IJCSE International Journal of Computer Sciences and Engineering Open Access

Research Paper Vol.-6, Issue-5, May 2018 E-ISSN: 2347-2693

# Packet-based Anomaly Detection using n-gram Approach

**Kajal Rai[1*], M. Syamala Devi[2], and Ajay Guleria[3]**

[1]Department of Computer Science and Applications, Panjab University, Sec-14, Chandigarh, India
[2]Department of Computer Science and Applications, Panjab University, Sec-14, Chandigarh, India
[3]Computer Center, Panjab University, Sec-14, Chandigarh, India

*Corresponding Author:kajalrai.pu@gmail.com, Phone: +91-9592366282*

**Available online at: www.ijcseonline.org**

Accepted: 21/May/2018, Published: 21/May/20182018

***Abstract*--**Intrusion detection systems monitor computer system events to discover malicious activities in the network. There are two types of intrusion detection systems, namely, signature-based and anomaly-based. Anomaly detection can be either flow-based or packet-based. In the flow-based approach, the system looks at aggregated information of related packets in the form of flow. Packet-based detection system inspects the complete packet which consists of a header as well as payload data. In this paper, a packet-based improved anomaly detection technique is proposed. In the training module, the normal profiles of the network traffic are generated by modeling the payload of the network using n-gram approach by applying length-wise clustering of packets according to payload length. Length-wise clustering is done to reduce the number of models for normal profiles. Then the mean and standard deviation is calculated which are used in detection module. In detection module, the distance between normal profiles and newly arriving data in the network is computed using cosine similarity. The standard dataset DARPA'99 and the Panjab University collected data are used for testing the proposed technique. Anomaly detection of the proposed technique is done on port numbers 21, 23 and 80 and the results are compared with the various n-gram techniques and other techniques used in literature for payload anomaly detection. It is concluded that this improved technique can reduce space and provide better results on port 21 and port 23 than on port 80.

***Keywords*--** Payload, anomaly detection, cosine similarity, n-gram, length-wise clustering

## I. INTRODUCTION

Security is the basic requirement for any type of computer network as any organization small or big uses Internet for their business purposes. Organizations have very important and confidential data which must be secure and should not get hacked by anyone especially by competitive companies. Intrusion Detection Systems (IDSs) are needed for security purpose of the computer systems in the network. IDS monitor system events and issue an alert if it discovers any malicious activity in the system. Misuse-based IDS and Anomaly-based IDS are the two major types of IDSs. Misuse-based IDS make use of several pattern matching techniques to find a match between previously stored attack signatures and coming packets in the network. Anomaly detection IDS analyses traffic patterns and generate profiles of normal traffic in the network. Algorithms for anomaly detection are based on the assumption that the behavior of attack packets is different from that of the normal packets. If the traffic in the network significantly deviates from the normal profiles generated, the anomaly detector classifies the network packet as an attack [1], [11], [14]. Difference between misuse-based and anomaly-based system is that misuse-based can only find known attacks while the anomaly-based system can find known as well as unknown attacks. The main limitation of anomaly-based IDS is that it causes a huge amount of false alarms.

There are two approaches for anomaly detection, namely, flow-based approach and packet-based approach. In the flow-based approach, the anomaly detector considers only the aggregated information of correlated packets in the form of flow. Patterns of network connections are provided by flow-based anomaly detector. While in the packet-based approach, the anomaly detector monitors the complete network packet which consists of a header as well as payload data [2], [15].

Techniques for anomaly detection have one most important requirement that the training data should not contain any anomalies for building a model to detect attacks. But there are certain limitations to this approach. First is that the anomaly free data for training the model is not easily available. Second, if the model is trained on the data which contains some anomalies, the model will treat it as normal data and will not be able to find attacks related to the anomalies present at training time [12].

For our experiments we had taken two datasets, one is the DARPA dataset which is available on-line and second dataset was collected from Panjab University live network.

