**Lab 5**

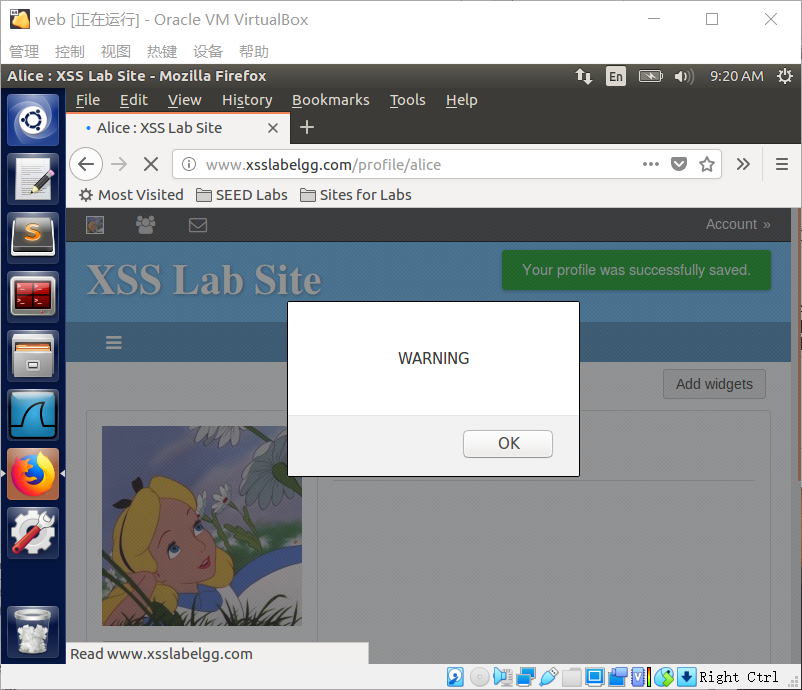
57118213 陈洪杰

**Task 1: Posting a Malicious Message to Display an Alert Window**

首先在brief description里输入下面的代码：



点击保存后，界面返回并出现弹窗：



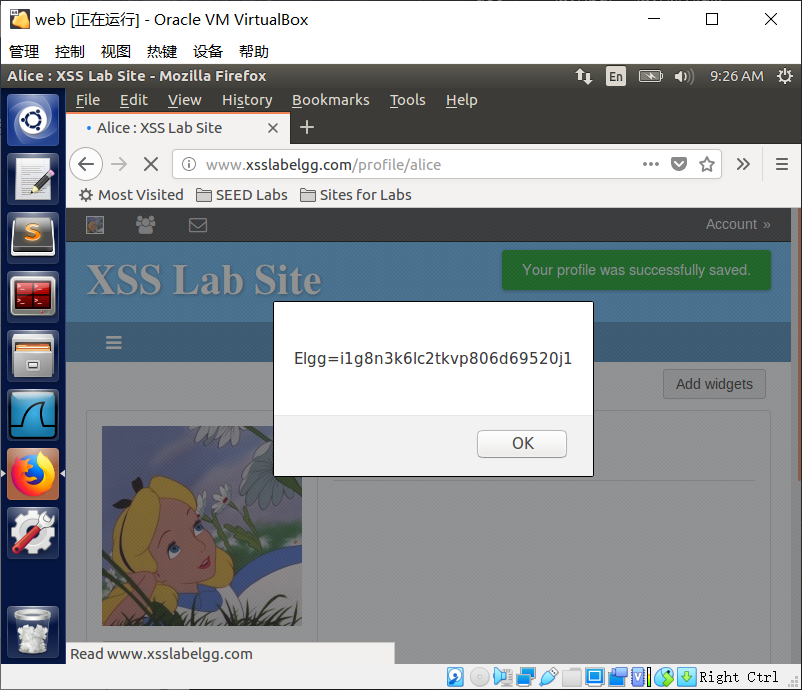
由此可见该语句执行成功。

**Task 2: Posting a Malicious Message to Display Cookies**

首先在brief description里输入下面的代码：



点击保存后，界面返回并出现弹窗：



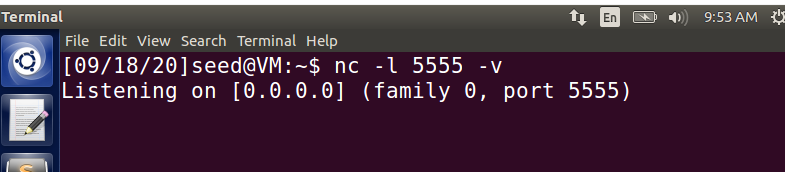
