International Journal of Advances in Engineering and Management (IJAEM)
Volume 2, Issue 7, pp: 831-835 www.ijaem.net ISSN: 2395-5252

An Ensemble-based Insider threat detection System for Stream Data

Ajayi Adebowale OAjayi Oluwabukola
Department of Computer ScienceDepartment of Computer Science
Babcock University, Ilishan-Remo, Ogun State, Nigeria Babcock University, Ilishan-Remo, Ogun State, Nigeria

Date of Submission: 01-10-2020 Date of Acceptance: 19-10-2020

ABSTRACT—Early detection of insider threats despite the large volumes of networked data and similarity of breach data points with legitimate network activity remains a viable research area in information security. Conceptualizing network data as stream data helps in applying stream analytics for effective handling of the velocity and volume of data prevalent on most networks nowadays.

This study adopted stream data methodologies for characterizing insider threat data as it is almost impossible to handle all the features in network data as its large size makes it impossible to store and the speed at which data points are collected makes it impossible to analyze all features at once. More importantly as attackers continually try to mimic legitimate actions, it is important to treat every new data point with a methodology that accommodates drifts in concepts.

This study presents an algorithm for quantized dictionary construction for a compressed and concise reference for user command sequences while taking into consideration the feature evolution and concept drift characteristics of stream data. The study recommends the application of stream analytics for tackling the insider threat menace.

Keywords—Insider threats, information security, stream data, quantized dictionary, stream analytics, concept drift, feature evolution

I. INTRODUCTION

Insider threat problem is one of the most difficult problems in computer security. While hackers and threats from outside of a company are often considered one of the biggest risks to an organization, corporation or business it may well be the insider threats which are the most problematic. The coronavirus pandemic has opened a great avenue for crackers to exploit the work from home situation and target unsuspecting employees to further their own malicious agendas. Losses due to insider threats (from malicious insiders or unsuspecting naïve ones) have been estimated to run into hundreds of millions of dollars (Ellen, 2020).

Traditionally, programs designed for insider threat detection can only respond based on data entered for each employee. This assigned specific access or operational protocols to provide specific information on which employees could access what information or data. From this set of protocols reports were generated showing insider threat detection, but only after the event. In addition, multiple false positive threats keeps getting generated for legitimate access to data or information. These inherent problems have led to a shift in focus to adopting a stream approach for early detection of insider threats.

Heraclitus famously said that "No man ever steps in the same river twice, for it is not the same river and he is not the same man". This saying aptly describes stream data and highlights the construct of concept shift. It is imperative to make quick assessments of input data in a network and necessary adjustments to detection boundaries to handle changes in both the data and the users whether malicious or not.

II. LITERATURE REVIEW

Insider threat research has attracted a great amount of attention in literature due to the gravity of the problem within many organizations. Around 2000, early workshops on insider threat highlighted various research issues surrounding the topic (Anderson, Bozek, Longstaff, Meitzler, Skroch, and Van, 2000), since then there has been a number of proposal to address these issues.

Various insider threat detection techniques have been proposed in literature. Although in reality there are no perfect detectors as the search for a holistic insider threat detection system remains elusive. Existing techniques for detecting adversarial insiders generally focuses on detecting insider in the act as opposed to proactive cues of the adversarial insider that can aid detection of such activities before the successful execution of such insider attacks. Malicious insider action can be potentially detected as anomalous activity on a network. Thompson (2004) detected anomalies in document accesses and

International Journal of Advances in Engineering and Management (IJAEM)
Volume 2, Issue 7, pp: 831-835 www.ijaem.net ISSN: 2395-5252

