UNIVERSITY OF NAIROBI

SCHOOL OF COMPUTING AND INFORMATICS

Security threat detection in the workplace: A behaviour-based artificial intelligence approach

BY
OSORO STEPHINE RATEMO
P52/35555/2019

Submitted in partial fulfilment of the requirements for award of Msc. Computational Intelligence

DECLARATION

I hereby affirm that this documentation, as presented in this report, is entirely my own work, and has to the best of my knowledge, not been submitted to any other institution of higher learning.

Student: Osoro Stephine Ratemo **Reg. Number:** P52/35555/2019

Signature: …………………………… **Date:** 2nd December, 2021 ……………………….

This documentation has been submitted as a partial fulfilment of requirements for the Master of Science in Computational Intelligence of the University of Nairobi with my approval as the University Supervisor.

Supervisor: Dr. Elisha Odira Abade

Signature: ………………………… **Date:** 2nd December, 2021 ……………………...

Table of Contents

CHAPTER 1: INTRODUCTION

BACKGROUND

The ever-increasing improvements in communication and network technologies have resulted in great results for organizations and our general lives. For example, great improvements in cloud infrastructures and distributed computing have eliminated existing geographical boundaries making it feasible for a lot to be achieved but this has also made it possible for cyber-attacks to originate from any part of the world. This has made intrusion detection a very difficult job in Cybersecurity since a wide range of security anomalies can be initiated.

Accordingly, cyber defence techniques must be i) increasingly intuitive, iii) more adjustable, and ii) vigorous to auto detect any threats and eliminate them. To meet these needs, corporations are using Artificial Intelligence methods to watch and tackle cyber-criminal activities (Wiafe et al., 2020). This highlights the growing importance of AI techniques in Cyber security.

Also based on recent research related to Cybersecurity, emails, and the internet browser activities are the most difficult to protect. According to these reports also, researchers have determined that almost half (49%) of all security incidents are caused by lack of end-user compliance (Arash et al., 2018). In the era of such arising security anomalies, having an intelligent pre-warning tool is key. One key gap in most existing security systems is that security teams in organizations usually focus on keeping their system secure without taking into account user experience of their end users who use the systems. Hence some users might not be able to uphold correctly the standards set by such security teams hence sometimes leaving loopholes that attackers might use to breach their security systems. One technique to overcome such attacks is by using intelligent behavior-based security systems.

Using behavior-based artificial intelligence to profile normal users behavior will help raise an alarm when security anomalies take place. User Entity Behavior Analysis, created by Gartner, is one such technology that uses network usage patterns of end users, and then applies machine learning algorithms to detect security threats from those learnt patterns. A research done by Digital Guardian, reveals that in 2020, organizations that don't have such security automation tools will experience a higher cost, by $3.58 billion, than those with security automation. This is how expensive a data breach is. The proposed model will use a multimodal-based UEBA to create a security profile of end user patterns using Convolutional Neural Network (CNN), which will help detect any security anomalies.

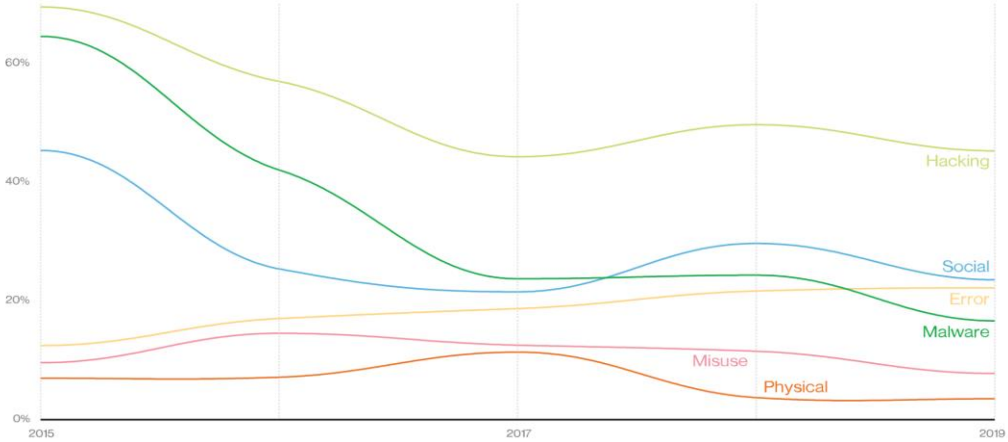
In this research, we shall use log data from University of Nairobi's (UoN's) servers, and the United States Computer Emergency Readiness team (CERT) data set. We shall aggregate the log data from these different sources using Fluentd. Fluentd is a local log aggregator that gathers all node logs and forwards them to a centralized storage facility. One of its key advantages is that it has low memory requirements and has a high throughput hence reducing system utilization. We shall then normalize all the data in the centralized storage to a "normal form" to improve its data integrity and reduce any redundancy in the data. This will ensure all our data, in all the records, reads and appears the same way.

Once done with the data pre-processing, The first part of our analysis is log correlation, which is looking for patterns in log events that are not evident in the separate log files. This connects the dots on related yet heterogenous data. Once this has been done, we shall then feed our correlated data to our deep learning, CNN model. Our deep CNN model will be an automatic log classification system that uses deep learning methods to predict the log event category that the collected logs belong to and allocate a given score to each classification prediction. This will help detect any anomaly which is not according to the profiled user patterns learned by our model.

We shall then visualize this learnt data using D3 charts for easier monitoring of security events and also help to identify data anomalies in the network infrastructure more easily visually.

PROBLEM STATEMENT

Security threats are continually evolving which makes them nearly impractical to be identified using traditional cybersecurity controls. In a recent security breach report, it was determined that the average median time between intrusion of a cyber attack and it's detection is about 14 days (Statista, 2021). Also, most cyber attacks take a few minutes with 68% of them going undiscovered and only 3% of them discovered as they are happening. This has led to an increased use of machine learning in security systems to help reveal cyber criminal patterns and to be able to reveal such activities as they happen. By 2019, the growing adoption of machine learning and advanced analytics helped reduce cyber security threats by 20% as seen



below.

OBJECTIVES

Objectives of this research:

1. To investigate how logs from multiple devices and applications could be aggregated in one normalized centralized storage.
2. To determine user characteristics important in cyber security.
3. Tie the user characteristics to the collected logs and extract these characteristics from the logs.
4. To develop a prediction classification model that can profile different normal end-user's usage patterns from the logs.
5. To evaluate the model and create dashboards for it.

Research Questions:

1. How can we aggregate logs for multiple sources to an aggregated normalized centralized data storage?
2. What are the common user characteristics that we can use to profile end users?
3. How can we retrieve user characteristics from logs and use them to create a security profile of them?

SIGNIFICANCE OF THE STUDY

This project will utilize deep learning techniques to create a behavior profile of each user, in particular LSTMs.

