
보안 위협 형태와 악성코드 분석 기법

2011.11.14

(주) 안철수연구소

ASEC (AhnLab Security Emergency response Center)

Senior Advanced Threat Researcher, MCSE, MCDBA, MCSA, CISSP

한국 CISSP 협회 보안연구 부분 간사

장 영 준 선임 연구원

(E-mail : zhang95@ahnlab.com, Twitter : @YoungjunChang)

I. 악성코드 정의와 특징

II. 안티 바이러스 엔진과 진단 기법

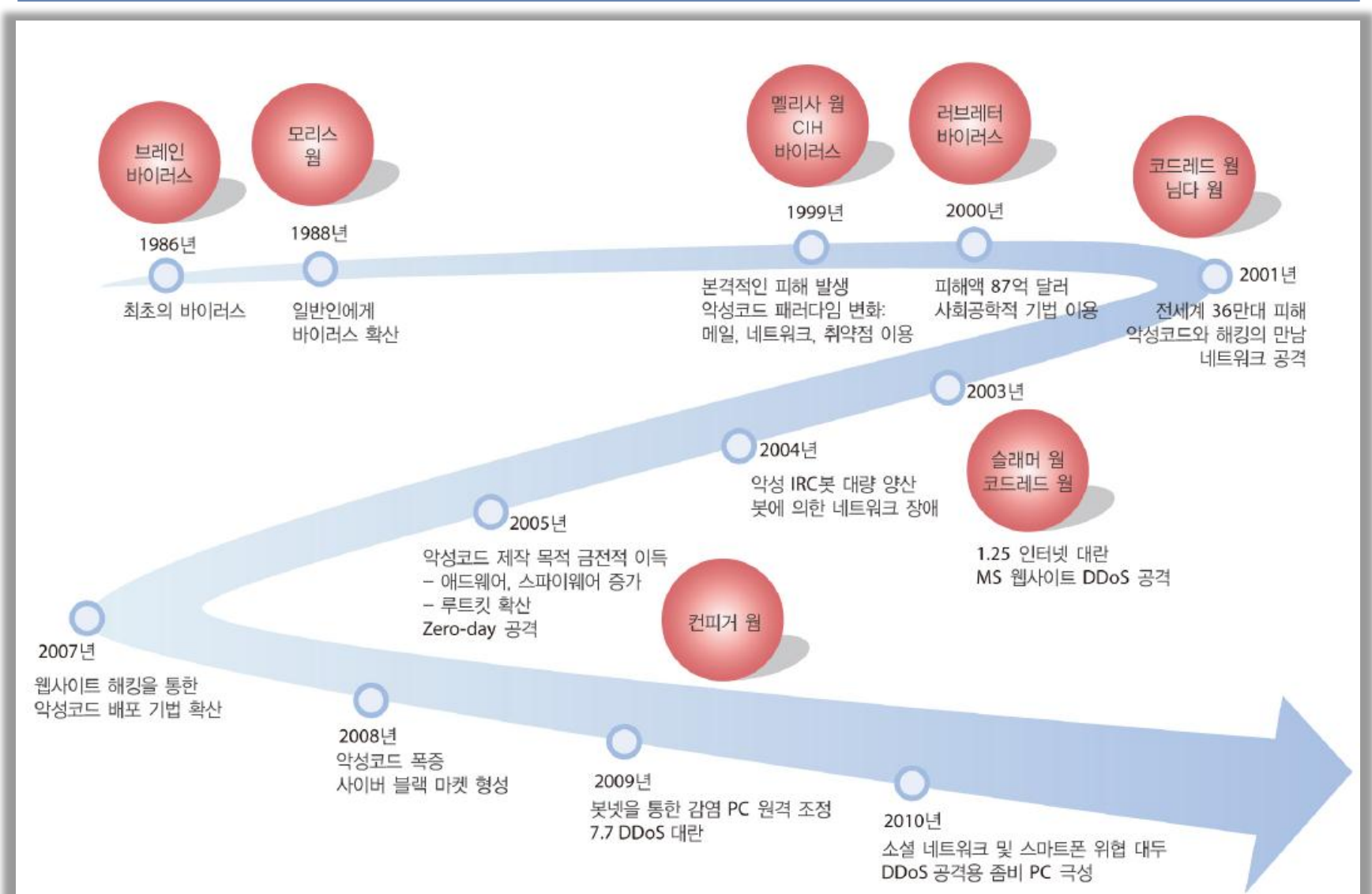
III. 악성코드 분석 방안

IV. 악성코드 분석 기법

1

악성코드 정의와 특징

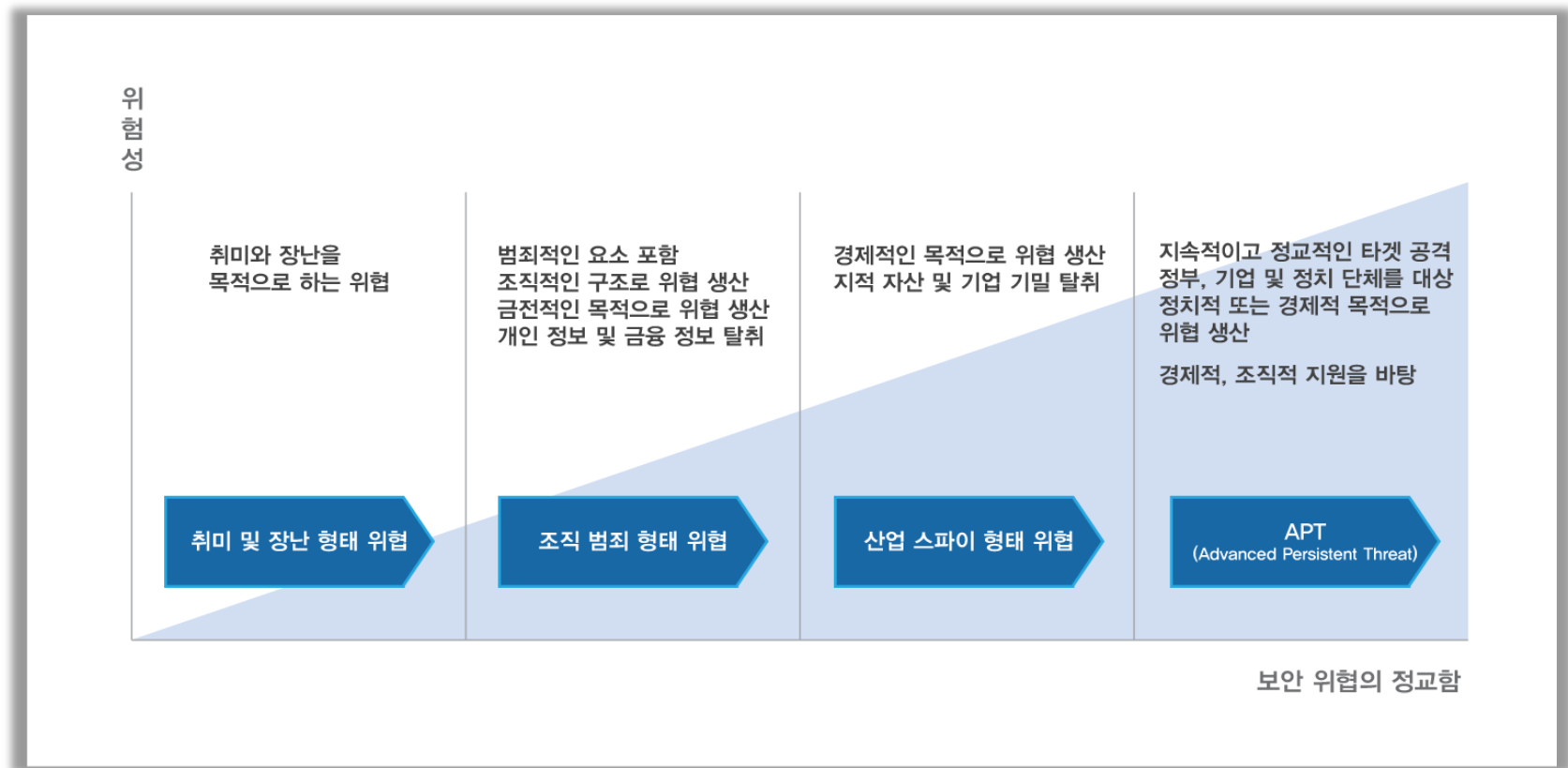
1. 악성코드 정의와 특징



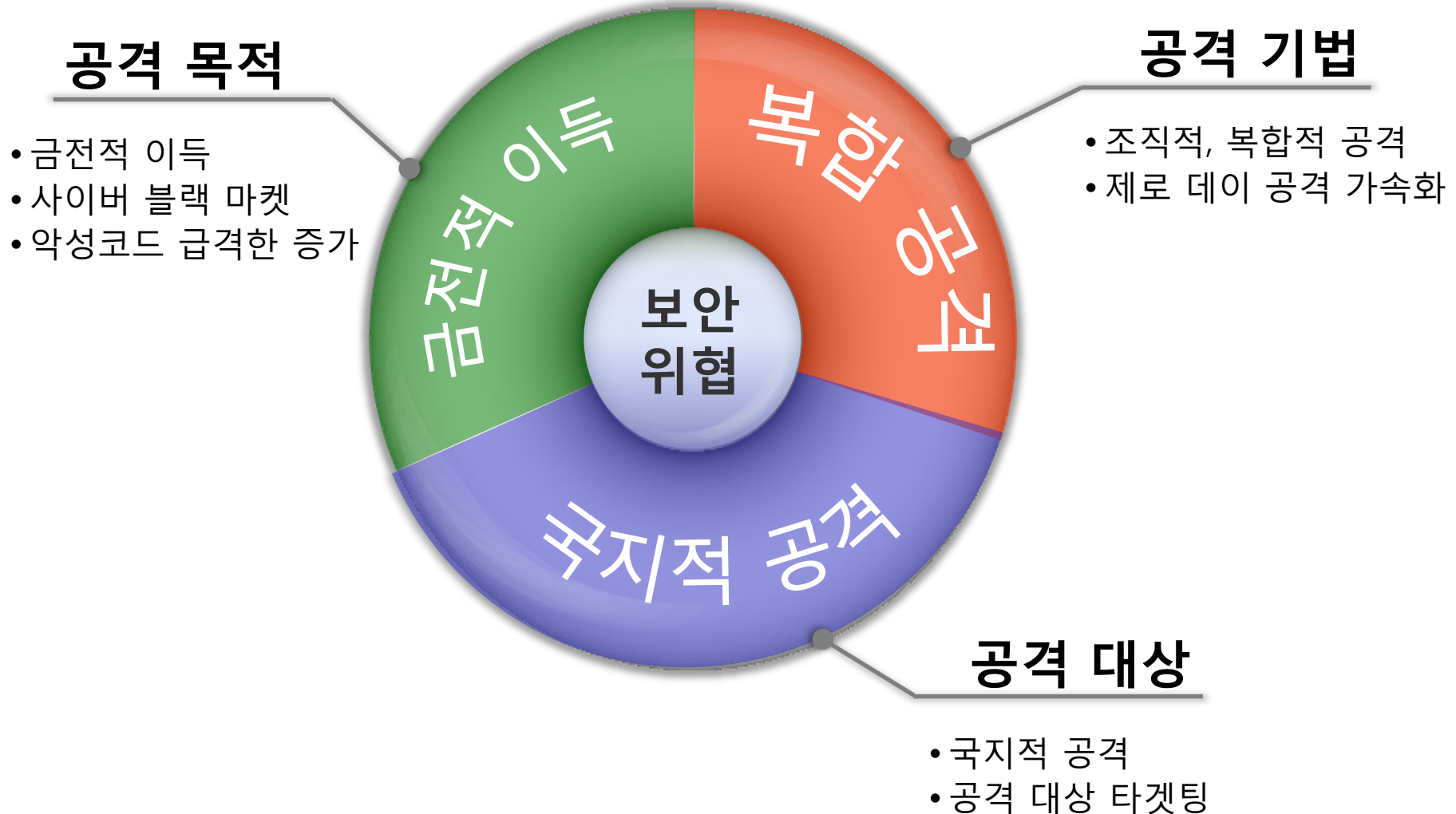
1. 악성코드 정의와 특징

◆ 악성코드 제작 목적의 변화

과거 악성코드 제작은 취미와 장난 그리고 자기 실력 과시를 위해 제작
현재는 사이버 범죄의 형태를 거쳐 APT(Advanced Persistent Threat) 형태로 발전