queries with respect to Hidden Markov Model of text content while Bradford & Hu (2005) modeled user processes and flags deviations from the model. Salem &Stolfo (2009) used machine learning to recognize malicious intent in information gathering commands. Some research augment this basic approach by introducing decoys onto the network to entrap adversial insider (Spitzner, 2003; Bowen, Hershkop, Keromytis & Stolfo, 2009). Moreover, different models of adversarial insiders have been developed in an effort to capture all characteristic data pertaining to insider threat detection. These models includes physical behaviors that are indicators of adversarial intents (Marbury et al, 2015), as well as variables related to personality, emotion and motivation (Herbig, 2008; Herbig & Wiskoff, 2002; Band et al, 2015; Keeney et al, 2015). Moreover, while all these models are valuable, none incorporated all the possible situational context variables, indicators and triggers. Some school of thought believe that such attributes are necessary to establish a connection between behavior and psychology. When building a model linking psychological variables and adversarial insiders there is substantial psychological research to draw upon. Specifically, years of research to define the taxonomy of personality attributes has led to the development of a general unifies structure of personality traits termed as the five factor model (Digman, 1990; Costa & McCrae, 1992). This model commonly known as "Big 5" identified the following general factors that represents the relationships among a host of more specific personality descriptors: Neuroticism, Extraversion, Openness,Conscientiousness and Agreeableness. Moreover, more recent works has established some links between personality and behavior through models that incorporate situational context variables, indicators and triggers in a Bayesian network designed to estimate the likelihood of the future behaviors (Sticha, Buede & Rees, 2005; Sticha, Buede & Rees, 2006).

III. INSIDER THREAT DETECTION FOR SEQUENCE DATA

A sequence is an ordered list of objects. In a sequence order matters, and hence, exactly the same elements can appear multiple times at different positions in the sequence (Qumruzzaman et al., 2013). For example, (A,G,C) is a sequence of letters with the letter 'U' first and 'D' last. This sequence differs from (C,G,A). The length of a sequence is defined as the number of ordered elements, sequence data can be finite or infinite in length. Infinite sequences are known as stream sequence data which is the focus of this work. Insider threat detection related sequence data is stream based in nature and these data may be gathered over time sometimes years. For this work continuous data stream will be converted into a number of chunks, each chunk will represent a week and contain the sequence in which the data arrived during that time period.

There are two possible causes of misclassification based on concept drift (Parveen et al. 2013).

Case 1: the decision boundary of the second chunk moves upwards compared to that of the first chunk. As a result, more normal data will be classified as anomalous by the decision boundary of the first chunk, thus FP will go up. A test point having true benign (normal) category classified as an anomalous by a classifier is known as a FP.

Case 2: the decision boundary of the third chunk moves downwards compared to that of the first chunk. So, more anomalous data will be classified as normal data by the decision boundary of the first chunk, thus FN will go up. A test point having true malicious category classified as benign by a classifier is known as a FN.

Most often the decision boundary of the current chunk may vary, which causes the decision boundary of the previous chunk to misclassify both normal and anomalous data. Therefore, both FP and FN may go up at the same time. This implies that a model built from a single chunk will not work. This motivated the adoption of adaptive learning and two approaches are exploited:

- Incremental Learning: A single dictionary is maintained. When a normative sequence pattern is learned from a chunk, it will be added to the dictionary. To find normative pattern, unsupervised stream based sequence learning (USSL) was be used.
- Ensemble Learning: A number of dictionaries are maintained. In the ensemble, k models are maintained and each model maintains a single dictionary. Unsupervised stream based sequence learning (USSL) to train models from an individual chunk. USSL identifies the normative patterns in the chunk and stores it in a quantized dictionary.

A. Unsupervised Stream Based Sequence Learning (USSL)

Normal user profiles are considered to be repetitive daily or weekly activities which are regular sequences of commands. These repetitive command sequences are called normative patterns and these patterns reveals the regular behavior of a user. When a user suddenly demonstrates unusual activities an alarm is flagged for potential insider threat.

Therefore, in order to identify an insider threat, a need to find normal user behavior is

International Journal of Advances in Engineering and Management (IJAEM)
Volume 2, Issue 7, pp: 831-835 www.ijaem.net ISSN: 2395-5252

required. For that, sequences of commands are collected and patterns are observed within these command sequences are identified in an unsupervised fashion. The unsupervised approach needs to identify normal user behavior in a single pass, one major challenge is the variability in length with these repetitive sequences. To combat this issue, a dictionary is generated which to contain combination of possible normative patterns existing in the gathered data stream. Potential variations that could emerge within the data include the commencement of new events, the omission or modification of existing events, or the reordering of events in the sequence e.g.,liftliftliftliftliftcomcomecomecomecome-come, is a sequence of commands represented by the alphabets given in a data stream. All patterns li,if,ft,tl, lif, ift, ftl, lift, iftletc., are considered as possible normative patterns (Parveen et al, 2013). However, the huge size of the dictionary presents another significant challenge. Figure 1 shows unsupervised stream based sequence learning from a chunk in an ensemble.

