JKIICE Journal of the Korea Institute of Information and Communication Engineering

한국정보통신학회논문지 Vol. 22, No. 11: 1544~1553, Nov. 2018

하둡 기반의 사용자 행위 분석을 통한 기밀파일 유출 방지 시스템

유혜림[1] · 신규진[2] · 양동민[3] · 이봉환[1*]

A Digital Secret File Leakage Prevention System via Hadoop-based User Behavior Analysis

Hye-Rim Yoo[1] · Gyu-Jin Shin[2] · Dong-Min Yang[3] · Bong-Hwan Lee[1*]

[1*]Department of Electronics, Information and Communications Engineering, Daejeon University, Daejeon 34520, Korea
[2]iREX Inc., 137 Munye-ro, Dunsan 2-dong, Seo-gu, Daejeon 35241, Korea
[3]Grade School of Archive and Records Management, Chonbuk National University, Baekje-daero, Jonju-si, Jeollabuk-do, 54896, Korea

요 약

최근 산업 보안 정책에도 불구하고 기업의 내부 정보 유출이 심각하게 증가하여 산업별로 정보 유출 방지 대책을 수립하는 것이 필수적이다. 대부분의 정보 유출은 외부 공격이 아닌 내부자에 의해 이루어지고 있다. 본 논문에서는 이동식 저장매체 및 네트워크를 통한 기밀 파일 유출방지를 위한 실시간 내부 정보 유출 방지 시스템을 구현하였다. 또한, 기업 내의 정보 로그 데이터의 저장 및 분석을 위해 Hadoop 기반 사용자 행동 분석 및 통계시스템을 설계 및 구현하였다. 제안한 시스템은 HDFS에 대량의 데이터를 저장하고 RHive를 사용하여 데이터 처리 기능을 개선함으로써 관리자가 기밀 파일 유출 시도를 인식하고 분석할 수 있도록 하였다. 구현한 시스템은 이동식 데이터 매체와 네트워크를 통해 기업 내부로의 기밀 파일 유출로 인한 피해를 줄이는 데 기여할 수 있을 것으로 사료된다.

ABSTRACT

Recently internal information leakage in industries is severely increasing in spite of industry security policy. Thus, it is essential to prepare an information leakage prevention measure by industries. Most of the leaks result from the insiders, not from external attacks. In this paper, a real-time internal information leakage prevention system via both storage and network is implemented in order to protect confidential file leakage. In addition, a Hadoop-based user behavior analysis and statistics system is designed and implemented for storing and analyzing information log data in industries. The proposed system stores a large volume of data in HDFS and improves data processing capability using RHive, consequently helps the administrator recognize and prepare the confidential file leak trials. The implemented audit system would be contributed to reducing the damage caused by leakage of confidential files inside of the industries via both portable data media and networks.

키워드 : 데이터 유출방지, 기밀 파일, 네트워크 보안, 파일 보안, 하둡

Key word : Data Loss Prevention, Confidential File, Network Security, File Security, Hadoop

Received 27 September 2018, **Revised** 20 October 2018, **Accepted** 29 October 2018

*** Corresponding Author** Bong-Hwan Lee(E-mail:blee@dju.kr Tel:+82-42-280-2553)
Department of Electronics, Information and Communications Engineering, Daejeon University, Daejeon 34520, Korea

Open Access http://doi.org/10.6109/jkiice.2018.22.11.1544 print ISSN: 2234-4772 online ISSN: 2288-4165

Ⅰ. 서 론

글로벌 경쟁시대에 각 기업의 핵심기술은 기업의 경쟁력만 아니라 국가의 경쟁력과도 관련이 있다. 우리나라의 중소기업 및 벤처기업들은 수준 높은 산업기술을 보유하고 있다. 그러나 대기업의 경우에는 고가의 보안 솔루션 도입 및 정책을 통하여 기밀파일 유출을 방지하고 있는 반면, 중소기업 및 벤처기업의 경우에는 고가의 보안 솔루션을 도입하기에 어려움이 있는 것이 현실이다. 따라서 중소기업의 내부 기밀파일 유출로 인한 피해액은 매년 증가하고 있다[1][2].

이러한 기밀파일 유출의 대부분은 기업 내부자에 의해 이루어지고 있다. 내부자를 통한 유출경로는 USB, 네트워크, 인쇄물 등 매우 다양하다. 그 중 USB를 통한 파일 복사 및 이동은 사용자 이외에는 파일이 옮겨졌는지 알지 못하고 관리가 어렵다. 사용자 자신도 모르게 기밀파일이 유출될 가능성이 매우 높으며, 사용자가 개인의 이익을 위해 악의적으로 기밀파일을 유출 할 수도 있다.

본 논문에서는 산업보안 시스템에 투자할 여력이 없는 중소기업 및 벤처 기업에 특화된 하둡(Hadoop) 기반의 사용자 행위 분석을 통한 내부정보 통제 솔루션을 통해 내부 정보 유출을 방지하는 방안을 제안한다. 하둡을 사용함으로써 중소기업 및 벤처기업의 상황에 맞는 확장성을 제공하고 로그 데이터 및 대량의 패킷 데이터를 저장하고 빠른 속도로 분석하여 관리자에게 제공할 수 있다. 먼저 본 시스템에서 다루는 기밀 파일은 특정 바이트 패턴인 시그니쳐를 포함한 파일로 정의한다. 각 Agent PC에서는 이동장치 기밀파일 유출을 차단하고, Ubuntu 서버를 통하는 모든 패킷을 스노트(Snort)를 통해 실시간으로 기밀파일 유출을 차단한다. 또한, NAS (Network Attached Storage) 서버에서는 WinPcap 라이브러리를 통해 패킷을 수집한다. 각 PC는 모든 패킷 파일과 차단 내역을 하둡 시스템에 전송하여 R-Hive를 이용하여 MapReduce 과정을 거쳐 관리자가 원하는 데이터를 조회하거나 분석할 수 있는 시스템을 구축하여 내부정보 유출을 차단 및 예방할 수 있다.

