

# Windows Kernel Debugging

## 환경구성하기

### (WinXP+VMWare(Win2000))

처음 작성 : 2005. 11. 20

최종 작성 : 2005. 11. 20

김 경 곤 (A.K.A. Anesra)

[anesra@{null2root.org,a3sc.co.kr}](mailto:anesra@null2root.org)

# 목 차

1. Debuggee 설정
2. Debugger 설정
3. Kernel Debugging

# 1 Debuggee 환경 설정

## 커널 디버깅 환경 설정

- 두대의 컴퓨터 필요(Debugger, Debuggee)
- Null-modem cable 또는 1394 cable로 연결

## 디버깅(Debuggee)환경 설정

### VMWare 부팅한 후 boot.ini 파일 편집

boot.ini

[boot loader]

timeout=30

default=multi(0)disk(0)rdisk(0)partition(1)\WINNT

[operation systems]

multi(0)disk(0)rdisk(0)partition(1)\WINNT="Microsoft Windows 2000 Debug" /DEBUG

/debugport=com1 /baudrate=115200

multi(0)disk(0)rdisk(0)partition(1)\WINNT="Microsoft Windows 2000 Professional"

/fastdetect

## 2 Debugger 환경 설정

### 디버거 환경 설정

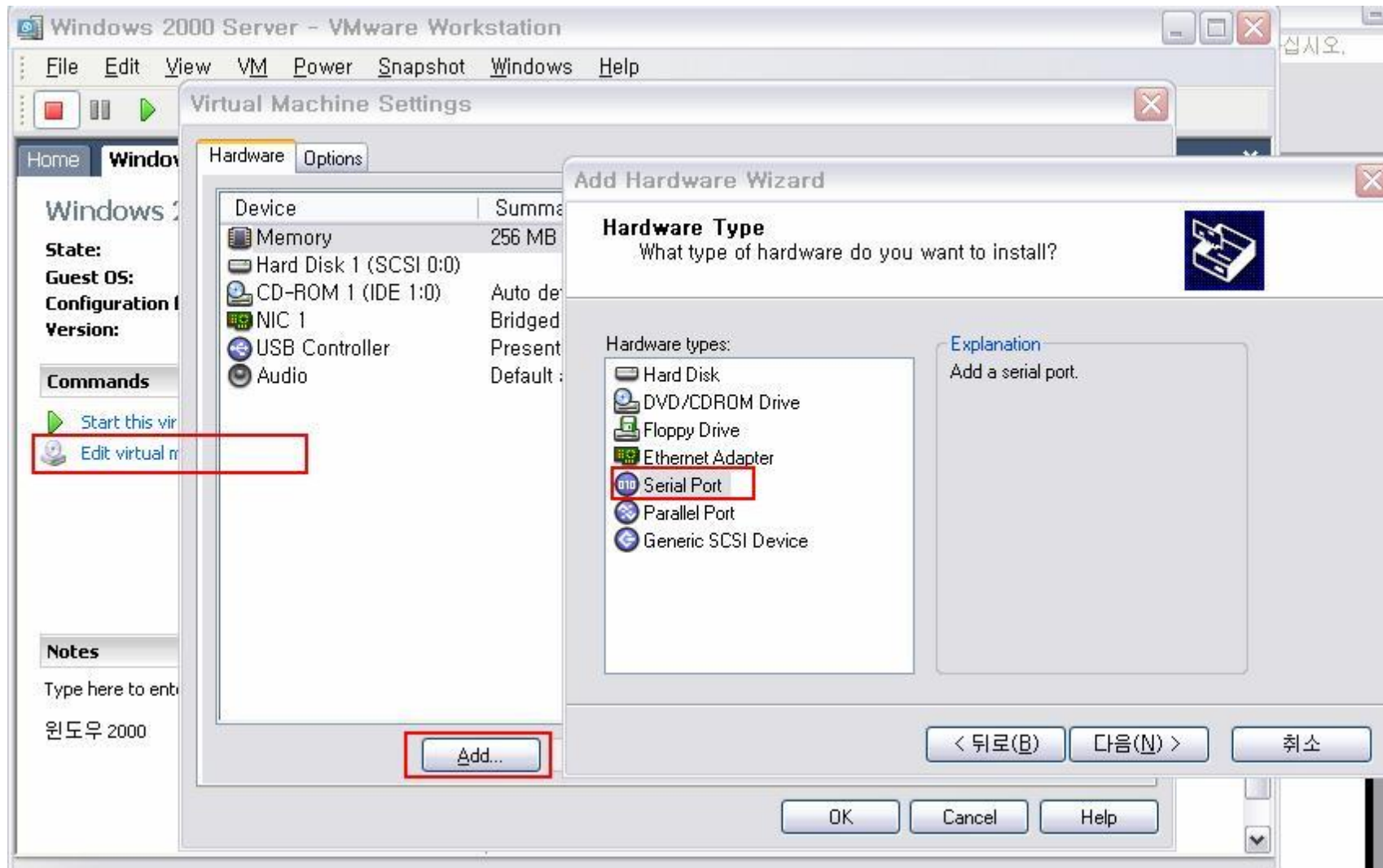
- WinDBG에서 File->Kernel Debug 선택

대상: windbg.exe -k com:pipe,port=\\.\pipe\xcom



## 2 Debugger 환경 설정 (VMware)

### VMWare에서 설정 (Serial Port 추가)



## 2 Debugger 환경 설정 (VMware)

### VMWare에서 Serial 연결



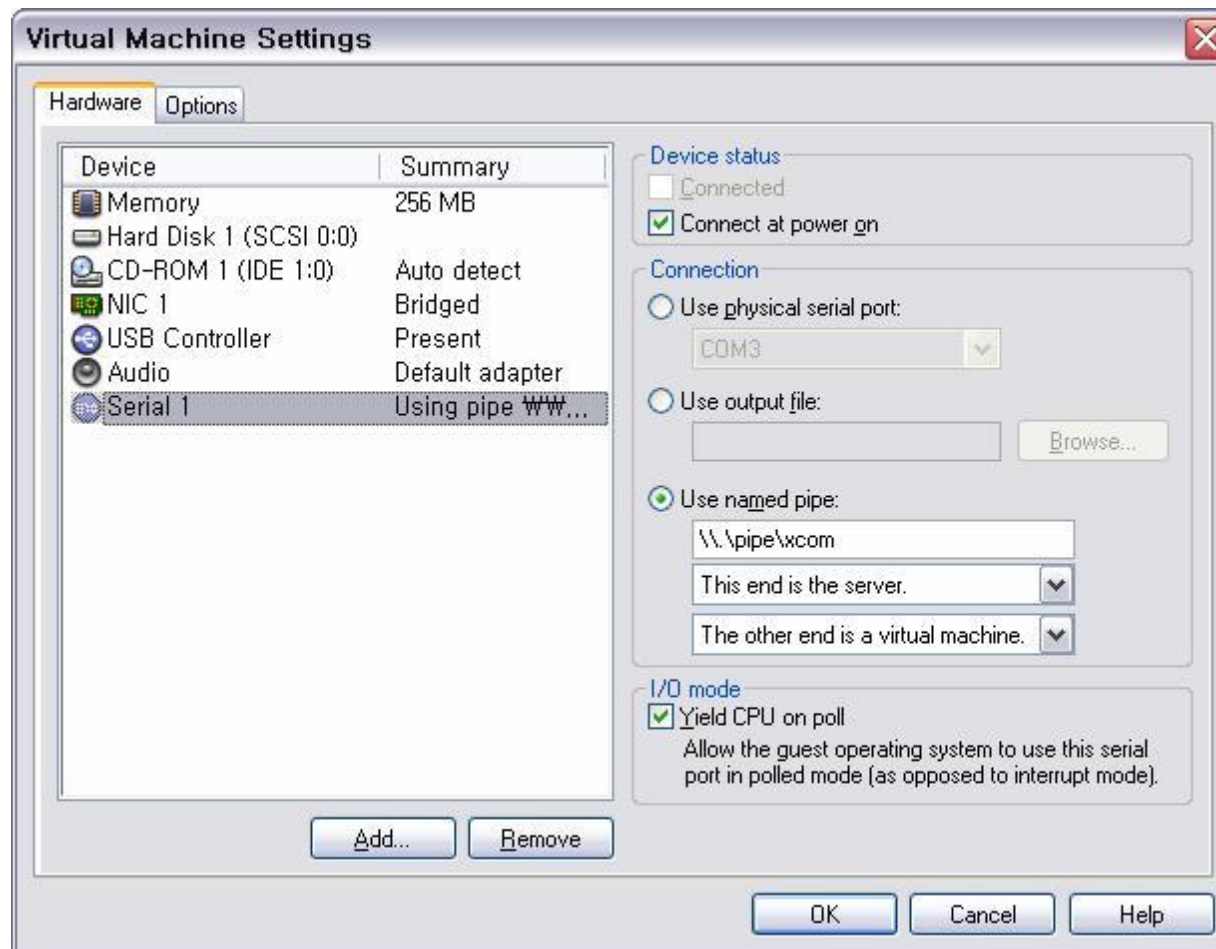
## 2 Debugger 환경 설정 (VMware)

