

# Trust Management Scheme for Mobile Ad-Hoc Networks

Raihana Ferdous, Vallipuram Muthukkumarasamy, Abdul Sattar

Institute for Integrated and Intelligent Systems

Griffith University, Australia

Email: raihana.ferdous@student.griffith.edu.au; v.muthu,a.sattar@griffith.edu.au

**Abstract**—The inherent freedom in self-organized mobile ad-hoc networks (MANETs) introduces challenges for trust management; particularly when nodes do not have any prior knowledge of each other. Furthermore in MANETs, the nodes themselves should be responsible for their own security. We propose a novel approach for trust management in MANETs that is based on the nodes own responsibility of building their trust level and node-level trust monitoring. The main contribution of this work is in the introduction of a Node-based Trust Management (NTM) scheme in MANET based on the assumption that individual nodes are themselves responsible for their own trust level. We explore and develop the mathematical framework of trust in NTM. We have defined a proposed metrics for nodes to establish and manage trust. We also have presented some new algorithms for trust formation in MANETs based on experience characteristics offered by nodes.

## I. INTRODUCTION

Mobile ad-hoc networks (MANETs) are dynamically configured, multi-hop wireless networks with varying topology. Mobile nodes may be considered as some programs launched by network users to accomplish certain tasks while migrating from one computation environment to another. They can be widely employed in many fields like electronic commerce, secure brokering, distributed information retrieval, and telecommunication network services. Due to the mobility of MANETs, most nodes are supposed to be as small as possible to be carried out or to be easy to install in hostile places. However, it is also true that a node may easily be stolen and become compromised. Thus, the trust between nodes in ad-hoc networks can not be guaranteed. Furthermore, this problem may increase the chance to tamper the stolen node. It is also vulnerable since every node in MANET uses radio wave to communicate. It is very hard to detect any node since there is no explicit evidence. In order to enforce cooperation within the networks, adjacent nodes should build up trust over time. Such trust establishment procedure can improve security, connectivity, and quality of service in the network so that performance is improved.

Self organization of ad-hoc network nodes into clusters has been studied in the literature [19] to induce distributed control in such networks, which are otherwise infrastructure-less. Present ad-hoc network clustering schemes use physical location as a metric to cluster the nodes[14]. Any node can elect to become a cluster head and can propagate cluster joining requests to its  $k$ -hop neighbors through various flooding

mechanisms. This choice of cluster formation is arbitrary and does not take security into account. A node that is malicious could initiate a cluster formation announcement and could potentially compromise all nodes that elect to join its cluster. The main problems that we address in this paper are the following: (a) to define a metric that the nodes can use to make decisions on whether to establish keys with other nodes in an ad-hoc network, given that the infrastructure for establishing such keys exists, and (b) to define a trust management scheme in which nodes in a mobile ad-hoc network (MANET) can securely group together in order to trust each other. We also propose a node-based trust management system architecture and the relevant algorithms to analyse the phases of this scheme. The proposed management scheme is based on a clustered wireless mobile sensor network with backbone and on a mobile agent system; it introduces a trust of a node within local management strategy with help from the mobile agents running on each node. That is, a node's trust-based information is stored as a history on the node itself and managed by the local mobile agent of the node. This paper is organized as follows:

Section II depicts some related work in introducing trust and security in MANETs. Section III describes the work on the theory of trust formalization. Section IV illustrates our proposed node-based trust management (NTM) scheme. Section V depicts the analytical part of NTM with the system architecture and some algorithms. Finally section VI concludes the paper.

## II. RELATED WORK

The idea of using trust to mitigate security threats has been an important area of research [26]. Trust establishment and management between entities (nodes or agents) can be done through a central trusted authority or in a distributed fashion by nodes [1], or a combination of both. Related work in this area [4], [27], [17], employ all these approaches. For example, Zhou et al. [13] proposed the idea of utilizing threshold cryptography to distribute trust in ad-hoc networks, Davis [3] proposes the use of certificates based on hierarchical trust model to manage trust, and Eschenauer et al. [12] contrast between trust establishment in ad-hoc networks and the Internet. Our approach is new in that we use trust as a metric to address the problems. Some research works have shown that rating nodes' trust level is an effective approach in distributed

environments to improve security [2][18], to support decision-making[24] and to promote node collaboration [22].

