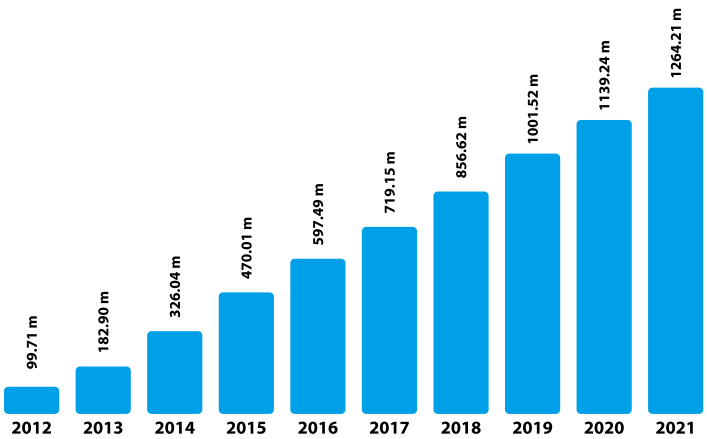
악성코드 통합진단명 생성절차 및 요구사항

이태진 호서대학교 조교수

1. 머리말

악성코드는 단일 컴퓨터, 서버 또는 컴퓨터 네트워크에 손상을 입히기 위해 고안된 모든 종류의 소프트웨어를 일컫는다. 악성코드를 이용한 침해공격은 지속적으로 증가하고 있으며, 점차 지능화된 형태로 발전하고 있다. AV-TEST에 따르면, 현재 악성코드는 12억 건에 이르며, 이는 10년 전에 비해 10배 이상으로 늘어난 규모이다.

한편, 이러한 악성코드에 대응하기 위한 많은 연구와 노력들이 이루어지고 있으며, 이는 악성코드를 고유하게 식별, 특정해야 새로운 위협에 대한 탐지, 분석, 공유 및 대응이 효과적으로 이루어질 수 있기 때문이다. 이를 위해, 악성코드 명명 규칙을 표준화하기 위한 시도가 있었으나 널리 사용되지 않았다. 이미 AV 벤더사마다 고



※출처: AV-TEST

[그림 1] 악성코드 출현현황

유의 악성코드 명명법이 존재하고 있고, 한번 정해진 이름은 이미 자사의 Anti-Virus에 반영되거나 발표된 이후라 통일이 쉽지 않았다. 무엇보다 AV 벤더사들 간의 이해관계가 복잡하게 관련되어 있어 악성코드 통합진단명의 제정은 요원한 일이 되었다. 본고에서는 기존의 복잡한 이해관계를 그대로 유지하면서도 사용 가능한 악성코드 통합진단명에 관해 다룬다. 2장에서는 악성코드 명명법 관련한 기존 추진노력들을 다루고, 3장에서는 악성코드 통합진단명이 갖춰야 할 요구사항을 담았으며, 4장에서는 통합진단명 생성 예시를 제시한다. 마지막으로 5장에서는 결론을 제시한다.

2. 관련표준 추진현황

악성코드 진단명 통일에 대한 노력은 1991년 한 차례 이루어졌다. CARO(Computer Antivirus Researchers Organization) 멤버들이 주축이 되어 악성코드 명명법을 만들었는데, 4개의 요소가 존재하며 각각은 '.'으로 구분된다. 각각은 FamilyName, GroupName, Major_variant, Minor_variant로 구성하며, 대소문자의 구분, 가독성 향상, 각 파트별 최대길이 조정 등의 내용을 담고 있다[1]. 발견장소나 제작자 이름 등은 포함하지 않았다. 이런 규칙은 빠르게 전파되는 악성코드에 적용하기에는 악성코드 유형이나 특성정보가 표현되지 않아 한계가 있었으며, 무엇보다 널리 사용되지 않았다. 이러한 문제를 해결하고자 닉 피츠제랄드(Nick FitzGerald)는 2002년 AVAR 컨퍼런스에서 새로운 진단명안을 마련하였다. 마이크로소프트, F-시큐어 등의 업체에서 이 안과 유사한 방식으로 진단명을 짓고 있지만, 여전히 진단명 통일은 이루어지지 않았다[2]. 아래 그림은 CARO에서의 FamilyName 관련한 규정 예시를 나타낸다.

3. 악성코드 통합진단명 요구사항

3.1 일반사항

악성코드 통합진단명은 Ground truth에 해당하는 AV 벤더사에서 악성코드 진단명을 토

1. Family names.

The Family_Name represents the family to which the virus belongs. Every attempt is made to group the existing viruses into families, depending on the structural similarities of the viruses, but we understand that a formal definition of a family is impossible.

