Secure and proficient cross layer (SPCL) QoS framework for mobile ad-hoc network

Santosh Sahu, Sanjeev Sharma
School of Information Technology, Rajiv Gandhi Prodoyogiki Vishwavidyalaya Bhopal, India

Article Info

Article history:

Received May 23, 2018
Revised Nov 14, 2018
Accepted Mar 10, 2019

Keywords:

Cross layer QoS framework
MANET
QoS framework
SPCL QoS framework

ABSTRACT

A cross layer QoS framework is a complete system that provides required QoS services to each node present in the network. All components within it cooperate together for providing the required services. In existing QoS frameworks there is no security mechanism provided while Security is a critical aspect for QoS in the MANET environment. Cross layer QoS framework tend to be vulnerable to a number of threats and attacks like, over/under-reporting of available bandwidth, over-reservation, state table starvation, QoS degradation, information disclosure, theft of services timing attack, flooding attack, replay attack, and denial of service (DoS) attack, attacks on information in transit and attacks against routing. So it is necessary when designing protocols for QoS framework, the harmony between security and QoS must be present as one impacts the others. In this work we proposed secure and proficient cross layer (SPCL) QoS frameworks which prevents from various types of threats and attacks. The proposed SPCL QoS framework achieves better performance compared to existing QoS frameworks in metrics of throughput, packet drop ratio, end-to-end delay, and average jitter in both condition when malicious node present in the network and when malicious node not present in the network.

Corresponding Author:

Santosh Sahu,
School of Information Technology,
Rajiv Gandhi Prodoyogiki Vishwavidyalaya,
Airport Bypass Road Bhopal MP, Bhopal, India
Email: santoshsahu@rgtu.net

1. INTRODUCTION

Mobile Ad hoc Network (MANET) is a collection of mobile nodes connected by wireless links which can be created on-the-fly without any infrastructure or administrative support [1]. These networks are characterized by self-organization and autonomy. Quality of service (QoS) is the performance [2] level of a service offered by the network to the user. The main component of any Cross layer QoS framework is the QoS service model which describes the way user requirements are fulfil. The other main components of the framework are, QoS signaling which is the combination of resource reservation, admission control and packet scheduling. QoS routing [3] which is used to find all or some of the feasible routs in the network. QoS medium access control, manage the accessibility of the shared medium in efficient way as shown in Figure 1. The combination of QoS service model and QoS signaling is called QoS provisioning [4]. QoS Provisioning is the extra functioning done by simple ad hoc network model to achieve quality of service.

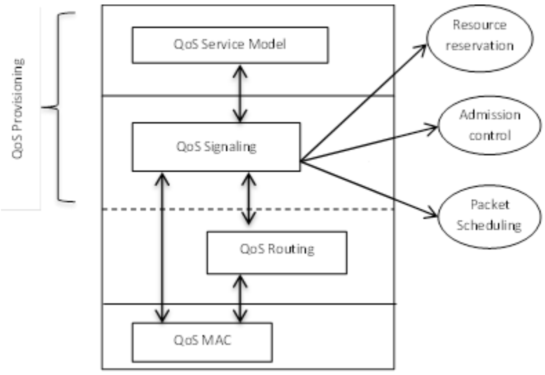

Figure 1. QoS framework model

Cross layer QoS framework design is one of the solutions for providing QOS in wireless networks. A cross layer QoS framework is a complete system in which all layers of network share network information between different layers and worked as a single system that tries to provide QoS services to every users or applications. All components among this technique virtually get together to providing the specified QoS services. Here virtually means that the user of cross layer framework illusion that all the layer of network worked together as single system. There are only four cross layer QoS frameworks of MANETs are available in literature.

Seoung-Bum Lee, et al [4] has proposed INSIGNIA framework that provides associate integrated approach to QoS provisioning by combining in band signal, decision admission management, and packet programing along. The soft state reservation theme employed in this framework ensures that resources ar quickly discharged at the time of path reconfiguration. But, this framework supports solely adaptive applications, maybe, multimedia system applications. This framework is clear to any mack protocol. Additionally as this framework assumes that routing protocol provides new routes within the case of topology changes. If enough resources don't seem to be accessible as a result of the ever-changing configuration, the improved QoS application is also downgraded to base QoS or perhaps to best-effort service. As this framework uses in-band signal, resources don't seem to be reserved before the particular information transmission begins. Thence badge isn't appropriate for period applications that have tight QoS necessities.

