International Journal of Applied Engineering Research
ISSN 0973-4562 Volume 10, Number 3 (2015) pp. 6151-6162

http://www.ripublication.com

An Enhanced Cross Layer Selfish node Detection Scheme to avoid False Negative Case in MANET

Mr. J.Govindarajan

Asst. Prof., CSE Dept., Amrita School of Engineering,
Amrita Vishwa Vidyapeetham University,
Coimbatore, Tamil Nadu, India.
+919942696870 j_govindarajan@cb.amrita.edu

ABSTRACT

Cooperation between the nodes is the main idea behind the MANET design. The presence of partial cooperative and non-cooperative nodes inside the network significantly reduces the performance of other nodes inside the network. Cross layer misbehavior detection is an approach to improve the detection efficiency with help of MAC and Network layer information. However, the existing detection schemes have not addressed the false negative case. The main aim of this work is to propose a modification to avoid false negative case in cross layer based misbehavior detection.

Keywords MANET, Cooperation, Cross Layer, False Negative case

1. INTRODUCTION

An ad-hoc network provides a dynamic network architecture, where end user mobile nodes are involved in the communication service without any infrastructure. Hence, the responsibilities of the networking devices like routing and forwarding are taken by end user mobile nodes. The nodes which are involved in routing and forwarding are called as cooperative nodes and the nodes which fail to involve are called as non-cooperative. Non-Cooperative nodes are also called as misbehavior nodes or selfish nodes. In general, the mobile nodes are constrained by the limited resources like power and processing time. A node shows its selfish behavior when it is not ready to share its resources with other nodes. Some cases of network layer misbehaviors are dropping of RREQ (Route Request) and

RREP (Route Reply) messages, delaying of RREQ and RREP messages, dropping of data packets and forwarding only small size packets.

The presence of selfish nodes inside MANET leads to the performance degradation like reduced throughput, high end-to-end delay and blocking of communication. To improve the network performance, we need to incorporate detection and punishment procedures in all the nodes. Misbehavior detection is the process of designating each node of the network, either as normal node or misbehavior node. The information about transmission and forwarding of each node are the basis for this classification. After the detection, punishment procedure must be invoked to protect the normal nodes from the misbehaved nodes.

2. RELATED WORK

To summarize the existing research contributions for selfish node detection, two types of classifications can be considered. In first classification, schemes are classified based on its objective towards handling the selfish node i.e. the schemes motivates the misbehaving node to participate in the forwarding process or identifying misbehaving nodes and isolated from the network. In the second classification, schemes are classified based on utilization of layers for the detection process i.e. schemes which utilizes only the specific layer information and schemes which utilizes information from two or more layer.

Virtual currency (also called as Nuglet) [5] is a detection scheme to motivate the nodes to involve in forwarding process. Virtual currency is analogy to our real-world currency i.e. A node, who earns the currency from providing "forwarding service" to other node, can get the "forwarding service" from other nodes through payment. Some of the schemes which are designed with motivation of isolating the selfish nodes are Watchdog [6], pathrater [7], Packet Conservation Monitoring Algorithm [8] and modification of routing table which is proposed PankajSharma [9]. In [6], a node after forwarding a packet to the next hop enters into watchdog state to monitor the forwarding action of its next hop. The pathrater [7] will help in choosing the most trusted path to the destination. In general, neighbor nodes which are sending or receiving packets to or from the specified node, only can give correct information about the node. This idea is used in Packet Conservation Monitoring Algorithm (PCMA) [8] to optimize the detection process. Security aspect of misbehavior is addressed by Pankaj Sharma [9]. He designed a secured AODV, called Trust AODV by incorporating trust model over AODV.

The above discussed schemes are based only on network layer information. Cross layer design is incorporated by Murugan et.al, in [11] and Prof. Rekha Patil et.al, in [12]. In [11], network layer and MAC layer information are collected and processed to detect MAC layer attack, packet dropping and packet modification misbehaviors. In [12], the author has proposed a solution to detect control packet (RREQ and RREP) dropping misbehavior.

