Warder: Online Insider Threat Detection System Using Multi-Feature Modeling and Graph-Based Correlation

Jianguo Jiang[†‡], Jiuming Chen[†‡§], Tianbo Gu[§], Kim-Kwang Raymond Choo[¶], Chao Liu[†‡], Min Yu[†‡*], Weiqing Huang[†‡], and Prasant Mohapatra[§]
[†] Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China
[‡] School of Cyber Security, University of Chinese Academy of Sciences, Beijing, China
[§] Department of Computer Science, University of California, Davis, CA, USA
[¶] Department of Information Systems and Cyber Security, University of Texas at San Antonio, San Antonio, TX 78249, USA
Email: {jiangjianguo,chenjiuming,liuchao, yumin, huangweiqing}@iie.ac.cn, tbgu,pmohapatra@ucdavis.edu, raymond.choo@fulbrightmail.org
* corresponding author

Abstract—**Existing insider threat detection models and frameworks generally focus on characterizing and detecting malicious insiders, for example by fusing behavioral analysis, machine learning, psychological characters, management measures, etc. However, it remains challenging to design a practical insider threat detection scheme that can be efficiently implemented and deployed in a real-world system. For example, existing approaches focus on extracting features from user behavioral activities but they lack in-depth correlation and decision making for suspected alerts; thus, resulting in high false positives and low detection accuracy. In this work, we propose a novel online insider threat detection system, *Warder*, which leverages diverse feature dimensions (using neural language processing) and fuses content and behavior features to create a user's daily profile to facilitate threat detection. Besides, hypergraph-based threat scenario feature tree is designed to correlate suspicious users' activities with threat scenarios to further screen the users. In practice, *Warder* can also be constantly updated using newly discovered features and threat scenarios. We evaluate the performance of *Warder* using the public CMU CERT dataset, as well as that of approaches from the Oxford group and CMU group. Findings from the evaluation demonstrate that *Warder* outperforms the other two competing approaches.**

Index Terms—**anomaly detection, insider threat detection, hypergraph, online activity**

I. Introduction

Defending against a privileged and motivated insider is challenging, as evidenced by the ongoing and increasing number of insider-related incidents. For example, according to a study performed by CA Inc, ninety percent of organizations surveyed reportedly felt vulnerable to insider attacks [1], and more than 53% of the surveyed organizations had experienced some insider attack in the past 12 months. Existing approaches, such as security controls and policies, data loss prevention (DLP) solutions, and encryption and access management solutions, were generally used to prevent unauthorized access. However, the effectiveness of such solutions is debatable.

In order to detect malicious insiders, a number of different detection frameworks or prototypes have been proposed to characterize malicious insiders and threat scenarios. These models generally focus on building comprehensive profile of users and then using anomaly detection [2], [3], scenario analysis [4] and graph analysis [5] to detect malicious insiders. Although both objective and subjective factors [6] of users have been utilized to build comprehensive frameworks, these models are not easily trained and evaluated due to lack of real-world data. Therefore, behavioral activities of users are widely used to characterize and build profiles of users, as evidenced by existing research [7]. However, such systems generally have high false positive and low detection accuracy in a practical deployment, due to reasons such as:

- Existing methods usually focus only on designing and extracting features from behavioral aspects of users. In other words, important content-based features, such as topics in email or sentiment of texts, are usually ignored despite their potential to be used as the indicators of an insider compromise.
- Anomaly behaviors are not exactly the same as malicious threats. Although existing approaches combine complex algorithms (e.g. neural networks) that could accurately predict anomalous activities, they typically lack follow-up analysis and correlation analysis about the predicted alerts, both of which are essential in reducing false positives.

To mitigate the above discussed limitations, we propose an online insider threat detection system adopting a two-step detection model. First, we build user profiles based on their behavioral features including not only the typical behavioral patterns but also the content from users (extracted via using some Neural Language Processing (NLP) and Information Retrieval (IR) methods). Then, we design a visualization and correlation model using hypergraph to analyze the suspected users' logs in order to reduce the false alerts and identify potential insiders which cannot be detected by the first step. Our approach can be summarized as follows:

- In order to have a more robust user profile, we extract content-based features using NLP and fuse the features with behavioral pattern features. This allows us to create diverse feature dimensions that can significantly enhance the performance of our threat detection model.
- We design a correlation model and hypergraph to analyze the flagged users and detect potential insiders; thus, achieving high accuracy and low false positive rate.
- The models and algorithms proposed in the work are

designed for real-time threat detection, and in comparison to other similar (after-the-fact analysis) models, our proactive online prediction and detection of malicious insiders can significantly reduce insider attacks.

