




BUUCTF_pyre

原创

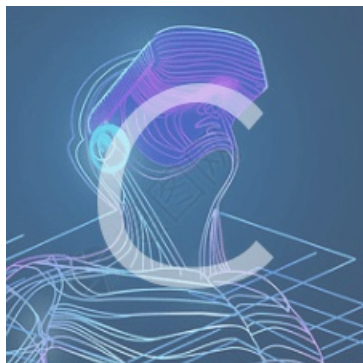
[ZYen12138](#)  于 2020-10-23 22:16:06 发布  214  收藏 1

分类专栏: [# BUUCTF 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/weixin_46009088/article/details/109250276

版权



[BUUCTF](#) 同时被 2 个专栏收录 

17 篇文章 2 订阅

订阅专栏



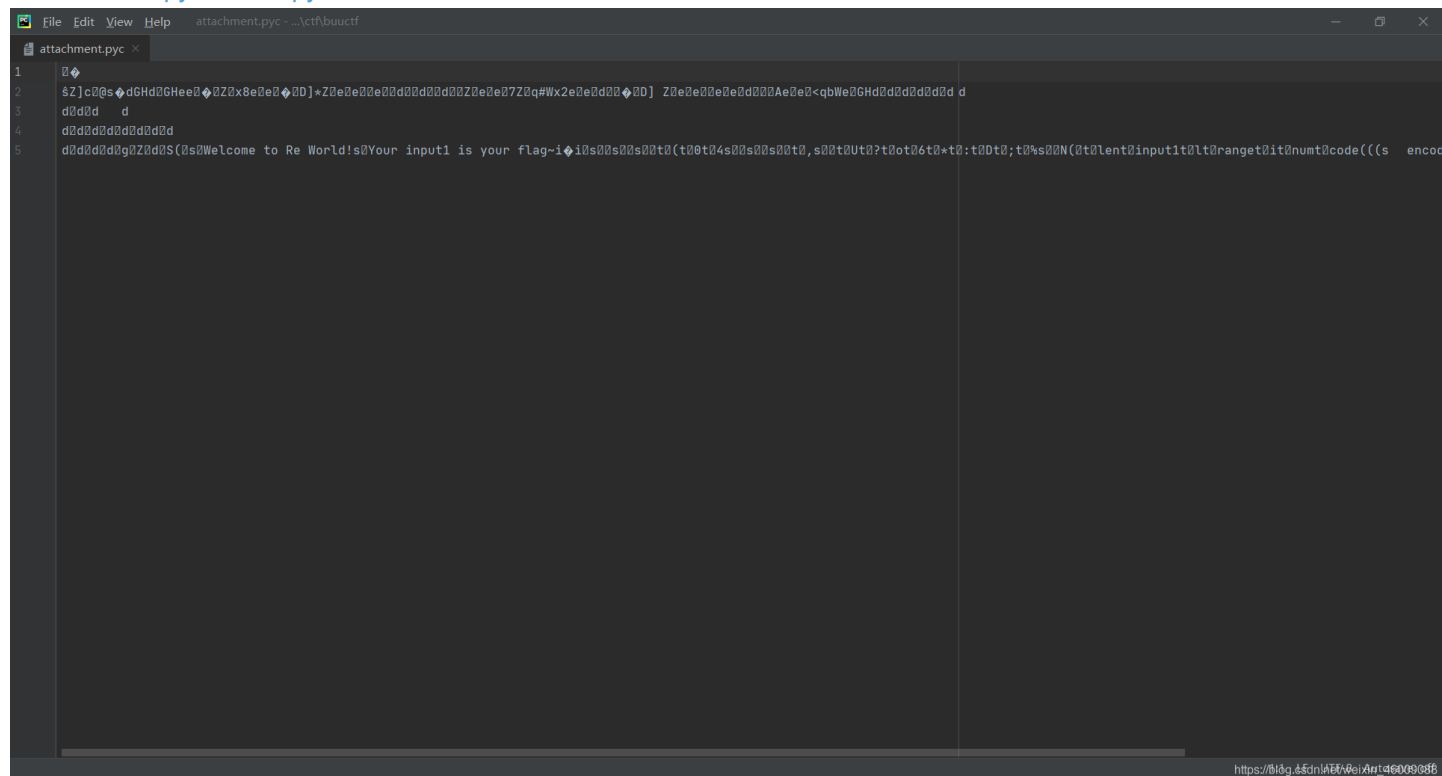
[CTF](#)