LSTM is a type of supervised machine-learning technique that is majorly applied in Natural language Processing and speech recognition. LSTMs are drilled to get the usual sequences then use the past to forecast the sequences of the next sequence state. The difference between the given prediction and the actual sequence is an proof of security threat identification in the system.

This study will give evidence that monitoring behavior of users and entities will enable us to detect most forms of traditional threats that cannot be detected using signature based techniques such as antiviruses and firewalls. Since we monitor how our users behave normally and any deviation from the normal behavior is flagged as an anomaly for the cyber security team to investigate further. This will help security threats to be discovered quickly on the go as they happen hence saving the loss that are associated with cyber security threats.

ASSUMPTIONS

1. Users tend to have a internet usage behavioral pattern
2. Users will use the same device to perform work/school work during the entire time of this research.

JUSTIFICATION

All the existing log analysis tools that use UEBA are commercial and their pricing is relatively high. Creating an open source UEBA tool will be a great addition to the open source community. Also, the ability to dynamically customize the model is lacking in all current existing UEBA tools. In this research, we shall make it possible for users/organizations to customize the model according to their organization structures which will make our tool more dynamic.

CHAPTER 2: LITERATURE REVIEW

In one of the recent research, (Barbara Filkins et. al, 2015), it states that most tools use either of these three security identification mechanisms - Signature-Based, Continuous System Health Monitoring or finally Anomaly (Behavior) Based detection.

a. Signature based detection.

This uses rules to detect anomalies by observing event patterns that have been documented before. The resulting signatures are compared to stored signatures stored in a signature database and if any match has been found, it will fire an alert that an security intrusion is/has taken place. Main advantage of this thread detection methodology is it produces less false positives than previous traditional methods and the disadvantage is that it can only identify anomalies that a defined signature is known and stored on the signature databases.

b. Anomaly based detection.

This detection mechanism involves creating a machine learning model of usual normal behavioral patterns for the usual system, application then the end users and any differences from this pattern is classified as an anomaly threat.
Advantage of this method is it detects intrusions without knowing the gravity for it while the disadvantage of this mechanism is that training the model in a very dynamic environment can be very challenging.

c. Continuous System Health Monitoring

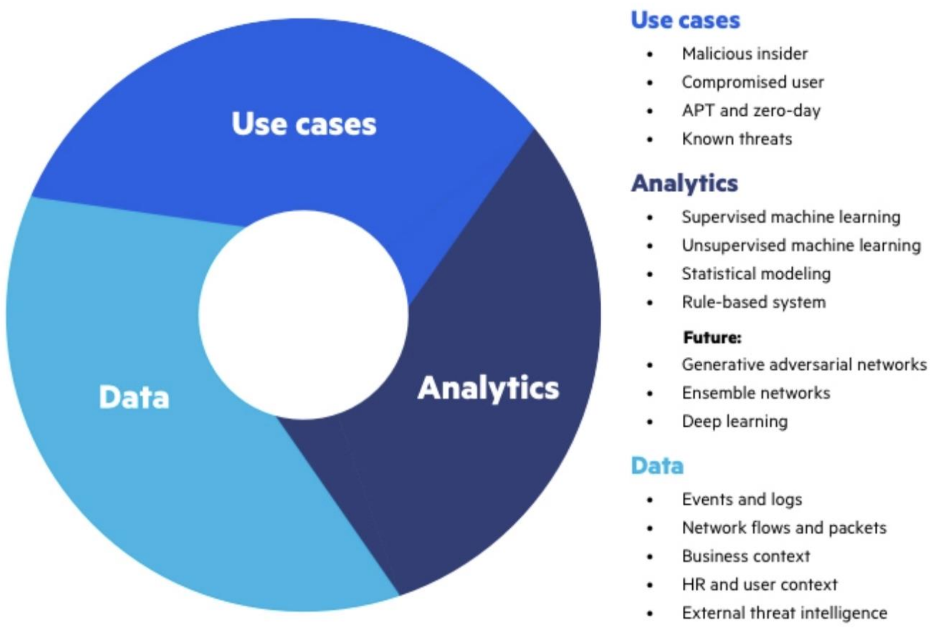
This traces system performance and health metrics to detect intrusion. For example, when resource usage such as RAM or CPU spikes abnormally over time, it might fire an alarm that an intrusion might be taking place. This may also involve learning the network protocols normally used, the ports accessed and the frequency and bandwidth utilization over a certain period.

USER ENTITY BEHAVIOR ANALYTICS (UEBA)

Is the behavioral analysis that provides insight by investigating the network and application logs that end-users create as they perform actions in their systems. Instead of tracking tools, UEBA tracks the end-user. (Wikipedia, 2021)

Gartner's interpretation consists of these three primary attributes of UEBA tools:

1. Use cases - they can disclose the users and entities behavior in a given network
2. Data sources - they can take data from any data repository such as data lakes and warehouses, Security Information and Event Management.
3. Analytics - UEBA tools separates security anomalies using analytical methods such as rules, threat signatures, statistical models and machine learning.



Three pillars of UEBA

MACHINE LEARNING TECHNIQUES.

A lot of machine learning techniques could be used to solve this problem. Some of the common approaches that have been used are as follows:

Classification Techniques

In machine learning, classification is a supervised machine learning technique that requires labeled data to supervise the learning process then assign a class to the input data.
The benefits of supervised learning methods is that they are fast and efficient, as they get the ‘accurate’ answers in the training phase, that is, they learn faster with clear feedback. But their main disadvantage is it’s difficult to get adequate, reliable, labeled data that contains no fault. Also, supervised machine learning techniques cannot predict totally unknown problems.

