03-134142-041 MUHAMMAD HAMZA AZIZ

03-134142-027 HAMZA AHMED

Anomaly Based Intrusion Detection System

In partial fulfilment of the requirements for the degree of

Bachelor of Science in Computer Science

Supervisor: Asghar Ali Shah

Department of Computer Sciences

Bahria University, Lahore Campus

June 2018

C e r t i f i c a t e

We accept the work contained in the report titled
"ANOMALY BASED INTRUSION DETECTION SYSTEM",
written by
MUHAMMAD HAMZA AZIZ
HAMZA AHMED

as a confirmation to the required standard for the partial fulfilment of the degree of Bachelor of Science in Computer Science.

Approved by:

Supervisor: Asghar Ali Shah (Assistant professor)

_____ (Signature)

June 4*th*, 2018

DECLARATION

We hereby declare that this project report is based on our original work except for citations and quotations which have been duly acknowledged. We also declare that it has not been previously and concurrently submitted for any other degree or award at Bahria University or other institutions.

Enrolment	Name	Signature
03-134142-041	MUHAMMAD HAMZA AZIZ	
03-134142-027	HAMZA AHMED	

Date : _____

Specially dedicated to

My beloved grandmother, mother and father

MUHAMMAD HAMZA AZIZ

My beloved grandmother, mother and father

HAMZA AHMED

ACKNOWLEDGEMENTS

We would like to thank everyone who had contributed to the successful completion of this project. We would like to express my gratitude to my research supervisor, Mr. Asghar Ali Shah for his invaluable advice, guidance and his enormous patience throughout the development of the research.

In addition, we would also like to express my gratitude to our loving parent and friends who had helped and given me encouragement.

Hamza Aziz
Hamza Ahmed

ANOMALY BASED INTRUSION DETECTION SYSTEM

ABSTRACT

As the research increased in computer science highlight the scientists mind for the growing research world towards security. Researchers have done a lot of research work in network Security. Cybersecurity has progressively become a zone of alarm for officials, Government agencies and industries, including big commercialized infrastructure, are under attack daily. First signature-based intrusion detection systems were developed, and it detects only known attacks. To detect strange attacks statistical IDS came into being recognized as anomaly-based IDS. It is not as much efficient as it detects all. This, study focuses on the efficiency of IDS using UNSW NB-15 dataset and suitable techniques to identify attacks.

Keywords: - ***Anomaly Based intrusion detection system, UNSW NB-15, IDS, Anomaly Based IDS***

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS / ABBREVIATIONS

OHE	*One hot Encoding*
RFT	*Random Forest Tree*
RFE	*Recursive feature elimination*
CV	*Cross Validation*
KDD	*Knowledge driven from data*
CNN	*Convolutional neural network*
ACCS	*Australia Centre of Cyber Security*
IG	*Information Gain*
RBI	*Rule Based Induction*
TCP	*Transmission control Protocol*
UDP	*User Datagram Protocol*
NN	*Neural network*
NB	*Naïve-Bayes*
EM	*Expectation Maximisation Clustering*
LR	*Logistic regression*

LIST OF APPENDICES

CHAPTER 2

INTRODUCTION

Networked computer playing chief role in information sharing. It keeping our society more modernized by providing ease to people in activities like efficient execution in transactions, running business processes, Government agencies secret services and social media etc. all have users in millions or billions. There is nothing aspect of our life where computer is not involved. The hardware or software that constitute these systems are rapidly changing.

The Internet, which is interconnection of millions of devices, designed for information sharing. It was designed to share information from simple binary numbers to complex real numbers. The internet users are increasing drastically day by day. Some statistics are discussed in Figure 1-1 [19]. These statistics are collected from internet world stats. As the number of internet users rise result in the upgradation of internet infrastructure as well.

WORLD INTERNET USAGE AND POPULATION STATISTICS DEC 31, 2017 - Update						
World Regions	**Population (2018 Est.)**	**Population % of World**	**Internet Users 31 Dec 2017**	**Penetration Rate (% Pop.)**	**Growth 2000-2018**	**Internet Users %**
Africa	1,287,914,329	16.9 %	453,329,534	35.2 %	9,941 %	10.9 %
Asia	4,207,588,157	55.1 %	2,023,630,194	48.1 %	1,670 %	48.7 %
Europe	827,650,849	10.8 %	704,833,752	85.2 %	570 %	17.0 %

Latin America / Caribbean	652,047,996	8.5 %	437,001,277	67.0 %	2,318 %	10.5 %
Middle East	254,438,981	3.3 %	164,037,259	64.5 %	4,893 %	3.9 %
North America	363,844,662	4.8 %	345,660,847	95.0 %	219 %	8.3 %
Oceania / Australia	41,273,454	0.6 %	28,439,277	68.9 %	273 %	0.7 %
WORLD TOTAL	7,634,758,428	100.0 %	4,156,932,140	54.4 %	1,052 %	100.0 %

Table 2-1 Internet usage statistics collect from Internet World Stats [19].

2.1 Background

Anomalies is nastiness in data that cause the operations deviate its working from normal behaviour [22]. Anomalies caused due to some reasons like, misconfiguration overload in network, malicious activities, devices malfunctioning etc. Anomalies are broadly divided into two categories

1. Network performance related anomalies.
2. Security related anomalies.

Performance related anomalies may occur due to network vulnerabilities. Vulnerabilities are the weakness in management of network devices, design and implementation. Poor design can be a flaw in software or hardware Like, in early versions of UNIX there is a “sendmail” flaw which enable the hackers to get access. Poor implementation refer to incorrect installation or configuration. For example, a system is configured with out-restricted access privileges on critical executions. It easily enable the hackers to tamper with these files. If no security routine check done and someone gain full access, this will poor management.

Security related anomalies Security anomalies in network occur due to several reasons e.g. flood anomalies, network operation anomalies and flash crowd anomalies. The malicious activities occur in network are categorize as, point anomaly, contextual anomaly, collective anomaly.

An IDS [1] is employed to differentiate every kind of nasty link traffic and device use that cannot be identified traditional firewall. This embrace network attacks in contradiction of susceptible facilities, data driven attacks on products, host-based attacks such as permission increase, unlicensed logins and entrée to sensitive data, and nasty files (i.e. viruses, Trojans, and worms).

IDS classification is done in three ways based on its deployment.

Network Intrusion Detection System acknowledge intrusions by investigating network traffics data and inspecting hosts data. Network IDS get permission to networks traffic by joining to a hub, network switch organized for port mirroring, or network faucet. An exemplar of a Network IDS is Snort.

Host-based IDS involves the working of an agent on a host machine which distinguishes intrusions by inspecting system calls, application log files, file-system variations (binaries, password files, capability/Access Control List databases) and other host activities and situations.

Hybrid IDS syndicates one or more methods. Host data is united with network info to create a comprehensive view of the system. A model of a HIDS is Prelude.

Based on recognition method classification is done in two ways:

Signature Based Detection method is specifically used for known patterns/novel attacks to detect nasty code. These specific patterns are termed as signatures. Detecting the worms in the network is an example of signature-based detection. These intrusions are said to be as misuse.

Anomaly Based Detection methods are designed to distinguish abnormal behaviour in the system to normal behaviour. The ordinary usage is base lined, and signals or messages are created when someone diverges from the standard behaviour.

Sony Pictures Entertainment [2] experienced one of the most distressing commercial attacks in history in the history of mankind. Thousands of records, grabbed by hackers

and were revealed online with personal details of around 6,000 Sony employees, forthcoming Sony feature films and the pay details of top management. The hackers also achieved to retrieve details about Deloitte financiers who are Sony's auditors.
The foremost data breach, which happened on 24th November, caused in the halt of the whole computer network of one of Hollywood's prime and most authoritative studios. Here have been collective reports that the hacking was carried out by North Korea in payback for the future release of a Sony comedy movie called "The Interview". The storyline tracks Seth Rogan and James Franco who are working in the CIA (Central Intelligence Agency) to eliminate Kim Jong-un the dictator of North Korea.

On July 13, 2017 a world known Organization Verizon was misled. A report express that 14 million supporters may have been influenced by this information rupture. This incorporate any individual who reached to client benefit in the previous a half year. These records were hung on a server that was controlled by Israel based Nice Systems. The information break was found by Chris Vickery, who is with the security firm, "UpGuard". He educated Verizon of the information presentation in late-June, and it took over seven days to secure the broke information. The genuine information that was acquired were log documents that progressed toward becoming created when clients of Verizon reached the organization by means of telephone [20].

Industrial think tanks guesstimate that almost 60,000 new, nasty computer programs and 315,000 new, nasty files are discovered daily. From 2006 to 2012, the number of security happenings stated by federal agencies amplified from 5,503 to 48,562 – a rise of 78.2% – and in 2013 McAfee investigation estimated that worldwide cybercrime failures might total $400 billion. Cyber-attacks are a risk to America's nationwide and financial security, in addition to separate privacy, and to the fundamental and most important factor, corporate strategies, and knowledgeable property for tom, dick and harry [3].

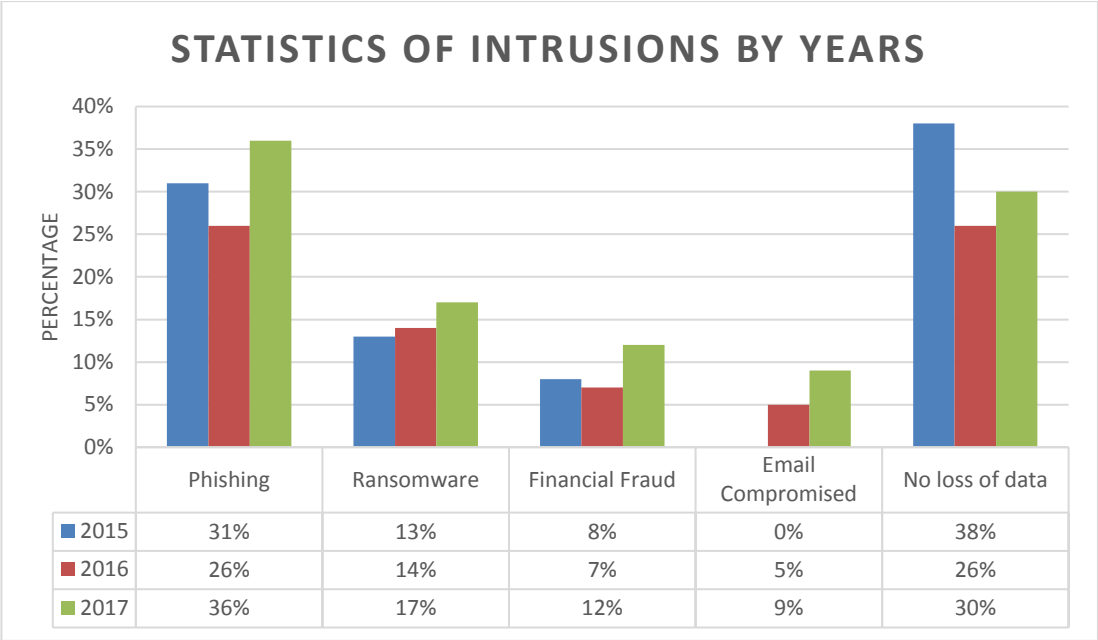


Figure 2-1 statistics of Intrusions from 2015-2017

As by Figure 1-2 there a decline in the overall number of incidents, phishing and ransomware attacks are on the rise, as is the number of companies that experienced losses from a cyberattack. As by cyberattack reports, the types of cybercrime on the rise. 36% of respondents say they were impacted by a phishing attack, up from 26 % the previous year. Ransomware attacks also rose, from 14% to 17%. Financial fraud jumped to 12% from 7%. Organizations are investing more and more in cyber security [21]. Intrusion discovery time is increasing each year because there is evolution of more sophisticated techniques in hacking. Detailed is in Figure 1-3.

