Experiment 3

Name	**Ruchika Raj**
Section/Group	**20BCS_WM-615_B**
UID	**20BCS9285**
Date	**04/09/2022**

1. **Aim/Overview of the practical:** Working of CSRF (Cross-site request forgery) attack

2. **Objective:** To test virtual box and Kali Linux

3. **Software/Hardware Requirements:** Windows 7 & above version

4. **Tools to be used:**

CSRF attack, we can use various vulnerable websites like BWAPP, DVWA etc. After installing Linux, we have to install these vulnerable websites in Kali Linux to test CSRF vulnerabilities.

Introduction:

CSRF : Cross-Site Request Forgery (CSRF) is an attack that forces an end user to execute unwanted actions on a web application in which they're currently authenticated. With a little help of social engineering (such as sending a link via email or chat), an attacker may trick the users of a web application into executing actions of the attacker's choosing. If the victim is a normal user, a successful CSRF attack can force the user to perform state changing requests like transferring funds, changing their email address, and so forth. If the victim is an administrative account, CSRF can compromise the entire web application.

Attack Surfaces:

The attack surfaces for CSRF are mostly HTTP requests that cause a change in something related to the victim, for example: name, email address, website and even password. It is sometimes used to alter the state of authentication as well. (Login CSRF, Logout CSRF) which are less severe but can still be problematic in some cases Exploitation:

Consider a website example.com and the attacker's website evil.com. Also assume that the victim is logged in and his session is being maintained by cookies. The attacker will:

1. Find out what action he needs to perform on behalf of the victim and find out its endpoint (for example, to change password on target.com a POST request is made to the website that contains new password as the parameter.)
2. Place HTML code on his website evil.com that will imitate a legal request to target.com (for example, a form with method as post and a hidden input field that contains the new password).
3. Make sure that the form is submitted by either using "autosubmit" or luring the victim to click on a submit button.

5. Steps for experiment/practical/Code:

CSRF attack on DVWA (Kali Linux)

Firstly download oracle virtual box and install kali linux**.**

Link to setup virtual box: **https://www.youtube.com/watch?v=4OPfRVdKmGY**

Link to download kali linux: **http://old.kali.org/kaliimages/kali-2017.1/** Install Kali linux in Oracle Virtual box.

Check video for installation:

https://www.youtube.com/watch?v=9ay_0dH5ZyA

To perform CSRF attack, we can use various vulnerable websites like BWAPP,DVWA etc. After installing Linux, we have to install these vulnerable websites in Kali Linux to test CSRF vulnerabilities.

To install BWAPP in Kali Linux refer the link :

https://www.youtube.com/watch?v=b0Jf4vccmeE OR

To install DVWA in Kali Linux refer the link: (terminal commands)

https://www.youtube.com/watch?v=PaB17Cc0dUg

Now open DVWA on kali Linux browser and check csrf attack working. Follow the steps in this link: https://youtu.be/Uzp64CNKSss

Login DVWA with credentials: ‘admin’ as user name and ‘password’ as password. Choose CSRF attack from left hand side. ‘Change your admin password’ page will open there. If we insert new password in input field then it will run successfully.
Login with new password after changing it and notice the behavior.

Save the URL after re-login. Goto Linux terminal and open

editor . Command: gedit

Paste URL here and save it with .txt file.

Go to DVWA website and right click to view source code. See the form tag code and copy it as it is. Paste it in editor and make some changes in code. Mandatory to add “value” field.

