디지털산업정보학회 논문지 제13권 제4호-2017년 12월
http://dx.doi.org/10.17662/ksdim.2017.13.4.125

네트워크 트래픽 수집 및 복원을 통한 내부자 행위 분석 프레임워크 연구*

고 장 혁** · 이 동 호***

A Study on the Insider Behavior Analysis Framework for Detecting Information Leakage Using Network Traffic Collection and Restoration

Kauh Janghyuk · Lee Dongho

〈Abstract〉

In this paper, we developed a framework to detect and predict insider information leakage by collecting and restoring network traffic. For automated behavior analysis, many meta information and behavior information obtained using network traffic collection are used as machine learning features. By these features, we created and learned behavior model, network model and protocol-specific models. In addition, the ensemble model was developed by digitizing and summing the results of various models. We developed a function to present information leakage candidates and view meta information and behavior information from various perspectives using the visual analysis. This supports to rule-based threat detection and machine learning based threat detection.

In the future, we plan to make an ensemble model that applies a regression model to the results of the models, and plan to develop a model with deep learning technology.

Key Words : Cyber, Insider Threat, Behavior Analysis, Machine Learning

Ⅰ. 서론

2000년대 초반부터 미국 국방성에서는 내부자 사이버 위협의 심각성을 인식하여 연구를 시작하였고 2001년부터는 CERT가 미 비밀사무국과 공동연구를 시작했다. 이 외에도 DARPA, ARDA, I3P 등에서 내부자 사이버 위협에 관한 연구개발을 주관하거나 지원하고 있다[1-3].

최근에는 GPU의 발전으로 머신러닝의 적용성이 좋아짐에 따라 MIT대학에서 머신러닝 기술을 이용한 AI2(Artificial Intelligent Analyst Intuition)프로젝트를 통해 지도학습(supervised learning)와 비지도 학습(unsupervised learning)을 조합 형태의 분석 기술을 발표하였고, 자체 발표에 따르면 기존의 내부 위협 85% 수준의 검출 성능을 보인다고 발표하였다[4-6]. 또한 상용 제품들은 데이터양을 고려하여 주로 정보보호 제품의 로그들을 대상으로 분석하

* 이 논문은 2017년도 광운대학교 교내 학술연구비 지원에 의해 연구되었음
** 국방과학연구소 선임 연구원
*** 광운대학교 소프트웨어학부 교수

는 제품들이 많이 나오고 있는데 대표적인 제품으로는 Splunk의 UBA (User Behavior Analytics)와 록히드 마틴의 Wisdom ITI (Insider Threat Identification) 등이 있다.

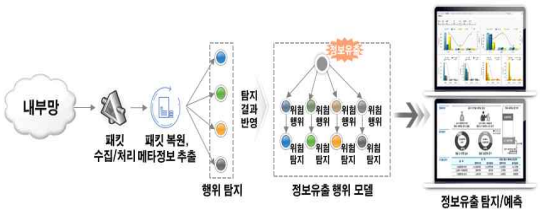
본 연구에서는 내부자에 대한 관찰 및 감시를 내부자가 인식하지 못하도록 호스트에 에이전트를 설치하는 방식이 아닌 네트워크 트래픽 분석을 통해 내부자의 정보 유출을 탐지 및 예측하는 분석 프레임워크를 제안한다. 또한 프레임워크에 의해 수집되는 다양한 정보들을 feature로 하여 머신 러닝 기술을 적용하여 내부자의 정보 유출 징후를 탐지 및 예측하는 기술에 대해 기술하려 한다.

2장에서는 먼저 네트워크 트래픽을 수집 및 복원하는 기술에 대해 간략하게 소개하고 3장에서는 수집된 네트워크 및 프로토콜 정보를 통해 행위 정보를 생성하고 정보유출 행위를 탐지하기 위한 분석 기능을 설명하며 이러한 정보들을 이용한 머신러닝 정보유출 탐지 기능에 대해 설명하고, 4장에서는 시험을 위한 모의 환경과 시험 결과 등에 대해 설명하며 5장에서는 결론으로 내부자 행위 분석의 성능을 높이기 위해 나아가야 할 방향과 추가 연구 필요성에 대해 기술한다.

Ⅱ. 네트워크 트래픽 수집 및 복원

우리는 내부자의 정보유출을 탐지하기 위해 네트워크에 흘러 다니는 패킷을 수집하여 프로토콜로 복원하는 기능과 이렇게 수집된 네트워크 정보들을 사용자의 행위와 연관시키기 위한 연구를 진행하였다.

본 연구의 프레임워크 구성도는 <그림 1>과 같다.



<그림 1> 프레임워크 구성도

분석 프레임워크의 기능 구성은 다음과 같다.

○ 수집기(Collect):
- 데이터 수집: 네트워크상에 흐르는 패킷을 수집
- 데이터 복원: 수집된 패킷을 분석가능한 형태로 변환
- 데이터 저장: 수집된 패킷 및 정보를 하둡으로 분산 저장/관리

○ 탐지기(Detect)
- 표시기(Indicator): 복원된 정보에서 사용자 행위를 예측
- 이상 탐지기: 다양한 측면에서 과도한 행위를 탐지
- 시나리오 탐지기: 알려진 정보유출과 관련된 일련의 행위를 탐지

○ 분석기(Analysis)
- 수치부: 행위정보들을 수치화
- 융합부: 머신 러닝을 통한 정보융합
- 탐지 및 예측부: 단계 및 임계값을 통한 정보유출 탐지 및 예측

○ 시각적 분석 및 가시화
- 분석도구: 정보유출
- 통계: 수집된 패킷 및 행위에 대한 통계

현재는 이러한 행위기반 분석 프레임워크를 대표적인 내부자 위협인 정보유출의 경우에 적용하여 <그림 1>과 같이 설계/구현하고 있다. 탐지부와 분

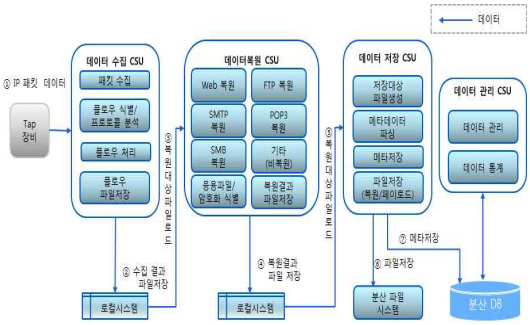
석부를 일부 수정하면 시스템 파괴나 사용자의 실수 등과 같은 내부자의 여러 위협에도 적용 가능할 것으로 판단된다[7].

