J. Sustain. Wireless Syst.
Vol.01/ No. 01
Pages: 42-61
http://irojournals.com/irosws/
DOI: https://doi.org/10.36548/jsws.2019.1.004

TRUST BASED ROUTING ALGORITHM IN INTERNET OF THINGS (IoT)

Dr. Jennifer S. Raj,
Department of ECE, Gnanamani College of Technology,
Namakkal, India. Email: jennifer.raj@gmail.com.

Alphina Stephy. S
Karunya University, Coimbatore.

ABSTRACT - The development in the area of networking is Internet of Things (IoT). This will interrelated the object and things together. The realization of IoT subsystems will be subjected to numerous constraints that include cost, power, energy, and lifetime. However, most challenging requirement will be trust. It is widely recognized that the attacks from malicious parties can activate from Internet to the physical word. Hence, trust of IoT is of essential importance. Therefore, trust management is considered as a efficient solution to IoT related issues. Trust management has useful technology for providing security service and it has been used in many applications such as P2P, Grid, adhoc network and so on. Thus the trust based routing algorithm in Internet of Things is proposed for providing a potential security system. With this, the major focus on the problem of trust on the malicious nodes in any environment.

KEYWORDS – Internet of Things, sensor, malicious nodes.

I. INTRODUCTION

Internet of Things (IoT) is to create a physical objects are merged into information networks in order to provide advanced services for humans. It embedded with electronics, software, sensors, and network connectivity that enables these objects to collect and exchange data. The internet of things allows objects to be sensed and controlled remotely across existing network infrastructure. Then it will integrated with the physical world into computer-based systems, and it improving efficiency, accuracy and economic benefit. The technologies are smart grids, smart homes, intelligent transportation and smart cities. Each thing is uniquely identifiable through its system. Internet of Things is also expected to generate large amounts of data from different locations, with the quick aggregation of the data. It will increase the storage and the process effectively. Internet of Things is one of the platforms Smart Energy Management Systems. In every field IoT is used. This systems will collecting the informations from various area.In home environment, they monitoring and security management and it will be controlled. In wearable devices IoT places a vital role. IoT produces the ultra low power. The other applications are smart management, transportation and environmental monitoring. In every field IoT finds its applications with memory, power and embedded devices. The other applications are smart management, transportation and environmental monitoring.

The network is suffering from the various attacks. The traditional approaches are not able to secure the system. The network requires the improvement in the security system. The trust is established between the nodes and the trust is evaluated. The survival of the network is depending on the trust in the network. The security and the trust are interdependent. The decentralized network needs more security in the networks. The Internet of Things having a problem in the trust system. Some methods are used to solve the problem in the network.

The network is involved with the people and the social networks based on the Internet of Things. It will bring the convergence of Internet of Things and the social networks. The social network in IoT lacks the understanding about the other member and the basic behavior of the objects. The trustworthiness is also needed in the Social Internet of Things. Reputation based trust can deal the malicious behavior of the nodes. The major problem is found in the autonomous system to find the peer that can provide the service. The social networking concept is the integration with the Internet of things, it is establish with the social relationships with respect to the owners. It is improving the scalability of the network information. For the trust management the model systems were used [1]. But the problem was it could not manage the network traffic. The trust management is divided into three layers and each is controlled by the trust mechanism [5].

For the adaptive trust management, the trust parameters were used to change the conditions in the network. But the performance in the application was increased [2]. The Internet of Things was integrated with environment. The trust were controlling by the keying technologies. In wireless sensor networks the key control method was not able to use. Because, wireless sensor networks having the bridge communication [3]. The architecture was used in the trust management system. The service oriented architecture is used in the heterogeneous system.

It will minimize the convergence time and estimation of the trust. The trust protocol is needed the more storage space [4]. IAM is demanded is to determine the trust for protecting the security. It is evaluating the behavior of the user [6]. The cloud centric internet of things is used for the public safety. Crowd management is used for the trustworthiness [7].

The energy efficiency is the major problem in Internet of Things. The energy efficient architecture is used to reduce the power level. But, it needed the hardware resources [8]. The scheduling algorithm is also used to for the energy efficiency. For the secure communication the signcryption is used. The diffie hellman problem in the wireless sensor networks is recovered. The benefits of the trustworthiness management in the IoT show the effectiveness in the malicious nodes in the network. The trust will reduce the misbehaving of the nodes.

II. NETWORK MODEL

The Internet of Things (IoT) integrates a large amount of everyday life devices from heterogeneous network environments, bringing a great challenge into security and reliability management. Recognizing that the smart objects in IoT are most likely human-carried or human-operated devices, trust is used. The trust routing algorithm is proposed for malicious nodes in Internet of Things.

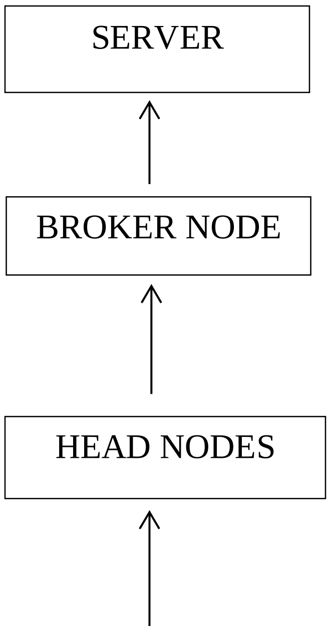


Fig.1 Network Model

The things/sensors are clustered into subgroups, each subgroup has a head node that delivers the messages originated from the group through the broker node to the ultimate receiver of the sensed data. The Fig.1 shows the network model. A network switch is a networking device that connects devices together on a computer network, by using packet switching to receive, process and forward data to the destination device. A bridge is a device that connects a local area network (LAN) to another local area network with the same protocol .

The head node can transfer the data to the other network through a switch and bridge. If the leaf nodes in each subgroup fail to work, the backup node can replace the failed node.

The proposed scheme is used to identify the proxy node. For the identification of proxy node two methods were used. The methods are distance estimation and angle estimation. For routing scheme were proposed to find the authentication, integrity and confidentiality.

I. DISTANCE ESTIMATION

For the distance estimation, the distances between the nodes were taken. The number of nodes is calculated and the separation of distance between the backbone node and broker node were taken. Every head nodes and broker node has their distance values. The permitted tolerance value is taken according to the network model. The estimated value of distance is defined as the sum of the distance separation of nodes and permitted tolerance. If the node moves away from the estimated distance of separation, then the node will be considered as proxy node.