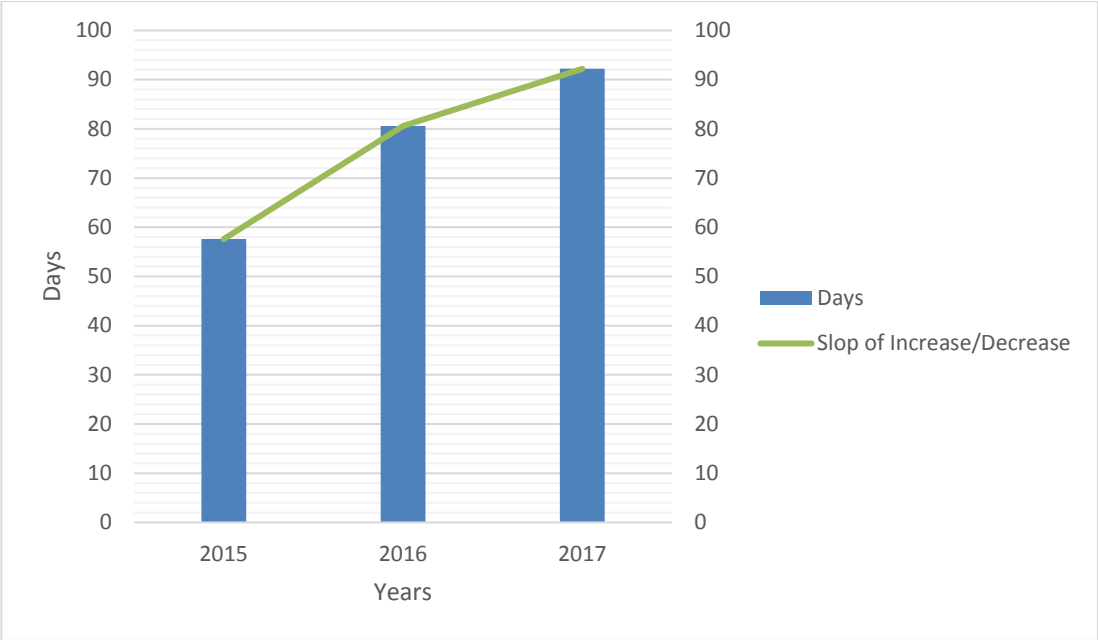


Figure 2-2 Average time of intrusion discovery is growing from 2015-2017

As The dawn of cloud computing, though, has taken new applicability to IDS structures, resulting in a flow in the IDS marketplace. A vital element of today's security top preparations, Intrusion Detection Systems are created to sense attacks that can happen, regardless of preventive procedures. In fact, Intrusion Detection System is today's unique top selling security equipment and it is predicted to remain to increase impetus. Despite everything, cloud security is far too multifaceted to be checked physically.

The logic and waya Intrusion Detection System usages is much related to these days technology. Through cloud computing, Intrusion Detection System has creäted a world where it can flourish and be most operative. By means of cloud computing, the fundament has engrossed with the Intrusion Detection technology.

So, computer security is very complex and always involves the human element.

2.2 Problem Statements

A number of IDS researchers as have utilised these datasets due to their public availability. However, many researchers have reported majorly three important disadvantages of these datasets [41] [42] [43] [44] [8] which can affect the transparency of the IDS evaluation. First, every attack data packets have a time to live value (TTL) of 126 or 253, whereas the packets of the traffic mostly have a TTL of 127 or 254. However, TTL values 126 and 253 do not occur in the training records of the attack [42]. Second, the probability distribution of the testing set is different from the probability distribution of the training set, because of adding new attack records in the testing set [43][8]. This leads to skew or bias classification methods to be toward some records rather than the balancing between the types of attack and normal observations. Third, the data set is not a comprehensive representation of recently reported low foot print attack projections [44].

Researchers have worked hard to detect intrusions but still there is a problem of efficiency. They have achieved up to 88% efficiency of anomaly detection. This study deals with efficiency of anomaly-based intrusion detection system. It improve the accuracy.

2.3 Aims and Objectives

The objectives of the thesis are shown as following:

i) The main objective of this research is to improve the detection accuracy of anomaly intrusion from the previously achieved 88.56% accuracy.

ii) Better pre-processing and classification of data to achieve high efficiency

2.4 Scope of Project

This project Research leads to the formation of an efficient Anomaly based IDS. The future IDS will merge all of the independent network components and tools which

exist today, into a complete and cooperative system, dedicated to keeping networks stable. There will be many distributed elements i.e. in an organization networks devices, specified network routes, IPS (Intrusion Prevention System), IDS (Intrusion Detection System), firewall etc. every element maintain its logs with specific information. Every element is performing its specific job, each passing the results onto a higher level for correlation and analysis. For example, in banking sector a bank track down its activities but regional office track down its underlying braches and so on

CHAPTER 3

LITERATURE REVIEW

Computer System suffer security vulnerabilities that are technical difficult and end economically costly. UNSW NB-15 and NSL-KDD are datasets used for evaluation of IDS's. KDD99 usage is increasing unexpectedly. Statistics by some resources are:

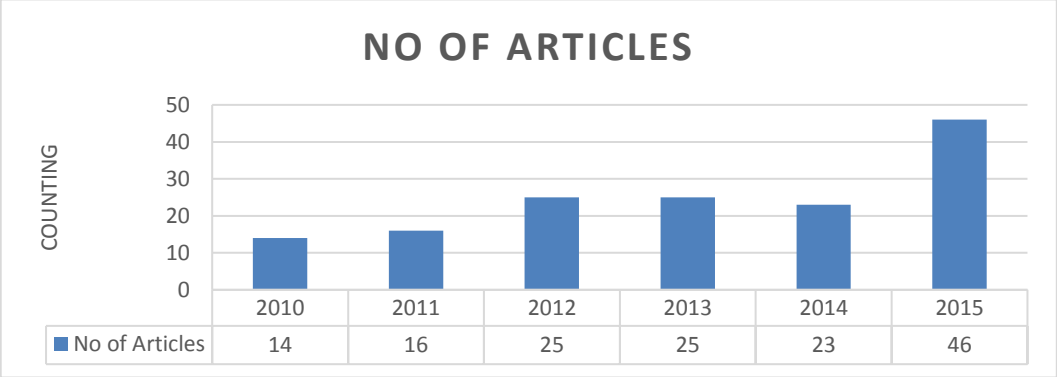


Figure 3-1 No of articles that use KDD99 Dataset [5].

NSL-KDD dataset is openly accessible for students and scientists. Although, the info set still suffers from many glitches mentioned in paper by McHugh [7] and Its won't be a perfect demonstration of existing actual networks, due to shortage of knowledge in dataset available publicly for network-based IDSs, it tends to have faith in it because it still often practical as an efficient benchmark knowledge driven to assist data scientist to compete completely different intrusion detection strategies.

There is a measure of some issues within the KDD dataset that causes the analysis results on this knowledge set to be dishonest. Because it contain redundant record. Data is over fitted when modelled.

Dataset social control is important to boost the efficiency of IDS once datasets square measure is big. Hence, technique used is Min-Max technique of social control.

3.1 Information gain method

Features will be selected based on information gain. It is calculated as

Let "D" be a group of training set with their match up labels [9]. Imagine there are "m" categories and the training set has "Di" category "I" and "D" are that the total variety of samples within the preparation set. Predictable data required to classify a sample, it is computed as:

A "D" named feature will split the training set into "v" subsets wherever "Dj" is that the set that has the worth "Aj" for feature "A" [9]. Moreover, let "Dj" contain "Dij" samples for category "i". Entropy of the feature "D" is calculated as:

$$E(D) = \sum_{j=1}^{v} \frac{D_{1j} + \cdots + D_{mj}}{D} * I\left(D_{1_j}, \ldots, D_{m_j}\right) \quad \ldots\ldots\ldots \text{ Equation 3.1}$$

Information gain for "A" is calculated "D" is treated as S is data:

$$I(s_1, s_2, s_3, \ldots, s_m) = -\sum_{i=1}^{m} \frac{s_i}{s} \log_2\left(\frac{s_i}{s}\right) \quad \ldots\ldots\ldots \text{ Equation 3.2}$$

The dependency magnitude relation is solely calculated therefore [10]

$$Dependency\ Ratio = \frac{HV}{TI} - \frac{OT}{TO} \quad \ldots\ldots\ldots \text{ Equation 3.3}$$

Where

H V = highest variety of occurrence variation for a category label in attribute A.

T I = total variety of occurrences of that category within the dataset.

O T = variety of occurrences for different category labels supported or a group of Variations.
T O = total variety of instances of category/class labels within the dataset creating OT. It helps to pick out options by high worth to low worth and so they're evaluated. On KDD99 test set there is a classification rate of 86% to nearly 100% [11].

Information gain (IG) measures how much. "Information" a feature gives us about the class, help to collect information for feature selection to reduce dimensionality so less computation power required.

3.2 Rule based Induction

Rule induction is one in all the chief varieties of data processing and is probably the foremost common variety of information discovery in unsupervised and supervised learning systems [12]. Rule induction is a vast responsibility wherever all doable patterns are completely force out of the information and so Associate in Nursing correctness and worth are accessorial to tell the user that how powerful the pattern is? And the probability it can happen another time.

Yu-Xin Ding et al., Min Xiao et al. and Ai-Wu Liu et al.[13] proposed a snort-based hybrid intrusion detection system using frequent episode rules and the 10% of the KDD99 Cup dataset. They create an anomaly detection module for Snort that can detect the unknown attacks and a signature generation module that extracts the signature of attacks that are detected by ADS module, and maps the signatures into snort rules. They achieve an average of detection rate of 94.07%.

For the how much the rule to be helpful there must be two things that provide a great information [14].

- Accuracy – however typically is that the rule corrects?
- Coverage – however typically will the rule apply?

3.3 Advance approaches

Machine learning scientist apply every possible method to increase the accuracy of IDS. M. Belouch [15] done a detailed analysis on UNSW NB-15 and NSL-KDD and result are:

Dataset		TCP		UDP		Other		Total
		Normal	Attack	Normal	Attack	Normal	Attack	
NSL-KDD	Training	53600	49089	12434	2559	1309	6982	125973
	Testing	7842	11038	1776	845	93	950	22544
UNSW NB-15	Training	39121	40825	13922	49361	2957	29155	175341
	Testing	27848	15247	8097	21321	1055	8764	82332

Table 3-1 Normal / Attack NSL-KDD and UNSW-NB15

According to table 2-1 there is a clear difference between entries and UNSW NB-15 has more categories then KDD99 or NSL-KDD. So, prior focus is UNSW NB-15.

