https://doi.org/10.33778/kcsa.2023.23.2.003

웹에 숨겨진 악성코드 배포 네트워크에서 악성코드 전파 핵심노드를 찾는 방안

김 성 진*

요 약

웹에 존재하는 악성코드 배포 네트워크에는 악성코드 배포를 위해 핵심 역할을 수행하는 중심 노드가 있다. 이 노드를 찾아 차단하면 악성코드 전파를 효과적으로 차단할 수 있다. 본 연구에서는 복잡계 네트워크에서 위험 분석이 적용된 centrality 검색 방법을 제안하였고, 이 방식을 통해 악성코드 배포 네트워크 내에서 핵심노드를 찾는 방법을 소개한다. 그 외에, 정상 네트워크와 악성 네트워트는 in-degree와 out-degree 측면에서 큰 차이가 있고, 네트워크 레이아웃 측면에서도 서로 다르다. 이 특징을 통해 우리는 악성과 정상 네트워크를 분별할 수 있다.

A Method to Find the Core Node Engaged in Malware Propagation in the Malware Distribution Network Hidden in the Web

Kim Sung Jin*

ABSTRACT

In the malware distribution network existing on the web, there is a central node that plays a key role in distributing malware. If you find and block this node, you can effectively block the propagation of malware. In this study, a centrality search method applied with risk analysis in a complex network is proposed, and a method for finding a core node in a malware distribution network is introduced through this approach. In addition, there is a big difference between a benign network and a malicious network in terms of in-degree and out-degree, and also in terms of network layout. Through these characteristics, we can discriminate between malicious and benign networks.

Key words : Centrality, Complex Network, Malicious URL, Malware, MDN

접수일(2023년 4월 5일), 수정일(1차: 2023년 4월 10일), 게재확정일(2023년 4월 13일)

* 제주한라대학교 인공지능공학과 조교수

1. 서 론

인터넷은 개방형 네트워크로 악성코드 유포가 용이하다. 공격자는 해킹한 웹사이트에 악성 URL을 삽입하여 최종 공격을 수행하는 공격코드인 exploit kit(EK)까지 연결시킨다. 사용자는 인터넷 서핑 중 공격자가 삽입해 놓은 공격코드가 자신의 브라우저로 로드되어 자신의 컴퓨터 내의 다양한 어플리케션을 공격하는지 모른 채 악성코드에 감염된다.

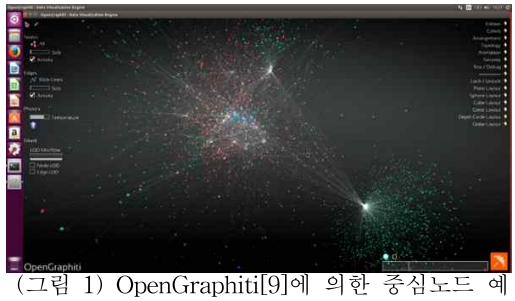
웹에는 악성코드를 전파하는 숨겨진 네트워크가 존재하는데, 그것은 우리가 접속하는 수많은 웹사이트 속에 URL의 형태로 서로 연결되어 존재한다. 우리가 이 URL를 찾는 것은 생각만큼 쉽지 않다. 공격자는 이 악의적인 URL의 탐지를 회피하기 위해 정상 URL처럼 보이려고 시도한다. 누군가는 기계학습 방법을 이용하여 악성 URL과 정상 URL의 차이를 높은 확률로 찾을 수 있다고 말하지만, 정상 URL도 악성 URL만큼 길고, 난독화된 코드도 존재하는 등 유사한 형태가 존재하여 실제 환경에서 이 방법을 적용해 찾아내는 경우는 드물다. 따라서, 우리가 이 악성 URL을 찾는 방법 중 한가지 방법은 Drive-by Downloads 과정에서 다운로되는 파일의 악성 여부를 판단하여 역으로 악성 URL을 한단계씩 역추적하여 찾아가는 방법이다. 이전 연구에서는[1, 2, 3, 4] 랜딩사이트(사용자가 처음 접속하는 사이트)부터 redirect되는 모든 URL을 수집해 놓았다가 drop되는 파일의 악성여부가 확인되면 관련된 URL들을 악성 URL로 판명하였다. 그 사용된 악성 URL들을 연결한 것이 악성코드배포네트워크(Malware Distribution Network, MDN)이다. 또한, 동일 해시값을 갖는 악성코드 및 유사성이 높은 해시값을 갖는 악성코드 배포 URL을 동일 공격자로 판단하여 같은 네트워크로 묶었다. 유사 IP(예를 들어 65.35.32.3, 65.35.32.6처럼 서브넷 마스크 /24가 동일한 IP)들을 동일 공격자로 판단하고 이런 유사성 및 동질성(homogeneity)을 갖는 악성코드, IP 들을 하나의 큰 네트워크로 연결하여 만든 것이 MDN이다. 이런 MDN들은 공격자 별로 서로 다르게 존재한다.

