

TRAS LAS SOMBRA: EVADIENDO COMO EL ADVERSARIO

CARLOS GARRIDO

WHOAMI

HELLO S-1-1-0

- Red Team Operations Leader en Pentraze Cybersecurity.
- Instructor de Ciberseguridad (Red Team) en USACH.
- Security Research:
 - CVE-2023-2705 (Gravity Forms gAppointments – WordPress)
 - CVE-2023-2707 (Gravity Forms gAppointments – WordPress)
 - DPAPI For Impact (<https://malapi.io/winapi/CryptProtectData>)
 - Process Hypnosis
 - Interactive Cross-Session Token Impersonation Terminal
 - Extracting Active Directory Credentials from an unsealed and unaunthenticated HashiCorp Vault.





Evasión de defensas



Windows 101



Arquitectura de un EDR



Subverting the Windows Kernel



ROP local para control de flujo sin corrupción de memoria



Process hypnosis: Debugger assisted control flow hijack

AGENDA



Evasión de defensas



Sobrevolando la Pirámide del Dolor



Windows 101

Win32 API

- API de Windows, también conocida como interfaz de programación de aplicaciones de Windows, facilita la comunicación entre los programas creados por el usuario y el sistema operativo Windows.

C++

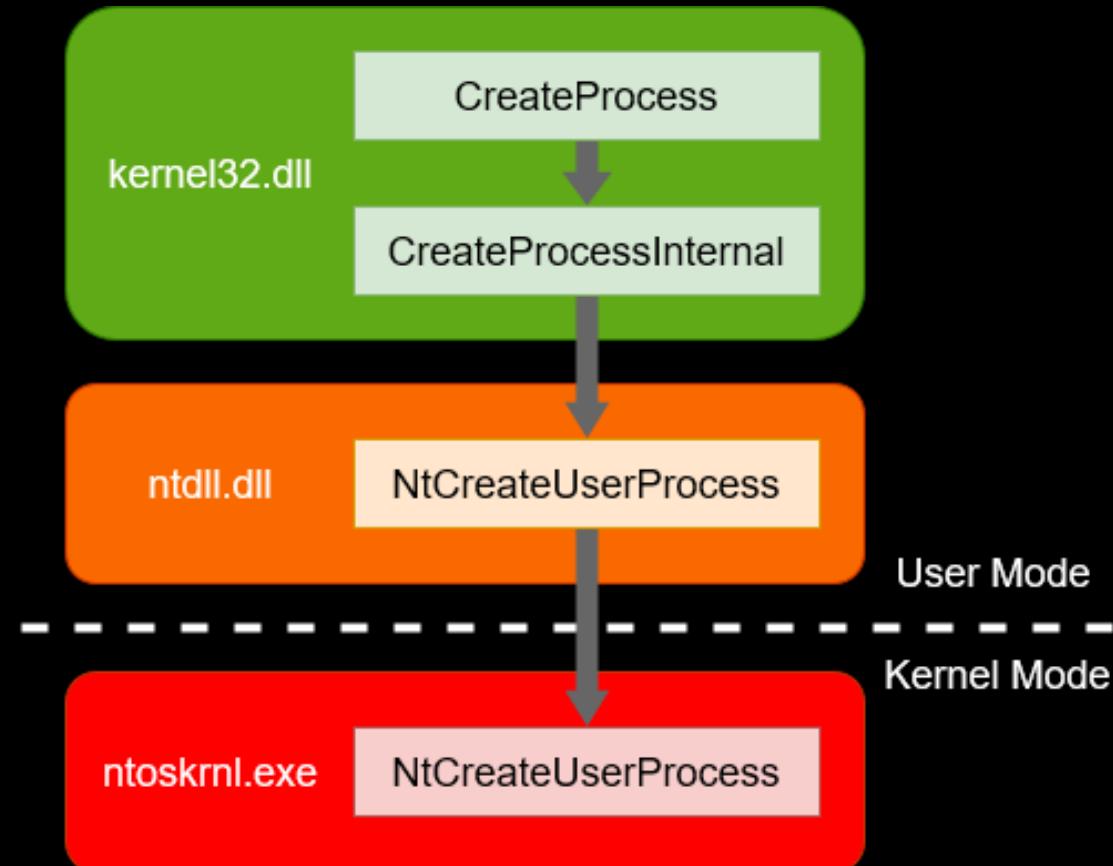
```
HANDLE CreateFileA(  
    [in]          LPCSTR          lpFileName,  
    [in]          DWORD           dwDesiredAccess,  
    [in]          DWORD           dwShareMode,  
    [in, optional] LPSECURITY_ATTRIBUTES lpSecurityAttributes,  
    [in]          DWORD           dwCreationDisposition,  
    [in]          DWORD           dwFlagsAndAttributes,  
    [in, optional] HANDLE          hTemplateFile  
);  
)?  
    [in]          HANDLE          lpTemplateFile  
    [in]          DWORD           dwFlagsAndAttributes  
);
```

C++

```
BOOL MoveFileA(  
    [in] LPCSTR lpExistingFileName,  
    [in] LPCSTR lpNewFileName  
);  
)?
```