17 篇文章 0 订阅

订阅专栏

BUUCTF_pyre

解压得到一个py文件，用pycharm打开，如图：



一堆乱码??? 打开反编译网站把它丢进去.1

```
print 'Welcome to Re World!'
print 'Your input1 is your flag~'
l = len(input1)
for i in range(l):
    num = ((input1[i] + i) % 128 + 128) % 128
    code += num

for i in range(l - 1):
    code[i] = code[i] ^ code[i + 1]

print code
code = [
    '\x1f',
    '\x12',
    '\x1d',
    '(',
    '0',
    '4',
    '\x01',
    '\x06',
    '\x14',
    '4',
    ',',
    '\x1b',
    'U',
    '?',
    'o',
    '6',
    '*',
    ':',
    '\x01',
    'D',
    ';',
    '%',
    '\x13']
```

宋与工具 Python 在线工具 清空

```

1 #!/usr/bin/env python
2 # visit http://tool.lu/pyc/ for more information
3 print 'Welcome to Re World!'
4 print 'Your input1 is your flag~'
5 l = len(input1)
6 for i in range(l):
7     num = ((input1[i] + i) % 128 + 128) % 128
8     code += num
9
10 for i in range(l - 1):
11     code[i] = code[i] ^ code[i + 1]
12
13 print code
14 code = [
15     '\x1f',
16     '\x12',
17     '\x1d',
18     '(',
19     '0',
20     '4',
21     '\x01',
22     '\x06',
23     '\x14',
24     '4',
25     ',',
26     '\x1b',
27     'U',
28     '?',
29     'o',
30     '6',
31     '*',
32     ':',
33     '\x01',
34     'D',
35     ';',
36     '%',
37     '\x13']

```

输入flag

然后对你输入的flag进行第一次操作

第二次操作

经过两轮加密后的flag

Welcome to l
Your input1 i

Traceback (m
File: script.p
l = len(inp
NameError: n

Exited with e

https://blog.csdn.net/weixin_46009088

我们将其算法进行逆向，写出python脚本：

```

code = [
    '\x1f',
    '\x12',
    '\x1d',
    '(',
    '0',
    '4',
    '\x01',
    '\x06',
    '\x14',
    '4',
    ',',
    '\x1b',
    'U',
    '?',
    'o',
    '6',
    '*',
    ':',
    '\x01',
    'D',
    ';',
    '%',
    '\x13']
l = len(code)
for i in range(l - 2, -1, -1):
    code[i] = chr(ord(code[i]) ^ ord(code[i + 1]))

for i in range(len(code)):
    print(chr((ord(code[i]) - i) % 128),end='')

```

总结:

1.

