AUTHENTICATION USING TRUST TO DETECT MISBEHAVING NODES IN MOBILE AD HOC NETWORKS USING Q-LEARNING

S.Sivagurunathan[1,] K.Prathapchandran[2] and A.Thirumavalavan[3]
[1, 2] Department of Computer Science and Applications
Gandhigram Rural Institute-Deemed University, Gandhigram-624 302
Tamilnadu, India
[3]Department of Computer Science
Arignar Anna Government Arts College, Attur
Tamilnadu, India

ABSTRACT

Providing security in Mobile Ad Hoc Network is crucial problem due to its open shared wireless medium, multi-hop and dynamic nature, constrained resources, lack of administration and cooperation. Traditionally routing protocols are designed to cope with routing operation but in practice they may be affected by misbehaving nodes so that they try to disturb the normal routing operations by launching different attacks with the intention to minimize or collapse the overall network performance. Therefore detecting a trusted node means ensuring authentication and securing routing can be expected. In this article we have proposed a Trust and Q-learning based Security (TQS) model to detect the misbehaving nodes over Ad Hoc On Demand Distance-Vector (AODV) routing protocol. Here we avoid the misbehaving nodes by calculating an aggregated reward, based on the Q-learning mechanism by using their historical forwarding and responding behaviour by the way misbehaving nodes can be isolated.

KEYWORDS

Mobile Ad hoc Networks Security, Routing, AODV, Historical, Response, Trust, Authentication & Q-Learning

1. INTRODUCTION

In the advancement of wireless communications technology, Mobile Ad Hoc Networks also called MANET plays vital role in today's communication technology. It is a collection of mobile nodes that are connected together over a wireless medium. It is also called Independent Basic Service Set (IBSS). Because all participating devices in the network are operating as a peer to peer structure and there is no fixed backbone and all the devices are connected with other nodes directly [1]. Nodes can be deployed easily and they move randomly as they want, without the support of any centralized structure and irrespective of time, hence due this self-configuration, self-healing and self-optimization characteristics, they are also called as self-organized networks (SON) [2]. In this network each node plays a dual role such as ordinary node; to perform network operations and router; to forward packets, hence there is no specialized router for forwarding the packet. Nodes in the network can join and leave at any time leading to dynamic topology. Such a special characteristic makes the network eligible various applications. At the same time providing security in such environment is difficult due to distinct nature hence probability rate of failure is very high compared to traditional network.

DOI: 10.5121/ijnsa.2016.8304

Providing security in mobile ad hoc networks is a challenging task because in general wireless channel is accessible to all kinds of users and nodes are moving with relatively poor physical protection hence we cannot make judgment that which one is a legitimate node and which one is a normal node. In addition to, nodes are having limited resources therefore continuous utilization of resource leads to shut down of nodes or slowdown work progress therefore probability of attacks is high. Finally each node depends on other nodes for forwarding so that it expects cooperation from neighbours but achieving such cooperation is difficult due to dynamic nature of the network. Though cooperation is achieved, we cannot predict nodes really behave well or not. Hence we have to ensure authentication among the communication devices.

Authentication is one of the important security requirements in MANET and it is defined as *"the ability of a node to ensure the identity to the receiver* [3]*"*. Typically authentication is carried out in two ways. The first one is initial authentication, which means all the participating devices in the network are authentic at the time of initial network deployment so that such authentication mechanism is called pre-authentication. The next one is called post- authentication; means over a period of time; every node in the network should ensure the identity of participating nodes [4]. In this work we concentrate on post-authentication mechanism. Once authentication is achieved, remaining security requirements such as confidentiality, integrity and non-reputation [3] can be achieved easily. To achieve authentication, shared secret, Public Key Infrastructure [PKI], digital signature, digital certificate [5] are used but these techniques are centralized, pre-determined and depend on trusted third party, thereby increasing computation power, memory and consumption of communication bandwidth and battery power but MANET has limited resource constrains. To provide security with limited computational capabilities trust comes into existence because it offers less memory overhead, less transmission overhead and less bandwidth consumption[6]. Trust is a word which is originally derived from the social sciences. Trust is defined as "*one entity (trustor) is willing to depend on another entity (trustee)[7]*" or *"the trustor abandons control over the actions performed by the trustee [8]"*. According to ad hoc networks, trust could be defined as "*the reliability, timeliness, and integrity of message delivery to a node's intended next hop* [9]*"*. Typically trust can be evaluated based on direct and indirect means recommendation of others [10-12]. Direct means information gathering from one hop neighbours and indirect trust means information gathering from other than one hop neighbours. But both trust information exhibit the historical interactions of nodes with respect to each other.

In TQS, we make use of Q-learning algorithm in order to enrich the proposed model that is discussed in section 4. Q-Learning algorithm is proposed by Watkins in the year 1989[13]. The algorithm involves with an agent, states *s* and a set of actions per state *a*. The state of environment will change after receiving the action *a*. After executing an action in a specific state, the agent gets reward. The goal is to try to find an optimal policy that encourages the agent to obtain the total reward during the whole operation [14] [15] and based on the total reward decision will be taken. The algorithm is defined as,

$$Q(s, a) = r(s, a) + MAX_{a'} \gamma (Q(s', a'))$$

where *r(s, a)* is an immediate reward, γ is a discount factor that determines the importance of future rewards. The value of discount between *0* and *1* range, *s'* represents the new state after action *a*, *a'* represents the action in state *s'* and *a* and *s* represent the current state and action respectively.

The main objective of this paper is to ensure authentication and detect misbehaving nodes. It can be achieved by measuring the historical interactions of neighbouring nodes using direct trust evaluation mechanism. Based on the trust value every node gets a reward using q-learning algorithm. Finally the aggregated reward is used to detect the misbehaving nodes. The rest of the

paper is organized as follows. Section 2 discusses the related work, section 3 discuss the AODV attack model, section 4 discuss the proposed TQS Model, section 5 analysis the performance of proposed model using mathematical modelling, section 6 discuss the simulation results and finally section 7 concludes the paper.

2. RELATED WORK

In the last decade, there have been a number of research papers showing their interest towards trust based models for mobile ad hoc networks.