已经显示浏览者的的cookie，语句执行成功。

**Task 3: Stealing Cookies from the Victim’s Machine**

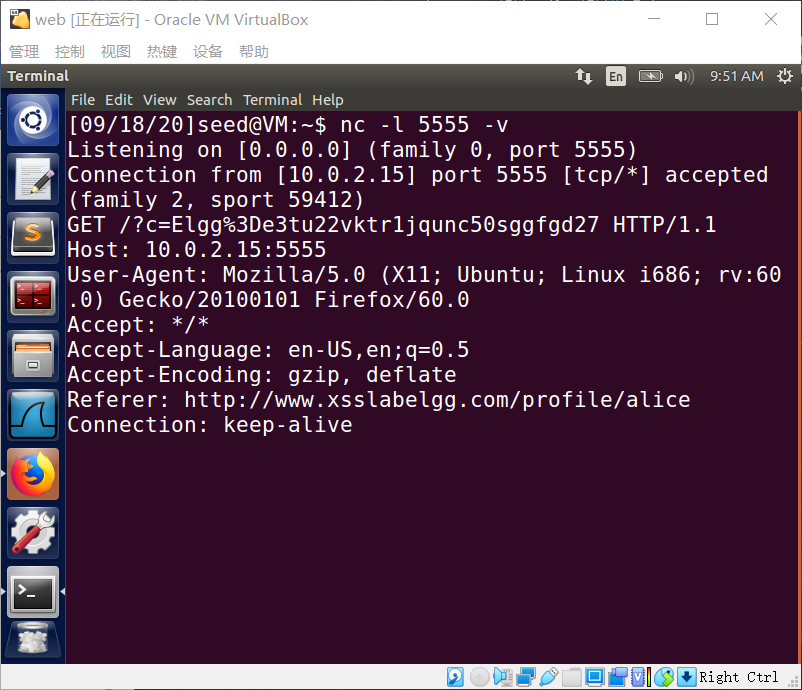
先点击右上键的edit HTML，然后输入以下代码：



然后用以下命令打开监听：



然后访问一下Alice的profile，发现终端已经打印出了攻击者的cookie

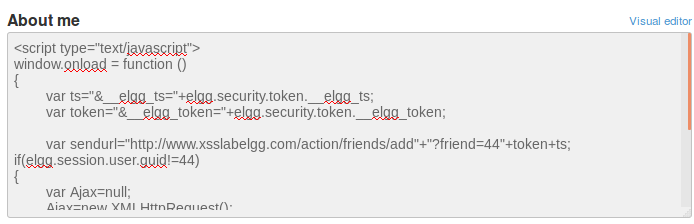


**Task 4: Becoming the Victim’s Friend**

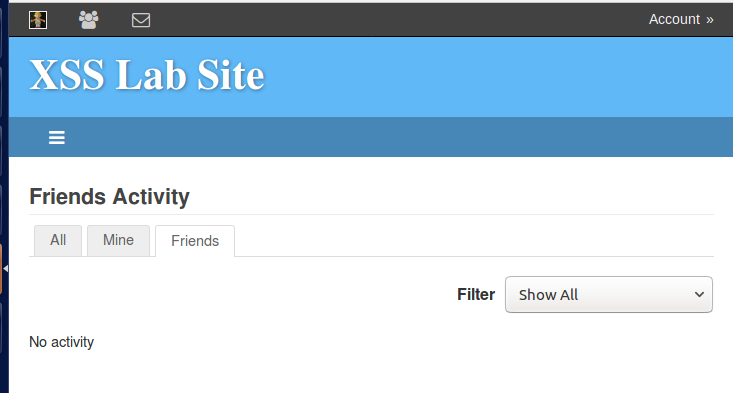
用Samy添加Alice好友，同时抓包获得Alice的id为44.