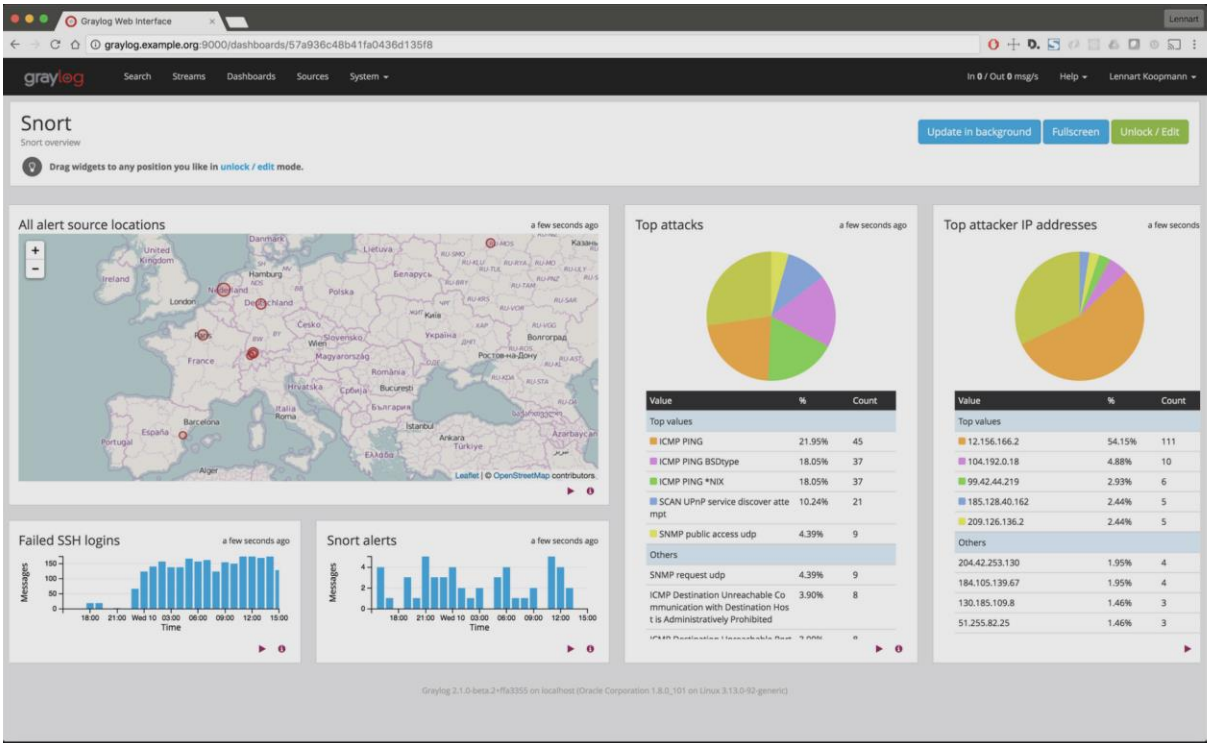
Clustering Techniques

Since getting fully clean data is almost impossible, most researchers usually use unsupervised learning. Many of them use clustering techniques. Clustering (Jiawei et. al, 2001) is the process where elements aggregated in the same cluster will have high similarities to each other but will be very dissimilar to elements in other clusters or classes. K-means is a classical partitioning technique that could be used for this but the major setback of K-means is that number of clusters (parameter K) needs to be predetermined by a human. Only two classes of either normal or anomalous logs are desired for this study. How to map the clusters of K-means into two classes is a problem that needs to be solved in our study using a given machine learning algorithm.

EXISTING SYSTEMS

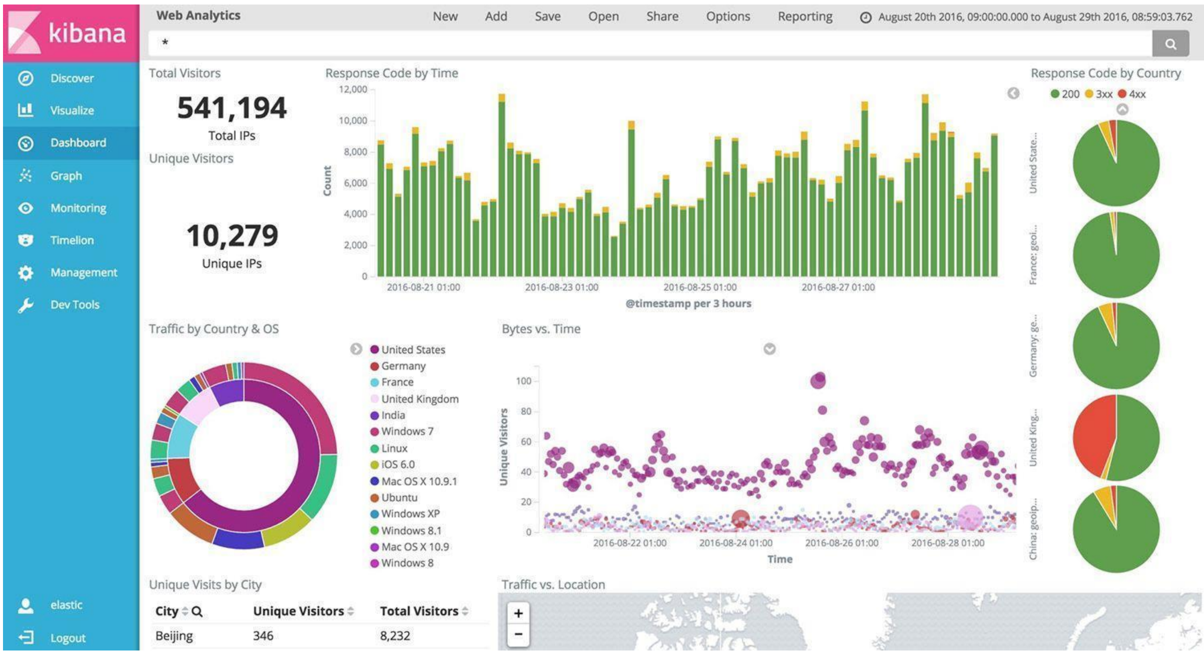
Graylog

Graylog is a centralized, enterprise log management software that stores, captures and enables real-time analysis of big data of machine data. It’s well known for its usability, scalability, and comprehensive access to complete data.



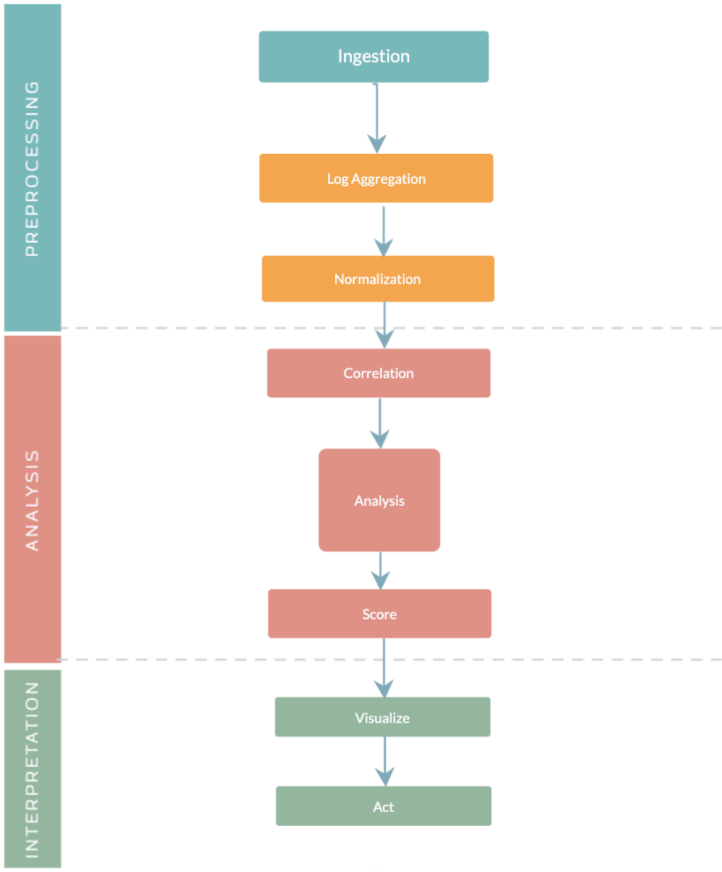
Kibana

Kibana is a data search and visualization tool used for logs visualization, operational intelligence problems and application software monitoring. Kibana is usually used with other combination of tools such as elastic search and logstash to form the well known ELK stack for log management.