D. Dharmaraju, et al [5] has proposed INORA that's higher than insignia in this it will search multiple methods with lesser QoS guarantees. It uses the badge in-band sign mechanism. Since no resources area unit reserved before the particular information transmission begins and since information packets need to be transmitted as best-effort packets just in case of admission management failure at the intermediate nodes, this model might not be appropriate for applications that need laborious service guarantees.

H. Ahn et al [6] has proposed SWAN framework that supporting real-time applications by forward a best-effort mac protocol and not creating any resource reservation. It uses feedback based mostly management mechanisms to control period of time traffic at the time of congestion within the network. As best-effort traffic is a buffer zone for period of time traffic, this model doesn't work well in eventualities wherever most of the traffic is period of time in nature. Even if this model is climbable (because the intermediate nodes don't maintain any per flow or mixture state information), it cannot give laborious QoS guarantees because of lack of resource reservation at the intermediate nodes. AN admitted period of time flow might encounter periodic violations in its information measure necessities.

V. Vivek, et al [7] has proposed PRTMAC that's acceptable in providing higher period of time traffic support and repair differentiation in high quality AWNs reminiscent of military networks fashioned by high speed combat vehicles, fleet of ships, fleet of air-crafts wherever the facility resource isn't a serious concern. In AWNs, fashioned by low power and resource affected hand-held devices, having associate degreeother channel might not be an economically viable answer.

In existing QoS frameworks there is no security mechanism provided while Security is a critical aspect for QoS in the MANET environment. Cross layer QoS framework tend to be vulnerable to a number of threats and attacks like, over/under-reporting of available bandwidth, over-reservation, state table starvation, QoS degradation, information disclosure, theft of services timing attack, flooding attack, replay attack, and denial of service (DoS) attack, attacks on information in transit and attacks against routing. So it is necessary when designing protocols for QoS framework, the harmony between security and QoS must be present as one impacts the others.

2. RESEARCH METHOD

Empirical or hypothesis-testing research design (generally known as experimental research design) is used in our research. Experimental research design are two types Informal experimental designs and Formal experimental designs. In our research we used Informal experimental designs. Informal experimental designs are further classified into three categories (i) Before-and-after without control design. (ii) After-only with control design. (iii) Before-and-after with control design. We used Before-and-after with control design. In research procedure both algorithms and pseudocode are used.

2.1. Proposed SPCL QoS framework

SPCL is a secure and proficient cross-layer QoS framework. In this work we proposed secure and proficient cross layer (SPCL) QoS frameworks which prevents from various types of threats and attacks. In SPCL QoS framework we design and implement Neighbour Node Surveillance Real Time MAC (NNSRT-MAC) protocol at MAC layer; Dynamic Secure Routing (DSR) at routing layer, SPCL In-band signaling for QoS Signaling system. A complete layered diagram of proposed SPCL cross-layer QoS framework is shown in Figure 2. The Detail description of each proposed module of each layer is given below.

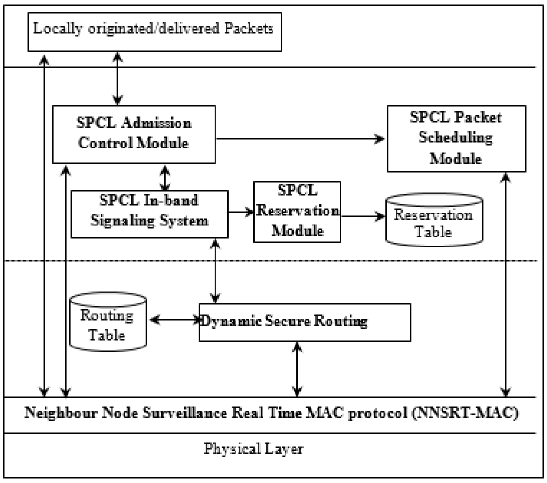
Locally originated/delivered Packets
SPCL Admission Control Module
SPCL Packet Scheduling Module
SPCL In-band Signaling System
SPCL Reservation Module
Reservation Table
Routing Table
Dynamic Secure Routing
Neighbour Node Surveillance Real Time MAC protocol (NNSRT-MAC)
Physical Layer

Figure 2. Proposed SPCL QoS framework

2.1.1. Secure real time MAC (NNSRT-MAC)

In proposed QoS framework SPCL we proposed NNSRT-MAC which is the secure version of Real Time MAC [8]. In NNSRT-MAC When source nodes have packets to send then it store that packet in buffer and send it. After sending a packet source node waits for a fixed time interval to overhear the neighbour node. When neighbour node forward that packet to next hope then source node compare the overheard packet to the buffered packet if packet is similar then source node assume that the corresponding node is the trusted node and increase the trust value by one of corresponding node in routing table. If source node don‘t overhear the send packet within fixed interval then source node assume that the corresponding node is black hole or wormhole node and broadcast a message in whole network that particular node is black hole node.

