

攻防世界misc高手进阶篇教程（6）

原创

锋刃科技 于 2020-05-29 16:35:55 发布 2240 收藏 2

文章标签: [攻防世界 ctf](#)

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/xuandao_ahfengren/article/details/106428165

版权

halo

简单的base64解密会出现奇怪的字符



```
import string

from base64 import *

b=b64decode("aWdxNDs1NDZSOzFpa1I1MWliT08w")

data=list(b)

for k in range(0,200):

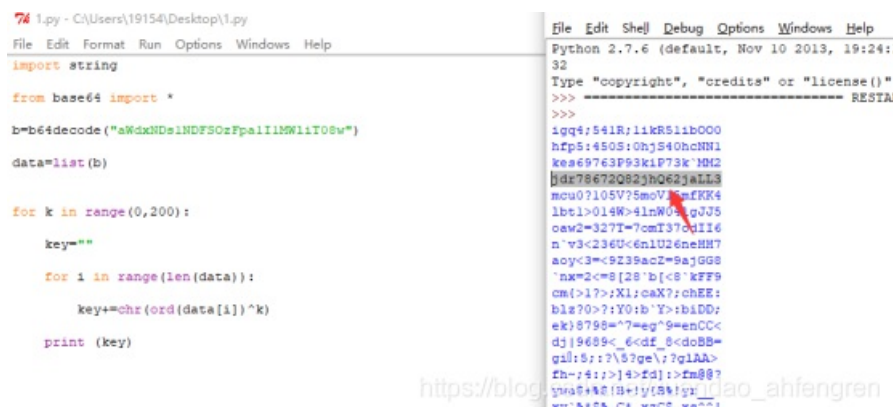
    key=""

    for i in range(len(data)):

        key+=chr(ord(data[i])^k)

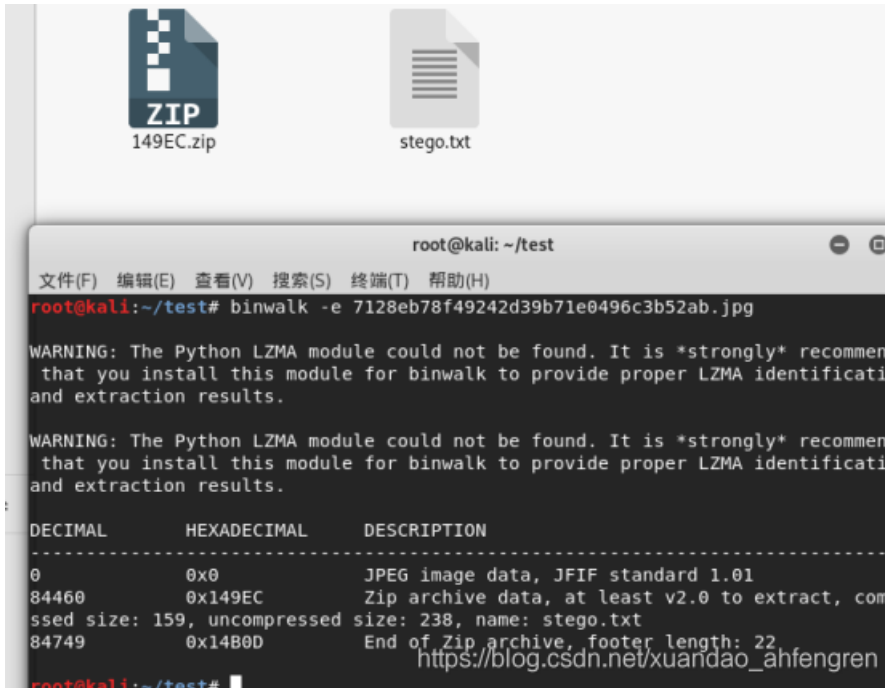
    print (key)
```