In the DARPA dataset, we had taken the data which is already attack free and for university dataset anomalies are removed manually using feedback from open source and commercial security tools available in the computer center with the help of the technical staff.

Rest of the paper is organized as follows, Section II contain the related work of payload-based anomaly intrusion detection, Section III explain the methodology of anomaly detection along with proposed algorithm, Section IV describes results and discussion, Section V concludes research work with future directions.

## II. RELATED WORK

Intrusion Detection Systems (IDSs) are those that monitor suspicious activities and issues alarm when undesirable events occur. Host-based IDS (HIDS) and Network-based IDS (NIDS) are the two major types of IDS based on network location. HIDS observes the behavior of a single host and detect suspicious activities against that host. Tripwire is an example of HIDS. NIDS aims to detect unauthorized access to the network resources by supervising network traffic. Most common NIDS are BRO, Suricata, and Snort [13]. Both HIDS and NIDS can be further categorized as Misuse-based IDS and Anomaly-based IDS based on methods used for detection. Misuse-based IDS rely on known attack signatures and use some pattern matching procedures to find attack signatures similar to the packets arrived in the network. Anomaly-based IDS first create normal profiles of system, users, and all the available network connections. Then it monitors newly arrived data in the network, and matches it with already stored profiles and calculates deviations from normal behavior. If there is significant deviation from normal stored profiles, they are termed as anomalies.

The authors in [3] present a payload-based anomaly detector, PAYL, for intrusion detection using n-gram approach. In the context of packet payload, the sequence of n-adjacent bytes in a payload unit is n-gram. A sliding window is used which is of width n and the occurrence of each n-gram is counted by passing this window over the entire payload. Further, the payload bytes are converted into ASCII characters and frequency vector is computed for each length. Frequency vector is the count of each ASCII character in the payload of packet. First the byte frequency distribution of payload is computed and then their mean and standard deviation of the payload flowing to a particular host and specific port is calculated during the training phase and make normal profiles. PAYL use Mahalanobis distance to calculate the distance between new data and the pre-computed profile, during the detection phase. The experiments were done on the DARPA’99 IDS dataset and a live dataset collected on the Columbia Computer Science department network.

Payload Content-based Network Anomaly Detection (PCNAD) was presented in [4]. It was an improvement to PAYL system. The difference between PAYL and PCNAD is that in PAYL the entire payload is considered for profile calculation and for anomaly detection also while PCNAD uses CPP (Content-based Payload Partitioning) technique in which the payload is divided into various partitions depending on the payload content.

SPADE (Statistical Packet Anomaly Detection Engine) [5] is a statistical anomaly detection system that is used as a plug-in for Snort. Snort is an open source NIDS which is used for real-time traffic analysis and logs packets to perform detailed analysis of packets. To compute the anomaly score of a packet, the authors used a simple frequency-based approach. The anomaly score of a packet increases with less number of times a given packet was seen. If the anomaly score of a packet was greater than the predefined threshold, that packet was then sent to a correlation engine that was intended to discover port scans. But this method generates huge number of false alarms because it classifies all unseen packets as attacks whether they are really intrusions or not. Besides machine learning, statistical approach can also be used for anomaly detection such as in SPADE. An important advantage of using statistical approach is that it does not need prior information of attacks and security flaws. But SPADE has very high false alarm rate as it reports all unseen packets as attacks.

The authors in [6] created six groups based on header field of the network packet. Byte frequency distribution of the ASCII characters was computed and assigned the ASCII characters to these six groups depending upon the value of their byte frequency distribution. This type of classification is very rigid classification.

Zhang and White in [7] proposed an approach which is used to find TCP based application level attacks. In this approach, Principle Component Analysis (PCA) was used to extract useful features from network packets. Total nine attributes were selected using PCA, namely, header length, packet length, IP version, source port, destination port, source IP, destination IP, payload size, and payload content. The first word of the first line of the payload was extracted and pairs were constructed with all other associated attributes selected by PCA. Anomaly score was calculated by considering all the possible combinations of each pair.

