그림 그리는 개발자

메모리 덤프(Memory dump) 분석하기 1

Language/MFC 2014.02.12 10:17
릴리즈후 테스트를 하는데 아침마다 프로그램이 죽어있는 현상으로 골머리를 썩히는 중...같이 일하시는 수석연구원님 께서 윈도우 이벤트 로그와 메모리 덤프 파일을 이용하는 방법을 찾아 주셨다. 아래 글을 참조하면 릴리즈 모드에서도 디버그가 어렵지 않을거 같은.-_-!!

1. "포스트모템 디버깅"과 "메모리 덤프"
"포스트 모템"이라는 말은 "사후(死後)"라는 의미입니다. 사후 세계를 믿는 종교를 "포스트모템 신앙"이라고 하죠.

"포스트모템 디버깅 (Post Mortem Debugging)" 이라는 말도 대략 비슷한 의미입니다. 디버거가 설치되어 있고 개발환경이 꾸며져 있는 PC에서 문제가 발생한다면야 별 걱정할 게 없겠지만... 그렇지 않다면 문제가 발생한 PC에서 "메모리 덤프"를 작성하여 분석가능한 개발PC로 가져와서 덤프분석을 수행해야 합니다. 이런 작업을 "포스트모템 디버깅"이라고 합니다.

대략은 아래와 같은 순서로 진행됩니다.

1. 문제가 발생하는 PC에 Just-In-Time Debugger를 등록 (관련된 내용은 여기를 참조)
2. 오류창이 발생하는 현상을 재현하면 JIT Debugger로 등록한 디버거가 실행되면서 문제 프로세스를 자동으로 Attach한다.
3. 실행된 디버거를 사용하여 메모리 덤프를 작성한다.
4. 디버거를 개발환경이 갖추어진 PC로 가져가 분석한다.

메모리 덤프란 프로세스의 메모리를 정해진 덤프 포맷에 따라 기록한 파일입니다. 커널메모리를 덤프뜨면 "커널 메모리덤프", Application 메모리를 덤프뜨면 "Application 메모리덤프"가 됩니다. ^^
이 덤프파일을 디버거로 분석하면, (Full Dump인 경우) 오류가 발생한 시점에 해당 프로세스에 Debugger를 Attach한 것과 같은 수준의 분석이 가능합니다.

이 포스트에서는...
"메모리덤프 분석은 커널드라이버 개발자들이나 하는 거다. Application 디버깅에는 해당 안되는 얘기일 거야"
라고 생각하시는 분들이나,
"WinDbg 같은건 너무 어려워서 난 몰라"
라고 생각하는 분들을 위해 메모리덤프 작성하는 방법과 WinDbg를 이용해 간단하게 덤프 분석하는 방법을 다루고자 합니다.
절대 고급 기술을 다루고자 함이 아니며, 저는 그럴 능력도 안되오니 고수분께서는 그냥 조용히 나가주시면 감사하겠습니다. ^^

이번 포스트에서는 "메모리덤프 작성하는 방법"을 주로 다루고자 합니다. 작성된 메모리덤프 분석하는 방법은 다음 번 포스트에서 다루겠습니다.

2. Dr.Watson을 이용한 메모리 덤프
Dr.Watson은 XP 이전까지의 Windows에 기본적으로 탑재되어 있는 디버거(?)로서, 다음과 같은 간단한 기능을 가지고 있습니다.

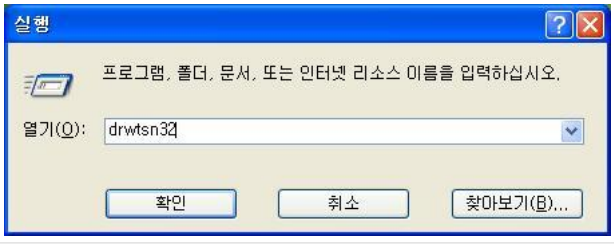
- Just-In-Time 디버거로 등록되어, 시스템의 프로세스 중에서 오류가 발생할 경우 오류가 발생한 주소와 오류코드, 콜스택 등을 파일로그로 기록
- 오류가 발생한 프로세스의 메모리덤프를 작성

Dr. Watson으로 메모리덤프를 작성하는 경우의 장점은 다음과 같습니다.