3. EXISTING SCHEME

The combined cross layer scheme proposed in [11] is considered as existing scheme for our work. The "packet dropping" misbehavior detection of the scheme is discussed here. The scheme is based on the parameter 'trust'. Trust can be considered as belief of a node about other node. Each node maintains a trust value for each of its neighbor in its table. Initially each node has an equal belief over its neighbors i.e. a constant Trust value (TV) is given to all neighbors.

$$TV(X) = c \quad \text{----------} \qquad (1)$$

Where, c is constant and X is neighbor node

After receiving each packet from the neighbor 'X', the receiving node increments Trust value of X by 1.

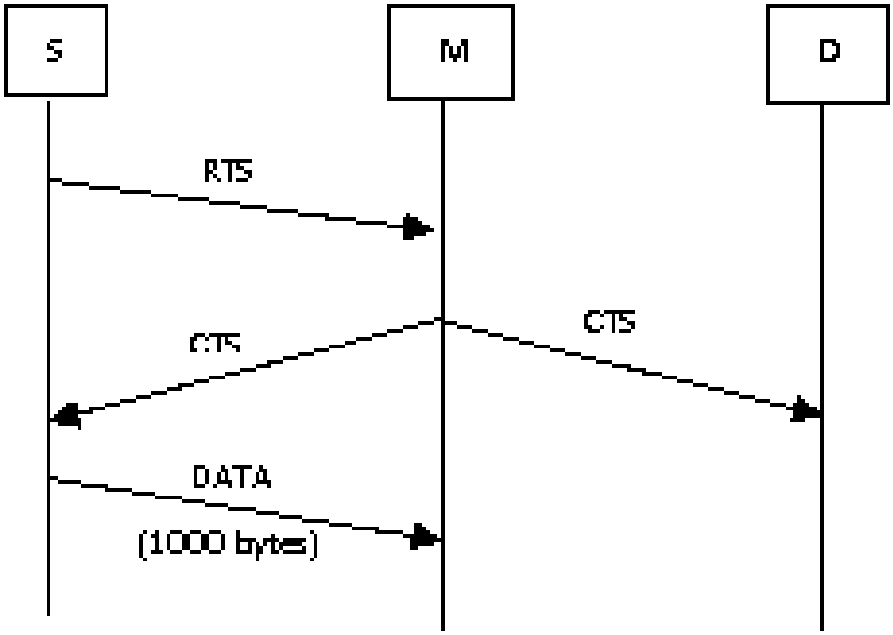
$$TV(X) = TV(X) + 1 \quad \text{----------} \qquad (2)$$

The updation of trust value is a continuous process and it is verified periodically with threshold value. The neighbor whose Trust value (TV) is less than the threshold is considered as selfish node.

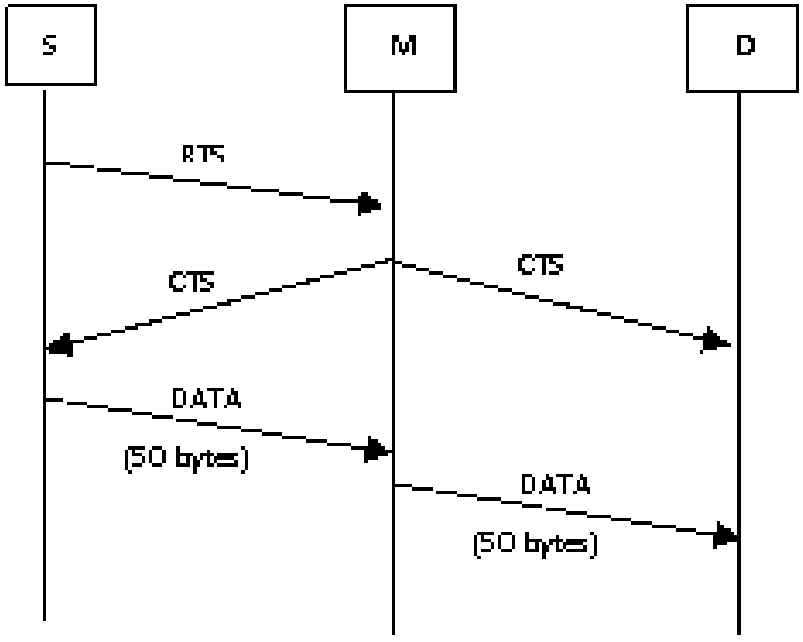
4. FALSE NEGATIVE CASE IN THE EXISTING SCHEME

