*Available online at http://www.ijcrls.com*

*International Journal of Current Research in Life Sciences*
*Vol. 07, No. 04, pp.1738-1745, April, 2018*

ISSN: 2319-9490

# RESEARCH ARTICLE

# AN ADVANCED INTRUSION DETECTION SYSTEM FOR THE NETWORKING USING DATA MINING APPROACH

***Shalini Dhar**

Department of Computer Science and IT, SHUATS, Allahabad, India

Received 20th February, 2018; Accepted 03rd March, 2018; Published 30th April, 2018

## ABSTRACT

Nowadays, many organizations and companies use Internet services as their communication and marketplace to do business. Due to increased number of internet users there is a problem due to intrusion which may damage data and information stored in computer server or data base server. So we need a filter which is able to filter malicious data and normal data. Intrusion detection is the process of monitoring and analysing the events occurring in a computer system in order to detect signs of security problems. The intrusion detection and other security technologies such as cryptography, authentication and firewalls have gained in importance in last few years. The present study gives an advanced Intrusion Detection System (IDS) along with data mining techniques by using k-means and outlier both approaches. The k-means approach uses clustering mechanisms to group the traffic flow data into normal and anomalous clusters. Outlier detection calculates an outlier score for each flow record. This score is called the neighbourhood outlier factor (NOF), whose value decides whether a particular traffic flow is normal or anomalous. The performance of these two approaches is compared by means of various confusion matrix and performance metrics like false positive rate, sensitivity, specificity, classification rate and precision and an analysis is done to find out that which one of the two approaches is better to be used for intrusion detection using traffic flows.

***Key words:*** Intrusion detection system, data mining, security, networks etc.

**Citation: Shalini Dhar, 2018.** "An advanced intrusion detection system for the networking using data mining approach" *International Journal of Current Research in Life Sciences*, 7, (04), 1738-1745.

## INTRODUCTION

Intuitively, intrusions in an information system are the activities that violate the security policy of the system, and intrusion detection is the process used to identify intrusions. Intrusion detection has been studied for approximately 20 years. It is based on the beliefs that an intruder's behaviour will be noticeably different from that of a legitimate user and that many unauthorized actions will be detectable. Intrusion detection systems (IDSs) are usually deployed along with other preventive security mechanisms, such as access control and authentication, as a second line of defence that protects information systems. There are several reasons that make intrusion detection a necessary part of the entire defences system. First, many traditional systems and applications were developed without security in mind. In other cases, systems and applications were developed to work in a different environment and may become vulnerable when deployed in the current environment. (For example, a system may be perfectly secure when it is isolated but become vulnerable when it is connected to the Internet.) Intrusion detection provides a way to identify and thus allow responses to, attacks against these systems.

Second, due to the limitations of information security and software engineering practice, computer systems and applications may have design flaws or bugs that could be used by an intruder to attack the systems or applications. As a result, certain preventive mechanisms (e.g., firewalls) may not be as effective as expected. Intrusion detection complements these protective mechanisms to improve the system security. Moreover, even if the preventive security mechanisms can protect information systems successfully, it is still desirable to know what intrusions have happened or are happening, so that we can understand the security threats and risks and thus be better prepared for future attacks. In spite of their importance, IDSs are not replacements for preventive security mechanisms, such as access control and authentication. Indeed, IDSs themselves cannot provide sufficient protection for information systems. As an extreme example, if an attacker erases all the data in an information system, detecting the attacks cannot reduce the damage at all. Thus, IDSs should be deployed along with other preventive security mechanisms as a part of a comprehensive defence system. Intrusion detection techniques are traditionally categorized into two methodologies: anomaly detection and misuse detection. Anomaly detection is based on the normal behaviour of a subject (e.g., a user or a system); any action that significantly deviates from the normal behaviour is considered intrusive.

****Corresponding author:*** **Shalini Dhar,**
Department of Computer Science and IT, SHUATS, Allahabad, India.

Misuse detection catches intrusions in terms of the characteristics of known attacks or system vulnerabilities; any action that conforms to the pattern of a known attack or vulnerability is considered intrusive. Alternatively, IDSs may be classified into host-based IDSs, distributed IDSs, and network-based IDSs according to the sources of the audit information used by each IDS. Host-based IDSs get audit data from host audit trails and usually aim at detecting attacks against a single host; distributed IDSs gather audit data from multiple hosts and possibly the network that connects the hosts, aiming at detecting attacks involving multiple hosts. Network-based IDSs use network traffic as the audit data source, relieving the burden on the hosts that usually provide normal computing services. Data Mining is the process of analyzing data from different perspectives and summarizing the results as useful information. It has been defined as "the non-trivial process of identifying valid, novel, potentially useful, and ultimately understandable patterns in data". process of data mining uses machine learning, statistics, and visualization techniques to discover and present knowledge in a form that is easily comprehensible. The word “Knowledge” in Knowledge Discovery Database refers to the discovery of patterns which are extracted from the processed data. A pattern is an expression describing facts in a subset of the data. Thus, the difference between KDD and data mining is that “KDD refers to the overall process of discovering knowledge from data while data mining refers to application of algorithms for extracting patterns from data without the additional steps of the KDD process”. Fayyad et al (1996) However, since Data Mining is a crucial and important part of the KDD process, most researchers use both terms interchangeably. The performance of the proposed intrusion detection system is evaluated using JAVA and MATLAB. The performance of the two approaches (k-means and outlier detection) for intrusion detection when implemented in the system. The main objectives of the research are to design an algorithm for intrusion detection system with data mining techniques with study the intrusion detection to identify normal and malicious actions on the system and also evaluate the Receiver operating characteristic (ROC) curves for analysis of intrusion detected data.

