A Review of Intrusion Detection Datasets and Techniques

Sadhana Patidar

M.Tech. Scholar

Department of CSE

Radharaman Institute of Technology
& Science, Bhopal, India
sadhanapatidar58@gmail.com

Priyanka Parihar

Assistant Professor

Department of CSE

Radharaman Institute of Technology
& Science, Bhopal, India
priyanka.parihar28113@gmail.com

Chetan Agrawal

Assistant Professor

Department of CSE

Radharaman Institute of Technology
& Science, Bhopal, India
chetan.agrawal12@gmail.com

Abstract—As network applications grow rapidly, network security mechanisms require more attention to improve speed and accuracy. The evolving nature of new types of intrusion poses a serious threat to network security: although many network securities tools have been developed, the rapid growth of intrusive activities is still a serious problem. Intrusion detection systems (IDS) are used to detect intrusive network activity. In order to prevent and detect the unauthorized access of any computer is a concern of Computer security. Hence computer security provides a measure of the level associated with Prevention and Detection which facilitate to avoid suspicious users. Deep learning have been widely used in recent years to improve intrusion detection in networks. These techniques allow the automatic detection of network traffic anomalies. This paper presents literature review on intrusion detection techniques.

Keywords: Anomaly, Intrusion Detection System, Supervised, Unsupervised, Web Security

I. INTRODUCTION

Now days, the evolution of internet and use of computer systems has resulted in huge electronic transformation of data which experienced multiple problems such security, privacy and confidentiality of information. A significant progress has been made in term of improving computer systems security. However, security, privacy and confidentiality of electronic systems are potentially major problems in computer systems. In fact, no system currently available in the world is 100% secure. In addition, we always can notice that there are huge Attack scenarios. Basically, if a new signature is found on the database of signatures, then the behavior will be considered as an attack [1, 2].

And, it can be exploited by either non authorized or authorized users. Among these tools is the intrusion detection systems (IDS) which allow us to monitor a range of computer systems: an information system, a network or a cloud computing. These IDS detect intrusions and defined as attempts to break the security objectives such as confidentiality, integrity and availability and non-repudiation. We will include the different approaches currently proposed by others on IDS system, network and cloud computing based vulnerabilities in most computer systems. And, it can be exploited by either non authorized or authorized users.

Table 1: Attack types with description

Attacks Category	Description	TCP/IP Layer
DoS	Denial-of-service (fake address generate)	Application Layer
DoS	Denial-of-service (fake address generate)	Transport Layer
U2R	Unauthorized access to local super user (root) privileges	Application Layer
R2L	Unauthorized access from a remote machine	Application Layer
R2L	Unauthorized access from a remote machine	Transport Layer
Probe	Surveillance and other probing	Application Layer
Probe	Surveillance and other probing	Transport Layer

DoS	Denial-of-service (fake address generate)	Transport Layer
-----	--	--------------------

It is based on the comparison between the observed behavior and corresponding reference signatures or known each signature describes a very specific attack and each attack can be detected by one or a sequence of events obtained by one or more sensors, collection of information. This approach is used to classify attacks into: attacks that can come from either a host (e.g., audit records, track of command execution, etc.) or a network. This means that their signatures exist in the database, and the databases are frequently updated in order to increase their effectiveness of detections.

In general, IDS generate an alert if there is a deviation between normal and observed behavior [3]. The basic idea of the approach is that to detects if a user has an abnormal behavior when comparing his/her usual uses. Using the profile generated from past events and compared it to the current collector profile [6]. However, this approach can give many false alarms as it might not be able to detect some attacks.

II. RELATED WORK

Sufyan T. Faraj et al. [2] proposed the intrusion detection model using BPANN for classification of anomalous network traffic from normal traffic and achieved the accuracy of about 93%. Anomaly detection system based on back-propagation Multi Layer Perceptron (MLP) to identify normal users' profile was proposed by Ryan et al. [3]. Their MLP model evaluates the users' commands for possible intrusions at the end of each log session. The top 100 important commands used by the user throughout the session was used to determine the user's behavior. They used a 3 layer MLP model with two hidden layers and found that their MLP model was able to correctly identify 22 cases out of 24.

Similarly, a method primarily based intrusion detection approach that gives the flexibility to generalize from previously determined behavior to acknowledge future unseen behavior was proposed by Ghosh et al. [4]. Their framework employs artificial neural networks (ANNs) and may be used for each anomaly findon so as to find novel attacks and misuse detection so as to detect best-known attacks and their variations.

