SoK: A Systematic Review of Insider Threat Detection

Aram Kim[1], Junhyoung Oh[2], Jinho Ryu[1], Jemin Lee[2], Kookheui Kwon[1], and Kyungho Lee[2*]
[1]*Korea Institute of Nuclear Nonproliferation and Control, Dajeon, South Korea*
{aramkim, halloyu, vivacita}@kinac.re.kr
[2]*Korea University, Seoul, South Korea*
{ohjun02, jeminjustinlee, kevinlee}@korea.ac.kr

Received: November 1, 2019; Accepted: December 7, 2019; Published: December 31, 2019

Abstract

Due to the subtle nature of the insider threat, government bodies and corporate organizations are forced to face the insider threat that is both malicious and accidental. In this paper, we provide a systematic understanding of the past literature that addresses the issues with insider threat detection. Our review consists of three items. First, we examine the different types of insider threats based on insider characteristics and insider activities. Second, we explore the sensors which make possible detecting insider threats in an automated way, and the public datasets available for research. Finally, the detection approaches used in related studies are examined from the perspective of technology, learning, input category, detection target, and interpretability. In particular, we have covered the state-of-the-art deep learning literature that was not covered in previous surveys.

Keywords: insider threat detection, machine learning, deep learning, survey

1 Introduction

An insider threat is defined as "a current or former employee, contractor, or other business partner who has or had authorized access to an organization's network, system, or data and who intentionally (or unintentionally) exceeds or misuses that access to negatively affect the confidentiality, integrity, or availability of the organization's information or information systems" [4]. Security professionals, government agencies, and corporate organizations have found an inherent need to prevent or mitigate attacks from both malicious and negligent insiders. According to Veriato's Insider Threat Report 2018 [71], a survey found that over 73% of the security practitioners have implemented security controls and policies to prevent and mitigate any impending threat. However, despite the implementation of security controls and policies, 90% of the cybersecurity professionals responded that they felt vulnerable to insider threats. The breach level index, a global database that tracks publicly disclosed breaches, revealed that 38.8% of the breaches were caused by unintended accidents (33.62%) or by insiders with malicious intents (5.25%) [28]. Oftentimes, organizations would not disclose their sensitive information breaches to the public. As such, it is difficult to identify and determine the scope and extent of the organization's vulnerability.

With the case of Edward Snowden [53], Bradley Manning [24], and Robert Hanssen [18] insiders can pose a serious threat to the organizations by revealing or exposing sensitive information. Insider threats are difficult to detect, as the insiders already have access to the organization's systems, networks, valuable data, and procedures. Insider also benefits from mobile device enhancements. Companies and governments increasingly use mobile devices that connect to the Internet, and insiders can threaten

Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications (JoWUA), 10(4):46-67, Dec. 2019
DOI:*10.22667/JOWUA.2019.12.31.046*

*Corresponding author: Center for Information Security Technologies (CIST), Korea University, Seoul, Tel: +82-(0)2-3290-4885, Web: `https://www.rimala.net/`

continuously through some manipulations of mobile devices, rather than discrete methods like simple storage like traditional thumb drives.

External attackers usually aim to perform malicious behavior by breaking into an organization's network or system. They are relatively clear to identify because they have to go from outside to inside to gain access to internal networks or systems. But the insiders extracting the information are individuals who are already familiar with the security protocols and systems. Thus, recognizing an insider threat has proven to be a much more difficult task that poses a great amount of threat. For instance, in an air-gapped environment, an external attacker would have to plan a sophisticated attack to overcome this, but an insider could connect to the air-gapped network without any difficulties (e.g., install internet-connected small mobile device in the USB port).

There are also other difficulties in detecting insider threats. In the case of an intelligent insider, he or she would try to conceal their intention and try to stealthily threaten by hiding himself behind in the activities of normal users. It is also challenging to detect threats from unintended negligent insiders because of their subtle nature.

An unintended insider is an attacker who adversely affects the system by human error, mistake, phishing site or virus infection, and may have relatively smaller damage, but the attack is passively conducted and may be difficult to detect. An example of this sort of insider threat is the Stuxnet [37]. The malicious code delivered through a thumb drive by the careless unintended insider which was detected only after more than 1,000 centrifuges had malfunctioned. In 2011, the Nonghyup, a bank located in South Korea, suffered from unintended insider damage [47]. In this case, Nonghyup's subcontractor connected a system maintenance laptop to the internet and inadvertently downloaded the malware, and the infected laptop was used as a channel of attack. The attacker could access the banking system through the air-gapped network using the infected laptop of this subcontractor as a bridge and successfully deleted the credit card customer records, destroyed the banking system and prevented users from using the banking system for three days.

A broad range of insider threat detection approaches have been proposed for insider threat detection, from user pattern analysis using Unix shell commands to behavior change analysis using machine learning, behavior change analysis using deep learning, and efforts to detect insiders proactively by analyzing personality and mood. There has been much progress in these studies, but still detecting insider threats remains a challenging problem. Intelligence Advanced Research Program Activity (IARPA) also launched the Scientific Advances to Continuous Insider Threat Evaluation (SCITE) research program in 2015 to support challenging researchers to study a new class of indicators called Active Indicators and develop Inference Enterprise Models (IEM) that forecast the accuracy of an enterprise in detecting potential threats, for the dramatic improvement of insider threat detection. Also, practical helpful guides such as "Common Sense Guide to Mitigating Insider Threats" [5], "Combating the Insider Threat" [11] by US CERT have been released.

In this paper we systematically study the insider threat including deep learning approaches which were not mentioned in other survey papers with perspective on insider's characteristics and used sensors to detect insider threats. In Section 2, we discuss the related works, and in section 3, we identify the different types of insiders and attack methods. We also present the appropriate detection factors and methods based on the situation. In Section 4, we address the sensors and public dataset from the perspective of an insider threat. In the penultimate section, we discuss machine learning and deep learning techniques for detecting insider threats. Finally, we present our findings in the conclusion.

2 Related Work

In recent years, the literature on insider threat detection has garnered much attention. Past studies have focused on insider threat profile [64, 54], and abnormal detection approaches [69, 7, 77, 29]. However, to the best of our knowledge, there is no study on insider threat detection survey that includes deep learning techniques.

Randazzo et al. [64] examined the actual insider attack cases identified through public reporting in the banking and finance sector. They analyzed the cases with behavioral and technical perspective. The paper also identifies behaviors and communications characteristics before and including the harmful activities. They provided findings that can give insight to develop policy and to conduct future research. However, the paper did not deal with insider threat detection approaches.

Salem et al. [69] provided a comprehensive report of using heterogeneous audit sources. They examined two-class based anomaly detectors using the Schonlau dataset [70] which includes truncated sequences of user commands in the UNIX environment. They also observed other approaches using the Greenberg data [23] that contains enriched data with flags and arguments [48]. They surveyed insider detection methods include Markov chain variations [30, 15], Uniqueness [70], Bayes variations [15, 49, 48, 83], Compression method, Incremental Probabilistic Action Modeling (IPAM) [13], Sequence-math [36], Semi-global alignment [10], Expectation Maximization (EM) [83], and Eigen Co-occurrence Matrix (ECM) [56]. They also surveyed the user profiling method in Windows, Web, and Network bases and concluded that in the Windows environment many audit sources exist and on the network level distinction of a user is challenging but valuable for detecting malicious or unusual activities. Although the paper provides a wide range of insight into insider threat detection, the dataset they used was limited to command sequence based and the state-of-the-art deep learning based detection approaches were not investigated.

Chandola et al. [7] provided a structured and comprehensive overview of anomaly detection techniques. They categorized anomaly detection techniques to provide guidance when applying it to the other domains. The categorized results are depicted in Table 1 and they surveyed each technique's key assumption, basic algorithms, variants, pros and cons, and computational complexity. They surveyed each technique's key assumptions, basic algorithms, variants, pros and cons, and computational complexity. The paper provided an insightful categorization of anomaly detection and provided a very wide range of information for each category. The main achievement of the paper is that it provides an insightful categorization of anomaly detection techniques and a fairly wide range of insight for each category. However, it has not considered anomaly insider, nor did it deal with the latest deep learning techniques.

Chandola et al. (2010) [77] in addition to their previous study focused on detecting anomalies in discrete/symbolic sequences. They classified the previous researches into three based on their problem formulation as follows: "1) Sequence based; 2) Contiguous subsequence based; and 3) Pattern based". They insisted that each problem formulation has a different definition of anomalies and use different anomaly detection techniques. The categorized results are depicted in Table 2 and they surveyed each technique's basic algorithms, variants, pros and cons, and relationship with other categories. They surveyed each technique's basic algorithms, variants, pros and cons, and relationship with other categories.