跑出来的结果有一个没有特殊符号



picture3

先用binwalk -e 分解出来



然后我们直接用stego.txt跑出结果即可

```

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

def solve_stego():

    with open('stego.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)

        pads_num = steg_line.count('=')
        if diff:
            bin_str += bin(diff)[2:].zfill(pads_num * 2)

        else:
            bin_str += '0' * pads_num * 2

    res_str = ''

    for i in xrange(0, len(bin_str), 8):

        res_str += chr(int(bin_str[i:i+8], 2))
    print res_str

solve_stego()

```

The screenshot shows a Python IDE window with the code from the previous block. The code is being executed, and the output is displayed in the console. The output is a long string of characters, including a copyright notice and the string 'Ba5e_640FivE'.

```

File Edit Format Run Options Windows Help
def get_base64_diff_value(s1, s2):
    base64chars = 'ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789+/'
    res = 0
    for i in xrange(len(s1)):
        if s1[i] != s2[i]:
            return abs(base64chars.index(s1[i]) - base64chars.index(s2[i]))
    return res

def solve_stego():

    with open('stego.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)

        pads_num = steg_line.count('=')
        if diff:
            bin_str += bin(diff)[2:].zfill(pads_num * 2)

        else:
            bin_str += '0' * pads_num * 2

    res_str = ''

    for i in xrange(0, len(bin_str), 8):

        res_str += chr(int(bin_str[i:i+8], 2))
    print res_str

solve_stego()

```

```

Python 2.7.6 (default, N
32
Type "copyright", "credit
>>>
Ba5e_640FivE
>>>

```

```

regexpire
from pwn import *
import rstr
import exrex
from time import sleep
import re

# conect to server
r = remote('misc.chal.csaw.io', 8001)

# Print the question string
print r.recvline()

# Counter
i=1

while True:
    # Recieve the regex pattern
    reg = r.recvline()[:-1]
    print "%d -----\n"%i
    print reg
    print "-----\n"
    ans=rstr.xeger(reg).replace('\n','') # Remove newlines!
    # ans=exrex.getone(reg).replace('\n','') # Another possible option
    r.sendline(ans)
    i+=1
    sleep(0.2)

```

flag{^regularly_express_yourself\$}

Crc

通过crc32爆破

然后把三个密码依次连接起来，后面发现密码是forum_91ctf_com_66

```

root@kali:~/crc32# python crc32.py reverse 0xCC86365B
4 bytes: {0x65, 0xd7, 0x1e, 0xf0}
verification checksum: 0xcc86365b (OK)
alternative: 05J728 (OK)
alternative: 0EvF7h (OK)
alternative: 2ysXnu (OK)
alternative: 3y2iul (OK)
alternative: R9DrOf (OK)
alternative: WQkoQX (OK)
alternative: avuKGt (OK)
alternative: d0875V (OK)
alternative: dSwk4B (OK)
alternative: forum (OK)
alternative: go3DvF (OK)
alternative: ldpDP2 (OK)
alternative: r6wKtc (OK)
alternative: s66zoz (OK)
alternative: yQGfVS (OK)

```

https://blog.csdn.net/xuandao_ahfengren

```
root@kali:~/crc32# python crc32.py reverse 0xBCEE7ED5
4 bytes: {0x1c, 0xeb, 0xe5, 0x41}
verification checksum: 0xbcee7ed5 (OK)
alternative: 2VSYDo (OK)
alternative: 50TgnD (OK)
alternative: 7sQy7Y (OK)
alternative: 9lctf_ (OK)
alternative: AVfsVk (OK)
alternative: N5K_u8 (OK)
alternative: 0YyCje (OK)
alternative: PgLpQi (OK)
alternative: aYUJmn (OK)
alternative: c425Xo (OK)
alternative: cePT4s (OK)
alternative: dlzWsP (OK)
alternative: pt05kx (OK)
alternative: rTzw3q (OK)
root@kali:~/crc32# python crc32.py reverse 0xccc7e74
4 bytes: {0x3f, 0x09, 0x32, 0xe4}
verification checksum: 0xccc7e74 (OK)
alternative: 1Atmb (OK)
alternative: 6XsSGI (OK)
alternative: EXFyUM (OK)
alternative: KWYiC (OK)
alternative: Qm0jH5 (OK)
alternative: Spkxd (OK)
alternative: TilZR0 (OK)
alternative: Uub7HB (OK)
alternative: ZfsjnA (OK)
alternative: cN30 z (OK)
alternative: com 66 (OK)
```

