

# 2019巅峰极客隐写

原创

[夜幕下的灯火阑珊](#)



于 2019-11-29 20:18:41 发布



192



收藏

文章标签: [misc](#)

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。

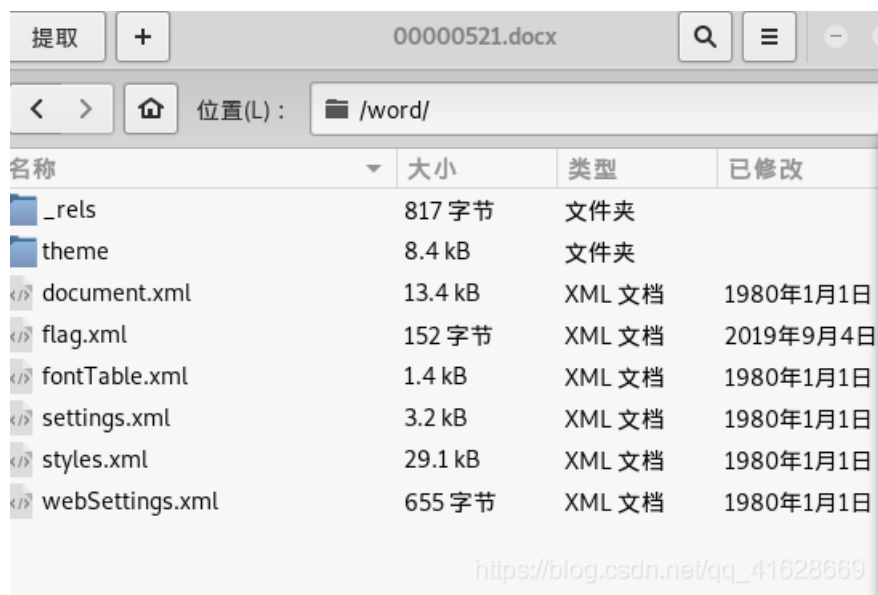
本文链接: [https://blog.csdn.net/qq\\_41628669/article/details/102647993](https://blog.csdn.net/qq_41628669/article/details/102647993)

版权

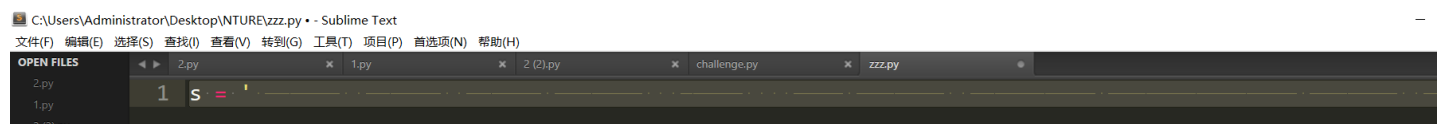
题目为一张png图片



foremost分离得到一个docx文件和一个加密压缩包  
docx文件分离可得flag.xml



打开flag.xml，一片空白，全选后发现有点东西，应该是隐藏文字，复制下来用sublime打开



把点和横杠记录下来



以0为点1为横杠转化成01字符串，再做处理，可得前半部分flag

```
#coding:utf-8
s = '.....'
k = ''
for i in s:
    if i == '.':
        k += '0'
    else:
        k += '1'
a = ''
b = [102,108,97,103,123,50,56,48,54,49,48,53,102,45,101,99,52,51,45]
for j in range(19):
    a += chr(b[j])
print a
#flag{2806105f-ec43-
```