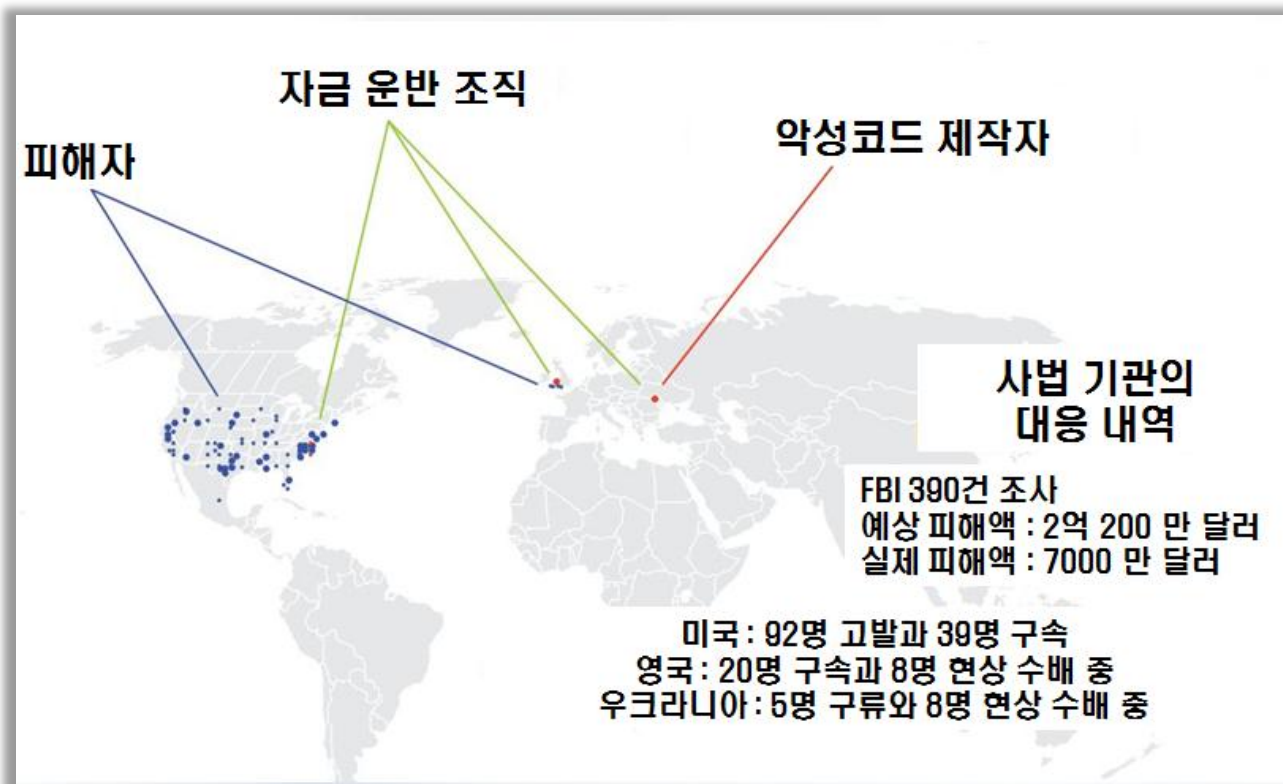
1. 악성코드 정의와 특징



1. 악성코드 정의와 특징

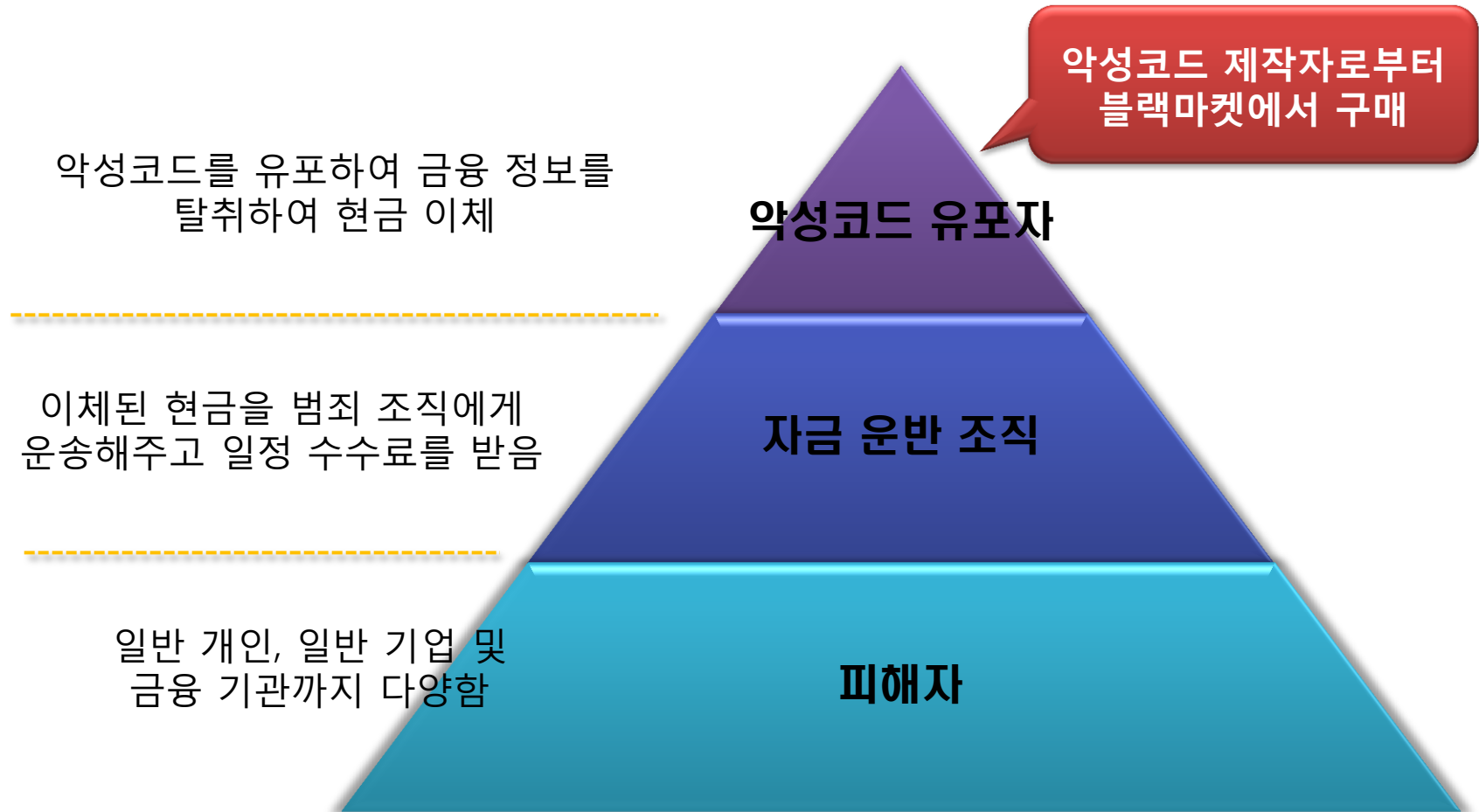
◆ 조직적인 사이버 범죄

사이버 범죄는 Black Market을 중심으로 러시아를 포함한 동유럽, 중국, 브라질에서 활발
대부분의 피해는 러시아를 포함한 동유럽에서 제작되는 보안 위협들로 인해 발생
뱅킹 트로이목마 Zeus의 경우 최소 7000만 달러에서 2억 200만 달러 피해 발생 추정



1. 악성코드 정의와 특징

◆ 사이버 범죄의 구성 요소

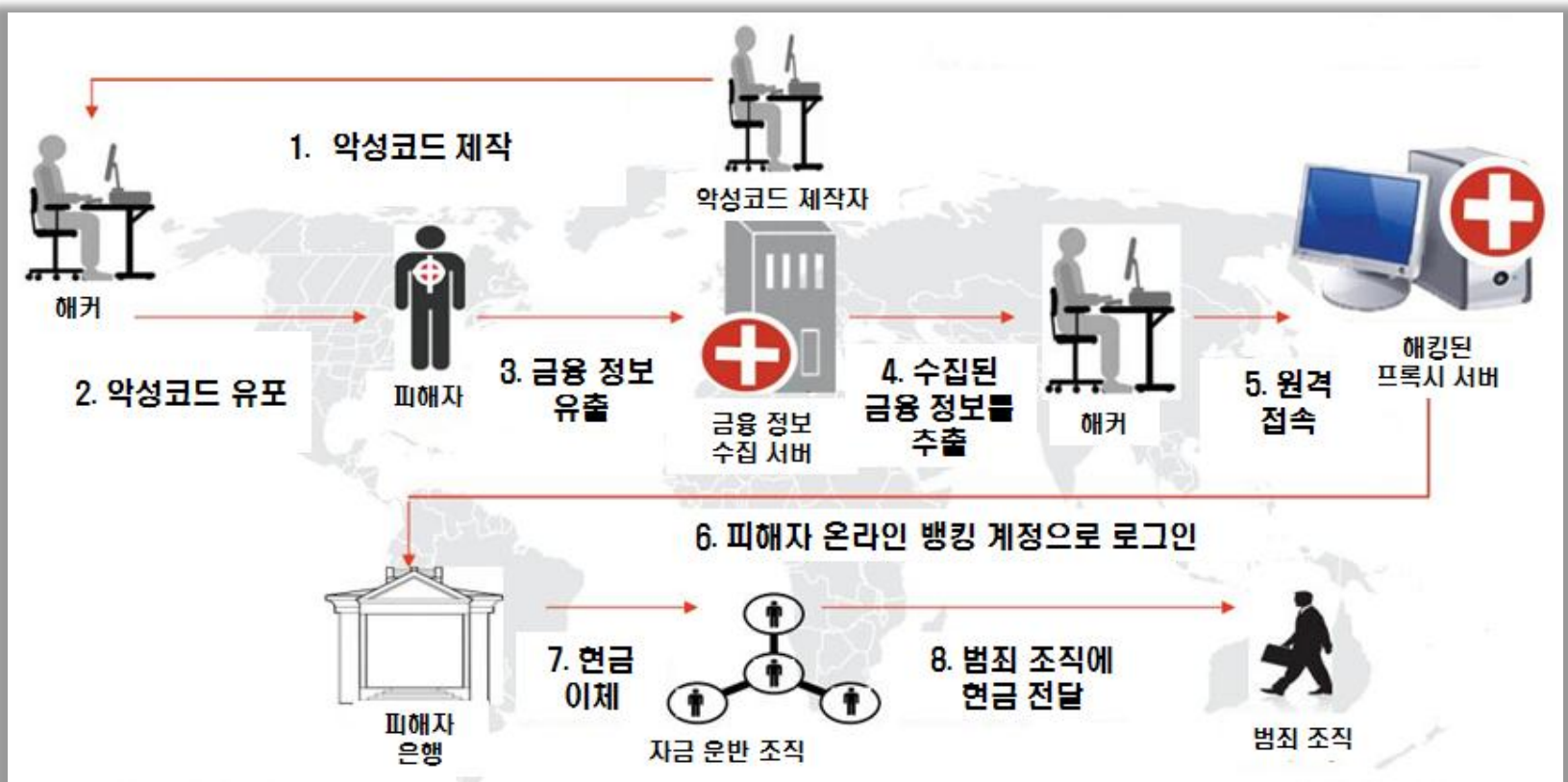


1. 악성코드 정의와 특징

◆ बैंकिंग 트로이목마를 이용한 사이버 범죄 구조

피해자는 일반인에서 금융 기관까지 다양하게 존재

자금 운반 조직은 현금을 범죄 조직에게 운송해주고 일정 수수료를 받음

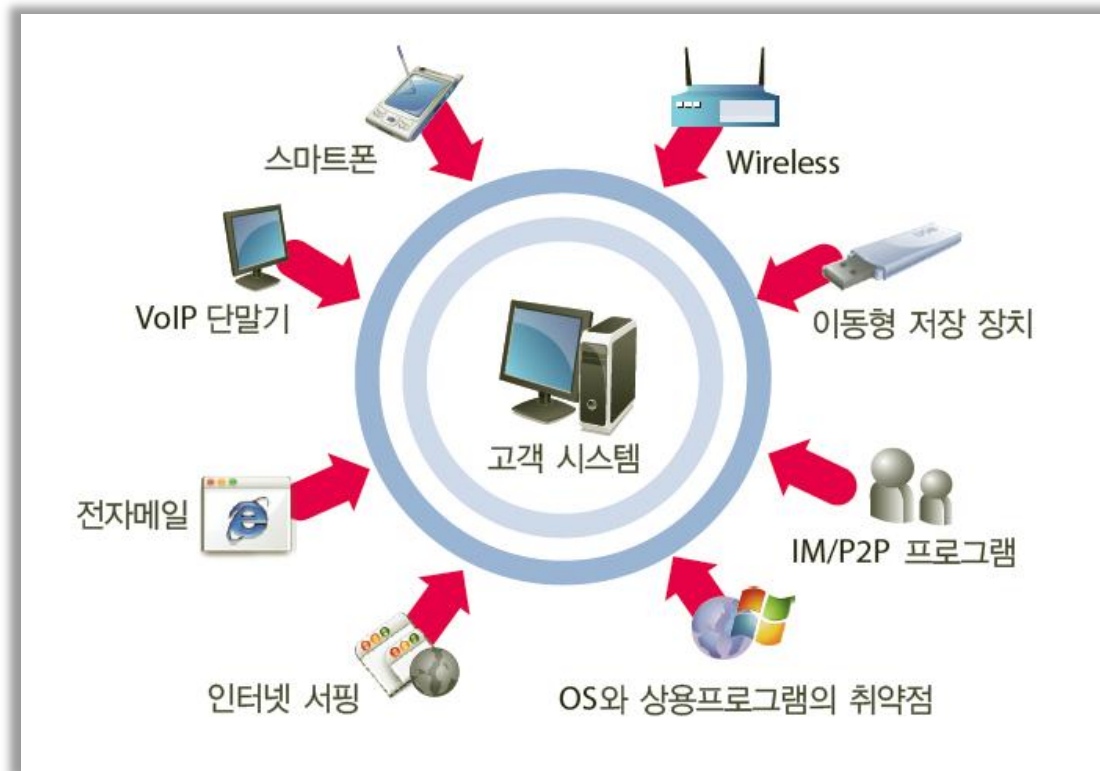


1. 악성코드 정의와 특징

◆ 악성코드 (Malware, Malicious Code, Malicious Software)