2.1 네트워크 및 프로토콜 Feature 수집

본 프레임워크에서는 네트워크 트래픽을 수집하고 프로토콜별로 복원하여 내부자의 행위 관련 정보를 획득한다. 물론 내부자 행위 외에도 시스템에서 발생하는 모든 트래픽을 수집하고 있으며 <표 11>에서 보이는 바와 같이 기타 트래픽의 비율이 약 70%이상을 나타나고 있다[8]. 이러한 트래픽에는 DNS 트래픽 및 통신 제어에 대한 트래픽이 해당되겠다. 본 프레임워크는 다음과 같은 프로토콜을 복원하며 복원하지 못하는 네트워크 트래픽도 헤더 및 페이로드를 수집 저장하여 메타 정보로 관리한다.

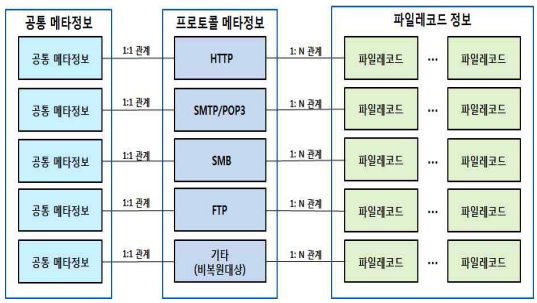
- HTTP (80) : HTTP 프로토콜을 사용하는 데이터 복원, HTTP 헤더 분석 / URL 추출 기능 / 송수신 데이터 복원
- FTP (20/21) : FTP 프로토콜을 사용하는 데이터 복원, 파일전송 / FTP 명령어 / 메시지 복원
- SMB (445) : SMB 프로토콜을 사용하는 데이터 복원, 파일 전송 데이터 복원 / 프린터 전송 데이터 복원
- SMTP (25) : SMTP 프로토콜을 사용하는 데이터 복원 메일의 본문과 송/수신자 및 첨부파일 복원
- POP3 (110) : POP3 프로토콜을 사용하는 데이터 복원, 메일의 본문과 송/수신자 및 첨부파일 복원
- 기타 : 복원대상 프로토콜이 아닌 나머지 데이터 복원, 패킷 데이터를 플로우별로 구분하여 파일로 저장

네트워크 및 프로토콜별 정보는 <그림 2>와 같이 TAP장비를 통해 작은 패킷들을 수집하고 Flow 별로 조립하여 프로토콜별로 복원한다.



<그림 2> 네트워크 트래픽 수집 및 복원

이렇게 수집된 정보는 <그림 3>과 같이 메타정보와 페이로드 파일로 저장된다.



<그림 3> 네트워크 및 프로토콜 정보 관리

메타 정보는 공통, 프로토콜 메타 정보로 나뉜다. 공통 메타 정보에는 <표 1>과 같이 패킷 관리번호, 어플리케이션의 종류, IP버전, 출발지IP, 도착지IP, 전송프로토콜, 전송프로토콜, 출발지 포트, 도착지 포트, 수집 장비 ID, 패킷 수집 시간, 세션 ID, 데이터 전송상태, 파일순서번호, 첨부파일 수 등이 저장된다. 프로토콜 메타 정보는 프로토콜마다 다르며 주로 프로토콜의 상태, 요청/응답, 명령어 등의 정

보가 저장된다. 프레임워크는 이러한 정보에서 사용자의 활동(Activity)을 탐지한다.

이러한 메타 정보가 머신러닝 모델을 학습하기 위한 Feature로 활용된다.

메타항목명	의미
packetnum	패킷번호
commapplicationtype	어플리케이션 타입
commsourceip	출발지 IP
commdestinationip	도착지 IP
commprotocol	전송프로토콜
commsourceportnumber	출발지 포트
commdestinationportnumber	도착지 포트
commcollectstarttime	수집시간
packetregdt	등록시간
sessionid	세션ID

<표 1> 프로토콜 공통 메타

프로토콜에 대한 메타정보는 종류별로 각각 존재하며 <표 2>는 사용자가 가장 많이 사용하는 프로토콜인 HTTP 프로토콜의 메타정보이다.

메타항목명	의미
packetnum	패킷번호
sessionid	세션ID
webhost	host명
lastflag	세션 지속여부
sequencenumber	시퀀스번호
attachcnt	첨부파일 수
packetregdt	등록시간

<표 2> HTTP 메타

메일관련 프로토콜은 SMTP 프로토콜과 POP3 프로토콜 2개가 있으며 다음과 같은 항목을 공동메타로 관리한다.

메타항목명	의미
packetnum	패킷번호
sessionid	세션ID
mailtype	메일타입
mailsenderid	메일송신자ID
mailreceiverid	메일수신자ID
mailccid	메일참조자ID
mailbccid	메일BCC ID
mailsubject	메일제목
mailmessage	메일메시지
lastflag	세션 지속여부
sequencenumber	시퀀스번호
attachcnt	첨부파일 수
packetregdt	등록시간

<표 3> MAIL 메타

또한 각각의 프로토콜에는 파일들이 존재하며 이에 대한 정보는 <표 4>와 같다. 주로 HTTP와 SMB 프로토콜을 통해 첨부 파일로 수집되며 프로토콜의 페이로드도 파일 형태로 저장된다.

메타항목명	의미
filenum	파일번호
packetnum	패킷번호
filecategory	파일카테고리
sessionid	세션ID
url	URL(HTTP)
httptype	HTTP Type(HTTP)
smbtype	SMB Type(HTTP)
flowdirection	패킷방향
filename	파일명
filetype	파일확장자
filesize	파일사이즈
encryptionflag	암호화여부
extensionflag	확장자변경여부

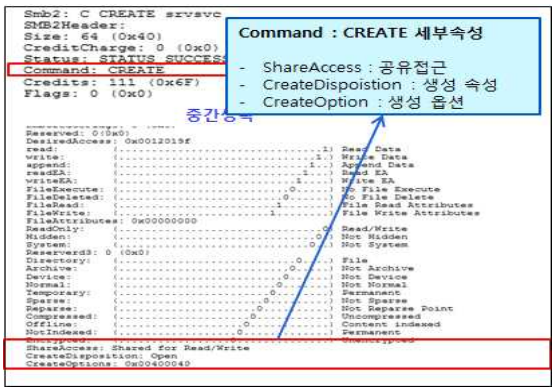
<표 4> 파일 메타

HTTP, FTP, SMTP, POP3 프로토콜 복원기술은 기존 연구[8]를 재활용하여 개발하였으며 본 연구에서는 사용자 LAN환경에서 많이 사용되는 SMB 프로토콜과 네트워크 프린팅 프로토콜 복원 기술에 중점을 두었다.

