

DeviceDriver





Seoul National University of Technology
Dept. Computer Science & Engineering
EC 14th Hyung Jin, Yoo

Goal

- 주제 : 윈도우 디바이스 드라이버 맛보기
- 기간 : 2008년 7월 ~ 8월
- 시간 : 저녁 7시
- 교재
 - 주교재
 - The Windows 2000 Device Driver Book, Jerry Lozano
 - 부교재
 - O'Reilly – Windows NT File System Internals, Nagar



Setup List

- VirtualPC Setup 
- VirtualPC Images(XPSP2) 
- Windwos Driver Kit(WDK) Setup 
- Visual C++ Setup
- WinDBG Setup 

Day-1

- 주제
 - 드라이버 개발 환경 구성
 - WinDBG를 이용한 유저모드 디버깅
 - WinDBG를 이용한 커널모드 디버깅
 - 초 간단 드라이버 제작



Environment

- 드라이버 개발환경을 구성해보자
 - VirtualPC

- 가상머신을 구현한 S/W로써 보통 디버깅환경을 구성한다는 것은 2대의 PC간의 물리적인 연결방법 (Serial, 1394, USB port)으로 인해서 연결 해야하지만 가상머신을 이용하여 컴퓨터 1대에서 동일한 작업을 할 수 있게 해준다.



Environment

– Windows Driver Kit(WDK)

- 윈도우 디바이스드라이버를 컴파일 하기 위한 툴킷
Vista 이전 버전에서는 DDK(Device Driver Kit)라는 이름이었지만 Vista에서부터는 WDK라는 이름으로 바뀌어 나오게 되었다.
- 추가된 사항
 - WDF Model, IFS Kit, 검증 툴, 배포 기능
 - Windows Logo Kit(WLK), etc...
- 자세한 사항은 이곳을 참고한다. 

Environment

– WinDBG

- 윈도우 커널 및 디바이스 드라이버 그리고, 애플리케이션을 디버깅할 수 있는 디버거(Debugger)이다.
- Microsoft에서 무료로 제공하고 있는 강력한 윈도우 커널 디버깅을 제공한다.
- WinDBG는 아쉽게도 Windows 9x(95/98/ME)는 지원하지 않고 있다. 그래도 디버깅을 하고 싶다면 Soft-ICE라는 디버깅 툴을 이용하면 된다.

What DeviceDriver

- 디바이스 드라이버란 무엇인가?
 - DeviceDriver
 - 장치(Device)를 사용할 수 있게 장치 정보를 가지고 있는 S/W라고 할 수 있으며, 이는 Windows와 밀접한 관계를 가지고 있다.
 - 예를 들자면 각종 장치드라이버들이 있는데, 그 중에 우리가 포맷을 했다고 가정하면 OS 설치 후 가장 먼저 해야 할 일은 각각의 드라이버 설치과정이다. 인터넷을 쓰고 싶어도(장치가 있어도) Lan Driver가 설치되어 있지 않다면 그에 해당하는 정보가 없기에 인식하지 않아 작동하지 않는다.

Debugging

- 디버깅에 대해서 알아보자
 - Debug
 - 프로그래밍을 해본 사람이라면 Debug라는 부분을 모를리는 없을 것이다. 간단히 표현하면 코드의 잘못된 부분을 찾아 고치는 작업을 일컫는 말이다.
 - 이 Debug의 유래는 초창기 시절에는 지금처럼 언어를 통한 프로그래밍이 아니라 일일이 하나씩 선을 연결하는 방식으로 작업을 했다. 그 과정에 있어서 프로그램이 제대로 작동이 되지 않아 원인을 찾던 중 기계 사이에 벌레(bug)가 들어가 있었다. 그래서 그때부터 Debug라는 말이 나오게 되었다.

Debugging

– Debugger

- Debug작업을 행하게 될 때 앞에서 가상머신을 소개한 부분에서 보면 2대의 컴퓨터를 연결하여 디버깅을 하는 작업에 있어서 주체가 되는 컴퓨터를 지칭하여 Debugger라고 한다.

– Debuggee

- 위 부분과 반대로 Debug라는 작업의 목적이 되는 컴퓨터를 지칭하여 Debuggee라고 한다.

– 작업 관점에서 따라 Debugger, Debuggee가 될수 있다.

Connect


- 컴퓨터 간 연결은 어떻게 해야 할까?
 - Serial Cable
 - 시리얼 케이블은 널모뎀 케이블이라고도 한다.
 - 2대의 PC 후면에 보면 시리얼포트가 내장되어 있는데 이곳에 서로 연결을 하면 된다.
 - 속도가 느리다는 단점이 있다.
 - 1394 & USB Cable
 - 1394 케이블 같은 경우는 다양한 기능을 가지고 있고, 속도 또한 빠르다. 별도의 장치를 필요로 하지 않다. 하지만 USB는 PC간 직접 연결할 수 없다.

Connect Setting

- 케이블 연결 후 설정은 어떻게 할까?
 - Serial Cable
 - Debugger - WinDBG 실행
 - File ⇒ Kernel Debug
 - baudrate = 115200, port = com1 으로 설정
 - pipe = 0, reconnect = 0, resets = 0 으로 설정
 - Debuggee - 디버그 모드로 설정
 - boot.ini 파일 수정
 - [operating systems] 밑에 내용 복사 후 아래 내용 추가
 - /debug /debugport = com1/ baudrate = 115200
 - debugport와 baudrate는 Debugger 설정과 동일해야 함


Connect Setting

– 1394 Cable

- 1394 케이블을 이용한 방법은 Windows XP, Server 2003 버전에서만 이용할 수 있다.
- Debugger
 - 제어판 ⇒ 네트워크 설정 ⇒ 1394 Connection 여부확인
 - 1394 속성 ⇒ TCP/IP 선택
 - IP입력 ⇒ IP : 192.168.0.1 // MASK : 255.255.255.0
- Debuggee
 - 위 과정과 동일하며 IP만 재입력 IP : 192.168.0.2
- 자세한 사항은 이곳을 참고한다. 