**System Architecture:** The system architecture is as shown in figure 3.1. The IDS developed in this thesis consists of the following modules.

**Packet capturing module:** The packets arriving from the internet are captured by this module in real time and are stored in a pcap file for further analysis. The captured packets are of any protocol as and when they are arriving.

**Packet reading module:** This module opens the PCAP file and reads the packets contained in it. The packets are grouped according to their protocols in a file. This module writes the TCP, UDP and ICMP packets to an external file for further analysis.

**Flow exporter module:** This module groups the packets into the flows. The features from the packets are extracted and read by this module based on which a flow record is generated. A flow generally consists of the following five parameters.

- Source IP.
- Destination IP.
- Protocol.
- Source port.
- Destination port.

If there is a deviation in any of these flow values then a new flow record is generated. However, the work presented in this thesis groups the packets in accordance with the most commonly used flow records protocol called as the Net Flow version 5. The flow exporter module, therefore, groups the packets into flows according to the following fields.

- Flow record ID.
- Layer 4 protocols (TCP, UDP or ICMP).
- Source IP.
- Source port.
- Destination IP.
- Destination port.
- Total packets in flow record.
- Total bytes in flow record.

**Anomaly detector module**

This module hosts the k-means and outlier detection algorithms to detect the intrusions present in each flow record. Each flow record is passed to each of the algorithms to detect the intrusions individually. The k-means approach makes use of the NSL-KDD Dataset and pcap file captured in international competitions to learn about the different types of anomalies in the network traffic. This knowledge was used to analyze the flow data by both the approaches in this module.

**Alert module**

Based upon the analysis done by the algorithms in the anomaly detector module on each flow record using k-means and outlier detection approach, the alert module declares each flow record as normal or anomalous individually by both the approaches.

## MATERIALS AND METHODS

**To Tools Used For Development, Testing and Analysis:** The implementation of the IDS is done using the following tools.

**Qt Creator (Used for development of IDS):** The Qt Creator was used as the development environment for the IDS on Linux platform. The IDS uses various Qt libraries installed in the Qt Creator for the IDS development. JAVA with Qt support was used to develop the source code of IDS.

**UDP attacker (Developed and used for testing IDS);** In order to flood the IDS with UDP Packets, a tool to inject UDP packets into the IDS starting from slow rate to very fast rate, was developed. It named this tool as ’UDP Attacker’ This tool was developed as a supporting tool to the IDS to test its capability to detect UDP flooding attacks. Microsoft visual studio was used with MFC classes in JAVA for the development of this tool on Microsoft windows platform.

**MATLAB (Used for analysis of IDS performance)**: It has used Matlab for the analysis of the data captured by the IDS. MATLAB was used to do the performance analysis of the k-means and outlier detection approaches.

**Analysis of algorithms**: The behaviour of the traffic as discussed in section 4.2 is taken as the baseline behaviour for further analysis.