메타항목명	의미
packetnum	패킷번호
sessionid	세션ID
smbversion	SMB버전
lastflag	세션 지속여부
sequencenumber	시퀀스번호
attachcnt	첨부파일 수
packetregdt	등록시간

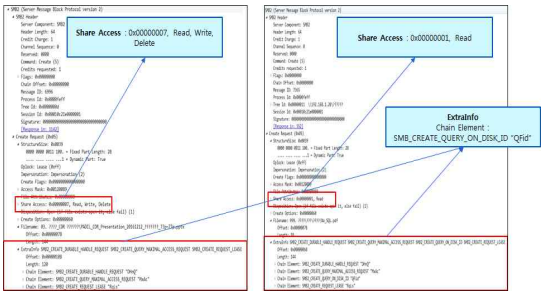
<표 5> SMB 메타

SMB 프로토콜은 초기에는 간단한 프로토콜로 시작하였으나 Microsoft사의 여러 프로토콜을 통합하여 방대한 프로토콜이 되었다. 그래서 본 프레임워크에서는 <표 5>, <표 6>와 같이 SMB 프로토콜에 대한 메타정보와 SMB COMMAND에 대한 메타정보로 분리해서 저장하고 있다. SMB 프로토콜은 19개의 명령어와 서브 명령어 및 파라미터를 통해 다양한 기능을 제공한다.



<그림 4> SMB COMMAND 및 파라미터 복원

<그림 4>는 SMB 프로토콜을 분석한 내용이다. SMB 프로토콜은 명령어뿐만 아니라 다양한 파라미터들이 존재한다. 내부자가 윈도우 상에서 탐색기를 통해 파일을 복사하거나 파일을 여는 행위는 <그림 5>와 같이 SMB 프로토콜을 통해 구현된다. 이러한 행위는 SMB 명령어 수준에서는 CREATE 명령으로 동일하다. 본 연구에서는 이러한 프로토콜 명령과 파라미터를 복원함으로서 두 행위를 구별할 수 있다.



<그림 5> SMB상에서 파일 오픈과 파일 복사 차이

메타항목명	의미
SMB_SEQ	명령어 관리번호
PACKETNUM	TB_PACKET 관리번호
FILENUM	TB_FILE 관리번호
REGDT	수정일시
COMMSOURCEIP	출발지 IP
COMMDESTINATIONIP	도착지 IP
COMMCOLLECTSTARTTIME	플로우 내 패킷 수집시작 시간
H_COMMAND	SMB COMMAND HEADER H_COMMAND
COMMAND	SMBCOMMAND 상세 COMMAND
~	
P_STRUCTURE_SIZE	SMBCOMMAND 상세 COMMAND P_STRUCTURE_SIZE
P_SHARE_TYPE	SMBCOMMAND 상세 COMMAND P_SHARE_TYPE
P_SESSION_FLAGS	SMBCOMMAND 상세 COMMAND P_SESSION_FLAGS
P_DIALECT_CNT	SMBCOMMAND 상세 COMMAND P_DIALECT_CNT
P_SECURITY_MODE	SMBCOMMAND 상세 COMMAND P_SECURITY_MODE
~	

<표 6> SMB COMMAND 메타

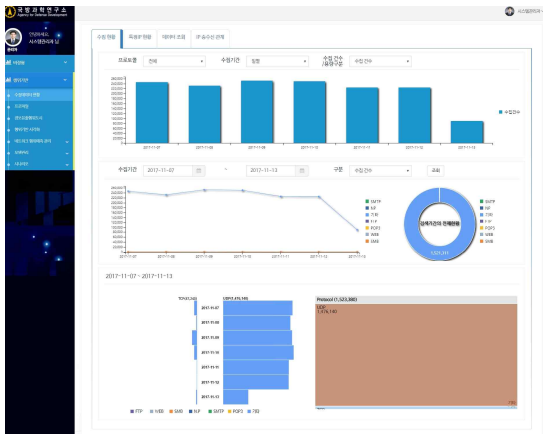
<그림 6> 네트워크 프린터(PJL) 프로토콜 복원

다음은 네트워크 프린팅의 경우로 본 프레임워크는 SMB 프로토콜을 이용한 프린팅과 HP에서 개발되어 사실상의 표준이 된 PJL(Printer Job Language)을 통해 제어하는 TCP/IP 프린터 전용 드라이버를 이용한 프린팅을 수집 및 복원한다. SMB 프로토콜을 이용한 네트워크 프린팅은 앞서 설명한 SMB 프로토콜 메타정보로 수집 및 관리된다. 네트워크 프린팅의 경우 대부분은 <그림 6>과 같이 TCP/IP 전용 프로토콜에 의해 송수신된다. 네트워크 프린팅에서 관리하는 메타정보는 <표 10>과 같다.

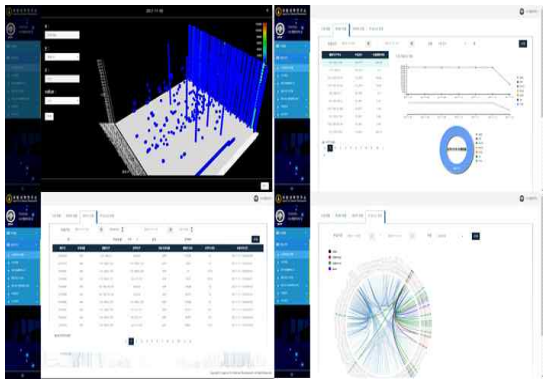
메타항목명	의미
packetnum	패킷번호
sessionid	세션ID
username	사용자명
appname	어플리케이션명
documentname	문서명
copies	부수
printdatatype	프린트데이터 타입
printresult	프린트 결과
pjlmessage	pjl 메시지
lastflag	세션 지속여부
sequencenumber	시퀀스번호
attachcnt	첨부파일 수
packetregdt	등록시간

<표 7> 네트워크 프린팅 메타

앞서 설명한 이러한 다양한 네트워크 및 프로토콜 메타정보 및 복원파일은 <그림 7>, <그림 8>과 같은 다양한 통계 및 가시화 기능을 통해 분석할 수 있다.