When selecting a Family_Name, the following guidelines must be applied:

"Must"

1) Do not use company names, brand names, or names of living people, except where the virus is provably written by the person. Common first names are permissible, but be careful – avoid if possible. In particular, avoid names associated with the anti-virus world. If a virus claims to be written by a particular person or company do not believe it without further proof.

2) Do not use an existing Family_Name, unless the viruses belong to the same family.

3) Do not invent a new name if there is an existing, acceptable name.

4) Do not use obscene or offensive names.

5) Do not assume that just because an infected sample arrives with a particular name, that the virus has that name.

6) Avoid numeric Family_Names like V845. They should never be used as family names, as the members of the family may have different lengths. When a new virus appears and a new Family_Name must be selected for it, it is acceptable to use a temporary name like _1234, but this must be changed as soon as possible.

"Should"

1) Avoid Family_Names like Friday 13th, September 22nd. They should not be used as family names, as members of the family may have different activation dates.

2) Avoid geographic names which are based on the discovery site – the same virus might appear simultaneously in several different places.

3) If multiple acceptable names exist, select the original one, the one used by the majority of existing anti-virus programs or the more descriptive one.

※출처: CARO

[그림 2] ARO 악성코드 명명법 예시

대로 요구사항과 생성절차에 따라 생성된다. 이러한 통합진단명은 여러 AV 벤더사에 대해 중립성을 지키며(neutrality), 자동으로 생성 가능하며(automatic), 특정 AV 벤더에 의존성이 없어야(vendor-independent)한다. 또한, 정규화(normalization)를 이용하여 최다 득표(plurality voting)된 결과를 선정하고, 플랫폼에 관계없이(cross-platform) 동작하며, 다양한 데이터셋을 이용해도 수량화된 정확도(quantified accuracy)를 나타내야 하며, 실행파일이 불필요(does not require executables)하며, 재현 가능(reproducible)해야 한다. 이러한 악성코드 통합진단명 요구사항을 통해 AV 벤더사들 간의 현실적인 이해관계를 극복하면서도 점차 지능화되는 악성코드의 식별, 분석, 공유 및 대응이 효과적으로 이루어질 수 있다.

3.2 요구사항

3.2.1 중립성(Neutrality)

악성코드 통합진단명의 목적은 악성코드의 이름을 효율적으로 표현하는 것이다. 따라서 AV 벤더사의 특정 AV 엔진에 영향을 받지 않고 중립적이어야 한다.

3.2.2 자동화(Automatic)

효율성을 극대화하기 위한 방법으로 자동화 기술을 사용한다. 악성코드 통합진단명 생성 과정에 있어서 일체의 수동 작업 없이 자동화된 방식으로 결과를 산출한다.

3.2.3 벤더 독립성(Vendor-independent)

기존 다양한 AV 벤더사의 AV 엔진을 통해 식별된 악성코드 진단명이 존재한다. 악성코드 통합진단명을 산출하기 위해서는 기존 AV Engine의 수, 종류에 상관없이 산출되어야 한다.

3.2.4 정규화(Normalization)

AV 벤더사별 악성코드 진단명을 그대로 사용하지 않고, 정규화를 통해 악성코드 통합진단명의 의미가 잘 드러나도록 선정해야한다.

3.2.5 최다 득표(Majority voting)

정규화 과정으로 처리된 이름에 대해 가장 많이 산출된 이름을 선정한다. 특정 이름이 절반 이상의 비율을 차지하는 것은 드물기 때문에 다수결을 이용하여 결과를 산출한다.

3.2.6 크로스 플랫폼(Cross-platform)

컴퓨터 플랫폼에서의 다양한 운영체제에서 악성코드가 악의적 사용, 탐지 및 분석에 이용된다. 악성코드 통합진단명은 플랫폼에 관계없이 지원 가능해야 한다.

3.2.7 수량화된 정확도(Quantified accuracy)

악성코드 통합진단명은 기존 AV 벤더사별 다른 진단명을 통합하는 데 의의를 둔다. 다양한 데이터셋에 대해서 악성코드 통합진단명은 항상 일관되며, 정확성 검증이 되어야 한다.

3.2.8 실행파일 불필요(Does not require executables)

악성코드가 없더라도 악성코드 통합진단명을 식별할 수 있어야 한다. 악성코드를 식별할 수 있는 해시값이 존재한다면, Virustotal과 같은 온라인 파일 검사 서비스를 통해 악성코드 통합진단명을 확인할 수 있어야 한다.

3.2.9 재현 가능(Reproducible)

한 연구를 똑같이 다시 반복함으로써 기존 원본 결과가 똑같이 다시 나타나야 재현성이 만족된다. 악성코드 통합진단명은 항상 재현 가능하며, 같은 결과가 나와야 한다.

