

Olly Debugger 사용 방법 강좌 1부

2005.4.14

지금부터 올리 디버거에 대해 간단하게 설명해 보겠습니다.

현재까지의 올리 디버거 최신 버전은 1.10이구요.

1.1 버전대의 BETA 버전에 계속 나오다가 최근이 FINAL 버전이 배포되었습니다.

1.1대에서의 FINAL 버전이라는 말입니다.

올리 디버거는 <http://home.t-online.de/home/Ollydbg/>에서 무료로 배포하고 있습니다.

올리 디버거는 잘 아시다시피 바이너리 파일 분석에 사용되는 도구이구요.
주요 기능을 살펴보면 다음과 같습니다.

- 기계어 DISASSEMBLE 기능
- CPU 레지스터 추적 기능
- 실행 중인 프로세스 디버깅 기능
- API, C 함수 분석 기능

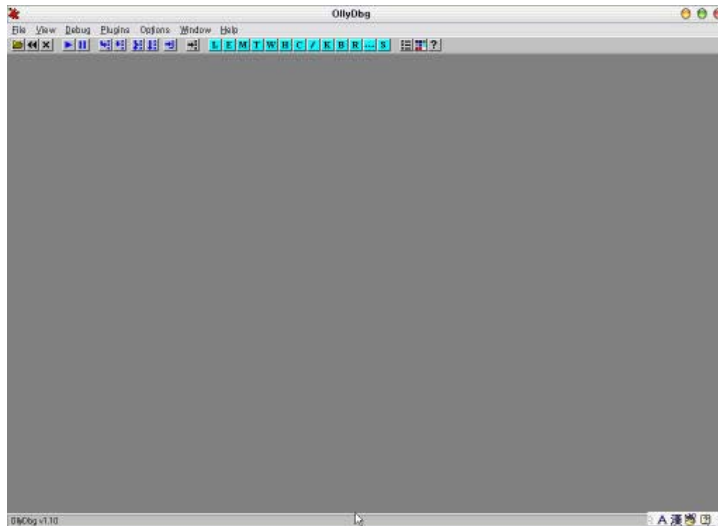
올리 디버거는 제가 지금까지 사용해 보았던 디버거들 중 가장 좋은 것 같구요.

유일하게 불편했던 점은 프로그램 내의 문자열 추적 기능이 비슷한 도구인 W32DASM에 비해 뒤떨어진다는 것인데 아쉽게도 이번 업데이트 버전에도 수정은 되어 있지 않았습니다.

이 문서에서 설명할 내용은 다음과 같습니다.

- 가) 올리 디버거의 인터페이스 설명
- 나) 각 메뉴 설명
- 다) 간단한 사용 실습 (2부)

이제 올리 디버거를 실행시켜 봅시다. 그럼 다음과 같은 화면을 보게 됩니다.



[그림 1]

위쪽에 보이는 주 메뉴들의 역할은 다음과 같습니다.

FILE : 디버깅할 파일 혹은 프로세스를 지정합니다.

View : 디버깅 대상에 대한 각종 정보를 출력합니다.

Debug : 디버깅에 관련된 기능들입니다.

Plugins : 디버깅에 유용한 각종 유틸리티를 플러그인 형태로 제공합니다.

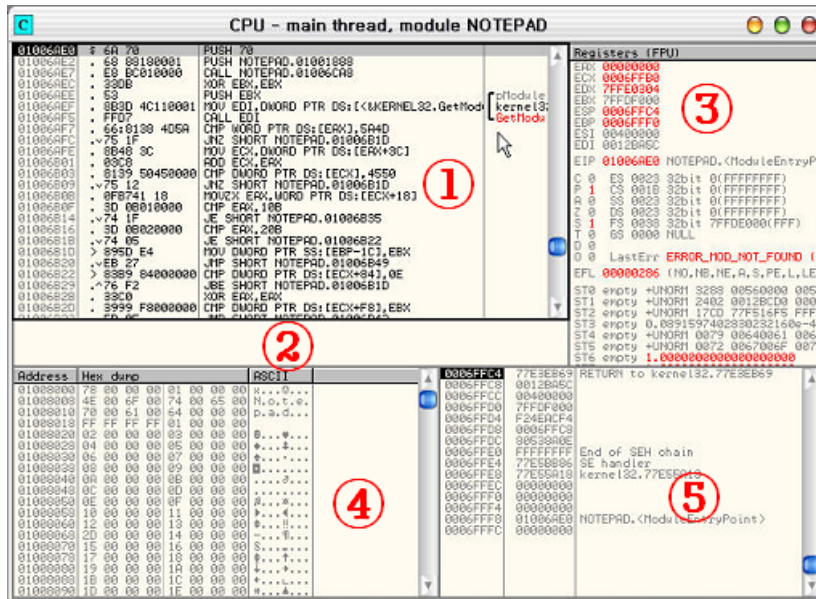
필요한 기능을 개인이 제작하여 배포하는 경우도 많으며, 다음의 주소에서 다운받아 사용할 수 있습니다.

