

Provided for non-commercial research and education use.  
Not for reproduction, distribution or commercial use.



This article was published in an Elsevier journal. The attached copy is furnished to the author for non-commercial research and education use, including for instruction at the author's institution, sharing with colleagues and providing to institution administration.

Other uses, including reproduction and distribution, or selling or licensing copies, or posting to personal, institutional or third party websites are prohibited.

In most cases authors are permitted to post their version of the article (e.g. in Word or Tex form) to their personal website or institutional repository. Authors requiring further information regarding Elsevier's archiving and manuscript policies are encouraged to visit:

<http://www.elsevier.com/copyright>



ELSEVIER

Available online at [www.sciencedirect.com](http://www.sciencedirect.com)

Computer Communications 31 (2008) 760–769

---



---

**computer**  
 communications
 

---



---

[www.elsevier.com/locate/comcom](http://www.elsevier.com/locate/comcom)

## Security in mobile ad-hoc networks using soft encryption and trust-based multi-path routing

Prayag Narula <sup>a</sup>, Sanjay Kumar Dhurandher <sup>a,\*</sup>, Sudip Misra <sup>b</sup>, Isaac Woungang <sup>c</sup>

<sup>a</sup> *Division of Information Technology, Netaji Subhas Institute of Technology, University of Delhi, India*

<sup>b</sup> *Department of Computer Science, Yale University, New Haven, CT, USA*

<sup>c</sup> *Department of Computer Science, Ryerson University, Toronto, Ont., Canada*

Available online 22 October 2007

---

### Abstract

Due to their applications in situations such as emergencies, crisis management, military and healthcare, message security is of paramount importance in mobile ad-hoc networks. However, because of the absence of a fixed infrastructure with designated centralized access points, implementation of hard-cryptographic security is a challenging prospect.

In this paper, we propose a novel method of message security using trust-based multi-path routing. Less trusted nodes are given lower number of self-encrypted parts of a message, making it difficult for malicious nodes to gain access to the minimum information required to break through the encryption strategy. Using trust levels, we make multi-path routing flexible enough to be usable in networks with ‘vital’ nodes and absence of necessary redundancy. In addition, using trust levels, we avoid non-trusted routes that may use brute force attacks and may decrypt messages if enough parts of the message are available to them. Simulation results, coupled with theoretical justification, affirm that the proposed solution is much more secured than the traditional multi-path routing algorithms.

© 2007 Elsevier B.V. All rights reserved.

*Keywords:* MANETs; Multi-path; Trust; Encryption

---

### 1. Introduction

MANETs are wireless and do not require any infrastructure to set up. This makes them ideal for military, rescue and relief operations. But this flexibility and the lack of a central server or access point creates a problem of security. Since the nodes co-operate to route messages, the presence of misbehaving and non-benevolent nodes is non-trivial. There are a number of security issues associated with co-operative routing in multi-hop wireless networks such as MANETs. Primarily, they are as follows [1]:

- *Confidentiality:* The confidentiality property refers to limiting unauthorized access to sensitive information. This includes both message data and network and routing information like network topology, geographic information and node placement. Due to the self governing nature of ad-hoc networks it is difficult to achieve. Various cryptographic techniques are used to implement message confidentiality.
- *Integrity:* Message integrity is concerned with ensuring that the message is modified or allowed to be altered by only authorized agents. It also includes ensuring that all nodes follow correct routing protocols. Various cryptographic methods have been devised to improve message integrity. But however as explained later the use of cryptography and key-based encryption schemes in MANETs.
- *Availability:* Message availability ensures that the nodes are safe from denial of service attacks. This incorporates defending against network and service overload attacks.

---

\* Corresponding author.

*E-mail addresses:* [prayag@coe.nsit.ac.in](mailto:prayag@coe.nsit.ac.in) (P. Narula), [sdhurandher@coe.nsit.ac.in](mailto:sdhurandher@coe.nsit.ac.in) (S.K. Dhurandher), [sudip.misra@yale.edu](mailto:sudip.misra@yale.edu) (S. Misra), [iwoungang@scs.ryerson.ca](mailto:iwoungang@scs.ryerson.ca) (I. Woungang).

- *Authentication*: Authentication requires that the communicating identities be assured the identity of each other. These identities should be authorized and recognized before the communication begins.
- *Access control*: Only the authorized agents should be able to access and use the resources and services provided by the network. This should be done in accordance to their access rights and group memberships.
- *Non-repudiation*: Non-repudiation refers to the property by the virtue of which a receiver cannot deny having received a message. The same holds well for sender too, that is, a sender cannot deny having sent a message if it has done so.

Various routing protocols have been proposed for MANETs. These vary from the table-driven protocols like Destination-Sequenced Distance Vector (DSDV) [2], which is based on the classical Bellman-Ford algorithm, to on-demand protocols like Dynamic Source Routing (DSR) [3] and Ad-hoc On Demand Distance Vector (AODV)[4]. These protocols work well in benign environments, but they have to be modified substantially if they are to be used in a hostile network. In a network in which malicious nodes might be present, protocols such as those mentioned above may cause serious security concerns.

We propose a method to securely route messages in an ad-hoc network using multi-path routing and trustworthiness of the nodes. Hence, we aim at addressing the issues underlying *message confidentiality*, *message integrity* and *access control*.

We divide the message into different parts and encrypt these parts using one another [5]. We then route these parts separately using different paths between a pair of source–destination nodes. An intermediate node can access different parts on the basis of its trustworthiness. That is, a more trusted node is allowed to feature in more paths than a less trusted node and hence access to more message parts than a less trusted node. This feature allows the routing algorithm to avoid nodes that are more likely to attempt ‘breaking-in’ the encryption. In addition, suspected nodes which have high computation power and are hence likely to be more successful in cryptanalysis can be given less parts to stymie their plans.

