ATTACK DETECTION AVAILING FEATURE DISCRETION USING RANDOM FOREST CLASSIFIER

Anne Dickson[1] and Ciza Thomas[2]

[1]Assistant Professor, Department of Computer Science & Engineering, Mar Baselios College of Engineering & Technology, Trivandrum, Kerala
[2]Senior Joint Director, Directorate of Technical Education, Trivandrum, Kerala

ABSTRACT

The widespread use of the Internet has an adverse effect of being vulnerable to cyber attacks. Defensive mechanisms like firewalls and IDSs have evolved with a lot of research contributions happening in these areas. Machine learning techniques have been successfully used in these defense mechanisms especially IDSs. Although they are effective to some extent in identifying new patterns and variants of existing malicious patterns, many attacks are still left as undetected. The objective is to develop an algorithm for detecting malicious domains based on passive traffic measurements. In this paper, an anomaly-based intrusion detection system based on an ensemble based machine learning classifier called Random Forest with gradient boosting is deployed. NSL-KDD cup dataset is used for analysis and out of 41 features, 32 features were identified as significant using feature discretion. Our observations confirm the conjecture that both the feature selection and stochastic based genetic operators improves the accuracy and the effectiveness. The training time is shown to be reduced tremendously by 98.59% and accuracy improved to 98.75%.

KEYWORDS

Statistical Traffic Properties, Traffic Classification, Segmentation, Deep Packet Inspection, Intrusion Detection System

1. INTRODUCTION

In the digitized world, the Internet has become an integral part of our life. Now all the transactions are becoming online and all are living in an online era. All the important transactions and documents are transferred using online ,e-mail, etc. As we depend too much on these services offered by internet the crime related to these are also exponentially increasing. So here comes the importance of an intrusion detection system (IDS), which forms the second layer of defense.It is our duty to keep our data credentials secure. Social engineering is the ultimate data source of real-time cyber threats. The enormous growth in internet applications leads to the challenging growth in cyber security. Exponential growth of network threats inadversely affected the confidentiality, integrity and availability which are the basic principles of information security. Firewall, IDS which are considered as the wall of defence failed to detect modern attack scenario. Deep network packet inspection and network behaviour analysis is not done appropriately by current IDS. Hence, analyzing and monitoring the network systems to detect anomalies and network threats are supposed to perform using variant approaches using integrated IDS such as machine learning, deep learning and other hybrid methods.

Intrusion detection is the process of identifying any abnormal incidents such as unauthorized access of a system or attack on a system. These systems can be implemented either in software or

hardware. The firewall is used to stop the unwanted traffic from outside. It does not indicate attacks, inside the system [1, 2]. IDS can be generally classified as anomaly and misuse-based. Misuse based detection uses the known signature to identify the attacks. It tries to match the signature of an attack with the signature in the database. However, this type of detection fails to identify a new attack. Anomaly-based detection identfies intrusion by observing deviations from normal behavior. Anything that slightly varies from normal is considered as anomaly. So in such cases the rate of false alarm will be more. However, this type of detection is suitable for detecting zero-day attacks [2] [3].

Anomaly detection can be done using various machine learning algorithms. Machine learning is a part of artificial intelligence and it learns and improves with experience. The main advantage of machine learning algorithm is the speed of detection. It uses trained data to form a model that can be used to predict the test data. Though there are several intrusion detection systems, still some attacks are not properly detected. Majority of these attacks comes under the category of minority attacks like remote to local (R2L) and user to root (U2R) [4][5]. Previous studies mentioned that the feature selection improves the speed of computation [6][7]. Feature selection determines the useful features from the whole feature set. There are mainly two types of feature selection, namely the wrapper method, and filter method. Wrapper method depends on the classifier whereas the filter method uses some suitable criteria. When coming to network trouble shooting for threat detection, we need network visibility. As intrusions increased with technology expansion, exploration of flow data traffic structure turn into an irreplaceable procedure. It is mandatory to identify the source and destination of packet flows and its configurations. Packet format should be identified precisely regardless of on-demand or full packet dissection. When the world bloom with technology exploration using computers and automation, the core challenge faced in current decade is in modelling secure network applications. IDS also faces different challenges in the areas of network topology, hardware involvement and in other functionality. The performance and availability differs with highly accessible and limited resources with different protocols.

In this paper, we propose an intrusion detection system based on random forest. Feature selection is done using the genetic algorithm. Weights are calculated for each feature. The rest of the paper is organized as: Related works are described in section II. Section III includes a brief description about theoretical background, proposed work and workflow. The experimental setup, brief discussion about NSL-KDD dataset and empirical results are described in section IV. Finally, the paper is concluded with future directions and discussions.

2. RELATED WORKS

This section reviews related works on detecting and analysing various subsisting network intrusions using varients of machine learning approaches. IDS is the most researched area among research community working in the field of network security. Anomaly based detection draws more attention than signature based due to its effectiveness in disclosing novel attacks [3].Though there are lot of work developed,the unavoidable fact is the failure in detecting serious network attacks.This problem comes in the case of the minority attack such as the R2L and U2R. This is because the number of samples of these attacks in the dataset is very less compared to the number of samples under the category of denial of service (DoS), probe and normal. If we are considering these aspects, there is a chance of getting improved detection rate.