Jiang et al. [29] surveyed academic publications from 2008-2015 that applied machine learning (ML) in security domains including network security, security service, software & application security, system security, malware, and social engineering & IDS. They described techniques and assumptions that are appropriate for each classification such as network security, security services, and system security but they did not treat insider threat detection.

Several surveys of insider threat detection have been published, but they have mostly focused on audit sources or detection techniques. However, no survey focuses on deep learning, a state-of-the-art technique that has received a lot of attention and progress in recent years, or data sources that have a strong

Factors	Contents
Nature of Input Data	Sequence data, Spatial data, Graph data
Type of Anomaly	Point Anomalies, Contextual Anomalies, Collective Anomalies
Type of Learning	Supervised, Semisupervised, Unsupervised Learning
Output of AD	Scores, Labels
Applications of AD	Intrusion Detection, Fraud Detection Medical and Public Health Anomaly Detection Industrial Damage Detection Image Processing, Text Data Anomaly Detection Sensor Networks, Other Domains
Classification AD	Neural Networks, Bayesian Networks, SVM based, Rule-Based
Nearest Neighbor AD	k^{th} Nearest Neighbor, Relative Density
Clustering AD	Cluster belonging, Distance between clusters, Density based
Statistical AD	Parametric Techniques, Nonparametric Techniques
Information Theoretic AD	Kolomogorov Complexity, entropy, relative entropy, and so on
Spectral AD	Subspacing Techniques

Table 1: Different Aspects of an Anomaly Detection Problem

Problem Formulation	Related Techniques
Sequence based	Similarity based, Window based, Markovian based Hidden Markov Models (HMM) based
Contiguous subsequence based	Window Scoring, Segmentation
Pattern based	Substring Matching, Subsequence Matching Permutation Matching

Table 2: Problem Formulation Based Classification and Related Techniques

relationship with detection performance and desired target. Therefore, we provide a high-level overview of the dataset used for insider threat detection and provides an analysis of the detection technique using deep learning.

3 Insider Threat

To understand insider threats, we will look into the types of insiders, insider activities, and find out the appropriate detection factors and detection methods based on each situation.

Type of Insiders: Insiders can be classified into *intended insiders* and *unintended insiders*. Intended insiders are who can conduct deliberately malicious activities targeted at any organization by a variety of motivations, including revenge, financial need, greed, dissatisfaction, health problems, proclaimed patriotism, notoriety, and political ideology. *Intended insiders* also can be divided into *Traitor* and *Masquerader* [69].

The *traitor* is an insider who already belongs to an organization and already has legitimate access to the organization's resources. Employees or contractors can belong to Traitor. Traitors can take the information more easily because he already knows valuable data, system vulnerable, and their legitimate authority. Besides, since the attack is performed based on the task and authority of the person in charge, there is no time constraint on preparation and execution, and thus, sufficient preparation time and sufficient attack time is already obtained. If a traitor, like Edward Snowden [53], has a lot of knowledge, it

can be more difficult to detect because he can bypass any known security measures and launch a stealthy attack. Also, these kinds of attacks use low frequency and sophisticated methods, making it difficult to be detected. Since the steps of preparation for acquiring the authority can be omitted, they are difficult to be detected during the preparation phase of the kill-chain and are likely to be detected only when or after malicious activity occurs. However, a recent Insider Threat Detection study dealt with proactively identifying people who are more likely to commit insider threats through psychological changes and language habits before insiders perform malicious activities [3, 32, 75]. These studies work best when applied mainly to the traitor.

The *masquerader* is an insider who does not have any legal authority for the desired attack, or who has lower privileges than he wants. They can be low-level employees, former employees, or contractors, and start without sufficient authority, so to perform the desired attack, the process of acquiring an adequate level of authority is necessary. Masquerader can use technical methods (malware installation, key logger installation, internal system sniffing, etc.) or social engineering methods (acquisition of password via an indirect path, use of terminal while away) to obtain authority. They have more time constraints compared to traitors, assuming that an organization enforces some security policy. For this reason, they may have different patterns of behaviors than the existing users, so that we can identify them through changes in behaviors or resource usage patterns. Proactive detection is likely to be less effective when applying to traitors because it affects to predefined ranges of targets.

The *unintended insiders* are who inadvertently launch attacks inside an organization due to inadvertent actions such as breaking security policy. CERT Insider Threat Team defined unintended insider threat as "An unintentional insider threat is (1) a current or former employee, contractor, or business partner (2) who has or had authorized access to an organization's network, system, or data and who, (3) through action or inaction without malicious intent,(4) causes harm or substantially increases the probability of future serious harm to the confidentiality, integrity, or availability of the organization's information or information systems." [74]. They identified threat vectors of unintended insider threats by accidental disclosure (DISC), UIT-HACK (malicious code), PHYS (improper or accidental disposal of physical records), and PORT (portable equipment no longer in possession). Among them, DICS and UIT-HACK can be directly detected using machine learning. Unintended insiders could harm the system without motivation. Stuxnet, the famous Advanced Persistent Threat (APT) attack on Iran's Natanz nuclear enrichment facility, is the case of the unintended insider attack. In this case, an unintended insider was able to bypass the air-gapped network by using a zero-day malware-infected thumb driver in the system and causing more than 1,000 centrifuges to malfunction. So it is difficult to use a proactive method for unintended insiders because they have no intention and an adverse effect would be detected only after occurring. For the unintended insider, a study was conducted to apply the detection method using system operation characteristics such as system usage and network usage.

Insider Activities: The report, published by the CERT Insider Threat Team, categorized four classes of malicious insider activity and analyzed 1,154 actual insider incidents in the United States [5]. Four classes of malicious insider activity are IT Sabotage (179 cases), Fraud (728 cases), Theft of Intellectual Property (268 cases), and Miscellaneous (65 cases). The numbers in parentheses indicate the number of events that occur in each class. Note that the theft of intellectual property includes industrial espionage involving outsiders and that the report did not cover espionage or accidental damage cases. Unintended insiders' activities were not included because this data was obtained through interviews with insiders during the investigation of insider incidents. Each class has the following meaning.

- **IT Sabotage**: Direct harm to an organization or an individual
- **Theft of Intellectual Property (IP)**: Steal IP from the organization
- **Fraud**: Unauthorized modification of an organization's data that to leads to identity crime

- **Espionage**: Practice of spying to acquire classified or proprietary info for foreign entities
- **Miscellaneous**: Cases in which the insider's activity was not for other classes

Table 3 is obtained according to the insider type and insider activities aforementioned. In this paper, we will systematically analyze the detection method according to this viewpoint.

		Traitor	**Masquerader**	**Unintended**
Tech. Level		Low	Medium	Low
Intentionality		Yes	Yes	No
Legitimate		Own Privilege	Privilege Escalation	Own Privilege
Frequency		Rare	Infrequent	Infrequent
Time Curb	**Preparation**	Plenty	Plenty	Not relevant
	Act	Plenty	Limited	Not relevant
Detection Factor	**Preparation**	Behavior Psychological	Behavior Psychological	Policy violation
	Act	System usage Network usage	System usage Network usage	System usage Network usage

Table 3: Insider Categories & Characteristics

4 Selection of Sensors and Public Dataset

4.1 Selection of Sensors

Selecting sensors to be monitored plays a significant role in insider threat detection approaches. The sensor collects data at the point where the insider threat detection system needs to observe. The type and number of sensors have highly correlated with the performance of detection. Poorly selected sensors make it difficult to detect insider threats. For example, if the system does not collect information about the use of the printer, no matter how great a system is, the system won't be able to detect theft of IP through the printed matter. Only properly tuned sensors can collect relevant events and can produce effective results [52].

The data obtained through the sensors can be used as training data for machine learning by binding the data collected for a certain period into a dataset or can be directly connected to a detection system when realtime detection is required. The collected dataset could be used to train a machine learning model and this machine learning model is used to detect malicious insiders by the automated method. Acquiring a dataset is essential because machine learning requires a large amount of data for learning. Insider threat detection also requires realistic data acquisition to train and to test detecting performance. Lots of researchers emphasise on the importance of realistic dataset [63, 69, 52, 62, 58, 8, 1, 17, 44, 65, 40, 76, 55].

Especially Myers et al. [52] pointed out that sensor selection is the major challenge when detecting insider threats. Therefore, in this section, we are going to analyze the existing studies to ascertain what purpose sensors are using. In the case of a study that collected data directly without publishing the dataset, the data used through the contents of the paper will be analyzed, and for the study using the published dataset, the analysis of the frequently used public dataset will be performed.

