# A Generic Approach to Big Data Alarm Prioritization

Askew Ossi, Darshit Mody, Ayushi Vyas, Branker Tiffany, Pedro Vasseur, and Stephan Barabassi
*Pace University, Pleasantville, New York*

**Abstract— This study focused on developing a process to confirm and prioritize true alerts, and to categorize alarms into one of the two: true positives or false positives, with the intention that the new processes will improve efficiency for analysts in the analysis of security logs from processes of Data Leak Detection. The study discussed various security applications, monitoring and Big Data tables. The intention is that, when implemented, refined processes will create a more manageable environment for the review of data security reports. The research additionally investigated different approaches of creating an automated interface that could be used to provide information about true positives from the machine learning model**.

*Keyword Terms:* ***False positives, Data Loss Prevention, Data anomaly Detection, Machine Learning, Predictive Analytics, Security Logs***

## I. INTRODUCTION

All institutions should be on their toes when it comes to the security of their precious data. Data Leakage and Data Loss are the common issues countered on a regular basis. The breach in security has become a potential threat considering the voluminous number of alerts generated by the data. X. Shu and D. Yao in their thesis define data breach as, "A **data breach** is an incident in which sensitive, protected or confidential **data** has potentially been viewed, stolen or used by an individual unauthorized to do so. **Data breaches** may involve personal health information (PHI), personally identifiable information (PII), trade secrets or intellectual property [1]."

Being generated as a result of data repositories that are being queried almost all day long, the alerts might be in the form of noisy data - false positives - which can be discarded. While examining these alerts in bulk, any security analyst can mistake a true positive for a false one and weed it out which might cause a huge monetary loss to the company, ruin its reputation, or worse. While doing business, sometimes sensitive data must be handed over to trusted third parties. For example, a hospital may give patient records to researchers who will devise new treatments [2].

In looking to relate the reduction of the number of false positives in security logs that track data breaches, and the confirmation of these alarms after the fact, we want to examine the issue from two perspectives. We began by analyzing the logs and identifying the specific circumstances, attributes, or actions that trigger false alarms with the intention of creating rules that would be effective at reducing their number. If we could confirm commonalities in false alarms, we may be able to instruct the application that does the detecting of the leaks how to do so more effectively. We also want to consider the best practices for configuring these applications, as well as the policies and procedures being employed in the environment generating the incidents that are being logged, to determine if preventative controls would be helpful at reducing the overall number of incidents reported, thereby also reducing the number of false alarms. These preventative controls could be in the form of continuous feedback of confirmed alarms to the machine learning model by the security analyst.

It is also important to consider the role that data status plays in the monitoring of data breaches. Sensitive data can be considered "at rest" – being stored and not in transit or in use – or "in motion", in transit across a network. The methods employed for protecting data will differ depending on the state of the data, and is therefore likely to trigger false positives for different reasons. Data at rest is usually monitored by access controls and file permissions; false alarms triggered by attempts to access this data may warrant re-evaluating file permissions and user groups after a vetting process done by the subject matter expert, the reviewing security analyst. Data in motion would be monitored by the Data Loss Prevention/Data Leak Detection (DLD/DLD) applications; false alarms triggered by data in traffic may require a more careful examination of the rules that govern incidents, and modification to make them more specific. The latter requires also the SME feedback for iterative fine tuning of the DLD.

## II. RELATED STUDIES

Vijay Bharti the head of Cyber Security practices at Happiest Minds Technologies referred to a Ponemon institute report in a recent post about the cost attributed to false positive alerts he stated "A January 2015 Ponemon Institute report stated that enterprises spend $1.3 million a year dealing with false positive alerts, which translates into around 21,000 hours of wasted time. The study, which surveyed more than 600 IT security enterprises in the US, found that organizations receive around 17,000 malware alerts on a weekly basis, of which only 19% are worthy of attention"[3].

