A Rule Induction Attribution Selection Algorithm for Intrusion Detection Systems

IJOTM
ISSN 2518-8623

Volume IV. Issue I
pp. 1-14, June 2019
ijotm.utamu.ac.ug
email: ijotm@utamu.ac.ug

Anthony A. Ashaba
Gulu University
Email: aashaba@gmail.com

Drake Patrick Mirembe
Makerere University

Daniel Ogenrwot
Gulu University

Robert Tumusiime
Uganda Technology and Management University (UTAMU)

Abstract

The high level of dependence of computer users on communication network infrastructures such as Internet and intranet is associated with increased level of threats to security resulting into outcomes such as interference to valid communication channels and loss of valuable information. Several network security tools have been developed over the past years, one being Intrusion Detection Systems (IDS). IDS use attributes to differentiate between normal and intrusive activities based on the behavior of users, networks or computer systems. However, with IDS, the expert's guess, experience and knowledge are central when choosing the features for detection which often results to false alarms and insufficiency of the detection system. This study investigated the possibility of enhancing the performance of IDS using data mining techniques. This study proposed Rule Induction technique of data mining to remove redundant or irrelevant attributes of IDS thereby enhancing accuracy, speeding up the computation time and minimizing false alarms. For effective generalization of Rule Induction Attribution Selection (RIAS), the algorithm was tested on KDD Cup99 dataset. Accuracy results from RIAS (53.98) were higher than that of Repeated Incremental Pruning to Produce Error Reduction (RIPPER) (0.48) while RIAS's (56121.53) computation time fell below that of RIPPER (902.47). The high accuracy results of RIAS indicate its capability to minimizing false alarms more than RIPPER. Clustering based on weighted support was applied to test the effectiveness of RIAS. Findings indicated that integrating data mining with IDS is effective in identifying useful information, hidden trends and associations from bulky of information.

Key words: *Intrusion Detection Systems, Rule Induction, Data Mining*

Introduction

The tremendous increase in internet applications has led most businesses and individuals to rely on the internet for their day to day activities. This increased dependence on the internet has numerous benefits although it is also possible that it can empower criminals to target individuals, private and government organizations [23]. Findings by Symantec [28] show that the magnitude of threats is continuously

increasing, and with the emergence of cheaper and readily available technologies and communication channels attracts malicious activity of all sorts. A threat to organizations is the insider threats that are generally caused by current ex-employees, contractors or partners who have authorized access to the organization's network and servers. Many organizations do not have adequate safeguards to detect or prevent attacks involving insiders [32]. McCue [17] states that 90% of controls are focused on external threats and yet 70 % of fraud is perpetrated by insiders rather than by external criminals. there is also an increasing trend of employees bringing their own devices (BYOD) to work making it difficult for organizations to enforce policies and restrictions on employees working with mobile devices and as such unauthorized use and modification of data due to deliberate or negligent actions is easily transferred to the organization.

The growth of threats necessitates strong countermeasures to detect flaws ahead of time and proactively prevent such attacks before they happen. It is important that the security mechanisms of IT infrastructure are designed to prevent unauthorized access to system resources and data. However, at present, completely preventing breaches of security appears to be unrealistic because most systems have security flaws due to increasingly sophiscated attacks, malware, and abuses from privileged insiders. Although security measures such as firewalls, virus scanners and encryption mechanisms exist, they are not sufficient to protect network data and its resources [22]. Firewalls for instance are hard to configure properly, and those who configure them may not have a good understanding of current threats and attacks. This is because firewalls are unable to protect against malicious code, insider attacks which subvert its purpose and those of unsecured modems [20]. Similarly, firewalls also have some inactive configurations that have no power of action to traffic directed towards them [36] and they are also vulnerable to errors in configuration and undefined security policies. In recent studies [1, 7, 11, 33, 34], it has been demonstrated that errors or conflicts in configuring rule sets can lower protection. Deploying an Intrusion Detection System (IDS) helps increase visibility and control within a corporate computing environment. They do not fully guarantee security, but when used with security policy, vulnerability assessments, data encryption, user authentication, access control and firewalls they can greatly enhance network security. The goal of IDS is to analyze data and determine indeed if any intrusion occurred although they are not always effective against emerging intrusion attempts [13]. The security analyst has to identify each alert and determine whether it is a false positive or a true positive.

Responding to and evaluating IDS alerts is labor intensive requiring vast human resource. The analysis process is often time consuming that the administrators do not have the resources to go through it all and find the relevant information. With the application of data mining algorithms, it is possible to determine the necessary attributes thereby helping IDS accurately determine what is normal and intrusive consequently reducing on the false positives. Data mining applied in IDS, can mine the features of new and unknown attacks well, which is a maximal help to the dynamic defense of Intrusion detection system [25]. In order to overcome the limitations of traditional intrusion detection system, a systematic and higher automation method should be employed in the design of IDS [36].

