CHAPTER 6

AN IMPROVED TRUST MODEL FOR MITIGATING DISHONEST RECOMMENDATION PROBLEM

This chapter presents a probability based trust model which uses recommendations from other nodes. Here, the possible attacks that can occur due to dishonest nodes while taking recommendation are investigated. Here, a filtering algorithm is proposed for resisting the effect of misbehaving nodes while taking recommendations for selecting valid nodes for recommendation.

6.1 Introduction

Many researchers have given the definition of trusts in different terms such as risk, belief, probability, quality of services etc. Also the trust can be achieved through different ways such as using reputation function, calculating direct trust, calculating trust through recommendation. Many researchers have used recommendation based trust model for filtering the malicious nodes. Li *et al.* (2009) proposed recommendation based trust model to screen the misbehaving nodes. But while taking recommendation from other nodes, it is difficult to filter out the malicious nodes from recommendation and dishonest recommendation may yield wrong trust. Dishonest recommendation may cause various types of attacks like collusion, bad mounting and ballot stuffing which may cause malfunctioning of trust frameworks. Some researchers have given various solutions for filtering the misbehaving nodes while recommending the trustor nodes but these are very limited and not much effective. Yu *et al.* (2011) proposes one approach to judge the goodness of recommending nodes by taking majority opinion from the recommenders. In this case screening out the malicious nodes is difficult when dishonest recommenders collude each other to perform a malicious attack. Zouridaki *et al.* (2005) proposes service reputation approach which uses recommendations from nodes which have higher trust values. But a node can be more trusted in terms of packet forwarding while may not be a trusted node for recommendation. Buchegger and Boudee (2005) proposed a trust model based on experiences to screen out the malicious nodes which are not compatible to evaluating nodes. This approach may not work when the trustee node has no prior experiences to the trust evaluating node. Thus resulting a confusing trust model for evaluating nodes trustiness.

In this chapter a recommendation based trust model is presented which excludes dishonest recommendation for evaluating trust. Here first we consider the majority rule to assure the consistency of recommendations in terms of time and location, personal experience based rule to assure the consistency of recommendation with experience held by the evaluating nodes and the service reputation based rule to assure the honesty of the recommender nodes. Second, a defense scheme is proposed to estimate the trustiness of the recommender using the parameters number of interactions, compatibility to the evaluating node using deviation test and association between nodes. The defense scheme considers social properties for estimating trustiness. For computing the performance of the algorithm different mobile nodes are taken into consideration against different mobile topologies.

6.2 Different types of attack related to recommendation in trust frameworks

In present days it's a great challenge to safeguard a network against various attacks. Recent researches are going on to tackle the dishonest nodes in terms of packet forwarding such as blackhole, wormhole, grayhole etc. It is also of great importance that trusts management frameworks are prone to attack by means of dishonest recommendations. There are various attacks at the time of recommendation propagation and aggregation such as bad mouthing attack, intelligent behaviour attack, ballot stuffing attack, selective misbehaviour attack, time-dependent attack and location-dependent attack. The attacks are explained below:

a) Bad Mouthing Attack (BMA): In BMA the colluding nodes gives negative ratings to good nodes in order to fade the reputation of the nodes. Such malicious behaviour confused the trust management framework.
b) Ballot Stuffing Attack (BSA): In BSA the collusive nodes mislead the trust mechanism by propagating fake positive rating to some low performing nodes.
c) Selective Misbehaviour Attack (SMA): This attack ill-treats some selective good nodes by false rating, but act normally to other nodes. These types of nodes are very difficult to detect for trust mechanism.

d) Intelligent Behaviour Attack (IBA): IBA gives high or low rating recommendation as per threshold. Such type of attack can perform malicious activity by dynamically responding to the threshold values.

e) Time-dependent Attack (TDA): In this type of attack nodes behave correctly for a certain period of time but change their behavior at other times.

f) Location-dependent Attack (LDA): In this type of attack nodes act differently at different location which affect the mobility property of MANET.