In the Existing Scheme, the trust increment rule (Equation 1) is based on verifying the forwarding action. Forwarding action is the node's participation in terms of forwarding the packet to the desired destination. However, the scheme fails to analyze the forwarding behavior before incrementing the trust value of the forwarding node. Let us consider cases where a node forwards only smaller size data packets or control packets like ACK packets as shown in figure 1. This can be named as a false negative case in which the misbehaving node is considered as a well behaved.

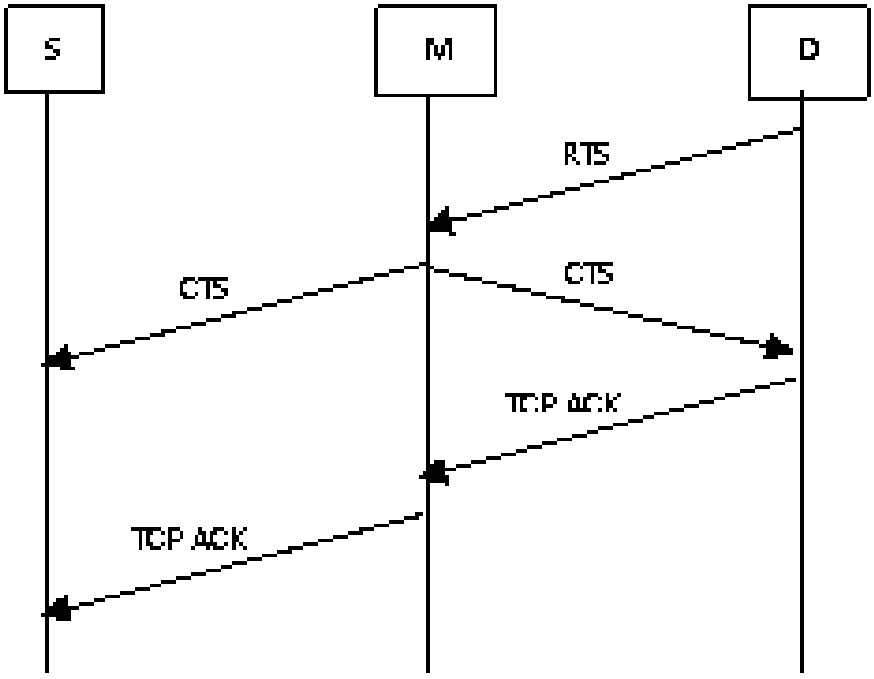
Suppose Source node (S) transmits 100 packets with packet size 1000 bytes and 200 packets are with packet size 50 bytes to Destination node (D). In this scenario the intermediate node (M) forwards only packets with size 50 bytes. According to existing algorithm, at the end node M achieves trust value of 200. The node 'M' achieves high trust value and it is declared as normal node if threshold is 100. However, the intelligent partial forwarding of node 'M' saved its power and escaped from misbehavior detection.



