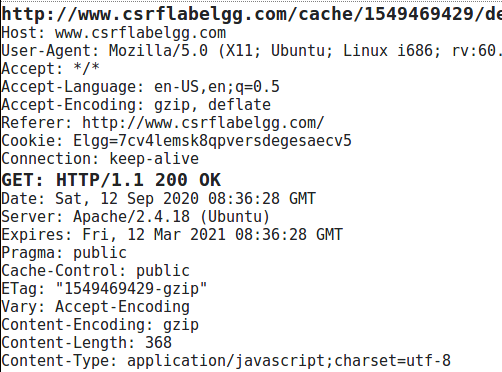
Lab4-report

57118113 蔡义涵

**3.1 Task 1: Observing HTTP Request.**

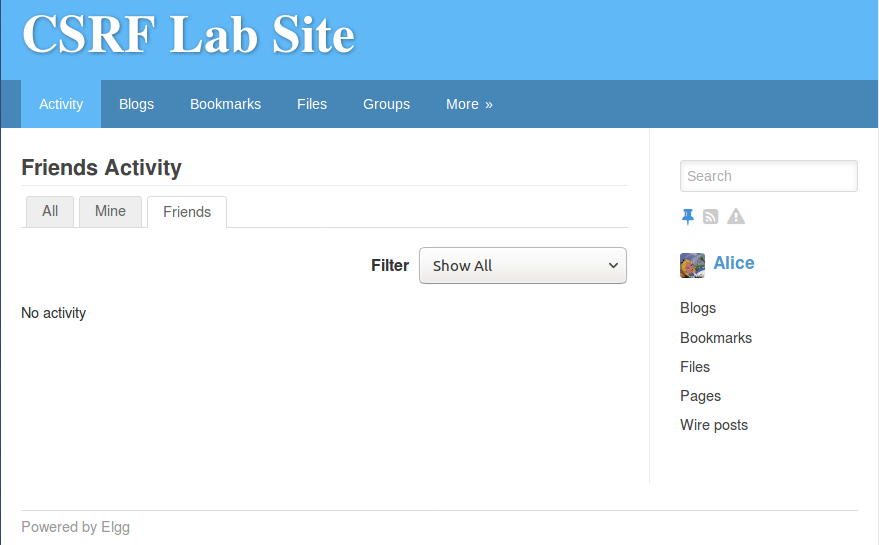




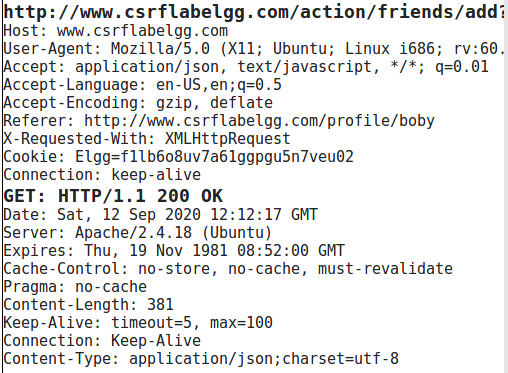
网络应用程序大部分服务均为GET/POST服务。GET请求通常用于查询、获取数据，而POST请求多用于发送数据。GET请求的参数在URL中，因此绝不能用GET请求传输敏感数据，而POST请求的参数在请求头中，安全性略高于GET请求。

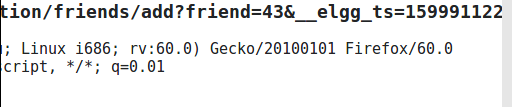
**3.2 Task 2: CSRF Attack using GET Request**

实验开始之前，先观察Alice的好友列表，发现为空。



首先，Boby先使用Charlie的账号添加Boby的账号为好友，从而查看到自己的id。



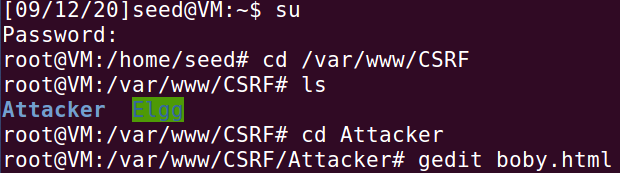


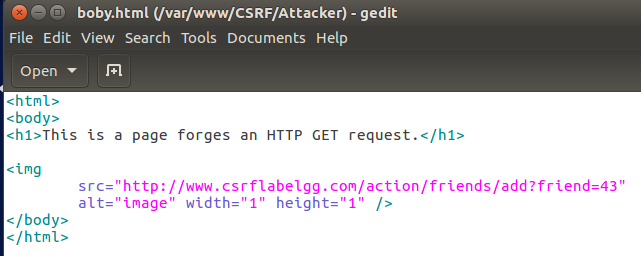
从图中可以得到，Boby的friend id为43

下面进行CSRF攻击

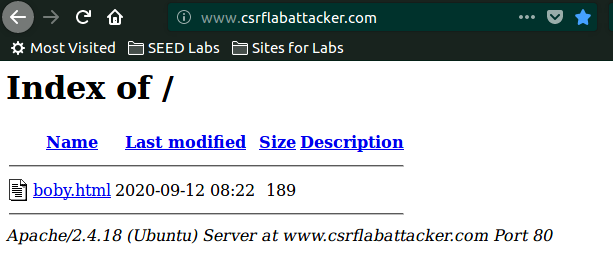
打开/var/www/CSRF/Attacker目录下的boby.html文件，