The most widely used sensor in insider threat detection research is system usage. This can be largely divided into host-based and network-based. Host-based sources are collected from each host (computer, network switch, application/service, etc.). Host-based can collect a wide range of user generated-data

such as Unix shell Command [60, 70, 63, 68], Operating System/Server logs [52, 8, 58, 76, 35, 55, 65, 75] and a wide range of data makes it the most prominent data source.

In the initial study, UNIX shell commands were used to detect insider threat using UNIX *acct* auditing mechanism. Unix shell commands are primarily used to detect masqueraders with the assumption that masquerader commands are different from normal user behaviors. Schonlau et al. [70] collect UNIX command data from 50 users and evaluate several statistical approaches for the detection of masqueraders. The data is composed of only "command name" and "user" with interspersed another 20 masqueraders command and they released the data used in the test for other researchers to use. Maxion et al. [63] point out why Schonlau dataset is not suitable for the masquerade detection with some reason (e.g., The data is not sequential, It is not clear commands are typed by human or script) and give some ideas to improve dataset as like equal lengths of time data, balanced data, time-stamp, removing unnecessary data (e.g., shared scripts), richer data, or job description. Maxion [48] used a modified Greenberg dataset [23] to improve their masquerader detecting accuracy with enriched data because Schonlau data did not allow for any enrichment - no information about flags, aliases, arguments or shell grammar was provided. Greenberg dataset contains the following data: session starts & end time, alias, the current working directory of the users, history use and error status. And this dataset also has four user groups: 1) novice-programmers, 2) experienced-programmers, 3) computer-scientists, 4) non-programmers. Maxion showed that they get improved hit rate and reduced error cost with an enriched dataset. Parveen et al. [59] also modified the Greenberg dataset and they extracted a repeated sequence pattern from the collected data and proposed a method to reduce false alarms in consideration of natural changes over time.

Salem and Salvatore [68] developed Windows and Linux host sensors. They thought that the Schonlau dataset, which is a combination of randomly mixed different user data and simulated masquerader attack, is not suitable for testing a willful act of malfeasance after identity theft. Their Windows sensor monitors all registry-based activity, process creation and destruction, window GUI access, and DLL libraries activity. and gathered data consist of "process name & ID, process path, parent of process, type of process action, process command arguments, success/failure, registry activity results, and timestamp". The Linux sensor collects all process IDs, process names, and process command arguments in real-time. They gathered data from 34 computer science students using developed sensors and through "capture the flag exercise" to obtain masquerader data. They extracted the following features from the gathered data and used them in the test.

- Number of search actions
- Number of non-search actions
- Number of user-induced actions
- Number of window touch (e.g. bringing a window into the fore ground, or closing a window)
- Number of new processes
- Total number of processes running on the system
- Total number of document editing application running on the system

Ted et al. [75] used the Defense Advanced Research Project's Anomaly Detection at Multiple Scales (ADAMS) program which is not allowed to be disclosed publicly. This dataset was collected using a commercial tool called SureView (Raytheon Oakley Systems, Inc.) and this program developed over 100 semantic features. SureView was installed in the user workstation and gathered user actions such as logins, file accesses, emails, instant messages, printer usage, browser usage, and process usage. The following shows the large categories of features and one of its example.:

Domains	Data
Logon	#logons, #PCs with logons, #after house logons #logons on dedicated PC, #of logons on other people's dedicated PC
Device	#device access, #PCs with device access #after house device access, #device usage on dedicated PC #device usage on other people's dedicated PC
File	#file access ,#PCs with file access, #distinct files, #after house file access #file access on dedicated PC, #file access on other people's dedicated PC
HTTP	#web access, #PCs with web access, #URLs visited #after hour web access, #URLs visited from other people's dedicated PC
Email	#emails, #distinct recipients, #internal emails, #internal recipients #emails sent after hour, #emails with attachments, #emails sent from non dedicated PC

Table 4: Daily Feature Vectors for each Domain

- Email : Count of attachments on sent emails (18 features)
- File : Count of file events to removable drives (28)
- Group : Shared printers (11)
- Login : Count of distinct workstations logged onto (4)
- Printer: Count of print jobs submitted (9)
- URL : Count of Blacklist events (13)
- Ratio : Ratio of file events on removable drives to all file events (28)

Lindauer et al. [42] outlined the use of a synthetic data generator for researchers. Although the synthesized data is not as effective as real data, it is essential for the research, so the process of creating test data for insider threat detection is explained, and the created dataset which is called CERT dataset is disclosed on the CERT website. CERT dataset is the most widely used dataset in recent researches. Rashid et al. [65] merged the dataset into user-based actions and performed an experiment using Hidden Markov Model (HMM) by analyzing the sequence of actions every week as a feature. Tuor et al. [76] extracted categorical user attribute features (user's role, department, supervisor) and continuous "count" features (count 408 activities for every 24 hours) from the CERT dataset. Legg et al. [40] also used the CERT dataset and they extracted features from the view of the user's daily observation, comparisons between the user's daily activity and their previous activity, and comparisons between the user's daily activity and the previous activity of their role. Yuan et al. [82] described their Deep Neural Network (DNN) model to detect insider threat and they used CERT dataset too.

The dataset used in Eldardiry et al. [17] is provided by a large defense contractor and not disclosed. They did not give detail about the used dataset but we could presume the dataset by the feature extraction description. They considered six different activity domains as follows: Removable device, logon/off, email sent/received/viewed, file access, and HTTP access. These domains are similar to the aforementioned dataset of Ted [75]. Gavai et al. [20] also used a similar dataset which is consisted of web browsing patterns, email frequency, and file/machine access pattern. The details are depicted in Table 4.

Network-based sources are collected from network communication. In general, data is collected by mirroring network packets used in network communication. If a network uses TCP/IP based protocol, such as source, destination address, service (port number), and network usage can be used as data sources.

If there is prior information on the used application, more detail data can be obtained through deep packet inspection. However, if the protocol information of the application is not disclosed, deep packet inspection cannot be used, so only general information through network flow analysis can be collected.

The research of insider threat detection through the system usage is carried out, and the research results to detect the insider threat by using the contextual of the insider, not the explicit change of the system state, was published. Contextual-based is an approach that uses additional data such as psychological status, employee information, physical movement record, and etc. alone or in conjunction with system usages to identify people who may be insiders before an event occurs. The application of psychological status can be obtained by 1) Analyzing the system usage by linking employee's information such as department, status, technical ability, work, etc. [31], 2) Analyzing the traits of employees through surveys [31], 3) Analyzing whether they have negative moods through language pattern analysis or social network analysis [3, 20, 32, 73], 4) Analyzing abnormal situations through correlations between physical location and system use [69] or other any possible novel approaches.

Kandias et al. [31] proposed a model consisting of Psychological Profiling and Real-Time Usage Profiling. They introduced a user taxonomy that categorized the user into four dimensions: 1) System Role, 2) Sophistication, 3) Predisposition, 4) Stress Level. As a way to get information related to taxonomy, for the system role company's employees information could be used, and interview, questionnaires, and psychometric tests were used for the other categories. For real-time usage profiling, they used system call data.

Brdiczka et al. [3] also proposed a combined model consisting of Structural Anomaly Detection (SA) from social and information networks and Psychological Profiling (PP) from the individual. Interestingly, they performed tests based on data obtained from the online game World of Warcraft (WoW), because there is a very limited amount of data available showing behaviors of malicious insiders. For SA detection, they collected information such as character, guild and fraction based on the list of characters connected to the server for each time slot (5 to 15 minutes) through a crawler. For the PP, they conducted a web-based survey of volunteer game players to gather demographic and personality information. The survey uses 20 items to measure the Big 5 factors (extroversion, agreeableness, conscientiousness, neuroticism, and openness) [67] introduced by the International Personality Item Pool. They analyzed a game character's personality based on 3 kinds of sources - behavioral, text analysis and social networking information. For behavioral they extracted 68 behavioral features that are related to gameplay as like achievements, different types of deaths, character skill. For text analysis they analyzed the "social tone" of game character guild names. And for networking information analysis, they analyzed the friendship and membership network. Although the data they used in the test is data collected from the game, it is noteworthy because it can be similarly applied in real environments.

Taylor et al. [73] conducted a four-stage experiment that was participated by 54 senior psychology students for six hours. Participants were divided into four teams with access to different databases, where some participants were asked to be insiders who leaked data from other users. They tried to find out if insiders could be distinguished by analyzing emails through these tests, and found singularities such as using more personal pronouns than other groups.

Kandias et al. [32] proposed a technique for detecting employees with a negative attitude towards authorities using YouTube as an open-source. They collected user-related data (profile, uploaded videos, subscriptions, favorite videos, playlist), video-related data (license, number of likes and dislikes received, category and tags), and comment-related data (comment text, number of received likes and dislikes) using YouTube's REST-based API for research.