Ⅱ. 관련 연구

2.1. DLP

DLP(Data Loss Prevention) 기술은 기밀 데이터 유출 방지를 위해 차단되어야 하는 기밀 데이터 패턴을 지속적으로 갱신해야 할 뿐 아니라 네트워크 기반의 DLP 경우에는 웹메일, 웹하드, 메신저 등의 프로토콜 변경에도 이를 즉각적으로 반영하고 갱신하여야 한다[3].

네트워크 기반 DLP는 이메일, 메시지, 웹하드, 웹게시판, FTP, P2P, SNS, 터미널 서비스, 프락시 서비스 등에서의 정보 유출에 대해 네트워크 상에서 감시하고 통제하는 역할을 수행한다. 네트워크 기반 DLP는 Monitor와 Prevent 등 2개의 그룹으로 나뉜다. Monitor는 Network Mirroring 방식으로 운영되며, 감시와 차단 전용으로 다양한 유출 경로에 대한 모니터링이 핵심이다. Prevent는 프록시 방식으로 원천적으로 유출을 사전 차단하는 방식을 의미하는데 대표적으로 Mail Proxy와 DB Proxy가 있으며, 사전 통제 기능 때문에 선호되지만 제공하는 프로토콜이 한정적이고 과도한 차단 현상으로 인해 실제 운영에서는 Monitor 형태가 선호된다. 종점 DLP/Discovery 기술은 USB 등의 미디어와 출력물을 통한 유출방지 기능을 처리한다. 예를 들어, USB로 복사하려는 파일에 주민번호, 계좌번호, 카드 번호 등 개인정보나 주요 기밀문서의 키워드 등이 포함되어 있을 경우 식별하고 차단한다. 기밀 데이터가 유출될 수 있는 다른 채널들은 접속을 금지하지만 USB나 출력물은 정당한 업무를 수행하기 위해 적극적으로 활용될 수 있기 때문에 통제의 대상이 된다. 또한, 최근 종점 DLP는 RF 통신, 블루투스 통신, 테더링 등 통신에 대한 차단 기능까지 제공한다[4][5].

2.2. 워터마크 기법

워터마크란 불법복제를 방지하기 위해 개발된 기술로 미술품이나 저작물 또는 지폐와 같은 한정된 경우에 사용되었으나 디지털 워터마크가 개발되면서 디지털 콘텐츠에 ID나 특수 기호를 삽입하거나, 영상, 음성 등의 신호에 특정한 코드나 유형 등을 삽입하는 기술이다. 이러한 방법으로 복제 등 불법행위를 방지하고, 데이터를 소유하고 있는 사람의 저작권이나 소유권을 보다 효과적으로 보호하기 위해 개발된 기술이다. 디지털 워터마크는 해당 프로그램의 소유자가 아닌 다른 사용자가 소프트웨어를 사용할 때 아무런 지장을 주지 않기 때문에 알아 챌 수 없고 원본이 유출될 경우에 복제된 프로그램의 경로를 찾아 낼 수 있다. 워터마크의 주요 기능

으로는 저작권 보호, 위조나 변조 판별, 불법 복제 추적, 무단 복사의 방지, 사용자 제어, 내용 보호, 내용 라벨링 등이 있다[6][7].

2.3. Snort [8]

Snort는 1998년 마틴 로시가 발표한 공개 네트워크 침입탐지시스템이다. 악의적인 사이트를 탐지하는 시스템 중 하나이며, 오랜 기간 동안 사용되어 오고 있는 오픈 소스 프로그램이다. 자동 Rule 설정 기능이 제공되며, 관리자가 수동으로 탐지 Rule를 설정하여 적용시킬 수도 있다[9].

Snort의 패킷처리 과정은 4가지로 나누어진다. 첫 번째는 Sniffer이며, Snort를 통과하는 모든 패킷을 수집하여 사용자가 보기 쉬운 형태로 변환한다. 두 번째로는 사전처리(Preprocessor)로 효율적인 공격 탐지를 위해 몇 가지 플러그인을 거쳐 매칭되는지 확인한다. 세 번째는 룰(rule) 기반의 탐지엔진(detection engine)으로 사전에 정의된 탐지 룰과 매칭 되는지 확인한다. 마지막으로 경고(Alert)/로깅(Logging)은 정책에 따라 로그를 기록하며, 로그는 텍스트 파일이나 데이터베이스에 저장할 수 있다.

2.4. Hadoop [10][11]

Hadoop 시스템은 여러 대의 컴퓨터를 마치 하나 처럼 묶어 대용량 데이터를 처리하는 기술이며, 구글의 GFS(Google File System)과 맵리듀스(MapReduce) 프레임워크를 구현한 시스템이다. Hadoop의 데이터 저장소인 HDFS(Hadoop Distributed File System)에 데이터들을 저장하여 분산처리 시스템인 맵리듀스 프레임워크로 데이터를 일괄 처리한다. Hadoop 2.0부터는 Yarn의 등장으로 상호 작용과 실시간 처리와 같은 다양한 데이터 처리를 제공하며, Hadoop 2.0의 시스템 구성도는 그림 1과 같다.

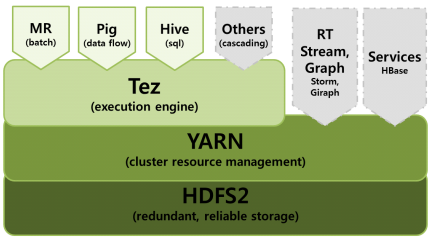

Fig. 1 Hadoop 2.0 system configuration[10]

HDFS는 Master/Slave 구조를 가지며, 일반적으로 HDFS 클러스터는 하나의 네임 노드(Name node)와 여러 개의 데이터 노드(Data node)로 구성된다. 네임노드는 이름과 위치 등의 메타 데이터를 관리해주며, 데이터 노드는 실제 데이터의 저장을 담당한다[9].