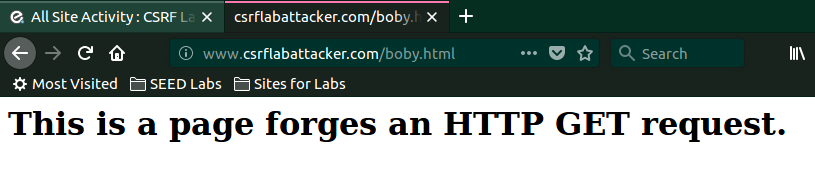
修改为如下：



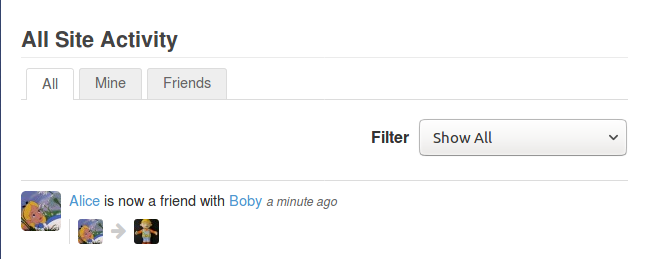


Alice点击恶意网页后：





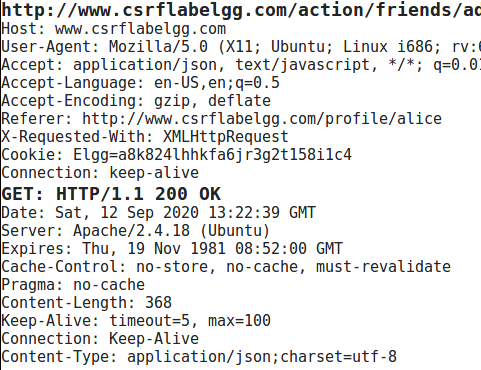
查看结果，发现已经添加了boby为好友，即代表攻击成功



**3.3 Task 3: CSRF Attack using POST Request**

下面我们开始使用POST请求进行攻击

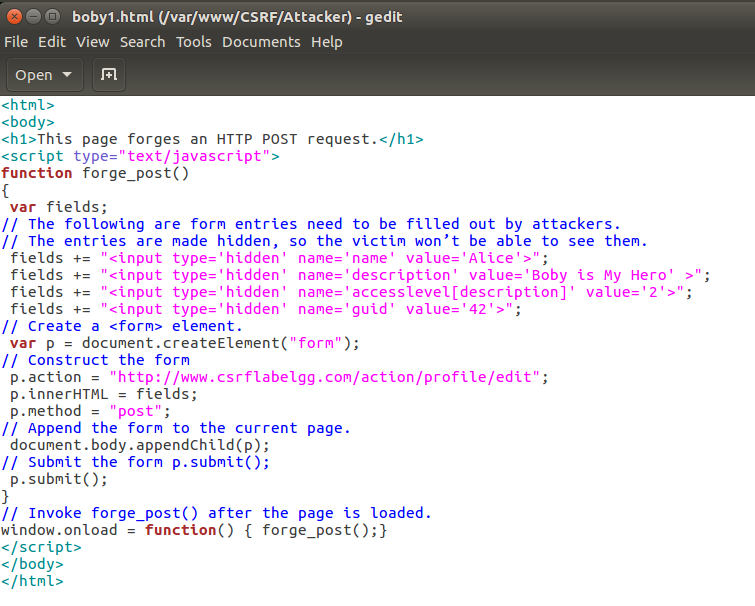
首先Boby先添加Alice好友，以获得Alice的friend id。



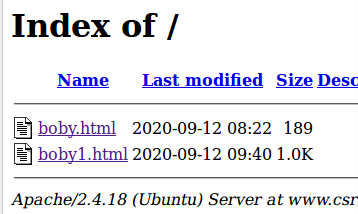


可以得到Alice的friend id为42

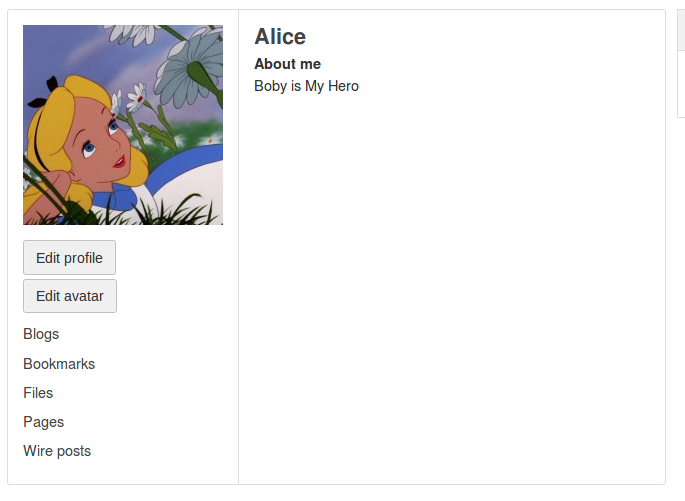