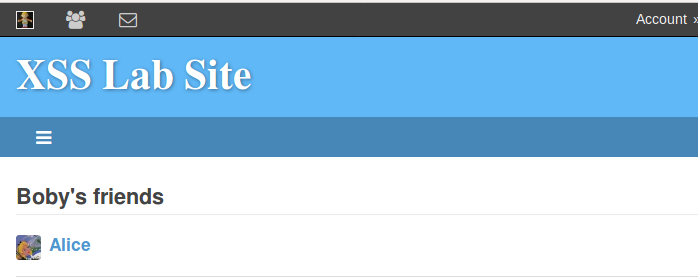
然后进入Alice账号，修改profile，先点击右上键的edit HTML，然后输入以下代码：



接着登录boby账号，可以发现此时在friend里面Alice不是Boby的好友。



然后用Boby访问Alice的profile，重新回到friend里面，发现Alice已经成为了Boby的好友：



Question 1: Explain the purpose of Lines ① and ②, why are they are needed?

第①行代码是为了获取\_\_elgg\_ts，第②行代码是为了获取\_\_elgg\_token，这两行代码是为了在发送的请求中能有正确的秘密令牌，即\_\_elgg\_ts和\_\_elgg\_token，才能实现攻击，否则攻击将不成功。

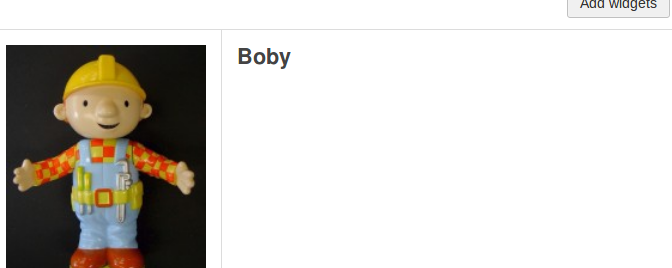
Question 2: If the Elgg application only provide the Editor mode for the "About Me" field, i.e.,you cannot switch to the Text mode, can you still launch a successful attack?

可以，通过使用浏览器扩展来删除请求中的格式化数据，进而成功攻击。

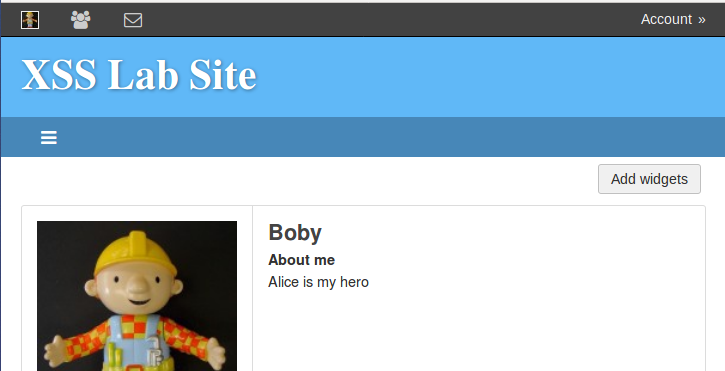
**Task 5: Modifying the Victim’s Profile**

在上个任务中已知Alice的id是44.进入Alice的账号，修改profile如下：

登录Boby的账号，首先查看自己的profile并无异常。

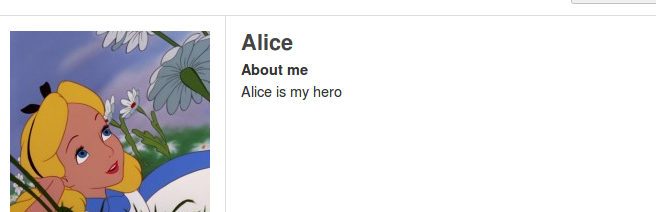


访问Alice的profile，再查看自己的profile，发现信息已被修改。



Question 3: Why do we need Line ①? Remove this line, and repeat your attack. Report and explain your observation.

