IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received March 11, 2020, accepted April 14, 2020, date of publication April 24, 2020, date of current version May 11, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2990195

A Review of Insider Threat Detection Approaches With IoT Perspective

ARAM KIM[1], JUNHYOUNG OH[2], JINHO RYU[1], AND KYUNGHO LEE[2]
[1]Korea Institue of Nuclear Nonproliferation and Control, Daejeon 34101, South Korea
[2]School of Information Security, Korea University, Seoul 02841, South Korea

Corresponding author: Kyungho Lee (kevinlee@korea.ac.kr)

This work was supported by the Nuclear Safety and Security Commission (id: 10.13039/501100003630).

ABSTRACT Security professionals, government agencies, and corporate organizations have found an inherent need to prevent or mitigate attacks from insider threats. Accordingly, active research on insider threat detection has been conducted to prevent and mitigate adverse effects such as leakage of valuable information that may be caused by insiders. Along with the growth of Internet-of-Things (IoT), new security challenges arise in the existing security frameworks. Attack surfaces are significantly enlarged which could cause a severe risk in terms of company insider threat management. In this work, we provide a generalization of aspects of insider threats with IoT and analyze the surveyed literature based on both private and public sources. We then examine data sources considering IoT environments based on the characteristics and the structure of IoT (perceptual, network, and application layers). The result of reviewing the study shows that using the data source of the network and application layer is more suitable than the perceptual layer in the IoT environment. We also categorized each layer's data sources according to their features, and we investigated research objectives and methods for each category. Finally, the potential for utilization and limitations under the IoT environment are presented at the end of each layer examination.

INDEX TERMS Insider threat detection, Internet-of-Things, dataset, survey.

I. 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'' [1]. 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 Gurucul's 2020 Insider Threat Report [2], a survey found that over 82% of the security practitioners respond that their organizations' insider threat effectiveness is ''some what effective'', ''very effective'', or ''extremely effective''. However, despite the implementation of security controls and policies, 68% of the cybersecurity professionals responded that they were moderately to extremely vulnerable [2]. Only 5% feel that they are not at all vulnerable to an insider attack. 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 intent (5.25%) [3]. 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 to the insider threat.

In the cases of Edward Snowden [4], Bradley Manning [5], and Robert Hanssen [6] insiders can pose a severe threat to organizations by revealing or exposing sensitive information. Insider threats are challenging to detect, as insiders already have access to the organization's systems, networks, valuable data, and procedures. Insiders also benefit from mobile device enhancements. Companies and governments increasingly use mobile devices that connect to the Internet, allowing insiders to launch attacks via these devices in addition to discrete methods like simple storage (i.e., traditional thumb drives).

External attackers usually aim to perform malicious behavior by breaking into an organization's network or system. They are straightforward to identify because they must go from outside to inside to gain access to internal networks or systems. But insiders extracting information are individuals

The associate editor coordinating the review of this manuscript and approving it for publication was Jason R. C. Nurse.

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 significant amount of risk. For instance, to overcome an air-gapped environment, an external attacker would have to plan a sophisticated attack, but an insider could connect to the air-gapped network without any difficulties with Internet-of-Things (IoT) devices (e.g., install Internet-connected small mobile device in the USB port).

The IoT is rapidly growing and fast becoming a fully realized technology. The growth of IoT systems means that billions of various smart things that can connect to the network exist around us, collecting, storing, and processing information [7], [8]. There are already a variety of easily noticeable IoT devices around us. According to Gartner's report in 2017 [9], over 20 billion smart things are expected to be connected by 2020, and Ericson's report [10] forecasts that 29 billion connected IoT devices will come in sight by 2022.

In terms of company insider threat management, the growth of IoT devices could pose a severe threat. Because there are many IoT devices around us, that can sense, store, compute, and communicate information, insiders could take advantage of them, and the boundary of the organization's system extends to all IoT devices. As such, along with the growth of IoT, new security challenges arise in the existing security framework, and attack surfaces are significantly enlarged [11], [12]. One of the attack surfaces that comes from the IoT environment is small-sized devices. Such small computing and network-enabled devices can be a useful attack vector for attackers. For example, in 2013, a report on state-owned TV in Russia reported that wireless spy chip installed irons and kettles could connect to unprotected WiFi networks and spread viruses [13]. And in 2014, Security Research Lab. presented in Black Hat that small IoT devices called ''Bad USB'' could attack a system to perform privilege escalation or change the configuration of the system [14]. Recently in 2019, ZDNet reported that hackers can abuse Amazon Alexa and Google Home devices to eavesdrop on user conversation secretly [15].

The proliferation of IoT devices also poses a massive challenge to threat of insiders. As Nurse *et al.* [16] explained, due to the effects of paradigms such as Bring Your Own Device (BYOD), insiders can more easily commit data leakage, malware, and Denial-of-Service (DoS) attacks through IoT devices. The authors also proposed attack vectors resulting from the introduction of IoT devices. These attack vectors contain taking a video of the login process or intellectual properties with a smart camera, recording a private conversation or meeting with an audio recorder, scanning sensitive IoT items (e.g., credit cards), connecting malware-infected IoT devices into the system, or conducting unauthorized network capturing.

However, research on insider threats considering the IoT environment are still hard to find. In particular, survey papers are more challenging to find. Therefore, in this paper, we examine the existing Insider Threat Detection (ITD) papers from the IoT perspective and suggest what we should consider when detecting insider threats in the IoT environment.

Although a variety of security technologies have been developed, there are always cases where insider threats make systems vulnerable. Because of continued insider threats, there are some patterns of behavior or detection of insider attacks [17]. However, there is a limit to detecting insider threats using existing research. Different studies have defined insider threats in different ways, and the nature and scope of the dataset are different. We analyze the survey papers on insider threats in Chapter II and find out what they do not cover. Chapter III summarizes the definition of insider threats in previous studies and presents the definitions we use. It also describes insider threats in the IoT environment by reflecting the characteristics of the IoT. Chapter IV provides details of each dataset, distinguished between public and private datasets. This allows readers to look at this paper and find other datasets and methodologies suitable for them by other researchers and industries. In Chapter V, we analyze insider threat detection studies by dividing them into perceptual, network and application layers. Chapter VI discusses the analyzed results, and the study concludes.

A. SURVEY APPROACH

The analyzed literature was found through the Google Scholar service using ''insider threat detection'' as queries. Of the first 100 results, we excluded 23 papers that were not published in the last ten years because they were already fully covered in other surveys. And after the evaluation of the abstracts and considering the number of citations, we did not include 38 papers that were not appropriate topics or had less than ten citations. In addition, for papers published by the same group of researchers (Eberle, Parveen, Legg, Kandias, Ted, and Agrafiotis), 15 overlapping papers were excluded by selecting only the most recent or with many citations.

In the process of examining each article's reference list, materials that were frequently mentioned [18]–[24] were added to the final list even though they were not published recently. We also included three papers that attempted a novel approach [25]–[27], one article that investigated network-based detection in IoT environments [28], one article about insider threat detection in IoTs [29], and two articles of attack vectors considering IoTs [12], [16]. Deep-learning related ITD researches [30], [31] that show the excellent capabilities of detecting unseen behavior patterns have also been added. Afterwards, the final 40 research papers were selected [12], [16], [18]–[55]. Figure 1 depicts categories of the selected literature.

II. EXISTING SURVEYS AND OUR CONTRIBUTION

Focusing on the computer security application field, Salem and Stolfo [56] suggested two types of malicious insiders based on an insider's knowledge about the target system

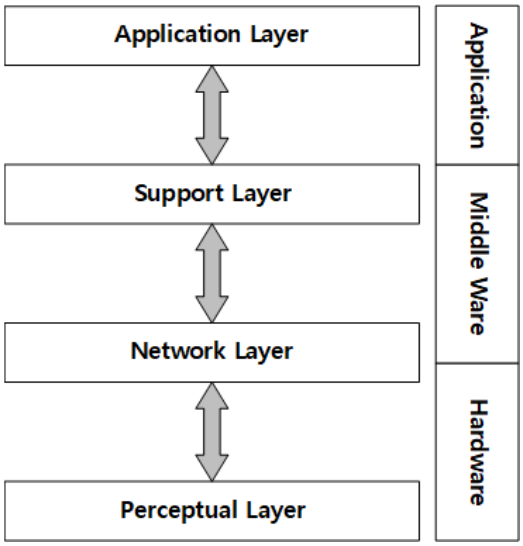

FIGURE 1. General architecture of IoTs.

TABLE 1. Selected literature.

Literature	Category
[32]–[55]	Google Scholar (Top 100 published in last 10 years)
[18]–[24]	Frequently mentioned among literature
[25]–[27]	Novel attempt (Mouse dynamics, Eye tracking)
[12], [16], [28], [29]	IoT Environment
[30], [31]	Deep learning approach

to attack: traitors and masqueraders. As a review of detection techniques, the authors reviewed related researches and divided them into three types of approaches: 1) host-based strategies (Unix, Windows, Web, Program), 2) network-level approaches (HTTP, SMB, SMTP, FTP), 3) honeypots, and 4) integrated methods. With this categorization, the authors claim that both a host-based and network-level approach can have a high chance of detecting traitors, while the host-based strategy may be successful in identifying masqueraders. Although the paper provides a wide range of insight into ITD algorithms in various environments, they did not provide a detailed analysis of the dataset.

For the different fields of applications, Chandola *et al.* [57] provide a survey on anomaly detection techniques. The authors made a discussion for several different application domains, including intrusion detection, fraud detection, medical anomaly detection, image processing, et cetera. As each domain may have different detection techniques, they grouped existing techniques into different categories based on the unique assumptions that each method has adopted. They also assessed the advantages and disadvantages of each technique. According to the authors, each assumption has a different notion of normal behavior. Thus the assumption can be used as guidelines to assess the effectiveness of the technique in that domain. Finally, they suggested promising directions for future studies, such as contextual, collective, and distributed anomaly detection techniques. In our work, we use their ''online anomaly detection'' category to evaluate the detection approach since it is suitable for IoT environments.

Jiang *et al.*'s [58] survey machine-learning techniques can be utilized for various computer security domains, including intrusion detection systems, software security, security policy management, identification of malware, mitigation of malware, et cetera. Their discussion not only focuses on the insider threat but the overall attack causing security problems. They analyzed the attack detection system in each study, summarizing the goal and conceptual system components, and selected the machine-learning technique of the system. As a result, they suggested a taxonomy of machine-learning used in various security domains and recommended system design of Game-Theory based and Human-in-the-loop machine-learning techniques. Their survey covered machine learning applied to security and is useful for approaching machine learning used for ITD, but has a limitation that does not cover insider threats.

Gheyas and Abdallah [59] conduct a systematic literature review of insider detection and prediction studies. They found the most popular research trends in the field as follows: dataset - Game Theory Approach (GTA), feature - insider's online activities, and algorithm - graph algorithm. They also ranked research works to suggest the recommended practice of detection and prediction study, considering the theoretical merits (how many the research addresses challenging points in the research field) and transparency of the study (how much the study is replicable). In our work, we were inspired by their feature space exploration section in describing challenges associated with IoT.

Homoliak *et al.* [60] present a comprehensive survey of insider threat research. They categorized existing insider threat papers into four different categories based on each paper's contribution to the research field: 1) Incidents and Datasets, 2) Analysis of Incidents, 3) Simulation, 4) Defense Solutions. According to this categorization, research regarding insider threats can adopt either a top-down (from Incidents and Datasets to other research fields) or bottom-up (from Defense Solutions&Simulation to Incidents and Datasets) approach. Moreover, the authors present the structural taxonomy of insider threat incidents based on 5W1H questions, and these questions are very helpful in figuring out the insider threat itself. With this, they provide a unified view by incorporating existing taxonomies, as well as making additional subcategorization to several areas that have never been discussed before. However, since they cover a very wide range of insider threats, they cover only a relatively small portion of detection approaches.

Our previous work [61] surveys ITD techniques. The research contains a categorization of insiders and an analysis of existing literature in terms of data sources and using datasets. The survey systematically reviewed ITD approaches of existing literature that were using machine learning in

terms of 1) type of insiders, 2) used system sensors, 3) used psychometric sensors, and 4) point of detection (proactive, reactive).

Previous ITD studies including our previous work [61] were conducted mainly considering the typical office environment. The terminals are composed of computers using widely used operating systems like Windows, Linux, or Unix. These terminals are connected to servers providing a service through a network using TCP/IP. The malicious insider sits in front of their own terminal or a colleague's terminal and tries to steal intellectual property, conduct espionage, or make illegal changes using its authority (traitor) or privilege escalation (masquerader). However, in an IoT environment where everyday things (e.g., smartwatch, smart sensors, air purifier, and smart health devices) connect to the Internet, the attack surfaces are much more significant, and insider detection studies also need to be analyzed from the IoT perspective. However, there is no survey paper on ITD considering the IoT environment.

