A Novel Trust Based System to Detect the Intrusive Behavior in MANET

Deverajan Ganesh Gopal[1] and R. Saravanan[2]
(Corresponding author: Deverajan Ganesh Gopal)
School of Computing Science and Engineering, Vellore Institute of Technology, India[1]
School of Information Technology and Engineering, Vellore Institute of Technology, India[2]
(Email: ganeshgopal@vit.ac.in and rsaravanan@vit.ac.in)
(Received Apr. 20, 2014; revised and accepted July 5 & Dec. 12, 2014)

Abstract

MANET is vulnerable to several challenging security breaches because of dynamic nature. In this work, we propose an IDS that is based on trust rates. The entire work of this system can be compartmentalized into three phases. They are Pre-eminent node selection, Inter-cluster trust rate computation, Intra-cluster trust rate computation. In the first phase of this work, a pre-eminent node is selected based on the trust rate. The reason for the incorporation of the cluster based IDS is the effective utilization of energy, which is essential for dynamic MANET. The next phase of this work focuses on the inter-cluster trust rate computation. This is carried out by the Bayes Theorem. The next phase of the work is intra- cluster trust rate computation and this is done by the Dempster-Shafer theory. The system is tested in both aspects of routing and intrusion detection. The proposed system shows remarkable accuracy rate.
Keywords: Bayes Theorem, Dempster-Shafer Theory; MANET.

1 Introduction

A Mobile Ad hoc Network (MANET) is composed of several mobile nodes that are geographically dispersed. The concept of centralised authority is absent in this type of network and hence the nodes depend on each other, in order to transmit packets. Hence, it is evident that the nodes act both as routers and as hosts. Some of the applications of MANET include sensor networks, conferences and emergency based network [3].

MANET is vulnerable to several challenging security breaches because of dynamic nature and the attacks can either be active or passive. Active attacks may be in the form of packet tamper, packet deletion and packet replication. Passive attacks can be triggered by eavesdropping or silent listening, which affects the confidentiality. These attacks may completely shatter the entire network [8]. The best way to get rid of all these attacks is the deployment of an effective Intrusion Detection System (IDS) [7].

An IDS is vigilant against security attacks, or simply it monitors the behavior of nodes. IDS can only detect the attacks but cannot respond to it. Thus, the goal of an effective IDS is keeping an eagle-eye over the nodes and to detect the security threats.

The IDS can be deployed in two ways. The first way is deployment of IDS in every node or in the cluster head alone, which is the second option. This concept is depicted in Figure 1.

It is not feasible to deploy IDS to all nodes, because of the energy restriction of the mobile nodes. This problem can be handled by the second way of IDS deployment. In this case, the entire network is partitioned into several clusters, such that all the nodes of the network come under any of the cluster.

The constituent nodes of each cluster select a pre-eminent node. The IDS can be deployed in such pre-eminent nodes alone, such that these nodes can take care of the constituent nodes. Red coloured nodes in Figure 1(b) are the pre-eminent nodes and the other nodes are constituent nodes. The dotted circles represent the cluster.

In MANET, an IDS can take any one of the three different forms. They are stand- alone, cooperative or hierarchical [9]. The stand-alone form of IDS is executed in all the nodes and local response can be observed. The main pitfall of this architecture is the detection accuracy [6].

In the cooperative architecture of IDS, all nodes possess the local IDS but they share the data among themselves and work cooperatively [6]. Finally, in the hierarchical architecture of IDS, the network is broken into several clusters and a cluster head is chosen, based on a valid criterion. These cluster heads are more powerful than the normal

Figure 1: (a). Stand alone architecture of IDS, (b) Cluster based IDS

nodes [7]. The main merit of this architecture is the effective utilization of energy and the drawback is that this type of architecture is complex to adapt highly mobile MANETs [6].

In this work, we propose an IDS that is based on trust rates. The entire work of this system can be compartmentalized into three phases. They are

1) Pre-eminent node selection;

2) Inter-cluster trust rate computation;

3) Intra-cluster trust rate computation.

In the first phase of this work, a pre-eminent node is selected based on the trust rate.

The reason for the incorporation of the cluster based IDS is the effective utilization of energy, which is essential for dynamic MANET. The next phase of this work focuses on the inter- cluster trust rate computation. This is carried out by the Bayes Theorem. The next phase of the work is intra-cluster trust rate computation and this is done by the Dempster-Shafer theory.

This is followed by the summation of trust rates calculated by both the second and third phase of the system and this rate is termed as genuine trust. The outcome of this step yields a promising trust rate and the behavior of the node is predicted. The genuine trust is fed into the Ant-based clustering algorithm. This algorithm clusters the normal and the abnormal nodes separately and saves it for future reference. This list of abnormal nodes are notified to the pre-eminent node for proceed with further action.

