Signature-based Denial of Service and Probe Detection, a Machine Learning approach

IJOTM
ISSN 2518-8623

Volume 3. Issue II
p. 11, Dec 2018
http://jotm.utamu.ac.ug
email: ijotm@utamu.ac.ug

Claire Babirye
cbabirye@utamu.ac.ug
Uganda Technology and Management University

Ernest Mwebaze
Uganda Technology and Management University

Abstract

Computer Networks and the internet are increasingly becoming the backbone of our social fabric. However, because of the diverse characteristics of these networks they are prone to various attacks and as a result the computer networks need to be highly secured to ensure confidentiality, integrity and availability of information. Presently, a key strategy in subduing these attacks is by use of Intrusion Detection (ID). Intrusion Detection Systems (IDSs) are used to detect attacks on a network. However, the uniqueness and frequency of these attacks calls for novel approaches such as the use of machine learning techniques to model the network traffic as it changes and detect anomalous traffic. In this paper we present some work on the detection of these Denial Of Service(DOS) and Probe attacks in network traffic using machine learning and data mining techniques. We build our models based on the common KDD dataset as well as live data from a wireless network at an institution of learning that has numerous and diverse users. We show the efficacy of machine learning algorithms for detecting these two attacks.

Key words: *Denial of Service, Network Security, Machine Learning*

Introduction

Background

Data communication networks such as the Internet are increasingly being used to connect millions of computers and personal networks at various organizations. There is an ever increasing dependency on these networks in all aspects of life. However, in parallel with the ever increasing network sizes has been a concomitant increase in the network traffic data which can contain highly confidential and valuable information communicated over the network [1].

To the network administrators and analysts this traffic is resourceful to understand network behavior, provide quality of service and set proper information security policies through monitoring network misuse and ensuring network security. It is very important to maintain a high level of network security to ensure safe and trusted communication of information between various organizations. However, secured data communication over the Internet and any other networks is always under threat of intrusions and misuses.

To control these threats, recognition of attacks is a critical matter. These attacks can be recognized through network monitoring; a continuous process that involves inspecting any or all kind of traffic that traverses a particular system or network of interest. Network monitoring is aimed at quickly detecting anomalies with in traffic behaviour such as attacks initiated by perpetrators looking to bring down a system or destroy or steal sensitive information [2].

Network monitoring has been facilitated by the use of Intrusion Detection Systems (IDS). An IDS can be classified as an anomaly IDS or a signature-based IDS. The signaturebased detection system performs the monitoring for intrusion through matching audit data with known patterns of intrusive network behavior while anomaly detection systems identify abnormalities from the normal network behavior. To recognize traffic as an attack, an IDS must be trained to recognize normal network activity.

However, there are no known models for normal network behavior, making it hard to develop an anomaly detector in the strictest sense [3]. Based on the inherent complexity in characterizing the normal network behavior, the problem of anomaly detection can be categorized as model-based and non-model based. In model-based anomaly detectors, it is assumed that a known model is available for the normal behavior of certain aspects of the network and any deviation from the norm is deemed as an anomaly [4]. In non-model based anomaly detection systems, no model is assumed.

A signature-based detector model must have access to a large library of data that can provide the required samples from which accurate estimates of a legitimate network behavior and anomaly like network behaviour is made [5]. Incoming patterns that match an element of the library are labelled as attacks. Unknown attacks that do not deviate much from the attacks listed in the library can be detected and labelled as neighbouring attacks.

In this study we propose a network-based misuse detector model based on machine learning techniques to be applied in the prediction of legitimate network behavior and behavior that deviates from the normal state referred to as anomalies. We focus on mainly 2 common attacks: DOS attacks and Probe attacks. These attacks affect the most networks globally on a daily basis [5] and thus the detection of the same is a profound research topic for researchers throughout the world. We carried out the study using the Knowledge Discovery in Data (KDD) data, a widely used data set to evaluate intrusion detection systems.

The next sections of this paper give a concise literature review of related work and methods. This is followed by sections that discuss the datasets we used, the experiments that were done with several machine learning algorithms and the results accruing from the experiments. We conclude with a discussion of the results and conclusions to the paper.