Meng et al. [8] compared ANN, SVM and DT schemes for anomaly detection in a uniform environment and concluded that J48 algorithm of DT gives better

performance than the other two schemes. The detection rate of low frequent attack types (U2R, R2L) was also high.

Sumaiya Thaseen Ikram et al. [9] proposed an intrusion detection model using chi-square feature selection and multi class support vector machine (SVM). A parameter tuning technique is adopted for optimization of Radial Basis Function kernel parameter namely gamma represented by ' γ ' and over fitting constant ' C '. These are the two important parameters required for the SVM model. The main idea behind this model is to construct a multi class SVM which has not been adopted for IDS so far to decrease the training and testing time and increase the individual classification accuracy of the network attacks.

Manjula et al. [10] proposed a classification and predictive models for intrusion detection which is built by using machine learning classification algorithms namely Logistic Regression, Gaussian Naive Bayes, Support Vector Machine and Random Forest. An experimental result shows that Random Forest Classifier out performs the other methods in identifying whether the data traffic is normal or an attack.

Saad Mohamed et al. [11] presented a hybrid approach to anomaly detection using of K-means clustering and Sequential Minimal Optimization (SMO) classification.

Ibrahim et al. [12] likewise applied a multi-level model with different machine learning techniques, such as C5, MLP, and Naïve Bayes. The study used one of the techniques at each level to classify one category, thereby confirming that multilevel techniques exhibit higher detection accuracy than a single technique

To reduce the false alarm rate of anomaly-based IDS, many machine learning techniques, including support vector machine (SVM) Feng et al. [13] applied extreme learning machine (ELM) along with models combining several techniques. Each model offers specific strengths and weaknesses, with overall generic detection rates steadily increasing. SVMs exhibit good detection performance with IDSs in terms of classifying the flow of a network into normal or abnormal behaviors.

Deshpande et al. [14] proposed classification and predictive models for intrusion detection are built by using machine learning classification algorithms namely Random Forest. The work is performed in divided into two stages. In the first stage data is normalized using

mean normalization. In second stage genetic algorithm is used to reduce number of features and further multilevel ensemble classifier is used for classification of data into different attack groups.

Kuang et al. [15] proposed an IDS based on a combination of the SVM model with kernel principal component analysis (KPCA) and genetic algorithm (GA). KPCA was used to reduce the dimensions of feature vectors, whereas GA was employed to optimize the SVM parameters. The average detection rate was 95.26%, whereas the average false alarm rate was 1.03%. ELMs exhibit performance comparable with that of SVMs in terms of classifying instances of IDS.

Gogoi, Bhattacharyya et al. [16] proposed a multi-level hybrid IDS using a combination of supervised, unsupervised, and outlier methods. This system was evaluated with three datasets, namely, real-time flow dataset, DDoS dataset, and the KDD Cup 1999 with NSL-KDD datasets. The system performance was good with a false alarm rate of 3.4% with the corrected KDD Cup 1999 dataset.

Wathiq Laftah Al-Yaseen et al. [17] proposes a multi-level hybrid intrusion detection model that uses support vector machine and extreme learning machine to improve the efficiency of detecting known and unknown attacks. A modified K-means algorithm is also proposed to build a high-quality training dataset that contributes significantly to improving the performance of classifiers. The modified K-means is used to build new small training datasets representing the entire original training dataset, significantly reduce the training time of classifiers, and improve the performance of intrusion detection system. The popular KDD Cup 1999 dataset is used to evaluate the proposed model. Compared with other methods based on the same dataset, the proposed model shows high efficiency in attack detection, and its accuracy (95.75%) is the best performance thus far.

III. MEASURABLE CHARACTERISTICS OF IDSS

Characteristics of IDSs can be measured quantitatively. Some of these characteristics are:

A. Coverage

Evaluating the detection of intrusion detection systems is a difficult task with many consequences. The range of any intrusion detection system depends on the attacks that IDS can detect under ideal conditions. The number

of dimensions that make up each attack makes assessment difficult. Each attack has a specific goal and works against certain software.

Attacks can also target a specific version of a protocol or a specific operating mode. Several websites may find some attacks more significant than others, which has a significant impact on the evaluation. For example, e-commerce websites may be very interested in finding distributed denial of service attacks, while military websites may pay close attention to surveillance attacks.