The list of normal nodes is considered in routing by taking the path in which maximum normal nodes are present. This work is analysed with respect to routing and intrusion detection accuracy. The experimental outcome proves the efficiency of this work.

2 Proposed Work

In this work, we propose an IDS that is based on trust rates. The entire work of this system can be compartmentalized into three phases and are explained in this section. The assumptions of this work are every cluster knows its neighbor or nearby cluster and they enter a mutual agreement, such that two different clusters mutually calculate the trust rates of the constituent nodes.

To exemplify this concept, consider four different clusters A, B, C and D, where A is the immediate neighbor of C and the immediate neighbor cluster of B is D. In this case, there-eminent node of A and C enter into an agreement, so do the B and D. The trust rates of constituent nodes are computed mutually by A and C and the same process are carried out by B and D. This assumption is conceived by this work while computing the intra-cluster trust rate computation. The next assumption is that a trustworthy node remains trustworthy till the recovery process. A cluster can be established only with the one hop neighbors.

The entire work of this system can be compartmentalized into three phases. They are

1) Pre-eminent node selection;

2) Inter-cluster trust rate computation;

3) Intra-cluster trust rate computation.

In the first phase of this work, a pre-eminent node is selected based on the trust rate. The reason for the incorporation of the cluster based IDS is the effective utilization of energy, which is essential for dynamic MANET. The next phase of this work focuses on the inter- cluster trust rate computation. This is carried out by the Bayes? theorem. The next phase of the work is intra-cluster trust rate computation and this is done by the Dempster-Shafer Theory.

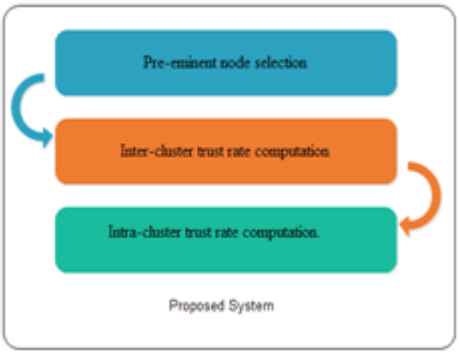

Figure 2: (Overflow of Proposed System

2.1 Cluster Establishment

In this work, we enforce a limit that every cluster can have the maximum of 7 nodes. This is to make the cluster to operate more effectively. After the deployment of nodes, if a cluster does not have maximum number of constituent nodes, it can send join_request to other nodes.

A node is chosen as the pre-eminent node among all the constituent nodes of the cluster. The pre-eminent node is chosen by taking the trust rate into account. A node with the highest trust rate in the cluster is chosen as the pre-eminent node. The trust rate is calculated by taking the packet delivery ratio and battery backup or energy into account.

Both the values range from 0 to 1 and a preferable value is set for both these parameters. The preferable value for packet delivery ratio and battery backup is 0.7 and 0.7 respectively. The threshold is fixed as 1.4. If a node's trust rate is greater than the fixed threshold, then that node is preferably the pre-eminent node. The trust rate of a node can be computed by the following.

$$Tr_{rate} = pdr + b_b \tag{1}$$

where pdr is the packet delivery ratio and b_b is the battery backup.

The packet delivery ratio can be computed by considering the in and out ratio of the forwarding history evidences. In normal case, in-ratio must be equal to the out-ratio. If the in-ratio is twice the out ratio, then the degree of selfishness will be 0.5. If the out- ratio is zero, then the node is considered to be completely selfish or it could be because of energy drop out.

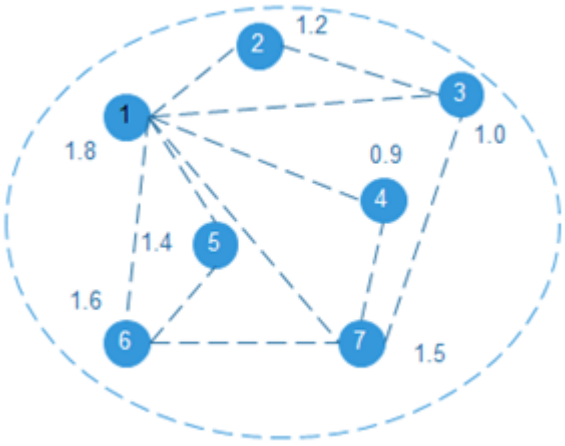

Figure 3: A Pre-eminent node selection

Let γ be the in-ratio of packets and μ be the out-ratio of the packets. If $\gamma = \mu$, then the node is trustworthy, that is, it forwards all the received packets and actively participates in the network. This type of node renders its fullest

cooperation to the network.

If $\gamma = \mu/2$, then the node is partially trustworthy. This type of node will forward the packets sometimes and not always. The behavior of such nodes varies with respect to time. If the value of $\mu = 0$, then the node is malicious and it may affect the entire network.