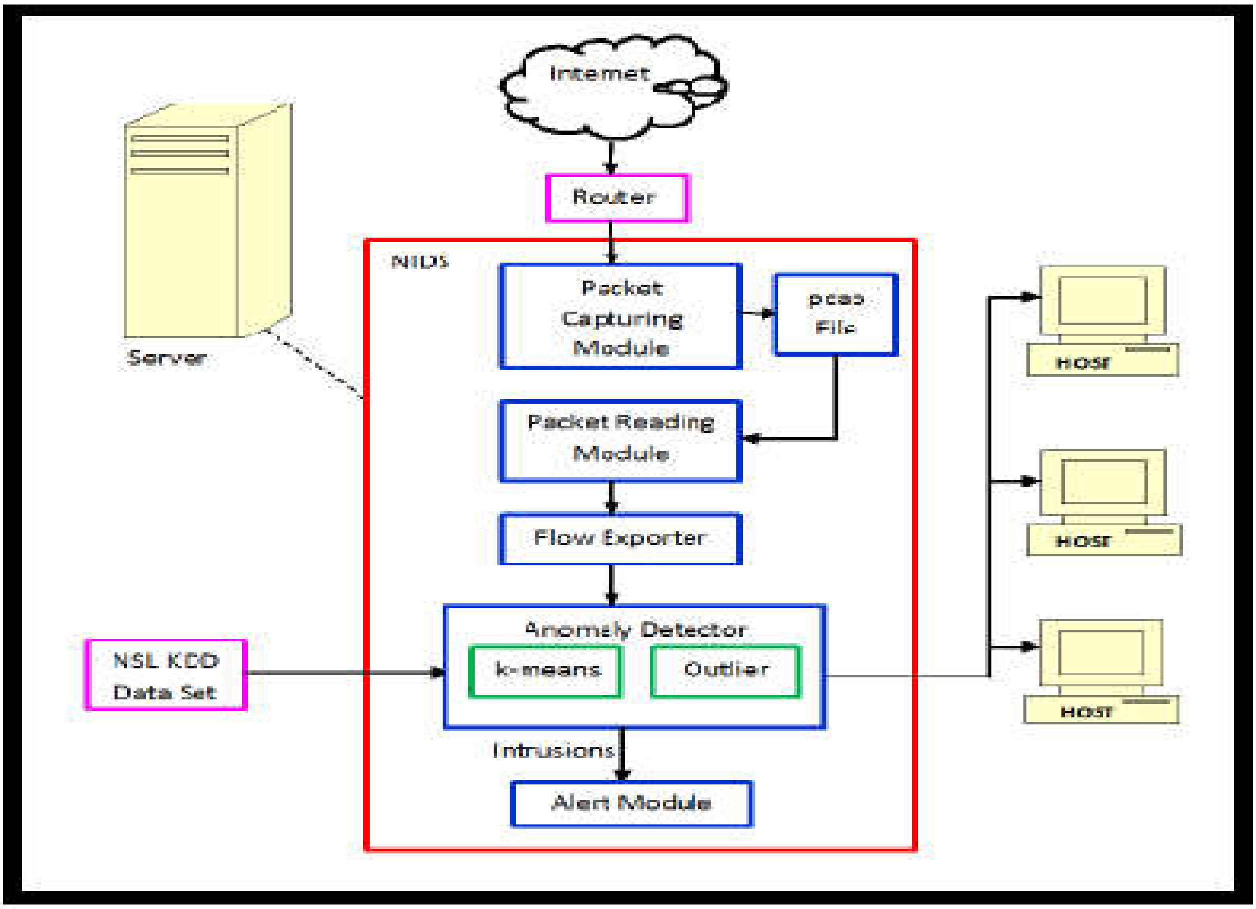


**Figure 1. System architecture of IDS**

The IDS classifies the behaviour of the traffic as normal or anomalous based on these criteria in this section. Once the behaviour is classified as normal or anomalous, the flow records are passed to the k-means and outlier detection methods to reclassify the behaviour of the traffic according to their own algorithms. The outcome of k-means and outlier detection is matched with the traffic behaviour based on the baseline behaviour and IDS declares whether the particular flow record is a false positive, false negative, true positive or true negative for both the approaches. The IDS then generates the parameters required for the performance analysis of the k-means and outlier detection approach. These parameters are as follows (discussed in detail in section 4.6).

- False positive rate (FPR).
- False negative rate (FNR).
- True positive rate (TPR).
- Sensitivity of IDS.
- Specificity of IDS.
- Classification rate of IDS.
- Precision of IDS.

These parameters are compared for both the approaches and the analysis outcomes are presented in results.

## Algorithms Description

### K-Means Approach

In k-means approach, it defines k as the number of clusters. The algorithm is described as follows.

**Step 1:** Given n objects, initialize k cluster centre.
**Step 2:** Assign each object to its closest cluster centre.
**Step 3:** Update the centre for each cluster.
**Step 4:** Repeat steps 2 and 3 until no change in each cluster centre.

The algorithm will assign a cluster to a particular flow record which is termed as objects in this algorithm. Thus each cluster shall contain similar types of flow records. The k-means algorithm implemented in this thesis creates two clusters, normal and anomalous. The algorithm outcome on each flow record is grouped into either normal or anomalous cluster and is declared as normal traffic if it belongs to the normal cluster or anomalous traffic if it belongs to the anomalous cluster.

### Outlier Detection Approach

This method makes use of neighbourhood outlier factor (NOF) which is used to measure the anomalies present in the datasets. This NOF is calculated for each flow record in the dataset and is termed as 'Outlier Score'. If the outlier is within a threshold, then the flow is normal otherwise it is anomalous. Based on the multiple runs of the IDS, it have derived the threshold values of the outliers. The detailed steps in this approach are as below. Here object refers to a single flow record.

**Step 1:** Given n objects, calculate the k-distance to the nearest neighbourhood of each object from every other object.
**Step 2:** Compute reach ability distance for each n objects with all other n-1 objects. The reach ability distance is max {k Distance (n), Distance (n, n-1)}.

**Step 3:** Calculate neighbour reach ability density for each n which is the inverse of the average reach ability distance to the nearest neighbours of n.

**Step 4:** Calculate NOF for all n objects which can be taken as an average of n's neighbour reach ability density and its nearest neighbour's neighbour reach ability density.