B. Probability of False Alarms

A false alarm is a warning caused by normal harmless background traffic. The probability of false alarms determines the percentage of false alarms generated by an IDS in a given environment during a certain period of time. Measuring false alarms can be difficult because an IDS can have different percentages of false alarms in different network environments. In addition, the various aspects associated with host activity and network traffic can make it difficult to determine which aspects cause false alarms.

In addition, configurable IDS that can be set to reduce the rate of false alarms make it difficult to determine the correct configuration of an IDS for a particular false alarm test. A noteworthy point is that there is a school of thought in the field of intrusion detection which believes that there are no false alarms. Each alarm is assumed to contain information in a well-designed system. For example, you can see some packages that look like a test for vulnerable systems. The administrator may want to know, even if it's not yet a problem and isn't actually the beginning of an attack. In this diagram, the system only reports alarms for important events for administrators, which significantly reduces the number of false alarms.

C. Probability of Detection

This measure, also known as the success rate, determines the frequency of attacks that have been correctly identified by an IDS in a given environment for a period of time. The number of attacks used in the IDS test largely determines the result of this measurement. Since the probability of detection is linked to the percentage of false alarms, we can repeat what has already been said about the configurable IDs and conclude that it is difficult to find the right configuration for a specific success rate test.

IDS ability to detect attacks is tied to its ability to identify attacks by marking them or assigning them to known categories. The probability of detection and the probability of false alarms play the most important role in the evaluation of intrusion detection algorithms. Different methods are then used to visually show how a given IDS behaves in relation to these two measures.

One of the most used methods is the operating characteristic curve of the receiver or ROC curve. The ROC curve is a graph of the probability of detection relating to the probability of false alarms. This can be achieved by varying the detection thresholds and maintaining a range of values. The x axis of the ROC graph shows the percentage of false alarms generated during a test, while the y axis shows the percentage of attacks detected for a certain percentage of false alarms.

D. Ability to Handle Stressful Network Conditions

This property shows how an IDS works when there is a lot of traffic. Attackers can send large amounts of data beyond the processing capacity of the host's network or intrusion detection system. Most IDSs should eliminate packets as traffic increases, which can lead to some attacks on deleted packets disappearing. It is up to the evaluation team to determine the threshold at which the performance of IDS and the monitored system significantly decreases [15].

E. Ability to Detect Novel Attacks

This feature shows how much an IDS is able to detect attacks that have not yet taken place. It goes without saying that this measure applies to intrusion detection systems designed to detect unknown attacks such as anomaly and specification-based systems. Signature-based systems are not subject to this measure because signature databases contain known attack patterns [16].

IV. INTRUSION DETECTION DATASETS

Evaluation records play an important role in validating an IDS approach by allowing us to evaluate the ability of the proposed method to detect intrusive behavior. The datasets used for analyzing network packets in commercial products are not readily available for data protection reasons. However, there are publicly available records such as DARPA, KDD, NSL-KDD and ADFA-LD which are commonly used as a reference. The existing datasets used to create and compare IDSs are

explained in this section with their functions and restrictions.

A. KDD Cup 99 Dataset

The earliest effort to create an IDS dataset was made by DARPA (Defense Advanced Research Project Agency) in 1998 and they created the KDD98 (Knowledge Discovery and Data Mining (KDD)) dataset. In 1998, DARPA introduced a programmer at the MIT Lincoln Labs to provide a comprehensive and realistic IDS benchmarking environment (MIT Lincoln Laboratory, 1999). Although this dataset was an important contribution to the research on IDS, its accuracy and capability to consider real-life conditions have been widely criticized.

These datasets were collected using multiple computers connected to the Internet to model a small US Air Force base of restricted personnel. Network packets and host log files were collected. Lincoln Labs built an experimental testbed to obtain 2 months of TCP packets dump for a Local Area Network (LAN), modelling a usual US Air Force LAN. They modelled the LAN as if it were a true Air Force environment, but interlaced it with several simulated intrusions. The collected network packets were around four gigabytes containing about 4,900,000 records. The test data of 2 weeks had around 2 million connection records, each of which had 41 features and was categorized as normal or abnormal. The extracted data is a series of TCP sessions starting and ending at well-defined times, between which data flows to and from a source IP address to a target IP address, which contains a large variety of attacks simulated in a military network environment. The 1998 DARPA Dataset was used as the basis to derive the KDD Cup99 dataset which has been used in Third International Knowledge Discovery and Data Mining Tools Competition (KDD, 1999).