Massberg et al. [43] did not explain direct data acquisition but presented a theoretical model that combined propositions with Dark Triad personality traits [61], Negative attitude, Malicious intent, Trigger, and Motive to overcome the limitations of the widely used Five-Factor Model. Park et al. [57] introduced techniques for detecting insider threat using insiders' sentiment through social network service.

They used the "Sentiment140" dataset which contains almost 1.6 million tweets and user information [21].

The Scientific Advances to Continuous Insider Threat Evaluation (SCITE) program supported the search for novel methods based on active indicators. Active indicators are a way to detect insiders by stimulating them to provoke an indicative response from potential insiders. There is also a study trying to distinguish average users from insiders through eye-tracking [45, 81, 46].

Table 5 summarizes what is described above with the perspective of the target, sensors for system usage, sensors for traits, user information, data sources, is a public dataset, and can detect proactively.

	I[1]	**Sensors (System)**	**Sensors (Psychometric)**	**User Info**	**O**[2]	**P**[3]
[70]	M[4]	Unix Command	N/A	N/A	✓	
[63, 59]		Unix Command	N/A	User ID	✓	
[68]		Process Commands	N/A	User ID	✓	
[42, 65, 76, 40, 82]	M T[5]	Email, File Logon, HTTP Portable Device	Psychometric score Extroversion score Conscientiousness score Latent job satisfaction	LDAP data	✓	
[17]		Email, File Logon, HTTP Portable Device	N/A	Job Role		
[20]		Email, File Logon, HTTP Application	N/A	Organize Info	✓	
[31]	M T U[6]	System Call	Predisposition Stress Level	Computer Skill		✓
[3]	T	Game data	Psychometric score Character Behavior	User name Social network	✓	✓
[73]		Email	N/A	User ID		✓
[32]		N/A	User, Video Comment related	User ID	✓	✓
[43]		N/A	Dark Triad Personality	User name		✓
[57]		N/A	Tweet Data	User ID	✓	✓

1: Type of Insider
2: Public Dataset
3: Detect Proactively
4: Masquerader
5: Traitor
6: Unintended

Table 5: Sensors for Insider Threat Detection

4.2 Public Dataset

As previously mentioned, data plays a significant role in insider threat detection when applying machine learning approaches. However, it is not an easy task for researchers to get their data or create a synthesized dataset using the red team. Fortunately, there are publicly provided datasets for insider threat detection researchers. Public datasets for insider threat detection are usually composed of normal data and synthesized anomaly data. In this section, we will look at some of the major datasets used to research insider threat detection.

Schonlau dataset: [70] introduced a truncated user command dataset, commonly called as Schonlau dataset or SEA (Schonlau Et Al.) dataset. The dataset had been used most widely for academic research. The dataset contains 15,000 UNIX shell commands which were generated with *acct()* system call per every 50 users (the other 20 users simulate masquerade activities). The first 5,000 commands for each user contain clean commands as training data and the rest contains masquerades' data with 5% of probability. Several researches used this dataset for masquerader detection [70, 69, 12, 79, 15, 30, 13]. The Schonlau dataset is just sequences of Unix commands and user names with no other information for example flag, aliases, timestamp, argument, or shell grammar [48, 62], which makes some limitations. Maxion et al. (2004) [63] pointed out why Schonlau dataset is not suitable for the masquerade detection with some reasons (e.g., The data is not sequential, It is not clear commands are typed by human or script) and suggested some ideas to improve dataset (1v49) [49].

Greenberg dataset: Greenberg et al. [23] collected 168 trace files from 168 different users of Unix *csh* (C Shell). They divided users into four groups: 1) novice-programmers, 2) experienced-programmers, 3) computer-scientists, 4) non-programmers. This dataset contains the following data: session start & end time, alias, the current working directory of the users, history use and error status. Greenberg dataset was enriched with information of session start & end time, alias, and the current working directory of the users. Several researches also used this dataset for insider threat detection researches [48, 60, 13, 22, 27, 34].

RUU Dataset: The RUU dataset [2] was created as part of the project to detect traitors and masqueraders. This dataset includes host-based sensors and active trapping technology to detect malicious insiders. However, the file link no longer worked and we could not find the details.

CERT Insider Threat Test Dataset: The CERT Insider Threat Test Dataset [6] is a synthetic insider threat test dataset that includes system logs with annotations of insider threat activity. The dataset is the de facto standard dataset in the insider threat detection domain and several papers [76, 65, 82, 40, 39] using the dataset.

The CERT dataset contains more and more data from r1 to r6.2, but the latest version contains the previous version of superset. CERT dataset is a synthesized one that of 4,000 employees' activities in a virtual organization. Insiders.csv contains the scenario number, detail scenario filename, user id, start and end time. CERT data r6.2 has five scenarios that could occur in a company, such as using a removable drive in off-duty hours, uploading data to wikileaks.org or surfing the job site, and stealing data using a removable drive before quitting the job. The number of insiders corresponding to each scenario are two. The overall number of malicious insider is 10.

Dataset consists of each CSV file according to the user's behavior, and each CSV file is as follows.

- logon.csv : PC on/off log (ID, date, user, PC, activity)
- devices.csv : Removable drive log (ID, date, user, PC, file_tree, activity)
- https.csv : Website access log (ID, date, user, PC, url, activity, content)
- email.csv : Email transceiver log (ID, date, user, PC, to, cc, bcc, from, activity, size, attachments, content)

- file.csv : Removable drive activity log (ID, date, user, PC, filename, activity, to_removable_media, from_removable_media, content)
- LDAP : Employee information (employee_name, user_id, email, role, projects, organization information, supervisor)
- psychometrics.csv : Psychometric information based on the Five Factor Model [67] (user_id, O, C, E, A, N, employee_name)

In addition to the data sets described above, there have been studies using published data. These studies could also confirm the use of Youtube data [32], LinkedIn [78], Text mining techniques [41], or even Online Game data [3]. However, acquiring an appropriate dataset is always a challenging part of the insider threat detection research.

We can also confirm that there are many limitations to the quality of Dataset. The recently created CERT dataset also focuses on information security, dealing only with known attack scenarios. (copying data using a portable drive, uploading data to websites, sending email, etc.). Such a dataset is not appropriate for testing attacks that are naturally occurring attacks by unintentional insiders (e.g., Stuxnet [37]), so there is a need to research more types of attacks samples and to develop datasets using them.

5 Detection Approaches

There have been numerous studies on Insider Threat Detection and most of them used an automated method based on data analysis to find proactive or actionable individuals who are likely to be malicious insiders. The techniques used for automated analysis can be classified into rule-based, machine learning, and deep learning.

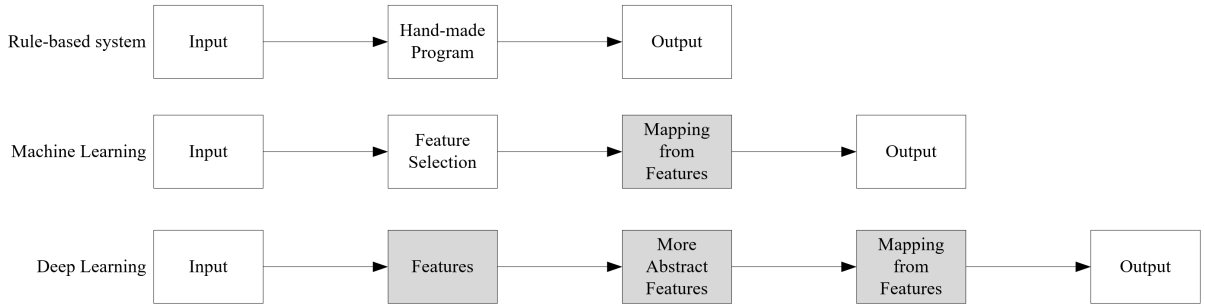

Figure 1: Rule-based, Machine learning, and Deep learning

Rule-based is a method of setting rules based on the knowledge of the domain and detecting the desired behaviors through it. It can detect desired behaviors with low false positives and high accuracy, but it is weak to detect unknown attacks or out of rules. Tom Mitchell provided a definition of machine learning: "A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P if its performance at tasks in T, as measured by P, improves with experience E." [51] Machine learning can detect insider threats by analyzing given data with relatively high accuracy without having to make rules. For this reason, many researchers have used machine learning techniques such as decision trees, support vector machines, and Bayesian networks. Li Deng defined Deep Learning as "A class of machine learning techniques that exploit many layers of non-linear information processing for supervised or unsupervised feature extraction and transformation, and for pattern analysis and

classification" [14]. Deep learning has been attempted in the insider threat detection field because of the feature that does not require feature engineering of deep learning. In this section, we will explore the researches and strengths and weaknesses of Insider Threat Detection using Machine Learning and Deep Learning, except Rule-based.

