IDA Remote Debugging

ASM Love
어셈블리어 개발자그룹
[어셈러브] asmlove.co.kr

2007. 01.
이강석 / certlab@gmail.com

어셈블리어 개발자 그룹 :: 어셈러브
http://www.asmlove.co.kr

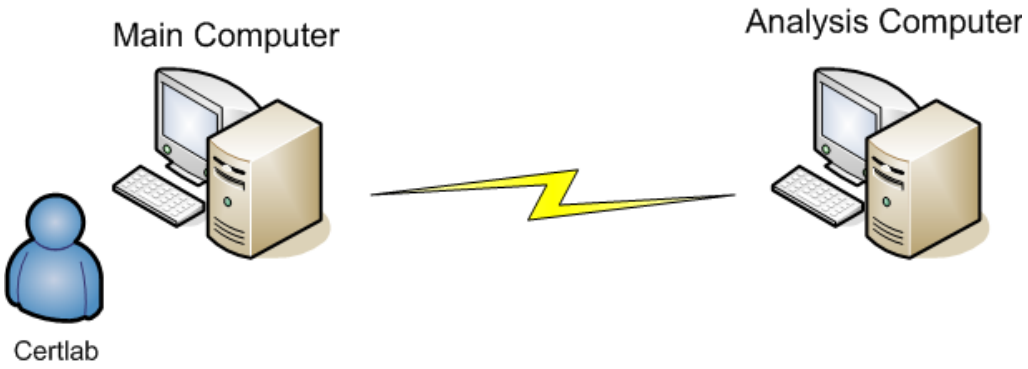
Intro

IDA Remote debugging에 대해 알아봅시다.
이런 기능이 있다는것을 잘 모르시는 분들을 위해 문서를 만들었습니다.
IDA 기능중에 분석할 파일을 원격에서 디버깅할수 있는 기능이 있는데 먼저 그림과 함께 예를 들어 설명해 보도록 하겠습니다.

Certlab은 바이러스와 같은 위험한 파일을 분석 할때 VMWare 안에서 IDA로 분석을 합니다.
쓰고 있는 Main Computer에서 분석을 하게되면 디버깅을 하다가 잘못해서 Main Computer에 감염이 되는 불안감과 그런경험이 있기 때문이죠.

디어셈블만 한다면 감염걱정은 없는데 Main Computer의 넓직한 모니터에 사양좋은 컴퓨터에서 IDA로 디버깅을 하고싶다고 소원을 빌게 됩니다..

그랬더니 꿈속에서 아이다가 나와 Remote Debugging 이라는 방법이 있다고 말해줍니다.
아이다 : Remote Debugging을 하면 디어셈블과 디버깅까지 Main Computer에서 할수 있고,
만약 디버깅과정에서 위험한 파일이 분석 도중 실행이 된다고 해도 설정해 놓은 분석컴퓨터에서 실행이 되요. 또한, 원격에 있는 Process를 attach시켜 디버깅할수 있구요.

