보안공학연구논문지(JournalofSecurityEngineering), 제 9 권
제 1 호 2012 년 2 월

공 격 용 툴 킷 및 변 형 코 드 의 유 사 성 기 준 선 정

정 용 욱 1), 노 봉 남 2)

Selectingfeaturesformeasuringsimilaritybetween attack toolkits and polymorphiccodes

Yong-WookChung[1],Bong-Nam Noh[2]

요 약

악 성 코 드 는 악 의 적 인 목 적 으 로 타 인 의 컴 퓨 터 에 영 향 을 끼 칠 수 있 는 프 로 그 램 으 로 최 근 에 는 대 부 분 의 악 성 코 드 들 이 최 초 공 개 된 소 스 에 기 반 하 여 그 감 염 방 법 과 증 상 들 이 다 양 해 지 는 변 형 코 드 로 발 전 하 며 진 화 하 고 있 다 . 실 시 간 으 로 변 형 되 는 악 성 코 드 를 기 존 에 생 성 된 바 이 러 스 시 그 니 처 로 탐 지 하 기 어 려 운 한 계 점 에 도 달 하 게 되 었 고 , 난 독 화 및 암 호 화 된 코 드 에 대 한 정 적 분 석 문 제 점 을 극 복 하 기 힘 든 어 려 움 으 로 인 하 여 동 적 분 석 방 법 에 대 한 많 은 연 구 가 이 루 어 져 왔 다 . 본 논 문 에 서 는 동 적 분 석 과 정 적 분 석 에 대 한 특 징 , 형 태 및 문 제 점 을 분 석 하 고 주 요 특 징 별 로 분 류 하 여 악 성 코 드 유 사 성 에 대 한 기 준 을 제 안 하 고 실 험 을 통 하 여 유 사 성 규 칙 의 정 확 도 를 보 인 다 . 생 성 된 규 칙 을 이 용 하 여 알 려 진 공 격 용 툴 킷 및 변 형 코 드 에 대 한 공 통 된 유 사 성 을 분 석 하 고 동 일 여 부 를 판 별 할 수 있 을 것 으 로 기 대 된 다 .

핵 심 어 : 공 격 용 툴 킷 , 악 성 코 드 , 정 적 분 석 , 동 적 분 석 , 유 사 성

Abstract

Others for the purpose of malicious code, malicious computer program that can have an impact on the most recently published their first source of malicious code based on the symptoms of the infection and how they developed into a diverse and variant code has evolved. The malicious code in real-time deformation generated previously difficult to detect the virus signature threshold is reached, and to code obfuscation and encryption for static analysis is difficult to overcome the problems due to the difficulty, many studies on the dynamic analysis method consists of have. In this paper, for dynamic analysis and static analysis features, form and analysis of issues and classified by the main characteristic malicious code based on similarities to suggest the accuracy of the experiment seems to rule through a similarity.

By rules created and transformed known attack code for the toolkit to analyze the similarities common to determine whether the same is expected.

Keywords : Attack toolkits, malicious code, static analysis, dynamic analysis, the similarity

접수일(2011 년 12 월 01 일), 심사의뢰일(2011 년 12 월 02 일), 심사완료일(1 차:2011 년 12 월 17 일, 2 차:2012 년 01 월 02 일) 게재일(2012 년 02 월 29 일)

[1] 500-757 광주 광역시 북구 용봉로 77 (용봉동), 전남대학교 정보보호협동과정. email:kit1989@paran.com [2](교신저자)500-757 광주 광역시 북구 용봉로 77(용봉동), 전남대학교 시스템보안연구센터 교수 email:bbong@jnu.ac.kr

1. 서론

악 성 코 드 로 인 하 여 국 내 에 서 는 수 능 시 험 자 료 를 유 출 하 는 사 고 , 미 국 대 통 령 등 유 명 인 사 들 의 트 위 터 계 정 해 킹 사 고 , 온 라 인 게 임 업 데 이 트 과 정 에 서 DDoS 공 격 을 일 으 킨 사 건 , 로 이 터 통 신 을 가 장 하 여 바 이 러 스 를 유 포 한 사 건 , 815 광 복 절 에 맞 춘 한 일 네 티 즌 간 커 뮤 니 티 사 이 트 대 상 DDoS 공 격 사 건 , 트 위 터 및 페 이 스 북 등 유 명 네 트 워 크 사 이 트 일 시 중 단 사 건 등 헤 아 릴 수 없 을 만 큼 다 양 하 다 . 과 거 에 는 백 신 사 에 서 수 작 업 으 로 상 세 분 석 하 면 서 악 성 코 드 의 목 적 에 맞 는 이 름 과 유 형 을 지 정 하 여 왔 지 만 , 매 일 새 롭 게 등 장 하 는 수 많 은 악 성 코 드 들 을 직 접 분 석 하 기 는 현 실 적 으 로 불 가 능 하 며 경 제 적 이 지 못 하 면 서 복 잡 해 지 고 있 다 .

