# 악성코드 수집 및 분석 기술 평가지표 제안

한경수[1], 김인경[2], 임을규[3]

# A Proposal of Evaluation Index for Malware Collecting and Analyzing

Kyoung-Soo Han[1], In-Kyoung Kim[2], Eul-Gyu Im[3]

요 약

악성코드의 형태가 다양해지고 그 수가 지속적으로 증가하면서 악성코드로 인한 위협이 확산되고 있다. 그 결과 사용자들의 피해 역시 급증하고 있으며, 이에 대응하기 위해서는 먼저 악성코드를 수집하고 분석해야 한다. 기존에는 악성코드의 수집과 분석에 있어서 수동적인 요소가 많이 존재하였으나 최근에는 자동으로 악성코드를 수집하고 분석하는 기술이 지속적으로 연구되고 개발되는 추세이다. 그러나 이에 대하여 성능이나 정확성 등을 판단할 수 있는 평가지표가 부족한 실정이다. 본 논문에서는 악성코드 자동 수집 및 분석 기술이 만족해야 하는 요구사항을 분석하여 이를 기반으로 성능 및 정확성 향상과 관련된 평가지표를 제시하고, 이에 상응하는 악성코드 수집 및 분석 프레임워크를 제시한다.

핵심어 : 악성코드 수집, 악성코드 분석, 평가지표

Abstract

As new malware constantly emerges, and the attacking and disguising techniques gets more complicated, it's great challenge to secure user data against threat from these malware. It's a common process to collect and analysis various malware for further detection. There has been some research on malware collector and analyzer, but deficit also exists. Recently, automatic collection and analysis techniques of malware has been a hot topic. While the existing researches is insufficient in view of evaluation on accuracy and performance. In this paper, we proposed the indices to evaluate the accuracy and performance of automatic malware collecting and analyzing techniques. Moreover, we proposed a framework corresponding to the evaluation indices.

Keywords : Malware collecting, Malware analyzing, Evaluation indices

접수일(2010년11월29일), 심사의뢰일(2010년11월30일), 심사완료일(1차:2010년12월15일, 2차:2011년01월08일)

게재일(2011년02월28일)

[1]133-791 서울시 성동구 행당1동 17 한양대학교 전자컴퓨터통신공학과.
email: 1hanasun@hanyang.ac.kr

[2]133-791 서울시 성동구 행당1동 17 한양대학교 전자컴퓨터통신공학과.
email: blackangel@hanyang.ac.kr

[3](교신저자) 133-791 서울시 성동구 행당1동 17 한양대학교 컴퓨터공학부 교수.
email: imeg@hanyang.ac.kr

# 1. 서론

최근 증가하고 있는 악성코드로 인하여 감염된 컴퓨터에서 사용자의 데이터 및 개인정보가 유출되거나, 공격자에 의해 감염된 컴퓨터가 스팸메일 발송과 분산서비스거부(DDoS) 공격 등의 도구로 악용되는 사례가 증가하고 있다. 그 결과 사용자들의 피해가 급증하고 있으며, 또한 전 세계적인 위협으로 확산되고 있다. 따라서 급격히 증가하는 악성코드에 대하여 신속한 대응이 절실히 요구되고 있으며, 악성코드를 예방하고 차단하기 위해서는 먼저 다양한 악성코드의 샘플들을 수집하고 분석해야 한다. 악성코드를 수집하는 방법으로는 이메일 수신 시스템에서 악성코드가 첨부된 스팸메일로부터 수집하거나[1], 악성코드를 유포시키고 있는 웹 사이트로부터 수집이 가능하며, 허니팟[2-3]이 이용되기도 한다. 최근에는 악성코드를 수집하기 위해 인터넷상에 존재하는 수많은 웹 사이트를 탐색하여 각종 정보 및 문서를 수집하고 데이터베이스에 저장하는 웹 크롤러[3-5]가 사용되고 있다. 이렇게 수집된 악성코드는 정적 분석과 동적 분석으로 나누어 분석한다. 정적 분석은 디버거와 디스어셈블러를 이용하여 실제 악성코드를 직접 실행하지 않고 디버깅하거나 코드를 분석하는 방법이며 최근에는 컨트롤 플로우 그래프를 이용한 방법[6-8], 명령어 빈도수를 이용한 방법[9], 함수의 길이를 이용한 방법[10], 함수 호출 그래프를 이용한 방법[11], 바이트 레벨 파일 컨텐츠 분석 방법[12] 등이 제안되었다. 동적 분석은 제한된 실행 환경 내에서 직접 실행해봄으로써 동작 과정 및 시스템 내에서의 행위를 모니터링 하는 방법이며, 악성행위 및 API 콜 순차패턴을 이용한 방법[13], 악성행위 기반 마이닝을 이용한 방법[14] 등이 제안되었다. 기존에는 악성코드의 수집 및 분석에 수동적인 요소가 대부분이었으며, 이를 위해 시간 및 비용이 많이 소요되는 단점이 존재하였다. 이를 극복하기 위해 최근에는 악성코드를 자동으로 수집하고 분석하는 기술과 시스템이 지속적으로 연구되고 개발되는 추세이다. 그러나 이에 대하여 성능이나 정확성을 판단할 수 있도록 평가할 수 있는 지표는 부족한 실정이다. 따라서 본 논문에서는 악성코드를 자동으로 수집하고 분석하기 위한 기술들에 대하여 성능 및 정확성을 평가할 수 있는 지표를 제시하고자 한다.