The number of broker nodes can be defined as S,

$$S = \{ S_0 , S_1 , S_2 ,......, S_n \} \tag{1.1}$$

The number of head nodes can be defined as B,

$$B = \{ B_0 , B_1 , B_2 ,......, B_k \} \tag{1.2}$$

S_iB_j - Distance of separation between a broker nodes and head nodes

$S_i\hat{B}_j$- Estimated distance of separation between the broker nodes and head nodes

$$S_iB_j = S_iB_j + \Delta S_lB_l \qquad (1.3)$$

ΔS_lB_l- permitted tolerance

II. ANGLE ESTIMATION

Every node in the network has their vertices and edges. According to that the angle estimation is calculated. The angle of separation between the broker and head nodes is taken. The sum of the angle separation of nodes and permitted tolerance angle value is considered as an estimated angle. The broker and head nodes will connect to the estimated angle. If the node moves beyond their estimated angle value, then the node is considered as a proxy node.

The number of broker nodes,

$$S = \{ S_0, S_1, S_2, \ldots\ldots, S_n \} \qquad (1.4)$$

The number of head nodes,

$$B = \{ B_0, B_1, B_2, \ldots\ldots, B_k \} \qquad (1.5)$$

$\angle S_{n_i}B_{k_j}$- Angle of separation between the broker nodes and head nodes

$\angle \hat{S}B_{n_ik_j}$ - Estimated tolerance between the broker nodes and head nodes,

$$\angle \hat{S}B_{n_ik_j} = \angle SB_{n_ik_j} + \Delta\theta \qquad (1.6)$$

$\Delta\,\theta$- Permitted tolerance

III. ROUTING ALGORITHM

For the path selection routing is used, it will send and receive the packets. Routing is used to select the path in the networks to send and receive the packets. The purpose of the routing algorithm is to determine the best path for data. The network is having subgroups. Each subgroup has a group of leaf nodes and a head node. A node which is having a maximum of edges is elected as a cluster head. The coordinator is selected according to the maximum of active links in the nodes.

For the routing algorithm the authentication, confidentiality and integrity is used.

A. AUTHENTICATION

Authentication is a process are compared to the authorized users information within an authentication server. The lifetime and packet handled is considered as authentication. The authentication is maximum of lifetime and maximum of packet handled by the node. Lifetime is defined as the ratio of difference between initial energy and residual energy to the entire computation time.

$$\text{Lifetime} = ((E_{int} - E_{res})/t) \tag{1.7}$$

$$\text{Authentication index, Ɉ} = \max(\text{Lifetime}) \,||\, \max(\text{Packet handled}) \tag{1.8}$$

B. CONFIDENTIALITY

In information security, that information is not available to the unauthorized entities. The success rate and retransmission is calculated. The confidentiality is the maximum of success rate and minimum of retransmission.

The successful delivery rate of message indicates the radio of the amount of receiving information of the destination node and the amount of sending information of the source node, in a certain time limit. The retransmission occurs due to packet loss and mobility of the nodes. If the retransmission is more then, the delay is more and the throughput is decreased. So, the minimum retransmission is needed.

Success rate for a node is defined as a node which has maximum throughput or minimum retransmissions.

$$\text{Success rate} = \max(\text{Throughput}) \,||\, \min(\text{retransmission}) \quad (1.9)$$

The maximum throughput is denoted by a threshold α. The α should be greater than 80 percentage. Throughput is defined as a number of packets successfully received in unit time. The minimum retransmission is denoted by a threshold β. The β should be less than 5 percentages.

$$\text{Confidentiality index, } \upsilon = \max(\text{Success rate}) \,||\, \min(\text{Retransmission}) \quad (1.10)$$

C. INTEGRITY

Integrity of information will protecting information from the unauthorized user by being modified. The maximum of neighborhood threshold and minimum of active links is calculated for integrity. The nodes in the network has its own threshold. The threshold is like covering range, the range should be maximum. The active links should be minimum.

$$\text{Integrity index, } \tau = \max(\text{Neighbor threshold}) \,||\, \min(\text{Active links}) \quad (1.11)$$

D. ROTUING ALGORITHM

An undirected planar graph G (V, E) with n nodes is given as an input.

Consider the head of leaf nodes,

$$C_i = \{C_0, C_1, C_2, \ldots\ldots, C_m\} \quad (1.12)$$

C_i - Head of leaf nodes

m – Number of leaf head nodes

P_j - Degree of trust, i is a positive integer, j=m

E_{int} - Initial energy of node

E_{res} - Residual energy of node

t – Computation time (secs)

V_n E_{pq}

- Vertex of a node
  - Edges of a node

Authentication index, Ɉ = max (Lifetime) || max (Packet handled)

Confidentiality index, υ = max (Success rate) || min (Retransmission)

Integrity index, τ = max (Neighbor threshold) || min (Active links)

The authentication, confidentiality and integrity is taken. A node is marked true in a trusted system only if it satisfies the following condition,

- Ɉ || υ & τ= true (set as priority)

When the node is marked true its priority is set

Condition 1

- If
- υ >τ > Ɉ - priority 0
- elseif
- (υ+τ) > 2Ɉ - priority 1
- else
- (Ɉ+υ) > 2τ –priority 2

According to the condition the priority is set and node is set as true.

IV. RESULTS

There were two methods used to find the identification of proxy node. They are distance estimation and angle estimation. The routing algorithms are also used to find the trust in the network. The network modelling is done using ns-3.

A. CREATION OF NETWORK MODEL

The nodes are created in grid topology. The leaf nodes are transmitting to the head node and the head node forward the data to the server through the router. The Fig.4.1 shows the network modelling. Bridge is used for the connection of two networks. In the network model the one group of leaf nodes will communicate to the other group through the switch and bridge. Then the data will send to the router and it reaches the destination node.

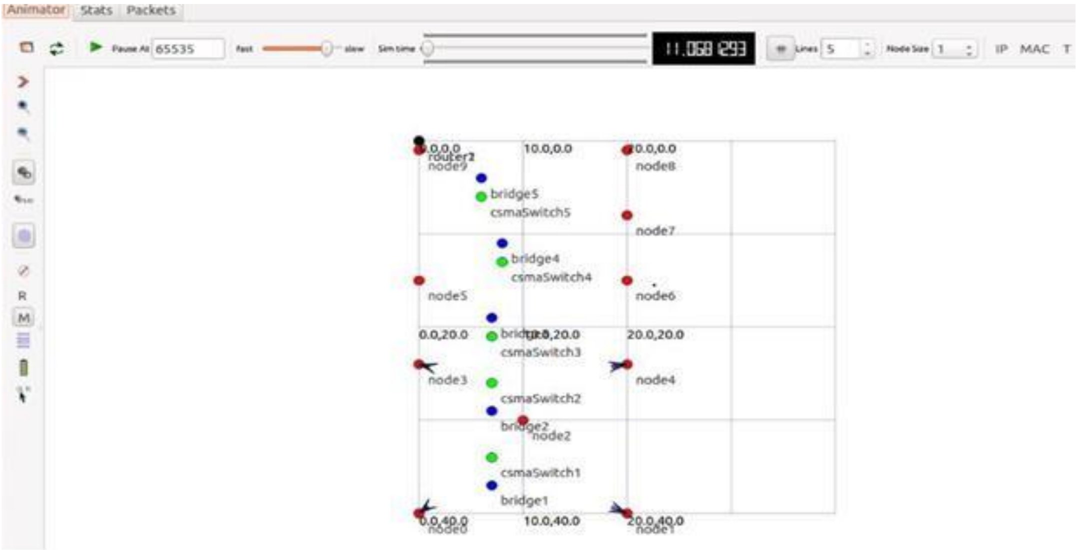


Fig.2. Creation of network model

The leaf node will communicate to a head node of each group. There were five leaf nodes are communicate to a head node. The Fig.3 shows communication between leaf node and head node. The link setup were made for the communication between the nodes.