In 2003 Mukkamala et al.[11] described a method for calculating the importance of the input attributes. They studied two classfiers named artificial neural network and support vector machine. One method deletes an attribute at a time and compares the performance of the system with full attributes. They ranked each feature as important and secondary based on the accuracy,

training time, and testing time. This is a general method irrespective of the modeling tool. Further, they described a ranking method specifically for SVM. They conclude by saying SVM performs better than the artificial neural network in terms of training time, running time, and prediction accuracy. An SVM based intrusion detection system was discussed by Pervez et al.[12]. In this work one feature is deleted and its accuracy is calculated. If the accuracy obtained is greater than the accuracy obtained using all features, the particular feature was eliminated from the dataset.

In 2012 Yinhui Li et al.[13] described four methods of feature reduction such as feature removal, sole feature, hybrid method and gradually feature removal method. They used the KDD CUP99 dataset and constructed a compact dataset by clustering and selected a small training dataset by Ant colony optimization(ACO). For the classifier purpose, RBF kernel-based SVM is used. The intrusion detection system developed by the Nelcileno Araujo et al.[14] in 2010 used a hybrid approach. First, information gain for 41 features of KDD CUP99 data set is calculated and then according to the value, the feature is ranked and the detection rate of the optimal feature is assessed. The K-means classifier was used to extract the feature with the highest information gain ratio (IGR). Rough set based feature reduction is done by Rung-Ching Chen et al.[15] in 2009.

In 2006 Wei Wang et al.[16] propose a method for identifying intrusion using principal component analysis (PCA). They also profile the behaviour of each individual attack. Fangjun Kuang et al. [17] propose an intrusion detection system (IDS) in which a feature reduction is done using kernel principal component analysis (KPCA). It is an improved version of PCA which adopts a non-linear kernel method. Cheng-Lung Hung et al. [18] describe the genetic algorithm for feature selection and parameter optimization for SVM. Aswani Kumar et al.[19] in 2017 describe an intrusion detection system in which feature selection is employed using chi-square method. Adriana et al.[20] use information gain method for feature selection. In this particle swarm optimization (PSO) and Ant colony optimization(ACO) are used for the parameter optimization. This method showed a considerable amount of reduction in the computational time and also improvement in the detection. Mostafa A.Salama and et al.[21] describes an IDS using support vector machine (SVM). In this feature reduction was done using deep belief network(DBN).

3. THEORETICAL BACKGROUND

Feature selection helps to work on problem with n dimensional feature space. Identifying the subgroup from the input variables by neglecting the irrelevant ones is said to be feature selection. There are two different types of feature selection such as supervised and unsupervised. Supervised are further categorized into wrapper, filter and intrinsic. Finding the relevant features helps in improving both accuracy and computation time. Feature extraction is a complicated task than feature selection. It is a procedure of creating new features when we could not have used raw features. This process includes some arithmetic operations on features for better extraction. Because only adequate feature extraction can yield better classification. From the input raw data, we will consider some sample of rows and columns, and it is subjected to row sampling and feature sampling.

Random Forest classifier or a regressor is a bagging technique. The base learner is decision tree. Decision tree has two properties.They are low bias and high variance. When we use many decision trees in the random forest,in full depth, it will get trained properly for our trained data set. So the error will be very less. Whenever we get new test data, these decision tree are prone to get more errors. Hence over fitting occurs. Multiple decision trees are taken in random forest. Finally we combine the decision trees for majority vote, the high variance will get converted into low variance. Because when we do row sampling and feature sampling and giving the records to

the decision tree, the decision tree turns into an expert with respect to these input samples. Hence random forest works very well with respect to most for the machine learning use cases[23]. Classifier uses majority vote whereas regression will find the mean or median of the particular output of decision trees. With the help of the hyper parameter, we can identify the number of decision trees that can be used for our problem domain.

4. FEATURE DISCRETION

As a pre-processing step, the feature discretion is done using random forest classifier to identify the most important feature that contributes towards the label, so that we can eliminate the least contributing feature thereby improving the computational efficiency.. Filter method and wrapper method are the two main different types of feature selection[28]. The importance of features is measured by their connection with the dependent variable or outcome variable, and features are chosen based on their results in various statistical tests. To rank all of the features in the data set, the filter approach employs an attribute evaluator and a ranker. Wrapper approaches use greedy search algorithms to examine all possible feature combinations and select the one that delivers the best result for a particular machine learning algorithm [29]. Sequential search algorithm and heuristic algorithm such as genetic algorithm comes under this category. If there are 'p' feature, then 2^p possible combinations of features are possible.

4.1. Feature Discretion using Passive Traffic Measurements

Initially features are selected using feature subset selection process. The practise of detecting and deleting as much useless and redundant information as feasible is known as feature subset selection. This decreases the data dimensionality, allowing learning algorithms to operate more quickly and effectively.