5.1 Machine Learning

Graph based: Eberle & Holder [16] proposed a Graph-Based Anomaly Detection (GBAD) system. GBAD is an unsupervised approach based on the SUBDUE graph-based knowledge discovery system which has a strong point at detecting using relational information [9]. GBAD represents a flow of information between entities using vertices and edges and finds a normative pattern using a Minimum Description Length (MDL) [66]. GBAD can detect abnormal when the substructure of the graph is different from the normative pattern by more than a certain percentage. These characteristics of GBAD is useful when using for insider threat detection because the insider can be judged abnormal when the insider is doing abnormal behavior. Eberle & Holder examined GBAD using the Enron e-mail dataset, Cell-phone traffic dataset provided by Visual Analytics Science and Technology (VAST), and simulated flow data using OMNeT ++ and GBAD showed good detection performance. However, GBAD can have difficulty finding mimicry behaviors such as using the boss's account to query data while the boss is away for a while. Also, since GBAD is limited to a static and finite-length dataset, real-time detection of stream data is difficult to apply. GBAD uses an unsupervised approach but does not reflect concept-drifted inputs, because normative patterns are already trained. Concept-drifted means that a new input comes in that you have not seen in learning from the same user. For example, GBAD would misidentify legitimate users as a masquerader and may raise a false positive alarm when a novice programmer develops his skill and becomes an expert.

Parveen et al. [58] proposed a new model that compensates for the limitation of the GBAD. The proposed method is an ensemble method that collects the results of a fixed number of GBADs and obtaining the results. This method uses the most recent chunk to create a new GBAD every time, and add it to the ensemble, and remove the worst GBAD from the ensemble to process the concept-drift and to process the stream data. They used the Lincoln Laboratory Intrusion Detection dataset [33] to examine the proposed ensemble system and to show that they could detect insider threats more precisely.

Brdiczka et al. [3] proposed the approach which combines Structural Anomaly Detection (SA) from social and information networks and Psychological Profiling (PP) of individuals. Their approach has the advantage of not only detecting changes in their behavior when detecting malicious insiders but also considering individual personality traits. They explain that they can proactively detect insiders who are likely to be malicious insiders by combining SA and PP to calculate the likelihood of becoming a malicious insider. For SA, they suggest using sequential Bayesian methods for tracking a dynamic entity in Euclidean space and also suggest support vector machine (SVM) or nearest-neighbor based methods for anomaly detection. For PP, they develop a dynamic psychological model that describes temporal patterns of activities leading up to an attack assuming that the emotional state and personality of the perpetrator are related to insider threats. For threat fusion and raking, they proposed a Bayesian method to mix data of SA and PP to reduce false alarm rates because they hope PP could provide semantics. They obtained moderate results using this data and confirmed the possibility of the proposed method. However, the paper did not describe the exact methods used in the experiment, making detailed tracking difficult.

Compression based: Schonlau et al. [70] examined the compression approach in their literature. The Lempel-Ziv-Welch (LZW) algorithm [80] was used to detect malicious insiders when different patterns of commands are typed. They used a simple dataset and resulted in a high missing alarm rate.

Parveen et al. [59] presented an unsupervised insider threat detection algorithm that can handle

evolving normal behavior using a compression-based technique to detect malicious insider. The method suggested by the authors aims to detect the masquerader when a user's command is used as input data, and the legitimate user's patterns are obtained to get out of the command. They use the Lempel-Ziv-Welch (LZW) algorithm [80], a well-known compression algorithm, to extract possible patterns from input data, and keep only the longest patterns of them. The Quantized dictionary (QD) is a set of longest patterns and their corresponding weights. When a new chunk arrives, a newly generated LZW dictionary from the new chunk and previous QD are merged. The dynamic nature of the QD can reduce false positives by reflecting on evolving user behavior. They made some modifications to the Greenberg dataset [23] to reflect realistic scenarios in which the novice programmer gained experience over time and examined their algorithm with high accuracy.

Moarkov Chain based: Schonlau et al. [70] examined Bayes One-Step Markov and Hybrid Multistep Markov in their literature. Those approaches are for detecting a masquerader whose command sequence is not consistent with the historical transition matrix. This method simply calculates the sequence probability of the user's Unix Shell command and attempts to detect malicious insiders, resulting in a high missing alarm rate.

Rashid et al. [65] described a novel method of modeling a normal behaviour of users to detect anomalies which may be indicative of an attack. They made use of Hidden Markov Models(HMM) to learn what constitutes normal behaviour and then use them to detect significant deviations from normal behaviour. The CERT Insider Threat Test Dataset [6] which is a synthetic dataset compromising mainly log files describing a user's computer-based activity was used as an input dataset. A sequence of actions the user took each week was used as an input to the HMM. HMM shows relatively accurate detection performance, but it has the disadvantage of increasing computational cost as the number of states increases.

Ensemble: Instead of using only one machine learning model, the ensemble method can be used to calculate results of several models at the same time and obtain more accurate results.

Ted et al. [75] describe a system that combines structural and semantic information to detect insider threats. In the system, they developed a visual language for specifying combinations of features, baseline, peer groups, time periods, and algorithms to detect insider threat behavior. The anomaly detection language allows users to specify various entities via visual language to detect anomaly domain independently. The paper applies 15 machine learning algorithms such as Relational Pseudo-Anomaly Detection (RPAD), Relational Density Estimation (RDE), Gaussian Mixture Model (GMM), and Ensemble Gaussian Mixture Model (EGMM) to find the right combination for outlier detection. They examined that the system enables flexible anomaly detection in various situations through visualize language and can achieve high accuracy through the ensemble of various detection algorithms. However, the dataset is not disclosed publicly so it can not be verified by other researchers.

5.2 Deep Learning

Statistical or Machine learning approaches have attracted many researchers and have been used for insider threat detection. Those approaches can learn the specific patterns and rules from the data itself and can eliminate efforts required to make rules and can update the model through continuous learning. Determining the performance of the model in machine learning is to find features that can best distinguish instances. Extracting features also require a lot of time based on domain experts' knowledge to be applied.

However, deep learning allows models to automatically extracting features. Deep learning has the form of complex function approximation through linear combination of weight, neuron values, and nonlinear activation function. Deep Neural Networks (DNN) is an artificial neural network that consists of several hidden layers between the input and output layers. DNN can model complex non-linear relation-

ships through many hidden layers. The conventional DNN processes only current inputs, so it is difficult to predict behaviors that reflect the past viewpoint by analyzing the pattern of the time-series data. To solve this problem, a Convolutional Neural Network (CNN) [38] or Long Short-Term Memory (LSTM) [25] has been proposed.

RNN is a kind of artificial neural network where hidden nodes are connected to edges with direction and form a circular structure and is suitable for processing of sequential data (e.g., speech recognition, language modeling, translation, image annotation). It is also widely used in recent years because they can accept inputs and outputs regardless of the sequence length, allowing various and flexible structures. LSTM is a model that adds cell-state to the hidden state of RNN to solve the vanishing gradient problem of RNN. LSTM is well suited to classify the time series data because it employs the LSTM cell to learn the historical experience and because of these characteristics widely used for insider threat detection.

Tuor et al. [76] proposed an online unsupervised deep learning approach to detect anomalous network activities from system logs in real-time. The approach is composed of a feature extractor, a batcher/dispatcher, and the number of RNNs/DNNs. The system user logs feed into the feature extractor and aggregated 408 user activities counts and output one vector for each day. These vector values are used to learn normal behavior using RNNs/DNNs and helped analysts to investigate if the user's action is likely to be anomalous behavior. The key contribution of this study is to decompose anomaly scores. Because decomposing anomaly scores into the contributions of individual user behavior features increase interpretability to aid analysts. To do this, they develop a traversal map which can traverse from right to left among count features. Using this map they get 414 counter features set. The collection of the traversal of all counter features is called "a counter feature set". For the experiment, the CERT dataset r6.2 was used and they showed their model outperformed PCA, SVM, and Isolation Forest. DNN does not detect temporal behavior unlike LSTM, but DNN and LSTM showed similar detection rates. The author speculated that this might be because the CERT dataset does not have enough temporal patterns unfolding over multiple days to offer any real advantage to the LSTM. They aggregate the user actions of each day and use them as input data, so the anomalous behavior happening within one day could not be detected.