Related Work

Network Traffic

Networks are mainly known to facilitate communication and information sharing, this makes them indispensable since information and communication are two of the most important strategic issues for the success of every enterprise [7].

Nearly today every organization uses a substantial number of computers and communication tools that are facilitated by networks such as the Internet to run day to day activities. Internet is a network of networks that facilitates various services such as online communication, information sharing while overcoming geographic separation problem.

These various activities that take place on the network form network traffic or data traffic and that is the amount of data moving across the network at a given point in time. This traffic is mainly categorized into two; Legitimate traffic and lethal traffic. Legitimate traffic includes the legit packets that are sent by a legitimate user on the network without any bad motive. Lethal traffic those are the malicious packets sent on the network sent by an attacker who lies somewhere on the network. Such packets are sent by attackers who have different bad motives. Malicious packets are sent with an intent of exploiting a vulnerability on the network and thus launching some form of attack. From thousands of known exploits [8,9,10], describe a taxonomy of attacks, grouping them into four categories: probes, Denial of service attacks, Remote to Local attacks and User to Root attacks. These are explained below:

- Probe attacks - These are launched when an attacker is testing a potential target to gather information[10] [11]. They are operated with an intention of identifying a weakness in a machine that can be exploited so as to compromise the system [8]. They are usually harmless (and common) unless a vulnerability is discovered and later exploited. According to [12], it is known that before launching the attack, the attacker selects a target and gathers information. Probes can be launched through a couple of activities such as: inside sniffing, port scans,ip sweep,vulnerability testing [9,15].
- Denial Of Service attacks - also known as DOS attacks, such aim at preventing normal operation of the network, such as causing the target host or server to crash, or blocking the network traffic [8]. This happens through overwhelming the target with high volumes of traffic making it unavailable to legitimate users [2,13].Such attacks happen through SYN flooding, session hijacking, and malicious programs, and they degrade the performance of a network.
- User to Root - these are attacks in which an authenticated user bypasses normal authentication gaining the privileges of another user, usually root [14].
- Remote to Local - unlike user to root attacks, for this case the root privileges are gained by an unauthorized user who is able to bypass normal authentication through exploiting the vulnerabilities in the system [8,14].

Intrusion Detection

Intrusion is any set of actions that attempt to compromise the integrity, confidentiality, or availability of a resource. An intrusion detection system (IDS) [17] is a system for the detection of such intrusions. There are three main components of an IDS: data collection, detection, and response [17]. Shakshuki et al [17] further describe these components: the data collection component is responsible for collection and preprocessing data tasks: transferring data to a common format, data storage and sending data to the detection module. IDS can use different data sources as inputs to the system: system logs, network packets, etc. In the detection component data is analyzed to detect intrusion attempts and indications of detected intrusions are sent to the response component [17]. We review 2 intrusion techniques in this study: Anomaly-Based Intrusion Detection and Misuse-Based Intrusion Detection.

Anomaly-Based Intrusion Detection

The approach looks out for any feature that is out of the ordinary. The ordinary can be defined with respect to the history of the test signal (unsupervised) or with respect to a collection of training data (semi-supervised) [18]. It is based on statistical behavior modeling [19], [20] normal operations of the members are profiled [19], [21] such as CPU usage [17] and any deviation from the normal behavior is flagged as an anomaly [20], [22]. The model of the normal behavior of the network is extracted [23] and it's compared with the current behavior of the network [20] to detect intrusion [17]. Adnan and Michael [20], [22] describe two phases of operation in anomaly detection systems: testing and training.

They describe training as a process of modelling the normal or expected behavior of the network or users, thus for any anomaly based IDS to be effective it must have a consistent and stable profile that

characterizes this behavior. Not only that they discuss the testing phase as a process which involves comparing the normal or expected behavior model derived during the training phase with the current model of the network or users.

However, this detection type presents a challenge as described by [19] of periodically updating the normal profiles since the network must change rapidly which may increase the load on the hardware resources used and thus making it expensive [17]. Sergio et al [24] discusses that this approach cannot be easily deployed in MANETs since the mobility and flexibility of MANETs nodes, harden the definition of normal and malicious behavior. They go on to say the mobility of nodes leads to changes in the network topology, increasing the complexity of the detection process. It is also more prone [22] to generate false positives than knowledge-based intrusion detection.

