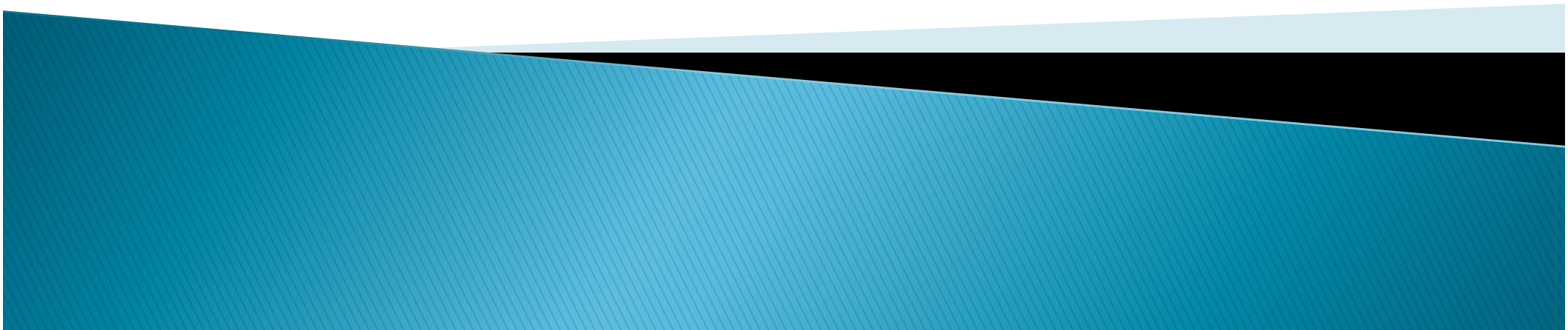
反-反外掛
從遊戲保護機制到Rootkit技術

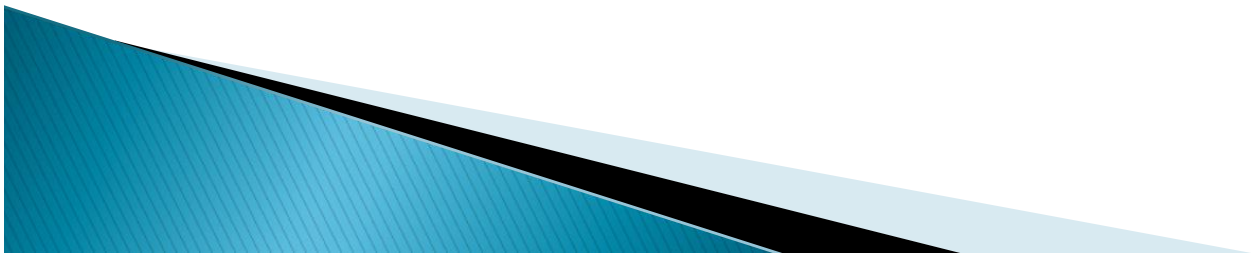
The Declaration of Hacker (TDOH)

LegBone(腿骨) & Singo



為甚麼要講這個議程...

腿骨說...搞了這麼多年的bypass，不來Hitcon 講一場 會很“幹”



About Me

Singo (TDOH中區召集人)
臺灣科技大學 資訊管理系

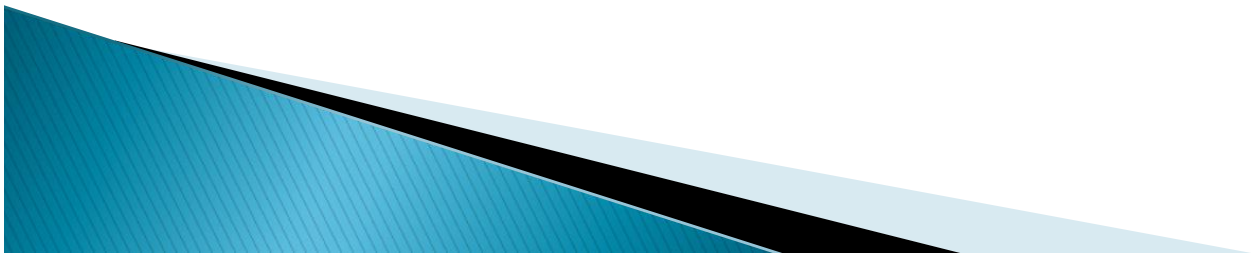
TDOH 2015 淺談逆向Android手遊
TDOH 2014 XSS新手初探之WorkShop
TDOH 2013 資安基礎入門
Sitcon 2015 第一次查課程評價就上手(網路爬蟲)

About TDOH

組織宗旨:

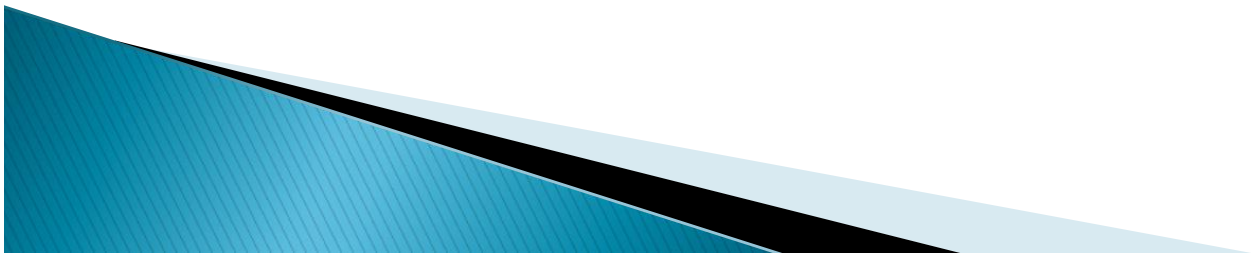
1. 培育更多資安人才
2. 推廣資安，改變大眾對駭客的刻板映像
3. 讓駭客們不再孤單，舉辦各種活動、聚會、網路論壇
4. 支援與聯合各學校社團，讓資安文化深入學校並傳承下去

談談今天的內容之前...



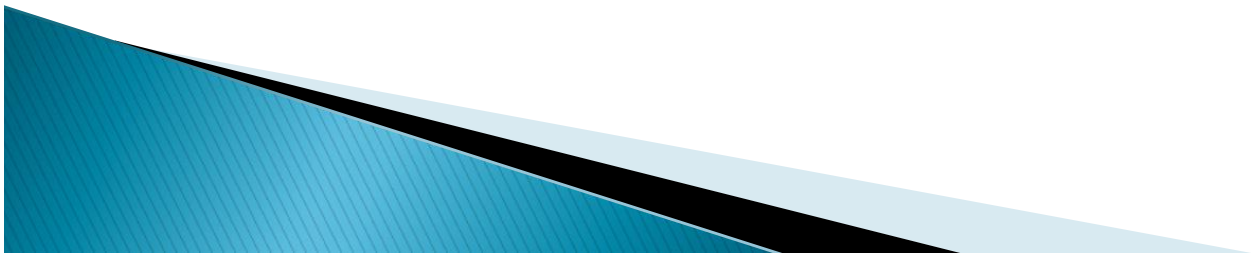
Ring3、Ring0是什麼? 能吃嗎?

- 遊戲常用的保護方式分為R3和R0:
  - –R3俗稱應用層
  - –R0俗稱內核層

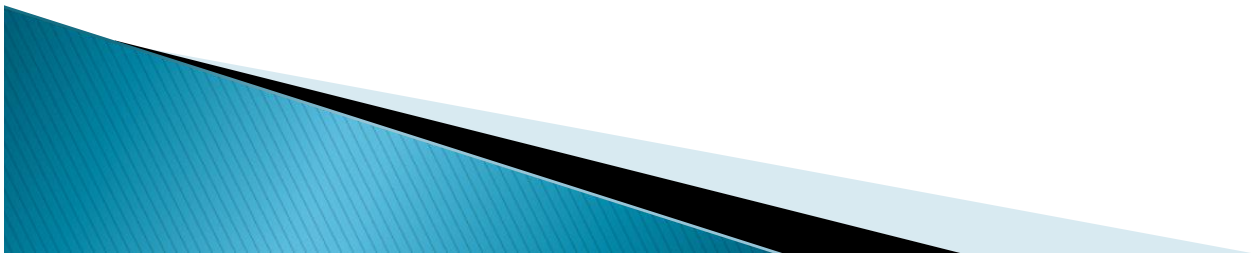


Ring3、Ring0是什麼? 能吃嗎?