The Genetic Method, developed by Holland in 1965, is a sophisticated stochastic search algorithm based on natural genetics and selection mechanisms. Evolution is an optimizing process. Genetic Algorithm is an Evolutionary optimization technique based on the concept of "Survival of the fittest", Darwin Theory. It simulates the concept of evolution. This is a bio-inspired and uses the concept of genetics and natural selection. It comes under the evolutionary algorithm. Genetic algorithm(GA) iterates through fitness assessment, selection, recombination, and population reassembly. Initially, a set of random solutions called population is created. Each person in the population is referred to as a chromosome, and each chromosome represents a solution to the problem at hand. Generations are the iterations in which the chromosomes evolve. During each generation, the chromosomes are evaluated using some measure of fitness. It gets evolved using the principle of variation, selection and inheritance. Crossover and mutation are used for the generation of offspring. In crossover two parent's genetic information's are mixed to produce offspring. To keep the population size constant, a new generation is generated by selecting some of the parents and offspring based on fitness values and rejecting others. Fitter chromosomes have a better chance of being chosen. The algorithms eventually converge on the best chromosome, which should reflect the optimal or suboptimal solution to the problem after numerous generations. The classic methods for crossover are one point, two point and uniform crossover. Mutation operator maintains diversity among the population. One simple method of performing the mutation is bit flip mutation. Selection can be done using roulette selection, tournament selection etc. For wide classes of problems, GA works reasonably well.

Initial population is created randomly. Selection is done using Tournament selection. Crossover and mutation are the subsequent steps. The termination condition used here is to stop the iteration after a fixed number of generation. Since the iteration terminated after a fixed number of

generation we can expect a nearly optimal solution or better solution than previous generation. The genetic algorithm creates initial population randomly and here it provides the best reduced feature subset in the last generation. The importance of each feature is checked by simply deactivating each feature one at a time from the reduced feature set and accuracy is observed. Calculated accuracy change by ($au-au_i$) where 'au' denotes accuracy with all features and 'au_i' is the accuracy by deleting one feature. The minimum and maximum value of accuracy change is noted. Calculate the weight of each feature using the expression,

$$\frac{ac - ac_{min}}{ac_{max} - ac_{min}}$$

where ac - Accuracy change of each feature

ac_{min} - Minimum accuracy change

ac_{max} - Maximum accuracy change

Cross validation is used to analyze the individual attacks detection rate. If the detection rate is low, then that particular attack is further studied. If any of the important feature is not included, then it is added to the earlier reduced feature set and the above steps are repeated until satisfactory detection rate is achieved.

4.2. Proposed Methodology

The following are the steps in the suggested method as given in Figure 1. Initialize the value obtained from the random forest classifier as the first random population. Calculate each particle's fitness value in the population. Determine the best population by sorting from the above obtained set. Cross over and mutation is performed for avoiding worst set and preserving the good one. Repeat the process until we reach the optimal value. Identify the population which gave the best value. This is a process that repeats over and over we start with population and then create a new population until we reach to identify network attack.

The work flow of proposed method consists of three parts, feature discretion, classification and detection[30]. Feature discretion is done using an evolutionary algorithm called genetic algorithm which works on the principle of natural selection.

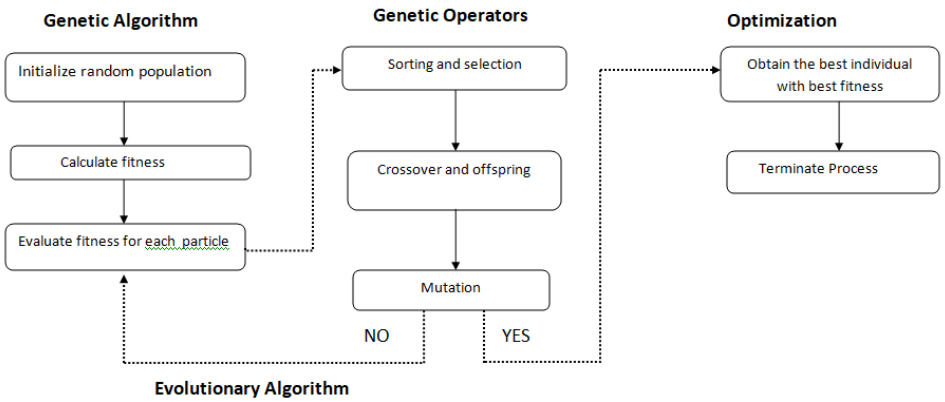

Figure 1: Proposed Method

4.3. Classification of Intrusions

Random forest, an ensemble based technique is used here for classifying the labeled dataset. In random forest the base learner is decision tree. Low bias and high variance are the two main properties of decision tree. Here the trees are going to split upto depth. By creating a decision tree in its entirety, we can ensure that it is correctly trained for our training data set. As a result, the training error will be minimal. In the case of test data, the number of instances is less compared to training samples[30]. By incorporating row sampling and feature selection, multiple decision tree models are combined in parallel so that the design will lead to high accuracy and low variance. Usually, the random forest works with respect to a scalar, it tries to find out the average of particular output from all decision trees. Binary classifier uses majority vote whereas regression problem uses mean or median of output of all decision trees.