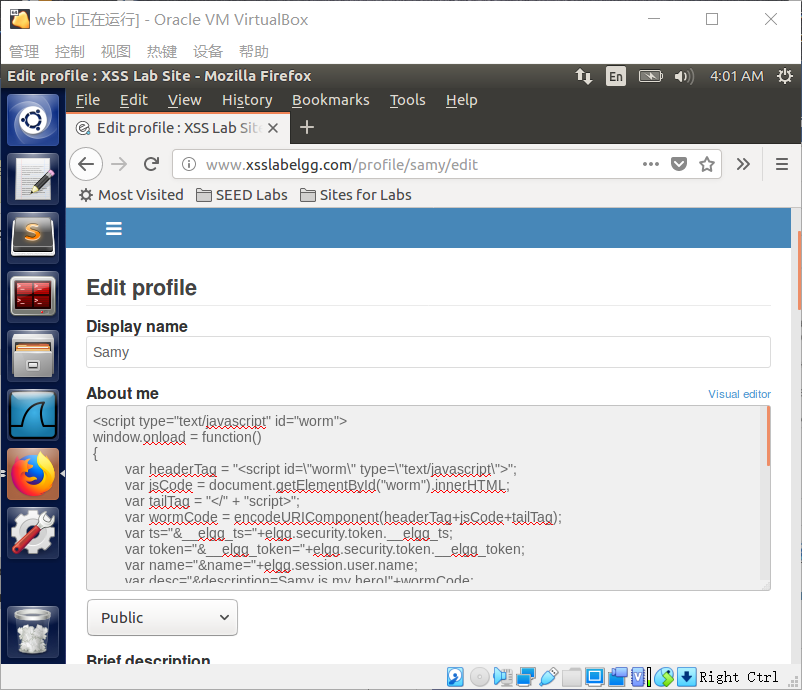
删除Line ①的代码后，发现Alice自己的主页被修改，而其他人访问后则无事发生。



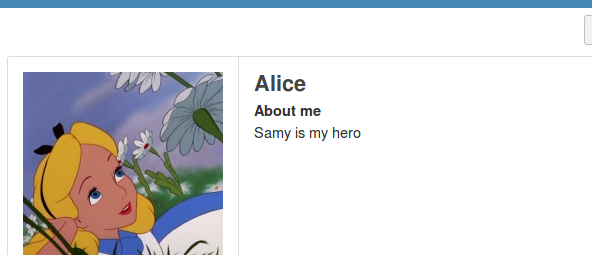
Line ①的代码是用来判断查看当前界面的id是否为攻击者自己的id。删去之后，Alice查看自己的profile后，会修改profile中的内容。其他人再进入Alice的profile，查看到的是修改之后的内容，因此攻击失败。