Classifier	Accuracy	Train	Test
NB Tree + Random Tree	89.24	50.29	0.93
REP Tree	89.85	1.17	0.24

Table 3-2 Performance comparison on NSL-KDD [15]

Classifier	Accuracy	Train	Test
Decision Tree	85.56	7.66	0.84
REP Tree	88.95	2.69	0.37

Table 3-3 Performance comparison on UNSW-NB15 [15]

In Table 2.2 and table 2.3 there is a comparison between NSL-KDD and UNSW NB-15. NSL-KDD got more accuracy then UNSW NB-15 because attack

category list is almost half then UNSW NB-15. Diversity of attack increase in UNSW NB-15.

In another research, Results using Expectation-Maximisation Clustering (EM), Logistic regression (LR) and Naïve Bayes (NB) are:

	UNSW NB-15	
	Accuracy	False Alarm Rate
EM	77.2	13.1
LR	83.0	14.2
NB	79.5	23.5

Table 3-4 Comparison of Accuracy for UNSW NB-15[16].

	UNSW NB-15				
Approaches	EM[16]	LR[16]	NB[16]	DT[15]	REP Tree[15]
Accuracy %	77.2	83.0	79.5	85.56	88.95

Table 3-5 Comparison of Accuracy between UNSW NB-15

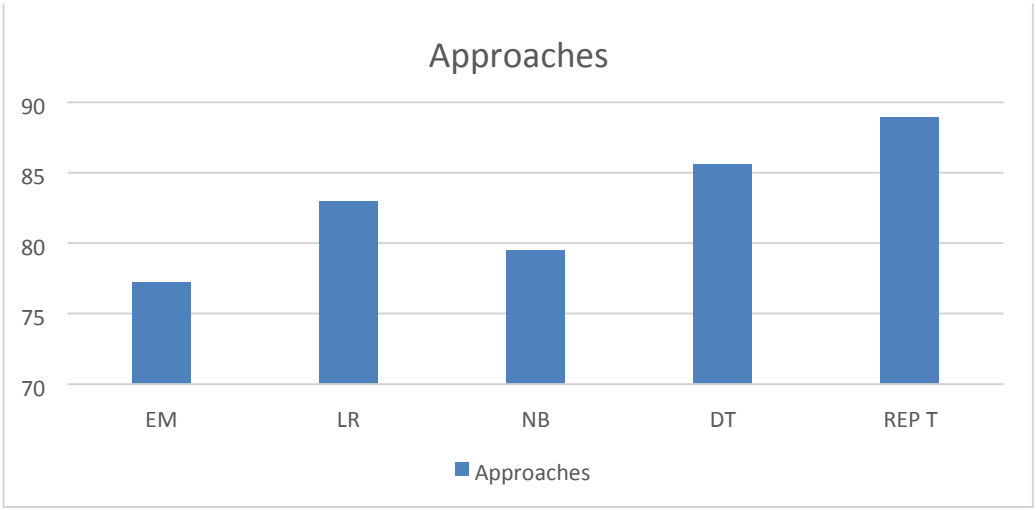


Figure 3-2 comparison of different approaches for UNSW NB-15

The maximum accuracy achieved is 88.95%. It is achieved by REPT classification.

CHAPTER 4

METHODOLOGY

Anomaly based refer to the statistical measure of system features. For this UNSW NB-15 dataset is used.

Anomaly based detection involves following steps

4.1 Pre-processing

It an important step in data mining process. It converts the raw into understandable format. There is a required of understandable format of training dataset for the learning of IDS. The main step in pre-processing is

1. Data cleansing
2. Data editing
3. Data reduction
4. Data wrangling

4.2 Feature Selection

In machine learning it is a procedure of choosing a subset of pertinent features/attributes used to create model. For specific results we need relevant features. Feature selection methods are adopted for following motives [17]:

- Oversimplification of model so it become easy to understand.
- Shorter training time.
- To avoid curse of dimensionality.
- Enhance generalization by reducing overfitting.

4.2.1 Feature Extraction

There is another approach called Feature Extraction related to Feature Selection. The goal of both approaches is to reduce the number of dimensions in a dataset. There are at least two important differences between feature selection and feature extraction.

1. A element choice technique decreases the dimensionality of an element space by choosing a subset of unique highlights, while a component extraction strategy, diminishes the dimensionality of an element space by straight or nonlinear projection of the n-dimensional vector onto a k-dimensional vector (k<n).
2. A feature selection method chooses features from the original n-dimensional set based on a measure such as information gain, correlation or mutual information and a user-defined threshold to filter out unimportant or redundant features.

For instance, in implanted or wrapper techniques, particular classifiers are utilized as a part of relationship with include choice to accomplish highlight determination and order in the meantime. Interestingly, highlight extraction techniques are transformative, i.e., a change is connected on the information to extend occasions to another element space with bring down measurement. Foremost Component Analysis (PCA) and Singular Value Decomposition (SVD) are cases of this.

4.2.2 Feature Relevance

A feature selection technique selects a subset of relevant features from the full set of features. The definition of relevance varies from technique to technique. Based on its notion of relevance, a feature selection technique mathematically formulates a criterion for evaluating a set of features generated by a scheme that searches over the feature space.

Kohavi and John [24] define two degrees of relevance, viz., strong and weak. A feature s is called strongly relevant if removal of s de-teriorates the performance of a classifier. A feature s is called weakly relevant if it is not strongly relevant and removal of a subset of features containing s deteriorates the performance of the classifier. A feature is irrelevant if it is neither strongly nor weakly relevant.

4.3 Classification

Classification is a process of arrangement of optimized parameters so that useful information can extract in data. It assigns items in a collection to categories or classes. It results in the formation of a model.
In machine learning and statistics, classification is the problem of identifying to which of a set of categories (sub-populations) a new observation belongs, on the basis of a training set of data containing observations (or instances) whose category membership is known. An example would be assigning a given email into "spam" or "non-spam" classes or assigning a diagnosis to a given patient as described by observed characteristics of the patient (gender, blood pressure, presence or absence of certain symptoms, etc.). Classification is an example of pattern recognition.

In the terminology of machine learning, classification is considered an instance of supervised learning, i.e. learning where a training set of correctly identified observations is available. The corresponding unsupervised procedure is known as clustering, and involves grouping data into categories based on some measure of inherent similarity or distance.

Classification is done by:

- Classification by Random Forest Tree (RFT)

4.3.1 Random Forest Tree

Random forests or random decision forests are an ensemble learning method for classification, regression and other tasks, that operate by constructing a multitude of decision trees at training time and outputting the class that is the mode of the classes (classification) or mean prediction (regression) of the individual trees.

The first algorithm for random decision forests was created by Tin Kam Ho[25] using the random subspace method,[26] which, in Ho's formulation, is a way to implement

the "stochastic discrimination" approach to classification proposed by Eugene Kleinberg.[27][28][29]

An extension of the algorithm was developed by Leo Breiman[30] and Adele Cutler,[31] and "Random Forests" is their trademark.[32] The extension combines Breiman's "bagging" idea and random selection of features, introduced first by Ho[25] and later independently by Amit and Geman[33] in order to construct a collection of decision trees with controlled variance.

Random decision forests is suitable for the decision trees to reduce the habit of overfitting to their training set.

4.4 Evaluation

Accuracy is used to evaluate the performance of a NIDS in terms of correctness. It measures detection and failure rates as well as the number of false alarms that the system produces [34], [35], [36]. A NIDS with an accuracy of 95% implies that it correctly classifies 95 instances out of 100 to their actual class. Usually attacks are very diverse in manner and the number of attack traffic instances is generally much smaller than normal instances [37], [38], [39]. As a result, most currently available NIDSes generate a large amount of false alarms. In other words, the current state-of-the-art systems are not as efficient and accurate as ideally desired. The accuracy metric helps evaluate a NIDS to determine how correctly it can detect an attack. The accuracy of a NIDS can be assessed in terms of five measures:

1. sensitivity and specificity,
2. misclassification rate
3. confusion matrix entries
4. precision-recall and F measures
5. receiver operating character

Model will be evaluated on the bases of confusion matrix. A confusion matrix can be used to show how a NIDS performs in a general manner. The confusion matrix can be used in the case of n-class problems, whereas the matrix discussed earlier is used for 2-class problems. The size of the matrix depends on the number of distinct classes

in the dataset to be detected. It compares the class labels predicted by the classifier against the actual class labels. Multiple scores are measured such as: accuracy, precision, recall, F-measure [18].

Actual / Predicted	0	1
0	TP	FP
1	FN	TN

Accuracy:

Accuracy is correct predictions of data. Accuracy has two definitions, More commonly, it is a description of systematic errors, a measure of statistical bias, as these cause a difference between a result and a "true" value, ISO (International organization for standardization) calls this trueness.
Alternatively, ISO (International organization for standardization) defines accuracy as describing a combination of both types of observational error above (random and systematic), so high accuracy requires both high precision and high trueness. It is calculated as :

$$Accuracy = \frac{TN + TP}{TN + TP + FN + FP} \quad \text{……. Equation 4.1}$$

Precision:

Precision refers to positive predictive values of data. Precision is defined as the fraction of retrieved objects (e.g., documents) that are relevant to a given query or search request. Mathematically, it is the fraction obtained by dividing retrieved objects ∩ relevant objects by total retrieved objects, i.e., retrieved objects.

$$Precision = \frac{TP}{TP+FP} \quad \text{... ... ...} \quad \text{Equation 4.2}$$

Recall:

Recall is the measure of sensitivity of data. Recall is the fraction of the objects that are relevant to a given query or search request and are correctly retrieved. Mathematically, it is the fraction obtained by dividing retrieved objects ∩ relevant objects by the total number of relevant objects. In the case of a 2-class problem, recall is the same as sensitivity. In other words, recall is the probability that a relevant document is retrieved by a search request or a query. It is possible to return all the relevant objects (mixed with a lot of irrelevant ones) with respect to a given query to achieve 100% recall. Thus recall alone is not sufficient to judge the effectiveness of a retrieval method.

$$Recall = \frac{TP}{TP+FN} \quad \text{... ... ...} \quad \text{Equation 4.3}$$

F-measure:

It is the measure of test accuracy using precision and recall. F-measure or Balanced F-score is calculated by combining precision and recall into a simple metric. The traditional F-measure (also known as the F1 measure) is the harmonic mean of precision and recall [40]. For an n-class intrusion classification problem, it is the most preferred accuracy metric. F1 is maximum when precision and recall both reach 100%, i.e., 1 signifying that the classifier has 0% false alarms and also detects 100% of the attacks. Thus, a good classifier must strive to achieve an F1 measure as high as possible. Mathematically it is calculated as:

$$F\ measure\ = 2 * \frac{Precision * Recall}{Precision + Recall} \quad \text{... ... ...} \quad \text{Equation 4.4}$$

CHAPTER 5

EXPERIMENTS

An experiment is a procedure carried out to support, refute, or validate a hypothesis. Experiments provide insight into cause-and-effect by demonstrating what outcome occurs when a particular factor is manipulated. Experiments typically include controls, which are designed to minimize the effects of variables other than the single independent variable. This increases the reliability of the results, often through a comparison between control measurements and the other measurements. Scientific controls are a part of the scientific method. Ideally, all variables in an experiment are controlled (accounted for by the control measurements) and none are uncontrolled. In such an experiment, if all controls work as expected, it is possible to conclude that the experiment works as intended, and that results are due to the effect of the tested variable.