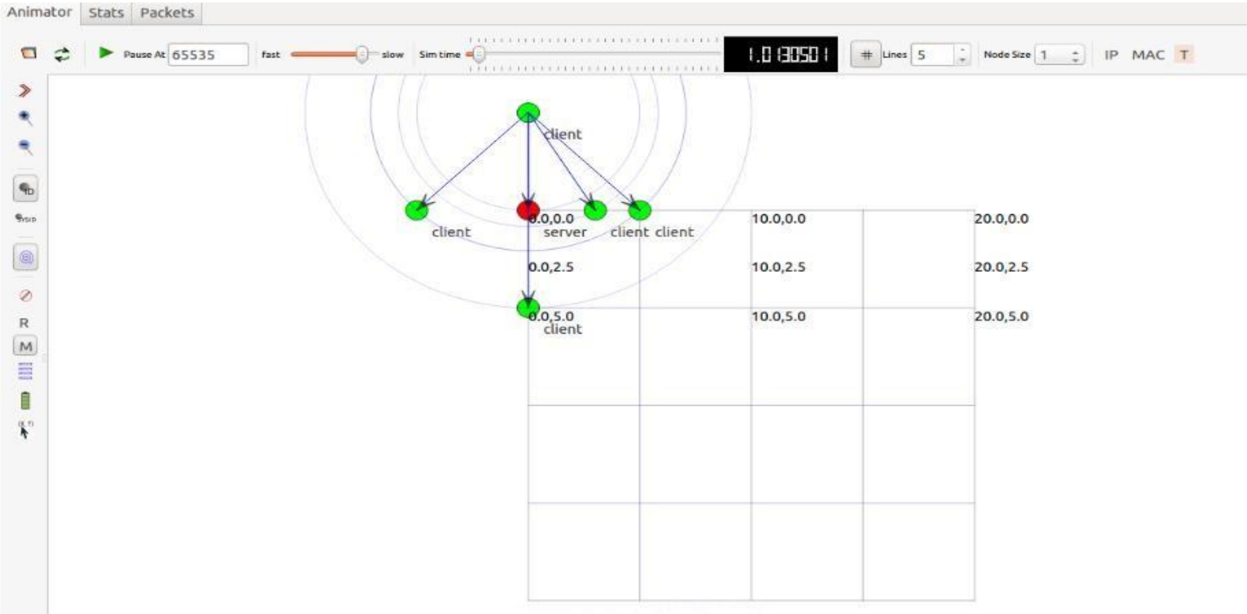


Fig.3. Communication between leaf node and head node

B. UNTRUSTED NETWORK

In the untrusted network, if the node is moved away, the moved node is not able to send or receive the data from the networks. The proxy node will act like an existing node and it send or receive the data from the other nodes. Because of the proxy node the data were lost and it corrupted by the node. The data packets were lost in the simulation process. So, the throughput is decreased. The untrusted network is shown in Fig.4. The leaf node will send a data to the head node. Bridge and switch is used to connect to the other network. Head node will send through bridge, switch and router to the other interconnecting network. The leaf node will send a data. At the same time, if the node fails or moves away, then the proxy node will act like an existing node. Then the data will be corrupted.

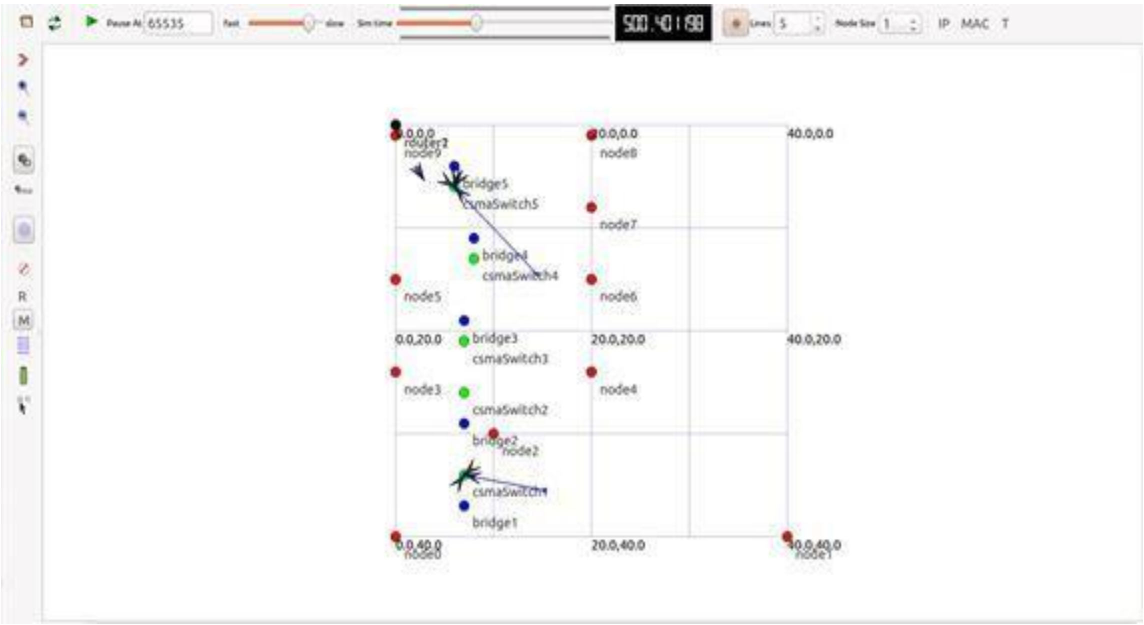
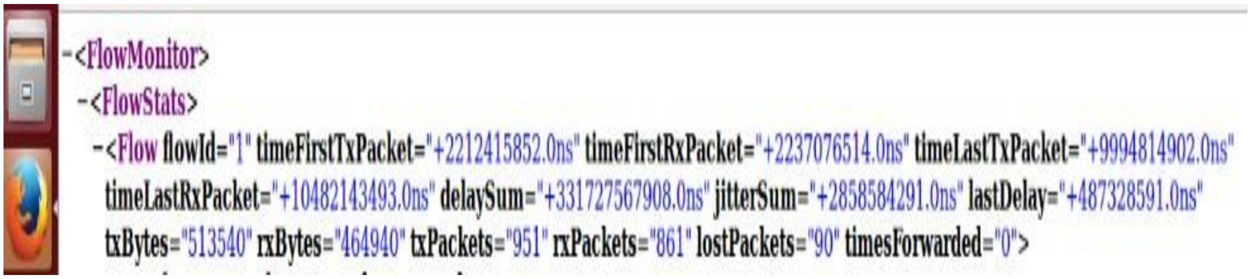


Fig.4. Untrusted Network

1. . FLOW MONITOR

The node moves away from the communication range, then it will not send or receive any data. So, the packet loss is occurs in untrusted network. There were 90 packets loss during the movement of the node. Because of the packet loss, delay is more. Flow monitor for untrusted network is shown in Fig.5.

