*Journal of The Korea Institute of Information Security & Cryptology* ISSN 1598-3986(Print)
VOL.33, NO.3, Jun. 2023 ISSN 2288-2715(Online)
https://doi.org/10.13089/JKIISC.2023.33.3.499

# 공공기관 실제 사례로 보는 랜섬웨어 탐지 방안에 대한 연구

박 용 주,[1†] 김 휘 강[2‡]
[1,2]고려대학교 정보보호대학원 (대학원생, 교수)

# A Study on Ransomware Detection Methods in Actual Cases of Public Institutions

Yong Ju Park,[1†] Huy Kang Kim[2‡]
[1,2]Korea university School of Cybersecurity (Graduate student, Professor)

## 요 약

최근 지능적이고 고도화된 사이버 공격은 악성코드가 포함된 파일을 이용하여 공공기관의 전산망을 공격하거나 정보를 유출하는 공격으로 그 피해가 커지고 있다. 다양한 정보 보호시스템이 구축된 공공기관에서도 기존의 시그니처 기반이나 정적 분석을 기반으로 하는 악성코드 및 랜섬웨어 파일 탐지하는 방식을 사용하는 경우는 알려진 공격은 탐지가 가능하나 알려지지 않은 동적 및 암호화 공격에 대해서는 취약하다. 본 연구에서 제안하는 탐지 방안은 공공기관에서 실제로 사용하는 정보보호시스템 중 악성코드 및 랜섬웨어를 탐지할 수 있는 시스템의 탐지 결과 데이터를 추출한 후 결합하여 여러 가지 속성을 도출해 내고, 머신러닝 분류 알고리즘을 통해 도출한 속성들이 어떻게 분류되고 어떤 속성이 분류 결과와 정확도 향상에 중대한 영향을 미치는지 실험을 통해 결과를 도출한다. 본 논문의 실험 결과에서는 특정 속성이 포함된 경우와 포함되지 않은 경우 알고리즘마다 상이하지만, 특정 속성이 포함된 학습에서는 정확도가 높아지는 결과를 보였으며 추후 정보보호시스템의 랜섬웨어 파일 및 이상행위 탐지 알고리즘 제작 시 속성 선택에 활용할 수 있을 것으로 기대한다.

## ABSTRACT

Recently, an intelligent and advanced cyber attack attacks a computer network of a public institution using a file containing malicious code or leaks information, and the damage is increasing. Even in public institutions with various information protection systems, known attacks can be detected, but unknown dynamic and encryption attacks can be detected when existing signature-based or static analysis-based malware and ransomware file detection methods are used. vulnerable to The detection method proposed in this study extracts the detection result data of the system that can detect malicious code and ransomware among the information protection systems actually used by public institutions, derives various attributes by combining them, and uses a machine learning classification algorithm. Results are derived through experiments on how the derived properties are classified and which properties have a significant effect on the classification result and accuracy improvement. In the experimental results of this paper, although it is different for each algorithm when a specific attribute is included or not, the learning with a specific attribute shows an increase in accuracy, and later detects malicious code and ransomware files and abnormal behavior in the information protection system. It is expected that it can be used for property selection when creating algorithms.

**Keywords:** Ransomware, Machain-Learning, Information Security System, Weka, Log

Received(04. 13. 2023), Modified(1st: 05. 04. 2023, 2nd: 05. 18. 2023), Accepted(05. 22. 2023)

† 주저자, hackers86@korea.kr
‡ 교신저자, cenda@korea.kr(Corresponding author)

## I. 서 론

디지털 시대에 모든 정보는 디지털화되고 정보의 가치가 높아짐에 따라 기관에 피해를 주기 위한 지능화된 사이버 공격이 늘어나고 있다

특히 악성코드는 기업의 중요 데이터 유출이나 랜섬웨어 감염, 전산망 마비 등 기관에 피해를 주기 위해 악의적으로 작성되어 유포되고 있다. 이러한 공격을 위해 악성코드 유포하기 위한 기술들은 점점 지능화 되고 그로 인한 피해도 커지고 있다. 기관들은 이러한 공격을 방어하고 기관의 정보보안을 위해 다양한 정책과 솔루션을 도입하고 있으며, 방화벽, DDoS, IPS, 안티-바이러스, 망 분리시스템 등과 같은 보안장비들은 각 역할에 맞게 위협을 탐지하며 취약점을 이용한 위험 요소를 차단하고 있다. 하지만 이러한 노력에도 불구하고 정보보안 사고는 지속해서 발생하고 있으며 그 원인에는 임직원이 사용하는 업무용 PC에서 주로 발생하고 있다. 기관에서 정보보안 정책을 강화하고 보안장비를 교체·도입을 해도 외부와 통신을 차단할 수는 없기 때문이다. 이러한 사용자 영역에서 인터넷을 통한 악성코드 유포는 IPS나 안티-바이러스에서 시그니처(Signature) 탐지로 대부분 삭제되거나 차단되지만, Zero-Day 공격에는 취약한 약점이 있다. 최근에는 APT 공격방지 시스템으로 이상 행위를 하는 파일에 대해 격리하거나 삭제가 가능하나 근본적인 사이버 위협에 대응하기 위해서는 악성코드 유입을 막아야 한다. 임직원이 PC 영역에서 파일이 유입될 가능성이 있는 경로를 살펴보면 성인, 음란물, 마약, 도박 등 유해한 정보의 사이트를 통해 악성코드를 다운로드하거나 개인정보나 계정 탈취를 위한 피싱(Phishing) 사이트에서 정보유출을 통한 추가 공격이 많이 사용되고 있다. 피싱은 유명한 사이트로 위장하거나 이메일을 통하여 링크를 첨부하여 사이트로 접속을 유도한 후 정보를 탈취하는 공격 기법이다. 이 기법은 사용자 계정, 이메일 주소, 카드 정보, 사진 등 개인정보와 관련된 정보뿐만 아니라 스팸메일 발송, 악성코드 유포, 전산망 침투 공격 등 사이버 공격을 추가로 수행할 수 있다. 정보보호 실태조사 보고서(2021년)에 따르면 사업체의 93.4%는 정보보호 제품을 이용하고, 제품군별로는 네트워크 보안 85.7%으로 가장 높았고, 시스템(단말)보안 74.7%, 인증 보안 38.2%, 콘텐츠/정보 유출 방지보안 37.0%, 기타 보안관리 32% 등의 순으로 조사되었다[1].