2.5. Apache Flume[12]

Apache Flume은 클라우데라에서 개발한 오픈 소스 로그 수집 소프트웨어이다. Flume은 데이터 수집을 위한 프레임워크로 다양한 로그 데이터 수집 및 모니터링이 가능하고 실시간 전송을 지원한다. JAVA로 구현되어 있기 때문에 다양한 운영체제에 설치가 가능하다. 또한, 장애에 쉽게 대처가 가능하며, 로그 유실에 대한 신뢰 수준을 상황에 맞게 변경할 수 있을 뿐만 아니라, 장애 발생 시 다양한 복구 메커니즘을 제공한다. Flume은 크게 OG, NG 두 가지의 버전으로 나누어지며, Flume-NG의 기본 아키텍처는 그림 2와 같이 Source, Channel 및 Sink로 구성되어 있다.

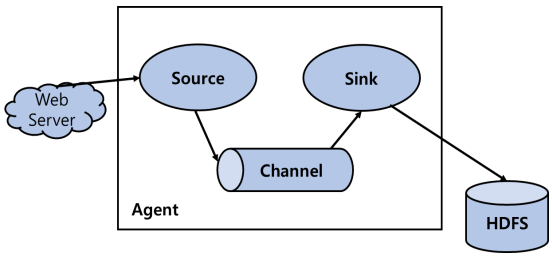

Fig. 2 The basic architecture of the Flume-NG

Ⅲ. 하둡 기반의 사용자 행위분석을 통한 내부 정보 유출방지시스템

본 논문에서 제안하는 기밀파일 유출 차단시스템의 구성도는 그림 3과 같다. 전체 시스템은 크게 시그니처 삽입관리자, 사용자 PC, NAS 서버, 우분투(Ubuntu) 서버, 하둡 시스템 등 5개 모듈로 구성된다. 사용자 PC들과 Ubuntu 서버는 하나의 스위치에 연결되어있고, 스위치의 포트 미러링 포트에는 NAS 서버가 연결되어 있다. 그리고 NAS 서버 후단에는 로그 데이터 등 대용량 데이터 처리를 위한 Hadoop 시스템이 연결되어 있다.

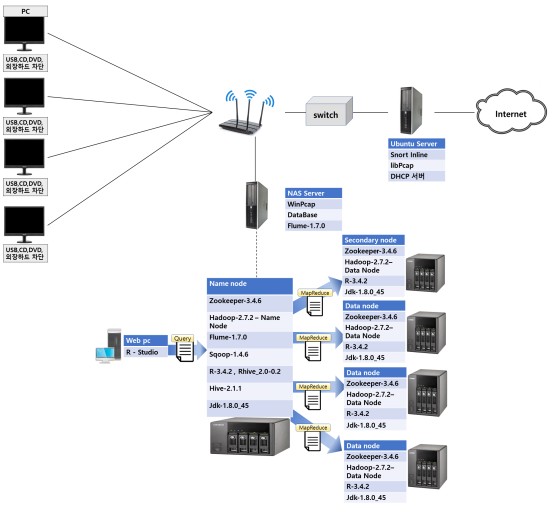

Fig. 3 Configuration of the secret file leakage prevention system

3.1. 시그니처 삽입 관리자

그림 4는 일반 파일을 기밀 파일로 전환하기 위해 시그니처를 삽입하는 알고리즘이다. 시그니처를 삽입하기 전 관리자가 폴더를 선택하고, 기밀파일로 변경하고자 하는 파일을 선택하면 관리자가 정의한 파일 확장자가 있는지 확인하고 파일 안의 시그니처 유무를 확인한다. 파일 안에 시그니처가 없을 경우 시그니처를 삽입하여 기밀파일로 분류한다. 시그니처 삽입 관리자는 .Net Framework에서 제공하는 File 클래스를 이용해 MS 오피스 파일은 XML파일을 추가하고, 일반 파일의 경우 .Net Framework에서 제공하는 Filestream 클래스를 이용하여 파일 필드 가장 마지막에 시그니처를 삽입한다.

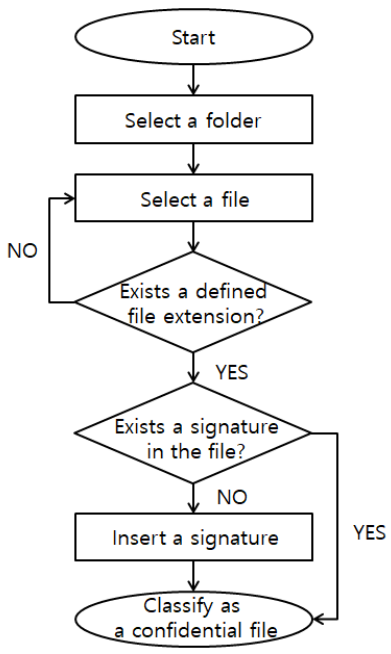

Fig. 4 Signature inserting process

3.2. 사용자 PC 시스템 구축 및 운용

그림 5는 사용자 PC에서 이동 매체를 통한 기밀파일 유출 차단 알고리즘을 나타낸 것이다. 이동식 저장매체를 통해 사전에 관리자가 정의한 기밀파일이 이동되면 시그니처 유무를 확인한다. 시그니처가 없을 경우에는 일반 파일로 분류하여 이동식 저장매체에 복사 또는 이동이 가능하지만, 시그니처가 있을 경우에는 시스템에서 기밀파일로 인식하여 해당 작업의 연산을 취소하여 기밀파일의 복사 및 이동을 방지한다. 차단된 내역은 NAS 서버의 데이터베이스로 전송한다.