CONCEPTUAL FRAMEWORK

The figure below illustrates the process of creating a behavioural profile of each user from log data. We shall ingest the United States Computer Emergency Readiness Team (CERT) dataset and data from the UoN server logs. We shall then aggregate all the data and normalize it to one form, then do log correlation and analysis using deep learning and create scores that will form the user's normal behavior. Any behavior that deviates from this will be considered as a security anomaly.



RESEARCH GAP

Unfortunately, most UEBA tools alone may be too illogical, leaving significant gaps in coverage for the companies using them.They often need tweaking by machine learning professionals to completely fit in with a particular company's needs and this can take a lot of time to complete. In a recent study by Crowdflower, more than half of a data scientist's time is spent collecting, cleaning, labeling and organizing data. Even with this noticeable effort, there's a high probability for the application to give false positives due to heterogeneous datasets and poor data training. Instead, augmenting such a UEBA solution with the file- and user-activity-based insider threat management implementation may infuse some of these gaps where machine learning fails today.

Also most UEBA tools are expensive and resource intensive, needing a lot of computational resources in order to work well. Creating an open source, less computational resourceful tool will be a key improvement to UEBA tools.

CHAPTER 3: METHODOLOGY

INTRODUCTION

This chapter describes how this research will be conducted using a behavior-based artificial intelligence approach and the procedures which will be followed to come up with the research results. We shall use behavior-based artificial intelligence which toils to identify the differences in the behavior between the endpoints (users) and which may stipulate a security anomaly.

RESEARCH DESIGN

Research design is the cogent steps taken to connect the objectives, research questions to the data collection, analysis and interpretation.

We shall conduct this study using quantitative research. Quantitative research is the process of collecting and using statistical techniques to analyze numerical data.

We shall use correlational research design, which analyzes the relation between variables without one manipulating any of the variables. It also defines the power of the relationship between the variables, in which our case is the behavioral profile of users and entities. Understanding the variable relationship will help predict security threats considering what is already known.

STUDY POPULATION AND SAMPLING METHODS

In this research, we shall use log data from University of Nairobi's (UoN's) servers, and the United States Computer Emergency Readiness Team: CERT data set (CERT). The CERT open dataset is created with diverse models including behavior, psychometric, and topic models by the Carnegie Mellon University CERT team and is the utmost famous dataset for insider based threat identification. I will choose version 4.2 of this open-dataset to evaluate this system since version 4.2 has been documented to consist a greater number of insider anomaly incidents than other releases, which makes it great to use for this purpose.

We shall use the quota sampling method which is a non-random sampling method where we use some predetermined characteristics to choose data so that the total sample will have equal spread of properties as the overall data. This will enable us to train our model with data which includes anomalies hence our model will have higher accuracies of determining anomalies from normal behavior.

Data Collection

Data collection is the process of gathering, organizing data or information of a particular interest. Our primary data source will be the server and network logs from the University of Nairobi (UoN) servers. I shall install fluentd agents on a couple of servers which shall be periodically checking for new logs and syncing the logs to the central database. The logs files of interest are sys logs, auth logs and nginx/apache logs which will contain all the security information that we need. This study will also use secondary data from the CERT dataset to complement the UoN data and CERT dataset contains security threat incidents hence it will be useful in training our model to accurately determine anomalies.

Pretesting (Validity and Reliability)

Validity and reliability is important for any given research design. Our data is reliable since the CERT dataset is a well known dataset and UoN servers contain log information from real systems. I shall conduct content validity to ensure all the data features required to create a user profile are included in the data and come up with a method of filling in missing data.

Data Analysis Method and the Model

Data analysis aims to look for relationships, patterns or themes in the collected data. We shall use quantitative data analysis which utilizes statistical mathematical approaches to examine our data to find relationships and patterns that will ultimately be classified as the user/entity behavior. Before we ultimately investigate our analyze our data using LSTM, the data shall pass through a series of steps listed below to ingest, aggregate, normalize the data, do some correlation on the data then finally analyze the data using machine learning to create a user/entity profile in our research and be able to predict if a given behavior is an anomaly when it deviates from the normal behavior.

A. Log ingestion

Ingestion is the process of refining and uploading data from different sources such as servers, applications and different platforms.
Almost all log strings consist of the three components listed below but only the message is mostly required.

1. Message

Message is a string that contains the explanatory piece of a log line and is normally preceded by a level and timestamp. A log message will contain a combination of variable and static substrings and allows for easy human interpretation.

2. Timestamp

Timestamp is required for all ingested log lines that we shall ingest. As a general rule, most timestamps follow the ISO 8601 format.

3. Log level

The log level illustrates the severity of the content in it. Common log levels are:

- CRITICAL
- ERROR
- DEBUG
- EMERGENCY
- INFO
- FATAL

- SEVERE
- ALERT
- TRACE
- WARN

Source information

Source information is also ingested and this includes the hostname, which is the source of the log line picked above. Other optional source information is also to be picked which are IP address and MAC address.

B. Log Aggregation

Log aggregation is the process of aggregating log data from different sources into a centralized storage where it can be cleaned, analyzed and used to extract meaningful insights.
We shall use Fluentd for the log aggregation. Fluentd is a local log aggregator that gathers all node logs and forwards them to a centralized storage facility. One of its key advantages is that it has low memory requirements and has a high throughput hence reducing system utilization.

C. Normalization

We shall then normalize all the data in the centralized storage to a “normal form” in order to improve its data integrity and reduce any redundancy in the data. This will ensure all our data, in all records, reads and looks the same way.
An example of normalization is how different tools respond to a successful authentication in the logs. Some applications will indicate “login successful”, another application will indicate “user authenticated”. These two pieces of information speak the same thing and would need to be normalized as one thing to increase the data integrity.

D. Correlation

Log correlation is looking for patterns in log events that are not evident in the separate log files. This connects the dots on related yet heterogenous data. Some cyber attacks are not noticeable when one log source is investigated hence the need to connect the dots between different log sources to reduces the number of false positives and provides a powerful confirmation that a security anomaly has indeed taken place.This shows the importance of log correlation.

E. Analysis

For the analysis, I'll use deep learning which utilizes Artificial Neural Networks (ANN), which are created to replicate the human brain neurons.

The advantage of deep learning models is that they have a higher better accuracy than traditional machine learning techniques, but need a lot of data to train to achieve their high accuracy. The figure below shows the accuracy of Deep Learning compared to other traditional machine learning methodologies.