### Input Data for Experiment

The process starts with the collection of input data and grouping them into flow records by the IDS. This input data containing real internet traffic is collected at four times with different types of traffic characteristics. This input data was captured from the network setup of few universities connected to each other to share knowledge and is used by students of the universities to access the internet. Each set have captured a smaller set of traffic ranging between 1-2 minutes for each dataset for analysis. Each dataset consists of varying both normal and attack data. This attack data consists of mainly TCP injection, UDP flooding and ICMP flooding attacks. Each dataset consists of the fields of flows as mentioned in Section 3.2. The most important fields of a flow record are the total number of packets in a flow and the total number of bytes in each flow. This combination of attributes of the flow helps in detecting anomalies in the total amount of traffic. Another combination of attributes in the flow to detect the anomalies is the source and destination IP and port pairs which provide input to detect the port scans. Table 3.1 shows the attributes of packets and flows contained in the datasets. The characteristics of the datasets as shown in Table 3.1 are as follows:

**Table 1. Input data attributes**

| Dataset | TCP Packets | UDP Packets | ICMP Packets | Total Flows |
|---|---|---|---|---|
| Dataset 1 | 2499 | 790 | 147 | 2563 |
| Dataset 2 | 1077 | 1073 | 230 | 1290 |
| Dataset 3 | 1592 | 438 | 43 | 1670 |
| Dataset 4 | 7356 | 816 | 154 | 7196 |

**Dataset 1:** This dataset consists of several TCP injection flows where the intention is to create multiple TCP connections between same source and destination IP. This dataset consists of many flows containing less number of packets and bytes per flow and is intended to congest the destination IP. It contains little amount of UDP flooding.

**Dataset 2:** This dataset consists of TCP injection as well as the highest amount of UDP flooding. It also contains a small amount of ICMP flooding attacks. The UDP flooding is another type of attack which consists of those flows where there are a heavy number of packets and bytes from one or more sources to the same destination. The intention is to keep the destination busy and to deviate it from its normal activities. However, the number of TCP injection flows is much more than the UDP or ICMP flooding flows.

**Dataset 3:** This dataset consists of less amount of TCP injection and UDP flooding flows when compared to dataset 1 and dataset 2. However, this dataset also contains TCP flows with less number of packets and bytes from genuine TCP connections which are normal in nature.

**Dataset 4:** This is a bigger dataset which consists of TCP injections, UDP and Flooding flows. The number of TCP injections is highest in this dataset when compared to all other datasets. However, it contains a moderate amount of UDP flooding flows.

**Performance Metrics for Ids:** The IDS implemented here using two different approaches (k-means and outlier detection) is judged by means of the following metrics. These metrics shall be calculated independently for the two approaches.

These metrics are then computed for both the approaches and then the results are compared in chapter 5.

**False positive rate (FPR):** The FPR is defined as the probability by which the IDS output an alert when the behaviour of the traffic is normal. In this case, the IDS incorrectly gives an alert as output. The FPR can be expressed mathematically as

$$FPR = \frac{FP}{Number\ of\ negatives} \tag{1}$$

**False negative rate (FNR):** The FNR is defined as the probability by which the IDS does not outputs an alert when the behaviour of the traffic is anomalous. In this case, the IDS incorrectly do not gives an alert as output. The FNR can be expressed mathematically as

$$FPR = \frac{FP}{Number\ of\ positives} \tag{2}$$

**Sensitivity:** Sensitivity of an IDS is defined as the proportion of normal behaviour in the entire traffic. In other words it is the ratio of correctly detected anomalous flows and total number of anomalous flows. If all anomalous flows are detected then the sensitivity value is 1 which is quite unusual for IDS. Sensitivity is also called as Detection Rate (DR) or the True Positive Rate (TPR). Mathematically it is expressed as,

$$Sensivity = \frac{TP}{TP+FN} \tag{3}$$

**Specificity:** It is defined as the proportion of true negatives from all the negative behaviour. If all flows are detected as normal then the specificity value is 1 which is quite unusual for an IDS where anomalous traffic is present. Specificity is also called as the True Negative Rate (TNR). Mathematically,

$$Specificity = \frac{TN}{TN+FP} \tag{4}$$

**Classification rate (CR) or accuracy:** The CR is the measure to find that how accurate the IDS is in detecting normal or anomalous traffic behaviour. It is defined as the ratio of all those correct instances according to baseline behaviour characteristics of the traffic to all instances.

$$CR = \frac{TN+TP}{TN+FP+TP+FN} \tag{5}$$

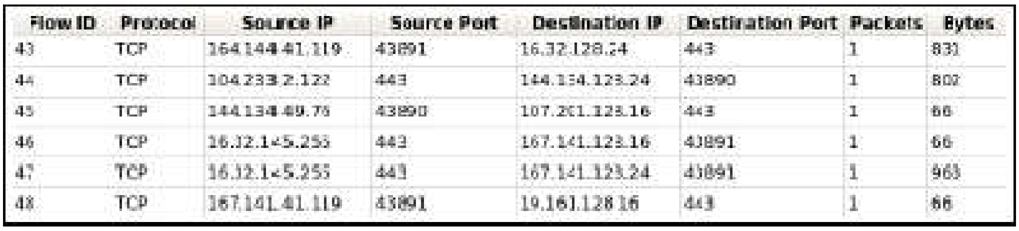
**Precision (PR):** It is the ratio of the number of flows that are detected as normal to those flows that are actually normal.

$$PR = \frac{TP}{TP+FP} \tag{6}$$

## RESULTS AND DISCUSSION

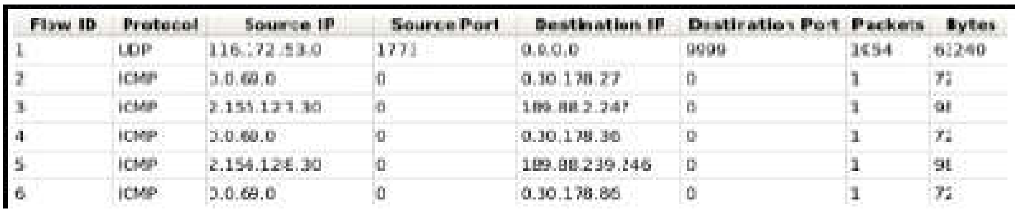
### Experiment I – Features extraction

The found results performed feature extraction from the captured datasets. The IDS extracted the set of flow attributes from the datasets and displayed them in the form of a table.