Windows 101

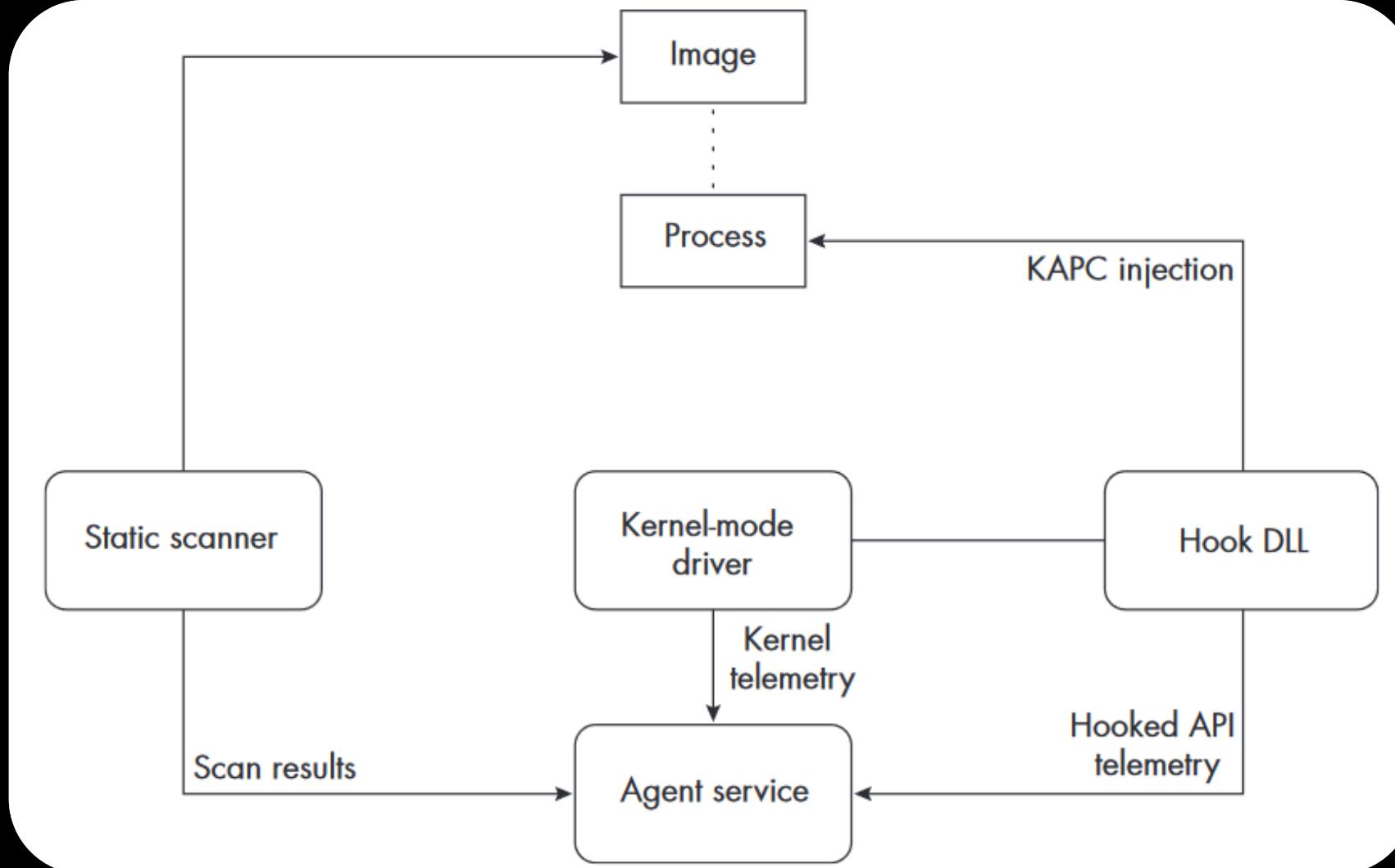
Funciones NT



Windows 101



EDR

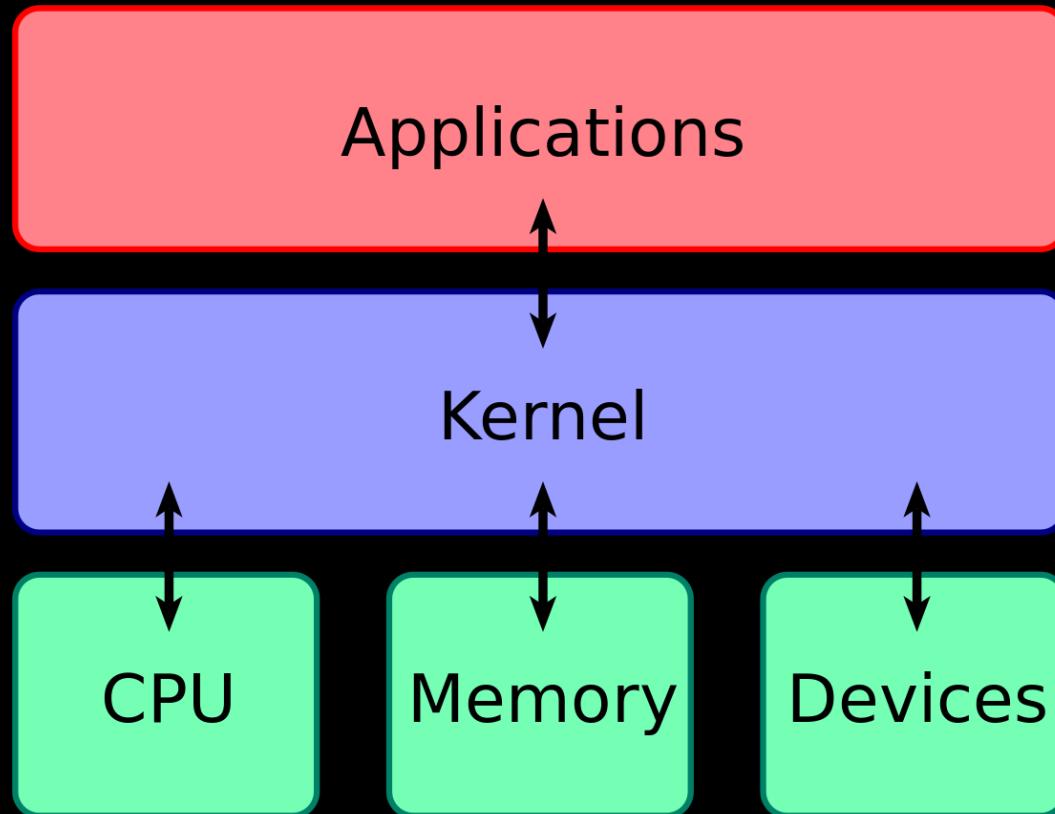


Retrieved From “Evading EDR” by Matt Hand

Arquitectura EDR



Subverting The Windows Kernel



Windows Kernel

Protected Process Light (PPL)

- PPL (Protected Process Light) es un mecanismo de seguridad.
- Permite la ejecución de programas con firmas especiales.
- Estos programas se ejecutan de manera protegida contra manipulaciones y terminaciones.
- Incluso los usuarios administrativos no pueden manipular o terminar estos programas.
- Su propósito es prevenir ataques de malware.
- También protege procesos críticos del sistema.

Protected Process Light (PPL) - LSASS

```
mimikatz # privilege::debug  
Privilege '20' OK
```

```
mimikatz # sekurlsa::ekeys  
ERROR kuhl_m_sekurlsa_acquireLSA ; Handle on memory (0x00000005)
```



ERROR: КПУТ-Ш-СЕКУЛСА-АКДАТУГЭА ? НЭҮҮДЭ ОУ ШИМОЛЫ (0x00000005)

Process	CPU	Private Bytes	Working Set	PID	Description	Protection	Integrity
lsass.exe	< 0.01	7,136 K	8,664 K	688		PsProtectedSignerLsa-Light	System