5.1 Classification by Random Forest Tree

5.1.1 Pre-processing

Data selected is UNSW NB-15 sets (obtained from https://cloudstor.aarnet.edu.au/plus/index.php/s/2DhnLGDdEECo4ys). Dataset has already training and testing set but, test set is double then training set.

Dataset labels are distinguished into two categories (0, 1). Category0 is representing normal and Category1 is attack.

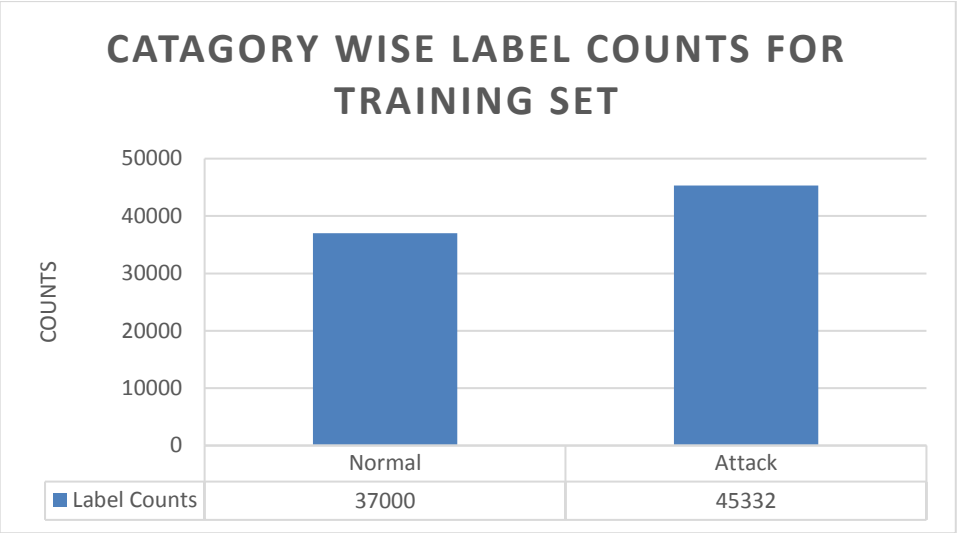


Figure 5-1 Training set Counts based on Label Categories.

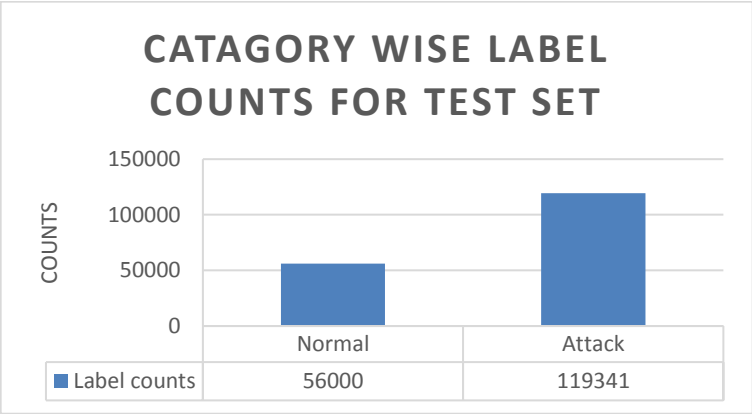


Figure 5-2 Testing set Counts based on Label Categories.

Lessens in training set then test set effect the accuracy of model. Because in training set there is less to overcome this firstly both datasets are merged and then split by 33% testing ratio.

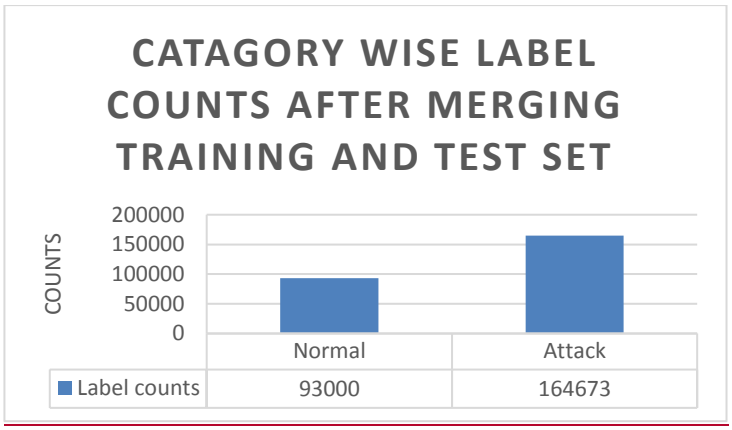


Figure 5-3 Merge the training and test set for fair split

All features are made numerical using one-Hot-encoding. The features are scaled to avoid features with large values that may weight too much in the results.

One-Hot-Encoding

In [11]:

```
enc = OneHotEncoder()
df_categorical_values_encenc = enc.fit_transform(df_categorical_values_enc)
df_cat_data = pd.DataFrame(df_categorical_values_encenc.toarray(),columns=dumcols)
# test set
testdf_categorical_values_encenc = enc.fit_transform(testdf_categorical_values_enc)
testdf_cat_data = pd.DataFrame(testdf_categorical_values_encenc.toarray(),columns=testdumcols)

df_cat_data.head()
```

Figure 5-4 Implementation of One-Hot Encoding

The dataset used is UNSW NB-15 and it contain seven type of attack categories and two type of labels that are collected from real network of UNSW. This dataset required pre-processing for the removal of extra data and convert categorical data to nominal data.

One Hot Encoding

One hot encoding is a process by which categorical variables are converted into a form that could be provided to ML algorithms to do a better job in prediction. One-hot encoding is often used for indicating the state of a state machine. When using binary or Gray code, a decoder is needed to determine the state. A one-hot state machine, however, does not need a decoder as the state machine is in the nth state if and only if the nth bit is high.

Advantages

1. Determining the state has a low and constant cost of accessing one flip-flop

2. Changing the state has the constant cost of accessing two flip-flops
3. Easy to design and modify
4. Easy to detect illegal states
5. Takes advantage of an FPGA's abundant flip-flops

Using a one-hot implementation typically allows a state machine to run at a faster clock rate than any other encoding of that state machine.[2]

Disadvantages

1. Requires more flip-flops than other encodings, making it impractical for PAL(Programmable array logic) devices
2. Many of the states are illegal.

5.1.2 Feature Selection

Eliminate redundant and irrelevant data by selecting a subset of relevant features that fully represents the given problem. When the subset is found Recursive Feature Elimination (RFE) is applied. RFE algorithm is applied to line up the parameters after optimization.

```
from sklearn.feature_selection import RFE
clf = RandomForestClassifier(n_jobs=2)
rfe = RFE(estimator=clf, n_features_to_select=25, step=1)
rfe.fit(X, y)
X_rfe=rfe.transform(X)
true=rfe.support_
rfecolindex=[i for i, x in enumerate(true) if x]
rfecolname=list(colNames[i] for i in rfecolindex)
print('Features selected for train:',rfecolname)
print()
```

```
Features selected for train: ['dur', 'dpkts', 'sbytes', 'dbytes', 'rate', 'sttl', 'sload', 'dload', 'sloss', 'sinpkt', 'dinpk
t', 'sjit', 'djit', 'stcpb', 'tcprtt', 'synack', 'ackdat', 'smean', 'dmean', 'ct_srv_src', 'ct_state_ttl', 'ct_dst_src_ltm', 'c
t_src_ltm', 'ct_srv_dst', 'state_FIN']
```

Figure 5-5 Implementation of Recursive Feature Elimination (RFE)

RFE (Recursive Feature Elimination)

RFE is a popular approach used with many classification algorithms to repeatedly construct a model and remove features with low weights. These approaches tend to be between filters and wrappers in terms of computational complexity.

5.1.3 Classification

In machine learning and statistics, classification is the problem of identifying to which of a set of categories (sub-populations) a new observation belongs, on the basis of a training set of data containing observations (or instances) whose category membership is known. An example would be assigning a given email into "spam" or "non-spam" classes or assigning a diagnosis to a given patient as described by observed characteristics of the patient (gender, blood pressure, presence or absence of certain symptoms, etc.). Classification is an example of pattern recognition. Random forest tree is applied by using sklearn in Figure 4-6

```
clf=RandomForestClassifier(bootstrap=True, class_weight=None, criterion='gini',
            max_depth=None, max_features='auto', max_leaf_nodes=None,
            min_impurity_decrease=0.0, min_impurity_split=None,
            min_samples_leaf=1, min_samples_split=2,
            min_weight_fraction_leaf=0.0, n_estimators=10, n_jobs=2,
            oob_score=False, random_state=None, verbose=0,
            warm_start=False)
```

Figure 5-6 Implementation of Random Forest classification

5.1.4 Evaluation

Exactness is utilized to assess the execution of a NIDS as far as accuracy. It gauges discovery and disappointment rates and in addition the quantity of false cautions that the framework produces. Using the test data to make predictions of the model. Multiple scores are considered such as: accuracy score, recall, f-measure, confusion matrix. Perform a 10-fold cross-validation.

```
pd.crosstab(y_test, y_pred, rownames=['Actual attacks'], colnames=['Predicted attacks'])
```

Figure 5-7 Confusion Matrix Implementation.

Actual / Predicted	0	1
0	28548	1996
1	2327	52162

Table 5-1 Confusion Matrix Results

```
from sklearn.metrics import classification_report

print("Model evaluation\n"+classification_report(y_test,y_pred))

from sklearn.model_selection import cross_val_score
accuracy = cross_val_score(clf, X_test, y_test, cv=10, scoring='accuracy')
print(accuracy)
print("Accuracy: %0.5f (+/- %0.5f)" % (accuracy.mean(), accuracy.std() * 2))
```

Figure 5-8 Implementation for evaluating results of UNSW NB-15

CHAPTER 6

RESULTS AND DISCUSSIONS

An outcome (additionally called upshot) is the last result of a grouping of activities or occasions communicated subjectively or quantitatively. Conceivable outcomes incorporate preferred standpoint, hindrance, pick up, damage, misfortune, esteem and triumph. There might be a scope of conceivable results related with an occasion contingent upon the perspective, chronicled separation or significance. Achieving no outcome can imply that activities are wasteful, inadequate, insignificant or imperfect.

6.1 Results

6.1.1 Classification by Random Forest Tree

In the field of machine learning and particularly the issue of factual characterization, a disarray grid, otherwise called a mistake lattice, is a particular table design that permits representation of the execution of a calculation, regularly a regulated learning one (in unsupervised learning it is typically called a coordinating network). Each line of the grid speaks to the occurrences in an anticipated class while every section speaks to the cases in a real class.