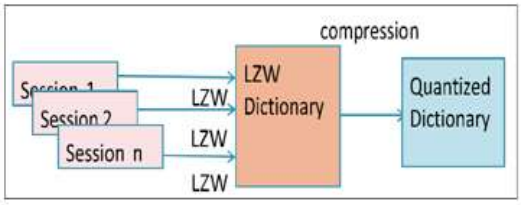

Fig 1. Unsupervised Stream based Sequence Learning (USSL) from a chunk in Ensemble based case (Parveen, 2013).

B. Construct the LZW Dictionary by Selecting the Patterns in the Data Stream

Initially, it is considered that insider data is not annotated. In other words, the possible sequence of future operations by a user is unknown. An, LZW algorithm (Ziv and Lempel, 1977) is used to extract the possible sequences that can be added to the dictionary. Unicode was used to index each command then the possible sequences are added to the dictionary. When a sequence is seen in the data stream for the second time, it will not be included in the LZW dictionary, instead the frequency is increased by 1 and it extends the pattern by concatenating it with the next character in the data stream, hence turning up a new pattern. The process continues until it reach the end of the current chunk.

C. Constructing the Quantized Dictionary

The longest and most frequent patterns in the LZW dictionary are kept and all their subsumed patterns are discarded. Algorithm 2 shows the steps on how a quantized dictionary is generated from LZW dictionary. Inputs of this algorithm are LZW dictionary D which contains a set of patterns P and its associated weight W.

Line 5 picks a pattern in the stream data, lines 7 - 9 finds all the closest patterns that are 1 edit distance away from it. Lines 13 - 16 keeps the pattern which has the highest weight multiplied by its length and then discards the other patterns. The steps are repeated until the longest, frequent pattern is identified. After that, a totally different pattern is taken and the steps are repeated until it has explored all the patterns in the dictionary. Hence, the iteration ends up with a more compact dictionary which will contain the useful and meaningful sequences. This dictionary is called quantized dictionary.

Algorithm 2 Quantized Dictionary

1: Input: D = {Pattern,Weight} (LZW Dictionary)

2: Output: QD (Quantized Dictionary)

3: Visited ← 0

4: while D ≠ 0 do
5: X ←Dj| j ∉Visited,Dj∈ D
6: Visited ← Visited ∪ j

7: for each pattern i in D do

8: if EditDistance(X, Di) = 1 then
9: P ← P ∪i

10: end if

11: end for

12: D←D−X

13: if P ≠ 0 then
14: X ←choose (argmaxi (wi×li))|li = Length(Pi), wi= Weight(Pi),Pi ∈ P
15: QD←QD∪X

16: D←D−P

17: end if
18: X ←Dj| j ∉Visited,Dj∈ D

19: Visited ← Visited ∪ j

20: end while

IV. TESTING ANONALY DETECTION

For a quantized dictionary, it is important to determine the sequence in the data stream which can raise potential threat. To formulate the problem, given a data stream S and Ensemble E where E = QD1,QD2,QD3,... and QDi = qdi1,qdi2,... , any pattern in the data stream is considered as an anomaly if it deviates from all the patterns qdij in E by more than X%.

To find the anomalies, matching patterns are identified and deleted from stream S, so that patterns from the data stream S that is an exact match or α edit distance away from any pattern, qdij in E is then considered as matching pattern. α can be ½, 1/3, or ¼ of the length of the particular pattern in qdij. The remaining pattens in the stream will then be considered as anomalies.

To identify the non- matching patterns in the stream S,a distance matrix L was computed which contained the edit distance between each pattern, qdij in E and the data stream S. When there is an exact match the proposed algorithm moves backwards exactly the length of qdij so as to find the starting point of that pattern in S and delete it from the data stream. But if there is an error in the match which is greater than (>) 0 but less than (<) α, the algorithm traverse either left or diagonal or up within the matrix according to which value is mentioned to find the starting point of that pattern in the data stream and when it is found, the pattern is deleted from the data stream and the remaining pattern will be considered as anomalous.

V. RESULTS

Insider threat detection remains a viable research area given the prevalence of insider attacks in recent years. Traditional learning methods use static data streams for evaluating and testing insider threat detection models. Few studies have conceptualised insider threat detection as a stream mining problem. This paper argues that the stationarity assumption represents a flaw in static stream approaches to insider threat detection as data pertaining to insider threat detection is usually unbounded and experiences concept drift and feature evolution characteristics of continuous stream data.

