软件安全实验报告

学号: 2310764 姓名: 王亦辉 班次: 计科一班

1 实验名称：

跨站脚本攻击

2 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。

3 实验过程：

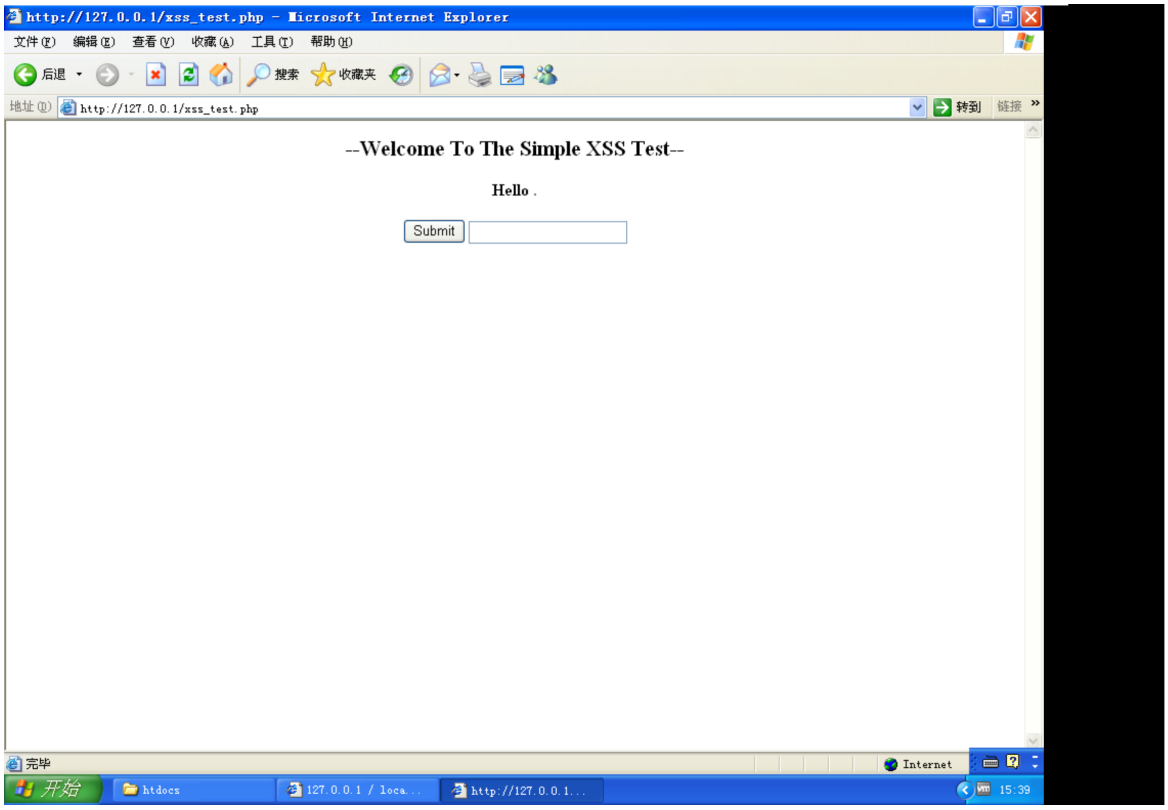
3.1 搭建受攻击网页

在 htdocs 下新建 `xss_test.php`，这是我们之后要攻击的网站

```
 <!DOCTYPE html>
<head>
<meta http-equiv="content-type" content="text/html;charset=utf-8">
<script>
window.alert = function()
{
confirm("Congratulations~");
 }
 </script>
 </head>
<body>
<h1 align=center>--Welcome To The Simple XSS Test--</h1>
<?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>
```