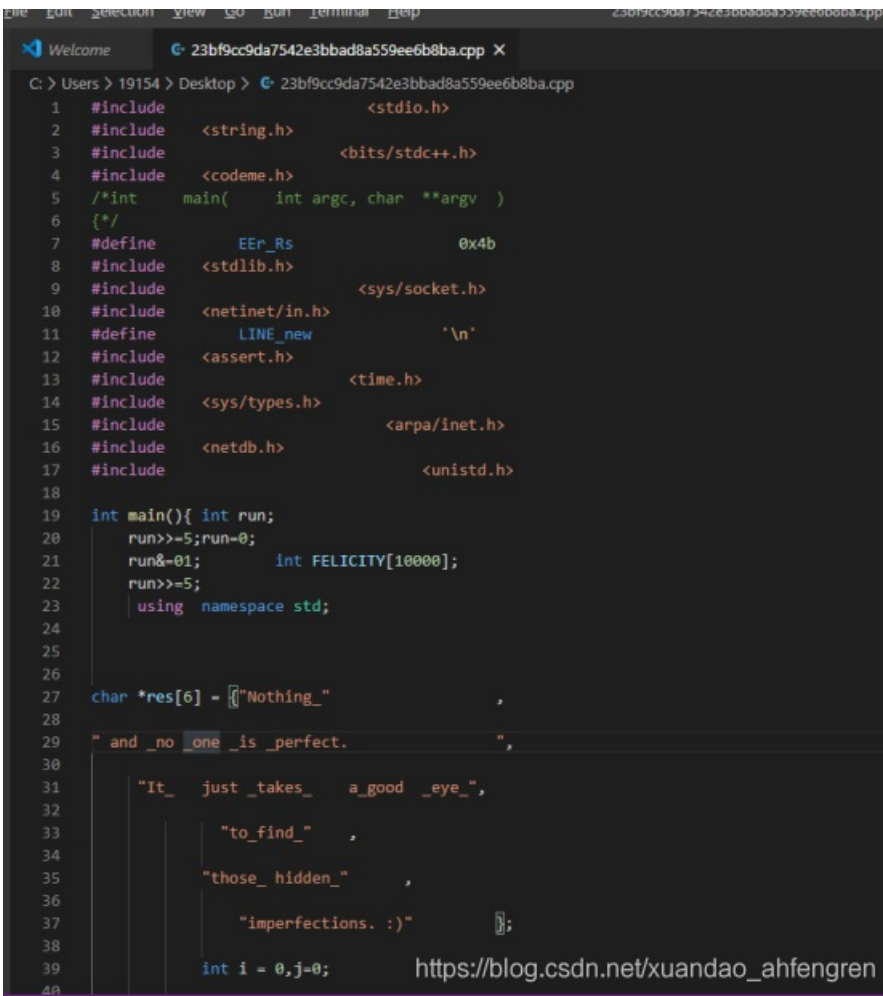
https://blog.csdn.net/xuandao_ahfengren

二进制转换成文本,然后保存到html文件中, 打开获得二维码, 扫描即可



https://blog.csdn.net/xuandao_ahfengren

下载源代码时，会注意到它具有所有这些奇怪的间距。



```
1 #include <stdio.h>
2 #include <string.h>
3 #include <bits/stdc++.h>
4 #include <codeme.h>
5 /*int main( int argc, char **argv )
6 {*/
7 #define EEr_Rs 0x4b
8 #include <stdlib.h>
9 #include <sys/socket.h>
10 #include <netinet/in.h>
11 #define LINE_new '\n'
12 #include <assert.h>
13 #include <time.h>
14 #include <sys/types.h>
15 #include <arpa/inet.h>
16 #include <netdb.h>
17 #include <unistd.h>
18
19 int main(){ int run;
20 run>>=5;run=0;
21 run&=01; int FELICITY[10000];
22 run>>=5;
23 using namespace std;
24
25
26
27 char *res[6] = {"Nothing_
28 " and_no_one_is_perfect.
29
30 "It_ just_takes_ a_good_eye_
31
32 "to_find_
33
34 "those_hidden_
35
36 "imperfections. :)"
37 };
38
39 int i = 0,j=0;
40
```