Hoffman *et al.* (2009) give a solution to such type of attack by using Bayesian statistical theory for computing the correctness of the recommendations. Some researchers have used majority opinion technique, fixing a threshold for positive and negative recommendation, sufficient interaction history etc. to correctly evaluate the honesty of the recommendation nodes but not sufficient till today. From the literature study we can say that trustiness of recommending nodes cannot be measured by a single approach, it should be done by using multiple properties like time, location, closeness between nodes, which is not present in the literature. With the purpose to improve the correctness and robustness of trust model, the effect of untrusted recommendation should be avoided.

6.3 The Proposed Trust Model

This section presents the proposed trust model which uses both direct and recommendation based trust value of each node to secure the MANET routing protocol. The proposed model has taken into consideration about the attacks discussed above occurs due to some dishonest nodes. The model used the same trust evaluating function for evaluating the trust value of each node, where we have considered both direct and indirect trust calculation function. The model addressed two types of attacks bad mounting and ballot stuffing to assess the functionality of the model. The bad mouthing and ballot stuffing are two types of attacks which occurred due to dishonest recommendation problem. The proposed model uses two components namely Trust Computation Module and Recommendation Manager Model.

A. Trust Computation Module

The Trust Computation Module uses direct as well as indirect trust for calculating the trust as explained below. The trust model uses direct trust when there is a pre initiated trust relationship between the trustor and trustee nodes. The model also uses a factor (µ) which gradually either increase for positive interaction or decrease for negative interaction, also it gradually decay as experience become past based on time. The trust computation module uses indirect trust when there is no direct trust value that there is no previous trust relationship exists. In such cases taking reference is important for calculating the trustiness of the nodes, but taking recommendation from all nodes may leads to malicious attacks. The attacker nodes may intentionally propagate dishonest recommendation for referring wrong route. The final trust value is calculated by combining both the direct and indirect trust.

The direct trust can be calculated using the following formula

$$DT_{ij} = \frac{X_{ij}}{X_{ij} + Y_{ij}} \qquad \text{Where,} \quad 0 \leq DT \leq 1 \qquad (6.1)$$

The recommendation trust can be calculated using the formula

$$RT_{ij} = \frac{\sum_{k=1}^{n} X_{kj}}{\sum_{k=1}^{n} (X_{kj} + Y_{kj})} \qquad \text{Where, } 0 \leq RT \leq 1 \qquad (6.2)$$

Indirect trust is calculated based on the sum of received recommendation in the form of ratings(X_{kj} ,Y_{kj}).

The Final Trust will be calculated as

$$FT_{ij} = W_d * DT_{ij} + W_i * RT_r \text{ Where,} 0 \leq DT \leq 1 \text{ , } 0 \leq RT \leq 1 \text{, and } DT + RT = 1 \qquad (6.3)$$

B. Recommendation Manager Module

The recommendation manager module sends recommendation request and collect recommendations for a node from a list of recommender nodes. It is designed mainly to detect and exclude the false recommendations. The recommendation manager module first sends recommendation request to the evaluating node's neighbours;

gathered received recommendations from the neighbours and runs the filtering algorithm. After running the filtering algorithm it sent back to the trust computation module a list of honest recommendations. Finally using the honest recommendations the evaluating node compute the trust value of a particular node.

Algorithm 6.1: Recommendation Manager Algorithm

```
Step 1: For each RecRequest from S the Recommendation Manager broadcasts
    {
Step 2: RecRequest → neighbours                    // Recommendation Request
Step 3: Gather received recommendation
Step 4: Construct RL= {n1, n2, n3, ………}             //Recommender nodes list
Step 5: Run the filtering algorithm
Step 6: Send the honest recommendation list to S
}
```

6.4 Selection of honest recommenders

In this section we have used multiple rules to select recommenders. These rules include majority opinion rule, personal knowledge rule and service reputation rules. These rules are used in combined to filter out the dishonest recommenders.