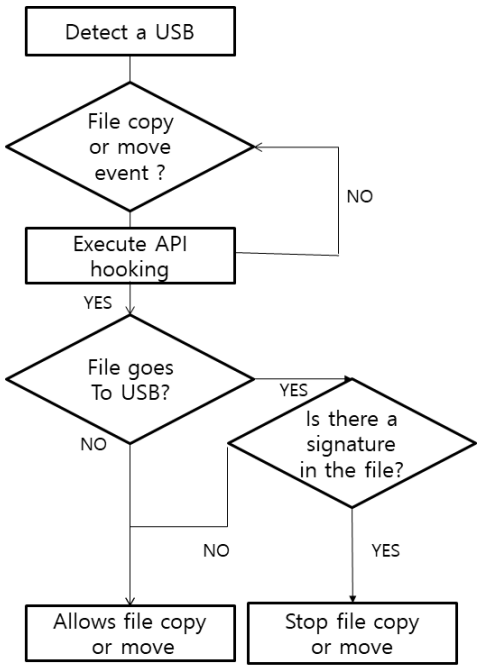

Fig. 5 Storage retrieval and confidential file protection process

3.2.1. 이동식 저장매체 탐지

USB, 외장하드, CD 등의 이동식 저장매체를 통한 기밀파일 유출을 탐지 및 방지하기 위해서는 시스템 상에서 먼저 USB, 외장하드, CD를 검출해야 한다. 우선 CD를 검출하기 위해서 .Net Framework에서 제공하는 DriveInfo 클래스를 활용한다. CD는 대부분 컴퓨터에 CD-ROM으로 먼저 연결이 되어 있기 때문에 드라이브의 작동 상태를 파악하여 CD 유무를 파악한다. USB와 외장하드의 유무 확인을 위해서는 Win32_Diskpartition과 Win32_LogicalDisk를 활용하여 시스템에서 확인한다.

3.3. NAS 서버 환경 구축 및 운용

NAS 서버에 데이터베이스를 구축하여 이동식 저장매체를 통한 유출 차단 로그를 데이터베이스에 저장하고, 스위치의 포트 미러링(port mirroring) 기능을 이용해 외부로 나가는 모든 패킷을 WinPcap 라이브러리를 이용해 사용자 PC의 네트워크 분석을 위한 Pcap 파일을 생성하고, 각각의 패킷에 시그니처 유무를 검사해 시그니처가 존재할 경우 데이터베이스에 패킷 정보를 저장한다. 또한, Hadoop으로 Pcap 파일을 전송하기 위해 Flume Agent를 설치해 설정한 폴더에 Pcap 파일이 생성될 경우 실시간으로 Hadoop 시스템으로 전송한다. 그림 6은 NAS 서버 구성도를 나타낸 것이다.

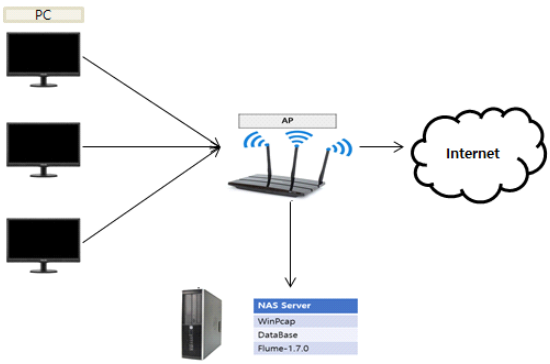

Fig. 6 NAS server configuration

3.3.1. 데이터베이스 구축

데이터베이스는 이동식 저장매체와 네트워크 유출 시도 내역을 저장하기 위하여 MySQL로 구축하였다. 이동식 저장매체나 네트워크에서의 차단은 가능하지만 관리자를 위해 로그를 남겨 관리한다.

3.3.2. Winpcap [13]

포트 미러링을 통해 전달되는 패킷들은 Visual Studio 2013에서 제공하는 Winpcap을 사용하여 구현하였다. 본 시스템에서는 내부 정보 유출 방지를 위한 시스템으로 내부에서 외부로 나가는 패킷만 Pcap 파일로 저장한다. NAS 서버에서는 패킷 차단을 수행하지 않고, 단일 패킷 페이로드만 검사하여 데이터베이스에 저장한다.

3.3.3. Flume Agent

Flume은 Collector와 사용자 PC에서 실행되는 Agent와의 연결이 필요하다. Collector 서버에서 Flume을 실행하면 Agent와 연결이 설정된다. Flume은 이기종 디바이스에서 실행이 가능하기 때문에 본 논문에서는 Windows에서 Flume Agent를 사용하여 Hadoop으로 Pcap 파일을 전송한다.

3.4. Ubuntu 서버 구축 및 라우팅 테이블 설정

Ubuntu 서버는 DHCP 서버와 Snort Inline으로 구성된다. Snort Inline과 DHCP 서버를 구축하기 위해서는 두 개의 랜카드(ens33, ens34)가 필요하다. Snort Inline을 이용하여 외부로 나가는 패킷의 페이로드에서 시그니처를 탐지하여 차단한다. 그림 7은 Ubuntu 서버 구성을 나타낸 것이다.

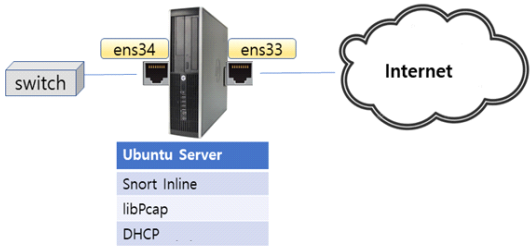

Fig. 7 Ubuntu server configuration

3.4.1. DHCP 서버 구축

ens34에 연결된 switch 하위 클라이언트 PC 및 AP들에게 IP를 자동으로 할당하기 위해 DHCP 서버를 구축한다. DHCP 서버를 구축하고 ens33과 ens34간의 네트워크 통신을 위해 라우팅 테이블을 설정한다.