이처럼 신속하게 악성코드 감염탐지, 개인정보 유출 등 정보보안 사고를 예방하기 위해선 전용 솔루션에 의존하고 있는 추세이다. 그러나 인터넷사이트 접속 시 URL은 수시로 신규로 생성되거나 삭제가 이루어지고, 일부 공격을 위한 URL은 알파벳(Alphabet)을 몇 개 변경하거나 대/소문자를 착각하게 하는 방식을 사용하여 블랙리스트(blacklist) 방식이나 시그니처 매칭(Matching) 방식으로 탐지하는 것은 불가능에 가깝다, 또한 이메일로 수신되는 악성 메일은 훈련을 통해 보안 의식을 고취하고 있지만 메일 및 첨부파일 열람 제한에 한계가 있다. 또각 네트워크 구간별로 설치되어 있는 안티-바이러스 소프트웨어는 시그니처 기반으로 최신 버전이 아니거나 시그니처 목록에 없으면 탐지가 불가능하다.

이러한 한계를 극복하기 위해 시그니처 기반의 탐지 기술에서 벗어나 머신러닝 및 딥러닝 기법을 이용하여 악성파일 유입경로를 파악하고, 이상 행위탐지(Abnormal Behavior Detection) 기술을 활용하여 악성파일 유입경로 탐지 및 차단 기법을 도입해야 한다. 현재 이와 관련된 분야에서 활발한 연구들이 진행되고 있다. 기존 연구에서는 특정 보안시스템에서의 이상 행위 탐지하거나 악성 URL의 특성을 이용하여 특정 사이트를 차단하는 연구들이 많았다. 특히 다양한 머신러닝 알고리즘의 성능을 증명하는 연구를 통해 정확도를 높이는 방법을 제안하고 있으나 대부분 전처리 과정을 통한 알고리즘 증명에 관한 결과만 제안하고 있어 기관에 실제 결과를 적용하기 어려웠다. 또한 랜섬웨어를 탐지하고 분석하는 대부분의 연구들은 악성코드를 포함하고 있는 파일을 리버싱을 통해 특성을 파악하고 행동을 분석하거나 사이트 URL과 같은 유입경로를 분석하여 탐지되는 경로의 특성을 찾아낸다. 하지만 이러한 연구들은 기존에 공개된 악성코드나 랜섬웨어 파일이거나 알려진 악성 URL 주소를 분석하는 연구로 이상 탐지나 새로운 형태의 공격 분석에는 한계가 있다.

이러한 기존 연구들은 악성코드나 랜섬웨어 파일 및 악성코드 배포 URL을 분석해 특정 알고리즘의 정확성 높이는 방안을 연구해 왔는데 이러한 연구는 기존에 공개되어있는 데이터에 대해 추가 정밀 분석이 가능하다는 장점이 있지만 새로운 파일이나 기술에 대해서는 즉각 대응이 어렵다는 단점이 있다.

본 연구에서는 파일 또는 URL을 분석하거나 특정 알고리즘의 정확도 분석이 아닌 악성코드 및 랜섬웨어를 탐지할 때 여러 정보보안시스템에서 발생한

탐지 데이터를 추출해 결합한 다음 그 데이터를 활용하여 속성을 도출해 내고 어떠한 속성 조합이 탐지 정확도를 높이는지 확인하고자 한다. 그 방법으로 이기종 정보보안시스템에서 결합한 데이터를 전처리를 통해 학습시킬 데이터를 만들고, 속성들의 조합을 상이하게 알고리즘에 적용함으로써 그 결과 값을 통해 최상의 속성 조합을 찾아낸다. 이는 만약 동일한 파일을 탐지했을 경우 A라는 시스템이 정상 파일로 분류했지만, B라는 시스템은 오탐지로 분류했을 경우 정확도는 떨어질 것이며, 이를 구분하기 위한 고유 속성을 찾아낸다. 이러한 속성을 찾아 탐지 알고리즘에 사용한다면 탐지 정확성을 높이는 성과를 보일 수 있다는 장점이 있지만 이기종 간의 데이터를 분석이 가능한 형태로 결합하고 수집하는 데이터의 내용이 다르기 때문에 분석을 위한 데이터를 만드는 데이터 전처리가 어렵다는 단점이 있다.

본 논문의 구성은 총 5장으로 구성되어 있으며 그 순서는 다음과 같다. 1장에서는 서론, 2장에서는 선행연구를 통해 기존 연구를 분석하고, 3장에서는 연구할 데이터 선택과 전처리 과정을 제안한다. 4장에서는 실험을 위한 데이터와 결과를 분석하여 설명하고, 5장에서 결론과 향후 목표로 끝을 맺는다.

## II. 관련연구

랜섬웨어 및 악성코드를 이용한 사고사례가 늘어나면서 망 분리를 운영하는 기관에 내부망 파일 유입에 대한 방안이 연구되고 있다. 특히 공공기관은 정부 정책이나 가이드라인을 통해 네트워크 통신 구간별로 정보보안시스템 구축을 권고하고 있으며, 구간별 대표적인 시스템은 망 분리(Network Segmentation), 스팸메일차단(Spam-filter), 안티-바이러스(Anti-Virus), 방화벽(Firewall), 망간 자료교환 및 스트리밍 시스템(File-Transfer), 악성 URL 차단(Malicious URL filter), 랜섬웨어탐지(Ransomware detection) 등이 있다.

이기종 시스템별 로그가 상이하고 전처리와 특성을 분류할 수 있는 패턴이 다르기 때문에 사용할 데이터셋을 지도학습(Supervised learning)과 비지도학습(Unsupervised Learning)으로 나누어 머신러닝 학습알고리즘을 분석하였다. 머신 러닝은 학습 데이터(Training Data)에 구분자(Label)가 있거나 없는 경우로 나누어 학습 방법을 구분할 수 있다. 지도 학습(Supervised Learning)은 구분자가 있는 경우에 분류에 용이하기 때문에 학습 방식으로 설명하고, 비지도 학습(Unsupervised Learning)은 구분자가 없는 경우 학습 방식으로 설명한다. 지도 학습 방식은 크게 분류(Classification)와 예측(Prediction) 알고리즘을 사용하여 예측 모델을 개발한다. 비지도 학습 방식은 군집(Clustering) 알고리즘으로 예측 모델을 개발한다[2].