Misuse-Based Intrusion Detection

Misuse-based intrusion detection (MBID) is also known as knowledge-based Intrusion Detection (KBID) [20], [22], signature-based intrusion detection [17], [18], [19], rule-based [19], pattern-based detection [18], supervised detection, intruder profiling. In this technique, a knowledge base [20] is maintained that contains signatures or patterns of [17], [19] well-known attacks and looks for these patterns in an attempt to detect a specific pattern [22] of misbehavior. An example of a signature [19] would be: "there are 3 login attempts within 5 minutes" for a brute force attack. Signature analysis [20] is also used by misuse-based intrusion detection, where the attacks or modelled through a sequence of events or patterns, which are then compared with the generated audit trails to indicate intrusion. Further Adnan and Michael [20] describe some Knowledge-Based Intrusion detection Systems (KBIDs) as those which apply rule based approaches to model the knowledge of known attacks in the form of a set of rules which are obtained through observations or by considering attack scenarios.
The main advantage of this technique is that [19] it can accurately and efficiently detect known attacks hence it has [17], [18], [22] a low positive rate and thus preferred for commercial IDs.

Ismail et al [19], described the main distinction between the anomaly based intrusion detection and misuse based intrusion detection as: "anomaly detection systems try to detect the effect of bad behavior but misuse detection systems try to recognize known bad behavior."

Current Detection Methods

Detection of DoS and Probe attacks using the genetic Algorithm The algorithm is based on the Darwin's theory of evolution; with a basic rule of Survival for the fittest, the algorithm handles a population of possible solutions where each solution is represented through a chromosome [8]. A chromosome is a threadlike structure of nucleic acids and protein found in the nucleus of most living cells, carrying genetic information in form of genes. The algorithm uses evolution and natural selection evolving chromosomes using selection, combination and mutation operators [25].

When the Genetic Algorithm is used for solving various problems three factors are considered to have a vital impact on the effectiveness of the algorithm and also of the applications. These factors include: fitness function, representation of the individuals and the parameters for the Genetic Algorithm. The determination of these factors often depends on applications and/or implementation [8].

How the algorithm functions in relation to detection of attacks in network traffic.

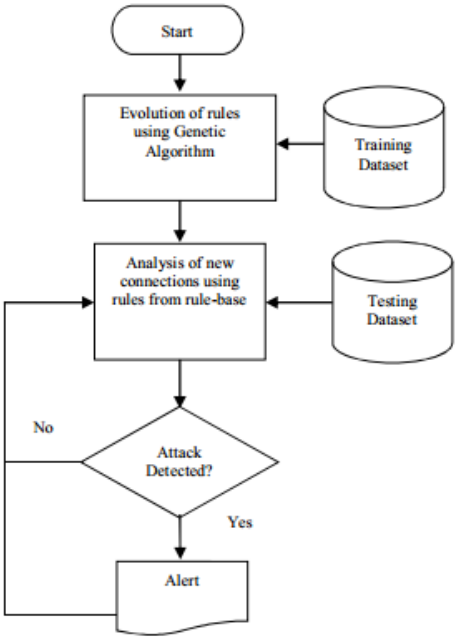
The algorithm works in two phases; learning [8] or training phase and the testing phase as shown in Figure 1.

Learning/Training phase. In the learning phase [25], network data which contains both normal network connections (normal network data) and attacks (abnormal data) is collected for audit. Then a network sniffer analyses this data and sends it to the genetic algorithm and the fitness function is applied to generate a set of rules for detecting intrusion. These rules are stored in a rule base.
The records from the learning phase are represented in the form of chromosomes. Each chromosome is a rule within which certain features of a connection are encoded in the form of fixed length vector. A fitness function is then applied to each chromosome in order to evaluate its goodness. If a chromosome helps to identify an attack correctly, it is considered good or fit else it is considered bad [25]. The algorithm proceeds with mutation and combination operators; combination and mutation operators are applied to the good chromosomes obtained from the fitness function to produce a new generation. The entire process is recurred by using the newly generated population. Thus the evolution process is repetitive until a solution is reached; a set of rules capable of detecting attacks is generated [5].

