Jarvis OJ -BASIC-WRITE-UP

原创

郁离歌 于 2018-04-05 17:23:01 发布 3226 收藏
分类专栏： CTF-WRITE-UP 文章标签： jarvisoj ctf学习 writeup basic

似乎寒假起就没有好好刷过ctf题了，感觉菜的和狗一样。愈发的感觉到无力。所以来JarvisOJ修炼了，写个WP记录一下。

BASIC

veryeasy

question：

使用基本命令获取flag

answer:

放winhex里面看下，没发现什么东西。题目说是基本命令，试了试binwalk，也什么都没有，再试strings。拿到flag。

关于USS Lab

question:

USS的英文全称是什么，请全部小写并使用下划线连接_，并在外面加上PCTF{}之后提交

answer:

百度谷歌一把梭。

base64?

question:

```
GUYDIMZVGQ2DMN3CGRQTONJXGM3TINLGG42DGMZXGM3TINLGGY4DGNBXGYZTGNLGGY3DGNBWMU3W
```

answer:**base32->hex->ascii**得flag。

-.-字符串

question:请选手观察以下密文并转换成flag形式

..-. .-.. .- --. ....--- ..--- ----- .---- ---.. -.. -.... -.... ....-- ---.. --... -.. .---- -.. .- ----. ...-- .---- ---.. .---- ..--- -... --... --... --... -.... ...-- ....- .---- -----

flag形式为32位大写md5

题目来源：CFF2016

answer:摩斯密码。解密得flag。

Help!!

question:出题人硬盘上找到一个神秘的压缩包，里面有个word文档，可是好像加密了呢~让我们一起分析一下吧！

answer:打开压缩包发现被加密了，放到winhex里面发现是伪加密，改一下得到word.docx,binwalk分离一下，在media文件夹下找到flag。

取证

---

answer:百度谷歌一把梭。

Secret

question:传说中的签到题

题目入口：http://web.jarvisoj.com:32776/

Hint1: 提交格式PCTF{你发现的秘密}

answer:f12一下，f5刷新一下页面，在 **请求头** 中发现flag。

veryeasyRSA

question:
已知RSA公钥生成参数：

e = 65537

求d =

请提交PCTF{d}

Hint1: 有好多小伙伴问d提交什么格式的，现在明确一下，提交十进制的d

answer:p和q都给了，直接上libnum模块求d，也可以自己算...

美丽的实验室logo

question：
出题人丢下个logo就走了，大家自己看着办吧。

answer：丢到winhex里面看看。没发现什么东西。然后丢到**stegsolve**里面浏览**frame**。拿到flag。

手贱

question：

某天A君的网站被日，管理员密码被改，死活登不上，去数据库一看，啥，这密码md5不是和原来一样吗？为啥登不上咧？

d78b6f302l25cdc811adfe8d4e7c9fd34

请提交PCTF{原来的管理员密码}

answer：md5哪来的l？去掉l,再爆破一下字符。解密得flag。

段子

question：

程序猿圈子里有个非常著名的段子：

手持两把锟斤拷，口中疾呼烫烫烫。

请提交其中"锟斤拷"的十六进制编码。(大写)

FLAG: PCTF{你的答案}

answer：百度谷歌一把梭。

握手包

question：

给你握手包，flag是Flag_is_here这个AP的密码，自己看着办吧。

提交格式：flag{WIFI密码}

answer:拿Elcomsoft Wireless Security Auditor或者aircrack-ng爆破一下密码，得flag。

德军的密码

question：
题目来源：CFF2016

answer：

```
a = '000000000000000000000000000000000000000000000000010111000011000100000010100000001'
b = 'WELCOMETOCFF'
c = ''
for i in range(len(a) / 7):
    c += a[i * 7: i * 7 + 7].zfill(8)
print hex(int(c, 2) ^ int(b.encode('hex'), 16))[2:-1].decode('hex')
# WELCOMECISRG
```

A Piece Of Cake

question：

nit yqmg mqrqn bxw mtjtm nq rqni fiklvbxu mqrqnl xwg dvmnzxu lqjnyxmt xatwnl, rzn nit uxnntm xmt zlzxuuk mtjtmmtg nq xl rqnl. nitmt vl wq bqwltwlzl qw yivbi exbivwtl pzxuvjk xl mqrqnl rzn nitmt vl atwtmxu xamttetwn xeqwa tsftmnl, xwg nit fzruvb, nixn mqrqnl ntwg nq gq lqet qm xuu qj nit jquuqyvwa: xbbtfn tutbnmqwvb fmqamxeevwa, fmqbtll gxnx qm fiklvbxu ftmbtfnvqwl tutbnmqwvbxuuk, qftmxnt xznqwqeqzluk nq lqet gtamtt, eqdt xmqzwg, qftmxnt fiklvbxu fxmnl qj vnltuj qm fiklvbxu fmqbtlltl, ltwlt xwg exwvfzuxnt nitvm twdvmqwetwn, xwg tsivrvn vwntuuvatwn rtixdvqm - tlftbvxuuk rtixdvqm yivbi evevbl izexwl qm qnitm xwvexul. juxa vl lzrlnvnzntfxllvldtmktxlkkqzaqnvn. buqltuk mtuxntg nq nit bqwbtfn qj x mqrqn vl nit jvtug qj lkwnitnvb rvquqak, yivbi lnzgvtl twnvnvtl yiqlt wxnzmt vl eqmt bqefxmxrut nq rtvwal nixw nq exbivwtl.