In NNSRT-MAC When source nodes have packets to send then it store that packet in buffer and send it. After sending a packet source node waits for a fixed time interval to overhear the neighbour node. When neighbour node forward that packet to next hope then source node compare the overheard packet to the buffered packet if packet is similar then source node assume that the corresponding node is the trusted node and increase the trust value by one of corresponding node in routing table. If source node don ‘t overhear the send packet within fixed interval then source node assume that the corresponding node is black hole or wormhole node and broadcast a message in whole network that particular node is black hole node. When this message is received by other node they update own routing table and decrease the trust value by one of corresponding node. And if neighbour node change or alter the field of data packets then source node found that packet comparison is dissimilar and assume that corresponding node is malicious or selfish node and broadcast a message that particular node is malicious or selfish. By using the NNSRT-MAC system we prevent the network by different types of attacks like black hole attack, wormhole attack [9], Dropping Attacks etc.

Consider a situation when source node itself a black hole or malicious node then how NNSRT-MAC system is worked. At this situation if source node is black hole node then it never send or forward the packet. If source node is malicious node then it can sends false information to the next hope node. This is the limitation of proposed NNSRT-MAC. Although this problem is removed in network layer in which we are using dynamic secure routing protocol which will be described in the next section.

As shown in Figure 3 source node S send the packet to neighbour node A and wait for fixed time interval. Since node A is also the neighbour of node S so node S also listen the packet that are send or forwarded by node A. Now node S compare the buffered packet with overhearing packet if packet is similar then node S broadcast the message in whole network that node A is trusted node. If node A not forwards the packet then node S not overhear the particular packet within fixed time interval. Then node S broadcast the message that node A is black hole node or wormhole node. And if node A alter the packet then forward it then node S found that comparison is dissimilar and broadcast a message that Node A is malicious node or selfish node.

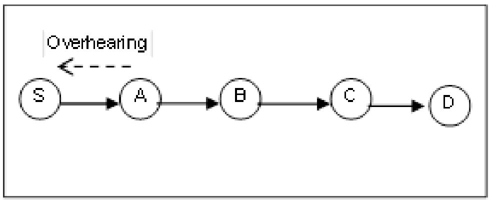

Figure 3. Operation of the “NNSRT-MAC”

2.1.2. Dynamic secure routing

When a source node have traffic to send. Before actual data transmission source node transmit a trial data packet. Dynamic secure routing finds available paths for the destination. And create routing table, add trust field in routing table. Initialize trust value by zero. After finding the paths; store the available path in path vector. Let routing algorithm select p_i(S-A-B-C-D) path vector as an efficient path to sending TRIAL_DATA packet. Each intermediate node that receive TRIAL_DATA packet check that number of hope is greater than two hope then it send back TRIAL_ACK to neighbour of neighbor node. If neighbour of neighbor node receives TRIAL_ACK then check that TRIAL_ACK received by them is from path vector p_i ‘s node and is the neighbour of neighbor node.

For example as shown in Figure 4 source node S first send a TRIAL_DATA packet through efficient path S-A-B-C-D. Now when node B received TRIAL_DATA packet then it checks that number of hope count of a packet is greater than two then it send TRIAL_ACK to node S through node A. If node S receives TRIAL_ACK then check that TRIAL_ACK received by them is from path vector p_i ‘s node and is from node B. Then node S flood message that node A is a trusted node or an authorized node. All nodes that received flood message increase the trust value by one in routing table of corresponding node. If Node S received TRIAL_ACK from another node then node S assumes that node A is malicious node or selfish node and flood message that node A is malicious node or unauthorized node. All nodes that receive flood message decrease the trust value by one in routing table at corresponding node. If node S does not receive any TRIAL_ACK within time interval then node S assumes that node A is malicious node or unauthorized node.
All nodes that receive flood message decrease the trust value by one in routing table at corresponding node. For selecting efficient path vector DSR prefer the node that have more trust value. Using this protocol we prevent various types of attacks like over-reservation [10], state table starvation, over/under-reporting of available bandwidth, QoS degradation, impersonation, information disclosure, theft of services timing attack, flooding attack, replay attack, and denial of service (DoS) attack [11], attacks on information in transit and attacks against routing.