<form action: paste copied URL here

<p> Special Offer</p>

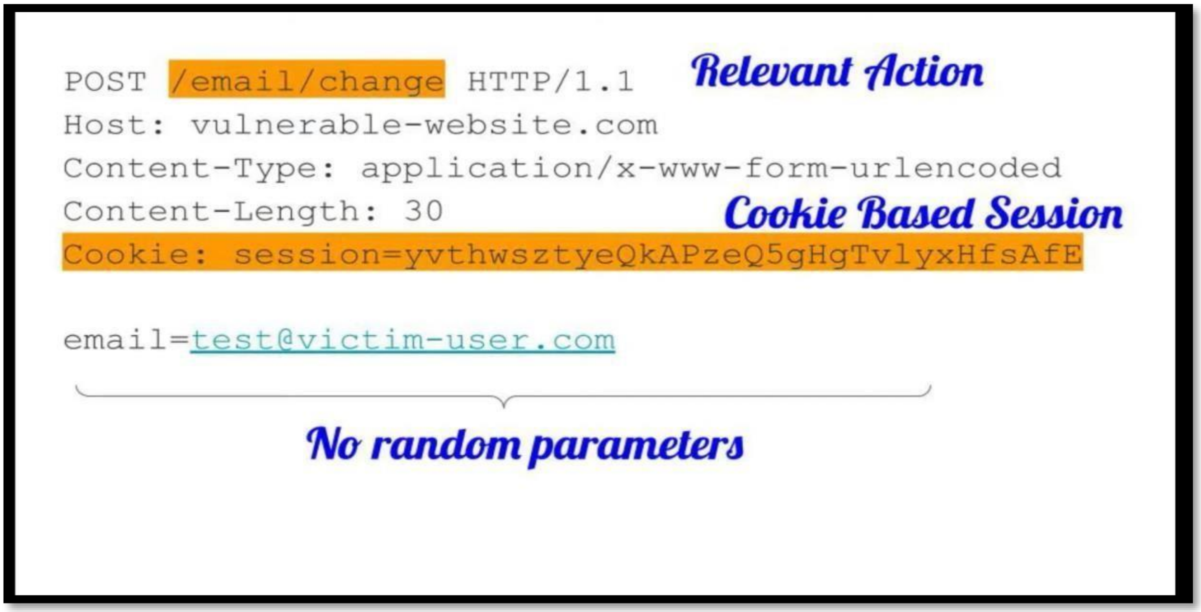
<input type=”hidden” AUTOCOMPLETE=”off” name=”password_new”

value=”hacked”> Save the file e.g. csrf.html. Types ls in Linux terminal to

check whether file exists.

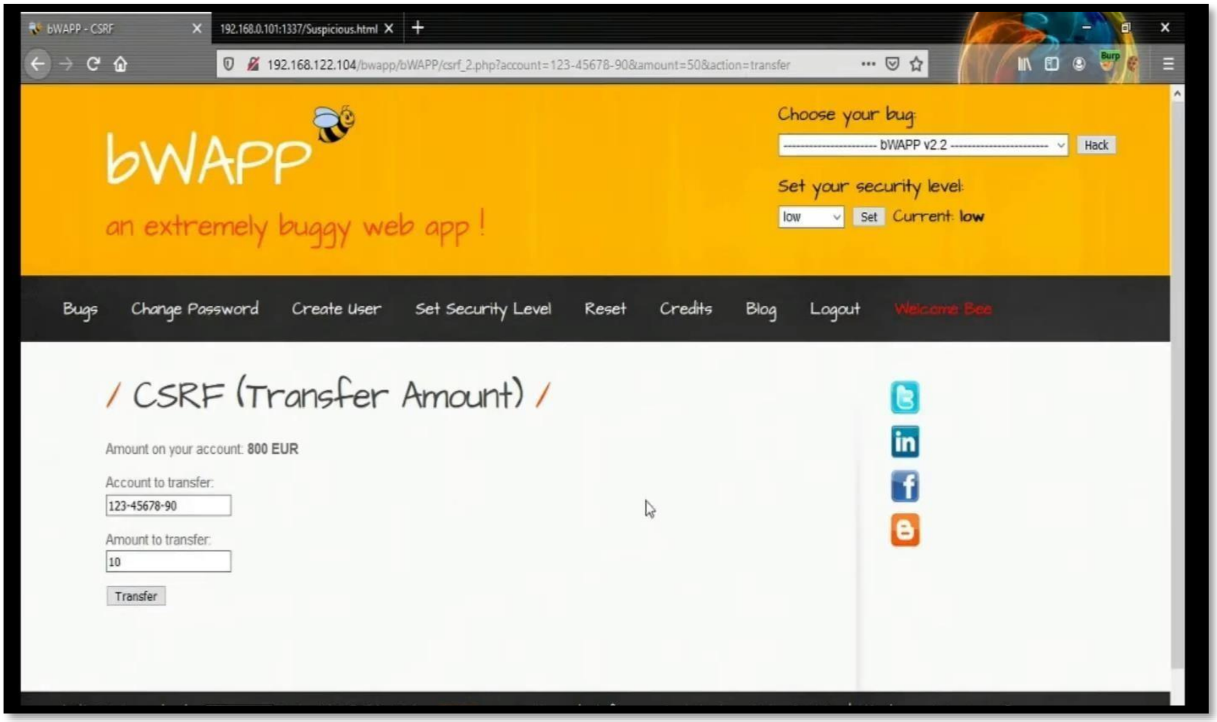
Open Firefox from terminal: Firefox csrf.html and you will see a button with name “ change” on page. This link is considered malicious sent by some attacker. End user will click on this button/link then password will be changed with one that hacker has added into the code. This is possible when user is login to the website.