Multiple-Classifier Payload-based Anomaly Detector (McPAD) [8] combines multiple one-class Support Vector Machine (SVM) classifiers. n-gram approach was used to model the payload of the network packet. The authors proposed a method to select features from payload by using sliding window approach that considers two bytes which are v positions apart in the payload. The experiments in this paper were done on the DARPA’99 dataset and real-time collected HTTP traffic. This detector was also able to find polymorphic blending attacks.

In [9], the authors used Linear Discriminant Analysis (LDA) and Mahalanobis distance map for feature selection. Euclidean distance was used to find the distance between normal profiles and newly coming data packet. Threshold was calculated by computing Euclidean distance between each normal training sample and the mean value of feature vector.

In [10], the authors proposed Text-Mining-based Anomaly Detection (TMAD) model based on PCA for dimensionality reduction. HTTP traffic of DARPA'99 dataset was used for experimentation. 1-gram approach was used to convert the payload into feature vectors. Weighted payload feature matrix was constructed by computing the weight of every single feature in the payload. The weight of each feature was calculated by using term frequency and inverse document frequency. Then from this matrix, the optimal number of features was selected by using the Guttman-Kaiser criterion by utilizing PCA. Mahalanobis distance was used to distinguish between normal and attack packets based on the selected features.

When we model the payload according to payload length, the number of models becomes very high which causes serious memory issues. To overcome this problem, we have used length-wise clustering of payload data before generating the models. This length-wise clustering uses packet header information first such as ports and IPs and payload length to cluster the packets of similar port and which are part of given network connection . Then it divides the packets into groups of 10 depending upon the length of the payload. For ex: packets on port 80 and having source and destination IP 172.16.128.240 and 151.32.168.20 respectively and having length 1-10 are in group1, length 11-20 are in group 2 and so on. After clustering, the normal profiles of similar connection are calculated using algorithm 1 for each group.

Algorithm 1 is based on the technique used by PAYL. The main difference between PAYL and our algorithm is that we have used length-wise clustering while in PAYL models are merged by continuously finding distance between nearby models whose payload length is similar and if the distance is very less than the nearby models are merged. By doing length-wise clustering of packets of nearby payload length, the time to merge the models after constructing them is reduced. Another difference is instead of using Mahalanobis distance for finding deviations from normal profiles, we have used Cosine Similarity as it is highly efficient for sparse matrix. Hence, even if payload contains missing values for some ASCII characters which are not available in payload, cosine similarity gives efficient results.

## III. PROPOSED TECHNIQUE FOR PAYLOAD BASED ANOMALY DETECTION

Our anomaly detection system has two modules, namely, the training module and the detection module. In the training module, the system generates normal profiles of the system and user behavior in the network using their known normal behaviors. In the testing module, the system matches the stored normal profiles with the new incoming data. If the system finds the significant deviation between them, it issues an alert message that the incoming packet is possibly an attack.

### III.I Training Module

Packet payload is the large piece of data in the form of bytes. In order to model the payload, we had fragmented this stream of bytes into sets based on port number, IPs, and payload length. The port numbers ranges from 0-65535 used by the transport layer protocols for host-to-host connectivity. The Internet Assigned Numbers Authority (IANA) had categorized port numbers into three groups.

(i) **Well-Known Ports:** Port numbers from 0-1023 are known as well-known ports. These ports are used mainly in a client-server application as destination ports. Each port is assigned a fixed application, common across networks and connections. For example, port 80 is reserved for HTTP connections, port 23 is for TELNET, and port 20 is reserved as an FTP control channel.

(ii) **Registered Ports:** Ports numbers from 1024-49151 are known as registered ports. These ports are assigned for specific services. For ex: port 6667 listens for IRC chat.