특 히 다 양 한 루 트 킷 , 백 도 어 , 드 로 퍼 , 다 운 로 더 기 능 을 하 나 씩 포 함 하 다 가 웜 으 로 확 산 되 는 성 향 으 로 가 면 서 분 류 하 기 가 갈 수 록 어 려 워 지 고 있 다 . 악 성 코 드 개 발 자 는 널 리 알 려 진 운 영 체 제 , 업 무 용 프 로 그 램 , 유 틸 리 티 등 의 취 약 점 을 계 속 해 서 찾 아 내 고 있 으 며 공 격 용 툴 킷 에 기 존 악 성 코 드 의 형 태 를 변 형 시 켜 취 약 점 을 이 용 한 변 종 을 만 들 어 내 고 있 기 때 문 이 다 . 일 반 사 용 자 는 사 용 하 고 있 는 컴 퓨 터 에 의 심 되 는 악 성 코 드 를 백 신 으 로 확 인 한 후 악 성 코 드 의 행 위 를 상 세 분 석 할 때 Anubis , BitBlaze , Comodo , CWSandbox , EUREKA , Joebox , NormanSandBox , ThreatExpert , Xandora 등 의 온 라 인 분 석 사 이 트 를 사 용 할 수 있 는 데 업 로 드 용 량 이 제 한 되 어 있 고 행 위 기 반 에 국 한 되 어 있 어 신 속 한 정 적 분 석 이 불 가 능 하 다 . 현 재 백 신 업 체 및 분 석 담 당 자 들 은 수 집 된 악 성 코 드 샘 플 을 분 석 시 스 템 에 서 정 적 분 석 을 수 행 한 후 개 별 적 으 로 수 동 분 석 을 수 행 하 고 있 다 . 결 과 보 고 서 및 분 석

샘 플 은 분 석 시 스 템 의 저 장 매 체 에 확 장 자 를 바 꾸 거 나 암 호 화 압 축 형 태 로 저 장 되 어 관 리 하 고 있 다 . 악 성 코 드 에 대 한 정 적 · 동 적 분 석 에 대 응 할 수 있 는 기 술 인 력 은 한 정 되 어 있 는 반 면 , 악 성 코 드 를 이 용 한 공 격 은 전 세 계 적 으 로 월 평 균 2 억 4 **천** 5 백 만 건 씩 탐 지 되 고 있 다 . 감 염 된 **1 l -** 일 의 **A** 료 목 적 을 위 한 백 신 진 단 이 **- 9 1** 공 격 용 툴 킷 에 대 한 상 세 분 석 을 통 해 개 발 자 의 특 징 , 고 유 속 성 , **원 천** 소 스 위 **A** 등 을 **1 l -** 악 하 기 위 해 서 는 변 형 되 는 악 성 코 드 를 판 별 하 기 위 한 유 사 성 에 대 한 기 준 이 **V .** 요 하 다 [1 , 2 , 3] .

최 신 악 성 코 드 는 국 내 **- 9 1** 백 신 과 **침 입** 탐 지 시 스 템 으 로 부 터 은 **닉** 하 도 록 난 독 화 **또** 는 암 호 화 코 드 로 상 시 변 경 하 여 공 격 용 툴 킷 을 통 하 여 전 **1 l -** 시 **1** 고 있 다 . 공 격 용 툴 킷 을 사 용 하 는 목 적 은 단 기 간 에 다 양 한 기 능 을 **4 - -** 가 시 켜 공 격 할 수 있 는 악 성 코 드 를 생 성 하 여 백 신 업 체 의 **6 3 .** 진 업 데 이 트 및 보 안 시 스 템 담 당 자 가 네 트 워 크 보 안 **V .** 터 **링** 규 칙 을 새 로 적 용 하 기 전 에 전 **1 l -** 시 켜 **피** 해 확 산 도 를 최 대 로 **? = -** 이 는 데 있 다 . 악 성 코 드 개 발 자 가 생 성 한 악 성 코 드 **1 l -** 일 안 에 는 **닉** 네 임 , 개 발 **V -** 경 , **설** 정 경 로 , 안 티 **M 1 = 1 1 7 c 1** 정 보 등 공 격 용 툴 킷 이 용 정 보 를 포 함 하 고 있 는 경 **우** 가 많 다 .

본 논 문 에 서 는 악 성 코 드 를 실 행 하 지 **a** 고 C h e c k s u m , P a c k i n g 을 통 한 **1 l -** 일 의 변 경 에 구 **6 1 1 '1 4** 지 **a** 는 정 적 상 태 , 악 성 코 드 를 실 행 할 때 **P l s** 리 에 남 아 있 는 **c L i g l l** 킹 상 태 , 그 리 고 , 네 트 워 크 상 에 서 악 성 코 드 가 주 고 **' 1 4** 는 **쿼** 리 등 을 통 해 유 사 성 판 단 의 기 준 점 을 제 안 하 **였** 다 . 공 격 용 툴 킷 에 대 한 정

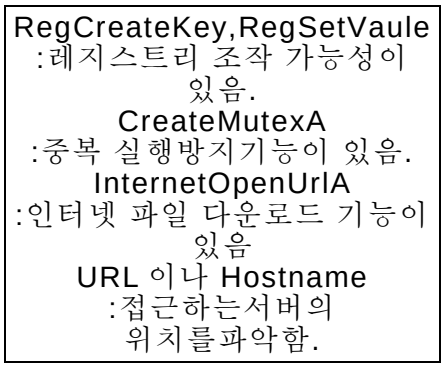
적·동적 분석을 위하여 툴킷의 특징으로부터 점수화 기법을 사용하였고, 제안한 기준점의 정확도를 실험을 통하여 확인하였다.

본 논문의 구성은 다음과 같다. 2 장에서는 기존의 악성코드 유사성 정적·동적 분석에 관한 연구들을 살펴보고, 3 장에서는 툴킷을 통한 실험과 기준속성을 선정하고 본 논문에서 제안하는 유사성 기준속성들에 대한 점수표 기반의 유사성 비교 정책을 수립한다. 마지막으로 결론과 향후 연구에 대해 기술한다.