이런 형태는 공격자가 생산적이고 자동화된 공격을 수행하기 위해 습관적으로 유사한 행동 패턴을 보여주기 때문에 나타나는 현상이다[5, 6]. 본 논문은 이런 속성들을 기반으로 악성코드배포네트워크를 구성하였다. 따라서 인터넷에는 각기 다른 공격자 MDN이 존재하고, 그 MDN들이 묶여진 큰 MDNs이 존재한다. 이 연구는 각 MDN의 위험수준을 평가하여 어떤 MDN의 위험수준이 가장 높은지를 판별하는 방법을 제시하고, 또한 MDN과 정상네트워크의 특성을 비교하여 어떤 차이점이 있는지를 밝히는데 그 목적이 있다.

2. 관련 연구

2.1 Degree Centrality

선행 연구들은 주로 social network(예를 들어, Facebook 등) 상에서 악성코드를 전파하는 노드들 간의 관계를 살펴보았다면[7], 본 연구는 악성코드 전파력을 고려한 위험도를 측정하여 네트워크의 영향력을 분석하였다. 이 논문은 MDN에서 중심성 및 구성요소 간 분석을 통해 MDN에서 핵심노드를 찾는 것이다. 이 연구는 주로 노드의 중심성을 찾기 위한 그래프 이론에 기반한다[8].



(그림 1) OpenGraphiti[9]에 의한 중심노드 예

네트워크를 구성하는 대상은 랜딩사이트, 악성 URL, 악성코드 등이다. 중심성은 중앙 노드에 미치는 영향력이다. Degree centrality는 해당노드가 가지는 다른 노드와의 연결개수를 말하며, 다른 노드들과 연결이 많을수록 중요도는 높아진다.

2.2 악성 URL

악성코드 유포 시점의 URL들은 서로 체계적으로 연결된다. 악성코드 유포에 사용된 URL들을 시간에 따라 누적해서 보면 복잡한 연결 구조성을 보인다. 즉, 상대적으로 정상네트워트보다 MDN이 더 동적 네트워크를 구성한다.

MDN을 구성하는 악성 URL 중에는 사용자가 처음 접속하는 랜딩사이트 URL, redirects, 배포 URL로 구성되어 있고, 배포 URL은 공격코드가 있는 웹사이트 URL과 공격 성공 후 악성코드 배포에 사용되는 URL로 구성된다. Redirect들은 랜딩과 배포사이트를 연결하는 n개의 패싱 URL들로 구성된다. 여기서 n은 보통 수개의 임의의 수이다. 이들의 역할은 서로 다른 웹페이지를 넘나들면서 보안전문가의 추적을 회피하거나 자신의 존재를 숨기는 역할을 한다. 때때로, 랜딩페이지라 불리기도 한다. 공격코드가 있는 사이트의 URL을 익스플로잇 URL과 같이 다양한 용어로 설명하고 사용되기도 한다.