악성코드는 '악의적인 목적을 위해 작성된 실행 가능한 코드'

실행 가능한 코드로는 프로그램, 매크로, 스크립트 뿐만 아니라 취약점을 이용한 데이터 형태들도 포함



1. 악성코드 정의와 특징

바이러스

- 자기 자신 혹은 자신의 변형을 복제
- 감염 대상 반드시 존재
- 시스템 및 데이터 자체에 대한 피해
- 사용자 작업 지연 혹은 방해

트로이 목마

- 자기 복제 능력 없고 정상 프로그램 위장
- 특정 조건 및 배포자의 의도에 따라 동작
- 개인 정보 유출, 감염 대상 원격 조정
- 주요 해킹 툴 유포 방법

웜

- 네트워크를 통한 자기 복제
- 매우 빠른 시간 안에 피해 확산
- E-Mail, Messenger, P2P등 다양한 감염경로
- Trojan과 악의적인 기능은 유사

기타

- Spyware (사용자 정보 수집)
- Adware (광고목적 제작)
- Hoax (심리적 위협/불안 조장)
- Joke (재미 및 흥미 위주로 제작)

종류 \ 특성	주요 목적	피해 가능성	자기복제	감염대상	대책
바이러스	데이터 손실/삭제, 손상	O	O	O	치료(복원, 복구)
웜	급속 확산	O	O	X	삭제 (차단-개인 방화벽)
트로이 목마	데이터 손실/유출	O	X	X	삭제
유해가능 프로그램	사용 불편, 심리적 거부	△	X	X	삭제 (수동 or 툴)

1. 악성코드 정의와 특징

◆ 바이러스 (Virus)

일반적으로 감염대상이 되는 프로그램 또는 코드에 자신의 코드 및 변형 코드를 삽입
주로 컴퓨터 시스템 내부에서만 감염 및 확산

예) Brian, Michelangelo, CIH, FunLove, Nimda, Klez 등

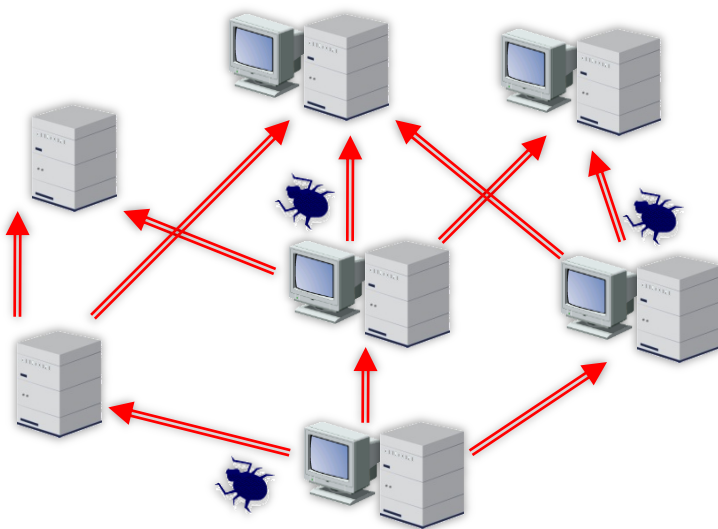


1. 악성코드 정의와 특징

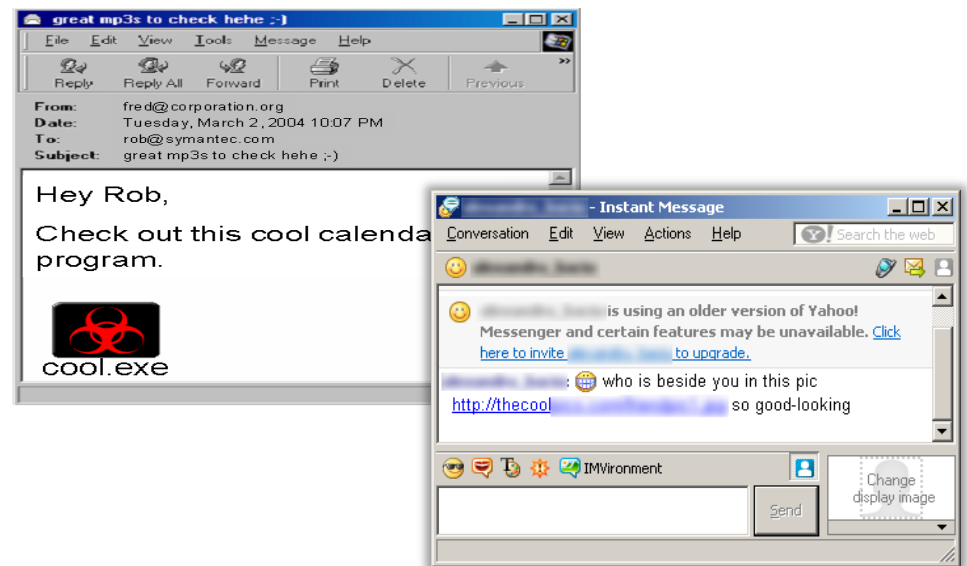
◆ 웜 (Worm)

컴퓨터의 기억장소 또는 내부에 코드 또는 실행파일 형태로 존재
실행 시 파일이나 코드를 네트워크, 전자메일과 인스턴트 메신저 프로그램 등을 통해 다른
시스템으로 자기 복제를 시도하는 형태

예) CodeRed, Blaster, Sasser, Bagle, Netsky, MyDoom 등



[네트워크를 이용한 전파]



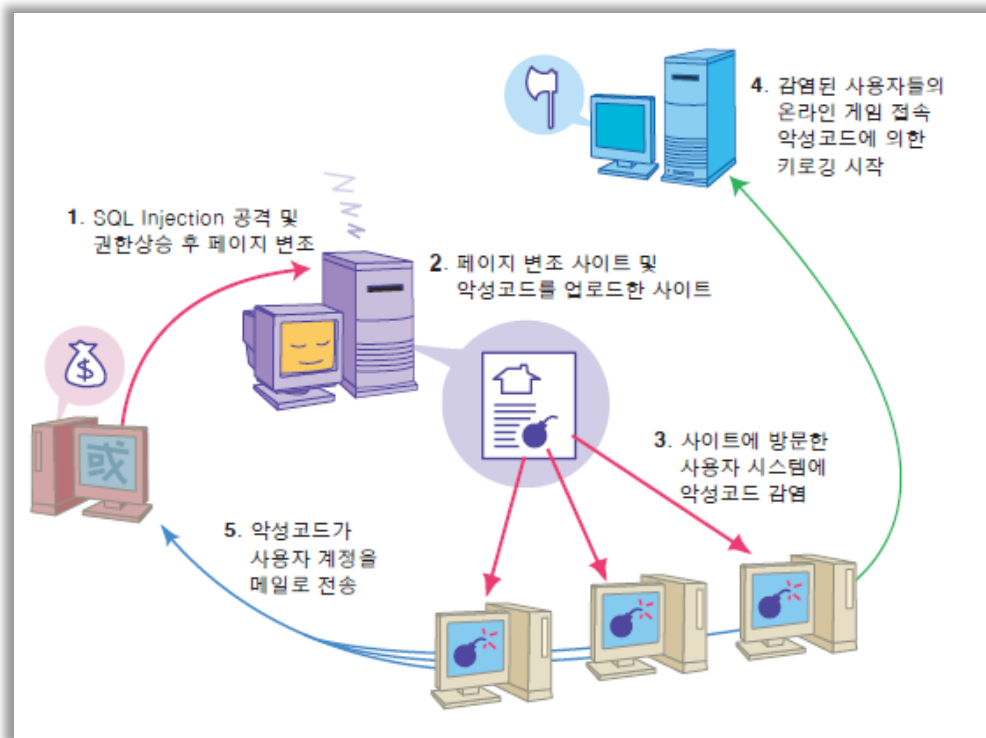
[전자 메일과 I.M를 이용한 전파]

1. 악성코드 정의와 특징

◆ 트로이 목마 (Trojan Horse)

자기 자신을 복제하지 않지만 악의적 기능을 포함하는 프로그램
악의적 목적에 적극적으로 활용되는 프로그램 또는 데이터 형태

예) OnlineGameHack, PeepViewer, Optix, BackOrifice 등



[웹 사이트 해킹을 통한 트로이목마의 대량 확산]