The KDD Cup 1999 dataset was used for the Third International Knowledge Discovery and Data Mining Tools Competition. Each connection instance is described by 41 attributes (38 continuous or discrete numerical attributes and 3 symbolic attributes). Each instance is labeled as either normal or a specific type of attack. These attacks fall under one of the four categories: DoS, Probe, U2R, and R2L. KDD Cup 1999 provided both the training and testing datasets, which are called 10% KDD and corrected dataset, respectively. The 10% KDD dataset contains 22 types of attacks,

whereas the corrected dataset features the same 22 types of attacks, along with 17 additional attack types. These datasets are out-of-date as they do not contain records of recent malware attacks. For example, attackers' behaviors are different in different network topologies, operating systems, and software and crime toolkits. Nevertheless, KDD99 remains in use as a benchmark within IDS research community and is still presently being used by researchers.

B. CAIDA Dataset

This dataset contains network traffic traces from Distributed Denial-of-Service (DDoS) attacks, and was collected in 2007. This type of denial-of-service attack attempts to interrupt normal traffic of a targeted computer, or network by overwhelming the target with a flood of network packets, preventing regular traffic from reaching its legitimate destination computer. One disadvantage of the CAIDA dataset is that it does not contain a diversity of the attacks. In addition, the gathered data does not contain features from the whole network which makes it difficult to distinguish between abnormal and normal traffic flows.

C. NSL-KDD Dataset

NSL-KDD is a public dataset, which has been developed from the earlier KDD cup99 dataset. A statistical analysis performed on the cup99 dataset raised important issues which heavily influence the intrusion detection accuracy, and results in a misleading evaluation of AIDS. The main problem in the KDD data set is the huge amount of duplicate packets. Tavallae et al. [18] analyzed KDD training and test sets and revealed that approximately 78% and 75% of the network packets are duplicated in both the training and testing dataset. This huge quantity of duplicate instances in the training set would influence machine-learning methods to be biased towards normal instances and thus prevent them from learning irregular instances which are typically more damaging to the computer system. Tavallae et al. [18] built the NSL-KDD dataset in 2009 from the KDD Cup'99 dataset to resolve the matters stated above by eliminating duplicated records. The NSL-KDD train dataset consists of 125,973 records and the test dataset contains 22,544 records. The size of the NSL-KDD dataset is sufficient to make it practical to use the whole NSL-KDD dataset without the necessity to sample randomly. This has produced consistent and comparable results from various research works. The NSL_KDD

dataset comprises 22 training intrusion attacks and 41 attributes (i.e., features). In this dataset, 21 attributes refer to the connection itself and 19 attributes describe the nature of connections within the same host.

D. ISCX 2012 Dataset

In this dataset, real network traffic traces were analyzed to identify normal behavior for computers from real traffic of HTTP, SMTP, SSH, IMAP, POP3, and FTP protocols. This dataset is based on realistic network traffic, which is labeled and contains diverse attacks scenarios.

E. CICIDS 2017 Dataset

CICIDS2017 dataset comprises both benign behavior and also details of new malware attacks: such as Brute Force FTP, Brute Force SSH, DoS, Heartbleed, Web Attack, Infiltration, Botnet and DDoS. This dataset is labelled based on the timestamp, source and destination IPs, source and destination ports, protocols and attacks. A complete network topology was configured to collect this dataset which contains Modem, Firewall, Switches, Routers, and nodes with different operating systems (Microsoft Windows (like Windows 10, Windows 8, Windows 7, and Windows XP), Apple's macOS iOS, and open source operating system Linux). This dataset contains 80 network flow features from the captured network traffic.

F. UNSW-NB 15 Dataset

The raw network packets of the UNSW-NB 15 data set was created by the IXIA PerfectStorm 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 behaviors. Tcpdump tool is utilized to capture 100 GB of the raw traffic (e.g., Pcap files). This data set has nine types of attacks, namely, Fuzzers, Analysis, Backdoors, DoS, Exploits, Generic, Reconnaissance, Shellcode and Worms. The Argus, Bro-IDS tools are used and twelve algorithms are developed to generate totally 49 features with the class label [19].