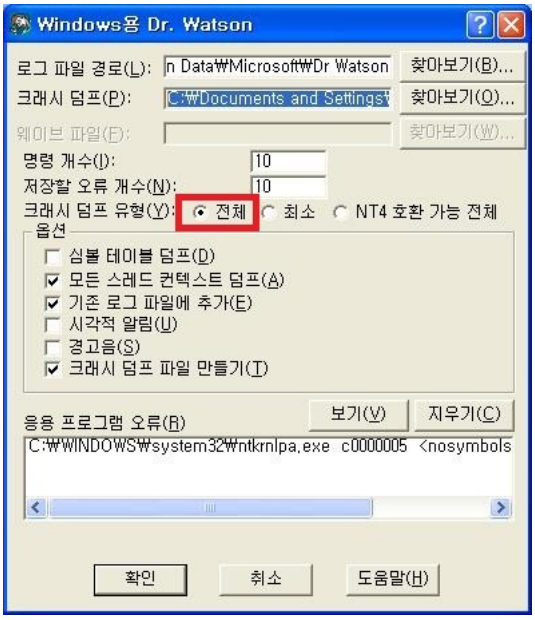
- 오류가 발생한 경우, 메모리 덤프를 얻기 위한 방법으로는 가장 간단한 방법임.
- 특히 Windows XP 이전의 OS인 경우 Windows에 기본으로 포함되어 있으므로 별도의 설치가 필요 없음.
- Dr.Watson은 오류 발생시 자동으로 덤프를 작성하기 때문에 svchost.exe 처럼 중요 시스템프로세스에서 오류가 발생하는 경우에도 덤프를 뜰 수 있습니다. 다른 디버거를 사용하면, 디버거가 시스템 프로세스를 Attach하면 그 Process가 멈추면서 시스템이 정상적으로 동작하지 않기 때문에 Remote Debugging 등 귀찮은 방법을 좀 동원해야 합니다.

사용법은 다음과 같습니다.

1) Dr.Watson에 메모리덤프 설정
– 먼저 시작 > 실행 창에서 "drwtsn32" 명령을 실행하여 Dr.Watson을 실행

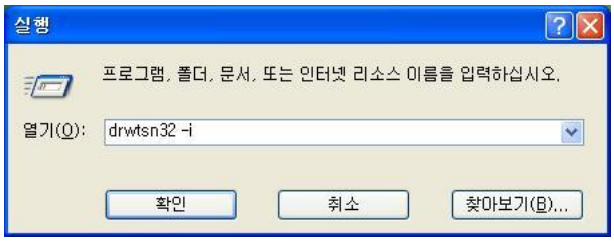


- 설정 창이 뜨면 "크래시 덤프 유형" 을 "전체"로 선택. (이때 크래시 덤프 생성 경로도 확인해둡니다.)