Actual / Predicted	0	1
0	28548	1996
1	2327	52162

Table 6-1 Results based on Confusion Matrix

Results are given below. All results are cross validated.

```
from sklearn.metrics import classification_report

print("Model evaluation\n"+classification_report(y_test,y_pred))

from sklearn.model_selection import cross_val_score
accuracy = cross_val_score(clf, X_test, y_test, cv=10, scoring='accuracy')
print(accuracy)
print("Accuracy: %0.5f (+/- %0.5f)" % (accuracy.mean(), accuracy.std() * 2))
```

Figure 6-1 Implementation for results calculation based on RFT

Figure 5-1 is the implementation of classification report. It automatically calculate results

	Precision	Recall	F1-Score	Support
0	0.92	0.93	0.93	30544
1	0.96	0.96	0.96	54489
Avg / total	0.95	0.95	0.95	85033

Table 6-2 Classification results report

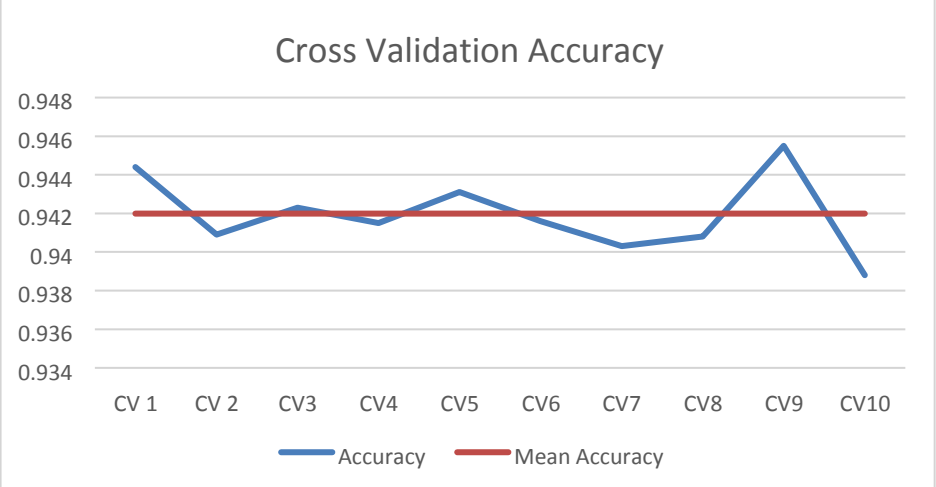


Figure 6-2 Cross Validation Accuracy measure using 10 folds

Accuracy=0.94199 (+/-0.00378).

Accuracy was effected in previous researches, because in training set there are less records then testing set. As always the training set is greater than test set. This issue is resolved by combine both sets and re-distribute them. This help to fairly distribute the data. If data is fairly distributed it help to train the model effectively. Results will be good.

6.2 Accuracy tracking by no of features

A cross validation graph which tells the ratio accuracy by no of features selected. As by figure 5-3 the accuracy increase drastically till 25 features. Then fluctuate to and far till last feature. This shows that we don't have to classify all feature but we can find an optimal point of features where accuracy defeat the collective computation power of all features. This help to reduce the computation cost and work for betterment of result.

```
# Create the RFE object and compute a cross-validated score.
# The "accuracy" scoring is proportional to the number of correct
# classifications
rfecv_DoS = RFECV(estimator=clf_DoS, step=1, cv=10, scoring='accuracy')
rfecv_DoS.fit(X_DoS_test, Y_DoS_test)
# Plot number of features VS. cross-validation scores
plt.figure()
plt.xlabel("Number of features selected")
plt.ylabel("Cross validation score (nb of correct classifications)")
plt.title('RFECV DoS')
plt.plot(range(1, len(rfecv_DoS.grid_scores_) + 1), rfecv_DoS.grid_scores_)
plt.show()
```

Figure 4-9 Show cross validation of data

This code telling that we are using ten cross-validations to get authentic results. This actually telling that how many features are involved in formation of label. So that minimum features will be use to achieve accuracy.

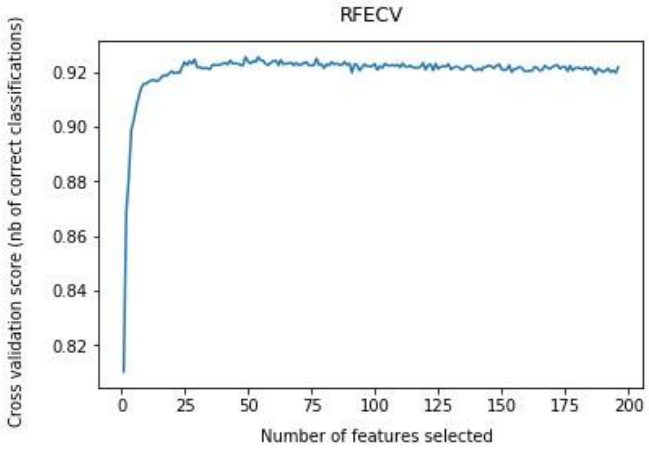


Figure 6-3 Cross validation accuracy graph by features selected

As by Table 5-1 when accuracy is compared with accuracy defined previous the results shows are:

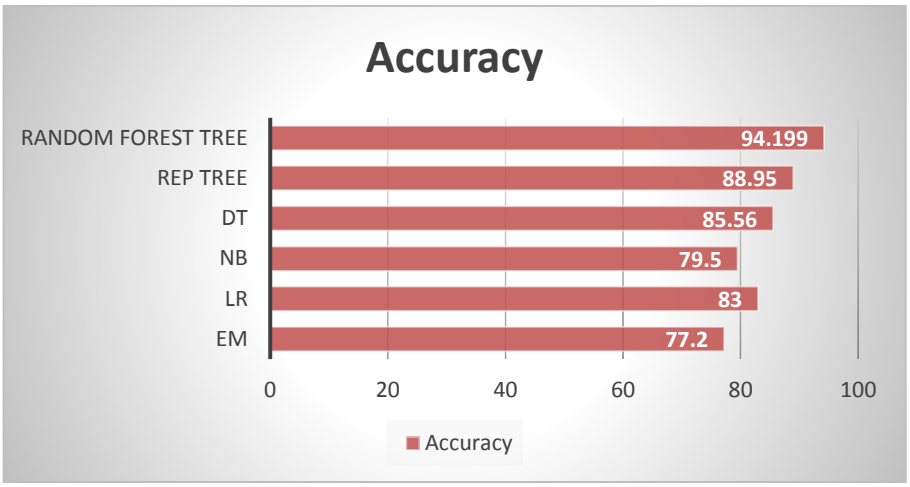


Figure 6-4 Accuracy comparison to previous Approaches

CHAPTER 7

CONCLUSION AND RECOMMENDATIONS

7.1 Conclusion

IDS is todays want because, it helps the people to stay up their confidentiality and integrity. Intrusion that disturbs the safety and secrecy of the structure, has become chief concern of several organizations.

Hence this problem is non-linear separable but RFT improve accuracy but there need more improvement for data. There is a want of robust IDS which could observe utterly completely different attack with high attack recognition accuracy. But there is a great gap in accuracy.

7.2 Recommendations

The efficiency of an individual classifier, either for supervised or un-supervised learning is not good for classification of all attack categories as well as normal instances. It is possible to obtain good classification accuracy by combination of multiple well performed classifiers. The objective is to create an IDS with its best performance

Hence this problem is non-linear separable and security never compromises but RFT improve accuracy but there need more improvement for data, hence something new is required for a great change like NN, CNN and neural network trained itself and its low learning rate help to improve accuracy at every iteration. Neural network will be

helpful. Deep forest can be used be a break through because it is the alternate of neural network. But there are limitations of neural network, it require a high processing power to train the model and it will be time consuming.

REFERENCES

[1] A. A. Rao "A Java Based Network Intrusion Detection System (IDS)". In Andhra university college of engineering, India, proceeding of the 2006 IJMEINTERTECH Conference, 2006.

[2] A. Ö. Corresp, H. Erdem "The impact of using large training data set KDD99 on classification accuracy" PeerJ PrePrints (2017).

[3] Newsweek. Sony Cyber Attack One of Worst in Corporate History. [online] Available at: http://www.newsweek.com/sony-cyber-attackhttp://www.newsweek.com/sony-cyber-attack-worstcorporate-history-thousands-files-areleaked-289230worstcorporate-history-thousands-files-areleaked-289230 [Accessed 22 Dec. 2017].

[4] B. Cha, K. Park, and J. Seo, "Neural Networks Techniques for Host Anomaly Intrusion Detection using Fixed Pattern Transformation" in ICCSA. LNCS 3481. 254-263, 2005.

[5] KDD Cup 1999 Data. [online] Available at: http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html [Accessed 11 May, 2017].

[6] The UNSW-NB15 data set description. [online] Available at: https://www.unsw.adfa.edu.au/unsw-canberra-cyber/cybersecurity/ADFA-NB15https://www.unsw.adfa.edu.au/unsw-canberra-cyber/cybersecurity/ADFA-NB15-Datasets/Datasets/ [Accessed 11 May, 2017].

[7] J. McHugh, "Testing intrusion detection systems: a critique of the 1998 and 1999 DARPA intrusion detection system evaluations as performed by Lincoln laboratory," ACM Transactions on Information and System Security, vol. 3, no. 4, pp. 262–294, 2000.

[8] M. Tavallaee, E. Bagheri, W. Lu, A. A. Gorbani, "A detailed analysis of KDD CUP 99 dataset" Submitted to Second IEEE Symposium on Computational Intelligence for Security and Defense Applications (CISDA), 2009.

[9] H. G. Kayacık, A. N. Z. Heywood, M. I. Heywood, "Selecting Features for Intrusion Detection: A Feature Relevance Analysis on KDD 99 Intrusion Detection Datasets",2005.

[10] A. A. Olusola, A. S. Oladele and D. O. Abosede, "Analysis of KDD '99 Intrusion Detection Dataset for Selection of Relevance Features" Proceedings

of the World Congress on Engineering and Computer Science 2010 Vol I WCECS 2010, October 20-22, 2010, San Francisco, USA ISBN: 97898817012-0-6, 2010.

[11] M. Tavallaee, E. Bagheri, W. Lu, A. A. Gorbani, "A Detailed Analysis of the KDD CUP 99 Data Set," Submitted to Second IEEE Symposium on Computational Intelligence for Security and Defense Applications (CISDA), 2009.

[12] J. Han, and M. Kamber, "Data Mining Concepts and Techniques" Morgan Kaufmann publishers .an imprint of Elsevier, ISBN 978-1-55860-901-3. Indian reprint ISBN 978-81-312-0535-8, 2010.

[13] Y. X. Ding, M. Xiao, and A. W. Liu, "Research and implementation on snort-based hybrid intrusion detection system," Proceedings of the 2009 International Conference on Machine Learning and Cybernetics, vol. 3, no. July, pp. 1414–1418, 2009.

[14] Anon. Saggi e Memorie di storia dell'arte. [online] Available at: http://www.ccianet.org/wpcontent/uploads/2014/04/Cybersecurity.pdf [Accessed 22 Dec. 2017].