The main contributions of this survey can be summarized as follows. First, to the best of our knowledge, this is the first work that surveys ITD literature with the IoT perspective and provides an analysis of the points to consider when performing ITD in the IoT environment. Second, we provide the characteristics of ITD in the IoT structure that researchers can use when conducting related research. Third, we provide an analysis of how to consider the IoT environment in selecting and utilizing data sources, the most critical point in ITD. Finally, we provide an analysis of the elements to consider when applying ITD approaches in the IoT environment.

III. INSIDER THREAT AND IoT CHARACTERISTICS

A. INSIDER THREATS

This section describes the definitions of insiders and insider threats and then categorizes their activities to establish common concepts related to such attacks.

1) DEFINITION OF THE INSIDER AND INSIDER THREAT

a: INSIDER

The Rand Corp [62] defined an insider as, ''anyone with access, privilege, or knowledge of information systems and services.'' They also defined a malicious insider as ''motivated to intentionally adversely impact an organization's mission. (e.g., deny, damage, degrade, destroy)'' According to Greitzer *et al.* [47] the definition of the insider refers to, ''an individual currently or at one time authorized to access an organization's information system, data, or network; such authorization implies a degree of trust in the individual.'' Parveen *et al.* [63] also depict insiders' features by saying, ''insiders are often intimately familiar with the internal working of a system and conceal their actions by molding them very closely to legitimate tasks and activities carried on by the system.'' According to Kont *et al.* [64] the definition of an insider is, ''a member of an organization, an associate (contractor, business partner or guest), anyone with authorization to perform certain activities, anyone who is authenticated by the system (including unauthorized users using valid credentials), or an unwilling or coerced accomplice to an external actor.''

In this paper, we expand the scope of the insider, because attacks such as collecting information is possible by simply placing small IoT devices. The definition of the insider we use is, ''someone who has the authority to enter an organization, whether employees, contractors, or guests, regardless of their authority of the information system.''

b: INSIDER THREAT

There are definitions for insider threats such as when Gavai *et al.* [43] defined the insider threat as ''threats with malicious intent directed towards organizations by people internal to the organization.'' Greitzer *et al.* [65] defined them as, ''harmful acts that trusted insiders might carry out something that causes harm to the organization, or an unauthorized act that benefits the individual.'' They also claimed, ''the insider threat is manifested when human behavior departs from compliance with established policies, regardless of whether it results from malice or disregard for security policies.'' According to Hunker and Probst [66] the definition of the insider threat refers to, ''an individual with privileges who misuses them or whose access results in misuse.''

2) CATEGORIES OF INSIDERS

Insiders can be classified into intended insiders and unintended insiders. Intended insiders are those 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 can also be divided into traitors and masqueraders [56].

a: TRAITOR

The traitor is an insider who already belongs to an organization and has legitimate access to the organization's resources. Employees or contractors may assume the role of a traitors. Traitors can take the information more easily because they already know where the valuable data is stored, how it is protected, and have knowledge of existing vulnerabilities. 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 and attack time is already obtained. Note that the traitor used in this paper, regardless of its dictionary meaning, technically means any employee with the proper authority to cause an insider threat. As such, the ethical judgment of whether a traitor is a whistleblower or a villain is beyond the scope of this paper. If a traitor has much knowledge, it can be more challenging 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, insiders are difficult to

be detected during the preparation phase of the kill-chain and are likely to be identified only during or after malicious activity occurs. However, a recent ITD 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 [20], [40], [41]. These studies work best when applied mainly to the traitor.

b: MASQUERADER

The masquerader is an insider who does not have any legal authority for the desired attack, or has lower privileges than they want. They can be low-level employees, former employees, or contractors, and start without sufficient authority to perform the desired attack thus requiring the insider to acquire the adequate level of authority as necessary. Masquerader can use technical methods (malware installation, key logger installation, internal system sniffing) 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 applied to traitors because it affects predefined ranges of targets.

c: UNINTENDED INSIDERS

The unintended insiders are those who inadvertently launch attacks inside an organization due to inadvertent actions such as breaking security policy. The CERT Insider Threat Team defined an unintentional insider as ''(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'' [67]. They identified threat vectors of unintended insider threats by accidental disclosure (DISC), UIT-HACK (malicious code), PHYS (improper or unintentional 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, was a case of an unintended insider attack. The Stuxnet operation had an extensive plan of causing the staff of the facility to insert a USB drive unintentionally. Hence, the employee who inserted the infected USB into the facility had no intention of spreading the worm. In this case, an unintended insider was able to bypass the air-gapped network by using a zero-day malware-infected thumb drive in the system and caused more than 1,000 centrifuges to malfunction. Therefore, 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, studies were conducted to apply the detection method using system operation characteristics such as system usage and network usage.

3) CATEGORIES OF INSIDER ACTIVITIES

The CERT Insider Threat Team published a guide to mitigating insider threats [68]. In the guide, they categorized the malicious insider activities into four types after analyzing 1,154 actual insider incidents in the United States. The four classes of malicious insider activity are IT Sabotage (179 cases), Fraud (728 cases), Theft of Intellectual Property (268 cases), and Miscellaneous (65 cases). 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)**: Stealing 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

B. INTERNET OF THINGS

1) DEFINITION OF IoTs

In 1999, Kevin Ashton introduced the concept of the Internet of Things (IoT) in his presentation [69]. At that time his idea was about linking the radio-frequency identification (RFID) in P&G's supply chain. However, with the development of technology and the widespread use of smart devices, ''things'' became almost everything, not just RFID. Ziegeldorf *et al.* [8] insist on the evolution order of IoT technologies from RFID, to wireless sensor network (WSN), to smartphones. Also, many IoT devices such as smartwatches, eHealth devices, smart home devices, and smart cams are widely used. After more than a decade, Haller *et al.*'s definitions did not change much. Haller *et al.* [70] defined IoT as, ''a world where physical objects are seamlessly integrated into the information network, and where the physical objects can become active participants in business processes. Services are available to interact with these smart objects over the Internet, query their state and any information associated with them, taking into account security and privacy issues''. Zhang *et al.* [11] expanded the physical object to the virtual object so their definition of ''thing'' in IoT is ''a physical or virtual object which connects to the Internet and has the ability to communicate with human users or other objects''.

2) GENERAL STRUCTURE OF IoTs

Security researchers generally categorized the IoT into several levels. Suo *et al.* [71] divided the IoT into four levels consisting of perceptual, network, support, and application layers. The *perceptual layer* is responsible for collecting information using physical sensors which include radio frequency identification (RFID) readers, temperature sensors, and cameras. The perceptual layer physical devices are generally small-sized mobile devices and are mainly powered by batteries, thus limiting the consumption of resources. The *network layer* reliably transmits the information of the perceptual layer. Networks for transmission are mainly wireless networks, and unlike the general IT environment that mainly uses WiFi, heterogeneous wireless communication methods like ZigBee, LTE, 5G and Bluetooth are used. The next level is the *support layer* which provides a stable support platform for the application layer. In this layer, many types of intelligent computing power are configured through the network grid or cloud computing. From this layer, computing resource limitation due to limited power support is eliminated. The last layer is the *application layer*. The application layer oversees interfacing with the user through the Internet and can provide personalized services according to the needs of the users. In this layer, protocols such as HTTP, CoAP, MQTT, XMPP, DDS, and AMQP are used. Swamy *et al.*'s [72] architecture is made up of three layers which are the perception, network, and application layers. In this case, there is no support layer, but it esentially has the same concept as the four layers.

On the other hand, Nurse *et al.* [16] divided the IoT into three layers: hardware, middleware, and application. But the three layers are also not very different from the four layers in concept. This is because the network layer of the four layers exists between the hardware and middleware of the three layers. That is, the hardware layer includes a perceptual layer and hardware side network layers, and the middleware includes middleware side network layers and a support layer. These three and four layers of architectures are depicted in Figure 1.

3) IoT FROM THE INSIDER THREAT PERSPECTIVE

In this chapter, we will examine the security features of IoT and evaluate the existing ITD techniques using the features presented in the next section.

a: HETEROGENEITY

The first characteristic of the IoT is heterogeneity. In existing IT environments, Windows, Unix/Linux, and TCP/IP are mostly used. But in the IoT world, there exists diverse operating systems and networks. For the operating systems (OS), we can easily find various OSes such as Contiki, Android things, Riot, Apache Mynewt, and Huawei LightOS. And for the network, in addition to TCP/IP, IoT-specific protocols such as 6LoWPAN, RPL, CoAP, MQTT, XMPP, DDS, and AMQP are used. This heterogeneity makes detection more challenging to the defender because of the broadened attack surface. For example, if connected IoT devices use ten types of operating systems, there is much more than ten times the vulnerability in the environment. And as defensive positions, they must have knowledge of at least ten environments and use ten times more resources when applying defense strategies. Moreover, considering that the software used in these various environments has not been fully validated, it can be a disastrous situation when the software of IoT is exploited. For this reason, many studies [7], [11], [16], [73], [74] have addressed the complexity of security incident detection due to the heterogeneity of IoT.

b: RESOURCE-CONSTRAINT

The second characteristic of the IoT is resource-constraint. IoT devices are usually operated by batteries. As powerful CPUs consume more battery power, IoT devices typically use lower power CPUs. To achieve long battery life of end devices, complex schemes or services that consume much battery are avoided. For this reason, the IoTs are characterized by resource-constraint. Especially in the perceptual layer, due to resource limitations, robust encryption cannot be used, and installation of agents is avoided [11], [75] and the use of firewalls on each end device [76] are also inefficient. For insider threats detection host-based are more suitable than network-based [77], but due to resource-constraints using the host-based method is restricted. The lack of user interfaces and interaction which can pose security threats are also resource-constraint features [78].

c: MOBILITY

The third characteristic of the IoT is mobility. IoT devices are highly mobile and can be located anywhere. This feature, combined with the increasingly smaller nature of IoT devices, makes detection more challengeable. These IoT devices can establish temporal networks instead of connecting to the fixed network, making them to bypass existing network-based detection techniques. Nurse *et al.* [16] presented attack vectors which reflects mobility characteristics. The attack vectors include 1) unauthorized video recording for should-surfing attack, 2) taking a photo or video of sensitive data or IP, 3) unauthorized audio recording of a private conversation or meeting, 4) unauthorized copy of sensitive data, 5) direct scans of sensitive items, 6) using the malware-infected IoT devices to comprise enterprise networks, 7) installation of hardware-based backdoors, and 8) installation of network analysis devices. In addition, the scattered nature of the device also means that insiders can easily access the device for privilege escalation.

IV. DATA SOURCES IN RESEARCH

A. PUBLIC DATASETS IN LITERATURE

The data plays a significant role in ITD 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 ITD researchers. Public datasets for ITD are usually composed of normal data and synthesized anomaly data. In this section, we will look at some of the significant datasets used in research ITD.

1) LINCOLN LABORATORY INTRUSION DETECTION DATASET

This dataset was synthesized and recorded on a network that simulated a United States Air Force facility network connected to the Internet by MIT Lincoln Laboratory to evaluate the DARPA intrusion detection system [79]. The simulated network is composed of more than 50 computers, and the dataset contains 32 types of attacks. The system logs were collected through the Basic Security Model (BSM) auditing program, each installed in a simulated environment. Each log consists of tokens that contain system information such as system call, date, time of execution, executing process, arguments, user ID, and group ID. The performed attacks were 1) denial of service (11 attacks), 2) user to root (7 attacks), 3) remote to user (9 attacks), and 4) probes (5 attacks). Parveen *et al.* [34] researched the malicious insider detection method using this dataset.

2) RUU DATASET

The RUU dataset [80] is a masquerader based dataset obtained through an experiment designed to collect realistic masquerader behavior logs [81]. Thirty-four normal users and 14 masqueraders generate the dataset. To obtain masquerader data, Salem and Stolfo [81] devised a ''capture the flag'' exercise, and 14 volunteers served as masqueraders. They were asked to perform masquerader activities such as finding information that could be used for financial gain on the normal user's computer with unlimited methods. For log collection, the Windows host sensor was installed and collected host-based information such as file system access, processes use, Windows registry, and dynamic library loading.

3) ENRON EMAIL DATASET

The Enron email dataset [82] contains 0.5M email messages from about 150 employees, mostly senior management of Enron company. Due to the request from affected employees, all attachments were removed, and some messages have been deleted. This dataset is a resource for email related researchers, but can also be used in an insider detection study. Homoliak *et al.* cited that this dataset can be used especially for email text analysis and social network analysis aimed at the detection of insider threats involving collaborating traitors [60]. Eberle and Holder [45] used this dataset in their study to find the principle actors of the Enron scandal.

4) VAST DATASET

The VAST dataset is provided by the IEEE Symposium on Visual Analytics Science and Technology (VAST). The IEEE Symposium on VAST provided a dataset which contains Wiki editors, migrant boats, cell phone calls, and evacuation traces in 2008. The cell phone calls data is a set of phone call records from Isla Del Sueño over a ten days period in 2006. Through these cell phone calls, data readers can extract critical information about the Catalano social network structure [83]. With this dataset, Eberle and Holder detected the inherent leading social network and anomalies in the network [45].