Since establishment of trust also requires cryptographic key exchange, we use a soft approach to trust [6]. Trust levels of peer nodes of the network are found using effort-return based trust model [6]. We use a variation of the model given in [7] and [8], which uses a combination of *derived trust and reputation* to estimate trust values of a node.

In sum, our contributions are as follows:

- We combine multi-path routing and trust with soft encryption technology to propose a scheme which is much more secure than traditional multi-path algorithms. By soft encryption, we mean having encryption

systems that are somewhat less secure than the sophisticated encryption methods, but are more efficient in terms of performance and require less resources [8].

- Our algorithm even works in the cases where a ‘vital’ node is present in the network structure, unlike most other multi-path security algorithms. This is further clarified in Section 2.
- We provide ‘soft’ encryption by using the message itself for encryption. This technique eliminates the need of Key Distribution Centers and key transfer.

The paper is organized as follows. We first discuss the previous works and discuss their limitations. We then discuss the motivations behind our work. This is followed by a detailed discussion of our algorithm including a discussion on our trust assignment, message encryption policy and routing strategy. We also present various lemmas, which provide an insight into the theoretical aspects on which our work is based. We then explain the simulation scenario and explain the results. We then present the future works that can be done in relation to the work and end with a few concluding remarks.

## 2. Previous works

Security in MANETs has been a topic of much discussion in the last few years. There are a plenty of works available in the literature that discuss security in MANETs (e.g., [1,5–16,22–32]). But efficiently providing complete message security in such networks still remains an open issue. We can broadly divide the various popularly known protocols for implementing security in MANETs into the following broad categories<sup>1</sup>:

- *Payment Systems* [9]: Systems that aim at providing monetary rewards to the nodes for benign behavior.
- *Reputation Systems* [11,12]: Systems that achieve security by awarding better reputation to nodes that show benign behavior. Nodes which have better reputation get better service from their peers in return of their own co-operating behavior.
- *Cryptography-based systems* [14]: Cryptography based systems use various cryptographic methods for implementing security. These systems generally use algorithms that make it computationally difficult for malicious nodes to break-in to the security that these algorithms provide such as message encryption, anti-modification and packet fabrication.

<sup>1</sup> It should be noted that this is a broad classification of some of the popularly known protocols. There may be some protocols which may not fall into any of these categories.

## 2.1. Payment systems

The purpose of payment systems is to encourage cooperation within a MANET by providing economic incentives to the benevolent and co-operating nodes. The security provided by payment systems is generally aimed at promoting better behavior rather than using any ‘hard’ security methodologies. Using virtual currency called *Nuglets* [9] is a popular payment based security system.

Nuglets is a virtual currency for charging and paying for server usage. Nodes can charge for the services they provide. Typically, an intermediate node may demand to be paid for forwarding a packet to the next node. The payment (done in Nuglets) is usually done by the source or the destination node depending on the model used.

If the source node pays for the services, it is known as the *Packet Purse Model (PPM)* [9]. In PPM, the source loads up the message with Nuglets after estimating the number of nodes lying in the path. The intermediate nodes acquire some Nuglets from the packet depending against the service provided. In this way, the packet is relayed to the destination. If the Nuglets are finished before the message reaches the destination, it is discarded and the process should start again. Hence, the source needs to make a near accurate estimation on the number of nodes lying on the route to the destination. This can be cumbersome or even impossible sometimes.

Hence, another model has been proposed which directs the intermediate nodes to buy a packet from the previous node and sell it to the next node on the route. In this model known as the *Packet Trade Model (PTM)* [9], the destination pays for the message by buying the message from its previous node. If the destination node refuses to buy the packet, the message is discarded.

By making the source pay for the packets sends, PPM discourages nodes from sending useless data. This prevents network from packet flooding attacks. On the other hand, PTM can lead to an overuse of the network by sending packets which the destination does not want. But in turn, PPM requires the source to accurately estimate the number of Nuglets it should include with the packet while PTM does not require any estimation on the part of the sender.

### 2.1.1. Limitations of payment systems

The payment systems require tamper resistant hardware for storing Nuglets with the message. If that’s not available, a central authority is required to calculate the charges and credits for various nodes. Tamper resistant hardware increases the cost, size and energy requirements of a mobile device and hence is an impractical assumption. Additionally, MANETs inherently lack a central authority and, hence, cannot be assumed to be present in pure ad-hoc networks. Moreover, most payment systems suffer from what is known as *locality problems*. Nodes in different location would have different chances of earning virtual money and, hence, such a model lacks fairness. For example, in most cases, nodes present at the edges would have less

chances of earning Nuglets as their chances of lying on a route are lesser than nodes lying in the center.

Payment systems, instead of providing message security, actually work towards promoting a healthy environment in MANETs. In addition, since payment systems directly provide economic incentives, they are more suitable for applications involving e-commerce. Involving direct monetary incentives make these systems a target of choice for malevolent agents. Hence, payment systems are not highly suitable for providing message security in general MANETs.

## 2.2. Reputation systems

Reputation systems are becoming increasingly popular for securing online transactions. Such systems promote agents with better reputation as better prospects for performing online transactions. Such systems are suitable for MANETs as the nodes act on the basis of a mutual *trust* that the peer nodes would act benevolently. Such a trust can be quantified by using a reputation system. But generally, in MANET applications, unlike other systems, each node maintains its own reputation rating for its peers, on the basis of direct observations or peer recommendations. Generally, the reputation systems work towards promotion of benign behavior among nodes, by providing better services to them in exchange of a co-operative and benign behavior. Although reputation systems bear some similarities with the payment systems, they are not directly economic in nature, though, indirectly it may lead to monetary advantages. For example, a node with better reputation may get an advantage in terms of forwarding its message earlier than a node with lower reputation.