2. 관련연구

2.1 정적 V:. 1 71V: *사 M VI

본문 악성코드를 실행하기 전에 정적 분석을 우선적으로 하는데 이진 실행파일에 대한 삽입된 문자열 분석을 하는 Strings, BinText 를 이용하여 문자열을 추출하여 분석하게 된다. 문자열 분석을 통하여 Import 함수 이름을 알 수 있고 악성코드의 특성을 파악할 수 있다.



[그림 1]문자열분석예제
[Fig.1]Stringanalysis

또 한 , 기 타 문 자 열 (P I N G , P O N G , P R I V M S G , J O I N) 을 통 해 악 성 코 드 가 특 정 프 로 토 콜 로 통 신 을 할 가 능 성 , 컴 파 일 러 및 신 용 카 드 관 련 정 보 , 옵 션 이 나 에 러 메 시 지 등 의 스 트 링 을 통 해 다 양 한 정 보 를 분 석 할 수 있 다 . 정 적 분 석 유 사 성 비 교 방 법 에 서 는 바 이 너 리 패 턴 매 칭 , 데 이 터 플 로 우 등 실 제 악 성 코 드 를 실 행 하 지 않 고 안 전 하 게 빠 른 분 석 을 할 수 있 다 는 장 점 을 가 지 고 있 다 . 그 러 나 , 동 적 분 석 과 달 리 상 세 분 석 에 필 요 한 숙 련 된 분 석 능 력 이 요 구 되 고 몇 가 지 가 시 적 특 성 만 을 판 별 하 게 되 어 오 탐 (f a l s e p o s i t i v e) 이 발 생 할 수 있 는 한 계 를 지 닌 다 [4] .

M e t a m o r p h i c 악 성 코 드 를 탐 지 하 기 위 한 코 드 그 래 프 에 서 는 M S W i n d o w s 운 영 체 제 환 경 의 공 용 라 이 브 러 리 를 호 출 하 여 사 용 하 는 I m p o r t 함 수 , 자 체 라 이 브 러 리 의 E x p o r t 함 수 , 코 드 내 부 의 사 용 자 함 수 등 에 서 순 차 적 수 행 원 리 의 연 관 관 계 를 코 드 그 래 프 로 표 현 하 고 유 사 성 을 비 교 하 고 있 다 [5] .

F-secure 사에서는보다 진보된악성코드분석기법으로,그래프의 유사도를 이용하여 악성코드의 차이점을발견하는방법을 제안하였다[6].악성코드의실행파일에서 함수 및 라이브러리 등 공통요 소를 행렬로 도출하고 함수 특징별로 시그니처를 생성하는데 Call-treesignature 방식을 사용한다 하지만,악성코드 자체가 아닌사용된함수의 유사도를기반으로분석하여변종여부를 판단하는데 사용하는 제한이따른다.

Microsoft 사에서제안한 자동화된 바이러스 분류 시스템은 악성코드자체와정상 프로그램에삽 입된 악성코드분석으로 구분되어있으며 Tracing 단계를 거쳐추출된 악성코드의 흐름을 Control Flow Graph 와 BasicBlock 으로 구분하여악성코드를비교하고 있다[7].이 방법의단점으로는 BasicBlock 간의거리를 구하는데사용된 Bloom Filter 알고리즘에서오탐이 발생하게 된다.또한, 코드 그래프를이용하는방법에서 있어,행위결과에 기반을 둔 동적 분석에대한악성코드 유사성 은 매번 실행해서 결과 값을 얻어야하는 번거로움이 있으며분석관의기술정도에따라분석시간이 유동적이다.이에비해정적분석은악성코드의 실행유무와 관계없이신속히 유사성을 비교할 수 있다.

22 동적 V.-.1 71v1 유사도 평 71-

현재까지알려진 악성코드유사성분석방법으로는크게 동적분석(dynamicanalysis)과 정적분석(staticanalysis)으로 나누어져연구되고있다[8].동적 분석에서는모니터링툴을 사용하여악성 코드가 접근 또는변경하는리소스를관찰하기 위해 Filemon,Regmon,ProcessMonitor,TDIMon

등의 실시간 모니터링 도구들을 이용할 수있다. 또한 WinAnalysis, RegShot, Autoruns 등의 도구 들을 이용하여 변경된레지스트리 및 자동실행여부를분석할 수있다. 특정 shadowbot 의경우 실행중인 explorer.exe 의자식프로세스를 모두 소멸시키므로 결과적으로모니터링 도구들이 모두 kill 되어정상적인 모니터링이힘든경우도 있다. 동적분석 기반의 유사성 분석방법에서는 대표 적으로 APIcall 함수의 순차적 행위분석을 통하여 악성코드 여부를탐지하게된다[9]. 실행압축 되거나난독화된 악성코드의경우, 메모리상에서 실제 행위를 볼 수 있는장점을 가지고 있다. 또 한 상위레벨수준의 프로세스의 개수, 파일 및 레지스트리 변경, 네트워크 행위 등을 비교하여악 성코드의 특징에 따른 유사성을나타낼 수있다. 그러나, 모듈단위로 공유된 악성코드에 대해서는 낮은 유사성을 보이는 단점을 지니기도 한다.