6. Result/Output/Writing Summary:



```
POST /email/change HTTP/1.1
Host: vulnerable-website.com
Content-Type: application/x-www-form-urlencoded
Content-Length: 30
Cookie: session=yvthwsztyeQkAPzeQ5gHgTvlyxHfsAfE

email=test@victim-user.com
```



bWAPP - CSRF | 192.168.0.101:1337/Suspicious.html

192.168.122.104/bwapp/bWAPP/csrf_2.php?account=123-45678-90&amount=50&action=transfer

Choose your bug: bWAPP v2.2

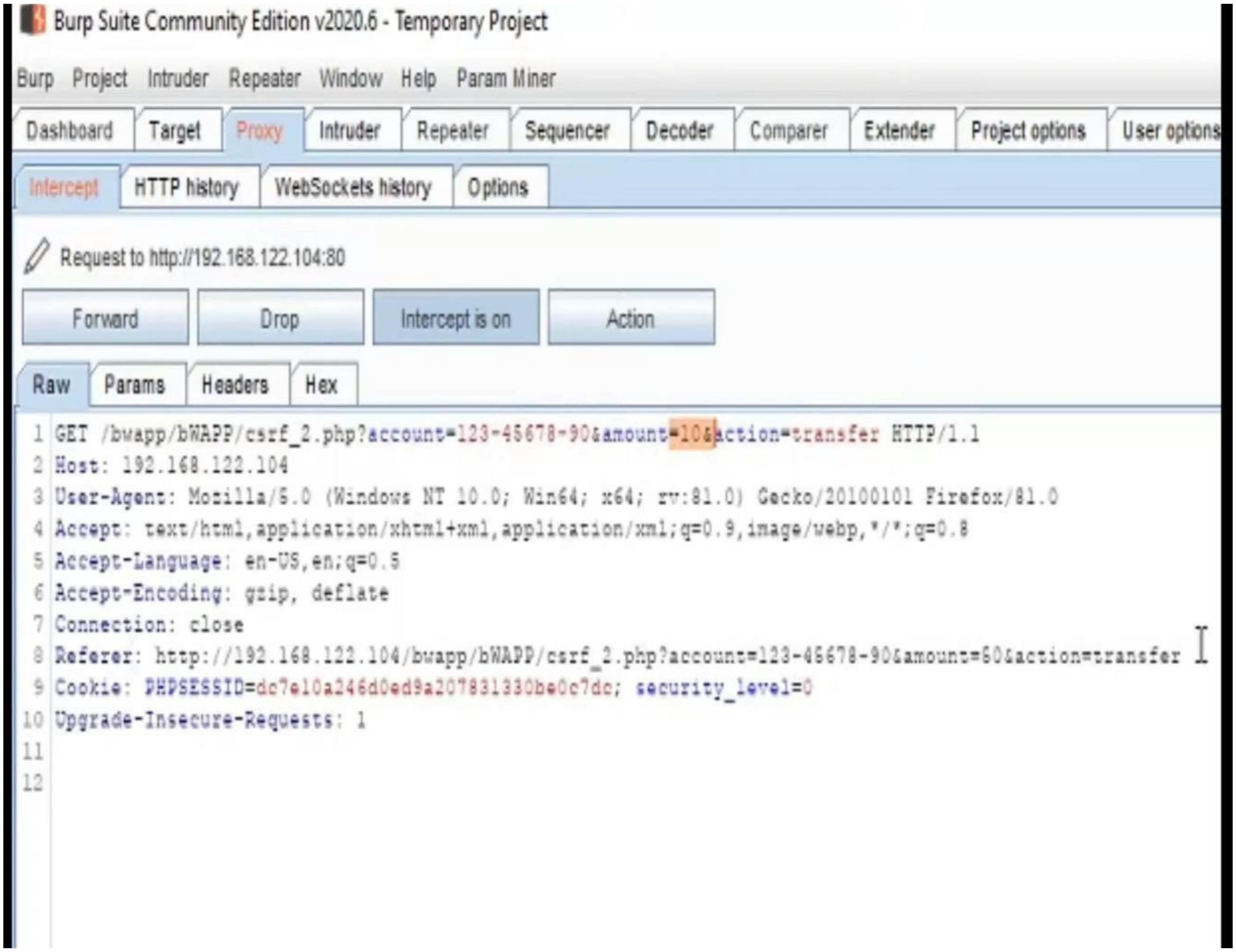
Set your security level: low Current: low