Charikleia Zouridaki et al. [16] proposed a byzantine robust trust establishment scheme to improve the reliability of packet forwarding by combining first-hand information and second hand information means recommendation from others. This work is the extension of their earlier work (Charikleia Zouridaki et al., 2006) which has the deficiency in identifying byzantine behaviour. Tao Jiang et.al [17], proposed an ant based adaptive trust evidence distribution in MANET based on swarm intelligence. This model is completely distributed and adaptive to mobility and trust evidence. This is presented in the form of certificates which is derived from the user's private key and public key. A.Boukerch et.al [18], proposed a trust based security for wireless ad hoc and sensor networks. The main contribution of this paper is to manage the trust and repudiation locally with minimal overhead in terms of extra messages and time delay. This model utilizes four components namely agent launcher; is responsible for generating and launching trust and repudiation assessors. Second, trust and repudiation assessors; is responsible for hosting trust and repudiation management for each host. Third, trust instruments; is responsible for message transmission between the transaction requester and receiver. Finally, repudiation certificate is responsible for local processing, periodically performed by the replica trust and repudiation of each node. Pedro B.Velloso et.al [19], proposed a trust management based on human based model and it integrates with scalable maturity to mitigate collude attack and improve the scalability. Here trust is evaluated based on the individual experience and recommendations. To exchange recommendations a special protocol called recommendation exchange protocol is used. The ultimate aim of utilizing maturity is to increase the efficiency of mobile nodes.

Anitha Vijayakumar et.al [20], proposed a self-adaptive trust based ABR protocol for MANET using Q-learning mechanism. Aim is to provide secure end to end routing among the mobile nodes. Trust evaluation is based on direct and indirect observation and q-learning algorithm. This is used to enable each node to adjust its route request forwarding rate according to its evaluated trust score. Sivagurunathan S et.al [21], proposed a light weight trust based security model to ensure authentication and detect the malicious nodes. In this model trust calculation is accomplished by combining the responses from its neighbors and depending upon the capacity of their work done, thereby ensures the authentication among the nodes by the way security can be achieved. This model is suitable for resource constrained devices.

Abort from the above the authors [22-25] proposed trust based secure routing for mobile ad hoc networks based on either direct or in direct trust and both trust establishment mechanisms.

3. AODV ATTACK MODEL

The proposed model is piggyback with AODV routing protocol hence the following section discusses the AODV routing functionality and execution of black hole attack in AODV. AODV is reactive or on-demand routing protocol the routes are created only when it is needed. There is no

need of periodic updates of routing tables compared with proactive or table driven protocol. The following section describes the working principle of AODV routing protocol.

3.1. Protocol Description

AODV creates routes only when it is needed, there is no need to maintain periodic updates of routing information. It is derived from DSDV and DSR and also called descendent of DSDV. From DSDV it inherits the route discovery, route maintenance and hop by hop routing, Sequence number from DSR. It supports both unicast and multicast transmission. In AODV each node in the network maintains the following fields in its routing table such as Destination address, Next hop address, Number of hops, Active neighbours, Destination sequence number and Life time. For providing effective and timely packet delivery AODV, uses Route discovery and Route Maintenance Phases.

Route Discovery Phase: Typically route discovery is achieved by flooding. To accomplish route discovery process two packets are used one is Route Request *(RREQ)* another one is Route Reply *(RPLY)*. When a node wishes to transmit a packet to a particular destination, first it checks the routing table entries. If it has a desired path to the destination, then it will forward the packet to next hop address otherwise it broadcasts route discovery process by *RREQ* packet to all its neighbouring nodes. The *RREQ* consist of Source address, Broadcast ID, Source Sequence number, Destination address, Destination Sequence number and the hop count. The combination of Source address and sequence numbers are used to uniquely identify the *RREQ*. Figure.1 illustrates the route discovery and route reply process.

Once the intermediate node received the *RREQ* packet, it compares the sequence number which is in the own routing table with *RREQ*'s sequence number. If the sequence number is less than *RREQ*'s sequence number then it rebroadcasts request to next neighbouring nodes otherwise it will set up a reverse path and stores the reverse path entry in its routing table. The reverse path entry consist of Source IP address, Source Sequence Number, Number of hops to source node, IP address of node from which *RREQ* is received. Subsequently the reserve path used for sending a *RPLY* to the node which sent *RREQ* previously. Route reply is carried out by unicast routing not flooding. Sequences numbers are used to determine the most recent entry in the routing table to avoid the routing loops. Likewise the route discovery process is carried out.

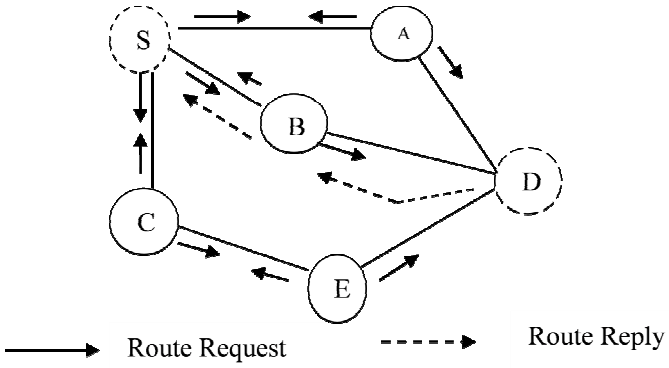

Figure 1. Route Request and Route Reply in AODV Routing

Route Maintenance Phase: Route maintenance is achieved by the Route Error (*RERR)* packet and also periodically propagating *HELLO* message to its neighboring nodes. The absence of *HELLO* messages from the receiving node depicts link failure so the source node again reinitiates

the route discovery process if the failure route is still in demand. When a node is unable to forward a packet to a particular destination it generates *RERR* to its predecessor node i.e. upstream nodes. So when a node receives *RERR* packet, it marks its own destination table as invalid and also sets the destination entry as infinity and deletes the particular route entry. Hence the source node reinitiates the route discovery process.

3.2. Black hole Attack in AODV Protocol

In AODV protocol, a node wants to send a packet to a particular destination, first it checks whether it has route to the destination in its routing table. If it has, simply use that route for relay the packet. Otherwise it initiates the route discovery process by using *RREQ* packet. Upon receiving the route request packet *RREQ* the intermediate nodes give response by sending the *RPLY* packet back to the source if they have desired route to the requested route request. According to the protocol specification, AODV protocol give response and send data packet to the first route reply from the neighboring nodes though it received multiple route replies. Here a black hole node takes this advantages and send a *RPLY* packet first without checking whether it has desired route or not to the destination. So that source node wrongly assumed that a black hole node has desired route to the destination by the way a black hole node can retain all the incoming data packets that intended to forward to the destination. But the source node could not able to know whether a data packet correctly reached the destination or not because lack of acknowledgement in AODV. The following Figure.2 shows the black hole attack model where *BH* is a black hole node it gives *RPLY* without checking whether it has desired route or not so that all the data packets that intended to forward is dropped.