3.4.2. Snort Inline rule set

Snort Inline을 실행하기 위해서는 libpcap 설치가 선행되어야 한다. Snort Inline에서 제공하는 NFQueue를

이용하여, 이더넷 랜카드에서 들어오는 모든 패킷은 NFQueue로 포워딩된다. Snort Inline에서는 시그니처를 탐색하고 차단하기 위해 선언된 룰에 의해 해당 패킷을 폐기한다.

3.5. 하둡 시스템 환경 구축 및 운용

분산파일 시스템을 구성하기 위해 총 5대의 노드를 이용한다. 즉, 한 대의 Name node, 한 대의 Secondary node 그리고 세 대의 Data node로 구성하였다. Name node는 Ubuntu 14.04 운영체제를 기반으로 하였으며, NAS 서버에 저장되는 정형화 데이터인 데이터베이스 정보를 받기 위해 Sqoop을 사용하고, 비정형화 데이터인 Pcap 파일을 받기위해 Flume을 사용하였다. 이 데이터 및 파일을 MapReduce를 사용하여 분석하기 위해 Hive를 구축한다. Hive 쿼리를 통해 얻어진 통계를 시각화하기 위해 R을 설치하고 Rserve와 RHive 패키지를 설치하여, 다른 PC에서 Rstudio웹을 통해 Name node의 R에 접속하여 Rhime 쿼리를 실행시켜 분석과 시각화를 용이하게 하였다. 그림 8은 분산파일 시스템의 구성도를 나타낸 것이다.

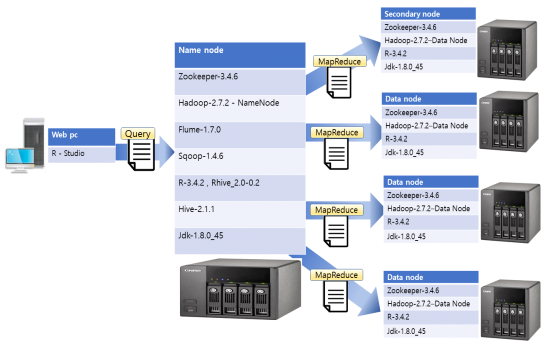

Fig. 8 Configuration of the Hadoop distributed file system

3.5.1. Pcap 파일 수집기

Apache Flume은 로그 데이터 수집 도구로 분산 환경에서 네트워크 패킷 등의 방대한 양의 데이터를 다양한 방법을 통해 사용자가 원하는 서버로의 전송이 가능하게 한다. 스위치에서 포트 미러링을 통해 수집되는 패킷을 지정한 특정 폴더에 저장할 때마다 지속적으로 Collector 서버에서 데이터를 수집한다. Flume은 사용자 PC Agent에서 서버 쪽으로 데이터를 보내는 구조를 가지고 있으며, 패킷을 수집하기 위해서는 Collector 서버를 먼저 실행하고 Agent Flume에서 클라이언트를 구동하여 연결 설정 후 전송하는 방식이다. Collector 서버와 Agent는 1대1 방식이 아닌 1대 다 방식으로 연결이 가능하며, 로그의 유형도 다양하기 때문에 Collector 서버는 다중 설정에 대한 다중 연결도 가능하다. Flume Agent도 Collector와 마찬가지로 Configuration 파일을 관리자의 요구사항에 맞게 명령어를 실행시킨다. 이와 같은 과정을 거치면 Collector와 Agent 사이에 패킷을 전송할 수 있는 환경이 구축된다.

3.5.2. Sqoop[14]와 데이터베이스 연동

Sqoop는 데이터베이스의 Table을 분산파일시스템으로 전송해주는 도구이다. Sqoop과 데이터베이스를 연결하여 데이터를 받기 위해서는 먼저 하둡의 jar 파일들과 연동을 해야 한다. 연동 후 해당 RDBMS에 사용 가능한 Connector를 다운로드 받아 Sqoop/lib 폴더 하위에 이동시킨다. 그리고 명령어 창에 Sqoop 실행 명령어와 연결할 IP 및 포트 데이터베이스 종류 등을 입력하여 import를 하면 HDFS로 개별 테이블을 가져온다. 테이블의 각 행은 HDFS의 레코드로 처리되며, 모든 레코드는 텍스트 파일에서 텍스트 데이터 또는 Avro와 Sequence 파일 및 바이너리 데이터로 저장된다.

3.5.3. RDBMS와 Hive[15] 연동

Sqoop을 통해 HDFS에 RDBMS의 데이터를 복사하였지만, Hive의 쿼리문을 사용하여 데이터를 분석하기는 불가능하다. HDFS 클러스터와 관련된 Hive 메타스토어가 있는 경우 Sqoop은 Hive Create 테이블에서 데이터 레이아웃을 정의하는 명령문을 생성 및 실행하여 데이터를 Hive로 가져올 수 있다. Hive에서 External Table을 생성하여 Location을 HDFS 위치로 지정하고 구분자를 두어 테이블에 맞게 저장한다.