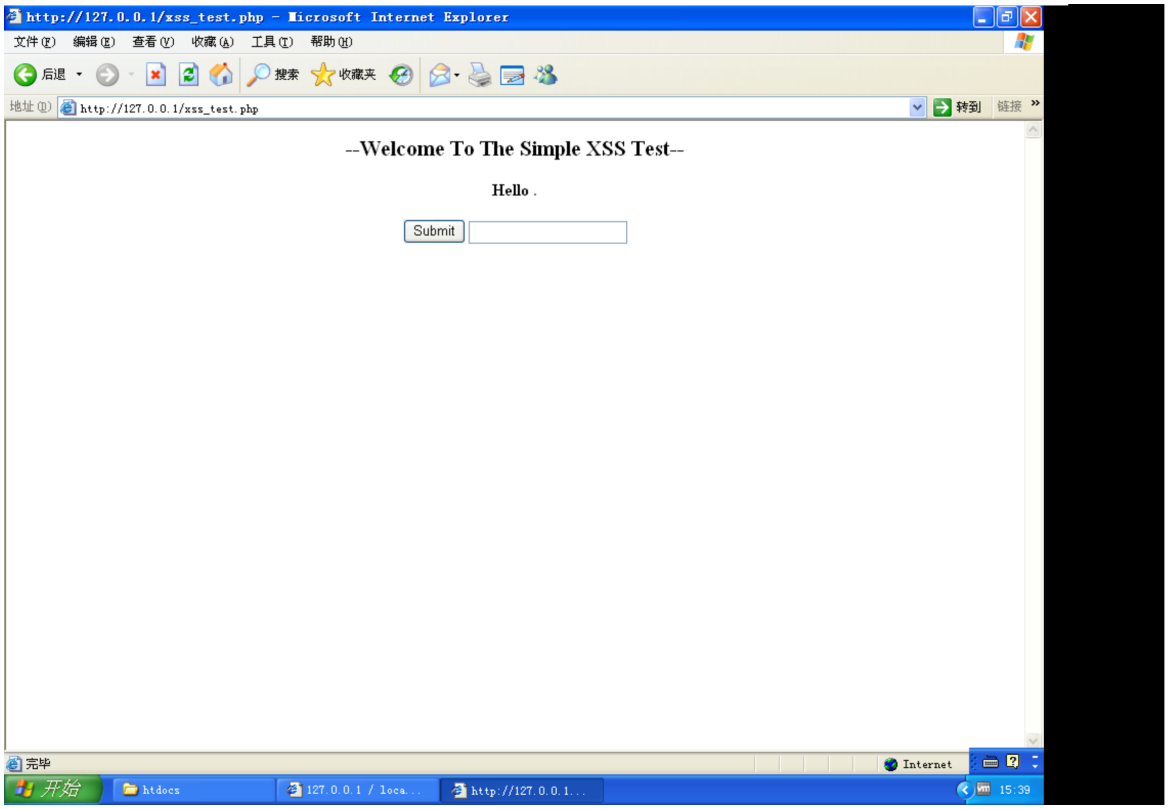
```
<input type=submit name=submit value=Submit />
<input name=keyword  value="'.$str4.'">
</form>
</center>';
?>
</body>
</html>
```