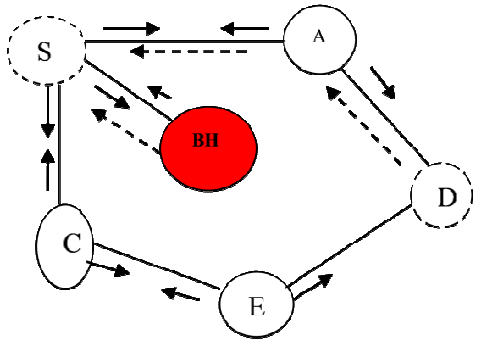

Figure 2. Black hole attack in AODV

4. TRUST AND Q-LEARNING BASED SECURITY MODEL

4.1. Assumptions

TQS model is based on the following assumptions. For simplicity we assume the network is small in size. All the nodes are behaving well at the time of initial network deployment since all the nodes are authentic and all the nodes are having well defined resources such as battery power, bandwidth and memory. Over the period of time, they may change their behavior and become black hole. We represent the misbehaving nodes as black hole nodes [26] where nodes try to drop every route request packets and always claiming a route to the requested node. Every node in the network maintains a table called TQtable where trust and reward values of their neighbor nodes can be stored. The structure of TQtable is shown in table 1. We also assumed that node's trust value as well as aggregated reward value as a continuous real numbers in the range 0 to 1 with

representation of 1 means completely trusted node, 0.5 means partially trusted node and 0 means untrusted misbehaving node. Figure. 3 shows the structure of the trust and Q-learning based model.

Table 1. TQtable of Node

Node ID	In	CP	DP	T	R	IR	MAX (R)	Γ	AR

where, ID – Node identity In – Number of Interactions, CP- Control Packet, DP- Data packet, T-Trust Value, R – Reward, IR – Immediate Reward, γ – Discount Factor and AR –Aggregated Reward

The TQS model consists of the following phases; trust computation, Aggregated reward computation and identifying and isolation of misbehaving nodes.

4.2. Trust Computation phase

As mentioned earlier, initially all the nodes are cooperating well. Over the period of time, a node wants to send a packet to a particular destination. According to our TQS model, initially all the nodes broadcast the *HELLO* packets instead of initiating route discovery process or checking their own routing table for desired route. So that every node ensures it's one hop neighboring nodes ultimately only one hop neighbors respond to the hello packets because they are in same communication range. From that every node can conclude how many nodes are staying as one hop neighbors. After that every node executes the trust evaluation mechanism on each of its neighboring nodes based on the following equation 1.

$$T_{AiAj}(n) = [CP_{AiAj}(n) + DP_{AiAj}(n)]/2 \quad \begin{cases} i=1 \\ n=1, 2, 3 \ldots \\ j=1, 2, 3\ldots \end{cases} \qquad (1)$$

where A_i denotes the evaluating node and A_j denotes evaluated node by A_i. *CP* denotes control packet (forwarding or responding ratio) and *DP* denotes data packets forwarding ratio over time with *n* number of interactions with the one hop neighboring nodes.

In AODV the following control packets are used. In route discovery, route request *(RREQ),* route reply *(RPLY)* packets are used. Route error *(RERR)* and *HELLO* packets are used in route maintenance process. While evaluating trust these packets are also considered because they provide significant contribution towards the routing operations. Though misbehaving nodes can also utilize such packets but utilizing probability of such packets are relatively low compared with well behaving nodes. Hence ratio of Control Packet forwarding (CP) or responding is calculated over the period of time based on the equation 2 with *n* interaction with the one hop neighboring nodes.

$$CP_{AiAj}(n) = (RREQ_{AiAj}(n) + RPLY_{AiAj}(n) + RERR_{AiAj}(n) + HELLO_{AiAj}(n))/4 \quad \begin{cases} i=1 \\ n=1,2,3\ldots \\ j=1, 2, 3\ldots \end{cases} \qquad (2)$$

The Data Packet (DP) forwarding ratio of each node is calculated ıe equation 3.

$$DP_{AiAj}(n) = NDF_{AiAj}(n) / NDR_{AiAj}(n) \quad \begin{cases} i=1 \\ n=1,2,3\ldots \\ j=1,2,3\ldots \end{cases} \quad (3)$$

where *NDF* denotes number of data packets actually forwarded and *NDR* denotes number of packets actually received over time with n number of interactions. Likewise every node could calculate the trust value of all its one hop neighbors and update its TQtable. Each node can monitor its neighboring nodes' forwarding behavior by using passive acknowledgment (Pirzada et al, 2006).