Most (not all) trust management systems in MANETs are reputation-based systems. Two examples of well-known reputation based systems are CONFIDANT [11] and CORE [12]. Besides CONFIDANT [11] and CORE [12], there are many other reputation systems for MANET. Without delving into cataloguing them, we can classify them into the following categories:

1. *Global reputation systems* [13]: In these systems, each node knows the reputation value of every other node in the network. This is achieved by exchanging an indirect reputation message over the network. CONFIDANT [11] and CORE [12] are examples of global reputation system.
2. *Local reputation systems* [13]: In these systems, each node only keeps the reputation value of its neighboring nodes. Instead of distributing the reputation value or information periodically, the local reputation systems usually update the reputation value based on its own observation.

### 2.2.1. Limitations of reputation systems

In all reputation systems, each node receives a feedback on what other nodes *think* of it. This mechanism can be either

direct, that is, based on reputation table broadcasts such as in the case of CONFIDANT [11], or indirectly by observing the positive recommendation about other nodes, as in the case of CORE [12]. This may lead to *grunge war* by the node which receives a negative feedback about itself.

Our proposed (described in Section 4) partially addresses the problem by providing an on-demand reputation system. Our system also discourages using promiscuous modes, and it uses active acknowledgements instead of passive ones and it promotes the use of directional antennas to enhance security. This ensures that a node's feedback remains hidden unless that node makes some efforts to snoop on other nodes. This decreases the probability of a grunge war.

### 2.3. Cryptographic methods

Several cryptographic methods have been proposed for MANETs. Most of them are based on existing routing protocols to install security features against attacks such as message modification, Denial of Service (DoS), message modification and others.

ARIDANE [14] is an example of a cryptographic method based on-demand protocols such as Dynamic Source Routing (DSR). ARIDANE has been designed to be effective against attacks such as Denial of Service (DoS) [1] attacks in ad-hoc networks. The advantages of ARIDANE lie in the fact that it is computationally un-intensive and only adds a message authentication code (MAC) to a message for broadcast authentication. ARIDANE primarily authenticates packets containing Route Request, Route Reply and Route Error, to prevent misbehaved nodes changing route information.

#### 2.3.1. Limitations of cryptographic methods

A key limitation of cryptographic methods is that they assume an effective key distribution mechanism. In systems as dynamic as MANETs, such an assumption is not practical. Key distribution is a non-trivial problem in MANETs. Nodes may join and exit a network at any point of time. Moreover, a key distribution server may not be able to communicate with all the nodes.

In view of these problems, few algorithms (e.g., [5]) have been proposed that use encryption based on method parts themselves. These systems, though not as strong against attacks as other cryptographic methods, are flexible.

We have selected such systems and implemented them in conjunction with a trust-based reputation system and a multi-path routing to provide a secure routing scheme which tries to alleviate some of the major problems associated to such systems.

### 3. Motivation

As mentioned in Section 2, various payment-based, reputation-based and cryptography-based schemes have been proposed in the literature. We have also reviewed the key limitations of each category of security systems. As already

discussed, the cryptographic methods are much better than the payment systems or reputation systems since they can well address message modification and fabrication attacks. However, cryptographic methods cannot be effective against packet dropping attacks since they have an inherent problem of key-distribution associated with them. Moreover, a stand-alone reputation-based system is insecure it is vulnerable to problems of multiple co-operating mischievous nodes. In fact, multiple nodes may co-operate to compromise the integrity and the “web-of-trust” type security provided by these reputation systems.

In addition, multi-path routing has been traditionally used in wired networks for providing various QoS guarantees [15]. A number of work has been done on using multiple paths to secure message transfer, but most of the algorithms are based on the strategy which consists of routing different parts of the message using different paths [16]. Most of these algorithms depend on finding  $k$  disjoint paths between the source and destination nodes, where a message is divided into  $n$  parts, where  $n > k$  [16]. There may not be enough redundancy in the network. In addition, there may be a *vital* node in the network which exists in all the valid paths. In such a scenario, these algorithms would not work even if the vital node is a trusted node and is allowed access to the complete message. A sample topology of such network is shown in Fig. 1.

In such a topology, even if we are sure about the non-malicious intent of node  $D$ , no secured routing can be established between nodes  $A$  and  $G$  since no disjoint paths can be found between nodes  $A$  and  $G$ . For the proposed algorithm to work, can be easily inferred that the trust level of node  $D$  has to be 4. If this is not the case, the sender can be made aware of the situation. A sender may consider increasing the trust level of the node so that the communication can take place, depending on whether or not the node wants to compromise security required for achieving the message delivery.

In most multi-path algorithms, it is often trivial to gain access to at least some parts of the messages. Neither of these algorithms takes into account the possibility that some of the nodes might be more malicious than others, either due to intent, or to capability (meaning that certain nodes may have more computation power to be able to perform more successful brute force attacks). Thus, most of these algorithms make the message parts available indiscriminately, therefore are vulnerable to brute force attacks, especially when a large part of the message is available to a compromised node. It is important to take into account the

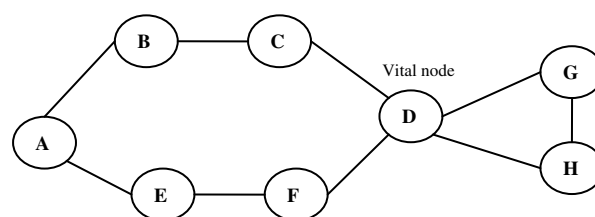


Fig. 1. No disjoint path to node  $G$ .

trustworthiness of the nodes and route the message parts accordingly. This would make multi-path routing algorithms more flexible by allowing a complete message to be routed via the trusted nodes, if required. In addition, this would also help in limiting data access to nodes more likely to carry out brute force and other attacks and if the trust level is low enough in avoiding them completely.

The problems associated with the standalone systems mentioned above make it difficult for them to provide a complete solution for secure routing in MANETs. We believe that these methods can be used together to provide a better security infrastructure, without making any non-standard assumptions about these networks. We use a combination of soft message encryption, trust establishment and multi-path routing and implement these techniques over DSR to propose a pragmatic approach to security in MANETs.