(iii) **Private Ports:** The port numbers from 49152 to 65535 is used for private or temporary services and cannot be assigned by IANA.

Network services have fixed pre-assigned port numbers for particular applications. Within one port the payload length also varies such as for TCP packets for port 80, the payload length ranges from 0-1460. To divide the payload into manageable groups, the stream of bytes is divided based on some useful attributes such as port numbers, payload length, and IP addresses. Normal profiles are built using TCP protocol on specific port for each source and destination IP and for each payload length. The 1-gram models are computed on a specific port for each payload length between each connection. But if we generate model $P_k$, on port j for each given length of payload between all the given network connections, the number of models becomes very high which in turn causes memory issues. To reduce the number of models, we have used length-wise clustering where the payload length for a particular port and between a given connection is divided into groups of 10 each from 1-1460. By using length-wise clustering, the number of models for any given port and connection reduces by approximately 10%. After this clustering, payload data is converted into ASCII characters and frequency vector is calculated for all length groups. Frequency vector is the frequency of each ASCII character in the payload of the packet. Then for all frequency vectors of same length group, mean and standard deviations are also computed. This mean and standard deviation of all frequency vectors for each payload length group, on the specific port and for each network connection are used as the model for given payload length group. Steps of generating normal profiles are given in algorithm 1.

**Algorithm 1:** Steps for generating normal profiles

```
Begin
while Not end of captured packets do
    Extracts the packet based on protocol type (TCP, UDP), destination port, payload length, source IP and destination IP address.
    Categorize packets using length wise clustering into groups according to payload length.
    Convert all packet contents of particular length group in its ASCII character.
    Calculate the byte frequency of each ASCII character for each packet of particular length group.
    Store the ASCII byte frequency distribution in a csv file.
    Read the ASCII csv file and calculate the mean and standard deviation of given payload length and store them in a database.
end while
End
```

### III.II Detection Module

In the detection module, the first step is to convert the payload of the incoming packet into its frequency vector and this is done by using algorithm 1 except the mean and standard deviation is not calculated for the newly coming packet. Then this calculated byte value distribution is compared against the pre-computed model. If the distance between the stored model and the payload distribution of the coming network packet is greater than the preset threshold then the packet is labeled as an attack packet and an alert message is generated. Cosine Similarity is used for calculating distance. We had used Cosine Similarity for our purpose as it is highly efficient for high-dimensional spaces. As ASCII character set has 256 characters and we are using 1-gram model thus, we have 256 dimensionality feature vector for each payload. Cosine Similarity is calculated as follows:

$$Cosine\ Similarity = \cos(\theta)$$

$$= \frac{X.Y.}{||X||.||Y||} \qquad (1)$$

where, X and Y are the components of vector.

### III.III Threshold for calculating distance

We find the threshold for differentiating normal and anomalous packets. We set different thresholds for each port and for each length group as payload data length and the content of the payload varies from port to port. First, we find the source and destination IP where the maximum number of packets transfers of a specific length group in the training data. Then, we find the mean and standard deviation for given payload length group by using algorithm 1. Then given formula is used to calculate the threshold.

$$Th_i = \mu_i \pm C\ \sigma_i \qquad (2)$$

where, $\mu_i$ and $\sigma_i$ are the mean and standard deviation of a given payload length group and $Th_i$ is the threshold for that group.

A number of experiments are done to find the value of C. The value of C is decided by varying its value and the optimum value is selected which maintains the balance between maximum detection rate and least false positive rate for each port.

## IV. IMPLEMENTATION AND TESTING OF THE PROPOSED TECHNIQUE

Algorithm 1 is implemented on a 64-bit Windows 10 operating system, with 8 GB of RAM and an Intel(R) i3 processor with CPU speed of 3.70GHz using Java and MATLAB. The experiments are done for performance comparison of different payload-based anomaly detection algorithm and the proposed algorithm. The testing of propose technique has been done on two datasets, namely, DARPA dataset and Panjab University collected dataset.