The use of IDS as an integrated part of security systems has become common and industry preferred form of security system design. However, the amount of false positive alarms generated by these systems requires immediate attention/solution so as to minimize the impact of attacker's exploitation of the security

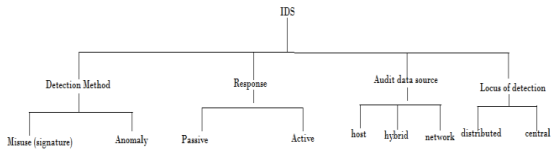
system vulnerabilities. This research proposed a rule-based attribute selection algorithm to IDS focusing on one of the key vulnerability of IDS that creates false positive alarm in the form of duplicate/redundant attributes. Through RIAS data mining algorithm, these attributes can be isolated and deleted to increase the effectiveness of the IDS. The results of this research therefore contribute to near-elimination of false positive alarms in IDS.

Intrusion Detection System

Intrusion Detection Systems have become an essential component of computer security to detect attacks that occur despite the best preventive measures. Intrusion detection is based on the assumption that system activities are observable and intrusive activities are noticeably different from normal system activities and thus detectable. There should be distinctive features between the normal data and intrusive activities. An IDS seeks to detect any malicious activity against computer systems by monitoring the behavior of users, networks or computer systems and reports to the administrator to take appropriate action against them [3], [4]. These systems have been designed and implemented based on the system builders' knowledge of a computer system and their understanding of known intrusions. As a result, the effectiveness and adaptability of the intrusion detection systems are limited in the face of new computing environments or newly invented attack methods [16]. Experts first analyze and categorize attack scenarios and system vulnerabilities, and hand-code the corresponding rules and patterns for misuse detection.

One major challenge in intrusion detection is that we have to identify the camouflaged intrusions from a huge amount of normal communication activities, making it demanding to apply data mining techniques to detect various intrusions. Data mining is capable of identifying legitimate, novel, potentially useful, and eventually understandable patterns in massive data. Data mining provide effective mechanism for understanding normal behavior inside the data and use this knowledge for detecting unseen intrusions. Another major challenge faced in today's IDS is its ability to effectively measure their performance. Measure of the effectiveness of intrusion detection refers to its ability to efficiently and correctly classify events as malicious or not. This is especially for anomaly detection, a situation where a normal activity is being considered as an intrusion which is false meaning the IDS is prone to mistakes. For anomaly to work the idea is that you assume what is deviating from normal profile as possible intrusions. What could be deviating from the normal profile is likely to be the new normal behavior that you have not observed previously. This results into a false positive and some deviations are necessarily not actual attacks therefore cases of false positives become a point of research.

IDS may be categorized on the basis of a number of characteristics



A wide variety of approaches have been explored to building better IDS like machine learning and Rule based or heuristic methods. Das et al. [8] pointed out that there is no heuristic to confirm the accuracy of results of IDS algorithms. The exact effectiveness of a network intrusion detection system's ability to identify malicious sources cannot be reported unless a concise measurement of performance is available.

Kalarani and Brunda [14] define data mining as the process of discovering interesting patterns or knowledge from large amounts of data. The type of interestingness could be frequency, rarity, correlation, length of occurrence, consistency, repeating / periodicity, "abnormal" behavior and so on. This interestingness is information which is not very obvious, something that is not visible directly. Fayyad, Piatetsky-Shapiro, and Smyth [10] indicate that these interesting patterns can be used to make predictions.

Reasons for use of data mining in IDS could be attributed to the fact that it is very hard to program an IDS using ordinary programming Languages that require the explicitation and formalization of knowledge and also the environment of IDS and its classification task highly depend on personal preferences. What may seem to be an incident in one environment may be normal in other environments. This way, the ability of computers to learn enables them to know someone's "personal" (or organisational) preferences and improve the performance of the IDS, for this particular environment [19]. The adaptative and dynamic nature of machine learning makes it a suitable solution for IDS.

Lee, Stolfo, and Mok [16] discusses a systematic framework for analyzing audit data and constructing intrusion detection models. Under this framework, a large amount of audit data is first analyzed using data mining algorithms in order to obtain the frequent activity patterns. These patterns are then used to guide the selection of system features as well as the construction of additional temporal and statistical features for another phase of automated learning. Classifiers based on these selected features are then inductively learned using the appropriately formatted audit data. These classifiers can be used as intrusion detection models since they can classify (i.e., decide) whether an observed system activity is "legitimate" or "intrusive".

Machine learning based techniques learn patterns, identify similar things which makes them adaptive, dynamic and requiring minimal intervention. This research applies the rule based methods which are able to find novel attacks or identify events never seen before. Traffic changes day by day but rule-based classifiers remain focused on any outliers on the traffic. As such, when an anomaly is flagged, a corresponding rule is triggered to confirm the anomaly thereby minimizing false alarms. This makes the rule-based methods simple, understandable and also dynamic depending on how the system is structured. Through Rule induction, the IDS is able to eliminate redundant or irrelevant features thereby enhancing the accuracy of detection which speeds up the computation time and reduces the number of false alarms.

