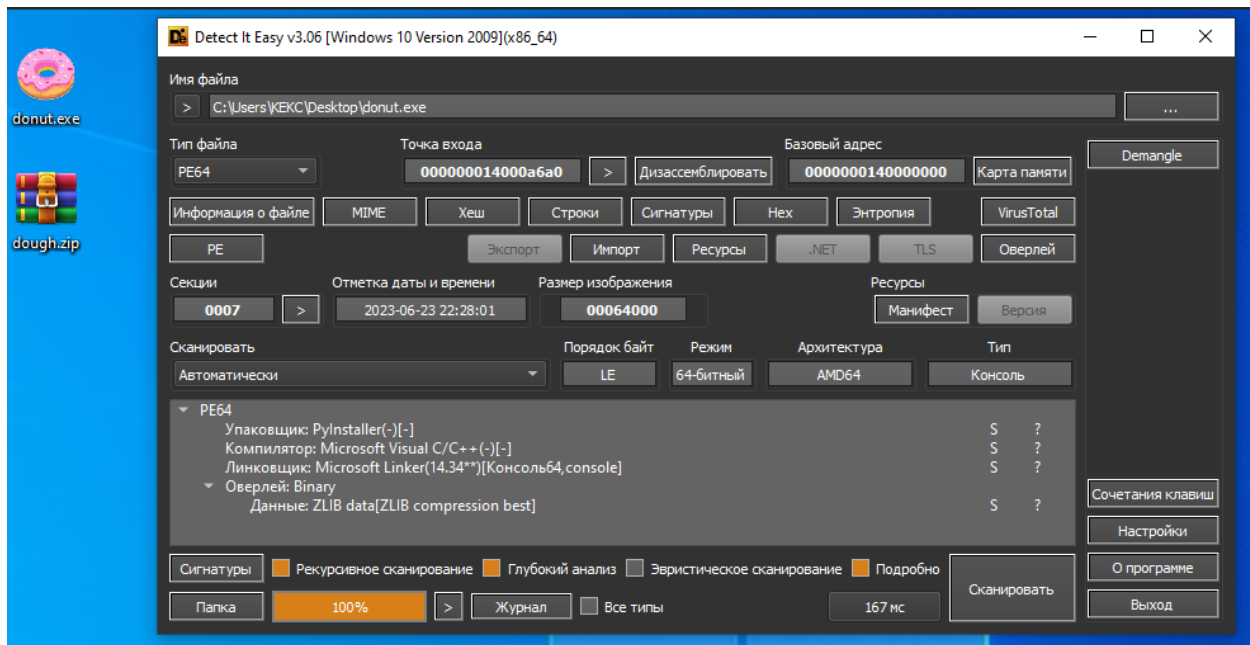




Название:	Пончик
Категория:	Реверс-инжиниринг
Уровень:	Легко
Очки:	300
Описание:	Я вкус пончиков обожаю, Но безопасность мне нужна. EXE, как тайный лабиринт воспринимаю, И точно знаю, что могу реверсить я без сна.
Теги:	Python, PyInstaller
Автор:	ROP

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.

```

Администратор: 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: pyiboot01_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.

```
C:\Users\KERO\Desktop>notepad++ [Administrator]
File Edit View Insert Format Shell Tools Plugins Window Help
1 # uncompress version 3.9.0
2 # Python bytecode version base 3.8.0 (3413)
3 # Decompiled from: Python 3.8.0 (tags/v3.8.0:fa519fd, 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('Настроено, не твой день...')
15     sys.exit(0)
16 inp = input('Введи пароль > ')
17 if inp != 'Maybe_th1$_1$_wR0ng':
18     print('Настроено, не твой 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('Настроено, не твой 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             print(ex)
32             finally:
33                 ex = None
34                 del ex
35
36 decrypt(ZIPFILE_NAME, 'Maybe_th1$_1$_wR0ng'.encode())
37
38 with open('dough', 'rb') as (file):
39     content = file.read()
40     data = bytearray(content)
41     data = [x for x in data]
```

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$_1$_wR0ng`

Так же этот скрипт декодирует файл. Мы можем его вручную восстановить и убрать ошибки uncompryle6:

```
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'])

```

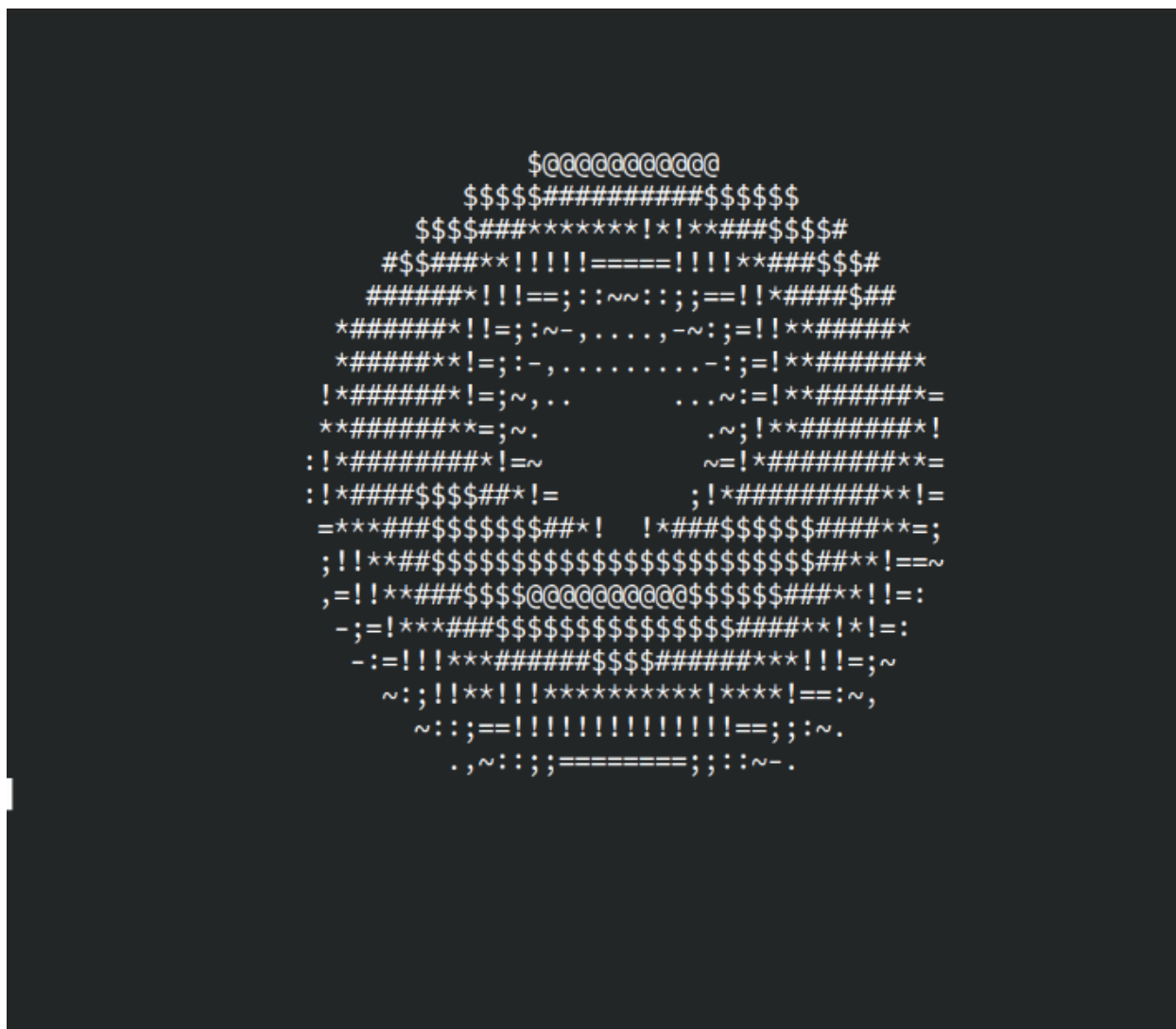
Пароль для декодирования: {MayBe_Th1\$_1S_wR0nG}

```

(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 .



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
gffffH
.,-~:;=!*$#@
9*3$
    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;_:=l@vE;;_:~-.
        ..--d0nUt--}-,*/

```

Собираем части флага.