- We provide and implement a prototype system for insider threat detection to demonstrate the practicability in a real-world insider threat detection scenario. Moreover, our system can be easily extended for newly found threat scenarios or new features.

The rest of the paper is organized as follows. We will briefly summarize related literature in the next section. In Sections III and IV, we will present the proposed model, and the evaluation setup and findings. Finally, we present the discussion and conclusion in Section V.

II. Related Work

To detect and mitigate insider threats, researchers have put forward a number of diverse solutions. In this section, we summarize the related literature on insider threat detection.

Behavioral activities or system logs of users are a rich data source to train and evaluate a robust insider threat detection system. Some existing approaches were designed to detect insider threats based on the established users profile of their behavioral activities [4]. Such approaches generally use users' behavioral logs, such as email, device, file [5], and logon patterns, as data source, and then use some anomaly based or machine learning based methods to detect insider threat [8]. However, anomaly is not the same as insider threats. In recent times, researchers have also realized the importance of a decision layer in a detection system [9]. In [10], for example, the authors proposed a system which designed a time window for correlation and threat decision.

However, existing approaches are not generally designed to combine feature based classification with threat decision to build a general system, which will result in improved detection accuracy. Such systems have the potential to achieve a relatively high accuracy in detecting known scenarios. Other desirable properties include the capability to extend the system' threat scenario updates (e.g. whenever a new attack trend is known), and facilitate online and real-time detection.

Therefore, in this paper, we aim to design an online insider threat system, which could be easily extended, achieve high detection accuracy, and monitor threats in real-time. The proposed system is described in the next section.

III. Proposed Insider threat Detection System

The proposed system is designed to monitor, identify and/or predict potential malicious insiders by analyzing the users' activities logs (e.g. email, file, web browsing, device, and login activities data) – see also Figure 1. Specifically, for each user, we build a daily activity profile and extract the content-based and behavior-based features to train the anomaly activities classification model. Then, a user whose daily behaviors deviate with its historical activities or other users' activities within the same organization can be detected. Subsequently, the suspected anomaly is further analyzed and screened, by using hyper graph visualization and correlating based feature trees. After the correlation and visualization phases, the system determines whether the user satisfies the definition of a malicious insider (e.g. based on the organization's threat scenarios in a specific context). Next, we will explain the key models, algorithms, and components of the online insider threat detection system.

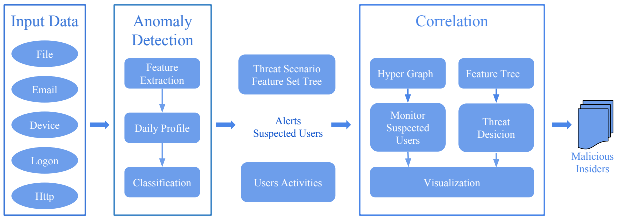


Fig. 1: Proposed insider threat detection system, *Warder*

A. Profile user state via diverse features fusion

In addition to user behavioral features, content-based features (e.g. topic of emails, sentiment tendency of emails, keywords of web pages, and topic of web pages) are also important feature dimensions that can be utilized to infer the attack motivation and identify and predict potential malicious insiders. Therefore, we extract and fuse content-based features with behavior-based features into the model to greatly enhance the model's performance. We also deign a learning based anomaly classification model to generate anomalous alerts to facilitate correlation and decision making.

TABLE I: Features for Building Daily Activity Profile of Users

Data Source	***Feature Type***
Logon/Logoff	Daily Logon/Logoff Times
Logon/Logoff	Daily Off-Work Hours Logon/Logoff Times
Logon/Logoff	Daily numbers of PC for Logon/Logoff
Device	Daily number of device connection
Device	Daily number off-work hours device connection
Device	Daily numbers of PC for device connection
File	Daily number of .exe files
File	Daily number of different files
File	Daily number of files
File	Daily number of files off-work hours
File	Daily numbers of PC for files
Email	Daily number of sent emails
Email	Daily number of sent emails out organization
Email	Daily number of sent emails within organization
Email	Daily number of attachments within emails
Email	Daily average email size
Email	Daily number of receivers of sent emails
Email	Daily number of sent emails off-work hours
Email	Daily number of PC for emails
Email	Daily number of sent emails within organization
Email	Daily number of topic-related emails
Email	Daily number of sentiment-related emails
Http	Daily number of web pages browsed
Http	Daily number of Wikileaks-related web pages
Http	Daily number of sentiment-related web pages
Http	Daily number of topic-related web pages
Http	Daily number of key-logger related web pages