Existing trust management schemes like [15],[16], [21] were designed originally for wired networks (e.g., P2P), and are thus not suitable for wireless sensor networks, where nodes have resource constraints. The schemes in [23], [9] are developed for wireless networks, but they focus mostly on trust and reputation modeling and seldom emphasize on the network performance issues. The most commonly used approaches by these schemes for acquiring reputation can be classified into two groups; on-demand and periodical. As trust information is distributed into the entire network, reputation computation requires network-wide flooding for trust aggregation. Whether reputation computation is done periodically or on-demand, it will incur bandwidth usage and energy consumption. Although the scheme presented in [23] restricts flooding to a subset of nodes, how to determine the subset such that it covers all the nodes (or a sufficient number of nodes) holding the required trust information becomes a problem.

Although many security mechanisms[28],[7] and fault-tolerance models [10], [8] have been proposed and developed for mobile nodes systems, there are still a number of issues that need to be addressed before having a robust mobile node system. A common solution provided by the researchers is as follows: when a node initiating route discovery determines the required minimal trust level for nodes participating in the query and reply propagation. Since only nodes at each trust level share symmetric encryption keys, intermediate nodes of different trust levels cannot decrypt in-transit routing packets or determine whether the required security attributes can be satisfied. Only the nodes with the correct key can read the header and forward the packet. So if a packet has reached the destination, then it must have been propagated by nodes at the same trust level. Therefore Routes discovered by trust aware routing comes with “quality of protection” guarantees. Moreover, mobile nodes are featured with autonomy, asynchrony, adaptivity and communicability. After a mobile node runs on a computer, it accesses the local resources and therefore the computation logic is encapsulated inside the mobile nodes, the transfer of control and data messages is minimized, and thus the possibility of network congestion decreases. Using mobile nodes to access distributed data creates an efficient method for data distribution, aggregation, and sharing in distributed network environments with bandwidth constraints. The new metric of trust represented in this paper differs from the previous trust models, as described in related works, in the sense that it considers a copy of each node within the environment for the trust monitoring aspect. It should be also noted that only two end nodes (source and destination) collect evidences and update their opinion on the trustworthiness of the communication path. In this context we demonstrate with new notations and algorithms to analysis the model. Therefore, firstly, we need to determine how the trust notion can be defined amongst mobile nodes. The following section depicts our work [20]related to the trust formalization and how this formalized notion helps to build our NTM scheme

in MANETs.

### III. THEORY OF TRUST FORMALIZATION

This section mainly describes the trust formalization of our previous work[20] so that the analysis of our proposed Node-based Trust Management (NTM) can be developed. In our NTM scheme, we need to compute TEs (Trust Evaluators) by grasping the TRUST-VALUE from equation 2. Our schemes draw ideas from the Watchdog and Pathrater schemes [25], utilized for cooperation of nodes in ad-hoc networks. We define a node  $n_i$ 's trust on another node  $n_j$ :

$$T_{n_i, n_j} = \alpha_1 n_i T_s^{n_j} + \alpha_2 n_i T^{n_j O} \quad (1)$$

In the above equation,  $T_{n_i, n_j}$  is evaluated as a function of two parameters:

- $n_i T_s^{n_j}$ : Node  $n_i$ 's self evaluated trust on  $n_j$ ;  $n_i$  computes this by directly monitoring  $n_j$ .
- $n_i T^{n_j O}$ : Weighted sum of other nodes' trust on  $n_j$  evaluated by  $n_i$ .

In eq. (1),  $\alpha_1$  and  $\alpha_2$  are weighting factors such that  $\alpha_1 + \alpha_2 = 1$ . Thus, by varying  $\alpha_1$  and  $\alpha_2$ ,  $n_i$  can vary the weight of self evaluated vs. others trust in calculating its total trust on  $n_j$ . Here,  $0 \leq T_{n_i, n_j, n_i} T_s^{n_j, n_i} T^{n_j O} \leq 1$ , and thus eq. (1) is normalized.