2.3 MDN 속성

본 논문은 악성코드 배포 노드에 위험 수준을 할당하기 위해 그래프 기반 지표(주로 중심성 지표)를 제안한다. 악성코드 배포에서 가장 영향력 있는 노드를 성공적으로 식별하기 위해 고위험 노드를 찾는 위험 분석 모델을 제안한다. 이 결과를 통해 악성코드 유포지점을 효과적으로 식별하고 제거할 수 있다. 현재는 웹의 악성 URL 발견 시 해당 노드를 하나씩 제거하는 방식을 사용하고 있다. 이 모델은 중요한 위험을 식별하고 완화 전략을 지원하기 위한 접근 방식을 제공한다. 일반적으로 공격코드인 EK는 해외사이트에 존재하여 우리의 제어밖에 존재함으로, 상대적으로 접근 가능한 중심노드를 찾는 방법은 매우 중요하다.

먼저 데이터 수집을 위해 LoGos[1], WebMon[2]과 Thug[3]가 사용되었다. 이들로부터 수집된 악성 URL들은 중심성을 찾기위해 NetMiner[10]를 사용하였고, 시각화를 위해 OpenGraphiti[9]와 Gephi[11]를 사용하였다.

본 논문은 MDN이 매 순간 바뀌는 상황(예를 들어, 공격자가 일시적으로 악성 URL을 삭제하여 기존 링크가 사라지고, 또다시 나타나는 상황 및 악성 URL이 다른 URL로 변경되는 일련의 과정)에서 악성코드를 배포하는 중심 경로를 신속히 찾고 차단하는 방법을 제공한다. 이를 위해 랜딩페이지의 Alexa[12] 순위가 중요하다. 상위 Alexa 순위를 갖는 웹사이트의 경우 전파력이 크고, 상대적으로 Alexa 순위가 낮은 사이트의 경우 감염에 따른 전파력은 낮다. 따라서, 악성코드 감염 측면에서는 핵심노드를 찾는게 중요하지만, 악성코드 전파력 측면에서는 전파력이 높은 웹사이트가 중요하다. 이를 종합적으로 고려한 방법이 본 연구가 제안하는 **위험분석기반 중심성 측정 방법**이다. 공격자는 전파력이 높은 Alexa 순위 웹사이트를 랜딩페이지로 사용하고, 전파력이 낮은 웹사이트는 공격코드를 삽입하여 악성코드 배포에 사용한다. 공격자는 이런 특성을 이용하여 악성코드 전파에 활용한다.

이 논문은 시시각각 변화하는 네트워크 레이아웃을 들여다보기 위해 NetMiner와 OpenGraphiti를 이용하여 시각화하였다. 시각화하여 보면, 악성 링크들은 중심노드로 향하고(그림 1), 이 중심노드에 사용자 시스템을 공격하는 EK가 위치한다.

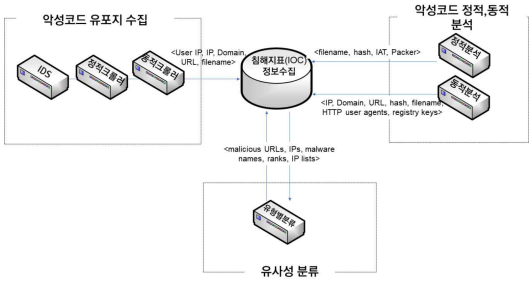
3. 제안 모델

3.1 데이터셋 수집

본 연구를 위해서 먼저, 악성 URL 수집 시스템을 구축했다[1, 2, 3]. 둘째, 악성 URL의 속성을 파악하였다[5]. 예를 들어, ccTLD를 분석해보면, 사용자가 접속한 사이트가 국내 웹사이트라도 공격코드가 존재하는 사이트는 해외에 위치하고 그 비중이 약 70%라는 사실이다[5]. 이런 지리적 차이로, 공격코드가 있는 해외 서버를 차단하기 어려워 사용자가 접속하는 국내 접속 사이트를 차단해야 하는 한계가 존재한다. 따라서, 우리는 이런 차단 측면에서 가장 핵심이 되는 악성 URL을 파악하는 것이 무엇보다 중요하다. 마지막으로 이런 악성 URL을 복잡계 네트워크로 재구성하여 악성코드배포네트워크의 속성을 파악하는 것이

다. 이 논문은 URL을 기준으로 복잡계 네트워크를 구성할 경우, 정상과 비정상 네트워크의 근본적 차이점을 분석하여 연구자들에게 이해를 돕고, 악성코드배포네트워크에서 핵심노드를 찾는 방안을 제시한다.