(a) Case 1 – Dropping Large Size Packet (No change in TV(M))



(b) Case 2 – Forwarding Smaller Size data Packet (After receiving the packet from Node M, Node increments the Trust value of node M by 1 i.e.TV (M) = TV(M) +1)



(c) Case 3 – Forwarding control packet (After receiving the packet from Node M, Node S increments the Trust value of node M by 1 i.e.TV(M)=TV(M+1)

Fig 1: Illustration of False Negative Case (S, M and D represent source node, Misbehavior node and Destination node respectively)

5. DESIGN OF PROPOSED SCHEME

Packet forwarding by a node requires resources like energy. The forwarding behavior can be related to the amount of energy that node spends to relay the data packet. We know that the resources like energy, power and processing time spent by a node in forwarding is directly proportional to the size of data packet that it forwards. Since the energy available to a node is limited, it may try to find a way to get a better trust value by spending less amount of energy. Hence, it may act as a well behaved node to participate in the actual routing process.

Based on the above mentioned facts, there is a need to consider packet size as a parameter for trust value computation. We modified the existing scheme by adding verification on the frame size at MAC level before incrementing the trust value. The maximum frame size and minimum frame size in IEEE 802.11 protocol are 2096 bytes and 30 bytes respectively. In our scheme, the maximum trust value '1' will be given to the node that forwards the frame with maximum size. For the frames with packet size less than the maximum, the node will get a trust value proportionally. The modified rule for trust value updation is as follows:

```
If (Pkt_size == 2096)
        Trust = 1;
Else
       Trust= frame size / 2096;
  If (neigbour addr! = srcipaddr(IP Packet))
    {
          TV (neighbor) = TV (neighbor) + trust
    }
```

In the modified scheme, trust value is calculated at MAC layer and it is passed to Network layer. Then in the Network layer, TV(neighbor) is incremented by the calculated trust value if the IP address of the received packet matches with IP address of neighbor node. This condition is to avoid the case where node will get trust value for its generated packet.

After the computation and updation of trust values according to the proposed modification, punishment mechanism will be invoked for the identified misbehaving nodes (i.e. TV(node) < threshold) . There are two approaches to punish the identified misbehaving nodes. First approach mainly concentrates on making the selfish node to participate in the forwarding process. Second approach concentrates on finding the misbehaving node and isolating it from the network for the future forwarding process. In this work, we applied the second approach of isolating the misbehaving node. The trust value of a node is broadcasted periodically, so each node will be having the information about the trust value of their neighboring nodes.

6. PROPOSED SCHEME: A SIMPLE VERIFICATION

Let us consider a node which sends 50 packets with no data. In the existing algorithm, for each received packet the trust value is incremented by '1'. At the end of forwarding 50 packets, the achieved trust value for the node will be 50.

Generally, the size of a frame only with header is 36 bytes. In our proposed algorithm, for each received packet the trust value is by (packet size/2096) i.e. (36/2096) equals to 0.0171. At the end of forwarding 50 packets, the achieved trust value for the node will be 0.85877 (50*0.0171).

According to existing trust algorithm, the above mentioned node will be reported as a well behaving node. This false report indicates the false negative case. However, in our scheme the above mentioned node is reported as misbehaving node. Hence, it is punished by isolating from the network.

7. SIMULATION DESIGN AND RESULTS

In our simulation, we assume nodes move in a 500m*500m region for the simulation time

of 30 sec. We assume that each mobile node moves independently with the same average speed. To evaluate our proposed scheme with the existing scheme, the following network topologies (figure 2 and figure 3) are considered.