Connect Setting

– Virtual PC

- Serial Cable 설정 방법과 거의 유사하다.
- Debugger - WinDBG 실행
 - File ⇒ Kernel Debug
 - baudrate = 115200, port = \\.\pipe\com1 으로 설정
 - pipe = 1, reconnect = 1, resets = 1 으로 설정
 - pipe란? 간단히 process간에 전송통로, 참고 
- Debuggee – 디버그 모드로 설정
 - boot.ini 파일 수정
 - [operating systems] 밑에 내용 복사 후 아래 내용 추가
 - /debug /debugport = com1/ baudrate = 115200


Debugging to User Mode

- 유저모드 디버깅은 뭐 하는 걸까?
 - User Mode
 - 윈도우라는 운영체제 내에서는 많은 애플리케이션들(IE, MS office, Winamp...)이 실행되는 곳을 유저가 접근할 수 있는 OS영역내에서 실행되는 모드, 유저모드라고 하며 Windows에서는 권한 Ring3
 - User Mode Debugging
 - 유저모드 디버깅이란 바로 이런 애플리케이션을 디버깅하는 작업을 지칭하는 말이다.

Debugging to Kernel Mode

- 커널모드 디버깅은 뭐 하는 걸까?
 - Kernel Mode
 - 커널 모드는 모든 시스템의 메모리와 모든 CPU처리에 액세스 할 수 있는 실행 모드이다.
 - Kernel Mode Debugging
 - 운영체제 코드(시스템 서비스나 장치 드라이버)를 디버깅하는 작업을 지칭하는 말이다.

Symbol

- 심볼은 무엇인가 알아보자.
 - 심볼은 우리가 디버깅 하는데 있어서 좀더 원활하게 편하게 접근할 수 있게 해준다.
 - .pdb라는 파일에 아래와 같은 파일이 있다.
 - 전역변수, 지역변수, 함수 이름과 EntryPoint의주소
 - FPO Data, Source-line번호
 - 자세한 사항은 이곳을 참고한다. 

WinDBG

- WinDBG 기본 명령어에 대해서 알아보자.
 - WinDBG 명령어 타입 3가지
 - Normal Commands – ex) bp
 - Meta Commands – ex) .reload
 - Extension Commands – ex) !process
 - 명령어는 크게 내장, 외장명령어로 나뉜다.
 - 내장명령어(Normal, Meta), 외장명령어(Extension)
 - 외장명령어 같은 경우 외부 dll로 구성되어 있다.
 - 사용자가 추가할 수 도 있으며 기본적으로 몇가지는 제공 되어진다.

WinDBG

– Break Point

- bp – break point를 설정한다.
- bl – break point 리스트를 출력한다.
- bc – break point를 삭제한다.
- bd - break point를 사용 안 함으로 설정한다.
- be - break point를 사용함으로 설정한다.

– Process

- !process 0 0 – debugee의 모든 프로세스 목록을 출력한다.

WinDBG

– Symbol

- .reload /f – 시스템의 모든 심볼을 로드한다.
- .reload /f kernel32.dll – kernel32 심볼을 로드한다.
- .reload /u – 시스템의 모든 심볼을 언로드한다.
- .reload /u kernel32.dll – kernel32 심볼을 언로드
- .reload /l – 현재 로드된 모듈 리스트를 출력한다.

– Module

- lm k – Kernel Mode 모듈을 표시한다.
- lm u – User Mode 모듈을 표시한다.
- lm m – 패턴을 검사하여 해당하는 것만 보여준다.

WinDBG

– Dump

- db – byte단위로 메모리 내용을 덤프한다.
- dw – word단위로 메모리 내용을 덤프한다.
- dd – dword단위로 메모리 내용을 덤프한다.
- eb, ew, ed – 메모리 내용을 수정한다.

– 기타

- g – go
- t – step info
- p – step over
- u – code to disassembly

WinDBG

– Stack trace

- k – 현재 콜 스택을 출력한다.
- kv – 지역변수 값과 함께 콜 스택을 출력한다.
- kvn – 지역변수, 스택프레임 번호와 함께 출력한다

Make Simple Driver

- 초 간단 드라이버를 만들어보자.
 - 드라이버 기본 구성 파일
 - hello.c - 소스 파일
 - sources - 빌드 환경 설정 파일
 - makefile - 메이크 파일
 - 드라이버 빌드 방법
 - WDK ⇒ Build Environment ⇒ Windows2000 ⇒ Checked Build Environment 선택
 - 명령어 : build -cwgZ makefile
 - hello.sys파일 생성

Make Simple Driver

- 실습을 해보자.
 - sources파일
 - TARGETNAME=hello
 - TARGETTYPE=DRIVER
 - SOURCES=sample1.c
 - makefile파일
 - 그다지 수정할 것이 없으므로 기존 WDK내 예제들 것을 사용해도 된다.
 - OSR Loader를 이용하여 드라이버를 로드

PostScript

- 오늘 1일차로 온통 처음 배우는 부분들이기에 이해가 되지 않은 부분이 더 많았고, 워낙 많은 것을 배웠기에 실습한 내용의 중간 중간 생각이 나서 다시 실습을 하려 하니 생각이 잘 나지 않아서 애를 먹고 있다. 현재 부족한 부분인 실습부분을 더 해보고 남은 부분을 채워 놓도록 하겠다.