a) Features exploration for Building of User Daily Profile: In this work, we design a daily multi-feature profile of users consisting of approximately 31 features that are extracted from users' behavior and content aspects, such as Login/Logoff, Device, File, Email, and HTTP – see also Table I. To build a robust detection model, we carve out a new path to explore content-based features shown in [11]. The language usage in emails is a valuable indication of users' subjective factors, such as emotion, motivation, and psychology. Therefore, we adopt the features from email content based on daily topic and daily sentiment. Besides, the web browsing habit and content trend may also reflect users' emotion, psychology, and motivation. Accordingly, the features from users' daily topic and sentiment are also integrated into our model for threat detection. The detailed processing of these contents is described in [11].

b) Detection of the insider threat: In order to determine whether users' daily activities are normal or anomalous, we build a classification model based on the features from the users' daily profile. It can be described as follows:

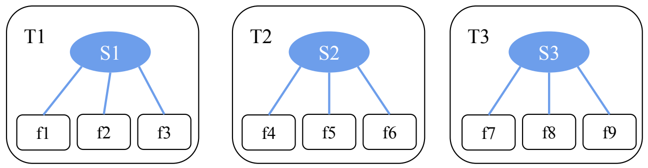
- *Extract the features from users' daily activities and build a daily features profile for each user*. The daily features profile for the user can be denoted as a matrix F(User, feature, Date) = M × D × S, where M is the number of users in an organization during a period of time, D denotes the monitored date sets, and S represents the number of features. The label of the feature matrix expresses whether a user's activities at a specific date are normal or anomalous.
- *Train the anomaly activity classification model*. After mapping user daily activities into the feature matrix, we design a classification model using machine learning algorithms, such as random forest, support vector machine (SVM), logistic regression, and neural network. The output of the model is the alerts about the users whose activities are determined to be anomalous.

B. Graph-based correlation and decision-making

Not all anomalous behaviors have malicious intent, as they can be due to benign users performing operations that deviate from their norm (e.g. due to a last-minute work request from their supervisors or other senior managers), which can result in false positive and incorrect alerts. Besides, a sophisticated malicious insider would deliberately hide their malicious activities within their normal behaviors, which cannot be easily detected and may be ignored by the detection system.

Therefore, to overcome the above shortcomings, a two-step algorithm is proposed for alert correlation and decision-making. The first part of the algorithm is alert decision algorithm as shown in Algorithm 15, which analyzes the alerts generated by the anomaly classification model and determines whether the users being flagged in the alerts are potentially malicious. The second part of the algorithm is to continuously monitor and filter the flagged (suspicious) users by correlating their daily activities with other threat scenarios and discovering the hidden malicious insiders – see Algorithm 19. Besides, a real-time visualization model using hyper graph is designed to exhibit current suspected users and their anomalous activities.

a) Alert Decision-making: For each recognized insider threat scenario, there are some key features indicating malicious insiders and threat activities. The probability of suspected users to be a malicious insider would be pretty high if these key features entirely occur during a period. Therefore, we firstly statistically probes the feature patterns from the known threat scenarios and create a feature tree for each scenario. Next, we compare the alerted users' anomalous feature set with the threat scenario feature trees T. The features tree that is correlated and built from a known insider threat scenarios is shown in Figure 2. In this work, we devise three different features trees $T = \{T1, T2, T3\}$ to characterize the three threat scenarios in the CMU CERT dataset [12].



T = {T1, T2, T3, ...} // Feature Trees for threat scenarios(could be extended)
Ti = {Ni, Li}
Ni: feature tree node for scenario i, identification of scenario i.
Li: feature tree leaves for scenario i, anomalous features related to scenario i.

Fig. 2: Feature trees for three threat scenarios in CMU CERT dataset [12]

Having more key features anomaly occurring within users' daily activities, will facilitate the making of an accurate decision about suspected users. To examine whether the alerted users' activities conform to pre-defined threat scenario feature trees, we design a decision-making algorithm using tree pruning principle – see Algorithm 1.