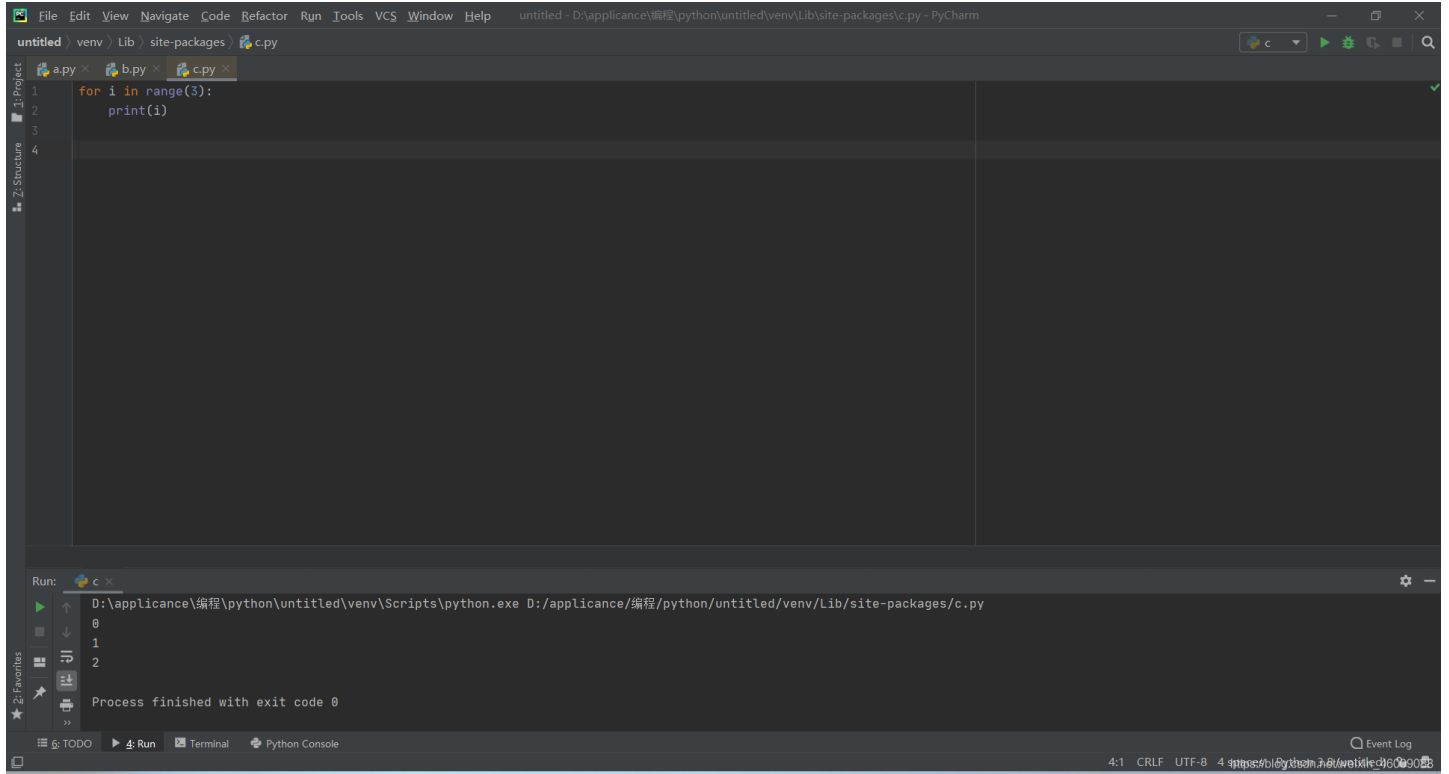
$(a+b)\%c == (a\%c+b\%c)\%c$

$((input1[i] + i) \% 128 + 128) \% 128 = ((input1[i] + i) \% 128 \% 128 + 128 \% 128) \% 128 = (input1[i] + i) \% 128$

2.

因为异或出现了 $i+1$ 所以需要反着遍历,for in 循环参数实验如下:

①一个参数



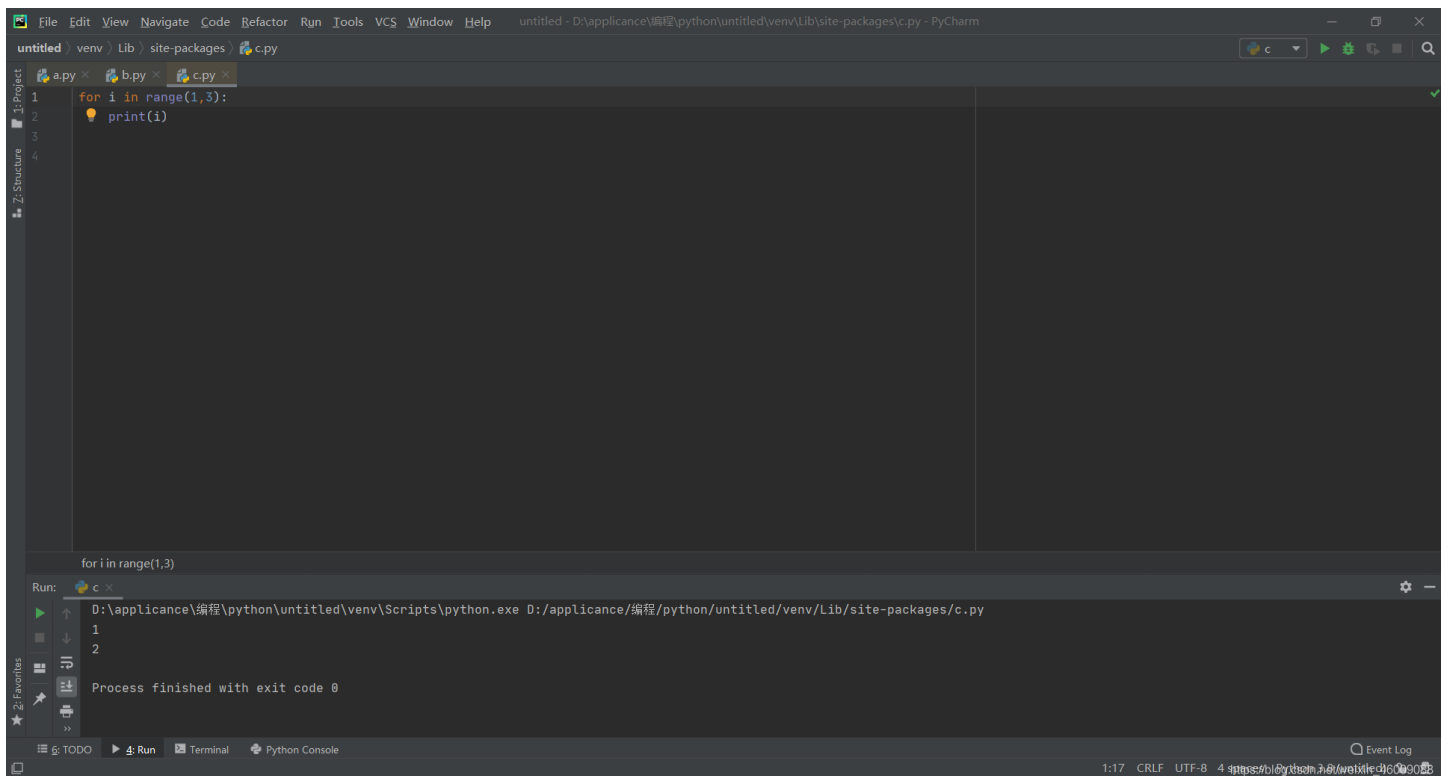
The screenshot shows the PyCharm IDE with a Python file named 'c.py'. The code in the editor is:

```
1 for i in range(3):  
2     print(i)  
3  
4
```

The Run console at the bottom shows the execution output:

```
Run: c x  
D:\apppliance\编程\python\untitled\venv\Scripts\python.exe D:/apppliance/编程/python/untitled/venv/Lib/site-packages/c.py  
0  
1  
2  
Process finished with exit code 0
```