실제 데이터셋을 수집하기 위해 IDS(Suricata 3.2.1), 정적 URL 및 EK 수집 크롤러 [1, 2]를 ESXi에 구현하여 악성파일 및 악성 URL을 수집하였고, 상용 가상머신 기반 분석기 Joe Sandbox 및 소셜 네트워크 분석 어플리케이션[10] 등을 통해 데이터를 분석, 그래프화하였다. (그림 2)는 데이터 수집, 분석, 그래프화 전체과정에서 사용된 시스템 구조도이다.



(그림 2) 수집·분석 시스템 구조

Suricata IDS는 로컬 사용자가 접속하는 도메인을 추출하는데 사용하였다. 이 수집 도메인을 기반으로 크롤러[1, 2]를 사용하여 접속 도메인의 악성 URL 포함 여부를 확인했다. 접속 도메인에 악성 URL이 포함된 경우, 악성코드 수집기를 통해 악성코드를 수집했다 [1].

이후 수집된 악성코드는 peframe과 ssdeep을 사용하여 악성코드 간 유사도를 측정했다. 90% 이상 해시 유사성은 동일 악성코드 변종으로 판단했다. 상용 분석기 Joe Sandbox를 통해 악성코드가 접촉하는 악성 IP도 추출하였다. 상용제품을 이용하여 악성코드를 랜섬웨어, 뱅커 등 악성코드 유형을 Computer Antivirus Research Organization (CARO) 기준에 따라 분류하였다.

본 실험에 사용된 데이터셋은 1,271개의 악성 URL, EKs(예를 들어, Gondad, Magnitude), C2서버, 악성코드 및 이들의 유사성 정보를 기반으로 한다. 악성코드와 악성 URL은 VirusTotal에 의해 확인되었다. 정상데이터셋은 8,376개의 URL로 구성되었으며, Alexa 상위 500개 사이트에서 샘플링되었고, VirusTotal로 검증하였다. 수집된 데이터셋은 2023년 3월 수집된 RedLine Stealer 악성코드는 동일한 RIG EK를 통해 유포되듯이, 현재도 유효하다.

3.2 MDN 위험 분석 모델

수집된 데이터셋으로 악성코드 전파의 영향을 검증하기 위한 구체적인 위험 분석 모델을 다음과 같이 제안한다. (1)에서 보듯이 중심노드의 영향력 C_i는 랜딩페이지의 개수(L^n)와 Alexa 순위가 갖는 영향력 A_i, 랜딩페이지 내에 숨겨져 있는 redirect 연결 링크 수 L_c의 곱과 비례하고, 랜딩페이지와 공격코드가 있는 배포 웹사이트 사이의 redirect 개수와 반비례한다. redirect와 중심노드 사이의 거리가 멀수록 탐지될 확률이 높아진다. $r_\alpha d_{ij}$에서 r_α는 redirect의 α번째를 의미하고. $r_\alpha d_{ij}$는 redirect와 중심 노드 사이의 거리를 의미한다.

$$C_i = \sum_{n=1}^{m} L^n A_i L_c \times \frac{1}{\sum_{\alpha=1}^{\beta} r_\alpha d_{ij}} \quad (1)$$

