1-1-2010

Trust-enhanced secure multipath routing for mobile ad hoc networks

Lubaid Ahmed
Ryerson University

TRUST-ENHANCED SECURE MULTIPATH ROUTING FOR MOBILE AD HOC NETWORKS

by

Lubaid Ahmed

M.Sc. Computer Science, NED University of Engineering & Technology, Pakistan, 2002

M.Sc. Applied Physics with Specialize in Electronics, University of Karachi, Pakistan, 1996

B.Sc. University of Karachi, Pakistan, 1993

A thesis
presented to Ryerson University
in partial fulfillment of the
requirements for the degree of

Master of Science

in the Department of
Computer Science

Toronto, Ontario, Canada, 2010

Author's Declaration

I hereby declare that I am the sole author of this thesis.

I authorize Ryerson University to lend this thesis to other institutions or individuals for the purpose of scholarly research.

Lubaid Ahmed

I further authorize Ryerson University to reproduce this thesis by photocopying or by other means, in total or in part, at the request of other institutions or individuals for the purpose of scholarly research.

Lubaid Ahmed

TRUST-ENHANCED SECURE MULTIPATH ROUTING FOR MOBILE AD HOC NETWORKS

A thesis for the degree of

Master of Science, 2010

by
Lubaid Ahmed

Department of Computer Science

Ryerson University

Abstract

Due to recent advances in computing and communication technologies, Mobile Ad hoc Networks (MANETs) are becoming networks of choice for various applications such as emergencies preparedness and response, military and crisis management, and healthcare, to name a few. The main reason for this is that in MANET, information exchange between nodes can happen dynamically without pre-existing fixed network infrastructure with designated centralized access points. However, this privilege also comes with some security drawbacks, especially from a message security viewpoint because the implementation of hard-cryptographic security now becomes a challenging prospect. In this thesis, we improve a recently proposed method of message security in MANET (so called benchmark scheme, also referred to as trust-based multipath DSR routing scheme), by introducing a trust model that makes multi-path routing flexible enough to avoid non-trusted routes that may use brute force attacks to decrypt messages travelling through the network en route to their destinations. Simulation results, coupled with theoretical justification, affirm that the proposed solution is much more secured than the above-mentioned benchmark method and traditional multi-path routing algorithms.

Acknowledgements

It is my honour to express my deep gratitude to people who made this challenge possible for me. I would like to thank my supervisor Dr. Isaac Woungang, for giving me the opportunity to work under him. He offered me such a challenging topic for my thesis, and with his strong support and constant guidance, I finally achieve my goals. He did not only guide me in the course of this thesis, but also provided me an opportunity to be benefited from his vast knowledge. He was always present to help me out and guide me whenever I needed his guidance.

Lastly, and most importantly, I would like to thank my mother Mrs. Lubna Farooqui and also my brothers Saad and Munad. It is their prayers and support that gave me courage to fulfill my goals.

Dedication

To my Mother

I would not be here without her support and strong belief in me.

Table of Contents

List of Tables

List of Figures

List of Abbreviations

MANET	Mobile Ad-Hoc Network
OSPF	Open Shortest Path First
RREP	Route Reply
RREQ	Route Request
DTN	Delay Tolerant Network
DoS	Denial of Service
DSDV	Destination-Sequenced Distance-Vector routing protocol
QoS	Quality Of Service
SRP	Secure Routing Protocol
CONFIDANT	Cooperation Of Nodes - Fairness In Dynamic Ad hoc NeTworks
CORE	A Collaborative Reputation Mechanism to Enforce Node Cooperation in Mobile Ad Hoc Networks
ARAN	Authentication Routing for Ad Hoc Networks
LARS	Locally Aware Reputation System
DSR	Normal Dynamic Source Routing
AODV	Ad hoc On Demand Distance Vector protocol
TB-MDSR	Trust-Based Multipath DSR Routing
ETB-MDSR	Enhanced Trust-Based Multipath DSR Routing
M-DSR	Multipath DSR Routing
MACA	Multiple Access with Collision Avoidance
CSMA	Carrier Sense Multiple Access
FAMA	Floor Acquisition Multiple Access

Chapter 1 : Introduction

1.1 Mobile Ad Hoc Networks

Wireless technology is undoubtedly one of the most promising technologies in the last decade because it can allow users to utilize devices that enable the access to information at any place and time. These needs make wireless networks the best solution for interconnecting devices and people. Wireless networks are composed of devices (nodes) that communicate through media such as radio signals, infrared, or other mediums. These networks can be classified into two categories: infrastructure-based and infrastructure-less wireless networks.