It is essential for any software organization to safeguard its infrastructure and business data against malicious security threats. Doing so becomes highly difficult as the current Data Loss Detection techniques produce numerous false alarms,

which consume valuable resources and business hours of examination of these alerts [4]. It should be noted that there is a tendency in the implementation of these DLD applications towards the generation of false positives at the expense of true positives. In some cases this is due to hasty deployment of the package with minimal fine tuning of the basic access rules. This profusion of false alarms is magnified in the case of Big Data security and breach monitoring due to the intrinsic large volume and replication of the of the data [5][6]

In "How to tackle false positives in big data security applications", Ram and Cody demonstrate the best practice to developing a model to reduce false positive anomalies using examples from Microsoft and Netflix. For instance, Netflix developed a user 'tagging' system to aid the analysts on what kind of alert is 'in vogue' with the system [7].

Performing testing of anomaly detection systems periodically is strongly recommended along with the formation of a procedural checklist of focus areas to be monitored. The checklist will aid the analysts of data logs in performing these checks, and aid in not overwhelming analysis by the sheer size of the alerts generated in the logs. These processes could improve the SME feedback to the machine learning model that confirms and prioritizes the alarms ***a posteriori,*** in other words, after the reports have been analyzed.

## III. APPROACHES TO FALSE ALARM DETECTION AND PRIORITIZING

The focus of this study was to develop a process to confirm and prioritize true alerts; the approach needed to address several of the issues that wasted productive hours in many organizations and exposed them to potential security breaches. The alarms fall into one of the two categories: true or false positives, according to Paul Cotter, security infrastructure architect at business and technology consulting firm West Monroe Partners. "False positives have always been a problem with security tools, but as we add more layers to our security defenses, the cumulative impact to these false positives is growing" [8].

Many of the approaches investigated to improve the efficiency of anomaly detection can be classified into the four main categories of machine learning, supervised learning model, unsupervised learning model, semi-supervised learning model, and a reinforcement-learning model. There are various ways that sensitive data can be accessed without proper credentials and copied by unauthorized individuals email, instant messaging, print medium, mobile devices and removable storage devices. It is crucial to identify sensitive data sources and properly label them by level of sensitivity for access purposes as a basis of any effective data loss prevention policy. These constraints are part of the accessibility rules used by the DLD while examining data base queries.

Supervised learning models or inductive learning model are algorithms that Gary Ericson a content developer with Microsoft, working on documentation for Azure Machine Learning describes as "predictions based on a set of examples [5]". Gary described the process by giving the example of stock prices; he stated, "Historical stock prices can be used to hazard guesses at future prices. Each example used for training is labeled with the value of interest—in this case the stock price. A supervised learning algorithm looks for patterns in those value labels. It can use any information that might be relevant—the day of the week, the season, the company's financial data, the type of industry, the presence of disruptive geo-political events—and each algorithm looks for different types of patterns. After the algorithm has found the best pattern it can, it uses that pattern to make predictions for unlabeled testing data—tomorrow's prices." [5] The supervised learning algorithm must be provided with the true or false nature of an alarm to begin with in order to be able to predict accurately in the future. To continue to be effective in its scoring confirmation process, the model should be iteratively re-trained by the analyst, the subject mate expert, whenever a new combination of predictor variables are discovered.

The supervised learning model is the most mature studied machine learning model and sometimes called classification, Classification or class label can be described as a problem of identifying the category of new instances on the basis of the training sets of data containing observations that are known. When used for the prediction of grouping labeled data points into categories such as normal, unusual, and highly unusual [6].

The unsupervised learning model is a much more fluid learning model, unsupervised learning models do not necessarily require all data points to be labeled, because the unsupervised learning model does not focus on pre-programmed characteristics, and there may be very little distinction between datasets that would be used for a training session and the actual testing data sets.

Unsupervised anomaly detection algorithms scores the data solely based on the fundamental properties of the datasets to estimate what is deemed normal behavior, and what is identified as an outlier outside the range of what is deemed normal behavior. Popular techniques include self-organizing maps, nearest-neighbor mapping, and k-means clustering and singular value decomposition. In Artificial Intelligence, a modern approach authors Stuart Russell, and Peter Norvig stated, "The most common unsupervised learning task is clustering" [5]

Clustering anomaly detection techniques group like data into clusters. Clustering techniques can be divided into two subgroups:

- Hard Clustering**:** In hard clustering, each data point either belongs to a cluster completely or not [8]
- Soft Clustering: In soft clustering, instead of putting each data point into a separate cluster, a probability or likelihood of that data point to be in those clusters is assigned.