[15] M. Belouch, S. E. Hadaj, M. Idhammad at "A Two-Stage Classifier Approach using RepTree Algorithm for Network Intrusion Detection" International Journal of Advanced Computer Science and Applications, Vol. 8, No. 6, 2017.

[16] N. Moustafa, J. Slay "A hybrid feature selection for network intrusion detection systems: central points and association rules" Originally published in the Proceedings of the 16th Australian Information Warfare Conference (pp. 5-13), held on the 30 November - 2 December,2015, Edith Cowan University, Joondalup Campus, Perth, Western Australia.

[17] J. F. C. Joseph, A. Das, B. C. Seet, B. S. Lee, "Cross-Layer Detection of Sinking Behavior in Wireless Ad Hoc Networks Using SVM and FDA." IEEE Transaction on dependable and secure computing, Vol. 8, No. 2, March April 2011.

[18] S. Peddabachigari, A. Abraham, C. Grosan, J. Thomas (2005). "Modeling Intrusion Detection Systems using Hybrid Intelligent Systems." Journal of Network and Computer Applications, 2005.

[19] de Argaez, E. Internet world stats, (Online) http://www.internetworldstats.com [Accessed 20 Feb. 2018].

[20] A look back at cybersecurity in 2017, (Online) https://www.csoonline.com/article/3239405/data-breach/a-look-back-at-cybersecurity-in-2017.html [Accessed 20 Feb. 2018].

[21] State of Cybercrime 2017 (Online) https://www.csoonline.com/article/3239405/data-breach/a-look-back-at-cybersecurity-in-2017.html [Accessed 20 Feb. 2018].

[22] M. Thottan, and C. Ji, Anomaly detection in IP networks. IEEE Transactions on Signal Processing 51, 8 (August 2003), 2191–2204.

[23] KDDcup99. Knowledge discovery in databases DARPA archive. (Online) http://www.kdd.ics.uci.edu/databases/kddcup99, 1999. [Accessed 20 Feb. 2018]

[24] Kohavi, R., and John, G. Wrappers for feature subset selection. Artificial Intelligence 97, 1 (1997), 273–324.

[25] Ho, Tin Kam (1995). Random Decision Forests. Proceedings of the 3rd International Conference on Document Analysis and Recognition, Montreal, QC, 14–16 August 1995. pp. 278–282.

[26] Ho TK (1998). "The Random Subspace Method for Constructing Decision Forests". IEEE Transactions on Pattern Analysis and Machine Intelligence. 20 (8): 832–844. doi:10.1109/34.709601.
[27] Kleinberg E (1990). "Stochastic Discrimination". Annals of Mathematics and Artificial Intelligence. 1 (1–4): 207–239. doi:10.1007/BF01531079.
[28] Kleinberg E (1996). "An Overtraining-Resistant Stochastic Modeling Method for Pattern Recognition". Annals of Statistics. 24 (6): 2319–2349. doi:10.1214/aos/1032181157. MR 1425956.
[29] Kleinberg E (2000). "On the Algorithmic Implementation of Stochastic Discrimination". IEEE Transactions on PAMI. 22 (5).
[30] Breiman L (2001). "Random Forests". Machine Learning. 45 (1): 5–32. doi:10.1023/A:1010933404324.
[31] Liaw A (16 October 2012). "Documentation for R package randomForest". Retrieved 15 March 2013.
[32] U.S. trademark registration number 3185828, registered 2006/12/19.
[33] Amit Y, Geman D (1997). "Shape quantization and recognition with randomized trees". Neural Computation. 9 (7): 1545–1588. doi:10.1162/neco.1997.9.7.1545
[34] Axelsson, S. The base-rate fallacy and its implications for the difficulty of intrusion detection. In Proc. of the 6th ACM Conference on Computer and Communications Security (New York, NY, USA, 1999), ACM, pp. 1–7.
[35] Axelsson, S. The base-rate fallacy and the difficulty of intrusion detection. ACM Transactions on Information and System Security 3, 3 (August 2000), 186–205.
[36] Lippmann, R. P., Fried, D. J., Graf, I., Haines, J., Kendall, K., McClung, D., Weber, D., Wyschogord, S. W. D., Cunningham, R. K., and Zissman, M. A. Evaluating intrusion detection systems: The 1998 DARPA offline intrusion detection evaluation. In Proc. of the DARPA Information Survivability Conference and Exposition (January 2000), pp. 12–26.
[37] Portnoy, L., Eskin, E., and Stolfo, S. J. Intrusion detection with unlabeled data using clustering. In Proc. of ACM Workshop on Data Mining Applied to Security (2001).
[38] Joshi, M. V., Agarwal, R. C., and Kumar, V. Mining Needle in a Haystack: Classifying Rare Classes via Two-phase Rule Induction. In Proc. of the 7th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (2001), ACM, pp. 293–298.
[39] Dokas, P., Ertoz, L., Lazarevic, A., Srivastava, J., and Tan, P. N. Data mining for network intrusion detection. In Proc. of the NSF Workshop on Next Generation Data Mining (November 2002).
[40] Weiss, S. M., and Zhang, T. The Handbook of Data Mining. Lawrence Erlbaum Assoc. Inc., 2003, pp. 426–439.
[41] P.Gogoi et al, "Packet and flow based network intrusion dataset."Contemporary Computing". Springer Berlin Heidelberg, 2012.P 322-334.
[42] McHugh, John, "Testing intrusion detection systems: a critique of the 1998 and 1999 DARPA intrusion detection system evaluations as performed by Lincoln Laboratory". ACM transactions on Information and system Security, 3, 2000, p 262-294.
[43] V.Mahoney, and K.Philip, "An analysis of the 1999 DARPA/Lincoln Laboratory evaluation data for network anomaly detection."Recent Advances in Intrusion Detection". Springer Berlin Heidelberg, 2003.

[44] A.Vasudevan, E. Harshini, and S. Selvakumar, "SSENet-2011: a network intrusion detection system dataset and its comparison with KDD CUP 99 dataset", Internet (AH-ICI), 2011, Second Asian Himalayas International Conference on. IEEE.

CHAPTER 8

APPENDICES

Appendix A

8.2 Datasets:

Many researchers have introduce many datasets for the assessment of intrusion system to differentiate known and unknown attacks. Datasets can be categorised in three types based on sources (i) Public datasets (ii) Private datasets (iii) network simulated datasets. In this the datasets we use will be network simulated dataset and they are created by simulating normal and attack in a considered scenarios

8.2.1 KDD CUP 1999 dataset

This dataset was created for The Third International Knowledge Discovery and Data Mining Tools Competition, which was held in conjunction with KDD-99 “The Fifth International Conference on Knowledge Discovery and Data Mining”. The competition task was to build a network intrusion detector, a predictive model capable of distinguishing between ``bad'' connections, called intrusions or attacks, and ``good'' normal connections. This database contains a standard set of data to be audited, which includes a wide variety of intrusions simulated in a military network environment [5].

KDD99 is benchmark dataset [23] for intrusion detection. In this connection is represented by 41 features. 37 features of them are numeric, 3 are ordinal/categorical and one is label feature. Description of features are:

Sr.	Features Name
1	Duration
2	Protocol type
3	Service
4	Flag
5	Source bytes
6	Destination bytes
7	Land
8	Wrong fragment
9	Urgent
10	Hot
11	Failed logins
12	Logged in
13	Num compromised
14	Root shell
15	Su attempted
16	Num root
17	Num file creations
18	Num shells
19	Num access files
20	Num outbound cmds
21	Is host login
22	Is guest login
23	Count
24	Srv count
25	Serror rate
26	Srv serror rate
27	Rerror rate
28	Srv rerror rate
29	Same srv rate
30	Diff srv rate
31	Srv diff host rate
32	Dst host count
33	Dst host srv count
34	Dst host same srv rate
35	Dst host diff srv rate
36	Dst host same source port rate
37	Dst host srv diff host rate
38	Dst host serror rate
39	Dst host srv serror rate
40	Dst host rerror rate
41	Dst host srv rerror rate
42	**Labels**

Table 8-1 Feature description of KDD99 dataset

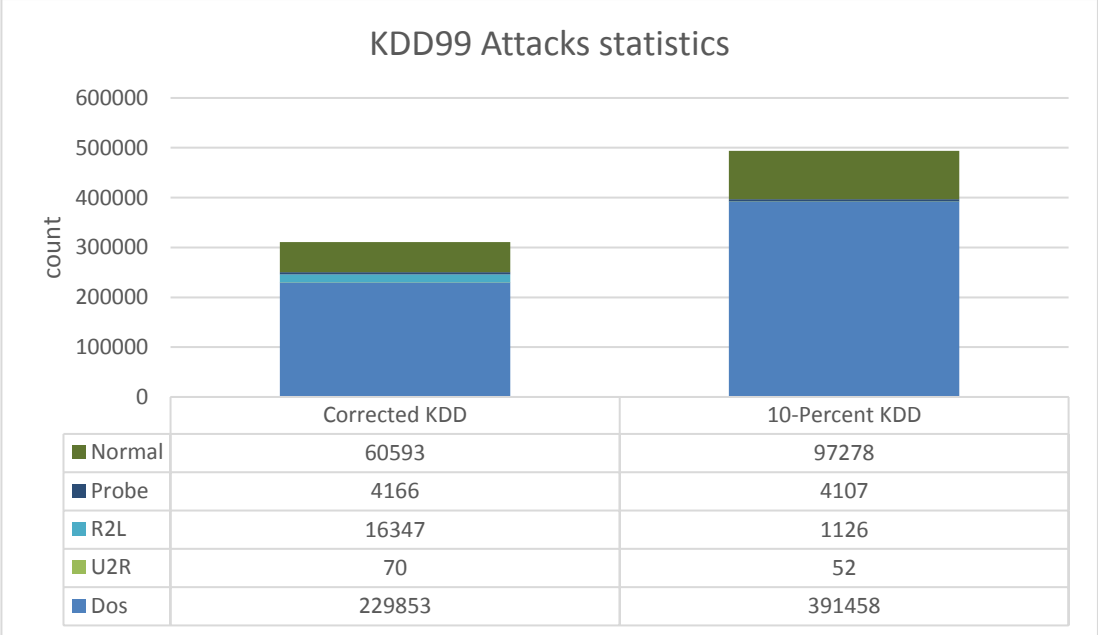
This dataset has four attack categories with a Normal

1. DoS (Denial of service) attack
2. U2R (User to root) attack
3. R2L (Remote to local) attack
4. Probe attack
5. Normal attack

KDD99 is available in two version

1. Corrected KDD dataset
2. 10-percent KDD

Detail of versions of KDD99 is described in Figure 6-1.