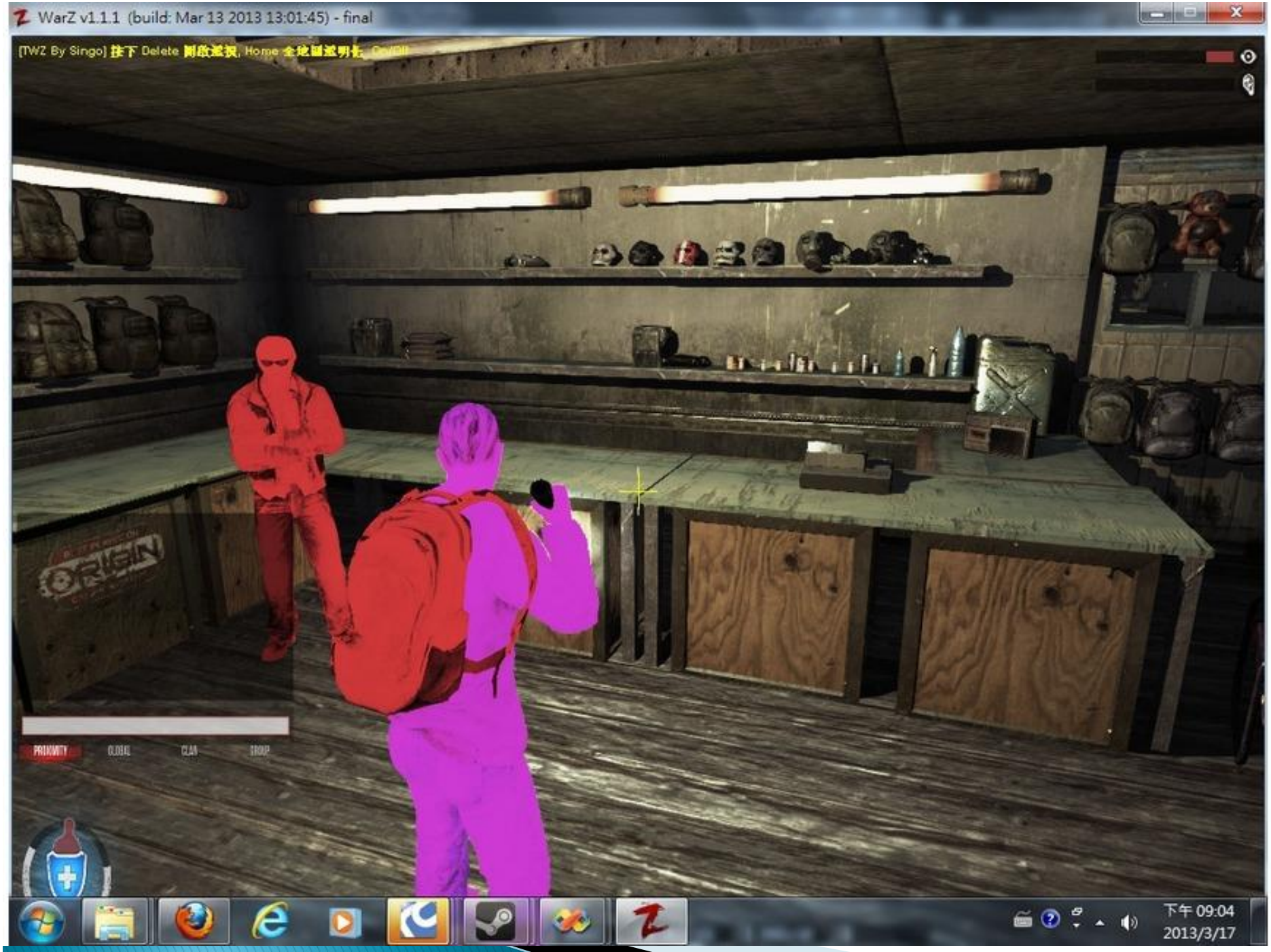
- 不僅系統有R3、R0之分
- 遊戲保護也有R3、R0之分



先聊聊我寫的遊戲外掛~~



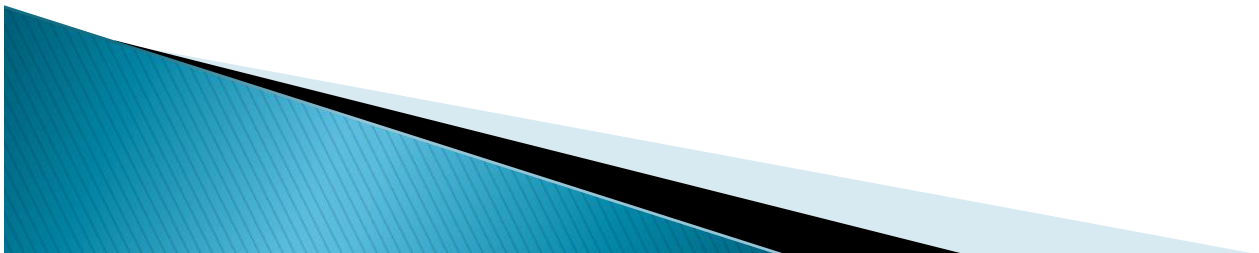
學習了DirectX 3D透視



簡單說明 射擊遊戲 透視的原理

- Hook DrawIndexedPrimitive函數
  禁用Z軸緩衝

- Hook EndScene 函數
  這可以將人物上色之類的



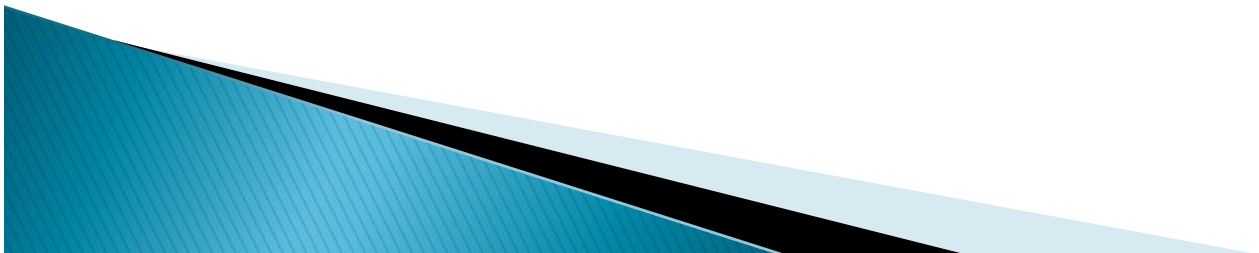

成吉思汗3智慧輔助 By Singo
人物信息
腳色名稱:
人物等級: 0
生命值: 0/0
魔力: 0/0
元氣: 0
經驗值: 0/0
輔助設置
生命低於 0 按 F1
生命低於 0 按 F1
真氣少於 0 按 F1
真氣少於 0 按 F1
元氣大於 200 按 F1
http://hot.fantasy.game.tw/
打怪設置
技能1 無
技能2 無
技能3 無
技能4 無
技能5 無
篩選怪物
開始
暫停
保存設置
讀取設置
關閉
答題警報
怪物篩選
自動選怪
打怪距離(1-9)
1
選怪設置

BUT….

人生中最厲害的就是這個BUT...

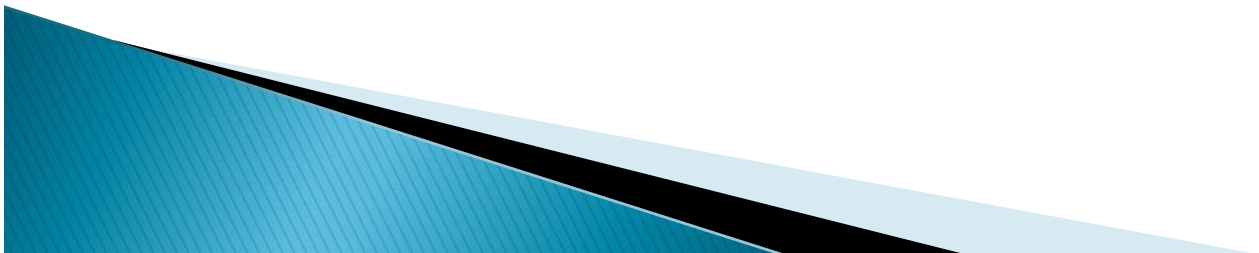
寫掛者討厭的東西?

去你的”驅動保護”...



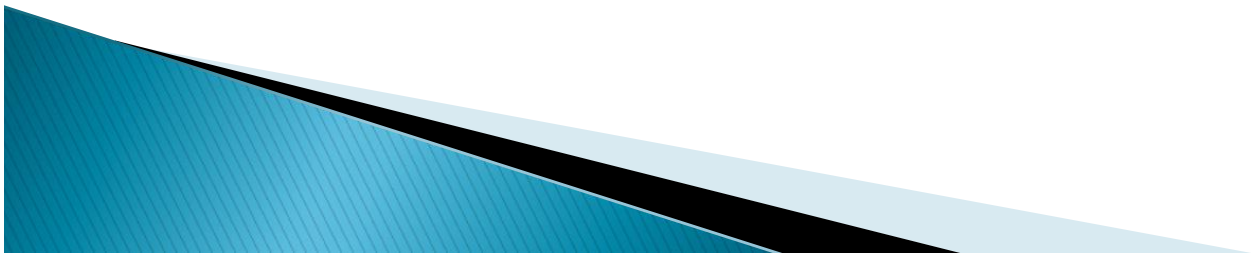

啊不就好棒棒

沒關係....

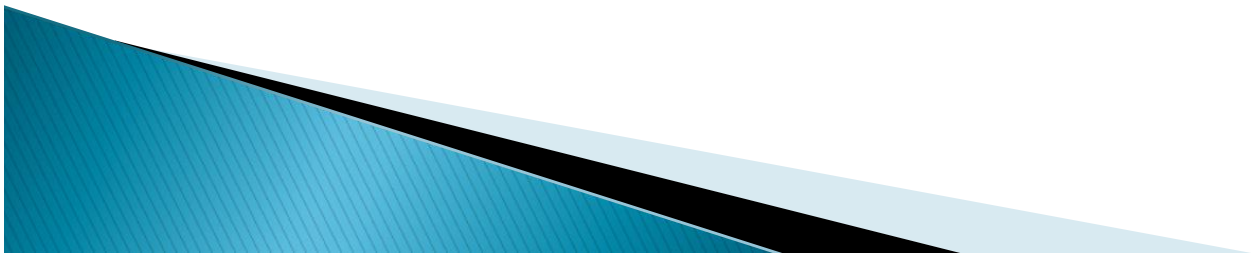


於是我認識了腿骨...

踏上了學習驅動之路...



我要特別講一下....

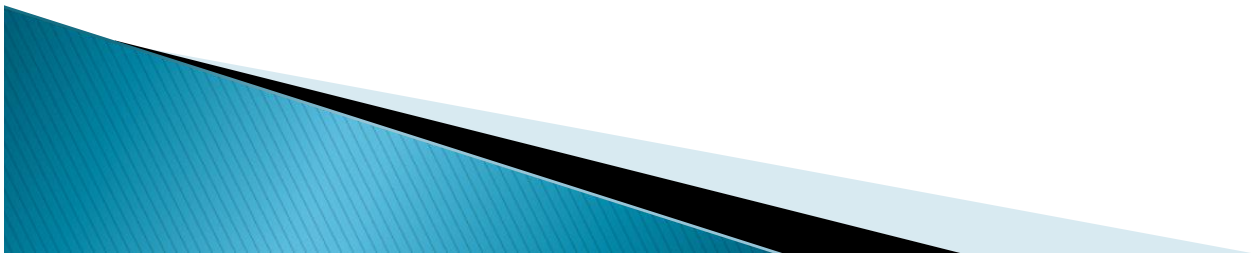


在資安個圈子裡面...

拜見強者的第一句話...

請先

2013年開始進行系統底層的研究



大概高2 還高3吧....

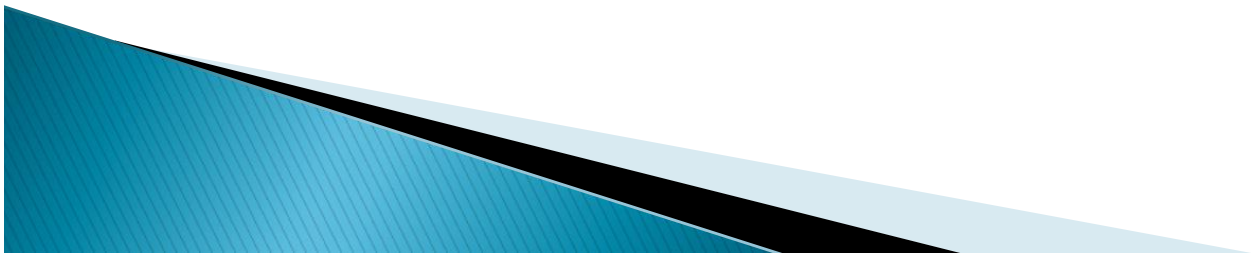
反–反外掛(防外掛機制)

矛盾大對決?

道高一尺，魔高一丈

R3的攻防方式

- R3的保護方式分為:
  - –遊戲本身自帶的
  - –額外包含進去的

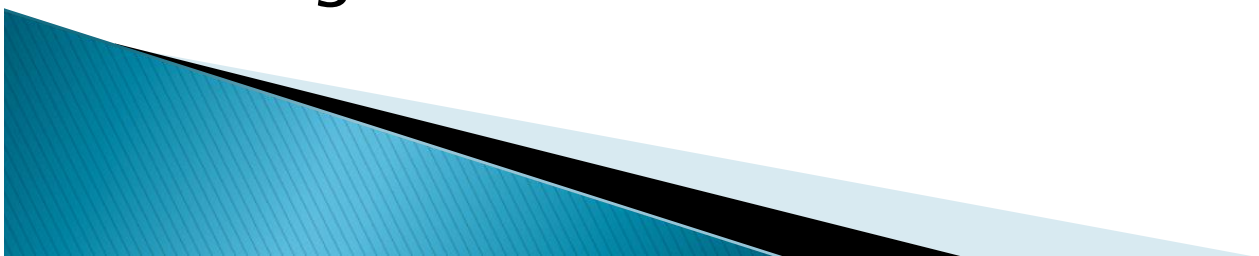


R3的攻防方式

- 談談R3常檢測的地方:
- FindWindow
- ProcessName
- PostMessage的lparam,wparam檢測

反調試:

DbgBreakPoint
DbgUserBreakPoint
DbgUiRemoteBreakin



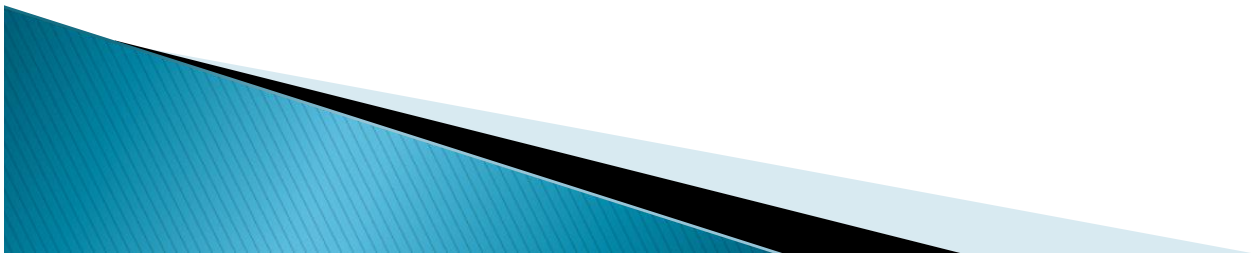
R3的攻防方式

- 反調試技術:

DbgUiRemoteBreakin
所有線程會去呼叫DbgBreakPoint

DbgBreakPoint 用來把當前的控制權交出去
Debugger就可以對他下int3斷點

DbgUserBreakPoint 長得和DbgBreakPoint一樣



R3的攻防方式

- 談談某防掛廠商反調試的方式:

掛鉤對象	掛鉤位置
len(1) ntdll.dll->DbgBreakPoint	0x77C740F0->_
len(5) ntdll.dll->DbgUiRemoteBreakin	0x77CDF125->0x77C9E...
len(1) ntdll.dll->DbgUserBreakPoint	0x77C740F0->_
len(5) ntdll.dll->NtProtectVirtualMemory	0x77C85F18->0x060921...
len(5) ntdll.dll->ZwProtectVirtualMemory	0x77C85F18->0x060921...

鉤子類型	掛鉤處當前值	掛鉤處原始值
inline	C3	CC
inline	E9 88 F2 FB FF	6A 08 68 E8 07
inline	C3	CC
inline	E9 D3 C2 40 8E	B8 D7 00 00 00
inline	E9 D3 C2 40 8E	B8 D7 00 00 00

R3的攻防方式

- 如果沒拔掉DbgUiRemoteBreakin的話:

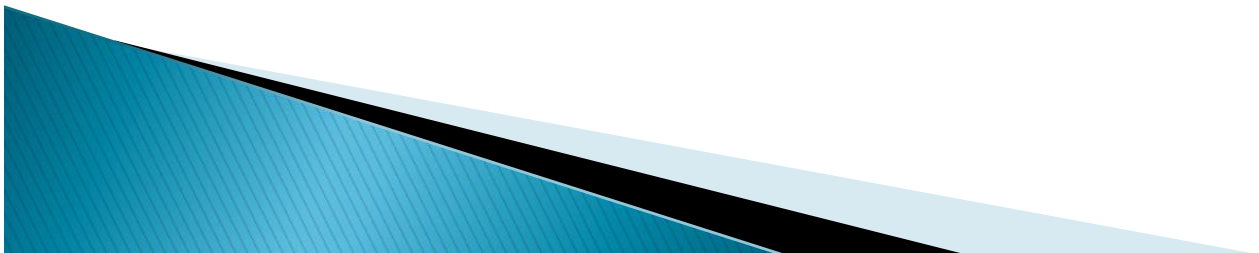
Address	Bytes	Opcode	
ntdll.DbgUiRemoteBreakin			
ntdll.DbgUiRemoteBreakin	E9 88F2FBFF	**jmp**	ntdll.LdrShutdownProcess
ntdll.DbgUiRemoteBreakin+5	36 77 E8	**ja**	ntdll.DbgUiStopDebugging+12
ntdll.DbgUiRemoteBreakin+8	DB 3A	**fstp**	tword ptr [edx]
ntdll.DbgUiRemoteBreakin+A	FB	**sti**	

R3的攻防方式

- 反-反外掛:

  修改ImagePath來偽裝成系統進程繞過檢測

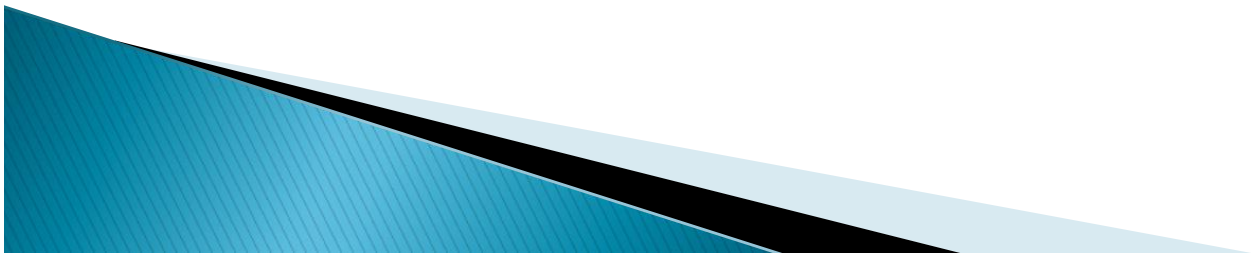
  有API HOOK的地方可以直接還原他的Hook
  (如果沒檢測Hook是否被還原的話...



