

西湖论剑2019-msc之奇怪的TTL

转载

[digupang7059](#) 于 2019-04-10 17:33:00 发布 671 收藏 1

原文链接: <http://www.cnblogs.com/mke2fs/p/10684883.html>

版权

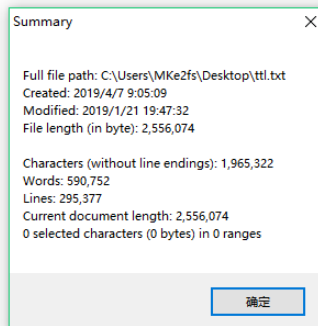
msc1给了一串很长的TTL字符, 参考一些隐写的文章, 猜测是在ttl中藏了信息, 题目是这样的

我们截获了一些IP数据报, 发现报文头中的TTL值特别可疑, 怀疑是通信方嵌入了数据到TTL, 我们将这些TTL值提取了出来, 你能看出什么

文本打开, TTL有29万行

S

```
1 TTL=127
2 TTL=191
3 TTL=127
4 TTL=191
5 TTL=127
6 TTL=191
7 TTL=127
8 TTL=191
9 TTL=127
10 TTL=191
11 TTL=127
12 TTL=63
13 TTL=63
14 TTL=255
15 TTL=191
16 TTL=63
17 TTL=127
18 TTL=191
19 TTL=127
20 TTL=191
21 TTL=127
22 TTL=191
23 TTL=127
24 TTL=191
25 TTL=127
26 TTL=191
27 TTL=127
28 TTL=127
29 TTL=63
30 TTL=255
31 TTL=63
32 TTL=127
33 TTL=63
34 TTL=255
35 TTL=63
36 TTL=63
37 TTL=63
38 TTL=255
```



分析一波之后发现一共有四种TTL值

```
63  00111111
127 01111111
191 10111111
255 11111111
```

可以看出后面六位全是1, 只有前面两位藏了数据, 也就是说一组TTL值可以隐藏一个字节

因此考虑写脚本跑。

下面是参考的一位大佬写的脚本, 本人就不献丑了

大佬写的writeup地址: <https://www.jianshu.com/p/13025b096f23>

```

f = open('ttl.txt','r')
TTL = f.readlines()
p = []
for i in TTL:
    p.append(int(i[4:]))
s = ''
for i in p:
    if i == 63:
        a = '00'
    elif i == 127:
        a = '01'
    elif i == 191:
        a = '10'
    elif i == 255:
        a = '11'
    s += a
print(type(s))
print(s)
import binascii
flag = ''
for i in range(0,len(s),8):
    flag += chr(int(s[i:i+8],2))
flag = binascii.unhexlify(flag)
wp = open('res1.jpg','wb')
wp.write(flag)
wp.close()

```

跑出来一张图片，jpg格式是首先计算前32位二进制位判断出来的



用foremost分离出六张二维码片段

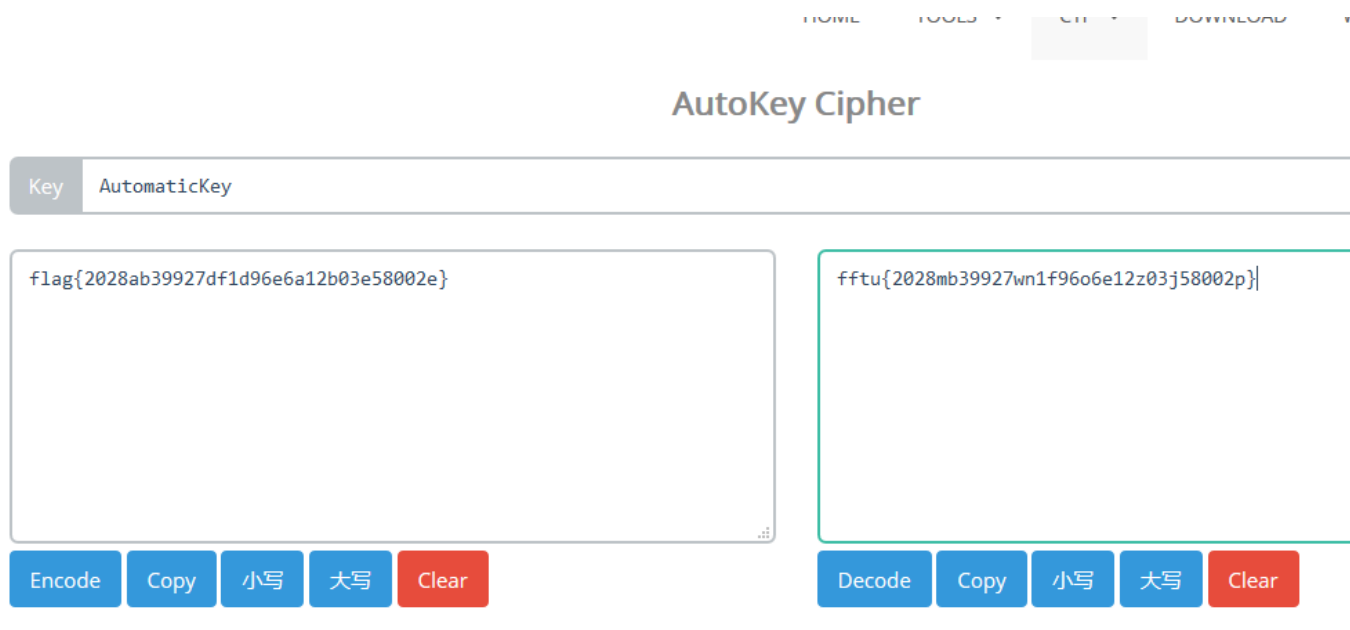


后面用PS把图片重新拼接，然后二维码识别



这里改正一下，根据key的信息，可以联想到一种常用的加密，Automatic加密

解密网站：<https://www.wishingstarmoye.com/ctf/autokey>



转载于：<https://www.cnblogs.com/mke2fs/p/10684883.html>