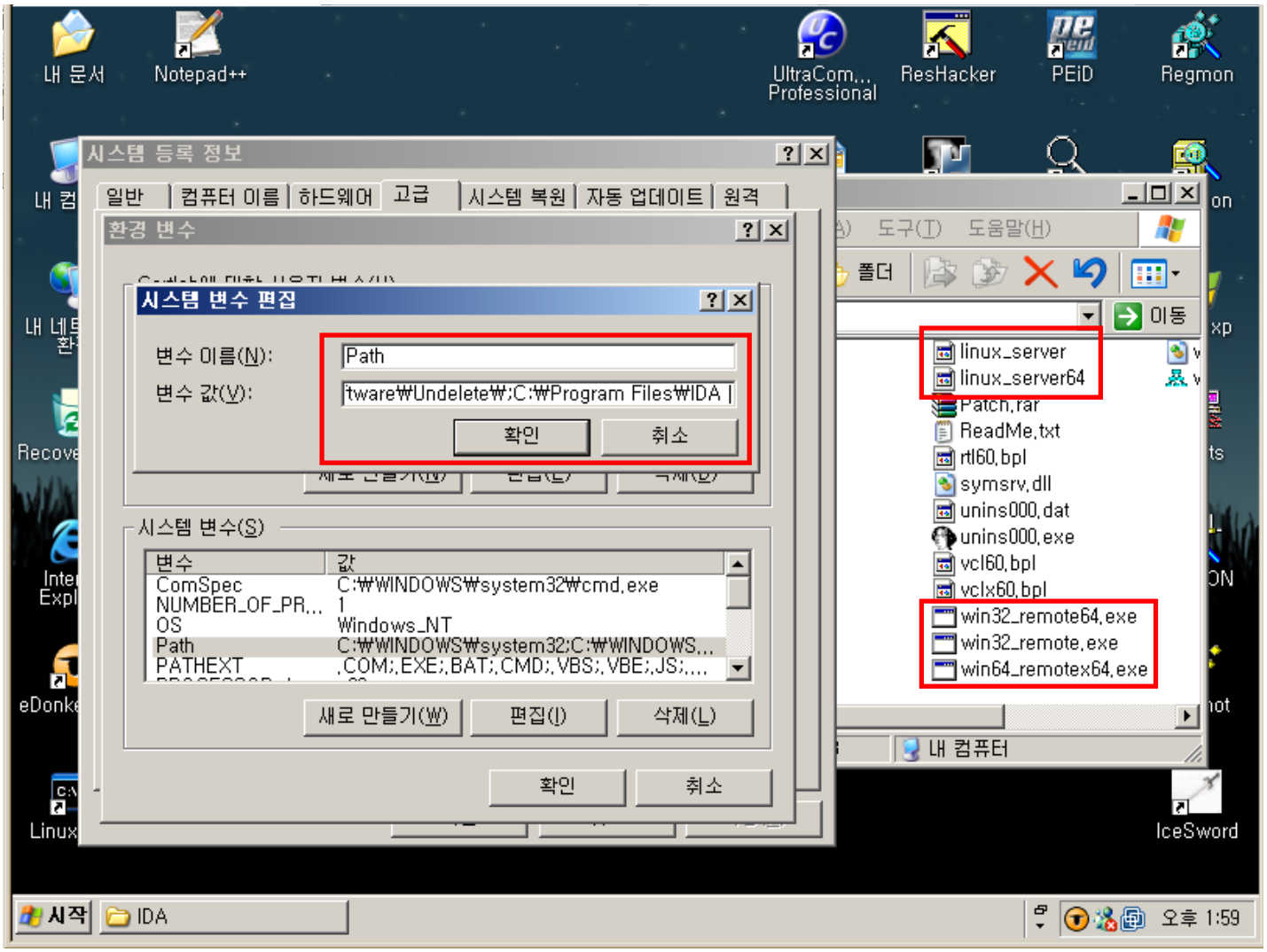


그림으로 보면 이해하기 쉬울것입니다.

이제 Remote debugging을 위한 설정을 해봅시다.
설정은 의외로 간단합니다.

디버깅 되어 실행이 될(분석도구가 있는 분석전용컴퓨터)가 서버가 됩니다.
Debugger Server 설정을 보도록 하겠습니다.

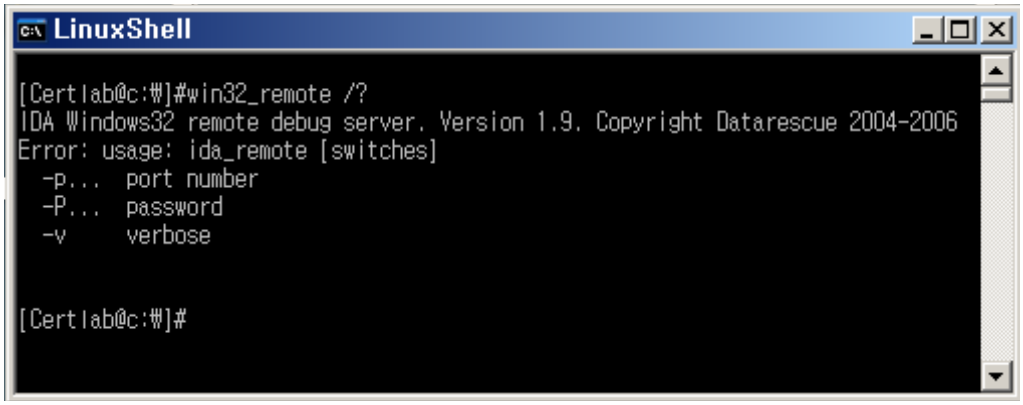
우선 IDA가 설치된 디렉토리를 Path에 넣어줍니다.



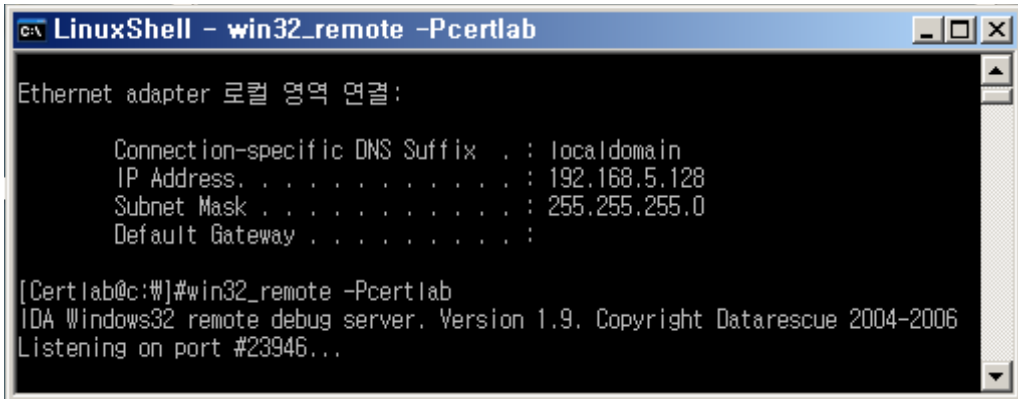
win32_remote, linux_server 파일이 있는것을 볼수 있고, 정리를 해보았습니다.
여기서 제일 위에 있는 win32_remote.exe 를 이용해 서버로 만들 것이고, 나머지는 각각의 플랫폼에 맞춰 구성을 하시면 됩니다.