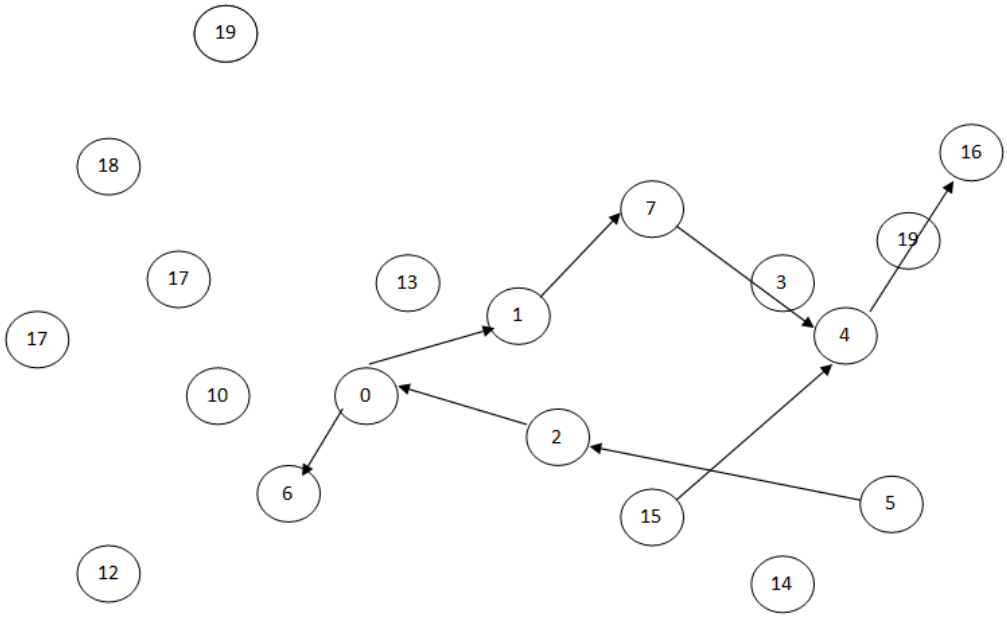


Figure 2: Topology 1 with 20 Nodes

Topology 1 consists of 20 nodes in the network with three connections. The source nodes are 0, 5 and 15, and the destination nodes are 4, 6 and 16. Two nodes kept as misbehaving nodes (node 4 and 7). The misbehavior node drops large sized (2096 bytes) frames. The routing paths between the source nodes and destination nodes are 0-1-7-4, 5-2-0-6 and 15-4-16 respectively. The frame sizes of the connections at MAC level are 400 bytes, 800 bytes and 2096 bytes.

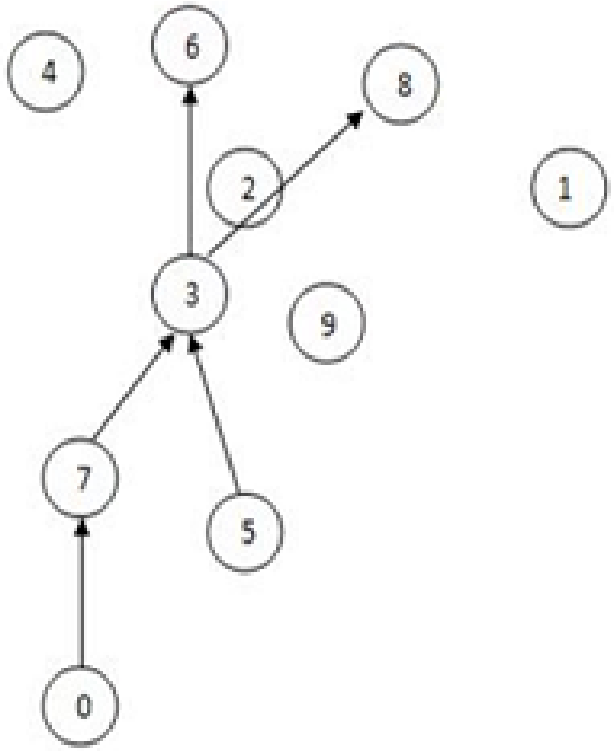


Figure 3: Topology 2 with 10 Nodes

Topology 2 consists of 10 nodes in the network with two connections. The source nodes are 0 and 5, and the destination nodes are 8 and 6. Node 3 is assumed as misbehaving node. The routing paths between the source nodes and destination nodes are 0-7-3-6 and 5-3-8 respectively. The Packet Sizes of the connections are 256 bytes and 2096 bytes.

The above mentioned topologies are simulated using NS2 simulator and the results are summarized in figure 4 (The achieved throughput for the connection between node 0 and node 4 with respect to topology 1) and figure 5 (The achieved throughput for the connection between node 0 and node 6 with respect to topology 2).