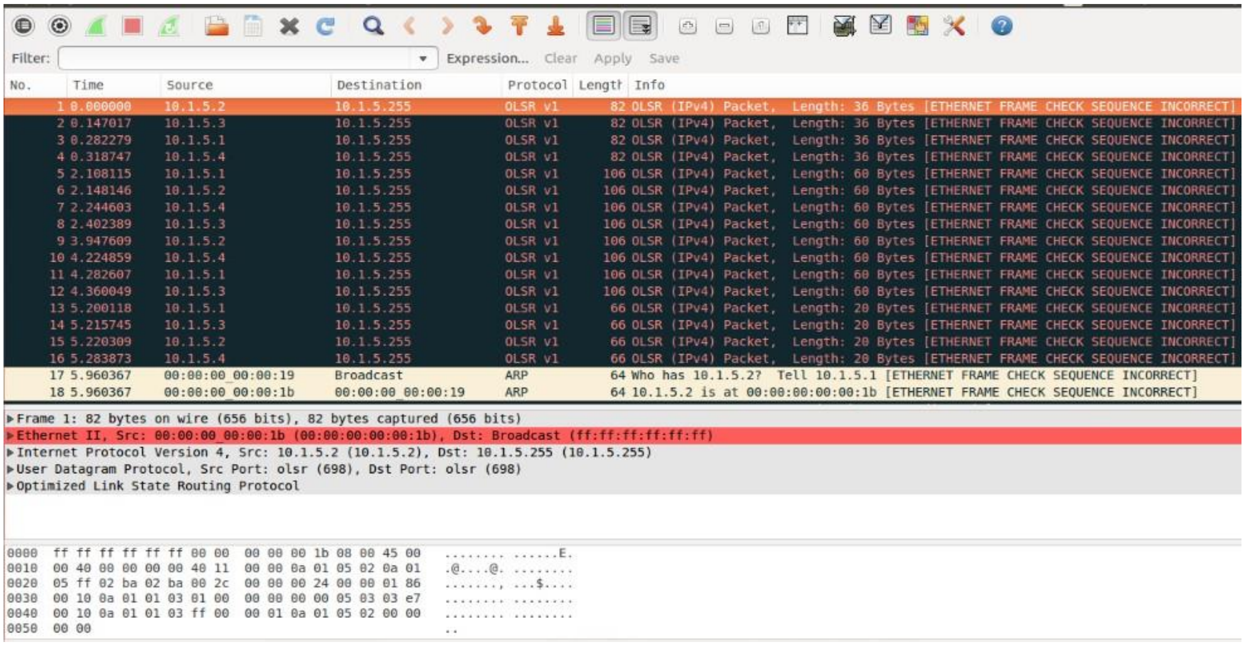


```
-<FlowMonitor>
 -<FlowStats>
  -<Flow flowId="1" timeFirstTxPacket="+2212415852.0ns" timeFirstRxPacket="+2237076514.0ns" timeLastTxPacket="+9994814902.0ns"
   timeLastRxPacket="+10482143493.0ns" delaySum="+331727567908.0ns" jitterSum="+2858584291.0ns" lastDelay="+487328591.0ns"
   txBytes="513540" rxBytes="464940" txPackets="951" rxPackets="861" lostPackets="90" timesForwarded="0">
```

Fig.5. Flow monitor for untrusted network

2. PCAP FROM WIRESHARK RESULT FOR TRUSTED NETWORK

The PCAP file will show the number of packets transmitted and received by the node. The untrusted network, the node moves away the range it cannot communicate with the other node. The leaf node will communicate to the other network through the head and broker nodes. Then the node will try to send the routing packets to the network. The Fig.6 shows the PCAP for untrusted network. The node will broadcast the OLSR packets.he length of the OLSR packets is 82, 106, and 66.

