

WebGoat HTTP Proxies

Twitter: @BlackSheepSpicy

Twitch: <https://twitch.tv/BlackSheepSpicy>

6. After clicking through 5 pages of information that is largely irrelevant to us (because we use Burp Suite rather than Zap) and instructing us how to complete the previous challenge for some reason we finally come to a challenge where we are tasked with modifying a packet in multiple ways:

Intercept and modify a request

Set up the intercept as noted above and then submit the form/request below by clicking the submit button. When your request is intercepted (hits the breakpoint), modify it as follows.

- Change the Method to GET
- Add a header 'x-request-intercepted:true'
- Change the input value 'changeMe' to 'Requests are tampered easily' (without the single quotes)

Then let the request continue through (by hitting the play button).

Note The two play buttons behave a little differently, but we'll let you tinker and figure that out for yourself.

Now that we know what they want us to do, let's turn on our intercepting function and snag a packet (after we forward through all the other packets burp captures, because annoyingly webgoat is pretty noisy on the wire):

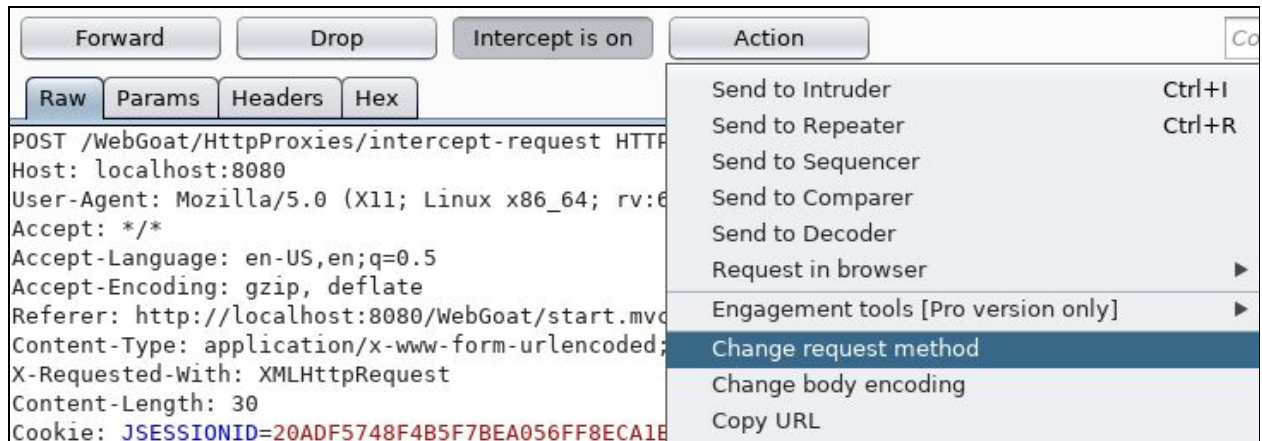
```
POST /WebGoat/HttpProxies/intercept-request HTTP/1.1
Host: localhost:8080
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:60.0) Gecko/20100101 Firefox/60.0
Accept: */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://localhost:8080/WebGoat/start.mvc
Content-Type: application/x-www-form-urlencoded; charset=UTF-8
X-Requested-With: XMLHttpRequest
Content-Length: 30
Cookie: JSESSIONID=20ADF5748F4B5F7BEA056FF8ECA1BE20
Connection: close

changeMe=doesn't+matter+really
```

- **Change the request Method to GET**

Like many people, my first attempt at changing the request method was just changing the verb at the top right. However because get POST and GET methods fill two different roles in HTTP, their structures are vastly different. Luckily rather than completely rewriting the request, we can use Burps **Change Request Method** functionality, found on the drop down menu after clicking **Action**:





The screenshot shows the Burp Suite interface. At the top, there are buttons for 'Forward', 'Drop', 'Intercept is on', and 'Action'. Below these are tabs for 'Raw', 'Params', 'Headers', and 'Hex'. The 'Raw' tab is selected, displaying a POST request to /WebGoat/HttpProxies/intercept-request. The 'Action' menu is open, showing options like 'Send to Intruder', 'Send to Repeater', 'Send to Sequencer', 'Send to Comparer', 'Send to Decoder', 'Request in browser', 'Engagement tools [Pro version only]', 'Change request method' (highlighted), 'Change body encoding', and 'Copy URL'.

This will change our request to a GET, unfortunately burp does not support other request types.

- **Add a header 'x-request-intercepted:true'**

In contrast to the previous objective, this objective is as trivial as it sounds: we literally just copy and paste the text in quotes into the header section of the request. I personally prefer to paste it right under “**X-Requested-with**” for aesthetic reasons:

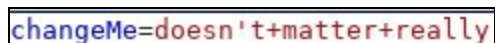


The screenshot shows the modified HTTP request in Burp Suite. The request is a GET to /WebGoat/HttpProxies/intercept-request?changeMe=doesn't+matter+really. The 'x-request-intercepted: true' header has been added below the 'X-Requested-With: XMLHttpRequest' header. The 'Cookie' is JSESSIONID=20ADF5748F4B5F7BEA056FF8ECA1BE20 and the connection is closed.

Headers are beautiful. Fight me.

- **Change the input value 'changeMe' to 'Requests are tampered easily' (without the single quotes)**

On the surface (especially after the last objective) this might look trivial, though there one curiosity that we must take into account. Take a look at the GET parameter:



The screenshot shows the GET parameter 'changeMe=doesn't+matter+really' highlighted in a box.

Instead of spaces, The string contains plus signs. According to RFC-1866 apparently only in request parameters can spaces be encoded as plus signs. No idea why that's a thing rather than just percent encoding, but regardless it's what we have to work with.



With that in mind, we can modify the parameter value keeping in mind the encoding schema:

```
changeMe=Requests+are+tampered+easily
```

Now we can let the packet fly and complete the challenge.

Pretty basic stuff, though in order to become better at your craft you must start at the fundamentals. If you want to see me do these challenges live be sure to drop by my Twitch when I'm live and also follow my Twitter for some fresh memes!