R3的攻防方式

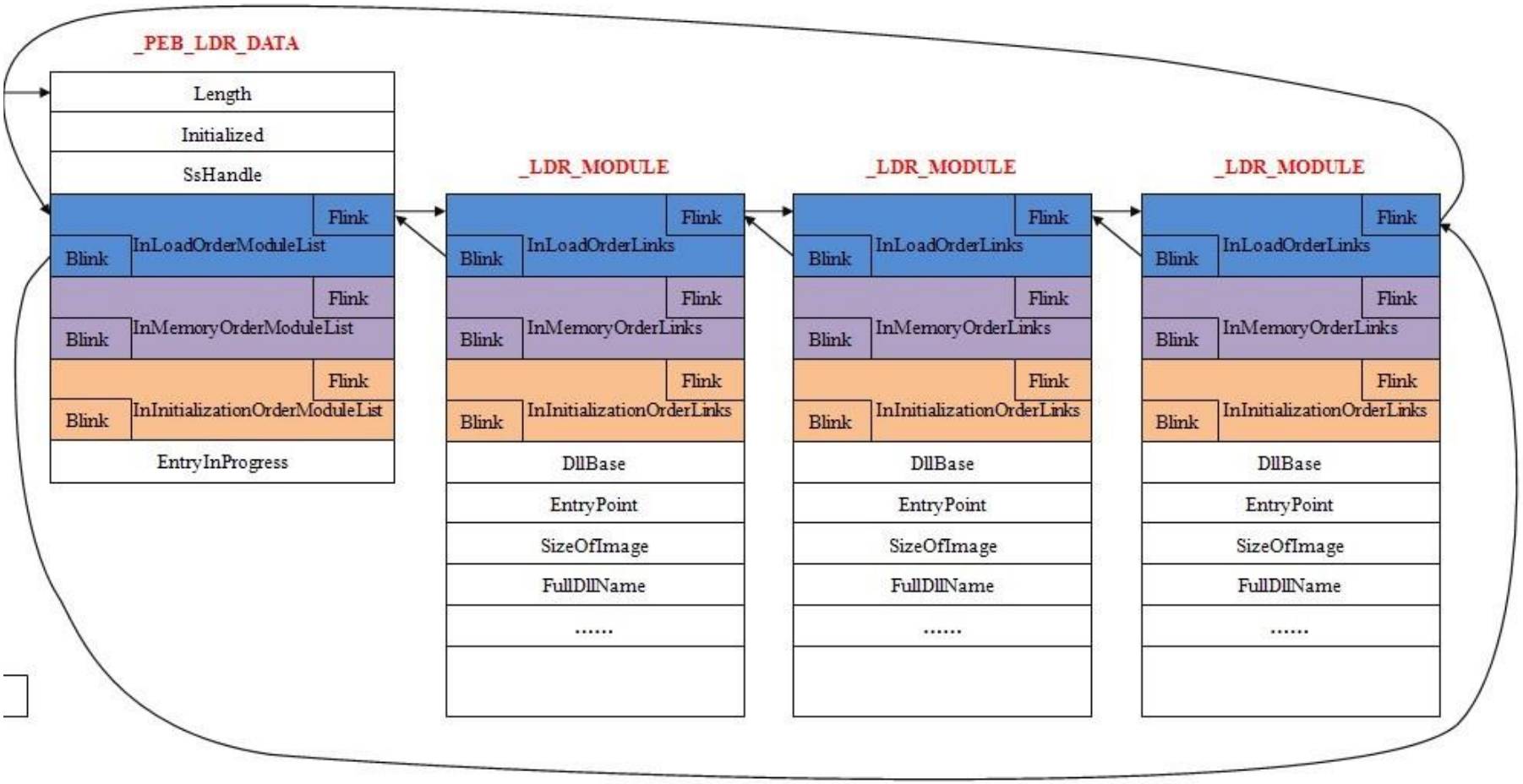
- 模塊隱藏:

  斷開_PEB_LDR_DATA 底下的三個鏈表來達到模塊隱藏
  InLoadOrderModuleList
  InMemoryOrderModuleList
  InInitializationOrderModuleList



R3的攻防方式

- 模塊隱藏:



R3的攻防方式

- 模塊隱藏:

```
ldm->InLoadOrderModuleList.Blink->Flink = ldm->InLoadOrderModuleList.Flink;
ldm->InLoadOrderModuleList.Flink->Blink = ldm->InLoadOrderModuleList.Blink;
ldm->InInitializationOrderModuleList.Blink->Flink = ldm->InInitializationOrderModuleList.Flink;
ldm->InInitializationOrderModuleList.Flink->Blink = ldm->InInitializationOrderModuleList.Blink;
ldm->InMemoryOrderModuleList.Blink->Flink = ldm->InMemoryOrderModuleList.Flink;
ldm->InMemoryOrderModuleList.Flink->Blink = ldm->InMemoryOrderModuleList.Blink;
```

R3的攻防方式

- 抹去PE標誌:

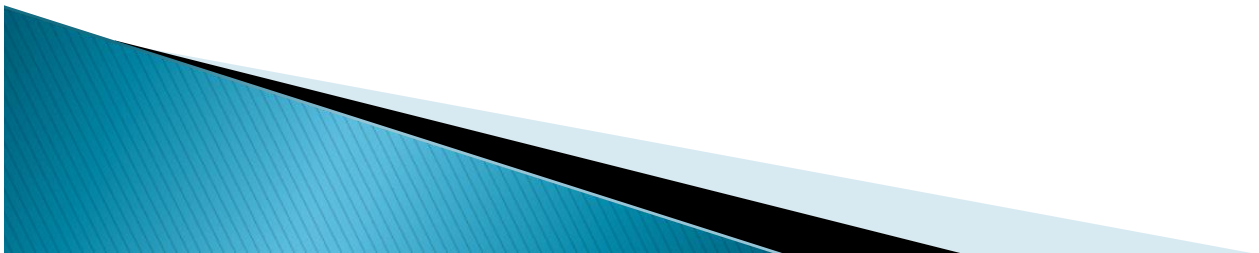
  將此處填0即可
  Signature=0;

R3的攻防方式

- 介紹常用的DLL注入方式:

1. 註冊表注入
2. 遠線程注入
3. 白名單注入
4. 輸入法注入

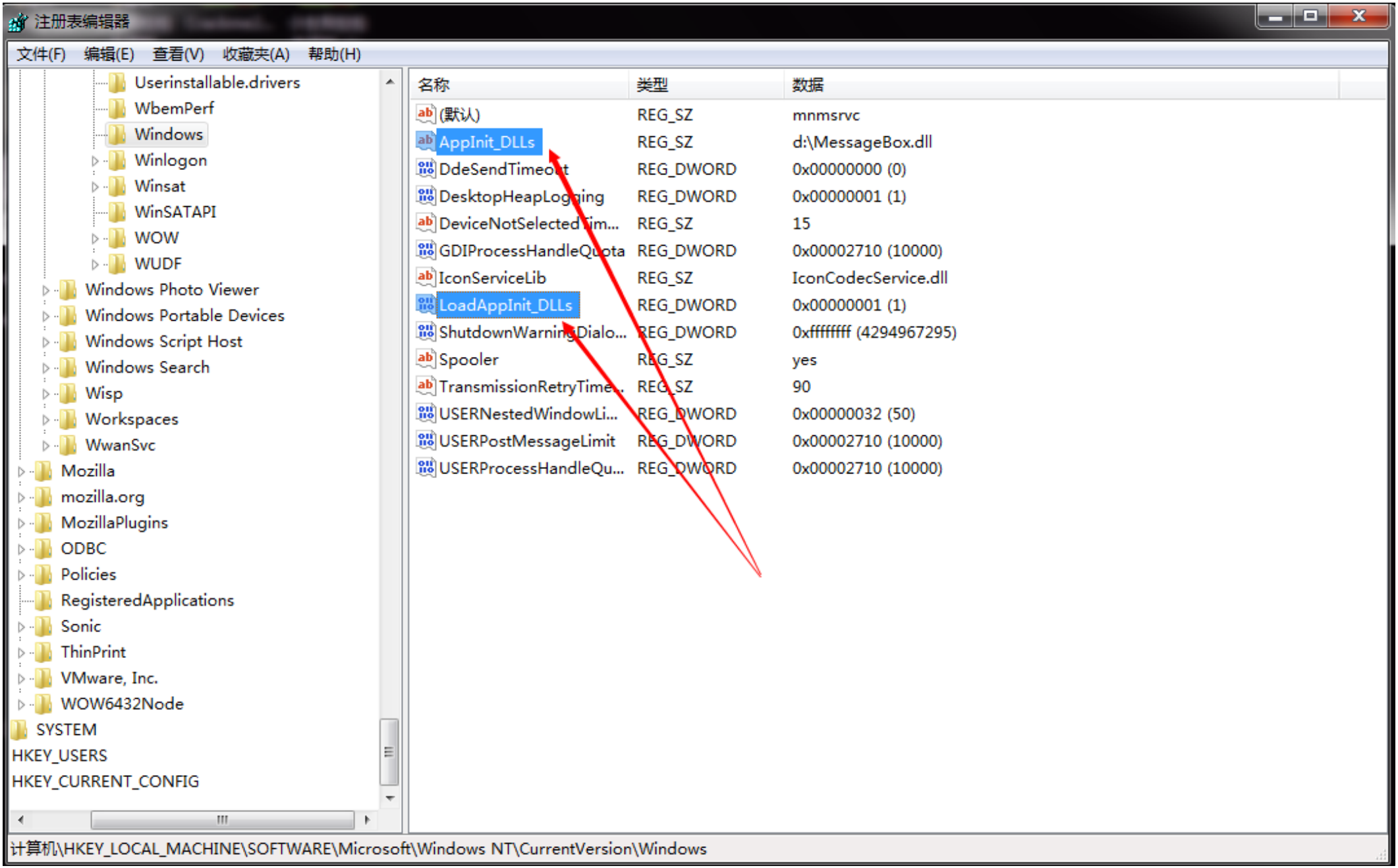
.....等等



R3的攻防方式

- 註冊表注入:

  利用系統的user32.dll啟動時，會加載註冊表AppInit_Dlls下所列出來的DLL，根據這個原理可以將欲注入的DLL路徑寫進AppInit_Dlls下，來達成注入。



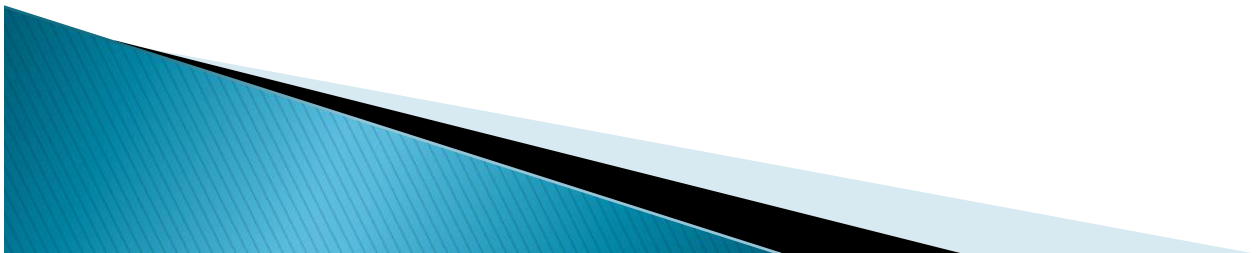
引用至:
http://blog.csdn.net/u013565525/article/details/28416279

R3的攻防方式

- 遠線程注入:

遠線程注入的核心概念是利用Windows提供的遠線程機制，在目標進程中開啟一個加載DLL的遠線程，使外掛DLL被該遠線程所加載到遊戲的記憶體內。

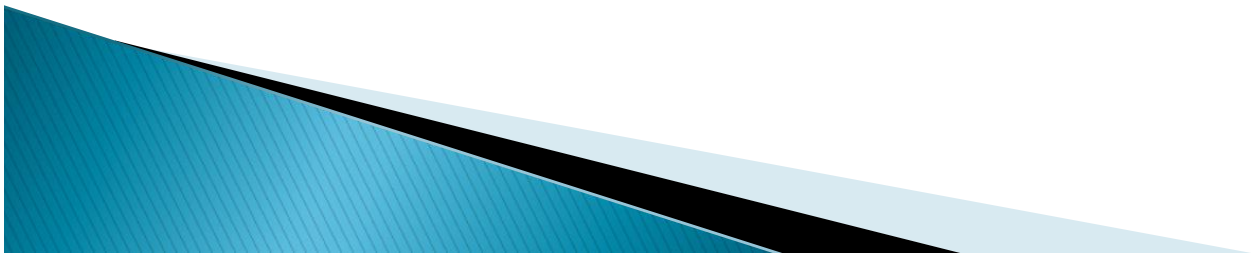
會用到的API有:
OpenProcess、GetProcAddress、VirtualAllocEx、WriteProcessMemory(WPM)、CreateRemoteThread、LoadLibrary。



R3的攻防方式

- 白名單注入:

  原理是利用系統重要的進程來幫我們進行DLL的注入也就是第三方注入。

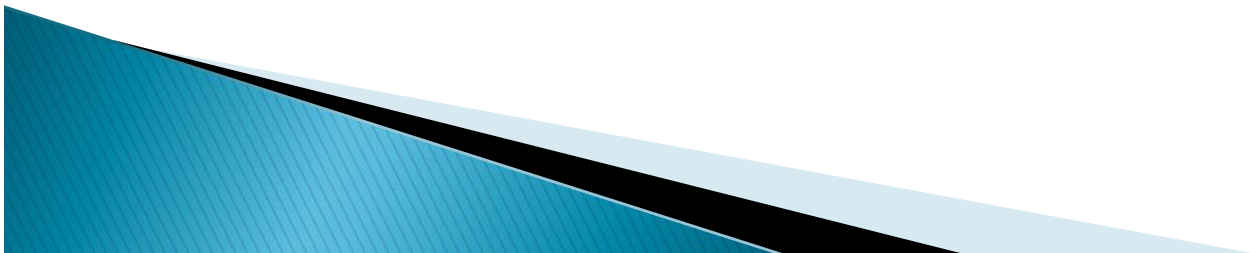


R3的攻防方式

- 輸入法注入:

由於大多數的反外掛都阻止了外部程序讀取遊戲或注入遊戲,但是卻有一個最重要的東西一定要載入遊戲，就是輸入法!

透過加載輸入法時的code在LoadLibrary的動作時,將我們的動態鏈結庫加載進遊戲記憶體中，來達到注入的效果!

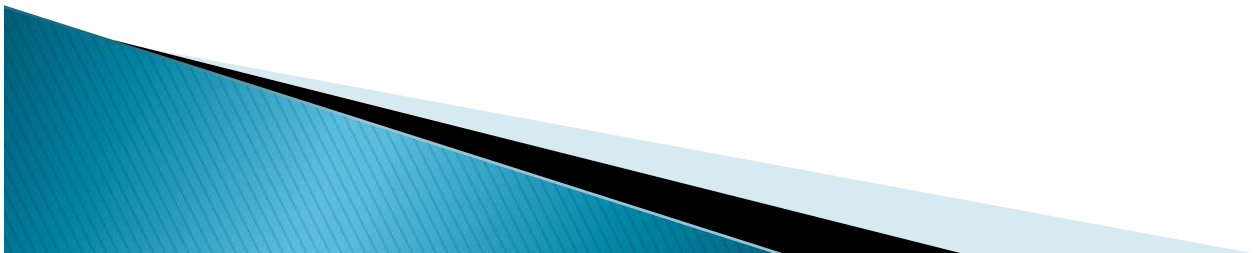


R3的攻防方式

- 注入的結論:

  其實注入方式不只這些

  只要能好好的將DLL注入進去就是好方法!!!



反-反外掛
從遊戲保護機制到 Rootkit 技術
Ring 0 層

TDOH-Singo
TDOH-LegBone

剛剛Singo講完後。。。。

約於 1 分鐘前

您好

蔡耀德大大

。。。。。。。。

About Me

撰寫BY PASS Hackshield
TDOHacker 南區召集人
SITCON 2014 2015 short talk 講者
東華大學資工週講者
中正大學逆向課程講者
死大學生
晶睿科技資安實習生
擅長惡意程式分析
常用的語言：asm,vc,vb,python
~~穩定單身中的小小魯蛇Q_Q~~
實際職業是個廚師 =3=

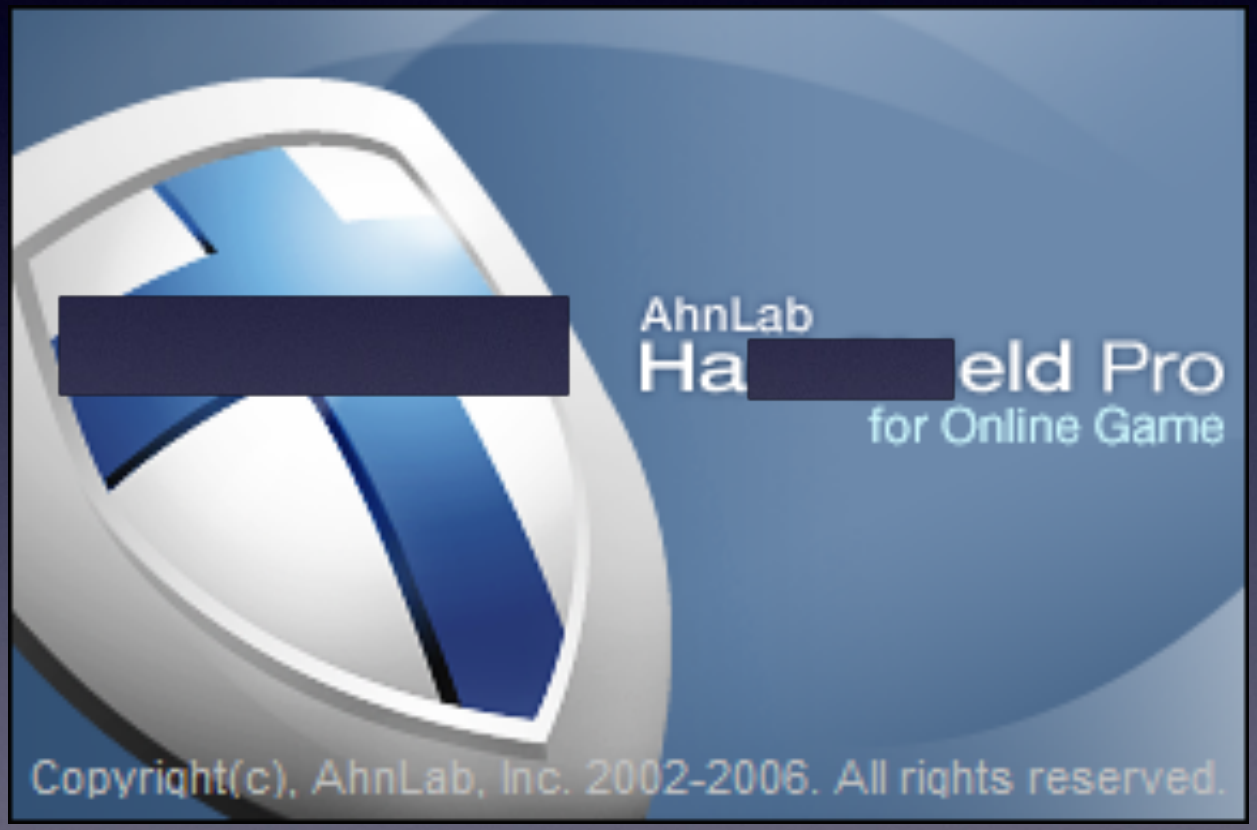
原本這次想穿這套上台講。。。。。

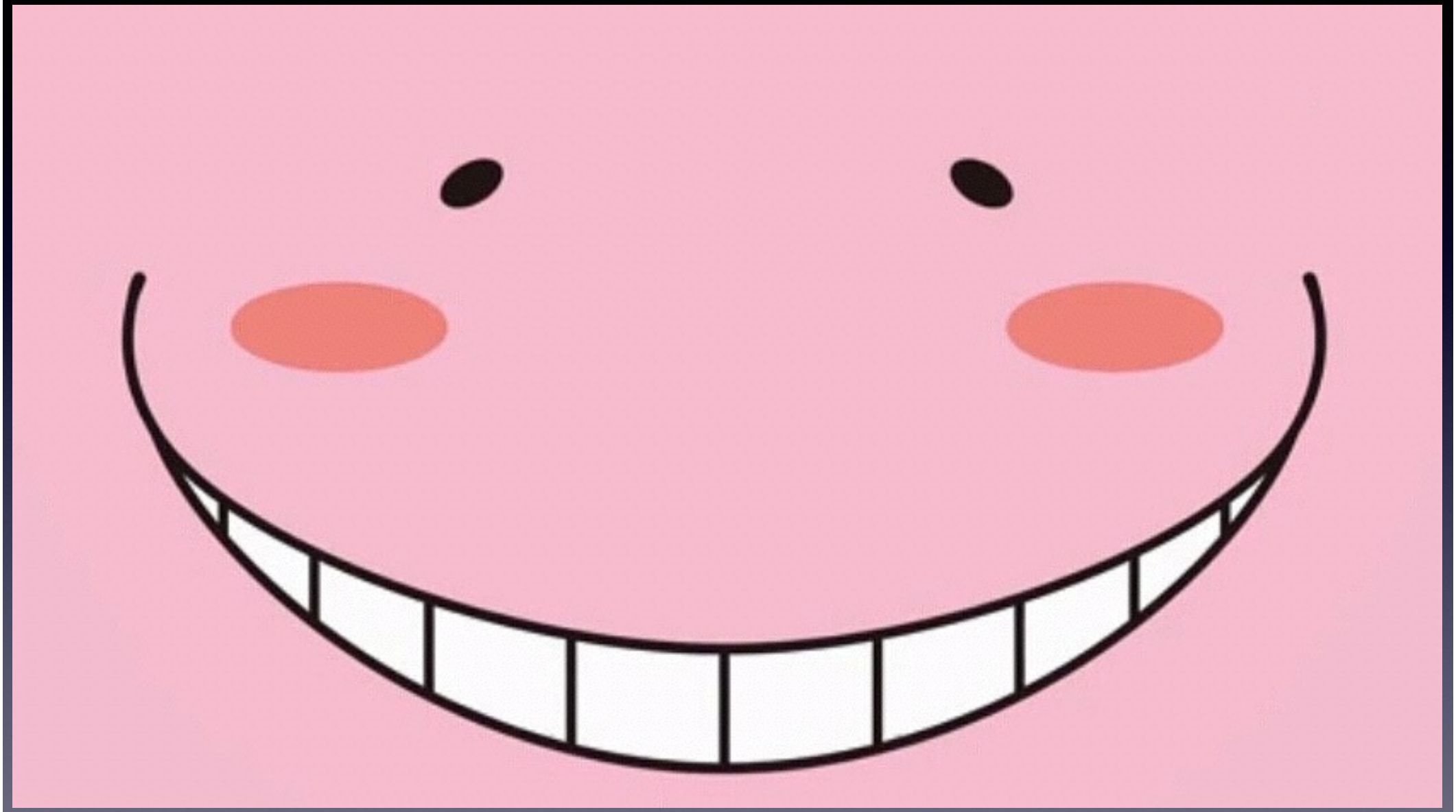
但是在上台的前幾天。。。


一路發
貨運網
此郵件是系統自動傳送，請勿直接回覆此郵件
※為確保您能每次收到信件，請將本信件設定為非垃圾信件。
親愛的蔡耀德，您好！
廈門站查件貨號[redacted],深圳口岸海關查車，等候放行後重新安排航班，請耐心等候謝謝。。

剛剛那隻黑黑的狗講了很多Ring3的東西

但是現在跟反外掛的戰爭打到Ring0了！





windows的函數調用過程

?.dll/exe調用了Openprocess

windows的函數調用過程

?.dll/exe調用了Openprocess

kernel32.OpenProcess

windows的函數調用過程

?.dll/exe調用了Openprocess

kernel32.OpenProcess

ntdll.NtOpenProcess

windows的函數調用過程

?.dll/exe調用了Openprocess

kernel32.OpenProcess

ntdll.NtOpenProcess

nt!KiFastCallEntry

windows的函數調用過程

?.dll/exe調用了Openprocess

kernel32.OpenProcess

ntdll.NtOpenProcess

nt!KiFastCallEntry

NtOpenProcess

windows的函數調用過程

?.dll/exe調用了Openprocess

kernel32.OpenProcess

ntdll.NtOpenProcess

nt!KiFastCallEntry

NtOpenProcess

反外掛

用PCHunter可以看到SSDT表。。。
被HOOK到了HS的EagleXNt.sys

SSDT | ShadowSSDT | FSD | 鍵盤 | I8042prt | 鼠標 | Partmgr | Disk | Atapi | Acpi | Scsi | 內核鉤子 | Object鉤子 | 系統中斷表