It can be noted from Figure 3 that the node with the highest trust value is chosen as the pre-eminent node. The trust value is computed by Equation (1) and the details are tabulated in Table 3.

Table 1: Sample Trust Value Table

Degree	In-Out Ratio	Description
1	$\gamma = \mu$	Trustworthy Node
2	$\gamma = \mu/2$	Partially Trustworthy Node
3	$\mu = 0$	Malicious Node

Table 2: Energy Value Table

Case	Energy value	Description
1	1	Full energized node
2	0.75	Pretty good energy
3	0.5	Half energized node
4	0.25	Poor energized node
5	0	Energy drained node

Table 3: Detecting the nature of node

Node ID	PDR	Battery Backup	Scenario	Status of the Node
Node 1	0.9	0.9	$1.8 > 1.4$	Trustworthy
Node 2	0.9	0.3	$1.2 < 1.4$	Untrustworthy
Node 3	0.3	0.7	$1.0 < 1.4$	Untrustworthy
Node 4	0.4	0.5	$0.9 < 1.4$	Untrustworthy
Node 5	0.7	0.7	$1.4 = 1.4$	Trustworthy
Node 6	0.8	0.8	$1.6 > 1.4$	Trustworthy
Node 7	0.7	0.8	$1.5 > 1.4$	Trustworthy

In this case, if the same node is retained as the pre-eminent node for a long time, then the battery back-up will deteriorate soon. By considering the dynamic nature of MANET and the aforementioned point, the cluster is re-established for every five seconds.

2.2 Inter-cluster Trust Rate Computation

The inter-cluster trust rate computation is carried out by Bayes' therorem. Bayes' theorem was proposed by Rev. Thomas Bayes, in the year 1763 [5]. The inter-cluster trust rate is computed by clubbing both the packet delivery ratio and battery backup together. The total trust is given by Equation (2):

$$P(tr|pdr, b_b) = \frac{P(pdr|tr, b_b) \times P(tr|b_b)}{P(pdr|b_b)} \tag{2}$$

In Equation (2), $P(tr|pdr, b_b)$ is the posterior probability. $P(pdr|tr, b_b)$ is the prior probability. $P(tr|b_b)$ is called as the likelihood and it provides the probability of trust rate for the battery backup. $P(pdr|b_b)$ is the normalizing factor.

Algorithm 1: Algorithm for pre-eminent node selection

```
Input: Set of nodes
Output: Clusters
Begin
cn_th = 7;
For every 5 seconds
Randomly select a node;
Draw a circle that encloses 7 nodes;
if (cn < 7)
Broadcast join request;
For every cn of a cluster
Do
Compute Tr_rate = pdr + b_b;
Find the ID of node with greatest Tr_rate
Declare it as pre-eminent node
End
```

The likelihood function of $P(pdr|tr, b_b) = P(pdr|tr)$. The probability of trust rate is given by

$$P(tr|pdr, b_b) = \frac{P(pdr|tr) \times P(tr|b_b)}{P(pdr|b_b)} \tag{3}$$

$P(pdr|tr)$ is calculated by Bayes' theorem and is given below.

$$P(pdr|tr) = \frac{P(tr|pdr) \times P(pdr)}{P(tr)} \tag{4}$$

Equation (4) is applied in Equation (3) and is presented in Equation (5).

$$P(tr|pdr, b_b) = \frac{P(tr|pdr) \times P(tr|b_b) \times P(pdr)}{P(pdr|b_b) \times P(tr)} \tag{5}$$

The normalizing factor is eliminated and the resultant equation is provided below.

$$P(tr|pdr, b_b) = P(tr|pdr) \times P(tr|P(tr|b_b). \tag{6}$$

The maximum trust value that can be obtained by this Bayes' theorem is 1. If the node's in-ratio is equal to the out-ratio of packets and if the node is fully energized, then the trust rate is 1. Trust rate may turn to 0 if the packet delivery ratio and the battery backup is not up to the mark.

This trust rate is calculated by the pre-eminent node for all its constituent nodes and is stored in the local table of every constituent node. This way of trust rate computation serves well.

2.3 Intra-cluster Trust Rate Computation

The intra cluster trust rate computation is done by the Dempster-Shafer theory. The assumptions of this phase are every cluster knows its neighbor or nearby cluster and they enter a mutual agreement, such that two different clusters mutually calculate the trust rates of the constituent nodes.

To exemplify this concept, consider four different clusters A, B, C and D, where A is the immediate neighbor of C and the immediate neighbor cluster of B is D. In this case, the pre-eminent node of A and C enter into an agreement, so do the B and D. The trust rates of constituent nodes are computed mutually by A and C and the same process are carried out by B and D. This assumption is conceived by this work while computing the intra-cluster trust rate computation.

Dempster-Shafer theory is introduced by Arthur P. Dempster [2]. This theory is also known as evidence theory. Dempster-Shafer theory is impressive because of the feature that it requires no prior knowledge of the probabilistic methodology, as in Bayes' theorem.