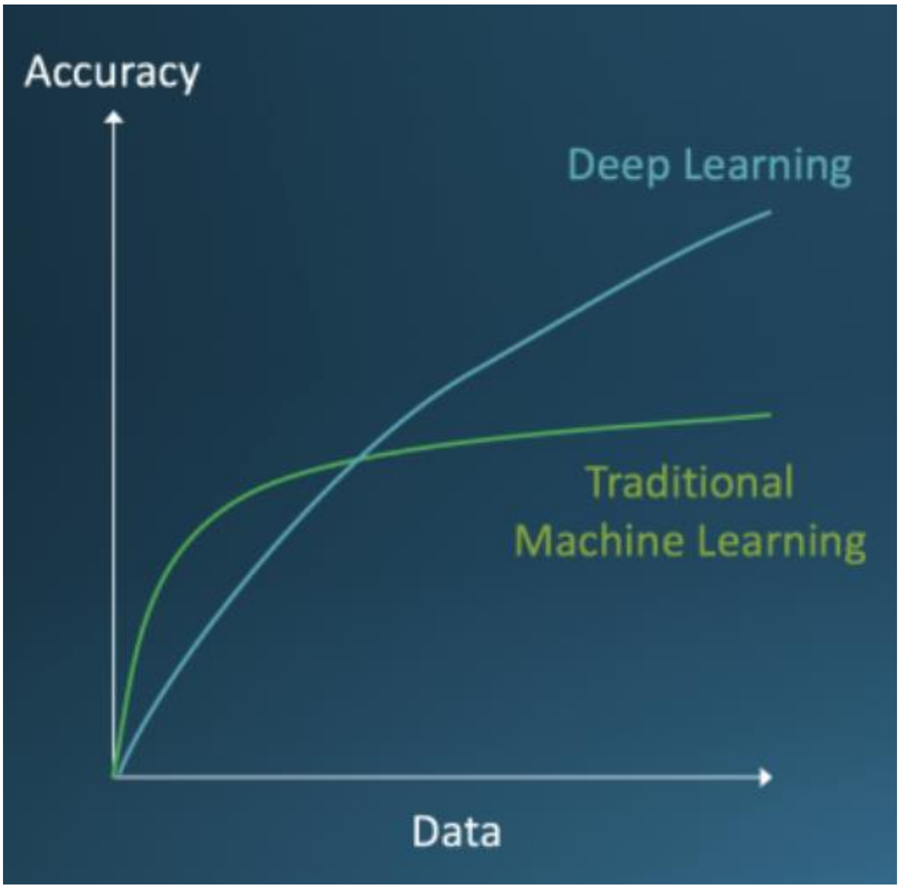


Image Source: IBM – Deep Learning Trends

LSTM

LSTM (Long Short Term Memory) is a type of supervised machine-learning technique that is majorly applied in Natural language Processing and speech recognition. LSTMs are drilled to get the usual sequences then use the past to forecast the sequences of the next sequence state.

The difference between the given prediction and the actual sequence is an proof of security threat identification in the system.

In this research, we shall use N given days of data to do the next sequence state prediction, taking into account that there's a difference in dimension of every user's action sequence. To illustrate this, the user's activities or action can be constituted as: log on, email, drive connect, internet browsing, drive disconnect, internet browsing,, USB connect,,,, log out - and add into our LSTMs. We shall use LSTM in this study due to its advantages.

ETHICAL CONSIDERATION

This research will obtain permission from UoN to use the server log data to create the user's profiles. To protect user privacy and anonymity, I shall code all the data before feeding it to the system to prevent tracing back the data to given users.

EXPECTED RESULTS/OUTPUTS

In this study, we shall investigate the challenge of catching insider initiated abnormal behaviours by using three viewpoints: users action and the user sequences of action, and the role features based on the users roles.. The main contribution of this study is to create and assess a novel tool that utilizes multiple machine learning algorithms to learn a given user's behavior patterns so as to detect any strange security behaviours.
To detect abnormal behaviours more precisely and to overcome false positives and false negatives, we'll use a Multi Layer Perceptron to make the final thorough decision by utilizing deviations produced by LSTM from every type of the three viewpoints mentioned above. Accordingly, the output of our Multi model Based System will show its prowess to identify insider initiated abnormal behaviours.

RESOURCES

Resources required are:

1. Microsoft Visual Code IDE for coding the system.
2. Django 3.1.7 for developing the system.

In addition to this, I shall need approximately 100 USD that will serve as data and transport cost.

CHAPTER 4: RESULT AND DISCUSSION

4.1 DESIGN AND ANALYSIS

To do the security threat detection, three key deep learning models are needed, one for action features, another one for action sequences, and the last one for role features, to create our multi-model-based system. (Zhihong et al., 2020)

At the beginning of this workflow, employees in the company dataset I'm using are grouped in accordance with their job roles, such as human resource employees, engineers, executives and software engineers. One assumption is that users in the same group will have the same characteristics since they do the same job and we can extract job role features through their daily job behaviour and through this we can be able to achieve anomaly detection through detection behavior that deviates from their normal job behaviors. For example, human resource departments are inclined to read resume , create meetings during most of their working hours. So if our system sees a user accessing files which they are not usually accessing and downloading them to external drives, this will be marked as suspicious and security personnels should investigate this.

In order to train our models for each type of unique user, past historical data is required. We shall use three key feature categories to extract useful information from this historical user data, which are action features and sequences, and role features. Then these three deep learning models are devised to learn which features will comprise the user's normal behaviours by learning the historical data and making the next state predictions for each feature. Then anomalous behavior can be detected when features deviate from the predictions that match a normal user behavior.

Feature extraction

In this step, we extract a feature map based on the users:

1. Logon data
2. Device data
3. Internet browsing
4. Emails

We identify relative information from log data sources and transform them into a normalized form that deep learning algorithms can predict deviations from normal user's behavior when an suspicious behavior occurs. (Zhihong et al., 2020).

- **Feature actions**: These are normalized numeric features obtained from the data to represent the user's daily activities for each particular time period.
- **action sequence**: This is a sequence of a user's recorded activities that's organized by time.

LSTM

LSTM (Long Short Term Memory) is a type of supervised machine-learning technique that is majorly applied in Natural language Processing and speech recognition. LSTMs are drilled to get the usual sequences then use the past to forecast the sequences of the next sequence state. The difference between the given prediction and the actual sequence is an proof of security threat identification in the system.

In this research, we shall use N given days of data to do the next sequence state prediction, taking into account that there's a difference in dimension of every user's action sequence. To illustrate this, the user's activities or actions can be constituted as: log on, email, drive connect, internet browsing, drive disconnect, internet browsing,, USB connect,,, log out - and add into our LSTMs. We shall use LSTM in this study due to its advantages.
Our model consists of two LSTM layers (100 and 160 units separately), then an activation layer of "tanh" after each layer, a 37-unit dense layer and a "relu" activation layer. Conventionally, 2 layers have shown to be enough to detect more complex features. Ideally, more LSTM layers are better but become more difficult to train.