序號	函數名稱	當前函數地址	Hook	原始函數地址	當前函數地址所在模塊
50	NtClose	0x877EEAB0->0x836...	inline hook	0x84095420	C:\Windows\system32\drivers\EagleXNt.sys
215	NtProtectVirtualMemory	0x881CC608->0x836...	inline hook	0x840934A9	C:\Windows\system32\drivers\EagleXNt.sys
277	NtReadVirtualMemory	0x87F99A80->0x836...	inline hook	0x840B08D2	C:\Windows\system32\drivers\EagleXNt.sys
399	NtWriteVirtualMemory	0x8836E980->0x836...	inline hook	0x840807C2	C:\Windows\system32\drivers\EagleXNt.sys

但是不只這些ＱＷＱ

len(5) NtClose[ntkrnlpa.exe]	[0x84095420]->[0x877EEAB0][->0x836DF950==>C:\Windows\system32\drivers\EagleXNt.sys]
len(4) NtDeviceIoControlFile[ntkrnlpa.exe]	[0x840C4498]->[0x8852C180][->0x836DF570==>C:\Windows\system32\drivers\EagleXNt.sys]
len(4) NtDuplicateObject[ntkrnlpa.exe]	[0x840825DF]->[0x9F8A1CF0][C:\Users\test\Downloads\VStart50\VStart50\TOOLS\sys\PCHunter_fre
len(4) NtOpenProcess[ntkrnlpa.exe]	[0x84062A81]->[0x86613AE8][->0x836DFB80==>C:\Windows\system32\drivers\EagleXNt.sys]
len(1) KiFastCallEntry[ntkrnlpa.exe]	[0x83E8E349]->[-]
len(4) NtGetContextThread[ntkrnlpa.exe]	[0x840E5E90]->[0x8814D590][->0x836E11F0==>C:\Windows\system32\drivers\EagleXNt.sys]
len(4) NtProtectVirtualMemory[ntkrnlpa.exe]	[0x840934B1]->[0x881CC608][->0x836DEDE0==>C:\Windows\system32\drivers\EagleXNt.sys]
len(4) NtReadVirtualMemory[ntkrnlpa.exe]	[0x840808DA]->[0x87F99A80][->0x836DFCE0==>C:\Windows\system32\drivers\EagleXNt.sys]
len(4) NtSetContextThread[ntkrnlpa.exe]	[0x8412DDFB]->[0x87FB3610][->0x836E1050==>C:\Windows\system32\drivers\EagleXNt.sys]
len(4) NtTerminateProcess[ntkrnlpa.exe]	[0x840ABAA8]->[0x9F8A2310][C:\Users\test\Downloads\VStart50\VStart50\TOOLS\sys\PCHunter_fre
len(4) NtTerminateThread[ntkrnlpa.exe]	[0x840C942D]->[0x9F8A2310][C:\Users\test\Downloads\VStart50\VStart50\TOOLS\sys\PCHunter_fre
len(4) NtWriteVirtualMemory[ntkrnlpa.exe]	[0x840807CA]->[0x8836E980][->0x836DFE60==>C:\Windows\system32\drivers\EagleXNt.sys]
len(1) [ntkrnlpa.exe]	[0x83EC795F]->[-]
len(1) [ntkrnlpa.exe]	[0x83EC7BAD]->[-]
len(22) [ntkrnlpa.exe]	[0x83EC7D52]->[-]
len(1) [ntkrnlpa.exe]	[0x83EC7D6F]->[-]
len(4) [ntkrnlpa.exe]	[0x83ED38A8]->[0x88529828][->0x836DEF90==>C:\Windows\system32\drivers\EagleXNt.sys]
len(4) [ntkrnlpa.exe]	[0x840613CB]->[0x882181A8][->0x836E1410==>C:\Windows\system32\drivers\EagleXNt.sys]
len(4) [ntkrnlpa.exe]	[0x8407DC31]->[0x882EB9B0][->0x836DF9B0==>C:\Windows\system32\drivers\EagleXNt.sys]
len(4) [ntkrnlpa.exe]	[0x840AEDC0]->[0x9F8A1AB0][C:\Users\test\Downloads\VStart50\VStart50\TOOLS\sys\PCHun
len(4) [ntkrnlpa.exe]	[0x840C7142]->[0x9F8A1AB0][C:\Users\test\Downloads\VStart50\VStart50\TOOLS\sys\PCHun
len(4) [ntkrnlpa.exe]	[0x840E5ECC]->[0x881C6608][->0x836E1390==>C:\Windows\system32\drivers\EagleXNt.sys]
len(4) [ntkrnlpa.exe]	[0x840E5FB5]->[0x881FD608][->0x836E0B40==>C:\Windows\system32\drivers\EagleXNt.sy
len(5) [win32k.sys]	[0x9551BAE9]->[0x9F8A0B80][C:\Users\test\Downloads\VStart50\VStart50\TOOLS\sys\PCHu