In this phase, we combine the trust rate computed by the corresponding pre-eminent node and the mutually agreed pre-eminent node, in order to arrive at a genuine trust rate. This can be given by the following.

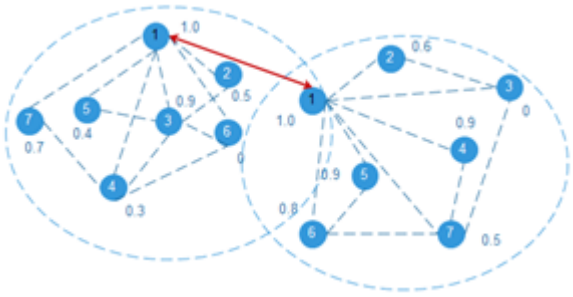

Figure 4: Intra-cluster trust computation

A node can either be trustworthy or untrustworthy and there is special case in which the node is either trustworthy or untrustworthy [**?**]. This can be written as

$$x : \Omega = \{T, \bar{T}\}. \tag{7}$$

T represents that the node x is trustworthy and $\bar{T}$ indicates that the node is untrustworthy. This could be explained more by the hypothesis, as shown below.

$$H = \{T\} \tag{8}$$
$$\bar{H} = \bar{T} \tag{9}$$
$$U = \Omega \tag{10}$$

In Equation (8), the equation represents that the node is trustworthy. The node is untrustworthy and is given in Equation (9). Equation (10) indicates that the node is either trustworthy or untrustworthy. Suppose, the probability of trustworthiness of a node is given by μ, then

$$\begin{cases} S1(H) & = \mu \\ S1(\bar{H}) & = 0 \\ S1(U) & = 1 - \mu \end{cases} \tag{11}$$

The probability function of un-trustworthiness of a node is given by Equation (12).

$$\begin{cases} S1(H) & = 0 \\ S1(\bar{H}) & = \mu \\ S1(U) & = 1 - \mu \end{cases} \tag{12}$$

In the next step, we combine the trust rate computed by the native pre-eminent node of a cluster along with the trust rate computed by the mutually agreed pre-eminent node. In this case, there is no need to check for the trustworthiness of pre-eminent node, as the node with highest trust rate is selected as the pre-eminent node, with faster recycling. The decisions made are combined and is provided below. There are three possible cases.

Case 1.

This is the case in which both the computations show that the node is trustworthy and it is given by

$$S1(H) \oplus S2(H)\frac{1}{w}[S1(H)S2(H) + S1(H)S2(U) + S1(U)S2(H)]. \tag{13}$$

Case 2.

This is the case in which both the computations show that the node is untrustworthy and it can be written as

$$S1(\bar{H}) \oplus S2(\bar{H})\frac{1}{w}[S1(\bar{H})S2(\bar{H}) + S1(\bar{H})S2(U) + S1(U)S2(\bar{H})]. \tag{14}$$

Case 3.

In this case the node can either be trustworthy or untrustworthy and it is represented as

$$S1(U) \oplus S2(U) = \frac{1}{w}S1(U)S2(U). \tag{15}$$

w in Equations (13)-(15) can be given by

$$\begin{aligned} w &= S1(H)S2(H) + S1(H)S2(U) + S1(U)S2(H) + S1(\bar{H})S2(\bar{H}) + S1(\bar{H})S2(U) \\ &\quad + S1(U)S2(\bar{H}) + S1(U)S2(U). \end{aligned} \tag{16}$$

The genuine trust rate falls between 0 and 1. If the node's trust rate is 1, then the node is completely trustworthy. In case, if the node's trust rate is 0.5, then the node is either trustworthy or untrustworthy and the final case is that if the node's trust rate is 0, then it indicates the fact that the node is untrustworthy. Accurate trust rates are obtained on combining the resultant trust rates of inter and intra cluster.

The trust rates obtained for all the nodes are fed into the Ant based clustering algorithm and is explained in the next section.

2.4 Ant Based Clustering Algorithm

Ant based clustering algorithm is known for its efficiency. In this work, the genuine trust rates obtained from the previous phase are fed as input. The trust rates are dispersed in the grid and the ant randomly chooses a trust rate. The ant moves over the grid and the trust rate is probabilistically placed only when the probability range is higher than the probability of trust rate placement. This process continues until all the trust rates are placed perfectly in cluster.

