Detection of Illegal Traffic Pattern using Hybrid Improved CART and Multiple Extreme Learning Machine Approach

J. Lekha[1], and Padmavathi Ganapathi[2]

[1]Department of Computer Science, Avinashilingam Institute for Home Science and Higher Education for Women, Coimbatore, India
[1]Department of Computer Science, Sri Krishna Arts and Science College, Sugunapuram, Kuniamuthur, Coimbatore, India
[2]Department of Computer Science, Avinashilingam Institute for Home Science and Higher Education for Women, Coimbatore, India

***Abstract*:** In the proposed hybrid intrusion detection process, misuse detection and anomaly detection model is integrated to detect the attack in traffic pattern. In misuse detection model, the traffic pattern is classified into known attack and not known attack. Each extracted normal data set does not have known attack and it contains small amount of varied connection patterns than overall normal data set. Anomaly detection model classifies the not known attack as normal data set and unknown attack thus improving the performance of normal traffic behavior. Experiment is carried out using NSL –KDD dataset and performance of proposed approach is compared with traditional learning approaches in terms of training time, testing time, false positive ratio and detection ratio. The proposed method detects the known attacks and unknown attacks with ratio of 99.8 % and 52% respectively.

***Keywords*:** Anomaly detection, misuse detection, traffic pattern, hybrid approach.

1. Introduction

IDS are an intrusion detection system [1] that detects the presence of attack in the given environment. The IDS is made available in the network to restrict the connection by analyzing all the incoming connections. The IDS detects abnormal activities of the node by analyzing the collected packets, acknowledging the admin, black listing the node, disrupting all existing connections with the attack node and prevents network from further damage. IDS are also connected with the firewall to enhance network security. Generally, intrusion detection algorithms are classified as: misuse detection (known attack) and anomaly detection (unknown attack) [2]. Misuse detection model detect attacks based on the known behavior of the traffic. They are effective in detecting known attacks that causes low attack to the network. However, they could not detect new attacks that possess different characteristics as that of known attack. Anomaly detection algorithms study normal traffic pattern and incoming node traffic behavior. The anomaly detection method assumes that the attacker user behavior differs to that of a normal user. They assume the node as an intruder if the behavior of the traffic is far different from normal traffic. Since various types of different connections are available anomaly detection cannot classify the attack accurately. This affects the performance of the algorithm [3].Anomaly detection algorithms detect new attack patterns, but the detection rate is not much effective as misuse detection models for known attacks. False positive rate is higher for anomaly detection, which is a ratio of misclassified normal traffic. To reduce the drawbacks of these two traditional intrusion detection methods, a hybrid intrusion detection method combining misuse detection method and anomaly detection method is used [2].Only if both the algorithms define the connection as an attack, the hybrid detection method considers it as an attack connection thus reducing the false classification rates. In the proposed work, Improved CART is used in misuse detection model and Extreme learning machine [4] is used in multiple anomaly detection models. Improved CART is used to classify the incoming traffic into known attack and not known attack in misuse detection. PSO is implemented in Anomaly detection method to improve the detection accuracy. It produces more accuracy when integrated with ELM. Multiple ELM based PSO detects the unknown attacks with high accuracy.

2. Related Works

An anomaly detection is proposed in [5] by calculating the interrelation of IP addresses in the outgoing traffic pattern at the doorway router. Through statistical analysis, anomaly detection is made efficient by transforming this interrelated data using discrete wavelet transform. Trace- driven evaluation shows that the proposed method works well in detecting anomalies closer to the source. The graph is presented indicating the anomalies detected using number of flows and port number relation. Flow based unusual network traffic detection is proposed in [6] to increase the detection accuracy. This method combines several packets that have identical flows thereby reducing processing costs of packet data. The detection mechanism uses a detection function to detect the attack when there is a change in the traffic pattern. This function can also detect mutant attack in case if it uses alternate port number. Also it can detect some flooding attacks that occur in the network. The parameters that can revert changes in traffic characteristics during attack are considered while detecting abnormal traffic. However if the attack does not cause any change in traffic pattern, this method remains difficult in detecting the attack. A payload based detection mechanism is proposed in [7] for preventing the network from unknown attacks. The proposed method focuses on application level network anomaly detection. It uses a keyword based approach which includes 2 phases: Training phase and Detecting phase. Training phase builds the dataset and detecting phase matches the incoming keyword with the stored keyword and sends an alarm if the keyword does not match. In low false alarm conditions, the detection rate is not satisfying. However the results show that

extracting useful data from packet payload results in reasonable performance.