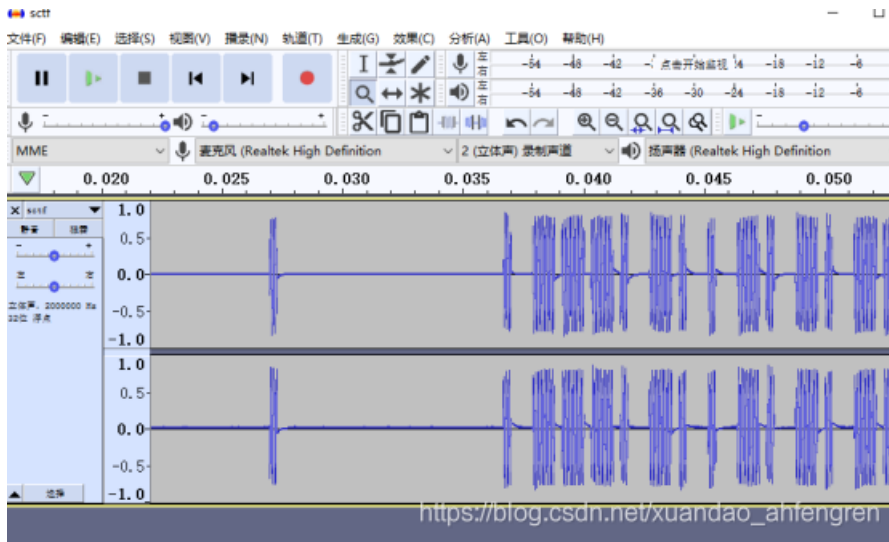
制表符本来是1，空格是0

再用python即可

```
>>> ''.join([ chr(int(i,2)) for i in x.split() ])
>>> x = "000001110100 000001101000 000001100101 000001011111 000001100110 000001
101100 000001100001 000001100111 000001011111 000001101001 000001110011 00000101
1111 000001010111 000001110000 000001010101 000001000001 000001001001 0000011101
00 000001110011 000001100001 000001100100 000001101101 000001101000 000001100001
000001101011 000001010 "
>>> ''.join([ chr(int(i,2)) for i in x.split() ])
'the_flag_is_WpUAItsadmhak\n'
>>>
```

打开电动车

是音频文件，直接打开



短的一段表示0，长的一段表示1

得到01111010010101001110

加上flag

sctf{01111010010101001110}

Hong

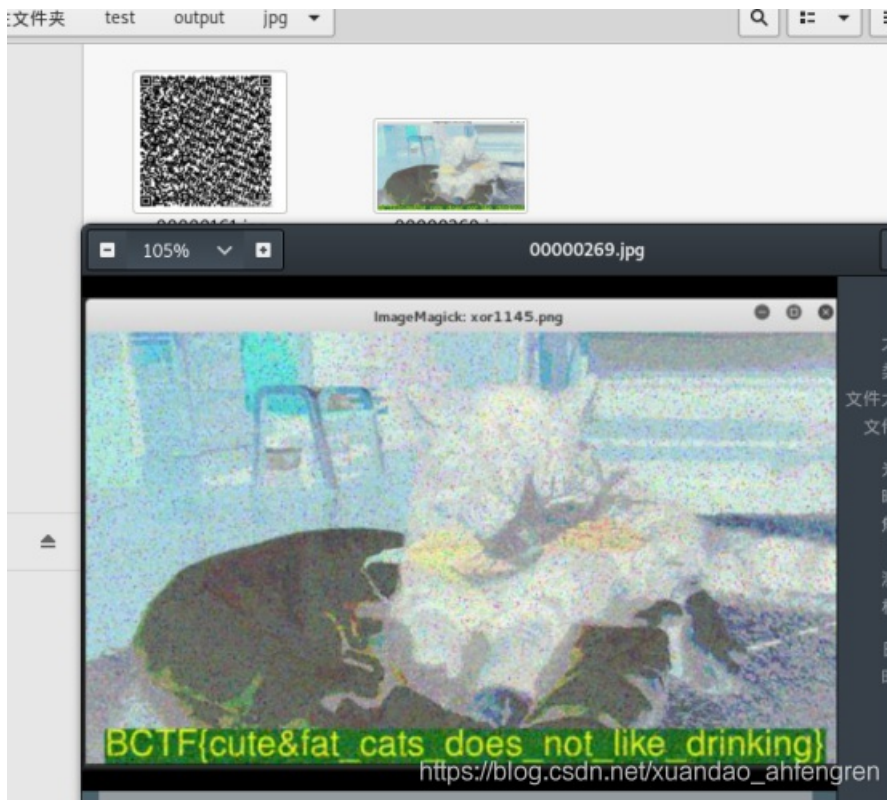
用binwalk分析发现有很多文件

```
root@kali:~/test# binwalk hong.mp3
```