본 논문의 구성은 다음과 같다. 2장에서는 악성코드의 수집 및 분석 기술에 대한 관련 연구를 기술하고, 3장에서는 이러한 기술들을 평가하기 위한 평가지표를 제시한다. 4장에서는 기존의 연구된 기술들에 제시한 평가지표를 적용한 결과를 기술하고, 5장에서는 평가지표를 만족하는 악성코드 자동 수집 및 분석 기술의 프레임워크를 제시한다. 마지막으로 6장에서는 결론 및 향후 연구 방향에 대하여 기술한다.

# 2. 관련연구

## 2.1. 이메일 수신 시스템을 이용한 악성코드 수집 기술

최근 악성코드가 첨부된 메일을 정상적인 메일처럼 위장하여 불특정 다수를 상대로 악성코드를 유포하는 방식이 급증하고 있다. 스팸메일을 통한 악성코드의 유포는 시스템에 대한 직접적인 공격보다 성공률이 높기 때문이다. 따라서 이메일 수신 시스템에서 스팸메일에 첨부된 악성코드를 수집하는 방법이 사용된다. 기존의 방법으로는 메일 서버의 전방에서 메일 서버로 전달되는 이메일을 분석하고, 첨부된 악성코드 은닉 문서를 탐지하거나 문서 내 악성코드를 탐지 및 제거하는 방법, 첨부된 파일을 분석하고 시그니처를 추출하여 보고된 악성코드의 것과 비교함으로써 악성코드를 탐지하는 방법이 이용되고 있다[1]. [3]에서 제안한 시스템은 이메일에서 웹 사이트의 URL과 첨부된 파일을 추출하고, VMWare 허니팟(Honeypot)을 이용하여 URL에 접속하거나 첨부된 파일을 실행한다. 그리고 실행 결과에서 악성행위가 탐지되면 허니팟에 설치된 모니터가 이를 알리게 되고, 해당 URL 또는 첨부된 파일을 로그와 함께 로그 서버로 전송한다.

### 2.2. 웹 크롤러를 이용한 악성코드 수집 및 분석 기술

웹 크롤러(Web Crawler)는 검색엔진에서 사용되며, 빠른 속도로 인터넷에서 웹 사이트의 URL 정보를 수집하는 탐색 프로그램이다[3]. 또한 웹 크롤러가 정보를 수집하는 과정을 웹 크롤링(Web Crawling)이라 한다. 그러나 악성코드 수집을 위한 웹 크롤링 기술은 검색엔진에서 사용되는 웹 크롤링과 다르게 웹 사이트에 포함된 악성코드 후보가 될 수 있는 실행파일이나 압축파일들을 선택하여 수집한다. 즉, URL의 파일 확장명이 .exe이거나 HTTP 헤더의 “Content type”이 “application/octet-stream”일 때, 웹 크롤러는 그 파일들을 실행파일로 간주하고 다운로드한다. 그리고 다운로드 된 파일들의 헤더 부분을 검사하여 실행파일인지 검사하는 과정도 포함된다. 실행파일뿐만 아니라 압축파일 및 MS 설치파일 역시 동일한 방식으로 검사하여 다운로드한다.

현재 웹 크롤링 기술을 이용한 악성코드 자동 수집 기술들이 다수 제안되었으며, 대부분 웹 크롤링 기술을 이용하여 웹 사이트들을 탐색하고, 허니팟을 이용하여 탐색된 웹 사이트들로부터 악성코드를 포함하고 있는지 판별한 후 다운로드하여 분석한다.