It must be noted, that unsupervised models would not add much value to this research, as demonstrated in previous efforts, and in turn would greatly increase the amount of time to finalize the eventual prioritization goal [15].

Reinforcement learning or semi-supervised learning in learning models where a set algorithm gets to choose an action in response to the supplied data, there is a gradual learning process that evaluates how good a choice is, based on the parameters. This technique best represents the final approach of this paper.

The semi-supervised model modifies the decision-making strategy to achieve the highest reward possible. Krzysztof J. Cios a Professor of Computer Science at The School of Engineering, Virginia University commented about reinforcement learning that "reinforcement learning requires the learner to extract a model of responses based on experience observations that include states, responses, and the corresponding reinforcements [7]".

Semi supervised learning is used for the same applications as supervised learning. But it uses both labeled and unlabeled data for training – typically a small amount of labeled data with a large amount of unlabeled data this type of learning can be used with methods such as classification, regression and prediction. Semi supervised learning is useful when the cost associated with labeling is too high to allow for a fully labeled training process. The agent will reach the goal much faster by following a good policy. So, the goal in reinforcement learning is to learn the best policy.

## IV. PREVIOUS WORK

In previous studies conducted by students, this topic gained a huge amount of ground in research, testing and analyzing techniques to assist in sifting the output of DLD applications. The data studied was an example of data in motion, and the data that was examined were generated alarms while in transit over the network. The alerts occurred over a span of four hours during normal business hours, and it recorded 352 incidents. Out of the 352 incidents, 40 were determined to be false alarms.

The first data analysis technique used was data clustering and attributes that were recorded in logs that were false positive. Due to the shortness of time and no computer assistant, the team decided that out of the 352 recorded incidents in this sampling the alarms were triggered by 23 different users. Only three out of the 23 were generating false alarms while the rest were generating legitimate alarms.

This information gave the team an advantage, knowing which data was false and which were not. One of the biggest challenges the team had were the three users that were generating the false alarms also generated true alarms. Previous teams found that the target UID path of the uploaded datasets triggered the false positives, and all the false positives shared the same UID path. By flagging the target UID path as a false alarm, the rate of false positives in the given data went from to 11% to 3.9%.

Two semesters ago the team discovered a pattern where any file uploads that took more than a second were classified as a false alarm, and any file that took a second or less was sorted based on the number of changes that were made to file ownership. With this data in mind, the team decided to use a decision tree. Most of the files had only one change in ownership and were split between actual and false alarms. Each file that had more than one change of ownership was a true alarm. The tree seems to be effective but still misidentified some of the cases. [15]

The decision tree algorithm proved to be more useful in multiple environments because it could consider various attributes of the data. It allowed the consideration of more attributes at a single time and how the value of one attribute might affect another. Unfortunately, due to project limitations, there were various aspects of the data the team were unable to incorporate into their analysis. This left many options for additional things to consider in future research.

Last semester's team leveraged as much as possible on the previous work done by applying to the "data at rest" condition especially in the Big Data security logs. The team's client created slides to visualize the status and desired stated of false alarms handling. The team's objective was to better understand the nature of false alarms by creating an algorithm that was predictive, and prioritize them to help limit the amount of logs the security analyst sees before a decision was made whether to act on, or discard an alarm.

The decision tree algorithm proved to be more useful in multiple environments because it could consider various attributes of the data. It allowed the consideration of more attributes at a single time and how the value of the one attribute might affect another. Unfortunately, due to project limitations, there were various aspects of the data the team were unable to incorporate into their analysis.

## V. DATA MINING TOOLS AND TECHNIQUES

"Data mining is the process of finding useful patterns from large amount of data. Data mining also called the knowledge extraction is a technique that finds patterns to help make important decisions for a business company [10]". The steps involved are:

- Pattern Exploration
- Identification
- Deployment

During exploration data is cleaned and transformed into another form, its nature determined. Pattern identification chooses patterns that make the best prediction. Patterns are deployed towards the end for a desired outcome.