Prevention of DDoS attack and Data modification attack is presented in [8]. This paper presents DDoS and Data Modification attack scenario and also provides the solution to prevent it. In case of data modification attack, it shows how easy to read/forward/modify the data exchanged between a cluster head node and computing nodes.

A Fuzzy Logic based Defense Mechanism against Distributed Denial of Service Attack is presented in [9]. A fuzzy logic based defense mechanism that can be set with predefined rules by which it can detect the malicious packets and takes proper counter measures to mitigate the DDoS attack is proposed. Also a detailed study of different kind of DDoS attack and existing defense strategies has been carried out.

A novel Intrusion Detection scheme named Intrusion Detection using Naïve Bayesis implemented in [10]. This IDNB is a traffic classification scheme to detect the intruded packets to increase the performance. The Naïve Bayes classifier is helpful to obtain a better solution to detect attack in an uncertain world because of its predictable feature. It also requires less number of training data to measure the parameters of a classification model. The results show the detection rate and false positive rates achieved using IDNB. The proposed scheme produces 92.34% accuracy in classifying packets which is higher than other existing algorithms. But with the increase in traffic, accuracy and detection rate decreases.

A new detection algorithm S3 is proposed in [11] to detect the important and short span anomalies. Bayes Net detects the anomaly in multiple input signals. Bit rate, correlation between the incoming and outgoing packet is considered as input signals that helps in identifying anomaly. The proposed algorithm does not deal with true positives. The experimental results ensure that S3 algorithm perform traffic anomaly detection with zero false positive.

Table 1 Review of Literature

Year	Author	Techniques used	Parameter used	Observations
2015	V. Hema and C. EmilinShyni	Intrusion Detection using Naïve Bayes	Accuracy. Detection rate and False positive rate.	It produces 92.34% accuracy in classifying packets. With the increase in traffic, accuracy and detection rate decreases.
2008	SeongSoo Kim and A. L. Narasimha Reddy	A traffic anomaly detector, operated in postmortem and in real-time, by passively monitoring packet headers of traffic.	Trace-driven evaluation.	It provides an effective means of detecting anomalies close to the source
2008	Jeff Kline, Sangnam Nam, Paul Barford, David Plonka, and Amos Ron	Detection algorithm called S3 that utilizes a Bayes Net	Accuracy False positives	Achieves over a 20% improvement in accuracy. It performs traffic anomaly detection with zero false positives.
2007	Like Zhang and Gregory B. White	Application level network anomaly detection	Total Attacks Detected. Payload related Attacks. Overall False Positive Rate.	It has slightly higher false positive rate. If there are very few payload related attacks, and they are difficult to detect. Able to detect more attacks always.
2004	Myung-Sup Kim , Hun-Jeong Kong , Seong-Cheol Hong , Seung-Hwa Chung and J. W. Hong	A Flow-based Method for Abnormal Network Traffic Detection	System overhead. Detection rate.	Detection accuracy is increased. This function detects mutant attacks that use new port numbers or a changed payload. If an attack does not influence network traffic, it is difficult to detect this type of attack.

3. Proposed Methodology

3.1. Proposed Hybrid misuse and anomaly detection for traffic

The proposed Hybrid Intrusion Detection techniques contain both misuse and anomaly detection to detect the known attack and unknown attack from the incoming connection. While establishing connection, illegal traffic is intruded to attack the network. Hence traffic features are extracted and analyzed using this hybrid approaches. In misuse detection, improved CART technique is used where as in anomaly detection extreme learning machine (ELM) algorithm is used. The proposed flow diagram is given in figure 1.