File name	IDA version	Operating system	Debugged programs
win32_remote.exe	32-bit	MS Windows 32-bit	32-bit PE files
win32_remote64.exe	64-bit	MS Windows 32-bit	32-bit PE files
win64_remotex64.exe	64-bit	MS Windows 64-bit	64 or 32-bit PE files
linux_server	32-bit	Linux	ELF files
linux_server64	64-bit	Linux	ELF files

win32_remote argument를 보겠습니다.
port number는 23946으로 default 설정되어있습니다.



password를 certlab으로 설정한후에 실행을 합니다.
Client로 설정할 Main Computer에서 접속을 할것이기 때문에 IP정보도 확인합니다. 192.168.5.128
그러면 default로 설정된 23946 port 로 listening 을 하게 되고, Debugger Server의 설정은 이것으로 끝입니다.

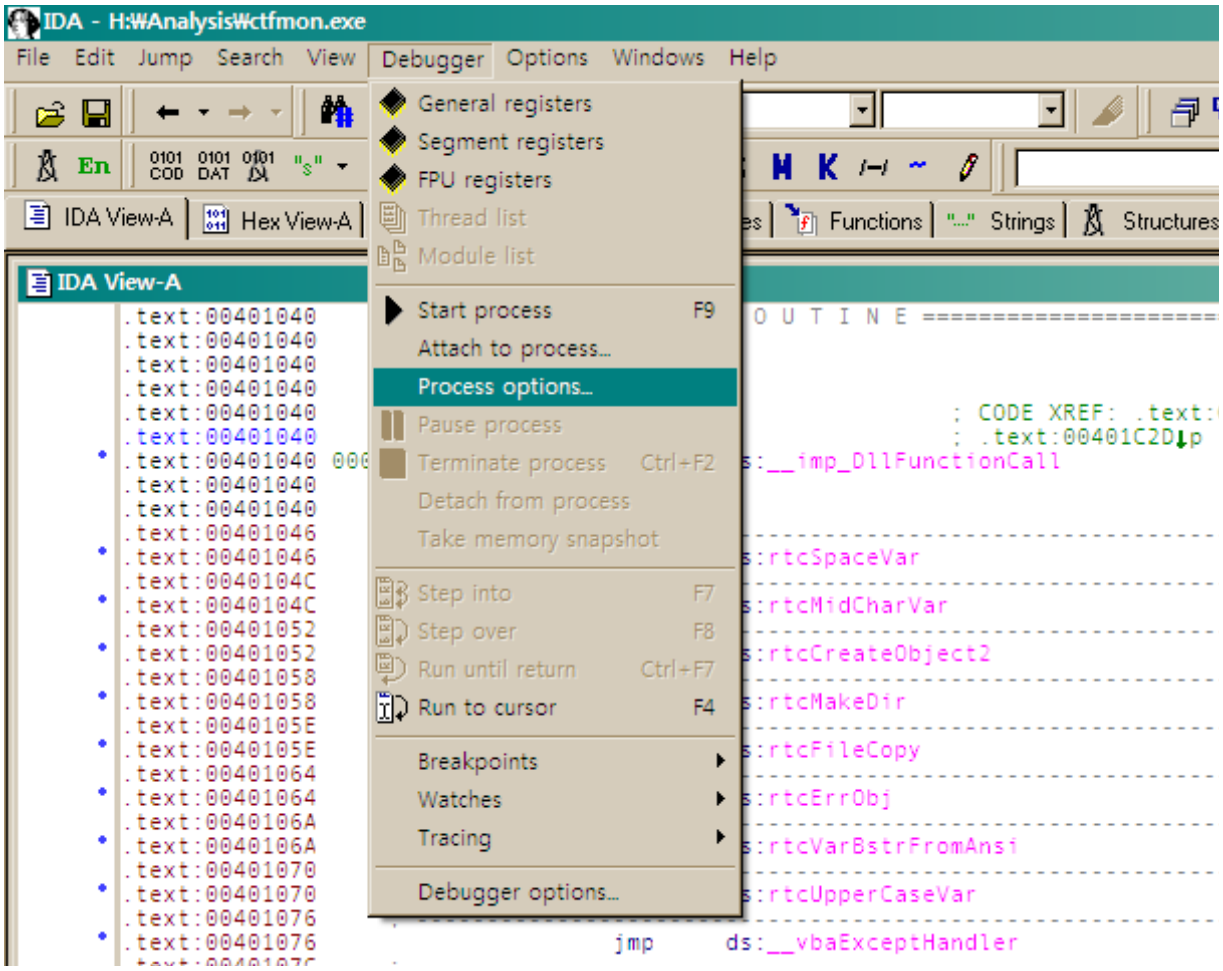


이제 디버깅을 하게될 컴퓨터인 Main Computer 의 설정. 즉 Debugger Client의 설정을 보겠습니다.
IDA를 실행하고 위험한 파일. 즉 악성코드를 IDA로 불러 들입니다.
그러면 악성코드의 디어셈블된 화면을 보실수 있고, 여기까지는 단순히 파일을 디어셈블만 하기 때문에 감염될 걱정은 없습니다.