### *False alarm category analysis:*

False alarms for a system or service can be of multiple type depending from their context and relationship to the primary (also called "root case") alarms. These include the following:

- Sympathetic alarms which are in relationship (cause-effect, child-parent or other) with the primary.
- Upstream alarms which are statistically correlated to the primary alarm. These are usually implications of a related primary cause but not in most of the case finding tangible dependency is hard to perform. The upstream events usually cause the larger business impact, but the solution of the problem usually lies

within diagnosing a smaller number of primary events.

Upstream and primary event data can be simply processed by R applications (Rstudio, Spark R, Python R), where we can formulate a null hypothesis of being correlated and with the expectation of easily confirming it.

A simple way to generate rich datasets of real-life Big-Data alarms, which contain these two categories of False Alarms, can be done by capturing the operation logs from complex Big-Data solutions, such as for example from the Cloudera Distribution of Hadoop (called CDH).

As a simple exercise, we have collected about 211,000 alerts from Cloudera Manager v 5.9, managing a complex CDH cluster with 10+ big data applications middleware and 20+ applications running mid to high-end workloads.

We have identified and isolated a series of sympathetic events within the informational and warning categories and upstream event within the warning and error categories (close to 1000 alarms).

As a simple exercise, we have uploaded the upstream-analysis candidate alarms into Rapid Miner and using multi-colored histogram type charting feature, we were able to demonstrate the presence and reoccurring nature of upstream category of events.

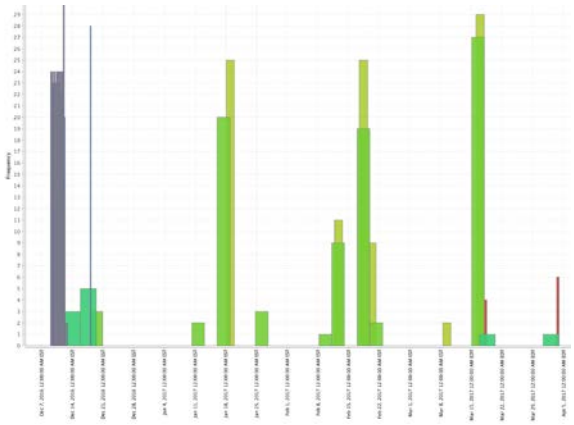

*Figure 1. Correlated upstream false alarms*

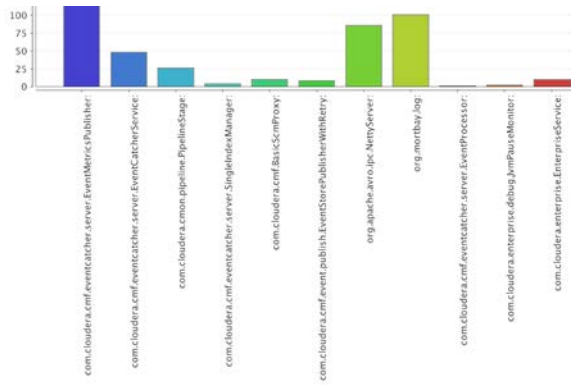


*Figure 2: Color coding of analyzed alarm sets*

Data Mining Techniques reviewed for this research:

Various algorithms and techniques like Classification, Clustering, Regression, Artificial Intelligence, Neural Networks, Association Rules, Decision Trees, Genetic Algorithm, Nearest Neighbor method etc., are used for knowledge discovery from databases.

### *A. Classification*

A classic data mining technique based on machine learning, classification is used to classify each item in a set of data into one of a predefined set of classes or groups. Mathematical techniques such as decision trees, linear programming, neural network and statistics are key concepts applied in classification. In classification, we develop the software that can learn how to classify the data items into groups. The task is to develop classification software that classifies data into separate groups [10].

### *B. Clustering*

A classic data mining technique based on machine learning, classification is used to classify each item in a set of data into one of a predefined set of classes or groups. Mathematical techniques such as decision trees, linear programming, neural network and statistics are key concepts applied in classification. In classification, we develop the software that can learn how to classify the data items into groups. The task is to develop classification software that classifies data into separate groups [10].