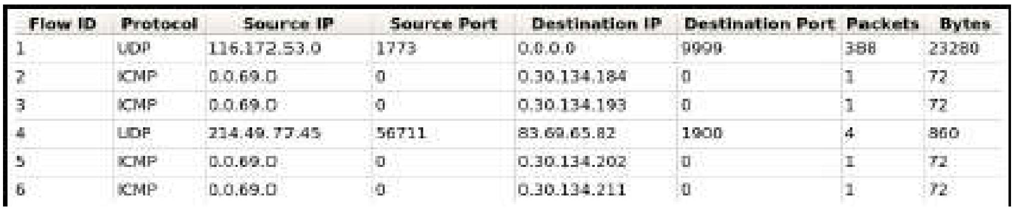
| Flow ID | Protocol | Source IP | Source Port | Destination IP | Destination Port | Packets | Bytes |
|---|---|---|---|---|---|---|---|
| 43 | TCP | 164.144.41.119 | 43891 | 16.32.128.24 | 443 | 1 | 831 |
| 44 | TCP | 104.233.2.122 | 443 | 144.134.123.24 | 43890 | 1 | 802 |
| 45 | TCP | 144.134.49.76 | 43890 | 107.201.123.16 | 443 | 1 | 66 |
| 46 | TCP | 16.32.145.255 | 443 | 167.141.123.16 | 43891 | 1 | 66 |
| 47 | TCP | 16.32.145.255 | 443 | 167.141.123.24 | 43891 | 1 | 963 |
| 48 | TCP | 167.141.41.119 | 43891 | 19.161.128.16 | 443 | 1 | 66 |

Figure 2. Flow records of dataset 1.



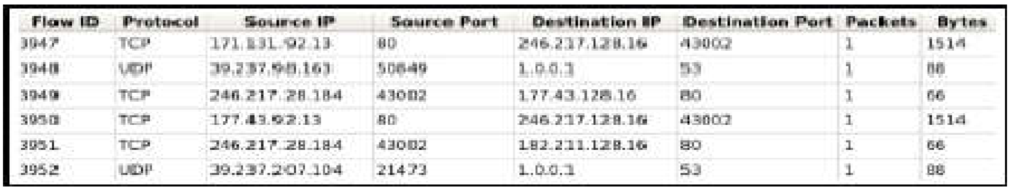
| Flow ID | Protocol | Source IP | Source Port | Destination IP | Destination Port | Packets | Bytes |
|---|---|---|---|---|---|---|---|
| 1 | UDP | 116.172.53.0 | 1771 | 0.0.0.0 | 9999 | 1654 | 62240 |
| 2 | ICMP | 0.0.69.0 | 0 | 0.30.178.27 | 0 | 1 | 72 |
| 3 | ICMP | 2.155.127.30 | 0 | 189.88.2.247 | 0 | 1 | 91 |
| 4 | ICMP | 0.0.69.0 | 0 | 0.30.178.36 | 0 | 1 | 72 |
| 5 | ICMP | 2.154.128.30 | 0 | 189.88.239.246 | 0 | 1 | 91 |
| 6 | ICMP | 0.0.69.0 | 0 | 0.30.178.86 | 0 | 1 | 72 |

Figure 3. Flow records of dataset 2.



| Flow ID | Protocol | Source IP | Source Port | Destination IP | Destination Port | Packets | Bytes |
|---|---|---|---|---|---|---|---|
| 1 | UDP | 116.172.53.0 | 1773 | 0.0.0.0 | 9999 | 388 | 23280 |
| 2 | ICMP | 0.0.69.0 | 0 | 0.30.134.184 | 0 | 1 | 72 |
| 3 | ICMP | 0.0.69.0 | 0 | 0.30.134.193 | 0 | 1 | 72 |
| 4 | UDP | 214.49.77.45 | 56711 | 83.69.65.82 | 1900 | 4 | 860 |
| 5 | ICMP | 0.0.69.0 | 0 | 0.30.134.202 | 0 | 1 | 72 |
| 6 | ICMP | 0.0.69.0 | 0 | 0.30.134.211 | 0 | 1 | 72 |

Figure 4. Flow records of dataset 3.



| Flow ID | Protocol | Source IP | Source Port | Destination IP | Destination Port | Packets | Bytes |
|---|---|---|---|---|---|---|---|
| 3947 | TCP | 171.131.92.13 | 80 | 246.217.128.16 | 43002 | 1 | 1514 |
| 3948 | UDP | 39.237.98.163 | 50849 | 1.0.0.1 | 53 | 1 | 88 |
| 3949 | TCP | 246.217.28.184 | 43082 | 177.43.128.16 | 80 | 1 | 66 |
| 3950 | TCP | 177.43.92.13 | 80 | 246.217.128.16 | 43002 | 1 | 1514 |
| 3951 | TCP | 246.217.28.184 | 43082 | 182.211.128.16 | 80 | 1 | 66 |
| 3952 | UDP | 39.237.207.104 | 21473 | 1.0.0.1 | 53 | 1 | 88 |