Algorithm 1 Alert Decision-making Algorithm

```
Input: date D, alerts $A$, feature set $F$
Output: Malicious Insiders $R$
initialize a empty hyper graph $H_{date}$ for the date;
for each $alert\_user \in A$ do
    initialize scenario tree set $T_i$ with feature node leaves;
    initialize a hyper graph node $H_i$;
    collect anomalous feature set $F_{user}$ for user during 30 days before date D;
    for each anomaly feature $f_j \in F_{user}$ do
        $H_i = H_i \cup f_j$;
        $T_i = T_i - f_j$;
        if $T_i = \phi$ then
            Generate a malicious insider decision;
            $R = R + alert\_user$;
        end if
    end for
    $H_{date} = H_{date} \cup H_i$
end for
```

After the detection system generates an anomalous alert, the system would immediately initialize a set of scenario feature trees for the alerted user. Then, the system filters the activity history of the alerted user during a period of time W. The values of the features are compared to its history and other users' data in the same organization. If the anomalous features are in the scenario tree, the leaf feature would be removed and the branch would be pruned. When the leaves of the feature tree for the suspected user become empty, the system automatically ascertains the user to be malicious insider. Meanwhile, the features with a large deviation are dynamically added to a visualization graph. Before the system makes a decision, the analysts can monitor and understand the current threat status and make an earlier decision based on the real-time visualization graph.

Algorithm 2 Threat Monitor and Screen Algorithm

```
Input: Scenario Tree $T$, feature set $F$, activities logs $L$
Output: Malicious Insiders $R$
initialize monitoring user list $U$(user to be monitored) ;
Initialize anomalous feature set $F_{user}$ for user during 30 days before current date;
for each feature $f_i \in T$ do
    Filter daily activities logs $L$;
    if $f_i \in F_{user}$ then
        $U = U \cup user$;
    end if
end for
for each monitored user $U_j \in U$ do
    initialize scenario tree set $T_i$ with feature node leafs;
    for each anomaly feature $f_j \in F_{user}$ do
        $T_i = T_i - f_j$;
        if $T_i = \phi$ then
            Generate a malicious insider decision;
            $R = R + alert\_user$;
        end if
    end for
end for
$H_{date} = H_{date} \cup H_i$
```

b) Threat Monitor and Screen: To detect the potential insiders described before, we design a real-time threat monitor and filter algorithm as described in Algorithm 2. This algorithm filters out users' daily features in threat scenario feature trees in real time. The feature values and corresponding scenarios feature tree for suspected users will be continuously monitoring. During a time window W, the threat scenario feature tree for the suspected user is constantly being pruned. After the time window, the scenario feature tree is utilized to confirm if the suspected user acts as a malicious insider, and if so, it will be deleted from the visualization model.

c) Hypergraph-based Threats Visualization: Visualizing suspected activities and alerts can empower analysts to promptly identify and react to a potential insider threat; thereby, mitigating the impact of the threat. Therefore, we implement a hyper graph to display the correlation between suspected users and their activities.

Hyper graph [13] is a graph where an edge can connect any number of vertices. We use the hyper graph H to represent the suspected users and their anomaly activities. In the graph, the vertices set X represents the anomalous feature set of the alerted users, and the hyper edges set E are denoted as the multi-attributes correlation between the suspected users and their activity features. The mathematical expression of the hyper graph is described as follows:

$$H = (X,\ E) \tag{1}$$

$$X = \{f1, f2, f3, f4, f5, f6, f7, f8, f9\} \tag{2}$$

$$E = \{user1, user2, user3, user4, user5, user6\} \tag{3}$$

$$user1 = \{f1, f2, f3\} \tag{4}$$

The hyper graph is dynamically generated, updated and pruned based on Algorithms 1 and 2. The hyper graph displays current potential suspected insiders and their abnormal activities, which assists analysts to monitor the potential threat and be proactive in their mitigation strategies.