In the rule base, the rules are stored in the following format [25]: if condition then act For example, a rule can be defined as [8]: if the connection has following information: source IP address 145.33.17.6; destination IP address 160.106.20.55; destination port number: 21; connection time: 10.1 seconds then stop the connection This implies: if there exists a network connection request with source IP address 145.33.17.6, destination IP address 160.106.20.55, destination port number 21, and connection time 10.1 seconds, then stop the connection establishment - since IP address 145.33.17.6 is recognized by the IDS as a blacklisted IP address. Thus, service request initiated from it, is rejected.

Testing phase. The testing entails detection of whether a real-time network connection is a normal connection or it is an intrusive attack [25]. This is obtained using rules stored in the rule base during the training phase. Since the algorithm is rule-based, if the characteristics of a new connection match with the condition section of some pre-defined rule in the rule base then the connection is considered as an attack else it is considered as a normal connection [5].
The sub attack labels such as smurf, mailbomb, among others are recognized with respect to the fitness criteria by selecting the best-fit chromosomes capable of detecting the attacks from every population [8]. Incase an attack is detected then IDS performs the necessary actions as defined by the security policies of the organization.

Figure 1: Flow of the Genetic Algorithm based IDS



Detection of DoS and Probe attacks using the Principal Component Analysis (PCA)
The model uses a multivariate statistical method called Principal Component analysis to detect Denial-of-service and network Probe attacks.

Principal Component Analysis is a multivariate statistical technique [6] applied to reduce the dimension of feature vectors and to achieve parsimony by extracting the smallest number components that account for most of the variation in the original multivariate data and to summarize the data with little loss of information to enable better analysis of the data [26]. The algorithm inputs data and portions of the data sets are processed to create a new database of feature vectors which represent the IP header of the packets [6]. The feature vectors are analyzed using PCA and various statistics are generated during this process including the principal components, their standard deviations, the loading of each feature on the principal components and bi-plots to represent a graphical summary of these statistics [27].

The variance and standard deviation of a random variable are measures of dispersion. The variance is the average value of the squared deviation from the variable's mean, and the standard deviation is the square root of the variance [28]. For instance, in IPsweep attacks, one or more machines (IPs) are sweeping through a list of server machines looking for open ports that can later be utilized in an attack while in port sweep attacks, one machine is sweeping through all ports of a single server machine looking for open ports. In both cases, there is an irregular use of port numbers that causes the variance in the principle components to vary, with an associated irregularity in the loading values [27].

Weaknesses in the existing current detection methods
- It is a difficult task to represent a problem space in the genetic algorithm, find the fitness function as well as choosing parameters for the algorithm and yet such factors determine the performance and effectiveness of the algorithm [11]. Configuration of a genetic algorithm based system is also known to be a hard task.
- The PCA algorithm has scalability issues; the cause of this is twofold; the algorithm reduces on the dimensionality by removing components with large Eigen values; this affects the sample space making some anomalies not detectable or traceable.

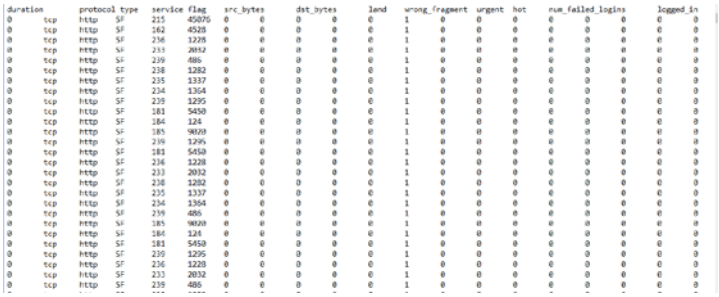
Experiments

The data
For the experiments we used two network traffic datasets. The first dataset is the popular KDD dataset that has clear feature extraction routines and has been used by several researchers in this field. The second dataset was data obtained from a live wireless network in a large academic institution with numerous users connecting to the network. The goal was to train the algorithms on the KDD dataset which has clear labelled data and test the algorithms on live network data to determine its efficacy.