<http://ollydbg.win32asmcommunity.net/stuph/>

Options : 각종 옵션을 변경합니다.

Window, Help : 화면(창) 설정과 도움말입니다.

이제 파일 하나를 불러와 보겠습니다. 불러오려는 파일은 NOTEPAD.EXE입니다. 그럼 다음과 같은 윈도우가 출력됩니다.

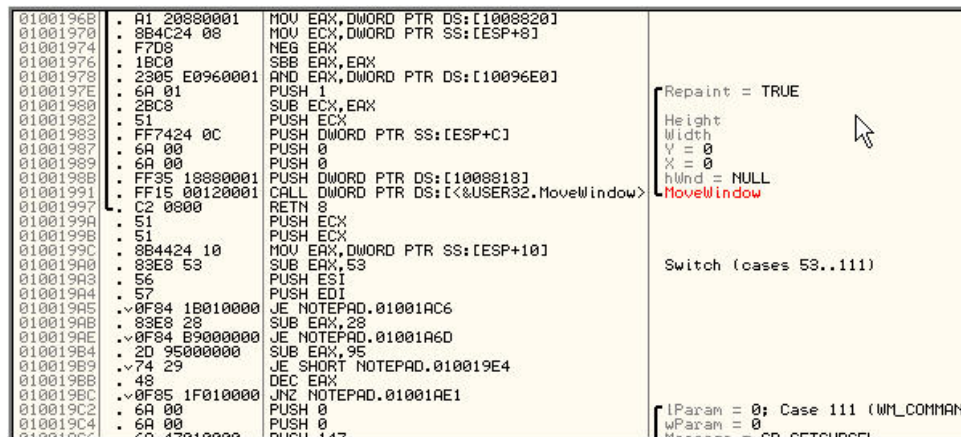


[그림2]

위 윈도우는 총 5개의 영역으로 구분할 수 있고요. 이를 설명하면 다음과 같습니다.

- ① 디버깅 대상의 Code 영역에 대한 Disassemble 정보
- ② 선택된 라인에 대한 추가 정보
- ③ 현 상태에서의 레지스터 정보
- ④ 선택된 주소에 대한 HEX와 ASCII 코드 정보
- ⑤ 현 상태에서의 스택 정보

①번 윈도우는 다시 네 개로 구분할 수 있으며, 의미는 다음과 같습니다.



[그림3]

- ① CODE의 주소
- ② 기계어
- ③ 어셈블리어
- ④ 관련 정보

그럼, 현재 보이고 있는 화면은 무엇을 의미할까요?

이는 NOTEPAD.EXE가 실행 된 후, 가장 처음 실행해야할 코드(ENTRY POINT)에 자동으로 BREAK가 걸린 것입니다.

이제 알아야할 것들은 오른쪽 버튼을 클릭했을 때 나오는 메뉴들입니다. 앞서 설명한 5개의 창에서 마우스 오른쪽 버튼을 누르면 또 다른 많은 메뉴들을 볼 수 있는데, 이것은 위쪽의 메인 메뉴와는 다른 것들입니다.

이제 이에 대한 설명을 드리겠습니다. 먼저 1번 윈도우에 대한 메뉴입니다.



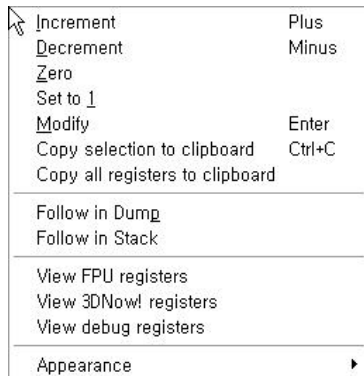
[그림4]

많은 메뉴가 있는데요. 주요 메뉴만 설명하면 다음과 같습니다.

- Backup : 현재의 정보를 백업하거나 복구합니다.
- Copy : 선택된 영역을 복사합니다.
- Binary : 선택된 라인의 기계어를 수정합니다.
- Assemble : 선택된 라인의 어셈블리어를 수정합니다.
- Label, Comment : 해당 라인에 주석 정보를 입력합니다.
- BreakPoint : BreakPoint를 설정합니다.
- Go To : 주요 지정 위치로 이동.
- Follow in Dump : 선택된 라인의 HEX 정보를 윈도우4에 표시
- Search For : 각종 정보 검색
- Find references to : 레퍼런스 정보 출력
- View : 디버깅 할 모듈 변경
- Copy to Executable : 선택된 내용을 파일로 저장
- Appearance : 윈도우 모양 변경

다음 2번 윈도우엔 윈도우 모양을 변경하는 Appearance 메뉴만 있습니다.

