A Better Approach towards Securing Mobile Adhoc Network

Amit Chauhan
Institute of Engineering & Science,
Indore, INDIA

Prof. Arti Patle
Institute of Engineering & Science,
Indore, INDIA

Prof. Anita Mahajan
Institute of Engineering & Science,
Indore, INDIA

ABSTRACT

A Mobile Adhoc Network is a group of wireless mobile computers in which nodes cooperate by forwarding packets for each other to allow them to communicate beyond direct wireless transmission range. Due to wide-ranging characteristics of the Ad Hoc Networks, it is always at a risk to internal as well as external attacks. Many solutions have been proposed and currently being improved in this area. Most of these solutions involve encryption, secure routing, quality of service etc. Each of them is designed to operate in a particular situation, which may fail to work successfully in other scenarios.

This paper offers an alternate approach to improve the trustworthiness of the neighbourhood nodes and secure the routing procedure. It helps in computing the trust in neighbours and selecting the most secured route from the available ones for communication. It also helps detecting the compromised node and virtually removing from the network.

Keywords

MANET, SID Trust.

1. INTRODUCTION

Mobile adhoc network MANET is a new concept in wireless communication world, where the networks are formed and destroyed on the fly without any centralized controlled. MANET is a collection of independent mobile nodes that can communicate to each other via radio waves. Mobile Ad-Hoc network is a system of wireless mobile nodes that dynamically self-organizes itself in arbitrary and temporary network topologies [2]. Each intermediate node acts as a router and is responsible for forwarding the packets and monitoring the network. Due to the lack of centralize management; security is a major concern in this dynamic, error prone, multi-hop wireless communication network. [1]. An ad hoc network a collections of mobile nodes without any predefined infrastructure. The network is very dynamic, here a node may enter and leave the network on frequent basis. Nodes may also be mobile, that they move within the network itself or from one ad hoc network to the other[5].

"*Trust, is a particular level of the subjective probability with which an agent will perform a particular action, both before we can monitor such action (or independently of his capacity of ever to be able to monitor it) and in a context in which it affects our own action*" [7]. Trust is a belief that the principal, when asked to perform an action, will act according to a predefined description, this implies that the principal will not attempt to harm the requester, regardless of how it carries out the request.

We can make three points of the definition above.

- Trust is subjective.
- Trust is affected by actions that cannot be monitored.
- The level of trust depends on our own actions.

Existing security trends provide the criteria to build certain level of trust in the network. For example, cryptographic algorithms for privacy and digital signatures, authentication protocols for providing authenticity and access control methods for managing authorization. However, these methods do not manage the general concept of "trustworthiness". For instance, a cryptographic algorithm is unable to say that, competent programmers have authored a piece of digitally signed code or a signed public-key certificate does not guarantee the owner's authenticity.

Trust has the following properties. [6]

Transitivity: Trust is not necessarily transitive, that is, if A trusts B and B trusts C, and A does not necessarily trust C.

Symmetry: Trust need not be symmetric, that is, A trusts B does not imply that B trusts A.

Reflexivity: Trust is assumed to be reflexive, that is, a node trusts itself completely.

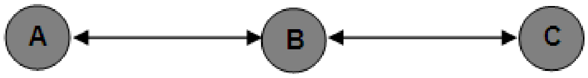

Figure 1. Trust between Network Nodes

A trust level is requested by a "Requester" to the "Recommender", in reply a recommender send its own trust level in the requested node. Based on experience gained via hearing the channel and trust level received from the neighbor, a node calculates its own trust in a particular node for a specific entity. Ad hoc networks are based on *"trust your neighbor"* relationships. This relationship originates, develop and expire on the fly [4]. A trust model can secure an ad hoc network from the attacks to some extent and identify the routes with certain measure of safe and confidence.

2. TRUST RELATIONSHIP

Trust relationship exists between one-hop neighbors. When one neighbor holds a belief about other, the same belief in the reverse direction need not exist at the same time. Mutual trust does exist between entities, but we represent them as two separate trust relationships, which can be manipulated independently.

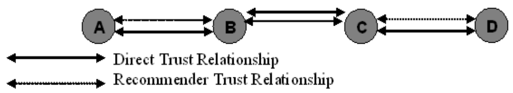

Figure 2. Trust Relationship in the Network

Trust can be seen in two ways, viz. 1. Direct Trust Relationship and 2. Recommender Trust Relationship. Each node maintains a database for its own use. Based on experience and recommendations, these values are changed in the database at any time. Depending on the behaviour of the direct (one-hop) neighbors, a node will calculate the trust value and will recommend the same in case of requested by any other node. For calculating the trust value of the remote node, a requester can demand recommendations from its neighbors.