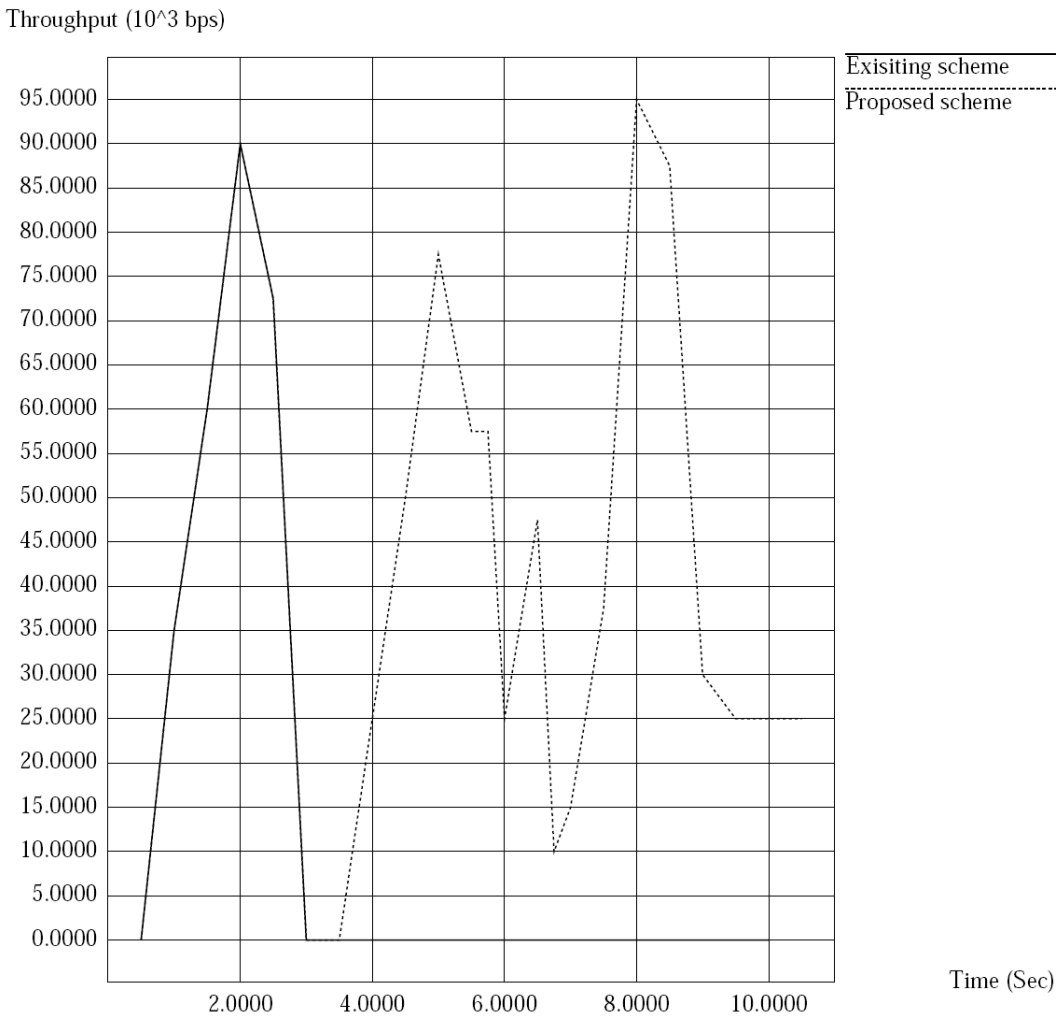


Figure 4: Throughput for Node 0 in Topology 1

In Topology 1, Node 15 is normal node. During the simulation we found no malicious node till 6th second. After 6th second, behavior of node 4 changes from co-operative to non-cooperative. In the case of existing scheme, the state change of node 4 has affected the transmission of connection between 15 and 16. A malicious node in the connection affects the performance of normal node. According to our proposed scheme, the node 4 gets the lower trust value and it was isolated from the network. This punishment has given an alternate path for the connection between node 15 and 16. Hence, the throughput has increased again after the route change.