Node  $n_i$  computes this value by directly monitoring  $n_j$  when  $n_j$  is in its radio range. We define  $n_i T_s^{n_j}$  as:

$$n_i T_s^{n_j} = f(\Phi, \Omega) \quad (2)$$

Node  $n_i$ 's self trust on  $n_j$  is a function ( $f$ ) of traffic statistic functions  $\Phi$  and  $\Omega$  computed by monitoring  $n_j$ . Precise definition of  $f$  can be implementation dependent. We assume  $f$  to be a weighted sum of  $\Phi$  and  $\Omega$ . Here,  $\Phi$  is a function of monitored traffic statistics pertaining purely to traffic volume and  $\Omega$  is a function of monitored traffic statistics pertaining to information integrity. Lee et al. [29] compile node monitoring statistics for one hop neighbors in ad-hoc networks. Based on these monitored statistics we define  $\Phi$  and  $\Omega$  as:

$$\Phi = g(\gamma_1, \gamma_2, \gamma_3, \gamma_4, \gamma_5, \gamma_6) \quad (3)$$

$$\Omega = h(\lambda_1, \lambda_2) \quad (4)$$

Here  $g$  and  $h$  can again be defined based on the implementation. Like  $f$ , we assume them to be weighted summation of their constituent parameters. These parameters are defined below:

$\gamma_1$ : packets sent by  $n_j$  to  $n_i$  that are dropped by  $n_j$

$\gamma_2$ : total packets dropped by  $n_j$

$\gamma_3$ : packets dropped by  $n_j$  due to congestion

$\gamma_4$ : packets dropped by  $n_j$  due to unknown reasons

$\gamma_5$ :  $n_i$ 's assessment of  $n_j$ 's priority to  $n_j$ 's self packets vs. all other nodes' packets

$\gamma_6$ : packet forwarding delay by  $n_j$

$\lambda_1$ : packets misrouted by  $n_j$

$\lambda_2$ : packets falsely injected by  $n_j$

In the representation  $n_i T^{n_j O}$ ,  $O$  is the set of other nodes,

TABLE I: Path chosen in proposed scheme

Next hop neighbor in the best path P1	T	U	T
Next hop neighbor in the best path P2	U	U	T

whose trust on  $n_j$  is utilized by  $n_i$  in evaluating its own trust on  $n_j$ .  $O$  is defined as:

$O = \forall \text{ node } o \in O \Rightarrow o \text{ is in the range of both } n_j \text{ and } n_i,$   
and  $\exists T_{no}, s.t. T_{no} \geq \text{“good”}$ .

Here “good” is a threshold value for demarcating Unknown and Good trust-regions.

In our Trust Model we adopt the concept of DSR[5] which is a routing protocol that is designed for use in a multi-hop environment like wireless mobile ad hoc networks. It allows mobile nodes to organize and configure themselves to form connections between them without any aid of an existing infrastructure or administration. DSR routing protocol reacts to the change of topology of the mobile ad hoc network caused by mobility of mobile nodes in the network or by interferences on the wireless communication links[6].

To communicate in a network, all mobile nodes must have a unique identifier, usually the IP address. However, in MANET the topology changes dynamically, thus creating difficulties for centralized administration to distribute IP addresses or any other identifier. This situation leads to a need for distributed, dynamic and automatic service. Therefore, towards establishing trust in MANETs, an integrated approach for auto-configuration, authentication and certification is needed. Auto-configuration provides a service that renders MANET more efficient and robust. Even though there are many approaches related to auto-configuration, none has been standardized for trusted network.

If the source node wants to communicate to a destination node to which it does not know the route to, the source node has to use the route discovery techniques to find the route to the intended destination. As it mentioned earlier that any node wishes to send messages to a distant node, its sends the ROUTE REQUEST(RTREQ) to all the neighboring nodes. The ROUTE REPLY(RTREP) obtained from its neighbor is sorted by trust ratings. The source selects the most trusted path. If its one hop neighbor node is a friend, then that path is chosen for message transfer. If its one-hop neighbor node is an acquaintance and if the one hop neighbor of the second best path is a Trustworthy then  $T$  is chosen otherwise it is  $U$  as depicted on the table 1. Similarly an optimal path is chosen based on the degree of trustworthiness existing between the neighbor nodes. The source selects the shortest and the next shortest path. Whenever a neighboring node is rated as trustworthy, the message transfer is done immediately. This eliminates the overhead of invoking the trust estimator between trusted partners. If it is an acquaintance or stranger, transfer is done based on the ratings. This protocol will converge to the DSR protocol if all the nodes in the ad-hoc network are being rated as trustworthy,  $T$ .