修改完成后，Alice进入恶意网页







发现Alice的主页被修改

问题1：伪造的HTTP请求需要Alice的用户ID（guid）才能正常工作。 如果Boby特别针对Alice，那么在攻击之前，他可以找到获取Alice用户ID的方法。 Boby不知道Alice的Elgg密码，因此他无法登录Alice的帐户来获取信息。 请描述Boby如何解决此问题。

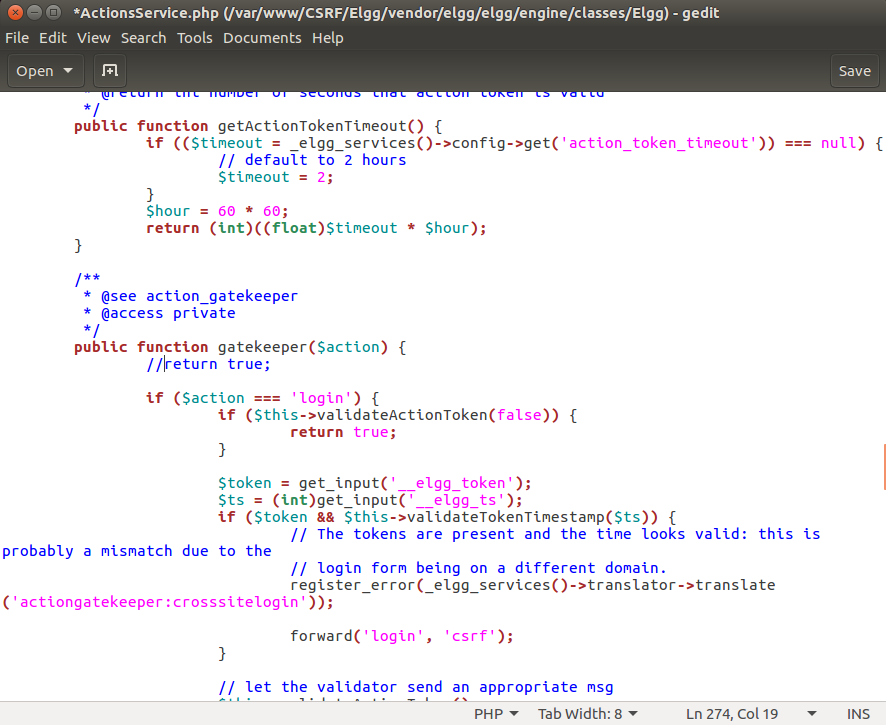
解答：Boby可以通过添加Alice为好友，查看GET请求以获得Alice的friend id为42。

问题2：Boby如果想对访问其恶意网页的任何人发起攻击。 在这种情况下，他不知道是谁预先访问了该网页。 他还能继续发动CSRF攻击来修改受害者的Elgg个人资料吗？ 请解释。

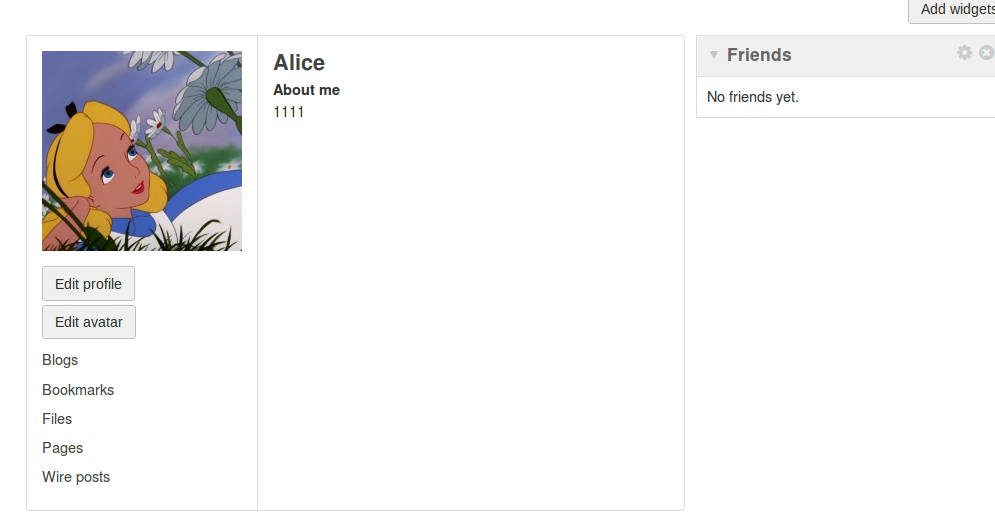
解答：可以通过编写JavaScript代码，自动获取访问页面者的friend id，从而实施恶意行为。