在地址栏输入 `http://127.0.0.1/xss_test.php` ,看到我们的网页已经成功搭建。

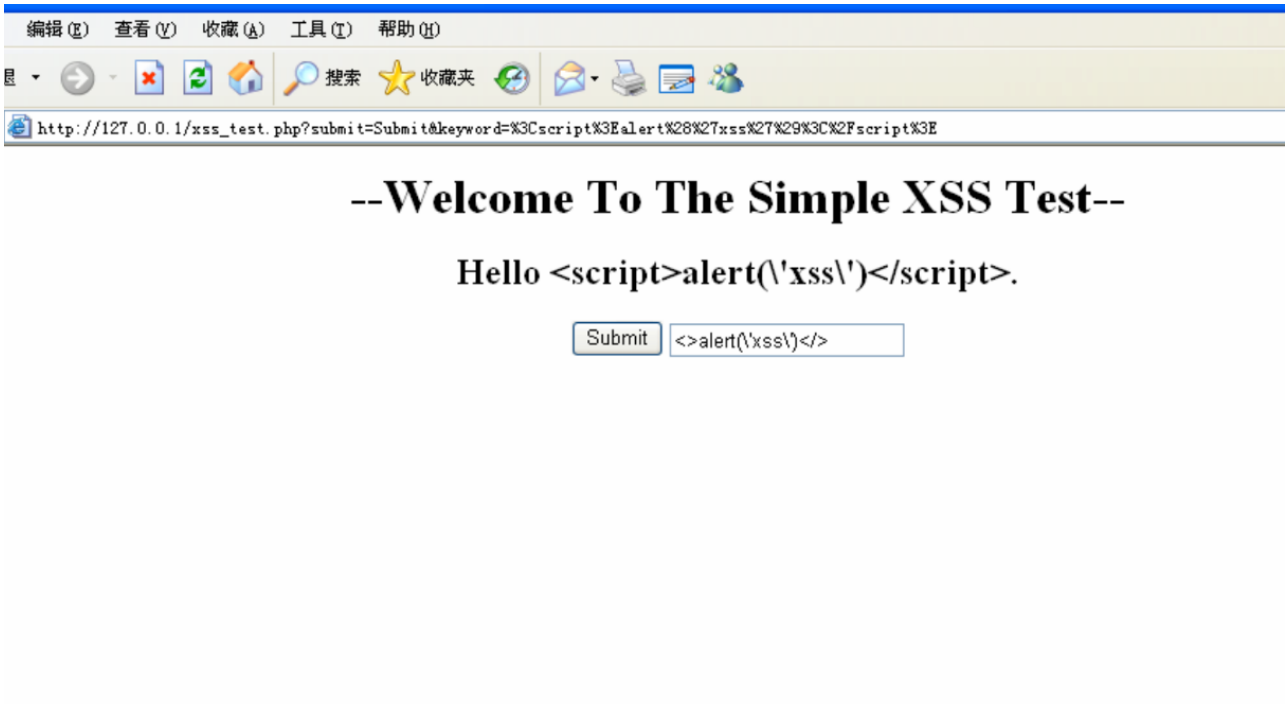


3.2 使用 script 进行攻击

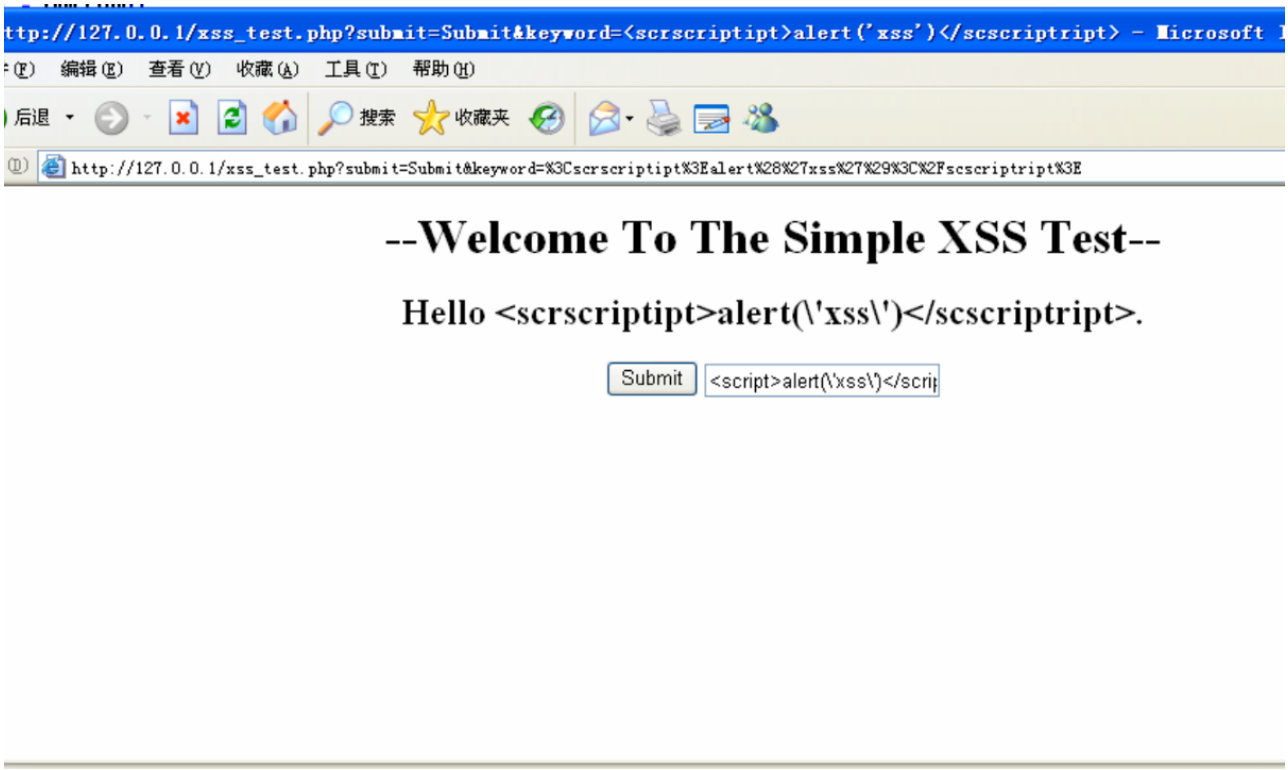
首先从黑盒测试的角度来进行实验：访问 URL：`http://127.0.0.1/xss_test.php` 页面显示效果如下：