기관의 네트워크 구성은 크게 인터넷 영역, 망 연계 영역, 내부망(업무망) 영역으로 나눌 수 있으며, 다양한 보안시스템으로 이루어져 있지만 본 논문에서는 시그니처 기반으로 악성파일을 탐지할 수 있는 시스템의 로그를 연구 데이터로 사용했다. Fig.1은 기관에서 현재 운영 중인 대표 시스템을 나타낸 구성도이며, 〈Table 1〉은 운영 중인 대표 보안시스템이다.

인터넷 영역 분야에서는 악성URL을 통해 악성파일이 유입되는 과정을 확인하고 탐지하는 연구를 통해 URL 구조의 특성(Feature)을 파악하고, 주요 특성을 24개로 구분하여 분류 알고리즘별(Decision Tree, Random Forest, Gradient boosting machine, XGBoost, Support Vector Machine) 정확도를 확인하였고, 그 결과를 바탕으로 최상의 조합을 찾기 위해 앙상블 알고리즘을 사용해 RF+XGB+GBM 모델 조합이 가장 예측에 효

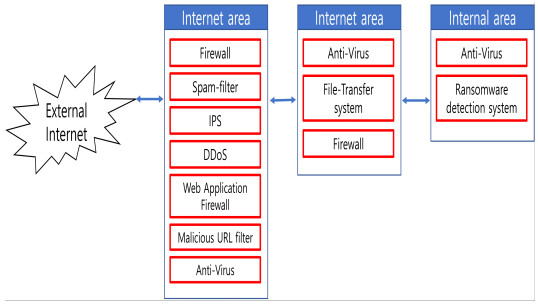


Fig. 1. Network diagram

Table 1. Representative system by section

| Area | Security system |
|---|---|
| Internet area | Firewall, Spam-filter, IPS, DDoS, WAF(Web Application Firewall), Malicious URL filter, Anti-Virus |
| Middle area | Anti-Virus, File-Transfer system, Firewall |
| Internal area | Anti-Virus, Ransomware detection system |

율적임을 증명했다[3].

망간 연계 구간에서는 내부자 자료교환시스템 로그를 활용하는 방법을 확인하기 위해 정보 유출을 탐지 및 예측하는 방안을 제시한 모델을 연구[4]하였으며, 이상 행위를 구분하거나 탐지하는 방법으로 자료교환 사용 패턴을 확인할 수 있는 주요 속성들을 데이터를 군집화를 위해 SOM 및 K-Means 알고리즘을 사용하여 데이터 마이닝(Data mining) 기반의 탐지 모델을 증명했다[5].

업무망 구간에서는 보안시스템에서 추출한 로그를 데이터셋으로 변환하기 위한 방법을 확인하기 위해 내부 직원이 정보시스템을 사용할 때 기록되는 로그를 사용자의 행위를 기반으로 일 단위로 추상화하여 일정 기간에 발생 빈도에 따라 사용자의 다양한 행위를 수치화된 벡터로 발생 빈도를 요약하고, 표현하는 내부 직원 이상 행위 모델링 기법을 확인하였다[6]. 정상, 비정상 행위 데이터가 함께 섞여 있는 데이터셋을 머신 러닝을 통해서 정상치와 이상치를 확인하고 추적했으며 본인이 속한 정상 집단과 다른 행동을 보이는 집단에서 이상 행위에 대한 탐지의 정확성과 적응성을 향상하는 결과를 증명하였다[7].

스팸메일 필터링 구간에서는 기존 많은 연구에서 성공을 거둔 나이브 베이즈 분류기(Naive Bayes Classifier)를 제안하였으며 그 기능과 정확도는 이미 수년 전 검증이 되어 있었다[8][9].

또한 사용자들의 이 상행위를 탐지하는 연구가 랜섬웨어나 악성코드를 탐지하는 방식과 유사하다고 생각하여 이상 행위탐지에 사용하는 머신러닝 알고리즘을 선행 연구하였다. 분류 알고리즘은 두 범주를 구분할 수 있는 경계면을 찾는 것이지만, 이상 탐지는 다수의 범주를 고려해 이상치가 아닌 데이터들의 섹터(Sector)를 구분 짓는 것이라고 할 수 있다. 이상 탐지를 위한 기법은 크게 분류 기반, NN(Nearest Neighbor) 기반, 군집화 기반, 통계적 기법, 스펙트럴 기법 등으로 나뉜다[10].

본 논문에서는 보안시스템의 악성코드 탐지 및 차단 능력을 확인하기 위해 기관에서 운영 중인 시스템의 로그를 추출 후 JOIN을 통해 통합 데이터셋을 만들고, 그 속성 중 최적의 성능을 낼 수 있도록 세분화하여 분류 알고리즘을 통해 학습하였으며, 실제 기관에서 악성코드에 감염된 데이터를 기반으로 검증하는 작업을 수행했다.

## III. 제안하는 방법

### 3.1 연구 데이터

본 논문에서 제안하는 랜섬웨어 탐지 방안은 현재 기관에서 운영 중인 보안시스템 중 Spam-filter, Anti-Virus, Malicious URL filter, File-Transfer system, Ramsomware detection system의 랜섬웨어 및 악성코드 탐지 로그를 수집된 데이터의 용량에 따라 1년 사이의 (2021년 ~2022년) 기간으로 설정하여 추출한다.

Spam-filter 로그는 탐지시간, 발송자IP, 발송자 이메일 주소, 수신자IP, 수신자 도메인 주소, 메일 제목, 송수신 성공 여부, 메일 종류(일반, Spam), Spam filtering 내역, 전달 여부, 잠김 항목으로 구성되어 있으며, Anti-Virus는 인터넷망(Ex), 망간 자료전송구간(Mid), 업무망(In)의 사용 중인 Anti-Virus 소프트웨어를 모두 통합하였고, 멀웨어 종류, 멀웨어명, 그룹명, 사용자명, IP주소, 감염 일자, 감염 횟수로 구성되어 있다. Malicious URL filter는 사이트 차단 발생일시, 사용자 IP, 서버 IP, 서버 포트, 카테고리, 호스트, 서브 URL, 프로토콜로 구성되어 있고, File-Transfer system 로그는 전송시간, 사용자, 멀웨어감염, 개인정보 검출, 승인 구분, 전송상태, 반입·출 구분, 제목, 파일명, 확장자, 사이즈로 구성되어 있다. 마지막으로 Ramsomware detection system은 sip, dip, 분석 대상(파일명), 파일 크기, 해쉬값, PC 명, 그룹, 결과탐지명, 파일명, 사용자, 해쉬값으로 구성되있다.