5) SCHONLAU DATASET

Schonlau *et al.* [19] introduced a truncated user command dataset, commonly called the Schonlau dataset or Schonlau Et Al. (SEA) dataset [84]. This dataset is a masquerader based dataset and 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 user (the other 20 users simulate masquerade activities). The first 5,000 commands for each user contain clean commands, such as training data, and the rest comprises of masquerades' data with a 5% probability. The Schonlau dataset is just sequences of Unix commands and user names with no other information, such as flag, aliases, timestamp, argument, or shell grammar [18], [85], which causes some limitations. Maxion and Townsend [86] pointed out why the Schonlau dataset is not suitable for masquerade detection for reasons such as the data not being sequential and not being clear whether commands are typed by a human or script, and suggested some ideas to improve the dataset (1v49) [87].

6) GREENBERG'S DATASET

Greenberg's dataset [88] is an authentication-based dataset and contains 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. Greenberg's dataset was enriched with information on session start & end time, alias, the current working directory of the users, history use, and error status. Maxion [18] conducted research on detecting malicious insiders using this dataset.

7) BALABIT MOUSE DYNAMICS CHALLENGE DATASET

The Balabit mouse dynamics challenge dataset [89] includes timing and positioning information of mouse pointers of 10 users in training_files folders. The purpose of the Balabit mouse dynamics challenge is protecting a set of users from the unauthorized usage of their accounts [89], and this dataset is used in the challenge. The dataset fields include record timestamp, client timestamp, button, state, and x & y coordination. Hu *et al.* [25] used this dataset in their study to show mouse biobehavioral features for ITD works.

8) CERT DATASET

The CERT Insider Threat Test Dataset [90] is a synthetic dataset that includes system logs with annotations of insider threat activity. It is the de facto standard dataset in the ITD domain for several studies [23], [29], [30], [32], [38], [39], [53], [55]. The CERT dataset contains more and more data from r1 to r6.2, but the latest version includes the previous version of superset. The CERT dataset is synthesized

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 job-searching websites, and stealing data using a removable drive before leaving the job. Two insiders are corresponding to each scenario, and the overall number of malicious insiders is 10. This dataset contains PC on/off logs, removable drive logs, website access logs, email transceiver logs, removable drive activity logs, and employee information in LDAP, and psychometric information based on the five-factor model [91].

B. PRIVATE DATASETS IN LITERATURE

Public datasets have an advantage in that they can easily use proven datasets for research. However, sometimes it is necessary to create a new dataset for their research. In this case, researchers use the data from their company or a specific company or from the test bed to collect the necessary data. Among these datasets, those that are not disclosed for some reason, such as containing sensitive information are called private datasets. Since it may be necessary to create a new dataset considering the IoT environment in conducting ITD studies, in this section, we analyze the private datasets used in the surveyed papers based on the data sources and features they used.

1) COMPUTER USAGE ACTIVITIES LOG

Computer usage activities logs are the most frequently used data source in our surveyed literature. Public datasets introduced in Section IV-A also fall into the computer usage activities log category except for the Enron Email dataset and the VAST dataset. Three papers [20], [33], [43] we surveyed use private computer usage activities log datasets. Ted *et al.* [20] collected computer usage activity in the business organization of about 5,500 people through a commercial tool called SureView. Installed SureView can capture user actions such as logins, file accesses, emails, instant messages, printer usage, browser usage, process usage, etc. Malicious behaviors are conducted by an independent expert red team based on known insider attack cases that include 1) destruction, 2) misuse or corruption, and 3) theft. 111 features of seven types are identified by a retired expert from the U.S. intelligence community. Seven types encompass email, file, group, login, printer, URL, and ratio. Eldardiry *et al.* [33] also conduct their experiment with a private dataset collected from a real company environment. Captured user actions are login/logoff, removable device usage, file access events, browser usage, and email with tags such as user id, host PC id, activity code, timestamp, after normal working hours, and by PC owner. Malicious behaviors are injected based on real malicious behavior accidents. Gavai *et al.* [43] also gather employee activities log from a single business unit of a large organization. Collected logs contain emails, application logs, login information, business unit hierarchy, etc. and realistic malicious actions are injected by a red team.

2) RESULT TUPLES OF RDBMS

Mathew *et al.* [24] suggest the technique to distinguish between normal and abnormal access patterns for the Relational Database Management System (RDBMS). Their idea is based on the fact that insider attacks against RDBMS are dangerous and tricky to detect. To distinguish abnormal access pattern, they take notice of semantics of the queries that are more suitable than their syntax. They devised a method to calculate the statistical summary of the result tuples for each query to analyze the semantics of queries. Their method computes the result tuples into the query summary vector that is called the S-Vector. S-vectors are obtained by calculating min, max, mean, median, and standard deviation for numeric attributes and using the non-numeric attribute to find the total count and the number of distinct values. For the anomalies they experiment with three types of anomaly cases: 1) different schema & different results, 2) similar schema & different results, and 3) similar schema & similar results.

3) NETWORK TRAFFIC

There are two pieces of literature which capture network traffic packets in their experiments and use captured packets as a dataset. Mayhew *et al.* [35] explains their mal-behavior analyzing system the Behavior-Based Access Control (BBAC). BBAC has analyzed the ability for mal-behavior through network connections, HTTP requests, text exchanges through emails or chat messages, and edit sequences to documents. Network connections and HTTP request logs are network traffic-related and they are from Bro [92] which characterizes IP network flow and individual HTTP requests and responses. For network connections, TCP connection log information associated with network flows including the total number of connections, size of traffic sent over the connections, and average duration of connections per host time indexed by day and hour-of-day are used. For HTTP request logs, HTTP headers associated features and WHOIS [93] related information including queries that are cached in a local database, whether the machine uses a DHCP IP address, and the country for the URL host are processed. Meidan *et al.* [28] conducted IoT botnet attack detection experiments. They set up a testbed that composed of IoT devices and Wi-Fi access points, and a wire connected switch. In their testbed, they sniffed the network traffic using Wireshark through port mirroring. To take a behavioral snapshot, they collected 23 traffic statistics over each 5-time window to summarize all the traffic. Time windows reflect the most recent 100 ms, 500 ms, 1.5 sec, 10 sec, and 1min. 23 features were embraced follows:

1) out bound packet size(mean, variance) aggregated by source IP, source MAC-IP, channel, socket
2) packet count aggregated by source IP, source MAC-IP, channel, socket

3) packet jitter(mean, variance, number) aggregated by channel
4) inbound/outbound packet size(magnitude, radius, covariance, correlation coefficient) aggregated by channel, socket

4) EHR SYSTEM ACCESS LOG

A Community-based Anomaly Detection System (CADS) introduced by Chen and Malin evaluate the anomaly detection models with a private dataset of real electronic health record (EHR) access logs from a Vanderbilt University medical center [21]. The EHR contains an electronic patient chart of 1.5 million patients, and for the research, a six months' duration of access logs were analyzed. The synthesized malicious behavior data was inserted by simulated users with the assumption that an anomalous user would not exhibit steady behavior.

5) EMAIL CONTENT

Email contents sent and received between people are popular data sources for ITD. The aforementioned Enron Email dataset and CERT dataset also include email contents. Taylor *et al.* [22] also conducted research using an email dataset for detecting insider threats through language change. To create their own private dataset, they designed the simulation program known as Confidential Operations Simulation (iCOS) which simulates the investigative tasks and organizational environment of a police investigation into organized crime. The iCOS provides an environment where information could only be communicated between teams through e-mail and print outs, thereby e-mail contents between participants could be obtained. In order to simulate an insider, they asked randomly selected players to provide additional information to a provocateur for an additional reward.

6) SENSITIVE INFORMATION ACCESS LOG

Oh *et al.* [31] develop a model to detect insider threats in an organization with certain business patterns. For the experiment they use a public institution's four years user-specific sensitive information access log with access time, access IP, access method, and access ID. The access method is composed of download, print, and view. To identify annual business process characteristics the action method's average and standard deviation are also used.

7) ORDER-PROCESSING DATA

There are studies trying to model normal behavior by using a graph composed of vertices and relationships and use this model as an ITD study. Eberle and Holder suggest graph-based approaches for ITD. To evaluate the approaches they use the Enron dataset, VAST dataset, and private dataset [45]. The private dataset is simulated using the public-domain discrete event simulator OMNeT++ and the simple order-processing model which contains a sales department, a customer, and a warehouse as vertices and relationships as order, delivery note, or order ACK.

8) BRAIN WAVE DATA

Some studies have used brain waves through Electroencephalography (EGG) signals instead of using a host-based or network-based dataset. Hashem *et al.* [42] utilize EGG signals for their study using a consumer-grade Brain-Computer Interface (BCI) device. Fourteen sensors of the BCI device are used to record different brain waves and recorded signals are decomposed into different frequency subsets using the Wavelet Packet Decomposition (WPD) method. Other features such as Microvolts (μV) mean value, maximum μV, minimum μV, number of peaks, the distance between the high and low μV, from each five-second time frame were also extracted.

9) AUTHENTICATE LOG

Ene *et al.* [94] used user profiles (Americas small and Americas large) obtained from Cisco access control firewalls for HP's external business partners connected to the HP network. Kaghazgaran and Takabi [54] also used the Americas large dataset for their experiment to assess role base access control extended system. The dataset is private and does not have much information. However, according to Kaghazgaran and Takabi's paper, the dataset contains 404 roles, 3,485 users, 10,127 permissions, 3,965 user assignment relations, 85,508 permission assignment relations, and the number of role hierarchy relations is 266 [54].

V. DATASET AND DETECTION APPROACH ANALYSIS WITH IoT PERSPECTIVE

In this section, we classify the papers we surveyed based on which layer of the IoT environment the data used in the study. Then, we classified the data of each layer according to its characteristics, how did the researchers use the collected data for the ITD, and what should be considered in collecting and using such data in the IoT environment. Besides, in a separate subsection, we present an analysis of other approaches. Table 2 shows the summarized data sources of the surveyed literature.

A. PERCEPTUAL LAYER

1) USER COMMAND BASED

Among the datasets used in the ITD research, the Schonlau dataset [84] and Greenberg dataset [88] use user commands, which is data available from the perceptual layer. User commands are useful for profiling user behavior because they are a collection of commands that the user enters directly into the system. However, only using the user command alone, it is hard to know the feedback of the system. Therefore the user command is mainly used for masquerade detection research, which is the method to detect the genuineness of the current user through the change of user behavior.

TABLE 2. Analysis of data sources in survey literatures.

Data Source (Literature)	Perceptual	Network	Application
Unix commands (Greenberg) [18]	✓		
Unix commands (Schonlau) [19]	✓		
Windows logs (RUU) [81]	✓		
Combined (CERT) [23], [29], [30], [32], [38], [39], [53], [55]	✓		
Email messages (Taylor) [22]	✓		
Working logs (Vegas) [43]	✓		
Working logs (Ted) [20]	✓		
Working logs (Eldardiry) [33]	✓		
Event logs (Lincoln) [34]	✓	✓	
Network traffic (Meidan) [28]		✓	
Working logs (Mayhew) [35]		✓	✓
Game play data (WoW Census) [40]			✓
Mouse behavior (Ballabit) [25]			✓
Email message (Enron) [39], [45]			✓
Combined (VAST) [45]			✓
Twitter tweets (Sentiment140) [26]			✓
Youtube (Kandias) [41]			✓
SQL result tuples (Mathew) [24]			✓
EEG signals (Hashem) [42]			✓
EEG signals (Almehmadi) [50]			✓
Information access (Oh) [31]			✓
EHR access log (Chen) [21]			✓
Order-processing data (Eberle) [45]			✓
Eye-tracking data (Matthews) [27]			✓
Authenticate log (Kaghazgaran) [54]			✓
Theoretical studies [47]–[49], [51], [52]			

The Schonlau dataset [84] is a simple data set of Unix csh commands for 50 users. The dataset consists of the 1) user number and 2) each users' truncated commands. About 5% of the test data contain masquerade commands, and because of this, several pieces of research [19], [86], [95] use the dataset to benchmark their masquerader detection performance.

The Greenberg dataset [88] is a dataset that collects 168 user's Unix csh commands. Unlike the Schonlau dataset, which contains only truncated commands, the Greenberg dataset is comprised of full command-line entries. Each trace file in the Greenberg dataset consists of seven entries as follows.

1) S: Session starting time
2) E: Session end time
3) C: User entered command line
4) D: Working directory path
5) A: The alias the command line invoked
6) H: Whether a history was used
7) X: Whether an error has occurred

Maxion [18] uses the Greenberg dataset to compare how accurate the truncated and enriched commands would be and shows that masquerade detection improves the hit rate from 70.9% to 82.1% when the enriched commands are used. In the experiment, Maxion uses only C and A of the seven entries in the Greenberg dataset and ignores the rest to focus on profiling user behavior.

```
header,134,2,ioctl(2),,Fri Jan 23 17:05:14 1998, + 20011000 msec
path,/etc/mnttab
attribute,100644,root,root,8388632,11027,0
argument,2,0x5401,cmd
argument,3,0xeffffa2c,arg
subject,2503,root,other,2503,other,5809,5807,24 0 192.168.1.30
return,failure: Inappropriate ioctl for device,-1
trailer,134
```

FIGURE 2. A sample record from Lincoln dataset.