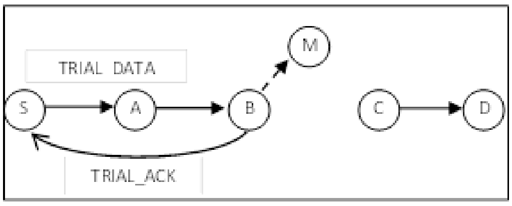

Figure 4. Operation of the dynamic secure routing

2.1.3. SPCL In-band Signaling System

SPCL In-band Signaling System is divided into three modules; SPCL admission control module, SPCL Reservation module and Packet Scheduling Module.

a. SPCL Admission Control Module

SPCL admission control module manages the admitted data traffic. According to their source traffic is classified in to two categories local generated traffic and carry forwarded traffic. Again data traffic is two types' best effort traffic and real time traffic. If traffic is locally generated and best effort then SPCL admission control module forwards the traffic to SPCL packet scheduling module. If local generated traffic is real time then SPCL admission control module calls the SPCL reservation module to reserve the path. If path is already reserved then SPCL admission control module forwards the traffic to the SPCL packet scheduling module. If path is free then it transmits the real time traffic to the secured and efficient reserved path.

If traffic comes from other node is best effort then SPCL admission control module first checks that what is the destination of data packet if destination is himself then it forward the data packet to the upper layers. Upper layers take appropriate action according to data packet. If destination is another node then it forwards the data packet to the SPCL packet scheduling module. If traffic comes from other node is real time traffic; it follows the already reserved path that was reserved by the source node of real time traffic then SPCL admission control module immediately forward the real time traffic to next hope node. If received node is destination node then it forwards the real time traffic to the upper layers. Upper layers take appropriate action according real time traffic.

b. SPCL Reservation module

SPCL reservation module is used for real time traffic to reserve the path in secured manner. To reserve the path SPCL reservation module sends the MAX_BW message end-to-end to each trusted available path. Each nodes of trusted available path mention own maximum available bandwidth in the MAX_BW message now destination node send back AVAIL_BW message to the source node of real time traffic. Now SPCL reservation module calculate the MIN_BW of each available path. For example Path P_1 MIN_BW = MIN [MAX_BW of N_1, MAX_BW of $N_{2,}$ ……. MAX_BW of $N_{i,}$…] Now send RESERVATION message on that path which satisfies the desired condition for real time traffic. Nodes that received RESERVATION message change own status from idle to reserve. If some node deny the RESERVATION then it follow the another path. If no path is available for reservation then it wait for a specific time period after that it retry to reserve the path. After completion of reservation process create reservation table. Now SPCL Admission Control Module transmits real time traffic using reserved path. The reserve path may efficient or not but it is secured.

c. SPCL Packet Scheduling module

SPCL packet scheduling module is used mainly for best effort traffic. It maintains a queue for scheduling. SPCL packet scheduling module assign the priority to different types of traffics like give first priority to local real time traffic, two priority to outside real time traffic, three priority to local best effort traffic, four priority to outside best effort traffic. After assign the priority use "priority based round robin algorithm" [12] to forward the data packets or real time traffic if node is not reserved. If node is reserved then wait until node become free.

2.2. Pseudo code or algorithm of SPCL QoS framework

2.2.1. Neighbour node surveillance real time MAC (NNSRT-MAC)

Use NNSRT-MAC protocol at MAC layer.

```
a. Store recently sent packets in a buffer and comparing each overheard packet with the packet in the
   buffer to see if there is a match.
   If yes; then do nothing
   Else; flood message that corresponding node is selfish node or malicious node
b. All nodes that receive flood message decrease the trust value by one in routing table at corresponding
   node.
c. After comparison remove recently packet from buffer.
```

2.2.2. Dynamic secure routing (DSR)

```
a. Let assume that Source node S has traffic to send
b. Firstly source transmit TRIAL_DATA packet before actual data transmission to the destination.
   1. Now TRIAL_DATA packet is given to "packet forwarding module".
   2. Dynamic source routing finds available paths for the destination. And create routing table, add trust
      field in routing table. Initialize trust value by zero.
```

Node	Next node	Hop count	Trust value
A	B	1	0
B	D	3	0

3. After finding the paths store the available path in path vector P_1, P_2, P_3………p_n like
   $P_1 = [N_1, N_2, N_3 …… N_i……..]$
   $P_2 = [N_1, N_2, N_3 …… N_i ……..]$
   .
   .
   .
   $P_n = [N_1, N_2, N_3…… N_i ………]$
4. Let routing algorithm select p_i path vector as an efficient path to sending TRIAL_DATA packet.
5. Each intermediate node that receive TRIAL_DATA packet check condition
   If path vector p_i node no. j is (j>2) then send two hop back TRIAL_ACK.
6. If node (j-2) receives TRIAL_ACK then check that

```
{
(A) TRIAL_ACK received by them is from path vector p_i 's node no j or not
(B) If yes; then node (j-2) flood message that node j-1 is a trusted node or an authorized node.
(C) All nodes that receive flood message increase the trust value by one in routing table at
    corresponding node.
}
Else
{
(A) Node (j-2) flood message that node j-1 is malicious node or unauthorized node
(B) All nodes that receive flood message decrease the trust value by one in routing table at
    corresponding node.
}
```