PPL Internals

MsMpEng.exe:2164 Properties

Image	Performance	Performance Graph	Disk and Network																			
GPU Graph	Services	Threads	TCP/IP	Security	Environment	Strings																
 User: NT AUTHORITY\SYSTEM																						
	SID: S-1-5-18																					
		Session: 0 Logon Session: 3e7																				
		Virtualized: No	Protected: PsProtectedSignerAntimalware-Ligh																			
<table border="1"><thead><tr><th>Group</th><th>Flags</th></tr></thead><tbody><tr><td>BUILTIN\Administrators</td><td>Owner</td></tr><tr><td>BUILTIN\Users</td><td>Mandatory</td></tr><tr><td>CONSOLE LOGON</td><td>Mandatory</td></tr><tr><td>CONSOLOGON</td><td>YoletpnuM</td></tr><tr><td>SYSTEM\INTERNA</td><td>YoletpnuM</td></tr><tr><td>SYSTEM\INTERNA</td><td>OWNER</td></tr><tr><td>SYSTEM\INTERNA</td><td>Logon</td></tr></tbody></table>							Group	Flags	BUILTIN\Administrators	Owner	BUILTIN\Users	Mandatory	CONSOLE LOGON	Mandatory	CONSOLOGON	YoletpnuM	SYSTEM\INTERNA	YoletpnuM	SYSTEM\INTERNA	OWNER	SYSTEM\INTERNA	Logon
Group	Flags																					
BUILTIN\Administrators	Owner																					
BUILTIN\Users	Mandatory																					
CONSOLE LOGON	Mandatory																					
CONSOLOGON	YoletpnuM																					
SYSTEM\INTERNA	YoletpnuM																					
SYSTEM\INTERNA	OWNER																					
SYSTEM\INTERNA	Logon																					

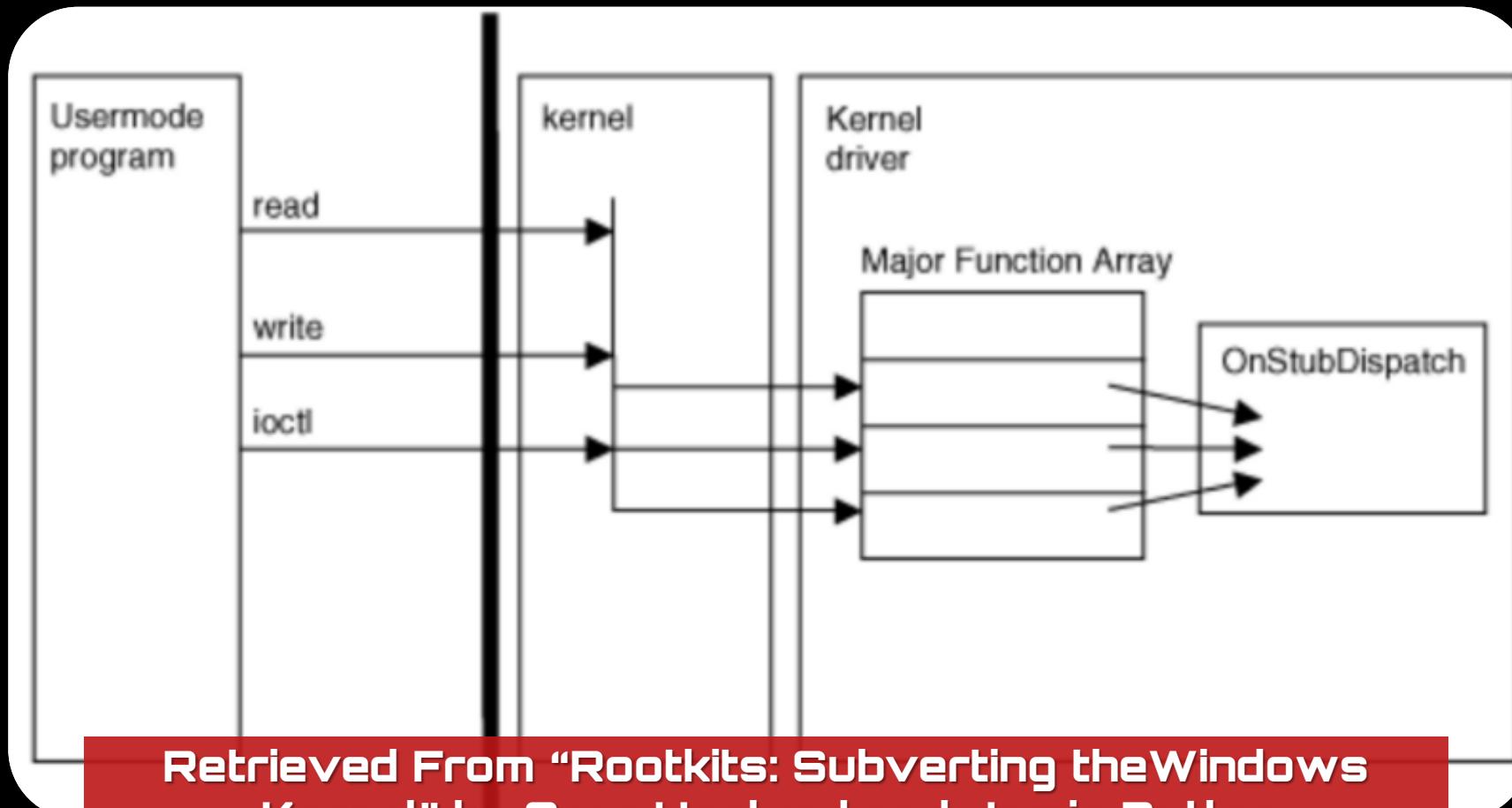
PPL Internals

```
0: kd> !process 0 0 Msmpeng.exe
PROCESS fffffad041e2850c0
    SessionId: 0 Cid: 0874 Peb: 3259603000 ParentCid: 0290
    DirBase: 533d3000 ObjectTable: fffff970c7c9c6180 HandleCount: 818.
    Image: MsMpEng.exe
    Image: MsMpEng.exe
```

```
0: kd> dt nt!_PS_PROTECTION (0xfffffad041e2850c0 + 0x87a)
+0x000 Level : 0x31 '1'
+0x000 Type : 0x001
+0x000 Audit : 0x0
+0x000 Signer : 0x0011
```

```
+0x000 Signer : 0x0011
```

```
0: kd> dt nt!_PS_PROTECTED_SIGNER
PsProtectedSignerNone = 0n0
PsProtectedSignerAuthenticode = 0n1
PsProtectedSignerCodeGen = 0n2
PsProtectedSignerAntimalware = 0n3
PsProtectedSignerLsa = 0n4
PsProtectedSignerWindows = 0n5
PsProtectedSignerWinTcb = 0n6
PsProtectedSignerWinSystem = 0n7
PsProtectedSignerApp = 0n8
PsProtectedSignerMax = 0n9
PsProtectedSignerMax = 0n9
PsProtectedSignerApp = 0n8
```