HOOK函數可以做什麼？

SSDT HOOK

- SSDT 系統描述服務表

SSDT HOOK

- SSDT 系統描述服務表
- KiFastCallEntry會從SSDT取得函數地址

SSDT HOOK

- SSDT 系統描述服務表
- KiFastCallEntry會從SSDT取得函數地址
- 替換掉裡面的函數地址！

SSDT HOOK

- SSDT 系統描述服務表
- KiFastCallEntry會從SSDT取得函數地址
- 替換掉裡面的函數地址！
- 調用函數的時候順便調用我們的函數！

inline HOOK

- 直接修改函數

inline HOOK

- 直接修改函數
- call或jmp走

inline HOOK

- 直接修改函數
- call或jmp走
- 之後記得跳回來

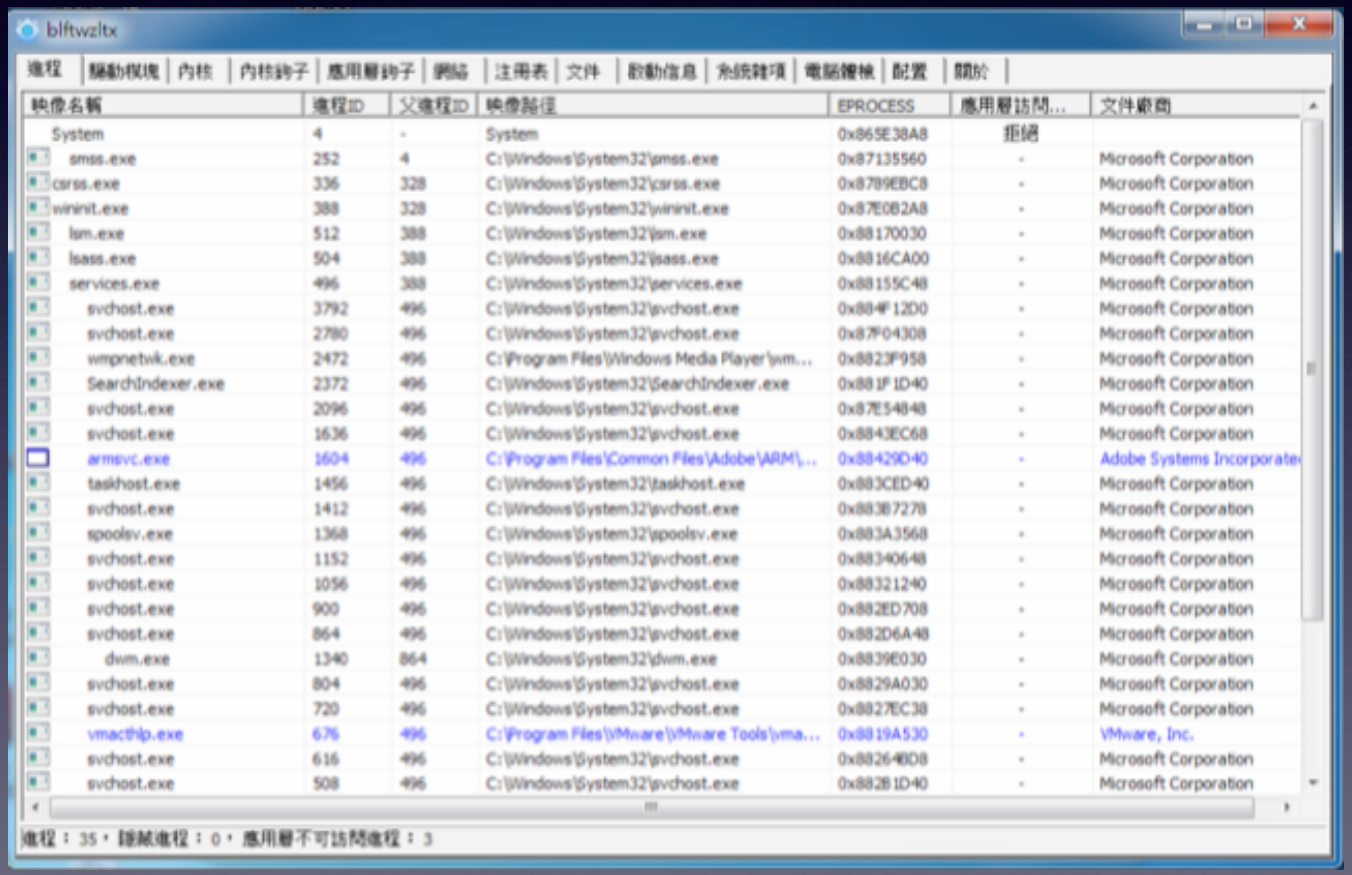
inline HOOK

- 直接修改函數
- call或jmp走
- 之後記得跳回來
- 調用函數的時候順便調用我們的函數！

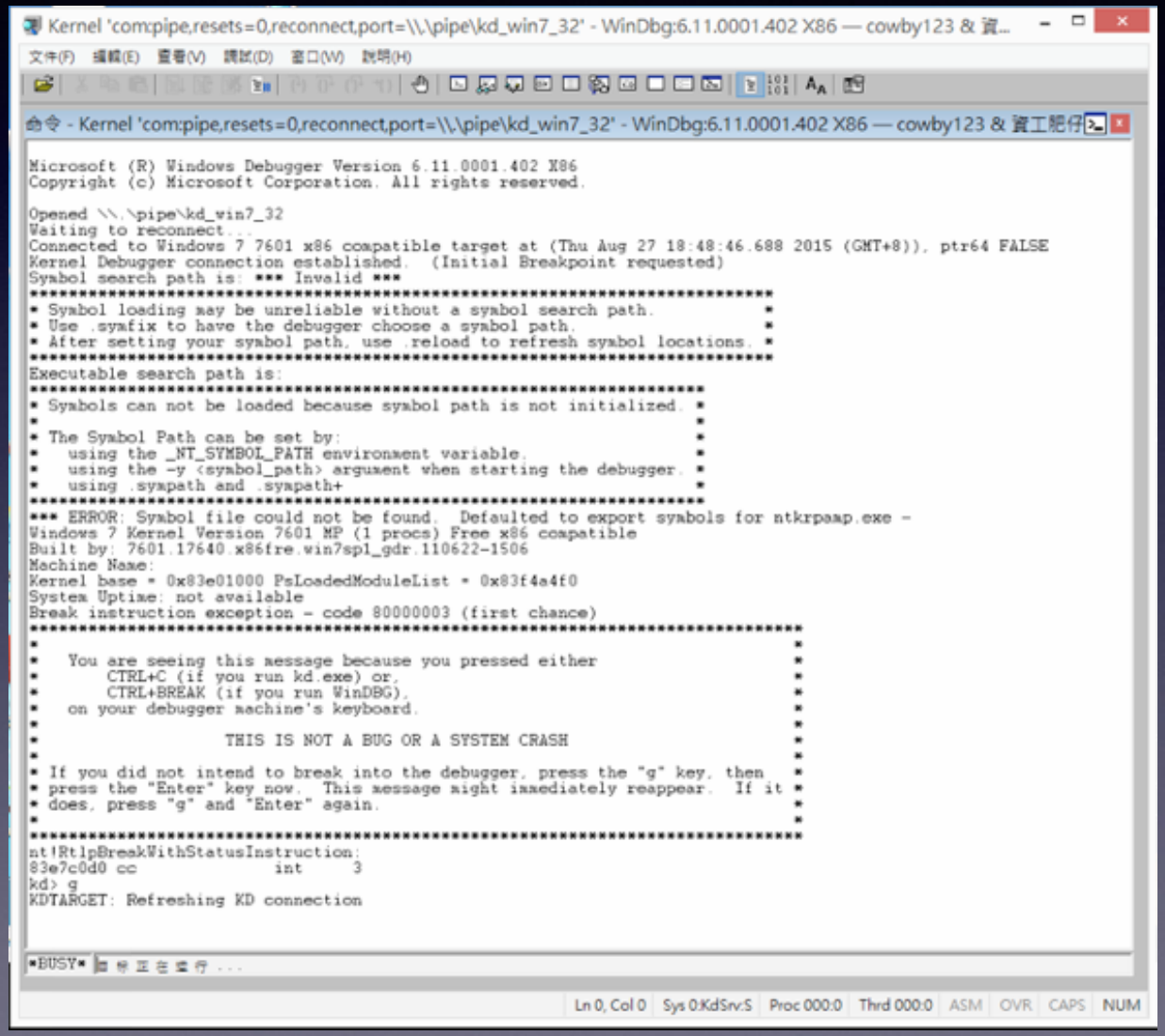
- 調用該函數的時候順便調用我們的函數！

分析工具

PCHunter

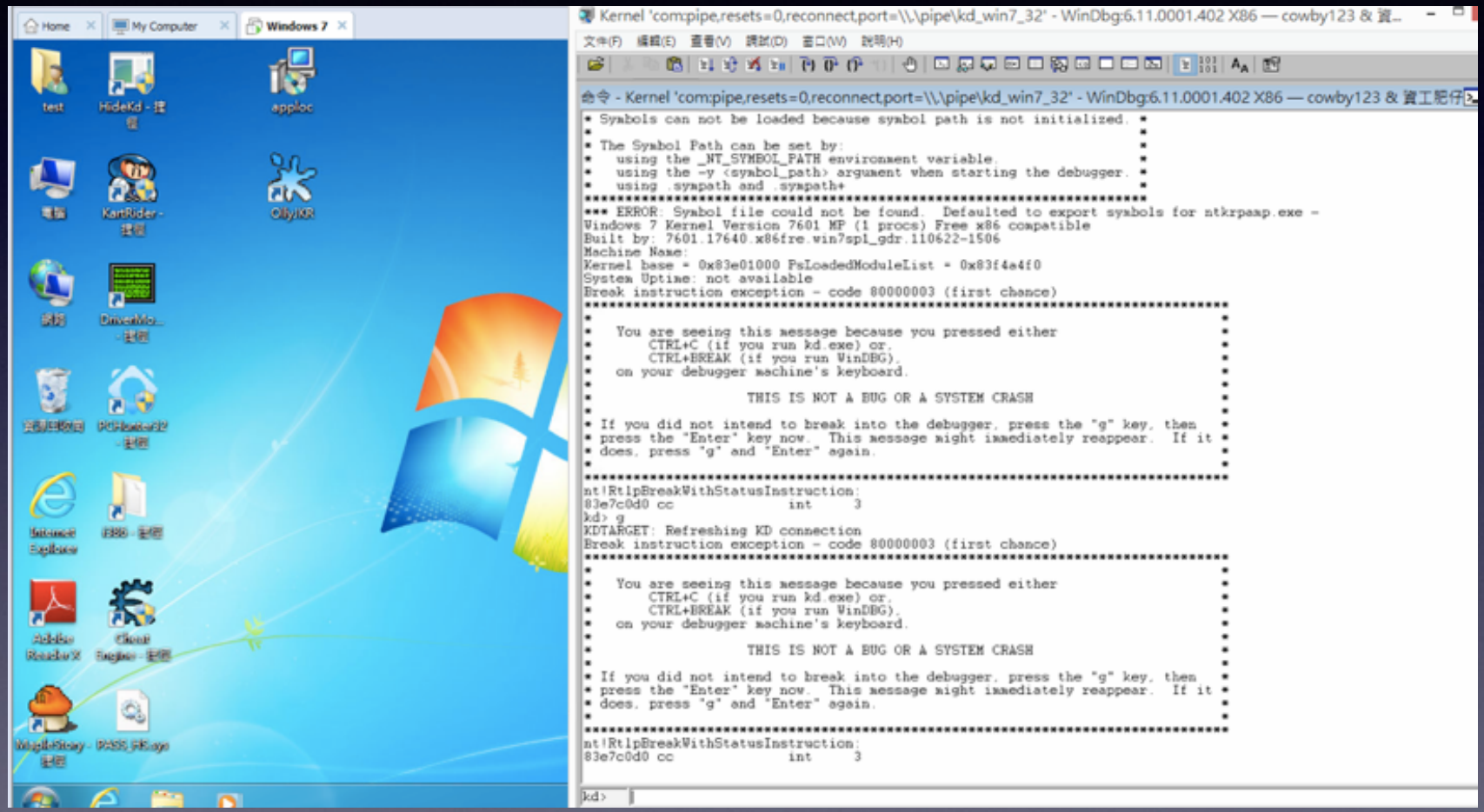

blftwzltx
進程 | 驅動模塊 | 內核 | 內核鉤子 | 應用層鉤子 | 網絡 | 注冊表 | 文件 | 啟動信息 | 系統雜項 | 電腦體檢 | 配置 | 關於
映像名稱 | 進程ID | 父進程ID | 映像路徑 | EPROCESS | 應用層訪問... | 文件廠商
System 4 - System 0x865E38A8 拒絕
smss.exe 252 4 C:\Windows\System32\smss.exe 0x87135560 - Microsoft Corporation
csrss.exe 336 328 C:\Windows\System32\csrss.exe 0x8789EBC8 - Microsoft Corporation
wininit.exe 388 328 C:\Windows\System32\wininit.exe 0x87E0B2A8 - Microsoft Corporation
lsm.exe 512 388 C:\Windows\System32\lsm.exe 0x88170030 - Microsoft Corporation
lsass.exe 504 388 C:\Windows\System32\lsass.exe 0x8816CA00 - Microsoft Corporation
services.exe 496 388 C:\Windows\System32\services.exe 0x88155C48 - Microsoft Corporation
svchost.exe 3792 496 C:\Windows\System32\svchost.exe 0x884F12D0 - Microsoft Corporation
svchost.exe 2780 496 C:\Windows\System32\svchost.exe 0x87F04308 - Microsoft Corporation
wmpnetwk.exe 2472 496 C:\Program Files\Windows Media Player\wm... 0x8823F958 - Microsoft Corporation
SearchIndexer.exe 2372 496 C:\Windows\System32\SearchIndexer.exe 0x881F1D40 - Microsoft Corporation
svchost.exe 2096 496 C:\Windows\System32\svchost.exe 0x87E54848 - Microsoft Corporation
svchost.exe 1636 496 C:\Windows\System32\svchost.exe 0x8843EC68 - Microsoft Corporation
armsvc.exe 1604 496 C:\Program Files\Common Files\Adobe\ARM\... 0x88429D40 - Adobe Systems Incorporate
taskhost.exe 1456 496 C:\Windows\System32\taskhost.exe 0x883CED40 - Microsoft Corporation
svchost.exe 1412 496 C:\Windows\System32\svchost.exe 0x883B7278 - Microsoft Corporation
spoolsv.exe 1368 496 C:\Windows\System32\spoolsv.exe 0x883A3568 - Microsoft Corporation
svchost.exe 1152 496 C:\Windows\System32\svchost.exe 0x88340648 - Microsoft Corporation
svchost.exe 1056 496 C:\Windows\System32\svchost.exe 0x88321240 - Microsoft Corporation
svchost.exe 900 496 C:\Windows\System32\svchost.exe 0x882ED708 - Microsoft Corporation
svchost.exe 864 496 C:\Windows\System32\svchost.exe 0x882D6A48 - Microsoft Corporation
dwm.exe 1340 864 C:\Windows\System32\dwm.exe 0x8839E030 - Microsoft Corporation
svchost.exe 804 496 C:\Windows\System32\svchost.exe 0x8829A030 - Microsoft Corporation
svchost.exe 720 496 C:\Windows\System32\svchost.exe 0x8827EC38 - Microsoft Corporation
vmacthlp.exe 676 496 C:\Program Files\VMware\VMware Tools\vma... 0x8819A530 - VMware, Inc.
svchost.exe 616 496 C:\Windows\System32\svchost.exe 0x882648D8 - Microsoft Corporation
svchost.exe 508 496 C:\Windows\System32\svchost.exe 0x882B1D40 - Microsoft Corporation
進程：35，隱藏進程：0，應用層不可訪問進程：3

WinDbg(繁中)


Kernel 'com:pipe,resets=0,reconnect,port=\\.\pipe\kd_win7_32' - WinDbg:6.11.0001.402 X86 — cowby123 & 資...
文件(F) 編輯(E) 查看(V) 調試(D) 窗口(W) 說明(H)
命令 - Kernel 'com:pipe,resets=0,reconnect,port=\\.\pipe\kd_win7_32' - WinDbg:6.11.0001.402 X86 — cowby123 & 資工肥仔
Microsoft (R) Windows Debugger Version 6.11.0001.402 X86
Copyright (c) Microsoft Corporation. All rights reserved.
Opened \\.\pipe\kd_win7_32
Waiting to reconnect...
Connected to Windows 7 7601 x86 compatible target at (Thu Aug 27 18:48:46.688 2015 (GMT+8)), ptr64 FALSE
Kernel Debugger connection established. (Initial Breakpoint requested)
Symbol search path is: *** Invalid ***
* Symbol loading may be unreliable without a symbol search path. *
* Use .symfix to have the debugger choose a symbol path. *
* After setting your symbol path, use .reload to refresh symbol locations. *
Executable search path is:
* Symbols can not be loaded because symbol path is not initialized. *
* The Symbol Path can be set by: *
* using the _NT_SYMBOL_PATH environment variable. *
* using the -y <symbol_path> argument when starting the debugger. *
* using .sympath and .sympath+ *
*** ERROR: Symbol file could not be found. Defaulted to export symbols for ntkrpamp.exe -
Windows 7 Kernel Version 7601 MP (1 procs) Free x86 compatible
Built by: 7601.17640.x86fre.win7sp1_gdr.110622-1506
Machine Name:
Kernel base = 0x83e01000 PsLoadedModuleList = 0x83f4a4f0
System Uptime: not available
Break instruction exception - code 80000003 (first chance)
* You are seeing this message because you pressed either *
* CTRL+C (if you run kd.exe) or, *
* CTRL+BREAK (if you run WinDBG), *
* on your debugger machine's keyboard. *
* THIS IS NOT A BUG OR A SYSTEM CRASH *
* If you did not intend to break into the debugger, press the "g" key, then *
* press the "Enter" key now. This message might immediately reappear. If it *
* does, press "g" and "Enter" again. *
nt!RtlpBreakWithStatusInstruction:
83e7c0d0 cc int 3
kd> g
KDTARGET: Refreshing KD connection
BUSY
Ln 0, Col 0
Sys 0:KdSrv:S
Proc 000:0
Thrd 000:0
ASM
OVR
CAPS
NUM

分析方法

windbg+vmware雙機調試



至於怎麼設置。。。。

Google第一頁就有了
別問我ORZ

用途？好處？

今天是拿HS開刀，他用的是inlint hook

這是原本的nt!NtOpenProcess

```
kd> u 84062a58 84062a58+0x30
nt!NtOpenProcess:
84062a58 8bff            mov     edi,edi
84062a5a 55              push    ebp
84062a5b 8bec            mov     ebp,esp
84062a5d 51              push    ecx
84062a5e 51              push    ecx
84062a5f 64a124010000    mov     eax,dword ptr fs:[00000124h]
84062a65 8a803a010000    mov     al,byte ptr [eax+13Ah]
84062a6b 8b4d14          mov     ecx,dword ptr [ebp+14h]
84062a6e 8b5510          mov     edx,dword ptr [ebp+10h]
84062a71 8845fc          mov     byte ptr [ebp-4],al
84062a74 ff75fc          push    dword ptr [ebp-4]
84062a77 ff75fc          push    dword ptr [ebp-4]
84062a7a ff750c          push    dword ptr [ebp+0Ch]
84062a7d ff7508          push    dword ptr [ebp+8]
84062a80 e88f440600      call    nt!PsOpenProcess (840c6f14)
84062a85 c9              leave
84062a86 c21000          ret     10h
```