7. If node (j-2) does not receives TRIAL_ACK then
   (A) Node (j-2) flood message that node j-1 is malicious node or unauthorized node
   (B) All nodes that receive flood message decrease the trust value by one in routing table at corresponding node.

2.2.3. SPCL In-band signaling system

SPCL In-band Signaling System is divided into three modules; SPCL admission control module, SPCL Reservation module and Packet Scheduling Module.

a. SPCL Admission Control Module

```
1. Checks that is traffic from local generated or received from other node.
2. If traffic is locally generated then checks that
   Is node already reserved or not
   (i) If already reserved then refer to "SPCL packet scheduling module"
   (ii) If no; then check that
        Is traffic is best effort or real time
        {
           If best effort then
           {
        (a) "SPCL Admission Control Module" select secure and efficient path to send data packet.
            }
            If traffic is Real time then
                {
(a) Refer to "SPCL Admission Control Module"
  }
        }
3. If traffic comes from other node then "SPCL Admission Control Module" checks that
       (i) Is this packet is own or not
  If yes; then refer it to upper layers. Upper layers take appropriate action according to data packet.
```

```
        (ii)  If no; then check that
   Is traffic is best effort or real time
                {
     (A)      If best effort then check that
                    {
                     Is already reserved or not
                          {
       (a) If yes; then refer to “SPCL packet scheduling module”
       (b) If no; then “SPCL Admission Control Module” select secure and efficient path to send data packet.
              }
        }
   }
     (B)       If traffic is Real time then check that
                  {
                    Is already reserved or not
                    {
             (a) If yes; then deny transmission or reservation
                       (b) If no; then
                          {
             (a) Refer to “SPCL Admission Control Module”
               }
                    }
                      }
```

b. SPCL Reservation Module

1. Reserve the Resource using following steps
   (i) Send MAX_BW message end-to-end to each available paths
   (ii) Each nodes of a path return own available maximum bandwidth in the MAX_BW message.
   (iii) Now calculate minimum band width of each available paths like
   Path P_1 MIN_BW = MIN [MAX_BW of N_1, MAX_BW of N2, ……. MAX_BW of N_{i},…]

   Path P_2 MIN_BW = MIN [MAX_BW of N_1, MAX_BW of N2, ……. MAX_BW of N_{i},…]
   .
   .
   .
   Path P_n MIN_BW = MIN [MAX_BW of N_1, MAX_BW of N2, ……. MAX_BW of N_{i},…]
2. Send RESERVATION message on that path which satisfies the desired condition for real time traffic.
3. After completion of reservation create reservation table.
4. Now Packet forwarding module transmit data packet to using reserved path. It may efficient or not.

c. SPCL Packet Scheduling Module

```
1.  Assign priorities to the different traffics in a queue
    {
      (i)    Assign one priority to local real time traffic.
      (ii)   Assign two priority to local real time traffic.
      (iii)  Assign three priority to local best effort traffic.
      (iv)   Assign four priority to outside best traffic.
    }
2. Continuously check that is resource become free or not
   If yes;
  {
    (a)  Use “priority based round robin algorithm”
    (b)  Refer traffic according to priority  to “SPCL Admission Control Module”
   }
3.  If no;
     {
       Wait until resource become free
      }
```

2.3. Simulation environment

Data is acquired by performing simulation experiments using fixed some variables and vary other variables. SPCL QoS framework is implemented in network simulator NS-2 [13] in Red hat enterprise edition 6.5. In this paper, we are comparing the performance of SPCL QoS framework with existing QoS frameworks in terms of throughput [14], packet delivery ratio and end-to-end delay. The simulation parameters are shown in Table 1.