DanielBilar 은통계구조를 통한악성코드구분 및분석에 필요한 Fingerprinting 을위해서정적 분석으로기계어 명령어별 사용빈도를 정상 프로그램과악성코드간의비교를 통해 차이점을 확인하고
있으며, 동적분석으로 Win32API 호출함수를악성코드 별로분석하여유사도를평가하고 있 다[10]. 또한 정적분석으로 SDG (System DependenceGraph)를사용하여 악성코드별제어및 데 이터의존도 특징등을비교한다. 하지만 악성코드개발자가사용하는명령어구조는백신을 우회 하기위한다양한방법 등으로 상시적인 변경가능성이 있어 명령어구조업데이트가요구된다.

그리고, 악성코드 유사도 평가를 하기 위한 추가적인 빈도별 통계 모델링 작업을 위해서는 비교적 긴 측정 시간이 소요된다.

3. 정적·동적 기반 유사성 기준

3.1 속성 선정

공격용 툴킷을 통해 선정한 기준점은 전체 5 가지 범주(개발국가, 네트워크, 개발언어, 포렌식, 이진파일) 안에 11 가지 속성으로 구성되어 있다.

국가코드

국가코드는 정적분석을 할 때 지역값에 의해 툴킷이 사용하고 있는 메뉴언어를 결정하는 기준점으로 사용한다. 중국어가 지역값일 때 "중국어로 'PRC' 스트링이 필요함." 라고 할 수 있으며, 지역값은 국가별 언어뿐 아니라 각 언어 안에서도 지역구분을 지원한다. 예를 들면, 국가언어별 반환값으로 MSDN 에서 다음과 같이 언어별 코드페이지를 찾아볼 수 있다.

```
0x0404Chinese(Taiwan)
0x0804Chinese(PRC)
0x0409English(UnitedStates)
```

[그림 2]국가코드예시
[Fig.2]Nationalcode

원격 동적 라이브러리

원격 동적라이브러리는툴킷에서변형 악성코드의설치방법을설정시 RemoteRegistry 서비스로 사용한다. 원격 동적라이브러리 이름및값의 변경 유무를

하나의 기준점으로 사용한다. 예를 들면 실제 regsvc.dll 을 툴킷을 통해 원격 동적 라이브러리인 regsvr.dll 로 변경하여 생성한 것을 말한다.

제어 및 명령

제어 및 명령은 노트북인지 데스크탑인지에 대한 쿼리, DNS 에 대한 접속 쿼리 및 실시간 데이터, USB 및 네트워크 상태, 시스템에 설치된 모든 드라이버 개수와 같은 감염 호스트 정보 등을 수집하는 데 사용하는 제어 명령어를 말한다. 바이너리 파일 내용에 암호화 되거나 난독화 된 제어 명령어들은 메모리상에서는 복호화 되어 정상적으로 처리되기 때문에 악성 코드 분류의 기준점으로 사용될 수 있다. 예를 들면, Operation Aurora 에서는 사용자 프로토콜을 사용하여 명령어를 숫자로 표시하고 있다.

컴 파 일 러 환 경

M i c r o s o f t V i s u a l S t u d i o 에 서 실 행 파 일 에 대 한 컴 파 일 러 환 경 이 정 적 또 는 동 적 연 동 런 타 임 라 이 브 러 리 선 택 , S t a n d a r d T e m p l a t e L i b r a r y 사 용 , 구 버 젼 i o s t r e a m 라 이 브 러 리 사 용 유 무 , 컴 파 일 러 별 임 의 함 수 및 변 수 이 름 을 변 경 하 는 것 들 이 판 단 의 기 준 점 이 된 다 .

Manifest 정보

M a n i f e s t 정 보 는 응 용 프 로 그 램 이 나 어 셈 블 리 안 에 포 함 되 거 나 외 부 X M L 파 일 로 제 공 되 는 X M L 문 서 다 . 이 름 , 설 명 , 플 랫 폼 , 독 립 모 듈 리 스 트 , 버 전 , 특 정 모 듈 인 식 토 큰 값 , 공 개 키 / 사 설 키 등 은 유 사 성 에 대 한 판 단 기 준 점 이 된 다 .

```
<assemblyIdentity
type="win32"
name="Microsoft.Windows.Common-Controls"
version="6.0.0.0"
processorArchitecture="*"
publicKeyToken="6595b64144ccf1df"
language="*"
/>
```

[그림 3]Manifest 정보 예시
[Fig.3]Manifestinformation

디 버 깅 정 보

PDB 파일의 버전,디버깅 횟수,Standardprogram database/Program Databaseforeditand continue/C7compatible 형식정보는 파악하는기준점이된다.

안 티 디 버 그

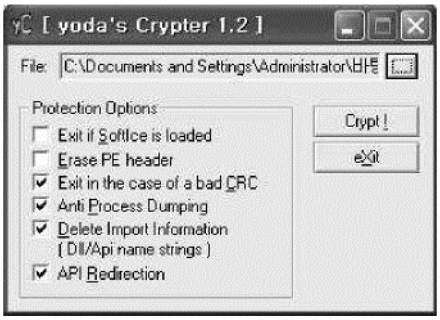
SEH handler,zeroerror 나누기,,IsDebuggerPresent,HeapManipulationFlags, NtQueryInformationProcess,RemoteDebuggerPresent,SetUnhandledExceptionFilter 등의 안티디버 깅 코드로 파악하는 기준점이된다.

타 임 라 인

타임라인에는 실행파일 헤더에위치한모듈 시간,PDB 확장자를가진디버그 시간을 파악하는 것은 유사성에대한 판단기준점이된다.예를들면 32 비트 파일의경우 국제표준(UTC)1995 년 1 월 1 일은 (0x2F05F080),2012 년 1 월 1 월 1 일은 (0x4EFFA200)로 표시된다.사용함수는 ctime()이사용된다.