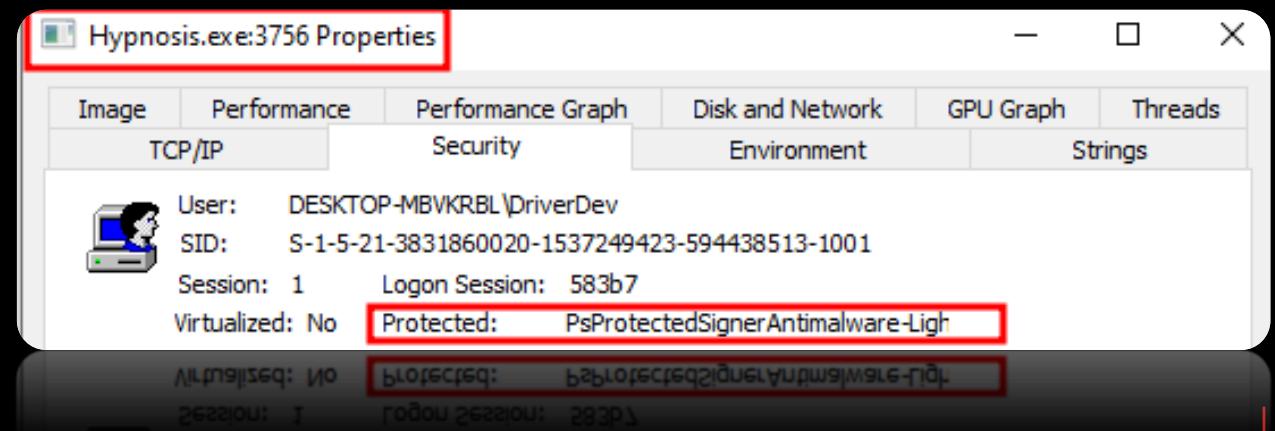


Retrieved From "Rootkits: Subverting the Windows Kernel" by Greg Hoglund and Jamie Butler

Kernel - Drivers

```
SERVICE_NAME: DriverDev
  TYPE          : 1  KERNEL_DRIVER
  STATE         : 4  RUNNING
                  (STOPPABLE, NOT_PAUSABLE
  IGNORES_SHUTDOWN)
  WIN32_EXIT_CODE   : 0  (0x0)
  SERVICE_EXIT_CODE : 0  (0x0)
  CHECKPOINT       : 0x0
  WAIT_HINT        : 0x0
  PID              : 0
  FLAGS            :
  ERROR            :
  BID              :
  INTENT           :
```

```
C:\DriverDev>.\Client_Driver.exe 3756 2
[+] Opening handle to driver
[+] Calling IOCTL_PROTECT_PROCESS...
```



Bring your own Anti-Malware

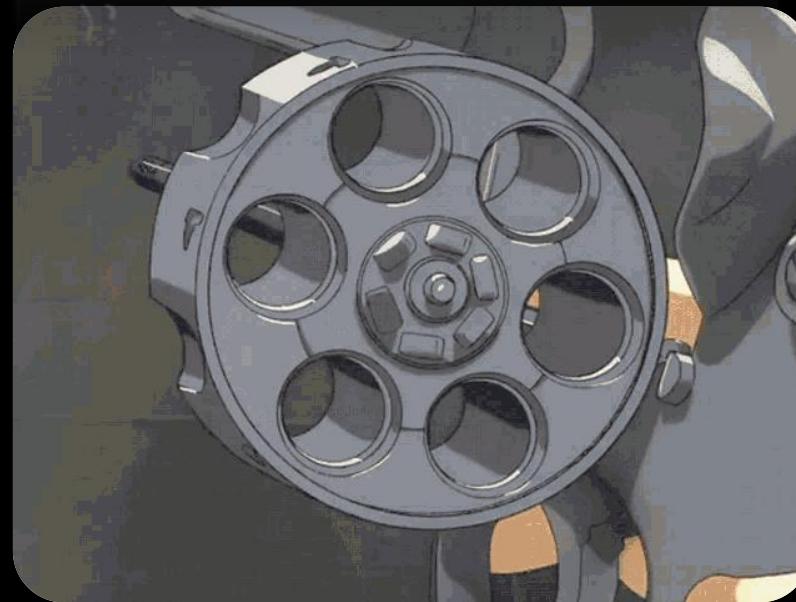


Your PC ran into a problem

Your PC ran into a problem

C++

```
void KeBugCheck(  
    [in] ULONG BugCheckCode  
)
```



Kernel Patch Guard (KPP) & Driver Signature Enforcement



This app has been blocked by your system administrator.

Contact your system administrator for more info.

[Close](#)

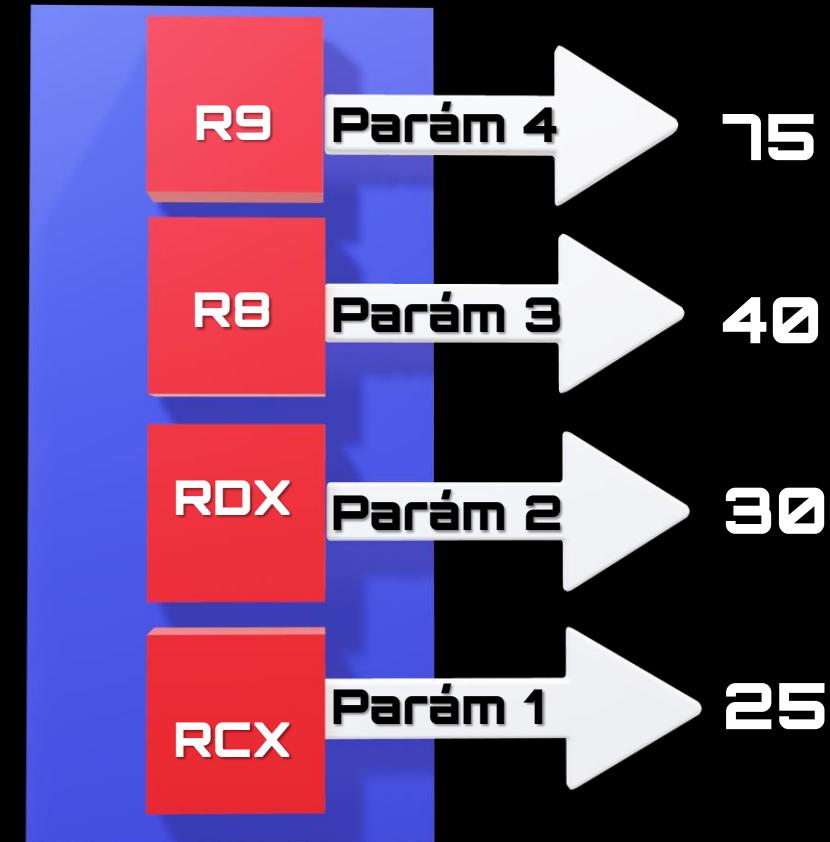
CVE-2024-21338: Windows Kernel Elevation of Privilege