Figure 5. Flow records of dataset 4

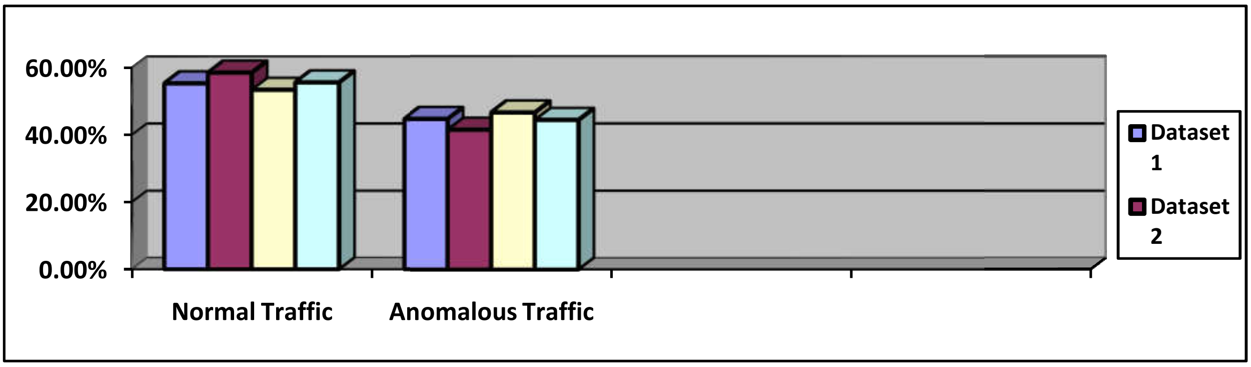


Figure 6. Baseline characteristics results

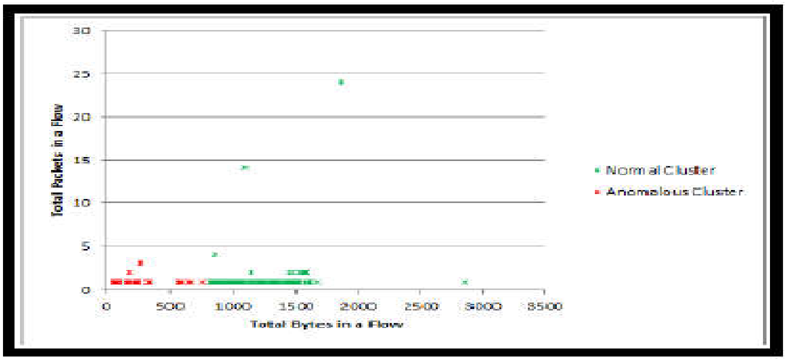


Figure 7. k-means clustering on dataset 1

Each row of this table is a separate flow record having a specific flow ID containing all the packets and bytes transacted between a source IP/port and destination IP/port pair using a specific protocol.

### Experiment II – Normal and Anomalous Characteristics

The IDS learned the characteristics of the normal and anomalous flows when pcap files were fed to it. Using these characteristics and knowledge from NSL-KDD dataset, the flows present in the input datasets were characterized as normal or anomalous. On the analysis of the pcap files based on Experiment II, the IDS learned the following knowledge which was used to categorize the traffic flows as normal or anomalous. This categorization of traffic is termed as the baseline behavior of the traffic and will be used to detect anomalies in the next incoming traffic.

