네트워크 악성행위 분석시스템

최선오, 최양서, 이종훈, 이주영, 김종현, 김익균
전자통신연구원 네트워크보안연구실
e-mail : suno@etri.re.kr

Network Abnormal Behavior Analysis System

Sunoh Choi, Yangseo Choi, Jonghoon Lee, Jooyoung Lee, Jonghyun Kim, Ikkyun Kim
Network Security Research Group, ETRI

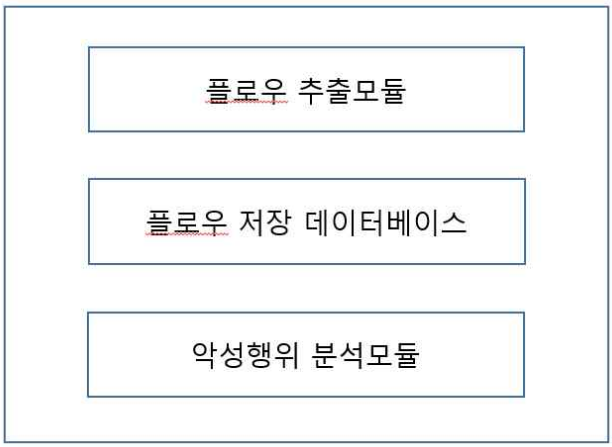
요 약

요즘 사이버 공격이 많이 발생함에 따라 기존의 디지털 포렌식 뿐만 아니라 네트워크 트래픽을 수집해서 사이버 공격을 분석하는 네트워크 포렌식에 대한 연구가 많이 수행되고 있다. 그러나 네트워크 포렌식을 수행하기 위해서는 여러가지 어려움이 존재한다. 이러한 문제를 해결하기 위하여 우리는 이 논문에서 네트워크 악성행위를 분석하기 위한 시스템 및 방법을 제안한다.

1. 서론

요즘 사이버 공격이 많이 발생함에 따라 기존의 디지털 포렌식 뿐만 아니라 네트워크 트래픽을 수집해서 사이버 공격을 분석하는 네트워크 포렌식에 대한 연구가 많이 수행되고 있다. 그러나 네트워크 포렌식을 수행하기 위해서는 여러가지 어려움이 존재한다. 특히 네트워크 포렌식에 대한 경험과 지식이 부족한 사람들이 네트워크 악성행위를 분석하는 일은 매우 어렵다.

이러한 문제를 해결하기 위하여 우리는 이 논문에서 네트워크 악성행위를 분석하기 위한 시스템 및 방법을 제안한다. 먼저 네트워크 악성행위 분석시스템의 세부모듈을 소개하고 그 다음으로 악성행위 분석 모듈에서 사용하는 네트워크 악성행위 분석방법을 소개한다.



(그림 1) 네트워크 악성행위 분석시스템

2. 네트워크 악성행위 분석시스템

이 논문의 목적은 사이버 공격을 분석할 수 있는 네트워크 악성행위 분석장치를 제안하고 그 방법을 소개하는 것이다.

이 논문은 크게 두 부분으로 이루어진다. 첫째는 네트워크 악성행위 분석시스템의 각 모듈의 기능을 소개하는 것이다. 둘째는 네트워크 악성행위 분석방법을 소개하는 것이다.

그림 1 과 같이 네트워크 악성행위 분석시스템은 크게 3 개의 모듈로 구성된다. 네트워크 악성행위 분석시스템의 입력은 네트워크 패킷들이 저장된 pcap 파일이다. Pcap 파일이 네트워크 악성행위 분석장치에 주어지면 첫째로 플로우 추출모듈에서 pcap 파일로부터 네트워크 플로우 메타정보를 추출하게 된다. 네트워크 플로우 메타정보는 다음과 같은 포맷을 가진다.

(sourcehost, sourceport, destinationhost, destinationport, packets, bytes, protocol, service, filetype, tcpflag, starttime, endtime)

네트워크 플로우 메타정보는 2 개의 네트워크 장비간에 네트워크 패킷들이 어떻게 전송되었는가에 관한 것이다. Sourcehost 는 네트워크 패킷을 보낸 장비의 아이피를 의미하고 sourceport 는 네트워크 패킷을 보낸 장비에서 사용되는 포트번호를 의미한다. Destinationhost 는 네트워크 패킷을 받는 장비의 아이피를 의미하고 destinationport 는 네트워크 패킷을 받는 장비에서 사용되는 포트번호를 의미한다. Packets 는 해당 플로우에서 보낸 패킷의 개수를 의미하고 bytes 는 해당 플로우에서 보낸 바이트수를 의미한다. Protocol 은 해당 플로우에서 사용되는 프로토콜을 의미하고 service 는 해당 플로우에서 사용되는 서비스를 의미한다. Filetype 은 해당 플로우에서 파일이 전

송된다면 플로우 추출모듈에서는 해당 파일의 시그니처를 탐지하여 파일타입을 기록한다. Tcpflags 는 해당 플로우에서 보내진 패킷에 있는 tcp 플래그 정보의 누적값이다. Starttime 은 해당 플로우의 시작시간을 의미하고 endtime 은 해당 플로우의 종료시간을 의미한다.

