

Обычная капиbara

ⓘ Краткая информация

Название: Обычная капиbara

Описание: Что такое, это обычная капиbara.

Категория: Реверс-инжиниринг

Сложность: Средняя

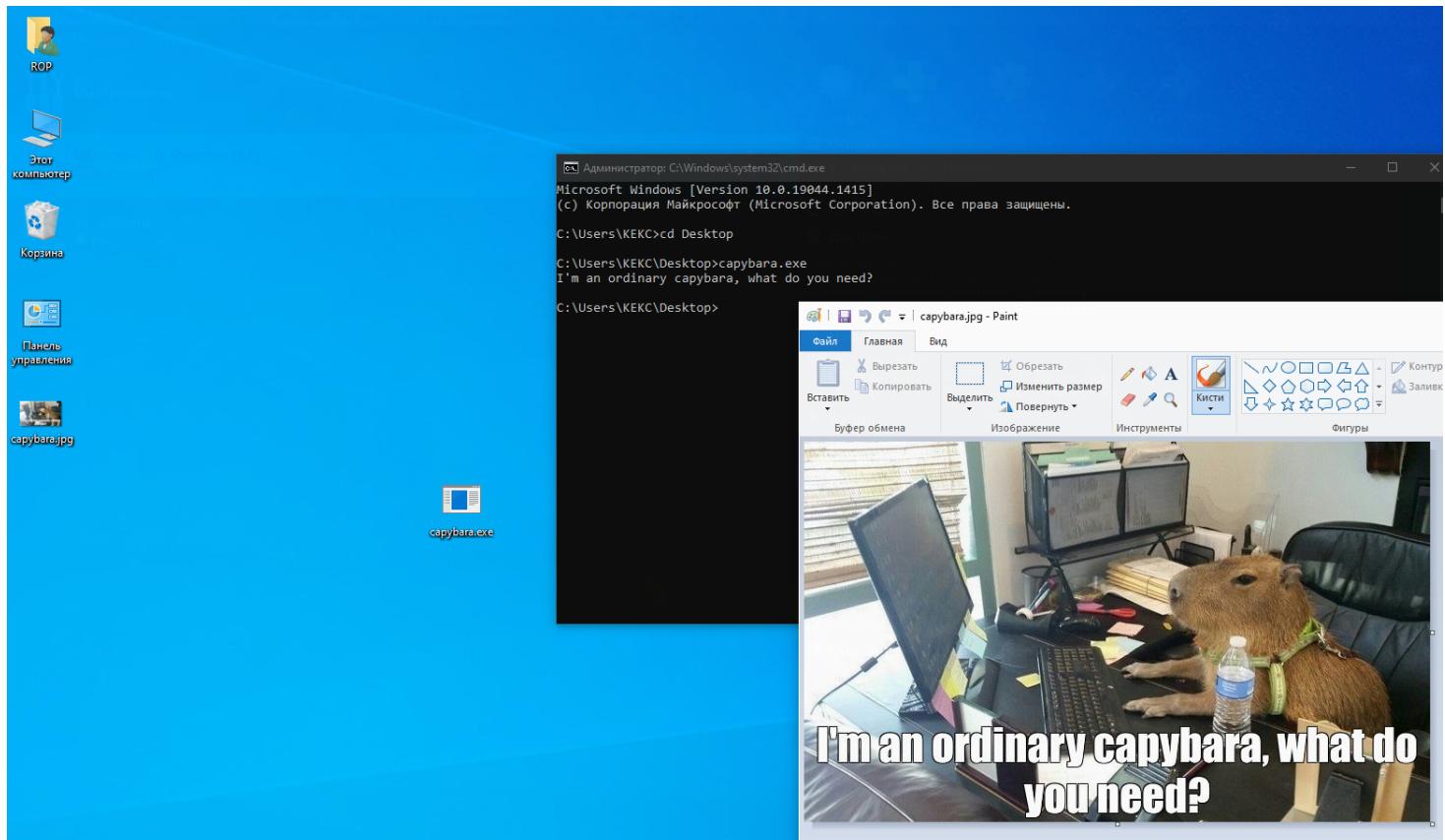
Очки: 450

Мероприятие/ивент: Киберколизей

Флаг: CODEBY{qqq_wE_l2ve_C@pYbaRa_1n_EXE}

Райтап

Распакуем архив и попробуем запустить EXE.



Создаётся картинка на рабочем столе.

Через UPX распакуем файл.

```

Administrator: C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.19044.1415]
(c) Корпорация Майкрософт (Microsoft Corporation). Все права защищены.

C:\Users\KEKC\Desktop\upx-4.0.2-win64>upx.exe -d capybara.exe
    Ultimate Packer for eXecutables
        Copyright (C) 1996 - 2023
UPX 4.0.2      Markus Oberhumer, Laszlo Molnar & John Reiser   Jan 30th 2023

File size      Ratio      Format      Name
-----      -----      -----      -----
424448 <-  307712  72.50%  win64/pe  capybara.exe

Unpacked 1 file.

C:\Users\KEKC\Desktop\upx-4.0.2-win64>

```

Это всё ложный вектор. Во время отладки доходим до сюда.