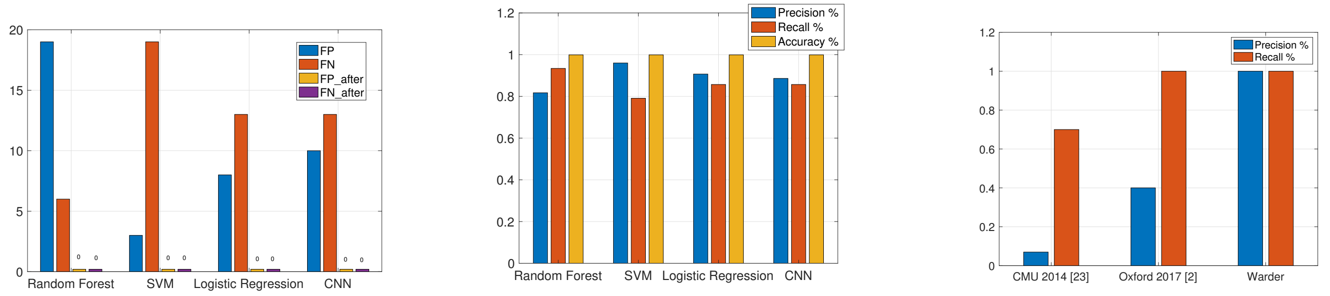
IV. Evaluation

We use the public dataset CMU CERT v4.2 [12] to evaluate the performance of *Warder*. The dataset collected and recorded users' behaviors activities data, such as logon/logoff, email, file actions, instant messages, printer, process, and web events. The dataset includes activities by normal users and labeled malicious insiders. Three threat scenarios, including information theft, IT sabotage, and corresponding malicious insider activities, are designed by domain experts and inserted to the normal users' logs. There are a total of 70 labeled malicious insiders inserted (i.e. 30 insiders in Scenario 1, 30 insiders in Scenario 2, and 10 insiders in Scenario 3). Each insider contains nearly 10 daily malicious activities. We use these labeled insiders and their activities as negative samples and divide these insiders as 7:3 to build the training and test dataset for the classification model.

A. Performance

We use four machine learning models (i.e, random forest, SVM, logistic regression and CNN) to evaluate the performance of the combined feature profiles for anomaly detection. After generating the anomalous alerts, we execute the alert correlation and threat decision-making model to further screen the suspected users. We use four conventional parameters TP, FP, TN, and FN to evaluate the performance of *Warder* and compared with some existing representative detection systems.

a) Threat Classification: Figure 3a and Figure 3b exhibit the anomaly classification performance of these models. Based on Figure 3a, the random-forest based model has the lowest FN, which means that the least number of malicious insiders is ignored. SVM based model has the lowest FP, which generates fewest false alerts. According to Figure 3b, the SVM based model has the highest precision (i.e. 96%), and



(a) FP and FN before and after decision

(b) Threat detection performance

(c) Threat detection comparison result

Fig. 3: Evaluation result for insider threat detection system *Warder*.

random forest based model has the highest recall (i.e. 93.4%). Furthermore, the average precision and recall are over 89.25% and 85.97% for the four models, which proves that *Warder* has minimal false alerts and high recall rate for true insiders. The accuracy of all the four models could reach nearly 100%, which demonstrates that regardless of the machine learning algorithms we use to train the model, the system can still achieve high detection. This demonstrates the advantage of fusing content-based and behaviors-based features.

However, there still exist some malicious insiders whose activities are pretty similar to normal activities; thereby, being ignored. Nonetheless, just one hidden malicious insider could hurt / harm the organization. Therefore, we design and implement an alert correlation and decision-making model to automatically provide an in-depth analysis of the suspected users and discover the hidden insiders.

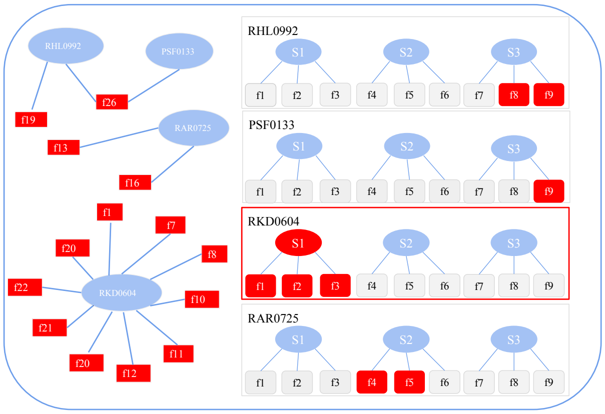


Fig. 4: Daily Threat Visualization on 2010-08-05

b) Alert Correlation and Decision-making: After generating the alerted users, the in-depth analysis via hyper graph and threat scenario futures tree is required to make a final decision about whether the user is a real malicious insider. We create the threat scenario feature trees for the suspected users and attach the trees to the visualization interface.