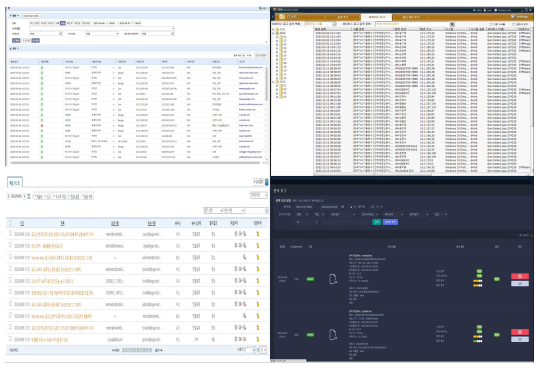

Fig. 2. Various logs from Security Systems

### 3.2 데이터 전처리

실제 어떤 구간에서 어떤 사용자가 어떤 파일이 가장 취약하고, 유입경로가 어디인지 확인할 수 있는지 연구되어야 하기 때문에 학습 전 데이터 분석 및 전처리가 필요하다. 각 보안시스템의 로그를 추출해 랜섬웨어 및 악성코드 감염에 대한 특징을 추출한다. 6개의 시스템 로그에서 추출한 텍스트 중 메일 도메인 주소, 제목, URL의 텍스트에 학습프로그램에 오류를 발생시키는 아스키(ASCII) 문자나 공백, 깨진 글자는 모두 삭제하였다. 사용자 이름의 경우 개인정보보호를 위해 E1, E2, E3 등으로 중복 값을 제외하고 나열했으며 동일 인물일 경우 같은 값으로 분류하였다. [Table 2]는 삭제한 내용을 나타낸다.

정상(Nomal)과 비정상(Abnomal)의 분류는 보안시스템에서 허가되거나(Allow) 일반 파일 및 URL이라고 분류된 내역에 대해 정상 판정을 하였고, 탐지내역 중 악성파일, 멀웨어, 악성 URL 등 허가되지 않은(Deny) 탐지내역에 대해선 비정상 판정을 하였다.

Table 2. Delete word

| Group | Delete word |
|---|---|
| ASCII | ‘(single quote), “(quote), %(percent), blank, ;(semicolon), :(colon), =(equal), !(exclamation mark), 〈(Less Than Sign) |
| Abnormal text | ex) 짱7횅L첩.찼〉V〉4z짖쨋M |

### 3.3 분류 알고리즘 선정

본 논문은 기관에서 운영 중인 이기종 보안시스템의 로그 데이터를 결합하여 데이터셋을 만들고 이를 기반으로 보안시스템이 악성파일을 탐지한 결과의 오용탐지, 미탐지 분류를 위한 가장 최적의 속성을 찾아낸다. 이를 위해 대표적인 분류 알고리즘을 통해 보안시스템에서 악성코드 및 랜섬웨어를 탐지한 결과를 정상과 비정상 조건 관계 분석하여 악성파일 탐지 정확도를 결정할 수 있는 방안을 제안한다.

분석 도구로는 New Zealand Waikato University에서 개발한 WEKA 3.9.6 버전을 사용하여 머신러닝 분석을 진행하였다. 사용한 WEKA 머신러닝 알고리즘은 지도학습 중 분류 알고리즘을 사용했으며, 총 5개의 알고리즘을 사용한다[11].

첫 번째, 트리 분류기(Tree classifier)는 결정트리 학습법을 사용하며 어떠한 값에 대한 관측값과 목푯값을 연결해주는 예측 모델로 결정트리를 사용한다. 대표적인 알고리즘은 ID3, C5.0, CART, CHAID, MARS 등이 있고 WEKA에서는 J48 알고리즘을 사용한다.

두 번째, 규칙 기반 분류기(Rule-based classifier)는 결정트리(Decision Tree) 방식에서 유래한 알고리즘이며 Ripper 알고리즘은 결정트리와 유사한 규칙을 생성하는 알고리즘이다. JRip은 WEKA에서 Ripper를 구현한 것으로 규칙 집합의 어림짐작의 기술인 휴리스틱(Heuristic)을 포함한 전역 최적화 알고리즘이다[12][13].

세 번째, 레이지 분류기(Lazy classifier)는 훈련 항목들을 저장하지만, 실질적인 작업은 분류 작업 전까지 하지 않는 알고리즘이다. IBk는 k-NN(k-Nearest Neighbor)로 최근접 이웃을 찾기 위해 여러 가지 서로 다른 검색 알고리즘을 이용하는 알고리즘이다[14,15]. 거리 측정 알고리즘은 맨하탄 거리(Manhattan Distance), 유클리드(Euclidean Distance), 해밍 거리(Hamming Distance) 등 거리 알고리즘이 있다[16][17].

네 번째, 함수 분류기(Functional classifier)는 분류 알고리즘 중에서 수학식으로 표현할 수 있는 알고리즘이다. SMO는 가우시안 커널(Gaussian Kernel)이나 다항식과 같은 커널 함수를 활용하여 훈련을 위한 순차적 최적·최소 알고리즘을 구현한 것으로 SVM을 활용하고 있다[18].

다섯 번째, 베이즈 정리(Bayes' theorem)를 적용한 확률 분류기의 일종인 베이지안 분류기(Bayesian classifier)이다[19]. 베이즈 정리는

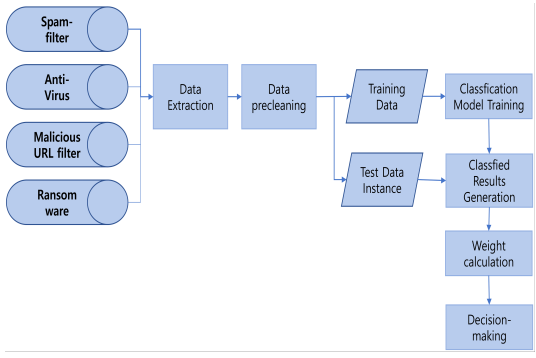


Fig. 3. Machine learning produces

두 확률 변수의 사전·사후 확률 사이의 관계를 나타내는 정리이며, 나이브 베이즈(Naive Bayes)는 베이즈 정리를 이용한 확률적 머신러닝 알고리즘으로 WEKA에서 구현한 것이다.