### *C. Prediction*

The prediction, as the name implies, is a data mining technique that discovers the relationship between independent variables and relationship between dependent and independent variables [10].

### *D. Sequential Patterns*

The analysis of identification of similar patterns, regular events or trends in transaction data over a business period is achieved using sequential patterns analysis [10].

### *E. Decision Trees*

The Decision tree is the easiest to understand and hence is the most widely used data mining technique. In decision tree technique, the root of the decision tree is a simple question or condition that has multiple answers. Each answer then leads to a set of questions or conditions that help us determine the data so that we can make the final decision based on it. [9] This is the preferred model for this paper to achieve high scoring of the alarms as true or false [10].

Data Mining Tools reviewed for this research:

### *A. Rapid Miner*

Written in the Java Programming language, this tool offers advanced analytics through template-based frameworks. In addition to data-mining, rapid miner also provides functionalities such as data preprocessing and visualization, predictive analytics and statistical modeling, evaluation, and deployment. Learning schemes, models and algorithms from WEKA and R scripts makes it a more powerful tool. This is the tool chosen for the scoring process in this research using the ID3 decision tree modeler [10].

### *B. WEKA*

The advanced Java based version of the tool is used in applications such as visualization and algorithms for data

analysis and predictive modeling. It is customizable which is a big plus when compared to rapid miner [10].

WEKA supports several standard data mining tasks, including data preprocessing, clustering, classification, regression, visualization and feature selection [10].

### C. R-Programming

R is highly extensible and easy to use. It is primarily written in C and FORTRAN. Majority of its modules are written in R itself. Popular among data miners for developing statistical software and data analysis, it is a free software programming language and software environment for statistical computing and graphics [10].

Besides data mining it provides statistical and graphical techniques, including linear and nonlinear modeling, classical statistical tests, time-series analysis, classification, clustering, and others [10].

### D. Orange

It is a Python based software that is easy to learn, an apt tool for experts as well as novices. In addition to visual programming and Python scripting Orange also has components for machine learning, add-ons for bioinformatics and text mining. It's packed with features for data analytics [11].

### E. KNIME

KNIME performs for all the three components of Data preprocessing: extraction, transformation and loading. The business intelligence and financial data analysts equally seek it after. It gives a graphical user interface to allow for the assembly of nodes for data processing. An open source data analytics, reporting and integration platform, KNIME also integrates various components for machine learning and data mining through its modular data pipelining concept.

Written in Java and based on Eclipse, KNIME is easy to extend and to add plugins. Additional functionalities can be added on the go [12].

### F. NLTK

NLTK works exceptionally well for language processing tools. NLTK provides a pool of language processing tools including data mining, machine learning, data scraping, sentiment analysis and other various language processing tasks. It's written in Python, one can easily build applications on top of it, customizing it for small tasks [13][14]

## VI. EXPERIMENTATION AND RESULT

To mine meaningful patterns that could be used for key decision-making tasks and the exponentially increasing consumption of data lead to companies feeding their information into very large repositories. A non-detected breach could prove extremely detrimental if not detected in time incurring important financial and reputational consequences. Hence, the information needed to be safeguarded against malicious threats and unauthorized access.

Data Leak Detection applications continually monitor the access to internal data repositories are employed by many companies. An alarm is generated in the form of a security log record whenever the DLD engine encounters any anomalous data querying. The security experts then analyze those logs to determine if the alerts were indeed malicious or benign, in a timely manner. To scrutinize these security logs efficiently and to prioritize true and false positives traditional data mining methods or Big Data Analytics approach are being proposed here to be used in order to ameliorate this risky process. The programmatic approach should confirm the nature and priority of the alerts and abate in reducing the time and cost to deal with malicious data accesses.