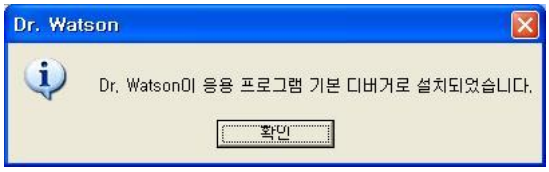


2) Dr.Watson을 기본디버거로 등록하기

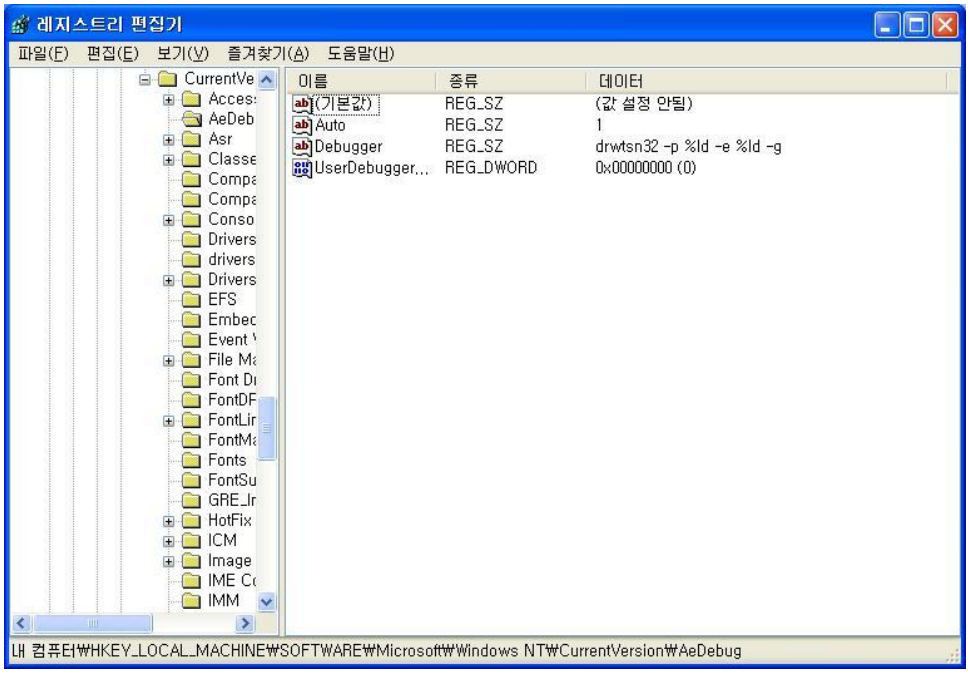
- 시작 > 실행 창에서 "drwtsn32 -i" 명령을 실행하여 기본디버거로 등록합니다.



- 디버거 등록이 완료되면 아래와 같이 창이 뜹니다.



- 기본디버거가 등록되면 "HKEY_LOCAL_MACHINE₩SOFTWARE₩Microsoft₩Windows NT₩CurrentVersion₩AeDebug" 레지스트리에 아래와 같은 값들이 등록됩니다. (원상복구를 원하신다면 이 키의 "Debugger" Value를 삭제하시면 됩니다.)

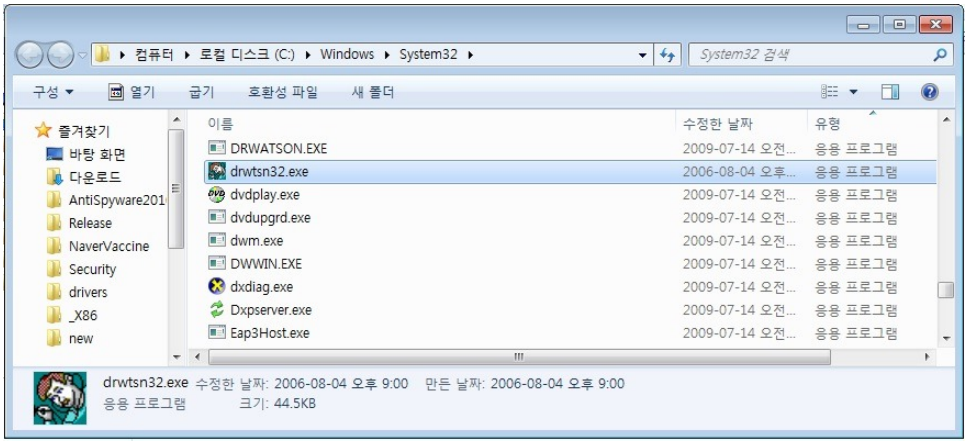


3)오류 상황을 재현하기

- 기본디버거가 등록된 후 오류가 발생하면 아래와 같이 로그와 덤프파일이 생성됩니다. "전체덤프" 설정일 때, 덤프파일은 해당 프로세스가 사용하는 물리메모리 크기와 같은 사이즈로 생성됩니다. 따라서 좀 사이즈가 클 수도 있습니다.