Yuan et al. [82] present an LSTM and CNN based model on user behavior to model the normal behavior of users and to detect anomalous behavior. They considered user action behavior like language and used LSTM with long term temporal dependencies on the user action sequence under the assumption that the language pattern changes as anomalous behavior. However, since the LSTM output contains only a single bit of information for each sequence, they fed the LSTM output as abstracted temporal features to the CNN classifier which calculate the anomalous probability of a user action sequence. They showed that their model has fairly accurate detection performance through the experiment with the CERT insider threat dataset [6]. LSTM-CNN method is supervised learning, so it is not suitable for use in the absence of labeled learning data.

Meng and Tian [50] proposed a method based on kernel Principle Component Analysis (PCA) and LSTM-RNN. Their method consisted of the event aggregator, feature extractor, several attribute classifiers, and anomaly calculator. For feature extractors, they used Kernel PCA which uses the Gaussian kernel as its kernel function and reduce the dimensionality of the initial variables while maintaining the variance as much as possible. As a classifier, they selected the LSTM because the LSTM provides good performance for sequence modeling and can learn historical experience. The CERT dataset was used as a training and testing dataset. They compared the performance with machine learning approaches (SVM, Isolation Forest, PCA) and gained outperformed results.

Hu et al. [26] proposed a method of user authentication by learning the movement of the mouse using Deep Learning which is called biobehavioral characteristics. They used an open-source mouse dynamic dataset which is called Balabit Mouse Challenge Dataset [19] as a train and test input. Existing researches require experience and time to extract features from datasets. However, this study eliminated the need for feature engineering by converting datasets into two-dimensional pictures and using CNN for

image processing. Their method offers the possibility of user authentication with shorter authentication time, lower False Acceptance Rate (FAR), and False Reject Rate (FRR) than previous studies.

Table 6 summarizes aforementioned literatures on insider threat detection in terms of technical, learning, input data category used for detection, algorithm, detection target, and interpretability.

	Technique	Type	Data	Algorithm	Detect Target	I[1]
[17]	Machine Learning	S[2]	Static	k-means + Markov model	Abnormal behavior	
[20]				Mutual information	Job quitting	
[57]				SVM, Linear, Naive bayes Decision tree	Negative attitude	
[3]				SVM, Nearest neighbor		
[32]			Stream	Naive bayes multinomial, SVM Logistic regression Dictionary based	Negative attitude	
[65]		U[3]	Static	Hidden Markov Model	Abnormal behavior	
[20]				Isolation forest		✓
[57]				EM, DBSCAN, k-means		
[16]				GBAD		
[55]				EM, k-means, Canopy Density-based		
[8]				KNN, PCA, KNN+PCA		
[70]			Stream	Uniqueness Bayes one-step Markov Hybrid multistep Markov LZW, IPAM, Sequence-math	Masquerader	
[48, 63]				Naive Bayes		
[60, 59]				LZW		
[68]				Hellinger distance, One-class SVM		
[75]				Ensemble (RDE, GFADD, GMM, RIDE, ...)	Abnormal behavior	
[73]				LIWC + LSM	Language change	
[58]				GBADs	Abnormal behavior	
[82]	Deep Learning	S	Static	LSTM-CNN	Abnormal behavior	
[50]		U		LSTM		
[26]				CNN	Masquerader	
[76]			Stream	DNN, LSTM	Abnormal behavior	✓

1: Interpretability
2: Supervised Learning
3: Unsupervised Learning

Table 6: Detection Approaches

6 Conclusion and Future Works

In this paper, we provide a systematic study of existing insider threat detection literatures from different perspectives. We try to classify the insider characteristics, possible sensors, and detection approaches. Specifically, we constructed a matrix at the end of each section that makes it easier to identify the contents of each chapter by our taxonomy. Through this survey, we could identify that various studies confirmed that the results obtained through the machine learning or deep learning based approaches and gathered data from sensors correlate with insider threat detection. Also, various kinds of data collection and various algorithm combinations have been tried to improve the accuracy, and it can be confirmed that it is effective to some extent. However, not much research has been published on active sensors or interpretability, so there is room for further researches in the future. Deep learning techniques usually require a great deal of learning data compare to machine learning techniques. Papers using deep learning showed high accuracy detection results, but it can be expected that the results will differ depending on the size of the training dataset used. We have not compared the performance of the techniques used due to resource limitations. But we could compare the performance of each technique directly in the future. Furthermore, it is necessary to study how to detect insider threats with limited size of data in a situation that data acquisition is challenged due to the technical constraints such as the nuclear power plant control systems that can not install a host agent. Future research can focus on analyzing the effect of event timestamp error on interleaved data and how to overcome it. Shamis et. al. pointed out that timestamps are not perfectly synchronized, because there is a latency over the synchronizing protocol [72]. They developed a very low latency time-synchronized protocol that achieves uncertainties in the tens of microseconds. However, since the typical network does not use this sophisticated method, it is necessary to account for timestamp errors due to synchronization delays.

References

[1] E. T. Axelrad, P. J. Sticha, O. Brdiczka, and J. Shen, "A bayesian network model for predicting insider threats," in *Proc. of the 2013 IEEE Security and Privacy Workshops (SPW'13), San Francisco, California, USA*. IEEE, July 2013, pp. 82–89.

[2] M. Ben-Salem, "Ruu dataset," http://ids.cs.columbia.edu/content/ruu.html [Online; accessed on August 15, 2019], 2009.

[3] O. Brdiczka, J. Liu, B. Price, J. Shen, A. Patil, R. Chow, E. Bart, and N. Ducheneaut, "Proactive insider threat detection through graph learning and psychological context," in *Proc. of the 2012 IEEE Security and Privacy Workshops (SPW'12), San Francisco, California, USA*. IEEE, May 2012, pp. 142–149.

[4] D. M. Cappelli, A. P. Moore, and R. F. Trzeciak, *The CERT guide to insider threats: how to prevent, detect, and respond to information technology crimes (Theft, Sabotage, Fraud)*. Addison-Wesley, January 2012.

[5] C. N. I. T. Center, "Common sense guide to mitigating insider threats, sixth edition," Carnegie-Mellon University Software Engineering Institute, Tech. Rep., December 2018, cMU/SEI-2018-TR-010.

[6] CERT, "Cert insider threat test dataset," https://resources.sei.cmu.edu/library/asset-view.cfm?assetid=508099 [Online; accessed on August 15, 2019], 2016.

[7] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection: A survey," *ACM computing surveys*, vol. 41, no. 3, pp. 15:1–15:58, July 2009.

[8] Y. Chen and B. Malin, "Detection of anomalous insiders in collaborative environments via relational analysis of access logs," in *Proc. of the first ACM conference on Data and application security and privacy (CODASPY'11), San Antonio, Texas, USA*. ACM, February 2011, pp. 63–74.

[9] D. J. Cook and L. B. Holder, "Graph-based data mining," *IEEE Intelligent Systems and Their Applications*, vol. 15, no. 2, pp. 32–41, April 2000.

[10] S. Coull, J. Branch, B. Szymanski, and E. Breimer, "Intrusion detection: A bioinformatics approach," in *Proc. of the 19th Annual Computer Security Applications Conference (ACSAC'03), Las Vegas, Nevada, USA*. IEEE, January 2003, pp. 24–33.

[11] N. Cybersecurity and C. I. Center, "Combating the insider threat 2014," https://www.us-cert.gov/sites/default/files/publications/CombatingtheInsiderThreat_0.pdf [Online; accessed on August 15, 2019], 2014.

[12] S. K. Dash, K. S. Reddy, and A. K. Pujari, "Adaptive naive bayes method for masquerade detection," *Security and Communication Networks*, vol. 4, no. 4, pp. 410–417, April 2011.

[13] B. D. Davison and H. Hirsh, "Predicting sequences of user actions," American Association for Artificial Intelligence, Tech. Rep. WS-98-07, 1998.

[14] L. Deng, D. Yu *et al.*, "Deep learning: methods and applications," *Foundations and Trends® in Signal Processing*, vol. 7, no. 3–4, pp. 197–387, June 2014.

[15] W. DuMouchel, "Computer intrusion detection based on bayes factors for comparing command transition probabilities," National Institute of Statistical Sciences, Tech. Rep. 91, May 1999.

[16] W. Eberle, J. Graves, and L. Holder, "Insider threat detection using a graph-based approach," *Journal of Applied Security Research*, vol. 6, no. 1, pp. 32–81, March 2010.

[17] H. Eldardiry, K. Sricharan, J. Liu, J. Hanley, B. Price, O. Brdiczka, and E. Bart, "Multi-source fusion for anomaly detection: using across-domain and across-time peer-group consistency checks." *Journal of Wireless mobile Networks, Ubiquitous Computing, and Dependable Applications*, vol. 5, no. 2, pp. 39–58, 2014.