3.5.4. Pcap 파일 Hive 연동

본 논문에서 제안하는 시스템은 보안 측면에서 내부자의 악의적인 행위로 인한 유출탐지를 관리자가 알 수 있도록 Pcap 데이터 및 유출차단 로그를 RHive를 사용하여 분석 및 통계 데이터를 산출한다. HDFS에 저장된 비정형 및 정형 데이터를 Hive의 쿼리를 이용하여 MapReduce를 사용하면 분석이 용이하고 HDFS 기반으로 Hive에서 테이블을 생성하여 HDFS 내용 검색이 가능하다. Hive에서 테이블을 생성한 후부터는 HDFS를

통해 Pcap 파일의 패킷 내용을 튜플 형식으로 조회를 할 수 있으며, Where절과 Group-by를 사용하면 MapReduce 작업을 거치면서 대량의 데이터를 빠른 속도로 처리하는 것이 가능하다. Pcap 파일 및 로그를 분석할 수 있는 기반을 마련한 후 R과 Hive를 연동해주는 RHive를 이용하기 위해서는 R과 Rserve를 설치하여 R과 Rserve를 실행한다. 다른 원격 PC에서 R-Studio 웹을 통해 RHive 라이브러리를 사용하여 분산처리환경의 데이터에 접근할 수 있다. RHive를 사용하여 로그 데이터 및 Pcap 데이터의 통계를 분석하여 관리자에게 시각화된 결과값을 제공할 수 있다.

Ⅳ. 실험 및 성능평가

본 연구에서 구현한 내부 정보유출방지 시스템의 성능을 검증하기 위해 실험 환경을 구축하여 성능을 분석하였다. 본 논문에서 제안하는 하둡 기반의 사용자 행위 분석을 통한 내부유출 방지 시스템에서 대량의 패킷 데이터 및 이동식 저장 매체와 네트워크의 차단 데이터를 수집하여 이를 하둡 시스템의 HDFS에 저장하여 Hive와의 연동을 통해 데이터를 분석하여 관리자에게 모니터링 서비스를 제공하는 과정을 실험하였다. 그리고 대용량의 데이터 처리 성능을 비교하기 위하여 MySQL과 Hive와의 비교 테스트 환경을 구성하고 성능 분석을 진행하였다.

4.1. 기밀파일 삽입 테스트

기밀 파일과 일반 파일을 구분하기 위해서는 기밀 파일 내에 시그니처 삽입이 필수적이다. 본 논문에서는 테스트를 한글파일, MS Office 파일 등 문서 파일에 대해 시그니처를 삽입하였다. MS Office 파일은 XML 형식의 압축 파일로 구성되어 있기 때문에 파일 필드에 시그니처를 삽입할 경우 파일이 손상된다. 따라서 .Net Framework에서 지원하는 ZipFile 클래스를 이용하여 루트 영역에 시그니처 이름을 가진 XML 파일을 추가하고 나머지 일반 파일들은 파일 필드 가장 마지막에 시그니처를 삽입한다. 그림 9 에서 음영 처리한 부분은 삽입한 시그니처를 나타내고 있다.

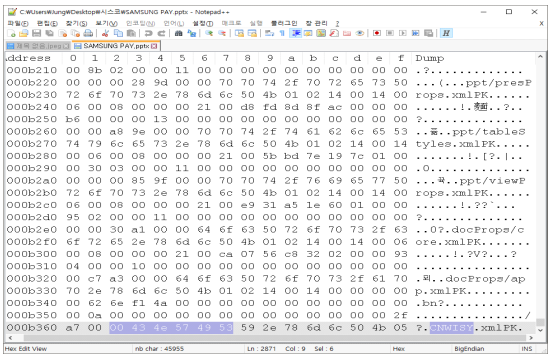

Fig. 9 Signature inserted file

4.2. 이동식 저장 매체를 통한 기밀 파일 유출 테스트

기밀파일 유출 차단은 사용자가 실수로 할 수도 있지만 고의로 유출할 수 있다. 이동식 저장매체의 경우 사용자 PC Agent로부터 유출을 차단하지 못하면 막을 수가 없다. 본 시스템은 모든 작업이 백 그라운드 환경에서 실행되지만 백 그라운드 환경에서는 직접적으로 탐지 및 기밀 파일 차단 여부를 확인할 수 없기 때문에 임의적으로 포어그라운드(Foreground) 환경에서 실행한 결과를 제시한다. 그림 10은 이동 매체로 기밀 파일과 일반 파일을 옮겼을 경우 포어그라운드 환경에서 테스트한 결과를 나타낸 것이다.

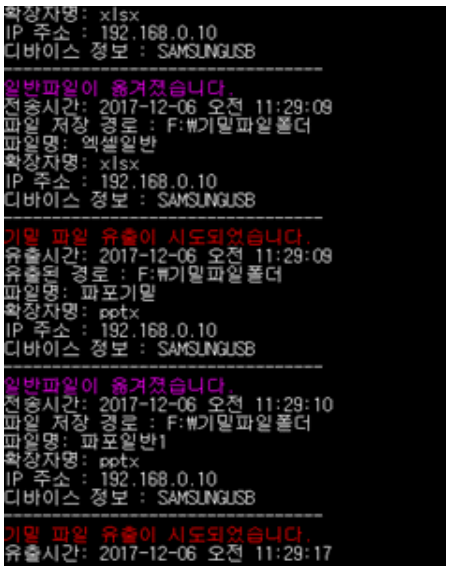

Fig. 10 Detection of confidential and ordinary file transfer to portable data medium

4.3. 네트워크를 통한 기밀파일 유출 테스트

네트워크를 통한 기밀파일 유출 차단은 DHCP 서버와 Snort Inline을 이용하여 유출을 차단한다. Snort Inline을 이용하여 모든 클라이언트로부터의 패킷을 수집하고 문자열 검색을 통해 관리자가 삽입한 시그니처를 검출하고 시그니처가 발견된 패킷을 삭제하여 메일이나 블로그에 올리지 못하게 차단한다.

4.4. 로그 분석시스템

로그 분석시스템에서는 RHive 명령어 Rhive.query()를 이용해 HDFS 가상 테이블을 생성한 Hive에서 대량의 데이터를 분석할 수 있게 하였으며, 분석시스템은 JAVA 언어로 구현되어 있다. 데이터베이스의 로그와 Pcap 파일을 쿼리를 사용하여 사용자별 기밀파일 차단 내역, 시간별 기밀파일 차단 내역, 시간별 네트워크 트래픽 양, IP별 네트워크 트래픽 양 등의 조회가 가능하도록 구현하였다.

