

[羊城杯 2020]login writeup

原创

禾兮兮 于 2021-12-29 22:20:09 发布 401 收藏

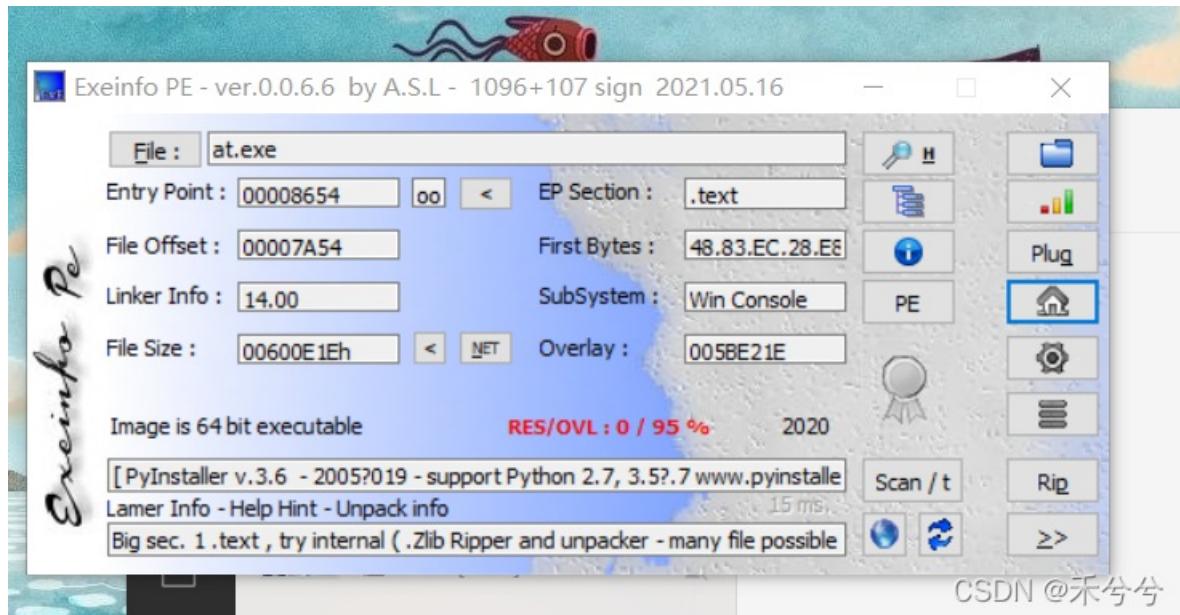
文章标签: 其他 经验分享

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/HLi1219/article/details/122224943>

版权

ExeinfoPE查壳, 之前我用0.3的版本查不出来, 因为没有壳, 打开发现什么都找不到, 更新了ExeinfoPE才找到了pyinstaller的壳。



至于pyinstaller, 翻了很多文章, 找到了它的打包原理, 原来是利用Cython把Python转换成C语言, 然后编译打包。[\(43条消息\) Pyinstaller原理详解_XHG78999的代码之家-CSDN博客_pyinstaller讲解](#)

利用pyinstxtractor.py 进行反编译, 修复生成文件下的login文件的头, 我用到了010editor, 修复方法参考了这篇文章struct里面提供了各个pyc文件修复头

[\[分享\]PyInstaller 解包-软件逆向-看雪论坛-安全社区|安全招聘|bbs.pediy.com](#)

Name	Size	Type	Date Modified
at.exe_extracted	4.0 KiB	folder	Today
at.exe	6.0 MiB	DOS/Windows executable	Yesterday
pyinstxtractor.py	12.4 KiB	Python script	Today

用010editor打开struct和login文件, 将struct里面E3之前的字节复制到login里面

文件(F) 编辑(E) 搜索(S) 视图(V) 格式(O) 脚本(I) 模板(L) 调试(D) 工具(T) 窗口(W)