3. EXISTING SOLUTION FOR TRUST CALCULATION

Trust A wide range of proposals are recommended to estimate the amount of trust between two communicating nodes in ad hoc network. Almost every method is based on situational trust for particular category of activity. Such type of design helps in assessing target node's various activities.

As suggested by Alfarez Abdul-Rahman and Stephen Halles in [3], a requester issues recommendation request message (RRQ) and receives recommendation message. These recommendations are time bound and refreshed on periodic basis. The recommended solution works as follows.

RRQ : : = Requestor_ID, Request_ID, Target_ID, Categories, RequestorPKC, GetPKC, Expiry
Categories : : = SET OF (Category_Name)
Recommendation Request (RRQ)

Recommendation : : = Requestor_ID, Request_ID, Rec_Path, [SEQUENCE OF {Recommendation_Set, TargetPKC} | NULL]
Rec_Path : : = SEQUENCE OF {Recommender_ID}
Recommendation_Set : : = SET OF Recommendation Slip
Recommendation_Slip : : = SET OF SEQUENCE {Target_ID, Category_Name, Trust_Value, Expiry}
Trust Recommendation

Requestor_ID - represents identity of the requester
Request_ID - is a unique identity of the request
Target_ID - represents identity of the target (about whom trust recommendation request is broadcasted)
Categories - set of category names that requestor is interested in inquiring about.
RequestorPKC – is a public key certificate, which can be used to encrypt the Recommendation_set (Optional)
GetPKC – requestor interested in target's public key for further communication (optional)
Rec_Path – contains the ordered sequence of recommender IDs.
Recommendatin_Set – includes multiple instances of Recommendation_Slip
Category_Name – Name of the category for which trust level is requested.
Expiry – Contains expiry period for RRQ

4. SHORTCOMINGS OF THE EXISTING SOLUTIONS

There are several problems in the existing solution. Some of them are,

1. The existing solution [3], computes the trust for a node in particular category. On the contrary, the proposed solution calculates the global trust. It considers the overall behaviour of the target node rather than looking at the certain types of activities.
2. Expiry timers are maintained for the recommendations. If the timer expires and the path is still active, again the original requester has to request for the trust value of the target. Thus, the process is duplicated even in case of unchanged trust value. This incurs more delays and waste of processing time and bandwidth.

The recommender is simply passing on its trust value of target node to the requester and the original requestor computes the value on its own. There are chances of malicious recommendation from one of the recommender, lies in between the original requester and the target node.

5. PROPOSED METHOD TRUST CALCULATION

Many solutions have been proposed to compute the trust level in ad hoc networks. Every solution has its own pros and cons and also designed and developed by keeping particular situation in mind. Thus, it may or may not work in the other condition.

Ad hoc networks are based on *"trust your neighbor"* relationships. Since there is no centralize control, each node is responsible for a secure data communication and as a process of providing secure communication path; each node monitors its neighbors. However, each node has to assure that, it is communicating with a trustworthy neighbor. This proposal offers a unique way of computing the trust level in the network and reduces the communication overhead by limiting the size of packet containing trust level information.

There are two different strategies in calculating target nodes trust value [3].

1. **Direct Trust Value:** This is relevant to the direct trust relationships, where a mobile node in a range can scrutinize the activities of its neighbors and calculate the trust value on its own.
2. **Recommender Trust Value:** This is relevant to recommender trust relationship. In this case, trust value of

out of range node is requested using RRQ (Recommendation Request).

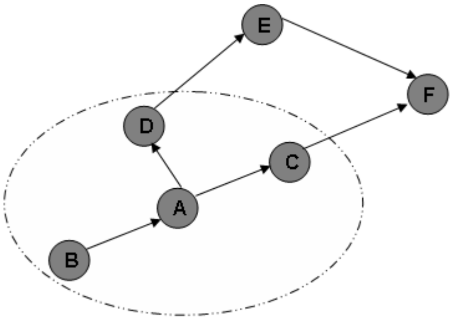

Figure 3. Communication Range

In figure 3, node A can directly communicate with the node B, C and D, as all of them are in A's communication range. Node E and F are out of range of node A, but it can still reach these out of range nodes via D and C respectively. Thus, node C and D will relay the message for A, in case A wishes to communicate with node E and F.