### 3.1. Message encryption

Most of the encryption-based security mechanisms are based on secured key exchanges. *A-priori* negotiations are required for key exchanges in dynamic ad-hoc networks. Pre-shared keys are used in networks which are less dynamic in nature [16]. Such networks are sometimes called “managed ad-hoc networks” [8].

In [17], the authors use a distributed approach in which multiple nodes collaborate to act as a Certification Authority [8]. A message encryption technique is used in which each part of the message is involved in encrypting the whole message itself. This helps avoiding the problem of key exchange since the message parts are themselves used as the keys. This encryption technique is used in this paper.

### 3.2. Trust establishment

The authors in [8] state that all the trust establishment protocols depend on a Central Trust Authority. In [6] a distributed model based on peer recommendation is presented. The authors define trust as a subjective entity which is transitive in nature under certain stated conditions. This definition generalizes the notion of trust, thus defines a complete trust model in which different nodes give a subjective, discrete and dynamic trust values to their peers based on repeated interactions. The authors in [7] and [8] define a trust model explicitly for MANETs. They give continuous and normalized trust levels depending on the benevolent behavior shown by the nodes. They use the Dynamic Source Routing (DSR) protocol [3] for routing via the trusted nodes only [18].

## 4. Proposed routing strategy

### 4.1. Trust

We use a variation of the trust models used in [6] and [7] in our algorithm. A node is assigned a discrete trust level in

the range of  $-1$  to  $4$ . A trust level of  $4$  defines a complete trust and a trust level of  $-1$  defines a complete distrust. These trust levels also define the maximum number of packets which can be routed via those nodes. A trust level of  $-1$  signifies that any packet coming from that node should be dropped. No packet is in turn routed to these nodes, leading to an isolation of malicious nodes.

### 4.2. Trust levels assignment

The trust level assigned to a node is a combination of direct interaction with its neighbors and the recommendations from its peers. A node assigns a direct trust level to its neighbor on the basis of the acknowledgements received. If the neighbor sends a prompt acknowledgement of the packet received, it is assumed that the node is not involved in a resource intensive brute-force attack and hence is assigned a higher trust level. The direct trust is then combined with the trust recommendation from its peers and a final trust level is assigned to it. Note that these trust levels are assigned dynamically and are cached by a node for performance enhancement. The trust recommendations are piggybacked on DSR routing packets that is explained in Section 4.5.

Let us consider Fig. 2. Let  $T_{xy}$  represents the direct trust in node  $Y$  by node  $X$  and let  $T_{yz}$  represents the trust recommended by the node  $Y$  in node  $Z$ . If  $T'_{xz}$  represents the direct trust of node  $Z$  in node  $X$ , then the trust assigned by  $X$  in  $Z$  is given in Eq. (1) below [7].

$$T'_{xz} = 1 - (1 - T_{xz}) \cdot (1 - T_{xyz}) \quad (1)$$

where

$$T_{xyz} = 1 - (1 - T_{xy})^{T_{yz}} \quad (2)$$

The trust levels are normalized to integer values using standard methods. Each node is given an integer trust value lying between  $-1$  and  $4$ .

If a new node joins the network, it sends a *hello* packet to its neighbors. The neighbors would assign an initial trust value of  $0$  to the node. The trustworthiness of the node can be increased if the node shows benevolent behavior. Similarly, when a node leaves the network, it would no longer respond to the messages. The neighbor may conclude that the network has lost its connectivity or the node has exited the network. In this scenario, the network

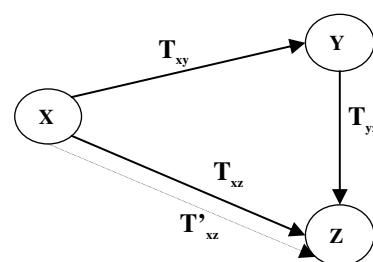


Fig. 2. Direct and derived trust.

would delete the node from its network's table and would broadcast this information to other nodes in the network. These nodes would then delete this table from their route cache.

### 4.3. Message encryption and routing

We use the message encryption method proposed in [5]. A  $4n$ -bits message is divided into four parts of  $n$  bits each. Let us denote these parts by  $a, b, c$  and  $d$ . We define the bit operation XOR on bit vectors  $k$  and  $l$  as follows:

$$\text{if } k = \{k_1, k_2, k_3 \dots k_n\}$$

$$\text{and } l = \{l_1, l_2, l_3, \dots l_n\}$$

then

$$k \text{ XOR } l =$$

$$\{k_1 \text{ XOR } l_1, k_2 \text{ XOR } l_2, k_3 \text{ XOR } l_3, \dots, k_n \text{ XOR } l_n\} \quad (3)$$

The aforementioned parts  $a, b, c$ , and  $d$  are then encrypted by means of the following equations:

$$d' = a \text{ XOR } c \quad (4)$$

$$b' = b \text{ XOR } d \quad (5)$$

$$c' = c \text{ XOR } b \quad (6)$$

$$d' = d \text{ XOR } a \text{ XOR } b \quad (7)$$

The parts  $a', b', c'$  and  $d'$  are now routed instead of  $a, b, c$  and  $d$ . Paths between the source and destination nodes are found using DSR. A node waits for intermediate multiple paths to the destination. Routing paths are selected from the set of paths using a novel *trust defined strategy*, which is described in Section 4.4.

### 4.4. Trust defined strategy

We define the *trust defined strategy* to secure routing as the policy in which a node with a trust level of  $x$  is given at most  $x$  parts of the packet to forward. This limits the possibility of using a brute force decryption of the message. For example, if the nodes are assigned four levels of trust (trust 1–4), excluding no trust and complete distrusts (trust level of 0 and  $-1$ ), the message would be divided into four parts. Hence,

- (A) A node with a trust level of 4 can read the message. Hence, only those nodes that have been certified to be completely safe can be given the right to read the full message. These might include nodes which are directly visible in case of military applications, or nodes with which keys have been exchanged securely.
- (B) A node with a trust level of 3 can be sure of finding  $2^n$  possible messages of which one would be correct, where  $n$  = number of bits used for encryption. For example, if a 32-bit message is sent as four 8-bit messages, then a node with trust level 3 would receive

3 bytes. Assuming that remaining byte out of 256 possibilities can be obtained through a brute force search, such node can find the entire message.