**Task 6: Writing a Self-Propagating XSS Worm**

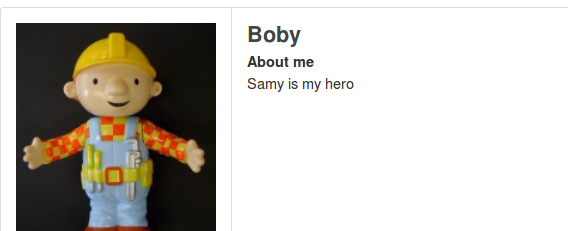
先登录samy的账号，修改profile如下：



然后登录Alice的账号访问Samy的profile，发现自己的profile被修改：

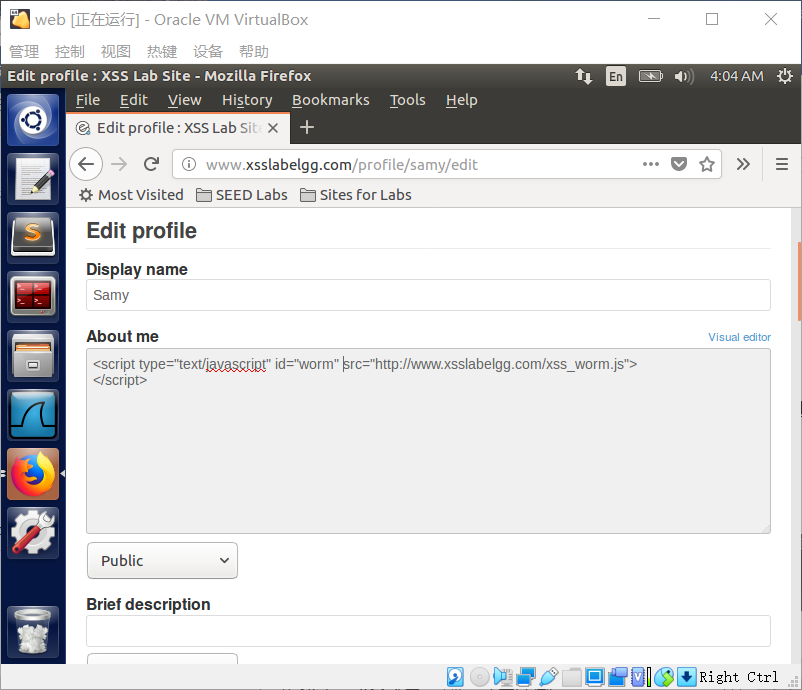


再用Boby的账号访问Alice的，发现也被修改：