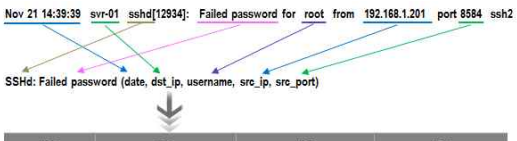


<그림 7> 수집 통계 기능



<그림 8> 가시화 분석 기능

<그림 9>와 같이 행위분석 프레임워크에는 기존의 시스템 및 정보보호제품의 이벤트 로그 정보를 행위정보로 변환하는 기능을 가지고 있다. 이를 통해 로그 정보도 포함하여 내부자의 이상행위를 분석 및 탐지할 수 있다.



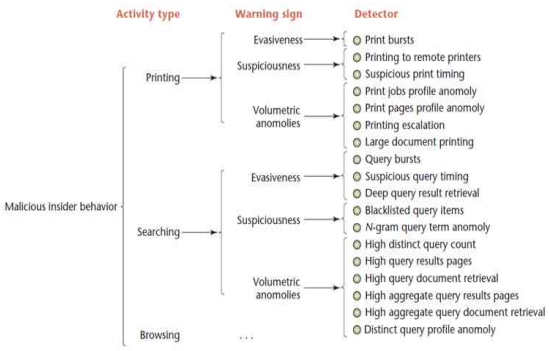
분류	항목	내용	비고
행위 정보	행위 내용	SSH: 잘못된 비밀번호	
	행위탐지기 종류	로그 분석	
	행위 분류	Access	
	행위 단계	3단계 - 접근 및 확대	
	위험 등급	정상	
수집 정보	수집시간	20161121143939	[YYMMDDHHmmss]
	출발지 IP	192.168.1.201	
	도착지 IP	svr-01	Hostname으로 표시
	출발지 Port	8584	
	도착지 Port	21	

<그림 9> 로그의 행위정보 변환

Ⅲ. 내부자 행위 분석기술

3.1내부자 행위 정보(Behavioral Signatures)

내부자의 위협분석은 작은 상태 및 이벤트 정보가 모여 지표(Indicator)가 되고 이러한 지표가 모여 IOC (Indicator of Compromise)가 되며 이러한 IOC들이 모여 하나의 행위 IOC를 나타내고, 이러한 행위 IOC들이 모여 정보유출, 시스템 파괴와 같은 내부자의 위협 사건을 이루게 된다. 행위(Behavior)는 여러 개의 활동(Activity)들로 이루어지며 이러한 내부자의 악의적인 활동은 <그림 10>과 같다[1, 2].



<그림 10> 사용자 이상 행위 분류[1,2]

본 연구에서는 사용자의 이상행위를 탐지하기 위해 볼륨의 이상적 증가와 같이 네트워크 트래픽 자체로 사용자의 이상행위를 예측할 수 있는 경우도 있으나 대부분의 경우에는 평소 사용자의 정상 행위에 대한 이상 행위임으로 사용자의 프로파일 정보가 중요하다. 행위에 대한 정보는 사용자 프로파일 정보와 비교하여 충돌되거나 지나치게 초과하는 경우를 탐지한다. 프로파일 정보는 인사(Human Resource)정보, 네트워크(Network) 정보, 디바이스(Device) 정보 그리고 개인 식별 정보 (PII: Personal Identity Information) 및 운용(Operation) 정보 등으로 구성된다[7, 9].

사용자의 행위 정보는 다음과 같이 5가지 범주로 분류할 수 있다[6, 9].

- Count, averages, standard, deviations
  예) 로그인 수, 평균 사용시간 등
- Indicators (or boolean variables)
  예) 로그인 실패여부, 근무시간 및 위치 이상 여부 등
- Relational features : 2개의 entity들로 계산
  예) 로그인 실패 회수 등
- Temporal behaviors : 경과된 시간
  예) 로그인 후 체크 아웃 까지의 사용시간 등
- Unique values
  예) 접속한 위치 정보 등

본 프레임워크에서는 정보 유출 행위를 사이버 킬 체인 모델을 참고하여 5단계로 분류하였고 이를 통해 행위 정보를 추가하였다.

내부자의 행위를 분석하기 위해서는 내부자의 행위 정보를 추적(Tracking)하거나 종합(Aggregation)할 수 있어야 한다[6].

행위 추적(Activity Tracking)은 사용자에 대한 행위정보 스트림을 구성해야 한다. 이를 위해서는 2

가지 방법이 존재하는데 짧은 주기의 시간(예, 30분, 1시간, 12시간, 24시간)으로 구성하는 방법과 이벤트 정보 검색(예, 동영상 요청 패킷 ~ 완료)으로 구성하는 방법이 있다.

행위 종합(Activity Aggregation)도 2가지 방법이 있는데 먼저 주어진 기간 내의 사용자의 모든 활동 기록을 탐색할 수 있어야 하며 사용자의 모든 활동 기록을 시간의 순서로 최종적으로 24시간에 걸쳐 사용자가 한 행동을 확인할 수 있어야 한다.

메타항목명	의미
COLLECT_DTIME	수집 시간 (시간 기준)
APPLICATION_TYPE	어플리케이션 타입
IP_ADDRESS	출발지 IP
PROTOCOL	전송프로토콜
COLLECT_CNT	카운트 수
COLLECT_TIME	수집 시간 (시간 기준)
COLLECT_DATE	수집 시간 (일 기준)
COLLECT_WEEK	수집 시간 (주 기준)
COLLECT_MONTH	수집 시간 (월 기준)
REGISTER_DTIME	등록 시간
SESSION_ID	세션 ID

<표 8> IP 기준 통계성 메타

본 프레임워크에서는 행위정보를 네트워크 트래픽을 수집 및 복원하여 기본적으로 Flow 단위로 저장하기 때문에 검색 이벤트로 구성할 수 있으며 <표 8>과 같이 IP 기준 통계성 메타정보를 저장 관리함으로 시간 주기로 검색 및 확인할 수 있다. 분석 프레임워크는 매일 많은 양의 행위정보를 분석할 수 있어야 하며 실시간으로 많은 정보에서 원하는 정보를 검색할 수 있어야 한다. 내부자의 모든 활동 기록을 분단위 기준으로 행위를 기록하여 관리한다면 행위정보는 1분마다 1개의 record를 생성하게 된다. 그럴 경우

24시간 X 60 분 = 1440 records

즉, 사용자의 특정 행위를 검색하기 위해 1440개의 record에서 검색을 해야 한다. 본 프레임워크에서는 성능 향상을 위해 행위 정보를 다음과 같이 저장하고 있다.