/ CSRF (Transfer Amount) /

Amount on your account: **800 EUR**

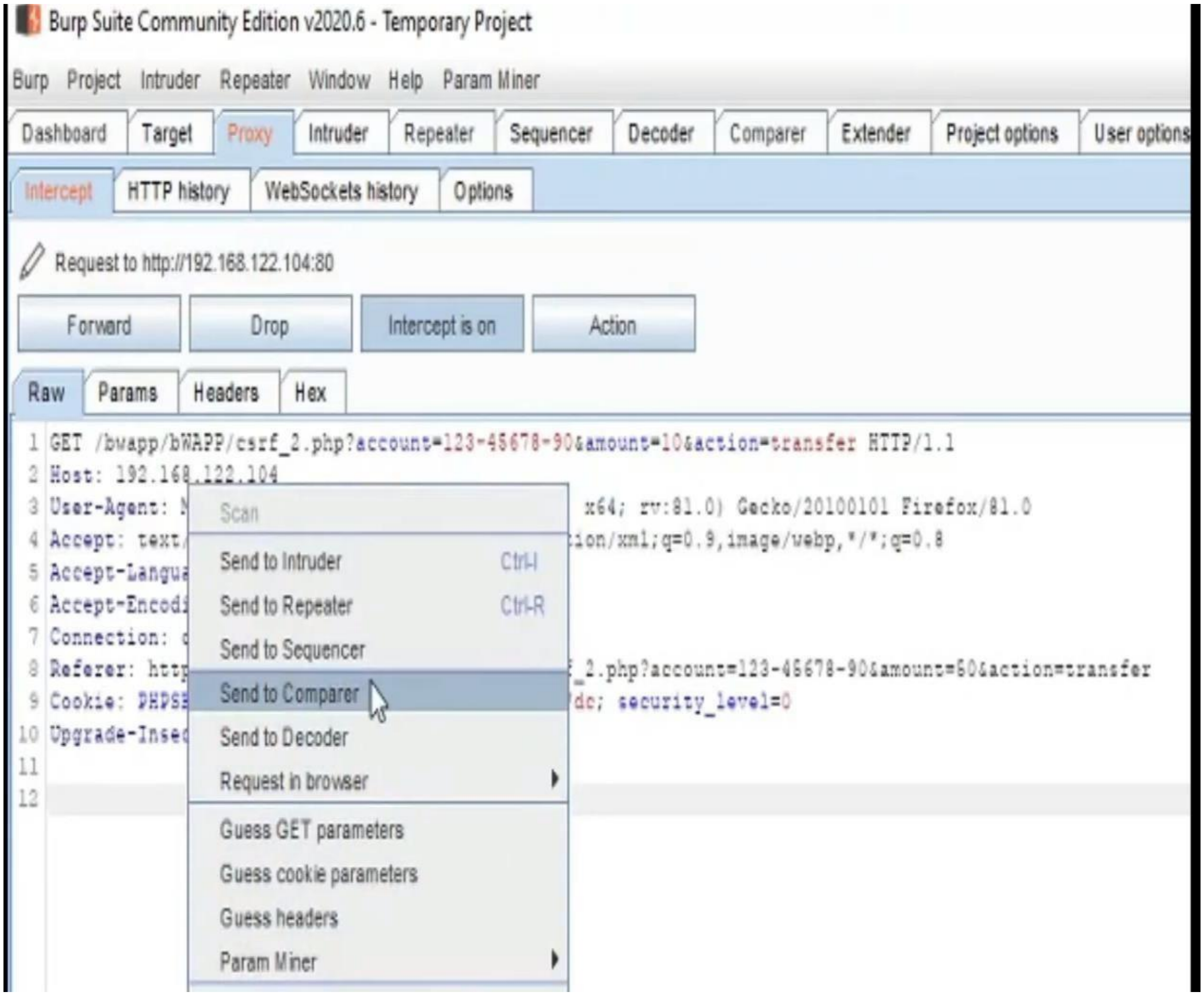
Account to transfer: 123-45678-90

Amount to transfer: 10


Burp Suite Community Edition v2020.6 - Temporary Project
Burp Project Intruder Repeater Window Help Param Miner
Dashboard Target Proxy Intruder Repeater Sequencer Decoder Comparer Extender Project options User options
Intercept HTTP history WebSockets history Options
Request to http://192.168.122.104:80
Forward Drop Intercept is on Action
Raw Params Headers Hex