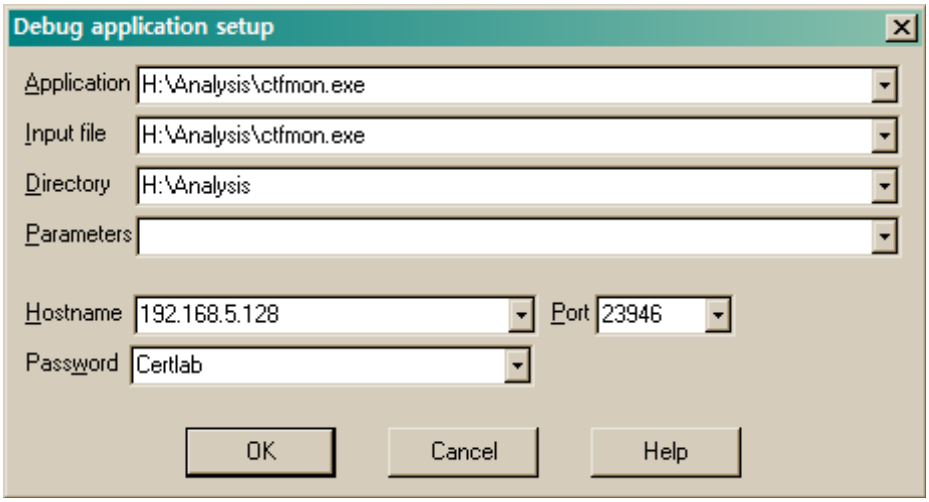
불러들일 악성코드는 다음과 같고, Blink Professional에서 검색된 결과입니다.

```
Event ID:  BLINK-MAL-205
Severity:  High
Description:  Blink has found a malware application
Virus found: W32/VBTroj.CUO
Item found: H:₩Analysis₩ctfmon.exe
Action: Repair
Alert: Yes
Name: W32/VBTroj.CUO
Second Action: Quarantine
Category: Trojan
```

Debugger -〉 Process options 를 누릅니다.

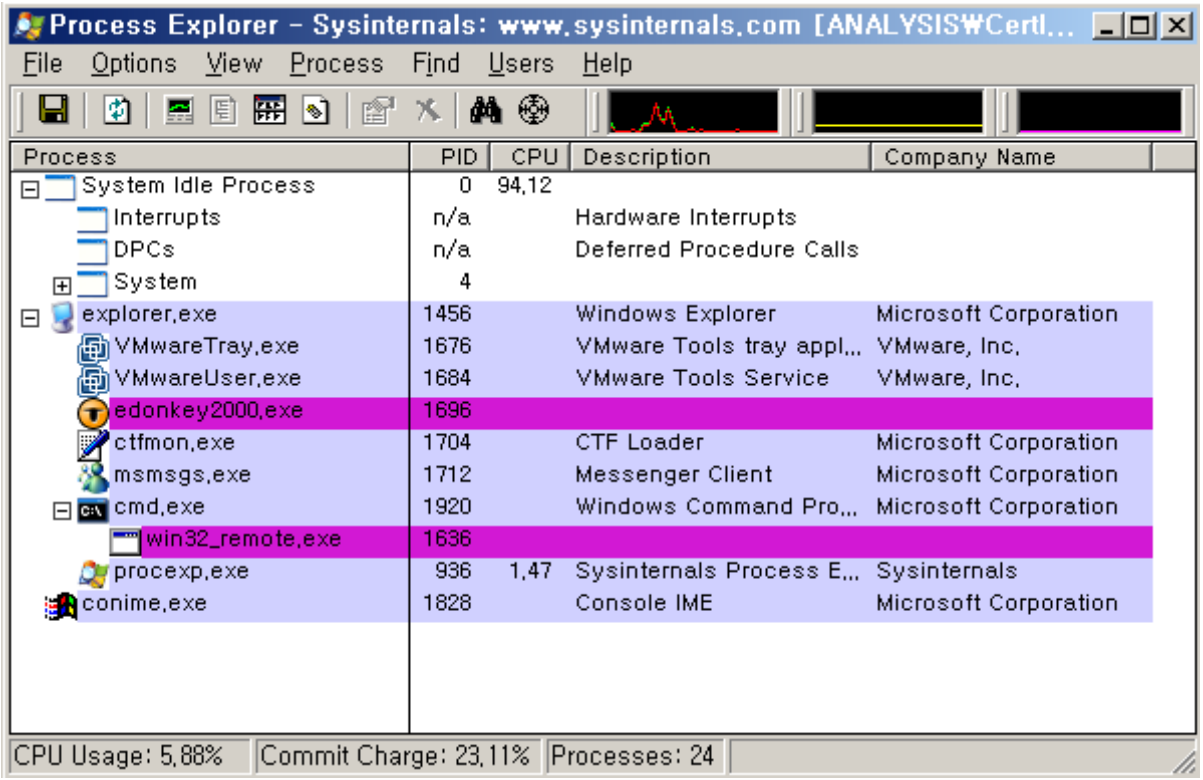


그러면 Debug application setup 창이 열리게 되고, 여기서
Hostname에 분석컴퓨터의 IP를 넣어주고, 설정한 패스워드를 입력한후에 OK를 누릅니다.