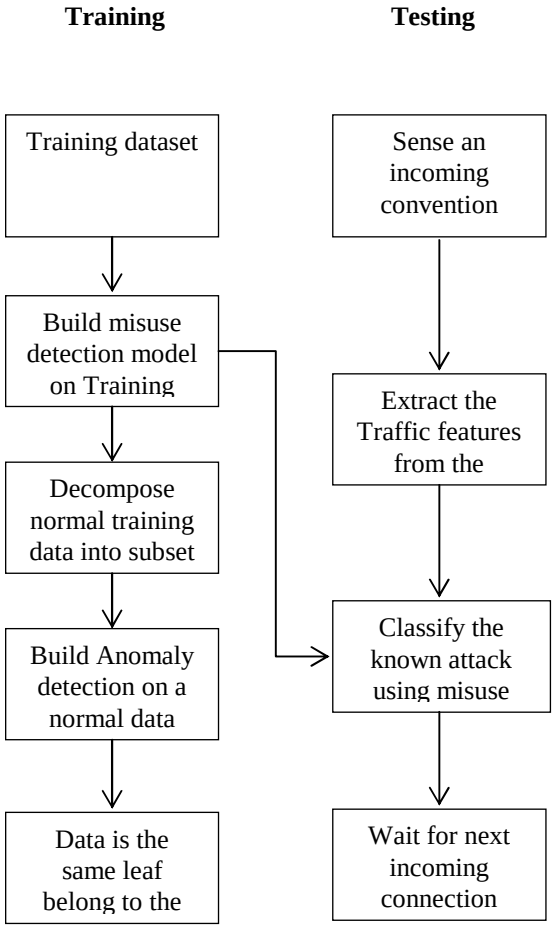

Figure 1. Proposed flow diagram

3.2 Improved CART (ICART)

Decision tree is tree structure that is used to classify the data based on the decision. An Improved Classification and Regression Trees (CART) is a one type of classifier which take historical data to construct the decision tree. A CART tree is a binary decision tree that node is splitting into two child nodes repeatedly, beginning with the root node that contains the whole learning sample [12].

The fundamental idea of constructing tree is to choose a split along with all the feasible splits at each node so that the resulting child nodes are accurate. In this algorithm consider only uni-variate splits. Each split depends on the value of only one predictor variable.

All possible splitting nodes find out the possible splits of each predictor. Check Y is a nominal categorical variable of J categories; there are $2^{J-1}-1$ possible splits for this predictor. Check Y is an ordinal categorical or continuous variable with M different values; there is M - 1 different split on Y. A tree is grown starting from the root node. The following procedure is used iteratively at each node.

3.2.1 Select best splitting predictor

In each continuous and ordinal predictor, arrange the values in ascending order and then check the value from the top node of tree to determine splitting point that examine best and maximize the splitting criteria when the node is split according to it. To observe each nominal predictor and possible subset under the categories(call it B, if y ∈ B, the condition is true goes to the left child node, otherwise, goes to the right.) to select the best split.

3.2.2 Select the best splitting node in the tree

The best splitting node to found in step 1, predict the particular one node that maximize the splitting criteria.

To identify the best splitting node found in step 2, to check stopping rules are not satisfied.

The categorical dependent variable if X is categorical, the splitting criteria available: Gini criteria

At node s, let probabilities q(n,s) , q(s) and q(n| s) be estimated by

$$Q(n, s) = \frac{\pi(n) R_{c,n}(s)}{R_{c,n}}$$

$$Q(s) = \sum_{n} Q(n, S)$$

$$Q(n \setminus s) = \frac{Q(n,s)}{Q(s)} = \frac{Q(n,s)}{\sum_{n} Q(n,s)}$$

Where,

$$r_{c,n} = \sum_{r \in t} p_{rg_r k(x_r = n)}$$

With k (d=e) being indicator function calling value 1 when d=e,0 otherwise.

Gini criteria,

$$Gini(f, a) = 1 - \sum_{z_{hdom}(f)} \left(\frac{a_{f=i_h a}}{|a|}\right)$$

Therefore the evolution criteria procedure for selecting the attribute f_i is denoted as

Rand-Gini gain (f_i, a_v)=

$$\text{Gini}(f, a_v) - \sum \frac{a_{fi=ve_{i,h}a_v}}{|a_v|} XGini(f, a_{fi=ve_{i,h}a_v}) \quad ve_{i,h}$$

The procedure for constructing new decision tree using RGG is given below:

New decision tree(S, Arr_list, RGG)
S - Data Partition
Arr_list - Attribute List
RGG –Randomized Gini Gain
L_R – Label root of n
Begin
Initialize a node n
Check samples in n ∈ to same class, P then
Return n as a leaf node
Check Arr_list≠ 0
Then apply Rand-GiniGain (f_i, a_v)
Set L_R as g (Arr_list)
For each outcome O of g (Arr_list) do
SubtreeO=New decision tree (s_i, Arr_list, RGG)
Join the root node n to subtreeO
Return n
End

In the improved CART techniques some stopping rules are applied to verify the tree growing process termination. The following subsequent stopping rules are used in this procedure:

i) If the node is accurate in all cases that have identical values and dependent variable then those nodes should not be split.
ii) In a node, if all cases contain identical values for each predictor, that node should not be split.
iii) When current tree depth reaches the user specified maximum depth limit at the time tree growing process should stop.
iv) The size of the node is less than the user specified minimum node ,the node should not be spilt.
v) The splitting node results in a child node whose node size is less than user specified minimum child node size value the node should not be spilt.