5.1 Neighbour Monitoring

In wireless manner, the nodes in a direct communication range, can transfer the data packet and updates routing paths directly. But before this, a node has to assure that it is communicating with a legitimate node. At this point, the concept of trust in the network appears. Each node has a different trust level in its neighbors and as stated earlier, this trust level is non-transitive, where A trusts B and B trusts C, but it does not imply that A trusts C, because A might have different trust level recommendations from its other neighbors about C. Similarly, trust is also non-symmetric; where A trusts B, that does not mean B also trust A or B may have different trust level value for A.

A node in a radio range of its neighbors can passively overhear the channel and ongoing activity at the other end. This is possible even if a node is not actively involved in a communication. Because of this unique characteristic of wireless networks, it is viable in ad hoc networks to monitor the neighborhood activities and record any offences conducted. Each node constantly monitors the activity of its neighbors in terms of amount of successful data packets and routing updates forwarded correctly. In case of any malicious activity, a node will broadcast SID (Single Intrusion Detection) against the malicious node. The affecting nodes (which are in a radio range of a malicious and SID originator node) will recompute the trust level of malicious nodes and raise their trust level database.

5.2 Validating Single Intrusion Detection

A malicious activity by any node can be detected and other nodes are informed using Single Intrusion Detection (SID). It is quite likely that a malicious node may broadcast a false SID against a legitimate node or due to the unavoidable circumstances like poor radio connectivity, error in received packets, etc. a node may get detected as a compromised node by its neighbors and an SID may broadcasted against it. Thus, instead of blindly accepting the SID, following parameters are considered by a node receiving SID broadcast

5.3 Algorithm to validate SID

1. Trust level of a node, which is broadcasting SID against a compromised node.
2. If a compromised node is in a radio range, it will observe a compromised node for a certain period.
3. It will request other neighbors for their recommendations about the compromised node.
4. Depending on its conclusion, a node may recompute the trust level for either the compromised node or SID broadcasting node.

5.4 Trust Recommendation

Structure of the recommendation request (RRQ) message is almost same as the one used in [3].

RRQ : : = Requestor_ID, Request_ID, Target_ID

Recommendation Request

In the above message, the RRQ does not contain Category and Expiry. We have not included these two parameters because we are not judging the trust of any node for a particular category. We are calculating the global trust, based on SID, hello beacons and acknowledgements. We have also not included the expiry time – during the ongoing communication, if the trust value of any active node is changed, the recommender will re-recommend the changed trust value to the requestor. This feature is based on the characteristic discussed earlier in which, a RRQ broadcasting node keeps track of the requester from whom the RRQ is received earlier.

Recommendation::= Requestor_ID, Request_ID, Recommender_ID, Target_ID, Trust_Value

Request_ID : represents identity of the request

Target_ID: represents identity of the target

In this proposal, each node computes its own trust level and forward it to the requestor, this process continues until the original requestor computes the trust level for the targeted node.

5.5 Trust Computation

As discussed earlier, each node computes its own trust based on its observation or recommendations from its neighbors. Unlike the scheme proposed in [3], where an original requester node computes the trust level recommended by intermediate as well as final recommender, here every node computes the trust level based on its own trust in its neighbors and forwards the computed trust towards the original requester.

A node constantly observes the activities of the other nodes in its radio range and computes the trust level for each node. In case of SID broadcast, the compromised node is not evicted out of the network immediately, rather trust level is computed and if it falls below certain threshold then only the node is expelled from the network.

A node will compute the trust level of its neighbors based on SID (either broadcasted by itself or other nodes), beacons and acknowledgements (during ongoing communication with a particular node). Trust level computation also depends on received / missing beacons and acknowledgements.

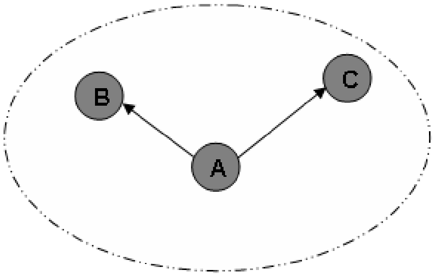

Figure 4. Network Node Computing Trust Level of its Neighbors

In fig 4, node B and C are in the radio range of node A. Each parameter viz. SID, Beacon and Acknowledgement is rated on the scale of 0 to 5 where [3]

Table 1. Trust Value Semantics

Value	Meaning	Description
0	Distrust	Completely Untrustworthy.
1	Ignorance	Cannot make trust-related judgment about entity.
2	Minimal	Lowest possible trust.
3	Average	Mean trustworthiness. Most entities have this trust level
4	Good	More trustworthy than most entities
5	Complete	Completely trustworthy