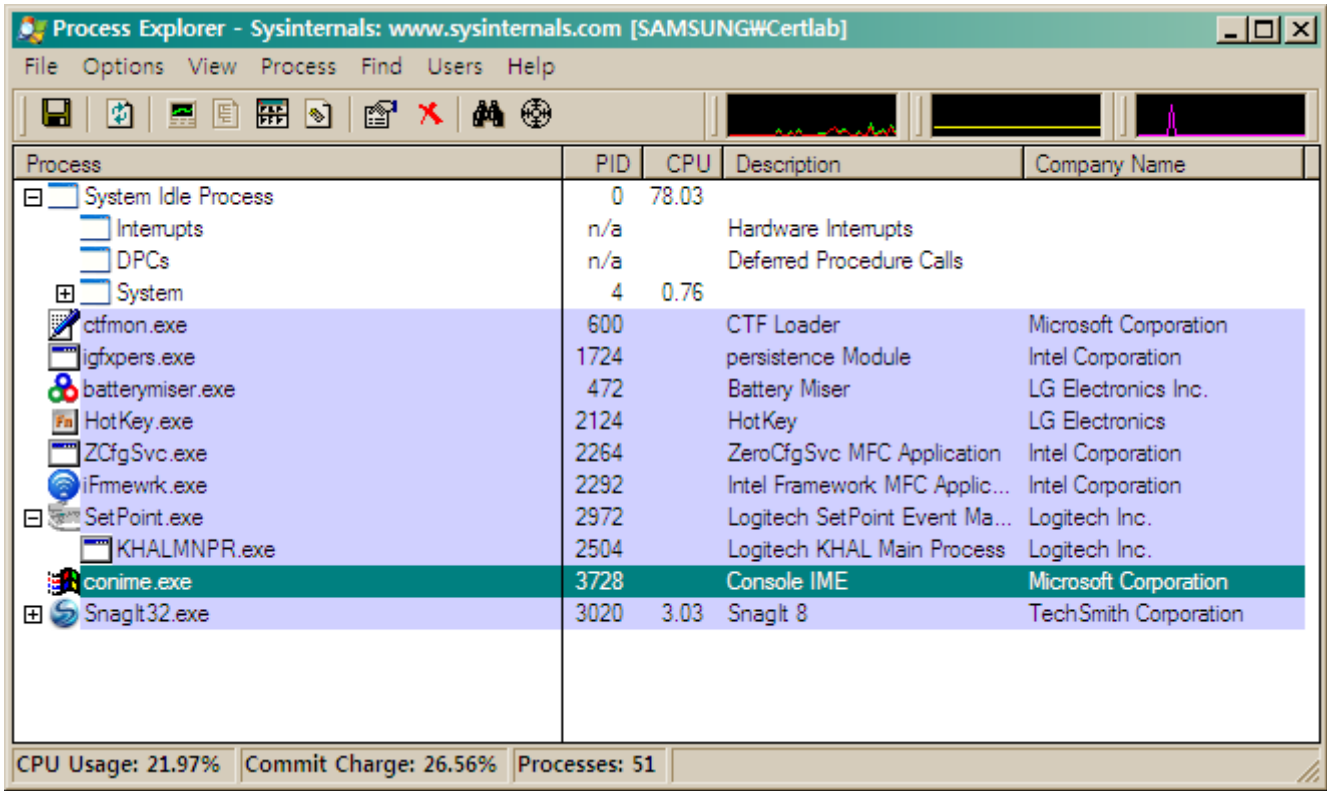


OK를 누르게 되면 반응이 없을것입니다. 지금한것은 Remote Debugging 설정을 한것이고,
악성코드를 이제 디버깅을 하게 되면 말그데로 remote debugging을 하게 되는것입니다.
디버깅을 하기 전에 각각의 컴퓨터에(Main Computer, Analysis Computer)
악성코드가 실행되었을때 악성코드가 실행되어지는 컴퓨터가 무엇인지 알아보기 위해 Process moniter로 반응을 보도록 하겠습니다.