In case the best split bs of node s, the development $\Delta k(bs, s)$= q(s) $\Delta j(bs, s)$ is smaller than the user-specified minimum development, that node should not spilt. Here, in this work Improved CART is used to classify the incoming traffic into known attack and not known attack in misuse detection.

3.3 Extreme learning machine

Extreme Learning Machine (ELM) theory was proposed by [4] for single hidden-layer feed forward networks (SLFNs). It is applied to real-world problems such as regression and classifications. In ELM, the number of hidden nodes must be defined, and then the input weights and biases are randomly assigned while output weights can be determined analytically by generalized inverse function. The training phase can be completed through nonlinear transformation without undergoing learning process. As the learning parameters are randomly assigned, they remain unchanged in training phase. Several variants of ELM [13] such as Incremental ELM, evolutionary ELM, error-minimized ELM, Pruning ELM, two-stage ELM, online sequential ELM, voting-based ELM, fully complex ELM, ordinal ELM and symmetric ELM are proposed to overcome that issue.

For S training samples (y_i, t_i), where
$y_i = [x_{i1}, x_{i2}, \dots, x_{im}]^T \in R^m$ and $t_i = [t_{i1}, t_{i2}, \dots, t_{in}]^T \in R^n$
with $\tilde{S}$ hidden nodes and activation function f(y).

$$\Sigma_{i=1}^{\tilde{s}} \beta_i f_i(y_j) = \Sigma_{i=1}^{\tilde{s}} \beta_i f(p_i . y_j + m_i) = O_j (1)$$

where j=1,2,…,S and $p_i = [p_{i1}, p_{i2}, \dots, p_{im}]^T$ is weight vector related to the i[th] hidden node and input nodes, $\beta_i = [\beta t_{i1}, \beta t_{i2}, \dots, \beta t_{in}]^T$ is weight vector relating i[th] hidden node and output nodes and m_i is the i[th] hidden nodes threshold.

set $p_i . y_j$ be the inner product of p_i and y_j. Then (1) can be written as,

$$N\beta = T (2)$$

where,

$$N = \begin{bmatrix} f(p_1 . y_1 + m_1) & \dots & f(p_{\tilde{s}} . y_1 + m_{\tilde{s}}) \\ \vdots & \dots & \vdots \\ f(p_1 . y_S + m_1) & \dots & f(p_{\tilde{s}} . y_S + m_{\tilde{s}}) \end{bmatrix} (3)$$

$$\beta = \begin{bmatrix} \beta_1^T \\ \vdots \\ \beta_{\tilde{s}}^T \end{bmatrix}_{\tilde{s} \times n} \text{ and } T = \begin{bmatrix} t_1^T \\ \vdots \\ t_S^T \end{bmatrix}_{S \times n} (4)$$

The output weights β can be calculated from,

$$\beta = N^{\dagger} T (5)$$

Where $N^{\dagger}$ is the Moore-Penrose generalized inverse operation of Matrix N.

ELM algorithm can be stated in three steps:

***Algorithm*:**
***Input:*S training samples (y_i, t_i), where**
$y_i = [x_{i1}, x_{i2}, \dots, x_{im}]^T \in R^m$ and
$t_i = [t_{i1}, t_{i2}, \dots, t_{in}]^T \in R^n$ with $\tilde{S}$ hidden nodes and activation function f(y).
***Output:*Output weight, $\beta = N^{\dagger} T$**
***Step 1.*Parameters of hidden nodes (y_i, t_i), where i=1,2,…,$\tilde{S}$ are randomly assigned.assign input weight p_i and bias m_i**
***Step 2.* Matrix of hidden node N is calculated**

$$N = \begin{bmatrix} f(p_1 . y_1 + m_1) & \dots & f(p_{\tilde{s}} . y_1 + m_{\tilde{s}}) \\ \vdots & \dots & \vdots \\ f(p_1 . y_S + m_1) & \dots & f(p_{\tilde{s}} . y_S + m_{\tilde{s}}) \end{bmatrix}$$

***Step3.Output* weightβ is calculated using $\beta = N^{\dagger} T$, where $T = [t_1, t_2, \dots, t_S]^T$**

In this work ELM is used to classify the incoming not known attack into normal and unknown attack in Anomaly detection.