這是HS啟動以後的nt!NtOpenProcess

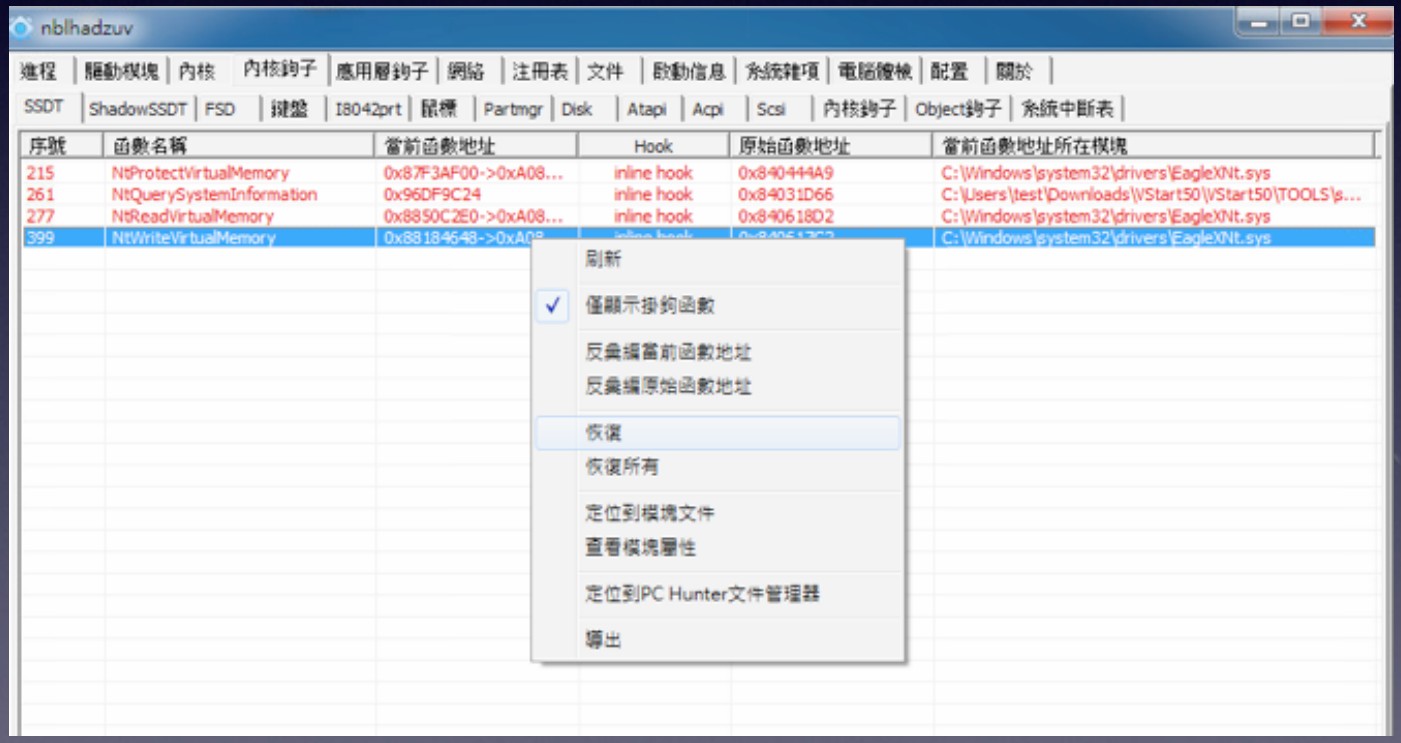
```
kd> u 84062a58 84062a58+0x30
nt!NtOpenProcess:
84062a58 8bff            mov     edi,edi
84062a5a 55              push    ebp
84062a5b 8bec            mov     ebp,esp
84062a5d 51              push    ecx
84062a5e 51              push    ecx
84062a5f 64a124010000    mov     eax,dword ptr fs:[00000124h]
84062a65 8a803a010000    mov     al,byte ptr [eax+13Ah]
84062a6b 8b4d14          mov     ecx,dword ptr [ebp+14h]
84062a6e 8b5510          mov     edx,dword ptr [ebp+10h]
84062a71 8845fc          mov     byte ptr [ebp-4],al
84062a74 ff75fc          push    dword ptr [ebp-4]
84062a77 ff75fc          push    dword ptr [ebp-4]
84062a7a ff750c          push    dword ptr [ebp+0Ch]
84062a7d ff7508          push    dword ptr [ebp+8]
84062a80 e863105b02      call    86613ae8
84062a85 c9              leave
84062a86 c21000          ret     10h
```

跟進 Call 86613AE8

```
kd> u 86613ae8
*** ERROR: Module load completed but symbols could not be loaded for EagleXNt.sys
86613ae8 e993c00cfd      jmp     EagleXNt+0x18b80 (836dfb80)
86613aed 90              nop
86613aee 90              nop
86613aef 90              nop
86613af0 e91f34abfd      jmp     nt!PsOpenProcess (840c6f14)
86613af5 90              nop
86613af6 90              nop
86613af7 90              nop
```

怎麼處理？

直接恢復嗎？


nblhadzuv
進程 | 驅動模塊 | 內核 | 內核鉤子 | 應用層鉤子 | 網絡 | 注冊表 | 文件 | 啟動信息 | 系統雜項 | 電腦體檢 | 配置 | 關於
SSDT | ShadowSSDT | FSD | 鍵盤 | I8042prt | 鼠標 | Partmgr | Disk | Atapi | Acpi | Scsi | 內核鉤子 | Object鉤子 | 系統中斷表
序號 函數名稱 當前函數地址 Hook 原始函數地址 當前函數地址所在模塊
215 NtProtectVirtualMemory 0x87F3AF00->0xA08... inline hook 0x840444A9 C:\Windows\system32\drivers\EagleXNt.sys
261 NtQuerySystemInformation 0x96DF9C24 inline hook 0x84031D66 C:\Users\test\Downloads\VStart50\VStart50\TOOLS\s...
277 NtReadVirtualMemory 0x8850C2E0->0xA08... inline hook 0x840618D2 C:\Windows\system32\drivers\EagleXNt.sys
399 NtWriteVirtualMemory 0x88184648->0xA0 C:\Windows\system32\drivers\EagleXNt.sys
刷新
僅顯示掛鉤函數
反彙編當前函數地址
反彙編原始函數地址
恢復
恢復所有
定位到模塊文件
查看模塊屬性
定位到PC Hunter文件管理器
導出

我的解法是。。。

```
nt!RtlpBreakWithStatusInstruction:
83e7c0d0 cc              int     3
kd> u nt!ntwritevirtualmemory
nt!NtWriteVirtualMemory:
840617c2 6a18            push    18h
840617c4 68008ae583      push    offset nt! ?? ::FNODOBFM::`string'+0x3e80 (83e58a00)
840617c9 e89af3e1ff      call    nt!_SEH_prolog4 (83e80b68)
840617ce 648b3d24010000  mov     edi,dword ptr fs:[124h]
840617d5 8a873a010000    mov     al,byte ptr [edi+13Ah]
840617db 8845e4          mov     byte ptr [ebp-1Ch],al
840617de 8b7514          mov     esi,dword ptr [ebp+14h]
840617e1 84c0            test    al,al
```

```
kd> u nt!ntwritevirtualmemory
nt!NtWriteVirtualMemory:
840617c2 e9b97c831c      jmp     PASS_HS!MyNtWriteVirtualMemory (a0899480)
840617c7 e583            in      eax,83h
840617c9 e87a2e1204      call    88184648
840617ce 648b3d24010000  mov     edi,dword ptr fs:[124h]
840617d5 8a873a010000    mov     al,byte ptr [edi+13Ah]
840617db 8845e4          mov     byte ptr [ebp-1Ch],al
840617de 8b7514          mov     esi,dword ptr [ebp+14h]
840617e1 84c0            test    al,al
```

判斷方式

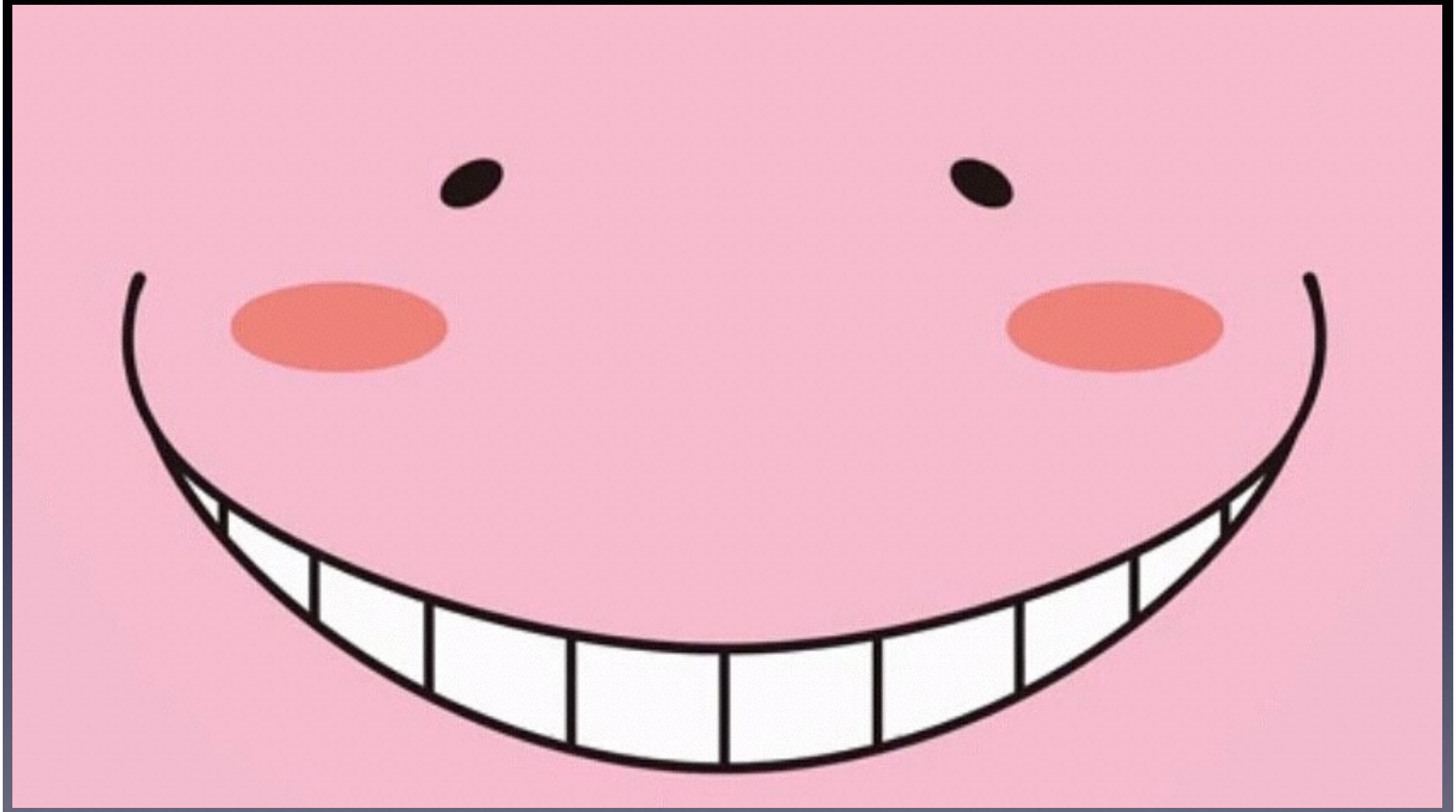
- 判斷是不是由遊戲本身或HS調用
- 如果是，就執行HS本身的HOOK
- 如果是由我們的程式調用
- 執行原本函數該執行的東西 A__A

怎麼判斷？

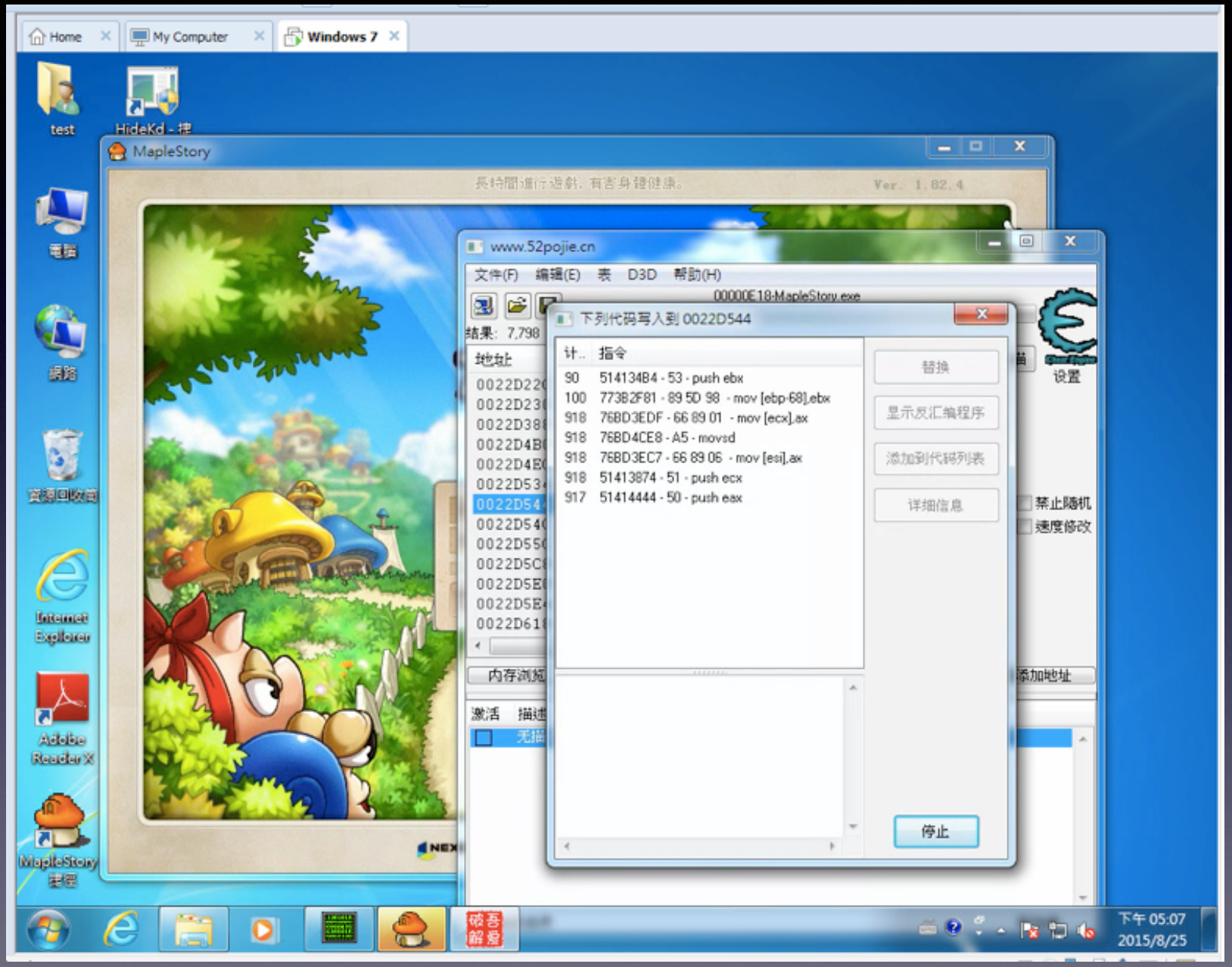
- 在調用時，有一個叫EPROCESS的結構體
- 該結構體的偏移0x16c(win7x86)是該處理程序名稱
- 可以從這邊判斷！