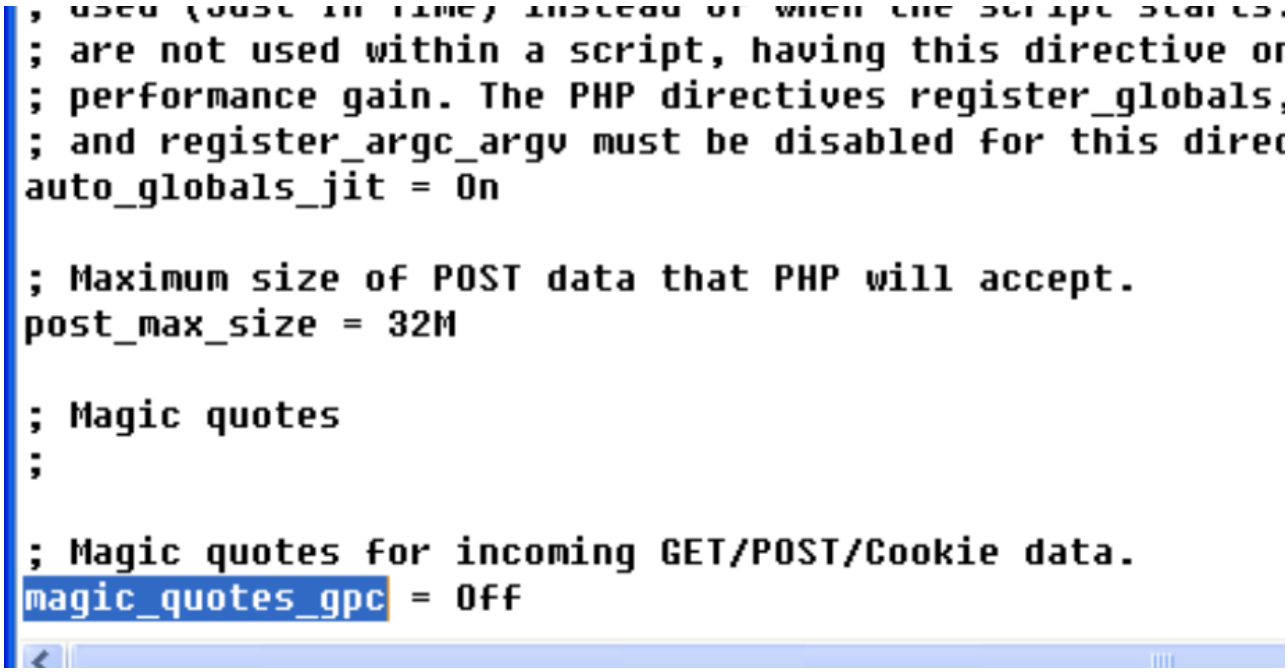
接着使用 `<script>alert ('xss')</script>` 提交进行测试，发现回显时被删掉了。