4.2 IMPLEMENTATION

In this research, we shall use log data from University of Nairobi's (UoN's) servers, and the United States Computer Emergency Readiness Team: CERT data set (CERT). The CERT open dataset is created with diverse models including behavior, psychometric, and topic models by the Carnegie Mellon University CERT team and is the utmost famous dataset for insider based threat identification. I will choose version 4.2 of this open-dataset to evaluate this system since version 4.2 has been documented to consist of a greater number of insider anomaly incidents than other releases, which makes it great to use for this purpose.

I used weighted deviation degree (WDD) to measure the deviational difference between the prediction and real features in order to do anomaly detection. For this scenario, some deviation may not indicate of abnormal behaviors. Therefore, we defined the WDD, which weighs the squared error linearly according to a weighted value. The WDD can be formulated as:

$$\mathrm{WDD} = \frac{1}{|V|} \sum_{\forall\, y \in V} w(y - \hat{y})^2,$$

where:

1. y is a single feature belonging to V
2. yˆ is the same feature as y but belongs to the predicted feature map
3. V is the set of all features in the real feature map
4. and w is a specially designed value according to the feature y.

In this research, "ProductionLineWorker" users were used for testing our model. Their features are presented in the figure below.

Table 1. Comprehensive Features

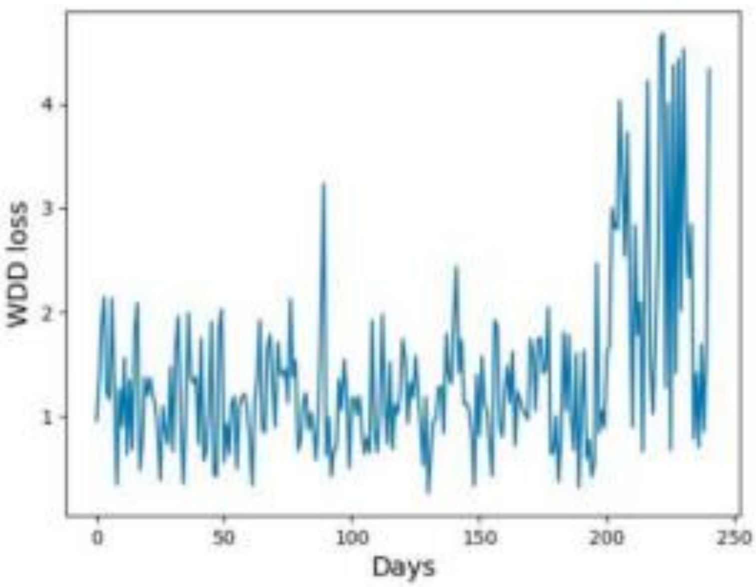
Action features
Weekday log on/log off (users logged on or logged off during working time)
After-working log on (users logged on or logged off beyond working time),
Weekend log on (users logged on or logged off during the weekend)
Online time (the online time)
Num. device (number of thumb drives used)
Files exe copy (users copied exe files to thumb drives)
Files jpg copy (users copied jpg files to thumb drives)
Files txt/doc/pdf copy (users copied txt/doc/pdf files to thumb drives)
Files zip copy (users copied zip files to thumb drives);
Num. emails sending (number of emails sent)
Internal email sends (users sent emails by using company emails)
Num. Internal email receive (number of receivers' emails that are company emails)
Num external email receive (number of receivers' emails that are other emails)
Size of emails (the size of emails), Num. attachments (the number of attachments)
Num. websites (times of visiting websites)
Num. career sites (number of visits to job websites)
Num. news sites (number of visits to news websites)
Num. tech sites (number of visits to techniques websites)
Action sequence
Types of actions include log on, log off, http, device connect, device disconnect, email.
Role features
The average of features selected from action features of all employees in this group. These features include Weekday log on/log off, After-working log on, Weekend log on, Online time, Num. emails sending, Internal email send, Num. Internal email receive, Size of emails, Num. websites, and Num. tech sites.

RESULTS

In this section, I will illustrate how I used LSTM on the CERT dataset to train the anomaly detection system. To train the models, I used LSTM on four day's features to predict features of the fifth day then calculate the deviations between predictions and the real features. Then the error between the predictions and the real fifth day of data from LSTM are calculated and optimized in this training phase.

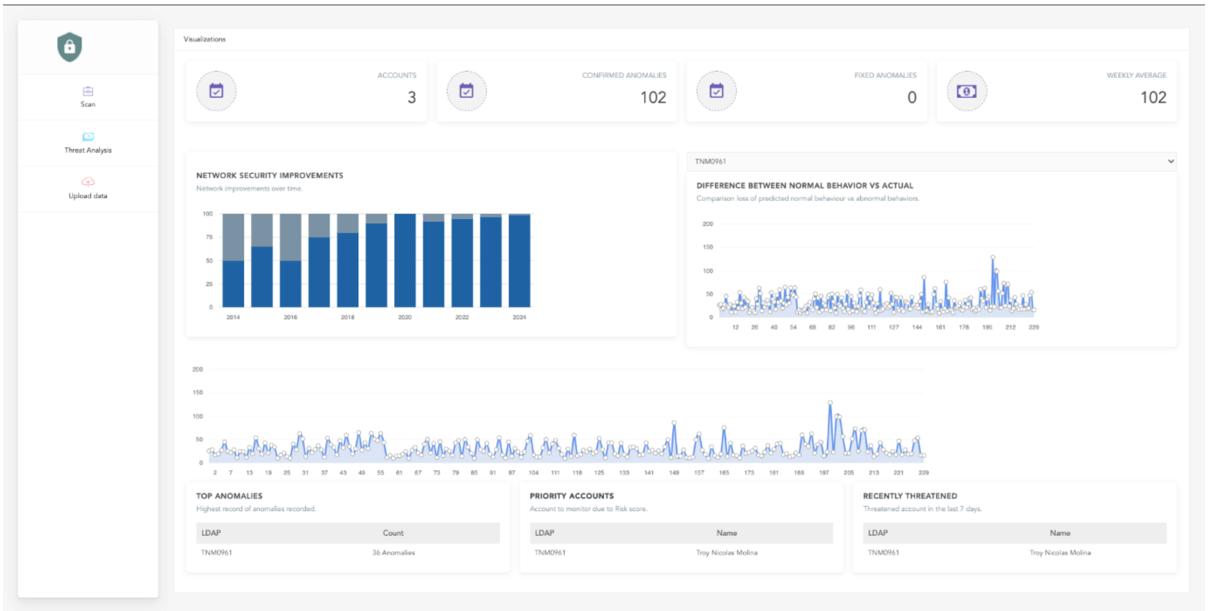
.