암호화/
난독화
36

공격용 툴킷을통해 제작되는 변형 코드는악의적인문자열, 취약점에대한 공격코드를쉽게인 지할 수 없도록 문자열 split 을 이용, XOR 인코딩 방식을 이용, 공격자 자체 함수에의한인코딩을 이용, 8-bitASCII 인코딩 값을 이용, BASE64 인코딩방식을이용, 알려진 암호화/난독화 도구 등을 이용한 우회 기법 등을 적용하게 되는데암호화/난독화 전후의패턴은 유사성에대한판단 기 준점이된다.



[그림 4]알려진 암호화/난독화 도구 정보 예시
[Fig.4]Knownencryption/obfuscationtoolinformation

파일경로

툴킷을 컴파일할때 사용했던 경로, 디버깅경로, 디바이스 드라이버경로 등을통해 툴킷의원 래 이름및 2 차실행파일이름을파악하는기준점이된다. 예를 들면, windows\ i386\ RESSDT.pdb 파일경로를 통해 i386 폴더에 설치및 Windowsbuild 관련파일을 포함하고있음을 알 수 있으며 SSDT(SystemServiceDescriptorTable) 파일명을통해 루트킷과후킹관련드라이버 파일이 2 차로 사용되고있음을파악할 수 있다.

ASCII 문자열

문자열관련 키워드(s,%d 등),에러 메시지(Unabletodetermine,Unknowntype 등)은툴킷 이 복사한원천소스이름을파악하는기준점이 된다.

3.2 점수 계산방법

본 논문에서는 11 개의 기준 속성에 대하여 가각의 정량적인배점기준을 중요도 순으로정하고 이에따른 점수를부여하였다.각 항목별 주어진 점수를 만점으로 총합계가 100 점기준으로 점수 를 부여한다.툴킷의 MD5 해쉬값과 Checksum 을우선적으로 비교하며이것이 동일하면 거의흡 사한 악성코드이며 100%이상의 유사도를보여준다.공격용툴킷 정적분석에서는위와 같은 속성점 수에 근거하여 툴킷들에 대한유사성을 비교한다.유사성비교를통하여동일한 공격유형별로분

류 가 능 하 며 이 를 통 한 개 발 자 의 성 향 도
파 악 이 가 능 할 것 이 다

[표 1]점수표
[Table1]Scorecard

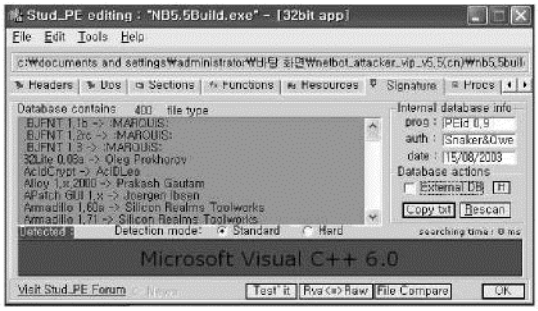
속성		점수
개발국가	국가코드	20
	합계	20
네트워크	원격 동적 라이브러	10
	제어 및	10
	합계	20
개발언어	컴파일러	10
	Manifest 정	5
	디버깅	5
	합계	20
포렌식	안티	5
	타임 라인	5
	암호화/난독	10
	합계	20
이진파일	파일경로	
	ASCII 문자	10
	합계	20
총 합계		100

예 를 들 면 , 정 적 분 석 을 통 하 여 국 가 코 드 인 중 국 (0 x 0 4 0 4) , 원 격 동 적 라 이 브 러 리 인 r e g s v r . d l l , 숫 자 로 변 환 된 제 어 와 명 령 코 드 , 컴 파 일 러 환 경 등 이 들 어 있 는 공 격 용 툴 킷 의 점 수 는 총 5 0 점 (= 2 0 + 1 0 + 1 0 + 1 0) 을 가 지 게 된 다 . 그 리 고 , 4 가 지 기 준 속 성 을 만 족 하 거 나 총 5 0 점 을 만 족 하 는 공 격 용 툴 킷 에 서 유 사 도 를 비 교 하 고 이 와 흡 사 한 기 준 을 추 출 하 게 된 다 . 속 성 별 점 수 비 교 에 의 해 유 사 도 를 확 인 하 며 최 종 점 수 를 통 해 공 격 용 툴 킷 개 발 자 의 성 향 을 파 악 하 는 기 준 으 로 도 사 용 할 수 있 을 것 으 로 기 대 한 다 .

3.3 -E-Ve *V: 'ati

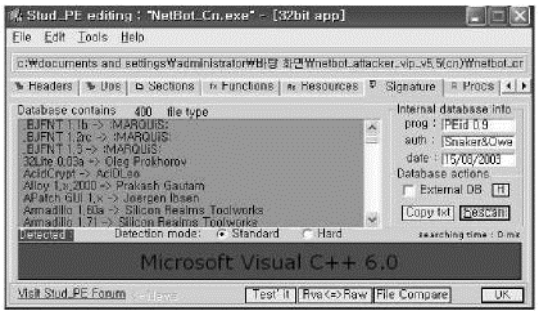
D D o S 공 격 과 백 도 어 기 능 을 가 진 대 표 적 인 툴 킷 으 로 알 려 진 N e t b o t A t t a c k e r 버 전 5 . 5 를 통 해 유 사 성 평 가 를 위 한 기 준 속 성 을 도 출 하 였 다 . N e t b o t A t t a c k e r 는

악성코드생성용빌더 (NB5.5Build.exe)와 해당 악성코드를관리하기위한 제어용 도구(NetBot_Cn.exe)로 구성되어 있다 실행파일 분석도구로 확인한 결과,이들 2 개의실행파일들은 MicrosoftsoftVisualC++6.0 버전으 로 컴파일 되었고암호화 관련 패킹 파일 속성을 지니고 있음을확인하였다.