- (C) Using a similar process, a node with a trust level of 2 can be sure of finding  $2^8 * 2^8$  possible messages.
- (D) Similarly, a node with a trust level of 1 can be sure of finding  $2^8 * 2^8 * 2^8$  possible messages.
- (E) A node with a trust level of zero is not given any part of the message. These nodes are either nodes that act as sinks, thus are not forwarding any message or nodes that mangle the messages before forwarding.
- (F) A node with a trust level of  $-1$  is a certified malicious node. All packets received from this kind of node are dropped immediately. Measures are taken to limit any promiscuous access of message parts by this node.

Hence the probability of comprehending the entire message decreases by a factor of  $2^n$  as the trust level decreases.

At the destination node, the message parts can be decrypted using the following equations [5]:

$$a = b' \text{ XOR } d' \quad (8)$$

$$b = a' \text{ XOR } b' \text{ XOR } c' \text{ XOR } d' \quad (9)$$

$$c = a' \text{ XOR } b' \text{ XOR } d' \quad (10)$$

$$d = a' \text{ XOR } c' \text{ XOR } d' \quad (11)$$

### 4.5. Implementation using DSR

DSR is an on-demand routing protocol. We implement our algorithm over DSR. When a node intends to route a message securely to a destination, it broadcasts a Route Request (R\_REQ) packet. If this packet reaches the destination, or a node which has a path to the destination in its cache, it sends a Route Reply (R\_REP) to the source. The R\_REP message is appended with the trust level of the previous node by the node sending the route backwards along a path.

For example, in the network presented in Fig. 3, the R\_REP packet sent by the destination node  $H$ , contains the path  $\{A, B, C, D, H\}$  back to the source node  $A$ , using the path in the reverse. When the packet reaches the node  $C$ , it appends to it a trust value in the packet for  $D$ . Similarly,  $B$  appends the trust value for  $C$ .

Node  $A$  can also explicitly request the trust values of nodes lying in the path from other nodes in the network,

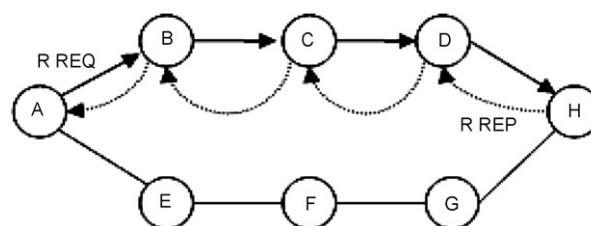


Fig. 3. DSR routing protocol.

by sending a recommendation request. This is explained in detail in [5], and is omitted here to maintain the brevity of this paper.

#### 4.6. Route selection

Whenever a new path is discovered and the trust levels of the nodes involved are available, efforts are made to select a secure route. The routes are selected using a greedy approach on the basis of path length, such that a node with a trust level  $T$  does not get more than  $T$  packets on the route.

The following steps are used to find the secure routes from a set of given routes:

1. Whenever a new route is found, the routes are rearranged in the increasing order of hop counts. This step ensures that the chosen route set consists of the smallest possible routes that can securely route the message without causing large overheads associated with the multi-path routing.
2. The first route is selected and the maximum numbers of parts of the message that can be routed via it are assumed to be routed. Note that no actual routing is done at this step.
3. The next route is selected and the maximum number of parts of the message that can be routed via it are assumed to be routed. If all the parts of message can be routed securely, the actual routing is done by the selected paths.
4. This process is repeated until secured routes are found.
5. If no secured routes are found, the algorithm is repeated by starting at Step 2, by selecting the second route as the first route.
6. This algorithm is repeated until all the combination of routes have been exhausted.
7. If no secured route is found, the algorithm waits for another route.
8. If all routes have been found or a specific time interval has been surpassed, the algorithm is assumed to have failed and a failure message is displayed.

Note that the algorithm has a worst case complexity of  $O(n^m)$ ; where  $n$  represents the number of paths and  $m$  represents the number of parts in a message. As we will see in Section 5, for our simulation purposes, we have assumed  $m$  to be equal to 4. It is acknowledged that the secured routes can be found in a more effective manner by using a back-tracking approach, since the computation time here is assumed to be negligible as compared to the time taken for finding a new path. We believe that this algorithm is computationally effective enough for the desired purpose.

Fig. 4 presents the algorithm used for selecting the routes to securely carry the data from source to destination.

```

Arrange the paths  $P = \{P_1, P_2, P_3, \dots, P_n\}$  in an increasing
order of path lengths

Initialize Count  $C_i$  for all nodes to 0

Select the smallest path from  $P\{$ 
  Select the next smallest path
  if( for all selected nodes  $i, C_i \leq T_i\{$ 
    if( four paths are selected)
      break the loop;
    else
      continue;
  }

  if(all paths are exhausted)
    wait for another path
}

if (no paths left)
  Print("Not possible to route securely")

```

Fig. 4. Algorithm to select secure routes.

**Lemma 1.** *All generic multi-path algorithms use a static and equal trust levels for all nodes present in a network*

In a generic multi-path routing algorithm, a message is divided into  $n$  parts, of which  $m$  parts are required to decrypt the message, where  $n \leq m$ . The  $n$  parts are then routed using  $n$  different paths to the destination node. We have modified our algorithm slightly and we have assumed that the message is divided into  $n$  parts, out of which at least  $m$  are required to decrypt the message. In addition, if we assume that the trust level of each node is constant and equal to 1, i.e.

$$T_i = 1$$

then, the proposed algorithm would route all the parts of the message using different routes. That is, the  $n$  parts of the messages are routed using  $n$  different paths.