4 악성코드 통합진단명 생성절차

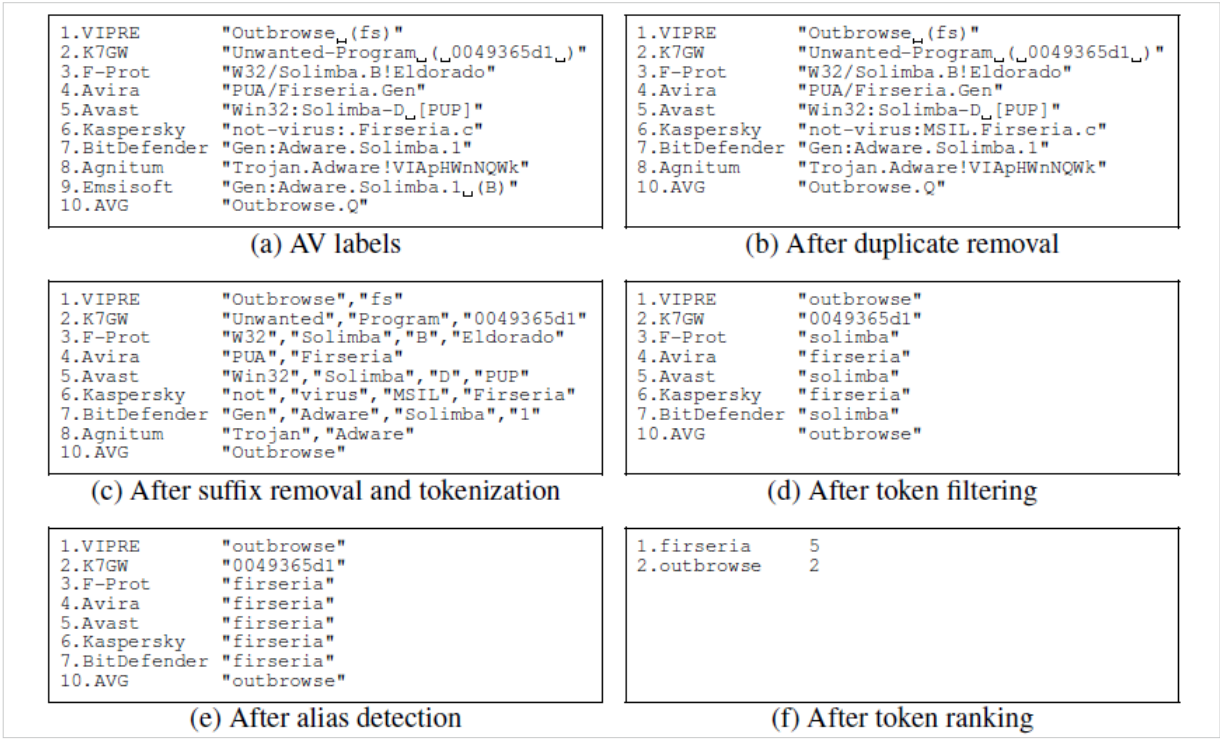
악성코드 통합진단명을 생성하기 위한 과정으로 Virustotal 파일 분석 사이트를 통한 악성코드의 AV 벤더사별 악성코드 이름을 추출한다. 동일 AV 엔진에 의한 진단명을 제거하고, 접미사 제거 및 토큰화 과정을 거치며 토큰 필터링을 수행한다. 동일 alias 그룹을 통합하고 토큰을 통해 악성코드 통합진단명을 생성한다. 아래 그림은 악성코드 통합진단명 생성절차 예시를 나타낸다.

4.1 동일 AV 엔진에 의한 탐지명 제거

동일한 AV 벤더사에 McAfee TrendMicro와 같은 다양한 엔진이 존재하고 이러한 엔진들은 서로 AV 라벨을 복사한다. 각 AV 벤더사별 라벨을 다르게 구성하고 키워드를 다르게 사용해도 AV 엔진은 라벨을 복사한다.

4.2 접미사 제거 및 토큰화

AV 라벨 대부분의 노이즈는 AV 라벨 뒷부분에 존재하는 접미사이다. 접미사를 제거하기 위한 규칙으로 3가지가 있다. 첫 번째로 17개의 AV 엔진의 경우 마지막 점 다음의 라벨을 자른다. 두 번째, AVG의 경우 접미사에 숫자나 대문자만 포함된 경우 마지막 점 뒤에서 자른다. 마지막으로 Agnitum의 경우 마지막 '!'문자 뒤에서 자른다.



※출처: AVClass[3]

[그림 3] 악성코드 통합진단명 생성절차 예시

4.3 토큰 필터링

family-name을 선별하는데 있어서 불필요한 토큰을 제거하기 위한 단계로 4가지 단계가 있다. 첫 번째, 각 라벨의 토큰을 소문자로 변환한다. 두 번째, 토큰 끝의 숫자를 제거한다. 세 번째, 네 글자 미만의 토큰을 제거한다. 마지막으로, 토큰이 샘플 해시의 접두사인 경우 제거한다.