IDA View-RIP

```
.text:00000000004013E5 call _cexit
.text:00000000004013EA mov eax, cs:dword_46B00C
```

```
.text:00000000004013F0
.text:00000000004013F0 loc_4013F0:
.text:00000000004013F0 add rsp, 98h
.text:00000000004013F7 mov rax, 401900h
.text:00000000004013FE push rax
.text:00000000004013FF retn
```

100.00% (-354,6857) (712,254) 000007FF|00000000004013FF: sub_401180+27F (Synchronized with RIP)

IDA - copybara.exe C:\Users\KEK\Desktop\copybara.exe

File Edit Jump Search View Debugger Options Windows Help

Library function Regular function Instruction Data Unexplored External symbol Lumina function

IDA View-RIP, General registers, Modules, Threads, Hex View-1, Stack view Structures Enums

General registers

RAX	0000000000004D000
RBX	000000000000008
RCX	000000000404020
RDX	00000000045101F
RSP	00000000000000000000000000000000
RSI	00000000000000000000000000000000
RDY	00000000000000000000000000000000
VIP	00000000000000000000000000000000
VIF	00000000000000000000000000000000
AC	00000000000000000000000000000000
VM	00000000000000000000000000000000

Modules

Path	Base	Size
C:\Users\KEK\Desktop\copybara.exe	0000000000000000	0000000000000000
C:\Windows\System32\ERNIEBASE.dll	000007FA80000000	0000000000000000
C:\Windows\System32\svrsv.dll	000007FA80000000	0000000000000000
C:\Windows\kernels\7kFRNFI.7.ni	000007FAFCAP0000	0000000000000000

Threads

Decimal	Hex	State	Name
4964	A364	Ready	copybara.exe
2704	A90	Ready	7FFABD102AD0

Hex View-1

Stack view

0000000000401380 00 48 89 10 4C 8B 05 55 9C 06 00 48 8B 15 56 9C H, L, U, D, N, V 00000000000000000000000000000000

Далее выполняем код до сюда.

Переходим в **var_10**, затем по адресу из него.

IDA View-RIP

Debug View

Structures

Stack[00001364]:000000000066FEAC db 0
Stack[00001364]:000000000066FEAD db 0D0h
Stack[00001364]:000000000066FEAE db 4
Stack[00001364]:000000000066FEAF db 0
Stack[00001364]:000000000066FEB0 dd 404020h
Stack[00001364]:000000000066FEB4 db 0
Stack[00001364]:000000000066FEB5 db 0
Stack[00001364]:000000000066FEB6 db 0
Stack[00001364]:000000000066FEB7 db 0
Stack[00001364]:000000000066FEB8 db 0
Stack[00001364]:000000000066FEB9 db 0
Stack[00001364]:000000000066FEBA db 0BBh
Stack[00001364]:000000000066FEBB db 41h ; A
Stack[00001364]:000000000066FEBC db 0
Stack[00001364]:000000000066FEBD db 0D0h
Stack[00001364]:000000000066FEBE db 4
Stack[00001364]:000000000066FEBF db 0
RBP Stack[00001364]:000000000066FEC0 db 0E0h

UNINOWN_000000000066FEB0: Stack[00001364]:000000000066FEB0 (Synchronized with RIP)

IDA View-RIP window showing assembly code for a Windows 10 Reverse API dump. The assembly code includes instructions like assume cs:_data, dword_404000 dd 0Ah, align 20h, and various db and mov instructions. The right side of the interface displays system information for Windows 10 Pro, including CPU, RAM, and disk details.

```
.data:0000000000404000 assume cs:_data
.data:0000000000404000 ;org 404000h
.data:0000000000404000 dword_404000 dd 0Ah
.data:0000000000404004 align 20h
RCX .data:0000000000404020 db 4Dh ; M
.RCX .data:0000000000404021 db 5Ah ; Z
.RCX .data:0000000000404022 db 90h
.RCX .data:0000000000404023 db 0
.RCX .data:0000000000404024 db 3
.RCX .data:0000000000404025 db 0
.RCX .data:0000000000404026 db 0
.RCX .data:0000000000404027 db 0
.RCX .data:0000000000404028 db 4
.RCX .data:0000000000404029 db 0
.RCX .data:000000000040402A db 0
.RCX .data:000000000040402B db 0
.RCX .data:000000000040402C db 0FFh
.RCX .data:000000000040402D db 0FFh
```

Ставим курсор на 0x402020 и нажимаем Shift+F2. Используем этот скрипт, чтобы сдампить программу из памяти.

```
auto fname = "C:\\dump.exe";
auto address = 0x404020;
auto size = 0x4D000;
auto file= fopen(fname, "wb");

savefile(file, 0, address, size);
fclose(file);
```

Размер узнали тут.

IDA View-RIP window showing assembly code for capybara.exe. The assembly code includes instructions like push, mov, sub, add, cmp, and ja. A green arrow points from the assembly code to a call graph window, which shows a flow from loc_401955 to loc_40192D, then to sub_401900+1D. The bottom right pane shows the end of the function with exit and endp instructions.

```
.text:0000000000401900 push  rbp
.text:0000000000401901 mov   rbp, rsp
.text:0000000000401904 sub   rsp, 40h
.text:0000000000401908 mov   [rbp+var_5], 41h ; 'A'
.text:000000000040190C mov   [rbp+var_6], 08Bh
.text:0000000000401910 mov   [rbp+var_10], 483FFDh
.text:0000000000401918 add   [rbp+var_10], 23h ; '#'
.text:000000000040191D mov   [rbp+var_14], 4D000h
.text:0000000000401924 mov   [rbp+var_4], 0
.text:000000000040192B jmp   short loc_401955
```

```
.text:0000000000401955 loc_401955:
.text:0000000000401955     mov    eax, [rbp+var_4]
.text:0000000000401958 cmp    [rbp+var_14], eax
.text:000000000040195B ja    short loc_40192D
```

```
.text:000000000040192D loc_40192D:
.text:000000000040192D     mov    eax, [rbp+var_4]
```

```
.text:000000000040195D mov    ecx, 0          ; Code
.text:0000000000401962 call   exit
.text:0000000000401962 sub_401900 endp
```

У нас будет файл `dump.exe` в диске С. Запустим его и получим новую картинку.



Флаг: `CODEBY{qqq_wE_l2ve_C@pYbaRa_1n_EXE}`