[그 V 5]NB5.5Build.exe 암호화여부

[Fig.5]NB5.5Build.exeencryptionchecking



[그 V 6]NetBot_Cn.exe 암호화 여부

[Fig.6]NetBot_Cn.exeencryptionchecking

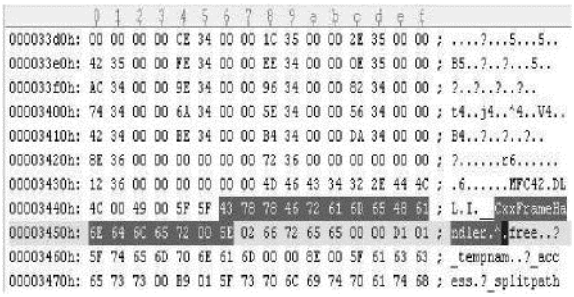
정적분석을 통해 NB5.5Build.exe,NetBot_Cn.exe 내용에는 임시 311-일명, 에러 I 시지등의 ASCII 문자열 속성을 사용하고 있*을알 수 있다.

File pos	Mem pos	ID	Text
A 00004070	00404070	0	FILE2_LEN87552
A 000040E0	004040E0	0	FILE1_LEN253500
A 00004150	00404150	0	~pmThis.tmp
A 0000416C	0040416C	0	ERROR 0x0000d: file corrupted
A 0000418C	0040418C	0	FILE2_PARADN
A 00004199	00404199	0	FILE2_PAR
A 000041FC	004041FC	0	ERROR 0x0000c: file corrupted
A 0000421C	0040421C	0	FILE2_NAMCross.exe
A 0000428C	0040428C	0	ERROR 0x0000b: file corrupted
A 000042AC	004042AC	0	FILE1_PAR
A 0000431C	0040431C	0	ERROR 0x0000a: file corrupted
A 0000433C	0040433C	0	FILE1_NAMNB5.5Build.exe
A 000043B4	004043B4	0	~__UNINST.EXE

[그 V 7]ASCII 문자열

[Fig.7]ASCIIstring

CxxFrameHandler 함수는 유니코드 문자열 안에 있는 문자 정보를 cg 는다



[그 림 8] 국 가 코 드

[Fig.8]Nationalcodeexample

N B 5 . 5 B u i l d . e x e , N e t B o t _ C n . e x e 파 일 이 생 성 된 시 점 에 대 한 타 임 라 인 정 보 를 보 면 생 성 시 간 , 접 근 시 간 , 수 정 시 간 등 을 알 수 있 다 .

NB5.5Build.exe

Property	Value
File Name	C:₩Documents and Settings₩Administrator₩바탕 화면₩NetBot,..
File Type	Portable Executable 32
File Info	Microsoft Visual C++ 6.0
File Size	357.06 KB (365628 bytes)
PE Size	24.00 KB (24576 bytes)
Created	Monday 09 January 2012, 13:31:41
Modified	Sunday 25 January 2009, 11:00:56
Accessed	Tuesday 10 January 2012, 14:45:03
MD5	4C89DBB1B680B05FADF5093C511F9E84
SHA-1	28C6A059D3452675E857287DDDC1C4348416BD5E

[그림 9]NB5.5Build.exe 타임라인

[Fig.9]NB5.5Build.exetimeline

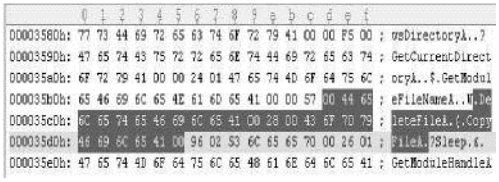
NetBot_Cn.exe

Property	Value
File Name	C:₩Documents and Settings₩Administrator₩바탕 화면₩NetBot,..
File Type	Portable Executable 32
File Info	Microsoft Visual C++ 6.0
File Size	2.15 MB (2256896 bytes)
PE Size	24.00 KB (24576 bytes)
Created	Monday 09 January 2012, 13:31:41
Modified	Sunday 25 January 2009, 10:57:26
Accessed	Wednesday 11 January 2012, 10:06:06
MD5	D85063CD229D9DE46CA29854028CCAE5
SHA-1	836C16E84632AB905B6AC8B9BAB71ADBC24F2DD8

[그림 10]NetBot_Cn.exe 타임 라인

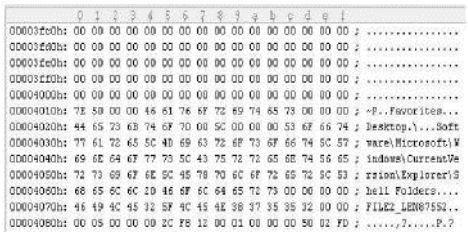
[Fig.10]NetBot_Cn.exetimeline

컴 파 일 러 별 사 용 하 는 함 수 및 변 수 이 름 을 통 해 파 일 을 복 사 하 는 C o p y F i l e A , 파 일 을 지 우 는 D e l e t e F i l e A 함 수 가 있 는 것 을 확 인 하 였 다 .