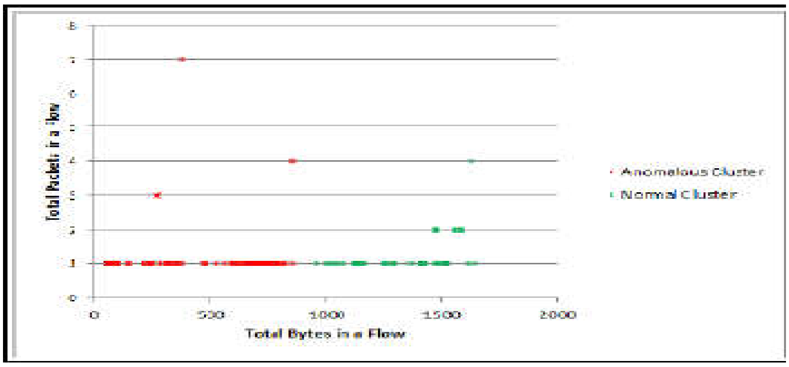


**Figure 8. k-means clustering on dataset 2**

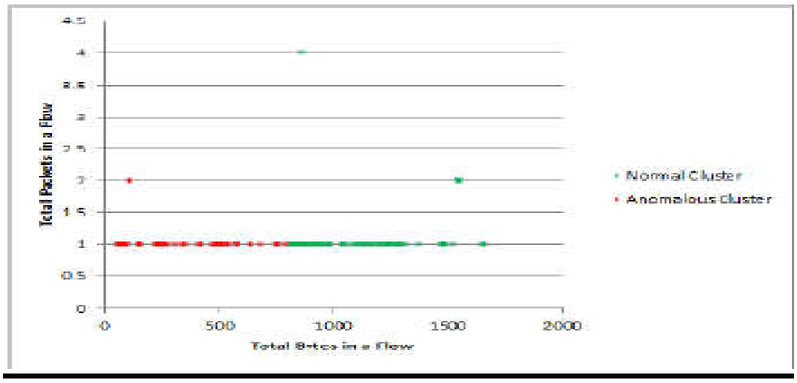


**Figure 9. k-means clustering on dataset 3**

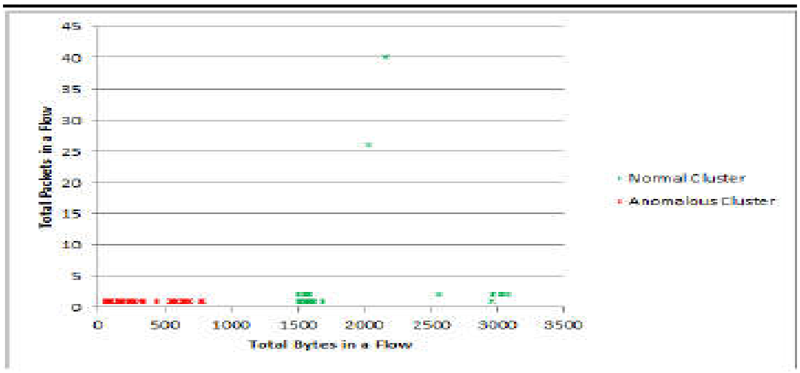


**Figure 10. k-means clustering on dataset 4**

**Average number of packets:** It observed that a normal network flow consists of more number of packets per flow whereas an anomalous flow consists of less number of packets. This indicates more number of TCP connections established between a single source IP and destination IP. In the pcap files analyzed, an anomalous traffic flow generally consists of less than 5 packets (possibly indicating DOS more than at least 100 packets per flow.

**Average size of packets:** It observed that the average size of the normal packets present per flow in the pcap files is more when compared to anomalous packets. There was TCP injection and UDP flooding attacks present in the pcap files and the average size of such packets were between 90 to 150 bytes. The average size of the normal packets present per flow is greater than 500 bytes in the pcap files.

**Average size of flows:** It was observed that anomalous flows are of a very small size as compared to the size of the normal flows. The size of anomalous flow was as small as the size of the packet itself which means that an anomalous flow may contain only a single packet.

**Average number of same source IP and port to different destination IP and port:** it observed that there is more number of different destination IP and destination port from the same source IP and source port using UDP and ICMP protocol. This lead to the generation of more number of flows and higher volume of traffic indicating possible UDP or ICMP flooding or port scanning attack.

**Average number of different source IP and port to same destination IP and port:** The pcap files also contain several entries of the flows where there are several flows which contain different source IP and port destined to same destination IPs and ports.

**Land attack flows:** There are those flows present in the pcap files where many packets were sent from same source IP and port to the same destination IP and port.

**Special feature:** Results noticed in the pcap files that there was some TCP traffic to the ports where communication between unprivileged and privileged port is happening. This type of traffic is considered as anomalous according to the following condition observed. Communication between a host A having IP/Port pair as IPa/Pa and host B having IP/Port pair as IPb/Pb over TCP is considered as anomalous if Pa is an unprivileged port and Pb is a privileged port i.e. Pa >=1024 and Pb < 1024.

### Experiment III – K-Means Evaluation

On the live traffic captured by IDS, the k-means algorithm was run. The input data for the k-means algorithm. k-means algorithm clustered the traffic into normal or anomalous flows. Results shows the percentage of the traffic clustered into normal or anomalous flows for each dataset using k-means. The total number of packets and the total bytes per flow were the attributes of each flow which were used by k-means to cluster the traffic. These figures show the clustered traffic when the total number of packets are plotted against the total number of bytes for each flow in each dataset. The k-means algorithm clusters the flow data into two clusters in the results presented in this section. However, it does not say that which cluster should be treated as normal or anomalous. We have learned this grouping when we imported pcap files into the IDS. We came out with an interesting observation that k-means was clustering all those flows into one cluster which are having a less average number of packets and bytes per flow than those flows which are grouped into another cluster. It, therefore, learned that the cluster in which there are more number of flows having less number of an average of packets and bytes should be considered as anomalous. The other cluster should be considered as normal. The analysis of dataset 3 revealed the same characteristics of the k-means algorithm where it has clustered 70.46% flows into the anomalous cluster and 29.54% to the normal cluster. As compared with the traffic characteristics of table 5.1, the anomalous cluster contains nearly 36% of the normal flows. However, on dataset 4 it was found that k-means was able to cluster 55.55% flows into the anomalous cluster and 44.45% to the normal cluster. k-means has shown improvement in clustering more amount of normal flows into the normal cluster for this dataset. This anomalous cluster contains nearly 11% of the normal flows which is least when compared to other three datasets.