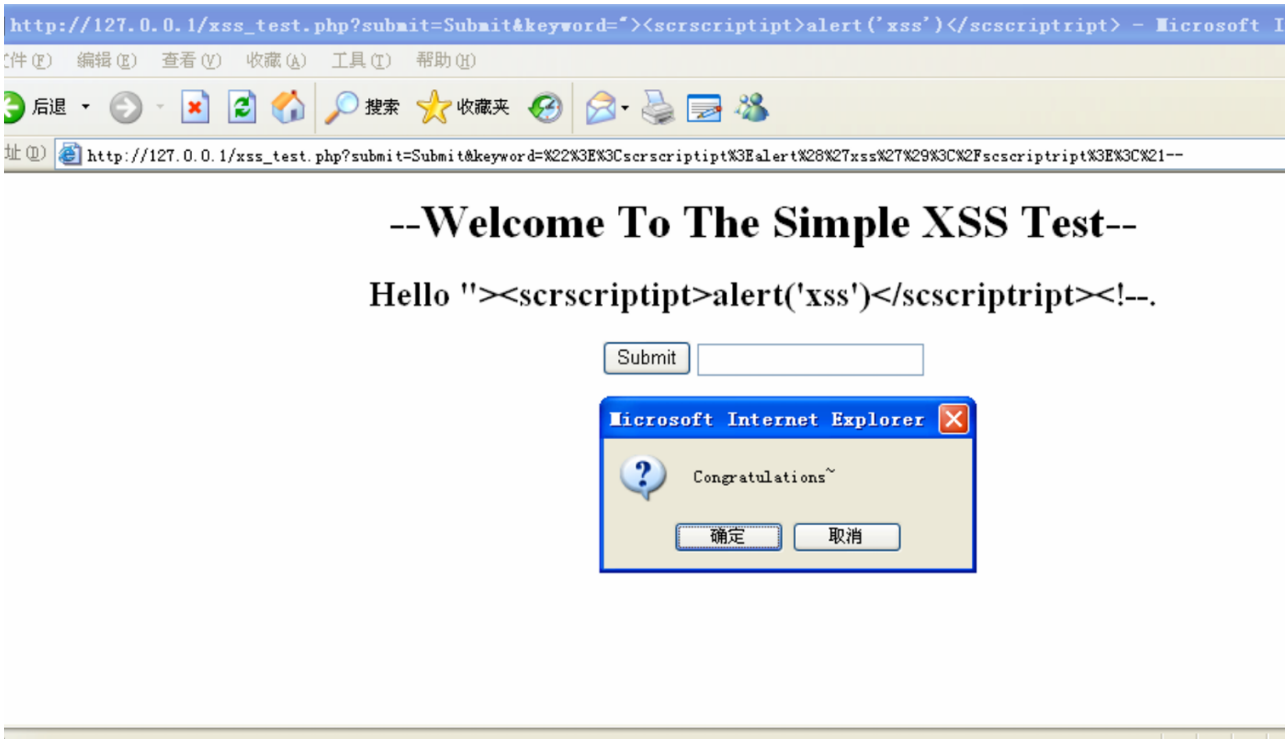
尝试双写突破，`<scrscriptipt>alert(‘xss')</scscriptript>`



将服务器的 `magic_guotes_gpc` 从 `On` 改为 `Off` ,这样服务器不会对输入的双引号进行转义。



想办法将前面的 `<input>` 标签闭合，于是构造如下脚本：`''><scrscriptipt>alert('XSS')</scscriptript><!-` 进行 Submit，可以看到攻击成功。