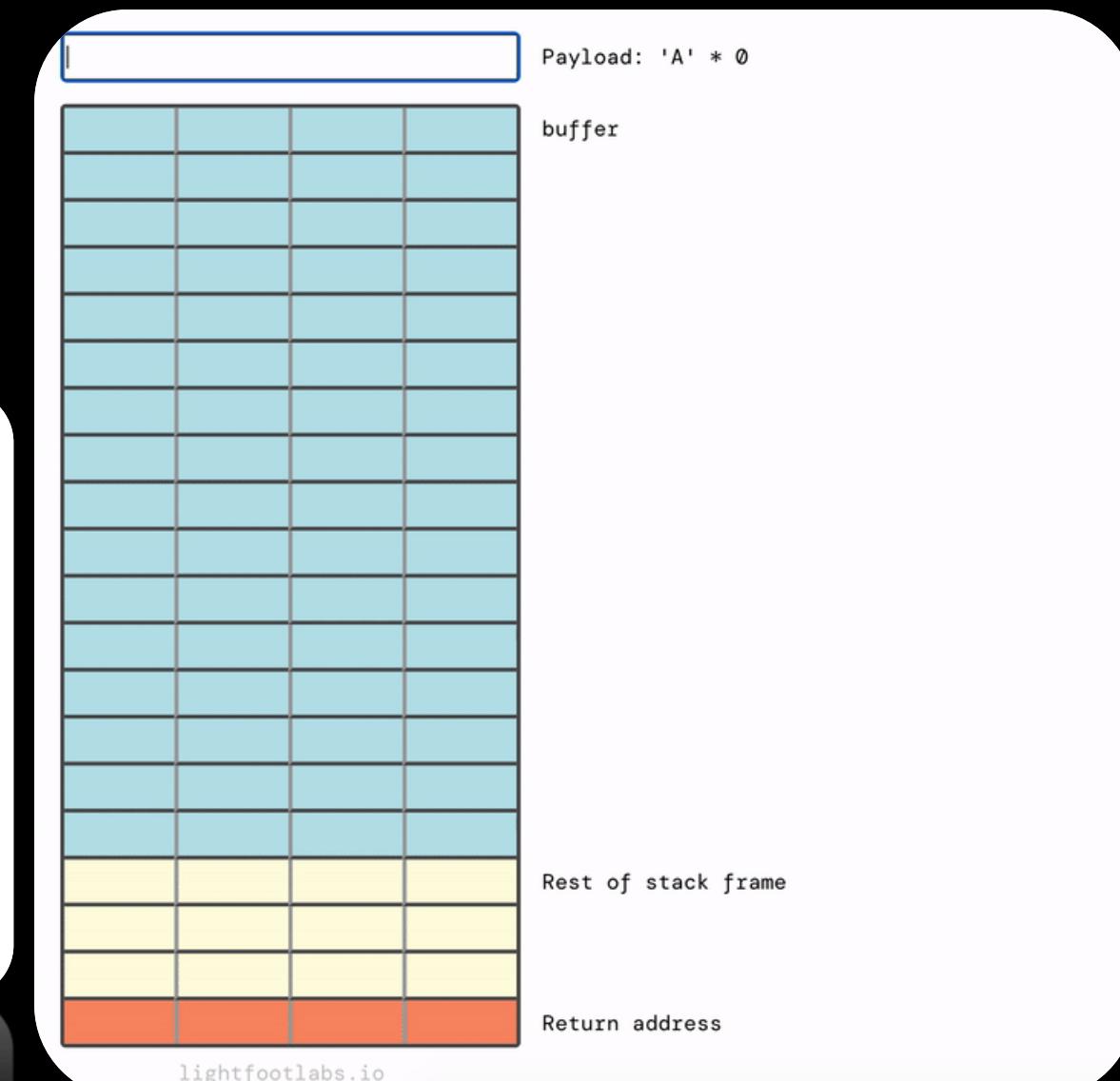
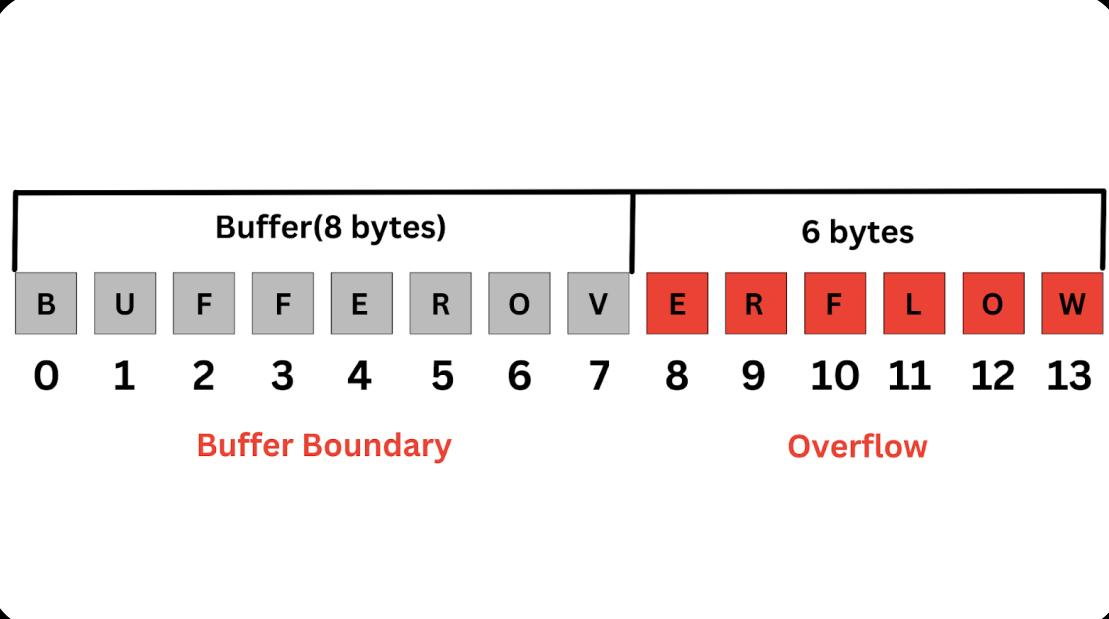


**ROP Local para control de flujo
sin corrupción de memoria**

X64 FastCall Calling Convention

```
1 int suma (int p1, int p2, int p3, int p4){  
2     int results = p1 + p2 + p3 + p4;  
3     return results;  
4 }  
5  
6 suma(25, 30, 40, 75);
```





ROP Local para control de flujo sin corrupción de memoria

memoria

Data Execution Prevention (DEP)