## IV. 실험 결과

### 4.1 속성 선택(Attribute Selected)

머신러닝 학습에서 학습 성과를 높이기 위해서는 속성 선택이 중요하다. 본 연구에서는 알고리즘별 정확도를 높이기 위한 속성 선택을 위해 결합한 데이터의 속성들의 상관관계를 분석했다. 분석에 사용한 Attribute Evaluator는 ReliefAttributeEval이며, 속성의 연관성을 평가하기 위해 인스턴스 기반의 학습을 이용하는 Relief를 이용해 속성 선택 방법이다. 탐지 시간(time), 공격 IP, 피해 IP, 전송구간(receive, send), type, 탐지 Filter 명, 전송상태, 공격 후 계정 잠김, 첨부파일, 해쉬값으로 이루어진 최초 데이터셋에서 〔Table 3〕과 같이 상관분석 결과가 나왔으며 연관도가 높은 속성 순으로 순위가 정해진 8개의 속성이 선택되었다.

Table 3. Feature Extraction

| Ranked | Attribute |
|---|---|
| 0.5846 | Destination ip |
| 0.5679 | Related system |
| 0.5370 | Source ip |
| 0.3102 | Employee name |
| 0.2582 | Detection time |
| 0.1323 | Department |
| 0.0453 | Employee number |
| 0.0373 | Position |

### 4.2 속성 선정

본 논문에서 과적합을 방지하고 최적화된 성능을 발휘하기 위해 3장에서 전처리한 데이터 중 8개의 속성을 추출하였다. 이기종 시스템 간의 로그는 내용이 모두 다르고 수집하는 형태가 다르기 때문에 가급적 공통으로 포함된 데이터를 속성으로 선정했으며, 〔Table 4〕와 같이 속성이 분류되었다. 각 시스템 로그에서 공통으로 기록하고 있는 탐지 날짜, 출발지 IP(송신지), 목적지IP(피해PC), 부서, 직위, 직원(수신자) 사원번호, 직원(수신자) 이름, 탐지 파일 이름을 선정하고 관련 시스템, 감염 타입을 따로 지정해 분류해 주었다.

Table 4. Detailed properties

| Group | Content |
|---|---|
| time | Dectection time<br>ex) 22-12-30 21:12 |
| sip | Source ip(i.e., sender ip) |
| dip | Destination IP(i.e., employee's ip) |
| dep | Department<br>ex) Directly operated, Gueonggi, Incheon, Tongyeong, etc. |
| po | Position<br>ex) Staff, Team Leader, etc. |
| id | Employee's identification number |
| name | PC user name<br>ex) A012454, A011152, etc. |
| system | Security system<br>ex) Malicious-site, Spam-filter, ransomware, Anti-Virus |
| file_type | nomal, spam-mail, virus, ransomware, deny-site |

구성된 학습 데이터를 각 분류 알고리즘에 대해 10-폴드 교차검증(10-fold cross validation)으로 성능 평가를 수행하였다. 비교 분석을 위한 알고리즘은 WEKA에 탑재된 J48, JRip, IBk, LibSMV, NavieBayes 5개 알고리즘을 사용했다. 또한 Malicious-site filter의 경우 접속 빈도가 너무 높아 동일 URL이 초당 몇천 건씩 발생하고, 랜섬웨어 탐지의 경우는 반대로 탐지 및 차단 빈도가 낮은 점을 고려해 데이터의 양을 조정했다. 〔Table 5〕는 총 실험에 사용한 Dataset 수이다.

Table 5. Dataset

| Type | Normal | Abnormal | Total |
|---|---|---|---|
| Spam-filter | 3,000 | 1,500 | 4,500 |
| Anti-Virus (ex,mid,in) | 2,000 | 1,000 | 3,000 |
| Malicious URL filter | 3,000 | 1,500 | 4,500 |
| ransomware | 2,000 | 1,000 | 3,000 |
| Total | | | 15,000 |

### 4.3 성능 평가 방법

성능 비교를 위해 1차 학습은 분류에서 가장 중요하게 생각한 고유 속성인 sip, dip를 제외한 탐지 결과만 학습시키고, 2차 학습시 sip, dip, id를 추가해 학습시켜 비교 평가한다.

학습 및 검증, 테스트하며 모델 분류의 정확도를 판단하기 위해 정확도(Accuracy), TP Rate(True Positive Rate), FP Rate (False Positive Rate), 정밀도(Precision), 재현율(Recall), F-measure(F1-score) 지표를 활용하여 예측모형별 성능을 측정 분석하고, 예측 유효성은 TP Rate, FP Rate, Accuracy, Recall, Precision, F-measure, ROC(Receiver Operating Characteristic) Area를 평가지표로 활용한다.

각각 대표 평가지표는 〈Table 6〉과 같이 설명할 수 있으며 제일 먼저 데이터를 혼합 행렬(Confusion matrix)로 분류하고 맞는 실제값을 올바르게 예측한 TP(True Positive), 틀린 실제값을 올바르게 예측한 TN(True Negative), 틀린 실제값을 맞다고 잘못 예측한 FP(False Positive), 맞는 실제값을 틀렸다고 잘못 예측한 FN(False Negative)으로 실제 라벨과 예측 라벨의 일치 개수를 Matrix로 표현한다.

Table 6. Classification evaluation metrics

| | | Predicted Class | | |
|---|---|---|---|---|
| | | **Positive** | **Negative** | |
| **Actual Class** | **Positive** | True Positive (TP) | False Negative (FN) Type2 Error | **Sensitivity** $\frac{TP}{(TP+FN)}$ |
| | **Negative** | False Positive (FP) Type1 Error | True Negative (TN) | **Specificity** $\frac{TN}{(TN+FP)}$ |
| | | **Precision** $\frac{TP}{(TP+FP)}$ | **Negative Predictive Value** $\frac{TN}{(TN+FN)}$ | **Accuracy** $\frac{TP+TN}{(TP+TN+FP+FN)}$ |

그 혼합 행렬을 기준으로 전체 대비 정확하게 예측한 개수의 비율 Accuracy를 구할 수 있으며, 가중치 평균(Weighted Average)은 개별치에 각각의 중요도, 영향도, 빈도 등에 따라 가중치를 곱하여 구해지는 평균으로 집단의 변량에 부의 값이 나타나지 않을 경우에 한해서 이용되며, 변량의 극단적인 값에 영향을 덜 받고 비율의 평균법으로 산술평균보다 훨씬 합리적인 장점이 있다.

$$W=\frac{a_1x_1+\cdots+a_nx_n}{a_1+\cdots+a_n}=\frac{\sum_{i=1}^{n}a_ix_i}{\sum_{i=1}^{n}a_i} \quad (1)$$

이와 관련한 수학식은 다소 복잡하여 WEKA에서 구현한 기능을 활용하였고, 그 결괏값을 추출해 반영하였다.