- 이 덤프파일(user.dmp)을 개발환경이 갖춰진 PC에 가져가서 분석하면 됩니다. 만약 덤프파일 분석이 곤란한 상황이라면 아쉬운대로 로그파일을 이용할 수도 있는데요, 이 경우 map 파일이나 cod 파일등을 사용해 분석하면 오류가 발생한 위치와 그때의 CallStack 등을 확인할 수 있습니다. (관련 내용은 여기를 참고)

- Windows Vista 이후에는 Dr.Watson이 설치되어 있지 않기 때문에 WinDbg 등 다른 방법을 이용해야 합니다. 하지만, Windows XP의 System32 폴더에서 drwtsn32.exe 파일을 복사해온다면 Vista 이후 OS에서도 Dr.Watson을 사용할 수 있습니다.



drwtsn32.exe

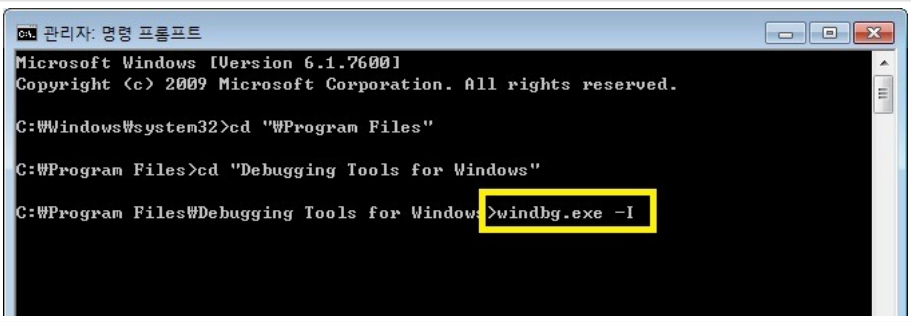
2. WinDbg 를 사용하는 방법
꼭 Dr.Watson이 아니더라도 메모리 덤프 기능이 있는 디버거를 기본 디버거로 등록만 한다면 오류 발생시 메모리덤프를 뜰 수 있습니다. 여기서는 WinDbg의 예를 들어 설명합니다.
WinDbg로 메모리덤프를 작성하는 경우, Dr.Watson과 비교했을 때 다음과 같은 장점들이 있습니다.

- WinDbg는 비교적 설치가 간단한 가벼운 디버거입니다. 따라서 Dr.Watson이 탑재되지 않은 Vista나 Win7 등에서 간단하게 다운로드 하여 사용할 수 있습니다.
- 오류가 발생하지 않는 경우 Dr.Watson은 메모리 덤프를 작성할 수 없지만, WinDbg는 디버거의 기능을 모두 갖추고 있기 때문에 오류가 발생하지 않는 경우에도 메모리 덤프 분석이 가능합니다. 예를 들어, Hang이 걸린 프로세스가 있을 때, WinDbg를 설치한 후 해당 Process를 Attach하면 간단하게 메모리덤프를 작성할 수 있습니다.

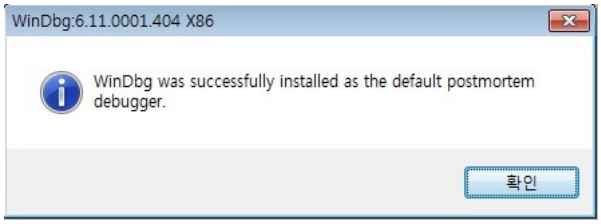
WinDbg를 이용해 덤프를 작성하는 방법은 다음과 같습니다.

1) 문제가 재현되는 PC에 WinDbg를 다운로드하여 설치 : 이부분은 설명을 생략합니다.