TABLE II: Proposed Notations for NTM

NAME	DESCRIPTION
NTM	Node-based Trust Management
$n_i$	A node in the network
$CNTXT_i$	A given context
$ID(n_i)$	ID of node $n_i$
SK	A symmetric secret key held by TM
COUNTR	Message counter
HB	Interaction History stored in a Buffer
TETBL	Trust evaluation Table
TEs	Trust evaluators
INFO	Trust Information

#### IV. PROPOSED NODE-BASED TRUST MANAGEMENT(NTM) SCHEME AND ITS SYSTEM ARCHITECTURE

As trust information is usually requested by nodes before they start communicating with each other, trust management schemes with poor trust and reputation acquisition latency are not acceptable in situations with strict real-time requirements, like battlefields and emergency rescue services. A backbone network construction algorithm, RCC, and its performance evaluation is presented in [11]. Under these circumstances, in this section, we propose a novel node-based trust management scheme (NTM) for MANETs. The main objective of the NTM is to effectively manage trust and reputation with minimal overhead in terms of extra messages and time delay. The notations to be used for describing the NTM can be found in Table II.

##### A. An Overview of the node

The NTM is based on a clustered MANET with backbone, and its core is a mobile node system. Differing from traditional trust and reputation management systems, NTM requires that a node’s trust information to be stored in the forms of Trust evaluators (TEs) by the node itself. Obviously, nodes cannot manage and compute their own trust and reputation. So, NTM further requires that every node locally hold a mobile node that is in charge of administrating the trust and reputation of its hosting node. In this sense, mobile nodes provide nodes a “one-to-one” trust and reputation management service.

In NTM, an arbitrary transaction is defined as the process of interaction between two nodes, the requester and the provider, and it is triggered by the requester and may be accepted/rejected by the provider. Before starting any transaction, the requester asks its local mobile node to obtain the TE of the provider by directly querying the provider’s local mobile node. Based on the provider’s TE, the requester decides whether or not to start the transaction. After a transaction is finished, the requester makes a trust evaluation on the provider based on the quality of service it gets from the provider during the transaction, and then submits the evaluation to its local mobile node which then accordingly generates a TE for the provider and sends the TE to the provider’s local mobile node. Based on the collected TEs, a mobile node periodically issues its hosting node updated TE.

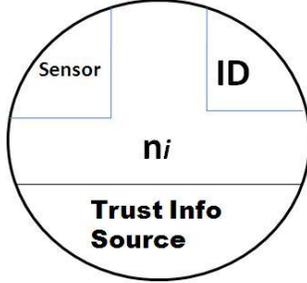


Fig. 1: A node in NTM

### B. NTM Network Model

In this paper, we use a network model based on sufficient backbone nodes with multiple long-range radios that are randomly scattered in the MANET. Every node has a unique ID by which it can be distinguished from others. Through a secure routing protocol, all the nodes (including backbone nodes) together compose a low-level network using a short-range radio, which is referred to as the original network. Likewise, the backbone nodes additionally constitute a high-level network, the backbone network, via a long-range radio. The original network is dynamically partitioned into a number of clusters by an effective clustering algorithm running on each node; each cluster has a backbone node elected as cluster head. Both backbone and non-backbone nodes may fail or become unavailable due to a system crash, power exhaust, or any other reason; however, the rest of the original network and backbone network should still be connected.

Since mobile nodes are designed to travel over the entire network and run on remote nodes, they must be launched by trusted entities. And clearly, compromised mobile nodes may not provide the expected services and can actually constitute a threat to network security. Therefore, in NTM, we assume (1) that there is a trusted authority that is responsible for generating and launching mobile nodes, and (2) that mobile nodes are resilient against the unauthorized analysis and modification of their computation logic. The architecture of NTM consists of three key segments: Node Initiators (NIs), Trust Monitors(TMs),and Trust Evaluators(TEs). Each node of NTM consists of four components: wireless sensor, ID of the node, Trust Info-score and Context (Fig-1).