3번 레지스터 윈도우의 메뉴는 다음과 같습니다.



[그림5]

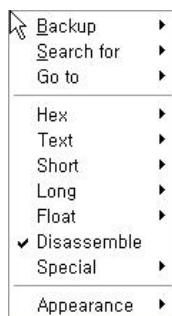
Increment, Decrement, Zero, Set To 1, Modify : 모두 레지스터 값 변경 과 관련된 메뉴입니다.

Copy ... : 선택된 영역 복사

Follow ... : 선택된 영역에 대한 내용을 4번 혹은 5번 윈도우에 출력

Appearance : 윈도우 모양 변경

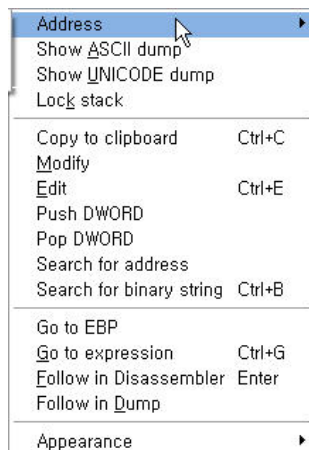
다음은 4번 윈도우의 메뉴에 대한 설명입니다.



[그림6]

Hex, Text ~ Disassemble : 출력 방식을 변경합니다.

마지막으로 5번 스택인 윈도우의 메뉴에 대한 설명입니다.



[그림7]

Address : 스택 주소의 표시 방식을 선택합니다. (절대주소, 상대주소)
 Show ... : ASCII 혹은 UNICODE 정보를 출력합니다.
 Modify, Edit : 스택 정보를 수정합니다.
 Push, Pop : 스택에 값을 넣거나 뺍니다.
 Search ... : 특정 값을 검색합니다.

이상으로 각 윈도우 창에서의 메뉴에 대한 설명을 마칩니다.
 다음은 왼쪽 View 메뉴에 있는 유용한 기능들을 알아보겠습니다.

View - Executable modules : 모듈 정보 출력

Base	Size	Entry	Name	File version	Path
01000000	00013000	01006AE0	NOTEPAD	5.1.2600.0 (xpc	C:\WINDOWS\notepad.exe
62340000	00008000	62342C82	LPK	5.1.2600.0 (xpc	C:\WINDOWS\System32\LPK.DLL
71950000	000E4000	7195F225	COMCTL32	6.0 (xpclient.0	C:\WINDOWS\WinSxS\x86_Microsoft.Windows.Common-Cont
72F00000	00050000	72F249BC	USP10	1.0407.2600.0 (C:\WINDOWS\System32\USP10.dll
72F50000	00023000	72F516FA	WINSPOOL	5.1.2600.0 (XPC	C:\WINDOWS\System32\WINSPOOL.DRV
762E0000	00010000	762E1270	IMM32	5.1.2600.0 (xpc	C:\WINDOWS\System32\IMM32.DLL
76300000	00045000	76301600	comdlg32	6.00.2600.0000	C:\WINDOWS\system32\comdlg32.dll
77230000	00064000	7729E4D3	SHLWAPI	6.00.2750.167 (C:\WINDOWS\system32\SHLWAPI.dll
77330000	00774000	773A2014	SHELL32	6.00.2600.0000	C:\WINDOWS\system32\SHELL32.dll
77BC0000	00053000	77BCE94F	msvcrt	7.0.2600.0 (xpc	C:\WINDOWS\system32\msvcrt.dll
77C20000	00030000		GDI32	5.1.2600.132 (x	C:\WINDOWS\system32\GDI32.dll
77CF0000	00030000	77CF514B	USER32	5.1.2600.0 (xpc	C:\WINDOWS\system32\USER32.dll
77D00000	00090000	77D81CFB	ADVAPI32	5.1.2600.0 (XPC	C:\WINDOWS\system32\ADVAPI32.dll
77E20000	0011F000	77E3A241	kernel32	5.1.2600.0 (xpc	C:\WINDOWS\system32\kernel32.dll
77F50000	000A9000		ntdll	5.1.2600.0 (xpc	C:\WINDOWS\System32\ntdll.dll
78000000	0006F000	7800739A	RPCRT4	5.1.2600.135 (x	C:\WINDOWS\system32\RPCRT4.dll

[그림8]