### IV.I DATASET USED

**(i) DARPA Dataset:** DARPA dataset is a well-known standard dataset for intrusion detection. DARPA'99 consists of three weeks of training data and two weeks of testing data. Each week contains data from Monday to

Friday. Week-1 and week-3 consist of attack-free data and week-2 consists of attack data. We trained our anomaly detector on attack-free data from both weeks and testing is done on week-4 and week-5 and also on trained data using 10-fold cross-validation technique. We only examined the inbound traffic from the hosts 172.16.xxx.xxx for ports 21, 23 and 80 so that we can compare our results from the work done by various researchers. Performance parameters which are used for our experiments are accuracy, detection rate, and false positive rate. Accuracy determines the total percentage of correctly classified examples; Detection Rate (DR) is the ratio of the correctly discovered attacks and the total number of attacks present in the network and False Positive Rate (FPR) is the ratio of the number of normal data packet recognised as attacks and the total number of normal packet instances.

Table 1 shows the accuracy rate of anomaly detection on DARPA'99 using payload on TCP protocol and destination port 80 which is Hyper Text Transfer Protocol (HTTP) port using 10-fold cross-validation technique.

**Table 1: Accuracy on port 80 using 10-fold cross-validation on DARPA dataset**

| Payload Length Range (in Bytes) | No. of data packets | Accuracy (in %) |
|---|---|---|
| 261-270 | 1400 | 93.88 |
| 491-500 | 206 | 80 |
| 611-620 | 14669 | 100 |
| 1021-1030 | 11708 | 94.31 |
| 1451-1460 | 11900 | 96.06 |

From the above table, it is clear that the accuracy of the model increases with increase in the number of instances used to train the model.

**Table 2: Comparison of proposed algorithm with various techniques**

| Port | Reference | Techniques used | DR (in %) | FPR (in %) |
|---|---|---|---|---|
| 21 | [3] | n-gram, Mahalanobis Distance | 15 | 0.1 |
| | [4] | CPP, Manhattan distance | 96.43 | 0.1114 |
| | [7] | PCA, payload keywords | 27.7 | 0.1 |
| | **Proposed algorithm** | **n-gram, Cosine Similarity** | **51** | **0.19** |
| 23 | [3] | n-gram, Mahalanobis Distance | 12 | 0.4 |
| | [7] | PCA, payload keywords | 22.5 | 0.4 |
| | **Proposed algorithm** | **n-gram, Cosine Similarity** | **20** | **0.32** |
| 80 | [3] | n-gram, Mahalanobis Distance | 100 | 0.3 |
| | [4] | CPP, Manhattan distance | 97.06 | 0.17 |
| | [7] | PCA, payload keywords | 19.6 | 0.3 |
| | [8] | PCA, SVM | 95 | $10^{-5}$ |
| | [9] | LDA, Euclidean Distance | 100 | 3.7 |
| | [10] | PCA, Guttman-Kaiser Criterion, Mahalanobis Distance Map algorithm | 99 | 0.1 |
| | **Proposed algorithm** | **n-gram, Cosine Similarity** | **90** | **0.35** |

### IV.II Performance Evaluation

We have also tested the proposed technique on test data of DARPA and table 2 and figure 1, 2 and 3 shows the comparison of performance of the proposed technique with various other techniques. Criteria for performance measurement are detection rate and false alarm rate. From table 2, it is clear that proposed technique for anomaly detection achieves better detection rate on port 21 and 23 than on port 80 with a less false positive rate. The results of the algorithm on various ports vary due to the training data available for each port and also on the threshold which was set different for each port depending upon mean and standard deviation of particular payload length group packets available on given port number. The accuracy of the proposed algorithm increases with the number of training data instances.