[3]에서는 악성코드가 포함된 웹 사이트를 탐색하기 위한 방법으로 웹 크롤러를 사용하였으며, 웹 크롤러의 수집 효율을 높이기 위해 웹 크롤링 과정에서 웹 페이지 필터링을 수행하였다. 웹 크롤링 수행 결과 악성코드가 포함된 URL을 수집하고, 클라이언트 측 VMware 허니팟을 이용하여 분석하였다. [4]에서는 에뮬레이트 된 시스템이나 서비스를 제공하여 행위에 제한을 두면서 악성코드를 분석할 수 있는 Low-Interaction Honeypot과 웹 크롤러를 기반으로 “Monkey-Spider”라는 시스템을 제안하였으며, 웹 크롤링 후 다운로드한 컨텐츠 분석을 통해 악성코드를 내포한 웹 사이트인지 판단한다. 이때 수행 시간을 줄이기 위하여 웹 크롤링과 스캐닝 분석을 각각 다른 머신에서 동작하도록 설계하였다.

### 2.3. 다양한 악성코드 분석 기술

[15]에서는 악성코드 변종에 신속하게 대응하기 위해 잘 알려지지 않은 공격패턴을 실시간으로 분석하고 생성, 차단함으로써 공격을 방지할 수 있는 공격 식별패턴 실시간 자동 생성 시스템(Zero-day Attack Signature Manufacture INfrastructure : ZASMIN)을 개발하였다. ZASMIN은 기존의 비정상 트래픽이나 공격 패킷이 같은 고유 패턴을 이용하여 탐지하는 방법뿐만 아니라, 네트워크상에서 전송되는 실행파일을 탐지하고 수집하여 재구성한다. 이를 이용하여 해당 실행파일이 악성인지 여부를 판단할 수 있는 시스템을 갖추고 있다. [16]에서는 보다 효율적인 악성코드 분석 및 탐지를 위해 ClamAV[18]를 확장시켜 빠른 탐지와 메모리를 적게 사용하는 것에 초점을 맞추고, 이를 위해서 Feed- Forward Bloom Filter(FFBF)를 사용한 “SplitScreen”을 제안하였다. 제안된 방법은 전체 파일을 w 길이의 슬라이딩 윈도우로 스캔하고, 스캔된 내용에 대하여 블룸필터를 통해 비트 벡터(Bit Vector)를 생성한 후 데이터베이스에 존재하는 기존 악성코드의 비트 벡터와 비교하여 동일한 값이 포함되어 있을 경우 악성코드를 탐지하는 방법이다. [17]에서는 동적 프로그램 슬라이싱(Dynamic Program Slicing)을 통해 악성코드 바이너리로부터 악성행위와 관련된 알고리즘을 추출하여 가젯(Gadget)을 생성함으로써 악성코드 자체를 실행시키지 않더라도 악성행위를 수행하도록 특정 태스크를 실행시킬 수 있는 “Inspector Gadget”이라는 시스템을 제안하였다. 이는 분석환경 내에서 악성코드에 대한 동적 분석을 수행하고, 특정 행위를 나타내는 가젯을 중간 코드 및 추가적인 데이터 영역과 함께 추출하며, 추출된 가젯은 특정 태스크를 수행하기 위한 가젯 플레이어(Gadget Player) 내에서 실행시키는 방법이다. Inspector Gadget은 악성코드로부터 추출한 일부분만을 독립적으로 실행시킬 수 있기 때문에 행위 분석 과정에서의 위험 노출을 감소키고, 잠복기를 가지는 악성코드의 경우 추출된 가젯을 실행시키면 즉시 행위 분석이 가능하며, 메모리 버퍼 내에 존재하는 암호화된 데이터를 확인할 수 있다.

## 3. 평가지표 제안

본 장에서는 향후 악성코드 수집 및 분석 기술을 개발하기 위해 만족해야 하는 요구사항을 기반으로 성능 향상에 관한 평가지표와 정확성 향상에 관한 평가지표로 나누어 기술한다.

### 3.1. 성능 관련 평가지표

#### 3.1.1. 선처리(Pre-Processing) 과정이 포함되어 있는가

최근 유포되는 악성코드의 수는 기하급수적으로 증가하고 있다. 따라서 악성코드 자동 분석 기술에서는 대량의 악성코드를 처리할 수 있도록 오버헤드를 줄이기 위하여 선처리 과정이 포함되어야 한다. 악성코드 변종 생성 기법 중의 하나인 Code Obfuscation을 통해 생성한 변종 분석 시 Code Obfuscation을 위해 추가된 부분을 제거하는 과정이 추가되어야 한다.