Figure 4 shows a visualization example on 2010-08-05. Three alerted users ”RHL0922”, ”RKD0604” and ”PSF0133” are being continuously monitored via building threat scenario features trees. The hyper graph in the left part of Figure 4 reveals four suspected users and their anomalous features. The anomalous features are displayed as the red node, and the users are marked as a blue circle and connected to the features nodes via a hyper edge. The hyper graph could be constantly monitored and updated when old users are excluded from the list of suspected users and new suspected users are detected.

The right part of Figure 4 shows the threat scenario trees for the suspected users. The number of feature trees is equal to the scenario feature trees defined in section III. The leaves of features trees are the principal features corresponding to a threat scenario. The leaf node is marked red and pruned if it deviates from the user’s historical activities. Further, if all the leaves of a threat scenario tree change to red, the suspected user is determined to be a real malicious insider.

We take the user ”RAR0725” as an example. Our threat detection system continually monitors and detects the user ”RAR0725” having the anomalous features f13 and f16. The two features represent anomaly on job-related web pages browse times and daily device usage times, which are important indicators for insider threat scenarios on stealing of data. So user ”RAR0725” is detected since its activities match the anomalous features in threat scenario 2. *Warder* is able to detect user ”RAR0725” although it is not detected by the anomaly classification model. User ”RKD0604” is finally detected as a malicious insider because its leaves in Scenario 1 turn red, which means its activities completely match threat scenario 1. The other suspected users with less red leaves in threat scenario trees will still be monitored and visualized until the system generates a final alert decision or the monitor time exceeds a fixed time window.

The anomaly classification model may produce false alerts and miss some real malicious insiders. After correlating the suspected users with defined threat scenarios, a final decision could be made for the alerted users. Besides, non-flagged insiders are continually visualized and monitored. As shown in Figure 3a, the FN and FP have been both reduced to 0. Figure 3c shows the undetected behaviors and false alerts could be overcame by the decision module, which suggests that our designed decision layer can greatly enhance the precision and recall rate of threat detection precision.

c) Performance comparison: We compare *Warder* with two representative approaches of the Oxford group [2] and CMU group [14]. They also evaluated their approach and provided the performance results based on the same dataset. There are also some competing approaches on insider threat detection, such as those in [15], [16]. However, these work do not provide the experiment result of their approaches in their papers. Moreover, some other work only focuses on specific threat types instead of giving a comprehensive detection system suitable for all threat types.

Figure 3c shows the performance comparison with the approaches of Oxford and CMU groups [2], [14]. The CMU work provides a representative detection scheme using machine learning-based methods to detect threat activities. They designed a bootstrap algorithm to train a robust machine learning model for anomaly detection on an unbalanced data set. Although the bootstrap algorithm could reduce the effect of lacking labeled malicious insiders and activities in training dataset, the detection model could only achieve 7% precision and 70% recall. This suggests that only using machine learning models may generate many false alerts and also miss some malicious insiders. The Oxford work is another representative work which compares users' daily activities with their historical records and compute the deviation to detect insider threats. Figure 3c shows that the approach could achieve 100% recall and 40% precision rate. However, nearly 40% precision means the generation of many false alerts. The result shows that machine learning based methods can accurately characterize and identify the difference between normal and anomalous behaviors compared with deviation based methods.

In our work, we design the two-step detection scheme, which draws upon the advantages of both machine learning and deviation based methods. The result in Figure 3c shows that *Warder* could largely reduce the false alerts and undetected behaviors using the two-step detection scheme, the false alerts and undetected behaviors from the first machine learning based classification module could be reduced to 0 by the second hypergraph based decision module. Moreover, *Warder* can dynamically add scenario feature trees and adapt to newly added threat scenario. Lastly, *Warder* provides a complete and practical solution for implementing a high-performance online insider threat detection system.

V. Conclusion

In this paper, we designed an online insider threat detection system, *Warder*, using multi-feature modeling and graph based visualization. In *Warder*, we combined multi-feature based anomaly detection with graph-based threat decision, in order to provide an extensible and high performing framework to facilitate the detection and monitoring of insider threats. Findings from evaluating *Warder* using the public CMU CERT v4.2 dataset demonstrated that by combining content based features with behavior based features to building users' daily activities, we can achieve high precision and recall rate for both known and unknown threat scenarios. For defined threat scenarios, *Warder* could accurately identify suspicious users and monitor them with hyper graph and feature tree. Moreover, the threat scenario feature trees can be easily formalized, which eases the implementation and extension of the system.