일반적인 자산평가에서 사용하는 위험분석 모델의 경우 Annual Loss Expectancy = Asset Value × Exposure Factor × Annual Rate Occurrence로 표현한다. (1)은 이와 유사하게 사용자가 처음 접속하는 URL 노드들에 대한 위험수준 평가를 위해 랜딩페이지 수, Alexa 순위 기반 방문자 수, 랜딩 페이지와 연결된 악성링크 수 및 공격코드 연결 시간에 비례하고, 최종 배포에 종사하는 랜딩웹페이지-공격코드 노드까지의 거리에 반비례한다. 노드간 거리가 멀수록 악성 URL이 탐지될 확률은 증가하기 때문이다. 네트워크에서 centrality는 노드들과의 연결(edge) 정도를 측정하여 각 노드들이 네트워크에서 얼마나 중심에 위치하는지를 degree 측면에서 측정할 수 있다. 따라서 위험수준은 하나의 네트워크로부터 산출되는 값으로 다른 네트워크와 비교되는 위험의 정도를 측정할 수 있다. (그림 8)은 MDN의 위험분석모델 (1)을 적용하여 그린 그래프로, 실제 (그림 8)의 2가 더 복잡한 네트워크 일지라도, 악성코드

배포 영향도 측면에서 (그림 8)의 1이 더 위험한 네트워크가 될 수 있다.

Alexa 순위가 높은 웹사이트들이 모두 안전하게 관리된다고 보장할 수 없다. 일반적으로 공격자는 랜딩사이트로 고순위의 Alexa 웹사이트를 사용하고 공격사이트로 저순위 Alexa 웹사이트를 채택한다. 그래야 악성코드 감염력이 높기 때문이다. 따라서 웹사이트를 통한 감염의 수준을 측정할 경우, Alexa 순위는 이런 특성을 반영하는 요소로 사용될 수 있다. 따라서 본 실험에 사용된 데이터셋은 Alexa 특성을 반영하여 전체 MDN의 위험도를 측정하도록 설계되었다. 이전 연구들은 어떻게 Drive-by Downloads를 효과적으로 탐지하는지에 더욱 초점화되어있어, 본 연구와는 차별화된다. 제안모델 (1)은 centrality node로 연결된 주변 노드들의 연결성과 centrality까지의 깊이와 관련이 있다. 단위 시간당 centrality로 연결된 노드의 수가 많아도 Alexa 순위가 낮은 랜딩사이트는 노드의 수가 적은 고순위 Alexa 랜딩사이트 보다는 덜 위험하다. 이를 반영한 수식 모델이 (1)이다.

4. 실험

이 섹션에서는 MDN에 대한 위험 분석 모델이 적용된 실제 네트워크 레이아웃 결과를 제시한다. 이를 위해 이전 섹션에서 말했듯이 MDN 및 정상네트워크 상에서 데이터셋을 수집했다. 특히 악성 데이터셋은 EK로 연결된 redirect들을 포함한다. 이 링크들을 조작할 때 MDN이 정상네트워크와 다른 근원적 속성차이를 보여준다.

4.1 데이터셋 수집

4.2 실험 환경

NetMiner 4, Java 1.3.0, Python 2.7.13을 사용하여 그래프를 생성하였다. Intel Xeon 6Core 2.4Ghz X 2ea, 128G M/M, 8TB HDD 및 1G Ethernet 2Port가 장착된 시스템에서 테스트하였다.

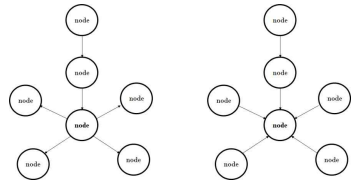
NetMiner는 Social Network Analysis 소프트웨어로 노드와 엣지를 가지는 MDN을 2차원 형상 그래프로 생성한다. 이 소프트웨어를 통해 공격자가 사용하는 네트워크 트리에서 중심성(Centrality) 노드에 대한 정보를 찾았다. 이 접근 방식은 각 MDN 별로 핵심노드를 찾고 별도로 작성한 Python 프로그램을 통해 전체적인 위험수준도 측정하였다.