#### 3.1.2. 고성능 시스템 리소스를 효율적으로 적용 및 활용할 수 있는가

최근 다양한 악성코드 분석 기술이 등장하고 그 규모가 짐에 따라 기존의 싱글 프로세서 및 싱글 코어 환경이 아닌 멀티 프로세서와 멀티 코어 환경에서의 시스템 적용이 가능해지고 있다. 따라서 악성코드 수집 및 분석 기술은 이러한 다양한 환경에 적용이 가능한 형태를 가지고 있어야 한다. 특히 여러 그룹의 악성코드와 유사도를 판별하여 분석하고 분류하는 기술의 경우 멀티 프로세서 및 멀티 코어를 활용하여 성능을 향상시킬 수 있다.

#### 3.1.3. 악성코드 분석에 특징 기반 클래스 후보군의 우선순위를 반영하는 과정이 포함되어 있는가

수집된 악성코드 후보 프로그램을 분석 및 분류하는 과정에서 후보 프로그램과 기존 악성코드 그룹의 정보를 기반으로 유사도 측정을 하게 되는데, 이 때 우선적으로 비교를 하는 그룹 후보군을 선정하는 과정 및 선정을 위한 선처리 과정이 포함되어야 한다. 수많은 악성코드 그룹에서 후보 프로그램의 분모그룹을 찾아내는 것은 복잡한 과정이기 때문에 선처리 과정을 통해 후보군을 선정하여 우선적으로 비교 연산을 수행함으로서 성능 향상을 기대할 수 있다.

#### 3.1.4. 시간 복잡도 최적화를 반영하는가

현재 발견된 악성코드는 물론, 하루에도 수없이 많은 악성코드 변종이 등장하고 있기 때문에 악성코드 자동 분석 기술을 통해 처리해야 하는 악성코드의 수 역시 증가하고 있다. 따라서 악성코드 분석 기술은 신속한 처리를 위해 분석 및 분류에 사용되는 알고리즘의 시간 복잡도를 최적화하는 과정을 포함해야 한다.

### 3.2. 정확성 관련 평가지표

#### 3.2.1. 악성코드 클래스의 특징을 충분히 반영하였는가

악성코드를 수집하고 분석하는 과정에서 중요한 것 중의 하나는 분석에 사용하기 위해 기존의 알려진 악성코드 그룹에서 특징을 추출하는 과정이다. 악성코드 특징의 추출 과정이 정확하게 이루어져야 향후 악성코드 후보 프로그램이 등장하였을 때 정확히 분석하여 분류할 수 있게 된다. 따라서 악성코드 자동 분석 기술에서는 기존 악성코드 그룹의 특징을 충분히 반영하여야 한다.

#### 3.2.2. 새로운 악성코드 분류를 위한 방법을 제시하였는가

악성코드 수집 기술을 통해 기존의 악성코드 그룹에 속하지 않는 새로운 악성코드가 수집될 수 있다. 이는 새롭게 개발된 악성코드이거나 정상 프로그램일 수 있다. 따라서 악성코드 수집 및 분

석 기술에서는 완전히 새로운 프로그램이 발견되었을 때 악성과 정상 여부를 판단하고, 악성일 경우에는 새로운 그룹을 생성하는 과정을 포함해야 한다.

### 3.2.3. 분석 결과를 얼마나 신뢰할 수 있는가

악성코드 자동 분석 기술의 목적은 악성코드를 같은 그룹으로 효율적으로 분류함으로써 차후에 등장하는 또 다른 변종의 탐지를 용이하게 하기 위한 것이다. 따라서 악성코드 수집 및 분석 기술에 의해 분석된 악성코드가 얼마나 신뢰할 수 있는 정보인지를 판단하는 과정이 필요하다. 또한 충분히 축적된 데이터 셋을 사용하여 효율적인 성능을 발휘할 수 있다는 실험결과로써 증명해야 한다.

### 3.2.4. 안티리버싱/ 안티디버깅 기법을 얼마나 효율적으로 극복하는가

최근 나타나는 악성코드들은 악성코드 분석자가 분석하기 어렵게 하기 위해 내부적으로 안티리버싱/안티디버깅 기법을 포함하고 있는 경우가 존재한다. 따라서 악성코드 수집 및 분석 기술에서는 이러한 안티리버싱/안티디버깅 기법이 적용된 악성코드에 대해서도 효율적인 수집 및 분석이 가능하도록 선처리 과정을 포함하거나 안티리버싱/안티디버깅 기법에 영향을 받지 않는 분석 기술이 개발되어야 할 것이다.