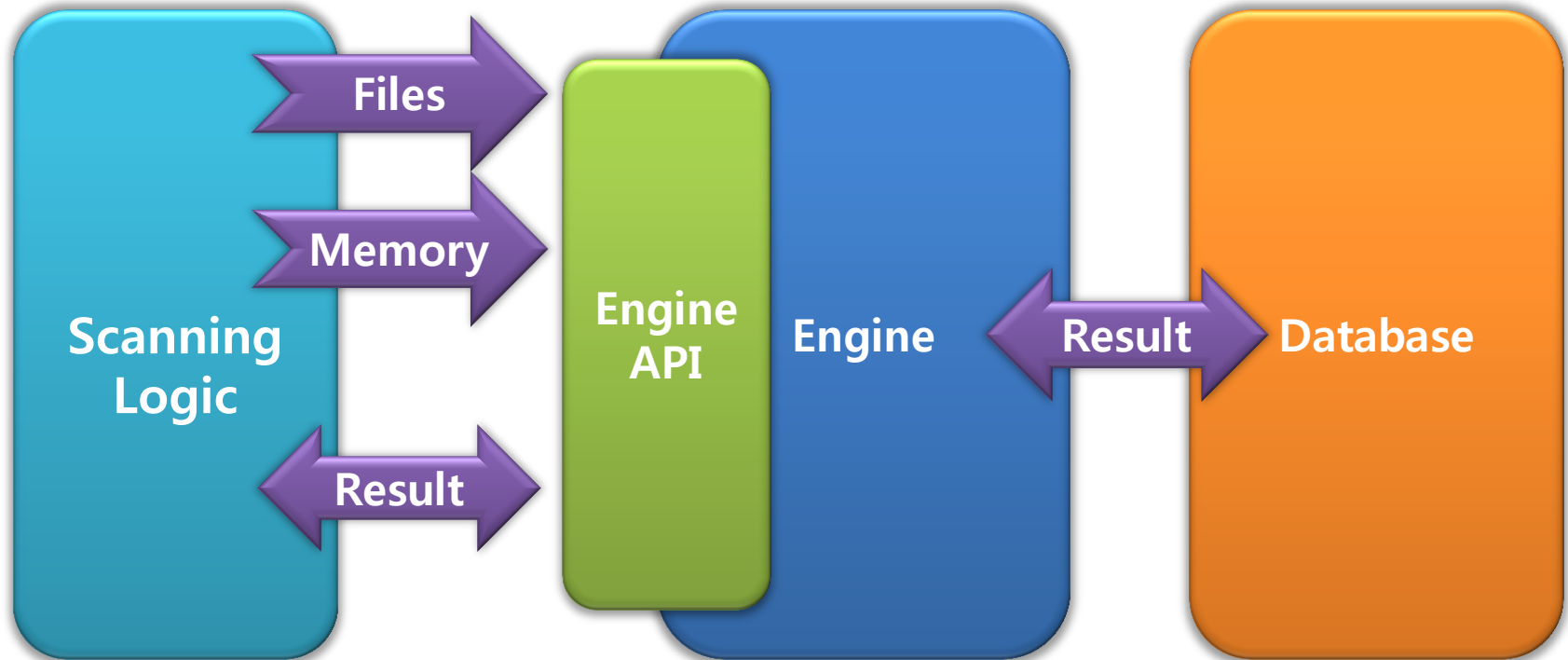
2

안티 바이러스 엔진과 진단 기법들

2. 안티 바이러스 엔진과 진단 기법들

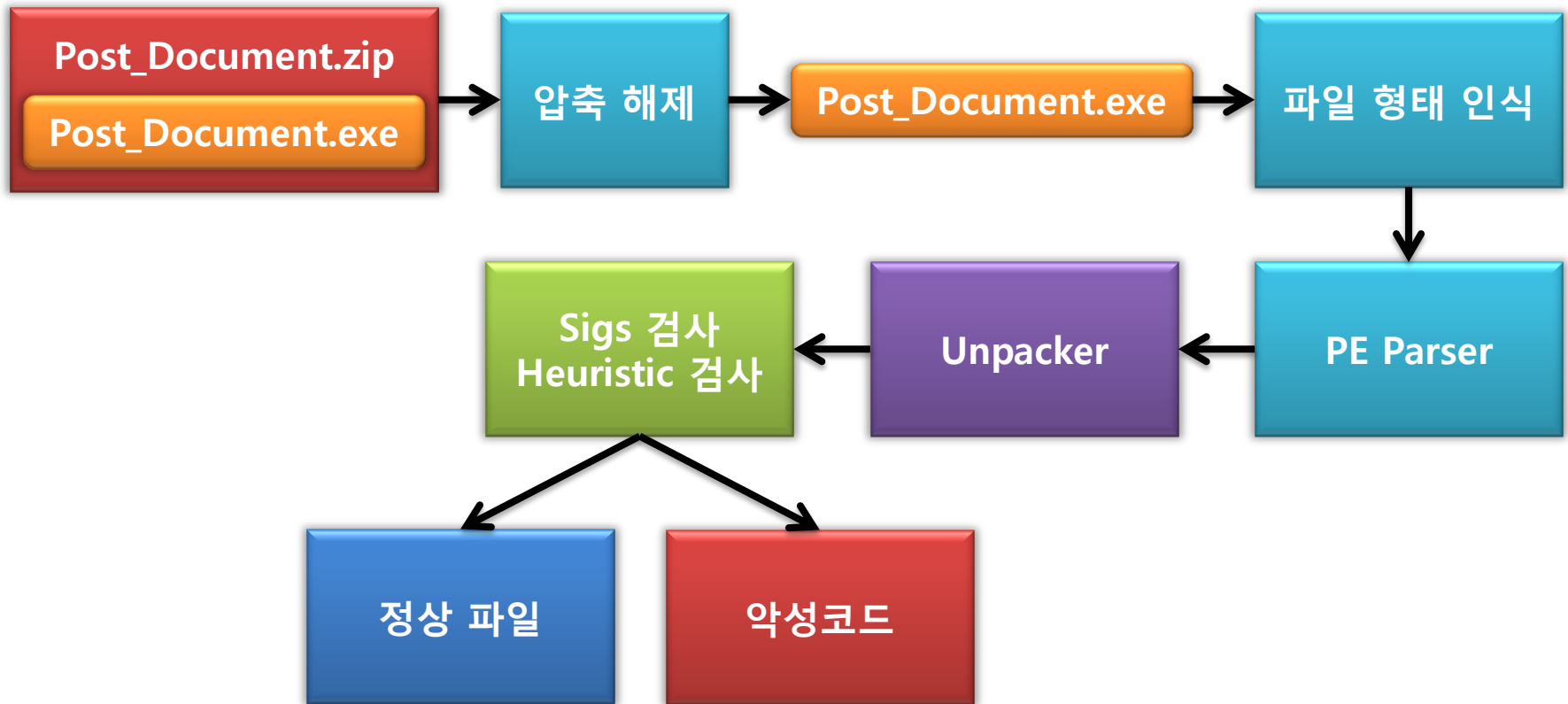
◆ 안티 바이러스 엔진

일반적으로 안티 바이러스 엔진은 3개 부분으로 구분 가능



2. 안티 바이러스 엔진과 진단 기법들

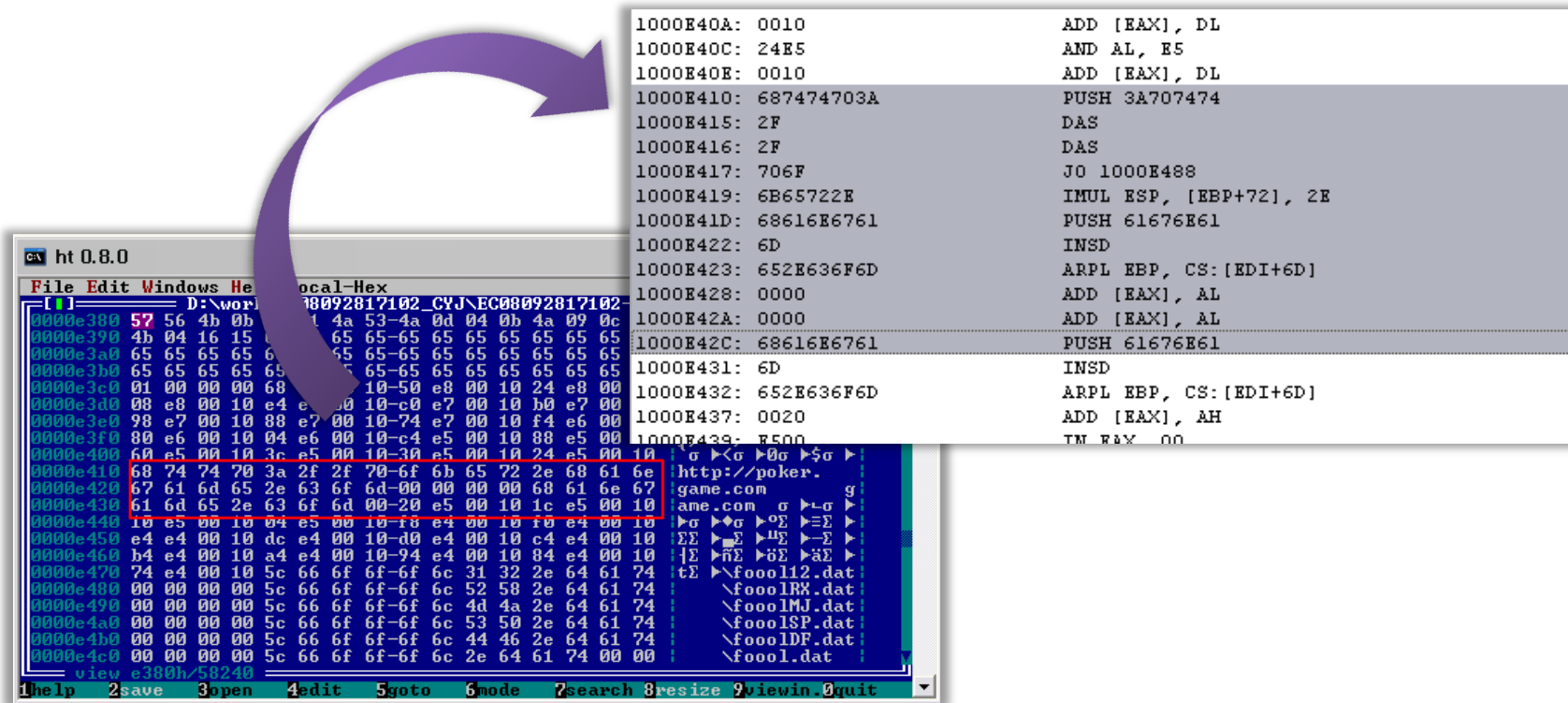
◆ 안티 바이러스 엔진을 이용한 파일 검사 과정



2. 안티 바이러스 엔진과 진단 기법들

◆ String 기반 진단

악성코드 내부 특정 코드 영역을 진단의 위치로 선정
가장 간단하고 빠르게 진단할 수 있는 기법



[Win-Trojan/OnlineGameHack.86016.CC 의 String 일부]

2. 안티 바이러스 엔진과 진단 기법들

◆ Generic 진단

특정 악성코드 집합의 공통된 OPcode(Operation Code) 영역을 진단 위치로 선정
알려지지 않은 변형들에 대해 유연하게 대응 가능

10001079	> 8D85 F8FDFFFF	LEA EAX,DWORD PTR SS:[EBP-208]	
1000107F	. 50	PUSH EAX	
10001080	. E8 19450000	CALL EC080928.1000559E	
10001085	. 8D85 F8FDFFFF	LEA EAX,DWORD PTR SS:[EBP-208]	
1000108B	. C70424 3CE10001	MOV DWORD PTR SS:[ESP],EC080928.1000E130	UNICODE "iexplore.exe"
10001092	. 50	PUSH EAX	
10001093	. E8 D1440000	CALL EC080928.10005569	
10001098	. 59	POP ECX	
10001099	. 85C0	TEST EAX,EAX	
1000109B	. 59	POP ECX	
1000109C	. 74 1B	JE SHORT EC080928.100010B9	
1000109E	. 8D85 F8FDFFFF	LEA EAX,DWORD PTR SS:[EBP-208]	
100010A4	. 68 20E10010	PUSH EC080928.1000E120	UNICODE "regsvr32.exe"
100010A9	. 50	PUSH EAX	
100010AA	. E8 BA440000	CALL EC080928.10005569	
100010AF	. 59	POP ECX	
100010B0	. 85C0	TEST EAX,EAX	
100010B2	. 59	POP ECX	

[Win-Trojan/OnlineGameHack.86016.CC의 I.E 인젝션 코드 일부]

2. 안티 바이러스 엔진과 진단 기법들

◆ Heuristic 진단

기존에 알려진 악성코드의 일반적인 특성을 바탕으로 그와 얼마나 유사한 코드를 가지고 있는가를 비교

❖ Static Heuristic Detection

악성코드의 실행 없이 기존 악성코드와 얼마나 많은 유사한 코드를 가지고 있는가를 비교 판단

❖ Dynamic Heuristic Detection

샌드박스(SandBox) 또는 가상화(Virtualization) 기술 등을 이용하여 악성코드 실행 시 나타나는 증상을 바탕으로 기존에 알려진 악성코드와 얼마나 유사한가를 판단

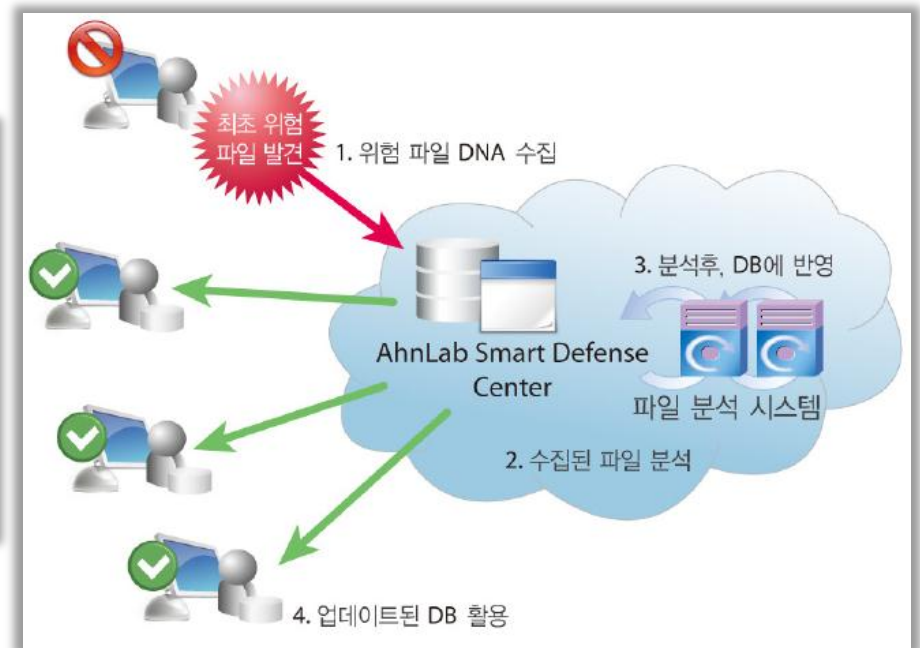
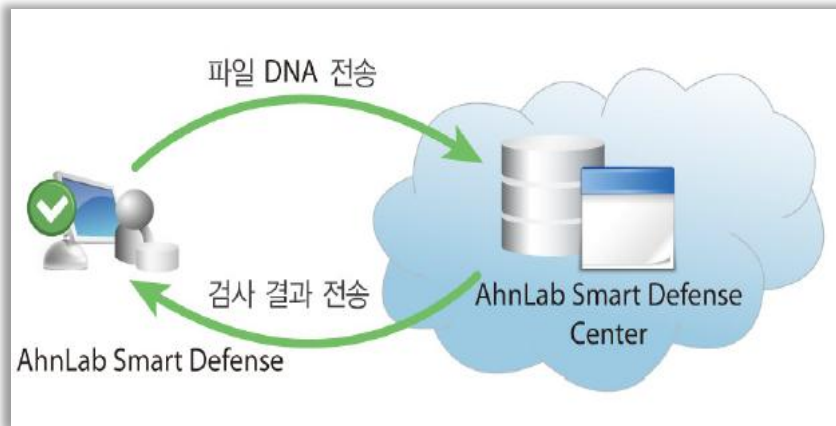
Reference

http://www.eset.com/resources/white-papers/Heuristic_Analysis.pdf

2. 안티 바이러스 엔진과 진단 기법들

◆ Cloud Anti-Virus

Cloud와 Virtualization 기술을 이용해 Cloud의 고사양, 대용량의 시스템으로 분석 악성코드의 각종 정보를 Database화 하여 Malware Mining으로 진단의 효율화 새로운 악성코드에 대한 신속한 확보 및 대응이 가능



2

악성코드 분석 방법론

2. 악성코드 분석 방법론

◆ Reverse Engineering

Reverse Engineering은 인공적으로 만들어진 사물(자동차, 제트 엔진, 소프트웨어 등)을 분해해서 설계나 구조와 같은 세밀한 사항들을 분석 하는 과정