Attribute Selection

The Attribution selection is to make better classification system by removing un needed attributions. Stanczyk and Jain [26] defined a feature as a trait of the system or object that can predict the behavior or state of the system. Many algorithms have been applied to develop the anomaly detection model for intrusion detection system. These algorithms, however, highly depend on input features, these input features give information to the learning algorithms which used in intrusion detection system in the form of the detection method. With irrelevant and redundant features, learning algorithms build detection models with less accuracy rate which leads to errors in the form of false negatives and false positives. Also, ambiguous features increase the time complexity and consume other computational resources as well. By removing these irrelevant and redundant features accuracy of the learning algorithms can be increased which in turn decreases computational complexity [21], [24], [35]. Feature selection is useful in the application domains that introduce a large number of input dimensions like intrusion detection.

Feature selection techniques hinges on two pillars namely relevancy and redundancy of the features [9]. Relevant features are those that predict the desired system response, on the other hand, redundant features have a high degree of correlation among themselves. Thus, removal of the redundant features is desired. The predictive accuracy of the machine learning algorithms can be increased by the feature selection. Reduced dataset also decreases dataset which acquire less storage space.

Sung and Mukkamala [27] presented several techniques for feature selection and compared their performance in the IDS application. They observed that with appropriately chosen features, both probes and DoS attacks could be detected in real time or near real time at the originating host or at the boundary controllers.

Balakrishnan et al.[2] proposed an Optimal Feature Selection algorithm based on Information Gain Ratio and Support Vector Machine (SVM) to detect attacks. They observed that computation time taken for detecting and classifying the records using all the forty one features of the KDD 99 cup data set was too large. The proposed feature selection algorithm selected only the important features that helped in reducing the time for detecting and classifying the records. Further the rule based classifier and SVM helped achieve a greater accuracy thereby reducing the false positive rates and the computation time.

Choi and Chae [5] investigated the performance of standard feature selection methods; CFS(Correlation based Feature Selection), IG(Information Gain) and GR(Gain Ratio). They proposed a feature selection method using attribute average of total and each class data. The classifier was evaluated on the NSL-KDD dataset to detect attacks and they obtained 22 relevant features. They observed that they attained the highest accuracy on 22 features than the full data set.

Olusola et al. [18] in their paper presented the relevance of each feature in KDD 99 intrusion detection dataset to the detection of each class. Results revealed that some features have no relevance in intrusion detection. These features include 20 and 21 (outbound command count for FTP session and hot login) while features 13, 15, 17, 22 and 40 (number of compromised conditions, su attempted, number of file creation operations, is guest login, dst host rerror rate respectively) are of little significant in the intrusion detection.

In current IDS, attributes are chosen from experience of the expert. So in this research the researcher provided an algorithm that abstracts effective attributes of IDS automatically using specification technology of data mining. We have to check whether the attributes lose useful information or not, to achieve this the researchers checked if the IDS treats normal traffic as a threat by providing clustering algorithm of intrusion detection data. Using clustering technology of data mining, the researcher was able to decide whether user's behavior was normal or abnormal considering the attributes under investigation. The use of IDS as an integrated part of security systems has become common and industry preferred form of security system design. However, the amount of false positive alarms generated by these systems requires immediate attention/solution so as to minimize the impact of attacker's exploitation of the security system vulnerabilities. This research proposed a rule-based attribute selection algorithm to IDS focusing on one of the key vulnerability of IDS that creates false positive alarm in the form of duplicate/redundant attributes. Through RIAS data mining algorithm, these attributes can be isolated and deleted to increase the effectiveness of the IDS. The results of this research therefore contribute to near-elimination of false positive alarms in IDS.

Therefore feature selection has a significant impact on Intrusion detection system performance in that it reduces computation time, removes irrelevant features and increases on the accuracy of the detection algorithm.

Rule Induction Based Attribution Selection (RIAS)

The Objective of RIAS is to select useful attributes and construct more correct rule set through learning by rule induction technology of data mining. Rule Induction Attribution selection has different phases involved in creating rules where the researchers through the designed algorithm were able to determine a distance and compare it with a threshold. The researchers then used laplace accuracy, using accuracy were able to remove the irrelevant features from the training set and finally clustering to group as either normal or attack.

RIAS is designed based on IDS knowledge area. This algorithm has four steps.