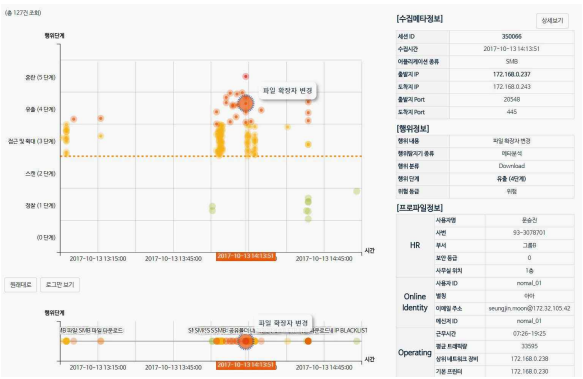
- 1분마다 저장 (정각 실행)
- 1시간마다 저장 (정각 실행)
- 1일마다 저장 (자정 실행)
- 1주일마다 저장(일요일 자정 실행)

이를 통해 24시간 record 검색시

23(시간) record + 60(분) record = 83 records

즉 검색대상의 record 수를 1440 record에서 83 record로 줄여 성능을 향상시켰다.

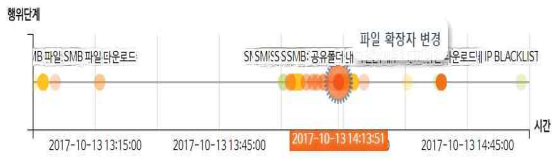
행위 추적 및 종합 기능은 앞서 설명한 수집정보 관리기능을 통해서도 확인할 수 있지만 <그림 11>과 같이 수집메타정보, 행위 정보, 사용자 프로파일 정보를 같이 보며 분석할 수도 있다.



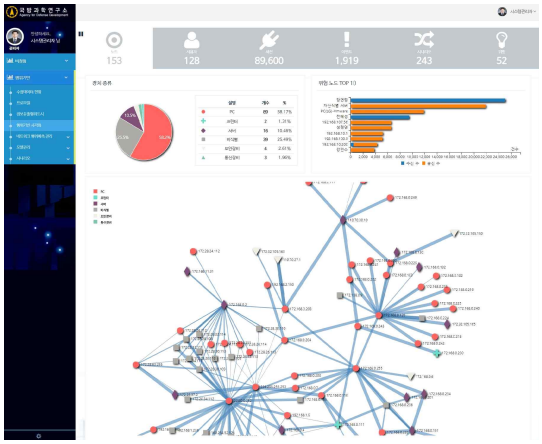
<그림 11> 내부자 행위 도시 기능

<그림 12>와 같이 시간의 흐름에 따른 시각적 분석 기능을 통해 확인할 수 있다.

분석을 위한 기능은 내부자 행위 도시기능 외에도 <그림 13>, <그림 14>와 같이 노드, 사용자, 세션, 이벤트, 시나리오, 위협 관점별로 분류되어 제공된다.



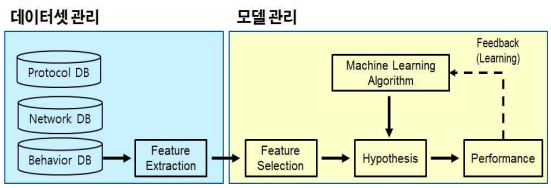
<그림 12> 시계열 분석 기능



<그림 13> 노드관점 분석 기능

<그림 14> 다양한 관점의 분석 기능

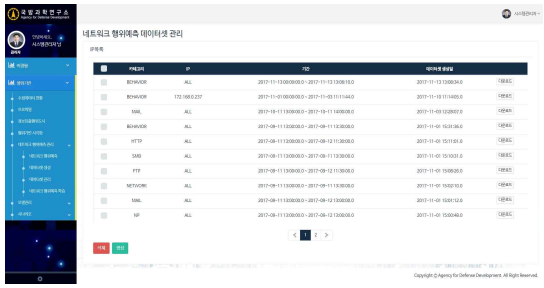
3.2 머신 러닝을 이용한 행위기반 분석



<그림 15> 머신러닝 프로세스

행위분석 프레임워크에서는 <그림 15>와 같이 머신러닝 프로세스를 지원하기 위해 데이터셋 관리 기능과 모델관리 기능 2가지로 구성이 된다.

데이터셋 관리기능은 네트워크 트래픽을 수집 및 복원해서 네트워크 및 프로토콜에 대한 다양한 정보를 메타정보로 관리하여 이러한 정보항목은 데이터 분석 및 머신 러닝 모델을 학습하기 위한 데이터 셋으로 생성해 주는 기능이다.



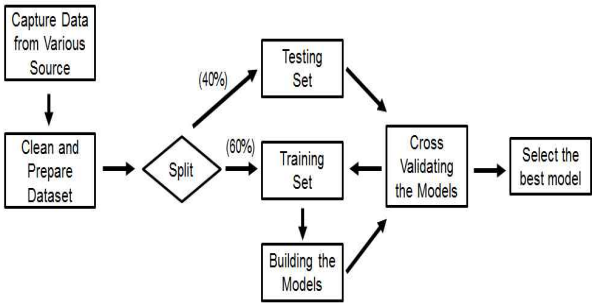
<그림 16> 데이터셋 관리기능

<그림 16>과 같이 행위, 네트워크, 그리고 프로토콜별(HTTP, FTP, SMB, MAIL, NP)로 7가지 종류의 데이터 셋을 생성할 수 있으며 각각에 대해서는 IP 범위(특정IP 또는 ALL)와 기간을 설정할 수 있게 되어 있다. 이를 통해 추후에도 머신러닝 모델을 지속적으로 학습시킬 수 있으며 다운로드를 통해 다른 시스템 및 분석방법을 이용할 수도 있다.

Feature Selection은 데이터셋으로부터 모델을 만들고 학습시킬 Feature를 결정하는 단계로 많은 시간과 경험이 필요한 분야이다. 본 연구에서는 네트워크 및 프로토콜 정보들에 대해 SFID(Size, Frequency, Interval and Delay) 측면의 feature들을 선택하여 각각의 모델을 만들고 학습시켰다. 기존에 많이 사용되는 네트워크 통계적 정보를 이용하는 모델을 포함해서 네트워크 특성 모델, 행위 통계 모델, 행위 특성 모델 그리고 프로토콜 별 모델을 만들어

학습시켰다.