Infrastructure-based wireless network consists of base stations localized in appropriate places. They can provide wireless connectivity to devices within their coverage areas. Examples such networks are cellular networks or wireless local area networks (WLANs).

Infrastructure-less (or ad hoc) wireless networks do not have a pre-established infrastructure. In other words, nodes connect to each other via automatic configuration when they are in the transmission range and are willing to forward data for other. This capability makes wireless ad hoc networks suitable for many applications where one central node may not be convenient, and where minimal configuration and quick deployment is required in emergency situations. In general, wireless ad hoc networks can be classified by their application in mobile ad-hoc networks (commonly referred to as MANETs), wireless sensor networks, and wireless mesh networks.

A MANET is a type of wireless ad hoc network that consists of a dynamic set of wireless mobile routers (with associated hosts), referred to as nodes. These nodes are interconnected using radio links. These radio links are said to be symmetric when two nodes are within each other's transmission range. Otherwise, they are said to be asymmetric. Fig. 1.1 depicts a typical graphical representation of a MANET.

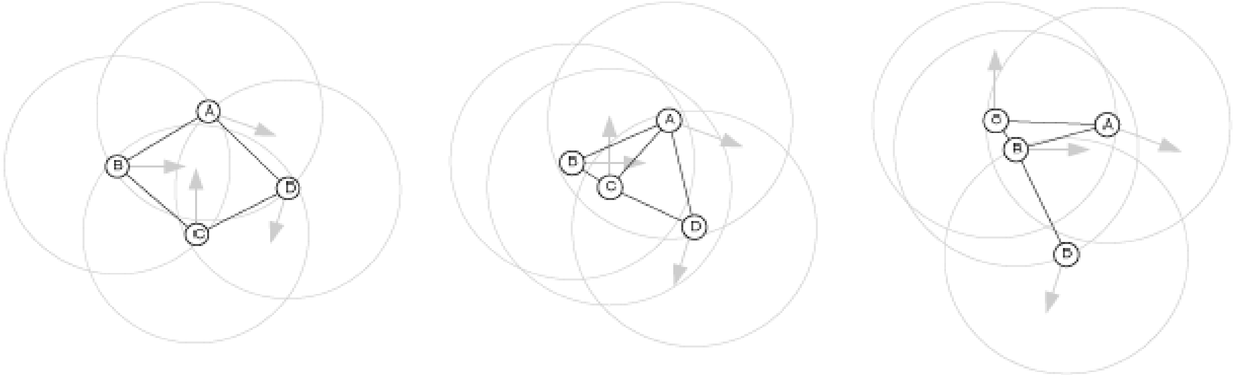

Figure 1.1: Example of a MANET [1].

As shown, this MANET is a connected, undirected graph in which the vertices represent the nodes and the edges represent asymmetric links. These nodes are free to move, creating a dynamic topology in which links can be created and destroyed in rapid succession. In Fig. 1.1, the circles around the nodes represent the reach of a nodes' radio, the arrows are the direction in which the nodes move. The solid lines between the nodes are communication links. Due to routing capabilities of the nodes themselves, the network is self-organizing in the sense that nodes that are not in each other's radio range can communicate using multi-hop routing.

With the advent of new consumer products such as laptops, mobile phones, PDAs, to name a few, that can act as wireless mobile routers, MANETs are becoming increasingly popular and resources are shifting away from wired backbone routers. With this growth, there is a clear demand for effective and secure multi-hop and multi-path routing protocols [2]. Part of the existing solutions to address this deficiency is to consider modifying substantially some traditional routing protocols. The solution proposed in this thesis follows this guideline by using the Dynamic Source Routing (DSR) protocol as underlying multi-path routing mechanism in MANETs.