Table 1. Parameters are set during simulation

S. N.	Parameter	Value
1.	Simulator	NS-2.35
2.	Area (Length*Width)	500*500
3.	Channel type	Wireless Channel
4.	Radio Propagation Model	Two Ray Ground [15]
5.	Interface queue Type	Drop Tail/ PriQueue
6.	Antenna	Omni directional Antenna
7.	MAC Protocol	Neighbour Node Surveillance Real Time MAC (NNSRT-MAC)
8.	Routing Protocol	DSR(dynamic secure routing)
9.	Signaling system	SPCL In-band Signaling System
0.	Admission control module	SPCL Admission control module
11.	Reservation Module	SPCL Reservation Module
12.	Packet Scheduling module	SPCL Packet Scheduling Module
13.	Packet Scheduling Algorithm	priority based round robin algorithm
14.	Type of traffic	CBR [16]
15.	Simulation Time	300 sec
16.	No. of Nodes	60
17.	Node Speed, Mobility type	30 m/s, Radom (in meter/second)
18.	No. of Malicious Nodes	10
19.	QoS Framework	SPCL, INSIGNIA, INORA, PRTMAC

3. RESULTS AND ANALYSIS

We have design two types of scenarios without malicious node and presence of ten malicious nodes. Further, we will compare the performance of SPCL with INSIGNIA, INORA, and PRTMAC QoS framework in terms of metrics throughput, end to end delay, packet delivery ratio and average jitter by plotting the graph.

3.1. Performance analysis of scenario when malicious node not present in the network

3.1.1. Throughput

Figure 5 represents the throughput of the QoS framework. The throughput of any network is degraded as speed of a node increased. Here we compare the average of all throughputs of all speeds. The throughput of SPCL framework are increased 11.73% compare to PRTMAC, 33.14 compare to INORA and 48.77 compare to INSIGNIA frameworks.

3.1.2. End-To-End Delay

Maximum End-To-End Delay can lead to low performance of the MANET and minimum End-To-End Delay is the indication of high efficiency and speed of the network. Figure 6 shows End to End delay in milliseconds. End to End delay of SPCL framework is decreased by 9.75% compare to PRTMAC, 19.07% by INORA and 24.13% by INSIGNIA framework.

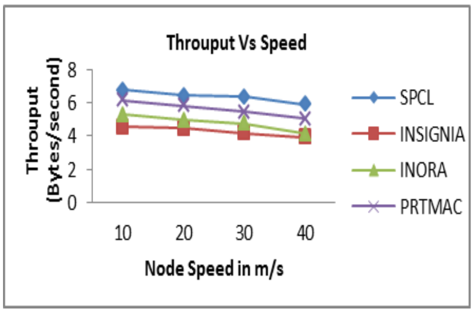

Figure 5. Throughput of QoS frameworks

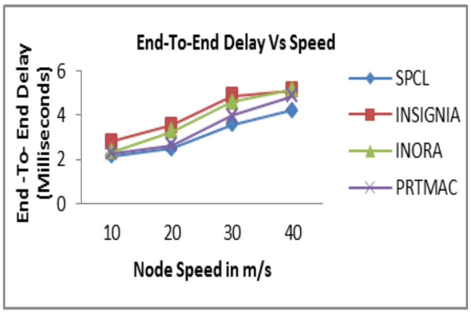

Figure 6. Throughput of QoS frameworks

3.1.3. Packet delivery ratio

Delivery ratio (PDR) is the packets that are successfully delivered to a destination divide by total number of packet send. Figure 7 shows graph between PDR and node speed. PDR of SPCL framework is increased 10.11% compared to PRTMAC, 15.46% by INORA and 18.92% compare to INSIGNIA framework.

3.1.4. Average jitter

Jitter is the delay variance in the time between packets arriving. It should be less for better performance. Average jitter of SPCL framework is lower than other frameworks as shown in Figure 8. Average jitter of SPCL framework is decreased by 13.86% compare to PRTMAC, 27.17% by INORA and 38.47% by INSIGNIA framework.

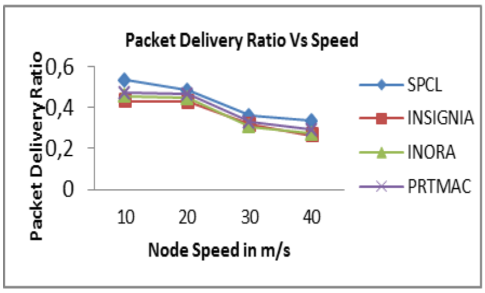

Figure 7. Packet Delivery Ratio of QoS frameworks

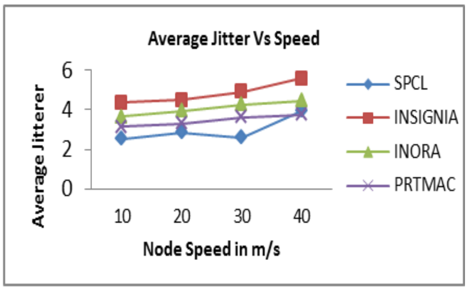

Figure 8. Average Jitter of QoS frameworks

3.2. Performance analysis of scenario when ten malicious node present in the network

3.2.1. Throughput

The throughput of QoS frameworks is degraded due to the presence of malicious node. But still SPCL has better throughput than other QoS frameworks as shown in Figure 9. The throughput of SPCL framework are increased 08.47% compare to PRTMAC, 44.52 compare to INORA and 55.28 compare to INSIGNIA frameworks.