4.3 실험 결과

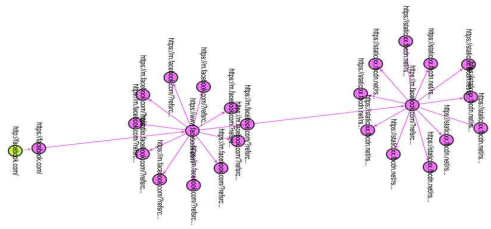
다음은 정상·악성네트워크 간 차별성을 설명한다.

4.3.1 네크워크 그래프 레이아웃에서 중심노드

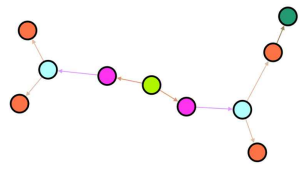
(그림 3)은 정상과 악성네트워크의 in-degree와 out-degree 측면에서의 실험으로 정상네트워크는 왼쪽 모습의 out-degree 형태를 보였으며, 악성네트워크는 in-degree 형태를 보였다. 예를 들어, (그림 4)은 Gephi[11]로 그린 Facebook의 그래프 레이아웃으로 화살표 방향이 중심에서 밖으로 향하는 out-degree 모습을 보였으며 반면에, 악성코드를 배포했던 ipeacetv.com 사이트(그림 5)는 반대로 중심노드로 모이는 in-degree 형태를 보였다. 실험한 모든 네트워크가 동일한 특징을 보였다.



(그림 3) 정상/악성네트워크 in-과 out-degree 비교



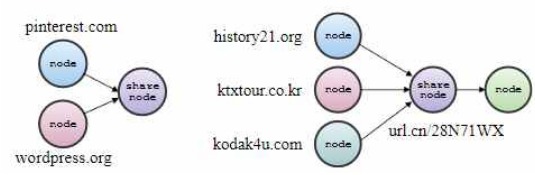
(그림 4) Facebook의 out-degree 레이아웃



(그림 5) ipeacetv.com의 in-degree 레이아웃

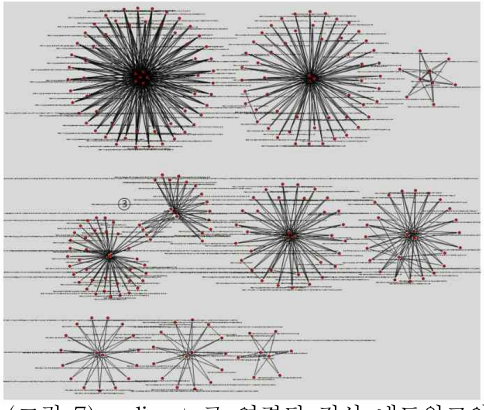
4.3.2 네트워크 그래프 레이아웃에서 공유노드

본 논문에서 사용한 데이터셋을 이용하여 노드-엣지 기반 네트워크를 그렸을 경우, 정상 랜딩 웹사이트인 pinterest.com와 wordpress.org의 경우 내부 웹사이트 코드에서 동일 URL을 공유하는 경우가 존재하였다 (그림 7의 3). 이런 경우, (그림 6)의 좌측과 같이 공유노드로 집중되지만, 공유노드를 통해 다른 노드로 확장되는 경우는 발생하지 않았고, 반면에 MDN의 공유노드는 (그림 6)의 우측과 같이 경유지로 사용되었다.



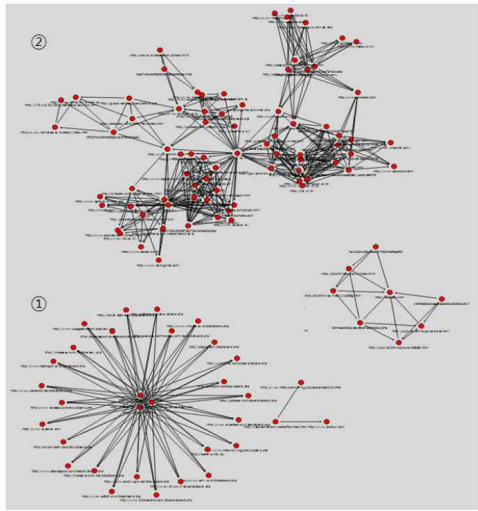
(그림 6) 정상 네트워크(왼쪽)와 악성 네트워크(오른쪽) 공유노드 차이