Thus, we can conclude that all the generic and pure multi-path algorithms use a static and equal trust level for all nodes in a network. Hence, we can infer that, inherently, all generic multi-path routing algorithms use trust based routing but the trust assigned is not dynamic and is constant.

**Lemma 2.** *Two sectors of a network can communicate, even when there is just one connecting node, given the vital node has the highest trust level.*

Let us consider the network given in Fig. 5. The two rectangles demark the two sectors of a network. It should be noted that node 4 is the vital node.

Now, if node 1 has to communicate with node 9, the multi-paths between these nodes will consist of a common node 4. If node 4 has complete trust in node 1, node 1 can communicate with the nodes in the other section (the rectangle on the right hand side, including node 9), and the secured route can be established.

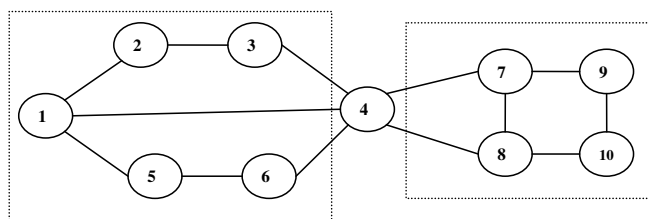


Fig. 5. Lemma 2 illustration.

### 5. Simulation and results

The simulation experiments conducted were evaluated in Global Mobile Information System Simulator (GloMoSim) [19]. GloMoSim is a scalable simulation environment for large wireless and wireline communication systems using parallel discrete-event simulation language called PARSEC [20]. GloMoSim is an event based system coded in C. GloMoSim implements all the seven layers of the ISO-OSI reference model and is customizable and assessable at all the layers. It supports various pre-compiled models and protocols at various layers including the DSR routing algorithm at the network layer, which was used as a basis for our system. On the Medium Access Control (MAC) layer, protocols such as CSMA, FAMA, MACA and IEEE 802.11 are currently available. On the application layer, models such as TCPLIB, CBR (Constant Bit Rate) and HTTP traffic are supported. It is at the application layer that the soft encryption using various message parts is implemented. A Java-based visualization tool is available for graphical visualization of the protocols implemented.

It is assumed that the trust levels of various nodes would be available to the source node via piggybacking the recommendations on the route response packets. To avoid complexity, each node randomly assigns trust levels to its peers. This is done in such a manner that most nodes have trust levels of either 2 or 3. Lesser number of nodes have trust level of 1 and even lesser number of nodes have a trust of 0 or 4. Very few nodes have a trust level of -1.

#### 5.1. Set up

The number of nodes was varied in a terrain measuring 2000 m × 2000 m. To maintain connectivity, radio transmission power was varied accordingly. The MAC protocol used is IEEE 802.11 [21]. Nodes were placed uniformly throughout the terrain and simulation was allowed to run for 600 s.

#### 5.2. Parameters

The algorithms were compared on the parameters measuring *security* and *route selection time*. Security was measured on the basis of access violation of the trust defined strategy that was presented in Section 4.4. *Trust compromise* is measured as the total sum of access violations in all the paths used for routing. *Access violation* is measured

as the difference between the number of packets a node gets and the trust level of that node, if the trust level is lesser than the number of packets. Formally, if  $S$  denotes the set of nodes used for routing, then for a node  $s$  with assigned trust  $T_s$  by the source, if  $s$  receives  $N_s$  different packets from all routes, then

$$\text{Trust compromise} = \sum_{s \in S} (N_s - T_s), \text{ where } N_s > T_s \quad (12)$$

The total trust compromise is calculated for all the paths selected for routing. Since the normal DSR uses the first path it receives to route a message, only one path is considered in DSR. It is clear that increasing the trust compromise for a path set would have the side effect of decreasing the level of message security. Ideally, the trust compromise of a path should be zero to ensure that only minimal access is given to the peer nodes to a message. Increasing the trust compromise would also increase the probability of a message to be compromised and broken by a malicious agent. The sum of individual trust compromises models the co-operation that may be take place between the malicious nodes. For example, if a message has a compromise of 4 and if each compromise takes place at a separate node for a separate message part, the whole message can be read if the malicious nodes are co-operating. Similarly, if the aggregated trust compromise is higher then there are more chances that compromising and co-operating malicious agents will break the message.

*Route selection time* is defined as the total time required for selecting a path set for routing. Since DSR uses the first path it receives, its path selection time is the time taken in getting the first route reply.

**Lemma 3.** *The trust compromise of the selected routes for soft encryption and trust based, multi-path routing is always zero.*

From Eq. (12):

$$\text{Trust compromise} = (N_1 - T_1) + (N_2 - T_2) + (N_3 - T_3) + \dots, \text{ wherever } N_i > T_i$$

Now in the algorithm, a node can never have more parts than its trust level. Hence,  $T_i \geq N_i$ . Hence, there exists no route for which  $N_i > T_i$  is true, thus the result.

#### 5.3. Results

The performance comparison is done in between the algorithms on the basis of the metrics described in Section 5.2. The results are indicated in the plots in Figs. 6 and 7. The results is in agreement with Lemma 2. Since trust based multi-path routing selects a route such that no node receives more parts of a message than its trust level, it has a trust compromise of zero in all cases. Moreover, since a multi-path routing using 2 disjoint paths sends message along different disjoint paths, its trust compromise is zero when all the involved nodes have a trust level greater than

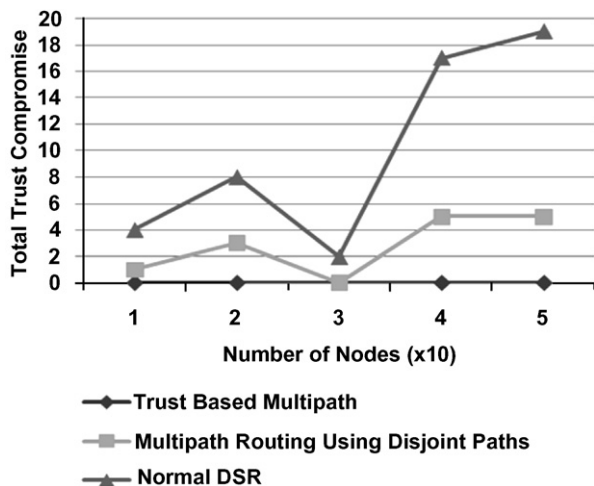


Fig. 6. Comparison of trust compromise.