V. DEEP LEARNING AND INTRUSION DETECTION

Deep learning models consist of diverse deep networks. Among them, deep belief networks (DBNs), deep neural networks (DNNs), convolutional neural networks (CNNs), and recurrent neural networks (RNNs) are

supervised learning models, while autoencoders, restricted Boltzmann machines (RBMs), and generative adversarial networks (GANs) are unsupervised learning models. The number of studies of deep learning-based IDSs has increased rapidly from 2015 to the present. For large datasets, deep learning methods have a significant advantage over shallow models. In the study of deep learning, the main emphases are network architecture, hyperparameter selection, and optimization strategy.

A. Autoencoder

An autoencoder contains two symmetrical components, an encoder and a decoder, as shown in Figure 1. The encoder extracts features from raw data, and the decoder reconstructs the data from the extracted features. During training, the divergence between the input of the encoder and the output of the decoder is gradually reduced. When the decoder succeeds in reconstructing the data via the extracted features, it means that the features extracted by the encoder represent the essence of the data. It is important to note that this entire process requires no supervised information. Many famous autoencoder variants exist, such as denoising autoencoders and sparse autoencoders.

Figure 1: The structure of an autoencoder

B. Restricted Boltzmann Machine (RBM)

An RBM is a randomized neural network in which units obey the Boltzmann distribution. An RBM is composed of a visible layer and a hidden layer. The units in the same layer are not connected; however, the units in different layers are fully connected, as shown in Figure 2. where v_i is a visible layer, and h_i is a hidden layer. RBMs do not distinguish between the forward and backward directions; thus, the weights in both directions are the same. RBMs are unsupervised learning models trained by the contrastive divergence algorithm, and they are usually applied for feature extraction or denoising.

Figure 2: The structure of the RBM

C. Deep Brief Network (DBN)

A DBN consists of several RBM layers and a softmax classification layer, as shown in Figure 3. Training a DBN involves two stages: unsupervised pretraining and supervised fine-tuning. First, each RBM is trained using greedy layer-wise pretraining. Then, the weight of the softmax layer are learned by labeled data. In attack detection, DBNs are used for both feature extraction and classification [20–22].

Figure 3: The structure of the DBN

D. Convolutional Neural Network (CNN)

CNNs are designed to mimic the human visual system (HVS); consequently, CNNs have made great achievements in the computer vision field. A CNN is stacked with alternate convolutional and pooling layers, as shown in Figure 4. The convolutional layers are used to extract features, and the pooling layers are used to enhance the feature generalizability. CNNs work on 2-dimensional (2D) data, so the input data must be translated into matrices for attack detection.

Figure 4: The structure of a CNN

E. Recurrent Neural Network (RNN)

RNNs are networks designed for sequential data and are widely used in natural language processing (NLP). The characteristics of sequential data are contextual; analyzing isolated data from the sequence makes no sense. To obtain contextual information, each unit in an RNN receives not only the current state but also previous states. The structure of an RNN is shown in Figure 5. Where all the W items in Figure 8 are the same. This characteristic causes RNNs to often suffer from vanishing or exploding gradients. In reality, standard RNNs deal with only limited-length sequences. To solve the long-term dependence problem, many RNN variants have been proposed, such as long short-term memory (LSTM), gated recurrent unit (GRU), etc.

Figure 5: The structure of an RNN

F. Generative Adversarial Network (GAN)

A GAN model includes two subnetworks, i.e., a generator and a discriminator. The generator aims to generate synthetic data similar to the real data, and the discriminator intends to distinguish synthetic data from real data. Thus, the generator and the discriminator improve each other. GANs are currently a hot research topic used to augment data in attack detection, which partly ease the problem of IDS dataset shortages. Meanwhile, GANs belong to adversarial learning approaches which can raise the detection accuracy of models by adding adversarial samples to the training set.

VI. SHALLOW MODELS COMPARED TO DEEP MODELS

Deep learning is a branch of machine learning, and the

effects of deep learning models are obviously superior to those of the traditional machine learning (or shallow model) methods in most application scenarios. The differences between shallow models and deep models are mainly reflected in the following aspects.