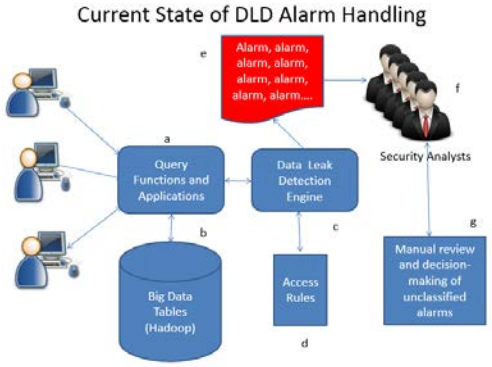


*Figure 3. The current state of DLD Alarm Handling*

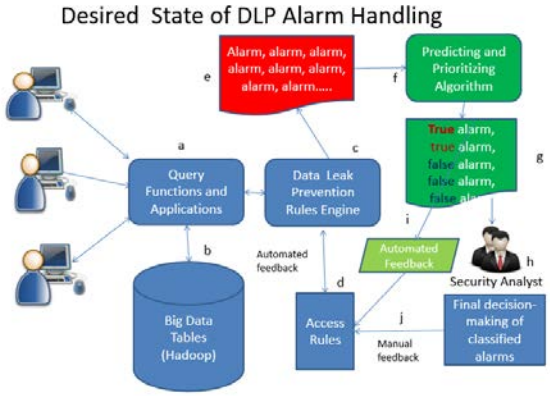


*Figure 4. Desired State of DLD Alarm Handling*

Figure 3 above provides an overview of the current state of Data Leak Detection Alarm Handling machine and figure 4 forms the desired state of DLD Alarm handling. The alarms generated by the DLD engine were prioritized and categorized as true and false positives. Rapid miner was used to attain prominent accuracy.

The final focus is on minimizing the time and effort required to manually update the Access Rules violation decision table by developing a programmatic approach to generate **component I** that could efficiently replace **component J**. The main aim is to be able to aid the security data analysts in dealing with a huge number of alerts created by the DLD machine from the overwhelming volume of queries to the larger repositories by scoring them correctly. [15]

The logic used is as follows:

If {TSIV1, TSIV2, TSIVn, TSTV} of *x*
is not equal to
{ARIV1, ARIV2, ARIVn, ARTV} of *y*

then execute *component j*.

Where,

- TSIV stands for the value of a Training Set Independent Variable
- TSTV stands for the value of a Training Set Target Variable e.g. True or False
- ARIV stands for the value of a Access Rules Independent Variable
- ARTV stands for the value of a Access Rules Target Variable e.g. True or False
- 1 to *n* is the distinct value of an Independent Variable in an instance of the Training Set,
- *x* is a record in a Training Set
- *y* is the corresponding record in the Access Rules table.

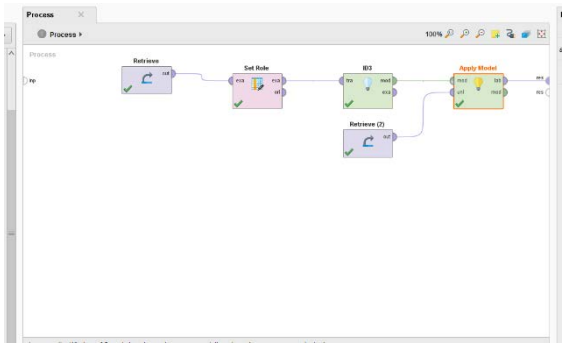


*Figure 5: Rapid Miner ID3 Algorithm Diagram with Test and Production Data Sets*

In setting up Rapid Miner, the usage of the ID3 algorithm was again employed. This time making changes within the process to reveal more precise results. Parameters were set to achieve correctness with the data in the spreadsheet. *Figure 3,* shows the process as set up for this example.

The results provided a view in terms of *confidence* of the data. This implies that the program was able to identify which alarms were nearly actual as being false or true in terms of 1 being positive and 0 being negative, shown in *Figure 5 and 6 below*.