Algorithm 2: Ant based clustering algorithm

1: Input: Genuine trust rates
2: Output: Clustered outcome of trustworthy and untrustworthy nodes
3: Distribute genuine trust rates (gtr) in grid
4: Each ant chooses a gtr and place randomly in grid
5: Select each ant randomly and it moves randomly over grid
6: Ant probabilistically drops gtr over grid
7: Continue this process until all gtrs are placed in grid
8: End process

Table 4: Sample Cluster Data

Trustworthy Nodes	Untrustworthy Nodes	Either Trustworthy or Untrustworthy Nodes
A1, A3, A7	A4, A6	A2, A5
B1, B4, B5, B6	B3	B2, B7

The outcome of this algorithm is a list of trustworthy, untrustworthy and nodes that are either trustworthy or untrustworthy. The list of untrustworthy nodes is forwarded to the pre-eminent node and it handles the problem. While routing, the path with maximum number of trustworthy nodes alone is chosen, such that the packet delivery ratio is terrifically improved.

2.5 Intrusion Detection

After the result update with the list of trustworthy, untrustworthy and nodes that are either trustworthy or untrustworthy, the pre-eminent node is responsible for handling such attacks. Handling the security breaches is out of the scope of this work. The detected report is submitted and then the intrusion is handled by the recovery process.

This work effectively makes use of the trust rate and the objective of the system is achieved. Clustering of trustworthy, untrustworthy and nodes that are either trustworthy or untrustworthy is achieved by ant based clustering algorithm. A small concern of routing is considered, in which the path with maximum number of trustworthy nodes is chosen.

3 Experimental Analysis

The performance of this system is tested in two angles. Initially, we focus on routing. In this work, only a small concern of routing is considered. The path with several trustworthy nodes is chosen as the route for forwarding

Algorithm 3: Overall algorithm

// Pre-eminent node selection
Begin
$cn_{th} = 7$;
For every 5 seconds
Randomly select a node
Draw a circle that encloses 7 nodes
if ($cn < 7$)
Broadcast join request
For every cn of a cluster
Do
Compute $Tr_{rate} = pdr + b_b$
Find the ID of node with greatest Tr_{rate}
Declare it as pre-eminent node
End
//Inter-cluster trust rate computation
Pre-eminent node calculates trust rate for all constituent nodes
S1: $P(tr|pdr, b_b) = P(tr|pdr) \times P(tr|P(tr|b_b)$
// Intra-cluster trust rate computation
S2: Neighbor pre-eminent node calculates trust rate for all constituent nodes mutually (based on assumption)
Calculate genuine trust by combining S1 and S2
//Cluster trustworthy and untrustworthy nodes separately
Disperse genuine trust rates in the grid
Ant randomly chooses the trust rates and clusters them in the grid
Continue the process until all the genuine trusts are located
Send the report to the pre-eminent node
//Situation handling
Track the untrustworthy nodes and take necessary action

packets and the analysis is carried out based on speed variation and the number of nodes variation.

The routing aspect of this work is compared with TSR1, TSR2, TDSR and DSR [10]. TSR1 and TSR2 are proposed in the same work but TSR1 renders more attention towards control packets and TSR2 pays more attention on data packets. The performance metrics considered are packet delivery ratio, end to end delay, and throughput.

3.1 Experimental Setup

The network size of our system is chosen as $1000 \times 1000m^2$. Wireless bandwidth is the data rate of the connection and is measured by bits/second. In our proposed system, the wireless bandwidth is 2 MB/Sec. The transmission range of the mobile node is fixed as 100 meters. In wired networks, the destination node can receive the packet from the source, only if the destination node is in the transmission range of the source node. In case, if the destination node is not within the transmission range of the source node, then the packet is transmitted via some intermediate node. In our case, the mobile nodes act both as a node and a router.

The random waypoint mobility model is exploited in this work and it makes sense that a mobile node remains in a location for a certain period of time, which is termed as 'pause'. After the time gets expired, the mobile node starts to choose a destination and the speed. Then, the node traverse towards the destination node at the chosen speed and again it will get paused. The maximum speed of the node of this work is 10m/sec. The node pause time is set as 20 seconds.

We have employed NS-2 for simulation. Un-slotted carrier-sense multiple access protocol is employed and it avoids collision [4] for packet transmission. We distribute 30 nodes in a 1000 by 1000 meter. The transmission radius of all nodes for a hop is given as 250 meters.

3.2 Performance Analysis

Performance metrics such as Packet delivery ratio, end-to-end delay and throughput, are employed to evaluate the performance of our algorithm and the graphs of these performance metrics are presented from Figure 2 - Figure 7. The proposed work is compared with the existing works such as DSR, TDSR, TSR1 and TSR2. Among these works,

Table 5: Simulation Parameters

Simulation Parameters	Value
Simulation Time	250 sec
Dimension	$1000m \times 1000m$
Node count	35
Mobility Model	Random waypoint
Traffic Nature	Constant Bit Rate
Transmission Radius	250 m

TSR1 and FTDSR2 are proposed in the same work but with different characteristic features. TSR1 pays more attention towards control packets and FTDSR2 focuses on data packets. On result analysis, it is proved that control packets are more important than data packets in MANET.