[18] FBI, "Famous cases & criminals: Robert hanssen," https://www.fbi.gov/history/famous-cases/robert-hanssen [Online; accessed on August 15, 2019], 2001.

[19] A. Fülöp, L. Kovács, T. Kurics, and E. Windhager-Pokol, "Balabit mouse dynamics challenge data set," https://github.com/balabit/Mouse-Dynamics-Challenge [Online; accessed on December 15, 2019], 2016.

[20] G. Gavai, K. Sricharan, D. Gunning, J. Hanley, M. Singhal, and R. Rolleston, "Supervised and unsupervised methods to detect insider threat from enterprise social and online activity data." *Journal of Wireless mobile Networks, Ubiquitous Computing, and Dependable Applications*, vol. 6, no. 4, pp. 47–63, December 2015.

[21] A. Go, R. Bhayani, and L. Huang, "Sentiment140," http://help.sentiment140.com/site-functionality [Online; accessed on December 15, 2019], 2016.

[22] K. Gopalratnam and D. J. Cook, "Online sequential prediction via incremental parsing: The active lezi algorithm," *IEEE Intelligent Systems*, vol. 22, no. 1, pp. 52–58, Jan 2007.

[23] S. Greenberg, "Using unix: Collected traces of 168 users," Tech. Rep., January 1988, research Report 88/333/45.

[24] T. Guardian, "Bradley manning prosecutors say soldier 'leaked sensitive information'," https://www.theguardian.com/world/2013/jun/11/bradley-manning-wikileaks-trial-prosecution [Online; accessed on August 15, 2019], 2013.

[25] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, December 1997.

[26] T. Hu, W. Niu, X. Zhang, X. Liu, J. Lu, and Y. Liu, "An insider threat detection approach based on mouse dynamics and deep learning," *Security and Communication Networks*, vol. 2019, p. 12, February 2019.

[27] J. A. Iglesias, P. Angelov, A. Ledezma, and A. Sanchis, "Creating evolving user behavior profiles automatically," *IEEE Transactions on Knowledge and Data Engineering*, vol. 24, no. 5, pp. 854–867, May 2011.

[28] B. L. Index, "Breach level index 2018," https://breachlevelindex.com/ [Online; accessed on July 30, 2019], 2018.

[29] H. Jiang, J. Nagra, and P. Ahammad, "Sok: Applying machine learning in security-a survey," arXiv preprint arXiv:1611.03186, November 2016.

[30] W.-H. Ju and Y. Vardi, "A hybrid high-order markov chain model for computer intrusion detection," *Journal of Computational and Graphical Statistics*, vol. 10, no. 2, pp. 277–295, June 2001.

[31] M. Kandias, A. Mylonas, N. Virvilis, M. Theoharidou, and D. Gritzalis, "An insider threat prediction model," in *Proc. of the 7th International conference Trust, Privacy and Security in Digitial Business (TrustBus'10), Bilbao, Spain*, ser. Lecture Notes in Computer Science, vol. 6264. Springer, Berlin, Heidelberg, 2010, pp. 26–37.

[32] M. Kandias, V. Stavrou, N. Bozovic, and D. Gritzalis, "Proactive insider threat detection through social media: The youtube case," in *Proc. of the 12th ACM workshop on Workshop on privacy in the electronic society (WPES'13), Berlin, Germany*. ACM, November 2013, pp. 261–266.

[33] K. K. R. Kendall, "A database of computer attacks for the evaluation of intrusion detection systems," Ph.D. dissertation, Massachusetts Institute of Technology, 1999.

[34] K. Kersting, L. De Raedt, and T. Raiko, "Logical hidden markov models," *Journal of Artificial Intelligence Research*, vol. 25, pp. 425–456, September 2006.

[35] H. Kim, J. Kim, M. Park, S. Cho, and P. Kang, "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. 276–287, August 2017.

[36] T. Lane, C. E. Brodley *et al.*, "Sequence matching and learning in anomaly detection for computer security," Tech. Rep., 1997, aAAI Technical Report WS-97-07.

[37] R. Langner, "Stuxnet: Dissecting a cyberwarfare weapon," *IEEE Security & Privacy*, vol. 9, no. 3, pp. 49–51, May 2011.

[38] Y. LeCun, P. Haffner, L. Bottou, and Y. Bengio, "Object recognition with gradient-based learning," in *Shape, contour and grouping in computer vision*, ser. Lecture Notes in Computer Science, vol. 1681. Springer, Berlin, Heidelberg, 1999, pp. 319–345.

[39] P. A. Legg, "Visualizing the insider threat: challenges and tools for identifying malicious user activity," in *Proc. of the 2015 IEEE Symposium on Visualization for Cyber Security (VizSec'15), Chicago, Illinois, USA*. IEEE, October 2015, pp. 1–7.

[40] P. A. Legg, O. Buckley, M. Goldsmith, and S. Creese, "Automated insider threat detection system using user and role-based profile assessment," *IEEE Systems Journal*, vol. 11, no. 2, pp. 503–512, June 2015.

[41] N. Liang, D. P. Biros, and A. Luse, "An empirical validation of malicious insider characteristics," *Journal of Management Information Systems*, vol. 33, no. 2, pp. 361–392, October 2016.

[42] B. Lindauer, J. Glasser, M. Rosen, K. C. Wallnau, and L. ExactData, "Generating test data for insider threat detectors," *Journal of Wireless mobile Networks, Ubiquitous Computing, and Dependable Applications*, vol. 5, no. 2, pp. 80–94, June 2014.

[43] M. Maasberg, J. Warren, and N. L. Beebe, "The dark side of the insider: detecting the insider threat through examination of dark triad personality traits," in *Proc. of the 48th Hawaii International Conference on System Sciences (HICSS'15), Kauai, Hawaii, USA*. IEEE, January 2015, pp. 3518–3526.

[44] P. Malhotra, A. Ramakrishnan, G. Anand, L. Vig, P. Agarwal, and G. Shroff, "Lstm-based encoder-decoder for multi-sensor anomaly detection," arXiv preprint arXiv:1607.00148, July 2016.

[45] G. Matthews, L. Reinerman-Jones, R. Wohleber, and E. Ortiz, "Eye tracking metrics for insider threat detection in a simulated work environment," *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, vol. 61, no. 1, pp. 202–206, September 2017.

[46] G. Matthews, R. Wohleber, J. Lin, L. Reinerman-Jones, V. Yerdon, and N. Pope, "Cognitive and affective eye tracking metrics for detecting insider threat: A study of simulated espionage," *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, vol. 62, pp. 242–246, September 2018.

[47] T. Maurer, A. Levite, and G. Perkovich, "Toward a global norm against manipulating the integrity of financial data," Economics Discussion Papers, Tech. Rep., 2017.

[48] R. A. Maxion, "Masquerade detection using enriched command lines," in *Proc. of the 2003 International Conference on Dependable Systems and Networks (DSN'03), San Francisco, California, USA*. IEEE, June 2003, pp. 5–14.

[49] R. A. Maxion and T. N. Townsend, "Masquerade detection using truncated command lines," in *Proc. of the 2002 International Conference on Dependable Systems and Networks (DSN'02), Bethesda, Maryland, USA*. IEEE, June 2002, pp. 219–228.

[50] F. Meng, F. Lou, Y. Fu, and Z. Tian, "Deep learning based attribute classification insider threat detection for data security," in *Proc. of the IEEE 3rd International Conference on Data Science in Cyberspace (DSC'18), Guangzhou, China*. IEEE, June 2018, pp. 576–581.

[51] T. M. Mitchell, *Machine Learning*. McGraw-Hill, Inc., March 1997.

[52] J. Myers, M. R. Grimaila, and R. F. Mills, "Towards insider threat detection using web server logs," in *Proceedings of the 5th Annual Workshop on Cyber Security and Information Intelligence Research: Cyber Security and Information Intelligence Challenges and Strategies*. ACM, 2009, p. 54.

[53] B. News, "Edward snowden: Leaks that exposed us spy programme," https://www.bbc.com/news/world-us-canada-23123964 [Online; accessed on August 15, 2019], 2014.

[54] J. R. Nurse, O. Buckley, P. A. Legg, M. Goldsmith, S. Creese, G. R. Wright, and M. Whitty, "Understanding insider threat: A framework for characterising attacks," in *Proc. of the 2014 IEEE Security and Privacy Workshops (SPW'14), San Jose, California, USA*. IEEE, May 2014, pp. 214–228.