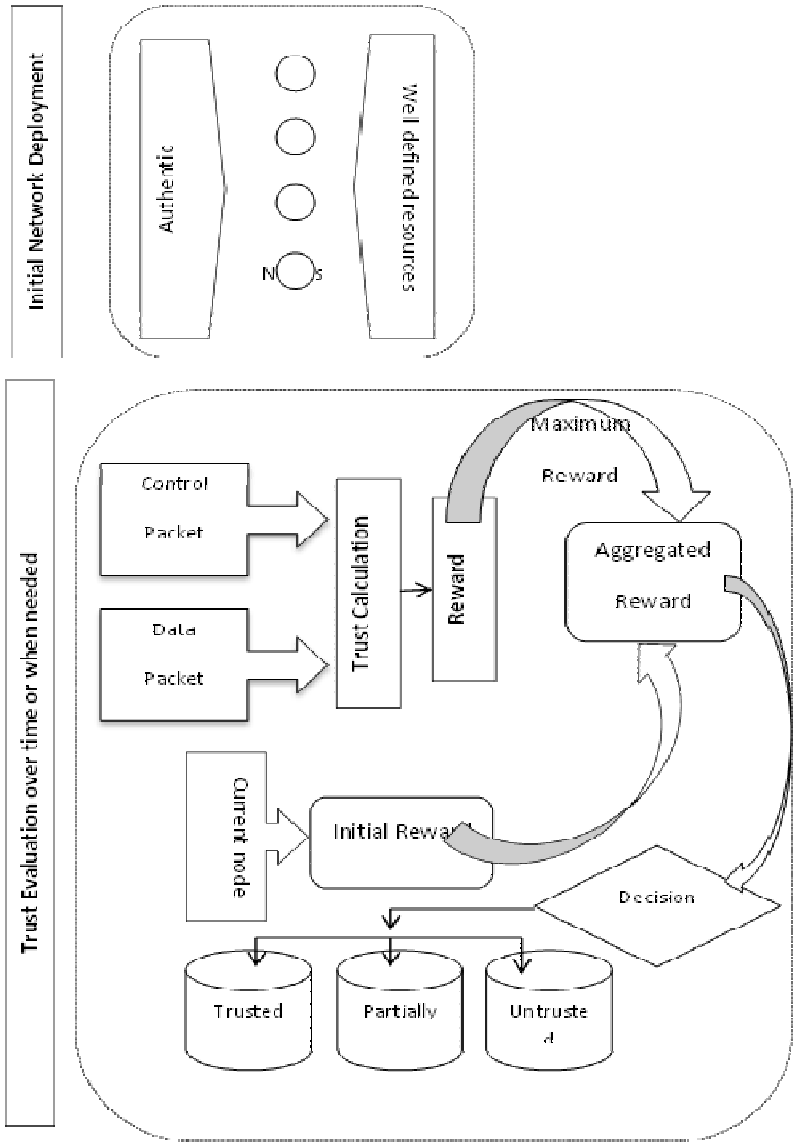

Figure 3. Trust and Q learning based Security model

4.3. Aggregated Reward Computation phase

After the trust computation phase, by using trust values obtained for every node, an evaluating node assigns a reward for each interaction that had with one hop neighboring nodes based on the threshold values. These threshold values can be changed according to the user specification. The reward value ranges between 0 and 1. 1 specifying the maximum, 0 specifying no reward and 0.5 specifying intermediate reward. The specification is given below.

$$\begin{aligned} &R=1 \text{ when } T_{AiAj}(t) >= TH1 \\ &R=0.5 \text{ when } T_{AiAj}(t) <= TH1 \text{ and } >=TH2 \\ &R= 0 \text{ when } T_{AiAj}(t) <=TH3 \end{aligned} \quad (4)$$

Thereafter evaluating node utilizes the Q-Learning algorithm to evaluate the overall performance of its neighbor nodes because a node can get high reward for some action and vice versa hence based on the equation 4 an evaluating node can get an aggregated reward means overall performance of its neighbors.

$$AR_{AiAj} = ([\ (IR)_{AiAj} + \gamma\ MAX(R)_{AiAj}(n)]/2\ )/n \quad (5)$$

where R represents the reward, IR denotes the immediate reward over time and AR denotes the aggregated reward. γ is a relative value and always>0. Immediate reward for all the neighbors is calculated based on its battery capacity, memory and bandwidth due to processing capabilities of each node these factors can change and also affect the overall network performance. Hence we consider each node's recent battery power, bandwidth and memory as immediate reward because they may change over time. So immediate reward can be calculated as,

$$(IR)_{AiAj} = (\text{Battery Capacity}_{AiAj} + \text{Memory}_{AiAj} + \text{Bandwidth}_{AiAj}\ )/3 \quad (6)$$

4.4. Identifying and isolation of misbehaving nodes

After the aggregated reward computation phase, now an evaluating node can take decision based on the aggregated reward of each of its neighboring nodes. This aggregated reward will be checked against the predetermined threshold value which is mentioned in table 2. Here the threshold values can also be changed according to the user specification.

Table 2. Threshold Table

Level	Threshold	Meaning
1	>= TH1	Trusted node
2	<= TH1 and >=TH2	Partially Trusted Node
3	<=TH3	Black hole node

These values are classified into three categories, first one is trusted nodes; we can allow those nodes in normal routing operation and data processing is actual data forwarding and receiving, second is partially trusted nodes; we allow those nodes to take part in the normal routing operation but they will not be involved in actual data processing. Finally, misbehaving nodes; black hole nodes those nodes are isolated from the network and information about the

misbehaving nodes can be broadcast by evaluating nodes before actual route discovery commences. Therefore those nodes are deleted from all the TQtables. By the way such nodes are isolated from route discovery process and therefore authentication is ensured by excluding those nodes. Every node can execute the TQS model over the period of time or when needed.

4.5. Algorithm for TQS model

The following algorithm explains the working flow of proposed TQS model.

Begins: TQS Model

Step 1: Initialize

Network is small scale in size; All the nodes are authentic and well defined resource constrains such as battery power, bandwidth and memory;
Node movements occur at random fashion;
n is represented as number of interactions where n=1,2,3…;
i is represented as evaluating node where i=1;
j is represented as evaluated node where j=1,2,3…;
CP denoted control packet forwarding or responding ratio;
DP denoted data packet forwarding ratio;
T is represented as trust;
NDF is represented as number of packet actually forwarded;
NDR is represented as number of packets actually received;
R denotes reward;
IR denotes immediate reward;
Th1, Th2 and Th3 is represented as pre-determined threshold;
γ is a relative value >0;
AR is represented as aggregated reward;

Step 2: Trust Computations

Each node evaluates the trust values (T) of its one hop neighbors over time or when needed, with n number of interactions.

$$T_{AiAj}(n) = [CP_{AiAj}(n) + DP_{AiAj}(n)]/2 \qquad \begin{cases} i=1 \\ n=1,\ 2,\ 3 \\ j=1,\ 2,\ 3\ldots \end{cases}$$

Control packet forwarding or responding ratio is calculated by,

$$CP_{AiAj}(n) = (RREQ_{AiAj}(n) + RPLY_{AiAj}(n) + RERR_{AiAj}(n) + HELLO_{AiAj}(n))/4 \qquad \begin{cases} i=1 \\ n=1,2,3\ldots \\ j=1,\ 2,\ 3\ldots \end{cases}$$

Data packet forwarding ratio is calculated by,

Step 3: Ag

$$DP_{AiAj}(n) = NDF_{AiAj}(n) / NDR_{AiAj}(n) \qquad \begin{cases} i=1 \\ n=1,2,3\ldots \\ j=1,\ 2,\ 3\ldots \end{cases}$$

Based on the trust value (T) an evaluating node assign a reward to evaluated node (j) for each interaction. The reward is given by,

R=1 when $T_{AiAj}(t) >= TH1$
R=0.5 when $T_{AiAj}(t) <= TH1$ and

Immediate reward is calculated by,

$$(IR)_{AiAj} = (\text{Battery Capacity}_{AiAj} + \text{Memory}_{AiAj} + \text{Bandwidth}_{AiAj}) / 3$$

Based on the immediate reward and reward an aggregated reward is calculated by,

$$AR_{AiAj} = [[(IR)_{AiAj} + \gamma\ MAX(R)_{AiAj}(n)]/2)/n$$

Step 4: Identifying and isolating misbehaving nodes
if (1<AR>=Th1) then
Trusted node; Allow for routing operation and actual data processing
else if(Th1>AR>=Th2) then
Partially trusted node; Allow only for routing activities
else (Th2<AR) then
Misbehaving nodes or black hole nodes; hence avoided and delete entry of such nodes from each node TQtable
End if
End if
End TQS Model

5. MATHEMATICAL ANALYSIS

In order to evaluate the proposed model, we make use of the following network structure which is shown in the Figure.4. The network consists of 5 nodes. S and D denotes source and destination nodes respectively and A, B and C are intermediate nodes. Over the period of time, according to our TQS model node S wants to transmit a packet to the destination D.