### 3.2.5. 악성코드 수집의 경로가 충분히 다양한가

악성코드 수집 기술로 알려진 웹 크롤링이나 이메일 수신 시스템 등을 살펴보면, 대부분 실행파일을 무작위로 수집하고 분류하는 과정을 포함하고 있다. 예를 들면 웹 크롤링을 통한 수집 시스템은 탐색하는 웹 페이지 선정 알고리즘을 포함하게 되는데, 이러한 알고리즘이 악성코드 유포자에게 파악되지 않도록 다양한 경로를 통해 악성코드를 수집할 수 있어야 한다.

### 3.2.6. 긍정 오류(False Positive)를 줄이기 위한 방법을 적용하는가

악성코드 수집 및 분석 기술에 있어 가장 큰 평가 기준 중 하나로 긍정 오류의 비율(False Positive Ratio)을 들 수 있다. 따라서 악성코드 수집 및 분석 기술에서는 반드시 긍정 오류의 비율을 줄이기 위한 방법이 포함되어야 한다. 악성코드의 그룹 판단 및 분류에서 한 가지의 기준이 아닌 다양한 기법을 활용한 판단 기준을 제시하는 방법이 긍정 오류를 줄이는 방법이 될 수 있다.

### 3.2.7. 피드백 프로세스가 존재하는가

현재 하루에도 수백 건 이상의 새로운 악성코드 변종들이 등장하고 있는 추세이기 때문에 악성코드 분석 및 분류 기술은 기존 데이터베이스를 활용하는 것은 물론, 새로이 수집 및 분류된 악성코드 역시 새로운 분류 기준을 반영해야 할 필요가 있다. 따라서 악성코드 수집 및 분석 기술에서는 최근의 악성코드를 분석 및 분류 기준에 반영함으로써 최신 동향의 악성코드가 수집될 수 있도록 하는 기능이 포함되어야 한다.

### 3.2.8. 기존 수집된 악성코드 DB를 얼마나 활용하는가

악성코드 수집 및 판단에 있어 가장 많은 근거로 활용되는 것 중 하나가 기존 악성코드와의 유사도 측정이다. 대부분의 악성코드가 새롭게 제작되는 것이 아니라 기존의 악성코드를 재사용하여 변종으로 생성하기 때문에 악성코드 수집 및 분석 기술은 기존의 악성코드 데이터베이스를 충분히 활용하여야 한다.

### 3.2.9. 기존 수집된 정상 프로그램의 DB를 활용하는가

기존의 악성코드 자동 수집기는 무작위로 네트워크상의 실행 파일을 수집하게 된다. 따라서 악성코드 수집 기술에서는 수집한 프로그램을 분석하기 전에 미리 정상 프로그램의 데이터베이스를 활용하여 White-List를 작성하고, 이를 프로그램 필터링에 사용하는 과정이 포함되어야 한다. 이미 안전하다고 알려진 프로그램에 대해 분석 및 분류 과정을 거치기 전에 후보 프로그램 군에서 제거함으로써 성능을 향상시키고 긍정 오류의 비율도 줄일 수 있는 효과를 기대할 수 있다.

## 4. 평가지표 적용

본 장에서는 제시한 악성코드 수집 및 분석 기술에 대한 평가지표를 관련연구에서 기술한 기존의 연구 및 기술에 적용하여 평가한다.

[표 1]은 기존 연구에 성능 관련 평가지표를 적용한 것이다. 기존의 각 연구에서 해당 평가지표와 관련된 내용이 포함되어 있는지 여부를 분석하였다. SplitScreen의 경우, 상대적으로 빠른 성능을 위한 제안이 연구 자료에 언급되어 있었다.

[표 1] 평가지표 적용 - 성능

[Table 1] Appling Evaluation Index on Performance

| 지표 / 연구 | 선처리 과정 | 고성능 시스템 활용 | 클래스 우선순위 | 시간 복잡도 최적화 |
|---|---|---|---|---|
| Internet Malware Collecting System[3] | ○ | × | × | × |
| Monkey-Spider[4] | × | × | ○ | × |
| ZASMIN[15] | ○ | × | ○ | × |
| SplitScreen[16] | ○ | ○ | × | ○ |
| Inspector Gadjet[17] | × | × | ○ | ○ |

[표 2] 및 [표 3]은 동일한 기존 연구에 정확성 관련 평가지표를 적용한 결과이다. 대부분의 악

성코드 연구는 정확성을 목적으로 하기 때문에 제안한 평가지표를 대부분 만족시키는 것으로 나타났다. 그러나 정상 프로그램을 활용하는 방법에 대해서는 5개 연구 모두 언급되지 않았다. 실제로 컴퓨터에 설치되는 프로그램은 악성코드보다 정상 프로그램이 더 많은 것이 일반적이다. 따라서 이러한 특성을 기반으로 악성코드 분석에 정상 프로그램의 데이터베이스를 활용한다면 선처리를 통해 성능을 향상시킬 수 있고, 긍정 오류를 최소화하여 정확성에 있어서 효과적인 결과를 기대할 수 있다.