[55] J. Oh, T. H. Kim, and K. H. Lee, "Advanced insider threat detection model to apply periodic work atmosphere," *KSII Transactions on Internet and Information Systems*, vol. 13, no. 3, pp. 1722–1737, March 2019.

[56] M. Oka, Y. Oyama, and K. Kato, "Eigen co-occurrence matrix method for masquerade detection," *Publications of the Japan Society for Software Science and Technology*, January 2004.

[57] W. Park, Y. You, and K. Lee, "Detecting potential insider threat: Analyzing insiders' sentiment exposed in social media," *Security and Communication Networks*, vol. 2018, pp. 1–8, June 2018.

[58] P. Parveen, J. Evans, B. Thuraisingham, K. W. Hamlen, and L. Khan, "Insider threat detection using stream mining and graph mining," in *Proc. of the 2011 IEEE Third International Prrivacy, Security, Risk and Trust (PASSAT'11), Boston, Massachusetts, USA*. IEEE, October 2011, pp. 1102–1110.

[59] P. Parveen, N. McDaniel, V. S. Hariharan, B. Thuraisingham, and L. Khan, "Unsupervised ensemble based learning for insider threat detection," in *Proc. of the 2012 International Conference on Privacy, Security, Risk and Trust (PASSAT'12), Amsterdam, Netherlands*. IEEE, September 2012, pp. 718–727.

[60] P. Parveen and B. Thuraisingham, "Unsupervised incremental sequence learning for insider threat detection," in *Proc. of the 2012 IEEE Intelligence and Security Informatics (ISI'12), Arlington, Virginia, USA*. IEEE, June 2012, pp. 141–143.

[61] D. L. Paulhus and K. M. Williams, "The dark triad of personality: Narcissism, machiavellianism, and psychopathy," *Journal of research in personality*, vol. 36, no. 6, pp. 556–563, December 2002.

[62] C. W. Probst, J. Hunker, M. Bishop, and D. Gollmann, *Insider threats in cyber security*. Springer US, 2010, vol. 49.

[63] R. A. Maxion and T. N. Townsend, "Masquerade detection augmented with error analysis," *IEEE Transactions on Reliability*, vol. 53, no. 1, pp. 124–147, March 2004.

[64] M. R. Randazzo, M. Keeney, E. Kowalski, D. M. Cappelli, and A. P. Moore, "Insider threat study: Illicit cyber activity in the banking and finance sector," Tech. Rep., June 2005.

[65] T. Rashid, I. Agrafiotis, and J. R. Nurse, "A new take on detecting insider threats: exploring the use of hidden markov models," in *Proc. of the 8th ACM CCS International Workshop on Managing Insider Security Threats (MIST'16), Vienna, Austria*. ACM, October 2016, pp. 47–56.

[66] J. Rissanen, "Modeling by shortest data description," *Automatica*, vol. 14, no. 5, pp. 465–471, September 1978.

[67] S. Rothmann and E. P. Coetzer, "The big five personality dimensions and job performance," *SA Journal of Industrial Psychology*, vol. 29, no. 1, pp. 68–74, October 2003.

[68] M. B. Salem and S. J. Stolfo, "Masquerade attack detection using a search-behavior modeling approach," Tech. Rep., 2009, technical Report CUCS-027-09.

[69] M. B. Salem, S. Hershkop, and S. J. Stolfo, *A survey of insider attack detection research*. Springer US, August 2008.

[70] M. Schonlau, W. DuMouchel, W.-H. Ju, A. F. Karr, M. Theusan, Y. Vardi *et al.*, "Computer intrusion: Detecting masquerades," *Statistical science*, vol. 16, no. 1, pp. 58–74, February 2001.

[71] H. Schulze, "Veriato insider threat report 2018," https://www.veriato.com/docs/default-source/whitepapers/insider-threat-report-2018.pdf [Online; accessed on July 30, 2019], 2018.

[72] A. Shamis, M. Renzelmann, S. Novakovic, G. Chatzopoulos, A. Dragojević, D. Narayanan, and M. Castro, "Fast general distributed transactions with opacity," in *Proc. of the 2019 International Conference on Management of Data (SIGMOD'19), Amsterdam, Netherlands*. ACM, June 2019, pp. 433–448.

[73] P. J. Taylor, C. J. Dando, T. C. Ormerod, L. J. Ball, M. C. Jenkins, A. Sandham, and T. Menacere, "Detecting

insider threats through language change," *Law and human behavior*, vol. 37, no. 4, pp. 267–275, June 2013.

[74] C. I. T. Team, "Unintentional insider threats: A foundational study," vol. 18, August 2013, cMU/SEI-2013-TN-022.

[75] E. Ted, H. G. Goldberg, A. Memory, W. T. Young, B. Rees, R. Pierce, D. Huang, M. Reardon, D. A. Bader, E. Chow *et al.*, "Detecting insider threats in a real corporate database of computer usage activity," in *Proc. of the 19th ACM SIGKDD International Conference on Knowledge discovery and data mining (KDD'13), Chicago, Illinois, USA*. ACM, August 2013, pp. 1393–1401.

[76] A. Tuor, S. Kaplan, B. Hutchinson, N. Nichols, and S. Robinson, "Deep learning for unsupervised insider threat detection in structured cybersecurity data streams," in *Proc. of the 31st AAAI Workshop on Artificial Intelligence for Cyber Security (AAAI'17), San Francisco, California, USA*. Association for the Advancement of Artificial Intelligence, February 2017.

[77] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection for discrete sequences: A survey," *IEEE Transactions on Knowledge and Data Engineering*, vol. 24, no. 5, pp. 823–839, May 2010.

[78] J. Wang, Y. Zhang, C. Posse, and A. Bhasin, "Is it time for a career switch?" in *Proc. of the 22nd international conference on World Wide Web (WWW'13), Rio de Janeiro, Brazil*. ACM, May 2013, pp. 1377–1388.

[79] K. Wang and S. Stolfo, "One-class training for masquerade detection." IEEE, January 2003.

[80] T. Welch, "A technique for high-performance data compression," *Computer*, vol. 6, no. 17, pp. 8–19, June 1984.

[81] V. A. Yerdon, R. W. Wohleber, G. Matthews, and L. E. Reinerman-Jones, "A simulation-based approach to development of a new insider threat detection technique: active indicators," in *Proc. of the 2018 International Conference on Applied Human Factors and Ergonomics (AHFE'18), Orlando, Florida, USA*. Springer, Cham, June 2018, pp. 3–14.

[82] F. Yuan, Y. Cao, Y. Shang, Y. Liu, J. Tan, and B. Fang, "Insider threat detection with deep neural network," in *Proc. of the 2018 International Conference on Computational Science (ICCS'18), Wuxi, China*, ser. Lecture Notes in Computer Science, vol. 10860. Springer, Cham, 2018, pp. 43–54.

[83] K. H. Yung, "Using self-consistent naive-bayes to detect masquerades," in *Proc. of the 8th Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD'04), Sydney, Australia*, ser. Lecture Notes in Computer Science, vol. 3056. Springer, Berlin, Heidelberg, 2004, pp. 329–340.

Author Biography

Aram Kim received a M.S. and B.S. degrees in Computer Science from Korea University, South Korea in 2005 and 2008. He is currently a Senior Researcher at KINAC, South Korea and is responsible for reviewing and inspecting cyber-security plans of nuclear facilities. His research interests include insider threat detection, abnormal detection and data-driven decision making.

Junhyoung Oh received B.S. degree from Korea University. He is currently studying as a Unified Master's and Doctor's Course Students at Korea University. He is also a member of risk management laboratory in Korea University. His research interests include Usable security, Security Evaluation, Privacy in IoT.

Jinho Ryu received M.S. degree from Seoul National University. He works as a researcher in KINAC, and has been working for improving the current framework of cyber security exercise. He is also participating research project regarding protection of nuclear facility against EMP threat. His research interests include cyber security exercise, protection of EMP threat.

Jemin Justin Lee is a Postdoctoral Associate in Korea University, Republic of Korea. He received his B.A. degree from Glion Institute of Higher Education, Switzerland, in 2013, and his M.S. degree and Ph.D. degree in Electrical and Computer Engineering from Yonsei University, Republic of Korea. His research interests are on Cloud Computing, Network Optimization, and Cyberwarfare.

Kookheui Kwon is currently a manager of the cyber security division at the KINAC's nuclear control headqueaters. He is a Ph.D. candidate in computer security at Chungnam National University and he was a former engineer at KEPCO E&C.

Kyoungho Lee received his Ph.D. degree from Korea University. He is now a professor in the school of information security at Korea University, and has been leading the risk management laboratory in Korea University since 2012. He was a former CISO at NAVER Corporation.