3.4 Particle Swarm Optimization

PSO was introduced by [14] used to solve optimization problems. It was developed as an inspired mechanism followed by bird flocking behavior. Consider the particles search for a single piece of food that is available in a given search space. The best idea is to follow the bird which is nearer to food. Here bird is referred as particle. Each particle has its own velocity which moves the particle towards the solution and fitness value calculated by fitness function $f(x) = (x1^2 + x2^2 + \dots + xn^2)$.

Initially PSO has a group of random solutions. From then it searches for the optimal solution by updating generations. Each particle is updated by two best values in each iteration. The first one is personal best and it is called as pbest. pbest is

the best fitness value obtained by the particle. Another one is global best and it is called as gbest. It is the fitness value tracked by PSO and it is the best value obtained by the particles in the population. The particles update pbest and gbest values in each iteration. The loop continues until all particles exhaust or the maximum iteration is reached.

The particles update its velocity and position after finding two best values. The following equation is used to find the velocity and position of each particle.

$$vel = vel + c1 * u1 * (pb - current) + c2 * u2 * (gb - current) \quad (6)$$

$$current = current + vel \quad (7)$$

PSO is implemented in Anomaly detection method to improve the detection accuracy. It produces more accuracy when integrated with ELM.

3.5 Hybrid misuse and anomaly detection

Using Improved CART mechanism the training model is built depending on the training dataset. The misuse detection model is effective in detecting known attacks that causes low damage. It detects only the known attacks with a small false positive rate. However, they could not detect new attacks that possess different characteristics as that of known attack. It is known that false positive rate of ICART method is low; each normal training dataset is trained by ELM.

ELM model is trained and is decomposed by ICART model. The reason behind decomposing normal dataset is that anomaly detection scheme using ELM can further categorize the normal dataset into unknown attack and normal dataset. There may be various normal patterns as stated in protocol type, service type and so on.

The ELM model is very conscious to the training dataset and can result in producing high false positives. To mitigate this issue, the normal dataset is dissolved into smaller data subsets. With these obtained subsets multiple ELM are built. Therefore the patterns of these smaller subsets will be less complex as that of complete dataset. For each dataset multiple models are built and this will be less flexible than building a single model for large dataset. This combined ICART-ELM model targets the normal data on smaller datasets as individual ELM on smaller region find out its appropriate normal pattern in the dataset. Hence, ICART-ELM model can detect the attack with low false positive rate. The training time of a dataset and the detecting time involved in anomaly detection can be improved by tree decomposition method.

4. Result and discussion

The proposed methodology is implemented using Java as front end and MySQL as back end. The experiment is conducted for evaluating the effectiveness of the proposed work using NSL-KDD dataset. The NSL-KDD is an enhanced version of KDD'99 in terms of redundant instance removal. KDD'99 dataset is found difficult if there are number of redundant instances in training and testing dataset. Hence NSL-KDD dataset was proposed by discarding all the redundant instances and remodeling the dataset for accurate evaluation of the proposed works.

The KDDTrain+.TXT and KDDTest+.TXT documents are modified in the NSL-KDD data set are organized for evaluation. These contain traffic details including information about normal traffic pattern features and a connection label that specifies the attack type.KDDTest+.TXT contain some attack types that may not be available in KDDTrain+.TXT. Using connection label the attack records in KDDTest+.TXT categorized as known attack and unknown attack. The attacks with the same label in the KDDTrain+.TXT and KDDTest+.TXT need not have similar traffic features to be categorized as known attacks.

Therefore the training data set and testing dataset were organized in order to categorize known attacks and unknown attacks in the following manner. The traffic data in KDDTest+.TXT is divided into two sets depending on the type of connection is known in KDDTrain+.TXT. The first set contains the connections that are known by KDDTrain+.TXT and second one contains connections that are not known by KDDTrain+.TXT and they are labeled as unknown attack. The first set is combined with KDDTrain+.TXT and then it is evenly divided into training dataset and testing dataset depending on the type of connection. In Testing dataset, two types of connections are available. Then KDDTest+.TXT were added to testing set thus completing the testing set organization.

