《软件安全》实验报告

姓名：曹瑜　　学号：2212794　　班级：密码科学与技术

实验名称：

跨站脚本攻击

实验要求：

复现课本第十一章实验三，通过 img 和 script 两类方式实现跨站脚本攻击，撰写实验报告。有能力者，可以自己撰写更安全的过滤程序。

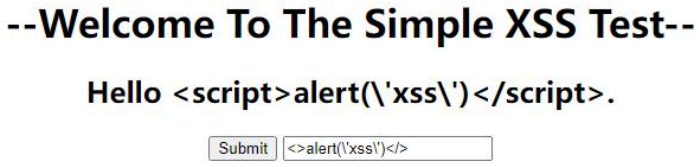
实验过程：

1、黑盒测试角度：

访问 URL：http://127.0.0.1/xss_test.php

显示页面如下：

输入学过的最简单的 XSS 脚本：<script>alert('xss')</script>来进行测试；点击 Submit 按钮以后，效果如下：



此时发现 Hello 后面出现了输入的内容，并且输入框中的回显过滤了 script 关键字，

这个时候考虑后台只是最简单的一次过滤，于是可以利用双写关键字绕过，构造脚本：

<scrscriptipt>alert('xss')</scscriptript>进行测试

执行效果如下：

发现虽然输入框中的回显为想要攻击的脚本，但是代码并没有执行；

原因是黑盒测试情况下，不能看到全部代码的整个逻辑，所以无法判断问题到底出在哪里；
此时在页面点击右键查看源码，尝试从源码片段中分析问题。右键源码如下：

File Edit View Help

```
1  <!DOCTYPE html><!--STATUS OK--><html>
2  <head>
3  <meta http-equiv="content-type" content="text/html;charset=utf-8">
4  <script>
5  window.alert = function()
6  {
7  confirm("Congratulations~");
8  }
9  </script>
10 </head>
11 <body>
12 <h1 align=center>--Welcome To The Simple XSS Test--</h1>
13 <h2 align=center>Hello &lt;scrscriptipt&gt;alert('xss')&lt;/scscriptript&gt;.</h2><center>
14 <form action=xss_test.php method=GET>
15 <input type=submit name=submit value=Submit />
16 <input name=keyword  value="<script>alert('xss')</script>">
17 </form>
18 </center></body>
19 </html>
```

接着查看到 16 行的<input>标签，正是唯一能输入且有可能控制的地方；
想办法将前面的<input>标签闭合，于是构造如下脚本：

"><scrscriptipt>alert('XSS')</scscriptript><!—

执行后弹出确认框，代表 XSS 攻击成功；
执行效果如下：

此时再从源码角度分析页面核心逻辑：

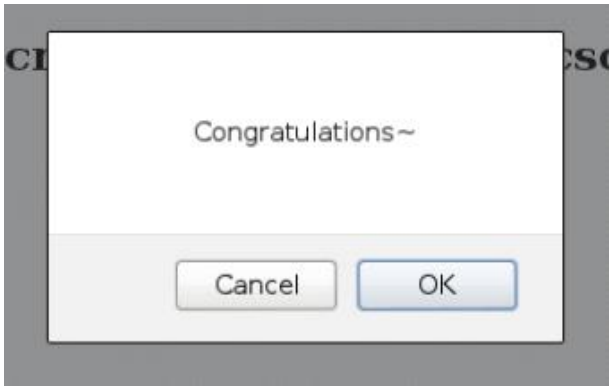
```
<?php
ini_set("display_errors", 0);
$str =strtolower( $_GET["keyword"]);
$str2=str_replace("script","",$str);
$str3=str_replace("on","",$str2);
$str4=str_replace("src","",$str3);
echo "<h2 align=center>Hello ".htmlspecialchars($str).".</h2>".'<center>
<form action=xss_test.php method=GET>
<input type=submit name=submit value=Submit />
<input name=keyword value="'.$str4.'">
</form>
```

```
    </center>';
    ?>
```

发现跟我们上面黑盒测试的情况差不多，但是也有没测试到的地方。比如，Hello 后面显示的值是经过小写转换的。输入框中回显值的过滤方法是将 script、on、src 等关键字都替换成了空，其实过滤的内容并不是很多。这也会导致攻击脚本的构造方法多种多样。

一种使用<img>标签的脚本构造方法：
<img src=ops! onerror="alert('XSS')">

弹出弹窗表明执行成功：



心得体会：

通过本次实验，成功复现课本第十一章实验三，通过 img 和 script 两类方式实现跨站脚本攻击，实现了黑盒白盒两种角度下的跨站攻击，成功学会了 php 网页攻击，实现基本的弹窗效果。