성능검증 절차는 <그림 17>과 같이 Train set(60%) / Test set(40%) 으로 분리한 후 Train set으로 학습시키고 Test set으로 predict 진행 후 label과 일치여부를 시험하였다.



<그림 17> 머신러닝 알고리즘 성능 검증 절차

현재는 classifier는 <표 9>와 같이 기존에 성능비교 했던 연구에서 우수한 성능을 보였던 RandomForest 모델을 이용하여 개발되었다[8].

알고리즘	Accuracy	F1-Score
LinearSVC	94.54	0.182
DecisionTree	95.59	0.636
RandomForest	96.29	0.591
GaussianNB	81.12	0.171
SGDClassifier	94.56	0.225

<표 9> Z-Scoring 정규화한 모델 성능비교[8]

모델관리 기능은 <그림 18>과 같이 네트워크 특성모델, 네트워크 통계모델, 행위 특성모델, 행위 통계모델, 그리고 프로토콜별 모델인 HTTP, FTP, SMB, MAIL, NP 모델로 구성된다.

<그림 18> 모델 종류

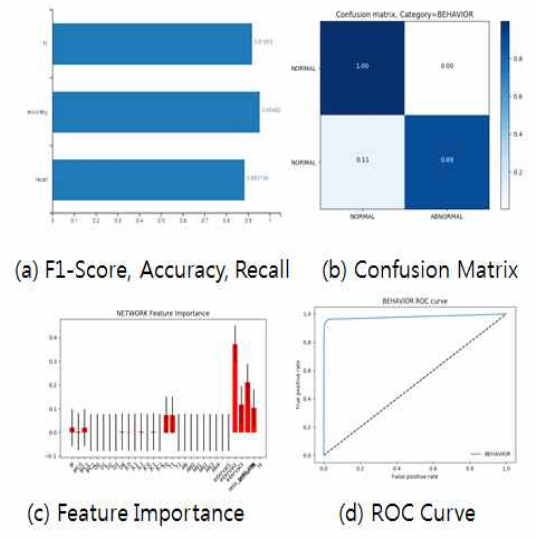
행위분석 프레임워크에서는 많은 내부자들 중에서 정보유출 후보군을 머신러닝 모델을 이용하여 선정하는 기능을 개발하였다. 여기에 사용된 모델은 앞서 설명한 다양한 모델들의 결과를 수치로 변환하여 합산하는 앙상블 모델로 <그림 19>와 같이 상위 100명을 점수순서로 목록화하여 이상행위 후보군을 도출하는 기능을 개발하였다. 이를 통해 대규모 네트워크에서 내부자의 위협을 보다 효율적으로 대응할 수 있다.

<그림 19> 이상행위 후보군 도출기능

목록에서 특정 IP를 선택함으로써 IP에 대해 각각의 머신러닝 모델들이 어떻게 판단하였는지를 <그림 20>과 같이 점수화된 수치를 통해 확인할 수 있다.

<그림 20> 네트워크 행위예측 결과

또한 모델정보를 선택하면 <그림 21>과 같이 머신러닝에서 일반적으로 제공하는 모델 성능 분석 지표인 F1-Score, Accuracy, Recall 등과 Confusion Matrix, Feature별 중요도, ROC 커브를 제공한다.



<그림 21> 모델 분석

Ⅳ. 시험

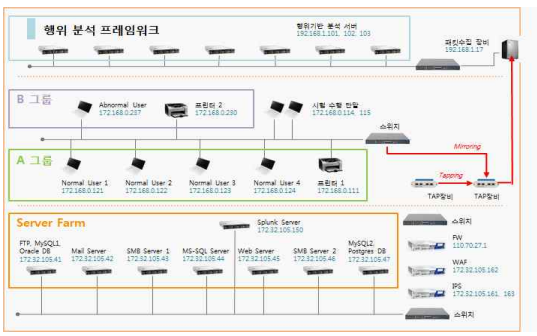
보안 분야에서 정보유출 데이터를 확보하는 일은 매우 힘든 일이다. 그 외에도 다음과 같은 이유로 힘들다.

- 레이블 데이터의 부족 : 정보유출과 관련된 데이터를 확보하기 힘들며 정상 데이터와 정보유출 데이터를 구분하기 힘들다.
- 공격유형 진화 : 정보유출이란 목적을 달성하기 위해 수행되는 다양한 공격은 매우 다양하고 빠르게 진화되고 있다.
- 제한된 시간과 예산 : 정보유출 사건은 평균적으로 2~3개월 정도의 시간을 거쳐 수행되며 그 동안의 네트워크 및 전사적인 데이터를 확보하기 위해서는 대용량의 데이터 수집 장비와 처리장비가 필요하다.

이러한 레이블 데이터 부족과 공격유형의 진화 문제를 해결하기 위해 본 연구에서는 전문가들을 통해 여러 가지 정보유출 시나리오를 작성했으며, 시나리오는 여러 단계로 구성되어 있어 각 단계를 변경함으로써 여러 시나리오로 확장할 수 있게 하였다.

제한된 시간과 예산 문제를 해결하기 위해서 본 연구에서는 정보유출 사건은 24시간동안에 발생하는 것으로 가정하고 시험을 수행하였으며 실제 적용할 때는 데이터 처리 및 저장 능력에 맞게 설정 변경하면 구현 가능하도록 개발하였다.

4.1 시험 환경



<그림 22> 시험 환경 구성도

시험 환경은 <그림 22>와 같이 내부자의 행위를 모의하는 네트워크가 있고, 여기서 유통되는 트래픽을 수집 및 복원하여 분석하는 행위 분석 프레임워크 네트워크로 구성된다. 시험을 위한 모의 환경은 사용자 영역은 A, B그룹과 서버 영역은 Server Farm 으로 구분하였다. 악의적인 행위는 사용자 영역에서 Server Farm으로 침투하는 종적인 접근도 있지만, 정보유출 및 악의적인 행위는 사용자 영역 A, B 그룹 사이에서 횡적으로 접근하기도 한다. 본 연구에서는 그래서 횡적인 접근에 많이 사용되는 SMB 프로토콜이나 네트워크 프린팅을 수집 및 복원하게 개발되었다.