Variant approaches are being utilized for intrusion detection but unfortunately none of the system existing so far is completely flawless. Secure data communication through the Internet, as well as any other network, is always vulnerable to hacking and misuse. As a result, it has become an essential component of computer and network security. It work from inside the network to catch attacks and breaches that make it through the firewall whereas firewall filter traffic on the network's periphery. It detects attackers and network anomalies and sends alerts through text, email to the management station.

5. EXPERIMENTAL SETUP

Various machine learning techniques are utilized to train and build multiple classification models that can classify attack type of network traffic verses normal type of network traffic. The test accuracies of various classification models are compared to identify the best model for performing network intrusion detection. Initially performance evaluators, essential and model required for analysis are imported followed by the path of the input data set. The dataset used is the twenty percent of labelled NSLKDD dataset. Initially import all the required libraries in Python such as Pandas, numpy, scipy, sklearn.seaborn, matlibplot with necessary models. Sklearn library is an open source machine learning tool widely used in python, with various tools for building statistical and machine learning models, including classification, clustering and dimensionality reduction. The dataset is then partitioned into train set and test set for further processing. Analyze the number of records and features. Using genetic algorithm eliminate features with low importance and check the effect on the accuracy of the model. Based on feature selection, the four levels are introduced. As all the features have some contribution to the model, we will keep all the features.The required model is fitted using train data. After we build the model using training data, we will test the accuracy of the model with test data and determine the appropriate model for this dataset. Finally, the response for test data is predicted by finding detection accuracy. The NSL KDD data set consists of 25191 rows in train data and 11850 rows in test data with 42 features. Train data set consists of twenty two labels with thirty eight numerical features and three categorical features. It has no missing values and duplicates. Training accuracy in Random Forest is 0.99875 with 0.8210 test accuracy.

6. ATTACK DESCRIPTION

Probing the network or system to gain information that helps in attack and thereby gaining a read access or write access to the system or disabling the services of the server are the main goals of an attacker. There are various tools available for the defenders of the network to study about various loopholes present in the real network environment that cause the cyber attacks. The research community has always come up with sophisticated defensive mechanisms that are

intelligent and adaptive to the ever evolving cyber world. Benchmark datasets are available for the researchers to evaluate these algorithms. One of the popular dataset for evaluation is the NSL-KDD dataset. There are twenty two attack types present in train set of NSL-KDD dataset and the test set includes thirty nine attack types as given in Table 1. Attacks belong to any one of the four classes, namely Probe, DoS, U2R or R2L.

Most serious attack category is the Denial of Service (DoS) such as unintended, distributed and application layer DoS. Most well known among this attack is the DNS flood. In order to slow or crash the service, an attacker simply floods it with requests from a faked IP address. The purpose is to reject new authorised TCP connections from any genuine client's side. If the attacker sets the source and destination information of a TCP segment as same, then it will give rise to Land attack, which is the Layer 4 DoS. Attempt to crash, destabilize, or freeze the targeted computer or service by sending malformed or oversized packets using a simple ping command will lead to Ping of Death (PoD). A smurf attack is a type of DDoS attack that employs the ICMP protocol to flood the victim's network with packets. TCP fragmentation attacks assaults target TCP or IP reassembly mechanisms, preventing from putting together fragmented data packets. As a result, the data packets overlap and quickly overwhelm the victim's servers, causing them to fail. It starts by sending the fragmented packets to a target machine. Such attacks are also termed as teardrop attacks.

When one source IP address transmits a predetermined number of ICMP packets to numerous hosts in a predetermined time period, it becomes an address sweep. Within a predetermined time interval, one source IP address transmits IP packets including TCP SYN segments to a predefined number of various ports at the same destination IP address. A backdoor is any method that allows authorised and unauthorised users to bypass typical security measures and achieve high-level user access or root access to a computer system, network, or software application. Password guessing is another sort of network attack. By detecting the user id or password combination of a genuine user, access rights to a computer and network resources are compromised.

Password guessing attacks can be categorized into two types such as Brute Force Attack and Dictionary attack respectively. A Brute Force assault is a form of password guessing attack that involves attempting every possible code, combination, or password until the correct one is discovered. This kind of attack could take a long time to finish. Another sort of password guessing attack is a dictionary attack, which employs a dictionary of frequent terms to figure out the user's password. A buffer overflow, also known as a buffer overrun, occurs when a fixed-length buffer is filled with more data than it can manage. This overflow normally causes a system crash, but it also gives an attacker the ability to run arbitrary code or exploit coding mistakes to cause harmful behaviour. Attackers leveraging password spraying technique are exploiting Internet Message Access Protocol (IMAP) to break into cloud accounts.

A rootkit is a form of malware that is meant to infect a target PC and install a suite of tools that give the attacker persistent remote access to the device. Powering down the machine and running a scan from a known clean system is one technique to discover the infestation. Another way of rootkit identification is behavioural analysis. Leveraging Internet Message Access Protocol (IMAP) for password-spray attacks to compromise cloud-based accounts lead to imap attack.