The performance of the system is analysed in two different aspects. The first aspect of analysis focuses on routing and the second aspect of analysis concentrates on intrusion detection. We have conducted two different tests for routing by varying the speed and nodes.

Packet Delivery Ratio (PDR). PDR is the ratio of packets that are successfully sent to the destination node from the source node and are computed by Equation (17) and the results are presented in Figure 1.

$$PDR = \frac{\sum Number - of - Packets - Received)}{\sum Number - of - packets - sent} \times 100. \tag{17}$$

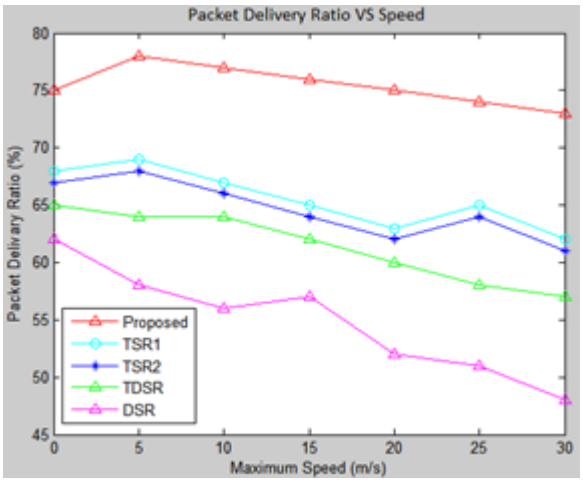

Figure 5: Packet delivery analysis vs. speed

Our system shows 75% packet delivery ratio with respect to speed and 80% packet delivery ratio with respect to number of nodes.

Average Latency (A_{ly}). The average time taken by the packets to reach the destination node from the source node and is calculated by

$$A_{ly} = \frac{\sum(Arrival\_Time - Sent\_Time)}{\sum Number\_of\_Connections} \tag{18}$$

Our proposed scheme proves least average latency and outperforms all the protocols. The end-to-end delay of our system starts from 16 seconds with respect to speed and 0.01 seconds with respect to node count.