3.2.2. End-To-End Delay

As we know that end to end delay of network is increased as malicious node present in the network. But if we compare it with other frameworks the SPCL has compare to lower end to end delay shown in Figure 10. End to End delay of SPCL framework is decreased by 7.66% compare to PRTMAC, 06.22% by INORA and 18.68% by INSIGNIA framework.

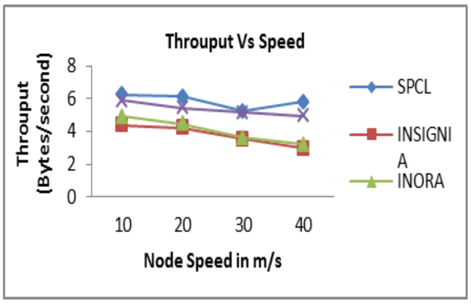

Figure 9. Throughput of QoS frameworks

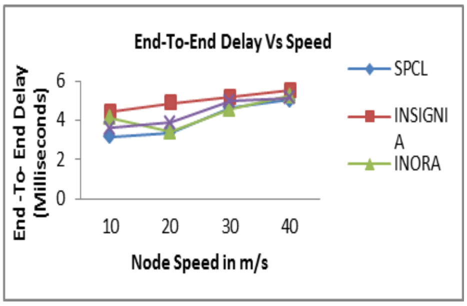

Figure 10. Throughput of QoS frameworks

3.2.3. Packet delivery ratio

PDR also affected by malicious node for better performance PDR must be high. Here PDR of SPCL QoS framework is better for varying node speed as shown in Figure 11. PDR of SPCL framework is increased 17.50% compared to PRTMAC, 25.39% by INORA and 29.43% compare to INSIGNIA framework.

3.2.4. Average jitter

Jitter is the delay variance in packet delivery so jitter must be lower for better performance. If we compare average jitter of SPCL QoS framework with other QoS framework, it is lower for every node speed as shown in Figure 12. Average jitter of SPCL framework is decreased by 11.16% compare to PRTMAC, 28.07% by INORA and 11.53% by INSIGNIA framework.

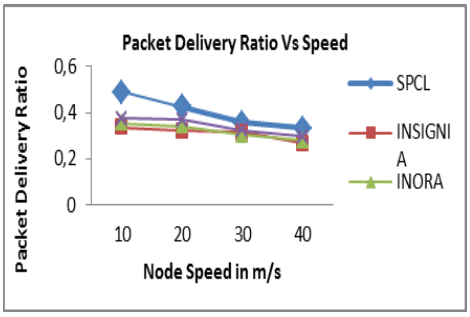

Figure 11. Packet Delivery Ratio QoS frameworks

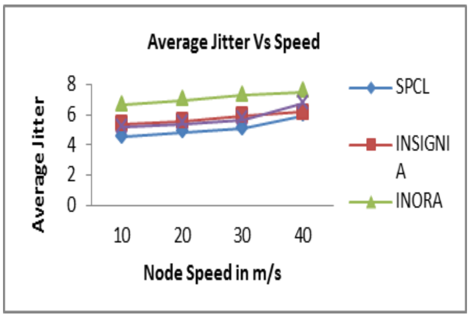

Figure 12. Average Jitter of QoS frameworks

4. CONCLUSION

SPCL QoS framework has various features that it differ from other existing QoS frameworks. The proposed NNSRT-MAC layer protocol detects the malicious nodes, prevent from various types of threats/attacks and maintain the performance level of shared medium. QoS signalling scheme is very simple and effective taking less time to reserve the resource and send packet to packet scheduling module which use priority based round robin algorithm for forwarding the packets. The proposed DSR routing algorithm finds the efficient path dynamically in secured manner. The SPCL QOS framework supports adaptive and real time traffic in safe and efficient way. A key contribution of our framework is that it uses very simple methods for security rather than complex algorithm exists in security. Using this framework we prevent various types of attacks like over-reservation, state table starvation, over/under-reporting of available bandwidth, QoS degradation, impersonation, information disclosure, theft of services timing attack, flooding attack, replay attack, and denial of service (DoS) attack, attacks on information in transit and attacks against routing. SPCL QoS framework gives better results in terms of throughput, packet delivery ratio and end-to-end delay in both situations when malicious node present and absent in the network.