Future research include maintaining an up-to-date knowledge of the ground truth for known threat scenarios and design the scenario feature trees by experts. We also plan to append some probability model to improve the threat decision.

Acknowledgement

The support provided by China Scholarship Council (CSC) for Jiuming Chen's stay at UC Davis is acknowledged.

References

[1] H. Schulze, "Insider threat report: 2018." https://www.ca.com/content/dam/ca/us/files/ebook/insider-threat-report.pdf.

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

[3] T. Gu and P. Mohapatra, "Bf-iot: Securing the iot networks via fingerprinting-based device authentication," in *2018 IEEE 15th International Conference on Mobile Ad Hoc and Sensor Systems (MASS)*, pp. 254–262, Oct 2018.

[4] P. Dutta, G. Ryan, A. Zieba, and S. Stolfo, "Simulated user bots: Real time testing of insider threat detection systems," in *2018 IEEE Security and Privacy Workshops (SPW)*, pp. 228–236, IEEE, 2018.

[5] F. Toffalini, I. Homoliak, A. Harilal, A. Binder, and M. Ochoa, "Detection of masqueraders based on graph partitioning of file system access events," in *2018 IEEE Security and Privacy Workshops (SPW)*, pp. 217–227, IEEE, 2018.

[6] F. Greitzer, J. Purl, Y. M. Leong, and D. S. Becker, "Sofit: Sociotechnical and organizational factors for insider threat," in *2018 IEEE Security and Privacy Workshops (SPW)*, pp. 197–206, IEEE, 2018.

[7] M. Mylrea, S. N. G. Gourisetti, C. Larimer, and C. Noonan, "Insider threat cybersecurity framework webtool & methodology: Defending against complex cyber-physical threats," in *2018 IEEE Security and Privacy Workshops (SPW)*, pp. 207–216, IEEE, 2018.

[8] A. C. Lin and G. L. Peterson, "Activity pattern discovery from network captures," in *Security and Privacy Workshops (SPW), 2016 IEEE*, pp. 334–342, IEEE, 2016.

[9] M. Kazdagli, C. Caramanis, S. Shakkottai, and M. Tiwari, "The shape of alerts: Detecting malware using distributed detectors by robustly amplifying transient correlations," *arXiv preprint arXiv:1803.00883*, 2018.

[10] P. Chattopadhyay, L. Wang, and Y.-P. Tan, "Scenario-based insider threat detection from cyber activities," *IEEE Transactions on Computational Social Systems*, no. 99, pp. 1–16, 2018.

[11] J. Jiang, J. Chen, K.-K. R. Choo, K. Liu, C. Liu, M. Yu, and P. Mohapatra, "Prediction and detection of malicious insiders' motivation based on sentiment profile on webpages and emails," in *MILCOM 2018-2018 IEEE Military Communications Conference (MILCOM)*, pp. 1–6, IEEE, 2018.

[12] J. Glasser and B. Lindauer, "Bridging the gap: A pragmatic approach to generating insider threat data," in *Security and Privacy Workshops (SPW), 2013 IEEE*, pp. 98–104, IEEE, 2013.

[13] L. Zhang, Y. Gao, C. Hong, Y. Feng, J. Zhu, and D. Cai, "Feature correlation hypergraph: exploiting high-order potentials for multimodal recognition," *IEEE transactions on cybernetics*, vol. 44, no. 8, pp. 1408–1419, 2014.

[14] A. Azaria, A. Richardson, S. Kraus, and V. Subrahmanian, "Behavioral analysis of insider threat: A survey and bootstrapped prediction in imbalanced data," *IEEE Transactions on Computational Social Systems*, vol. 1, no. 2, pp. 135–155, 2014.

[15] Y. Hashem, H. Takabi, M. GhasemiGol, and R. Dantu, "Towards insider threat detection using psychophysiological signals," in *Proceedings of the 7th ACM CCS International Workshop on Managing Insider Security Threats*, pp. 71–74, ACM, 2015.

[16] M. Kandias, D. Gritzalis, V. Stavrou, and K. Nikoloulis, "Stress level detection via osn usage pattern and chronicity analysis: An osint threat intelligence module," *Computers & Security*, vol. 69, pp. 3–17, 2017.