起始页 article_group.php struct x

编辑方式: 十六进制(H) 运行脚本 运行模板

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0000h:	33	0D	0D	0A	70	79	69	30	01	01	00	00	E3	00	00	00
0010h:	00	00	00	00	00	00	00	00	00	08	00	00	00	40	00	00
0020h:	00	73	38	00	00	00	64	00	64	01	64	02	64	03	64	04
0030h:	64	05	64	06	64	07	67	08	5A	00	64	08	64	09	6C	01
0040h:	54	00	64	08	64	0A	6C	01	6D	02	5A	02	01	00	64	08
0050h:	64	0B	6C	01	6D	03	5A	03	01	00	64	0C	53	00	29	0D
0060h:	DA	08	63	61	6C	63	73	69	7A	65	DA	04	70	61	63	6B
0070h:	DA	09	70	61	63	6B	5F	69	6E	74	6F	DA	06	75	6E	70
0080h:	61	63	6B	DA	0B	75	6E	70	61	63	6B	5F	66	72	6F	6D
0090h:	DA	0B	69	74	65	72	5F	75	6E	70	61	63	6B	DA	06	53
00A0h:	74	72	75	63	74	DA	05	65	72	72	6F	72	E9	00	00	00
00B0h:	00	29	01	DA	01	2A	29	01	DA	0B	5F	63	6C	65	61	72
00C0h:	63	61	63	68	65	29	01	DA	07	5F	5F	64	6F	63	5F	5F
00D0h:	4E	29	04	DA	07	5F	5F	61	6C	6C	5F	5F	DA	07	5F	73
00E0h:	74	72	75	63	74	72	0B	00	00	00	72	0C	00	00	00	A9
00F0h:	00	72	0F	00	00	00	72	0F	00	00	00	FA	1A	64	3A	5C
0100h:	61	6E	61	63	6F	6E	64	61	33	5C	6C	69	62	5C	73	74
0110h:	72	75	63	74	2E	70	79	DA	08	3C	6D	6F	64	75	6C	65
0120h:	3E	03	00	00	00	73	0C	00	00	00	0A	01	02	03	02	03
0130h:	06	03	08	01	0C	01										

CSDN @禾兮兮

文件(F) 编辑(E) 搜索(S) 视图(V) 格式(O) 脚本(I) 模板(L) 调试(D) 工具(T) 窗口(W) 帮助(H)

起始页 article_group.php struct login.pyc x

编辑方式: 十六进制(H) 运行脚本 运行模板

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	0123456789AI
000h:	33	0D	0D	0A	70	79	69	30	01	01	00	00	E3	00	00	00	3...pyi0...
010h:	00	00	00	00	00	00	00	00	00	07	00	00	00	40	00	00
020h:	00	73	5E	07	00	00	64	00	64	01	6C	00	5A	00	65	01	.s^...d.d.l
030h:	64	02	83	01	5A	02	65	03	65	02	83	01	64	03	6B	03	d.f.Z.e.e.f
040h:	72	2C	65	04	64	04	83	01	01	00	65	00	6A	05	83	00	r,e.d.f...e
050h:	01	00	67	00	5A	06	78	32	65	07	64	05	83	01	44	00	..g.Z.x2e.d
060h:	5D	26	5A	08	65	06	6A	09	65	0A	65	02	65	08	19	00]&Z.e.j.e.e
070h:	83	01	65	0A	65	02	65	08	64	06	17	00	19	00	83	01	f.e.e.e.d..
080h:	41	00	83	01	01	00	71	3A	57	00	65	06	6A	09	65	0A	A.f...q:W.e
090h:	65	02	64	05	19	00	83	01	83	01	01	00	65	06	64	07	...d...f.f...
0A0h:	19	00	5A	0B	65	06	64	06	19	00	5A	0C	65	06	64	00	..Z.e.d...z