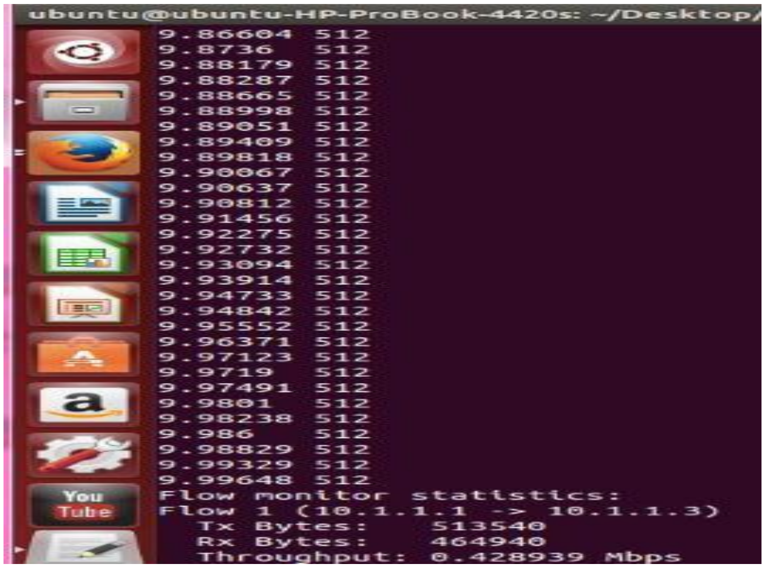


```
No.   Time       Source               Destination          Protocol  Lengtł Info
    1 0.000000   10.1.5.2             10.1.5.255           OLSR v1      82 OLSR (IPv4) Packet,  Length: 36 Bytes [ETHERNET FRAME CHECK SEQUENCE INCORRECT]
    2 0.147017   10.1.5.3             10.1.5.255           OLSR v1      82 OLSR (IPv4) Packet,  Length: 36 Bytes [ETHERNET FRAME CHECK SEQUENCE INCORRECT]
    3 0.282279   10.1.5.1             10.1.5.255           OLSR v1      82 OLSR (IPv4) Packet,  Length: 36 Bytes [ETHERNET FRAME CHECK SEQUENCE INCORRECT]
    4 0.318747   10.1.5.4             10.1.5.255           OLSR v1      82 OLSR (IPv4) Packet,  Length: 36 Bytes [ETHERNET FRAME CHECK SEQUENCE INCORRECT]
    5 2.108115   10.1.5.1             10.1.5.255           OLSR v1     106 OLSR (IPv4) Packet,  Length: 60 Bytes [ETHERNET FRAME CHECK SEQUENCE INCORRECT]
    6 2.148146   10.1.5.2             10.1.5.255           OLSR v1     106 OLSR (IPv4) Packet,  Length: 60 Bytes [ETHERNET FRAME CHECK SEQUENCE INCORRECT]
    7 2.244603   10.1.5.4             10.1.5.255           OLSR v1     106 OLSR (IPv4) Packet,  Length: 60 Bytes [ETHERNET FRAME CHECK SEQUENCE INCORRECT]
    8 2.402389   10.1.5.3             10.1.5.255           OLSR v1     106 OLSR (IPv4) Packet,  Length: 60 Bytes [ETHERNET FRAME CHECK SEQUENCE INCORRECT]
    9 3.947609   10.1.5.2             10.1.5.255           OLSR v1     106 OLSR (IPv4) Packet,  Length: 60 Bytes [ETHERNET FRAME CHECK SEQUENCE INCORRECT]
   10 4.224859   10.1.5.4             10.1.5.255           OLSR v1     106 OLSR (IPv4) Packet,  Length: 60 Bytes [ETHERNET FRAME CHECK SEQUENCE INCORRECT]
   11 4.282607   10.1.5.1             10.1.5.255           OLSR v1     106 OLSR (IPv4) Packet,  Length: 60 Bytes [ETHERNET FRAME CHECK SEQUENCE INCORRECT]
   12 4.360049   10.1.5.3             10.1.5.255           OLSR v1     106 OLSR (IPv4) Packet,  Length: 60 Bytes [ETHERNET FRAME CHECK SEQUENCE INCORRECT]
   13 5.200118   10.1.5.1             10.1.5.255           OLSR v1      66 OLSR (IPv4) Packet,  Length: 20 Bytes [ETHERNET FRAME CHECK SEQUENCE INCORRECT]
   14 5.215745   10.1.5.3             10.1.5.255           OLSR v1      66 OLSR (IPv4) Packet,  Length: 20 Bytes [ETHERNET FRAME CHECK SEQUENCE INCORRECT]
   15 5.220309   10.1.5.2             10.1.5.255           OLSR v1      66 OLSR (IPv4) Packet,  Length: 20 Bytes [ETHERNET FRAME CHECK SEQUENCE INCORRECT]
   16 5.283873   10.1.5.4             10.1.5.255           OLSR v1      66 OLSR (IPv4) Packet,  Length: 20 Bytes [ETHERNET FRAME CHECK SEQUENCE INCORRECT]
   17 5.960367   00:00:00_00:00:19    Broadcast            ARP          64 Who has 10.1.5.2?  Tell 10.1.5.1 [ETHERNET FRAME CHECK SEQUENCE INCORRECT]
   18 5.960367   00:00:00_00:00:1b    00:00:00_00:00:19    ARP          64 10.1.5.2 is at 00:00:00:00:00:1b [ETHERNET FRAME CHECK SEQUENCE INCORRECT]

▶Frame 1: 82 bytes on wire (656 bits), 82 bytes captured (656 bits)
▶Ethernet II, Src: 00:00:00_00:00:1b (00:00:00:00:00:1b), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
▶Internet Protocol Version 4, Src: 10.1.5.2 (10.1.5.2), Dst: 10.1.5.255 (10.1.5.255)
▶User Datagram Protocol, Src Port: olsr (698), Dst Port: olsr (698)
▶Optimized Link State Routing Protocol

0000  ff ff ff ff ff ff 00 00  00 00 00 1b 08 00 45 00   ........ ......E.
0010  00 40 00 00 00 00 40 11  00 00 0a 01 05 02 0a 01   .@....@. ........
0020  05 ff 02 ba 02 ba 00 2c  00 00 00 24 00 00 01 86   ......., ...$....
0030  00 10 0a 01 01 03 01 00  00 00 00 00 05 03 03 e7   ........ ........
0040  00 10 0a 01 01 03 ff 00  00 01 0a 01 05 02 00 00   ........ ........
0050  00 00                                              ..
```

Fig.6. PCAP for untrusted network

3. TERMINAL OUTPUT

The number of transmitted and received bytes are shown in the terminal using flow monitor. The node statistics is also available in the terminal. There were 512 data packets is received. The untrusted network, the throughput is decreased because of packet loss. Fig.7 shows the terminal for untrusted network.



```
ubuntu@ubuntu-HP-ProBook-4420s: ~/Desktop/
9.86604  512
9.8736   512
9.88179  512
9.88287  512
9.88665  512
9.88998  512
9.89051  512
9.89409  512
9.89818  512
9.90067  512
9.90637  512
9.90812  512
9.91456  512
9.92275  512
9.92732  512
9.93094  512
9.93914  512
9.94733  512
9.94842  512
9.95552  512
9.96371  512
9.97123  512
9.9719   512
9.97491  512
9.9801   512
9.98238  512
9.986    512
9.98829  512
9.99329  512
9.99648  512
Flow monitor statistics:
Flow 1 (10.1.1.1 -> 10.1.1.3)
  Tx Bytes:    513540
  Rx Bytes:    464940
  Throughput:  0.428939 Mbps
```

Fig.7. Terminal output for untrusted network

C. TRUSTED NETWORK

The trusted network has the backup nodes. If the node is moved away from the network, then the backup node will replace the existing node. The proxy node identification is done by using distance and angle estimation. The angle and distance estimation is calculated for every node. The permitted tolerance value is fixed. The sum of estimated value and permitted tolerance is calculated. If the node which is having a value greater than the estimated value is considered as a proxy node. The node failure and movement of node problem can be replaced by the free node in trusted network. The packet loss is minimized, because the backup node.so, the throughput is also increased. In the trusted network the characteristics were taken and according to that the proxy node is analysed. The authentication, confidentiality and integrity were analysed according to the success rate, retransmission, lifetime, active links, packets handled and neighbor threshold. The Fig.8 shows the trusted network.