### 4.4 비교 분석

〔Table 7〕과 〔Table 8〕은 데이터별 성능 지표를 설명하고 있다. 2개의 비교실험의 경우 보안시스템 로그에서 고유 속성으로 지정한 sip, dip, id를 속성을 제외한 데이터에 대한 분류기와 포함했을 때 대한 성능(Performance)을 설명하고 있다. 정확성, 재현율, TP Rate는 값이 클수록 좋은 성능을 표현하는 것이고, FP Rate는 오답을 정답으로 분류하고 있는 비율이므로 값이 낮을수록 좋은 것이다.

따라서 〔Table 7〕의 제안 알고리즘 중 J48, JRip이 좋은 성능을 보여주고 있지만 공격자나 피해자를 구분할 수 있는 고유 속성을 제외하고 결과를 측정한 결과 전반적으로 낮은 수치의 정확도를 보여주고 있으며 이는 탐지 패턴을 분류할 때 사용하는 분류 항목과 악성 파일명이 한정되어 있고, 탐지한 내역을 분류하는데 단순 탐지한 내역의 시그니처 매칭으로는 정확도가 낮다는 것을 보여준다.

〔Table 8〕은 공격자의 IP, 직원 PC IP, 사번 등 특정 배포지나 사람을 구분할 수 있는 고유 식별 정보를 추가하여 분류하였을 때 알고리즘별 성능을 보여주고 있다. 이전 고유 속성을 제외한 결과보다 크게 향상된 결과를 보여주고 있으며 이는 추가된 속성값이 다양한 탐지내역에서 공격자를 분류하고, 실제 피해가 발생하면 그 피해자를 지정하고 분류하는데 영향이 크게 작용한 것으로 판단된다.

### 4.5 학습 데이터 모델 수행 결과

Table 7. Performance comparison of machine learning algorithms(NOT Included sip, dip, id)

| Algorithm | J48 | JRip | IBk | SMO | Naive Bayes |
|---|---|---|---|---|---|
| Accuracy(%) | 84.3733 | 78.32 | 78.3267 | 66.6667 | 90.22 |

| Algorithm | Type | Performance [%] | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | | TP Rate | FP Rate | Precision | Recall | F1-score | MCC | ROC Area |
| J48 | nomal | 0.993 | 0.449 | 0.816 | 0.993 | 0.896 | 0.657 | 0.974 |
| | spam-mail | 0.554 | 0.006 | 0.909 | 0.554 | 0.689 | 0.689 | 0.914 |
| | virus | 0.418 | 0.000 | 0.998 | 0.418 | 0.590 | 0.631 | 0.960 |
| | ransomware | 0.000 | 0.000 | - | 0.000 | - | - | 0.948 |
| | deny-site | 0.994 | 0.002 | 0.984 | 0.994 | 0.989 | 0.988 | 0.998 |
| | Weighted Avg. | 0.844 | 0.300 | - | 0.844 | - | - | 0.968 |
| JRip | nomal | 0.985 | 0.619 | 0.761 | 0.985 | 0.858 | 0.501 | 0.688 |
| | spam-mail | 0.295 | 0.008 | 0.789 | 0.295 | 0.429 | 0.455 | 0.693 |
| | virus | 0.196 | 0.002 | 0.892 | 0.196 | 0.321 | 0.401 | 0.645 |
| | ransomware | 0.061 | 0.001 | 0.847 | 0.061 | 0.114 | 0.217 | 0.600 |
| | deny-site | 0.809 | 0.001 | 0.992 | 0.809 | 0.891 | 0.886 | 0914 |
| | Weighted Avg. | 0.783 | 0.414 | 0.802 | 0.783 | 0.732 | 0.509 | 0.702 |
| IBk | nomal | 0.861 | 0.361 | 0.827 | 0.861 | 0843 | 0.511 | 0.829 |
| | spam-mail | 0.564 | 0.030 | 0.650 | 0.564 | 0.604 | 0.569 | 0.926 |
| | virus | 0.500 | 0.037 | 0.527 | 0.500 | 0.513 | 0.474 | 0.892 |
| | ransomware | 0.387 | 0.036 | 0.436 | 0.387 | 0.410 | 0.371 | 0.893 |
| | deny-site | 0.943 | 0.002 | 0.985 | 0.943 | 0.964 | 0.960 | 0.997 |
| | Weighted Avg. | 0.783 | 0.249 | 0.778 | 0.783 | 0.780 | 0.549 | 0.863 |
| SMO | nomal | 1.000 | 1.000 | 0.667 | 1.000 | 0.800 | - | 0.500 |
| | spam | 0.000 | 0.000 | - | 0.000 | - | - | 0.500 |
| | virus | 0.000 | 0.000 | - | 0.000 | - | - | 0.500 |
| | ransomware | 0.000 | 0.000 | - | 0.000 | - | - | 0.500 |
| | deny-site | 0.000 | 0.000 | - | 0.000 | - | - | 0.500 |
| | Weighted Avg. | 0.667 | 0.667 | - | 0.667 | - | - | 0.500 |
| Naive Bayes | nomal | 0.918 | 0.114 | 0.941 | 0.918 | 0.930 | 0.794 | 0.966 |
| | spam-mail | 0.873 | 0.054 | 0.617 | 0.873 | 0.723 | 0.703 | 0.979 |
| | virus | 0.751 | 0.004 | 0.933 | 0.751 | 0.832 | 0.826 | 0.979 |
| | ransomware | 0.822 | 0.007 | 0.899 | 0.822 | 0.859 | 0.850 | 0.994 |
| | deny-site | 0.991 | 0.000 | 1.000 | 0.991 | 0.996 | 0.995 | 1.000 |
| | Weighted Avg. | 0.902 | 0.082 | 0.914 | 0.902 | 0.905 | 0.812 | 0.973 |