The following table describes the dataset used for training and testing of different protocols. In the given dataset, partial amount of data is used for training and remaining is used for testing purpose.

Table 2. Training and Testing Dataset

Dataset	TCP	UDP	ICMP
KDDTrain+20%	20526	3011	1655
KDDTrain+	102689	14993	8291
KDDTest+	18880	2621	1043

ELM and PSO classification training accuracy for proposed methodology is given in the table below. It is represented by percentage (%)

Table 3. Training accuracy for classifiers

Classifier	Accuracy (%)
ELM training accuracy	92
ELM with PSO training accuracy	96

From the table, it is clear that the ELM with PSO classifier achieves high training accuracy of 96% where ELM alone produces only 92% accuracy.

Table 4. Training time comparison between proposed and other methods

Methods	Training time(sec)
Proposed approach	40.2
DT-SVM	56.8
Conventional hybrid method (serial)	76.63
Conventional hybrid method (parallel)	76.63
Misuse detection method(DT)	35.21
Anomaly detection method(1-class SVM)	41.42

Training time for proposed method and other methods are listed in the above table. DT-SVM method takes 56.8 seconds; Conventional hybrid method takes 76.63 seconds, Anomaly detection takes 41.42 seconds. Our proposed method takes 40.2 seconds which is lesser than other methods except misuse detection which takes 35.21 seconds.

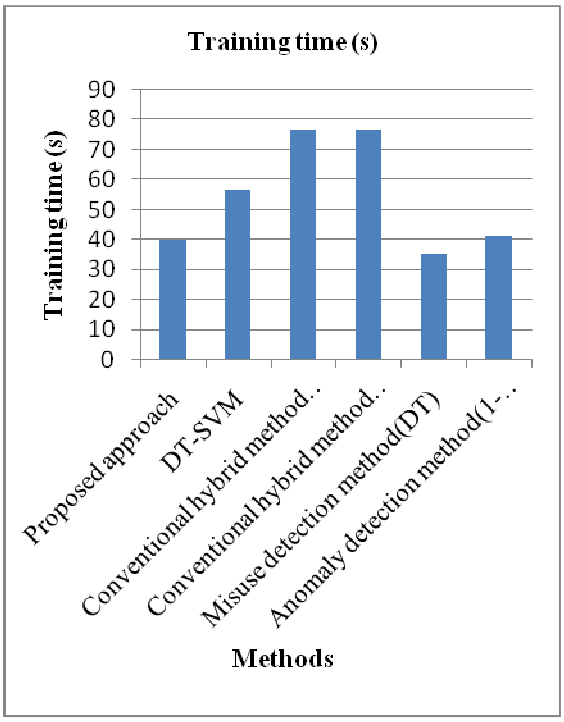

Figure 2. Training time of existing and proposed approach

From figure 2, it is understand that the proposed work of training time is lower than existing approach. It is represented in second(s).

Table 5. Testing time comparison between proposed and conventional methods

Methods	Testing time(sec)
Proposed approach	8.3
DT-SVM	11.2
Conventional hybrid method (serial)	15.62
Conventional hybrid method (parallel)	30.17
Misuse detection method (DT)	1.07
Anomaly detection method(1-class SVM)	29.1

Table 4lists the testing time taken for proposed method and other conventional methods. DT-SVM method takes 11.2 seconds; Conventional hybrid method takes 15.62 and 30.17 seconds, Misuse detection takes 1.07 seconds and Anomaly detection takes 29.1 seconds.It is clear that the proposed approach takes less testing time i.e. 8.3 seconds.