DECIMAL	HEXADECIMAL	DESCRIPTION
55334	0xD826	Zlib compressed data, default compression
56734	0xDD9E	Zlib compressed data, default compression
82483	0x14233	JPEG image data, JFIF standard 1.01
82513	0x14251	TIFF image data, big-endian, offset of first image
directory: 8		
112115	0x1B5F3	Zlib compressed data, default compression
112861	0x1B8DD	Zlib compressed data, default compression
136938	0x216EA	Zlib compressed data, default compression
138170	0x21BBA	JPEG image data, JFIF standard 1.01
138200	0x21BD8	TIFF image data, big-endian, offset of first image
directory: 8		
192612	0x2F064	Zlib compressed data, default compression
194332	0x2F71C	Zlib compressed data, default compression
195518	0x2FB8E	Zlib compressed data, default compression
274685	0x430FD	Zlib compressed data, default compression
275977	0x43609	Zlib compressed data, default compression

foremost hong.mp3分解出来

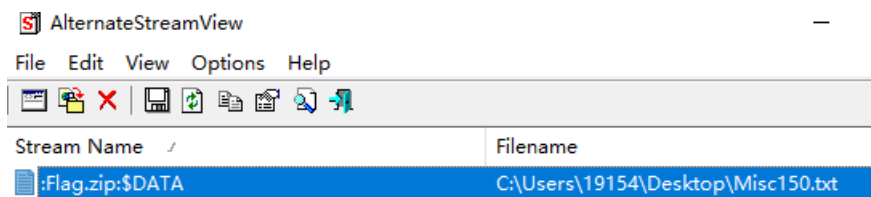
直接就发现了flag



BCTF{cute&fat_cats_does_not_like_drinking}

神奇的压缩文件

解压后使用 **AlternateStreamView** 扫描发现了 **flag.zip** 文件



压缩包注释段隐藏信息

将空格与TAB组成的空行转换为01

得到字符串 110110011000111110100110011011110 110110110110010001100110110110011
01110110111110010011001001111101