Throughput. The amount of data transmitted per unit time and it is calculated by

$$Throughput = \frac{Size\_of\_the\_Packets(bits)}{Time\_Taken(sec)} \tag{19}$$

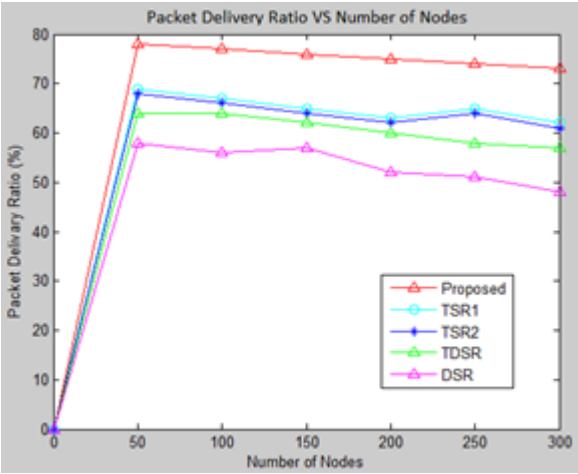

Figure 6: Packet delivery analysis vs. node count

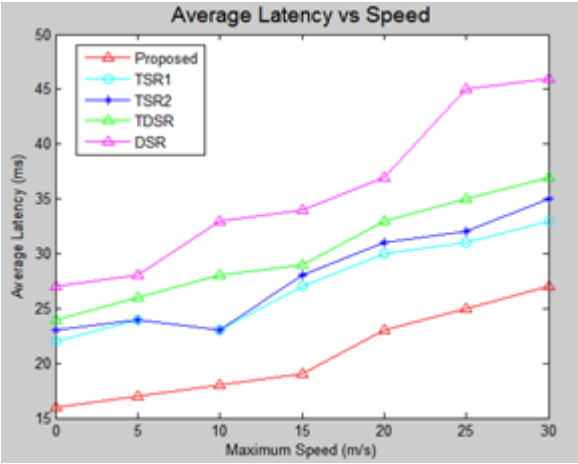

Figure 7: Average latency vs. speed analysis

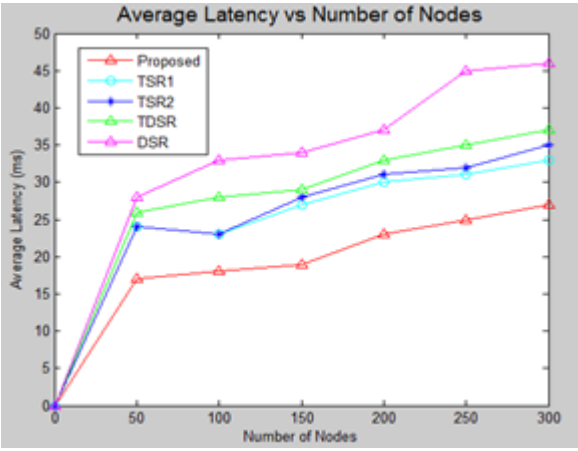

Figure 8: Average latency analysis vs. node count

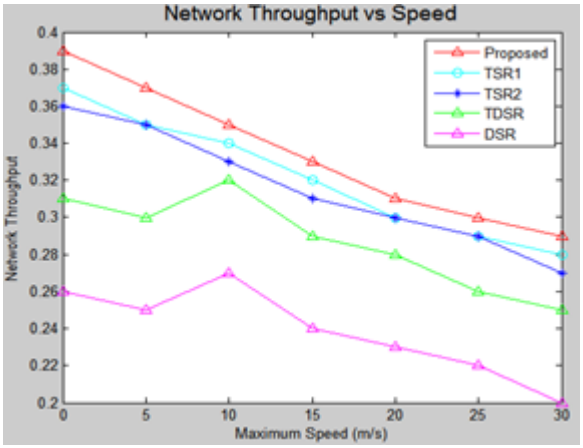

Figure 9: Throughput analysis vs. speed

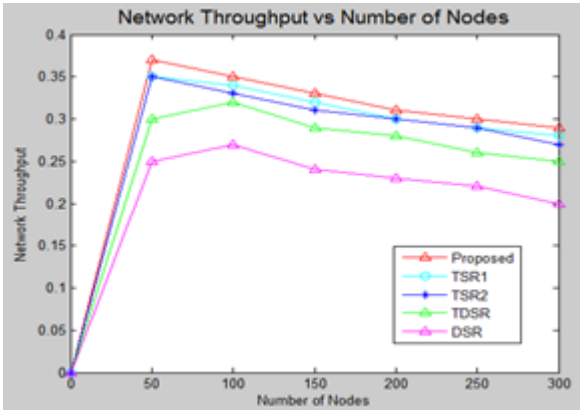

Figure 10: Throughput analysis vs. node count

Throughput is directly proportional to the packet delivery ratio and thus the throughput of the proposed work is higher than the existing works. The proposed work can transmit 0.39 packets per second with respect to speed and 0.37 packets per second with respect to node count.

Intrusion Detection Accuracy. It is the accuracy rate of intrusion detection and it can be calculated by

$$Intrusion\_detection\_accuracy = \frac{Number\_of\_correct\_detection\_of\_intrusion}{Number\_of\_intrusions\_detected} \tag{20}$$

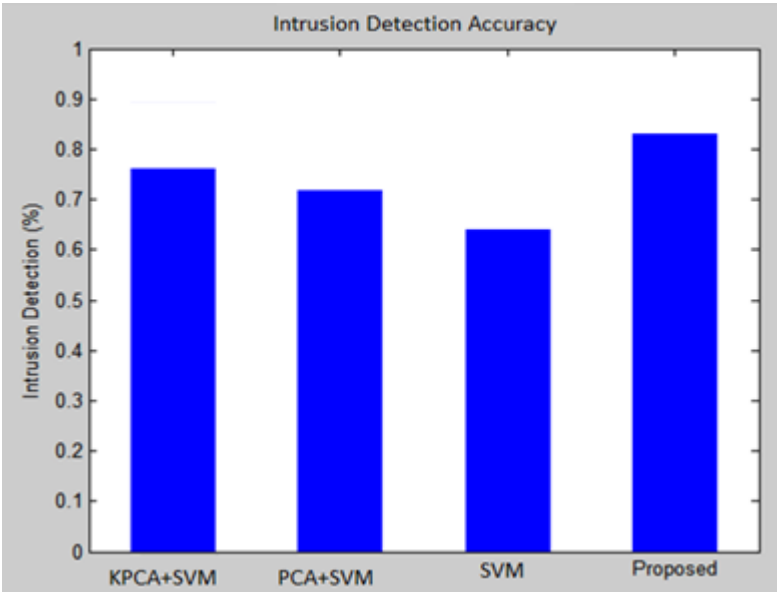

Figure 11: Intrusion detection accuracy

Intrusion detection accuracy of the proposed work is compared with the system that employs $KPCA+SVM$, $PCA+SVM$ and SVM alone. The detection accuracy of our system is 94.3% and is better than the compared systems.

Intrusion Error Rate. It is the wrong detection of intrusion out of the total intrusions detected and it can be calculated by

$$Intrusion\_detection\_accuracy = \frac{Number\_of\_wrong\_detection\_of\_intrusion}{Number\_of\_intrusions\_detected} \tag{21}$$

Intrusion error rate of our system is much lesser than the existing works such as $KPCA+SVM$, $PCA+SVM$ and SVM alone. The error rate of the proposed system is 5%, and is tolerable to some extent.

From the experimental results, it is evident that the intrusion detection system works well with lesser error rate.