Knowledge Discovery in Data (KDD) Data set
The KDD dataset is composed of 41 attributes and these are categorized into three groups[29][30]: intrinsic attributes which are extracted from the packet headers; content attributes which are extracted from the contents area of the network packets based on expert person knowledge and finally traffic attributes which are based on previous connections: those which occured in the past 2 seconds and those which occured in a sequence[6][7]. However, some of these features or attributes in the KDD data set are relevant in the detection of Denial Of Service attacks and Probe attacks whilst others are irrelevant.

The data set used had 24,972 records, of these 13,449 belonged to the normal category; 9234 belonged to the DOS category and 2289 belonged to the Probe category. The dataset included different attack types which were categorized under DoS or Probe attacks. Under the DOS category, the attack types included: Back, Land, Neptune, Pod, Smurf and teardrop whilst ipsweep, nmap, portsweep and satan belonged to the probe category[7]. Figure 2 depicts an example of this data. As is evident, this data contains a number of uniquely calculated features.

Figure 2: Sample of the KDD dataset

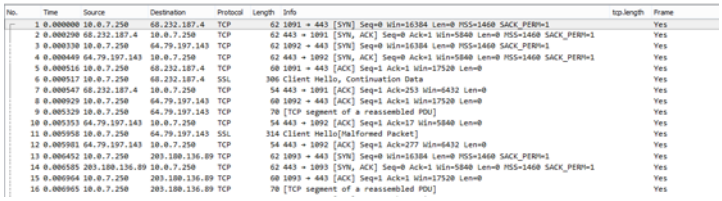


duration	protocol_type	service	flag	src_bytes	dst_bytes	land	wrong_fragment	urgent	hot	num_failed_logins	logged_in
0	tcp	http	SF	215	45076	0	0	0	0	0	0
0	tcp	http	SF	162	4528	0	0	0	0	0	0
0	tcp	http	SF	236	1228	0	0	0	0	0	0
0	tcp	http	SF	233	2032	0	0	0	0	0	0
0	tcp	http	SF	239	486	0	0	0	0	0	0
0	tcp	http	SF	238	1282	0	0	0	0	0	0
0	tcp	http	SF	235	1337	0	0	0	0	0	0
0	tcp	http	SF	234	1364	0	0	0	0	0	0
0	tcp	http	SF	239	1295	0	0	0	0	0	0
0	tcp	http	SF	181	5450	0	0	0	0	0	0
0	tcp	http	SF	164	124	0	0	0	0	0	0
0	tcp	http	SF	185	9020	0	0	0	0	0	0
0	tcp	http	SF	239	1295	0	0	0	0	0	0
0	tcp	http	SF	181	5450	0	0	0	0	0	0
0	tcp	http	SF	236	1228	0	0	0	0	0	0
0	tcp	http	SF	233	2032	0	0	0	0	0	0
0	tcp	http	SF	238	1282	0	0	0	0	0	0
0	tcp	http	SF	235	1337	0	0	0	0	0	0
0	tcp	http	SF	234	1364	0	0	0	0	0	0
0	tcp	http	SF	239	486	0	0	0	0	0	0
0	tcp	http	SF	185	9020	0	0	0	0	0	0
0	tcp	http	SF	164	124	0	0	0	0	0	0
0	tcp	http	SF	181	5450	0	0	0	0	0	0
0	tcp	http	SF	239	1295	0	0	0	0	0	0
0	tcp	http	SF	236	1228	0	0	0	0	0	0
0	tcp	http	SF	233	2032	0	0	0	0	0	0
0	tcp	http	SF	239	486	0	0	0	0	0	0

Live wireless network data

Live network traffic data from an institution of higher learning was recorded over some time. The data was collected at times when usage of the network was heavy and as such the data is rich with different variations of what ordinary normal network traffic will look like. Because this is live data, it was not possible to obtain the ground truth for this data. We used it as a validation dataset for our machine learning algorithm. Figure 3 depicts and example of this data.