Majority Opinion Rule

In majority opinion rule, the trust approach takes majority opinion from all recommendations and categorizes them as per deviation test. The node whose deviation is very high from the majority opinion is treated as dishonest and ignores them from trust calculation.

Personal Knowledge Rule

The personal knowledge rule considers the node as malicious if its value deviates much from the opinion of the evaluating node. This rule applies deviation test to the receiving recommendations and eliminates those recommendations which have higher deviation value than the predefined threshold.

Service Reputation Rule

The service reputation rule consider that there is uniformity between service providing and recommendation. The recommendation from a reputed node is considered more trustworthy and given more weight while taken for evaluating trust. The recommendation can be taken on the basis of rank of the nodes. The nodes whose service reputation is more will be ranked as higher ranked nodes and will be given higher weightage in calculating recommendation.

Algorithm 6.2: Filtering Algorithm for honest recommendations

```
Step 1: For each recommendation list $L$ Do
Step 2: For each rating vector in the list (x,y) Do
Step 3: Calculate trust value for the recommender as equation (6.2)
Step 4: Calculate deviation values from all the recommenders    $d_{ik} = |DT_{ij} - RT_{kj}|$
Step 5: Rank the deviations as per $d_{ik}$ , lowest $d_{ik}$ as highest rank and given highest
        weightage.
Step 6: Weight $RT_{kj}$ as $WRT_{kj} = RT_{kj} * W_{kj}$ on the basis of $d_{ik}$ values
        End For
Step 7: For each $d_{ik}$ Calculate $Avgd_{ik}$
        End For
Step 8: If ($Avgd_{ik} \leq D$) Then        // D is the deviation threshold
Step 9: Return trustworthy Recommender
        End For
```

6.5 Experimental setup

The experiment is conducted in a MANET environment to test the performance of the proposed filtering algorithms for alleviating the impact of dishonest recommendations. In this experiment false rating data is propagated to simulate the ballot stuffing and bad mouthing problem. The aim of the experiment is the proper selection of recommenders for calculating the trustworthiness of a certain node in the presence of attacks. The simulation is performed in NS2 simulator with adding the required module to the simulator. We have used 60 random nodes in an area of 500 m × 500 m area. For simulating bad mouthing and ballot stuffing attack several nodes are used to send false rating. Different scenarios are taken into considerations using different number of malicious nodes. We have taken a maximum of 50% misbehaving nodes. We have chosen the threshold value at 0.5 for considering the nodes as trusted. The Table 6.1 depicts the parameters used in the simulation process. Results are obtained after multiple run of the simulation.

Table 6.1: Network Configuration Parameter used in Simulation

Parameter	**Value**
No. of Nodes	60
Area	500 m X 500 m
Speed	30 m/s
Radio Range	250 m
Movement	Random waypoint model
Routing Protocol	DSR
MAC	802.11
Application	CBR
Packet size	512 B
Simulation time	500 s
Trust threshold	0.5
Publication timer	30 s
Deviation threshold, D	0.3

6.6 Experimental Results

There are several types of attacks which can occur due to dishonest recommendation problem. In this work only bad mouthing and ballot stuffing attacks are considered in proposed model. These two types of attacks are appropriate to show the performance of the model to minimize the dishonest recommendation problem. The expected trust value is computed in the simulation by considering the normal behavior of nodes. The model considers the average of the trust values recommended by other nodes. The simulation is carried out with the filtering algorithm and without the filtering algorithm with 0% to 50% of attacking nodes that is with no attacker node and half of the nodes is attacker. The simulation results are plotted in figure 6.1, 6.2, 6.3 and 6.4.