2) SYSTEM BEHAVIOR BASED

The Lincoln dataset [96] and RUU dataset [80] are datasets that collects data in terms of system behavior. While user command-based datasets collect the commands entered by the user in sequential order, the datasets in this section provide rich information, including timestamped user commands and the resulting system events.

The Lincoln dataset [96] provides separate list files of audit data and raw packet data. For audit data, Solaris BSM software was used, and for raw packet data, tcpdump was used. We do not cover raw packet data in this section because it is not data obtained from the perceptual layer. The BSM logs are organized in a format consisting of a head beginning and a trailer ending to represent all system calls invoked by all processes.

Figure 2 shows the format of the BSM logs. The header line reports the record length in bytes, an audit record structure version, the system call (event ID), and the time and date the record was created in millisecond resolution. The second line reports an absolute path of the process. The third line reports the file access mode, the user ID, the owner group ID, tile file system ID, inode ID, and device ID. A path token usually accompanies the attribute token. The fourth and fifth lines report the system call argument ID, the argument value, and an optional text string. The subject line reports the audit ID, effective user and group ID, real user and group ID, process ID, session ID, and terminal device and machine ID. Finally, the last line reports the total number of audit record characters.

Parveen *et al.* [34] tested their algorithm with audit data of the Linconln dataset. They filtered the dataset audit data by user-affiliated system calls (e.g., exec, execve, utime, login, logout, su, rsh, rexecd, passwd, rexd, and ftp) because their intention was an insider threat, and these system calls correspond to logging in/out or file operations performed by users. Their result shows that with unsupervised graph-based detection algorithms, low false negatives (up to zero) and relatively high false positives (up to 42%) could be achieved.

The RUU dataset [80] provides Windows and Linux monitoring data. Salem and Stolfo [95] use Windows and Linux supporting host sensors to collect system behavior and transmit collected information to the data collection server. The Windows sensor could monitor all registry-based activity,

TABLE 3. Mutl domain business data based datasets.

Dataset	email	logon	web	file	device	printer
CERT [90]	✓	✓	✓	✓	✓	
Eldardiry [33]	✓	✓	✓	✓	✓	
Vegas [43]	✓	✓	✓			
Ted [20]	✓	✓	✓	✓		✓
Taylor [22]	✓					

processes activity, GUI access, and DLL libraries activity through low-level system drivers. The Linux sensor could collect all process IDs, process names, and process command arguments using *auditd* daemon. This dataset is a public dataset, but unfortunately, the link to the dataset [80] is now forbidden, so the detailed data structure is inevitably found in the paper [95].

The authors show that using the RUU dataset and one-class SVM with context features, they achieved a 100% detection rate with a very low false-positive rate (0.1%), and an average AUC score of 0.996. Context features refer to information about previous user events.

3) MULTI DOMAIN BUSINESS DATA BASED

The user command-based and system behavior based datasets are those that consider the direct interface between the user and system, and system behavior upon interaction. Several studies [18], [19], [29], [34], [53], [86], [95] have shown that such data is useful for ITD. However, the ITD field depends highly on the characteristics of human behavior, so the data need to model the micro and macro effects of human behavior [97]. The demand for modeling realistic human behavior had led to the need for the dataset that represents a business organizations' interactions with realism in a large number of dimensions.

Among such multi domain business data-based datasets, CERT [90], Eldardiry [33], Vegas [43], Ted [20], and Taylor [22] are datasets containing perceptual layer data. These datasets include similar domains such as email, login, web, file, and device activities because they assume a typical organizational business environment. Table 3 shows what business domain data each dataset contains. Some of these domains (e.g., email, logon, and web) can also collect data through the application layer. However, we considered them as perceptual layers because more detailed data can be collected at the perceptual layer.

Among the domains, email is information of emails sent and received by users. Email domain has attributes of date, subject, to, cc, bcc, from, content, and attachments in most datasets, and the action record of read, send, and view. Logon domain contains data of user login and logout with information of user, PC, and date. For web activities, information of date, user, PC, URL, contents, and browser information could be collected. Web activities could be upload, download, visit, or categories of visiting sites (e.g., career site, web mail, entertainment, and social media). File domain includes date, user, PC, filename, removable media related, contents, and activities (e.g., open, write, copy, rename, and delete). Device domain involves removable media-related activities, such as USB thumb drive or portable hard disk connection and can collect data of file trees or activities of connection and disconnection. Printer domain comprises print job submitted-related data.

The use of multi-domain data enables elaborate user behavior analysis. Researchers have devised masquerader detection when using the single-domain dataset (user command or system behavior). However, with the multi-domain studies have been attempted to detect traitors (i.e., Hiding Undue Affluenc scenario [43]) in addition to masquerader detection. Studies usually utilize data of each domain to create a user activity profile and use it to detect user anomaly. Among the studies we surveyed, some studies [20], [23], [30], [32], [33], [53], [55], [81] attempted to detect insiders by creating user activity profiles. In addition to the user activity profiles, some studies [22], [39], [43] also try to use text mining for domains with text data (e.g., email, web, and file) to detect user's sentiment state or writing style change.

4) CONSIDERATION IN AN IoT ENVIRONMENT

The data from the perceptual layer is beneficial to observe the behavior of the system directly. The ITD system could utilize this data such as login, file access, device connection, audit data, or user commands to detect system anomalies promptly. In the perceptual layer, the ITD system could collect data required for detection and execute detection algorithms using such data. However, in the IoT environment, executing a detection algorithm in the perceptual layer is not a suitable choice due to the unique nature of IoT devices of limited capabilities (processing, power, and memory constraints) [8], [16]. Because security mechanisms require a fair amount of computation power, this can affect the devices' performance adversely [11].

Therefore, in the IoT environment, transmitting collected data to a dedicated detection system and executing detection algorithms in the system is realistic. For this approach, the operating system or third-party software should provide data collecting and transmitting functionality. In this paper, we call this operating system functionality or third-party software a ''host-agent''.

In the typical IT environment, the organization can manage devices inside of the organization from a security point of view. Organizations can enforce security policies to install specific security software or to configure initiating OS auditing services for connecting to the network. If the device could not connect to the network, it would not pose a threat to security. However, in the IoT environment, devices can connect to their network, including the Internet, without the organization's permission. This means that there are devices without host-agents that could pose a security threat to the organization, and ITD systems could not utilize that data.

Even though the organization can enforce installation of a host-agent through a strict security policy, the problem remains. IoT devices that are small in size or embedded in

TABLE 4. Analysis of network layer data.

Data Source	Observing Activities
TCP/IP dump	
Lincoln, Meidan, Mayhen	packet dump

other devices can possibly bypass the security gate, and they can come into the organization without a host-agent. Even if security guards can detect all IoT devices, it is almost impossible to develop and install host-agents corresponding to heterogeneous IoT devices.

As described above, when applying the ITD in the IoT environment, the insider could bypass sending perceptual layer data to the ITD system and hide from the system. For this reason, when conducting IDT research in the IoT environment, it is essential to keep in mind that perceptual layer data alone does not achieve the desired purpose.

As we mentioned in previous sections, existing studies used datasets of user commands (Section V-A1), system behavior (Section V-A2), or multi-domain business data (Section V-A3) as data sources from the perceptual layer. Among these, user command based and system behavior based contain only system behavior information and have no high dimensional data for related activities. Thus, related literature mainly focuses on detecting masqueraders and has difficulty detecting insiders who have privileges (traitors). Existing studies solved this difficulty by using a multi-domain business data-based dataset, but additional consideration is needed in the IoT environment.

Existing studies assumed a PC-based work environment, so most of the data were collected from business-specific domains such as email, web, devices, files, and printers. However, IoT devices can use different domains from these domains, so it is not clear whether they will be equally effective. For example, if an insider attacks a smart sensor in a power plant and alters the sensed value of the device, this data does not belong to an existing business domain. Therefore, when conducting ITD research in the IoT environment, it is necessary to analyze the domain and select the data source for each environment.

B. NETWORK LAYER

Network traffic is a valuable data source in the security field because it allows the development of a threats and anomalies detection mechanisms [98]. The same applies to ITD, so among our surveyed literature, three datasets contain the TCP/IP packet dump from the network layer. Table 4 depicts the datasets using the network layer data. The Lincoln dataset has outbound and inbound packet dump separately but in the experiment Parveen *et al.* [36] did not use the network data. Meidan *et al.* [28] conducted their experiment using MAC, IP, and TCP related protocol information. Mayhew *et al.* [35] also attempted ITD using application level protocols (e.g., HTTP, XMPP, and SMTP) in addition to MAC, IP and TCP.

1) NETWORK PACKET CAPTURE BASED

The Lincoln dataset [96] contains outbound and inbound network dumps as a pcap file per each session using tcpdump. Median *et al.* [28] also collected the raw network traffic data as a pcap file using port mirroring on the switch. Median *et al.* extracted 23 features for each five-time windows (100 ms, 500 ms, 1.5 sec, 10 sec, and 1 min) to summarize the traffic group by originating from the same IP (network layer), from the same MAC (data link layer), per network layer channel (same source and destination IP), and per transport layer (same source and destination TCP/UDP sockets). The experiment uses network data up to level 4 (transport layer) among OSI 7 layers (Stateful Packet Inspection). It shows that they can detect BASHLITE Attacks and Mirai Attacks that are characterized by scanning and flooding (UDP, TCP, Ack, Syn). However, the author did not utilize the application layer of the OSI 7 layer, so the detection model did not detect the behavior of the context from the business side, but only detected the attack.

Mayhew *et al.* [35] extended their detection scope to application level protocols (e.g., HTTP, XMPP, and SMTP), and this method is called Deep Packet Inspection. The authors take advantage of the connection behavior of IP network flows through TCP connection log information, and they also extended the scope by analyzing HTTP request behavior using HTTP. As a result, they achieved 99.6 % of true-positive-rate with a 0.9% false-positive rate by using an individual word vector for each string feature and including the WHOIS data features for detecting malicious URLs.

2) CONSIDERATION IN AN IoT ENVIRONMENT

In IoT environments, mobility characteristics should be considered when detecting insider threats with network layer data. However, the datasets used in the surveyed papers assume a fixed network architecture. The fixed network architecture could more efficiently manage the connection of computers to the network through operational controls or technical controls than the IoT network. In the IoT environment, the topology and the nodes can be changed continuously. In a wireless sensor network environment, if one routing node is turned off due to low battery, other sensors should organize new network topology. A smart device can also establish a network for a short time on one nearby device and then connect to another device to form a new network. This means that the members of the network could be continuously changed, and subnetworks could be continually created and destroyed. Figure 3 depicts these IoT network architectures.

This variability network makes it challenging to identify and track the devices connecting to the network and determine its users. In the situation when an insider detection system is monitoring TCP/IP packet through network switch tapping, it is hard to observe the Bluetooth network between a smartwatch and a smartphone. In the IoT world, there are diverse network methods such as WiFi, Bluetooth, and

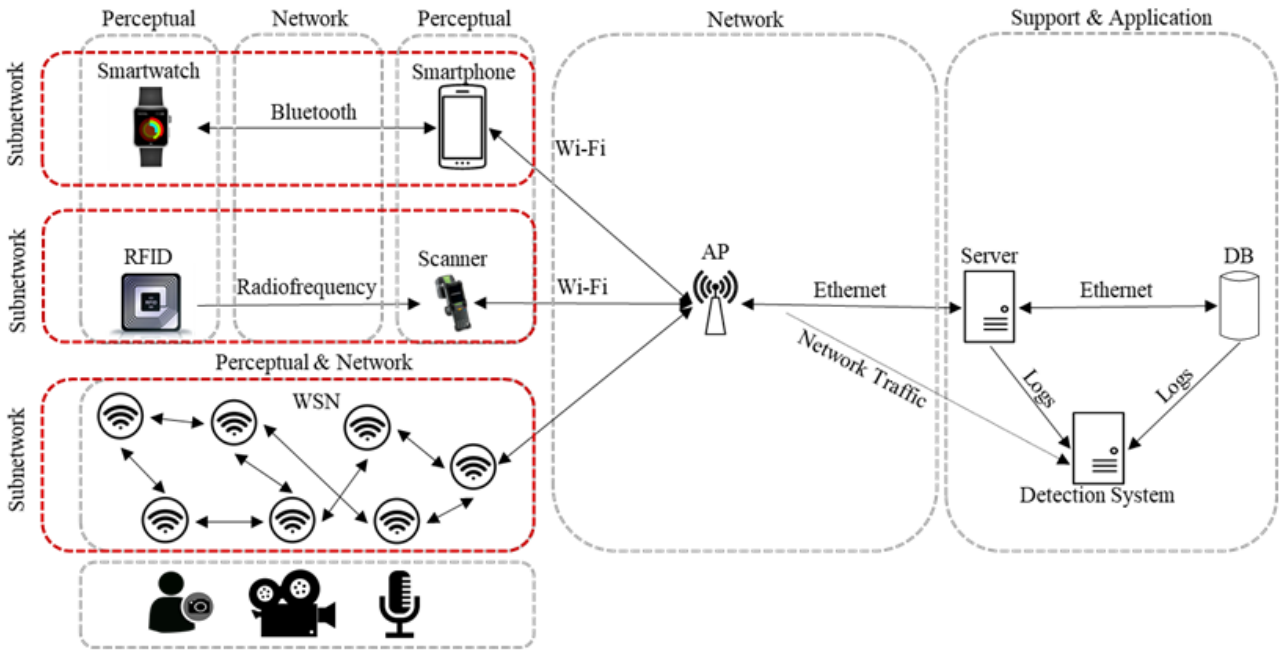

FIGURE 3. IoT network with temporal subnetwork.

