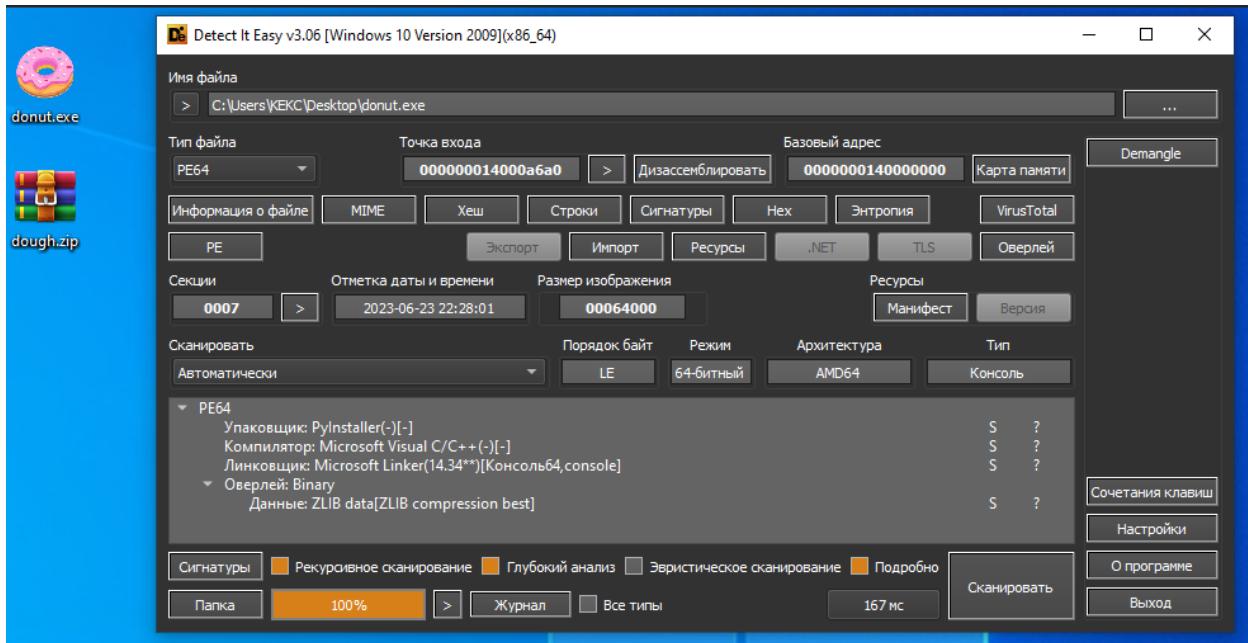




Название:	Пончик
Категория:	Реверс-инжиниринг
Уровень:	Легко
Очки:	300
Описание:	Я вкус пончиков обожаю, Но безопасность мне нужна. EXE, как тайный лабиринт воспринимаю, И точно знаю, что могу ревёрсить я без сна.
Теги:	Python, PyInstaller
Автор:	ROP

Прохождение:

Изучаем файл donut.exe в DIE. Видим, что это pyinstaller.



Используем pyinstxtractor (<https://github.com/extremecoders-re/pyinstxtractor>) uncompyle6 (установка через pip) и получаем исходник.

```
Administrator: C:\Windows\system32\cmd.exe
C:\Users\KEKC\Desktop>uncompyle6 donut.exe

# file donut.exe
# path donut.exe must point to a Python source that can be compiled, or Python bytecode (.pyc, .pyo)

C:\Users\KEKC\Desktop>pyinstxtractor.py
[+] Usage: pyinstxtractor.py <filename>

C:\Users\KEKC\Desktop>pyinstxtractor.py donut.exe
[+] Processing donut.exe
[+] Pyinstaller version: 2.1+
[+] Python version: 3.8
[+] Length of package: 7169924 bytes
[+] Found 102 files in CArchive
[+] Beginning extraction...please standby
[+] Possible entry point: pyboot01_bootstrap.pyc
[+] Possible entry point: donut.pyc
[!] Warning: This script is running in a different Python version than the one used to build the executable.
[!] Please run this script in Python 3.8 to prevent extraction errors during unmarshalling
[!] Skipping pyz extraction
[+] Successfully extracted pyinstaller archive: donut.exe

You can now use a python decompiler on the pyc files within the extracted directory
C:\Users\KEKC\Desktop>cd donut.exe_extracted
C:\Users\KEKC\Desktop\donut.exe_extracted>uncompyle6 donut.pyc
```

Получили декомпилированный код.