Reverse Engineering은 RE(Reverse Engineering) 또는 RCE(Reverse Code Engineering)의 약칭



2. 악성코드 분석 방법론

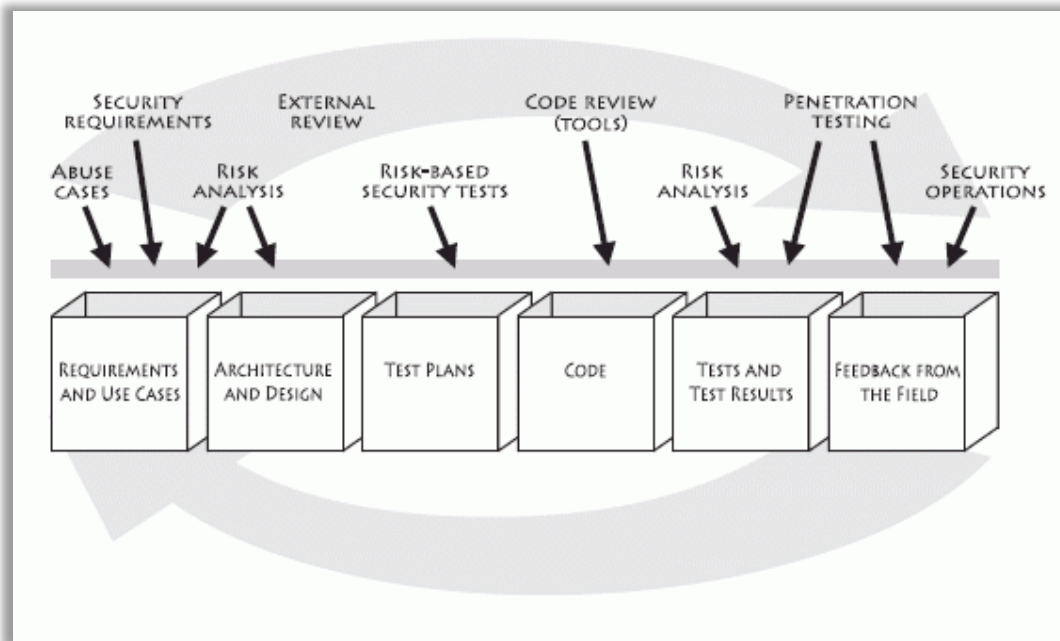
◆ Software Reverse Engineering

Software Reverse Engineering은 소스코드나 관련 문서 없이 프로그램에서 설계나 구현 내용을 알아 내는 작업

◆ Software Reverse Engineering 범위

보안 - 악성 코드 분석, 암호화 알고리즘 분석, 프로그램 바이너리 감사

소프트웨어 개발 - 소프트웨어 상호 운용 검증, 소프트웨어 품질 및 안정성 검증



2. 악성코드 분석 방법론

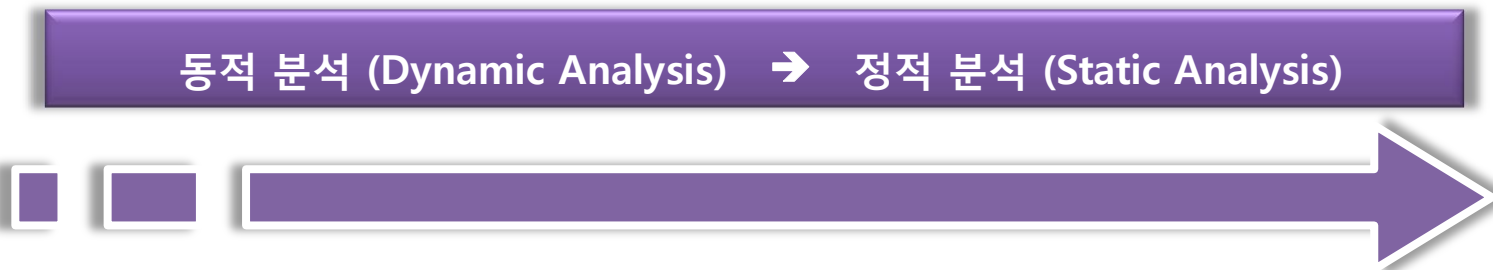
◆ Reverse Engineering Process

❖ 동적 분석 (Dynamic Analysis, System Level Reversing)

각종 툴과 운영체제의 다양한 서비스를 이용해서 프로그램 실행 파일과 입출력 값 등을 조사해서 정보를 분석하는 일련의 과정

❖ 정적 분석 (Static Analysis, Code Level Reversing)

소프트웨어 개발 및 CPU와 운영체제에 대한 깊은 이해를 바탕으로 Low Level에서 소프트웨어가 어떻게 동작하는지 분석하는 일련의 과정



2. 악성코드 분석 방법론

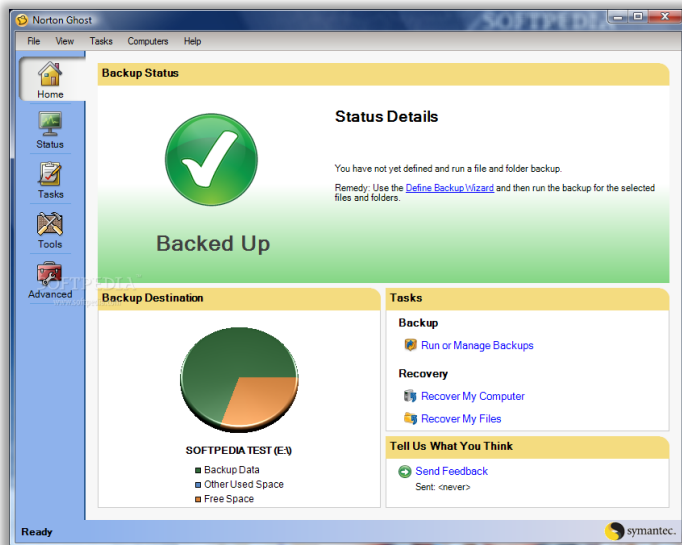
◆ System Level Reversing 환경

❖ 실제 컴퓨터 시스템 환경

시스템 복구 솔루션을 사용하여 일반 하드웨어에서 윈도우 시스템으로 외부 네트워크 단절된 독립된 시스템 구축

❖ 가상화 시스템 환경

시스템 가상화 솔루션을 이용하여 가상의 윈도우 시스템과 가상의 네트워크를 구성



2. 악성코드 분석 방법론

동적 분석 (Dynamic Analysis) → 정적 분석 (Static Analysis)

파일 분석

1. 파일 형태 분석
2. 사용 API 분석
3. 문자열 분석

증상 분석

1. 시스템 분석
2. 프로세스 분석
3. 레지스트리 분석
4. 네트워크 분석
5. 기타 증상 분석

정보 분석

1. 증상 추가 분석
2. 각종 정보 수집
3. 관련 사항 확인

코드 분석

1. 디스어셈블링
2. 디버깅

엔진 제작

1. 악성코드 판단
2. 진단 시그니처 및 함수 제작
3. 분석정보 작성

분석 프로세스

3

악성코드 분석 기법

3. 악성코드 분석 기법

◆ Reverse Engineering Tools

❖ System Monitoring Tools

리버싱 대상 어플리케이션과 동작 환경에 대해 수집된 정보들인 네트워킹, 파일 접근, 레지스트리 접근, 뮤텍스, 파이프, 이벤트 등의 정보들을 보여주는 유틸리티

❖ Disassembler와 Debugger

Disassembler - 프로그램의 실행 바이너리를 입력 받아 전체나 일부분을 어셈블리 언어 코드로 변환 해주는 프로그램

Debugger - 실행 중인 프로그램의 코드를 추적할 수 있도록 해주는 프로그램

3. 악성코드 분석 기법

◆ System Level Reversing

- 1) 각 종 툴과 운영 체제의 다양한 서비스를 이용해서 프로그램 실행 파일과 입출력 값 등을 조사해 정보를 추출하는 일련의 과정
- 2) System Level Reversing은 분석 대상이 되는 파일을 실행한다는 의미에서 동적 분석 또는 Dynamic Analysis 라고도 함
- 3) System Level Reversing의 주요 관점은 Black-box Testing과 유사
- 4) Black-box Testing 기술은 악성코드의 악의적인 기능들과 감염 기법들을 빠르게 파악하는데 유효함
- 5) Code Level Reversing과 비교하여 분석 시간은 빠르나 상세한 기능들을 파악하기는 어려움