네트워크 트래픽을 수집 및 복원하고, 행위 정보를 생성, 분석하며, 가시적 분석 기능 및 머신러닝을 통한 분석을 수행하는 행위 분석 프레임워크의 장비 사양은 <표 10>과 같다. 메타정보와 프로파일정보 및 시스템에 대한 자료는 DB를 통해 저장관리되며, 복원 파일은 하둡(Hadoop)을 이용하여 분산처리 하였다.

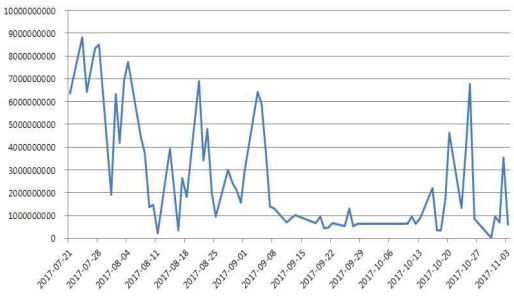
분류	장비명	CPU	OS
분석/저장 장비	행위기반 서버-1	Intel(R) Xeon(R) CPU E5-2623 v3 @ 3.00GHz	CentOS
	행위기반 서버-2	Intel(R) Xeon(R) CPU E5-2623 v3 @ 3.00GHz	CentOS
	행위기반 서버-3	Intel(R) Xeon(R) CPU E5-2623 v3 @ 3.00GHz	CentOS
	시각화/ DB 서버	Intel(R) Xeon(R) CPU E5-2620 v3 @ 2.40GHz, 2cpu	CentOS
	패킷수집 장비	Intel(R) Core™ i7-6700 CPU @ 3.40GHz	Windows 10
모의 환경 장비	FTP 서버	Intel(R) Core™2 Duo CPU T6500 @ 2.10GHz	Ubuntu
	MySQL 서버	Intel(R) Core™2 Duo CPU T6500 @ 2.10GHz	Ubuntu
	Oracle DB서버	Intel(R) Core™2 Duo CPU T6500 @ 2.10GHz	Ubuntu
	메일 서버	Intel(R) Core™ i7-6700 CPU @ 3.40GHz	Windows 10
	SMB 서버	Intel(R) Core™ i7-3770 CPU @ 3.40GHz	CentOS
	MS-SQL 서버	Intel(R) Core™ i7-6700 CPU @ 3.40GHz	Windows 10
	웹서버	Intel(R) Core™ i7-6700 CPU @ 3.40GHz	Windows 10
	Postgres DB 서버	Intel(R) Core™ i7-3770 CPU @ 3.40GHz	CentOS

<표 10> 행위분석 프레임워크 사양

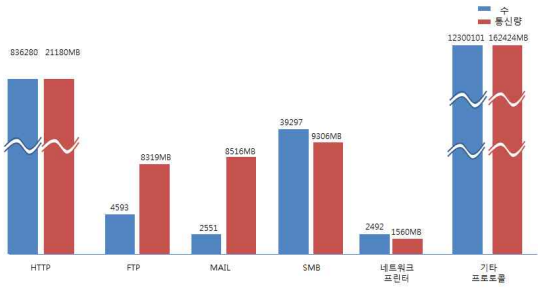
시험 기간은 '17.7.20~'17.11.06 까지 76일간 수행하였고, 시험 횟수는 459회로 시나리오를 각 단계를 다양하게 조합해서 시험했으며 정상적인 네트워크 트래픽도 흐르게 하였다. 행위 탐지기(IOC)는 135개 개발하였고, 2~3개 IOC를 조합하여 새로운 IOC를 생성할 수 있게 개발하였다.

4.2 시험 결과

시험기간 동안 네트워크 트래픽을 수집 및 복원하여 탐지된 사용자 행위의 수는 1,687,951건이며, 시나리오에 의한 악의적인 행위 94,608건 탐지되었다. 시험기간 동안 수집된 네트워크 트래픽은 <그림 23>과 같으며 프로토콜별 건수 및 통신량은 <그림 24>, <표 14>와 같다.



<그림 23> 수집된 총 네트워크 트래픽양(76일간)



<그림 24> 시험기간 동안 수집 건수 및 통신량

	건수	통신량(MB)
HTTP	836,280	21,180.59
FTP	4,593	8,319.90
MAIL	2,551	8,516.23
SMB	39,297	9,306.92
네트워크 프린트	2,492	1,560.15
기타 프로토콜	12,300,101	162,424.26
IP	27,439	-
전체	13,212,753	211,308.05

<표 11> 시험기간 동안 수집 건수 및 통신량

다음은 머신 러닝 모델에 대한 시험 결과이다. 네트워크 통계모델이나 행위 통계모델과 같은 통계모델이 우수한 성능을 보이는 것은 당연한 결과라고 보겠다. 그럼에도 불구하고 네트워크 특성이나 행위 특성을 이용한 모델도 어느 정도 내부자의 이상행위를 판단할 수 있는 성능을 보이는 것을 확인했다.

	F1-Score
행위 특성 모델	0.918
행위 통계 모델	0.998
네트워크 특성 모델	0.818
네트워크 통계 모델	0.997
HTTP 프로토콜 모델	0.884

<표 12> F1-Score

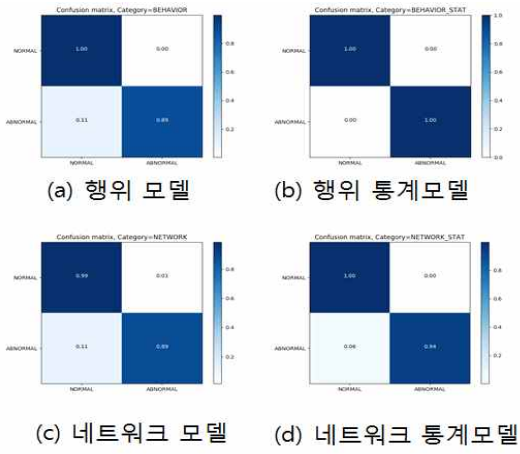
	Accuracy
행위 특성 모델	0.954
행위 통계 모델	0.996
네트워크 특성 모델	0.759
네트워크 통계 모델	0.996
HTTP 프로토콜 모델	0.893

<표 13> Accuracy

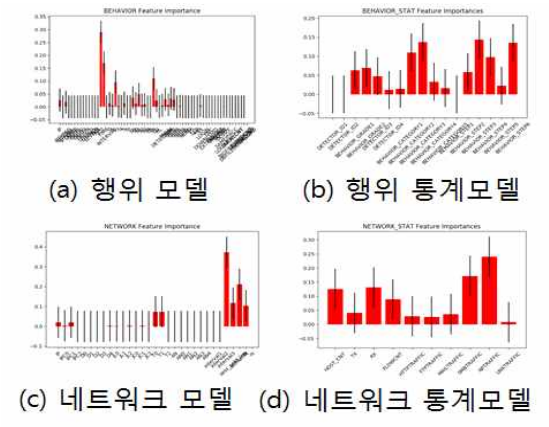
	Recall
행위 특성 모델	0.885
행위 통계 모델	0.997
네트워크 특성 모델	0.887
네트워크 통계 모델	0.998
HTTP 프로토콜 모델	0.876

<표 14> Recall

<그림 25>, <그림 26>은 현재 모델들의 Confusion Matrix와 Feature 중요도이며 계속 Feature Selection과 Feature의 weight를 수정해 가면서 모델 학습 적용중이다.