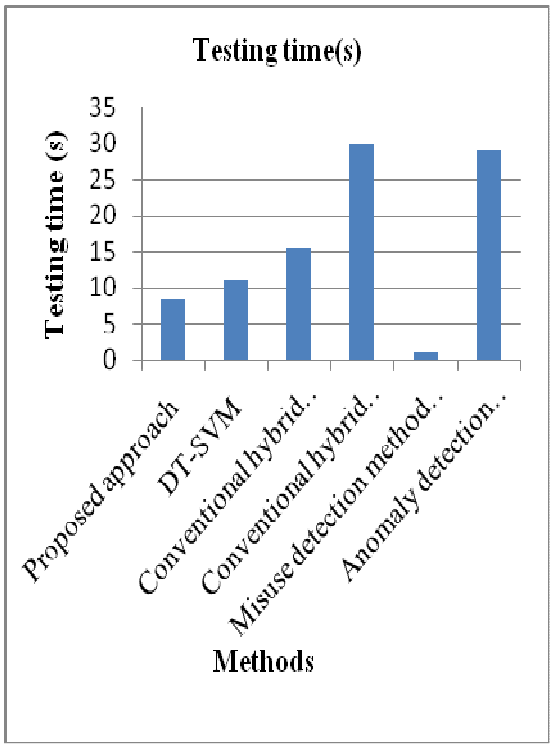

Figure 3 Testing time of existing and proposed approach

From figure 3, it is understand that the proposed work of testing time is lower than existing approach. It is represented in second(s).

Table 6. False positive, Detection rate known, Detection rate unknown comparison between proposed and DT-SVM

Metric	Proposed approach	DT-SVM
False positive (%)	1.8	1.2
Detection rate known (%)	99.8	99.1
Unknown (%)	52	30.5

From the above table, it is understood that false positive, Detection rate known and unknown is compared and proposed method is better than DT-SVM with higher values.

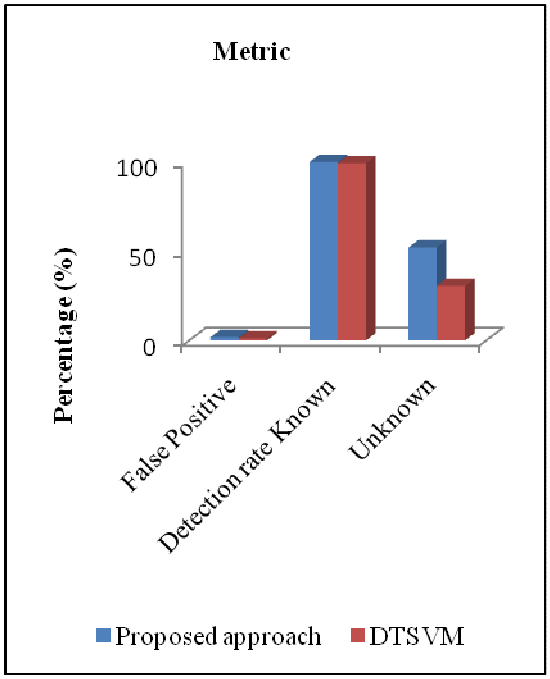

Figure 4. False positive, detection rate of known and unknown attack of existing and proposed approach

In figure 4, it is understand that the proposed work of false positive rate, detection rate of known and unknown attack are higher than existing approach. It is represented in percentage (%).

Detection time is compared between proposed approach and other methods. Proposed method takes lesser time of 35 seconds whereas existing DT-SVM method takes 42 seconds; Conventional hybrid method takes 65 and 72 seconds, Misuse detection takes 78 seconds and Anomaly detection takes 81 seconds.

In figure 5, it is understand that the proposed work of detection time is lower than existing approach. It is represented in second(s).

First, the detection performance of the proposed method was evaluated. The detection performance of proposed work was compared with the existing approaches. This experiment demonstrates that the proposed hybrid intrusion detection method is better than the conventional methods in terms of Accuracy, detection performance, training time, and testing time.

Table 7. Detection time comparison between proposed and other methods

Algorithm	Detection time(sec)
Proposed approach	35
DT-SVM	42
Conventional hybrid method (serial)	65
Conventional hybrid method (parallel)	72
Misuse detection method(DT)	78
Anomaly detection method(1-class SVM)	81