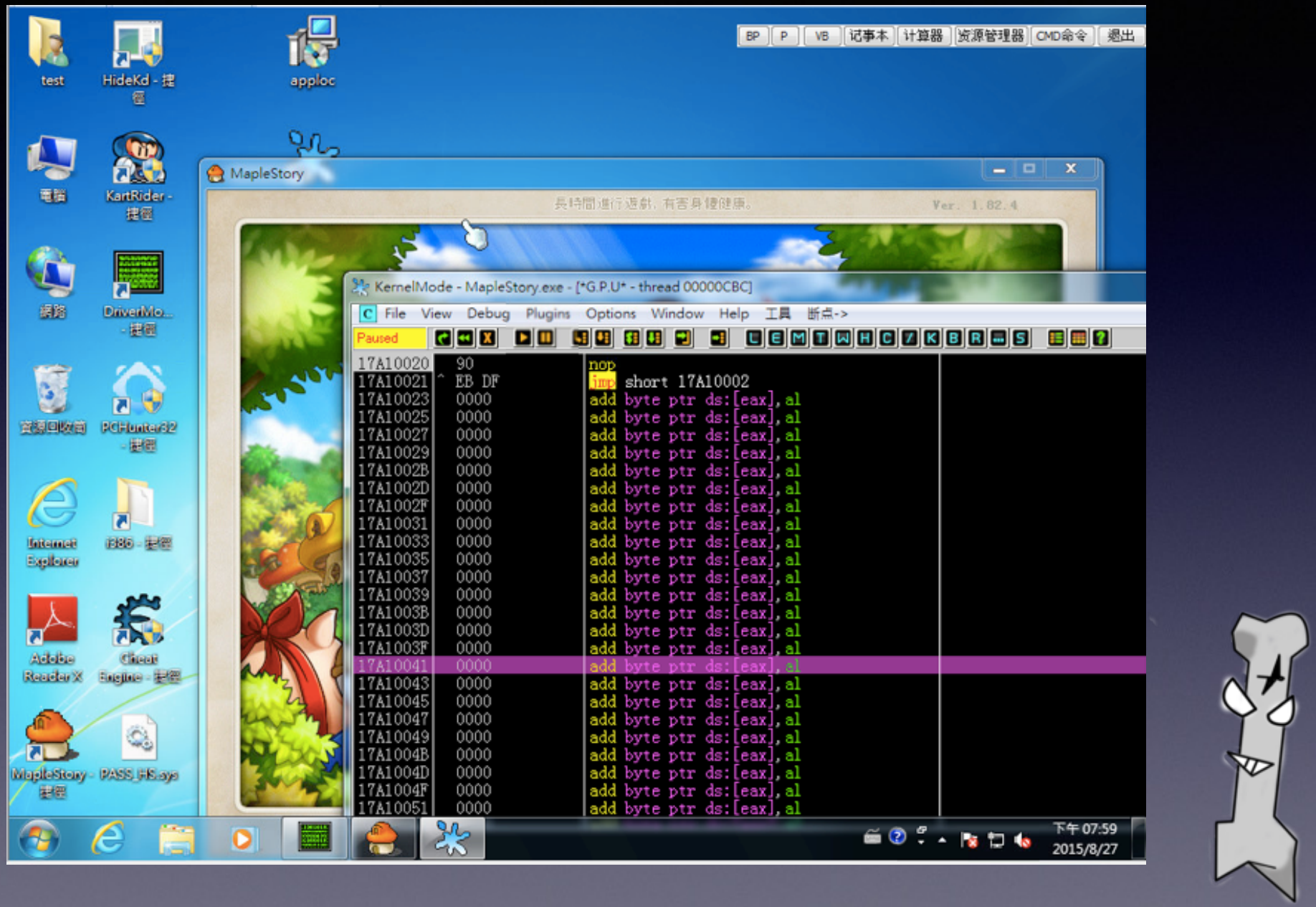
據說這方法從腿骨剛開始摸楓之谷

到現在都活著（燦笑



ByPass了以後能幹麻？


Home
My Computer
Windows 7
test
HideKd - 捷
MapleStory
長時間進行遊戲，有害身體健康。
Ver. 1.62.4
電腦
網路
資源回收筒
Internet Explorer
Adobe Reader X
MapleStory 捷徑
www.52pojie.cn
文件(F) 編輯(E) 表 D3D 幫助(H)
00000E18-MapleStory.exe
結果: 7,798
地址
0022D544
下列代码写入到 0022D544
计.. 指令
90 514134B4 - 53 - push ebx
100 773B2F81 - 89 5D 98 - mov [ebp-68],ebx
918 76BD3EDF - 66 89 01 - mov [ecx],ax
918 76BD4CE8 - A5 - movsd
918 76BD3EC7 - 66 89 06 - mov [esi],ax
918 51413874 - 51 - push ecx
917 51414444 - 50 - push eax
替换
显示反汇编程序
添加到代码列表
详细信息
停止
设置
禁止随机
速度修改
内存浏览
添加地址
激活
下午 05:07
2015/8/25


BP
P
VB
记事本
计算器
资源管理器
CMD命令
退出
test
HideKd - 捷徑
apploc
電腦
KartRider - 捷徑
網路
DriverMo... - 捷徑
資源回收筒
PCHunter32 - 捷徑
Internet Explorer
i386 - 捷徑
Adobe Reader X
Cheat Engine - 捷徑
MapleStory - 捷徑
PASS_HS.sys
MapleStory
長時間進行遊戲，有害身體健康。
Ver. 1.82.4
KernelMode - MapleStory.exe - [*G.P.U* - thread 00000CBC]
File View Debug Plugins Options Window Help 工具 断点->
Paused
17A10020 90 nop
17A10021 EB DF jmp short 17A10002
17A10023 0000 add byte ptr ds:[eax],al
17A10025 0000 add byte ptr ds:[eax],al
17A10027 0000 add byte ptr ds:[eax],al
17A10029 0000 add byte ptr ds:[eax],al
17A1002B 0000 add byte ptr ds:[eax],al
17A1002D 0000 add byte ptr ds:[eax],al
17A1002F 0000 add byte ptr ds:[eax],al
17A10031 0000 add byte ptr ds:[eax],al
17A10033 0000 add byte ptr ds:[eax],al
17A10035 0000 add byte ptr ds:[eax],al
17A10037 0000 add byte ptr ds:[eax],al
17A10039 0000 add byte ptr ds:[eax],al
17A1003B 0000 add byte ptr ds:[eax],al
17A1003D 0000 add byte ptr ds:[eax],al
17A1003F 0000 add byte ptr ds:[eax],al
17A10041 0000 add byte ptr ds:[eax],al
17A10043 0000 add byte ptr ds:[eax],al
17A10045 0000 add byte ptr ds:[eax],al
17A10047 0000 add byte ptr ds:[eax],al
17A10049 0000 add byte ptr ds:[eax],al
17A1004B 0000 add byte ptr ds:[eax],al
17A1004D 0000 add byte ptr ds:[eax],al
17A1004F 0000 add byte ptr ds:[eax],al
17A10051 0000 add byte ptr ds:[eax],al
下午 07:59
2015/8/27

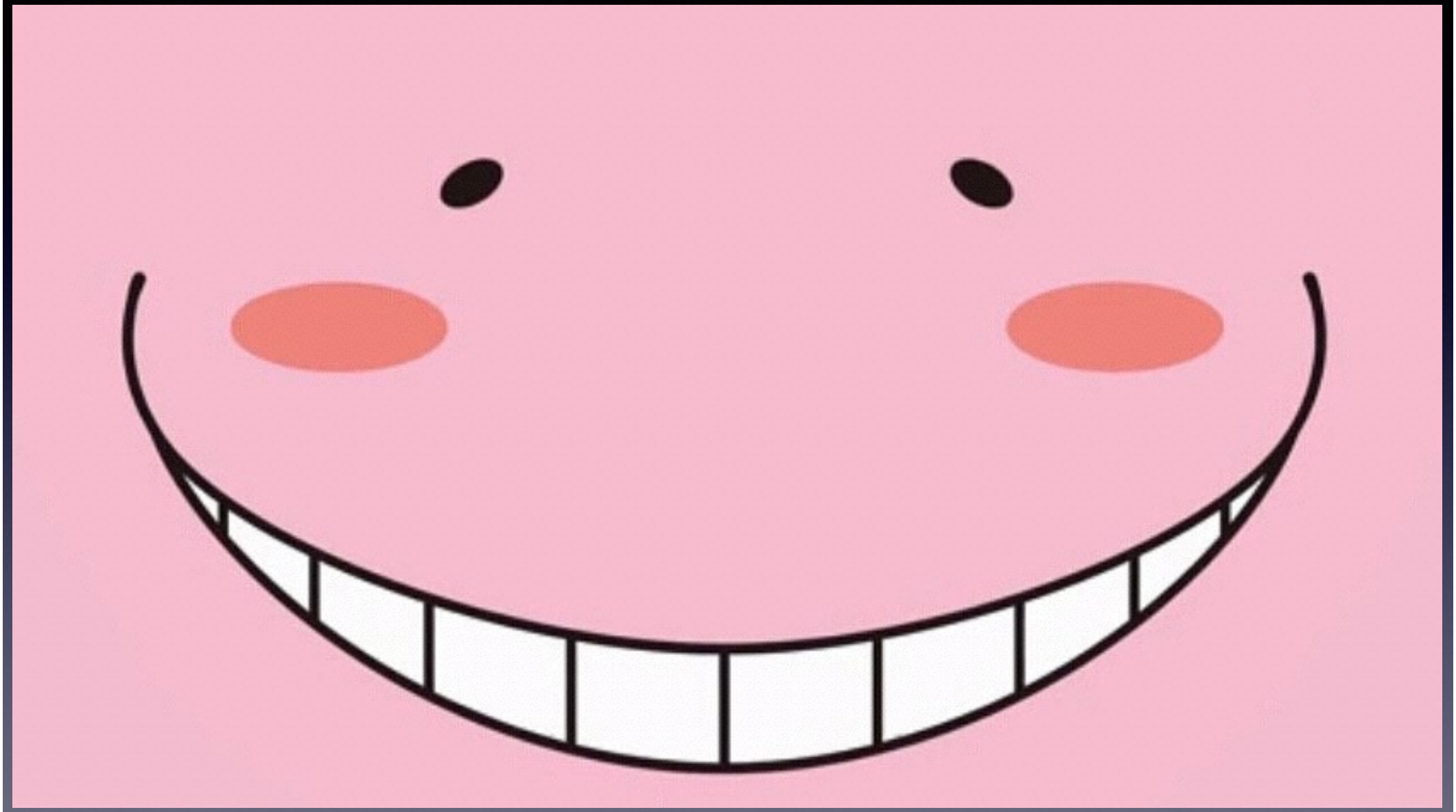
DEMO

BUT

應該算一個HS的0day吧(?)

但是只要遊戲的主程式呼叫有執行到紅匡處他就當作HOOK沒有被修改！

```
kd> u 86613ae8
*** ERROR: Module load completed but symbols could not be loaded for EagleXNt.sys
86613ae8 e993c00cfd      jmp     EagleXNt+0x18b80 (836dfb80)
86613aed 90              nop
86613aee 90              nop
86613aef 90              nop
86613af0 e91f34abfd      jmp     nt!PsOpenProcess (840c6f14)
86613af5 90              nop
86613af6 90              nop
86613af7 90              nop
```



除了上述的方法以外，還有像內核重載

- 重新加載一份內核ntkrplen.exe

除了上述的方法以外，還有像內核重載

- 重新加載一份內核ntkrplen.exe
- 對剛剛加載的內核作重定位

除了上述的方法以外，還有像內核重載

- 重新加載一份內核ntkrplen.exe
- 對剛剛加載的內核作重定位
- 對新內核的SSDT表重定位

除了上述的方法以外，還有像內核重載

- 重新加載一份內核ntkrplen.exe
- 對剛剛加載的內核作重定位
- 對新內核的SSDT表重定位
- 用剛剛講過的技術去 Hook KiFastCallEntry

除了上述的方法以外，還有像內核重載

- 重新加載一份內核ntkrplen.exe
- 對剛剛加載的內核作重定位
- 對新內核的SSDT表重定位
- 用剛剛講過的技術去 Hook KiFastCallEntry
- 收工！

實際資安上的應用

工商服務

工商服務

關於莫風

身高：比腿骨矮一點

體重：比腿骨瘦很多

號稱：大媽殺手

夢想：徵伴侶

條件：男女不拘老少通吃物種不限（我們支持各種形式的愛

http://莫風.isalways.one

Q & A