- i. Running time: The running time includes both training and test time. Due to the high complexity of deep models, both their training and test times are much longer than those of shallow models.
- ii. Number of parameters: There are two types of parameters: learnable parameters and hyperparameters. The learnable parameters are calculated during the training phase, and the hyperparameters are set manually before training begins. The learnable parameters and hyperparameters in deep models far outnumber those in shallow models; consequently, training and optimizing deep models takes longer.
- iii. Feature representation: The input to traditional machine learning models is a feature vector, and feature engineering is an essential step. In contrast, deep learning models are able to learn feature representations from raw data and are not reliant on feature engineering. The deep learning methods can execute in an end-to-end manner, giving them an outstanding advantage over traditional machine learning methods.
- iv. Learning capacity: The structures of deep learning models are complex and they contain huge numbers of parameters (generally millions or more). Therefore, the deep learning models have stronger fitting ability than do shallow models. However, deep learning models also face a higher risk of overfitting, require a much larger volume of data for training. However, the effect of deep learning models is better.
- v. Interpretability: Deep learning models are black boxes. The results are almost uninterpretable, which is a critical point in deep learning. However, some traditional deep learning algorithms, such as the decision tree and naïve Bayes, have strong interpretability.

VII. PERFORMANCE EVALUATION MEASURES

To evaluate the performance of IDS algorithms, it is concentrated on three indications of performance: detection rate, accuracy and False Alarm Rate (FAR).

If one sample is an anomaly and the predicted label also stands anomaly, then it is called as true positive (TP).

If one sample is an anomaly, but the predicted label stands normal, then it is called as false negative (FN).

If one sample is a normal and the predicted label also stands normal, then it is true negative (TN).

If one sample is normal, but the predicted label stands anomaly, then it is termed as false positive (FP).

TP stands the number of true positive samples, FN stands the number of false negative samples, FP stands the number of false positive samples, and TN stands the number of true negatives.

To evaluate the proposed algorithm, it is concentrated on three indications of performance: detection rate, accuracy and False Alarm Rate (FAR).

If one sample is an anomaly and the predicted label also stands anomaly, then it is called as true positive (TP).

If one sample is an anomaly, but the predicted label stands normal, then it is called as false negative (FN).

If one sample is a normal and the predicted label also stands normal, then it is true negative (TN).

If one sample is normal, but the predicted label stands anomaly, then it is termed as false positive (FP).

TP stands the number of true positive samples, FN stands the number of false negative samples, FP stands the number of false positive samples, and TN stands the number of true negatives.

From equation (1-5), the accuracy, detection rate, False Positive rate (FPR), False Negative Rate (FNR) and False Alarm rate (FAR) is calculated.

$$\text{Accuracy} = (\text{TP} + \text{TN}) / (\text{TP} + \text{TN} + \text{FP} + \text{FN}) * 100 \quad (1)$$

$$\text{Detection Rate} = \text{TP} / (\text{TP} + \text{FN}) * 100 \quad (2)$$

$$\text{False Negative Rate (FNR)} = \text{FN} / (\text{FN} + \text{TP}) * 100 \quad (3)$$

$$\text{False Positive Rate (FPR)} = \text{FP} / (\text{FP} + \text{TN}) * 100 \quad (4)$$

$$\text{False Alarm Rate (FAR)} = (\text{FPR} + \text{FNR}) / 2 \quad (5)$$

VIII. CONCLUSION

In this paper, a detailed survey of intrusion detection system methodologies, types, and technologies with

their advantages and limitations. Several machine learning techniques that have been proposed to detect attacks are reviewed. However, such approaches may have the problem of generating and updating the information about new attacks and yield high false alarms or poor accuracy. In addition, the most popular datasets used for IDS research have been explored and their data collection techniques, evaluation results and limitations have been discussed. As normal activities are frequently changing and may not remain effective over time, there exists a need for newer and more comprehensive datasets that contain wide-spectrum of malware activities.