플로우 추출모듈은 pcap 파일을 읽어서 위의 포맷에 따라 플로우 메타정보를 생성하고 이것을 네트워크 악성행위 분석장치의 두번째 모듈인 플로우 저장 데이터베이스에 저장한다.

두번째로 플로우 저장 데이터베이스는 고속로딩 및 고속검색을 지원하기 위하여 비트맵 인덱스[1, 2]를 사용한다. 여기서 로딩이라고 함은 네트워크 플로우 정보를 데이터베이스의 테이블 형태로 저장하는 과정을 의미한다. 비트맵 인덱스는 대용량 데이터 검색을 위하여 인덱스를 비트맵 형태로 관리하는 것을 의미한다.

네트워크 악성행위 분석시스템의 세번째 모듈인 악성행위 분석모듈은 미리 정의된 네트워크 악성행위 분석방법에 다라 플로우 저장 데이터베이스에 질의를 하게 된다. 그리고 질의에 해당하는 결과가 있으면 사용자에게 결과를 보여주게 된다.

다음으로 이 논문의 두번째 부분인 네트워크 악성행위 분석방법에 대해 소개한다. 이 논문에서는 대표적으로 지속적 외부연결에 해당하는 네트워크 악성행위 분석방법을 소개한다.

지속적 외부연결은 내부시스템이 지속적으로 외부시스템에 연결을 하기 위해 SYN 패킷을 보내지만 외부시스템의 해당 포트가 열려져 있지 않기 때문에 외부시스템이 내부시스템으로 RST 패킷을 보내는 것이다. 이것은 주로 악성코드에 감염된 내부시스템이 외부 C&C 시스템에 접근하려고 할 때 많이 나타나는 증상이다. 이것은 악성행위 분석모듈에서 다음과 같은 질의를 플로우 저장 데이터베이스로 보내는 것을 통해 분석할 수 있다.

```
Select sourcehost, sourceport, destinationhost, count(destinationport) from flowtable
Where tcpflags = 20 group by sourcehost, sourceport, destinationhost,
Having count(destinationport) >= threshold;
```

이 질의의 의미는 다음과 같다. 여기서 tcpflags 의 값이 20 이라는 것은 이진수로 010100 을 의미한다. Tcp 플래그는 URG, ACK, PSH, RST, SYN, FIN 의 6 비트로 구성된다. 010100 은 해당 플로우에서 ACK, RST 플래그가 1 로 세팅된 패킷이 전송되었다는 의미이다. 이것은 소스아이피에 해당하는 장비가 목적지아이피에 해당하는 장비로 RST 패킷을 보냈다는 의미이다. 그리고 위의 질의에서 목적지아이피에 해당하는 장비가 소스아이피에 해당하는 장비로 SYN 패킷을 보냈을때 소스아이피에 해당하는 장비가 목적지아이피에 해당하는 장비로 RST 패킷을 보냈는지 찾는 것이다. 이것을 모든 플로우에 대해 한번에 처리하기 위하여 우리는 sourcehost, sourceport, destinationhost 로 그룹화하고 사용되는 destinationport 의 개수가 미리 정의된 threshold 이상인 것을 지속적 외부연결에 해당하는 악성행위로 정의한다.

예를 들어 위의 질의의 결과가 (A, B, C, D)와 같이 주어진다고 하면 감염시스템 C 는 외부시스템 A 의 B 번포트로 D 번 지속적으로 외부연결시도를 한 것에 해당된다. 외부시스템 A 의 B 번 포트가 아직 열려져 있지 않기 때문에 외부시스템 A 는 감염시스템 C 로 D 번 RST 패킷을 보낸 것이다.

3. 결론

우리는 이 논문에서 네트워크 악성행위를 분석하기 위한 시스템을 제안하고 네트워크 악성행위 분석방법을 제안하였다. 이 시스템과 분석방법을 사용함으로써 네트워크 포렌식에 대한 경험과 지식이 부족한 사용자들도 네트워크 악성행위 분석을 수행할 수 있을 것으로 기대한다.

ACKNOWLEDGEMENT

이 논문은 2016 년도 정부(미래창조과학부)의 재원으로 정보통신기술진흥센터의 지원을 받아 수행된 연구임 (No. R-20160222-002755, 맞춤형 보안서비스 제공을 위한 클라우드 기반 지능형 보안 기술 개발)

참고문헌

[1] https://en.wikipedia.org/wiki/Bitmap_index

[2] Sunoh Choi et.al., Performance Comparison of Relational Databases and Columnar Databases Using Bitmap Index for Fast Search of 10Gbps Network Flows, International Conference on Computer Science and its Applications (CSA), 2015