1. Searching step: search subset of attribution from attribution space
2. Estimating step: Estimate each candidate attribution
3. Cramming area knowledge: selecting attribution needed certainly based on application area
4. Classification step: completing classification system by attribution selected

KDD Cup99 dataset was used as a benchmark, for evaluation the dataset was set as rule data and training data which was the same data used for both rule and training set. For each feature of the rule data was checked against all the features of the training data set. To determine the distance between a given rule data (R) and training data ($\bar{X}$) for features for the 41 of the KDD Cup99 dataset, the distance can be determined as below:

$$\Delta(R,\bar{x}) = \sum_{i=1}^{|F|} \delta^2(i) \quad \bar{x}' = (x_1', x_2', \ldots\ldots\ldots, x_{|F|}', c_i') \tag{1}$$

Since KDD Cup99 data set consists of both characters and numeric values, from equation (2) for zero case if α_1 is not a number or α_1 is also null the δ at the feature is zero which applies to features 2,3 and 4 of KDD cup99 data set. $\rho(c_n/\alpha_1)$ is probability that the rule at ith attribution value is α_1 is included in label c_n hand $\rho(c_n/x_i)$ is probability that the training data at i^{th} attribution value x'_i included in label c_n. To determine the distance the equation below is used.

$$\delta(i) = \begin{cases} 0, & a_i = \emptyset \text{ or } x_i' = \emptyset \\ \sum_{i=1}^{|J|} |P(c_h/a_i) - P(c_h/x_i)|, & a_i \text{ is character} \\ \delta_{num(i)}, & a_i \text{ is numerical value} \end{cases}$$

The equation below is used to determine the distance for the numeric values in the dataset.

$$\delta_{num(i)} = \begin{cases} 0, & a_{i,lower} \le x_i' \le a_{i,upper} \\ \frac{x_i' - a_{i,upper}}{a_{i,max} - a_{i,min}}, & x_i' > a_{i,upper} \\ \frac{a_{i,lower} - x_i'}{a_{i,max} - a_{i,min}}, & x_i' < a_{i,lower} \end{cases}$$

After determining the distance, the distance is compared against the threshold, if distance ≤ threshold we consider whether it is positive training data or negative training data but if the distance > threshold we do not consider.

For each record we calculate Accuracy (R) based on training data set. We compare the distance between Rule data set and training data set. If the distance is greater than the threshold we ignore but if the distance is less than threshold then we check if labels are the same then it is a positive training data and if the labels are different then it is referred as a negative training data.

$$Accuracy \quad r_1 = \frac{|u^+| + 1}{|u^+| + |u^-| + |J|}$$

u^+ - Number of positive training data
u^- - Number of negative training data
J - Number of labels

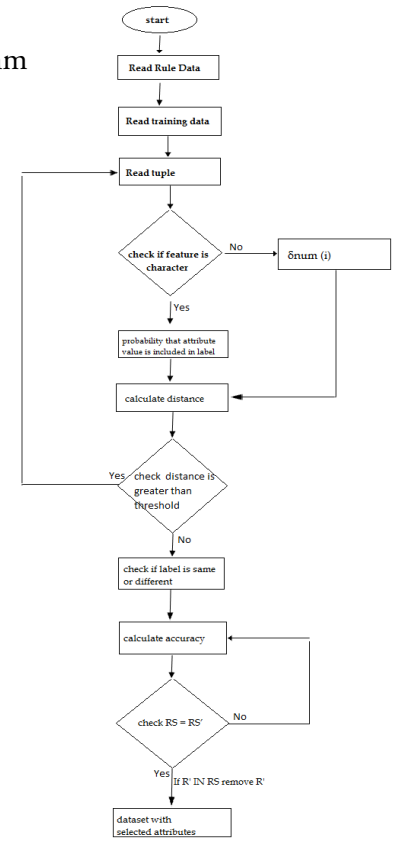
$$Accuracy\ of\ ruleset\ (RS) = \frac{\sum_{i=1}^{N} r_i}{N}$$

RIAS Algorithm

```
Begin
U ⇒ RS;
Acc(RS) calculation on the U
Repeat
    For each rule R in RS
    Find out example z that rule R get but not cover
    R' = generalization(R, E)
    RS' = (RS/R)UR'
    if Acc(RS') ≥ Acc(RS)
    RS = RS' if there is R' in RS, then remove R'
Until Acc(RS) is constant
    F' = attribute_select()
Output Return F'
End
```

The distance between data belonging to the same label is closer and the distance between data belonging to different labels is further. Like this, the magnitude of data is reduced through attribute selection process by removing the interference of useless attribution.

Figure 1:
Flowchart of RIAS algorithm



The design evaluation and performance of RIAS algorithm was done on a well-known multiclass classifier Repeated Incremental Pruning to Produce Error Reduction (RIPPER). RIPPER was developed by William Cohen [19] based on Incremental Reduced Error pruning (IREP) algorithm. The reason for choosing RIPPER against all other algorithms to be used to test the results of RIAS is because both RIPPER and RIAS depend on learning based on rule induction. Also RIPPER has been successfully used in a number of data mining based anomaly detection algorithms to classify incoming audit data and detect intrusions.