DOI: https://doi.org/10.36548/jsws.2019.1.004

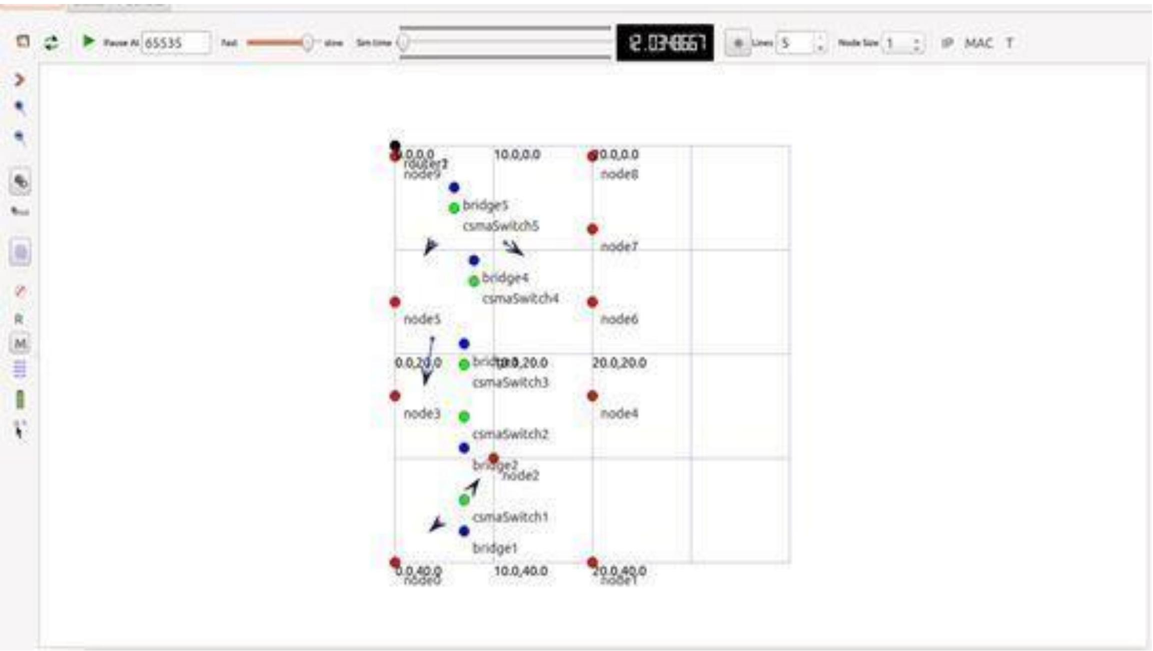
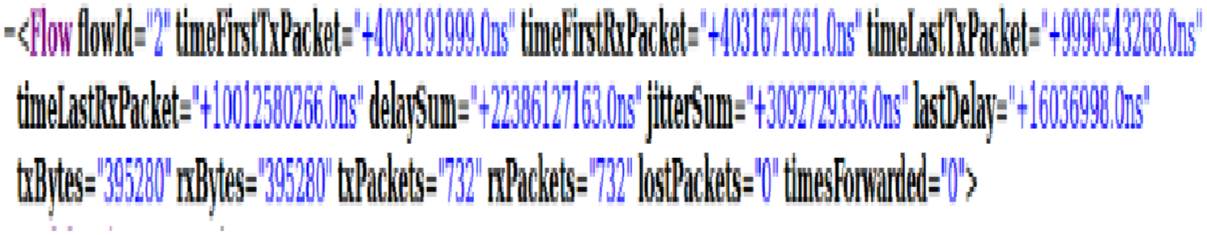


Fig.8. Trusted network

1. FLOW MONITOR RESULT

The transmitted and received packets are shown in flow monitor. The node is identified as proxy node then the free node will take over the function of existing node. The packet loss is minimum. The Fig.9 shows flow monitor for trusted network.

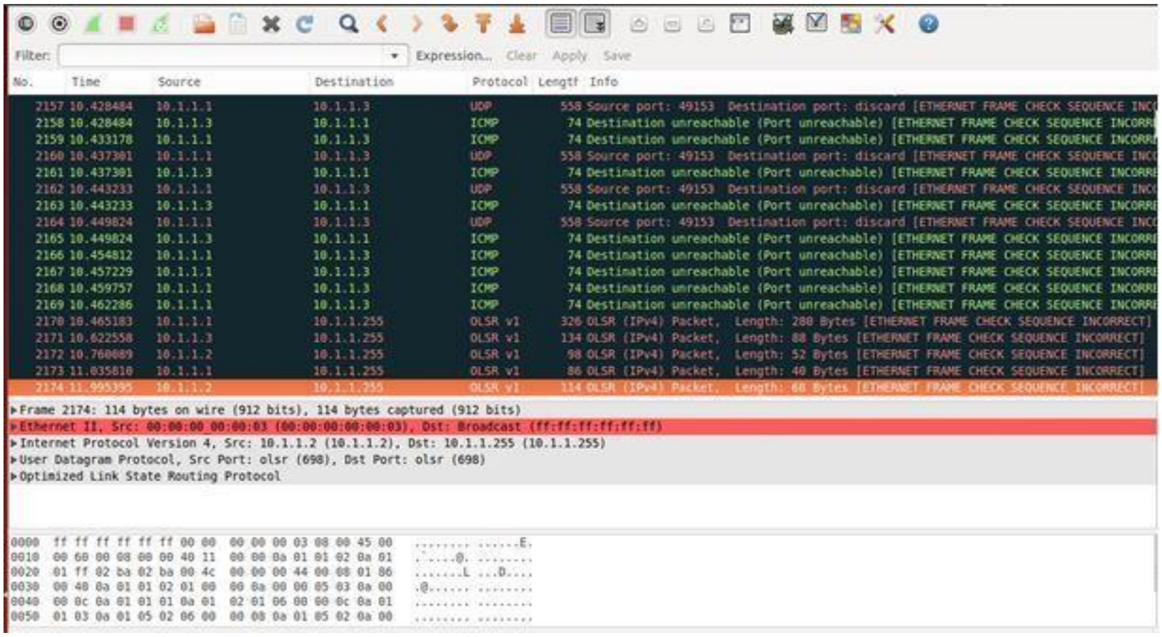


```
-<Flow flowId="2" timeFirstTxPacket="+4008191999.0ns" timeFirstRxPacket="+4031671661.0ns" timeLastTxPacket="+9996543268.0ns"
 timeLastRxPacket="+10012580266.0ns" delaySum="+22386127163.0ns" jitterSum="+3092729336.0ns" lastDelay="+16036998.0ns"
 txBytes="395280" rxBytes="395280" txPackets="732" rxPackets="732" lostPackets="0" timesForwarded="0">
```

Fig.9. Flow monitor for trusted network

2. PCAP FROM WIRESHARK RESULT FOR TRUSTED NETWORK

The wireshark will show the sending and receiving of data packet in the networks. In this the data packets were sent. The general scenario the node will send the UDP packets to the destination node. Once the node is moving from the range of communication, then the node will drop the packets. At the time the backup node will send a data to the destination. The moved node will broadcast the OLSR packets to the networks. The PCAP for trusted network is shown in Fig.10.