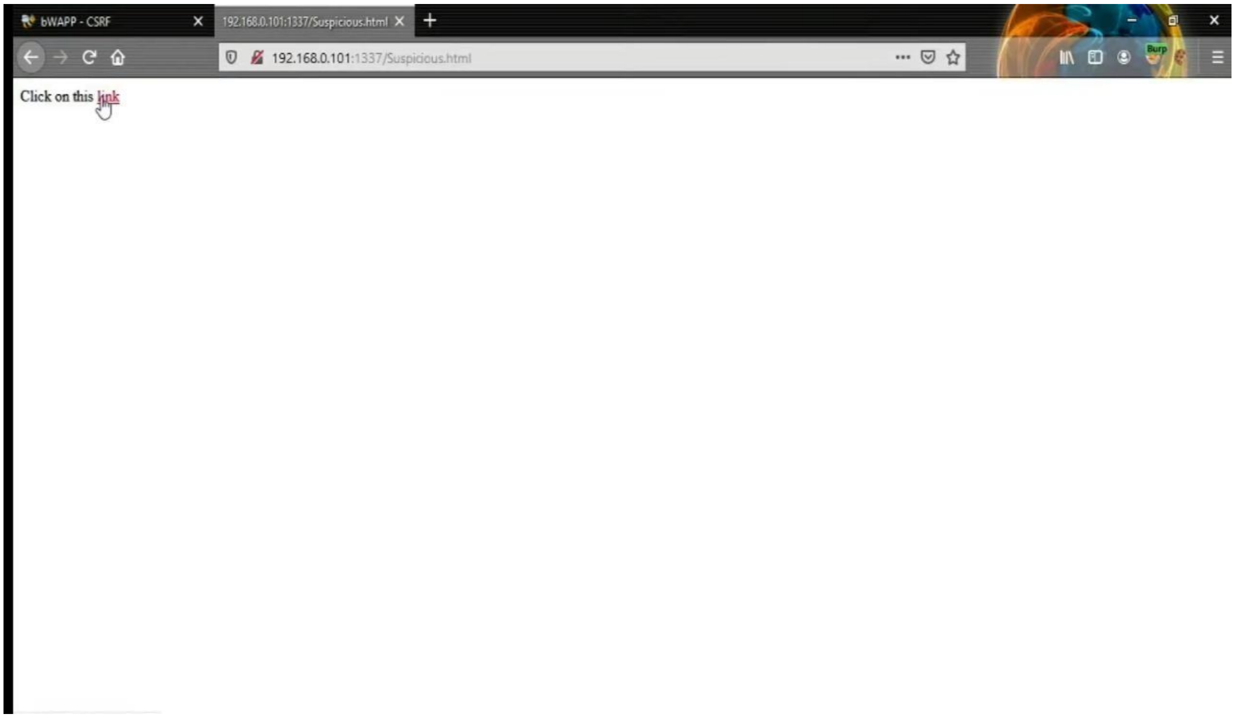
```
GET /bwapp/bWAPP/csrf_2.php?account=123-45678-90&amount=10&action=transfer HTTP/1.1
Host: 192.168.122.104
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:81.0) Gecko/20100101 Firefox/81.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: close
Referer: http://192.168.122.104/bwapp/bWAPP/csrf_2.php?account=123-45678-90&amount=60&action=transfer
Cookie: PHPSESSID=dc7e10a246d0ed9a207831330be0c7dc; security_level=0
Upgrade-Insecure-Requests: 1