Near-Field Communication (NFC), so only capturing TCP/IP packets via port mirroring on the network switch may not be enough. This situation means that there is a network beyond the TCP/IP network in the IoT environment.

C. APPLICATION LAYER

The IoT architecture described in Section III-B2 is composed of four layers, but in this paper, we consider the support layer combined with the application layer from the perspective of the data collection target since the support layer plays a supporting role for the service. The application layer oversees interfacing with the user through the Internet and can provide personalized services according to the needs of the users. In terms of analyzing existing datasets, the application layer is considered the server side where the information is gathered and provides a service.

Among 20 datasets, 11 datasets use the application layer data. The data of this layer may include data of various values and types according to the structure and purpose of the system. In the CERT dataset [90], the application layer data includes the website access log, email transceiver log, and employee information in LDAP. In the Enron dataset [82] and email messages [22], the emails are the data of this layer, and in the VAST dataset [83], the phone call records are the data. In the research about sentiment140 [26], YouTube case [41], and WoW Census case [40], they use open-source data such as social network service (SNS) or gameplay data (WoW Census) to identify sentiment of users proactively. Access logs are used for sensitive information access [31] and EHR access logs [21]. Result tuples [24], which are datasets for database semantics, and electroencephalography (EGG) signals [42], [50], which are brain wave related data, were also classified as application layer data.

1) DATABASE QUERY RESULT BASED

Mathew *et al.* [24] propose a data-centric approach to model user access patterns by profiling the data points that users access. To do this, the authors attempted to obtain an access pattern that describes what the user is trying to access using an S-vector that represents a statistical measurement of the result tuple of the SQL query used by the RDBMS. S-vectors are obtained by calculating min, max, mean, median, and standard deviation for numeric attributes and using the non-numeric attribute to find the total count and the number of distinct values. The authors trained machine learning algorithms (Naive Bayes, Decision Tree, SVM, and Clustering) with inputs of the query results' statistics on the real Graduate Admission database, user name, and user-role. As a result, the authors got a nearly 10% improved result compared to the syntax-centric approach by Kamra *et al.* [99].

2) SENTIMENTAL BASED

The researches using YouTube [41], Sentiment140 [26], and WoW Census [40] have the goal of detecting changes in the user's attitude or sentiment, rather than detecting malicious behavior.

Kandias *et al.* [100] used data from the video streaming web service, YouTube. The authors gathered user related (profile, uploaded videos, subscriptions, favorite videos, and playlists), video related (license, # of likes, # of dislikes, category, and tags), and comment related (comment, # of likes, # of dislikes) data from the YouTube service. Then they showed that using machine learning or a dictionary-based

classification algorithm they could classify each video's comments into a positive and negative attitude. They had an assumption that malevolent insiders have a close relation to the psychosocial trait of negative attitudes towards law enforcement and authorities in their study. With the assumption, we can use this data to track malicious intent.

Park *et al.* [26] utilized the Sentiment140 dataset with 1.6 million tweets [101]. The Sentiment140 dataset contains user ID, date, tweets, and sentiment of tweets in the.csv file. The authors trained unsupervised learning (Naive Bayes, SVM, Linear, Decision Tree) and supervised learning (K-Means, EM) algorithms with the dataset and calculated the sentiment scores. They achieved the highest accuracy when using a Decision Tree for detecting possible malicious insiders with the highest accuracy of 99.7%.

Brdiczka *et al.* [40] used the public online game play dataset (WoWCensus) [102]. The authors proposed a more sophisticated model than using text mining to classify emotional states. Their sophisticated model uses behavior, text analysis, and social network data to predict personality. For behavioral analysis, the authors used behavioral data such as milestone achievement, type of death, and character skill from the World of Warcraft online game playing dataset. To analyze text, the authors choose features from the names (character, guild, role, race, actions) and chat messages. Finally, to uncover social network use, the authors analyzed friendship and membership networks. Through the combination of three analysis, they could obtain results of anomalous behavior (i.e., guild quitting) possibly detected through structural analysis of social networks in the game, and a player's personality could be captured using behavioral and text analysis.

Greitzer *et al.* [47] tried to combine traditional cybersecurity audit data and psychosocial data in their research. In the research, they presented five legal/privacy ethical free data source for assessing psychosocial factors to identify candidate insiders. These are 360 Profiler and other tools that are used in staff performance evaluations, competency tracking, disciplinary tracking, timecard records, proximity card records, and pre-employment background checks. They also gave the list of should not use for monitoring insider threats as follows: arrest records, use of Employee Assistance Program (e.g., for family counseling), use of Employee Complaint Mechanism, life events (such as marriage, divorce, births, or deaths in family), and health events (medical records).

Maasberg *et al.* [49] proposed a theoretical model based on the Theory of Planned Behavior and the Capability, Motive, and Opportunity (CMO) model. The study tries to find the relationship between insider threat and malicious intent. Then they would like to explain Dark Triad personality traits and the insider threat with the relationship.

3) RELATION BASED

Among the surveyed literature, the Enron email dataset [82], VAST dataset [83], Order process dataset [45], and EHR access log [21] are data providing relations between actors. These datasets provide overall relations between users or devices rather than using single device monitoring.

The Enron email dataset [82] contains0.5M of email messages of about 150 employees, mostly from senior management of the Enron company. Because the Enron email dataset is a dataset that collects the mail stored in the company's mail server, we classified it as an application layer, unlike the email data belonging to multi-domain business data (Section V-A3) which also includes user activity added to the email itself. Eberle and Holder [45] constructed relation graphs (i.e., vertices and edges) using the Enron email dataset [82], VAST dataset [83] and their own simulated order process dataset using the OMNeT++ public-domain discrete event simulator [103]. The authors argue that graph-based algorithms could overcome the drawback of classification algorithms that do not consider relational information. They used Graph-Based Anomaly Detection (GBAD), an unsupervised approach based on the SUBDUE graph-based knowledge discovery system [104], to construct a graph of the typical pattern (most prevalent substructure). Then, they show that the pattern different from the structure obtained by GBAD could be detected and the approach could be applied to the ITD.

Legg *et al.* [39] also used the Enron email dataset, but they did not use the relation-based method because they tried to identify psychological context by using the bag of words and Linguistic Inquiry Word Count (LIWC) [105].

The VAST dataset [83] consists of cellphone communication data. This data has attributes of from, to, date/time, duration, and cell tower. The aforementioned Eberle and Holder study [45] utilized the dataset and obtained the relational graph.

Kim *et al.* [51] proposed the ITD model using a graph-based approach to compute employee behavior. The proposed model is a set of independent graphs that represent each domain (e.g., subjects, email, files, and web) of activities. With a combined set, the model could compute an integrated behavior score.

4) PERSONAL IDENTIFICATION BASED

The Balabit mouse dynamics challenge dataset [89] contains timing and positioning information of mouse pointers. The Balabit dataset is suitable for user authentication or identification purpose based on mouse dynamics. User authentication could have a relation with masquerader detection from an insider threat viewpoint. The goal of both approaches is to determine whether the current user is a real user with the correct privileges. Hu *et al.* [25] researched the ITD approach based on mouse dynamics and deep learning and got a relatively high accurate result of a false acceptance rate of 2.94% and a false rejection rate of 2.28% within every seven seconds.

Mayhew *et al.* [35] gathered text data from Wiki, Twitter, and emails from their private system. They also conducted tests using data such as HTTP and TCP but also introduced

a method of personal identification using a method called stylometry. Using stylometry, the authors could measure various traits of a user's writing style from Twitter and email. Also, from MediaWiki [106], they could compute knowledge scores [107] for each user using features of identity information, document topic, and length of change. Through experiments, the authors had a true-positive-rate of over 93% in writing style detection for Twitter and Email, and a 76% true-positive-rate for the Wiki knowledge score method.

5) BRAINWAVE BASED

Almehmadi and El-Khatib [50] used the electroencephalography (EEG) response to visualize stimuli to demonstrate the possibility of using the user's intent as a means of access control. Physiological signals such as EEG are involuntary and have the advantage of hard to control the response by him/herself. The authors could detect the user's knowledge of intention with high accuracy by using P300 which is a 300ms delay positive peak in the brain signal. Since P300-based concealed information detection accuracy is reported at 90% to 100% [108], the authors explained that it was used in the experiment. For the experiment, the authors analyzed the P300 response by stimulating a participant in the presence of bad intention and motivation for a particular resource, such as showing pictures of malicious behavior. The result showed that verifying the intent of access using EGG is possible instead of identity.

Hashem *et al.* [42] constructed a dataset for their research with a consumer-grade EEG device that could record the brainwave signal from 14 different parts of the brain. The authors collected the EEG signal by recording a regular activity task (browsing, document work, email work), and a malicious activity task (data harvest, hacking) for each participant. The classification test using SVM showed that the authors could detect malicious behavior with 84% to 89% accuracy with the brainwave dataset.

6) ACTIVE INDICATOR BASED

The active indicator based detection method analyzes the feedback of the insider after stimulating a special significance to the insider (Active Indicator). Matthews *et al.* [27] experimented using a purpose-built simulation environment. Under the simulation environment, participants support intelligence in uncovering information about terrorist plots and installed eye-tracking devices monitoring participants for detecting illicit activity. At this time, active indicator probes such as showing critical cue on the screen were generated to identify espionage and attempted to catch insider by comparison with eye movement with normal circumstances. The experiment based on the assumption of lying is typically more complicated than telling the truth. Therefore the liar can expect to control and regulate verbal and non-verbal, so we anticipate detecting deception through these changes.

7) CONSIDERATION IN AN IoT ENVIRONMENT

Even in the IoT environment, IDT using the application layer is useful in that in order to harvest large amounts of data, it is necessary to connect to the service and to monitor the operation status of the service in its entirety. In particular, research using the database access pattern [24] is expected to be useful in the IoT environment in that all data is eventually stored in the database, and the data is extracted through a query.

Sentimental analysis has attracted much attention from researchers because it can proactively detect insider threats. The sentimental analysis is based on the assumption that the malicious insider shows early warning signs of mal-intent such as language style change, work progress, facial expressions, and so on [109]. Of course, as Homoliak *et al.* [60] mentioned, detecting a sentimental change is only for proactively preventing, and not evidence of whether the individual was involved in any malicious activity. However, using sentimental data is recommended in addition to cyber data to support the analysis [47]. Moreover, due to the nature of the ITD, which has a higher false-positive-rate than the rule-based method, it would be useful if the method could narrow the scope of the observing target through sentimental analysis. It also has the advantage of proactively finding insiders that are more likely to execute sophisticated attacks that cannot be found only by system behavior. However, for sentimental analysis, it is necessary to collect text-based information, which may cause legal or ethical issues [40], [41], [110], [111].

Graph-based detection has the advantage of applying to various types of data and considering relational characteristics. However, we could not find many studies in the ITD field. Furthermore, researchers should consider that if the malicious IoT device does not belong to the monitored network, graph-based approaches could not detect the insider threat, as mentioned in Section V-A4.

Personal identification based datasets are useful for detecting user characteristics rather than system behavior. In particular, the mouse is the most directly controlled device and is a suitable dataset for observing user behavior directly. Using the stylometry method can also be useful for detecting masqueraders. However, there are some things to consider when applying these methods to the IoT environment. First, in an IoT environment, there may be more machine to machine interfaces than user interfaces. In this case, since there may be no direct user behavior, it is necessary to find and apply a feature suitable for the environment of each IoT domain. Of course, there are IoT devices with user interfaces, but since they provide a different interface than the mouse, it will be possible to monitor the relevant data and apply it to research. Secondly, text writing data must take into account legal considerations of collecting data and the difficulty of learning user patterns for users who do not write much text. It is also necessary to consider that there is not much content creation in IoT devices, unlike the general IT environment.