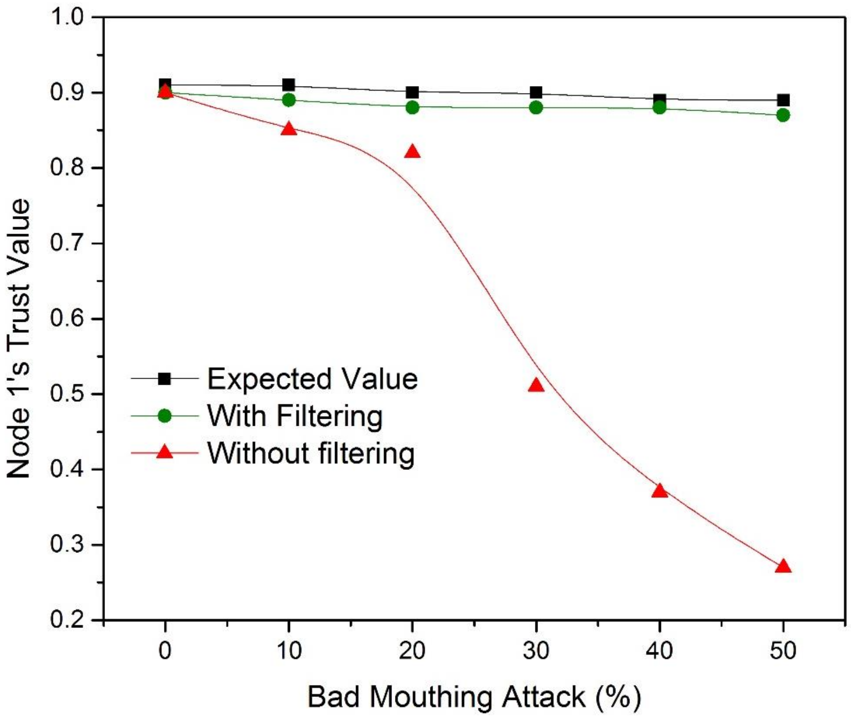

Figure 6.1: Trust value of node 1 in the presence of bad mouthing attack.

Figure 6.1 represents the trust value of a node which is considered as good; in this experiment node 1 is considered as good node. As the number of dishonest recommender nodes increases the trust value of node 1 is distorted when filtering algorithm is not applied because the dishonest recommenders propagate more false ratings. But from the figure it is clear that when filtering algorithm is applied, the trust value is as per expected value even when 50% nodes are considered as bad mouthing attackers.