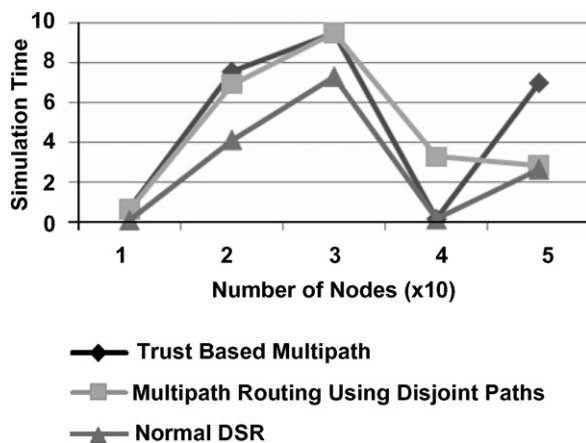


Fig. 7. Route selection time.

or equal to 2. But since that may not necessarily be the case, trust compromise for such an algorithm varies between 0 and 5 in the results. In addition, since DSR routes a message via the first message it received, the message security in DSR is minimal compared to other algorithms. The trust compromise for the normal DSR varies between 2 and 14 in the observed results.

The route selection times for the three algorithms are presented in Fig. 7. Since DSR selects the first path it receives as the path set, the route selection time of DSR is minimum. The disjoint multi-path routing algorithm has to wait for at least two paths till it can select a path set. In addition, it may have to wait for a longer time depending on how much time it takes to receive a disjoint path. A node may not receive a disjoint path in case a 'vital' node exists or if there is not enough redundancy in the network. Trust based multi-path routing generally (but not always) takes the longest time in route selection since it requires trusted paths which may take a longer time to come. But in cases where all the nodes of the path received first are trusted, the route selection time of the trust based multi-path routing can be equal to that of a

normal DSR. The route selection times of all three algorithms depend on the placement of the source and destination nodes in a network as well as on the network topology.

Hence, we observe that there is a compromise between message security (trust compromise) and routing time (route selection time), which is generally the case with most security algorithms. A balance must be made between these two concepts in order to provide a maximum security level without causing a substantial delay for a user of the network.

## 6. Conclusions

In this paper, we proposed a new routing strategy towards message security in MANETs. We first presented a comprehensive discussion on previous works that have been done on the topic of security in MANETs. We discussed various methods that have been proposed and highlighted their respective advantages and limitations in various scenarios. We have provided some ideas of possible solutions to these problems and we have discussed how our proposed system can incorporate these solutions.

Based on these settings, we have introduced a trust-based multi-path algorithm for message security in MANETs. We have discussed the message encryption and the trust establishment methodologies that can be used in the system. Then, we have proposed a *trust based strategy* for route selection. The implementation of this trust-based approach using DSR was then discussed. Finally, the simulation results obtained from our algorithm are compared against the results obtained using traditional algorithms such as normal DSR and multi-path routing using disjoint paths, used as benchmarks.

Our proposed solution proved to be much more secured than the results obtained from traditional multi-path routing algorithms.

In the future, we plan to investigate how to design a more efficient algorithm for selecting the routes from a set of routes. For example, a backtracking technique can provide a better solution. Strong encryption methodologies would be analyzed to provide a better security mechanism. Implementation using other routing algorithms such as AODV and TORA [33] should also be researched.

## References

- [1] A. Mishra, K.M. Nadkarni, Security in wireless ad-hoc networks, in: M. Ilyas (Ed.), *The Handbook of Wireless Ad-Hoc Networks*, CRC Press, 2003, Chapter 30, ISBN 0849313325.
- [2] C.E. Perkins, P. Bhagwat, Highly Dynamic Destination-Sequenced Distance-Vector Routing (DSDV) for mobile computers, in: *Proceedings of the Conference on Communications Architectures, Protocols and Applications*, London, UK, 1994, pp. 234–244 (August 31–September 2).
- [3] D. Johnson, D. Maltz, Dynamic source routing in ad hoc wireless networks, in: T. Imielinski, H. Korth (Eds.), *Book Chapter in Mobile Computing*, Kluwer Academic Publishers, Dordrecht, Netherlands, 1996, pp. 131–181.