b数组是由01字符串转化而得，舍去了尾部的一些字符

至此，前半部分完成

再看压缩包，需要密码

打开docx文件，可以看到一堆base64字符串，是base64隐写，从网上获得解码代码

```
#coding:utf-8
def get_base64_diff_value(s1, s2):
    base64chars = 'ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789+/'
    res = 0
    for i in xrange(len(s2)):
        if s1[i] != s2[i]:
            return abs(base64chars.index(s1[i]) - base64chars.index(s2[i]))
    return res

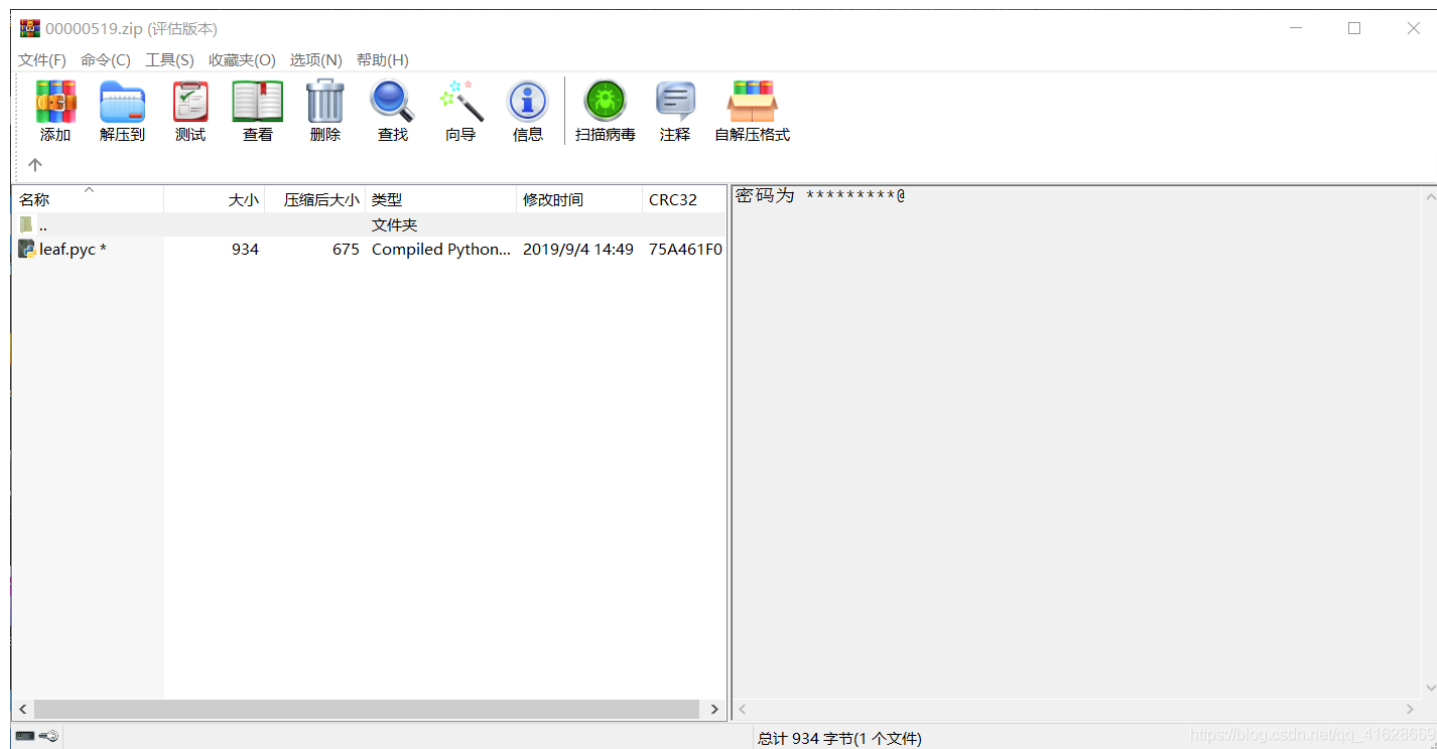
def solve_stego():
    with open('123456.txt', 'rb') as f:
        file_lines = f.readlines()
        bin_str = ''
        for line in file_lines:
            steg_line = line.replace('\n', '')
            norm_line = line.replace('\n', '').decode('base64').encode('base64').replace('\n', '')
            diff = get_base64_diff_value(steg_line, norm_line)
            print diff
            pads_num = steg_line.count('=')
            if diff:
                bin_str += bin(diff)[2:].zfill(pads_num * 2)
            else:
                bin_str += '0' * pads_num * 2
        print goflag(bin_str)

def goflag(bin_str):
    res_str = ''
    for i in xrange(0, len(bin_str), 8):
        res_str += chr(int(bin_str[i:i + 8], 2))
    return res_str

if __name__ == '__main__':
    solve_stego()
```

得到一堆字符串

由压缩包注释可知密码为l4mtHek3y@



在线反编译leaf.pyc可得

```

#!/usr/bin/env python 3.6 (3379)
#coding=utf-8
# Compiled at: 2019-09-04 01:49:32
#Powered by BugScanner
#http://tools.bugscanner.com/
#如果觉得不错,请分享给你朋友使用吧!
from numpy import *
from random import random
import turtle as t
t.reset()
x = array([[0.5], [0.5]])
p = [0.85, 0.92, 0.99, 1.0]
A1 = array([[0.85, 0.04],
            [-0.04, 0.85]])
b1 = array([[0], [1.6]])
A2 = array([[0.2, -0.26],
            [0.23, 0.22]])
b2 = array([[0], [1.6]])
A3 = array([[-0.15, 0.28],
            [0.26, 0.24]])
b3 = array([[0], [0.44]])
A4 = array([[0, 0],
            [0, 0.16]])
cnt = 1
while True:
    cnt += 1
    if cnt == 2000:
        break
    r = random()
    if r < p[0]:
        x = dot(A1, x) + b1
    else:
        if r < p[1]:
            x = dot(A2, x) + b2
        else:
            if r < p[2]:
                x = dot(A3, x) + b3
            else:
                x = dot(A4, x)
    t.color('green')
    t.color('green')
    t.up()
    t.goto(x[0][0] * 50, x[1][0] * 40 - 240)
    t.down()
    t.dot()

hint = 'I am not the reverse'

```

在kali下用stegosaurus解密可得后半部分flag

```
root@kali: ~/tools/zzctf/stegosaurus
文件(F) 编辑(E) 查看(V) 搜索(S) 终端(T) 帮助(H)
root@kali:~/tools/zzctf/stegosaurus# python3 '/root/tools/zzctf/stegosaurus/steg
osaurus.py' -x leaf.pyc
Extracted payload: 57f3-8cb4-1add2793f508}
root@kali:~/tools/zzctf/stegosaurus# python3 '/root/tools/zzctf/stegosaurus/steg
osaurus.py' -x leaf.pyc
Extracted payload: 57f3-8cb4-1add2793f508}
root@kali:~/tools/zzctf/stegosaurus#
```

```
P,LOOPBACK,RUNNING> mtu 65536
27.0.0.1 netmask 255.0.0.0
:1 prefixlen 128 scopeid 0x10<host>
velen 1000 (Local Loopback)
work 8 bytes 1512 (1.4 KiB)
ors 0 dropped 0 overruns 0 frame 0
kets 28 bytes 1512 (1.4 KiB)
ors 0 dropped 0 overruns 0 carrier 0
ktop#
```

CONTRIBUTORS.md

README.md

桌面

Documents

Music

Pictures

Videos

回收站

sql-connections

Desktop [https://blog.csdn.net/qq\\_41628669](https://blog.csdn.net/qq_41628669)



[创作打卡挑战赛](#)

[赢取流量/现金/CSDN周边激励大奖](#)