4 Conclusion

In this work, we propose an IDS that is based on trust rates. The entire work of this system is compartmentalized into three phases and they are Pre-eminent node selection, Inter-cluster trust rate computation, Intra-cluster trust rate computation. Inter-cluster trust rate is computed by Bayes? theorem and the intra-cluster trust rate is calculated by the Dempster-Shafer theory. The computed genuine trust is fed into the ant based clustering algorithm. Finally, the proposed work is analysed in terms of routing and intrusion detecting potential. The system shows better accuracy rates for intrusion detection with minimum error rate. Trust rates are computed in two different ways and then, the computed trust rates are combined. This combined trust rate is named as the genuine trust rate and is fed into Ant based clustering algorithm. In future, the problem of computation overhead can be addressed.

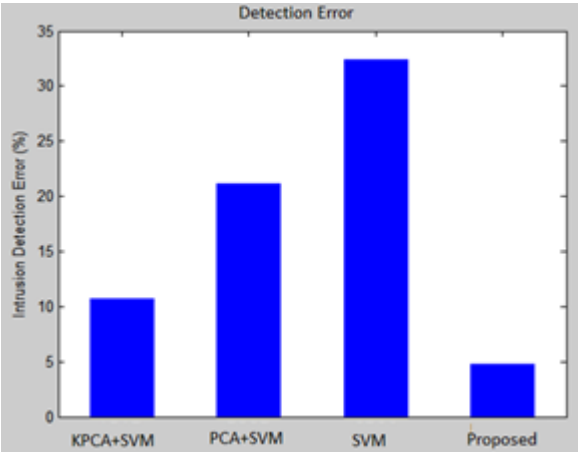

Figure 12: Detection error rate

References

[1] T. M. Chen and V. Venkataramanan, "Dempster-safer theory for intrusion detection in ad hoc networks", *IEEE Internet Computing*, vol. 9, pp. 35–41, 2005.

[2] A. P. Dempster, "Upper and lower probabilities induced by a multivalued mapping", *The Annals of Mathematical Statistics*, vol. 38, no. 2, pp. 325–339, 1967.

[3] G. G. Deverajan, R. Saravanan, "Selfish node detection based on evidence by trust authority and selfish replica allocation in MANET", *International Journal of Information and Communication Technology*, 2014.

[4] K. Govindan and P. Mohapatra, "Trust computations and trust dynamics in mobile adhoc networks: A survey," *IEEE Communications Surveys & Tutorials*, vol. 14, no. 2, pp. 279–298, 2012.

[5] J. Grover, "An Introduction to Bayes' theorem and Bayesian belief networks (BBN)", *Strategic Economic Decision-Making*, vol. 9, pp 1–9, 2013.

[6] S. Mutlu, G. Yilmaz, "Distributed cooperative trust based i ntrusion detection framework for MANETs", in *The Seventh International Conference on Networking and Services*, pp. 292–298, 2011.

[7] C. Panos, C. Xenakis, I. S. Stavrakakis, "A novel intrusion detection system for MANETS", *International Conference on Security and Cryptography*, pp.1-10, 2010.

[8] R. Sharman, S. Sharma, "Performance analysis of intrusion detection in MANET", *Computer Technology and Applications*, vol. 2, pp. 456–462, 2011.

[9] R. Shrestha, K. H. Han, D. Y. Choi, S. Jo Han, "A novel cross layer intrusion detection system in MANET", in *24th IEEE Conference on Advanced Information Networking and Applications*, pp. 647–654, 2010.

[10] Bo Wang, X. Chen, W. Chang, "A light-weight trust-based QoS routing algorithm for ad hoc networks", *Pervasive and Mobile Computing*, vol. 13, pp. 164–180, 2014.

[11] X. Wang, "An intrusion detection model based on ant principle", in *International Forum on Information Technology and Applications*, pp. 362–366, 2010.

Deverajan Ganesh Gopal working as Associate Professor in School of Computing Science and Engineering. He is an active researcher. He is a PhD student working under the supervision of Dr. R. Saravanan. His research areas include wireless networks, network security and cloud computing.

R. Saravanan completed his doctoral thesis in the area of Approximation Algorithms in 1997 at the Ramanujan Institute for Advanced Study in Mathematics and obtained the Ph.D degree from University of Madras. He obtained M.E in the branch of Computer Science and Engineering at the College of Engineering, Guindy, Anna University, Chennai. He has about two decades of teaching and research experience. He has rich research experience in areas of algorithms and published more than seventy five research papers. His areas of research include approximation algorithms, mobile computing, cryptography, and network security. He is a life member of Computer Society of India, Cryptology Research Society of India and Ramanujan Mathematical Society and also he is a member of IEEE. He served as a director during Aug 2010 - Jan 2013 and also as a dean during Feb 2013 - Jan 2014 at VIT University.