```

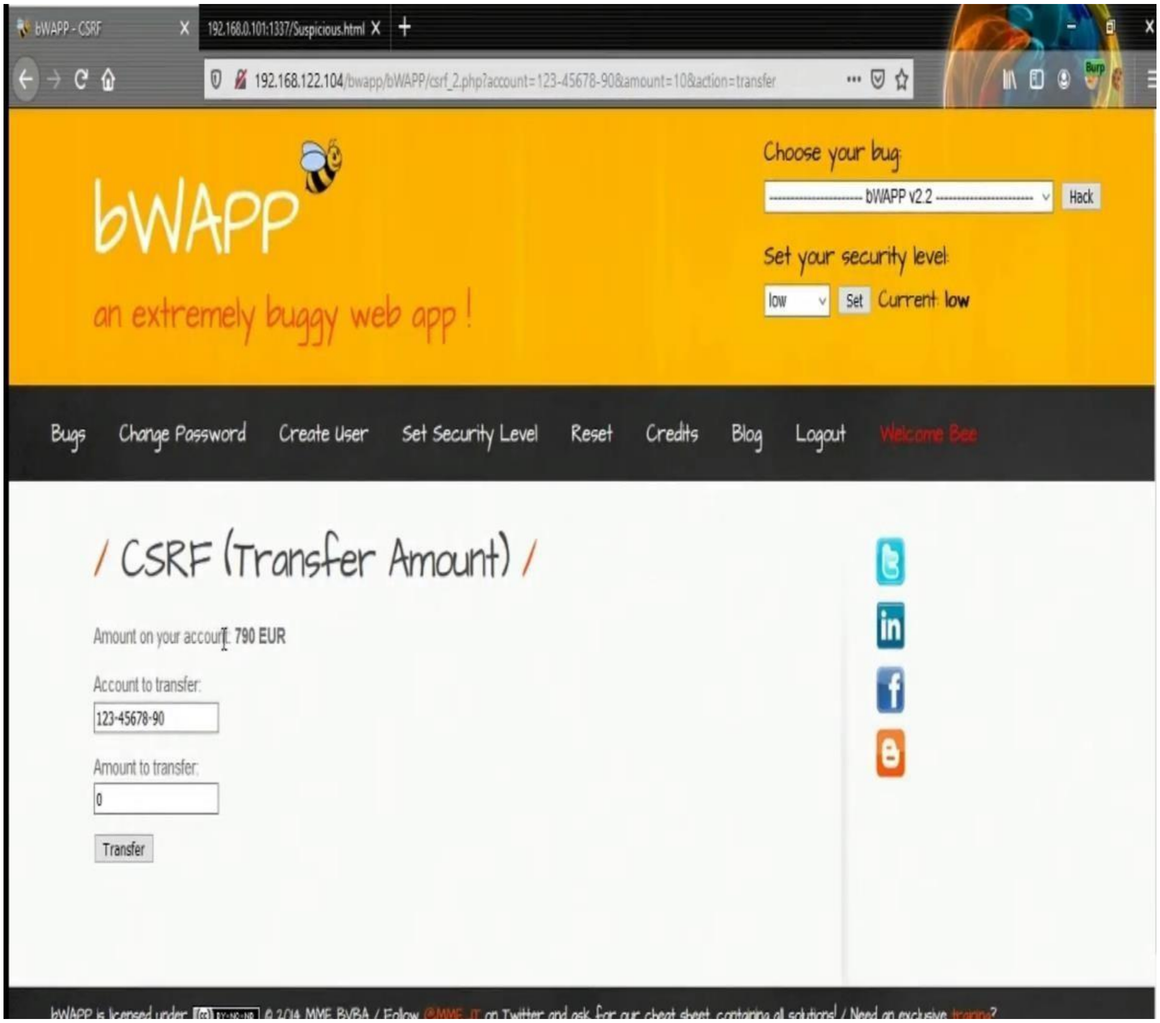


Burp Suite Community Edition v2020.6 - Temporary Project
Burp Project Intruder Repeater Window Help Param Miner
Dashboard Target Proxy Intruder Repeater Sequencer Decoder Comparer Extender Project options User options
Intercept HTTP history WebSockets history Options
Request to http://192.168.122.104:80
Forward Drop Intercept is on Action
Raw Params Headers Hex
Scan
Send to Intruder Ctrl-I
Send to Repeater Ctrl-R
Send to Sequencer
Send to Comparer
Send to Decoder
Request in browser
Guess GET parameters
Guess cookie parameters
Guess headers
Param Miner

```
GET /bwapp/bWAPP/csrf_2.php?account=123-45678-90&amount=10&action=transfer HTTP/1.1
Host: 192.168.122.104
User-Agent: M                                    x64; rv:81.0) Gecko/20100101 Firefox/81.0
Accept: text/                                    tion/xml;q=0.9,image/webp,*/*;q=0.8
Accept-Langua
Accept-Encodi
Connection: c
Referer: http                                    f_2.php?account=123-45678-90&amount=60&action=transfer
Cookie: PHPSE                                    dc; security_level=0
Upgrade-Insec
```


bWAPP - CSRF
192.168.0.101:1337/Suspicious.html
192.168.0.101:1337/Suspicious.html
Burp

Click on this link



192.168.122.104/bwapp/bWAPP/csrf_2.php?account=123-45678-90&amount=10&action=transfer

Current: low

/ CSRF (Transfer Amount) /

Amount on your account: 790 EUR

Account to transfer: 123-45678-90

Amount to transfer: 0

bWAPP is licensed under BY-NC-ND © 2014 MME BVBA / Follow @MME_IT on Twitter and ask for our cheat sheet, containing all solutions! / Need an exclusive training?

Learning outcomes (What I have learnt):

1. A CSRF attack targets Web applications failing to differentiate between valid requests and forged requests controlled by attacker

2. CSRF tokens can prevent CSRF attacks by making it impossible for an attacker to construct a fully valid HTTP request suitable for feeding to a victim user.