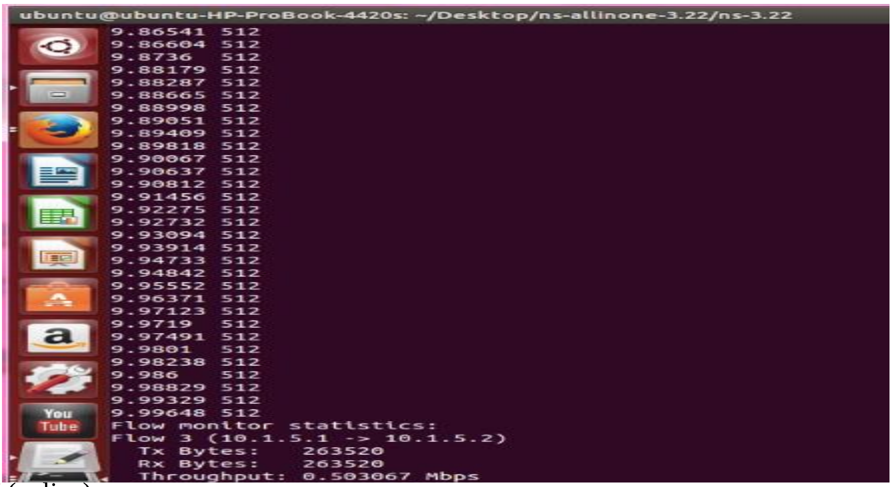


```
No.   Time        Source      Destination   Protocol Lengtl Info
 2157 10.428484   10.1.1.1    10.1.1.3      UDP       558 Source port: 49153  Destination port: discard [ETHERNET FRAME CHECK SEQUENCE INCO
 2158 10.428484   10.1.1.3    10.1.1.1      ICMP       74 Destination unreachable (Port unreachable) [ETHERNET FRAME CHECK SEQUENCE INCORR
 2159 10.433178   10.1.1.1    10.1.1.3      ICMP       74 Destination unreachable (Port unreachable) [ETHERNET FRAME CHECK SEQUENCE INCORR
 2160 10.437301   10.1.1.1    10.1.1.3      UDP       558 Source port: 49153  Destination port: discard [ETHERNET FRAME CHECK SEQUENCE INCO
 2161 10.437301   10.1.1.3    10.1.1.1      ICMP       74 Destination unreachable (Port unreachable) [ETHERNET FRAME CHECK SEQUENCE INCORR
 2162 10.443233   10.1.1.1    10.1.1.3      UDP       558 Source port: 49153  Destination port: discard [ETHERNET FRAME CHECK SEQUENCE INCO
 2163 10.443233   10.1.1.3    10.1.1.1      ICMP       74 Destination unreachable (Port unreachable) [ETHERNET FRAME CHECK SEQUENCE INCORR
 2164 10.449824   10.1.1.1    10.1.1.3      UDP       558 Source port: 49153  Destination port: discard [ETHERNET FRAME CHECK SEQUENCE INCO
 2165 10.449824   10.1.1.3    10.1.1.1      ICMP       74 Destination unreachable (Port unreachable) [ETHERNET FRAME CHECK SEQUENCE INCORR
 2166 10.454812   10.1.1.1    10.1.1.3      ICMP       74 Destination unreachable (Port unreachable) [ETHERNET FRAME CHECK SEQUENCE INCORR
 2167 10.457229   10.1.1.1    10.1.1.3      ICMP       74 Destination unreachable (Port unreachable) [ETHERNET FRAME CHECK SEQUENCE INCORR
 2168 10.459757   10.1.1.1    10.1.1.3      ICMP       74 Destination unreachable (Port unreachable) [ETHERNET FRAME CHECK SEQUENCE INCORR
 2169 10.462286   10.1.1.1    10.1.1.3      ICMP       74 Destination unreachable (Port unreachable) [ETHERNET FRAME CHECK SEQUENCE INCORR
 2170 10.465183   10.1.1.1    10.1.1.255    OLSR v1   326 OLSR (IPv4) Packet,  Length: 280 Bytes [ETHERNET FRAME CHECK SEQUENCE INCORRECT]
 2171 10.622558   10.1.1.3    10.1.1.255    OLSR v1   134 OLSR (IPv4) Packet,  Length: 88 Bytes [ETHERNET FRAME CHECK SEQUENCE INCORRECT]
 2172 10.760089   10.1.1.2    10.1.1.255    OLSR v1    98 OLSR (IPv4) Packet,  Length: 52 Bytes [ETHERNET FRAME CHECK SEQUENCE INCORRECT]
 2173 11.035810   10.1.1.1    10.1.1.255    OLSR v1    86 OLSR (IPv4) Packet,  Length: 40 Bytes [ETHERNET FRAME CHECK SEQUENCE INCORRECT]
 2174 11.995395   10.1.1.2    10.1.1.255    OLSR v1   114 OLSR (IPv4) Packet,  Length: 68 Bytes [ETHERNET FRAME CHECK SEQUENCE INCORRECT]

▸Frame 2174: 114 bytes on wire (912 bits), 114 bytes captured (912 bits)
▸Ethernet II, Src: 00:00:00_00:00:03 (00:00:00:00:00:03), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
▸Internet Protocol Version 4, Src: 10.1.1.2 (10.1.1.2), Dst: 10.1.1.255 (10.1.1.255)
▸User Datagram Protocol, Src Port: olsr (698), Dst Port: olsr (698)
▸Optimized Link State Routing Protocol

0000  ff ff ff ff ff ff 00 00  00 00 00 03 08 00 45 00   ........ ......E.
0010  00 60 00 08 00 00 40 11  00 00 0a 01 01 02 0a 01   .`....@. ........
0020  01 ff 02 ba 02 ba 00 4c  00 00 00 44 00 08 01 86   .......L ...D....
0030  00 40 0a 01 01 02 01 00  00 0a 00 00 05 03 0a 00   .@...... ........
0040  00 0c 0a 01 01 01 0a 01  02 01 06 00 00 0c 0a 01   ........ ........
0050  01 03 0a 01 05 02 06 00  00 08 0a 01 05 02 0a 00   ........ ........
```

Fig.10. PCAP for trusted network

3. TERMINAL OUTPUT

The throughput is analysed by using flow monitor. There were 512 bytes were transmitted. The proxy node and node movement problem is overcome by the trusted network. The trusted network there were no packet loss and also the throughput is increased. The Fig.11 shows the terminal of trusted network.



```
ubuntu@ubuntu-HP-ProBook-4420s: ~/Desktop/ns-allinone-3.22/ns-3.22
9.86541 512
9.86604 512
9.8736  512
9.88179 512
9.88287 512
9.88665 512
9.88998 512
9.89051 512
9.89409 512
9.89818 512
9.90067 512
9.90637 512
9.90812 512
9.91456 512
9.92275 512
9.92732 512
9.93094 512
9.93914 512
9.94733 512
9.94842 512
9.95552 512
9.96371 512
9.97123 512
9.9719  512
9.97491 512
9.9801  512
9.98238 512
9.986   512
9.98829 512
9.99329 512
9.99648 512
Flow monitor statistics:
Flow 3 (10.1.5.1 -> 10.1.5.2)
  Tx Bytes:   263520
  Rx Bytes:   263520
  Throughput: 0.503067 Mbps
```