It computes the trust level as follows. Consider that node A is computing the trust level of node B. A node computing trust level, rates each of these events from 0 (zero) to 5 (five) based on its experience. This scheme proposed percentage based computation. It takes 60 % of SID, 20% Hello Beacons and 20% of Acknowledgements. In case of no data transfer there will not be any communication between two nodes and hence no acknowledgements, in this case it takes 60% of SID and 40% of Acknowledgements.

tv = 0.6 * sid + 0.2 * bcn + 0.2 * ack ……..… (1)

Here,

tv = Trust Value

sid = Single Intrusion Detection

bcn = Beacon

ack = Acknowledgement

Lets assume that node A has gathered following details about node B after observing it for a particular period of time.

sid = 4, bcn = 3, ack = 2

tv = 0.6 * 4 + 0.2 * 3 + 0.2 * 2

= **3.4**

Each node will use the above-described formula to compute its trust level in the neighboring node. Even if it is recommending the trust level of its neighbors to the requestor, it will first calculate the trust level in above described manner and forward it to the recommendation requester node.

If a node (either intermediate or original RRQ generator) is seeking the trust level for a node, which is not in its radio range, it will broadcast RRQ to its neighbors.

In case of recommending the trust value of the target node, the recommender will calculate the trust using the equation (1) and forward it to the requester node. Upon receiving the recommendations from the neighbors, original requester will calculate the final trust value for the target node as follows.

Computing Trust Proportion of the Neighbor

$$S_r = \sum_{i=1}^{n} Ni \text{........................ (2)}$$

where,

S_r = Sum of trust of neighbors (recommenders) of a requester

i = Neighbors of a requester

N = Trust value of i

Computing Trust Proportion Between a Requester and a Recommender

$T_{Rr(i)} = [100 * T_{r(i)}] / S_r$ ....(3)

Where,

$T_{Rr(i)}$ = Calculated trust proportion between a requester and a recommender

$T_{r(i)}$ = Trust recommended by a recommender

S_r = Sum of trust of neighbors (recommenders) of a requester

Computing the Final Trust Value of a Target

$$T_t = \sum_{i=1}^{n} T_{Rr(i)} \text{ …………..... (4)}$$

Where,

T_t = Sum of trust proportion of neighbor nodes.

$T_{Rr(i)}$ = Calculated trust proportion between a requester and a recommender

The original RRQ requester node will calculate the trust level of a target node in above described manner. Firstly, it will calculate the total trust value of its neighbor and compute the individual trust proportion of each neighboring node based on it. At last, it will calculate the trust value for target node by adding the recommended trust values based on the proportional trust in the recommender.

5.6 Algorithm to compute Trust in Adhoc Network

1. Check whether the target node is in the communication range or not. If it is not in a communication range then, broadcast RRQ in the vicinity.
2. Compute the trust percentage of each node against the sum of the trust of all the neighbors.
3. Consider the percentage of recommended trust value based previously computed trust level proportion of the neighbors.

Add the calculated recommendations from the neighbors and compute the final trust value of the target node.

Example:

In this scenario as shown in figure 5, node A wishes to communicate with node E. As a prerequisite for a secure communication, node A requests for a trust level for node E. Since E is not in a communication range of the node A, it will broadcast the RRQ message to its neighbors.

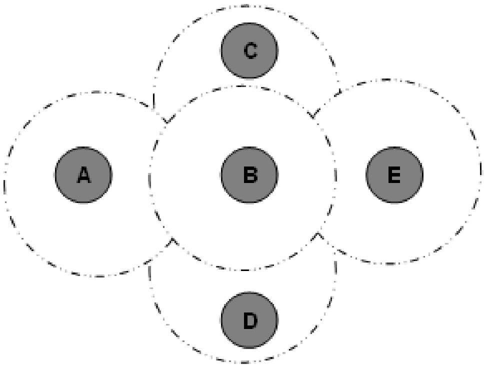

Figure 5. Requesting Trust Level of Out of Range Node

Node B, C and D can directly communicate with node A

Node A, C, D and E can directly communicate with node B

Node A, B and E can directly communicate with node C

Node A, B and E can directly communicate with node D

Node B, C and D can directly communicate with node E

As stated above, as a part of secure communication with node E, node A broadcasts RRQ to request. Here, node A and E is not in a direct communication link of each other. Neighbors of node A viz. node B, C and D, will receive the RRQ packets and fortunately, all of these nodes are in a direct radio communication rage of node E. All of these nodes have calculated the trust level of node E previously, using the equation (1).

A → B : A,rrqA01,E

A → C : A,rrqA01,E

A → D :A,rrqA01,E

As a recommendation reply each node will send its own recommendation to node A – the requestor.