Table 1. Attacks present in NSL-KDD Dataset

Dataset	Attacks
NSL-KDD Train	Back, Land, Neptune, Pod. Smurf, Teadrop, Satan, IPsweep, Nmap,
	Portsweep, guesspasssword,ftp-write, imap, phf, multihop, warezmaster,
	warezclient, spy, buffer overflow, loadmodule, rootkit, perl
NSL-KDD Test	Back, Land, Neptune, Pod, Smurf, Teardrop, Satan, IPsweep, Nmap,
	Portsweep, guesspassword, ftp_write,imap, phf, multihop, warezmaster,
	warezclient, spy,buffer overflow, loadmodule,rootkit,perl, Apache2,
	Mailbomb,processtable,udpstorm,snmpgetattack,snmpguess,named,
	worm, sendmail,sqlattack,httptunnel,xterm,ps,xlock,xsnoop,mscan,saint

7. RESULTS AND INFERENCES

The evaluation started with the feature selection. Since the genetic algorithm is a stochastic process, feature selection was implemented many times and most repeated features were taken. Selected features are included in the Table 2.

Table 2. Reduced Feature Set

Reduced Feature Set	
Protocol type	Rerror rate
Dst bytes	Same service rate
Land	Diff service rate
Logged in	Srv diff host rate
Num compromised	Destination host count
Root shell	Dst host srv count
Num Root	Dst host same srv rate
Num access files	Dst host same src port rate
Num outbound cmds	Dst host srv diff host rate
Is guest login	Dst host srv serror rate
Srv count	Dst host rerror rate
Srv serror rate	

With these features, the classification was done. Analysis of individual detection rate has been done. From the thorough analysis, it is understood that due to feature selection, some features that contributed to the detection of attacks listed in Table 1 were missing for classification process. Table 6 shows the classification results using Random Forest for labelled dataset.

The individual attacks detection rates are analysed and it is found that the detection rate of the attack named ping of death (pod) is very less. Only 30% of the attack was correctly detected. The features unique to the attacks that have a low detection rate are introduced in various levels of development of the enhancement algorithm proposed and implemented in this work. Table 3 clearly indicates the probability of detection of individual attacks in the different stages of development such as Level I with 23 features, Level II with seven additional features of nmap, pod and teardrop. Level III with two additional features of warezclient and warezmaster. Level IV with all the 41 features. Table 4 and Table 5 indicates the precision and F-score of individual

attacks in each level. For the first level of detection beyond pod we considered two more attacks teardrop and nmap. From the analysis, it is understood that due to feature selection some attributes that contributed to the detection of the above mentioned intrusions were missing for classification process.

Table 3. Recall of 20 percent of dataset

Attacks	Level I	Level II	Level III	Level IV
Back	0.67	0.75	0.97	0.97
bufferoverflow	1	1	1	1
guesspassword	1	1	1	1
ipsweep	0.86	0.86	0.98	0.98
Neptune	1	1	1	1
nmap	0.84	0.92	0.92	0.92
Normal	0.98	0.99	0.99	0.99
pod	0.30	1	1	1
portsweep	0.97	0.97	0.97	0.97
satan	0.78	0.89	0.91	0.89
smurf	0.98	0.99	0.99	1
teardrop	0.74	1	1	1
warezclient	0.71	0.71	0.93	0.93
warezmaster	1	1	1	1

Teardrop is an attack that comes under the category of the denial of service which mainly uses the fragmented traffic to damage the victim machine. Fragmentation is a natural effect when traffic is moving over a network having a fluctuating size of the MTU (Maximum Transmission Unit). It can additionally take place when a host wishes to put a datagram on the network that exceeds its own networks maximum transmission network. At last from the destination host they were reassembled. Teardrop attack takes advantage of these fragments with the coinciding offset fields. While reassembling the fragments at the destination host some system may hang, crash or reboot. On understanding the different features of the dataset, it is found that the feature wrong fragment is related to fragmentation. So it is decided to add to the set of above 23 features to increase the rate of detection of attack teardrop.

Table 4. Precision of 20 percent of dataset

Attacks	Level I	Level II	Level III	Level IV
Back	0.75	0.73	0.97	0.95
bufferoverflow	0.33	0.33	1	1
guesspassword	1	1	1	1
ipsweep	0.93	0.95	0.95	0.95
Neptune	1	1	1	1
nmap	0.80	0.84	0.82	0.81
Normal	0.98	0.98	0.99	0.99
pod	0.40	0.95	0.95	0.95
portsweep	0.97	0.98	0.99	0.99
satan	0.98	0.92	0.94	1
smurf	1	1	1	1
teardrop	0.77	1	1	1
warezclient	0.62	0.74	0.70	0.70
warezmaster	0.67	0.67	1	1

Ping of death (Pod) is also a DoS attack and it is related to fragmentation. One of the features of TCP/IP is that single IP packet can be broken into smaller packets. When a packet is broken into small fragments it is possible to add up to a large amount than the allowed number of bytes. Attackers make use of this property to crash or freeze a system. This can be detected by identifying ICMP packets larger than 64KB. For this, we identified that the important features missing are service and wrong fragment.