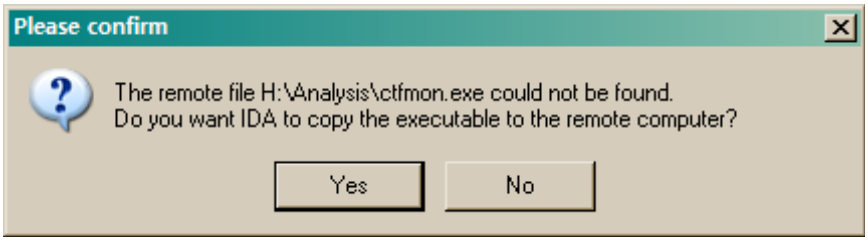
Analysis Computer



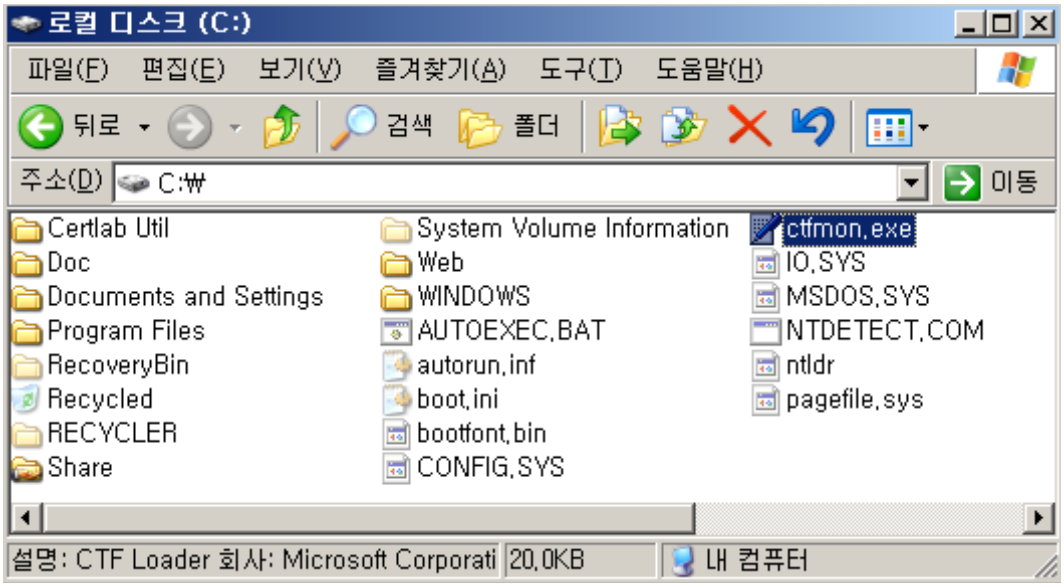
Main Computer


Process Explorer - Sysinternals: www.sysinternals.com [SAMSUNG₩Certlab]
File Options View Process Find Users Help
Process PID CPU Description Company Name
System Idle Process 0 78.03
Interrupts n/a Hardware Interrupts
DPCs n/a Deferred Procedure Calls
System 4 0.76
ctfmon.exe 600 CTF Loader Microsoft Corporation
igfxpers.exe 1724 persistence Module Intel Corporation
batterymiser.exe 472 Battery Miser LG Electronics Inc.
HotKey.exe 2124 HotKey LG Electronics
ZCfgSvc.exe 2264 ZeroCfgSvc MFC Application Intel Corporation
iFrmewrk.exe 2292 Intel Framework MFC Applic... Intel Corporation
SetPoint.exe 2972 Logitech SetPoint Event Ma... Logitech Inc.
KHALMNPR.exe 2504 Logitech KHAL Main Process Logitech Inc.
conime.exe 3728 Console IME Microsoft Corporation
SnagIt32.exe 3020 3.03 SnagIt 8 TechSmith Corporation
CPU Usage: 21.97% Commit Charge: 26.56% Processes: 51

이제 디버깅을 해보도록 하겠습니다.
디버깅 버튼을 누르게 되면 다음과 같은 창이 뜨게 되는데 현재 Main Computer의 악성코드 경로인 H:₩Analysis₩ctfmon.exe 파일이 remote로 설정한 컴퓨터의 H:₩Analysis₩ctfmon.exe 에 없다는 뜻이고, remote computer에 복사를 하겠냐고 물어보는 창입니다. Yes를 누릅니다.