1) *Node Initiator(NI)*: An NI is an authority responsible for generating and launching TMs into the network. It may be a piece of software, a node, or an organization, and it could be either inside or outside the network. In NTM, we assume that there is only one NI that is always-existing and trusted by every node. The NI launches one TM each time in a broadcast fashion into the backbone network. Before launching a TM, the NI associates it with a symmetric secret key, SK, and a monotonically increased version number (starting from 1). The purpose of SK is to secure trust aggregation and reputation propagation, while the version number is used to support agent consistency verification. In case the SK of the current TM is

TABLE III: A Table of TETBL of node  $n_i$

ID	CNTXT	EVAL	TSTMP	COUNTR
$ID(n_i)$	Context	$t_{ij}$	T	1

stolen or broken, the NI may periodically launch a new TM with a higher version number and a fresh SK to replace old ones according to application-specific security requirements.

2) *Trust Monitor(TM)*: A TM is a mobile agent generated by the NI. It is designed to be distributed into every node and to provide its hosting node with a trust and reputation management service. Each node will hold a copy of the TM's current version. For an arbitrary node  $n_i$ , its copy TM,  $TM(n_i)$ , locally maintains three data structures, i.e. a trust evaluation table TETBL, an interaction history buffer HB and a message counter COUNTR. The trust evaluations that  $n_i$  recently made on other nodes are kept in TETBL, while the TEs issued to  $n_i$  by the local copy TMs of other nodes are also stored in TETBL. HB accommodates  $n_i$ 's TE last issued by  $TM(n_i)$ . As for COUNTR, it is incremented whenever  $TM(n_i)$  receives a message from a node for the first time since the last COUNTR resetting.

As illustrated in table, TETBL is composed of five fields, ID, CNTXT, EVAL, TSTMP, COUNTR among which ID and CNTXT together constitute the primary key of the table. Field ID contains the IDs of the evaluated nodes; field CNTXT implies trust contexts; and field EVAL stores the trust evaluation values; field TSTMP holds the time when evaluations are made. For any node  $n_i$ , field CNTXT implies trust contexts. Field TSTMP holds the time when TEs are issued, while field COUNTR reflects how many times a TE is acknowledged, and its default value is 1. A copy TM stays on its host until it is replaced by the copy of a higher-version TM, and in the meantime it offers its host the trust and reputation management service. When TM replacement takes place, the new local TM will take over all the data structures maintained by the old one and reset COUNTR to 0.

3) *Trust Evaluators(TEs)*: A Trust Evaluator is a segment of data that is organized with a special structure and issued by the copy TM of a node (*sender*) to another node (*receiver*). It is stored in the TETBL on its receiver node. Considering any two nodes  $n_i$  and  $n_j$ , the TE issued by  $TM(n_i)$  to  $n_j$  under context, **Context** is defined as:

$$TI(n_i, n_j, CNTXT) = EVAL_{SK}(D) \quad (5)$$

where  $D = (ID(n_i), ID(n_j), CNTXT, T, t_{i,j})$  and T is a time-stamp implying the time when the TE is issued. From the above definition, we can see that a TE implicitly indicates the temporal property of trust by the use of a time-stamp T. TE is driven by transactions, and it involves message transmission between the copy of TMs of the sender and receiver.

## V. ANALYSIS OF NTM

The execution of NTM involves two phases; network formatting phase and the Trust Management interaction routine phase. As soon as NTM starts, the network formatting phase

is initiated. The purpose of this phase is to distribute a TM to every node. What follows is the trust management interaction routine phase during which the trust and reputation service is provided. The following subsections depict the details of these two phases.

#### A. Network Formatting

The network formatting phase consists of two stages. In the first stage, the NI launches a TM in the network in a broadcast fashion. Considering an arbitrary node,  $n_i$  in the backbone network, when it receives a TM for the first time,  $n_i$  makes a copy of the TM and then forwards the TM to all its immediate neighbors in the backbone network. If  $n_i$  receives an already-received TM, it just discards the TM. Once  $n_i$  has a copy TM, it enters the second stage. In the second stage,  $n_i$  checks whether it is a cluster head itself. If so,  $n_i$  broadcasts its copy TM within its cluster in order to distribute the copy TM to all its cluster members, otherwise it keeps silent. The network formatting phase is run at the beginning of the execution of the NTM, and it may also be re-run later from time to time, to update the copy TMs depending on application-specific security requirements.