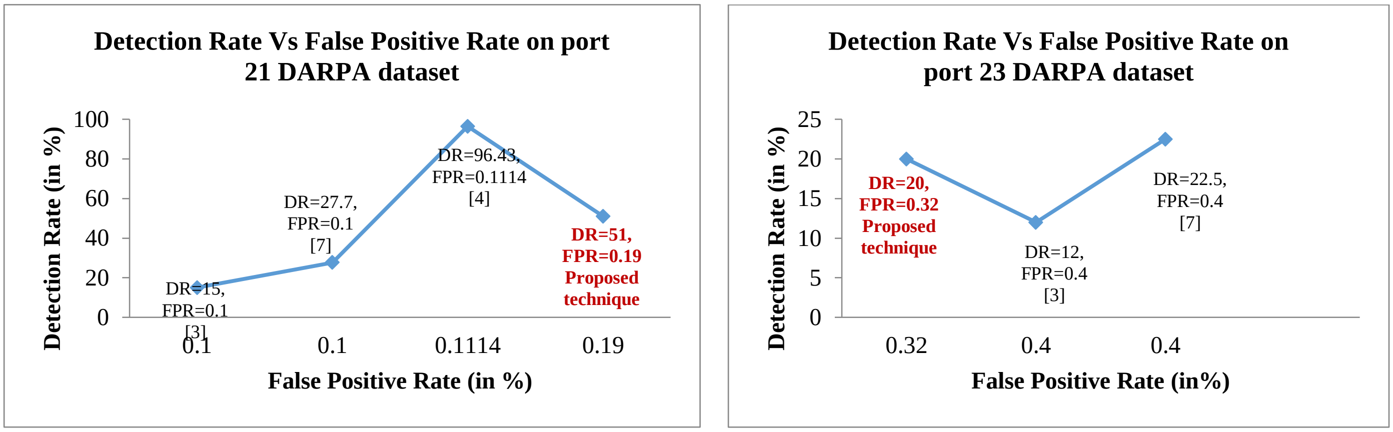

**Figure** Error! No text of specified style in document.**1: Comparison of proposed technique with other techniques on port 21 and port 23**