Table 8. Performance comparison of machine learning algorithms(Included sip, dip, id)

| Algorithm | J48 | JRip | IBk | SMO | Naive Bayes |
|---|---|---|---|---|---|
| Accuracy(%) | 91.56 | 92.7333 | 86.06 | 98.4067 | 94.2533 |

| Algorithm | Type | performance [%] | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | | TP Rate | FP Rate | Precision | Recall | F1-score | MCC | ROC Area |
| J48 | nomal | 0.245 | 0.891 | 0.891 | 0.997 | 0.941 | 0.815 | 0.945 |
| | spam-mail | 0.002 | 0.979 | 0.979 | 0.844 | 0.907 | 0.901 | 0.991 |
| | virus | 0.001 | 0.988 | 0.988 | 0.997 | 0.993 | 0.992 | 1.000 |
| | ransomware | 0.000 | - | - | 0.000 | - | - | 0.854 |
| | deny-site | 0.000 | 0.999 | 0.999 | 0.985 | 0.992 | 0.991 | 0.999 |
| | Weighted Avg. | 0.164 | - | - | 0.916 | - | - | 0.953 |
| JRip | nomal | 0.990 | 0.193 | 0.911 | 0.990 | 0.949 | 0.841 | 0.903 |
| | spam-mail | 0.989 | 0.001 | 0.990 | 0.989 | 0.990 | 0.989 | 0.999 |
| | virus | 0.989 | 0.001 | 0.987 | 0.989 | 0.988 | 0.987 | 0.999 |
| | ransomware | 0.073 | 0.006 | 0.453 | 0.073 | 0.126 | 0.161 | 0.668 |
| | deny-site | 0.974 | 0.001 | 0.995 | 0.974 | 0.984 | 0.982 | 0.992 |
| | Weighted Avg. | 0.927 | 0.129 | 0.902 | 0.927 | 0.904 | 0.835 | 0.912 |
| IBk | nomal | 0.905 | 0.211 | 0.896 | 0.905 | 0.900 | 0.698 | 0.907 |
| | spam-mail | 0.935 | 0.006 | 0.942 | 0.935 | 0.939 | 0.933 | 0.967 |
| | virus | 0.657 | 0.016 | 0.767 | 0.657 | 0.708 | 0.688 | 0.889 |
| | ransomware | 0.389 | 0.051 | 0.355 | 0.389 | 0.371 | 0.324 | 0.856 |
| | deny-site | 0.965 | 0.002 | 0.984 | 0.965 | 0.975 | 0.972 | 0.985 |
| | Weighted Avg. | 0.861 | 0.146 | 0.863 | 0.861 | 0.861 | 0.721 | 0.915 |
| SMO | nomal | 0.991 | 0.024 | 0.988 | 0.991 | 0.968 | 0.985 | 0.986 |
| | spam-mail | 0.991 | 0.001 | 0.991 | 0.991 | 0.990 | 0.999 | 0.985 |
| | virus | 0.989 | 0.001 | 0.989 | 0.989 | 0.989 | 0.999 | 0.982 |
| | ransomware | 0.880 | 0.007 | 0.903 | 0.880 | 0.884 | 0.985 | 0.826 |
| | deny-site | 1.000 | 0.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| | Weighted Avg. | 0.984 | 0.017 | 0.984 | 0.984 | 0.969 | 0.989 | 0.976 |
| Naive Bayes | nomal | 0.936 | 0.021 | 0.989 | 0.936 | 0.962 | 0.894 | 0.995 |
| | spam-mail | 1.000 | 0.009 | 0.921 | 1.000 | 0.959 | 0.956 | 1.000 |
| | virus | 0.898 | 0.000 | 1.000 | 0.898 | 0.947 | 0.944 | 1.000 |
| | ransomware | 0.898 | 0.046 | 0.583 | 0.898 | 0.707 | 0.701 | 0.982 |
| | deny-site | 0.999 | 0.000 | 0.999 | 0.999 | 0.999 | 0.999 | 1.000 |
| | Weighted Avg. | 0.943 | 0.018 | 0.958 | 0.943 | 0.947 | 0.901 | 0.995 |

# V. 결 론

## 5.1 결론

본 논문에서는 사이버 위협이 증가하고 있는 현 시대에 정부의 정책을 준수해야하는 공공기관이 정보보안 체계를 적절하게 유지하고 있다는 가정하에 보안시스템들의 로그 내용을 바탕으로 탐지 능력을 확인하고, 문제점을 도출한 뒤 탐지 능력을 높이는 방법을 제안했다.

실험에 사용한 연구데이터는 현재 운영 중인 보안시스템 로그를 사용하였으며, 지도학습 알고리즘은 중 WEKA의 분류 알고리즘을 선택했다. 연구데이터셋 속성에서 단순 탐지명, 파일 이름 등을 사용한 데이터에서는 비교적 낮은 정확도를 보여주었으며, 발신자의 ip, 수신자의 ip, 직원의 사번, 이름 등의 고유성을 가진 속성을 추가하여 학습시킨 결과에서는 높은 정확도를 보여주었다. 실험에서 도출된 결과를 보면 공격자가 이메일, 악성파일 유포 URL, 파일명 등은 수정 용이하고 시그니처 패턴을 우회할 수 있는 다양한 형태로 수정할 수 있기 때문에 탐지에 어려움이 있다. 하지만 공격하는 곳이나 피해받는 곳의 고유 속성을 적용한 결과 기존 속성보다 분류 정확도를 향상시킬 수 있었다. 이러한 결과는 현재 이기종의 시스템이 다중으로 보호하는 형태로 구간별 탐지 및 차단하고 있지만 머신러닝 실험을 통해 확인한 결과 단순 비교 탐지는 오용탐지 및 미탐지 내역을 가지고 있다는 것을 보았다. 이는 앞으로 시그니처 기반의 보안시스템이 가지고 있는 한계를 극복하기 위해 발신자나 주로 공격당하는 직원의 정보를 이용해 이 상행위를 탐지하고, 공격당하더라도 사후 대응을 위한 체계를 구축해 대응해야 한다. 이를 위해 추후 차세대 미들웨어(Middle-ware)나 전용 시스템이 도입되어야 함을 보여주었다.