- [4] C.E. Perkins, E.M. Royer, Ad-Hoc On demand distance vector routing, in: *Proceedings of IEEE WMCSA'99*, New Orleans, LA, February 1999, pp. 90–100.
- [5] T. Haniotakis, S. Tragoudas, C. Kalapodas, Security enhancement through multiple path transmission in ad hoc networks, *IEEE International Conference on Communications (2004)* 4187–4191, June.
- [6] A. Abdul-Rahman, S. Hailes, A distributed trust model, in: *Proceedings of the 1997 Workshop on New Security Paradigms*, Langdale, Cumbria, UK, September 23–26, 1997, ACM Press, NY, pp. 48–60.
- [7] A.A. Pirzada, A. Datta, C. McDonald, Propagating trust in ad-hoc networks for reliable routing, in: *Proceedings of 2004 International Workshop on Wireless Ad-Hoc Networks*, May–June 2004, pp. 58–62.
- [8] A.A. Pirzada, C. McDonald, Establishing trust in pure ad-hoc networks, in: Estivill-Castro (Ed.), *Proceedings of the 27th Australasian Conference on Computer Science*, Dunedin, New Zealand, pp. 47–54.
- [9] L. Buttyan, L., J.-P. Hubaux, Stimulating Cooperation in Self-Organizing Mobile Ad Hoc Networks. Technical Report DSC/2001/046, EPFL-DI-ICA, August 2001.
- [10] R. Norcen, A. Uhl, Encryption of wavelet-coded imagery using random permutations, *Proceedings of 2004 International Conference on Image Processing 5 (2004)* 3431–3434, October 24–27.
- [11] S. Buchegger, J.-Y. LeBoudec, Performance analysis of the CONFIDANT protocol: cooperation of nodes—fairness in dynamic ad-hoc networks, in: *Proceedings of IEEE/ACM Symposium on Mobile Ad Hoc Networking and Computing (MobiHOC)*, Lausanne, Switzerland, June 2002.
- [12] P. Michiardi, R. Molva, CORE: a collaborative reputation mechanism to enforce node cooperation in mobile ad hoc networks, in: *Proceedings of the IFIP TC6/TC11 Sixth Joint Working Conference on Communications and Multimedia Security*, 2002, pp. 107–121.
- [13] S.D. Kamvar, M.T. Schlosser, H.Garcia-Molina, The EigenTrust Algorithm for reputation management in P2P networks, in: *Proceedings of the Twelfth International World Wide Web Conference*, Budapest, Hungary, 2003, pp. 640–651.
- [14] Y.-C. Hu, A. Perrig, Johnson D.B. Ariadne, A secure on-emand routing protocol for ad hoc networks, in: *Proceedings of MOBICOM 2002*, Atlanta, Georgia, USA.
- [15] I. Cidon, R. Rom, Y. Shavitt, Analysis of multi-path routing, *IEEE/ACM Transactions on Networking* 7 (6) (1999) 885–896, December.
- [16] W. Lou, W. Liu, Y. Fang, SPREAD: enhancing data confidentiality in mobile ad hoc networks, in: *Proceedings of the Twenty-third Annual Joint Conference of the IEEE Computer and Communications Societies, INFOCOM 2004*, March 2004, pp. 2404–2413.
- [17] W. Wang, Y. Zhu, B. Li, Self-managed heterogeneous certification in mobile ad hoc networks, *Proceedings of the Vehicular Technology Conference (2003)* 2137–2141, VTC 2003.
- [18] A.A. Pirzada, A. Datta, C. McDonald, Trust-based routing for ad-hoc wireless networks, in: *Proceedings of the Twelfth IEEE International Conference on Networks*, 2004. (ICON 2004), pp. 326–330, Nov. 2004.
- [19] M. Takai, L. Bajaj, R. Ahuja, R. Bargrodia, M. Gerla, GloMoSim: a scalable network simulation environment, Technical Report 990027, Department of Computer Science, University of California, Los Angeles, USA, 1999.
- [20] R. Bargodia, R. Meyer, M. Takai, Y.-A. Chen, X. Zeng, J. Martin, H.Y. Song, PARSEC: A Parallel Simulation Environment for Complex Systems, *IEEE Computer* 31 (10) (1998) 77–85, Oct..
- [21] IEEE Computer Society LAN MAN Standards Committee, Wireless LAN Medium Access Protocol (MAC) and Physical Layer (PHY), IEEE Std. 802.11–1997. The Institute of Electrical and Electronic Engineers, 1997.
- [22] P. Papadimitratos, Z.J. Haas, Secure routing for mobile ad hoc networks, in: *Proceedings of SCS CNDS*, San Antonio, TX, January 27–31, 2002, pp. 193–204.
- [23] Y. Hu, A. Perrig, D.B. Johnson, Secure efficient distance vector routing for mobile wireless ad hoc networks, *Ad Hoc Networks* 1 (1) (2003) 175–190, Jul..
- [24] P. Papadimitratos, Z.J. Haas, Secure QoS-aware route discovery in ad hoc networks, in: *Proceedings of 2005 IEEE Sarnoff Symposium*, Princeton, NJ, April 2005, pp. 176–179.
- [25] Papadimitratos, P., Secure and Fault-Tolerant Communication in Mobile Ad Hoc Networks, Ph.D. Thesis, Cornell University, Ithaca, NY, January 2005.
- [26] S. Marti, T.J. Giuli, K. Lai, M. Baker, Mitigating routing misbehavior in mobile ad hoc networks, in: *Proceedings of Sixth MobiCom*, Boston, MA, August 2000, pp. 255–265.
- [27] B. Dahill, B. Neil, E. Royer, C. Shields, A secure protocol for ad hoc networks, in: *Proceedings of IEEE ICNP*, 2002.
- [28] V. Gupta, S. Krishnamurthy, M. Faloutsos, Denial of Service attacks at the mac layer in wireless ad hoc networks, in: *Proceedings of IEEE MILCOM*, 2002.
- [29] Y. Hu, D. Johnson, A. Perrig, Sead: Secure Efficient Distance vector routing for mobile wireless ad hoc networks, in: *Proceedings of IEEE WMCSA*, 2002.
- [30] B. Awerbuch, D. Holmer, C. Nita-Rotaru, H. Rubens, An on-demand secure routing protocol resilient to byzantine failures, in: *Proceedings of ACM Workshop on Wireless Security (WiSe)*, 2002.
- [31] H. Yang, X. Meng, S. Lu, Self-organized network layer security in mobile ad hoc networks, in: *Proceedings of ACM Workshop on Wireless Security (WiSe)*, 2002.
- [32] Y. Hu, A. Perrig, D. Johnson, Packet leashes: a defense against wormhole attacks in wireless networks, in: *Proceedings of IEEE INFOCOM*, 2002.
- [33] E.M. Royer, C.K. Toh, A review of current routing protocols for ad hoc mobile wireless networks, *IEEE Personal Communications* 6 (1999) 46–55. April.