The brainwave based studies present approaches that using biometric signal to detect malicious intention proactively. This approach has a limitation of requiring the use of sensing equipment to collect data from the user explicitly. However, when using the measure as access control for significant areas, it is expected to detect potential insiders accurately. The method could also be used effectively to detect intentions to use small-scale IoT devices that are difficult to find.

The active indicator-based method has the advantage of finding insider proactively like the sentimental based method. However, the method is based on observations of people rather than on the monitoring of devices. Therefore, it is expected that the difference, according to the environment, will be small from the environment in the existing IoT research.

D. OTHER APPROACHES

In the previous sections, we analyzed which data was used for what purpose based on the data that can be collected in each layer of the IoT structure. However, not all of the analyzed studies conducted experiments using datasets. One example of unexperimented studies are theoretical studies, and these papers are difficult to assign to the structures described above because they do not directly use specific data. That is why we analyze these papers in this section separately.

1) PROCESS ANALYSIS BASED

Most insider threat studies attempt to detect anomalous behavior or identify individuals who are more likely to be insiders. However, Bishop *et al.* [48] present the method to analyze how the process is vulnerable and suggest countermeasures to increase the process' resistance to insider attacks. To find vulnerabilities, they utilized the fault tree analysis, a static analysis technique. The approach is based on the property that, simple boolean algebra can compute cut sets and minimal cut sets. Cut sets are sets of event literals (primary events or negations of primary events) that could cause the hazard, and minimal cut sets are cuts that cannot be further reduced. One minimal cut set refers to a vulnerability in the system since if all events in the minimal cut sets occurs, it causes the hazard corresponding to the root of the fault tree. The authors presented how to analyze vulnerabilities of the processes related to a sabotage and data exfiltration attack by insiders through finding minimal cut sets.

We expect this method would be useful in enhancing the system's resistance to an insider attack by securing the process. However, this method has the disadvantage of applying only to critical industries where fault tree analysis has already been performed, such as aviation or nuclear field. Moreover, in the IoT environment, the number of events could increase dramatically, and this can raise the complexity of designing a fault tree [112]. Researchers should consider that this increase in complexity can lead to missing events in process analysis, which can affect the reliability of vulnerability analysis.

2) ROLE BASED

Sandhu *et al.* introduce the role based access control (RBAC) [113], and the use of RBAC rules means that the detection system considers the user's role during the behavior analysis. Nellikar [52] devised a policy engine to inject the user's role in the log by feeding RBAC rules to the policy engine. The author also described a simulator can simulate the insiders and generate access information with logs form. The simulator could model an insider/normal user and generates the access information based on Markov chain, and expected to overcome the difficulty of obtaining real data by simulating the real organization experience. Five classification algorithms (OC-SVM, Support Vector Data Description, One class classification, Filter for detecting outliers using interquartile ranges, and Fast Adaptive Mean Shift) were used for ITD using simulator generated log files and showed that when considering roles, higher accuracy could be obtained. The comparison between using and not using a user's role is covered in this paper, but detection using a user's role has been tried in other papers too [33], [39], [114], [115].

Kaghazgaran and Tabaki proposed deception techniques and an access control model combined approach in the paper [54]. The methods introduced honey permission that is an extended RBAC for detecting insider threats. In the assumption that someone tries to access sensitive resources not associated with a given role could be a potential insider, the proposed system could detect insider who is using honey permission. For the system, the authors introduced the calculation method of the sensitivity level of the object, the risk of the permission, the risk of the role, which permissions are appropriate for honey permission, and the number of candidate roles for honey permission assignment. With an experiment, they examined that the method increases the cost of the resulting RBAC model reasonably.

Role-based approaches are useful for detecting insider threats by finding deviation between user behavior and similar role groups or by monitoring unauthorized resources accessing. Although this method is reported applicable in situations where network or system boundaries are distinct and fixed [52], [54], in the IoT environment where the boundaries are not vague, more validation should be conducted.

3) ATTACK VECTOR MODELING

We surveyed papers related to ITD, but only a few of them were IoT related. In this section, we surveyed the paper that deals with threats caused by insiders using IoT devices, not ITD considering IoT.

Nurse *et al.* [16] addressed the issues of insider threats with the IoT perspective by focusing on the devices that employers bring to the enterprise. They presented attack vectors using VERIS's A4 (assets, actors, attributes, actions) modeling approach and context of attacks. As attack vectors, they provide eight attack vectors caused by malicious insiders and eight attack vectors caused by unintentional insiders.

Also, as an attack vector modeling technique for IoT attack capture, the authors extended Howard and Longstaff's taxonomy [116] for identifying the key aspect constituting the attack. For this extension, they added the ''physical device functionality'' category and subcategories (camera, audio recorder, storage system, device scanner, network scanner, access point, and location tracker) for the tool aspect. They also added ''records'' (picture taking, video recording, and audio recording) category to the Action aspects that possess typical attack categories and added ''personnel'' and ''event'' categories to the assets aspect.

Kammüller *et al.* [12] formally modeled insider attacks related to IoT using in the interactive theorem prover Isabelle [117]. The author describes a formal method, including a social explanation of insider threats and a representation of an attack tree, and exhibits the ability to model employee blackmail through an extended formal language and identify vulnerabilities.

VI. DISCUSSION AND CONCLUSION

In this paper, we analyzed the insider threat detection literature with the perspective of the IoT environment. The primary purpose of the analysis was to determine what features each paper uses to detect insiders, and what data were collected and utilized to detect those features. To do this, in Section I-A we first surveyed studies related to insider threat detection. We defined the concepts of insider, insider threat, insider activities, and IoT structure in Section III. After that, using the four layers of the IoT structure suggested by Suo *et al.*, we extracted three layer (perceptual, network, and application) [71]. Among the four layers, the support layer and the application have no benefit of distinguishing them from the ITD perspective, so we integrated them into the application layer. Finally, in Section V, we analyzed and summarized the data that ITD studies collected in each layer, asking ourselves how the researchers used the collected data for the ITD, and what should be considered in collecting and using such data in the IoT environment. Through this analysis, we categorized what data are used in the perceptual, network, and application layers in the surveyed literature.

In the perceptual layer, we were able to classify data used as 1) user command based, 2) system behavior based, and 3) multi-domain business data based. Studies conducted in each classification indicated that higher and more accurate detection rates could be obtained if there were more relevant information than a single data source, and that it would also be beneficial to analyze the causes of detection. Among the categories, multi-domain business data is mainly obtained through analysis of the IT environment of a company; thus, domains (email, logon, web, file, device, and print) reflecting the general office IT environment were used.

We found that multi-domain business data have the most abundant data structure. However, some data domains (logon, file, and device) can be applied to the IoT environment. In contrast, others (email, web, and print) will be difficult to apply directly to the IoT environment.

For this reason, ITD research in the IoT environment will require a data collection strategy that reflects the characteristics of the IoT environment. For example, Khan *et al.* [29] generated a dataset using NS-2 simulation based on the CERT dataset [90] for ITD research in IoT environments. However, due to the lack of information in their paper, we are not able to provide a detailed description of the dataset of Khan *et al.* in our study. Besides this, we described why, in the IoT envrionment, transmitting collected data to the detection system is more realistic than performing direct detection at the device. Furthermore, we mentioned the difficulty of installing additional software to perform data collection and transmitting functions in the IoT environment.

In the network layer, we only identified network packet-based information. For network packet-based data, both stateful packet inspection that analyzes network flow using TCP/IP/MAC of captured packets and deep packet inspection (i.e., HTTP) that analyzes application layer protocols are used. These studies mainly used methods in which the detection system collects data through such as port mirroring of the switches. However, considering that several separate networks can be configured in the IoT environment, we explained why using various types of wireless communications such as Bluetooth, NFC, Sensor network, RFID should be considered in the IoT environment.

In the application layer, we categorized data into 1) database query result based, 2) sentiment based, 3) relation based, 4) personal identification based, 5) brainwave based, and 6) active indicator based in the application layer. We explained that the data collected by the application layer are independent of the device because device can collect all transmitted and received data to provide services. We also mentioned that, even in the IoT environment, application layer data can be useful regardless of the environment because the service provider can acquire all transmitted and received data related to the service. In particular, among all categories, we expect that database query results will be most beneficial in IoT environments, given that most services are based on databases. Also, we expect that sentiment-based will provide higher coverage when used in integration with other approaches, since such data help us to detect who is mostly to be an insider proactively while other methods focus on detecting abnormal behavior.

Finally, we surveyed studies that did not perform experiment using the dataset. This includes topics such as detection model, attack vector analysis, and vulnerability detection through process analysis. We categorized other approaches into 1) process analysis based, 2) role based, and 3) attack vector modeling. The process analysis-based literature [48] presented a methodology to identify process vulnerability from insiders through static analysis technologies such as the fault tree analysis. The role-based studies suggested how to detect an insider that deviates from the behavior of a similar group based on the user's role, and by accessing data that the user's role does not need to be accessed through honey-

permission. The attack vector modeling studies proposed insider attack taxonomies and attack vectors in the IoT environment and how to represent attacks through formal modeling.

Through analysis, we would like to suggest how data can be collected and utilized in the industry to detect insider threats in the IoT environment. As we have found in our literature survey, for insider detection, using data from only one layer is insufficient. Therefore, we suggest combining three phase methods to detect insider threats in the IoT environment: 1) sentiment change detection (proactive Detection), 2) user identification change detection, and 3) user behavior change detection.

Sentimental change detection uses the user's psychological state data, which can be collected in the application layer to predict employees who are more likely to become insiders. With sentiment change detection, we can not discover insider threats directly. However, we can reduce the scope of targets to monitor and pay more attention to a limited range of employees. Reducing the scope of targets increases the detection efficiency when using machine learning or deep-learning techniques that have relatively high false-positive rates, compared to signature-based or rule-based systems.

User identification change detection allows the ITD system to detect masqueraders by distinguishing whether they are real users. Surveyed studies used data collected from perceptual and application layers for this purpose. In particular, user-command-based, system behavior-based, and personal identification-based datasets from the application layer are utilized for this purpose.

User behavior change detection allows the ITD system to extract user profiles and detect anomalies when current user behavior deviates from extracted user profiles. This method can be used to detect both masqueraders and traitors. For this method, the surveyed literature combined user information (e.g., user role, user department) and other data (multi-domain business data based on perceptual layer, deep packet inspection of network layer, and database query and relation based on application layer).

If our three-phase ITD system is adopted in the industry, we expect that the system will detect the insider threats that the surveyed literature considered. Of course, even with the adoption of such a system, insider threats on IoT devices that are not connected to the monitored network, such as the shooting of sensitive information, remain outside the detection range. Therefore, in addition to the methods used in the surveyed studies, research on ITD that considers the IoT environment needs to be conducted. We expect that future researchers, including us, will devise breakthrough ideas to fill current gaps in knowledge.

ACKNOWLEDGMENT

This article was presented in part at the MobiSec 2019 (The 4th International Symposium on Mobile Internet Security).

REFERENCES

[1] 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)*. Reading, MA, USA: Addison-Wesley, 2012.

[2] H. Schulze. (2020). *2020 Insider Threat Survey Report*. Accessed: Jan. 29, 2020. [Online]. Available: https://gurucul.com/2020-insider-threat-survey-report

[3] Breach Level Index. (2018). *Breach Level Index 2018*. Accessed: Jul. 30, 2019. [Online]. Available: https://breachlevelindex.com/

[4] B News. (2014). *Edward Snowden: Leaks That Exposed US Spy Programme*. Accessed: Aug. 15, 2019. [Online]. Available: https://www.bbc.com/news/world-us-canada-23123964

[5] The Guardian. (2013). *Bradley Manning Prosecutors Say Soldier 'Leaked Sensitive Information'*. Accessed: Aug. 15, 2019. [Online]. Available: https://www.theguardian.com/world/2013/jun/11/bradley-manning-wikileaks-trial-prosecution

[6] (2001). *Famous Cases & Criminals: Robert Hanssen*. Accessed: Aug. 15, 2019. [Online]. Available: https://www.fbi.gov/history/famous-cases/robert-hanssen

[7] Q. M. Ashraf and M. H. Habaebi, "Autonomic schemes for threat mitigation in Internet of Things," *J. Netw. Comput. Appl.*, vol. 49, pp. 112–127, Mar. 2015.

[8] J. H. Ziegeldorf, O. G. Morchon, and K. Wehrle, "Privacy in the Internet of Things: Threats and challenges," *Secur. Commun. Netw.*, vol. 7, no. 12, pp. 2728–2742, Dec. 2014.

[9] M. Hung, "Leading the IoT, gartner insights on how to lead in a connected world," Gartner Res., Stamford, CA, USA, Tech. Rep., 2017, pp. 1–29. [Online]. Available: https://www.gartner.com/imagesrv/books/iot/iotEbook_digital.pdf

[10] Ericson. (2016). *Internet of Things Forecast*. Accessed: Jan. 28, 2020. [Online]. Available: https://www.ericsson.com/en/mobility-report/internet-of-things-forecast

[11] Z.-K. Zhang, M. C. Y. Cho, C.-W. Wang, C.-W. Hsu, C.-K. Chen, and S. Shieh, "IoT security: Ongoing challenges and research opportunities," in *Proc. IEEE 7th Int. Conf. Service-Oriented Comput. Appl.*, Nov. 2014, pp. 230–234.