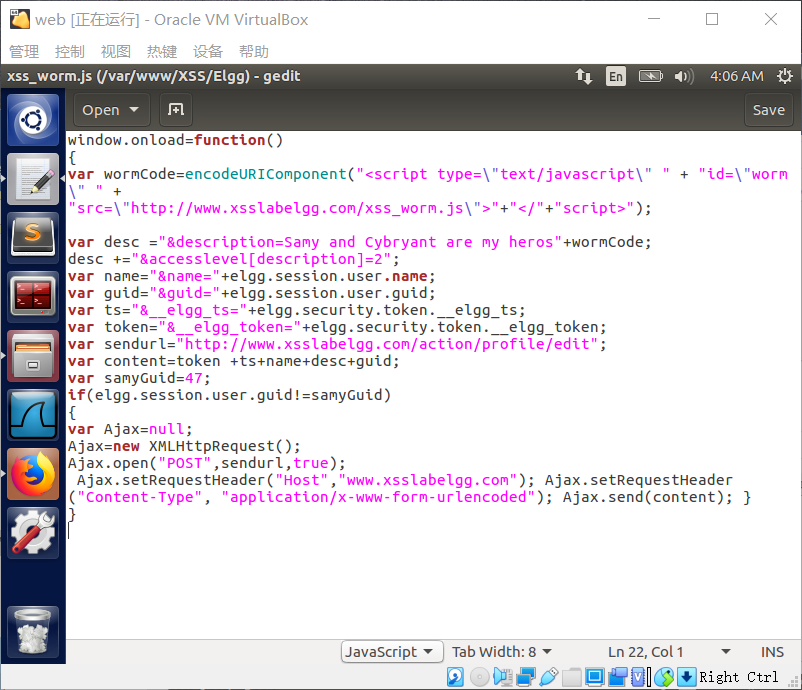


由此可知 dom方法能够传播worm。

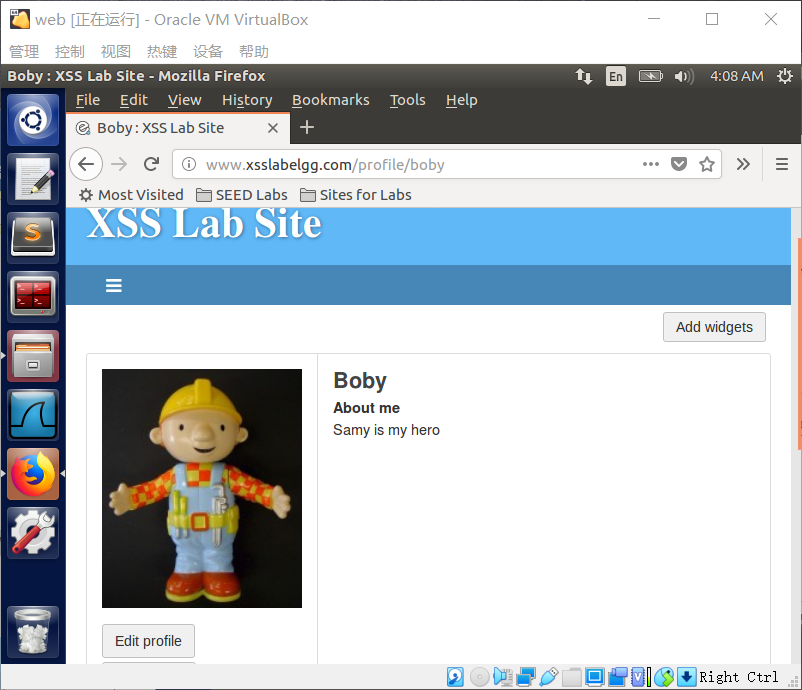
再来执行link，先修改samyd的file：



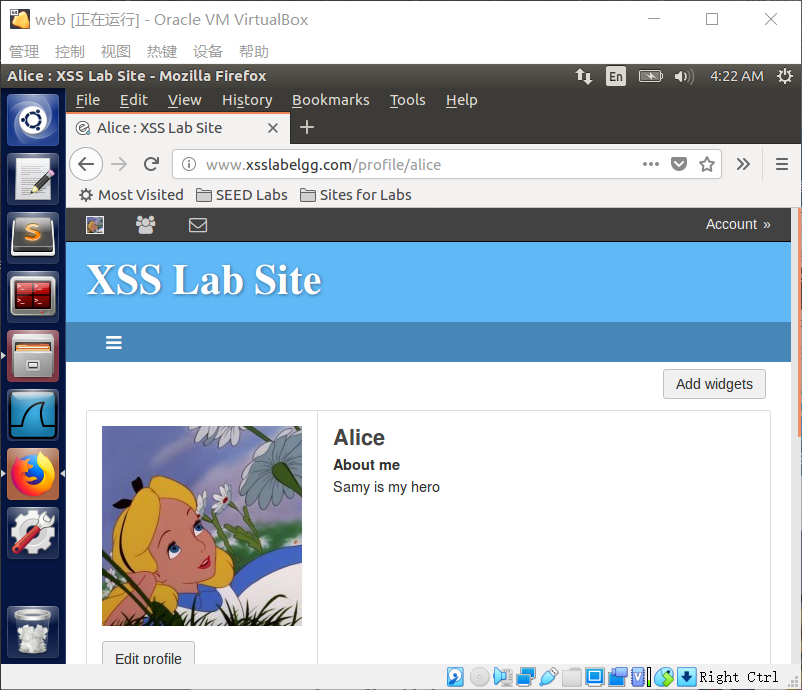
在/var/www/XSS/Elgg中创建文件xss\_worm.js，写入内容如下：