#### B. Trust Management interaction routine

As long as a node has a local copy TM, the trust and reputation service provided by the copy TM is available to the node, and thus we say that the node is in the service-offering phase. The trust and reputation service is composed mainly of two types of sub-services:

*Trust value acquisition* and *Trust management service routine*. The first sub-service is transaction-driven and involves the message transmission between the requester and provider, whereas the other sub-service involves merely the local processing periodically performed by the copy TM of each node. Because of the asynchronous execution, nodes are unlikely to enter the Trust management interaction routine phase at the same time. Specifically, it is possible that some nodes do not yet have a local copy of TM when they are asked by other nodes for TEs. Clearly, the asynchronous execution may lead to the failure of the trust management service. The trust value acquisition service consists of three algorithms to follow illustrated in fig-2, 3, 4 (Algorithm-1, 2, 3 respectively). Algorithm 1 depicts that a node  $n_i$ , in the network, requests TRUST-VALUES from another node  $n_j$ . If  $n_i$  does not get any response from  $n_j$  in a given period of time (TTMSTMP), then  $n_i$  re-sends the request to the same destination ( $n_j$ ). If  $n_j$  does not reply for the second time, then it is ( $n_j$ ) discarded by node  $n_i$ . Otherwise, with a given reply by  $n_j$  within given time period, TM of  $n_i$  then validates the TRUST-VALUE by decrypting it and then retrieve TE by equation 2. When the validity checks runs successfully, then TM of  $n_i$  computes the trust of  $n_j$  based on previously-run trust evaluation function. If the validity checks fails, then the TRUST-VALUE is being treated as 'illegally modified value'. Now, according to Algorithm 2, the transmission delay between each message transaction can be computed by deducting the timestamps between message

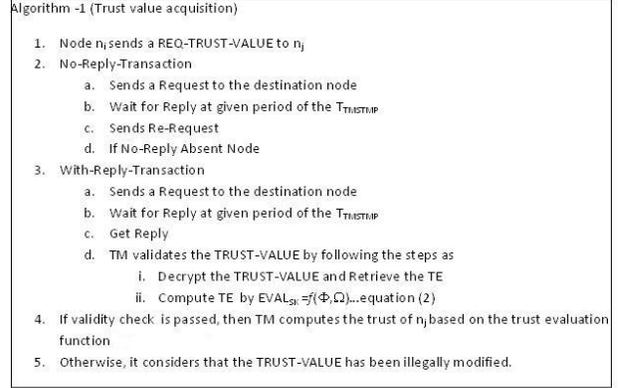


Fig. 2: An Algorithm for Trust value acquisition

transmission and message reception. It takes into account some relative delays and average delays as well. Finally, in algorithm 3, the pseudo-code reveals that for each and every trust evolution, there exists a trusted route to propagate TEs to the trusted nodes in NTM. All verifications and validations are conducted by using the time-stamp to keep track of the history of interactions. At the end of any successful interaction, it (e.g. node  $n_i$ ) sends ACK = 1 to the destination node to terminate the session. In this trust management interaction routine, each intermediate node accommodates incoming TRUST-INFO in proper buffers (HBs). Upon receiving the first INFO, the node will calculate the appropriate time duration for holding INFO in the buffer before forwarding it to the next node. This duration is computed from a predefined buffer threshold and a transmission interval of each INFO. When the first INFO of a flow is due, the node starts forwarding INFOs of that flow according to the sequence of their arrivals and the transmission interval of the flow. To take into account of lost INFOs, the mechanism can calculate a due time of each of the following INFOs from their sequence numbers(COUNTR) and the transmission interval. The intermediate node starts forwarding the first INFO after it reaches two time slots and transmits the following INFOs at their due time according to their sequences.

In NTM, we classify each incoming INFO as based on each INFO's arrival time, and adaptively determine a suitable due time for each INFO individually. Whether INFO is early, in time, or late depends on its relative delay.

It should be noted that the transmission power of the nodes, which directly influences the number of TM nodes, has been chosen to a transmission range of certain distances(in meters). The transmission range decreases due to the consideration of the ground in the radio propagation model with an increasing distance  $d$  m. The distribution time is measured as the time period between the initial sending and the earliest time at which all TM nodes have received the information.