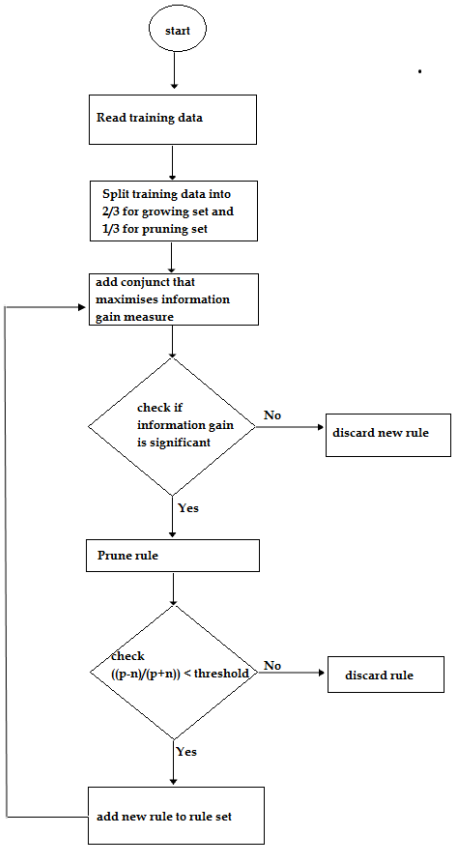
To show the essence of RIAS, we compared accuracy and execution time results of RIAS and RIPPER.

The comparison between RIAS and RIPPER reveals that the accuracy values of RIAS outperform those of RIPPER however the computation time is longer than that of RIPPER and this is attributed to the fact that RIAS uses the whole dataset whereas RIAS uses two thirds of the entire dataset. The measure that was used for evaluating the performance of RIAS was laplace accuracy and laplace estimator accuracy for RIPPER algorithm. This research did not apply RIAS to other datasets but only on the KDD Cup'99 dataset. Time complexity is another parameter that was used for evaluation, RIAS's time complexity depends mainly on the entire training examples while RIPPER on two thirds of the training examples.

$$P_{classA} = \frac{n+1}{N+C}$$

Equation 1: general Laplace estimator to determine accuracy of the ruleset

Figure 2: Flowchart of RIPPER algorithm



Experimental Results And Analysis

Description of the Benchmark dataset

There are several publicly available benchmark datasets used for IDS evaluation and these include DARPA'98 [6], [15], KDD Cup 1999 [30], [29],[12], Nsl-Kdd [31]. KDD Cup '99 dataset [116] one of the most commonly used dataset for intrusion detection evaluation was used for experimental purpose though focus was on the corrected KDD. It is the most comprehensive dataset that is still valid and applied to compare and measure the performance of IDSs. The researcher used the corrected KDD cup 99 dataset that has 311029 records. The dataset has two types of attributes numeric and character. One of the critics of the KDD Cup '99 dataset state that the datasets are outdated and do not represent today's network traffic characteristics. The dataset also has so many duplicate records, so the researcher integrated

kutools with excel and it can be noted that after redundancy removal there is a large reduction in the attack classes as compared to the normal class.

Experimental Setup

So, we randomly selected 50,000 records out of 77758 of the unique values of the KDD Cup 99 dataset as samples used for evaluation purposes which were grouped into different datasets of cases 100, 500, 1000, and 2000. All these files were then saved as CSV files to be read with in the program. All files for the different cases were fed into the RIAS algorithm while using different thresholds of 0.1, 0.15, 0.2 and 0.25. The idea of threshold is that it enables you to study whether one needs to stop the algorithm or continue running the algorithm until certain accuracies are achieved. The accuracy of any algorithm depends on the stopping criteria. The stopping criteria is what defines the threshold otherwise the algorithm runs forever or is likely to remove more features. The result for this was csv files with columns marked for removal. The researchers then took all the files for each case and looked through the entire column, as long as an entire column was marked for removal that meant that it is the redundant field. The columns that were partially marked were not removed. To evaluate the performance of the RIAS algorithm, RIPPER was used as a benchmark to check on accuracy and time computation. And also to check that RIAS was effective, the entire dataset and dataset with removed features were clustered based on weighted support and results showed that the dataset with removed features created more clusters and a higher accuracy than the entire dataset thus making RIAS effective.

Experiments were conducted on a 500HDD computer core i7 running at 2.3 GHz with 8GB DDR3 RAM. The test data as rule data and training data were passed through the designed algorithm (RIAS) for each group. With RIPPER one set of test data was used which was split into two thirds as growing rules and one third as pruning rules. Then the accuracy and time taken for each of the classifiers studied.

In RIAS algorithm, for each case the same file was treated as both rule dataset and training dataset. Through RIAS algorithm we were able to remove the irrelevant features from the 41 features. With RIPPER for each case we have only we file as the training set which was split into 2/3 for the grow set and 1/3 as the prune set. Through RIPPER we began with an empty dataset and the RIPPER algorithm was able to add useful attributes.

The researcher also applied clustering based on weighted support on the entire dataset and a dataset with removed features. The aim of clustering was to prove the effectiveness of RIAS algorithm.

Performance Evaluation

Evaluation is based on comparison of algorithms and computation time. Comparison was looking at how well RIAS algorithm scales with RIPPER in terms of accuracy.