然后将login加上pyc后缀，利用uncompyle6将pyc文件反编译成py文件得的源代码：

```
# uncompyle6 version 3.8.0
# Python bytecode 3.6 (3379)
# Decompiled from: Python 3.8.10 (tags/v3.8.10:3d8993a, May  3 2021, 11:48:03) [MSC v.1928 64 bit (AMD64)]
# Embedded file name: login.py
# Compiled at: 1995-09-28 00:18:56
# Size of source mod 2**32: 257 bytes
import sys
input1 = input('input something:')
if len(input1) != 14:
    print('Wrong length!')
    sys.exit()
else:
    code = []
    for i in range(13):
        code.append(ord(input1[i]) ^ ord(input1[(i + 1)]))

    code.append(ord(input1[13]))
    a1 = code[2]
    a2 = code[1]
    a3 = code[0]
    a4 = code[3]
    a5 = code[4]
    a6 = code[5]
    a7 = code[6]
    a8 = code[7]
    a9 = code[9]
    a10 = code[8]
    a11 = code[10]
    a12 = code[11]
    a13 = code[12]
    a14 = code[13]
    if (a1 * 88 + a2 * 67 + a3 * 65 - a4 * 5 + a5 * 43 + a6 * 89 + a7 * 25 + a8 * 13 - a9 * 36 + a10 * 15 +
        print('flag is GWT{md5(your_input)}')
        print('Congratulations and have fun!')
    else:
        print('Sorry,plz try again...')

# okay decompiling login.pyc
```

```

import hashlib
from z3 import *

a1=Int("a1")
a2=Int("a2")
a3=Int("a3")
a4=Int("a4")
a5=Int("a5")
a6=Int("a6")
a7=Int("a7")
a8=Int("a8")
a9=Int("a9")
a10=Int("a10")
a11=Int("a11")
a12=Int("a12")
a13=Int("a13")
a14=Int("a14")

a = Solver()

a.add(a1 * 88 + a2 * 67 + a3 * 65 - a4 * 5 + a5 * 43 + a6 * 89 + a7 * 25 + a8 * 13 - a9 * 36 + a10 * 15 + a
a.add(a1 * 89 + a2 * 7 + a3 * 12 - a4 * 25 + a5 * 41 + a6 * 23 + a7 * 20 - a8 * 66 + a9 * 31 + a10 * 8 + a1
a.add(a1 * 28 + a2 * 35 + a3 * 16 - a4 * 65 + a5 * 53 + a6 * 39 + a7 * 27 + a8 * 15 - a9 * 33 + a10 * 13 +
a.add(a1 * 23 + a2 * 34 + a3 * 35 - a4 * 59 + a5 * 49 + a6 * 81 + a7 * 25 + (a8 * 128) - a9 * 32 + a10 * 75
a.add(a1 * 38 + a2 * 97 + a3 * 35 - a4 * 52 + a5 * 42 + a6 * 79 + a7 * 90 + a8 * 23 - a9 * 36 + a10 * 57 +
a.add(a1 * 22 + a2 * 27 + a3 * 35 - a4 * 45 + a5 * 47 + a6 * 49 + a7 * 29 + a8 * 18 - a9 * 26 + a10 * 35 +
a.add(a1 * 12 + a2 * 45 + a3 * 35 - a4 * 9 - a5 * 42 + a6 * 86 + a7 * 23 + a8 * 85 - a9 * 47 + a10 * 34 + a
a.add(a1 * 79 + a2 * 62 + a3 * 35 - a4 * 85 + a5 * 33 + a6 * 79 + a7 * 86 + a8 * 14 - a9 * 30 + a10 * 25 +
a.add(a1 * 8 + a2 * 6 + a3 * 64 - a4 * 85 + a5 * 73 + a6 * 29 + a7 * 2 + a8 * 23 - a9 * 36 + a10 * 5 + a11
a.add(a1 * 67 - a2 * 68 + a3 * 68 - a4 * 51 - a5 * 43 + a6 * 81 + a7 * 22 - a8 * 12 - a9 * 38 + a10 * 75 +
a.add(a1 * 85 + a2 * 63 + a3 * 5 - a4 * 51 + a5 * 44 + a6 * 36 + a7 * 28 + a8 * 15 - a9 * 6 + a10 * 45 + a1
a.add(a1 * 47 + a2 * 64 + a3 * 66 - a4 * 5 + a5 * 43 + a6 * 112 + a7 * 25 + a8 * 13 - a9 * 35 + a10 * 95 +
a.add(a1 * 89 + a2 * 67 + a3 * 85 - a4 * 25 + a5 * 49 + a6 * 89 + a7 * 23 + a8 * 56 - a9 * 92 + a10 * 14 +
a.add(a1 * 95 + a2 * 34 + a3 * 62 - a4 * 9 - a5 * 43 + a6 * 83 + a7 * 25 + a8 * 12 - a9 * 36 + a10 * 16 + a

print(a.check())
print(a.model())
#注意login.py中a1 = code[2], a3 = code[0], a1和a3位置互换, a10和a9
a = [10, 24, 119, 7, 104, 43, 28, 91, 108, 52, 88, 74, 88, 33]
flag = ""

for i in range(12, -1, -1):
    a[i] = a[i] ^ a[i + 1]

print(a)
for i in a:
    flag += chr(i)
print(flag)
print(hashlib.md5(flag.encode(encoding='UTF-8')).hexdigest())

```

解密脚本如上