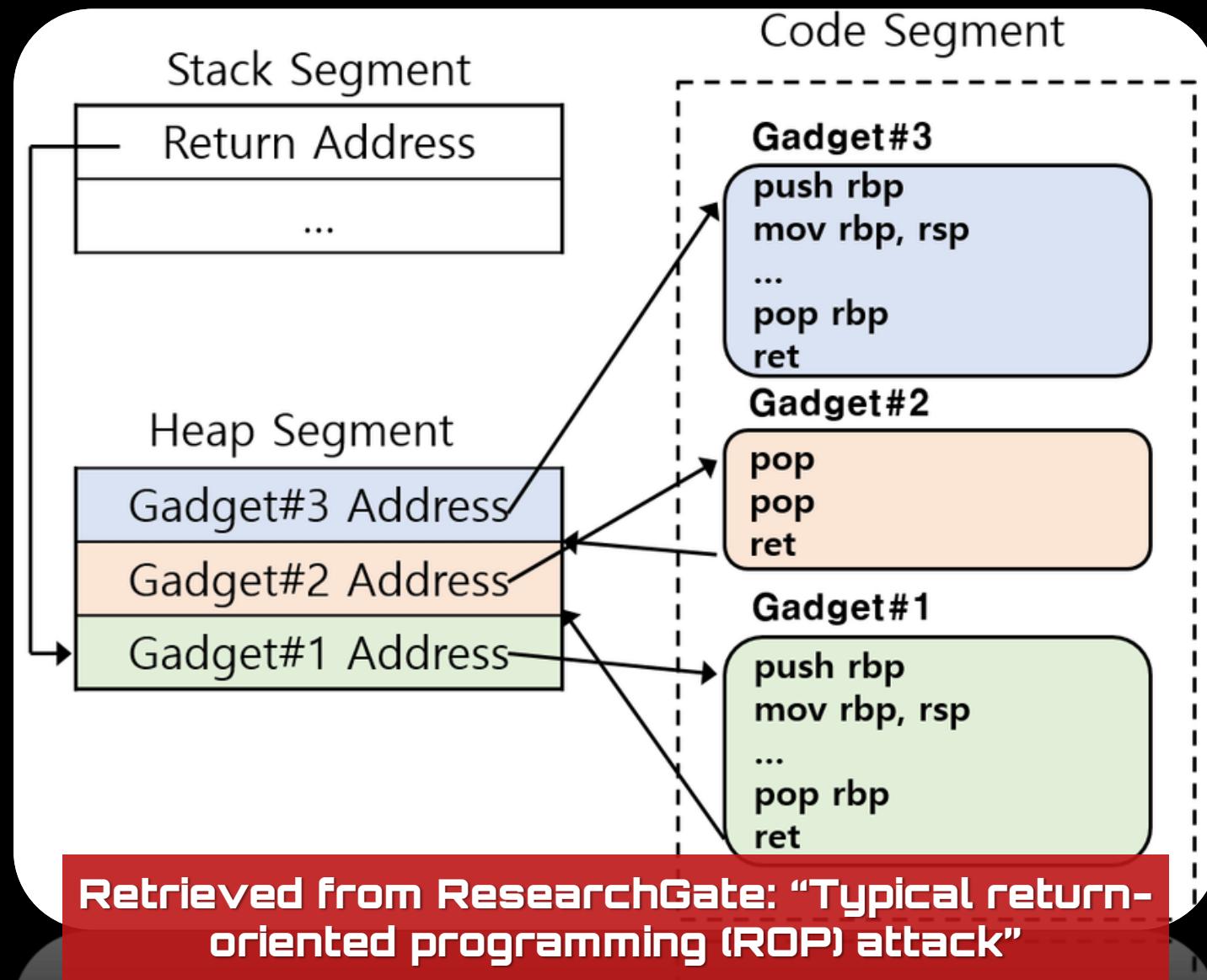
EXCEPTION ACCESS VIOLATION



```
773b1430 cc          int     3
0:006> !vprot eip
BaseAddress: 773b1000
AllocationBase: 77320000
AllocationProtect: 00000080 PAGE_EXECUTE_WRITECOPY
RegionSize: 00087000
State: 00001000 MEM_COMMIT
Protect: 00000020 PAGE_EXECUTE_READ
Type: 01000000 MEM_IMAGE
```

```
0:006> !vprot @esp
BaseAddress: 0661f000
AllocationBase: 065e0000
AllocationProtect: 00000004 PAGE_READWRITE
RegionSize: 00001000
State: 00001000 MEM_COMMIT
Protect: 00000004 PAGE_READWRITE
Type: 00020000 MEM_PRIVATE
```

ROP Local para control de flujo sin corrupción de memoria



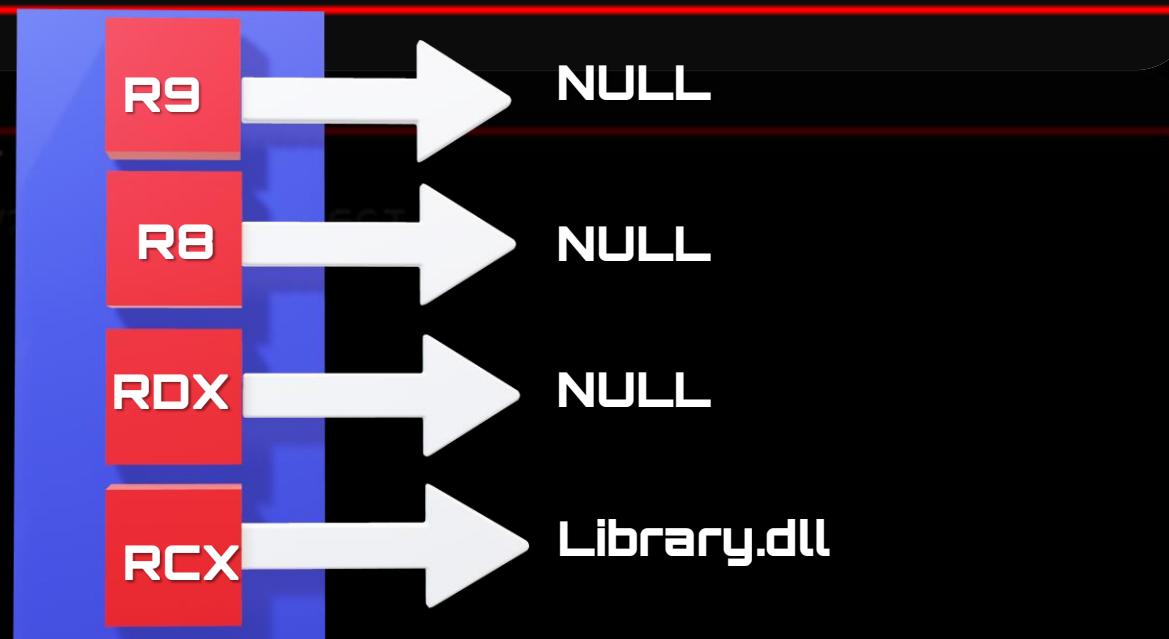
Return Oriented Programming

```
[+] Base Address of the process : 0x00007FF710B80000
[+] KernelBase.dll, POS: 3 ADDRESS: 00007FF827E50000
[+] LoadLibraryW Address: 0x7ff827ed2480
[+] .data Section RVA: 0x7000
[+] .data Section Virtual Size: 0x730
[+] Target Data Cave: 0x00007FF710B87730
[+] Aligning Stack to 16 bytes boundary: 0x00007FF710B87750
[+] Gadget #1: POP RCX ; NOP ; MOV EAX, 0x00000007F ; RET --> KERNELBASE.dll + 0xa095e
[+] Gadget #2: POP RBX ; RET --> KERNELBASE.dll + 0xbec1
[*] Gadget #3: CALL RBX --> KERNELBASE.dll + 0x7ab7
```