3. 악성코드 분석 기법

◆ System Level Reversing의 주요 관점

❖ System Level Reversing에서 주요한 시스템 정보 수집 대상

File Change Monitoring

Registry Change Monitoring

Process와 Thread Monitoring

Network Port Monitoring

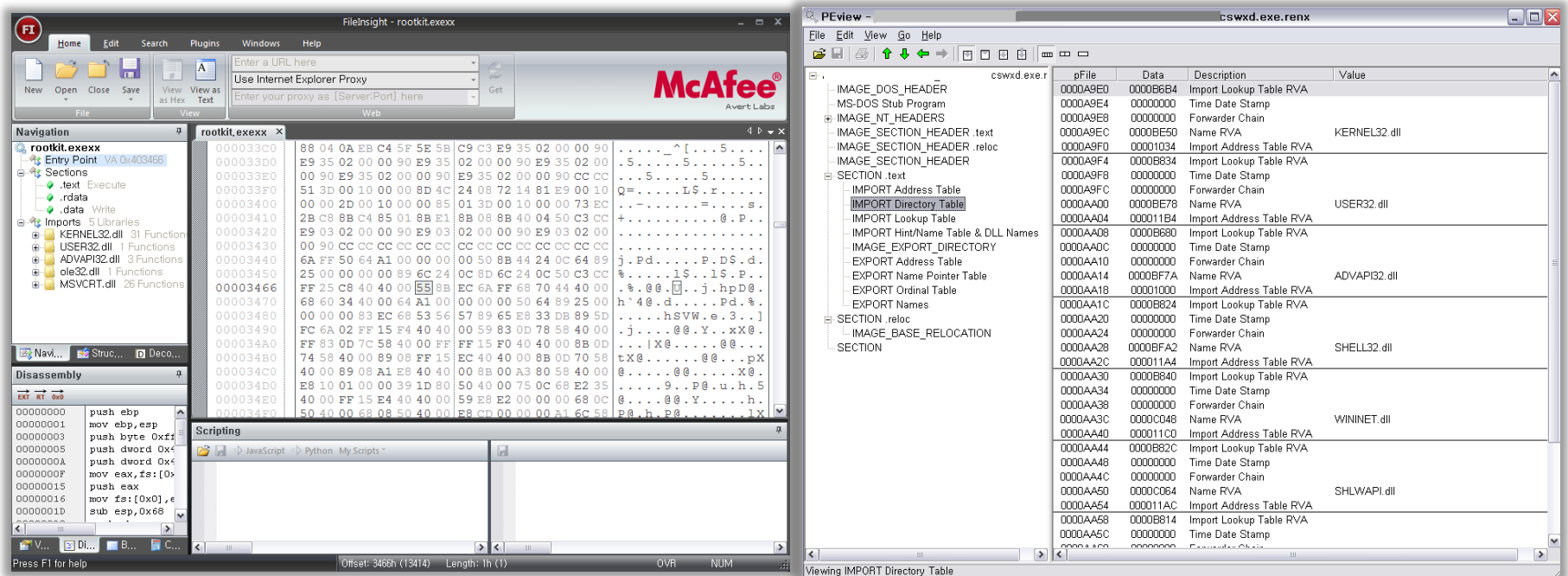
Network Sniffing과 Packet Capturing

System Call Monitoring

3. 악성코드 분석 기법

◆ PE File 분석 – Fileinsight와 PVIEW

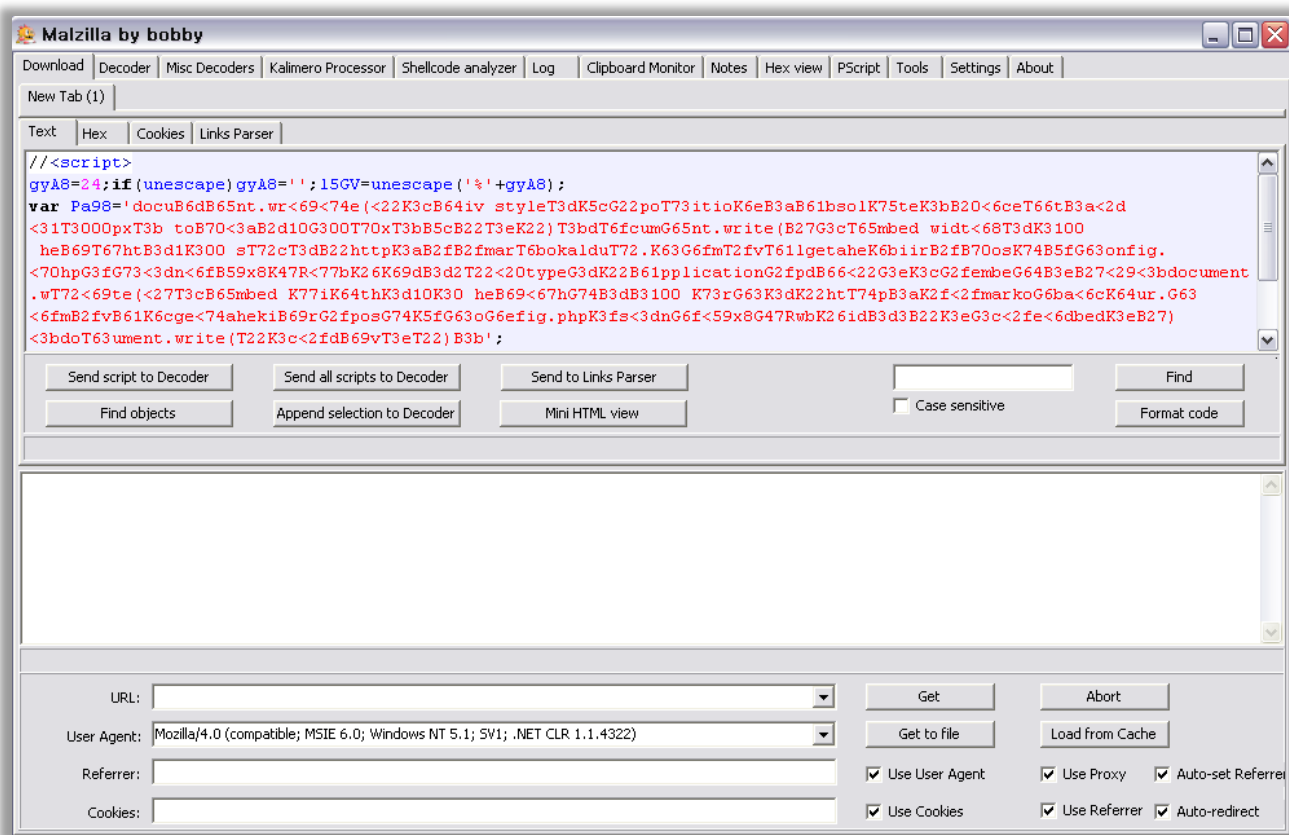
4개의 창으로 구분 지원, 구조에 따른 블록화 표시
Hex Editing, 파일 구조, EP 계산, IAT와 EAT 구분
Disassembly 지원, 파일 다운로드 기능



3. 악성코드 분석 기법

◆ Script File 분석 - Malzilla

스크립트 파일 구조 및 Decoding 분석
Shellcode 분석 및 파일 다운로드 지원



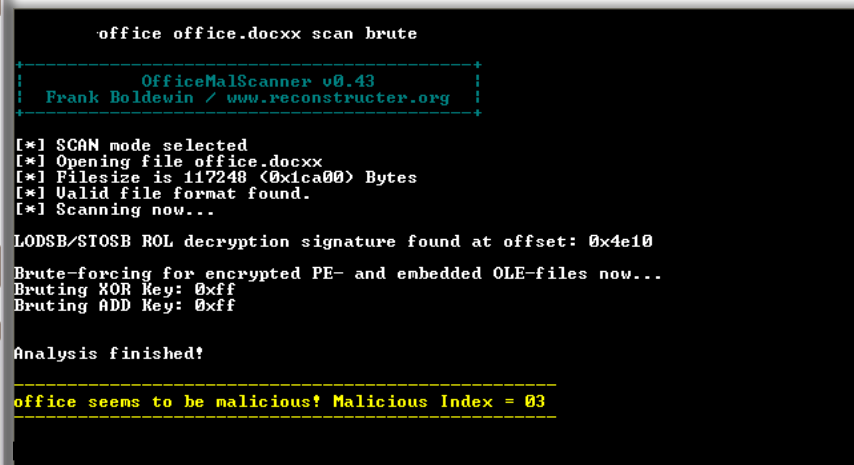
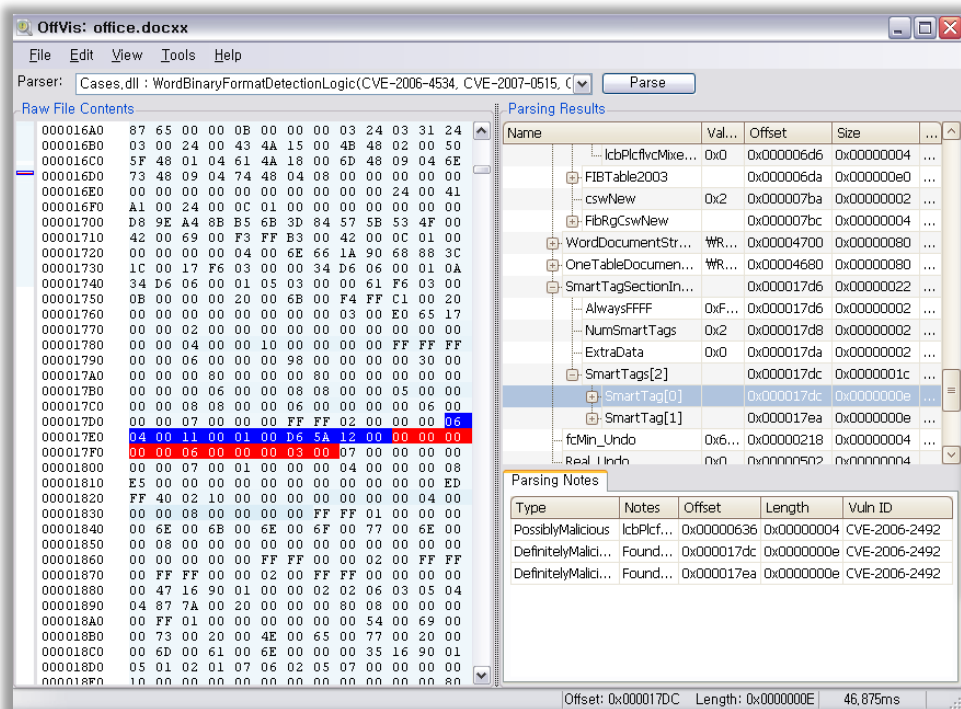
3. 악성코드 분석 기법

◆ Office File 분석 - Offvis와 OfficeMalScanner

Office 파일에 포함된 취약점 확인

Office 파일의 구조 분석

취약한 Office 파일의 파일 Offset 위치 확인



3. 악성코드 분석 기법

◆ PDF File 분석 - PDFid와 Pdftk

PDF 파일의 Zlib 압축 해제

PDF 파일 구조 중의 취약한 부분 탐색

PDF 파일 내부의 Java Script 추출

```
pdfid -s 1.aaa
PDFid 0.0.6 1.aaa
PDF Header: %PDF-1.3
obj          14
endobj       14
stream       2
endstream    2
xref         1
trailer      1
startxref    1
/Page        1
/Encrypt     0
/ObjStm      0
/JS          2
/JavaScript  3
/AA          0
/OpenAction  1
/JBIG2Decode 0
```

```
D:\work2\1>pdftk
SYNOPSIS
  pdftk <input PDF files ! - ! PROMPT>
  [input_pw <input PDF owner passwords ! PROMPT>]
  [<operation> <operation arguments>]
  [output <output filename ! - ! PROMPT>]
  [encrypt_40bit ! encrypt_128bit]
  [allow <permissions>]
  [owner_pw <owner password ! PROMPT>]
  [user_pw <user password ! PROMPT>]
  [flatten] [compress ! uncompress]
  [keep_first_id ! keep_final_id] [drop_xfa]
  [verbose] [dont_ask ! do_ask]

Where:
  <operation> may be empty, or:
  [cat ! attach_files ! unpack_files ! burst !
  fill_form ! background ! stamp ! generate_fdf
  dump_data ! dump_data_fields ! update_info]

For Complete Help: pdftk --help
```



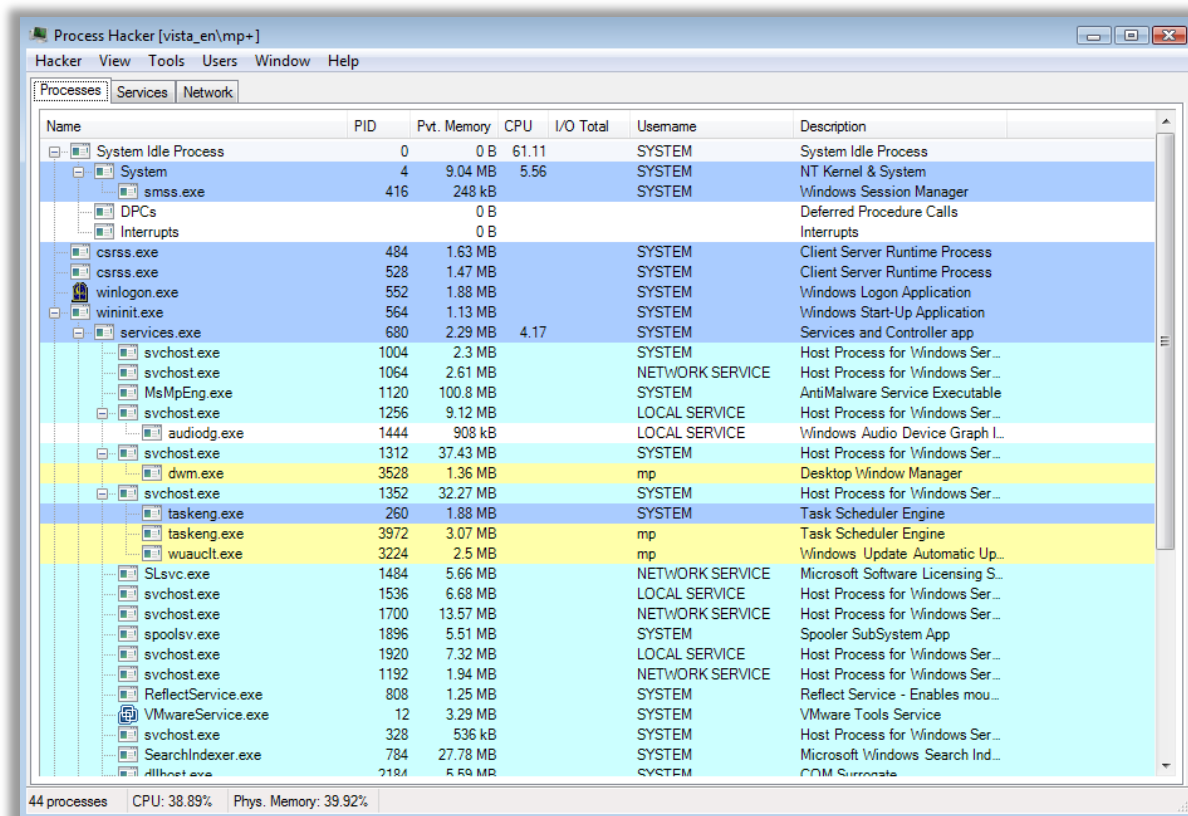
```
94 stream
95     function PkAHNt() {
96 function lhiEPAS(arg) {
97     var out = "";
98     for (var i=0; i<arg.length;i=i+4) {
99         var br1 = parseInt('0x'+arg[i] + arg[i+1], 16).toString(16);
100        var br2 = parseInt('0x'+arg[i+2] + arg[i+3], 16).toString(16);
101        if(br2.length == 1) { br2 = "0" + br2; };
102        if(br1.length == 1) { br1 = "0" + br1; };
103        out = out + "%u" + br1 + br2;
104    }
105    return out;
106 }
107 function nUM9G4HuY0xQV() {
108     Q04mL5t7CQzOV = unescape;
109     return Q04mL5t7CQzOV(lhiEPAS('414'+'14149494'+'94941494'+"94949E"+
110 ));
111     Q04mL5t7CQzOV = unescape;
112     ed81Vu = nUM9G4HuY0xQV();
113
114     var grftV1T = new Array();
115     var njzJdJ;
```