1.2 Benefit of Multi-Path Routing

Multipath routing can be defined [3] as the property that at any given time, the source node can choose multiple paths to a particular destination by taking advantage of the connectivity redundancy of the network. This concept may be used alternately- i.e. traffic taking one path at a time, or concurrently - i.e. traffic flowing through multiple paths simultaneously.

The dense deployment of nodes in MANETs makes the multipath routing a promising technique to cope with the frequent topological changes and consequently with unreliable communication services. For instance, multipath routing can be used to improve the robustness of data delivery, to balance the traffic load and power consumption among nodes [4], to reduce the end-to-end delay and frequency of route discoveries [5], or to improve the network security [6], just to name a few. In this thesis, the use of multipath routing falls within the latter case.

1.3 Context of Our Study

MANETs are wireless networks that do not require any infrastructure to set up. This makes them ideal for military, rescue and relief operations. Nevertheless, this flexibility and the lack of a centralized server or access point create a message security problem. Indeed, since the nodes are assumed to co-operate to route messages, the identification of misbehaving and non-benevolent nodes is non-trivial and message security cannot be implemented in multi-hop MANETs without a dedicated message encryption strategy. In this context, using traditional key-based encryption techniques would require certification authorities and key distribution centers to trustfully transfer keys between nodes. This will in turn require a centralized or partially distributed authorization, which is difficult to achieve in such an open and improvised environment of ad-hoc networks.

In fact, these are various security issues associated with cooperative routing in multi-hop wireless ad hoc networks such as MANETs [7]. The basic ones are confidentiality, integrity, availability, authentication, access control, and non-repudiation.

- *Confidentiality*: means that certain message information is kept secure from unauthorized party. The information includes the application data that gets sent over the routing protocol, the routing information itself, the network topology and the geographical location. Since, MANETs are generally multi-hop in nature, a message is relayed to all the nodes lying in the path of the route. Therefore, message confidentiality is difficult to achieve and certain cryptographic methods need to be used to implement it.
- *Integrity*: concerns ensuring that the transmitted message and other system assets are modified only by authorized parties during transmission. At the routing level, integrity requires that all nodes in the network follow the correct routing procedures. The main challenge of ensuring integrity is that without centralized infrastructure and powerful computing capabilities, it is difficult to apply existing cryptography and key management schemes.

- *Availability*: refers to the normal service provision in face of all kinds of attacks. This feature requires that services or devices are exempt from denial of service, which is normally done by interruption, network or server overload. Typical examples of denial of service attack are *radio jamming*, in which a misbehaving node transmits radio to interfere with other nodes' communications, and *battery exhaustion*, in which a misbehaving node interacts with a node to deliberately consume its battery energy.
- *Authentication*: This property requires that the communicating entity's identification is recognized and proved before communication starts.
- *Access control*: This property requires restricting resources, services or data to special identities according to their access rights or group membership.
- *Non-repudiation*: means that the origin of a message cannot deny having sent the message. This property is meant to ensure that when data are sent from sender to receiver, the sender cannot deny that it has sent the data and the receiver cannot deny that it has received the data.

Among the above-mentioned security services, authentication is the most complex and important issue in MANETs. Indeed, without knowing exactly which nodes talk among them-selves, it is worthless to protect the message from being read or modified. Once authentication is achieved, confidentiality can be realized by encrypting the session using a certain key material that the communicating parties would have agreed upon, prior to communication. It should be noticed that the above-mentioned security services are often implemented either in an individual basis or as a combination of methods,

According to the various attack means [8], common attacks to routing in MANETs can be classified into two major categories, namely *passive attacks* – where data exchange is done maliciously in the network without disrupting the operation of the communications, and *active attacks* – where the data exchange involves information interruption, fabrication, or modification, when disrupting the normal functionality of the MANET operation. In both cases, attacks commonly occurred at the routing discovery phase [9], the maintenance phase, or the Data forwarding phase [10], by not following the specifications of the targeted routing protocols. Defense mechanisms against these types of attacks that aim at addressing all or some of the above-mentioned security services are still some challenging issues.