4.3.3 정상·악성 네트워크 그래프 레이아웃 형태



(그림 7) redirects로 연결된 정상 네트워크의 그래프 레이아웃 실제 예

Alexa 순위가 높은 웹사이트는 Alexa 순위가 낮은 웹사이트에 비해 공격자들이 숨긴 악성 링크를 찾아 제거할 확률이 더 높다. Alexa 순위가 높은 웹사이트일수록 안전하게 관리될 가능성이 높기 때문이다. 그러나 Alexa 순위가 높은 웹사이트가 손상되면 위험의 크기는 더욱 크다. 고위험이란 상대적으로 방문자 수가 많은 노드를 말하며, 이는 악성 웹사이트에 접속하는 사용자 PC의 수가 많다는 것을 의미한다. 따라서 PC의 악성코드 감염 가능성이 높다. 따라서 Alexa 순위가 높은 웹사이트는 악성 사이트로 사용될 경우 위험도가 높지만, 실제 환경에서는 관리가 잘 되고 있어 위험도 측면에서 Alexa 순위가 낮은 웹사이트보다 안전하다.

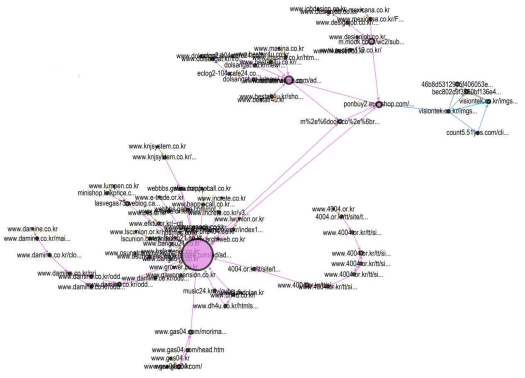


(그림 8) redirects들을 포함한 MDN 실제 예.

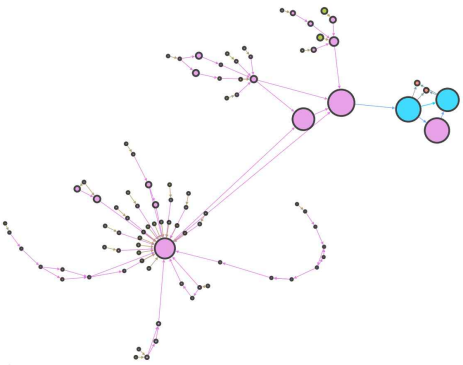
(그림 7)는 실제 Alexa 순위가 높은 정상 웹사이트들(Facebook, Youtube 등)을 NetMiner로 생성한 그래프로 1개월 기간의 변화를 나타낸 모습이다. in-degree 형태를 가지고 있고, 방사형 형태로 정형화되어 있다. MDN 보다 상당히 정적인 네트워크이다. 반면에 MDN의 경우 (그림 8)의 1처럼 초기에는 정상네트워크와 유사한 형태를 가지고 출발하지만 시간이 지속될수록 2와 같이 복잡한 MDN의 형태로 변화한다. MDN은 중심 node로 집중하는 in-degree 형태를 보이지만, (그림 8)에서 보듯이, 다양한 형태의 그래프 레이아웃을 보이기도 한다. 정상 네트워크에 비해 네트워크 레이아웃의 변화가 상당히 동적이다.