Other parameters such as dataset and threshold were also used for evaluation. The KDD Cup ’99 dataset is used as a benchmark because most research on intrusion detection problems with machine learning have been applied on this dataset [87]. The threshold value is set by the programmer and it is not definite, so the study looked at thresholds of 0.1, 0.15, 0.2, and 0.25 for RIAS and a threshold of 0.5 for RIPPER and clustering. For instance in RIAS, the threshold determines whether a given rule is a positive example or a negative example and whether accuracy of a given rule data based on train data affects which feature is to be removed. Therefore the threshold determines both the accuracy and the number of features to be removed and indirectly which feature will be selected for removal. In clustering, the threshold determines

the number of clusters built thereby affecting accuracy. When a feature is removed accuracy increases and also the number of clusters increases implying more clusters better accuracy.

Results and Discussion

Case	0.1	0.15	0.2	0.25
100	4.79	4.51	4.55	4.54
500	577.01	536.47	529.77	582.04
1000	5374.45	5271.42	6724.88	6553.89
2000	56121.53	57626.1	61346.81	91364.17

Table 1: Execution time for RIAS with a threshold of 0.1, 0.15, 0.2 & 0.25

Case	0.5
100	6.70
500	480.66
1000	901.28
2000	902.47

Table 2: Execution time for RIPPER with a threshold of 0.5

Case	0.1	0.15	0.2	0.25
100	2.72	2.81	2.86	2.92
500	13.93	13.69	13.92	13.62
1000	27.77	28.36	25.14	25.16
2000	53.98	51.83	51.85	49.03

Table 3: Accuracy for RIAS with a threshold of 0.1, 0.15, 0.2 & 0.25

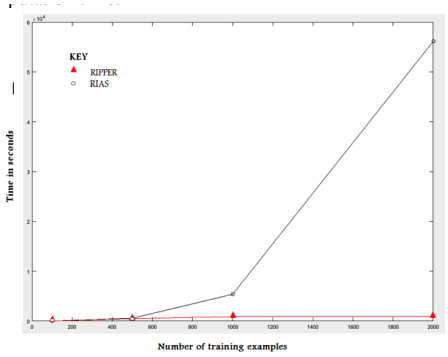
Case	0.5
100	0.44
500	0.45
1000	0.49
2000	0.48

Table 4: Accuracy for RIPPER with a threshold of 0.5

Observing Table 1 and Table 3, column wise shows that for a smaller threshold of 0.1, it would give higher accuracy and lower computation time. There is likelihood in the experiment that as you increase threshold, accuracy tends to decrease and also computation time increases. This implies that if we lower the threshold below 0.1 we are likely to get higher accuracies, more features selected and possibly a decrease on the computation time.

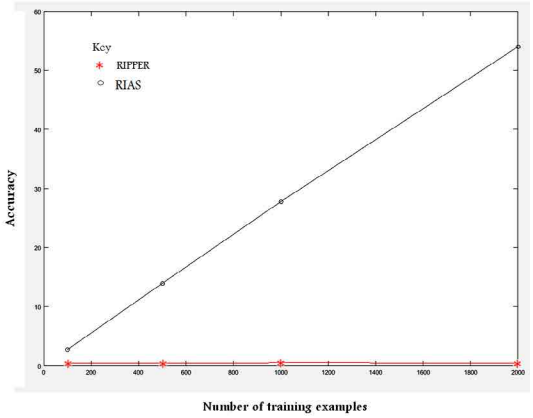
The results obtained from running RIAS and RIPPER algorithm on the KDD Cup dataset were applied on MATLAB 8.5.0(R2015a) for data visualization in order to help analyze the performance of the algorithms in terms of accuracy and computation time.

Figure 3: The Time cost of RIPPER and RIAS



At the start when the size of data is small RIAS's time complexity is better however when the size of data increases i.e looking at 500 case, 1000 and 2000 case, RIPPER's time complexity looks constant yet RIAS 's time complexity grows exponentially meaning that in terms of time complexity RIPPER is better than RIAS. From Figure 3 it can be noted that the execution time of RIPPER is linear (O 2n) whereas for RIAS is quadratic. Running RIAS is like looking at a matrix of n rows and n columns, meaning the number of operations will be n2. Therefore a training set of size n, RIPPER's performance scales as O(nlog2n) and that of RIAS as O(nlog2n^2).

Figure 4: The Accuracy of RIPPER and RIAS



With accuracy, RIAS is better than RIPPER (as seen in Figure 4) and this is because RIAS treats the whole dataset as training data while RIPPER uses two thirds of the dataset for training. The accuracy of RIAS also grows exponentially when the size of data increases. Much as this study was able to achieve high accuracy values for RIAS, however the computation time of RIAS is not as good as that of RIPPER. We still argue that the main interest in this work is on the accuracy other than the computational time because it could be more costly to act on false alarms than the computation time.

When RIAS was applied to the 100, 500, 1000, and 2000 case while applying thresholds of 0.1, 0.15, 0.2 and 0.25, feature "num_outbound_cmds" in the KDD Cup dataset was removed. As the size of training examples increases more features are selected for removal though the only challenge with that is that a bigger size of data takes a lot of computing resources but we are able to achieve higher accuracy values. When Running RIPPER for all the cases (100, 500, 1000 and 2000) with a threshold hold of 0.5 features "dst_host_count", "dst_host_rerror_rate" and "dst_host_srv_rerror_rate" are identified as irrelevant features and thus are not included in the final dataset.