### Experiment IV – Outlier Detection Evaluation

It ran the algorithm of outlier detection using neighborhood outlier factor on the pcap files to compute the value of the NOF in the form of outlier score. We came to know from this experiment that normal flow records are having almost similar NOF whereas the anomalous flow records are having NOF value with bigger difference with the normal flow record densities. It found that maximum number of normal and anomalous traffic flows were categorized as normal or anomalous respectively when the value of NOF <= 1.2 for normal flows and NOF > 1.2 for anomalous flows. The NOF flow record density and its k-nearest neighbor's density.For dataset 1, outlier detection was able to classify 28.06% out of 44.69% of the anomalous flows as anomalous. For normal flows, 71.94% in excess to 55.31% were assigned score <=1.2. In this case, nearly 15% of the normal traffic was given an outlier score greater than 1.2 and were declared as anomalous. For dataset 2, 20.24% out of 41.5% of the anomalous flows were declared as anomalous. For normal flows, 79.76% in excess to 58.5% were assigned score <=1.2. In this case, again nearly 15% of the normal traffic was given an outlier score greater than 1.2 and were declared as anomalous. The same type of figures exists for dataset 3 where 21.92% out of 46.61% of the anomalous flows were declared as anomalous and 78.08% in excess to 53.39% were assigned score <=1.2 and declared normal. Here also, nearly 15% of the anomalous flows were classified as normal. In dataset 4 which contains a large number of TCP injection traffic, outlier detection has performed badly and classified only 12.07% out of 44.4% of the anomalous traffic as anomalous. The remaining was classified as normal.

### Conclusion

It could, therefore, be concluded that k-means was able to cluster those TCP flows as anomalous which exhibited the similar type of behavior in terms of less number of packets and bytes. However, UDP flows containing more number of packets and bytes per flow which could be an indication of UDP flooding, are assigned to the normal cluster by k-means. This happened in the case of dataset 1, 3 and 4 where the total number of packets and bytes was less than 1000 and 50000 respectively for some individual flows with UDP protocol. It can conclude from the results of outlier detection that outlier detection is better in detecting ICMP flooding. However, it was not able to detect TCP injection where the number of bytes per flow is less than 100 which is generally the case. But in the case of TCP flows where the total bytes exceed 100, it was able to classify them as anomalous which is not always true. However, it was also not able to detect UDP flooding as anomalous in any case. The ground truth in the case of outlier detection says that the value of the threshold should be chosen in such away that maximum amount of anomalous traffic is detected. This requires real reverse engineering to find out such a value of threshold which has been assigned by the algorithm on anomalous flows in multiple learning datasets. Generally, in the case of other outlier detection approaches, the outlier scores are between the range of 0.0 to 1.0 for normal traffic. The anomalous flows have higher to much higher values of outlier scores above 1.0. However, it depends entirely upon the variety of anomalous data contained in the datasets which are used to train the IDS. Sometimes, the normal traffic can lie within a score of up to 1.5, which has happened in the case of neighborhood outlier factor in this thesis.

## REFERENCES

Jaiganesh, V., Dr. Sumathi, P. and Vinitha, A. 2014. Classification Algorithms in Intrusion Detection System: A

Survey. *Int.J.Computer Technology & Applications, Vol 4 (5),746-750*

Jaiganesh, V., Mangayarkarasi, S. and Dr. Sumathi, P. 2013. Intrusion Detection Systems: A Survey and Analysis of Classification Techniques. *International Journal of Advanced Research in Computer and Communication Engineering, Vol. 2, Issue 4, April 2013.*

Jalla Hanumantha Rao and Girija P. N. 2014. Distance based transformation for privacy preserving data miningusing hybrid transformation. *Computer Science & Information Technology ,(CS & IT) pp. 15–23, 2014. © CS & IT-CSCP 2014.*

Kalyani G. and Lakshmi A. Jaya, 2012. Performance assessment of different classification techniques for intrusion detection. *IOSR Journal of Computer Engineering (IOSRJCE) ISSN: 2278-0661, ISBN: 2278-8727 Volume 7, Issue 5 (Nov-Dec. 2012), PP 25-29 www.iosrjournals.org*

Maheshwar Kamini and Singh Divakar, 2013. A Review Of Data Mining Based Intrusion Detection Techniques. *International Journal of Application or Innovation in Engineering & Management (IJAIEM) Web Site: www.ijaiem.org Email: editor@ijaiem.org, editorijaiem@gmail.com Volume 2, Issue 2, February 2013 ISSN 2319 – 4847*

Manish Joshi, 2012. Classification, clustering and intrusion detection system. *International Journal of Engineering Research and Applications (IJERA) ISSN: 2248-9622 www.ijera.com Vol. 2, Issue 2,Mar-Apr 2012, pp.961-964.*

Phutane Trupti and Apashabi Pathan, 2014. Survey of Intrusion Detection System Using Different Data Mining Techniques. *International Journal of Innovative Research in Computer and Communication Engineering, (An ISO 3297: 2007 Certified Organization) Vol. 2, Issue 11, November 2014.*

Senthilnayaki Balakrishnan, Venkatalakshmi, K. and Kannan, A. 2014. Intrusion detection system using feature selection and classification technique. *International Journal of Computer Science and Application, (IJCSA) Volume 3 Issue 4, November 2014 www.ij□csa.org*

Zibusiso Dewa and Leandros A. Maglaras, 2016. Data mining and intrusion detection systems. *(IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 7, No. 1, 2016.*

*******