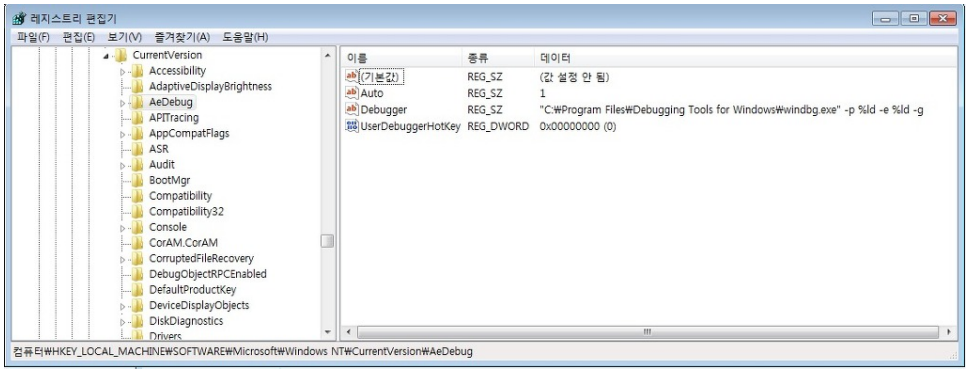
2) WinDbg를 기본 디버거로 등록
- 관리자 권한의 커맨드 쉘을 실행해 WinDbg 설치 위치로 이동한 후 아래와 같이 "windbg.exe -I" 명령을 실행해주면 됩니다.



– 정상적으로 기본디버거 등록이 완료되면 아래와 같은 메시지 창이 뜹니다.



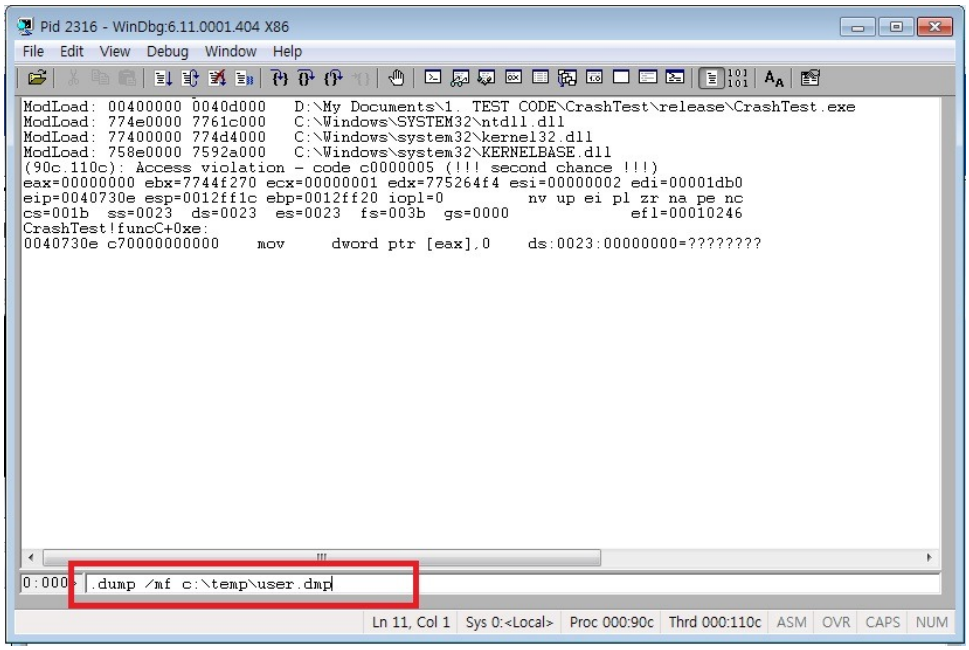
– 기본디버거가 등록되면 역시 "AeDebugger" 레지스트리 키에 "Debugger" Value에 디버거 실행커맨드가 등록됩니다. 기본디버거 설정을 해제하고 싶으면 이 Value를 삭제하면 됩니다.