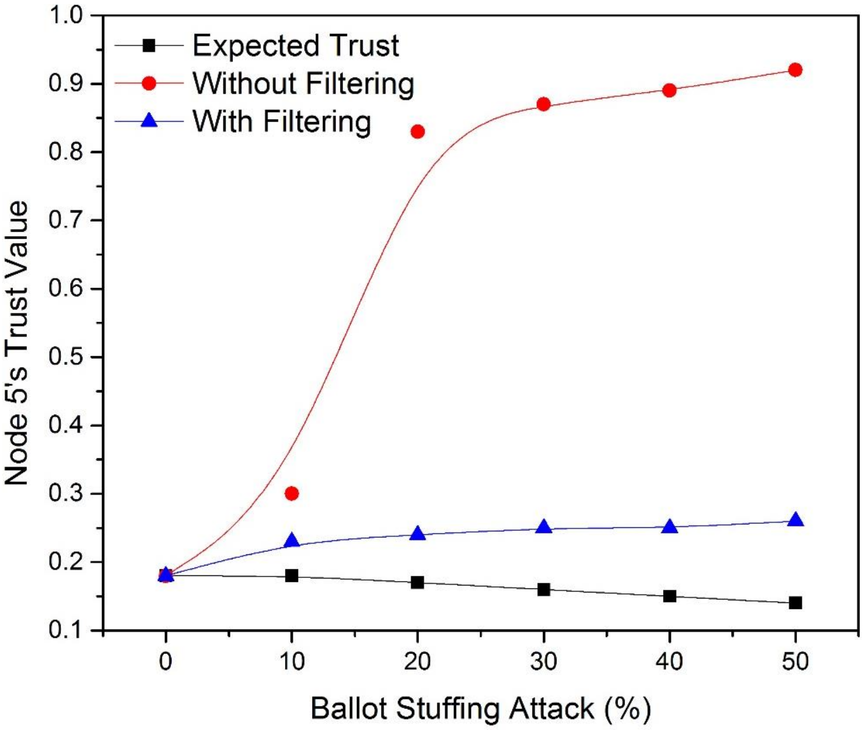

Figure 6.2: Trust value of node 5 in the presence of ballot stuffing attack

Figure 6.2 shows the effect of ballot stuffing attack. The figure shows the computed trust value of a bad node (node 5) when there are no dishonest recommender nodes, which evaluate the expected value; when there are dishonest recommendation nodes but filtering algorithm is disabled and when the filtering algorithm is active. The figure shows that the attacker nodes propagate dishonest recommendation for misleading the node while calculating the trust value. Thus when the attacker nodes percentage is 50 %, it can mislead the judgment by other nodes up to 90%. The proposed filtering algorithm can mitigate the effect of dishonest recommendations to a reasonable extent. Figure 6.3 presents the influence of dishonest recommenders in the present of filtering algorithm and without the filtering algorithm on the basis of performance metrics throughput. Figure shows the throughput of simulated network with and without the defense scheme with the presence of 0% to 50 % dishonest recommender nodes. The proposed defense scheme is able to maintain the throughput performance up to 80 % even there is higher dishonest nodes. From the figure it is clear that dishonest recommender nodes can significantly effect on the network throughput. The proposed mechanism can maintain the throughput level up to a reasonable acceptable level even after there is a high increase in the dishonest nodes.

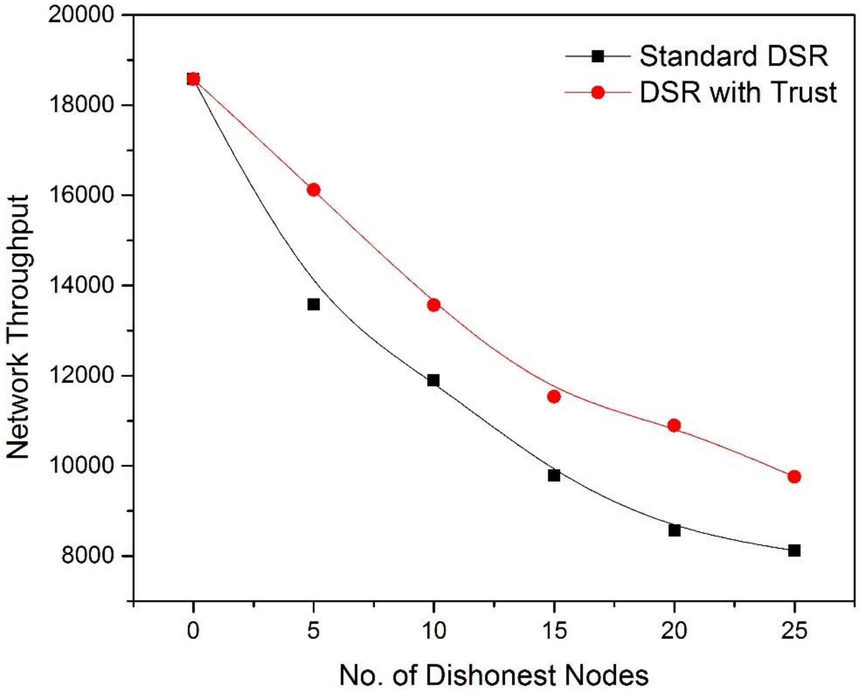

Figure 6.3: Throughput of the network

6.7 Chapter Summary

This chapter presents a recommendation based trust model which uses probabilistic theory for calculating trust value. Here the ballot stuffing and bad mouthing attacks are considered while taking recommendations. This chapter also addresses the dishonest recommender's problem which gives false recommendations about a node which leads to wrong trust value calculation. Here, a filtering algorithm is applied to resist the dishonest nodes from taking part in giving recommendations. A reasonable outcome is observed in resisting the ballot stuffing and bad mouthing attackers in producing dishonest recommendations. In future a more strict mechanism may be designed for proper authentication in taking recommendations so that the no attacker can attack the network.