View - Memory : 메모리 정보 출력

Address	Size	Owner	Section	Contains	Type	Access	Initial	Mapped as
00010000	00001000				Priv	RW	RW	
00020000	00001000				Priv	RW	RW	
0005E000	00001000			stack of ma	Priv	RW	Gua: RW	
0005F000	00011000				Priv	RW	Gua: RW	
00070000	00001000				Map	R	R	
00080000	00002000				Map	R	R	
00090000	00005000				Priv	RW	RW	
00190000	00005000				Priv	RW	RW	
001A0000	00001000				Map	RW	RW	
001B0000	00016000				Map	R	R	
001D0000	00034000				Map	R	R	
00210000	00041000				Map	R	R	
00260000	00006000				Map	R	R	
00270000	00041000				Map	R	R	
002C0000	00004000				Priv	RW	RW	
002D0000	00003000				Map	R	R	
002E0000	00006000				Map	R	R	E
003A0000	00002000				Map	R	R	E
003B0000	00103000				Map	R	R	
004C0000	00008000				Priv	RW	RW	
004D0000	000B1000				Map	R	R	E
007D0000	00001000				Priv	RW	RW	
007E0000	00001000				Priv	RW	RW	
007F0000	00002000				Map	R	R	
00800000	00002000				Map	R	R	
00810000	00003000				Priv	RW	RW	
01000000	00001000	NOTEPAD		PE header	Inag	R	RWE	
01001000	00007000	NOTEPAD	.text	code, import	Inag	R	RWE	
01008000	00002000	NOTEPAD	.data	data	Inag	R	RWE	
0100A000	00009000	NOTEPAD	.rsrc	resources	Inag	R	RWE	
62340000	00001000	LPK		PE header	Inag	R	RWE	
62341000	00004000	LPK	.text	code, import	Inag	R	RWE	
62345000	00001000	LPK	.data	data	Inag	R	RWE	

[그림9]

View - Thread : 스레드 정보 출력

Ident	Entry	Data block	Last error	Status	Priority	User time
00000210	01006AE0	7FFDE000	ERROR_MOD_NOT_FOU	Active	32 + 0	0.0000 s

[그림 10]

View - Handles : 핸들 정보 출력

Handle	Type	Refs	Access	T	Info	Name
0000002C	Desktop	2116	000F01FF			\Default
00000008	Directory	57	00000003			\KnownDlls
00000014	Directory	29	000F000F			\Windows
00000020	Event	3	001F0003			
0000000C	File (dir)	2	00100020			c:\WINDOWS
0000001C	File (dir)	2	00100020			c:\WINDOWS\WinSxS*86_Microsoft.Windows.Common-Com
00000040	File (dir)	2	00100020			c:\WINDOWS\WinSxS*86_Microsoft.Windows.Common-Com
00000034	Key	2	000F003F			HKEY_LOCAL_MACHINE
0000003C	Key	2	000F003F			HKEY_CURRENT_USER
00000004	KeyedEvent	27	00000003			\Kerberos\Objects\CritSecOutOfMemoryEvent
00000010	Mutant	27	00000001			\NlsCacheMutant
00000018	Port	3	001F0001			
00000024	Section	26	000F001F			
00000028	WindowStation	58	000F037F			\Windows\WindowStations\WinSta0
00000030	WindowStation	58	000F037F			\Windows\WindowStations\WinSta0

[그림 11]

View - BreakPoints : 설정된 브레이크 포인트 정보 출력

Address	Module	Active	Disassembly
010069E8	NOTEPAD	Always	JNZ SHORT NOTEPAD.01006A33
010069F0	NOTEPAD	Always	JNZ SHORT NOTEPAD.01006A33
010069F8	NOTEPAD	Always	PUSH EAX
010069FE	NOTEPAD	Always	MOV ESI,DWORD PTR DS:[&KERNEL32.Lo
01006A04	NOTEPAD	Always	TEST EAX,EAX

[그림 12]

View - Source : 소스 코드 정보 출력

View - File : 특정 파일의 HEX, ASCII 정보 출력

View - Text : 특정 파일의 텍스트 정보 출력

메뉴와 관련된 설명은 여기서 마치도록 하겠습니다.
이처럼 올리 디버거엔 강력하고 다양한 기능들이 포함되어 있습니다.
이 올리 디버거 하나만 가지고도 윈도우즈 파일에 대한 이해와 분석을
충분히 할 수 있을 거라 생각합니다.

특히 View - Memory의 정보는 윈도우즈의 실행 파일이 차지하는
가상 메모리에 대한 구조를 개괄적으로 나타내주기 때문에 PE 파일
포맷 구조, 또 리눅스에서와의 차이점 이해에 큰 도움이 됩니다.

그럼 이번 강좌는 여기서 마치도록 하겠습니다.