Figure 3: Sample of the dataset collected on the institution network WiFi interface



No.	Time	Source	Destination	Protocol	Length	Info	tcp.length	Frame
1	0.000000	10.0.7.250	68.232.187.4	TCP	62	1091 → 443 [SYN] Seq=0 Win=16384 Len=0 MSS=1460 SACK_PERM=1		Yes
2	0.000290	68.232.187.4	10.0.7.250	TCP	62	443 → 1091 [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MSS=1460 SACK_PERM=1		Yes
3	0.000330	10.0.7.250	64.79.197.143	TCP	62	1092 → 443 [SYN] Seq=0 Win=16384 Len=0 MSS=1460 SACK_PERM=1		Yes
4	0.000449	64.79.197.143	10.0.7.250	TCP	62	443 → 1092 [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MSS=1460 SACK_PERM=1		Yes
5	0.000516	10.0.7.250	68.232.187.4	TCP	60	1091 → 443 [ACK] Seq=1 Ack=1 Win=17520 Len=0		Yes
6	0.000517	10.0.7.250	68.232.187.4	SSL	306	Client Hello, Continuation Data		Yes
7	0.000547	68.232.187.4	10.0.7.250	TCP	54	443 → 1091 [ACK] Seq=1 Ack=253 Win=6432 Len=0		Yes
8	0.000929	10.0.7.250	64.79.197.143	TCP	60	1092 → 443 [ACK] Seq=1 Ack=1 Win=17520 Len=0		Yes
9	0.005329	10.0.7.250	64.79.197.143	TCP	70	[TCP segment of a reassembled PDU]		Yes
10	0.005353	64.79.197.143	10.0.7.250	TCP	54	443 → 1092 [ACK] Seq=1 Ack=17 Win=5840 Len=0		Yes
11	0.005958	10.0.7.250	64.79.197.143	SSL	314	Client Hello[Malformed Packet]		Yes
12	0.005981	64.79.197.143	10.0.7.250	TCP	54	443 → 1092 [ACK] Seq=1 Ack=277 Win=6432 Len=0		Yes
13	0.006452	10.0.7.250	203.180.136.89	TCP	62	1093 → 443 [SYN] Seq=0 Win=16384 Len=0 MSS=1460 SACK_PERM=1		Yes
14	0.006585	203.180.136.89	10.0.7.250	TCP	62	443 → 1093 [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MSS=1460 SACK_PERM=1		Yes
15	0.006964	10.0.7.250	203.180.136.89	TCP	60	1093 → 443 [ACK] Seq=1 Ack=1 Win=17520 Len=0		Yes
16	0.006965	10.0.7.250	203.180.136.89	TCP	70	[TCP segment of a reassembled PDU]		Yes

Feature Extraction

Intrinsic, content and traffic attributes were extracted from the data set collected on the wireless interface using Clion C++ software.

Feature Selection

Effectiveness of an intrusion detection model or any model to be used in prediction is dependent on the features selected for use in building the model on a given task. Without eliminating the irrelevant features prior to training phase of the model; the model size is susceptible to increment in size, computational cost and decrementing in its performance in terms of the performance metric.

We determine the relevancy of each feature in the data set based on information gain, a concept that helps in discovery of how much information each feature in the data space has on each target class in the data.

Machine learning based detection

We worked with four machine learning classification algorithms: Random Forest Algorithm, Decision Trees, Support Vector Machines(SVM) and K-Nearest Neighbors (KNN) and present the results in a confusion matrix as shown in Figure 2.

Random Forest Algorithm

Random Forest algorithm is an ensemble learning method for classification, regression and other tasks that operates by constructing a multitude of decision trees which forms a forest[30], at training time and outputting the class that is the mode of classes for a classification task or problem. The algorithm corrects for decision trees' habit of overfitting to their training set [33].

K-NN Algorithm

K-NN is a classification algorithm used under supervised learning. The idea is to search for closest match of the test data in feature space. Here, if a sample point has features similar to the ones of points of a particular class, then it belongs to that class. These points are known as nearest neighbors. The algorithm also involves a parameter k that specifies the number of neighbors (neighboring points) used to classify one particular sample point. Finally, the assignment of a sample to a particular class is done by having the k neighbors considered to be legal. In this fashion, the class represented by the largest number of points among the neighbors ought to be the class that the sample belongs.