进行ASCII转换,得到flag: lctf{6d3677dd}

北京地铁

根据题目描述, 应该是AES-ECB加密, 目前通过低位隐写拿到了base64(ciphertext), 密钥16bytes未知.hint:Color Threshold 则通过gimp2查看Color Threshold, 等到hint: 发现魏公村颜色不对, 根据提示, 可能是密钥。


```

ERTYNBVCXSWERFGRDXCVBMNBVCDRGTGHUTYUIOJMWXSXTRFVBWSXNBVCXSWERFGRDXCVBZAQWDFRQWERTYUIOJMIUYH
ERFWSXCDEMNBVCDRGTGHUWSXWSXCDEZAQWDFRFRFBVBSXSCDEQWERTYWSXZAQWDFRQWERTYUHYHNBVWSXC
QWDFVRCVGRDQWERTYGRDXCVBQWERTYXSWEFYHNGRDXCVBTRFVBFVGYHNNWSXZAQWDFRWSXCDE,QWERTY
WSXCDEWSXCVWSXCQWERTYGRDXCVBIUYHNBVQWERTYTRFBTGBNMJUJYZAQWDFRWSXCFETGBNMJUJTRFBTYUI
DXCVBZAQWDFRWSXCFEQWERTYMNBNVCDRGTGHUWSXCDEGRDXCVBTRFBTYUIOJMWXSXZAQWDFRQWERTY
VGYHNBWSXCDEQWERTYUHYHNBVTGBNMJUJYMNBNVCDRGTGHUTYUIOJMQWERTYTGBNMJUYTRFVQWERTYGRDXCVBTY
UYHNBVQWERTYTRFVGTGBNMJUJYTGBNMJUJYZAQWDFRWSXCFEQWERTYWSXZAQWDFRQWERTYTYUIOJMRVGYHNI
GRDXCVBWSXCVQWERTYFVGYWDCFTTGBNMJUJYMNBNVCDRGTGHUWSXCVWSXCFEQWERTY
{WSX,WSXCDE,,QWERTYHNMKJTGBNMJUCVGRDQWERTYHNMKJTGBNMJUYTGBNMJUJYZAQWDFRQWERTYUJMEFVTQM
MJUYCVGRDQWERTYHNMKJTGBNMJUCVGRDQWERTYHNMKJTGBNMJUYTGBNMJUJYZAQWDFRQWERTYUJMEFVTQM
BVUYHNBVWSXVTGBNMJUJYZAQWDFRGRDXCVBWSXCVQWERTYUHYHNBVWSXCDETYUIOJMTYUIOJMWXSXZAQWDFR
CDEMNBVCDRGTGHUWSXCDEQWERTYGRDXCVBMNBVCDRGTGHUWSXCDEQWERTYEFVTTGBNMJUJYTGBNMJUMNBVCDR
GRD
[WSXIUHYHNBVTRFBTRFBQWERTYQAZSCEWSXCDEEFVTYHNMKJTGBNMJUYGRDXCVBMNBVCDRGTGHUWSXCFEQWER
CDEMNBVCDRGTGHU]QWERTYMNBNVCDRGTGHUWSXCDEEFVGYWSXCDEMNBVCDRGTGHUIUYHNBVWSXCDE-
WSXCDEZAQWDFRQWDFRQWERTYUJYZAQWDFRWSXCFEQWERTYUJYZAQWDFRQWERTYUJYZAQWDFRQWERTYUJYZAQWDFR
NMJUJYMNBNVCDRGTGHUQAZSCEQWERTYUHYHNBVZAQWDFRWSXTRFVGTFRVGSXZAQWDFRQWERTYUJYZAQWDFR
OJMTGBNMJUJYTRFBTGBNMJUJYWSXCVQWERTYGRDXCVBZAQWDFRGRDXCVBWSXCFEFTIUYHNBVWSXIUHYHNBV,Q
SXCDXSWEFYHNGWERTYGRDXCVBWSXCFEFSWEFYHNSXZAQWDFRWSXIUHYHNBVTYUIOJMMNBVCDRGTGHUGRD
R,QWERTYNBVCXSWERFMNBVCDRGTGHUTGBNMJUJYCVGRDQWERTYHNMKJTGBNMJUYTRFVGSXCFEQWERTYUJYZAQWDFR
WDVFRWSXCFEQWERTYTRFBMNBVCDRGTGHUEFVTNBVCXSWERFUYUIOJMGDXCVBZAQWDFRGRDXCVBWSXCFEFT

```

键盘字母连起来对应另一个字母:

```

TRFVG F
WSXCV L
GRDXCVB A
CVGRED G
{WSX I
IUYHNBV S
TRFVB C
TRFVB C
QWERTY -
QAZSCE K
WSXCDE E
EFVT Y
YHNMKJ b
TGBNMJUJ O
GRDXCVB A
MNBVCDRGTGHU R
WSXCFE D
QWERTY -
TRFVB C
WSX I
NBVCXSWERF P
RFVGYHN H
WSXCDE E
MNBVCDRGTGHU R}

```

得到flag: FLAG{ISCC-KEYBOARD-CIPHER}