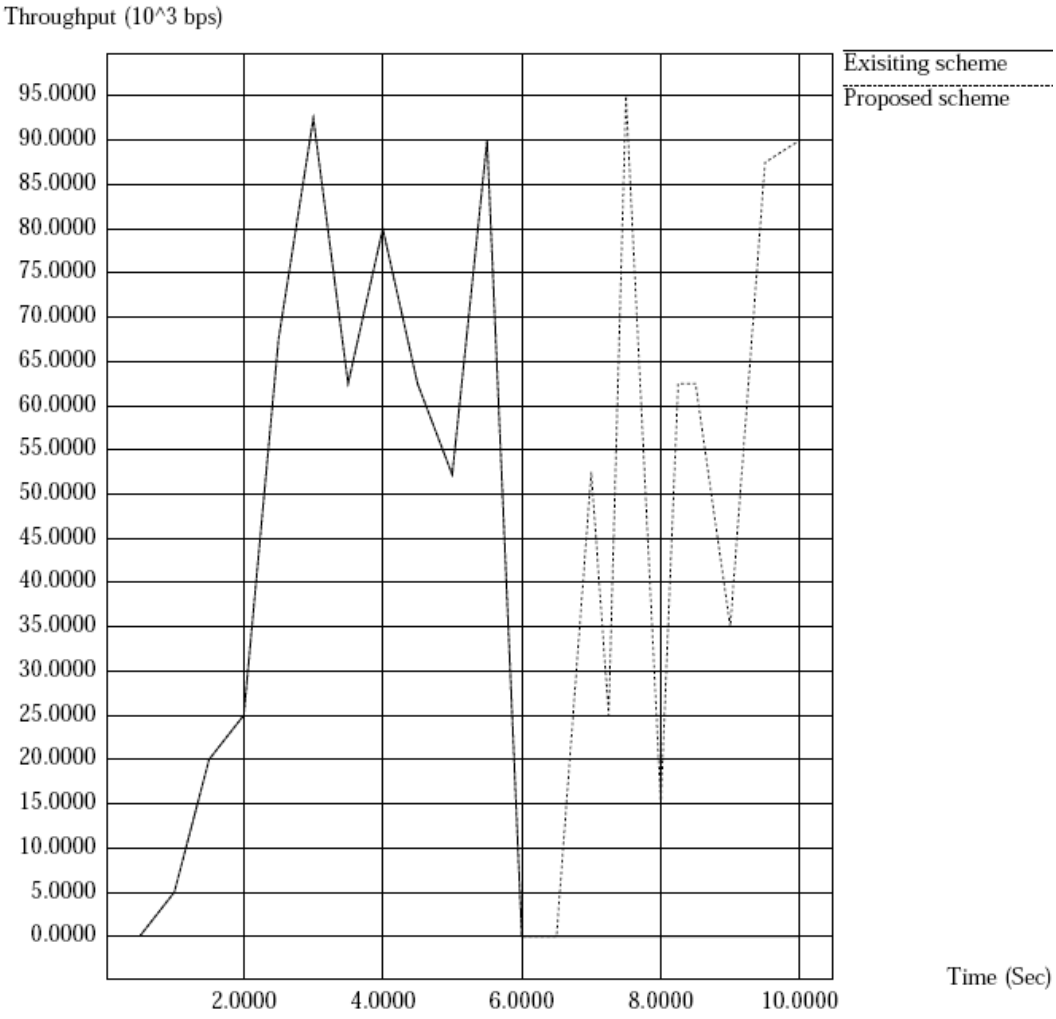


Figure 5: Throughput for node 3 Topology 2

In Topology 2, the transmission of connection between Node 0 and 6 is affected by the misbehaving node 7. However, in the existing scheme, the misbehavior of node 7 was not reported. According to our proposed scheme, the node 7 was isolated and alternate path between 0 and 6 has been calculated.