**3.4 Task 4: Implementing a countermeasure for Elgg**

首先进入/var/www/CSRF/Elgg/vendor/elgg/elgg/engine/classes/Elgg目录打开文件ActionsService.php，并注释掉gatekeeper函数中的“return true;”

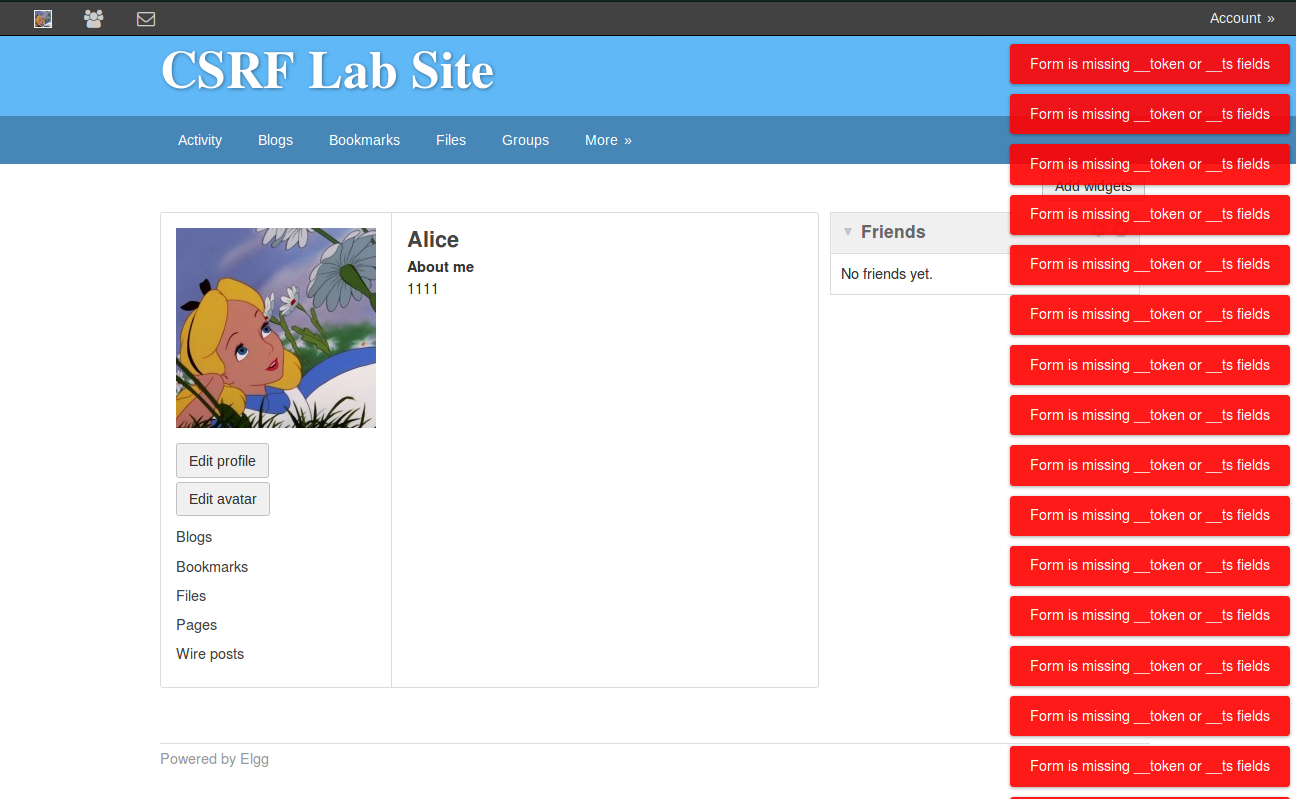


再将Alice好友列表中的Boby删除，修改about me内容为其他



随后，再次执行前两次攻击





两次攻击都失败了，这是因为Web应用Elgg使用secret-token方法进行保护。它在GET和POST请求的URL字符串中嵌入了两个参数，分别是时间戳elgg ts和elgg token。因而不同源的恶意网站无法读取web上的elgg token，从而导致攻击失败。