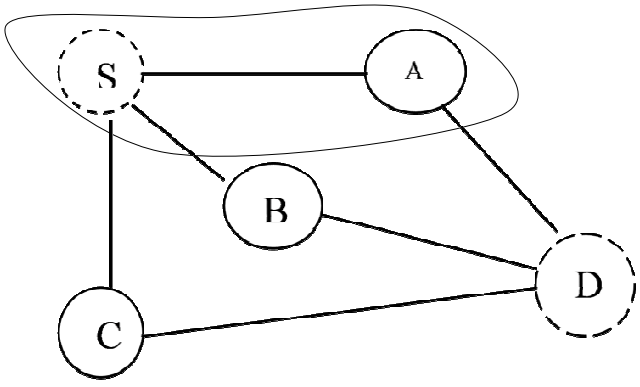

Figure 4.Example Network

Initially node S broadcast a *Hello* packet, so that node *A, B* and *C* can response the *Hello* packet because they are one hop neighbors. Now node *S* can assume three neighboring nodes are staying in touch with. Next it executes a trust evaluation mechanism based on the equation 1.

$$T_{AiAj}(n_i) = [CP_{AiAj}(n_i) + DP_{AiAj}(n_i)]/2$$

Hence, we assume node *S* evaluate a trust value of one of its neighboring node *A* overtime with number of interactions say 3 so it marked in the above figure. So the above equation is rewrite into,

$$T_{AiAj}(n1) = [CP_{SA}(n1) + DP_{SA}(n1)]/2 \text{ for } n=1$$

$$T_{AiAj}(n2) = [CP_{SA}(n2) + DP_{SA}(n2)]/2 \text{ for } n=2$$

$$T_{AiAj}(n3) = [CP_{SA}(n3) + DP_{SA}(n3)]/2 \text{ for } n=3$$

from the above equations, first each node evaluate the Control Packet (CP) forwarding and responding ratio for 3 interactions based on the equation 2 so that,

$$CP_{SA}(n1) = (RREQ_{SA}(n1) + RPLY_{SA}(n1) + RERR_{SA}(n1) + HELLO_{SA}(n1)) /4$$

$$CP_{SA}(n2) = (RREQ_{SA}(n2) + RPLY_{SA}(n2) + RERR_{SA}(n2) + HELLO_{SA}(n2))/4$$

$$CP_{SA}(n3) = (RREQ_{SA}(n3) + RPLY_{SA}(n3) + RERR_{SA}(n1) + HELLO_{SA}(n3))/4$$

Assume $RREQ_{SA}(n1) = 0.6$, $RPLY_{SA}(n1) = 0.6$, $RERR_{SA}(n1) = 0.0$ and $HELLO_{SA}(n1) = 0.5$.

Therefore,

$$CP_{SA}(n1) = (0.6 + 0.6 + 0.0 + 0.5) / 4 = 0.4$$

Likewise, assume $CP_{SA}(n2) = 0.7$ and $CP_{SA}(n3) = 0.9$

Then, Data Packet (DP) forwarding ratio based on the equation 3 with number of interactions say 3. Therefore,

$$DP_{SA}(n1) = NDF_{SA}(n1) / NDR_{SA}(n1)$$

$$DP_{SA}(n2) = NDF_{SA}(n2) / NDR_{SA}(n2)$$

$$DP_{SA}(n3) = NDF_{SA}(n3) / NDR_{SA}(n3)$$

Assume $NDF_{SA}(n1) = 0.7$ and $NDR_{SA}(n1) = 0.9$

Therefore,

$$DP_{SA}(n1) = 0.7/0.9 = 0.7$$

Likewise, assume DP_{SA} (n2) = 0.6 and DP_{SA} (n3) = 0.4.
The Control Packet (CP) and Data Packet (DP) forwarding and responding ratio values are substitute in equation 1. So that the answer will be,

$T_{SA}(n1)= [CP_{SA}(n1) + DP_{SA}(n1)]/2$
$= (0.4+0.7)/2$
$= 0.6$
$T_{SA}(n2)= [CP_{SA}(n2) + DP_{SA}(n2)]/2$
$= (0.7+0.6)/2$
$= 0.7$
$T_{SA}(n3)= [CP_{SA}(n3) + DP_{SA}(n3)]/2$
$= (0.9+0.4)/2$
$= 0.7$

After that node *S* assign reward for node *A* for each interactions based on equation 4. Here we assume the threshold values TH1=0.7, TH2=0.6 and TH3=0.4 so that, $T_{SA}(n1)$= the reward is 0.5, $T_{SA}(n2)$= the reward is 1 and $T_{SA}(n3)$= the reward is 1.
Next node *S* calculates the Immediate Reward (IR) based on equation 6 so that,

$$(IR)_{SA} = (\text{Battery Capacity}_{SA} + \text{Memory}_{SA} + \text{Bandwidth}_{SA}) /3$$

Assume, Battery Capacity_{SA} = 0.7, Memory = 0.9 and Bandwidth_{SA} = 0.6. For reconciled with all means, we consider the above factors also fall between (0.0 - 0.1) and for experimental results also we followed that will discuss in next section.
Therefore,

$(IR)_{SA} = (0.7+0.9+0.6)/3$
$= 0.7$

Finally node *S* evaluates overall reward of node *A* based on the equation 5.
Hence,

$$AR_{SA} =[(IR)_{SA} + \gamma\, MAX(R)_{SA}(n)]/2$$
$= (0.7 + 0.7(1))/2$
$=0.7$

Table 3.TQtable of Node *S*

Nodes	**In**	**CP**	**DP**	**T**	**R**	**IR**	**MAX (R)**	**γ**	**AR**
A	n1	0.4	0.7	0.6	0.5	0.7	1	0.7	0.7
	n2	0.7	0.6	0.7	1				
	n3	0.9	0.4	0.7	1				
B	n1	0.5	0.6	0.55	0.5	0.8	1	0.7	.75
	n2	0.9	0.0	0.45	0.5				
	n3	0.7	0.8	0.7	1				
C	n1	0.2	0.3	0.25	0	0.8	0	0.7	0.4
	n2	0.4	0.3	0.35	0				
	n3	0.5	0.1	0.30	0				

In the above calculation γ denotes the relative value we assume γ =0.7. So the aggregated trust value of node A which is evaluated by node S is, 0.7. By using this aggregated trust value a node can be classified into trusted, partially trusted and black hole node based on the threshold values. Likewise node S calculates the aggregated reward for node B and node C. The overall TQtable of node *S* is presents in the table 3.