The command completed successfully.

C++

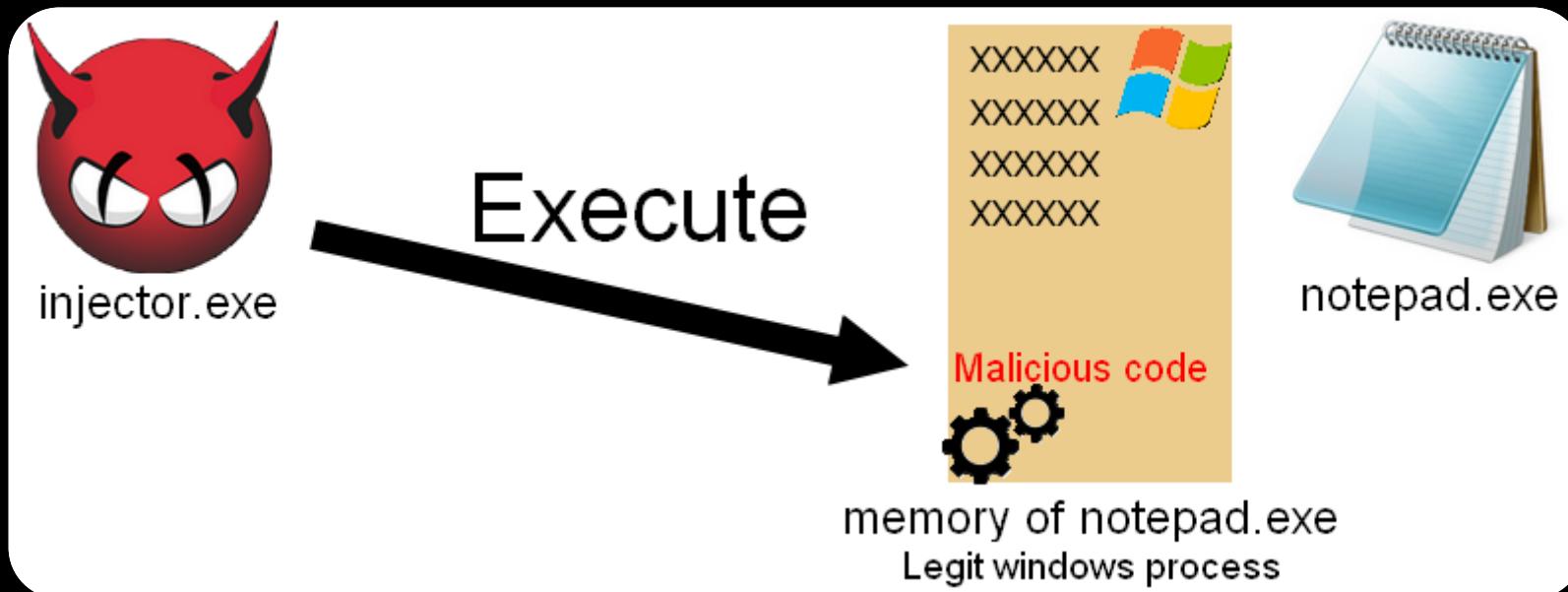
```
HMODULE LoadLibraryW(
    [in] LPCWSTR lpLibFileName
);
```



Return Oriented Programming



Process Hypnosis



Process Injection (Clásico)

Process Injection Clásico - Procedimiento

1.



Vinculación
OpenProcess();

2.



Reservar espacio en la memoria del proceso remoto

VirtualAllocEx();

3.



Escribir **ShellCode** en el espacio previamente reservado

WriteProcessMemory();

Process Injection Clásico - Procedimiento

4. Opcional



Cambiar las **protecciones** de la memoria

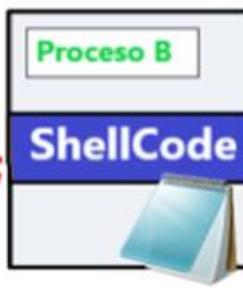
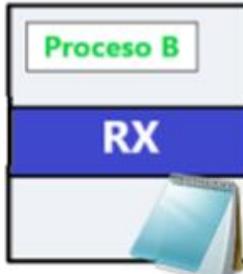
VirtualProtectEx();

5.



Ejecución

CreateRemoteThread();



Process Injection (Clásico)

PROCESS HYPNOSIS

GetProcAddress
GetModuleHandle

CREATE_SUSPENDED

VirtualAllocEx ResumeThread

CreateRemoteThread



Freeze

```
0: kd> bl
 0 e Disable Clear  ffffff807`6e6b2184
 1 e Disable Clear  ffffff807`6e71a744
```

```
0001 (0001) nt!PsFreezeProcess "!process -1 0; g"
0001 (0001) nt!PsThawProcess "!process -1 0"
```

```
0: kd> g
PROCESS fffff878b73339080
  SessionId: 1 Cid: 1558 Peb: 86d32fc000 ParentCid: 0fac
  DirBase: 4c5a8000 ObjectTable: fffffaf0e72a8e9c0 HandleCount: 0.
  Image: MRT.exe
```

```
PROCESS fffff878b73339080
  SessionId: 1 Cid: 1558 Peb: 86d32fc000 ParentCid: 0fac
  FreezeCount 1
  DirBase: 4c5a8000 ObjectTable: fffffaf0e72a8e9c0 HandleCount: 0.
  Image: MRT.exe
```

```
nt!PsThawProcess:
fffff807`6e71a744 88542410    mov    byte ptr [rsp+10h],dl
```