3. 악성코드 분석 기법

◆ Process, Memory 및 Thread 분석 - Process Hacker

Process, Services와 Network 실시간 모니터링

Memory Dump, String Scan, Thread와 Handle 모니터링



Name	PID	Pvt. Memory	CPU	I/O Total	Username	Description
System Idle Process	0	0 B	61.11		SYSTEM	System Idle Process
System	4	9.04 MB	5.56		SYSTEM	NT Kernel & System
smss.exe	416	248 kB			SYSTEM	Windows Session Manager
DPCs		0 B				Deferred Procedure Calls
Interrupts		0 B				Interrupts
csrss.exe	484	1.63 MB			SYSTEM	Client Server Runtime Process
csrss.exe	528	1.47 MB			SYSTEM	Client Server Runtime Process
winlogon.exe	552	1.88 MB			SYSTEM	Windows Logon Application
wininit.exe	564	1.13 MB			SYSTEM	Windows Start-Up Application
services.exe	680	2.29 MB	4.17		SYSTEM	Services and Controller app
svchost.exe	1004	2.3 MB			SYSTEM	Host Process for Windows Ser...
svchost.exe	1064	2.61 MB			NETWORK SERVICE	Host Process for Windows Ser...
MsMpEng.exe	1120	100.8 MB			SYSTEM	AntiMalware Service Executable
svchost.exe	1256	9.12 MB			LOCAL SERVICE	Host Process for Windows Ser...
audiodg.exe	1444	908 kB			LOCAL SERVICE	Windows Audio Device Graph L...
svchost.exe	1312	37.43 MB			SYSTEM	Host Process for Windows Ser...
dwm.exe	3528	1.36 MB			mp	Desktop Window Manager
svchost.exe	1352	32.27 MB			SYSTEM	Host Process for Windows Ser...
taskeng.exe	260	1.88 MB			SYSTEM	Task Scheduler Engine
taskeng.exe	3972	3.07 MB			mp	Task Scheduler Engine
wuauclt.exe	3224	2.5 MB			mp	Windows Update Automatic Up...
SLsvc.exe	1484	5.66 MB			NETWORK SERVICE	Microsoft Software Licensing S...
svchost.exe	1536	6.68 MB			LOCAL SERVICE	Host Process for Windows Ser...
svchost.exe	1700	13.57 MB			NETWORK SERVICE	Host Process for Windows Ser...
spoolsv.exe	1896	5.51 MB			SYSTEM	Spooler SubSystem App
svchost.exe	1920	7.32 MB			LOCAL SERVICE	Host Process for Windows Ser...
svchost.exe	1192	1.94 MB			NETWORK SERVICE	Host Process for Windows Ser...
ReflectService.exe	808	1.25 MB			SYSTEM	Reflect Service - Enables mou...
VMwareService.exe	12	3.29 MB			SYSTEM	VMware Tools Service
svchost.exe	328	536 kB			SYSTEM	Host Process for Windows Ser...
SearchIndexer.exe	784	27.78 MB			SYSTEM	Microsoft Windows Search Ind...
dllhost.exe	2184	5.59 MB			SYSTEM	COM Surrogate

44 processes CPU: 38.89% Phys. Memory: 39.92%

3. 악성코드 분석 기법

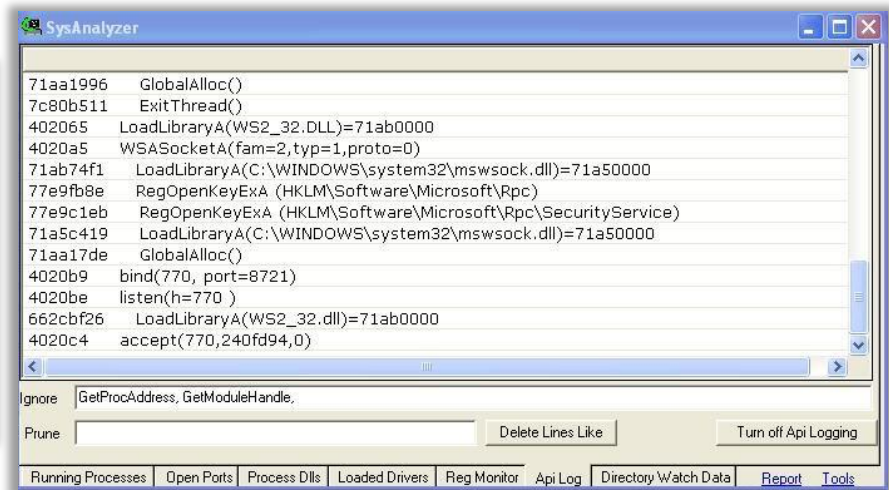
◆ File과 Registry 변화 분석 - SysAnalyzer

특정 파일에 의해 File과 Registry 변화 모니터링

특정 파일에 의해 호출되는 API 모니터링

Network Traffic 현황 모니터링

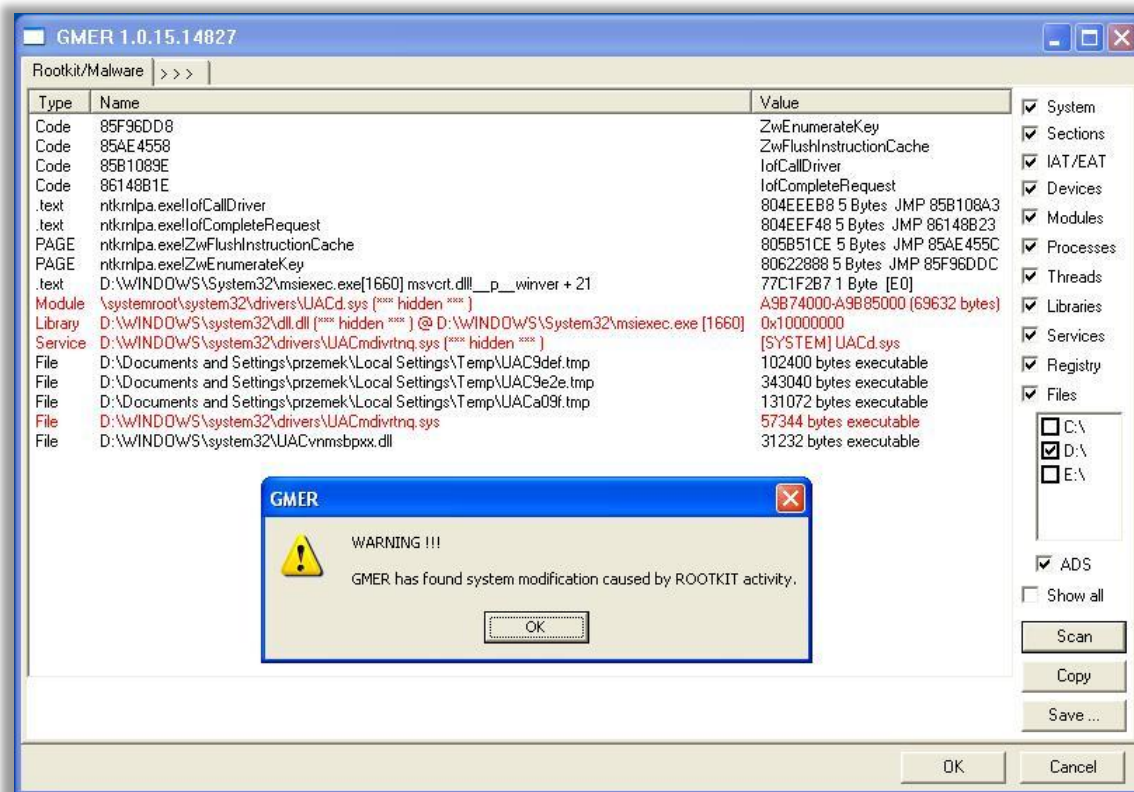
사용하는 Network Port 모니터링



3. 악성코드 분석 기법

◆ 은폐형 파일 분석 - Gmer

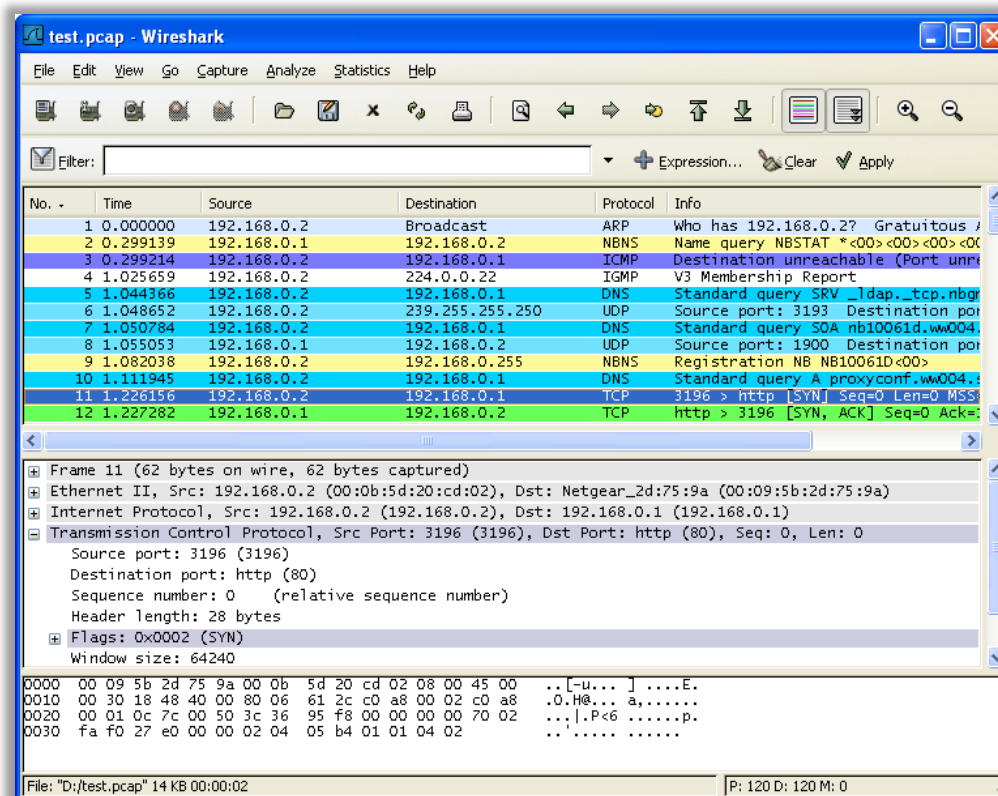
다양한 은폐 기법으로 은폐된 프로세스, 파일 및 레지스트리 분석
Process, Modules, Services와 자동 시작 부분 분석



3. 악성코드 분석 기법

◆ Network Traffic 분석 - Wireshark

시스템의 Network Traffic 실시간 분석
Protocol과 Packet 형태 분석



3. 악성코드 분석 기법

◆ Code Level Reversing

- 1) 프로그램의 설계 의도를 간파하거나 프로그램 바이너리에서 알고리즘을 파악하기 위해 소프트웨어 개발, CPU, 운영체제 등에 대한 깊은 이해를 바탕으로 정보를 프로그램 제작 목적 및 방식을 파악하는 일련의 과정
- 2) Code Level Reversing은 분석 대상이 되는 파일을 실행하지 않는다는 의미에서 정적 분석 또는 Static Analysis 라고도 함
- 3) System Level Reversing과 비교하여 더 많은 분석 시간을 요구하나 프로그램의 상세한 기능들을 파악 할 수 있음

3. 악성코드 분석 기법

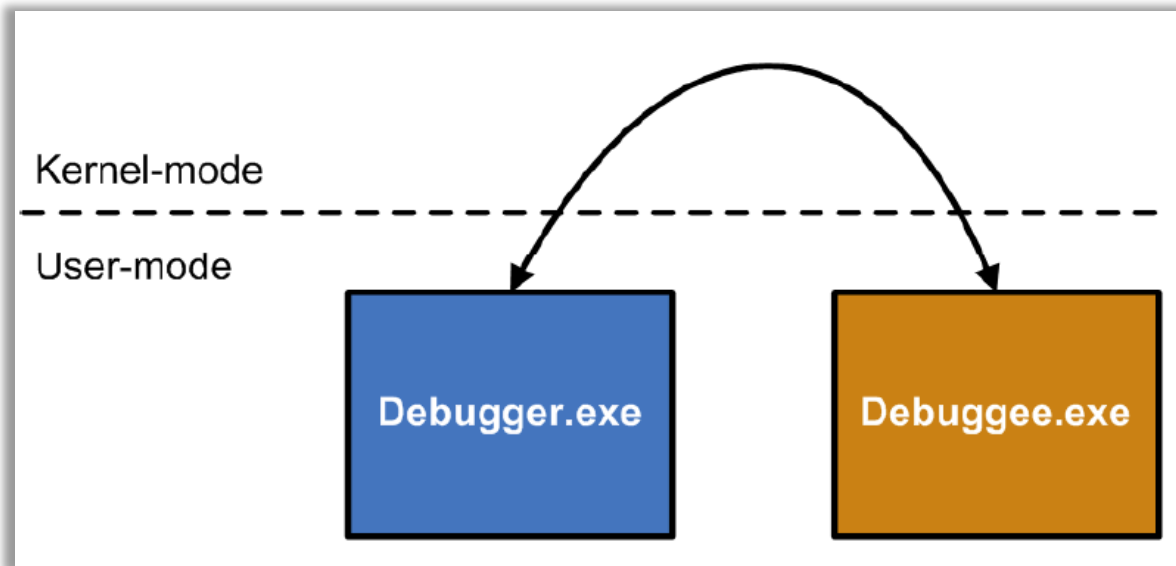
◆ Debugger

- 1) Debugger는 소프트웨어 개발자가 프로그램의 오류를 찾아내고 수정 할 수 있도록 도와주기 위한 유틸리티
- 2) 대부분의 Debugger는 소스코드 없이 어셈블리 어언 상에서 실행의 흐름을 따라갈 수 있는 기능을 제공
- 3) Software Breakpoint는 프로그램 실행 시에 디버거가 프로그램 코드 사이에 삽입하는 명령으로 그 곳에 도달하면 프로세서는 프로그램 실행을 일시 정지하고 제어 권을 디버거에게 넘김
- 4) Hardware Breakpoint는 CPU의 특별한 기능으로 어떤 특정 메모리 주소에 접근이 이루어지면 프로세서가 프로그램 실행을 일시 정지하고 제어권을 디버거에게 넘김
- 5) Debugger는 크게 User Mode Debugger와 Kernel Mode Debugger로 구분

