A. CHAPTER 4

B. Result Analysis of Three Tier Trust Manager

This chapter presents analysis of the results which are implemented in the previous chapter. Also analysis of trust variables like reputation and reliability is presented in this chapter. The Proposed Algorithm presents the design and implementation of a policy enforcing mechanism based on a kernel-level trusted execution monitor.

4.1 Simulation

The Proposed algorithms are implemented using OPNET Modeler 14.0. This modeler has a powerful GUI and a rich library of different protocols and functions. The algorithm is implemented in different levels the trust tier is implemented using state transition diagram [19]. Each processor modules include a trust model implemented using state transition diagram. Each wireless node includes processor module. So there is a hierarchy in the implementation.

This chapter includes the results and the statistics collected after the simulation. The implementation of the presented algorithm was completed and the simulation of one of the attacks was implemented at the trusted tier at the kernel level. For routing tier types of routing AODV, DSR, and OLSR were used. With each routing algorithm the no. of nodes were 25, 50, 100.

With start of simulation a pair of source and destination nodes was selected. Then after starting the simulation the randomized function selected any two nodes randomly. These nodes were selected for misbehaving or attack. Then after completion after completion of the simulation calculation following attributes was done: total packets dropped Delay Traffic received (packets/sec), Traffic sent (Packets/sec), Load (bits/sec), Throughput (bits/sec).
4.2 Types of attacks implemented

Following Scenarios are considered for attack analysis.

**Eavesdrop Method**: It is the simplest approach which intercepts all traffic from two nodes A to B on the local interface of the intermediate node P.

**Route Invasion**: If the route from source to destination exists, then the attacker first generates a False_Request basic attack with a larger sequence number for destination. It will make all nodes, including source, update their routes to destination using current node as the next hop. Then, the attacker generates a second False_Request attack, which will launch a route discovery process to establish the route from current to destination. Eventually, current node will be injected in the route from source to destination.

The figure 4.1 shows the values of successful and unsuccessful interactions based on the equation 3.1 and 3.2. Nearest integer function for trust is calculated for 360 minute.

![Figure 4.1 values of successful and unsuccessful interactions.](image)
4.2.1 Throughput Comparison against attacks

A comparison of throughput is presented in figure 6 with varying number of nodes from 10 to 100. The throughput is good in case of no attack with an average of 93.8 \%. While with eaves drop method there was an average throughput 90.5 \%. With route invasion there is an average throughput 89.2 \%. In both type of attack similar results were obtained.

4.2.2 Trust Implementation Overhead: this section calculates the overhead for implementation as the round trip time taken to send a join request then calculation of trust \( T_c \) for the requested node after that round time to reply the requested node with trust implementation.

\[
TIO = \frac{(T_{n2} - T_{n1})}{(T_{n2} - T_{n1} + T_c)} \times 100
\]

Eq. 4.1

\( T_{n2} \) is the time at which reply was received the request. \( T_{n1} \) is the time at which request was sent. The network was implemented with a network size in meters. The trust implementation overhead was calculated in the stationary network and mobile
With stationary network the overhead increases with distance. But with mobility the overhead percentage varies from 12 to 7%. The overhead % in randomwalk [92] with speed 4.5 m/s is 18%. MTTM has an average throughput of 13%.

With a speed of 1.5 m/s MTTM has an average throughput 4.77% whereas average throughput in [92] is 9%.
Figure 4.5 and 4.6 presented the results for AODV, DSR and OLSR routing. Here 25, 50 and 100 nodes have been taken for consideration. With AODV routing an average throughput of 90% with range of 88-92% was obtained. Here with 25 nodes an average false positive of 13.5, with 50 nodes an average false positive of 14.6 and with 100 nodes an average false positive 18 and an average throughput 87% was calculated. In AODV it was observed that with increasing number of nodes the average throughput decreases and the false positive rate increases. With DSR protocol scenarios 25, 50 and 100 nodes are included. In DSR protocol and 25 nodes an average throughput of 91% and an average false positive rate 9.9 was obtained. With 50 nodes an average throughput 91.5 and average false Positive of 12.5 was received. For 100 nodes an average throughput 84.8 and false positive rate of 10.5 was obtained. With DSR protocol it was observed that the throughput is decreasing with no of nodes and false positive rate increases. However as compared with AODV throughput is same as DSR but False positive rate is low. For 25 nodes in OLSR an average throughput of 92.8 and a false positive rate of 4.1 were received. With 50 nodes an average throughput of 91.1 and 7.8 false positive rate was obtained. For
100 nodes an average throughput of 91.1 and a false positive of 7.6 were calculated. Here it was observed that the throughput remains same there is a quite negligible change in throughput. Comparing with the false positive rate was seen high in AODV, lesser in DSR and least in OLSR algorithm.

4.2.3 Comparison of Throughput with Different IDS

The Figure 4.6 has presented a comparison of throughput intrusion detection of our system by using 100 nodes. The comparison is among different intrusion detection systems. The average throughput of CONFIDANT protocol is 86.5 %. With CORE the average throughput obtained was 85.2 and CORMAN IDS gives average throughput 84.6 with 100 nodes. Policy enforcement mechanism has an average throughput of 90.8%. With MTTM detection rate is always above 90 percentages and an average throughput 93.6%. But the detection of PEM methods ranges from 82-99 %.

Figure 4.6 Comparison of throughput with other IDS.
4.3 Conclusion

This chapter included the result analysis of Multitier Trusted method. The implementation was completed with OPNET simulator. The simulation creates a process model which consists of state transition Diagram. This Process model is included in Node Model and Node Model is included inside a MANET node. The simulation was run for variable time duration for which result is included.