[표 2] 평가지표 적용 - 정확성(1)
[Table 2] Appling Evaluation Index on Accuracy(1)

| 지표 / 연구 | 클래스 특징 반영 | 새로운 악성코드 처리 | 분석 결과 신뢰성 | 안티리버싱 및 안티디버깅 극복 |
|---|---|---|---|---|
| Internet Malware Collecting System[3] | ○ | × | × | × |
| Monkey-Spider[4] | ○ | ○ | ○ | ○ |
| ZASMIN[15] | ○ | ○ | ○ | △ |
| SplitScreen[16] | ○ | × | ○ | ○ |
| Inspector Gadjet[17] | ○ | ○ | ○ | ○ |

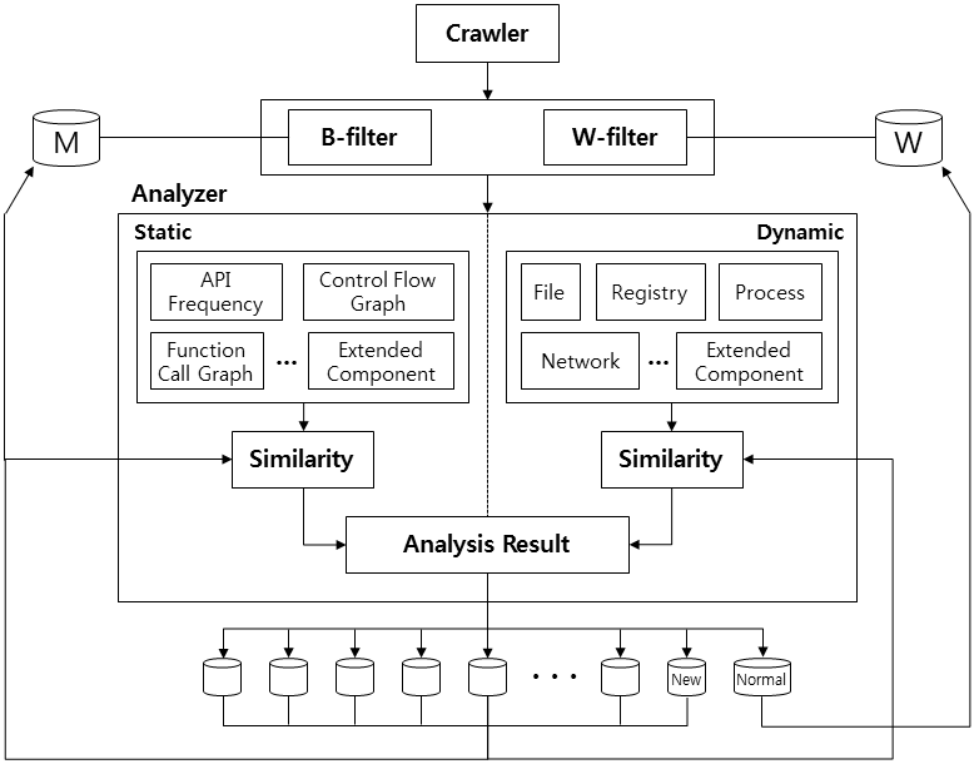
[표 3] 평가지표 적용 - 정확성(2)
[Table 3] Appling Evaluation Index on Accuracy(2)

| 지표 / 연구 | 악성코드 수집 경로 다양성 | 긍정 오류 감소 방안 | 피드백 프로세스 | 악성코드 DB 활용 | 정상 프로그램 DB 활용 |
|---|---|---|---|---|---|
| Internet Malware Collecting System[3] | ○ | × | ○ | × | × |
| Monkey-Spider[4] | ○ | × | ○ | ○ | × |
| ZASMIN[15] | - | × | ○ | ○ | × |
| SplitScreen[16] | - | ○ | ○ | ○ | × |
| Inspector Gadjet[17] | - | × | ○ | ○ | × |

기존의 연구를 대상으로 제시한 평가지표를 적용해본 결과, 상기 평가지표는 연구 자료의 우수성을 판단하는 근거로 사용될 수 있다. 또한 자체 평가지표로 활용하여 연구 자료를 보완하고 발전시키는 방향을 제시해 줄 수 있을 것으로 기대된다.