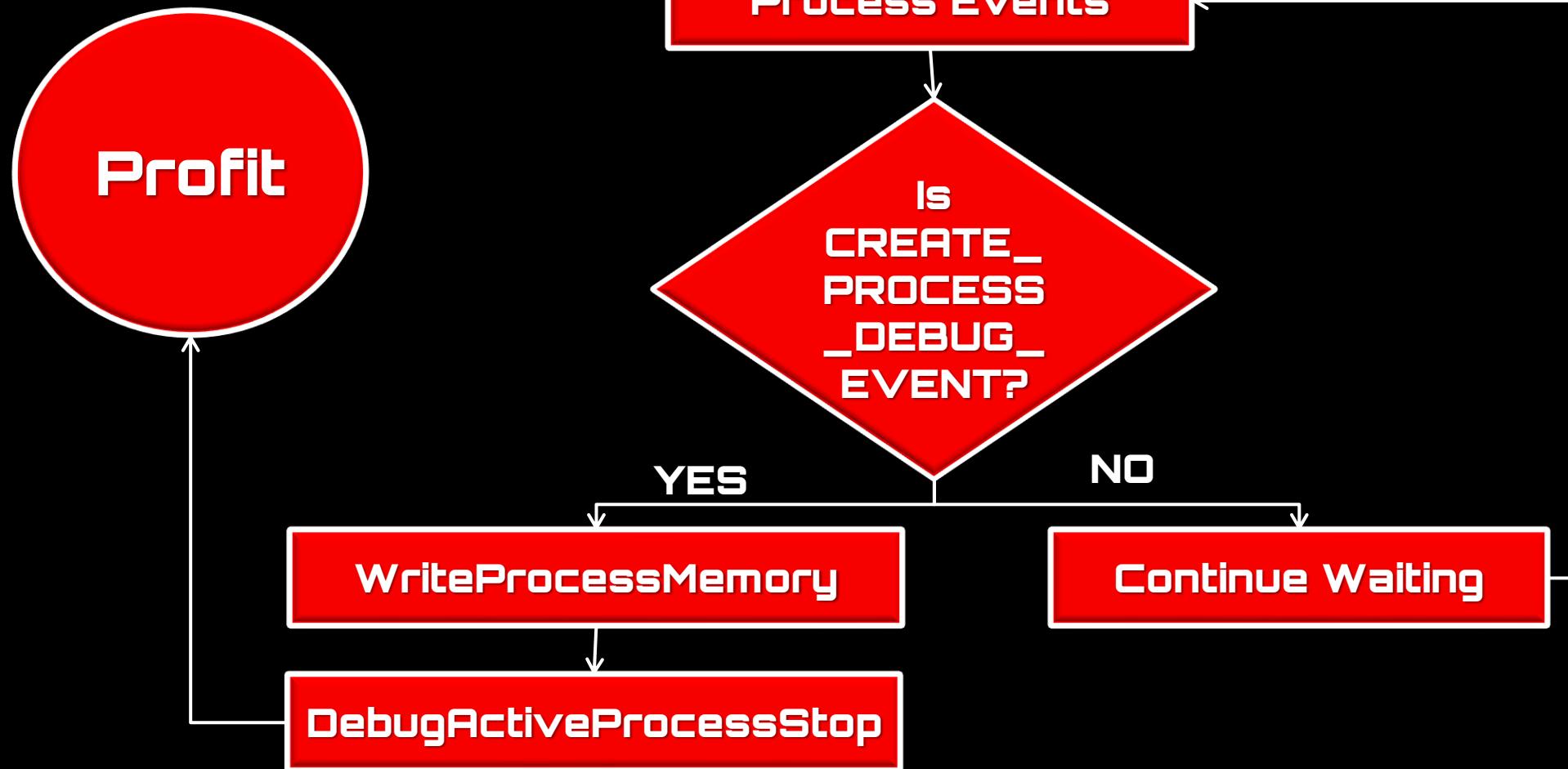
fffff807`6e71a744 88542410

mov byte ptr [rsb+10h],dl

nt!PsThawProcess

Freeze

Process Hypnosis



```
[+] Process created successfully - PID: 20512
[+] New Process Created - PID: 20512
[+] New Thread Created - TID: 16868
[+] Process lpStartAddress: 0x00007FF69F835BC0
[+] Process Main Thread: 0x00000000000000144

[+] DLL Remote Address: 0x000000DAB9780268
[+] DLL Name: ntdll.dll
[+] DLL Base Address: 0x00007FFD269F0000
[+] DLL hFile: 0x00000000000000134

[+] CreateRemoteThread Address: 0x00007FFD2684B190

[+] DLL Remote Address: 0x0000026022E27C50
[+] DLL Name: C:\WINDOWS\System32\KERNEL32.DLL
[+] DLL Base Address: 0x00007FFD26810000
[+] DLL hFile: 0x00000000000000160

[+] DLL Remote Address: 0x0000000000000000
[+] DLL Base Address: 0x00007FFD26810000
[+] DLL Name: C:\WINDOWS\System32\KERNEL32.DLL
[+] DLL Remote Address: 0x0000000000000000
```

```
[+] New Thread Created: 0x00007FFEE89A2B30
[+] New Thread Handle: 0x00000000000000114
[+] New Thread ThreadLocalBase: 0x000000AA85F8C000
```

Process Hypnosis

```
[+] Process [DEBUG] created successfully - PID: 7436
[+] New Process Created - PID: 7436
[+] New Thread Created - TID: 6836
[+] Process lpStartAddress: 0x00007FF7244D5BC0
[+] Process Main Thread: 0x0000000000000000DC

[+] Shellcode was successfully written [327 bytes]

[+] Successfully detached from the DEBUG process. Continuing the process' flow execution...
[+] Successfully detached from the DEBUG process. Continuing the process, from execution...
```

Process
Hypnosis



EDR

¿Qué evadimos?

Process	CPU	Private Bytes	Working Set	PID	Description	Company Name	Protection	
MRT.exe	< 0.01	6,604 K	3,544 K	5448	Microsoft Windows Malicious... Microsoft Corporation			
< >								
 Handles	 DLLs	 Threads						
 Stack	 Module	 Terminate						
State	Wait Reason	TID	User Time	Kernel Time	CPU	CPU Time	Start Time	Start Address
 Waiting	Suspended	3948	00:00:00	00:00:00		00:00:00	03/02/24 17:...	!RtlUserThreadStart
 Waiting	Executive	4504	00:00:00	00:00:00		00:00:00	03/02/24 17:...	!TpReleaseCleanupGroupMembers+0x450
 Waiting	UserRequest	7964	00:00:00	00:00:00		00:00:00	03/02/24 17:...	!RtlQueryProcessDebugInformationRemote
 Waiting	UserRequest	2584	00:00:00	00:00:00		00:00:00	03/02/24 17:...	!RtlQueryProcessDebugInformationRemote

RtlQueryProcessDebugInformationRemote –

Función no documentada

Detección



THANK
YOU