Table 5: clustering Accuracy before and after removing irrelevant features

Case	Cluster Accuracy	No. of clusters created
Full KDD dataset	470.43	599
One Feature removed	475.22	605

Through clustering based on weighted support, we tested the effectiveness of RIAS algorithm. The first case we clustered the entire KDD cup data set and in the second case we clustered the selected attributes of the KDD cup data after running RIAS. Results show that the cluster accuracy in the second case is higher than the first case as reflected in table 5, meaning RIAS is useful and effective.

Conclusion

IDS depend on features we rule as redundant or irrelevant to remove and through that the IDS can be enhanced. Data mining integrated with an IDS helps identify the relevant, hidden trends and associations from a large bulk of information for effectiveness and with less execution time [37]. In this research, we investigated the possibility of enhancing feature selection in IDS. In doing so, current trends in IDS were examined, an algorithm to enhance performance of IDS technologies designed and the performance of the designed algorithm evaluated. The experimental results show that the designed algorithm is able to reduce features achieving a higher accuracy as compared to RIPPER classifier, from this the researcher was able to test the sensitivity or effectiveness of the designed algorithm.

RIAS algorithm developed as a result of this research helps remove redundant or irrelevant features which contributes to near-elimination of false positive alarms in IDS. RIAS algorithm should work on any given large dataset or big data.

Since this research was based on data simulated by the MIT Lincoln lab to generate an algorithm for detection. Based on the research results, it was concluded that the RIAS performance is better on bulky data in terms of its accuracy as compared to RIPPER. This is because RIAS compares each tuple with other tuples a task RIPPER does not perform. The research can conclude that RIAS is a more efficient algorithm when handling bulky data in terms of accuracy. For future work emphasis should be put on one improving the computational time of the designed algorithm and two on detecting real time traffic to classify it using association rule as either normal or intrusion upon entry. The research suggests that frequent sequence using association rules be applied on real-time traffic.

References

[1] E. S. Al-Shaer and H. H. Hamed, "Discovery of policy anomalies in distributed firewalls," in INFOCOM 2004. Twenty-third AnnualJoint Conference of the IEEE Computer and Communications Societies, vol. 4. IEEE, 2004, pp. 2605–2616.

[2] S. Balakrishnan, K. Venkatalakshmi, and A. Kannan, "Intrusion detection system using feature selection and classification technique," International Journal of Computer Science and Application, 2014.

[3] R. Bhadauria and S. Sanyal, "Survey on security issues in cloud computing and associated mitigation techniques," arXiv preprint arXiv:1204.0764, 2012.

[4] N. Chakraborty, "Intrusion detection system and intrusion prevention system: A comparative study," International Journal of Computing and Business Research (IJCBR), vol. 4, no. 2, pp. 1–8, 2013.

[5] S.-H. Choi and H.-S. Chae, "Feature selection using attribute ratio in nsl-kdd data," in International Conference Data mining, Civil and Mechanical Engineering (ICDMSMEŠ2014), Bali (Indonesia), Fecb, 2014, pp. 4–5.

[6] R. K. Cunningham, R. P. Lippmann, D. J. Fried, S. L. Garfinkel, I. Graf, K. R. Kendall, S. E. Webster, D. Wyschogrod, and M. A. Zissman, "Evaluating intrusion detection systems without attacking your friends: The 1998 darpa intrusion detection evaluation," DTIC Document, Tech. Rep., 1999.

[7] F. Cuppens, N. Cuppens-Boulahia, and J. Garcia-Alfaro, "Detection and removal of firewall misconfiguration," in Proceedings of the 2005 IASTED International Conference on Communication, Network and Information Security, vol. 1, 2005, pp. 154–162.

[8] V. Das, V. Pathak, S. Sharma, M. Srikanth, G. Kumar, and T. Nadu, "Network intrusion detection system based on machine learning algorithms," 2010.

[9] I. Düntsch and G. Gediga, "Rough set data analysis–a road to non-invasive knowledge discovery," 2000.

[10] U. Fayyad, G. Piatetsky-Shapiro, and P. Smyth, "From data mining to knowledge discovery in databases," AI magazine, vol. 17, no. 3, p. 37, 1996.