B → A : A,rrqA01,B,E,2

C → A : A,rrqA01,C,E,3

D → A : A,rrqA01,D,E,2

Node A will calculate the trust level as follows.

Initially,

- Node A trust node B value 1
- Node A trust node C value 3.5
- Node A trust node D value 5

Recommendations about node E from the neighbors of node A.

- Node B trust node E value 2
- Node C trust node E value 3
- Node D trust node E value 2

As per Equation (2)

Sr = 9.5

As per Equation (3)

$\mathbf{T_{Rr(B)} = 0.21}$

$\mathbf{T_{Rr(C)} = 1.11}$

$\mathbf{T_{Rr(D)} = 1.05}$

Finally, Trust of Node A in a Target Node E according to the equation (4)

$\boxed{\mathbf{T_t = 2.37}}$

6. BENEFITS OF THE PROPOSED SCHEME

The proposed scheme has significant variations and their benefits as follows.

1. The RRQ and recommendation reply messages are simple and do not contain unnecessary parameters and hence it reduces the overhead generated in the network.
2. Each node itself computes the proportional trust level of its neighbors and forwards it to the requester. Thus, there is no need of sending the list of intermediate recommenders in form of rec_path or rec_slip.
3. We do not calculate the trust for any specific category. We are calculating the global trust, based on SID, hello beacons and acknowledgements.
4. As stated earlier, in case of any change in the trust value of any active node, the recommender will re-recommend the changed trust value to the requestor, with the help of the

forward path established previously. Hence, we do not maintain any Expiry Timer.

5. The original requester node calculates the trust level of target node in proportion with the trust level of his neighborhood territory.

7. LIMITATIONS OF THE PROPOSED SCHEME

There are serveral shortcomings of the proposed scheme as follows:

1. Since the proposed protocol is designed to prevent DoS attacks, it does not calculate the trust level in any perticular category, hence it may not help in selecting a route demanding a quality of service (higher bandwidth etc.)
2. Each node computes its own trust level rather than simply forwding it to the requester, it may take more processing power and time to generate and forward its recommendation.
3. I have ignored the memory requrements for storing reputations and behaviour of the recommendation protocol (since it is not being implemented and tested in any kind of live environment).

8. CONCLUSION

Security is vital in Ad Hoc Networks. Securing the Ad Hoc Networks starts from the neighbor verification in the local community also termed as a cluster – collection of wireless nodes in a particular group. As a proposed solution trust is not calculated for any particular situation instead, it is computed based on a summary of behavior of the node for a specific amount of period, instead of a target node, calculation is made on overall trust, a neighbor itself will calculate the percentage based trust and recommend it to the requester. In case of any malicious behavior, a Single Intrusion Detection (SID) packet is broadcasted against compromised node and all the participating neighbors are informed about the malevolent activity performed.

9. REFERENCES

[1] Hao Yang, Xiaoqiao Meng, Songwu Lu, “Self-organized network-layer security in mobile ad hoc networks”. In Proc3rd ACM workshop on Wireless Security.

[2] Kortuem G., Schneider J., Preuitt D., Thompson T.G.C., F’ickas. S., Segall.Z, “Whwn Peer-to-Peer comes Face-toFace: Collaborative Peer-to-Peer Computing in Mobile Adhoc Networks”, 1st International Copnference on Peer-to-Peer Computing, August, Linkoping, Swedan,2001. Pp. 75-91.

[3] Alfarez Abdul-Rahman & Stephen Halles, “A Distributed Trust Model”, New Security Paradigms Workshop**,** Proceedings of the 1997 workshop on New security paradigms. pp. 48-60.

[4] Asad Amir Pirzada and Chris McDonald, “Establishing Trust in Pure Ad-hoc Networks”. ACM International Conference Proceeding Series; Vol. 56, Proceedings of the 27th conference on Australasian Computer Science- Volume 26. pp 47-54.

[5] C. Perkins and E. Royer. Ad hoc on-demand distance vector routing. In Proc. IEEE Workshop on Mobile Computing Systems and Applications, 1999.

[6] Catharina Candolin and Hannu H. Kari, “Distributing Incomplete Trust in Wireless Ad Hoc Networks”. In Proceedings of the New Security Paradigms Workshop, ACM, 1997.

[7] Gambetta Diego (2000) ‘Can We Trust?’, in Gambetta, Diego (ed.) “Trust Making and Breaking Cooperative Relations” electronic edition, Department of Sociology, University of Oxford, chapter 13, pp. 213-237.