Algorithm – 2 (Computation of Transmission Delay)

1. Node  $n_i$  receives  $t_{ij}$  from  $n_j$ .
2. Compute
  - a.  $Relative\_delay_{TSTMP} = delay_{TSTMP} - average\_delay_{TSTMP}$
  - b.  $average\_delay_{TSTMP+1} = (1-w) \times average\_delay_{TSTMP} + delay_{TSTMP}$
  - c.  $delay_{TSTMP} = Recieve_{TSTMP} - Send_{TSTMP}$

Where  $w$  is the weight.

Fig. 3: An Algorithm for Computation of Transmission Delay

Algorithm – 3 Trust Propagation and Termination of  $TM$

1. For every trust evolution,  $TE (ID(n_i), CONTEXT, t_{ij}, T)$  submitted by node  $n_i$  from node  $n_j$ .
2. Find  $ROUTE\_REQUEST(RTREQ)$  and  $ROUTE\_REPLY(RTREP)$  to propagate  $TE$  to the trusted nodes in  $NTM$
3.  $TM$  verifies the validity of each node by  $Tl(n_i, n_j, CNTXT) = EVAL_{SK}(D)$
4. If  $Tl(n_i, n_j, CNTXT)$  is invalid then it is discarded by  $TM$ .
5. Else  $TM$  first retrieves the timestamp  $T$  from  $Tl(n_i, n_j, CNTXT)$  Then sends an  $ACK = l$  to the destination node.

Fig. 4: An Algorithm for Trust Propagation and Termination of  $TM$

## VI. CONCLUSIONS

The goal of this paper was to provide a simple node-based trust management scheme for MANET with multiple perspectives on the concept of trust, an understanding of the properties, which should be considered in developing a trust metric, and insights on how a trust metric can be customized to meet the requirements and goals of the NTM scheme. The model is simple, flexible and easy to be implemented. After introducing and analysing the concept of node-based trust in MANET, we suggested future research directions to develop trust management schemes with desirable attributes such as adaptation to environmental dynamics, scalability and reliability. The proposed model will be simulated to gain further insight.

## REFERENCES

- [1] A. Rahman and S. Hailes. *A Distributed Trust Model*. New Security Paradigms Workshop 1997. ACM, 1997.
- [2] A. Rahman, S. Hailes, *Supporting trust in virtual communities*, In Proc. 33rd Ann. Hawaii Int'l Conf. Syst. Sci. (HICSS 33), vol. 6, (2000) pp. 6007-6016.
- [3] C. Davis. *A localized trust management scheme for ad hoc networks*. Proceedings of 3rd International Conference on Networking (ICN'04). Mar. 2004.
- [4] D. Balfanz, D. Smetters, P. Stewart and H. Wong. *Talking to Strangers: Authentication in Ad-hoc Wireless Networks*. NDSS. San Diego, 2002.
- [5] D. B. Johnson. Routing in Ad Hoc Networks of Mobile Hosts. Proceedings of the Workshop on Mobile Computing Systems and Applications, pp. 158-163, IEEE Computer Society, Santa Cruz, CA, December 1994.