1.4 Research Problem

Several routing protocols have been proposed for MANETs. These vary from table-driven protocols such as the Destination-Sequenced Distance Vector (DSDV) [11] - which is based on the classical Bellman-Ford algorithm, to on-demand protocols such as the Dynamic Source Routing (DSR) [12], and the Ad-hoc On Demand Distance Vector (AODV)[13]. These protocols work well in benign environments. But in a network in which malicious nodes might be present, they might cause serious security concerns. Therefore, they have to be modified substantially if they are to be used in a hostile network. Proposing such modifications, for instance, on the DSR protocol, is the problem that we addressed in this thesis. It should be emphasized that our solution targets only the protection against attacks on routing messages.

1.5 Our Approach

In this thesis, we propose a multi-path DSR-based method to securely route messages in MANET using trustworthiness of nodes. By doing so, we aim at addressing the issues underlying *message confidentiality, message integrity* and *access control*.

Our proposed scheme is designed as follows. First, a message is divided into different parts. Second, these message parts are encrypted using one another [14]. Third, these encrypted message parts are routed separately via different paths between a source-destination nodes pair. During this process, an intermediate node is allowed to access different parts of the message on the basis of its trustworthiness. In other words, a more trusted node is allowed to pertain in more paths than a less trusted node and hence to have 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 process. Furthermore, suspected nodes which have high computation power and hence are likely to be more successful in cryptanalysis, can be given less parts to stymie their plans.

Since our scheme is an improvement of the one proposed in [15], we use a probability trust model [16] instead of the soft approach to trust [17] adopted in [15]. A combination of derived trust and reputation is used to estimate the trust value of each node.

1.6 Contributions

The contributions of this thesis are as follows.

- We have re-implemented the message security algorithm introduced in [15] (so-called Trust-Based Multipath Routing scheme).

- We have proposed an enhancement to the above-mentioned Trust-Based Multipath design, which consist in replacing the underlying trust model by a recently targeted probabilistic trust model [16], leading to a message security scheme which is much more secure than the Trust-Based Multipath routing scheme and traditional multi-path algorithms.

The design of the Trust-Based Multipath DSR Routing (denoted as TB-MDSR) scheme and our enhanced version (so-called Enhanced Trust-Based Multi-path Routing (denoted as ETB-MDSR) both consist of a combination of multi-path routing with a soft-encryption methodology [18] and trust management mechanism. In both designs, soft encryption is achieved by using the message itself for encryption [8], eliminating the need of key distribution centers and key transfer.

Contrary to the trust model used in [15], our adopted a probabilistic trust-based model [16] can account for multiple trust values to assess each aspect of a node's behavior. The final trust value assigned to each node is also obtained as a combination of direct interaction with its neighbours and the recommendations from its peers. These particular features make this trust model more suitable for real MANETs applications.

1.7 Thesis Organization

This thesis is composed of the following Chapters.

Chapter 2: Background Research

In this chapter, we discuss previous works on the subject and their limitations. We then discuss the motivations behind our work.

Chapter 3: Trust-Enhanced Secure Multi-Path Routing

The chapter constitute the core of this thesis. We describe our algorithm including a discussion on our trust assignment, message encryption policy and routing strategy and contrast them against the benchmark method in [103]. We also present various lemmas, which provide an insight into the theoretical aspects on which our work is based.

Chapter 4: Performance Evaluation

Validating the proposed enhanced method is of course an essential part of this research work. In this chapter, we describe the simulation setup, scenario and performance parameters and results.

Chapter 5: Conclusion

We conclude our work and present future possible works that can be done to extend the scope of the work carried in this thesis.

Chapter 2 : Background Research

In the past few years, security in MANETs has been a topic of many discussions in the research community. Many works are available in the literature that discusses this problem. Few representative ones are [7], [14]-[27], [28]-[37]. More recently, most attention have been devoted to pragmatic evolution of MANETs referred to as challenged ad hoc networks, by proposing incentive-aware routing schemes aim at enforcing cooperation among nodes when dealing with the routing and forwarding processes [38]. Few representatives such works are available in [39]-[46]. Nonetheless, providing a complete and efficient message security in MANETs is still a challenging issue.