Likewise all the nodes in the network can able to calculate the aggregated trust value of its one hop neighbors so that they can easily identify the black hole nodes. According to the threshold value node *A and B* are trusted node and *C* is untrusted nodes means black hole node hence it will be avoided from further communication.

6. EXPERIMENTAL RESULTS

The TQS model is implemented in Network Simulator 3(NS3). The study area is 500mx1000m for simulation with random way point mobility model. The number of nodes involved for simulation is 50. The following table 4 gives illustrate the simulation parameters. The aim of the simulation experiment is to identify and isolate the misbehaving nodes hence we chose the black hole nodes in a random fashion and include in the network to validate the performance of TQS model. We also set source and destination in a random fashion. We increase the number of black hole nodes step by step and run the experiment. According to the algorithm specification, we have taken four numbers of interactions; hence we run the simulation for four times with varying number of malicious nodes for analysis over the period of time. We have done the following experiments,

Table 4. Simulation Parameters

System Parameters	**Values Utilized**
Number of Mobile Nodes	50
% of Black hole nodes	25%, 50%, 75%
Mobility Model	Random Way point Mobility
Simulation Duration	100 Sec
Time interval	.5 Sec
Simulation Size	500mx1000m
Routing Protocol	Ad Hoc on Demand Distance Vector
Data rate	3072bps
Packet Size	64 Bytes
Wi-Fi Ad Hoc	802.11b
Data Traffic	UDP
Maximum Node Speed	20m/s
Node Pause	0s
Transmission Range	7.5dbm
Threshold Value1	>=5.5
Threshold Value2	<=5.5.>4.0
Threshold Value3	<=4.0

1. Our aim is to identify the black hole nodes, so that it is necessary to know the impact of black hole nodes. In this regard, we include the black hole nodes by increasing percentage and observed the packet dropping ratio.

2. Include the black hole nodes with increasing number in normal AODV routing protocol and identify them by using TQS model.

3. Comparing the packet delivery ratio, packet dropped ratio and end to end delay of AODV and TQS

Experiment 1: Observe the packet dropping ratio by increasing the number of black hole nodes. The Figure. 5 shows when number of black hole nodes increases, packet dropping ratio is also increased proportionally.

Experiment 2: The ultimate aim of the TQS model is to identify the misbehaving nodes. Hence we increase the black hole nodes by 25%, 50% and 75% respectively to assess the performance of TQS model. The following Figures 6, 7 and 8 depict the performance of TQS model in the presence of increasing black hole nodes. According to the aggregated reward from the TQS model, the detection of misbehaving nodes can be executed, which is shown in the figures. From the Figures 6, 7 and 8 circle surrounded by numbers denotes the node's identity and numbers from 0.0 to 1.0 which is shown vertically i.e the aggregated reward. We observe that as the number of black hole nodes has increased, the detection ratio of TQS model is also increased proportionally. According to the threshold values, which are shown in the simulation parameter table, aggregated reward less than or equal to 4 will be treated as misbehaving nodes. So those nodes will be avoided prior to next communication by the way authentication can be achieved. Next, aggregated reward between 4 and 5.5 will be treated as partially trusted nodes. Finally aggregated reward over 5.5 will be treated as fully trusted nodes.

Experiment 3: Performance analysis of TQS model over AODV is done in order to compare the TQS model with AODV routing protocol, the following performance like metrics packet dropping ratio, packet delivery ratio and end to end delay evaluated.

Packet dropped ratio: This metrics is calculated by difference between the total number of packets actually sent and the total number of packets actually received during the simulation. Hence the Figure 9 clearly shows that packet dropping ratio of TQS model is relatively low compared with AODV routing protocol. During the simulation the misbehaving nodes could be isolated by using TQS model so that TQS shows better results over AODV.

Packet delivery ratio: This metric analyses the packet delivery ratio of each node as well as overall network. It is measured by number of packets actually received divided by number of packets actually sent. Figure 10 depicts the packet delivery ratio of TQS very high over AODV because as mentioned earlier, black hole nodes are isolated from the network. Since they will not be involved in routing operation.

End to End delay: It is measured by the average time taken by a packet from the source to the destination. Hence it is calculated by difference between the arrival times and sending time of packets from the source to the destination and the results will be divided by total number of connections between the sources to the destinations. The Figure 11 shows the end to end delay of TQS model is low compared with AODV. From the above performance metrics, it is clear that the TQS model is better compared with normal AODV routing.
.