- [6] D. Johnson and D. Maltz. *Dynamic Source Routing in Ad Hoc Wireless Networks*, Mobile Computing, T. Imielinski and H. Korth, Ed., Kluwer, 1996.
- [7] D. Scott, A. Beresford, A. Mycroft, *Spatial security policies for mobile agents in a sentient computing environment*, In Proceedings of FASE 2003, Lecture Notes in Computer Science, vol. 2621, Warsaw, Poland, Apr. 2003, pp. 102-117.
- [8] F.B. Schneider, Towards fault-tolerant and secure agency, In Proc. 11th Int'l Workshop on Distributed Algorithms, Saarbrücken, Germany, Sep. 1997, pp. 1-14.
- [9] J. Liu, V. Issarny, *Enhanced reputation mechanism for mobile ad hoc networks*, In Proceedings of the Second International Conference on Trust Management, vol. 2995, Mar. 2004, pp. 48-62.
- [10] K. Rothermel, M. Straber, *A fault-tolerant protocol for providing the exactly-once property of mobile agents*, In Proc. 17th IEEE Symp. Reliable Distr. Syst., 1998, pp. 100-108.
- [11] K. Xu, X. Hong, M. Gerla, *Landmark routing in ad hoc networks with mobile backbones*, J. Parallel Distrib. Comput. 63 (2) (2003) 110-122.
- [12] L. Eschenauer, V. Gligor and J. Baras. *On Trust Establishment in Mobile Ad-Hoc Networks*, Proceedings of 10th International Workshop of Security Protocols, Springer Lecture Notes in Computer Science (LNCS), Apr. 2002.
- [13] L. Zhou and Z.J. Haas. *Securing ad hoc networks*. IEEE Network Magazine, vol. 13, no. 6, pp. 24-30, November 1999.
- [14] M. Bechler, H.-J. Hof, D. Kraft, F. Phlke, L. Wolf. *A Cluster-Based Security Architecture for Ad Hoc Networks*. IEEE Infocom. 2004.
- [15] M. Blaze, J. Feigenbaum, J. Lacy. *Decentralized Trust Management*, In Proceedings of 1996 IEEE Conference on Privacy and Security, Oakland, US, 1996, pp. 164-173.
- [16] M. Gupta, P. Judge, M. Ammar, *A Reputation System for Peer-to-peer networks*, In Proc. 13th Int'l Workshop Netw. Oper. Syst. Support for Digital Audio and Video, Monterey, US, June 2003, pp.144-152.
- [17] M. Virendra and S. Upadhyaya. *Securing Information through Trust Management in Wireless Networks*. Workshop on Secure Knowledge Management (SKM 2004). Buffalo, NY, 2004.
- [18] P. Dewan, P. Dasgupta, *Securing P2P networks using peer reputations: is there a silver bullet?*, In Proc. IEEE consumer communications and networking conference (CCNC 2005), Las Vegas, US, 2005.
- [19] P. Krishna, M. Chatterjee, N. Vaidya, D. Pradhan. *A Cluster-based Approach for Routing in Ad-Hoc Networks*. Proc 2nd Symposium on Mobile and Location-Independent Computing. Apr. 1995.
- [20] R. Ferdous, V. Muthukkumarasamy, A. Sattar, *Trust Formalization in Mobile Ad-Hoc Networks*. In proceedings of the Sixth International Workshop on Heterogeneous Wireless Networks (HWISE), Perth, Australia April 20 to 23, 2010.
- [21] R. Jurca, B. Faltings. *An incentive compatible reputation mechanism*, In Proc. IEEE Int'l Conf. E-Commerce, June 2003.
- [22] S. Buchegger, J.L. Boudec, *Nodes bearing grudges: towards routing security, fairness, and robustness in mobile ad hoc networks*, In Proc. Tenth Euromicro Workshop Parallel, Distr. Netw.-based Process., 2002, pp. 403-410.
- [23] S. Buchegger, J. Boudec, *A robust reputation system for P2P and mobile ad-hoc networks*, In Proc. Second Workshop the Economics of Peer-to-Peer Systems, Cambridge, US, June 2004.
- [24] S.D. Kamvar, M.T. Schlosser, H. Garcia-Molina. *The Eigentrust algorithm for reputation management in P2P networks*, In Proc. 12th Int'l World Wide Web Conf., Budapest, Hungary, 2003, pp. 640-651.
- [25] S. Marti, T.J. Giuli, K. Lai, and M. Baker. *Mitigating Routing Misbehavior in Mobile Ad Hoc Networks*. Mobicom 2000, August 2000, pp. 255-265.
- [26] T. Beth, M. Borcherdig and B. Klein. *Valuation of trust in open networks*. Proceedings of ESORICS 1994, November 1994.
- [27] T. Hughes, J. Denny, P. Muckelbauer, J. Ettl. *Dynamic Trust Applied to Ad Hoc Network Resources*. Autonomous Agents and Multi-Agent Systems Conference, Melbourne, Australia, 2003.
- [28] T. Sander, C. Tschudin, *Protecting mobile agents against malicious hosts*, In Mobile Agents and Security, Lecture Notes in Computer Science, vol. 1419, Heidelberg, Germany, 1998, pp. 44-60.
- [29] Y. Huang and W. Lee. *A Cooperative Intrusion Detection System for Ad Hoc Networks*. Proceedings of the ACM Workshop on Security of Ad Hoc and Sensor Networks (SASN '03). Fairfax VA, October 2003.