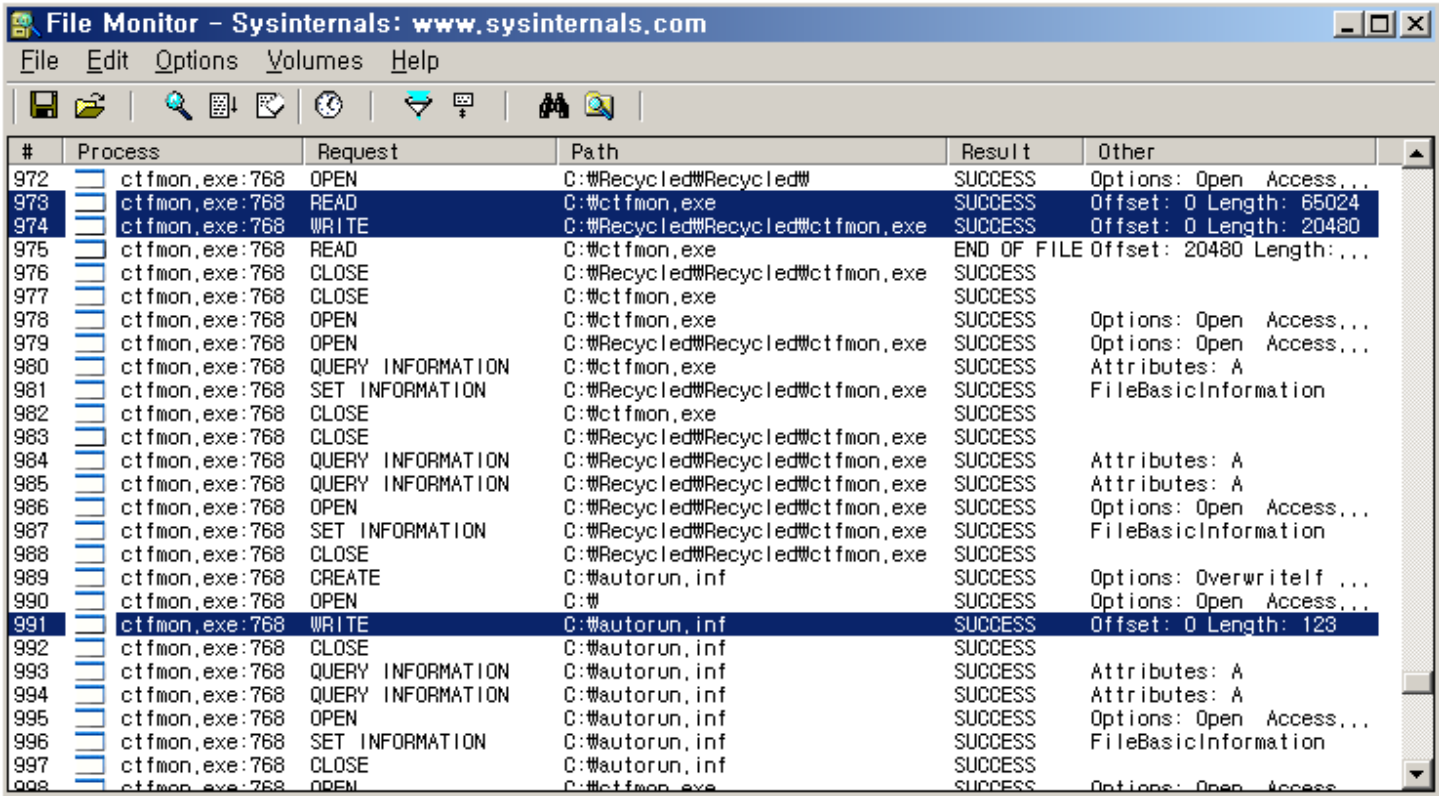


Yes를 누르게 되면 IDA Debugging mode 로 전환이 되고, Debugging 후에는 분석컴퓨터의 c:₩ctfmon.exe 악성코드가 복사되고, 실행이 된것을 보실수 있습니다.



물론 Main computer에서는 악성코드가 실행되지 않았고, 악성코드가 실행된 분석 컴퓨터에는 악성코드가 실행되어 여러 가지 일들을 수행하는것을 볼수 있습니다.

다음은 악성코드가 휴지통 밑에 휴지통 밑에 ctfmon.exe 라는 악성코드를 숨기는것과 c:₩autorun.inf 파일을 생성하는것을 볼수 있습니다.

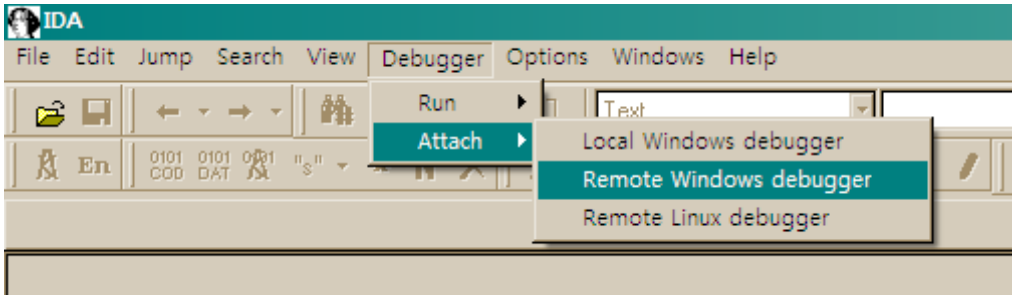


#	Process	Request	Path	Result	Other
972	ctfmon.exe:768	OPEN	C:₩Recycled₩Recycled₩	SUCCESS	Options: Open Access...
973	ctfmon.exe:768	READ	C:₩ctfmon.exe	SUCCESS	Offset: 0 Length: 65024
974	ctfmon.exe:768	WRITE	C:₩Recycled₩Recycled₩ctfmon.exe	SUCCESS	Offset: 0 Length: 20480
975	ctfmon.exe:768	READ	C:₩ctfmon.exe	END OF FILE	Offset: 20480 Length:...
976	ctfmon.exe:768	CLOSE	C:₩Recycled₩Recycled₩ctfmon.exe	SUCCESS	
977	ctfmon.exe:768	CLOSE	C:₩ctfmon.exe	SUCCESS	
978	ctfmon.exe:768	OPEN	C:₩ctfmon.exe	SUCCESS	Options: Open Access...
979	ctfmon.exe:768	OPEN	C:₩Recycled₩Recycled₩ctfmon.exe	SUCCESS	Options: Open Access...
980	ctfmon.exe:768	QUERY INFORMATION	C:₩ctfmon.exe	SUCCESS	Attributes: A
981	ctfmon.exe:768	SET INFORMATION	C:₩Recycled₩Recycled₩ctfmon.exe	SUCCESS	FileBasicInformation
982	ctfmon.exe:768	CLOSE	C:₩ctfmon.exe	SUCCESS	
983	ctfmon.exe:768	CLOSE	C:₩Recycled₩Recycled₩ctfmon.exe	SUCCESS	
984	ctfmon.exe:768	QUERY INFORMATION	C:₩Recycled₩Recycled₩ctfmon.exe	SUCCESS	Attributes: A
985	ctfmon.exe:768	QUERY INFORMATION	C:₩Recycled₩Recycled₩ctfmon.exe	SUCCESS	Attributes: A
986	ctfmon.exe:768	OPEN	C:₩Recycled₩Recycled₩ctfmon.exe	SUCCESS	Options: Open Access...
987	ctfmon.exe:768	SET INFORMATION	C:₩Recycled₩Recycled₩ctfmon.exe	SUCCESS	FileBasicInformation
988	ctfmon.exe:768	CLOSE	C:₩Recycled₩Recycled₩ctfmon.exe	SUCCESS	
989	ctfmon.exe:768	CREATE	C:₩autorun.inf	SUCCESS	Options: OverwriteIf ...
990	ctfmon.exe:768	OPEN	C:₩	SUCCESS	Options: Open Access...
991	ctfmon.exe:768	WRITE	C:₩autorun.inf	SUCCESS	Offset: 0 Length: 123
992	ctfmon.exe:768	CLOSE	C:₩autorun.inf	SUCCESS	
993	ctfmon.exe:768	QUERY INFORMATION	C:₩autorun.inf	SUCCESS	Attributes: A
994	ctfmon.exe:768	QUERY INFORMATION	C:₩autorun.inf	SUCCESS	Attributes: A
995	ctfmon.exe:768	OPEN	C:₩autorun.inf	SUCCESS	Options: Open Access...
996	ctfmon.exe:768	SET INFORMATION	C:₩autorun.inf	SUCCESS	FileBasicInformation
997	ctfmon.exe:768	CLOSE	C:₩autorun.inf	SUCCESS	