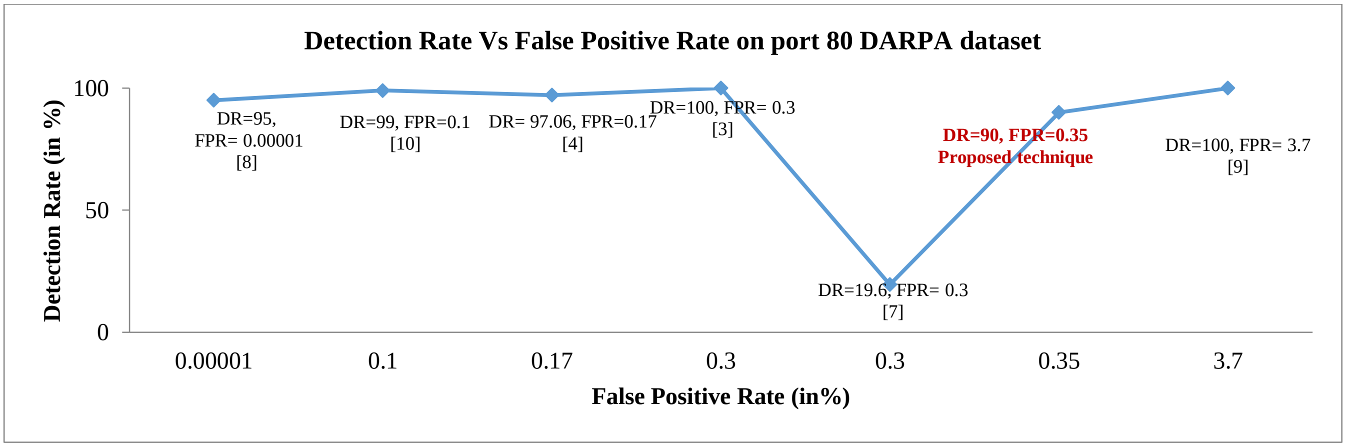

**Figure 3: Comparison of proposed technique results with other techniques on port 80**

**(ii) Panjab University Dataset:** Testing of proposed algorithm is also done on Panjab University dataset. For this purpose Wireshark is used to collect 3 hours of real-time data which consists of LAN, and Wi-Fi data. The network of a university department is taken into consideration and the network traffic on two ports, port 80 (HTTP) and port 443 (HTTPS) are used for training and testing. There are total 124 hosts in this network. Table 3 shows the result of testing on university dataset using 10-fold cross-validation. It is clear from table 3 that accuracy of anomaly detector increases as the number of training instances increases.

**Table 3: Results on Panjab University dataset for port 80 and 443**

| Port | Payload Length Range (in Bytes) | No. of instances | Accuracy (in %) |
|---|---|---|---|
| **80** | 1371-1380 | 63390 | 98.2 |
| | 1381-1390 | 771 | 94 |
| | 1281-1290 | 317 | 93 |
| | 1441-1450 | 202 | 80.1 |
| **443** | 1371-1380 | 44766 | 97 |
| | 1381-1390 | 1096 | 95 |
| | 31-40 | 676 | 93.76 |

## V. CONCLUSION

In this paper, a system which can detect anomalous packets by taking into consideration the contents of the packets is proposed. The proposed system models payload statically using mean and standard deviation of payload features and language independent n-gram approach. The experimental results show that this method is better in attack detection on port 21 and 23 from PAYL and also on port 80 it gives good results and is effective to detect attacks against web application through the analysis of HTTP payloads using cosine distance. In this work, the technique is evaluated on DARPA’99 dataset as well as on university dataset. Presently, the system is showing less accurate results on university data on port 80 for payload length above 1440 due to less training data of that length. However, the system is proficient in discriminating between normal patterns and attack patterns on DARPA as well as university dataset. As future work, it is planned to collect more data on port 80 for length above 1440 and then train and test the system to achieve higher accuracy. Also, it is planned to train and test the effectiveness of the technique for other ports such as 25, and 53 and to extend the work to distributed network environments also.

## ACKNOWLEDGMENTS

This study was supported by the Panjab University, Chandigarh. The authors gratefully acknowledge the technical staff of APJ Abdul Kalam Computer Center, Panjab University, Chandigarh for their help in data collection and labeling of the same.

## REFERENCES

[1] N. M. Jacob, and M. Y. Wanjala, "A Review of Intrusion Detection Systems", International Journal of Computer Science and Information Technology Research, Vol. 5, Issue 4, pp. 1-5, 2017.

[2] H. Alaidaros, M. Mahmuddin, and A. Mazari, "An Overview of Flow-based and Packet-based Intrusion Detection Performance in High Speed Networks", Naif Arab University for Security Sciences, pp. 1–9, 2011.

[3] K. Wang, J.S. Stolfo, "Anomalous Payload-based Network Intrusion Detection", International Workshop on Recent Advances in Intrusion Detection, Springer, Berlin, Heidelberg, Vol. 3224, pp. 203-222, 2004.

[4] S.A. Thorat, A. K. Khandelwal, B. Bruhadeshwar, and K. Kishore, "Payload Content based Network Anomaly Detection", In the Proceedings of the 2008 International conference on the Applications of Digital Information and Web Technologies, IEEE, pp. 127-132, 2008.

[5] S. Staniford, J.A. Hoagland, J.M. McAlerney, "PracticalAutomated Detection of Stealthy Portscans", Journal of Computer Security, Vol.10, pp. 105-136, 2002.

[6] C. Krugel, T. Toth, and E. Kirda, "Service Specific Anomaly Detection for Network Intrusion Detection", In the Proceedings of the 2002 ACM symposium on Applied computing, pp. 201-208, 2002.