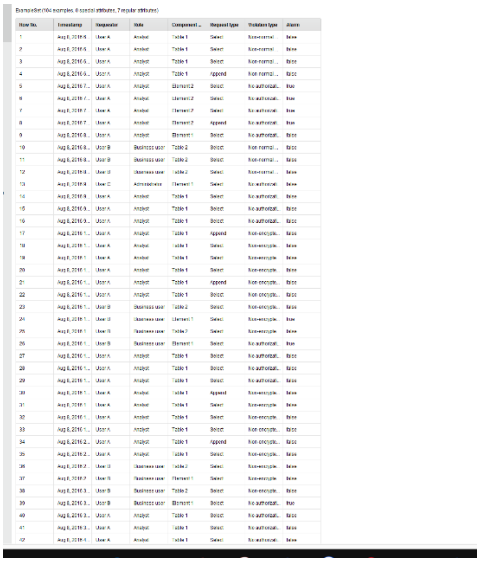

*Figure 6: Rapid Miner Results Confidence of Data*

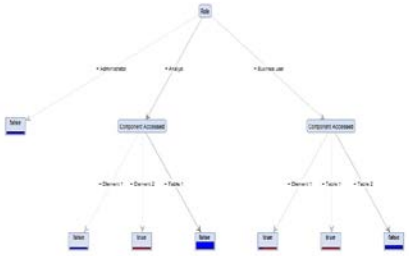

*Figure 7: Rapid Miner ID3 more efficient tree structure*

## VII. FUTURE WORK

The team ran test data sample sets on Rapid miner and leveraged on the results obtained by the previous semester's project work. Initially no changes were made to the weights and the tool was allowed to determine results. A second test data sample was run against the previous one and the system was tested to check for false positives. The results obtained were not 100 percent accurate. The future task would include obtaining real-time data from companies and running them across RapidMiner to efficiently detect true positives and confirm the false ones.

Simple R interference analysis can prove the correlation in between Upstream and Primary event data and one can perform this using Rstudio for subsequent enhancements of this study.

## VIII. LIMITATIONS

Obtaining real-time data had its own challenges. Our team faced a major issue in finding a source willing to share sample data for obvious reasons, security and privacy. There is a "masking" process that needs to be done also if we were to obtain real alarm logs that would increase the cost of provision the reports from real organizations [16].

The team has continued to work on simulated feed from previous semesters until a real one is obtained. Results have been expected to be more precise once the real feed is obtained, and as long as the data that will be provided has been scored by a security analysts, differentiating true and false alarms. Otherwise, the unsupervised methods that would be needed will elongate the research even further [17].

## IX. REFERENCES


[1] X. SHU AND D. (D. YAO, "DATA LEAK DETECTION AS A SERVICE: CHALLENGES AND SOLUTIONS," THESIS.

[2] ROUSE, MARGRET "WHAT IS DATA BREACH? - DEFINITION FROM WHATIS.COM," SEARCHSECURITY. [ONLINE]. AVAILABLE: HTTP://SEARCHSECURITY.TECHTARGET.COM/DEFINITION/DATA-BREACH.

[3] BHARTI, VIJAY" "THE HIGH COST OF FALSE POSITIVES TO AN ORGANIZATION," DIGITAL TRANSFORMATION BLOGS BIG DATA IOT M2M MOBILITY CLOUD. [ONLINE]. AVAILABLE: HTTP://WWW.HAPPIESTMINDS.COM/BLOGS/THE-HIGH-COST-OF-FALSE-POSITIVES-TO-AN-ORGANIZATION/.

[4] J. BUCKRIDGE, E. FINKELSTEIN, M. HASANRAMAJ, AND P. VASSEUR, "IMPROVING DATA LEAKAGE DETECTION AND PREVENTION SOLUTIONS BY REDUCING FALSE POSITIVES IN SECURITY LOGS.",2016, SEIDENBERG SCHOOL OF CSIS, PACE UNIVERSITY, PLEASANTVILLE, NEW YORK

[5] D. BRADBURY, "OVERWHELMED WITH ALERTS AND FALSE POSITIVES: WHY SECURITY ANALYTICS IS ON THE RISE," IT WORLD CANADA. [ONLINE]. AVAILABLE: HTTP://WWW.ITWORLDCANADA.COM/ARTICLE/OVERWHELMED-WITH-ALERTS-AND-FALSE-POSITIVES-WHY-SECURITY-ANALYTICS-IS-ON-THE-RISE/375046#IXZZ4DQUFY4VZ.2015

[6] E. DAMIANI, "TOWARD BIG DATA RISK ANALYSIS," 2015 IEEE INTERNATIONAL CONFERENCE ON BIG DATA (BIG DATA), NOV. 2015.