## 5. 제안하는 프레임워크

[그림 1]은 각각의 평가지표에 상응하는 악성코드 수집 및 분석 기술의 프레임워크를 나타낸 것이다. 먼저 웹 크롤러가 인터넷에서 실행 파일들을 수집하면, 정상 파일들의 정보를 포함하고 있는 White-list DB 및 악성코드의 정보를 포함하고 있는 Black-list DB가 연결된 필터에서 필터링을 함으로써 선처리 단계를 거친다. 이는 기존에 수집되어 있는 악성코드를 중복 수집하지 않도록 할 뿐만 아니라, 정상 파일 역시 수집하지 않게 함으로써 저장 공간의 효율성을 향상시키는 것이다. 악성코드로 의심되는 파일은 분석기에서 정적 분석과 동적 분석이 수행된다. 정적 분석에서는 컨트롤 플로우 그래프[6-8], 함수 호출 그래프[11], API 빈도[19] 등과 같이 악성코드에서 나타날 수 있는 요소들을 분석한다. 동적 분석에서는 파일, 레지스트리, 프로세스, 네트워크 관련 행위들을 모니터링한다. 그 다음 정적 분석 및 동적 분석을 통해 기존의 악성코드와 유사도를 계산하고 결과를 종합하여 각 악성코드 클래스 별로 분류하며, 분류가 불가능할 경우 새로운 DB에 저장한다. 또한 그 결과는 선처리 필터와 연결된 DB에 정보가 전달되며, 특히 White-list DB가 확장될수록 긍정 오류(False Positive)는 감소시키도록 한다.




[그림 1] 제안하는 프레임워크의 전체 흐름도

[Fig. 1] The Overview of Proposed Framework

## 5. 결론 및 향후 연구

악성코드의 형태가 다양해지고 그 수가 지속적으로 증가하면서 악성코드로 인한 위협이 확산됨에 따라 이에 대응하기 위하여 악성코드를 수집하고 분석하는 기술이 지속적으로 연구되고 있다. 그러나 악성코드를 수집하고 분석함에 있어서 정확성이나 성능을 평가할 수 있는 지표는 부족한 실정이다. 본 논문에서는 향후 악성코드 수집 및 분석 기술을 개발할 때 만족해야 하는 요구사항들을 분석하고 이를 기반으로 성능 및 정확성 향상과 관련된 평가지표를 제시하였다. 악성코드를 수집하고 분석할 때 요구되는 비용을 최소화하고 악성코드의 특징을 고려하며, 필요한 프로세스와 요소들을 기술하였다. 향후에는 지속적으로 악성코드 수집 및 분석 기술에 평가지표를 적용하고 추가적인 요구사항을 도출함으로써 평가지표를 개선하고 일반화해야 한다. 또한 평가지표에 대한 우선순위를 두고 정량적 분석을 할 수 있도록 업그레이드해야 한다.

## 6. 사사

본 연구는 교육과학기술부 재원으로 한국연구재단의 중견연구자지원사업으로 수행되었음. (NRF 2009-0084870)

## 참고문헌

[1] 양경철, 이수연, 박원형, 박광철, 임종인, “전자우편을 이용한 악성코드 유포방법 분석 및 탐지에 관한 연구”, 정보보호학회논문지, 제19권 제1호, pp. 93-101, 2009년 2월.

[2] 한경수, 임광혁, 임을규, “허니넷을 이용한 P2P 기반 Storm 봇넷의 트래픽 분석”, 정보보호학회논문지, 제19권 제4호, pp. 51-61, 2009년 8월.

[3] X. Sun, Y. Wang, J. Ren, Y. Zhu, and S. Liu, “Collecting Internet Malware Based on Client-side Honeypot”, Proceedings of the 9th International Conference for Young Computer Scientists, pp. 1493-1498, November 2008.

[4] A. Ikinci, T. Holz, and F. Freiling. “Monkey-Spider: Detecting Malicious Websites with Low-Interaction Honeyclients”, Proceedings of Sicherheit, Schutz und Zuverlässigkeit, April 2008.