### VMWare에서 Serial 연결



## 2 Debugger 환경 설정 (VMware)

### VMWare에서 Serial 연결





## 2 Debugger 환경 설정 (VMware)

### 심벌 파일(Symbol File) 설정

-WinDBG에서 모든 커널 관련 명령어 사용하기 위해서는 WinDBG에서 사용할 디버거(Debugee)의 심벌 파일을 지정해 주어야 함

WinDBG에서 File->Symbol File 심벌 패스 지정



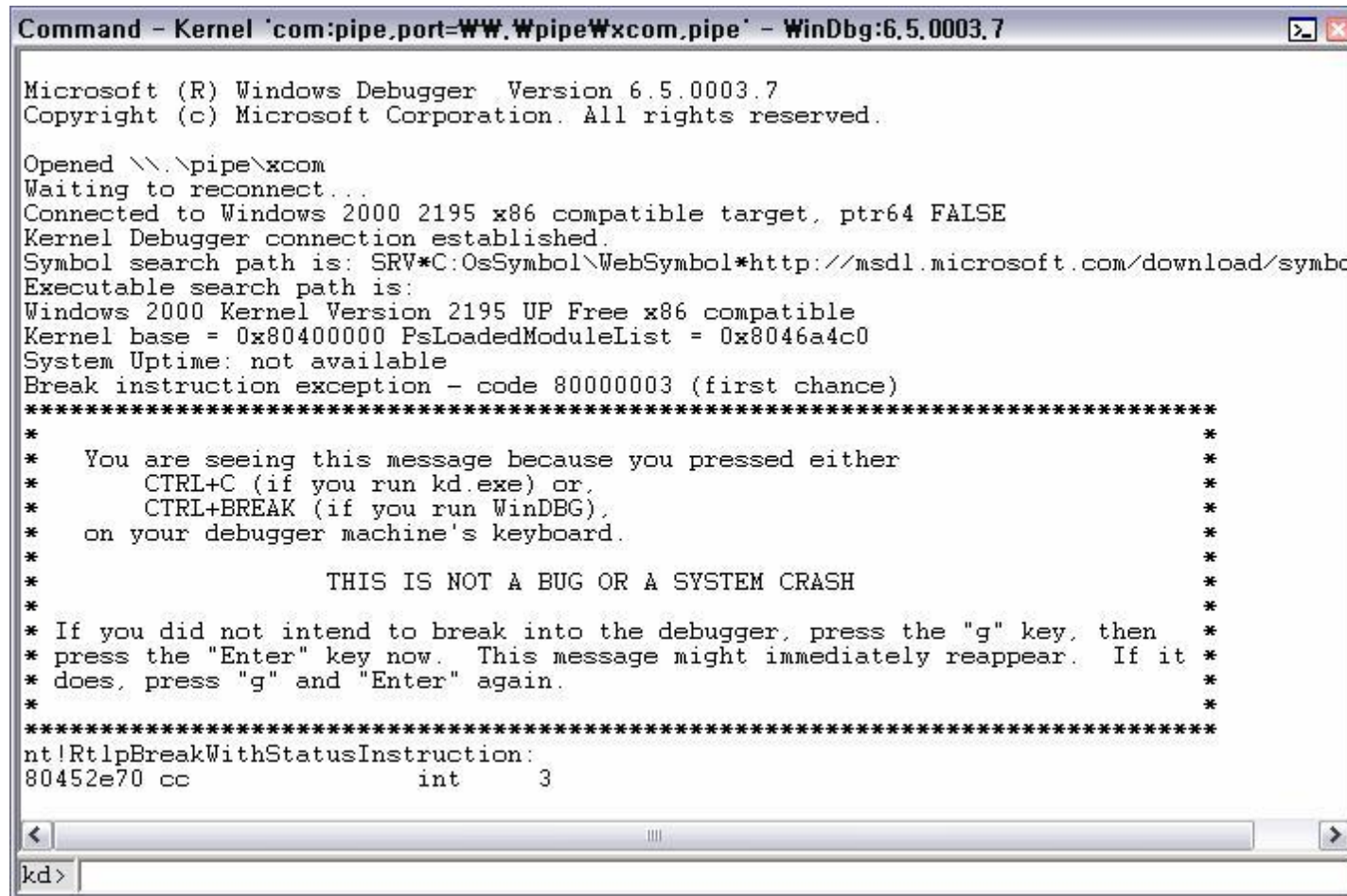
SRV\*C:\\WebSymbols\*http://msdl.microsoft.com/download/symbols  
C:\\WebSymbols 폴더 만들어 줘야함

## 2.1 WinDBG를 사용한 프로세스 구조체 살펴보기

### VMWare에서 커널 디버깅 성공

\*WinDBG에서 command창 눌러야 확인 가능(이거 안해서 안되는지 알았음;; )

\*Break걸어야지 대기에서 Connect함 , g는 디버깅 실행



```
Command - Kernel 'com:pipe,port=\\.\pipe\ww.wpipe\wxcom.pipe' - WinDbg:6.5.0003.7

Microsoft (R) Windows Debugger Version 6.5.0003.7
Copyright (c) Microsoft Corporation. All rights reserved.

Opened \\.\pipe\xcom
Waiting to reconnect...
Connected to Windows 2000 2195 x86 compatible target, ptr64 FALSE
Kernel Debugger connection established.
Symbol search path is: SRV*C:\OsSymbol\WebSymbol*http://msdl.microsoft.com/download/symbols
Executable search path is:
Windows 2000 Kernel Version 2195 UP Free x86 compatible
Kernel base = 0x80400000 PsLoadedModuleList = 0x8046a4c0
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:
80452e70 cc          int          3
kd>
```