	Corrected KDD	10-Percent KDD
Normal	60593	97278
Probe	4166	4107
R2L	16347	1126
U2R	70	52
Dos	229853	391458

Figure 8-1 KDD99 Detailed statistics for attack

8.2.1.1 DoS (Denial of service)

A Denial-of-Service (DoS) attack is an attack meant to shut down a machine or network, making it inaccessible to its intended users. DoS attacks accomplish this by flooding the target with traffic, or sending it information that triggers a crash. In both instances, the DoS attack deprives legitimate users (i.e. employees, members, or

account holders) of the service or resource they expected. Statistics for 10-percent KDD99 attacks are described in 6-2 below

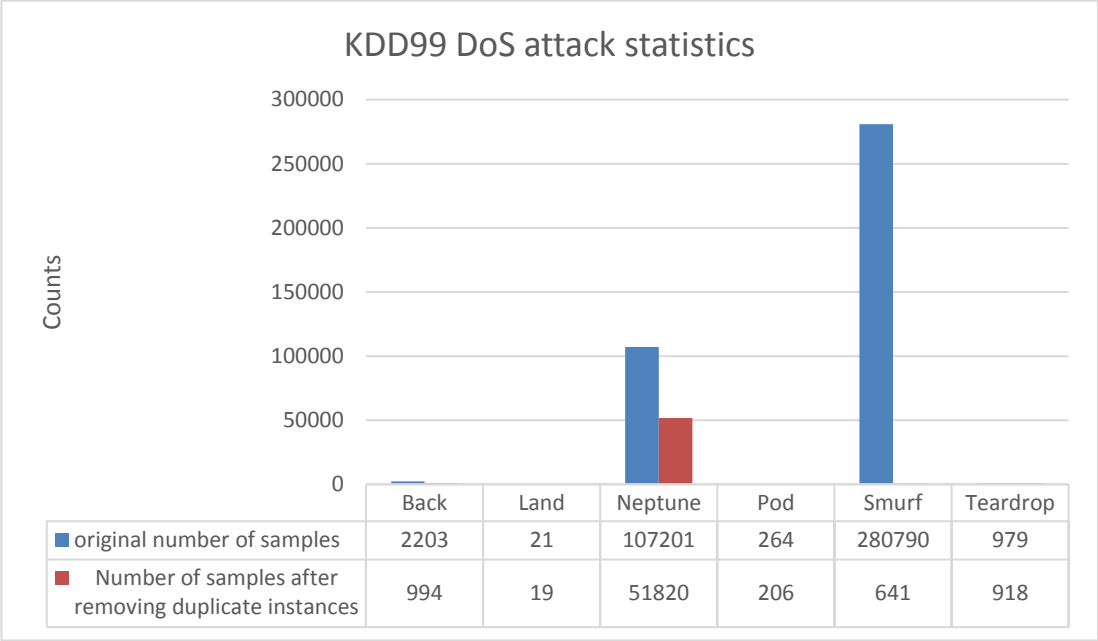


Figure 8-2 10-Percent KDD99 Detailed statistics for DoS attack

8.2.1.2 U2R (User to root)

In this attack type the hacker attempt to gain access of victim machine as an authorized user by some way. The hacker try to find some vulnerability in the system to gain access as super user. Through super user the hacker gain full access. By gain full access the hacker can install backdoor, exploits, manipulate system files. Some tools are Yaga, Sqlattack etc. Detailed of user to root access is described in Figure 6-3.

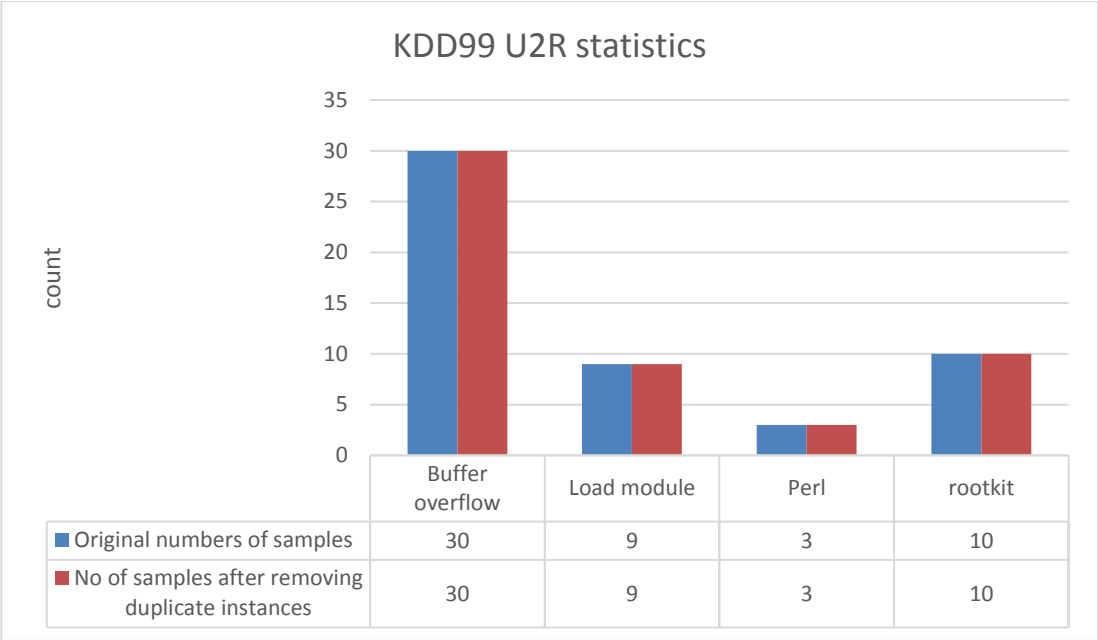


Figure 8-3 10-Percent KDD99 Detailed statistics for U2R attack

8.2.1.3 R2L (Remote to local) attack

The attack in which the remote attacker without having account on local machine send packet to machine to gain access based on vulnerability. To gain access two types of dictionary attacks are used. Online dictionary attack and offline dictionary attack to acquire password by guessing possible username and password. Some tools are Netcat, ntfsdos. Detailed statistics are below in Figure 6-4.

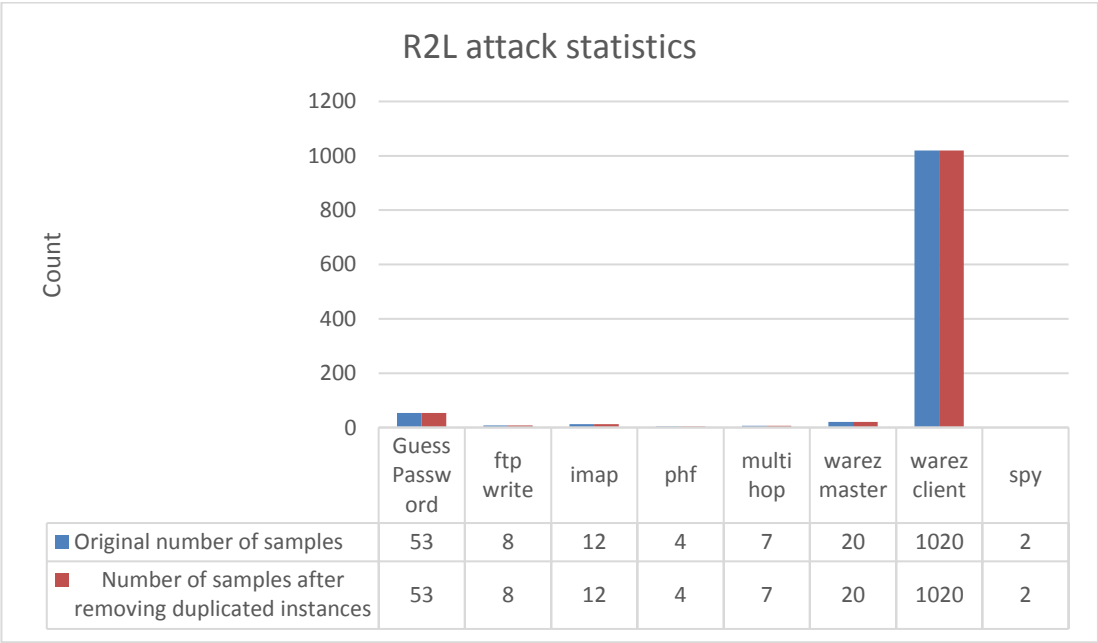


Figure 8-4 10-Percent KDD99 Detailed statistics for R2L attack

8.2.1.4 Probe attack

Probe is a type of foot printing attack. Foot printing is a method to gather information about the victim. Like to send an empty method either the destination exist or connected to network. For this purpose PING is a utility. Other tools are Nmap, P0f, Xprobe, Queso etc. Detailed of probe attacks in KDD99 10-percent is in Figure 7-5

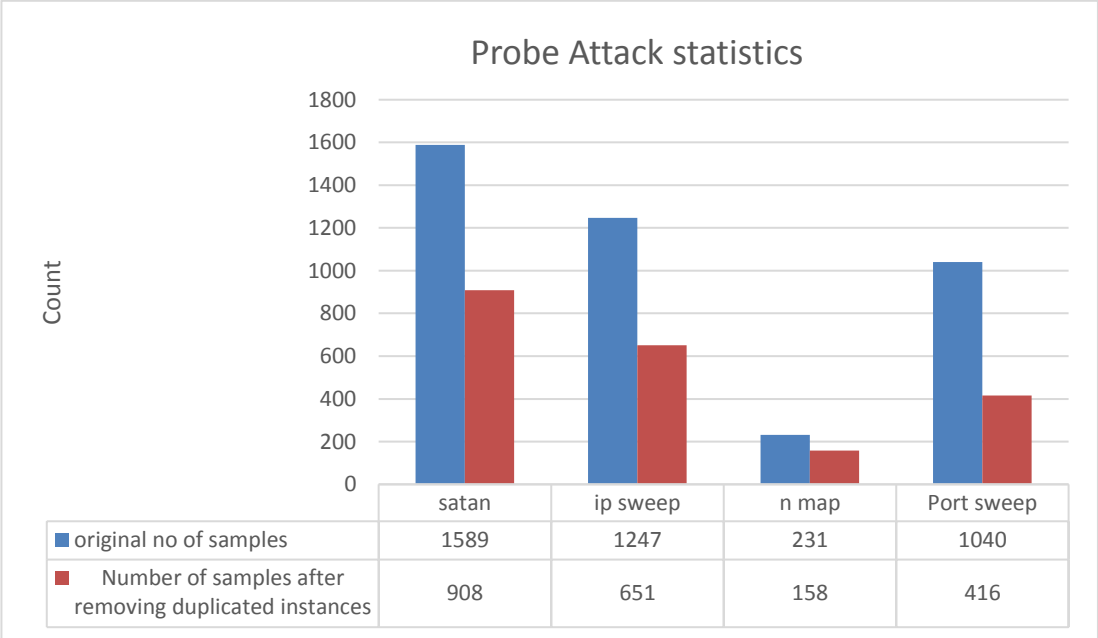


Figure 8-5 10-Percent KDD99 Detailed statistics for R2L attack

8.2.1.5 Normal

Normal category is said to be normal operations in a network.