The next attack evaluated was Nmap. It is a network scanner. In a computer network, It is used to distinguish between the host and the services. It can perform different types of scanning such as port scan including SYN, FIN and ACK scanning with TCP and UDP as well as ICMP scanning. So the port scan can be identified by examining the network packet through TCP, UDP or only FIN packets or only SYN packets which have been sent to many ports on the target machine or group of target machines on some duration of time. So the features duration, num shell, count, serror rate, srv rerror rate are added to the above reduced feature set.After the addition of seven more features to the reduced dataset, second level of detection was performed. The results indicates a good improvement in the detection rate of these attacks.

In the case of teardrop and pod, the detection rate increased to 1 from 0.74 and 0.30. For the nmap, the detection rate increased from 0.84 to 0.98. The attacks warezclient and warezmaster were considered for the tird level detection. These two attacks come under the category of (R2L) root to local attack.

The warezmaster utilizes the system bug which is related to the ftp server. This attack occurs when write permission is given b y mistake to the user on the system by the ftp server. Then the attacker can login and upload any files. The attacker login to the system using a guest account while the attack is occuring. Then hidden directory is created and illegal copies of the software are uploaded. This attack can be identified when many data were sent from source to destination during ftp session.

Table 5. F-Score of 20 percent of dataset

Attacks	Level I	Level II	Level III	Level IV
Back	0.71	0.74	0.97	0.96
bufferoverflow	0.50	0.50	1	1
guesspassword	1	1	1	1
ipsweep	0.89	0.90	0.97	0.96
Neptune	1	1	1	1
nmap	0.82	0.88	0.87	0.86
Normal	0.98	0.98	0.99	0.99
pod	0.40	0.95	0.97	0.95
portsweep	0.98	0.98	0.98	0.98
satan	0.87	0.91	0.92	0.94
smurf	0.99	1	1	1
teardrop	0.75	1	1	1
warezclient	0.67	0.73	0.80	0.80
warezmaster	0.80	0.80	1	1

The warezclient attack occurs after the warezmaster attack. In this, the warez is downloaded which is actually loaded during warezmaster attack. Downloading files from an FTP server always seems to be a legal process. This can be identified during an FTP session when hot indicators were notably triggered for a small duration of time. This may be due to downloading warez. So hot is an important feature. For the third level of detection two additional features src

bytes and hot were incorporated. At this stage, detection is performed using 32 features and the detection rate of the warezclient increases from 0.71 to 0.93.

The fourth level of detection was done using all the 41 feature and its accuracy and detection rate was observed. The overall accuracy improved than the above levels. The results of recall precision and F-score for four different levels are shown in Table 6.

For the experimentation, only twenty percent of the dataset was used. Since some of the attacks which has only less representation or less samples are unsampled. Hence the tables are provided with fourteen types of attacks. Tables 7 to 9 provides the recall, precision and F-Measure of the unsampled data set having all the twenty two types of attacks present in the entire dataset.

Table 10 and Table 11 shows the overall accuracy obtained during each stage of the experiment. The results of classification accuracy reveals the fact that feature discretion as well as providing weight to the reduced feature set improves accuracy. This work shows the importance of detecting all types of attacks present in the data set without compromising accuracy. Because some of the major attacks that contribute most were easily detected without detecting the minority attacks, which are very hard to detect. This issue is being addressed in this using a level wise detection. Out of 41 features, 32 features were proved to be very important. Thus by using this level by level reduced feature set, we were able to identify missing features along with few attacks such as Pod, Nmap, teardrop, warezclient and warezmaster.

Table 6. Precision, Recall and F-Measure of twenty percent of dataset

Attacks	Recall	Precision	F-Measure
Neptune	1	1	1
warezclient	0.939	0.971	0.955
ipsweep	0.989	0.994	0.998
portsweep	0.986	0.997	0.991
teardrop	1	1	1
Nmap	0.983	0.983	0.983
Satan	0.973	0.994	0.983
Smurf	1	1	1
Pod	1	1	1
Back	1	1	1
guesspassword	1	1	1
bufferoverflow	0.667	0.800	0.727
Imap	0.600	1	0.750
Warezmaster	0.714	0.833	0.769
spy	0	0	0

Table 7. Recall of twentytwo attacks in the dataset

Attacks	Level I	Level II	Level III	Level IV
Back	0.84	0.92	0.98	0.98
Bufferoverflow	1	1	1	1
ftp write	0.67	0.67	0.67	0.67
guesspassword	1	1	1	1
Imap	1	1	1	1
ipsweep	0.90	0.90	0.98	0.99
Land	1	1	1	1
Loadmodule	0.74	0.89	1	1
Multihop	1	1	1	1
Neptune	1	1	1	1
Nmap	0.75	0.93	0.91	0.88
Normal	0.98	0.98	0.98	0.99
Phf	1	1	1	1
Pod	0.29	1	1	1
portsweep	0.97	0.98	0.97	0.98
Rootkit	0.23	0.23	0.77	1
Satan	0.82	0.93	0.93	0.93
Smurf	0.99	0.99	1	1
spy	1	1	1	1
teardrop	0.68	1	1	1
warezclient	0.66	0.66	0.91	0.89
warezmaster	0.86	0.86	0.86	0.86