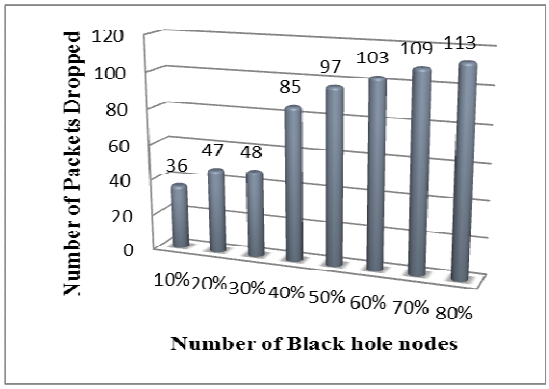

Figure 5.Packet dropped ratio with number of black hole nodes

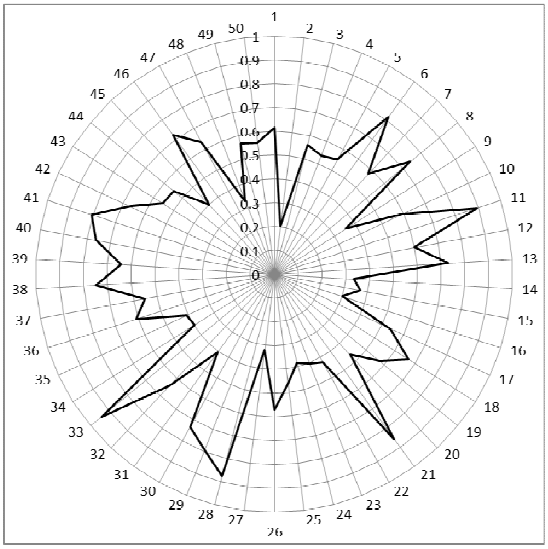

Figure 6. Detection ratio of malicious nodes under 25% of black hole nodes

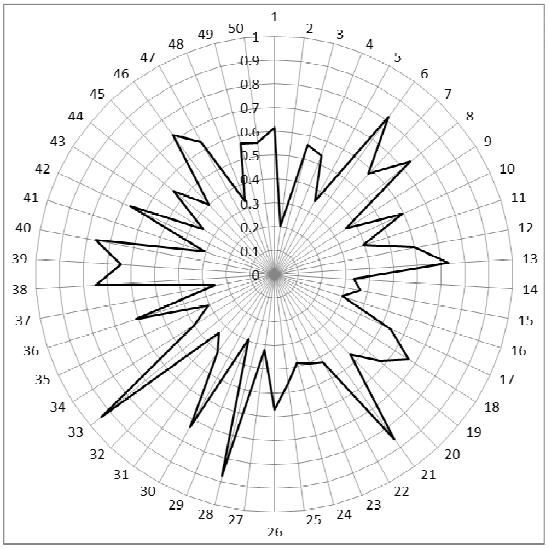

Figure 7. Detection ratio of malicious nodes under 50% of black hole nodes

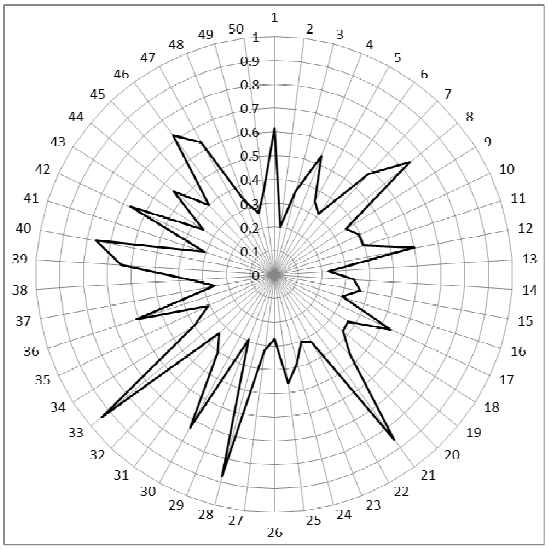

Figure 8. Detection ratio of malicious nodes under 75% of black hole nodes

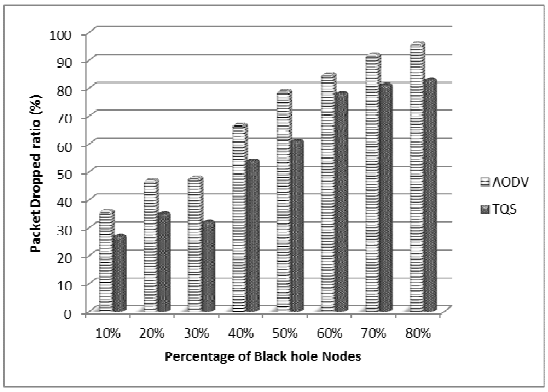

Figure 9. Packet Dropped Ratio, TQS vs AODV

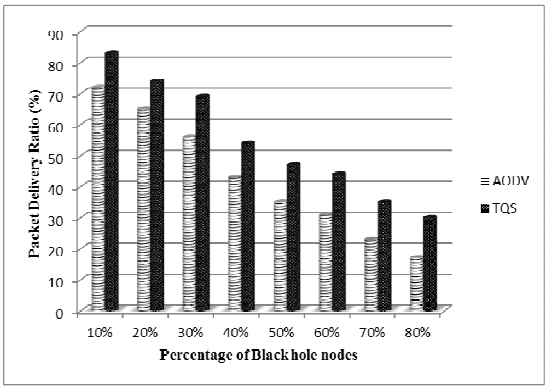

Figure 10. Packet Delivery Ratio, TQS vs AODV

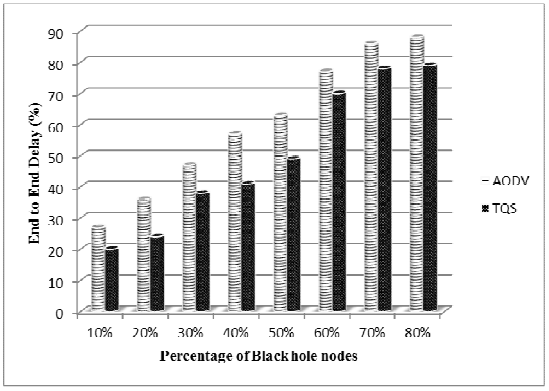

Figure 11. End to End Delay TQS vs AODV

7. CONCLUSIONS

The success rate of MANET application depends on how its security is defined. As applications of MANET have increases, weakness in security of MANET is also increases proportionally. Authentication is a primary security requirement in mobile ad hoc networks to ensure the correct identity. We make use of each and every control packets in AODV to calculate the trust evaluation because we cannot say all the packets are well processed in this distinct environment. In addition to, we utilize the data packet forwarding ratio for each node; it helps to assess the behavior of each node. We do not focus on indirect trust, because it always leads to processing overhead. Q-learning algorithm is used to enrich the trust calculation by giving immediate reward for each node by assessing their current resources and maximum reward during the interactions. It helps to make correct decision on nodes by the way authentication is achieved and black hole nodes can be detected.

8. ACKNOWLEDGEMENT

This research work is supported by University Grant Commission, India, through a Major Research Project, Grant (UGC.F.No: 42-128/2013 (SR)).