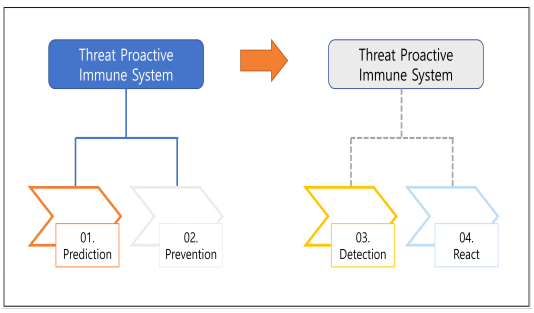


Fig. 4. Future detection system

## 5.2 향후 연구

본 논문에서 사용한 로그들은 실제로 운영되고 있는 보안시스템에서 추출한 것이기 때문에 다양한 문자가 섞여 있고 전처리하는데 상당한 어려움이 있었다. 또한 로그를 대량으로 추출하다 보니 추출 도중 에러가 나거나 멈추는 등 고충이 많았다. 그러한 이유로 다양한 속성을 활용하지 못했고 학습 모델의 정확도를 높이기 위해 더 많은 속성을 적용하는 것이 필요해 보이며, 여기에서 언급하지 않은 기관 내 정보보안시스템과의 조합도 고려해 볼 수 있을 것이다.

그 외에도 Java 기반으로 만들어진 WEKA 프로그램을 통해 실험을 진행함에 따라 다양한 라이브러리 적용이나 전처리 과정, 메모리 부족 등의 제한이 있었다. 추후에는 파이썬(Python), 아나콘다3(Anaconda3), 판다스(Padas) 등 머신러닝 툴을 활용해 실시간 악성URL 탐지 학습, 동적분석을 통한 랜섬웨어 파일 분석 등을 연구할 필요가 있다.

## References

[1] Korea Internet & Security Agency,“ 2021 Information Security Survey Report,” Jan. 2021.

[2] Tae ho Kim, “Feature Selection Optimization in Unsupervised Learning for Insider threat Detection,” Korea University master thesis, Jun. 2018.

[3] Hong Koo Kang, Sam Shin Shin, Dae Yeob Kim, Soon Tai Park, “Design and Implementation of Malicious URL Prediction Systembased on Multiple Machine Learning Algorithms”, Journal of Korea Multimedia Society Vol. 23, No. 11,pp. 1396-1405, Nov. 2020.

[4] In Jae Son, Huy Kang Kim,,“A Study on the Abnormal Behavior Detection Model through Data Transfer”, Journal of The Korea Institute of Information Security & Cryptology

VOL.30, NO.4, Aug. 2020.
〔5〕 Hyun Song Jang, "Data-mining Based Anomaly Detection in Document Management System," Journal of Knowledge Information Technology and Systems(JKITS), Vol. 10, No. 4, pp. 465-473, Aug. 2015.
〔6〕 Hae-dong Kim, "Insider Threat Detection based on User Behavior Model and Novelty Detection Algorithms", Journal of the Korean Institute of Industrial Engineers Vol.43, No.4, pp. 276-278, Aug. 2017.
〔7〕 Eldardiry, H., Sricharn,k.,Liu, j., Hanley,J., Price,B., Brdiczka, O., & Bart,E., "Multi-source fusion for anomaly detection: using across-domain and across-timepeer-group consistency checks", Palo Alto Research Center, Jun. 2014.
〔8〕 Le Zhang, Jingbo Zhu, and Tianshun Yao, "An Evaluation of statistical spam filtering techniques," ACM Transaction on Asian Language Information Processing, 2006.
〔9〕 Vangelis Metsis, "Spam Filtering with NaiveBayes-Which Naive Bayes?," CEAS, June. 2006.
〔10〕 Miae Oh, "A Study on anomaly detection based on MachineLearning", Korea Institute for Health and Social Affairs, Dec. 2018.
〔11〕 Moo-Hun Lee, Min-Gyu Kim, "Meteorological Information Analysis Algorithm based on Weight for Outdoor Activity Decision-Making", Mar. 2016.
〔12〕 William W. Cohen, "Fast Effective Rule Induction", International Conference on Machine Learning, pp. 15-123, July. 1995.
〔13〕 Indrė Žliobaitė, Albert Bifet, Bernhard Pfahringer, and Geoff Holmes, "Active learning with drifting streaming data", IEEE Transactions on Neural Networks and Learning Systems, Vol. 25, No. 1, pp. 27-39, Jan. 2013.
〔14〕 N. Bhatia, "Survey of Nearest Neighbor Techniques", International Journal of Computer Science and Information Security, Vol. 8, No. 2, Jul. 2010.
〔15〕 D. Aha, D. Kibler, "Instance-based learning algorithms". Machine Learning. pp. 37-66, Jan. 1991.
〔16〕 〔16〕 Deza, M.; Deza, E. "Encyclopedia of Distances", Springer-Verlag, pp.94, Jan. 2009.
〔17〕 David M. J. Tax, Robert Duin, and Dick De Ridder, "Classification, Parameter Estimation and State Estimation: An Engineering Approach Using MATLAB", John Wiley and Sons. pp. 440, Sep. 2004.
〔18〕 S.S. Keerthi, S.K. Shevade, C. Bhattacharyya, K.R. K. Murthy, "Improvements to Platt′s SMO Algorithm for SVM Classifier Design", Neural Computation. Vol. 13, No. 3, pp. 637-649. Mar. 2001.
〔19〕 George H. John, Pat Langley, "Estimating Continuous Distributions in Bayesian Classifiers", Eleventh Conference on Uncertainty in Artificial Intelligence, San Mateo, pp. 338-345, Aug. 1995.

## 〈저자소개〉

박 용 주 (Yong Ju Park) 정회원
2013년 2월: 호서대학교 정보보호학과 졸업
2013년 6월~현재: 한국가스기술공사 정보보안부 근무
2020년 2월 고려대학교 정보보호대학원 사이버보안학과 석사과정
〈관심분야〉 정보보호 정책, 머신러닝, AI, 빅데이터 분석, 취약점 분석

김 휘 강 (Huy Kang Kim) 종신회원
1998년 2월: KAIST 산업경영학과 학사
2000년 2월: KAIST 산업공학과 석사
2009년 2월: KAIST 산업및시스템공학과 박사
2004년 5월~2010년 2월: 엔씨소프트 정보보안실장, Technical Director
2010년 3월~현재: 고려대학교 정보호호대학원 교수
〈관심분야〉 온라인게임 보안, 네트워크 보안, 네트워크 포렌직, 침입탐지시스템, 봇넷탐지