[11] M. G. Gouda and A. X. Liu, "Structured firewall design," Computer networks, vol. 51, no. 4, pp. 1106–1120, 2007.
[12] S. Hettich and S. Bay, "Kdd cup 1999 data," The UCI KD Archive, Irvine, CA: University of California, Department of Information and Computer Science, 1999.
[13] S. Joshi and V. S. Pimprale, "Network intrusion detection system (nids) based on data mining," International Journal of Engineering Science and Innovative Technology (IJESIT), vol. 2, no. 1, pp. 95–98, 2013.
[14] P. Kalarani and S. S. Brunda, "A survey on efficient data mining techniques for network intrusion detection system (ids)." International Journal of Advanced Research in Computer and Communication Engineering, vol. 3, no. 9, pp. 8028–8031, 2014.
[15] K. Kendall, "A database of computer attacks for the evaluation of intrusion detection systems," DTIC Document, Tech. Rep., 1999.
[16] W. Lee, S. J. Stolfo, and K. W. Mok, "Adaptive intrusion detection: A data mining approach," Artificial Intelligence Review, vol. 14, no. 6, pp. 533–567, 2000.
[17] A. McCue, "Beware the insider security threat," CIO Jury, 2008.
[18] A. A. Olusola, A. S. Oladele, and D. O. Abosede, "Analysis of kddŠ99 intrusion detection dataset for selection of relevance features," in Proceedings of the World Congress on Engineering and Computer Science, vol. 1, 2010, pp. 20–22.
[19] M. Panda and M. R. Patra, "Ensembling rule based classifiers for detecting network intrusions," in Advances in Recent Technologies in Communication and Computing, 2009. ARTCom'09. International Conference on. IEEE, 2009, pp. 19–22.
[20] S. Peisert, M. Bishop, and K. Marzullo, "What do firewalls protect? an empirical study of firewalls, vulnerabilities, and attacks," 2010.
[21] H. B. RAIS and T. MEHMOOD, "Feature selection in intrusion detection, state of the art: A review." Journal of Theoretical & Applied Information Technology, vol. 94, no. 1, 2016.
[22] U. Ravale, N. Marathe, and P. Padiya, "Feature selection based hybrid anomaly intrusion detection system using k means and rbf kernel function," Procedia Computer Science, vol. 45, pp. 428–435, 2015.
[23] S. Sanyal, A. Shelat, and A. Gupta, "New frontiers of network security: The threat within," in Information Technology for Real World Problems (VCON), 2010 Second Vaagdevi International Conference on. IEEE, 2010, pp. 63–66.
[24] B. Shah and B. H. Trivedi, "Reducing features of kdd cup 1999 dataset for anomaly detection using back propagation neural network," in Advanced Computing & Communication Technologies (ACCT), 2015 Fifth International Conference on. IEEE, 2015, pp. 247–251.
[25] C. Song and K. Ma, "Design of intrusion detection system based on data mining algorithm," in 2009 International Conference on Signal Processing Systems. IEEE, 2009, pp. 370–373.
[26] U. Stanczyk and L. C. Jain, Feature selection for data and pattern recognition. Springer, 2015.
[27] A. H. Sung and S. Mukkamala, "The feature selection and intrusion detection problems," in Advances in Computer Science-ASIAN 2004. Higher-Level Decision Making. Springer, 2004, pp. 468–482.
[28] Symantec Corporation. (April 2015(accessed June 5, 2017)) Internet security threat report. [Online]. Available:https://www.symantec.com/content/en/us/enterprise/other_resources/21347933_GA_RPTinternetsecuritythreatreportvolume202015.Pdf
[29] M. Tavallaee, E. Bagheri, W. Lu, and A. A. Ghorbani, "A detailed analysis of the kdd cup 99 data set," in Computational Intelligence for Security and Defense Applications, 2009. CISDA 2009. IEEE Symposium on. IEEE, 2009, pp. 1–6.
[30] The UCI KDD Archive. ((accessed December 5, 2016)) Kdd cup 1999 data. [Online]. Available: kdd.ics.uci.edu/databases/kddcup99/kddcup99.html
[31] unb. ((accessed December 5, 2016)) Nsl-kdd data set for network-based intrusion detection systems. [Online]. Available: http://www.unb.ca/cic/research/datasets/nsl.html
[32] D. M. Upton and S. Creese, "The danger from within," Harvard business review, vol. 92, no. 9, pp. 94–101, 2014.
[33] T. E. Uribe and S. Cheung, "Automatic analysis of firewall and network intrusion detection system configurations," Journal of Computer Security, vol. 15, no. 6, pp. 691–715, 2007.
[34] L. Yuan, H. Chen, J. Mai, C.-N. Chuah, Z. Su, and P. Mohapatra, "Fireman: A toolkit for firewall modeling and analysis," in Security and Privacy, 2006 IEEE Symposium on. IEEE, 2006, pp. 15–pp.
[35] D. Zheng and C. Zhang, "Selecting feature subset for large-scale data via fuzzy rough approach," Journal of Convergence Information Technology, vol. 8, no. 9, p. 109, 2013.

[36] Q. Zhou and Y. Zhao, "The design and implementation of intrusion detection system based on data mining technology," Research Journal of Applied Sciences, Engineering and Technology, vol. 5, no. 14, pp. 3824–3829, 2013.
[37] G. Nadiammai and M. Hemalatha, "Effective approach toward intrusion detection system using data mining techniques," Egyptian Informatics Journal, vol. 15, no. 1, pp. 37–50, 2014