The screenshot shows a Notepad++ window with the title 'C:\User\XEC\Desktop\text.py - Notepad++ [Administrator]'. The code in the editor is as follows:

```
1 # uncompyle6 version 3.9.0
2 # Python bytecode version base 3.0.0 (3433)
3 # The source code is Python 3.8.0 (tags/v3.8.0:f9919fd, Oct 14 2019, 19:37:50) [MSC v.1916 64 bit (AMD64)]
4 # Embedded file name: donut.py
5 import subprocess, signal, os, sys, platform
6 ZIPFILE_NAME = 'dough.zip'
7
8 def signal_handler(sig, frame):
9     os.remove('.donut')
10    sys.exit(0)
11
12 plat = platform.system()
13 if plat == "Windows":
14     print("Напоминай, не такой день...")
15     sys.exit(0)
16 imp = input('Весит пароль == ?')
17 if imp != 'MayBe_Th1$_1S_Wr0NG':
18     print("Напоминай, не такой день...")
19     sys.exit(0)
20 current_directory = os.getcwd()
21 if not os.path.exists(os.path.join(current_directory, ZIPFILE_NAME)):
22     print("Напоминай, не такой день...")
23     sys.exit(0)
24 import pyzipper
25
26 def decrypt(file_path, word):
27     with pyzipper.AESZipfile(file_path, 'r', compression=pyzipper.ZIP_LZMA, encryption=pyzipper.WZ_AES) as (extracted_zip):
28         try:
29             extracted_zip.extractall(pwd=word)
30         except RuntimeError as ex:
31             try:
32                 print(ex)
33             finally:
34                 ex = None
35             del ex
36
37
38 decrypt(ZIPFILE_NAME, 'MayBe_Th1$_1S_Wr0NG'.encode())
39 with open('dough', 'rb') as (file):
40     content = file.read()
41     data = bytearray(content)
42     data = [x for x in data]
43
```

Изучаем его и понимаем, какой пароль у архива, что идёт вместе с EXE-файлом. Так же понимаем, что файл в архиве запускается в Linux.

Пароль к архиву: **M@ybe_tH1\$_1S_Wr0NG**

Так же этот скрипт декодирует файл. Мы можем его вручную восстановить и убрать ошибки uncompyle6:

```
import subprocess
import signal
import os
import sys
import platform

ZIPFILE_NAME = 'dough.zip'

def signal_handler(sig, frame):
    os.remove('.donut')
    sys.exit(0)

plat = platform.system()

if plat == "Windows":
```

```
print("Наверное, не твой день...")
sys.exit(0)

inp = input("Введи пароль => ")
if inp != "{MayBe_Th1$_1S_wR0nG}":
    print("Наверное, не твой день...")
    sys.exit(0)

current_directory = os.getcwd()

if not os.path.exists(os.path.join(current_directory, ZIPFILE_NAME)):
    print("Наверное, не твой день...")
    sys.exit(0)

import pyzipper

def decrypt(file_path, word):
    with pyzipper.AESZipFile(file_path, 'r', compression=pyzipper.ZIP_LZMA, encryption=pyzipper.WZ_AES) \\
        as extracted_zip:
        try:
            extracted_zip.extractall(pwd=word)
        except RuntimeError as ex:
            print(ex)

decrypt(ZIPFILE_NAME, "M@ybe_th1$_1S_Wr0NG".encode())

# Открываем файл в бинарном режиме
with open('dough', 'rb') as file:
    content = file.read()
data = bytearray(content)
data = [x for x in data]
```

```

# Добавляем к каждому элементу массива 0x50
for i in range(len(data)):
    data[i] = (data[i] + 0x50)%256

# Выполняем XOR с 0x50
for i in range(len(data)):
    data[i] = (data[i] ^ 0x50)%256

# Записываем результат в файл .donut
with open('.donut', 'wb') as f:
    for b in data:
        f.write(bytes([b]))

subprocess.run(['chmod', '+x', '.donut'])
os.remove('dough')

# Запускаем файл .donut
signal.signal(signal.SIGINT, signal_handler)
subprocess.run(['./.donut'])