REFERENCES

[1] Lee Barken, Eric Bermel, John Eder, Matthew Fanady, Alan Koebrick, Michael Mee & Marc Palumbo, (2004) "Wireless Hacking projects for Wi-Fi Enthusiasts, Sungress. [1]
[2] Pedro B.Velloso, Rafael P.Laufer, Daniel de o.Cunha, otta Carlos M.B.Duarte & Guy Pujolle, (2010) "Trust Management in Mobile Ad Hoc Networks Using a Scalable Maturity based Model", IEEE Transactions on Network and Service Management, Vol.7, No.3. [2]
[3] William Stallings,(2003) "Cryptography and Network Security", Pearson Education.
[4] Y.Xiao, X.Shen & D.Z.Du, (2007) "Wireless Network Security", Springer.
[5] Sivagurunthan .S & Prathapchandran. K, (2014) "Trust based Security schemes in Mobile Ad Hoc Networks – A Review" 978-1-4799-3966-4/14, DOI 10.1109/ICICA.2014.67, IEEE.
[6] Xiaoyong Li, Feng Zhou & Junping Du,v(2013) "LDTS: A Lightweight and Dependable trust system for Clustered Wireless Sensor System", IEEE Transactions on information forensics and security, Vol.8, No.6.
[7] R C Mayer, J H Davis & F D Schoorman, (1995) "An Integrative Model of Organizational Trust-Academy of Management Review", vol. 20 (3), pp. 709-734.
[8] Bamberger and Walter, (2010) "Interpersonal Trust – Attempt of a Definition", Scientific Report.
[9] Liu z, JoyA.W & Thompson R A, (2004) "A dynamic trust model for mobile ad hoc networks", Proceeding of the 10 th IEEE International Workshop on Future trends of distributes computing systems, pp-80-85.
[10] A.Boukerch & K.EL-Khatib, (2007) "Trust based Security for Wireless ad hoc and Sensor networks", Computer Communications, vol.30, pp.2413-2424.
[11] Feng Zhang, Zhi-Ping Jia, Hui Xia, Xin Li & H.M. Sha Edwin, (2010) "Node trust evaluation in mobile ad hoc networks based on multi-dimensional fuzzy and Markov SCGM(1,1) model", Computer Communications, vol.35, pp.589-596.
[12] Mentari Djatmiko, Roksana Boreli, Aruna Seneviratne & Sebastian Ries, (2013) "Resources aware trusted node selection for content distribution in mobile ad hoc networks", Wireless networks, vol.19, pp.843-856.
[13] Q-Learning Algorithm, Retrieved on January, 12, 2015 from http://www.wikipedia.com
[14] Li X & Wu J, (2006) "Improve searching by reinforcement learning in unstructured P2Ps," International Conference on Distributed Computing Systems Workshops, pp.1-6.
[15] Wang Q & Zhan Z, (2011) "Reinforcement Learning Model, algorithms and its Applications", International Conference on Mechatronic Science, Electric Engineering and Computer, pp.1143-1146.
[16] Charikleia Zouridaki, Brian L.Mark, Marek Hejmo & Roshan K.Thomas, (2009) "E-Hermes: A robust cooperative trust establishment scheme for mobile ad hoc networks", Ad hoc Networks, vol.7, pp.1156-1168.
[17] Tao Jiang & John S.Baras, (2004) "Ant-based Adaptive Trust Evidence Distribution in MANET", Proceedings of the 24 th International Conference on Distributed Computing System Workshops, IEEE.
[18] A.Boukerch & K.EL-Khatib, (2007) "Trust based Security for Wireless ad hoc and Sensor networks", Computer Communications, vol.30, pp.2413-2424.
[19] Pedro B.Velloso, Rafael P.Laufer, Daniel De o.Cunha,oot carlos M.B.Duarte & Guy Pujolle, (2010) "Trust Management in Mobile Ad Hoc Networks Using Scalable Maturity Based Model", IEEE Transactions on Network and Service Management, Vol.7, No.3.
[20] Anitha Vijaya kumar & Akilandeswari Jeyapal, (2014) "Self Adaptive Trust Based ABR Protocol for MANET using Q-Learning", The Scientific World Journal.
[21] Sivagurunathan. S & Prathapchandran.K, (2015) "A Light weight Trust based Security model for Mobile Ad Hoc Networks", Proceedings of the International Conference on Electrical, Instrumentations and Communication Engineering- Recent Trends and Research Issues, 2015.

[22] Srinivasan A, Teitelbaum J & Wu J (2006) "DRBTS: distributed reputation based beacon trust system", In: Proceddings of the 2nd IEEE International Symposium on Dependable, Autonomic and Secure Computing, pp.277–283.
[23] Arya M & Jain Y K, (2011) "Grayhole attack and prevention in mobile ad hoc network",International Journal of Computer Applications, Vol. 27, No.10.
[24] Ameza F, Assam N & Beghdad R, (2010) "Defending AODV routing protocol against the black hole attack" International Journal of Computer Science and Information Security, Vol. 8, No.2,pp. 112–117
[25] Manikandan S P &Manimegalai R (2013), "Trust based routing to mitigate black hole attack in MANET" Life Science Journal, Vol.10,No.4,pp. 490–498 .
[26] M.Mohanapriya & Ilango Krishnamurthi, (2013) "Trust Based DSR Routing Protocol for Mitigating Cooperative Black Hole Attacks in Ad Hoc Networks", Arabian Journal of Science and Engineering, Vol.39, no.3, pp.1825-1833

BIOGRAPHY

Dr.S.Sivagurunathan is an Assistant Professor in the Department of Computer Science and Applications, Gandhigram Rural Institute-Deemed University, Gandhigram, Tamilnadu, India. He received his B.Sc degree in Physics from Madurai Kamaraj University in the year 1995 and the M.C.A degree in Computer Applications and M.Phil degree in Computer Science from Madurai Kamaraj University, Madurai in the year 1998 and 2004 respectively. He received his Ph.D. degree in Network Security from Anna University in the year 2010. He has seventeen years of experience in teaching. He is a life member of Computer Society of India (CSI). He has more than twenty publications in reputed Journals and Conferences and four publications in as book chapters. His areas of interest are Computer Networks, Mobile Ad hoc Networks, Network Security, Cloud Computing and Internet of Things. (E-Mail:svgrnth@gmail.com)

Mr.K.Prathapchandran is a Research Scholar in the Department of Computer Science and Applications, Gandhigram Rural Institute-Deemed University, Gandhigram, Tamilnadu, India. He received his B.C.A degree in Computer Applications from Madurai Kamaraj University in the year 2005, the M.C.A degree from Gandhigram Rural Institute – Deemed University in the year 2008 and the M.Phil degree in Computer Science from Bharathidasan University in the year 2010. He has two years of experience in teaching. He has fifteen publications in reputed Journals as well as conference proceedings. His areas of interest are Computer Networks, Mobile Ad Hoc Networks, Internet of Things (IoT) and Network Security.(E-Mail:kprathapchandran@gmail.com)

Mr.A.Thirumavalavan is an M.Phil Scholar in the Department of Computer Science, Arignar Anna Government Arts College, Attur, Tamilnadu, India. He received his B.Sc degree in Computer Science from Arignar Anna Government Arts College in the year 2006 and M.C.A degree in Computer Applications from Periyar University in the year 2009. He has two years of teaching experience. His areas of interest are Computer Networks, Mobile Ad Hoc Networks, and Network Security. (E-Mail:thiruma.atr@gmail.com)