Table 8. Precision of twentytwo attacks in the dataset

Attacks	Level I	Level II	Level III	Level IV
Back	0.77	0.75	0.91	0.96
Bufferoverflow	0.85	0.92	0.91	0.92
ftp write	0.67	0.86	0.75	0.88
guesspassword	1	1	0.99	1
Imap	1	0.97	0.85	0.97
ipsweep	0.94	0.94	0.95	0.93
Land	1	1	1	1
Loadmodule	0.89	0.93	1	0.98
Multihop	0.49	0.52	0.58	0.58
Neptune	1	1	1	1
Nmap	0.94	1	0.98	0.97
Normal	0.95	0.97	0.99	0.99
Phf	1	1	1	1
Pod	0.22	1	0.88	0.92
portsweep	0.98	0.98	1	0.98
Rootkit	1	1	0.88	0.92
Satan	0.97	0.91	0.90	0.98
Smurf	1	0.99	0.99	1
spy	1	1	1	1
teardrop	0.93	1	1	1
warezclient	0.66	0.70	0.65	0.76
warezmaster	1	1	1	1

Table 9. F-Score of twentytwo attacks in the dataset

Attacks	Level I	Level II	Level III	Level IV
Back	0.80	0.82	0.94	0.97
Bufferoverflow	0.92	0.96	0.95	0.96
ftp write	0.80	0.75	0.71	0.82
guesspassword	1	1	1	1
Imap	1	0.98	0.92	0.98
ipsweep	0.92	0.92	0.96	0.96
Land	1	1	1	1
Loadmodule	0.81	0.91	1	0.99
Multihop	0.66	0.69	0.73	0.73
Neptune	1	1	1	1
Nmap	0.83	1	0.95	0.92
Normal	0.97	0.98	0.99	0.99
Phf	1	1	1	1
Pod	0.25	1	0.93	0.93
portsweep	0.98	0.98	0.98	0.98
Rootkit	0.37	0.37	0.82	0.96
Satan	0.89	0.92	0.91	0.96
Smurf	1	0.99	1	1
spy	1	1	1	1
teardrop	0.79	1	1	1
warezclient	0.66	0.68	0.76	0.82
warezmaster	0.92	0.92	0.92	0.92

Table 10. Overall Accuracy for weighted and non-weighted features

Number of Features	Accuracy in Percentage
41 Features	81.04
23 Features	90.87
Weighted 23 Features	97.12
Weighted 30 Features	98.01
Weighted 32 Features	98.74
Weighted 41 Features	98.86

Table 11. Overall Accuracy for Random Forest.

Classifier	Number of Features	Accuracy	Time Taken (Seconds)
Random Forest	23 Features	97	3.48
	23+7 (additional features of nmap, teardrop, pod)	99.36	3.14
	23+7 (additional features of warezclient and warezmaster)	99.65	3.86

8. CONCLUSION

This manuscript aims at finding the optimum search point in detecting intrusion among network. This was accomplished using feature discretion using the best ensemble based machine learning classifier known as gradient based random forest by importing the data set by splitting into test and train. The sequential workflow includes feature selection, classification and searching

technique. Stochastic based evolutionary algorithm is applied for getting the optimized result. As future direction, simulating this data set using various hybrid methods will help the research community to explore more challenges in the field of threat analysis and network security.

REFERENCES

[1] Stefan Axelsson. Intrusion detection systems: A survey and taxonomy. Technical report, Technical report, 2000.

[2] Ciza Thomas and N Balakrishnan. Performance enhancement of intrusion detection systems using advances in sensor fusion. Supercomputer Education and Research Centre Indian Institute of Science, Doctoral Thesis, 304pp. Available at: http://www. serc. iisc. ernet.in/graduation-theses/CizaThomas-PhD-Thesis. pdf, 2009.

[3] Aleksandar Lazarevic, Vipin Kumar, and Jaideep Srivastava. Intrusion detection: A survey. In Managing Cyber Threats, pages 19–78. Springer, 2005.

[4] Chirag Modi, Dhiren Patel, Bhavesh Borisaniya, Hiren Patel, Avi Patel, and Muttukrishnan Rajarajan. A survey of intrusion detection techniques in cloud. Journal of Network and Computer Applications, 36(1):42–57, 2013.

[5] Swati Paliwal and Ravindra Gupta. Denial-of-service, probing & remote to user (r2l) attack detection using genetic algorithm., International Journal of Computer Applications 60(19):57–62, 2012.

[6] Ron Kohavi and George H John. Wrappers for feature subset selection. Artificial intelligence,97(1-2):273–324, 1997.

