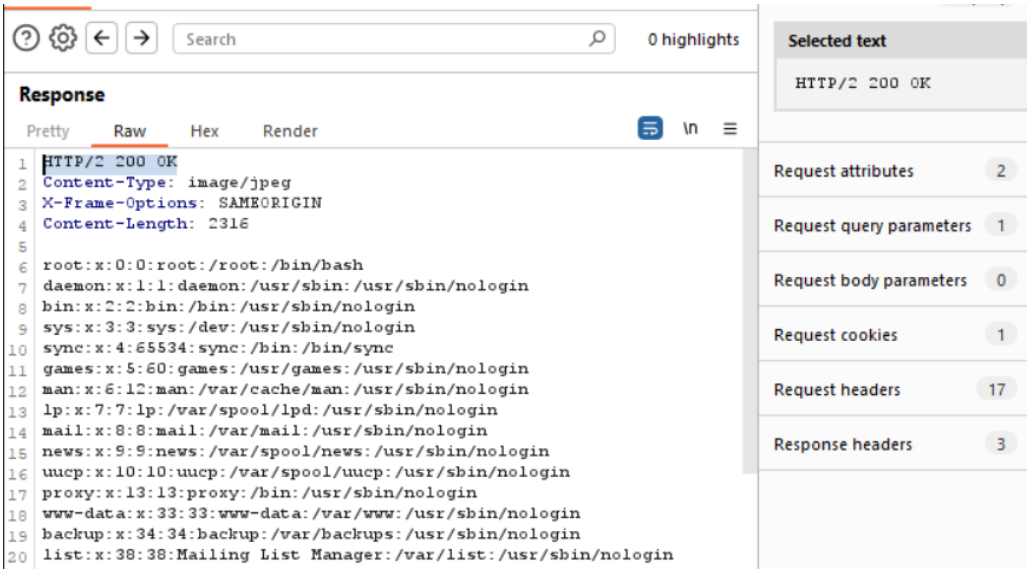
3. Modify the Request with Absolute Path

- Sent the intercepted request to Burp Repeater**.**
- Replaced the filename parameter with an absolute path to the /etc/passwd file:



4. Observe the Server Response

- The server responded with the contents of the `/etc/passwd` file.
- Confirmed that path traversal filtering was bypassed using an absolute path**.**



5. Submit the Solution

Vulnerability Analysis:

- **Issue:** The application attempts to block traversal sequences like `../`, but it fails to account for the use of absolute paths.
- **Exploit Method:** Supplying /etc/passwd directly as the filename bypasses the filter.
- **Impact:** Unauthorized access to sensitive files on the server's filesystem.
- **Risk Level:** High

Mitigation Recommendations:

- Normalize all user input paths (resolve symlinks, absolute paths, etc.) before access.
- Enforce strict allow-lists of permitted filenames and directories.
- Disallow absolute paths and reject any inputs containing `/`, `\`, or starting with `/`.
- Run the application in a sandbox or chroot jail to isolate it from sensitive system files.

Conclusion:

This lab demonstrated a bypass of path traversal protections by using an absolute file path instead of traversal sequences. The /etc/passwd file was accessed directly, proving that the application failed to properly validate or sanitize the filename input.

End…