[7] RAM SHANKAR SIVA KUMAR, CODY RIOUX, "HOW TO TACKLE FALSE POSITIVES IN BIG DATA SECURITY APPLICATIONS - STRATA HADOOP WORLD IN SAN JOSE 2016," BIG DATA CONFERENCE: STRATA HADOOP WORLD, MARCH 28 - 31, 2016, SAN JOSE, CA. [ONLINE]. AVAILABLE: HTTPS://CONFERENCES.OREILLY.COM/STRATA/STRATA-CA-2016/PUBLIC/SCHEDULE/DETAIL/47132.

[8] B. VIOLINO, "SECURITY TOOLS' EFFECTIVENESS HAMPERED BY FALSE POSITIVES," CSO ONLINE, 02-NOV-2015. [ONLINE]. AVAILABLE: HTTP://WWW.CSOONLINE.COM/ARTICLE/2998839/DATA-PROTECTION/SECURITY-TOOLS-EFFECTIVENESS-HAMPERED-BY-FALSE-POSITIVES.HTML.

[9] E. AMLIE, P. GELSOMINO, A. G. GIRI, J. RODRIGUEZ, AND P. VASSEUR, "BIG DATA FALSE ALARMS: IMPROVING DATA LEAKAGE DETECTION SOLUTIONS." HTTP://CSIS.PACE.EDU/~CTAPPERT/SRD2017/2016FALLPROJ/D4T09.PDF

[10] P. NORVING AND S. RUSSEL, ARTIFICIAL INTELLIGENCE: A MODERN APPROACH. S.L.: PEARSON EDUCATION LIMITED, 2013.

[11] K. J. CIOS, W. PEDRYCZ, AND R. W. SWINIARSKI, DATA MINING: METHODS FOR KNOWLEDGE DISCOVERY. BOSTON, MA: KLUWER ACADEMIC, 2000

[12] Z. JADIDI, V. MUTHUKKUMARASAMY, E. SITHIRASENAN, AND K. SINGH, "INTELLIGENT SAMPLING USING AN OPTIMIZED NEURAL NETWORK," JOURNAL OF NETWORKS, VOL. 11, NO. 01, 2016.

[13] "MACHINE LEARNING: WHAT IT IS AND WHY IT MATTERS," WHAT IT IS AND WHY IT MATTERS | SAS. [ONLINE]. AVAILABLE: HTTPS://WWW.SAS.COM/EN_US/INSIGHTS/ANALYTICS/MACHINE-LEARNING.HTML.

[14] S. KAUSHIK, T. SRIVASTAVA, F. SHAIKH, S. KASHYAP, AND SAURABH.JAJU2, "AN INTRODUCTION TO CLUSTERING & DIFFERENT METHODS OF CLUSTERING," ANALYTICS VIDHYA, 10-DEC-2016. [ONLINE]. AVAILABLE: HTTPS://WWW.ANALYTICSVIDHYA.COM/BLOG/2016/11/AN-INTRODUCTION-TO-CLUSTERING-AND-DIFFERENT-METHODS-OF-CLUSTERING/

[15] E. AMLIE, P. GELSOMINO, A. G. GIRI, J. RODRIGUEZ, AND P. VASSEUR, "BIG DATA FALSE ALARMS: IMPROVING DATA LEAKAGE DETECTION SOLUTIONS." HTTP://CSIS.PACE.EDU/~CTAPPERT/SRD2017/2016FALLPROJ/D4T09.PDF

[16] "DATA MINING TECHNIQUES," ZENTUT. [ONLINE]. AVAILABLE: HTTP://WWW.ZENTUT.COM/DATA-MINING/DATA- MINING-TECHNIQUES.

[17] CHANDAN GOOPTA "SIX OF THE BEST OPEN SOURCE DATA MINING TOOLS," THE NEW STACK, 07-SEP-2015. AVAILABLE: HTTP://THENEWSTACK.IO/SIX-OF-THE-BEST-OPEN-SOURCE-DATA-MINING-TOOLS/.