SVM Algorithm

A support vector machine is a supervised machine learning algorithm used for data classification and estimating the relationships between variables. It is a supervised algorithm because there is an initial training phase involved where you feed the algorithm data that has already been classified (labeled).
After this initial training phase is completed, future data sets given to the algorithm can be classified with no or minimal human intervention.

Decision Trees

The general motive of using Decision Tree is to create a training model which can use to predict class or value of target variables by learning decision rules inferred from prior data which is the training dataset[35]. The decision tree algorithm tries to solve the classification problem, by using tree representation as shown in Figure 4. Each internal node of the tree corresponds to an attribute, and each leaf node corresponds to a class label.
Entropy, a measure of impurity in the dataset is used to determine how informative a particular input attribute is about the output attribute for a subset of the training data.

Figure 4: Basic structure of a Decision Tree

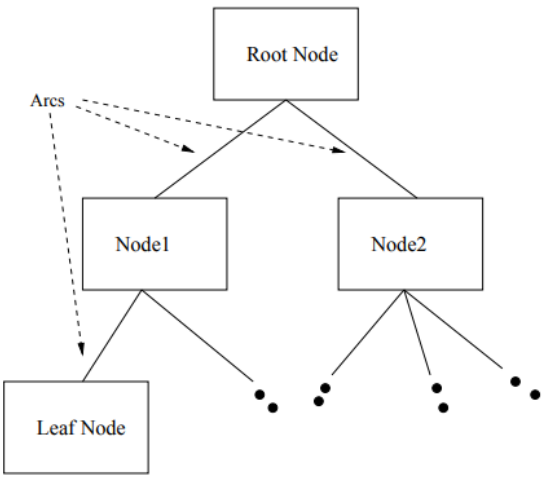
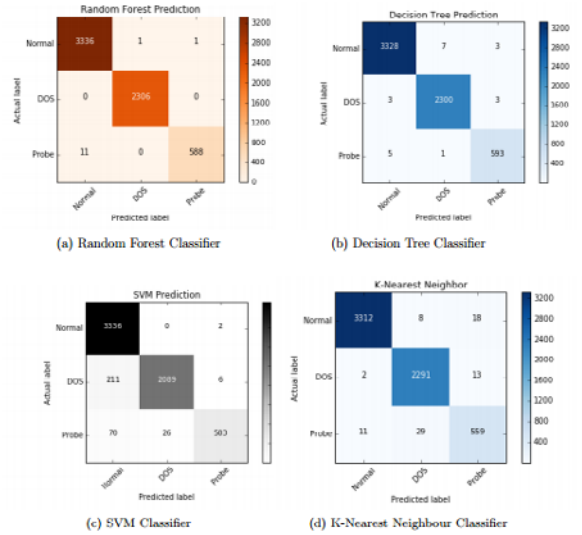


Figure 5: Predicted results from the classification algorithms used



(a) Random Forest Classifier (b) Decision Tree Classifier

(c) SVM Classifier (d) K-Nearest Neighbour Classifier

Results

We measured and assessed the performance of the classifiers based on different evaluation metrics such as accuracy, precision, recall, F_Score and False Positive Rate. We computed this using the values depicted in the confusion matrix that is True Positives (TP), True Negatives (TN), False Positives(FP) and False Negatives (FN).The rates obtained under each metric is depicted in Figure 5.

After computation of information gain using Random Forest algorithm, 21 features of the 41 features in the KDD data set were considered as the most relevant, since they had a high information gain.

Figure 6: Performance of classifiers based on Evaluation Metrics

Random Forest Classifier			
Accuracy	**Precision**	**Recall**	**F_Score**
99.79	1.00	1.00	99.79
Decision Tree Classifier			
Accuracy	**Precision**	**Recall**	**F_Score**
99.64	1.00	1.00	99.64
KNN			
Accuracy	**Precision**	**Recall**	**F_Score**
98.70	0.99	0.99	98.69
SVM			
Accuracy	**Precision**	**Recall**	**F_Score**
94.95	0.95	0.95	94.89

Random Forest and Decision Trees where the best classifiers; as depicted in Figure 6 and this is because in a feature space of multi-features; the algorithms calculate the entropy of every feature to measure the impurity in the data and also decide effectively on how to split the data. The performance of K-NN varies depending on the number of neighbors considered during the testing phase.