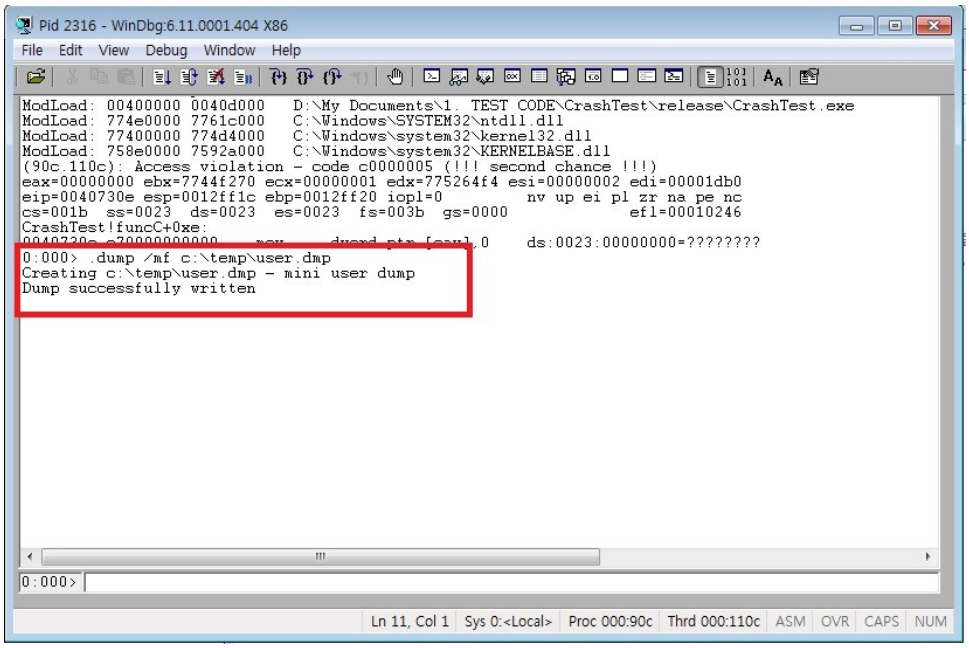


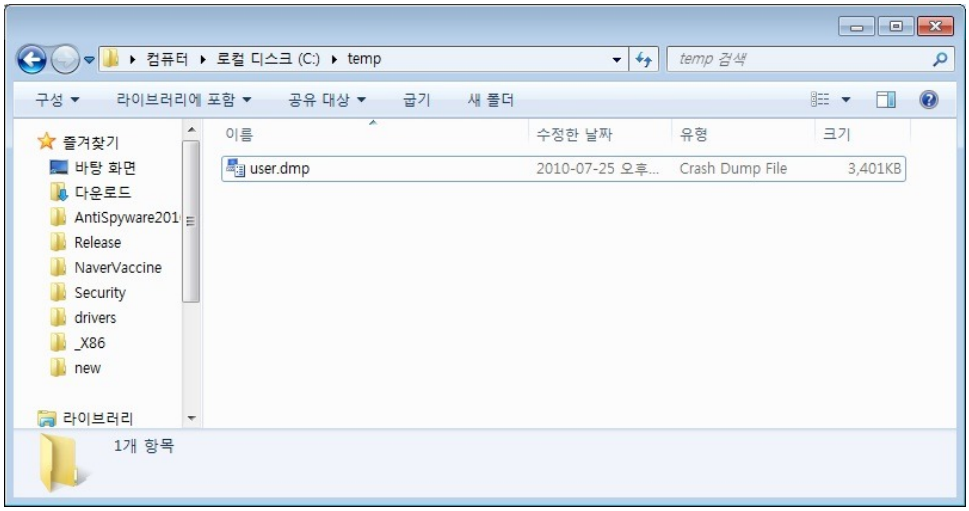
3) 문제 현상을 재현하면 자동으로 WinDbg가 실행됨

4) 실행된 WinDbg 에서 아래와 같은 명령을 실행하여 덤프를 생성

```
.dump /ma c:₩temp₩user.dmp
```

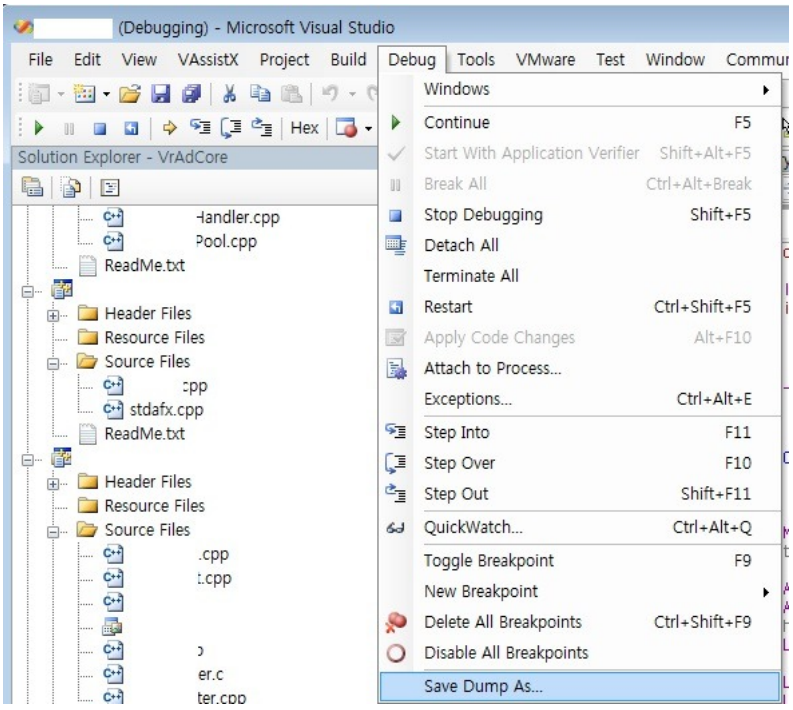




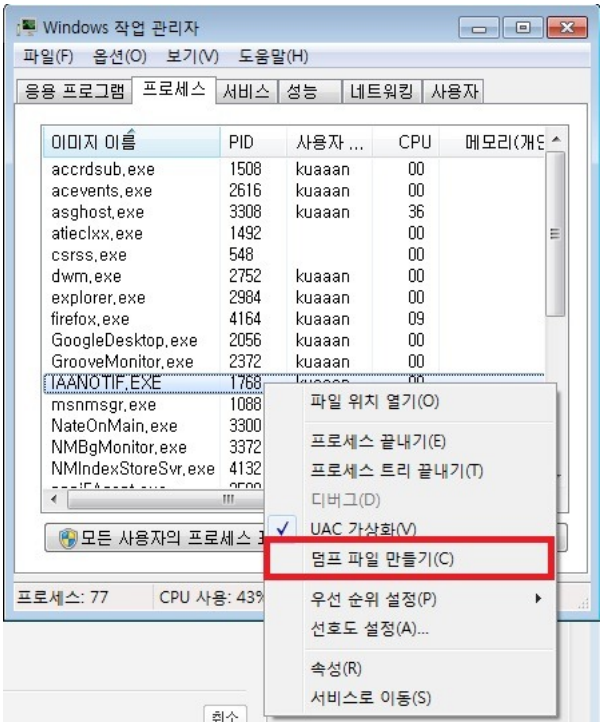


3. 기타 메모리덤프 생성 방법
사실 메모리덤프 생성하는 방법은 이 외에도 많습니다.

VisualStudio로 디버깅 중에도 메모리 덤프를 뜰 수 있구요...



Vista 이후 OS에서는 작업관리자에서도 덤프를 작성할 수 있습니다.



이 외에도 Olly 등 웬만한 디버거에는 메모리덤프 기능이 있지요. 어떤 방법을 이용해도 마찬가지구요. 해당 상황에서 적합한 방법을 사용하시면 되겠습니다.

출처 :(kuaaan) http://kuaaan.tistory.com/213

공감

Posted by MAKUBEX
TAG Debug, dr.watson, log남기기, MEMORY DUMP, memrory dump debug, windbg, windebug, 닥터왓슨, 메모리 덤프 트랙백1, 댓글0
이전 1 … 5 6 7 8 9 10 11 12 13 … 158 다음