提交格式：PCTF{flag}

answer：词频分析。拿到quipquip上面去跑一下。拿到替换规则得flag。

Shellcode

question：

作为一个黑客，怎么能不会使用shellcode?

这里给你一段shellcode，你能正确使用并最后得到flag吗？

answer：用github上现成的shellcode执行器。跑一下就弹窗flag了。

https://github.com/inquisb/shellcodeexec

Easy Crackme

quetion:

都说逆向挺难的，但是这题挺容易的，反正我不会，大家来挑战一下吧~~:)

answer:拖进ida里面f5看一下加密过程，就是一个取余和异或的过程，逆一下得flag。

```
n = [251, 158, 103, 18, 78, 157, 152, 171, 0, 6, 70, 138, 244,
     180, 6, 11, 67, 220, 217, 164, 108, 49, 116, 156, 210, 160]
key = [0xab, 0xdd, 0x33, 0x54, 0x35, 0xef]
flag = ''
for i in range(26):
    flag += chr(n[i] ^ key[i % 6])
print flag
```

熟悉的声音

question：

两种不同的元素，如果是声音的话，听起来是不是很熟悉呢，据说前不久神盾局某位特工领便当了大家都很惋惜哦

XYYY YXXX XYXX XXY XYY X XYY YX YYXX

请提交PCTF{你的答案}

answer：仔细观察发现就X和Y和空格三种字符。应该是摩斯密码了。把X替换成. Y替换成- 得到

```
JBLUWEWNZ
```

交一下flag，发现并不对，应该是有意义的字符了，凯撒什么的跑一下。得flag。

ROPGadget

question:

都说学好汇编是学习PWN的基础，以下有一段ROPGadget的汇编指令序列，请提交其十六进制机器码(大写，不要有空格)

XCHG EAX,ESP

RET

MOV ECX,[EAX]

MOV [EDX],ECX

POP EBX

RET

提交格式：PCTF{你的答案}

answer:直接拿pwntools上。

```
from pwn import *
print asm('XCHG EAX,ESP\nRET\nMOV ECX,[EAX]\nMOV [EDX],ECX\nPOP
EBX\nRET'.lower()).encode('hex').upper()
```

Easy RSA

question：

还记得veryeasy RSA吗？是不是不难？那继续来看看这题吧，这题也不难。

已知一段RSA加密的信息为：0xdc2eeeb2782c且已知加密所用的公钥：

(N=322831561921859 e = 23)

请解密出明文，提交时请将数字转化为ascii码提交

比如你解出的明文是0x6162，那么请提交字符串ab

提交格式:PCTF{明文字符串}

answer：最简单的rsa，直接拿脚本跑了。

```
import libnum
from Crypto.Util.number import long_to_bytes

c = 0xdc2eeeb2782c
n = 322831561921859
e = 23
# http://factordb.com/index.php
q = 13574881
p = 23781539

d = libnum.invmod(e, (p - 1) * (q - 1))
m = pow(c, d, n)
print long_to_bytes(m)
```

爱吃培根的出题人

question：

听说你也喜欢吃培根？那我们一起来欣赏一段培根的介绍吧：

> *bacoN is one of aMerICa'S sWEethEartS. it's A dARlinG, SuCCulEnt fOoD tHAt PaIRs FlawLE*

什么，不知道要干什么？上面这段巨丑无比的文字，为什么会有大小写呢？你能发现其中的玄机吗？

提交格式：PCTF{你发现的玄机}

answer:培根密码，大小写的培根搞一下，出flag。

Baby's Crack

question：

既然是逆向题，我废话就不多说了，自己看着办吧。

answer：爆破字符吧。

```
import string

def ecrypto(a):
    if a != '\xff' and a:
        if ord(a) > 47 and ord(a) <= 96:
            a = chr(ord(a) + 53)
        elif ord(a) <= 46:
            a = chr(ord(a) + ord(a) % 11)
        else:
            a = chr(61 * (ord(a) / 61))
    return a

enc = r'jeihjiiklwjnk{ljj{kflghhj{ilk{k{kij{ihlgkfkhkwhhjgly'
m = ''
for i in enc:
    for j in string.printable:
        if i == ecrypto(j):
            m += j
            break
print m.decode('hex')
```

公倍数

question：

请计算1000000000以内3或5的倍数之和。

如：10以内这样的数有3,5,6,9，和是23

请提交PCTF{你的答案}

answer：妈耶，编程题。

```
n = 0
for i in xrange(1000000000):
    if i % 3 == 0:
        n += i
    elif i % 5 == 0:
        n += i
print n
```

神秘的文件

question：

出题人太懒，还是就丢了个文件就走了，你能发现里面的秘密吗？

answer：拖到winhex里面并没有发现什么东西，binwalk一下发现

可以挂载。mount一下看看。

mount haha ./1

打开发现有254个文件，那应该是要拼接了。我们使用cat或者dd拼接文件（头铁可以这样做）

其实只有把其中的文件分析一下，发现一个文件里面只有一个字符，写脚本把支付提取出来就行了。

```
path = 'lost+found/'
out= ''
for i in range( 254):
        out+=open(path+str(i), 'r').read()
print out
```

O98K，ak了basic。

开始看别的题了，加油！