[12] F. Kammüller, J. R. C. Nurse, and C. W. Probst, "Attack tree analysis for insider threats on the IoT using Isabelle," in *Proc. Int. Conf. Human Aspects Inf. Secur., Privacy, Trust*. Cham, Switzerland: Springer, 2016, pp. 234–246.

[13] C News. (2013). *Hacked From China: Is Your Kettle Spying on You?* Accessed: Jan. 28, 2020. [Online]. Available: https://www.cbsnews.com/news/hacked-from-china-is-your-kettle-spying-on-you/

[14] K. Nohl and J. Lell, "Badusb—On accessories that turn evil," *Black Hat USA*, vol. 1, no. 9, pp. 1–22, Jul. 2014.

[15] ZDNet. (2019). *Alexa and Google Home Devices Leveraged to Phish and Eavesdrop on Users, Again*. Accessed: Jan. 30, 2020. [Online]. Available: https://www.zdnet.com/article/alexa-and-google-home-devices-leveraged-to-phish-and-eavesdrop-on-users-again/

[16] J. R. C. Nurse, A. Erola, I. Agrafiotis, M. Goldsmith, and S. Creese, "Smart insiders: Exploring the threat from insiders using the Internet-of-Things," in *Proc. Int. Workshop Secure Internet Things (SIoT)*, Sep. 2015, pp. 5–14.

[17] A. P. Moore, M. L. Collins, D. A. Mundie, R. M. Ruefle, and D. M. McIntire, "Pattern-based design of insider threat programs," Softw. Eng. Inst., Carnegie-Mellon Univ., Pittsburgh, PA, USA, Tech. Rep. CMU/SEI-2014-TN-024, 2014.

[18] R. A. Maxion, "Masquerade detection using enriched command lines," in *Proc. Int. Conf. Dependable Syst. Netw., . Proceedings.*, 2003, pp. 5–14.

[19] Y. Vardi, M. Theusan, A. F. Karr, W.-H. Ju, W. DuMouchel, and M. Schonlau, "Computer intrusion: Detecting masquerades," *Stat. Sci.*, vol. 16, no. 1, pp. 58–74, Feb. 2001.

[20] T. E. Senator *et al.*, "Detecting insider threats in a real corporate database of computer usage activity," in *Proc. 19th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining (KDD)*. New York, NY, USA: ACM, 2013, pp. 1393–1401.

[21] Y. Chen and B. Malin, "Detection of anomalous insiders in collaborative environments via relational analysis of access logs," in *Proc. 1st ACM Conf. Data Appl. Secur. Privacy (CODASPY)*. New York, NY, USA: ACM, 2011, pp. 63–74.

[22] 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 Hum. Behav.*, vol. 37, no. 4, p. 267, 2013.

[23] T. Rashid, I. Agrafiotis, and J. R. C. Nurse, "A new take on detecting insider threats: Exploring the use of hidden Markov models," in *Proc. Int. Workshop Manag. Insider Secur. Threats (MIST)*. New York, NY, USA: ACM, 2016, pp. 47–56.

[24] S. Mathew, M. Petropoulos, H. Q. Ngo, and S. Upadhyaya, "A data-centric approach to insider attack detection in database systems," in *Proc. Int. Workshop Recent Adv. Intrusion Detection*. Heidelberg, Germany: Springer, 2010, pp. 382–401.

[25] 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," *Secur. Commun. Netw.*, vol. 2019, pp. 1–12, Feb. 2019.

[26] W. Park, Y. You, and K. Lee, "Detecting potential insider threat: Analyzing insiders' sentiment exposed in social media," *Secur. Commun. Netw.*, vol. 2018, pp. 1–8, Jul. 2018.

[27] 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," in *Proc. Hum. Factors Ergonom. Soc. Annu. Meeting*, vol. 62, no. 1. Los Angeles, CA, USA: Sage, 2018, pp. 242–246.

[28] Y. Meidan, M. Bohadana, Y. Mathov, Y. Mirsky, A. Shabtai, D. Breitenbacher, and Y. Elovici, "N-BaIoT—Network-based detection of IoT botnet attacks using deep autoencoders," *IEEE Pervas. Comput.*, vol. 17, no. 3, pp. 12–22, Jul./Sep. 2018.

[29] A. Y. Khan, R. Latif, S. Latif, S. Tahir, G. Batool, and T. Saba, "Malicious insider attack detection in IoTs using data analytics," *IEEE Access*, vol. 8, pp. 11743–11753, 2020.

[30] F. Meng, F. Lou, Y. Fu, and Z. Tian, "Deep learning based attribute classification insider threat detection for data security," in *Proc. IEEE 3rd Int. Conf. Data Sci. Cyberspace (DSC)*, Jun. 2018, pp. 576–581.

[31] J. Oh, T. H. Kim, and K. H. Lee, "Advanced insider threat detection model to apply periodic work atmosphere," *KSII Trans. Internet Inf. Syst.*, vol. 13, no. 3, pp. 1722–1737, 2019.

[32] 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. Workshops 31st AAAI Conf. Artif. Intell.*, 2017, pp. 224–231.

[33] H. Eldardiry, E. Bart, J. Liu, J. Hanley, B. Price, and O. Brdiczka, "Multi-domain information fusion for insider threat detection," in *Proc. IEEE Secur. Privacy Workshops*, May 2013, pp. 45–51.

[34] P. Parveen, J. Evans, B. Thuraisingham, K. W. Hamlen, and L. Khan, "Insider threat detection using stream mining and graph mining," in *Proc. IEEE 3rd Int. Conf. Privacy, Secur., Risk Trust IEEE 3rd Int. Conf. Social Comput.*, Oct. 2011, pp. 1102–1110.

[35] M. Mayhew, M. Atighetchi, A. Adler, and R. Greenstadt, "Use of machine learning in big data analytics for insider threat detection," in *Proc. MILCOM - IEEE Mil. Commun. Conf.*, Oct. 2015, pp. 915–922.

[36] P. Parveen and B. Thuraisingham, "Unsupervised incremental sequence learning for insider threat detection," in *Proc. IEEE Int. Conf. Intell. Secur. Informat.*, Jun. 2012, pp. 141–143.

[37] K. Brancik and G. Ghinita, "The optimization of situational awareness for insider threat detection," in *Proc. 1st ACM Conf. Data Appl. Secur. Privacy (CODASPY)*, 2011, pp. 231–236.

[38] F. Yuan, Y. Cao, Y. Shang, Y. Liu, J. Tan, and B. Fang, "Insider threat detection with deep neural network," in *Proc. Int. Conf. Comput. Sci.* Cham, Switzerland: Springer, 2018, pp. 43–54.

[39] P. A. Legg, O. Buckley, M. Goldsmith, and S. Creese, "Automated insider threat detection system using user and role-based profile assessment," *IEEE Syst. J.*, vol. 11, no. 2, pp. 503–512, Jun. 2017.

[40] 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. IEEE Symp. Secur. Privacy Workshops*, May 2012, pp. 142–149.

[41] M. Kandias, V. Stavrou, N. Bozovic, and D. Gritzalis, "Proactive insider threat detection through social media: The YouTube case," in *Proc. 12th ACM Workshop Workshop Privacy Electron. Soc. (WPES)*. New York, NY, USA: ACM, 2013, pp. 261–266.

[42] Y. Hashem, H. Takabi, M. Ghasemigol, and R. Dantu, "Towards insider threat detection using psychophysiological signals," in *Proc. 7th ACM CCS Int. Workshop Manag. Insider Secur. Threats (MIST)*. New York, NY, USA: ACM, 2015, pp. 71–74.

[43] 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," *J. Wireless Mobile Netw., Ubiquitous Comput., Dependable Appl.*, vol. 6, no. 4, pp. 47–63, 2015.

[44] D. L. Costa, M. L. Collins, S. J. Perl, M. J. Albrethsen, G. J. Silowash, and D. L. Spooner, "An ontology for insider threat indicators development and applications," Softw. Eng. Inst., Carnegie-Mellon Univ., Pittsburgh, PA, USA, Tech. Rep. ADA615757, 2014.

[45] W. Eberle, J. Graves, and L. Holder, "Insider threat detection using a graph-based approach," *J. Appl. Secur. Res.*, vol. 6, no. 1, pp. 32–81, Dec. 2010.

[46] A. Gamachchi, L. Sun, and S. Boztas, "A graph based framework for malicious insider threat detection," 2018, *arXiv:1809.00141*. [Online]. Available: http://arxiv.org/abs/1809.00141

[47] F. L. Greitzer and D. A. Frincke, "Combining traditional cyber security audit data with psychosocial data: Towards predictive modeling for insider threat mitigation," in *Insider Threats in Cyber Security*. Boston, MA, USA: Springer, 2010, pp. 85–113.

[48] M. Bishop, H. M. Conboy, H. Phan, B. I. Simidchieva, G. S. Avrunin, L. A. Clarke, L. J. Osterweil, and S. Peisert, "Insider threat identification by process analysis," in *Proc. IEEE Secur. Privacy Workshops*, May 2014, pp. 251–264.

[49] 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. 48th Hawaii Int. Conf. Syst. Sci.*, Jan. 2015, pp. 3518–3526.

[50] A. Almehmadi and K. El-Khatib, "On the possibility of insider threat prevention using intent-based access control (IBAC)," *IEEE Syst. J.*, vol. 11, no. 2, pp. 373–384, Jun. 2017.

[51] Y. Kim and F. Sheldon, "Anomaly detection in multiple scale for insider threat analysis," in *Proc. 7th Annu. Workshop Cyber Secur. Inf. Intell. Res. (CSIIRW)*, 2011, p. 1.

[52] S. Nellikar, "Insider threat simulation and performance analysis of insider detection algorithms with role based models," Ph.D. dissertation, Dept. Elect. Comput. Eng., Univ. Illinois Urbana-Champaign, Urbana, IL, USA, 2010.

[53] P. Chattopadhyay, L. Wang, and Y.-P. Tan, "Scenario-based insider threat detection from cyber activities," *IEEE Trans. Comput. Social Syst.*, vol. 5, no. 3, pp. 660–675, Sep. 2018.

[54] P. Kaghazgaran and H. Takabi, "Toward an insider threat detection framework using honey permissions," *J. Internet Services Inf. Secur.*, vol. 5, no. 3, pp. 19–36, 2015.

[55] O. Lo, W. J. Buchanan, P. Griffiths, and R. Macfarlane, "Distance measurement methods for improved insider threat detection," *Secur. Commun. Netw.*, vol. 2018, pp. 1–18, Jan. 2018.

[56] M. B. Salem, S. Hershkop, and S. J. Stolfo, "A survey of insider attack detection research," in *Insider Attack and Cyber Security*. Boston, MA, USA: Springer, 2008, pp. 69–90.

[57] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection: A survey," *ACM Comput. Surv.*, vol. 41, no. 3, p. 15, 2009.

[58] H. Jiang, J. Nagra, and P. Ahammad, "SoK: Applying machine learning in security–a survey," 2016, *arXiv:1611.03186*. [Online]. Available: http://arxiv.org/abs/1611.03186

[59] I. A. Gheyas and A. E. Abdallah, "Detection and prediction of insider threats to cyber security: A systematic literature review and meta-analysis," *Big Data Analytics*, vol. 1, no. 1, p. 6, Dec. 2016.

[60] I. Homoliak, F. Toffalini, J. Guarnizo, Y. Elovici, and M. Ochoa, "Insight into insiders and IT: A survey of insider threat taxonomies, analysis, modeling, and countermeasures," *ACM Comput. Surv.*, vol. 52, no. 2, pp. 1–40, May 2019.

[61] A. Kim, J. Oh, J. Ryu, J. Lee, K. Kwon, and K. Lee, "Sok: A systematic review of insider threat detection," *J. Wireless Mobile Netw., Ubiquitous Comput., Dependable Appl.*, vol. 10, no. 4, pp. 46–67, Dec. 2019.

[62] R. C. Brackney and R. H. Anderson, "Understanding the insider threat. Proceedings of a March 2004 workshop," RAND Corp., Santa Monica, CA, Tech. Rep. ADA429854, 2004.

[63] P. Parveen, N. McDaniel, V. S. Hariharan, B. Thuraisingham, and L. Khan "Unsupervised ensemble based learning for insider threat detection," in *Proc. Int. Conf. Privacy, Secur., Risk Trust Int. Conf. Social Comput.*, Sep. 2012, pp. 718–727.