4.4.1. 이동매체를 통한 사용자별 기밀파일 차단내역

사용자별 기밀파일 차단 내역은 누가 무슨 파일을 외부로 유출 시도를 하였는지 판단하기 위한 매우 중요한 로그이다. Rhive 쿼리를 사용하여 유출을 시도한 사용자 리스트를 확장자 별로 산정하고 시각화하여 관리자에게 제공한다. 사용자가 일과 시간이 아닌 시간에 유출 시도를 하였을 경우에는 악의적인 행동이라고 의심할 수 있다. 그림 11은 사용자별 기밀 파일 유출을 확장자별로 시각화한 것이다.

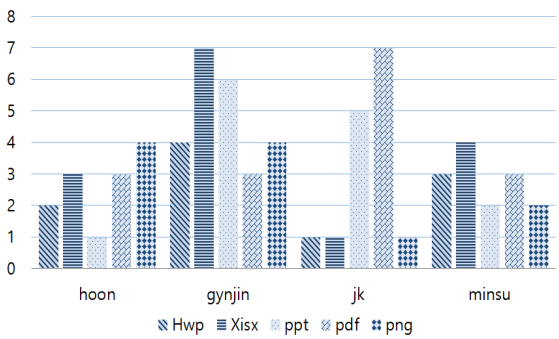

Fig. 11 Visualization of leakage of confidential file by both user name and file extension

4.4.2. 네트워크 트래픽 분석

기관 내부의 네트워크에서 이동하는 모든 트래픽은 사용자가 인지하지 못하는 상황에서 NAS 서버에서 패킷 형태로 쌓이게 된다. 사용자 개인 PC에서 시간별 IP별 트래픽 발생을 관리자가 파악할 수 있도록 하여 이상징후 발생 시 악의적인 행위를 추적할 수 있도록 한다. 그림 12는 시간별 특정 사용자 단말에서 발생된 네트워크 트래픽 양을 나타낸 것이다.

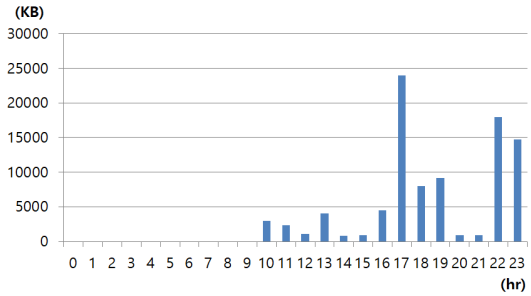

Fig. 12 Amount of network traffic by time

네트워크에 연결되어 있는 사용자 단말의 IP 주소별 트래픽 발생 현황을 파악하는 것은 정확하지 않은 경우가 있다. 기밀파일 차단 내역은 사용자 PC의 이름을 가져오지 못하기도 하고 IP 주소 또한 유동적으로 변하기 때문에 변하지 않는 MAC 주소별 차단 내역을 정리하여 관리자가 어느 사용자 단말에서 기밀 파일 유출을 시도하였는지 파악할 수 있도록 구현하였다.

4.5. Hive와 MySQL의 성능 비교

제안하는 하둡 기반의 내부 정보유출 방지시스템에서의 데이터 처리 성능 테스트를 위해 Hadoop 기반의 Hive와 RDBMS기반의 MySQL의 성능을 비교하였다. 먼저 기존 방식인 RDBMS에 저장한 네트워크 패킷을 얼마나 빠르게 처리하는지 알아보기 위해 사용자 PC에서 생성되는 패킷 데이터를 NAS 서버의 MySQL에 저장하였다. 다음으로 패킷을 대량으로 발생시키고 Sqoop을 이용하여 Hive로 구성된 노드에 패킷 데이터를 저장하여 데이터 로드 시간과 쿼리 실행 시간을 각각 비교하였다. 성능 테스트는 60MB, 100MB, 200MB 등 세 개의 데이터 셋에 대한 데이터 로드 시간과 10GB, 20GB, 30GB 등 세 개의 데이터 셋에 대한 쿼리 실행 시간을 측정하였다. 테스트 결과 MySQL과 Hive의 데이터 로드 시간은 데이터가 상대적으로 작은 60M일 때는 MySQL의

로드시간이 짧았지만, 100MB 이상 대용량 데이터의 경우에는 Hive의 데이터 로드 시간이 훨씬 짧아짐을 확인할 수 있었다. 한편, MySQL과 Hive에 대한 쿼리 실행 시간을 비교한 결과 데이터 처리 속도는 Hive가 MySQL에 비해 약 2배 이상 빠름을 보였고 데이터 크기가 커질수록 차이는 더 커짐을 확인할 수 있었다. 그림 13은 MySQL과 Hive의 데이터 처리 속도를 비교한 것이다.