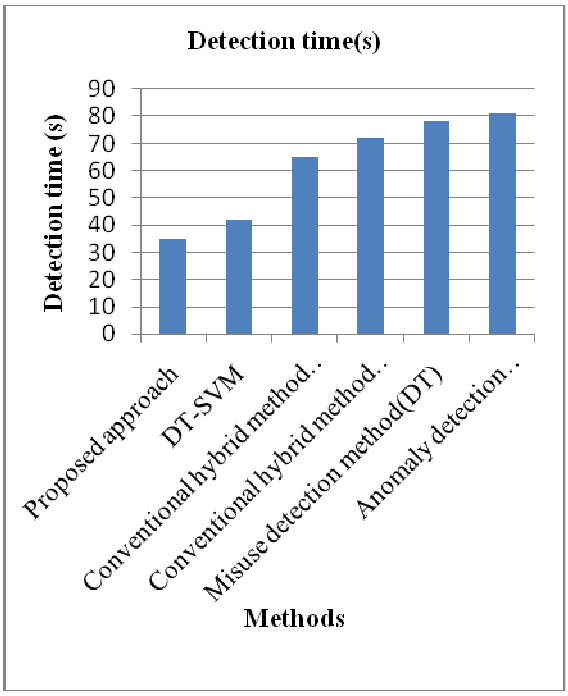

Figure 5 Detection time of existing and proposed approach

5. Conclusions

The proposed hybrid intrusion detection method is integrating a misuse detection model and an anomaly detection model in a decomposition structure. First, the ICART was used to create the misuse detection model that is used to decompose the normal training data into smaller subsets. Then, multiple ELM is used to create an anomaly detection model in each decomposed region. The experiments demonstrated that the proposed hybrid intrusion detection method could improve the IDS in terms of

detection performance for unknown attacks and detection speed.

References

[1] B. Santos Kumar et al, “Intrusion Detection System- Types and Prevention”, (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 4 (1), 2013, 77 – 82, ISSN:0975-9646

[2] Depren O, Topallar M, Anarim E, Ciliz MK., “An intelligent intrusion detection system (IDS) for anomaly and misuse detection in computer networks”, Expert systems with Applications. 2005 Nov 30;29(4):713-22.

[3] Jungsuk S, Takakura H, Okabe Y, Yongjin K,. “Unsupervised anomaly detection based on clustering and multiple one-class SVM”, IEICE transactions on communications, 2009 Jun 1;92(6):1981-90.

[4] Huang GB, Zhu QY, Siew CK,. “Extreme learning machine: theory and applications”, Neuro-computing, 2006 Dec 31;70(1):489-501.

[5] Kim SS, Reddy AL.,.” Statistical techniques for detecting traffic anomalies through packet header data”, IEEE/ACM Transactions on Networking (TON), 2008 Jun 1;16(3):562-75.

[6] Kim MS, Kong HJ, Hong SC, Chung SH, Hong JW, “A flow-based method for abnormal network traffic detection”, InNetwork operations and management symposium, 2004, NOMS 2004, IEEE/IFIP 2004 Apr 23 (Vol. 1, pp. 599-612), IEEE

[7] Zhang L, White GB, “Analysis of payload based application level network anomaly detection”, In System Sciences, 2007,HICSS 2007, 40th Annual Hawaii International Conference on 2007 Jan (pp. 99-99). IEEE

[8] B. Kodada, P. Gaurav, and R. Alwyn.Pais, "Protection Against DDoS and Data Modification Attack in Computational Grid Cluster Environment", International Journal of Computer Network and Information Security (IJCNIS), Vol.4, No.7, pp. 12-18, (2012).

[9] N. Iyengar, B. Arindam, and G. Gopinath, "A Fuzzy Logic Based Defense Mechanism against Distributed Denial of Services Attack in Cloud Environment”, International Journal of Communication Networks and Information Security (IJCNIS), vol.6, No.3, pp. 233-245, (2014)

[10] V. Hema and C. EmilinShyni, “DoS Attack Detection Based on Naive Bayes Classifier”, Middle-East Journal of Scientific Research 23 (Sensing, Signal Processing and Security), page no. 398-405, 2015

[11] Kline J, Nam S, Barford P, Plonka D, Ron A, “Traffic anomaly detection at fine time scales with bayes nets”, In Internet Monitoring and Protection, 2008, ICIMP'08,The Third International Conference on 2008 Jun 29 (pp. 37-46). IEEE

[12] Breiman L, Friedman JH, Olshen R, Stone CJ, “Classification and Regression Trees”, Wadsworth & Brooks/Cole Advanced Books& Software. Pacific California, 1984.

[13]Ding S, Xu X, Nie R, “Extreme learning machine and its applications. Neural Computing and Applications”, 2014 Sep 1;25(3-4):549-56.

[14] Eberhart R, Kennedy J, “A new optimizer using particle swarm theory”, In: Proceedings of the sixth international symposium on micro machine and human science, vol 43 IEEE. New York. 1995.