3. 악성코드 분석 기법

◆ User Mode Debugger

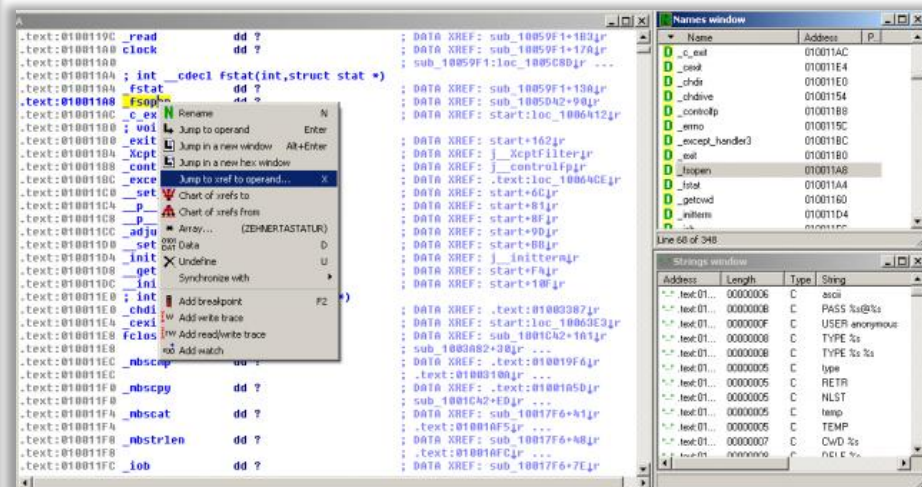
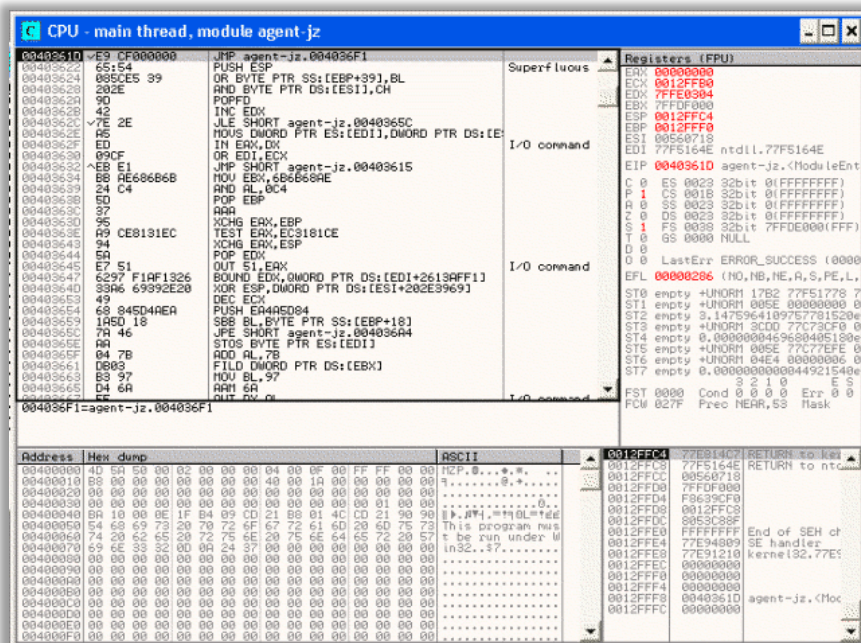
- 1) User Mode Debugger는 일반적인 User Mode 어플리케이션을 디버깅하기 위해 사용
- 2) User Mode Debugger는 디버깅 대상(Debuggee) 프로세스에 붙어 해당 프로세스에 대한 모든 제어를 수행할 수 있는 전통적인 어플리케이션
- 3) User Mode Debugger는 Kernel Mode Debugger와 달리 디버거 자체가 시스템 상의 프로그램임으로 설치 및 사용이 간편



3. 악성코드 분석 기법

◆ User Mode Debugger 어플리케이션

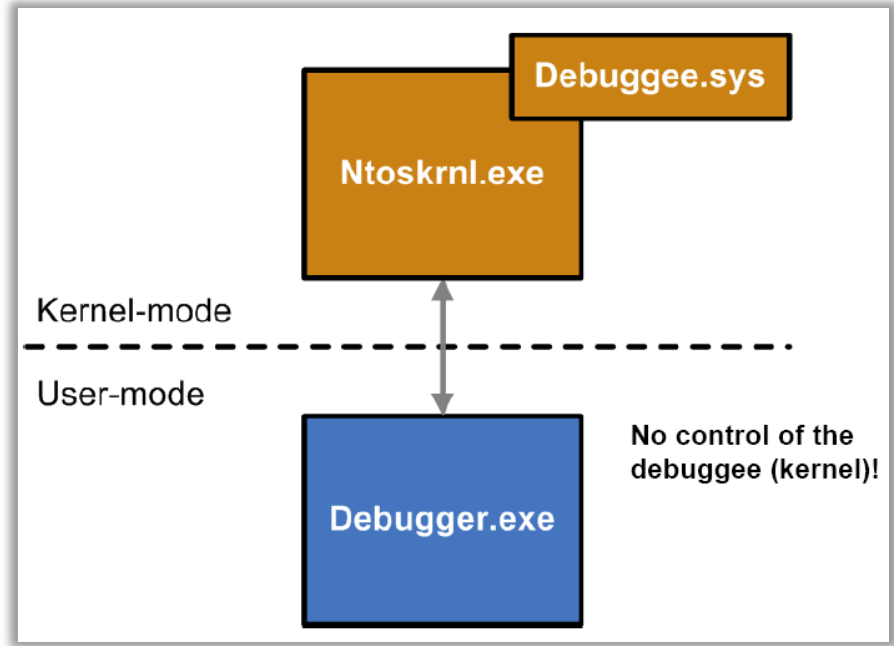
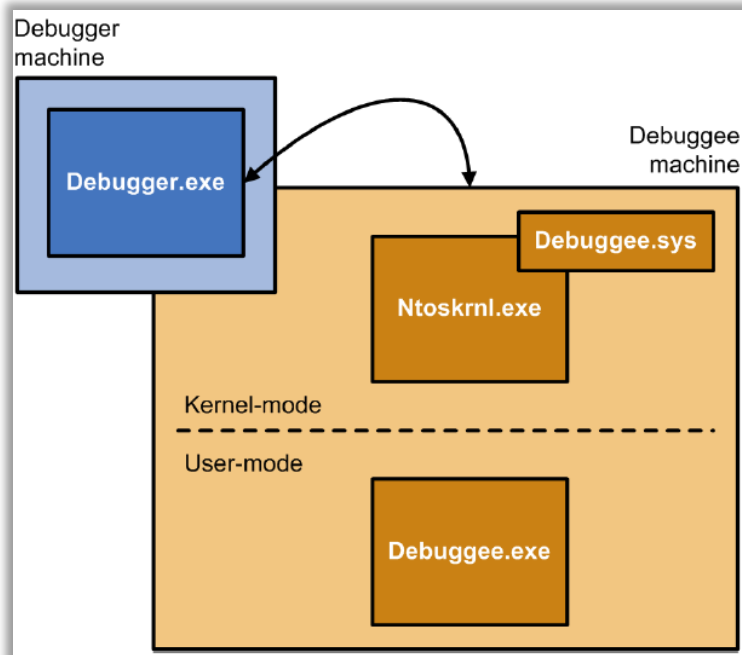
- 1) OllyDbg는 리버싱을 위해서 설계된 디버깅 프로그램
- 2) WinDbg는 마이크로소프트에서 개발한 디버깅 프로그램 커맨드 라인 인터페이스 제공
- 3) IDA Pro는 강력한 디스어셈블리어인 동시에 유저 모드 디버깅 프로그램



3. 악성코드 분석 기법

◆ Kernel Mode Debugger

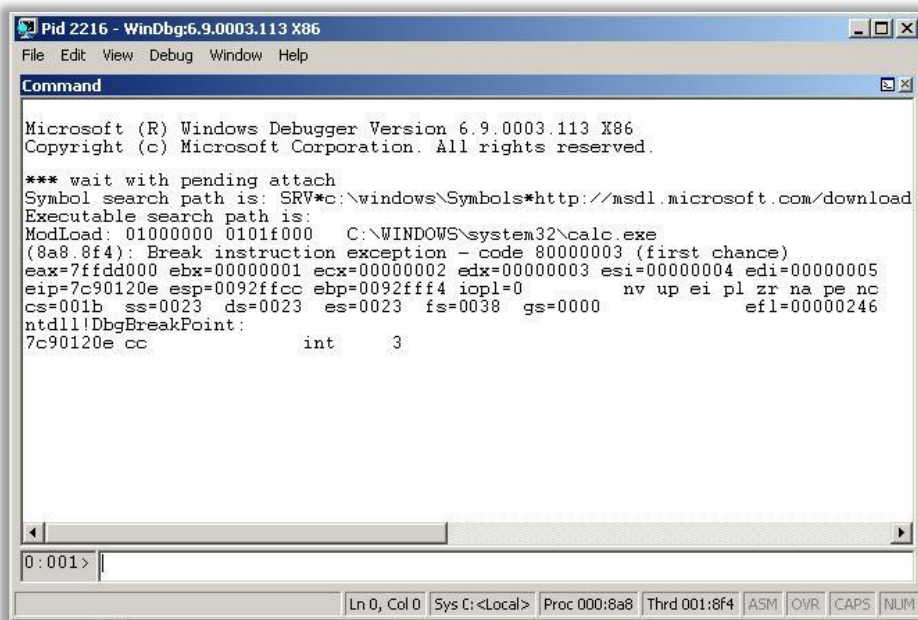
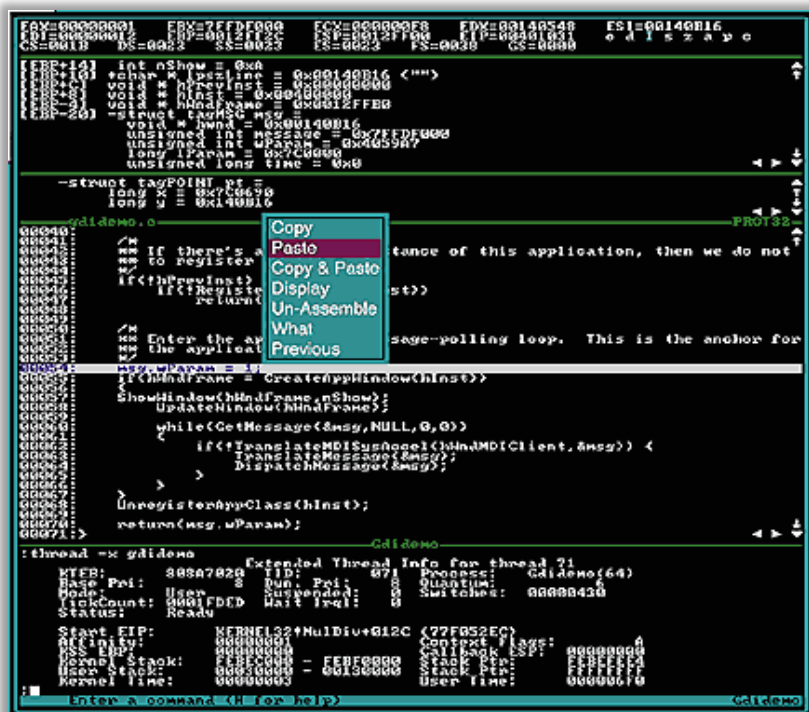
- 1) Kernel Mode Debugger는 대상 시스템 전체를 제어 할 수 있으며 시스템에서 발생하는 어플리케이션 코드와 운영체제 코드 내부에서 발생하는 모든 것을 볼 수 있음
- 2) Kernel Mode Debugger는 운영체제에서 실행되는 하나의 어플리케이션이 아니라 시스템의 커널과 동등한 컴포넌트로 존재하여 전체적인 시스템을 관찰하거나 실행을 중지 할 수 있음



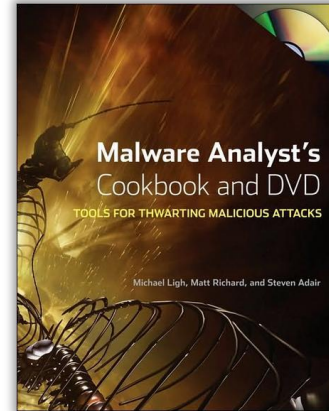
3. 악성코드 분석 기법

◆ Kernel Mode Debugger 어플리케이션

- 1) SoftICE는 윈도우 디바이스 드라이버 개발 툴로 개발되어 로컬 커널 디버깅 수행
- 2) WinDBG는 커널 모드 디버깅이 가능한 원격 커널 모드 디버깅 수행



Reference



- 1) The Art of Computer Virus Research and Defense
- 2) 리버싱 : 리버스 엔지니어링 비밀을 파헤치다
- 3) Malware Analyst's Cookbook and DVD : Tools for Thwarting Malicious Attacks
- 4) 소프트웨어 보안 코드 깨부수기

감사합니다

세상에서 가장 안전한 이름

Ahn 안철수연구소