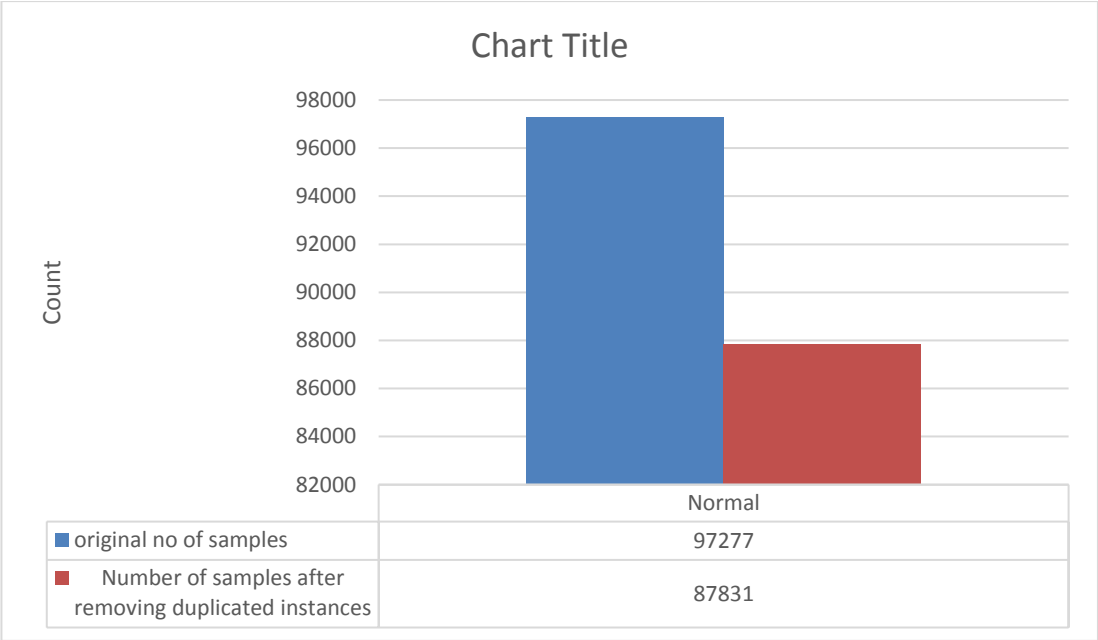


Figure 8-6 10-Percent KDD99 Detailed statistics for Normal

8.2.2 NSL-KDD

NSL-KDD is network baased intrusion detection dataset. IT is the filtered version of KDD CUP 1999 benchmark intrusion detection dataset. In KDD 99 dataset there is a large amount of redundant records. To resolve this only one copy of redundant record is kept. NSL-KDD consist of two datasets. (i) KDDTrain+ and (ii) KDDTest+. Statistics are discussed below.

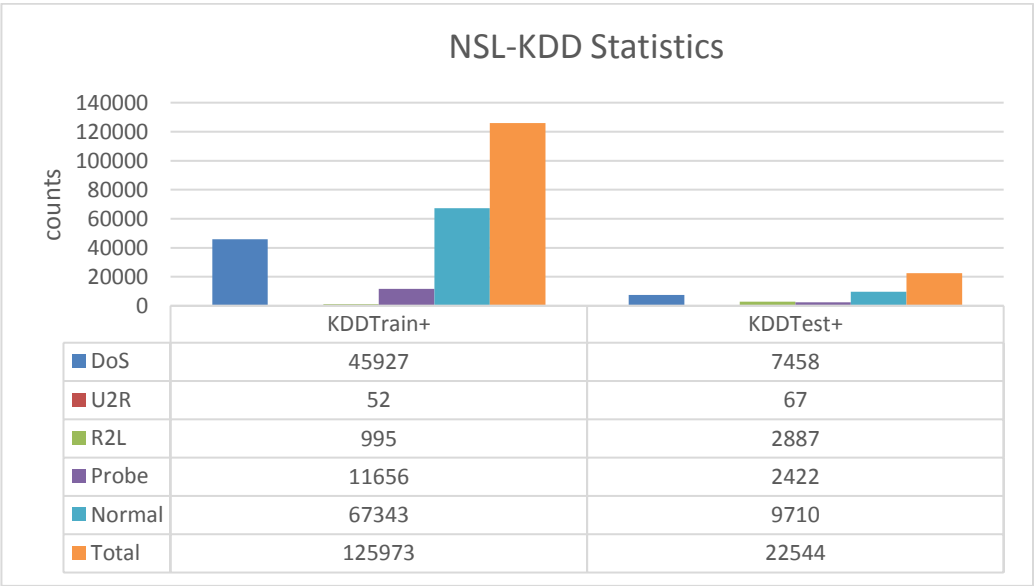


Figure 8-7 Attack Distribution of NSL-KDD dataset

8.2.3 UNSW NB-15

The raw network packets of the UNSW-NB 15 data set is created by the IXIA Perfect Storm tool in the Cyber Range Lab of the Australian Centre for Cyber Security (ACCS) for generating a hybrid of real modern normal activities and synthetic contemporary attack behaviours. Tcp dump tool is utilised to capture 100 GB of the raw traffic (e.g., Pcap files). This dataset has nine families of attacks, namely, Fuzzers, Analysis, Backdoors, DoS, Exploits, Generic, Reconnaissance, Shellcode and Worms. The Argus, Bro -IDS tools are utilised and twelve algorithms are developed to generate totally 49 features with the class label [6]. Features are discussed below.

No.	Name	Type	Description
1	srcip	nominal	Source IP address
2	sport	integer	Source port number
3	dstip	nominal	Destination IP address
4	dsport	integer	Destination port number
5	proto	nominal	Transaction protocol
6	state	nominal	Indicates to the state and its dependent protocol, e.g. ACC, CLO, CON, ECO, ECR, FIN, INT, MAS, PAR, REQ, RST, TST, TXD, URH, URN, and (-) (if not used state)
7	dur	Float	Record total duration
8	sbytes	Integer	Source to destination transaction bytes
9	dbytes	Integer	Destination to source transaction bytes
10	sttl	Integer	Source to destination time to live value
11	dttl	Integer	Destination to source time to live value
12	sloss	Integer	Source packets retransmitted or dropped
13	dloss	Integer	Destination packets retransmitted or dropped
14	service	nominal	http, ftp, smtp, ssh, dns, ftp-data ,irc and (-) if not much used service
15	Sload	Float	Source bits per second
16	Dload	Float	Destination bits per second
17	Spkts	integer	Source to destination packet count
18	Dpkts	integer	Destination to source packet count
19	swin	integer	Source TCP window advertisement value
20	dwin	integer	Destination TCP window advertisement value
21	stcpb	integer	Source TCP base sequence number
22	dtcpb	integer	Destination TCP base sequence number

23	smeansz	integer	Mean of the ?ow packet size transmitted by the src
24	dmeansz	integer	Mean of the ?ow packet size transmitted by the dst
25	trans_depth	integer	Represents the pipelined depth into the connection of http request/response transaction
26	res_bdy_len	integer	Actual uncompressed content size of the data transferred from the server's http service.
27	Sjit	Float	Source jitter (mSec)
28	Djit	Float	Destination jitter (mSec)
29	Stime	Timestamp	record start time
30	Ltime	Timestamp	record last time
31	Sintpkt	Float	Source interpacket arrival time (mSec)
32	Dintpkt	Float	Destination interpacket arrival time (mSec)
33	tcprtt	Float	TCP connection setup round-trip time, the sum of 'synack' and 'ackdat'.
34	synack	Float	TCP connection setup time, the time between the SYN and the SYN_ACK packets.
35	ackdat	Float	TCP connection setup time, the time between the SYN_ACK and the ACK packets.
36	is_sm_ips_ports	Binary	If source (1) and destination (3)IP addresses equal and port numbers (2)(4) equal then, this variable takes value 1 else 0
37	ct_state_ttl	Integer	No. for each state (6) according to specific range of values for source/destination time to live (10) (11).
38	ct_flw_http_mthd	Integer	No. of flows that has methods such as Get and Post in http service.
39	is_ftp_login	Binary	If the ftp session is accessed by user and password then 1 else 0.
40	ct_ftp_cmd	integer	No of flows that has a command in ftp session.
41	ct_srv_src	integer	No. of connections that contain the same service (14) and source address (1) in 100 connections according to the last time (26).
42	ct_srv_dst	integer	No. of connections that contain the same service (14) and destination address (3) in 100 connections according to the last time (26).
43	ct_dst_ltm	integer	No. of connections of the same destination address (3) in 100 connections according to the last time (26).
44	ct_src_ ltm	integer	No. of connections of the same source address (1) in 100 connections according to the last time (26).
45	ct_src_dport_ltm	integer	No of connections of the same source address (1) and the destination port (4) in 100 connections according to the last time (26).
46	ct_dst_sport_ltm	integer	No of connections of the same destination address (3) and the source port (2) in 100 connections according to the last time (26).

47	ct_dst_src_ltm	integer	No of connections of the same source (1) and the destination (3) address in in 100 connections according to the last time (26).
48	attack_cat	nominal	The name of each attack category. In this data set , nine categories e.g. Fuzzers, Analysis, Backdoors, DoS Exploits, Generic, Reconnaissance, Shellcode and Worms
49	Label	binary	0 for normal and 1 for attack records

Table 8-2 UNSW-NB 15 features description

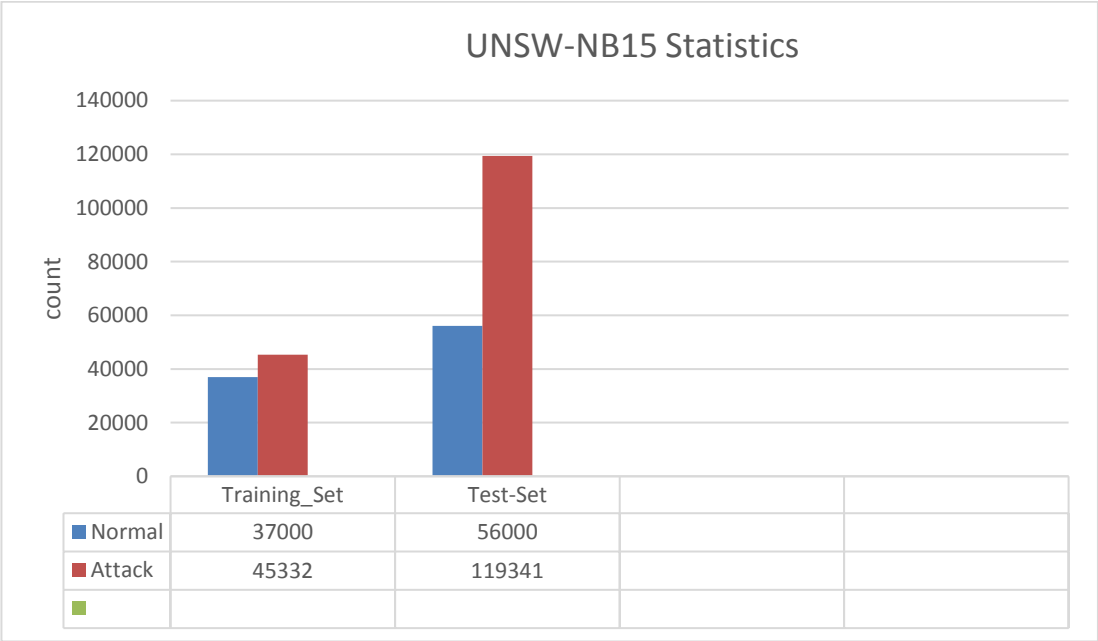


Figure 8-8 UNSW-NB training and testing dataset statistic