```

Уберём строки ниже и декодер готов.

```

signal.signal(signal.SIGINT, signal_handler)
subprocess.run(['./.donut'])

```

Пароль для декодирования: {MayBe_Th1\$_1S_wR0nG}

```

(venv) rop@rop-pc:/tmp/donut$ python donut.py
Введи пароль => {MayBe_Th1$_1S_wR0nG}
(venv) rop@rop-pc:/tmp/donut$ 

```

Попробуем запустить файл `.donut`.

\$@C@C@C@C@C@C@
\$\$\$\$\$#####\$\$\$\$\$
\$\$\$\$#*****!*!*#\$\$\$#
#\$\$##**!!!!!=!!*#\$\$#
#####*!!=;::~~::;;==!!*###\$##
####!=;:~-,...,-~:;=!!*###*
####!=;:-,...,...-:;=!!*###*
!*####*!=;~,.. .~:=!!*###*
**####*!=;~. .~;!!*###*
:!*####*!=~ ~=!!*###*
:!*####*!=~ ;!!*###*
=*###\$####*!= !*###\$####*!=;
;!!*###\$####*!= ;!!*###*
,=!!*###\$####*!=@C@C@C@C@C@C@C@
-;=!!*###\$####*!=:
-:=!!*###\$####*!=;~
~:;!**!!!!*****!====:~,
~:;==!!!!!!!=;:~.
.~:;=====;:~-.
.

ВИДИМ ПОНЧИК. Если заглянем к нему в строки, то увидим исходник и части флага.

```
(venv) rop@rop-pc:/tmp/donut$ strings .donut
/lib64/ld-linux-x86-64.so.2
_ITM_deregisterTMCloneTable
__gmon_start__
_ITM_registerTMCloneTable
putchar
__libc_start_main
__cxa_finalize
memset
printf
libm.so.6
libc.so.6
GLIBC_2.2.5
GLIBC_2.34
PTE1
u+UH
ffffH
..,-~:;=!*#$@9*3$"int k;double sin()
,cos();main(){float A=
0,B=0,i,j,z[1760];char b[
1760];printf("\x1b[2J");for(;
){memset(b,32,1760);memset(z,0,7040)
;for(j=0;6.28>j;j+=0.07)for(i=0;6.28
>i;i+=0.02){float c=sin(i),d=cos(j),e=
sin(A),f=sin(j),g=cos(A),h=d+2,D=1/(c*
h*e+f*g+5),l=cos(i),m=cos(B),n=s\
in(B),t=c*h*g-f*
e;int x=40+30*D*
(l*h*m-t*n),y=
12+15*D*(l*h*n
+t*m),o=x+80*y,
N=8*((f*e-c*d*g
)*m-c*d*e-f*g-l
*d*n);if(22>y&&
y>0&&x>0&&80>x&&D>z[o])z[o]=D;;;b[o]=
"..,-~:;=!*#$@[N>0?N:0];}/*#*****!!--*
printf("\x1b[H");for(k=0;1761>k;k++)
putchar(k%80?b[k]:10);A+=0.04;B+=
0.02;}/******#####*****!!=;:~-
~::==!!!CODEBY{****!!==:-.
.,~~=I;_:=l0vE;_;:~-.
.,--d0nUt--}-,*/
```

Собираем части флага.