3.3 使用 img 进行攻击

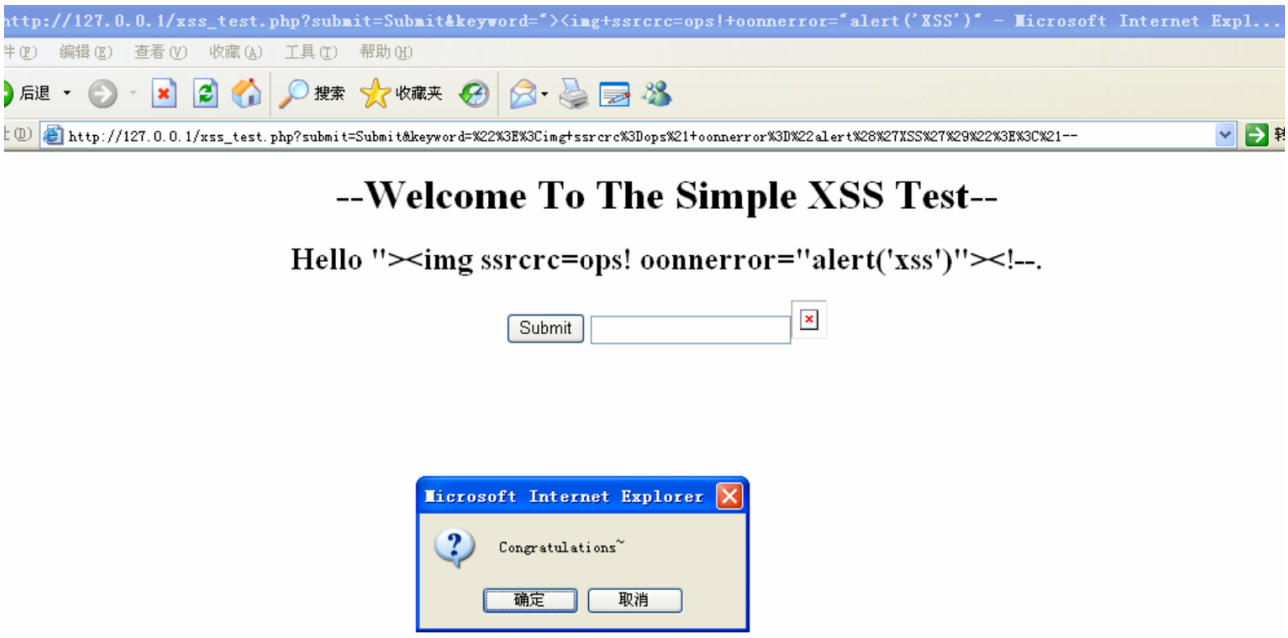
借助 img 标签的行为，我们也可以实现攻击，像 `<img src=ops! onerror="alert('xss')">` 这样的代码，当从 src 获取资源失败时，会触发 onerror 事件，从而执行相应函数，这是我们的攻击原理。

我们先修改 `xss_test.php` 以去除对 onerror 中的 on 关键词的删除。

```
<!DOCTYPE html>
<head>
<meta http-equiv="content-type" content="text/html;charset=utf-8">
<script>
window.alert = function()
{
confirm("Congratulations~");
 }
 </script>
 </head>
<body>
<h1 align=center>--Welcome To The Simple XSS Test--</h1>
<?php
```

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

使用 `"><img ssrcrc=ops! oonnerror="alert('XSS')">`，将前面标签闭合，以进行攻击，可以看到攻击成功。



4　心得体会：

本次实验中的这种跨站脚本攻击，需要用户有一定的配合，实行起来比较麻烦，但是也为我们揭示了网页的运作原理。我们通过一些操作，可以修改本地的网页，从而在本地执行一些脚本，这些网页时可以自由地被我们操作的。