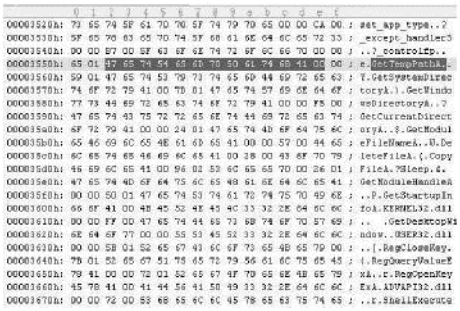
[그림 11]컴파일러환경(CreateFileA)
[Fig.11]Compilierenvironment(CreateFileA)

레지스트리내 HKEY_CURRENT_USER 또는 HKEY_LOCAL_MACHINE 내 Software\Microsoft\ Windows\CurrentVersion\Explorer\ShellFolders 경로를통해 바탕화면,즐겨찾기,공유문서 위치 등을 변경할 수 가 있음을 알 수 있다.



[그림 12]파일경로
[Fig.12]Filepath

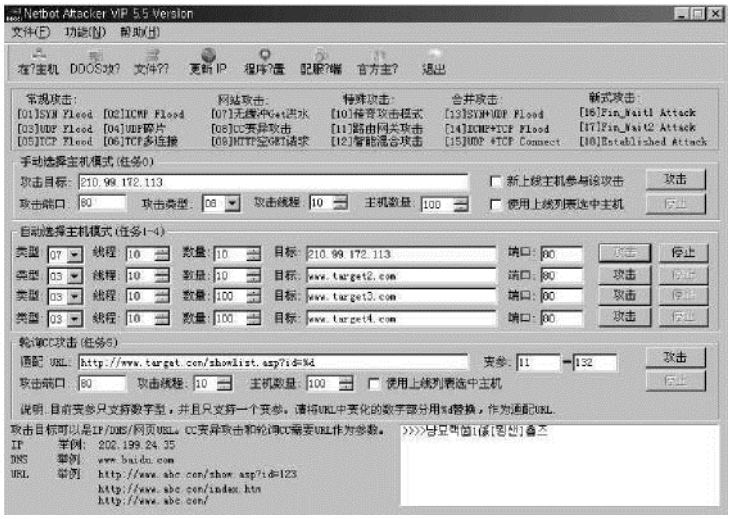
GetTempPathA 함수는 임시폴더의 경로를저장하고,GetSystemDirectoryA 함수는윈도우가설치 된 폴더를
저장하며,GetCurrentDirectoryA 함수는현재프로세스에대해 현재폴더를 바꾸는 작 업을 한다,GetModuleFileNameA 함수는 파일의 절대 경로를 저장하며,GetWindowsDirectoryA 함수는 윈도우가 설치된폴더를알아내는 기능을하며,GetModuleHandleA 핸들하는 파일의절대 경로를 저장하는 것을 확인 할 수 있다.GetStartupInfoA,ShellExecuteA 은 외부 프로그램을 실행 할 수 있으며,GetDesktopWindow 함수는 윈도우바탕화면에 내용을 출력한다.



[그림 13] 제어 및 명령 함수

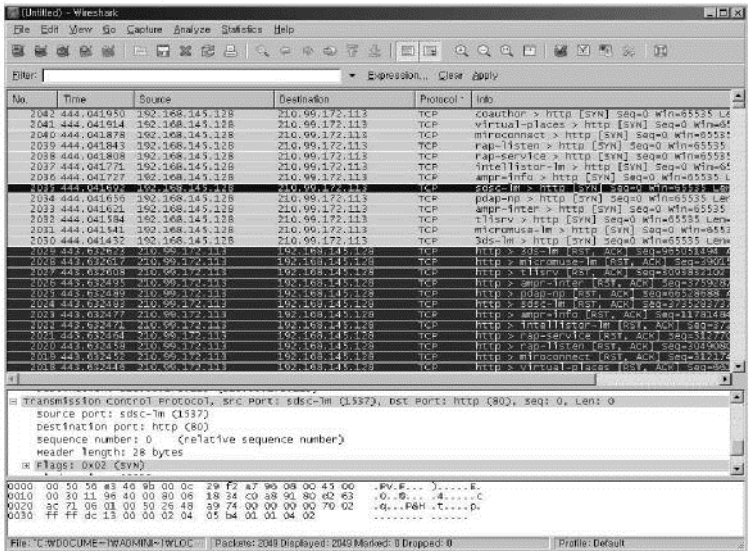
[Fig.13]Controlandcommandfunction

NB5.5Build.exe 를 실행하면 C:\DocumentsandSettings\Administrator\LocalSettings\Temp 에 Cracked.exe 가 UPXg11 킹되어생성되며감염된 t PC 에서 실행 시 고재부 1 을 시킨 후 +9_격 으로 NeBot_Cn.exe 도구를 통해 제어하게된다.NeBot_Cn.exe 에서 지정된 타 41 으로 옵션메뉴 7 번 NnCacheGetFlooding 공격을 지정된 횟수만큼 공격하기 위한 제어 명령어(01000032)를 전달 하%.*을 확인할수 있다



[그 V 14] 제어 및 명령 프로그램

[Fig.14]Controlandcommandprogram



[그 V 15] 제어 및 명령 상태

[Fig.15]Controlandcommandstatus

4. 결 론

악성코드 시그니처 기반 스캐너의 검색을 회피하기 위해 공격용 툴킷을 사용한 시간대별 최신 변종에서는 암호화 알고리즘을 적용하기도 한다. 암호화 또는 난독화된 공격용 툴킷에 대한 동적 분석, 특정 시간에 동작하는 네트워크 쿼리에 대한 동적 분석, 그리고, 공격용 툴킷에 대한 정적 분