On a general perspective, 95% of the data records were correctly classified with a negligible false positive rate. The ones correctly identified are the ones which appeared on the main diagonal in the confusion matrix. This was as a result of building the model on the relevant features in categorizing each target class; and thus malicious traffic was more likely to be detected on unknown future datasets.

Discussion

In this work we presented an alternate machine learning based approach to developing DOS and probe detection systems. Having robust IDS system is key in this era of big data. The results show that this approach can have very good performance. Performance is evaluated using four machine learning algorithms which are differently oriented. As evident from the tables, decision tree based algorithms tend to provide superior performance compared with the other algorithms.

One advantage with decision tree algorithms is they provide interpretability of the results. A network administrator can clearly tell what the algorithm is doing by looking at the decision tree and by tuning different parameters can determine how strict the IDS is by controlling the depth of the tree for example. We extend this idea of interpretability to tease out the most relevant features that influence the performance of the algorithm greatly. Knowing which features most affect the performance of the algorithms is important because the system administrator or the person using the IDS can deliberately intervene on these to control the strength of the security of the network.

Conclusion

IDS have been presented as a security tool to subdue these attacks. A big percentage of these attacks can be viewed as normal traffic if the IDS is poorly configured; and as a result selection of relevance features prior to design of the model is very substantial. In this study we focused on detection of DOS and Probe attacks using data mining techniques. Relevant features were determined using the concept of information gain to reduce on the bias towards multivalued attributes and thus decrease on the error percentage. This model can be deployed in the detection of DOS and Probe attacks in offline network traffic with a high accuracy score.

References

[1] Becker, R. A., Eick, S. G., & Wilks, A. R. (2015). Visualizing network data. IEEE Transactions on visualization and computer graphics, 1(1), 16-28.
[2] Aggarwal, N., & Dhankhar, K. (2014). Attacks on Mobile Adhoc Networks: A Survey. International Journal of Research in Advent Technology, 2(5), 307-316.
[3] Thottan, M., Liu, G., & Ji, C. (2010). Anomaly detection approaches for communication networks. In Algorithms for Next Generation Networks (pp. 239-261). Springer London.
[4] Jyothsna, V., Prasad, V. R., & Prasad, K. M. (2011). A review of anomaly based intrusion detection systems. International Journal of Computer Applications, 28(7), 26-35.
[5] Kshirsagar, V. K., Tidke, S. M., & Vishnu, S. (2012). Intrusion detection system using genetic algorithm and data mining: An overview. International Journal of Computer Science and Informatics, 2231, 5292.
[6] Labib, K., & Vemuri, V. R. (2004, June). Detecting and visualizing denialof-service and network probe attacks using principal component analysis. In Third Conference on Security and Network Architectures, La Londe,(France).
[7] Patra, M. P. M. R. (2009). Evaluating machine learning algorithms for detecting network intrusions. Int. J. of Recent Trends in Engineering and Technology, 1(1).
[8] Paliwal, S., & Gupta, R. (2012). Denial-of-service, probing & remote to user (R2L) attack detection using genetic algorithm. International Journal of Computer Applications, 60(19), 57-62.
[9] Kendall, K. (1999). A database of computer attacks for the evaluation of intrusion detection systems. Massachusetts Inst Of Tech Cambridge Dept Of Electrical Engineering And Computer Science
[10] Tavallaee, M., Bagheri, E., Lu, W., & Ghorbani, A. A. (2009, July). A detailed analysis of the KDD CUP 99 data set. In Computational Intelligence for Security and Defense Applications, 2009. CISDA 2009. IEEE Symposium on (pp. 1-6). IEEE

[11] Kayacik, H. G., Zincir-Heywood, A. N., & Heywood, M. I. (2005, October). Selecting features for intrusion detection: A feature relevance analysis on KDD 99 intrusion detection datasets
[12] Hansman, Simon Luke. "A taxonomy of network and computer attack methodologies." (2003). [13] Douligeris, C., & Mitrokotsa, A. (2004). DDoS attacks and defense mechanisms: classification and state-of-the-art. Computer Networks, 44(5), 643-666.