<그림 25> Confusion Matrix



<그림 26> Feature Importances

Ⅴ. 결론

본 연구에서는 내부자의 악의적인 행위를 탐지하기 위해 내부자의 네트워크 트래픽 수집 및 복원하여 내부자의 행위를 분석할 수 있는 프레임워크를 개발하였고, 이를 통해 수집되는 다양한 네트워크 정보, 프로토콜 정보, 그리고 내부자의 행위에 관련된 정보를 이용하여 머신러닝 기술을 통해 분석하는 기술을 제안하였다. 네트워크 트래픽에 대한 각 레이어별 프로토콜 헤더 및 페이로드 정보들에 SFID(Size, Frequency, Interval and Delay) 측면의 feature들을 추가하여 각각의 모델을 만들어 머신러닝을 수행했으며 기존의 통계적 정보를 이용하는 머신러닝 모델도 수행을 하였다. 본 연구에서는 각각의 모델을 실행하는데 그치지 않고 내부자의 이상행위를 판단하기 위해 각 모델들의 결과를 수치화하여 합산하는 앙상블 모델을 제안하였고, 이를 통해 대규모 네트워크에서 정보유출이 의심되는 IP후보군들을 정확도순으로 도출할 수 있었다. 이를 통해 기존 연구[8]의 룰 기반의 시나리오 탐지기술과 결합하여 내부자 행위의 알려진 위협뿐만 아니라 새로운 형태의 위협에도 대응할 수 있게 되었다.

추후 연구할 분야는 현재 개발된 네트워크 특성 모델, 네트워크 통계모델, 행위 특성모델, 행위 통계모델과 각 프로토콜별 모델들의 결과를 합산하는 수준의 앙상블 모델이 아닌 모델간 balance를 잘 만족하는 회귀분석(regression)을 이용한 앙상블 모델을 개발하려 하며 또한 최근 이슈가 되고 있는 딥러닝 기술을 이용하여 모델을 재개발할 계획이다. 그러나 이를 위해서는 역시 레이블 데이터의 확보가 필수적이며 어려운 부분이라고 하겠다. 최근에는 이러한 정상/이상 위협 데이터를 구축하는 방안에 대한 연구가 많이 이루어지고 있으며 GAN 모델과 같은 인공지능적인 방법을 통해서도 해결책을 강구하고 있으므로 시험 데이터 확보를 위한 기술의 발전을 통해 내부자의 위협을 탐지 및 예측하는 기술 및 모델도 발전할 것으로 기대된다.

참고문헌

[1] Richard C. Brackney, Robert H. Anderson, “Understanding the Insider Threat,” RAND, 2005.

[2] Marcus A. Maloof, Gregory D. Stephens, “ELICT: A System for Detecting Insiders Who Violate need-to-know,” RAID(Recent Advances in Intrusion Detection), 2007, pp. 146-166.

[3] Ted E. Senator 외 26인, “Detecting Insider Threats in a Real Corporate Database of Computer Usage Activity,” ACM SIGKDD, 2013, pp. 1393-1401.

[4] 고장혁,이동호, “GPU를 이용한 정보시스템 성능 향상에 관한 연구,” 한국군사과학기술학회, 종합

학술대회, 2013, pp.391-392.
[5] 고장혁, 이동호, “국방정보시스템 성능향상을 위한 효율적인 GPU 적용방안 연구,” 디지털산업정보학회, 제11권, 제1호, 2015, pp.27-35.
[6] Kalyan Veeramachaneni 외 2인, “AI2: Training a Big Data Machine to Defend,” IEEE BigDataSecurity-HPSC-IDS, 2016, pp.49-54.
[7] 고장혁, 이동호, “정보 유출 탐지를 위한 머신 러닝 기반 내부자 행위 분석 연구,” 디지털산업정보학회, 제13권, 제2호, 2017, pp.1-11.
[8] Richard Bejtlich, “Practice of Network Security Monitoring,” 2013.
[9] 고장혁 외 6인, “Indicator-based Behavior Ontology for Detecting insider Threats in Network Systems,” KSII Transactions on Internet and Information Systems, Vol. 11, No.10, 2017, pp.5062-5079.
[10] Nutan Farah Haq 외 5인, “Application of Machine Learning Approaches in Intrusion Detection System: A Survey,” International Journal of Advanced Research in Artificial Intelligence, Vol.4, No.3, 2015, pp.9-18.
[11] Jeffrey Cleveland 외 3인, “Scalable Machine Learning Framework for Behavior-Based Access Control,” Resilient Control Systems(ISRCS), 2013 6th International Symposium, pp.181-185.

■ 저자소개 ■

고 장 혁
(Kauh Janghyuk)

1998년 3월~현재 국방과학연구소 선임 연구원
2014년 2월 광운대학교 컴퓨터과학과 박사과정수료
1998년 2월 광운대학교 컴퓨터과학과 이학석사
1996년 2월 광운대학교 컴퓨터과학과 이학사

관심분야 : 정보 보호, 내부자 위협, 병렬처리, 머신러닝
E-mail : jhkauh@add.re.kr

이 동 호
(Lee Dongho)

1984년 9월~현재 광운대학교 소프트웨어학부 교수
1988년 2월 서울대학교 컴퓨터공학과 공학박사
1983년 2월 서울대학교 컴퓨터공학과 공학석사
1979년 2월 서울대학교 전자공학과 공학사

관심분야 : 컴퓨터 네트워크, 차세대 인터넷, BAN
E-mail : dhlee@kw.ac.kr

논문접수일 : 2017년 11월 20일
수 정 일 : 2017년 12월 01일
게재확정일 : 2017년 12월 05일