석 연구 등을 포함한 통합적인 분석에 대한 연구는 아직

미흡한 실정이다. 본 논문에서는 공격용

툴킷에 대한 정적·동적 분석을 이용한

악성코드 유사성에 대한 기준 속성

을 제안 하였다. 공격용 툴킷을 통한 실험에서 고유 특성을 추출하고 특징을 분류하기 위해 점수 화 기법을 사용하였다. 이를 통하여 변종 악성코드들에 대한 더욱 정확한 분류가 가능할 것으로

기대한다. 향후 연구에서는 항목별로 유사한

속성을 미리 계산하고 사용자 평가 상위 항목

중 유사한 항

목들의 가중치에 대한 목록을 생성하여 트리 구조로 전개하여 비교 서브셋을 설정하고 임계값을 결정할 것이다. 이와 동일한 트리 구조로 형성되는 공격용 툴킷과 유사성 기준 속성의 가중치에 따라 비교하는 방법이 있을 수 있는데, 이를 통해 악성코드 식별을 위한 가독성과 정확성을 향상시킬 수 있을 것이다. 그리고, 공격용 툴킷 기반 악성코드를 정적·동적 분석 하여 툴킷 기준 속성별로 적용한 알고리즘을 반영하는

방법은 유사한 악성코드의 식별과 새로운 변종 구분을 위한 효과적이고 새로운 기준을 제시할 것이다. 또한, 더욱 다양한 공격용 툴킷 샘플들에 적용하여 비교해 보고 그 결과를 시각적으로 표현할 수 있을 것이다.

참 고 문 헌

[1] 서희석, 최중섭, 주필환, "윈도우 악성코드분류 방법론의 설계", 정보보호학회 논문지, 제 19 권, 제 2 호, pp.84-91, 2009 년 4 월.

[2] 이성훈, "Program 유사 정도에 대한 사례 연구", 한국인터넷정보학회 논문지, 제 7 권, 제 2 호, pp 373-377, 2006 년 11 월

[3] Andrew Walenstein and Arun Lakhotia. The Software Similarity Problem in Malware Analysis. In Proceedings Dagstuhl Seminar 06301: Duplication, Redundancy, and Similarity in Software, 10 pp., Dagstuhl, Germany, July 2006

[4] 조은선, "악성 행위 방지를 위한 프로그램 분석" 프로그래밍언어 논문지, 제 19 권, 제 1 호, pp 26-32, 2005 년 8 월.

[5] Jusuk Lee, Kyoochang Jeong, Heejo Lee, "Detecting Metamorphic Malwares using Code Graphs", SAC'10 proceedings of the 2010 ACM Symposium on applied computing, pp.22-26, 2010

[6] E. Carrera et al, "Digital Genome Mapping: Advanced Binary Malware Analysis", Proceedings of 15th Virus Bulletin International Conference (VB 2004), pp.187-197, 2004

[7] Marius Gheorghescu, "An Automated Virus Classfication System", Proceedings of 16th Virus Bulletin International Conference (VB 2005), pp.294-300, 2005

[8] 임 을 규 외 7 인 , " 악 성 코 드 유 형 에 따 른 자 동 화 분 석 방 법 론 연 구 보 고 서 , " 한 국 정 보 보 호 진 흥 원 , K I S A - W P - 2 0 0 9 - 0 0 2 0 , p p . 5 5 - 5 7 , 2 0 0 9 년 6 월 .

[9] 한 경 수 , 김 인 경 , 임 을 규 , " A P I 순 차 적 특 징 을 이 용 한 악 성 코 드 변 종 분 류 기 법 " , 보 안 공 학 연 구 논 문 지 , 제 8 권 , 제 2 호 , p p . 3 1 9 - 3 3 3 , 2 0 1 1 년 4 월

[10]DanielBilar, "Statistical structures: Fingerprinting malware for classification and analysis", Blackhat.com, Sep. 2005

저 자 소 개

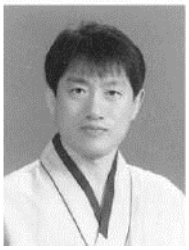
정용욱 (Yong-WookChung)

1 9 9 5 년 2 월 : 금 오 공 과 대 학 토 목 공 학 과 졸 업 (학 사)

2000 년 10 월 :RoyalHolloway,UniversityofLondon,InformationSecurity MSc

2 0 0 6 년 ~ 현 재 : 전 남 대 학 교 정 보 보 호 협 동 과 정 박 사 과 정

관 심 분 야 : 몰 웨 어 포 렌 식 , 모 바 일 포 렌 식 , 네 트 워 크 포 렌 식 , 디 스 크 포 렌 식



노봉남 (Bong-Nam Noh)

1 9 7 8 년 2 월 : 전 남 대 학 교 수 학 교 육 과 졸 업 (학 사)

1 9 8 2 년 2 월 : K A I S T 대 학 원 전 산 학 과 졸 업 (석 사)

1 9 9 4 년 2 월 : 전 북 대 학 교 대 학 원 전 산 과 졸 업 (박 사)

1 9 8 3 년 ~ 현 재 : 전 남 대 학 교 전 자 컴 퓨 터 공 학 부 교 수

2 0 0 0 년 ~ 현 재 : 전 남 대 학 교 시 스 템 보 안 연 구 센 터 소 장

관 심 분 야 : 사 이 버 사 회 와 보 안 , 정 보 보 안 , 시 스 템 및 네 트 워 크 보 안