用Boby账号访问Samy：

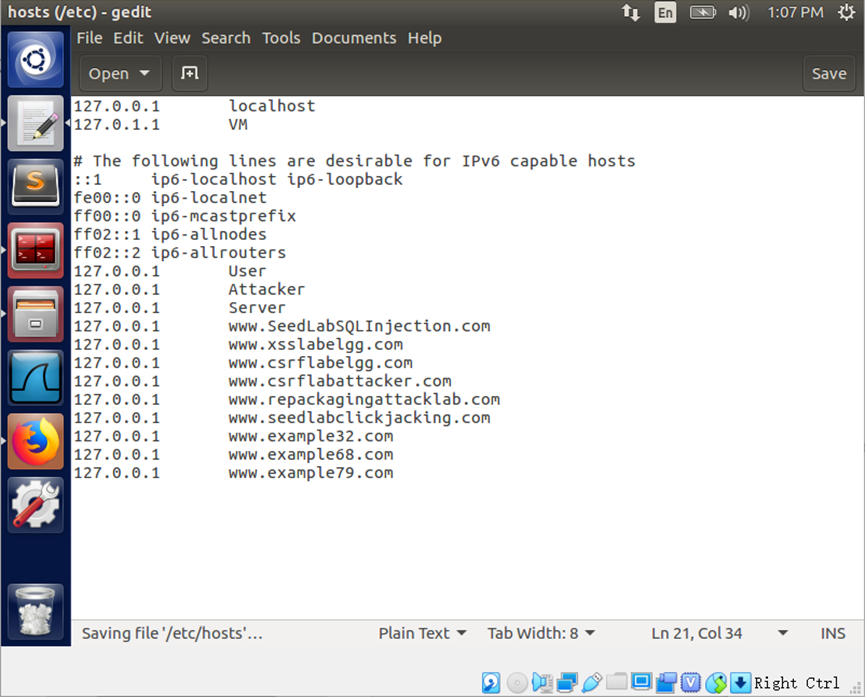


用Alice账号访问Boby，发现也被修改，所以link也能传播worm



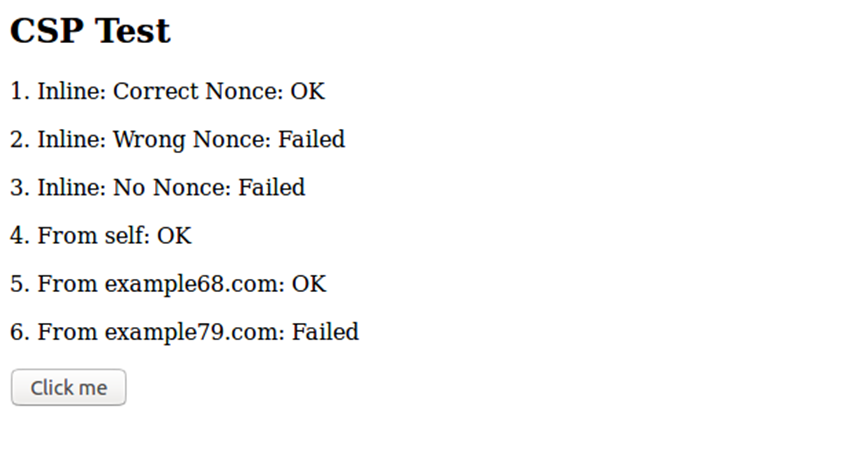
**Task 7: Defeating XSS Attacks Using CSP**