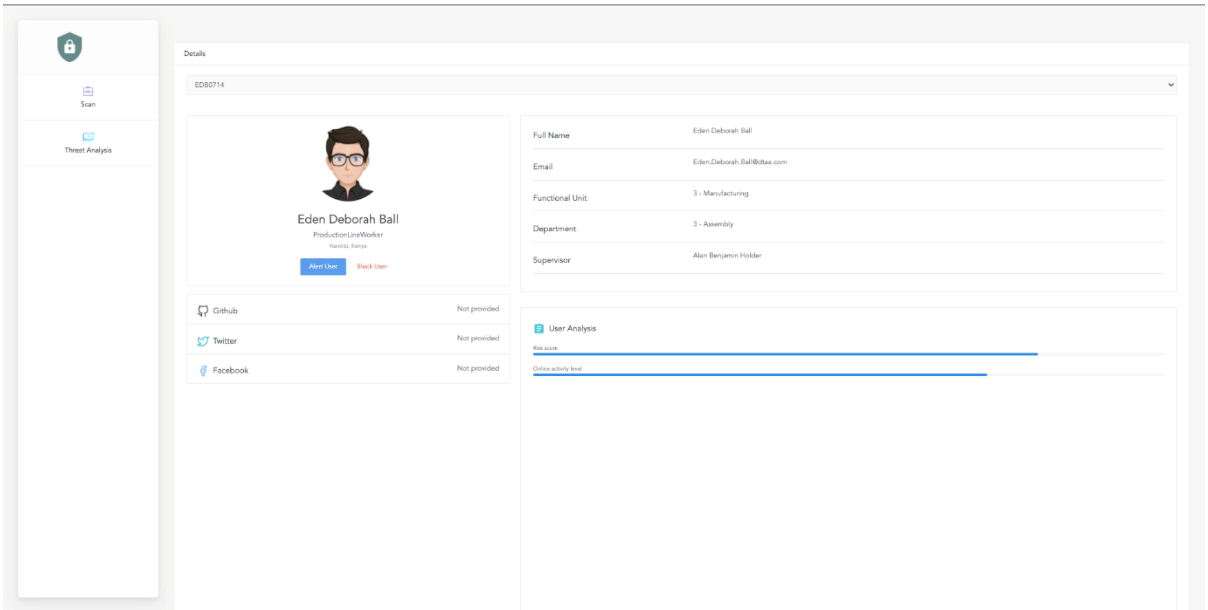
As indicated in the figure below, in the first 200 days, the deviation between the user's daily features and the standard role features fluctuated within the range of 0–2, but 200 days later, the user started to exhibit some suspicious behaviour, and there was a significant difference during the training phase. This indicates that the role features can reflect whether the user's daily behaviour conforms to normalcy to some extent and can be indicative of abnormal behaviour detection.

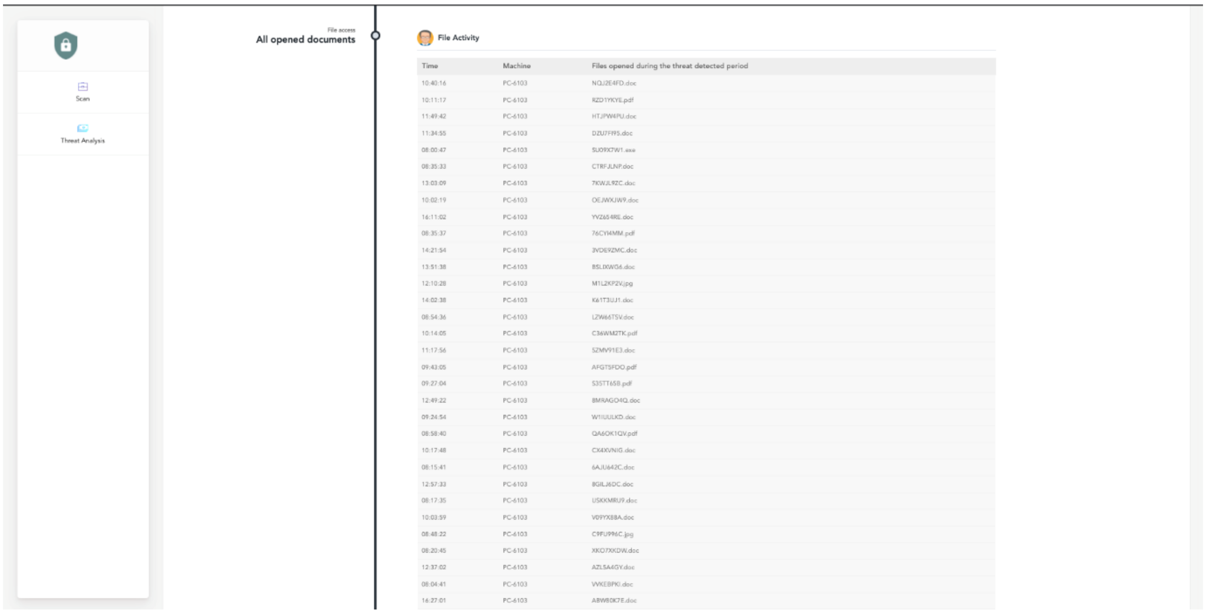


In order to accurately raise alarm when abnormal behaviour is detected, I use MLP to learn the relationship between the three deviations. I concatenated the three types of behavior deviations and used them to train the MLP to learn these potential links. Finally, the MLP after the training can accurately determine whether the user has abnormal behaviours on a particular day.

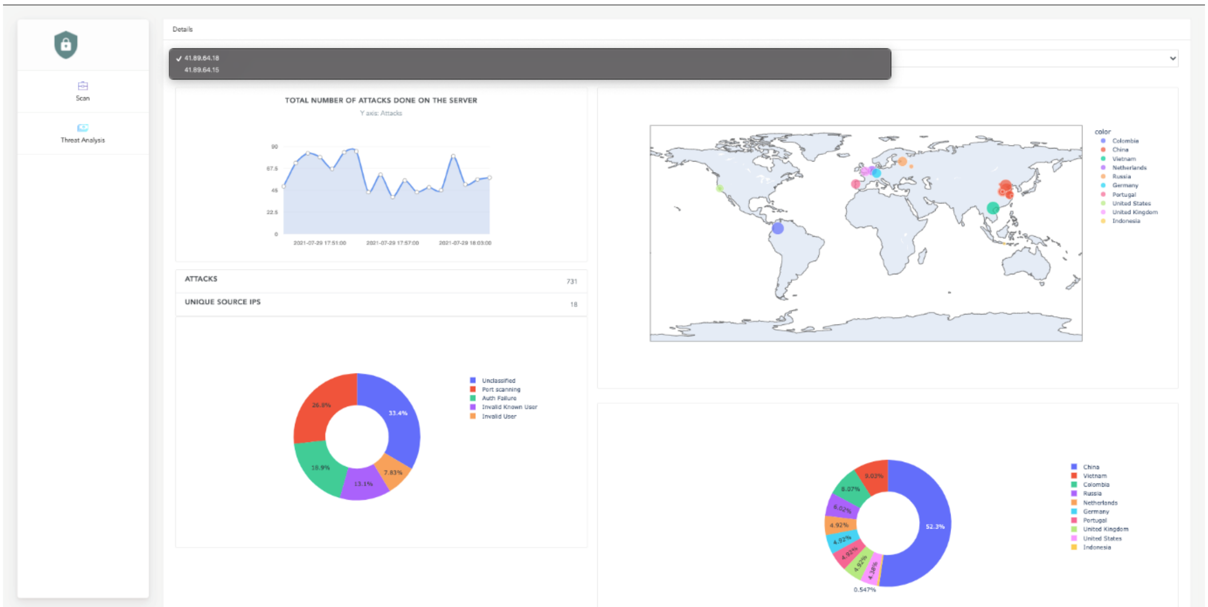
Below are some of the screenshots of the system:







Time	Machine	Files opened during the threat detected period
10:40:16	PC-6103	NQJ2E4FD.doc
10:11:17	PC-6103	RZD1YKYE.pdf
11:49:42	PC-6103	HTJPW4PU.doc
11:34:55	PC-6103	DZU7FI95.doc
08:00:47	PC-6103	5U09X7W1.exe
08:35:33	PC-6103	CTRFJLNP.doc
13:03:09	PC-6103	7KWJL9ZC.doc
10:02:19	PC-6103	OEJWXJW9.doc
16:11:02	PC-6103	YVZ654RE.doc
08:35:37	PC-6103	76CYI4MM.pdf
14:21:54	PC-6103	3VDE9ZMC.doc
13:51:38	PC-6103	BSLIXWG6.doc
12:10:28	PC-6103	M1L2KP2V.jpg
14:02:38	PC-6103	K61T3UJ1.doc
08:54:36	PC-6103	LZW66TSV.doc
10:14:05	PC-6103	C36WM2TK.pdf
11:17:56	PC-6103	5ZMV91E3.doc
09:43:05	PC-6103	AFGT5FDO.pdf
09:27:04	PC-6103	535TT65B.pdf
12:49:22	PC-6103	8MRAGO4Q.doc
09:24:54	PC-6103	W1IUULKD.doc
08:58:40	PC-6103	QA6OK1QV.pdf
10:17:48	PC-6103	CX4XVNIG.doc
08:15:41	PC-6103	6AJU642C.doc
12:57:33	PC-6103	8GILJ6DC.doc
08:17:35	PC-6103	USKKMRU9.doc
10:03:59	PC-6103	V09YX88A.doc
08:48:22	PC-6103	C9FU996C.jpg
08:20:45	PC-6103	XKO7XKDW.doc
12:37:02	PC-6103	AZL5A4GY.doc
08:04:41	PC-6103	VVKEBPKI.doc
16:27:01	PC-6103	ABW80K7E.doc



4.3 DISCUSSION AND CONCLUSION

In this research, we investigated the problem of detecting abnormal behaviours from insiders by considering three angles: action features and sequences, and role features. The main contribution is the development and evaluation of a multi model based system that learns a user's normal pattern of behaviours to identify suspicious behaviours.

Even though every type of features has the potential for predicting anomalous behaviour, there are false positives and false negatives when using every single feature to identify such deviations. To accurately do anomaly detection, I sued an MLP to perform a comprehensive decision by taking advantage of deviations from every type of feature. Consequently, experimental results of the MBS show its promising ability to detect abnormal behaviours from insiders.

Limitations

The most significant limitation is assuming the user's normal behavior is represented on the historical data used to train the models. This assumption will lead to wrong algorithm detections if let's say a user launched attacks from the very beginning.

Also, since different operating systems, hosts and applications generate their own logs, several issues arise when performing log analysis. This is because different log sources have inconsistent log content, inconsistent formats and inconsistent timestamps. (Kahonge et al., 2012)

REFERENCES

1. Manya Ali Salitin, Ali Zolait (2018). The role of User Entity Behavior Analytics to detect network attacks in real time.
2. Verizon (2020). Data Breach Investigations Report.
3. Wikipedia (2021). Retrieved from https://en.wikipedia.org/wiki/User_behavior_analytics
4. Imperva (2020). Retrieved from https://www.imperva.com/learn/data-security/ueba-user-and-entity-behavior-analytics/
5. Verizon (2018). Data Breach Investigations Report.
6. Arash Habibi Lashkari, Min Chen and Ali A. Ghorbani (2018, October 31). A Survey on User Profiling Model for Anomaly Detection in Cyberspace.
7. B. Filkins (2015). The Expanding Role of Data Analytics in Threat Detection.
8. S. Muddu, C. Tryfonas. (2015, August 15).Network security threat detection by user/user-entity behavioral analysis," 15 August 2015. [Online]. Retrieved from. https://patents.google.com/patent/US9516053B1/en
9. Dr. Satnam Singh (2018). Retrieved from https://www.analyticsvidhya.com/blog/2018/07/using-power-deep-learning-cyber-security/
10. Computer Vision and Pattern Recognition (2015, June 5). https://arxiv.org/abs/1506.02025
11. Federal Agency Security Breaches Caused by Lack of User. (2016). Retrieved from http://www.businesswire.com.
12. (2021 ,February 16) Top 10 Deep Learning Algorithms You Should Know in 2021 https://www.simplilearn.com/tutorials/deep-learning-tutorial/deep-learning-algorithm
13. Weixi Li (2013, November) Automatic log analysis using machine learning.
14. Nikhil Mangrulkar, Arvind Ramdas Bhagat Patil, Abhijit S. Panden (2014, February). Network Attacks and Their Detection Mechanisms: A review.
15. Stigler, S. M. (1989), “Francis Galton’s Account of the Invention of Correlation,” Statistical Science, 4, 73–79.
16. Cristina Abad, Jed Taylor, Kenneth E. Row (November, 2003). Log Correlation for Intrusion Detection: A Proof of Concept.

17. (2021, January 25) Median time period between intrusion, detection, and containment of industrial cyber attacks worldwide from 2014 to 2019. Retrieved from: https://www.statista.com/statistics/221406/time-between-initial-compromise-and-discovery-of-larger-organizations/
18. Jianji Wang, Nanning Zheng (2014). Measures of Correlation for Multiple Variables
19. H. Jiawei and M. Kamber (2001). Data mining: concepts and techniques vol. 5.
20. F. V. Jensen (1996). An introduction to Bayesian networks. UCL press London, vol. 74.
21. D. Barbara, N. Wu, and S. Jajodia (2001) Detecting novel network intrusions using bayes estimators.
22. V. Chandola, A. Banerjee, and V. Kumar (2009) Anomaly detection: A survey.
23. Y. Guan, A. A. Ghorbani, and N. Belacel (2003) Y-means: A clustering method for intrusion detection.
24. Zhihong Tian, Chaochao Luo, Hui Lu, Man Zhang (2020). User and Entity Behavior Analysis under Urban Big Data.
25. A. Lakhina, M. Crovella, and C. Diot (2004) Diagnosing network-wide traffic anomalies.
26. Digital Guardian, Chris Book (2020, August 18). What Does a Data Breach Cost in 2020? Retrieved from: https://digitalguardian.com/blog/what-does-data-breach-cost-2020
27.