VI. CONCLUSION

Manually monitoring network activity in a very large organisation securely is a humanly impossible task. Vital tracking clues that can point to unusual activities are swiftly drowned out by the plethora of other information. In such a circumstance, data analytics can make a huge difference.

The increasing occurrence of insider attacks and inadequacy of available detection systems in combating the menace is a major motivation for this research.

This paper conceptualised insider threat problems as a stream mining problem that applies to continuous data stream and proposed an ensemble of unsupervised learning methods for efficiently detecting anomalies in stream data. An evolving ensemble of classifier models was used to cope with concept- drift as the behaviour of valid and invalid agents varies over time.

REFERENCE

[1]. Ellen S. (2020) Cybercrime ramps up amid coronavirus chaos, costing companies billions. Technology Executive council, July 29, 2020.
[2]. Anderson, R., Bozek, Longstaff, T., Meitzler, W., Skroch, M., and VanWyk, K., (2000). Research on mitigating the insider threat to information systems. In Proceedings of the Insider Workshop.
[3]. Thompson, P. (2004). Weak models for insider threat detection. Proceedings of the SPIE Vol. 5403, Sensors and Command, Control, Communications and Intelligence (C3I) Technologies for Homeland Security and Homeland Defense III.
[4]. Bradford, P., and Hu, N. (2005). A layered approach to insider threat detection and proactive forensics. ACSAC.
[5]. Salem, M., and Stolfo, S. (2009). Masquerade attack detection using a search behavior modeling approach. Columbia University Computer Science Department, Technical Report # cucs-027-09.
[6]. Spitzner, L. (2003). Honeypots: catching the insider threat. In Computer Security Applications Conference, 2003. Proceedings. 19th Annual, pages 170–179.
[7]. Bowen, B., Hershkop, S., Keromytis, A., and Stolfo, S. (2009). Baiting inside attackers using decoy documents. SecureComm.
[8]. Marbury, M., Chase, P., Cheikes, B., Brackney, D., Matzner, S., Hetherington, T., Wood, B., Sibley, C., Marin, J., Longstaff, T., Spitzner, L., Haile, J., Copeland, J., and Lewandowski, S. (2015). Analysis and Detection of Malicious Insiders. Technical Paper, Case #05-0207.
[9]. Herbig, K. (2008). Changes in espionage by Americans: 1947-2007. Department of Defense Technical Report 08-05.
[10]. Herbig, K., and Wiskoff, M. (2002). Espionage against the United States by American citizens 1947-2001. PERSEREC Technical Report 02-5.

[11]. Band, S., Capppelli, D., Fischer, L., Mooore, A., Shaw, E., and Trzeciak, R. (2015). Comparing Insider IT Sabotage and Espionage: A Model-Based Analysis. Technical Report CMU/SEI-2006-TR-026.
[12]. Keeney, M., Kowalski, E., Cappelli, D., Moore, A., Shimeall, T., and Rogers, S. (2015). Insider Threat Study: Comuter System Sabotage in Critical Infrastructure Sectors. U. S. Secret Service and CERT Coordination Center/SEI.
[13]. Digman, J. (1990). Personality structure: An emergence of the five factor model. Annual Review of Psychology, 41, 417-440.
[14]. Costa, Jr., and McCrae, R. (1992). Revised NEO Personality Inventory (NEO-PI-R) and NEO Five-Factor Inventory (NEO-FFI) manual. Odessa, FL: Psychological Assessment Resources.
[15]. Sticha, P., Buede D., and Rees, R. (2005). APOLLO: An analytical tool for predicting a subject's decision making. International Conference on Intelligence Analysis Proceedings. Bedford, MA: the MITRE Corporation.
[16]. Sticha, P., Buede D., and Rees, R. (2006). Bayesian model of the effect of personality in predicting decision maker behavior. Proceedings of the 22nd conference on uncertainty in artificial intelligence.
[17]. Parveen, P., McDaniel, N., Evans, J., Thuraisingham, B., Hamlen, K., and Khan, L. (2013). Evolving insider threat detection stream mining perspective. International Journal on Artificial Intelligence Tools (World Scientific Publishing) 22 (5), 1360013-1-1360013-24.