[7] L. Zhang, and G.B. White, "Anomaly Detection forApplication Level Network Attacks Using Payload Keywords", In the Proceedings of IEEE Symposium on Computational Intelligence in Security and Defense Applications, CISDA, pp.178-185, 2007.

[8] R. Perdisci, D. Ariu, P. Fogla, G. Giacinto, and W. Lee, "McPAD : A Multiple Classifier System for Accurate Payload-based Anomaly Detection," Elsevier Science Journal of Computer. Networks, Vol. 5, Issue. 6, pp. 864–881, 2009.

[9] Z. Tan, A. Jamdagni, X. He, and P. Nanda, "Network Intrusion Detection based on LDA for Payload Feature Selection", in Proc. of IEEE Globecom Workshops, pp. 1545–1549, 2010.

[10] M. Kakavand, N. Mustapha, A. Mustapha, and M.T.Abdulla, "Effective Dimensionality Reduction ofPayload- Based Anomaly Detection in TMAD Model for HTTP Payload", Transactions on Internet and Information Systems, Vol. 10, Issue. 8, pp. 3884-3910,2016.

[11] G. Kim, S. Lee, and S. Kim, "A Novel Hybrid Intrusion Detection Method Integrating Anomaly Detection with Misuse Detection", Expert Systems with Applications,Elsevier, Vol. 41, Issue 2, pp. 1690-1700, 2014.

[12] E. Eskin, "Anomaly Detection over Noisy Data UsingLearned Probability Distributions", in Proceedings ofThe International Conference on Machine Learning, pp.255-262, Czech Republic, Aug 2000.

[13] K. Scarfone, and P. Mell, "Guide to Intrusion Detectionand Prevention Systems (IDPS)", Technical report NISTSpecial Publication Vol. 800, Issue 94, Feb. 2007.

[14] P. Rutravigneshwaran, "A Study of Intrusion Detection System using Efficient Data Mining Techniques", International Journal Science Research in Network Security and Communication, Vol. 5, Issue 6, pp.5-8, December 2017.

[15] M. Shivakumar, R. Subalakshmi , S. Shanthakumari and S.John Joseph, "Architecture for Network-Intrusion Detection and Response in open Networks using Analyzer Mobile Agents", International Journal Science Research in Network Security and Communication, Vol. 1, Issue 4, pp. 1-7, Oct 2013.

### Authors Profile

**Kajal Rai** is a research scholar in the department of Computer Science and Applications, Panjab University, Chandigarh. She has done Masters of Computer Application from R.G.P.V., Bhopal and Bachelors of Computer Application from Jiwaji University, Gwalior. She is doing her research in the areas of Multi Agent Systems and Network Security. She has the experience of teaching in various colleges of Chandigarh and Amritsar.

**M. Syamala Devi** is a professor in the department of Computer Science & Applications, Panjab University, Chandigarh. She received her Ph.D. degree in Computer Science and Systems Engineering from Andhra University, Visakhapatnam and M.E. in Computer Science and Engineering from NIT, Allahabad. She has completed M.Sc. in Applied Mathematics from Andhra University, Visakhapatnam with first position in the University. Before joining Panjab University, she served Indian Space Research Organization, Sriharikota, and national Institute of Technical Teachers Training and Research, Chandigarh. Her areas of expertise include algorithms, Image Processing, Distributed Artificial Intelligence and Educational Computing.

**Ajay Guleria** is a system manager in Computer Center, Panjab University, Chandigarh. He received Ph.D. in computer science and engineering from National Institute of Technology, Hamirpur, India. His research interests span the areas of vehicular Adhoc networks, network security, wireless sensor networks and data mining. He has 24 years of experience in IT Infrastructure Management and Teaching in Panjab University, Chandigarh. And National Institute of Technology, Hamirpur, (H.P.).