[7] Jason Weston, Sayan Mukherjee, Olivier Chapelle, Massimiliano Pontil, Tomaso Poggio, and Vladimir Vapnik. Feature selection for svms. In Advances in neural information processing systems, pages 668–674, 2001.

[8] Alessia Mammone, Marco Turchi, and Nello Cristianini. Support vector machines. Wiley Interdisciplinary Reviews: Computational Statistics, 1(3):283–289, 2009.

[9] Marti A. Hearst, Susan T Dumais, Edgar Osuna, John Platt, and Bernhard Scholkopf. Support vector machines. IEEE Intelligent Systems and their applications, 13(4):18–28, 1998.

[10] Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine learning*, 20(3):273–297, 1995.

[11] Srinivas Mukkamala and Andrew Sung. Feature selection for intrusion detection with neural networks and support vector machines. *Transportation Research Record: Journal of the Transportation Research Board*, (1822):33–39, 2003.

[12] Muhammad Shakil Pervez and Dewan Md Farid. Feature selection and intrusion classification in nsl-kdd cup 99 dataset employing svms. In *Software, Knowledge, Information Management and Applications (SKIMA), 2014 8th International Conference on*, pages 1–6. IEEE, 2014.

[13] Yinhui Li, Jingbo Xia, Silan Zhang, Jiakai Yan, Xiaochuan Ai, and Kuobin Dai. An efficient intrusion detection system based on support vector machines and gradually feature removal method. *ExpertSystems with Applications*, 39(1):424–430, 2012.

[14] Nelcileno Araújo, Ruy de Oliveira, Ailton Akira Shinoda, Bharat Bhargava, et al. Identifying important characteristics in the kdd99 intrusion detection dataset by feature selection using a hybrid approach. In *Telecommunications (ICT), 2010 IEEE 17th International Conference on*, pages 552–558. IEEE, 2010.

[15] Rung-Ching Chen, Kai-Fan Cheng, Ying-Hao Chen, and Chia-Fen Hsieh. Using rough set and support vector machine for network intrusion detection system. In *Intelligent Information and Database Systems, 2009. ACIIDS 2009. First Asian Conference on*, pages 465–470. IEEE, 2009.

[16] Wei Wang and Roberto Battiti. Identifying intrusions in computer networks with principal component analysis. In *Availability, Reliability and Security, 2006. ARES 2006. The First International Conference on*, pages 8–pp. IEEE, 2006.

[17] Fangjun Kuang, Weihong Xu, and Siyang Zhang. A novel hybrid kpca and svm with ga model for intrusion detection. *Applied Soft Computing*, 18:178–184,2014.

[18] Cheng-Lung Huang and Chieh-Jen Wang. A ga-based feature selection and parameters optimizationfor support vector machines. *Expert Systems with applications*, 31(2):231–240, 2006.

[19] Ikram Sumaiya Thaseen and Cherukuri Aswani Kumar. Intrusion detection model using fusionof chi-square feature selection and multi class svm. *Journal of King Saud University-*

Computer and Information Sciences, 29(4):462–472, 2017.

[20] Adriana-Cristina Enache and Victor Valeriu Patriciu. Intrusions detection based on support vector machine optimized with swarm intelligence. In *Applied Computational Intelligence and Informatics (SACI), 2014 IEEE 9th International Symposium on*, pages 153–158. IEEE, 2014.

[21] Mostafa A Salama, Heba F Eid, Rabie A Ramadan, Ashraf Darwish, and Aboul Ella Hassanien. Hybrid intelligent intrusion detection scheme. In Soft computing in industrial applications, pages 293–303. Springer, 2011.

[22] Stephanie Forrest. Genetic algorithms: principles of natural selection applied to computation. Science, 261(5123):872–878, 1993.

[23] Randy L Haupt, Sue Ellen Haupt, and Sue Ellen Haupt. Practical genetic algorithms, volume 2. Wiley New York, 1998.

[24] Sean Luke. Essentials of metaheuristics, volume 113. Lulu Raleigh, 2009.

[25] Bineet Mishra and Rakesh Kumar Patnaik. Genetic Algorithm and its varients: Theory ans Applications, PhD thesis, 2009.

[26] L Dhanabal and SP Shantharajah. A study on nsl-kdd dataset for intrusion detection system based on classification algorithms. International Journal of Advanced Research in Computer and Communication Engineering, 4(6):446–452, 2015.

[27] Charles E Metz. Basic principles of roc analysis. In Seminars in nuclear medicine, volume 8, pages 283–298. Elsevier, 1978.

[28] Kristopher Robert Kendall. A database of computer attacks for the evaluation of intrusion detection systems. PhD thesis, Massachusetts Institute of Technology, 1999.

[29] Stephen Northcutt and Judy Novak. Network intrusion detection. Sams Publishing, 2002.

[30] Ciza Thomas, Vishwas Sharma, and N Balakrishnan. Usefulness of darpa dataset for intrusion detection system evaluation. In Data Mining, Intrusion Detection, Information Assurance, and Data Networks Security 2008, volume 6973, page 69730G. International Society for Optics and Photonics,2008.