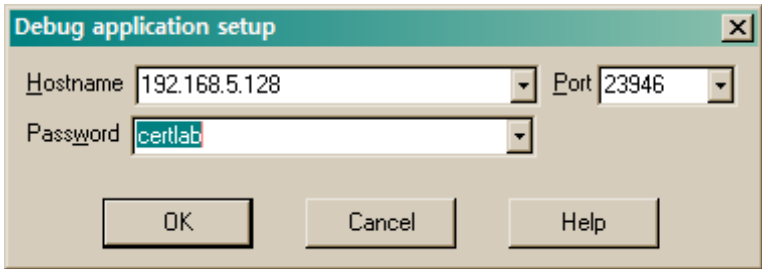
그 외에 여러 가지 dll 파일들을 생성하고, 시작 Registry에 추가하는 악성코드입니다.
이렇게 remote debugging을 통해 분석을 하는 방법을 알았고,

remote computer 의 process 를 attach 해서 분석하는 방법을 알아봅시다.

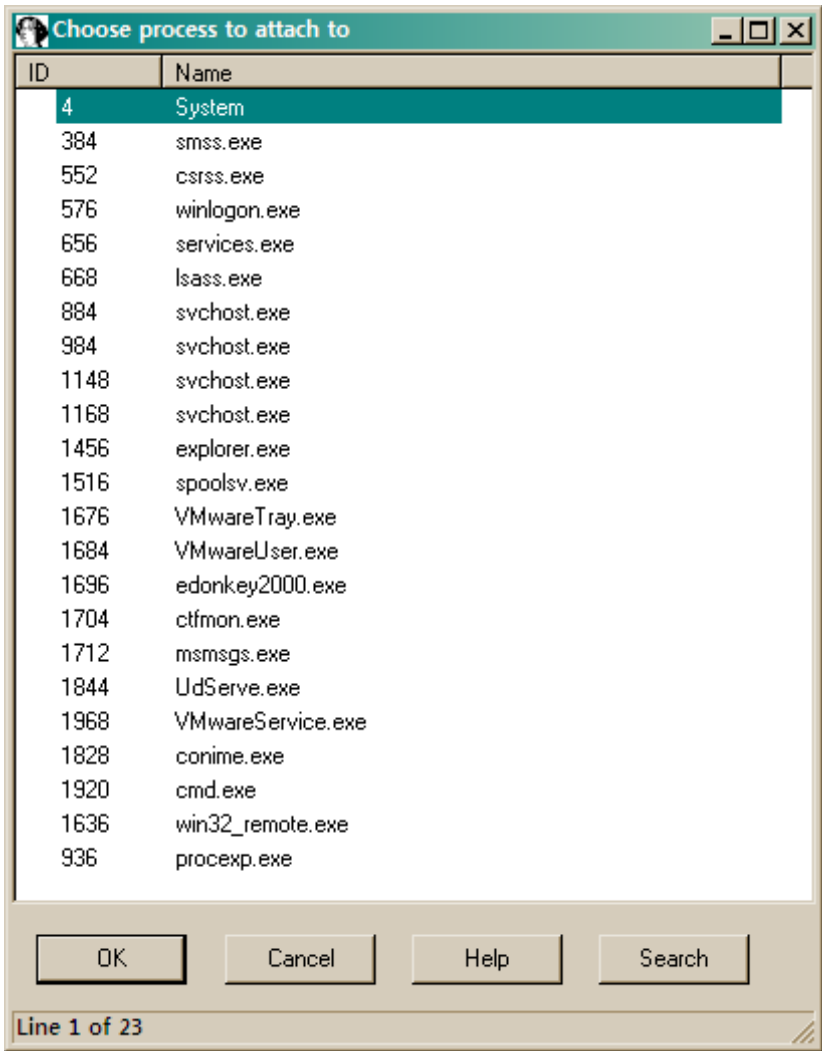
win32_remote.exe 로 Debugger Sever 로 설정을 하고,
Debugger Client에서 다음과 같이 Remote Windows debugger를 누릅니다.



그럼 다음과 같이 Debug application setup 창이 뜨게 되고, OK 를 누릅니다.



그럼 다음과 같이 Remote computer의 현재 Process list를 볼수 있고,
Attach할 Process를 확인후 분석을 하시면 됩니다.



이렇게 Remote Debugging에 대해 알아보았습니다.
이것을 응용해 **분 석**은 여러분의 몫입니다. :)

문서 내용중에 틀린부분이나 잘못된 내용은 메일을 보내주시면 감사드리겠습니다.
http://www.certlab.org
mailto:certlab@gmail.com