Fig.11. Terminal output for trusted network

DOI: https://doi.org/10.36548/jsws.2019.1.004

D. ANALYSIS OF PACKET DELIVERY RATIO

Packet delivery ratio is defined as the ratio of delivered data packet to the destination. This illustrates the level of delivered data to the destination. The table.4.1 shows the Packet Delivery Ratio.

$$PDR = \Sigma \text{ Number of packet receive} / \Sigma \text{ Number of packets send} \qquad (4.1)$$

Table 1.1 Packet delivery ratio

PARAMETERS	**TRUSTED NETWORK**	**UNTRUSTED NETWORK**
Number of packet send (Bytes)	263520	513540
Number of packet received (Bytes)	263520	464940
Packet Delivery ratio (%)	100	90.5

E. ANALYSIS OF THROUGHPUT

Throughput is the rate of successful message delivery over a communication channel. Throughput is usually measured in bits per second , and sometimes in data packets per second (p/s or pps). The Fig.12 shows throughput. The trusted and untrusted network values were taken. The node moves away then the throughput is decreased. In the trusted network, the backbone node will operate at the time of the moved node and the throughput is getting increased.

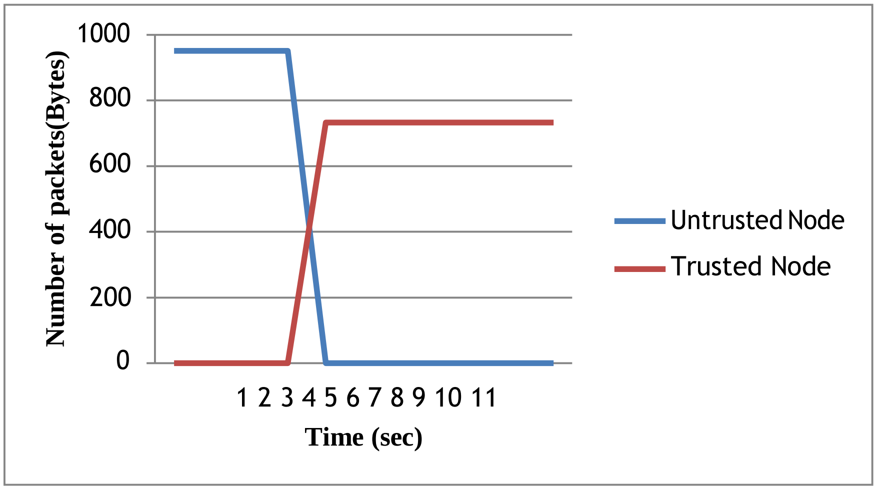


Fig.12. Throughput

F. SIMULATION PARAMETERS

Jitter is a variation in packet transit delay caused by queuing, contention and serialization effects on the path through the network. In heavily congested link, the higher levels of jitter will be more. The table.4.2 shows the simulation parameters. The delay will specifies the time taken for a bit of data to travel the network from one node to another. It is measured in multiples or fractions of seconds. By the location of the communicating nodes delay may vary.

The packet loss happens in the untrusted network, because of lacking the backbone node. The maximum throughput and minimum packet loss is occurs in trusted network.

TABLE.1.2 SIMULATION PARAMETERS

PARAMETERS	UNTRUSTED NETWORK	TRUSTED NETWORK
Throughput	0.428 Mbps	0.503 Mbps
Delay	3.31 s	2.2 s
Packet loss	90 packets	0
Jitter	2.8 s	2.09 s

V. CONCLUSION

The proxy node was identified using distance and angle estimation. The node velocity is 0.01m/sec. If the node movement velocity is more, then the data transmission will be difficult. In this trust based system having the free node to overcome the problem of proxy node identification and node movement. The computation time is more because of replacing node in the system. To overcome the problem of link detection we need an efficient trust based system. For that the keying and security methods to be used. The proxy node and node movement occurs in the untrusted system. The problem is overcome with the help of a trusted system by using the backup nodes and routing algorithm. On doing this, it was found that the PDR and throughput were found to be high and better for the trust based system.

VI. REFERENCES

[1] Michele Nitti, Roberto Girau, and Luigi Atzori, ” Trustworthiness management in the social internet of things”, IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, VOL. 26, NO. 5, MAY 2014,(PP:1253-1266).

[2] Ing-Ray Chen, Fenye Bao, and Jia Guo, “Trust-based service management for social internet of things systems”, IEEE Transactions on Dependable and Secure Computing, 2014.

[3] Yan Liu Kun Wang, ” Trust control in heterogeneous networks for internet of things“,Computer Application and System Modeling, 2010, (PP:632-636).

[4] Ing-Ray Chen, Jia Guo, and Fenye Bao, “Trust management for SOA-based IoT and its application to service composition” IEEE Transactions on Services Computing, 2014.

[5] GU Lizet, WANG JingpeP, SUN Bi,” Trust management mechanism for internet of things, China Communications, February 2014, (PP:148-156).

[6] Kai Kang, Zhibo Pang, Li Da Xu, Liya Ma, and Cong Wang, “An interactive trust model for application market of the Internet of Things”, IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS, VOL. 10, NO. 2, MAY 2014, (PP:1516-1526).

[7] Burak Kantarci, Hussein T. Mouftah, “Trustworthy sensing for public safety in cloud-centric internet of things”, IEEE INTERNET OF THINGS JOURNAL, VOL. 1, NO. 4, August 2014, (PP: 360-368).

[8] Navroop Kaur and Sandeep K. Sood,” An energy-efficient architecture for the internet of

things (IoT)", IEEE SYSTEMS JOURNAL,2015, (PP:1-9).

[9] Saima Abdullah · Kun Yang, "An energy efficient message scheduling algorithm considering node failure in IoT environment", Wireless Pers Commun, 79 (2014), (PP:1815–1835).

[10] Fagen Li, Pan Xiong, "Practical secure communication for integrating wireless sensor networks into the internet of things", IEEE SENSORS JOURNAL, VOL. 13, NO. 10, October 2013, (PP: 3677-3684).

[11] Maria Rita Palattella, Luigi Alfredo Grieco,Thomas Engel, "On optimal scheduling in duty-cycled industrial iot applications using ieee802.15.4e tsch", IEEE SENSORS JOURNAL, VOL. 13, NO. 10, OCTOBER 2013, (PP:3655-3666).

[12] Reema Sharma, Navin Kumar, Namratha B Gowda," Probabilistic prediction based scheduling for delay sensitive traffic in internet of things" Procedia Computer Science 52(2015), (PP:90 – 97).