8. CONCLUSION

The widespread usage of Mobile Ad hoc Networks (MANETs) has given a new research direction on managing the selfish nodes. In this work, a modification on the existing trust

based scheme has been proposed to reduce the false negative case. The modification is implemented with a punishment policy and numbers of simulations with different topologies are performed in NS2 Simulator. With the help of the simulation results, it has been verified that the proposed scheme reduces the false negative rate and increases the performance of cooperative nodes.

In the proposed scheme, the packet size is the only parameter to analyze the forwarding behavior of a node. The solution to solve false negative case may leads to false positive case. As a future enhancement, a detailed investigation on false positive case can be done and a detection scheme with detailed analysis of forwarding behavior can be designed to learn the intention of node's forwarding.

REFERENCES

[1] G.VijayaKumar,Y.VasudevaReddy and Dr.M.Nagendra , "Current Research Work on Routing Protocols for MANET: A Literature Survey", (IJCSE) International Journal on Computer Science and Engineering, Vol. 02, No. 03, 2010.

[2] Mehajabeen Fatima, Roopam Gupta, T.K.Bandhopadhyay, " Performance Analysis of Route Discovery by Cross Layer Routing Protocol- RDCLRP *", I. J. Computer Network and Information Security*, 2013.

[3] Sagarpadiya, Rakeshpandit, Sachinpatel."Survey of innovated techniques to detect selfish nodes in MANET",*International Journal of Computer Networking, Wireless and Mobile Communications (IJCNWMC)*,ISSN 2250-1568,Vol. 3, Issue 1, Mar 2013.

[4] Dipali K. Dakhole, , G.H. Raisoni, Archana R. Raut, Comparitive analysis of broadcast techniques in multi-hop relay MANETS. "*International Journal of Advanced Technology & Engineering Research (IJATER)*",ISSN No: 2250-3536, E-ICETT 2014.

[5] DeekshaNarula, MohitLalit, ParveenChaudhary, "Various Protocols to Manage Cooperation and Reputation in MANET (A Review)", in *International Journal of Advanced Research inComputer Science and Software Engineering*, Volume 3, Issue 8, August 2013.

[6] DipaliKoshti, SupriyaKamoji, "Comparative study of Techniques used for Detection of Selfish Nodes in Mobile Ad hoc Networks", *International Journal of Soft Computing and Engineering (IJSCE),* ISSN: 2231-2307, Volume-1, Issue-4, September 2011.

[7] D.Anitha, Dr.M.Punithavalli," A Collaborative Selfish Replica with Watchdog and Pathrater in MANETS", in *International Journal of Computer Science and Mobile Computing,* Vol. 2, Issue.3, March 2013, pg.112– 119.

[8] TaragFahad , Robert Askwith, "A Node Misbehaviour Detection Mechanism for MobileAd-hoc Networks", School of Computing & Mathematical Sciences,Liverpool John MooresUniversity,Liverpool, UK.

[9] Pankaj Sharma, Yogendra Kumar Jain, "TRUST BASED SECURE AODV IN MANET" in *Journal of Global Research in Computer Science,* Volume 3, No. 6, June 2012.

[10] MehajabeenFatima,RoopamGupta,T. K. Bandhopadhyay, "Cross Layer Preemptive Route Repair Scheme for MANET", in *International Journal of Computer Applications,*Volume 77 - Number 10,2013.

[11] R. Murugan, A. Shanmugam, "A Combined Solution for Routing and Medium Access Control Layer Attacks in Mobile Ad Hoc Networks", in Journal of Computer Science , No. 1416-1423, 2010.

[12] Prof. Rekha Patil, Shilpa Kallimath, "Cross Layer Approach for Selfish Node Detection in MANET", International Journal of Advanced Research in Electronics and Communication Engineering (IJARECE), Volume 1, Issue 3, September 2012.