18th January 2023

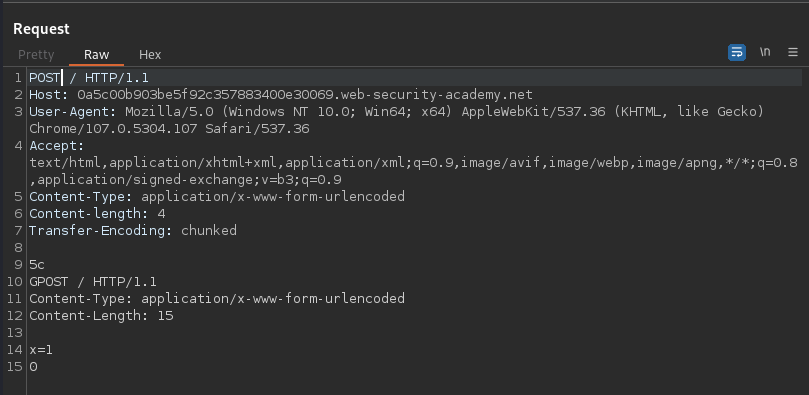
**HTTP REQUEST SMUGGLING LAB 2**

**Summary of Results**

This lab involves a front-end and back-end server, and the back-end server doesn't support chunked encoding. The front-end server rejects requests that aren't using the GET or POST method. The back-end server processes the ‘Content-Length’ header and does not process ‘Transfer-Encoding’ header. Hence, a request is smuggled to the back-end server, so that the next request processed by the back-end server appears to use the method GPOST.

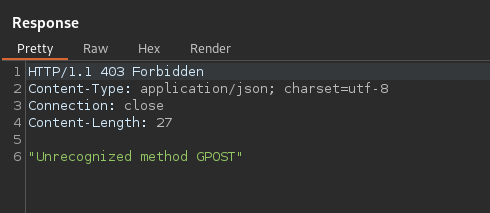
**Attack Narrative**

A request to the application is intercepted in the proxy, and a POST method is added. Both the HTTP specifications ‘Content-length’ and ‘Transfer-Encoding’ headers are used. The ‘Update Content-Length’ is unchecked in burp to prevent it from interfering in the request.



Here, the ‘Content-length’ specifies 4 bytes up to the start of the line following 5c. The front-end server processes the second chunk, which is stated to be zero length, and so it is treated as terminating the request. This request is forwarded on to the back-end server.

When the request from the repeater is forwarded twice, the server responds with the error message stating ‘Unrecognized method GPOST’.

****

This confirms the presence of TE.CL vulnerability.