②两个参数



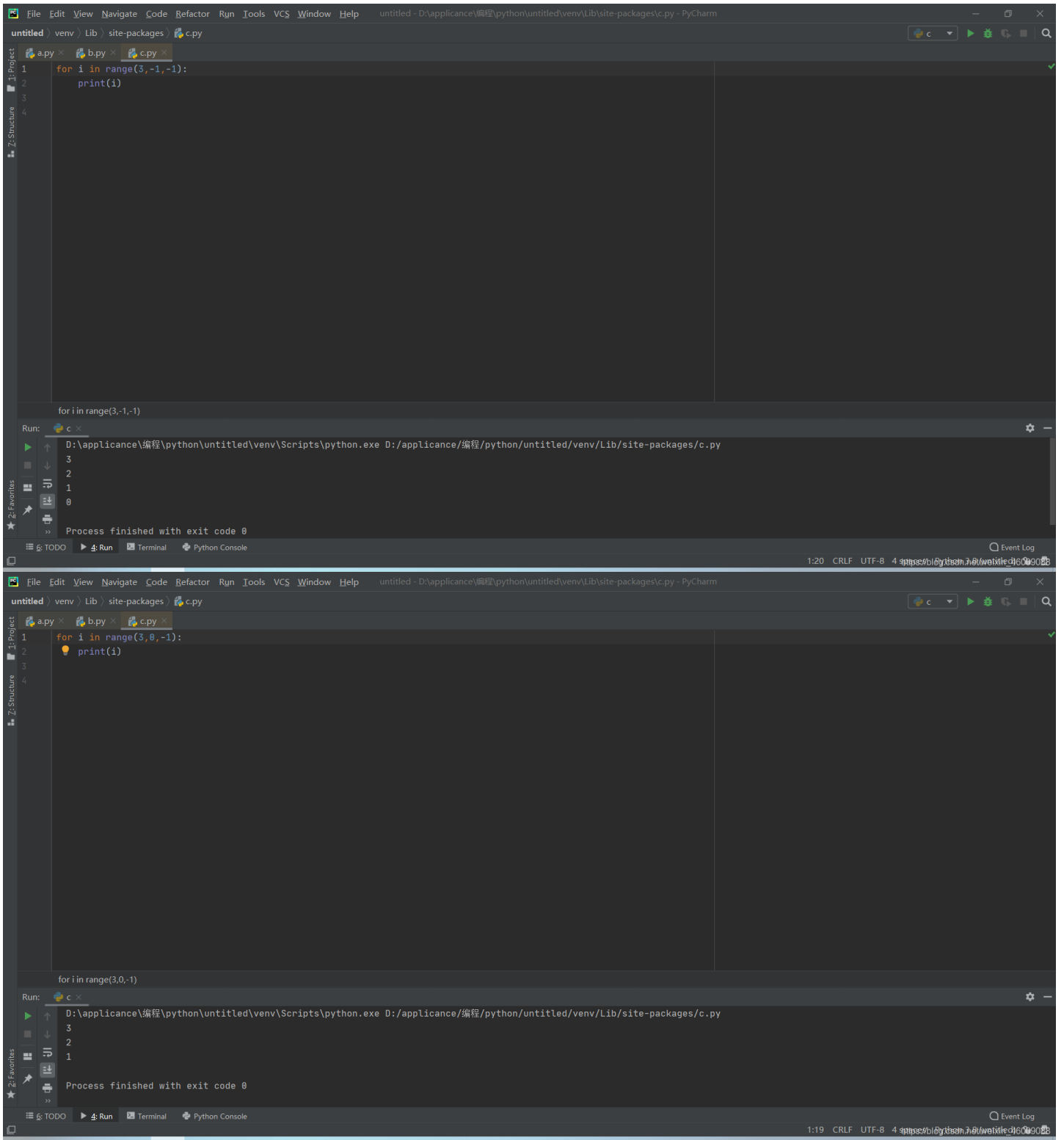
The screenshot shows the PyCharm IDE with a Python file named 'c.py'. The code in the editor is:

```
1 for i in range(1,3):  
2     print(i)  
3  
4
```

The Run console at the bottom shows the execution output:

```
Run: c x  
D:\apppliance\编程\python\untitled\venv\Scripts\python.exe D:/apppliance/编程/python/untitled/venv/Lib/site-packages/c.py  
1  
2  
Process finished with exit code 0
```

③三个参数 (第一个为初始值, 第二个为结束值, 第三个为迭代值)



3.

`print` 打印完了结束会加给换行符，`end = ''` 是将结尾的换行换成空字符串，然后接下来的字符将接下去就不换变成下面这样：

```
G  
W  
H  
T  
{  
J  
u  
s  
t  
-  
R  
e  
-  
1  
s  
-  
H  
a  
6  
6  
y  
!  
}
```

https://blog.csdn.net/weixin_46009088

反编译的网站: <https://tool.lu/pyc/> ↩️