(그림 9)는 위험분석 모델이 적용되지 않은 악성 URL의 연결성을 보여주는 악성코드배포네트워크이고, (그림 10)은 위험분석 모델 (1)을 적용하여 그린 악성코드배포네트워크이다. (그림 10)에서 보듯이 노드의 크기가 (그림 9)와 다르며, 중요도 측면에서도 상당이 다름을 알 수 있다. 단순히 node-edge 연결성만을 고려한다면 연결수가 많은 노드가 중요하지만, 노드가 가질 수 있는 영향력을 고려하면, (그림 10)과 같이 c

entrality는 다르게 된다. 우리는 이 노드의 크기를 고려하여 악성 링크를 차단하는 것이 악성코드 배포를 신속히 차단하는 측면에서 효과적이라고 생각한다.



(그림 9) 위험분석 없는 MDN 형태.



(그림 10) 위험분석 적용 MDN 형태.

5. 결 론

MDN은 악성 URL로 구성된 악성코드 배포 네트워크이다. 악성 URL, 악성코드 및 C2 서버를 포함하고 방향성을 갖는 엣지와 노드들로 구성된다. 특히 노드는 위험 크기를 포함하고 있고, 엣지는 악성코드 전파경로를 나타내는 다양한 연결로 구성된다. 이전 연구는 이런 MDN의 구조적 특성을 설명하지 못했다. 따라서 이 논문은 정상네트워크와 MDN의 서로 다른 특성을 설명하고, 악성코드 유포에 관여도가 높은 노드를 탐색하는 모델을 제시하였다. 실 환경에서는 해외에 위치한 주요 노드를 제거하는 것이 어렵다. 따라서, 중심노드 대신 제어 가능한 중요 노드를 신속히 찾아 제거하는 것이 필요하다. 이런 측면에서 본 논문은 그 가능성을 제시한다. 이 방법은 이전 연구를 바탕으로 MDN의 존재를 알리고, 핵심 노드를 찾아 효과적으로 제거하기 위한 방법을 제시한 측면에서 큰 의미가 있다고 본다.

참고문헌

[1] S. J. Kim, S. K. Kim and D. H. Kim, “LoGos: Internet-Explore-Based Malicious Webpage Detection”, ETRI Journal, Vol. 39, No. 3, pp. 406-416, 2017.

[2] S. J. Kim, J. K. Kim, S. W. Nam and D. H. Kim, “WebMon: ML-and YARA-based malicious webpage detection”, Computer Networks, Vol. 137, pp. 119-131, 2018.

[3] Thug, https://buffer.github.io/thug/, 2023.

[4] T. Nelms, R. Perdisci, M. Antonakakis and M. Ahamad, “Webwitness: Investigating, categorizing, and mitigating malware download paths”, In 24th {USENIX} Security Symposium ({USENIX} Security 15), pp. 1025-1040, 2015.

[5] S. Kim, J. Kim and B. B. Kang, “Malicious URL protection based on attackers′ habitual behavioral analysis”, Computers & Security, Vol. 77, pp. 790-806, 2018.

[6] S. Huh, S. Cho, J. Choi, S. Shin and H. Lee, "A Comprehensive Analysis of Today’s Malware and Its Distribution Network: Common Adversary Strategies and Implications", EEE Access, Vol. 10, pp. 49566-49584, 2022.

[7] H. Gao, J. Hu, C. Wilson, Z. Li, Y. Chen and B. Y. Zhao, “Detecting and characterizing social spam campaigns”, In Proceedings of the 10th ACM SIGCOMM conference on Internet measurement, pp. 35-47, 2010.

[8] Centrality, https://en.wikipedia.org/wiki/Centrality.

[9] OpenGraphiti, https://www.opengraphiti.com/, 2015.

[10] NetMiner, http://www.netminer.com, 2023.

[11] The Open Graph Viz Platform, https://gephi.org/, Gephi, 2022.

[12] Alexa, https://en.wikipedia.org/wiki/Alexa_Internet.

〔저자소개〕

김 성 진 (Sungjin Kim)
2000년 8월 Ohio State Univ. 학사
2004년 2월 서강대학교 석사
2019년 2월 KAIST 박사
2019년 9월 ~ 현재 제주한라대학교
인공지능공학과 조교수
email : r3dzon3@chu.ac.kr