修改/etc/hosts 文件，添加最后三条语句如图所示。



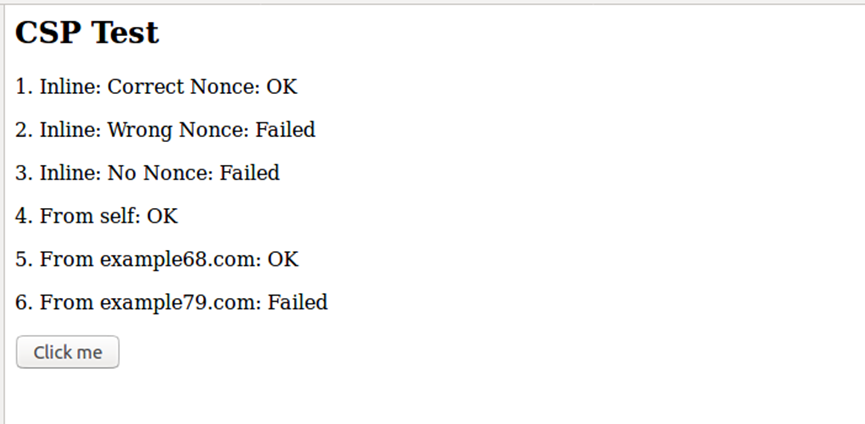
下载csp文件后解压，运行http\_server.py。

在浏览器输入<http://www.example32.com:8000/csptest.html>，结果为：



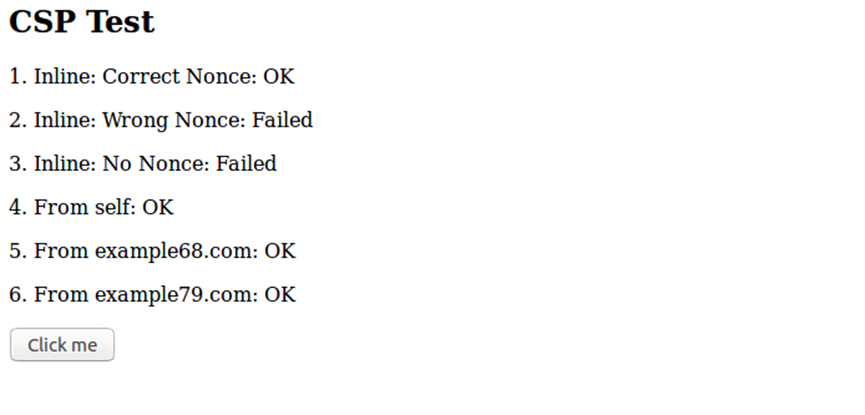
Field 1、4、5显示OK，其余为Failed。

进入http://www.example68.com:8000/csptest.html，结果为：



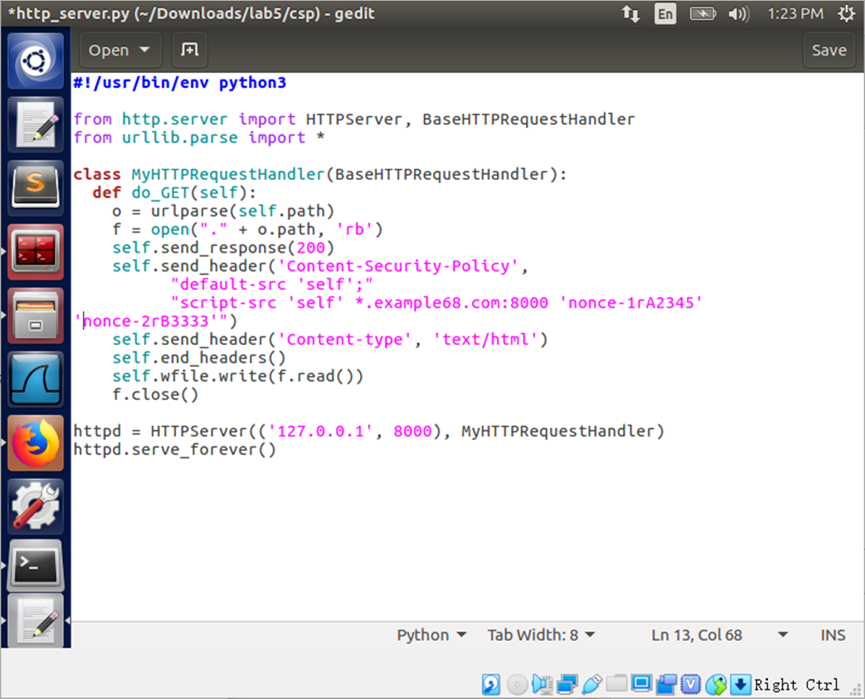
Field 1、4、5显示OK，其余为Failed。

进入http://www.example79.com:8000/csptest.html，结果为：



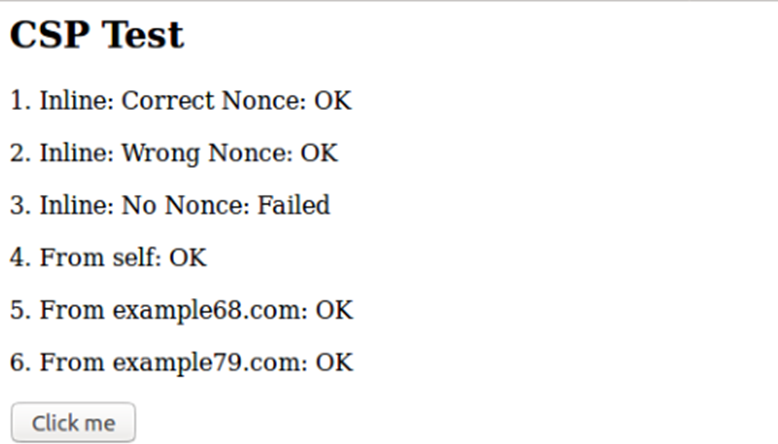
Field 1、4、5、6显示OK，其余为Failed。

修改 http\_server.py 文件，在 script-src 后添加 nonce-2rB3333。



重新编译运行修改后的 http\_server.py，再次在浏览器中输入

<http://www.example79.com:8000/csptest.html>，结果如下



1、2、4、5、6都显示OK。