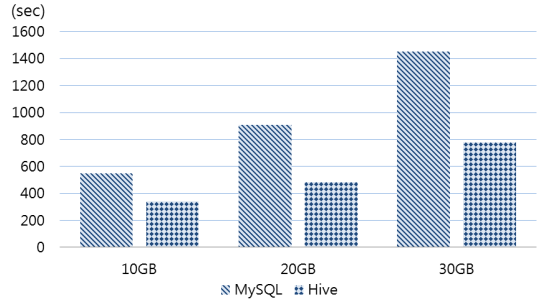

Fig. 13 Comparison of data processing time between MySQL and Hive

Ⅴ. 결론 및 향후 과제

전 세계적으로 기업들의 정보유출이 빈번히 발생되고 있으며, 정보유출로 인한 기업의 피해는 해마다 증가하고 있다. 본 논문에서는 중소기업이나 벤처기업 환경에서 기밀파일 유출을 차단 및 감시하기 위한 시스템을 구현하였다. 기밀파일을 정의하고 시그니처 삽입 시스템을 구현 및 실험하였고, 이동식 저장매체나 네트워크를 통한 기밀파일 유출을 차단하는 시스템을 구현 및 실험하였다. 또한, 네트워크나 저장매체의 로그 및 네트워크 패킷을 하둡 시스템에 수집 및 저장하고 로그분석시스템을 구축하여 관리자가 쉽게 모니터링 할 수 있도록 하는 시각화 시스템을 구현하였다. 시각화를 기반으로 관리자는 기밀파일 유출시도를 실시간으로 인지할 수 있고 기밀파일 유출에 대해 신속히 대응할 수 있다.

향후 연구로는 이동식 저장매체나 네트워크 상에서의 유출뿐만 아닌 인쇄물, 캡처링 등 다양한 정보 유출 방법에 대한 차단 방법에 대한 연구가 필요하다.

ACKNOWLEDGEMENT

This work was supported in part by Daejeon University fund (2018) and MSIT(Ministry of Science and ICT), Korea, under the SW master's course of a hiring contract program(2015-9-00999) supervised by the IITP(Institute for Information & communications Technology Promotion).

REFERENCES

[1] J. S. Lee and K. H. Lee, “A study on security container to prevent data leaks,” *Journal of the Korea Institute of Information Security & Cryptology*, vol. 24, no. 6, pp. 1225-1241, June 2014.

[2] M. B. Hyun and S. J. Lee, “The proactive threat protection method from predicting resignation throughout DRM log analysis and monitor,” *Journal of The Korea Institute of Information Security & Cryptology*, vol. 26, no. 2, pp. 369-375, Feb. 2016.

[3] J. H. Choi and S. Y. Rhew, “Monitoring system of file outflow through storage devices and printers,” *Journal of the Korea Institute of Information Security & Cryptology*, vol. 15, no. 4, pp. 51-60, April 2005.

[4] T. K. Ju and W. Shin, “A new filtering system against the disclosure of sensitive internal information,” *Journal of the Korea Institute of Information and Communication Engineering*, vol. 19, no. 5, pp. 1137-1143, May 2015.

[5] J. U. Choi and Y. J. Lee, “E-DRM-based privacy protection technology for overcoming technical limitations of DLP-based solutions,” *Journal of the Korea Institute of Information Security & Cryptology*, vol. 22, no.5, pp. 1103-1113, Oct. 2012.

[6] S. J. Hee and H. B. Park, “ Multiple barcode watermarking technique for improve robustness and imperceptibility,” *Journal of the Korea Institute of Information and Communication Engineering*, vol. 20. no. 9. pp. 1723-1729, Sept. 2016.

[7] G. J. Shin, G. H. Jung, D. M. Yang, and B. H. Lee, “A USB DLP Scheme for Preventing Loss of Internal Confidential Files,” *Journal of the Korea Institute of Information and Communication Engineering*, vol. 21. no. 12. pp. 2333-2340, Dec. 2017.

[8] Snort Users Manual [Internet]. Available: https://snort.org/documents.
[9] S. N. Park, A. Y. Kim, and H. K. Jung, “A study on signature-based wireless intrusion detection systems,” *Journal of the Korea Institute of Information and Communication Engineering*, vol. 19. no. 5, pp. 1122-1127, May 2014.
[10] Apache Hadoop [Internet]. Available: https://hadoop.apache.org/.
[11] R. D. Caytiles, “Big Data is not just Hadoop,” *Asia-pacific Journal of Multimedia Services Convergent with Art, Humanities*, and Sociology, vol. 1, no. 1, pp. 11-16, June 2012.
[12] Apache Flume [Internet]. Available: https://flume.apache.org/.
[13] The industry-standard windows packet capture library. Available: https://www.winpcap.org/.
[14] Apache Sqoop [Internet]. Available: http://sqoop.apache.org/.
[15] Apache Hive [Internet]. Available: https://hive.apache.org/.

유혜림(Hye-Rim Yoo)

2005년 대전대학교 정보통신공학과(학사)
2007년 대전대학교 대학원 정보통신공학과(석사)
2011년 대전대학교 대학원 정보통신공학과(박사수료)
현재 대전대학교 전자 · 정보통신공학과 강의전담교수
※ 관심분야 : 딥러닝, 머신러닝, 영상처리, 네트워크보안 등

신규진(Gyu-Jin Shin)

2016년 대전대학교 정보통신공학과(학사)
2018년 대전대학교 대학원 정보통신공학과(석사)
2018 현재 ㈜아이렉스넷 연구원
※ 관심분야 : 사물인터넷, 빅데이터, 네트워크보안 등

양동민(Dong-Min Yang)

2000년 POSTECH 컴퓨터공학과(학사)
2003년 POSTECH 컴퓨터공학과(석사)
2009년 9월 ~2011년 9월 삼성전자 Senior Engineer
2011년 POSTECH 컴퓨터공학과(박사)
2011년 9월 ~2017년 9월 대전대학교 정보통신공학과 조교수
2017년 9월 ~ 현재 전북대학교 대학원 기록물 관리학과 조교수
※ 관심분야 : Archives & Records Information Security, IoT, Manet

이봉환(Bong-Hwan Lee)

1985년 서강대학교 전자공학과졸업(학사)
1987년 연세대학교 대학원 전자공학과 졸업(석사)
1993년 Texas A&M 대학교 대학원 전기 및 컴퓨터공학과 졸업(박사)
1995년 3월~현재 대전대학교 전자 · 정보통신공학과 교수
※ 관심분야 : 클라우드 컴퓨팅, 사물인터넷, 네트워크보안 등