[64] M. Kont, M. Pihelgas, J. Wojtkowiak, L. Trinberg, and A.-M. Osula, "Insider threat detection study," NATO CCD COE, Tallinn, Estonia, Tech. Rep., 2015. [Online]. Available: https://ccdcoe.org/uploads/2018/10/Insider_Threat_Study_CCDCOE.pdf

[65] F. L. Greitzer, A. P. Moore, D. M. Cappelli, D. H. Andrews, L. A. Carroll, and T. D. Hull, "Combating the insider cyber threat," *IEEE Secur. Privacy Mag.*, vol. 6, no. 1, pp. 61–64, 2008.
[66] J. Hunker and C. W. Probst, "Insiders and insider threats-an overview of definitions and mitigation techniques," *J. Wireless Mobile Netw., Ubiquitous Comput., Dependable Appl.*, vol. 2, no. 1, pp. 4–27, 2011.
[67] CERT Insider Threat team, "Unintentional insider threats: A foundational study," Softw. Eng. Inst., Carnegie Mellon Univ., Pittsburgh, PA, USA, Cahier De Recherche Tech. Note CMU/SEI-2013-TN-022 , 2013, vol. 18.
[68] CERT National Insider Threat Center, "Common sense guide to mitigating insider threats, sixth edition," CMU Softw. Eng. Inst., Pittsburgh, PA, USA, Tech. Rep., 2018.
[69] K. Ashton, "That 'Internet of Things' thing," *RFID J.*, vol. 22, no. 7, pp. 97–114, 2009.
[70] S. Haller, S. Karnouskos, and C. Schroth, "The Internet of Things in an enterprise context," in *Proc. Future Internet Symp.* Heidelberg, Germany: Springer, 2008, pp. 14–28.
[71] H. Suo, J. Wan, C. Zou, and J. Liu, "Security in the Internet of Things: A review," in *Proc. IEEE Int. Conf. Comput. Sci. Electron. Eng. (ICCSEE)*, vol. 3, Mar. 2012, pp. 648–651.
[72] S. N. Swamy, D. Jadhav, and N. Kulkarni, "Security threats in the application layer in IOT applications," in *Proc. Int. Conf. I-SMAC (IoT Social, Mobile, Analytics Cloud) (I-SMAC)*, Feb. 2017, pp. 477–480.
[73] D. Lavrova and A. Pechenkin, "Applying correlation and regression analysis to detect security incidents in the Internet of Things," *Int. J. Commun. Netw. Inf. Secur.*, vol. 7, no. 3, p. 131, 2015.
[74] A. Haroon, M. Ali, Y. Asim, W. Naeem, M. Kamran, and Q. Javaid, "Constraints in the IoT: The world in 2020 and beyond," *Int. J. Adv. Comput. Sci. Appl.*, vol. 7, no. 11, pp. 252–271, 2016.
[75] A. Sehgal, V. Perelman, S. Kuryla, and J. Schonwalder, "Management of resource constrained devices in the Internet of Things," *IEEE Commun. Mag.*, vol. 50, no. 12, pp. 144–149, Dec. 2012.
[76] Z. Shelby and C. Bormann, *6LoWPAN: The wireless embedded Internet*, vol. 43. Hoboken, NJ, USA: Wiley, 2011.
[77] H. Kozushko, "Intrusion detection: Host-based and network-based intrusion detection systems," *Independ. Study*, pp. 1–23, Sep. 2003. [Online]. Available: https://pdfs.semanticscholar.org/471b/6047150e82d5b94cbcf1fed36586dcf929c1.pdf
[78] A. P. Sample, D. J. Yeager, and J. R. Smith, "A capacitive touch interface for passive RFID tags," in *Proc. IEEE Int. Conf. RFID*, Apr. 2009, pp. 103–109.
[79] K. K. R. Kendall, "A database of computer attacks for the evaluation of intrusion detection systems," Ph.D. dissertation, Massachusetts Inst. Technol., Cambridge, MA, USA, 1999.
[80] M. Ben-Salem. (2009). *Ruu Dataset*. Accessed: Aug. 15, 2019. [Online]. Available: http://ids.cs.columbia.edu/content/ruu.html
[81] M. B. Salem and S. J. Stolfo, "Modeling user search behavior for masquerade detection," in *Proc. Int. Workshop Recent Adv. Intrusion Detection*. Heidelberg, Germany: Springer, 2011, pp. 181–200.
[82] CALO Project. (2015). *Enron Email Dataset*. Accessed: May 1, 2020. [Online]. Available: https://www.cs.cmu.edu/~enron/
[83] V C 2008. (2008). *Mc3—Cell Phone Calls*. Accessed: Sep. 1, 2020. [Online]. Available: http://www.cs.umd.edu/hcil/varepository/VASTChallenge2008/challenges/MC3-CellPhoneCalls/
[84] M. Schonlau. (2020). *Masquerading User Data*. Accessed: Feb. 23, 202. [Online]. Available: http://www.schonlau.net/intrusion.html
[85] C. W. Probst, J. Hunker, M. Bishop, and D. Gollmann, *Insider Threats in Cyber Security*, vol. 49. Boston, MA, USA: Springer, 2010.
[86] R. A. Maxion and T. N. Townsend, "Masquerade detection augmented with error analysis," *IEEE Trans. Rel.*, vol. 53, no. 1, pp. 124–147, Mar. 2004.
[87] R. A. Maxion and T. N. Townsend, "Masquerade detection using truncated command lines," in *Proc. Int. Conf. Dependable Syst. Netw.*, 2002, pp. 219–228.
[88] S. Greenberg. (2020). *Www and Unix Data Sets*. Accessed: Feb. 23, 2020. [Online]. Available: http://saul.cpsc.ucalgary.ca/pmwiki.php/HCIResources/HCIWWWUnixDataSets
[89] A. Fülöp, L. Kovács, T. Kurics, and E. Windhager-Pokol, "Balabit mouse dynamics challenge data set," Tech. Rep., 2016. [Online]. Available: https://github.com/balabit/Mouse-Dynamics-Challenge
[90] CERT. (2016). *Cert Insider Threat Test Dataset*. Accessed: Aug. 15, 2019. [Online]. Available: https://resources.sei.cmu.edu/library/asset-view.cfm?assetid=508099
[91] S. Rothmann and E. P. Coetzer, "The big five personality dimensions and job performance," *SA J. Ind. Psychol.*, vol. 29, no. 1, pp. 68–74, 2003.
[92] V. Paxson, "Bro: A system for detecting network intruders in real-time," *Comput. Netw.*, vol. 31, nos. 23–24, pp. 2435–2463, Dec. 1999.
[93] L. Daigle, *Whois Protocol Specification*, document RFC 3912, IETF, Sep. 2004.
[94] A. Ene, W. Horne, N. Milosavljevic, P. Rao, R. Schreiber, and R. E. Tarjan, "Fast exact and heuristic methods for role minimization problems," in *Proc. 13th ACM Symp. Access Control Models Technol. (SACMAT)*, 2008, pp. 1–10.
[95] M. B. Salem and S. J. Stolfo, "Masquerade attack detection using a search-behavior modeling approach," Comput. Sci. Dept., Columbia Univ., Tech. Rep. CUCS-027-09, 2009.
[96] DARPA. (1998). *1998 Darpa Intrusion Detection Evaluation Dataset*. Accessed: Feb. 23, 2020. [Online]. Available: https://www.ll.mit.edu/r-d/datasets/1998-darpa-intrusion-detection-evaluation-dataset
[97] J. Glasser and B. Lindauer, "Bridging the gap: A pragmatic approach to generating insider threat data," in *Proc. IEEE Secur. Privacy Workshops*, May 2013, pp. 98–104.
[98] L. Santos, C. Rabadão, and R. Gonçalves, "Flow monitoring system for IoT networks," in *Proc. World Conf. Inf. Syst. Technol.* Cham, Switzerland: Springer, 2019, pp. 420–430.
[99] A. Kamra, E. Terzi, and E. Bertino, "Detecting anomalous access patterns in relational databases," *VLDB J.*, vol. 17, no. 5, pp. 1063–1077, Aug. 2008.
[100] M. Kandias, A. Mylonas, N. Virvilis, M. Theoharidou, and D. Gritzalis, "An insider threat prediction model," in *Proc. Int. Conf. Trust, Privacy Secur. Digit. Bus.* Heidelberg, Germany: Springer, 2010, pp. 26–37.
[101] A. Go, R. Bhayani, and L. Huang. (2013). *Sentiment140*. Site Functionality, Abruf Am. Accessed: 2016. [Online]. Available: http://help.sentiment140.com/site-functionality
[102] W. Realms, *Wow Census*. 2015.
[103] A. Varga, "OMNeT++," in *Modeling and Tools for Network Simulation*. Heidelberg, Germany: Springer, 2010, pp. 35–59.
[104] D. J. Cook and L. B. Holder, "Graph-based data mining," *IEEE Intell. Syst. Their Appl.*, vol. 15, no. 2, pp. 32–41, Mar./Apr. 2000.
[105] Y. R. Tausczik and J. W. Pennebaker, "The psychological meaning of words: LIWC and computerized text analysis methods," *J. Lang. Social Psychol.*, vol. 29, no. 1, pp. 24–54, Mar. 2010.
[106] D. J. Barret, *Mediawiki*. Sebastopol, CA, USA: O'Reilly Media 2008.
[107] J. Segall and R. Greenstadt, "The illiterate editor: Metadata-driven revert detection in Wikipedia," in *Proc. 9th Int. Symp. Open Collaboration (WikiSym)*, 2013, pp. 1–8.
[108] J. B. Meixner and J. P. Rosenfeld, "A mock terrorism application of the P300-based concealed information test," *Psychophysiology*, vol. 48, no. 2, pp. 149–154, Feb. 2011.
[109] 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)*. Reading, MA, USA: Addison-Wesley, 2012.
[110] L. Mitrou, "The impact of communications data retention on fundamental rights and democracy–the case of the eu data retention directive," in *Surveillance and Democracy*. Milton, U.K.: Routledge-Cavendish, 2010, pp. 143–163.
[111] F. L. Greitzer, D. Frincke, and M. Zabriskie, "Social/ethical issues in predictive insider threat monitoring," in *Information Assurance and Security Ethics in Complex Systems: Interdisciplinary Perspectives*. Hershey, PA, USA: IGI Global, 2011, pp. 132–161.
[112] L. Podofillini, B. Sudret, B. Stojadinovic, E. Zio, and W. Kröger, *Safety and Reliability of Complex Engineered Systems: ESREL 2015*. Boca Raton, FL, USA: CRC Press, 2015.
[113] R. Sandhu, D. Ferraiolo, and R. Kuhn, "The NIST model for role-based access control: Towards a unified standard," in *Proc. ACM Workshop Role-Based Access Control*, vol. 10, Jul. 2000, pp. 344287–344301.
[114] P. A. Legg, O. Buckley, M. Goldsmith, and S. Creese, "Caught in the act of an insider attack: Detection and assessment of insider threat," in *Proc. IEEE Int. Symp. Technol. Homeland Secur. (HST)*, Apr. 2015, pp. 1–6.
[115] A. Riahi, Y. Challal, E. Natalizio, Z. Chtourou, and A. Bouabdallah, "A systemic approach for IoT security," in *Proc. IEEE Int. Conf. Distrib. Comput. Sensor Syst.*, May 2013, pp. 351–355.
[116] J. D. Howard and T. A. Longstaff, "A common language for computer security incidents," Sandia National Labs., Albuquerque, NM, USA, Tech. Rep. SAND98-8667, 1998.

[117] T. Nipkow, L. C. Paulson, and M. Wenzel, *Isabelle/HOL: A Proof Assistant for Higher-Order Logic*, vol. 2283. Springer, 2002.

ARAM KIM received the B.S. and M.S. degrees in computer science from Korea University, Seoul, South Korea, in 2005 and 2008, respectively. From 2008 to 2015, he was a Senior Engineer with the KEPCO Engineering and Construction Comapny Inc. Since 2015, he has been a Senior Researcher with the Cyber Security Department, Korea Institute of Nuclear Nonproliferation, and Control. His research interests include insider threat detection, abnormal detection, and data-driven decision making. Cybersecurity policy implementation to prevent and to mitigate radiological consequences of sabotage at nuclear facilities and unauthorized removal of nuclear material is another interest.

JUNHYOUNG OH received the B.S. degree from Korea University, in 2017. He is currently pursuing the Unified Master's and Doctor's Course Students with the Graduate School of Information Security, Korea University. He is also a member of the Risk Management Laboratory, Korea University. His research interests include usable security, and data analysis and Privacy in the IoT.

JINHO RYU received the B.S. and M.S. degrees from Seoul National University, Seoul, South Korea, in 2017 and 2015, respectively. Since 2017, he has been a Researcher with the Korea Institute of Nuclear Nonproliferation and Control (KINAC). His research interests include improving the current framework of cyber security exercise for nuclear facilities, and protection of nuclear facilities against EMP threat.

KYUNGHO LEE received the Ph.D. degree from Korea University. He has been leading the Risk Management Laboratory, Korea University, since 2012. He was a former CISO with Naver Corporation. He was a CIO, CISO, and CPO with Korea University. He is currently a Professor with the Graduate School of Information Security, Korea University.