REFERENCES

- [1] A. Kartit, A. Saidi, F. Bezzazi, M. El Marraki & A. Radi, "A new approach to intrusion detection system," *Journal of Theoretical and Applied Information Technology*, Vol. 36, No. 2, 2012, pp. 284-289.
- [2] Sufyan T Faraj Al-Janabi, Hadeel Amjed Saeed, "A neural network based anomaly intrusion detection system", *IEEE*, 2011.
- [3] J. Ryan, M. Lin, and R. Miikkulainen, "Intrusion Detection with Neural Networks," *Conference in Neural Information Processing Systems*, 943-949.
- [4] A. K. Ghosh and A. Schwartzbard, "A Study in Using Neural Networks for Anomaly and Misuse Detection," *Conference on USENIX Security Symposium*, Volume 8, pp. 12-12, 1999.
- [5] P. L. Nur, A. N. Zincir-heywood, and M. I. Heywood, "Host-Based Intrusion Detection Using Self-Organizing Maps," in *Proceedings of the IEEE International Joint Conference on Neural Networks*, pp. 1714-1719, 2002.
- [6] K. Labib and R. Vemuri, "NSOM: A Real-Time Network-Based Intrusion Detection System Using Self-Organizing Maps," 2000.
- [7] Sharma, R.K., Kalita, H.K., Issac, B., "Different firewall techniques: a survey", *International Conference on Computing, Communication and Networking Technologies (ICCNT)*, *IEEE*, 2014.
- [8] Meng, Y.-X., "The practice on using machine learning for network anomaly intrusion detection", *International Conference on Machine Learning and Cybernetics (ICMLC)*, vol. 2, *IEEE*, 2011.
- [9] Sumaiya Thaseen Ikram, Aswani Kumar Cherukuri, "Intrusion detection model using fusion of chi-square feature selection and multi class SVM", *Journal of King Saud University -Computer and Information Sciences*, 2016.
- [10] Manjula C. Belavagi and Balachandra Muniyal, "Performance Evaluation of Supervised Machine Learning Algorithms for Intrusion Detection, *Procedia Computer Science*", Elsevier, 2016.
- [11] Saad Mohamed Ali Mohamed Gadal and Rania A. Mokhtar, "Anomaly Detection Approach using Hybrid Algorithm of Data Mining Technique", *International Conference on Communication, Control, Computing and Electronics Engineering*, *IEEE*, 2017.
- [12] Ibrahim, H. E., Badr, S. M., & Shaheen, M. A., "Adaptive layered approach using machine learning techniques with gain ratio for intrusion detection systems", *International Journal of Computer Applications*, vol. 56, issue 7, pp. 10-16, 2012.
- [13] Wen Feng, Qinglei Zhang, Gongzhu Hu, Jimmy Xiang Huang, "Mining network data for intrusion detection through combining SVMs with ant colony networks", *Elsevier*, Vol 37, pp 127-140, 2014.
- [14] Deshpande, A., & Sharma, R., "Anomaly Detection using Optimized Features using Genetic Algorithm and MultiEnsemble

- Classifier”, International Journal Online of Sports Technology & Human Engineering(IJOSTHE), 5(6), 2018. Retrieved from <https://ijosthe.com/index.php/ojssports/article/view/79>.
- [15] Kuang, F., Xu, W., & Zhang, S., “A novel hybrid KPCA and SVM with GA model for intrusion detection”, Applied Soft Computing Journal, vol. 18, pp. 178–184, 2014.
- [16] Prasanta Gogoi, D.K. Bhattacharyya, B. Borah¹ and Juga, K. Kalita, “MLH-IDS: A Multi-Level Hybrid Intrusion Detection Method”, The Computer Journal, Vol. 57 issue 4, pp. 602-623, 2014.
- [17] Wathiq Laftah Al-Yaseen , Zulaiha Ali Othman ,Mohd Zakree Ahmad Nazri, “Multi-Level Hybrid Support Vector Machine and Extreme Learning Machine Based on Modified K-means for Intrusion Detection System”, International Journal in Expert Systems With Applications, Elsevier, 2017.
- [18] M. Tavallaei, E. Bagheri, W. Lu, and A. A. Ghorbani, "A detailed analysis of the KDD CUP 99 data set," in 2009 IEEE symposium on computational intelligence for security and defense applications, 2009, pp. 1–6.
- [19] Nour Moustafa & Jill Slay, “The evaluation of Network Anomaly Detection Systems: Statistical analysis of the UNSW-NB15 data set and the comparison with the KDD99 data set”, Information Security Journal: A Global Perspective, 2015.
- [20] Zhao, G.; Zhang, C.; Zheng, L. “Intrusion detection using deep belief network and probabilistic neuralnetwork”, In Proceedings of the 2017 IEEE International Conference on Computational Science andEngineering (CSE) and IEEE International Conference on Embedded and Ubiquitous Computing (EUC),Guangzhou, China, 21–24 July 2017; Volume 1, pp. 639–642.
- [21] Alrawashdeh, K.; Purdy, C. “Toward an online anomaly intrusion detection system based on deep learning”,In Proceedings of the 2016 15th IEEE International Conference on Machine Learning and Applications(ICMLA), Anaheim, CA, USA, 18–20 December 2016; pp. 195–200.