4.4 동일 Alias 통합

AV 벤더사마다 다른 진단명을 가지고 있지만, 같은 의미를 갖는 family-name들이 존재한다. 아래 그림은 같은 악성코드 family에 대해 다른 Alias를 갖는 예시를 나타낸다. 이들은 동일 진단명으로 통합하는 것이 합리적이다. [그림 4]는 같은 악성코드에 대한 Alias를 통합하는 과정을 나타낸다.

4.5 토큰 순위 및 악성코드 통합진단명 선택

라벨의 토큰을 포함하는 엔진 수가 감소하여 토큰 순위를 매기게 된다. 하나의 AV 엔진에서만 나타나는 토큰을 제거한다. 남은 토큰의 개수를 카운트하여 최상위 순위의 토큰을 샘플의 통합진단명으로 선택한다.

5. 맺음말

지금까지 악성코드 통합진단명 생성절차 및 요구사항에 대해 기술하였다. AV 벤더사는 이미 많은 양의 악성코드 샘플과 이에 대한 진단명을 보유하고 있어, 자체 악성코드 진단명을 바꿀 경우 기존 진단명 체계를 모두 바꿔야 하고, 이에 따른 제품 개발 등의 어려움이 존재한다. 하지만 기존 악성코드 진단명을 변경하는 것이 아닌, AV 벤더마다 상이한 진단명과 별도로 통합 진단명으로 재조합 및 생성할 경우 악성코드에 대한 평판 정보값을 표준화할 수 있고, 통합진단명이 동일한 많은 악성코드들을 관리, 탐지가 수월해진다. 또한, 동일한 통합진단명을 가진 악성코드 간 변종 관계를 파악하기 쉬워져 신종 악성코드에 대해서도 통합진단명을 통해서 탐지 및 관리가 용이해진다.

Family	Aliases	Example Aliases
wapomi	12	pikor, otwycal, protil
firseria	10	firser, popeler, solimba
vobfus	9	changeup, meredrop, vbobfus
virut	8	angryangel, madangel, virtob
gamarue	7	debris, lilu, wauchos
hotbar	7	clickpotato, rugo, zango
bandoo	6	ilivid, seasuite, searchsuite,
gamevance	6	arcadeweb, gvance, rivalgame
loadmoney	6	ldmon, odyssey, plocust
zeroaccess	6	maxplus, sirefef, zaccess

※출처: AVCLass[3]

[그림 4] 상위 10개 Alias기준 family-name 예시

※ 본 연구는 2021년도 정부(과학기술정보통신부)의 재원으로 정보통신기획평가원의 지원을 받아 수행된 연구임(No. 2018-0-00276, 딥러닝 기반 악성코드 패턴룰셋 생성 자동화 원천기술 개발)

주요 용어 풀이

- **사이버 공격**(cyber attack) : 하나 이상의 컴퓨터에서 다른 컴퓨터, 여러 컴퓨터 또는 네트워크에 대해 시작된 공격.
- **악성코드**(malicious code) : 악의적인 목적을 위해 작성된 실행 가능한 코드를 의미함. 실행 가능한 코드에는 프로그램, 매크로, 스크립트 뿐만 아니라 취약점을 이용한 데이터 형태도 포함됨. 악성 소프트웨어는 가장 광범위한 개념이며 자기 복제 능력과 감염대상 유무에 따라 바이러스, 웜, 트로이목마, 스파이웨어 등으로 분류됨.
- **토큰**(token) : 문법적으로 더 이상 나눌 수 없는 언어요소.
- **플랫폼**(platform) : 컴퓨터 시스템의 기반이 되는 소프트웨어가 구동 가능한 하드웨어 구조 또는 소프트웨어 프레임워크의 일종으로, 구조(architecture), 운영체제(operating System), 프로그래밍 언어, 그리고 관련 런타임 라이브러리 또는 그래픽 사용자 인터페이스(GUI: Graphic User Interface) 등을 포함함.
- **해시값**(hash value) : 임의의 길이로 구성된 입력 메시지를 다양한 길이를 가진 데이터를 고정된 길이를 가진 데이터로 매핑(mapping)한 값.

참고문헌

[1] CARO Virus Naming Convention. http://www.caro.org/articles/naming.html

[2] FitzGerald, Nick. 'A virus by any other name: Towards the revised caro naming convention.' Proc. AVAR (2002): 141-166.

[3] Sebastián, Marcos, et al. 'Avclass: A tool for massive malware labeling.' International symposium on research in attacks, intrusions, and defenses. Springer, Cham, 2016.