[5] K. Whitehouse, G. Tolle, J. Taneja, C. Sharp, S. Kim, J. Jeong, J. Hui, P. Dutta, and D. Culler, “Marionette: Using RPC for Interactive Development and Debugging of Wireless, Embedded Networks”, Proceedings of the 5th International Conference on Information Processing in Sensor Networks, pp. 416-423, April 2006.

[6] Q. Zhang, and D. S. Reeves, “MetaAware: Identifying Metamorphic Malware”, Proceedings of the 23rd Annual Computer Security Applications Conference, pp. 411-420, September 2007.

[7] S. Cesare, and Y. Xiang, “A Fast Flowgraph Based Classification System for Packed and Polymorphic

Malware on the Endhost", Proceedings of the 24th IEEE International Conference on Advanced Information Networking and Applications, pp. 721-728, April 2010.

[8] G. Bonfante, M. Kaczmarek and J.-Y. Marion, "Morphological Detection of Malware", Proceedings of the 3rd International Conference on Malicious and Unwanted Software, pp. 1-8, October 2008.

[9] A. Karnik, S. Goswami, and R. Guha, "Detecting Obfuscated Viruses Using Cosine Similarity Analysis", Proceedings of the 1th Asia International Conference on Modelling & Simulation, pp. 165-170, March 2007.

[10] R. Tian, L.M. Batten, and S.C. Versteeg, "Function Length as a Tool for Malware Classification", Proceedings of the 3rd International Conference on Malicious and Unwanted Software, pp. 69-76, October 2008.

[11] J. Lee, K. Jeong, and H. Lee, "Detecting Metamorphic Malwares using Code Graphs", Proceedings of the 2010 ACM Symposium on Applied Computing, pp. 1970-1977, March 2010.

[12] S. M. Tabish, M. Z. Shafiq, and M. Farooq, "Malware Detection using Statistical Analysis of Byte-Level File Content", Proceedings of the ACM SIGKDD Workshop on CyberSecurity and Intelligence Informatics, pp. 23-31, June 2009.

[13] 박남열, 김용민, 노봉남, "우회기법을 이용하는 악성코드 행위기반 탐지 방법", 정보보호학회논문지, 제16권 제3호, pp. 17-28, 2006년 6월.

[14] Matt Fredrikson, Somesh Jha, Mihai Christodorescu, Reiner Sailer, and Xifeng Yan, Synthesizing Near-Optimal Malware Specifications from Suspicious Behaviors, Proceedings of the 2010 IEEE Symposium on Security and Privacy, pp. 45-60, May 2010.

[15] 한국전자통신연구원, "네트워크 위협의 제로데이 공격 대응을 위한 실시간 공격 시그니처 생성 및 관리기술 개발", 지식경제부, 2009.

[16] SK. Cha, I. Moraru, J. Jang, J. Truelove, D. Brumley, and D. G. Andersen, "SplitScreen: Enabling Efficient, Distributed Malware Detection", Proceedings of the 7th USENIX Conference on Networked Systems Design and Implementation, April 2010.

[17] C. Kolbitsch, T. Holz, C. Kruegel, and E. Kirda, "Inspector Gadget: Automated Extraction of Proprietary Gadgets from Malware Binaries", Proceedings of the 31st IEEE Symposium on Security and Privacy, pp. 29-44, May 2010.

[18] ClamAV, http://www.clamav.net/

[19] 권오철, 배성재, 조재익, 문종섭, "Native API 빈도 기반의 퍼지 군집화를 이용한 악성코드 재그룹화 기법연구", 정보보호학회논문지, 제18권 제6호, pp. 115-127, 2008년 12월.

## 저자 소개

**한경수 (Kyoung-Soo Han)**
2008년 상지대학교 컴퓨터정보공학부 학사
2010년 한양대학교 전자컴퓨터통신공학과 석사
2011년 2월 현재 한양대학교 전자컴퓨터통신공학과 박사과정
관심분야 : 악성코드 분석, 네트워크 보안, 정보보호

**김인경 (In-Kyoung Kim)**
2010년 한양대학교 컴퓨터공학부 학사
2011년 2월 현재 한양대학교 전자컴퓨터통신공학과 석사과정
관심분야 : 악성코드 분석, 네트워크 보안, 정보보호

**임을규 (Eul-Gyu Im)**
1992년 서울대학교 컴퓨터공학과 학사
1994년 서울대학교 컴퓨터공학과 석사
2002년 University of Southern California 컴퓨터과학과 박사
2011년 2월 현재 한양대학교 컴퓨터공학부 조교수
관심분야 : 네트워크 보안, 악성 프로그램 분석, RFID 보안, SCADA 보안