# 2.1 WinDBG를 사용한 프로세스 구조체 살펴보기

## Kernel 디버깅에서 eprocess 보기

The screenshot shows the WinDBG kernel debugger interface with three windows. The left window displays the structure of an EPROCESS object, with fields like DirectoryTableBase[2] highlighted. The middle window shows the loaded modules list, with the 'lm' command used to view it. The right window shows the 'x ntdll!' command being used to examine the ntdll module's internal structure. A purple callout box on the right provides a legend for the commands used in the debugger.

```
kd> .load kdex2x86
kd> !strct eprocess
struct _EPROCESS (sizeof=648)
+000 struct _KPROCESS Pcb
+000 struct _DISPATCHER_HEADER
+000 byte Type
+001 byte Absolute
+002 byte Size
+003 byte Inserted
+004 int32 SignalState
+008 struct _LIST_ENTRY
+008 struct _LIST_ENTRY
+00c struct _LIST_ENTRY
+010 struct _LIST_ENTRY
+010 struct _LIST_ENTRY
+014 struct _LIST_ENTRY
+018 uint32 DirectoryTableBase[2]
+020 struct _KGDTENTRY LdtDescriptor
+020 uint16 LimitLow
+022 uint16 BaseLow
+024 union __unnamed
+024 struct __unnamed
+024 byte Base
+025 byte Flags
+026 byte Flags
+027 byte Base
...
kd> !strct eprocess
+27c uint32 VadPhysicalPages
+280 uint32 AweLock
kd> lm
start end module name
77f80000 77ff9000 ntdll (pdb symbols)
80062000 80075d20 hal (pdb symbols)
80400000 80590b40 nt # (pdb symbols)
a0000000 a01a5500 win32k (deferred)
be21a000 be22ecc0 ipsec (deferred)
be40f000 be433040 Fastfat (deferred)
be510000 be512f20 spud (deferred)
be54c000 be587260 srv (deferred)
be7b4000 be7ef000 udmpd (deferred)
kd> x ntdll!
77f9ce110 ntdll!Last64BitTickCount = <no type information>
77f936dc ntdll!RtlpValidateCurrentDirectory = <no type information>
77f9a1fd ntdll!ZwCreateEventPair = <no type information>
77f94139 ntdll!ZwSetSecurityObject = <no type information>
77fcd378 ntdll!WorkerThreadTimerQueue = <no type information>
77fcf2a0 ntdll!LdrpNumberOfDllTags = <no type information>
77f9522a ntdll!ZwAccessCheckByTypeAndAuditAlarm = <no type information>
77fc6449 ntdll!_eFCOM32 = <no type information>
77f96c59 ntdll!ZwOpenSemaphore = <no type information>
77fb288d ntdll!RtlpDebugPageHeapSetUserValue = <no type information>
77f81122 ntdll!NtCreateMailslotFile = <no type information>
77f822fa ntdll!RtlDelete = <no type information>
77f93587 ntdll!NtQueryAttributesFile = <no type information>
77f84562 ntdll!RtlpGetDefaultsSubjectContext = <no type information>
77f938e8 ntdll!ZwQueryInformationThread = <no type information>
77faf9ad ntdll!RtlpVerCompare = <no type information>
77fa8454 ntdll!RtlNumberOfSetBits = <no type information>
```

**구조체보기**  
kd>!strct 구조체명  
**로드된 모듈보기**  
kd>lm  
**모듈로드하기**  
kd>ld ntdll  
**모듈내심볼보기**  
kd>x ntdll(모듈명)

# 참고 자료

- VMWare에서 커널 디버깅 설정하기

[http://www.devpia.com/etc/msn/view\\_board.asp?forumnameforumtype=vc\\_lec@1@6854@2@](http://www.devpia.com/etc/msn/view_board.asp?forumnameforumtype=vc_lec@1@6854@2@)

[http://www.vmware.com/support/ws2/doc/serial\\_ws\\_win.html](http://www.vmware.com/support/ws2/doc/serial_ws_win.html)

[http://myhome.naver.com/andy\\_jung/action/h\\_list.php?id=andy\\_jung\\_5&work=list&st=&sw=&cp=4](http://myhome.naver.com/andy_jung/action/h_list.php?id=andy_jung_5&work=list&st=&sw=&cp=4)

[http://myhome.naver.com/andy\\_jung/action/download.php?id=andy\\_jung\\_1&nid=10696](http://myhome.naver.com/andy_jung/action/download.php?id=andy_jung_1&nid=10696)

<http://snoya.ye.ro/driver/windbg/windbg.html>



# Discussion