REFERENCES

[1] C. K. Toh, “Ad hoc mobile wireless networks: protocols and systems,” Pearson Education, 2001.

[2] T. B. Reddy, *et al*., “Quality of service provisioning in ad hoc wireless networks: a survey of issues and solutions,” *Ad Hoc Networks*, vol/issue: 4(1), pp. 83-124, 2006.

[3] McNerney, *et al.*, “A 2-dimensional approach to QoS provisioning in adversarial mobile ad hoc network environments,” *Proceedings of the 15th ACM international conference on Modeling, analysis and simulation of wireless and mobile systems*, pp. 143-150, 2012.

[4] S. B. Lee, *et al.*, “INSIGNIA: An IP-based quality of service framework for mobile ad hoc networks,” *Journal of Parallel and Distributed Computing*, vol/issue: 60(4), pp. 374-406, 2000.

[5] D. Dharmaraju, *et al.*, “INORA—A unified signalling and routing mechanism for QoSsupport in mobile ad hoc networks,” *Proceedings of ICPPW 2002*, pp. 86-93, 2002.

[6] H. Ahn, *et al.*, “Supporting service differentiation for real-time and best-effort traffic in stateless wireless ad hoc networks,” *IEEE Transactions on Mobile Computing*, vol/issue: 1(3), pp. 192-207, 2002.

[7] V. Vivek, *et al.*, “A novel out-of-band signaling mechanism for enhanced real time support in tactical ad hoc wireless networks,” *Proceedings of IEEE RTAS2004*, 2004.

[8] B. S. Manoj and C. S. R. Murthy, “Real-time traffic support for ad hoc wireless networks,” *Proceedings of IEEE ICON 2002*, pp. 335-340, 2002.

[9] S, Tripathi and A. K. Jain, “Secure Routing Protocol for Integrated UMTS and WLAN Ad Hoc Networks,” *Bulletin of Electrical Engineering and Informatics*, vol/issue: 5(4), pp. 469-488, 2016.

[10] C. Zouridaki, *et al.*, “Analysis of Attacks and Defense Mechanisms for QoS Signaling Protocols in MANETs,” *Wireless Information Systems*, pp. 61-70, 2005.

[11] K. Pelechrinis, *et al.*, “Denial of service attacks in wireless networks: The case of jammers,” *IEEE Communications surveys & tutorials*, vol/issue: 13(2), pp. 245-257, 2011.

[12] J. P. Sheu, *et al.*, “A priority MAC protocol to support real-time traffic in ad hoc networks,” *Wireless networks*, vol/issue: 10(1), pp. 61-69, 2004.

[13] U. C. Berkeley and USC LBL, “ISI and Xerox Parc, NS-2 documentation and software, version 2.35,” in T. Issariyakul and E. Hossain, “Introduction to network simulator NS2,” Springer Science & Business Media, 2011.

[14] S. Sharma and R. Gupta, “Simulation study of blackhole attack in the mobile ad hoc networks,” *Journal of Engineering Science and Technology*, vol/issue: 4(2), pp. 243-250, 2009.

[15] A, Schmitz and M. Wenig, “The effect of the radio wave propagation model in mobile ad hoc networks,” *Proceedings of the 9th ACM international symposium on Modeling analysis and simulation of wireless and mobile systems*, pp. 61-67, 2006.

[16] M. L. Sharma, *et al.*, “Performance Evaluation of MANET Routing Protocols under CBR and FTP traffic classes,” *Int. J. Comp. Tech. Appl*, vol/issue: 2(3), pp. 393-400, 2010.

BIOGRAPHIES OF AUTHORS

Santosh Sahu received his M.E degree in CCN from Madhav Institute of Technology and Science, Gwalior, MP, India in 2007. He is currently pursuing Ph.D in School of Information Technology, Rajiv Gandhi Proudyogiki Vishwavidyalaya, Bhopal. He is having overall teaching experience of 10 years at UG and PG level. His major research Interests in Mobile Ad-hoc Network.

Sanjeev Sharma received his Ph.D degree in Information Technology from Rajiv Gandhi Proudyogiki Vishwavidyalaya, MP, India in 2010. He is currently working as a Head of Department and Professor in School of Information Technology, Rajiv Gandhi Proudyogiki Vishwavidyalaya, Bhopal. He is having overall teaching experience of 30 years at UG and PG level. His major research Interests in Mobile Ad-hoc Network.