The above-mentioned representative protocols for implementing security in MANETs can be grouped into four main categories, namely, credit-based systems (commonly referred to as payment systems) [19], [39]-[46], reputation-based systems [21], [22], [23], Tit-for-Tat (TFT)-based schemes [47], [48], and cryptography-based systems [24].

2.1 Credit-Based Systems

This type of systems assigns credits to nodes that forward packets. Typically, credits are taken away from nodes that do not cooperate and credits are given to those that participate in packet forwarding. In this capacity, a dedicated secure hardware or third trusted party is responsible for the management of credits.

In general, credit-based schemes can be realized in two different ways: game theoretical schemes and security protocol based schemes. Game theory based schemes deal with non-cooperative communication scenarios using game theoretical approaches

[49]. Security protocol based schemes deal with cryptography as a mean for ensuring the security of the credits [44].

The purpose of credit-based systems is to encourage cooperation within a MANET by providing economic incentives to the benevolent and co-operating nodes. The security provided by these incentive methods systems is generally aimed at promoting better behaviour rather than using any ‘hard’ security methodology. Some of the above-mentioned popular credit-based systems use virtual currency (as credits).

In [19], nuglets is used as virtual currency for charging and paying for server usage. Nodes are allowed to charge for the services that they provide. Typically, an intermediate node may demand to be paid for forwarding a packet to the next node. The payment (in the form of nuglets) is done by the source or the destination node depending on the model used. If the source node pays for the services, the charging model is referred to as Packet Purse Model (PPM). In this model, the source loads up the message with nuglets after estimating the number of nodes lying in the path. The intermediate nodes then acquire some nuglets from the packet based on the service that they provided, i.e. their successful participation in data forwarding. By doing so, packets are relayed to the destination. If the nuglets are finished before the message has reached the destination, then the message is discarded and the process restarts again. Hence, the source node is requested to make a good estimation of the number of nodes lying on the route to the destination, which might be difficult or even impossible in some cases.

To circumvent this difficulty, the authors in [19] proposed another model called Packet Trade Model (PTM). In this model, intermediate nodes are required to buy a packet from the previous node and sell it to the next node on the route. 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 that are sent, the PPM discourages nodes from sending useless data. This help preventing the network from packet flooding attacks. On the other hand, the PTM can lead to an overuse of the network by sending unwanted packets to the destination node. As opposed to PPM, PTM does not require any estimation on the part of the sender.

SeIP [39] is an incentive protocol implemented in a secure module resigning at each node. It focuses on assigning a non-forged stamp on each packet forwarded as the proof of forwarding. Based on this, intermediate nodes are remunerated, while source and destination nodes are charged with appropriate credits. The same charging rate applies for all nodes while the remuneration is given in the form of a reward of the total credits from the source node via the source's secure module. Here, an intermediate node is free to decide about its participation and can do so by not propagating the session request without bearing any punishment. Finally, a pairing-based method [46] involving an identity-based cryptography (IBC) is used for securing the charging and rewarding processes.

The Secure Multi-Layer Credit based Incentive Scheme (SMART) [40] uses a layer concatenation technique, where a layered coin provides virtual electronic credits as incentive to stimulate the cooperation among nodes while effectively restraining selfish behavior on the network layer. Its rewarding and charging mechanisms are implemented via a profit sharing model, where the reward and punishment are controlled by means of a virtual bank that runs credit clearance. Node that have correctly fulfills their packets forwarding tasks should receive some compensation from the virtual bank, in the form of a dividend of the total credits from the source node. Hence, nodes are naturally motivated to participate in the packets forwarding in order to gain as much credits as possible. In SMART, security is enforced by the use of hash functions in the design of PKI-based certificates. Each node has a unique public key certificate assigned by an Offline Security Manager (OSM), and a chain of signatures is constructed when designing the layered chain, which in turns, determines the secure path to be used for packets forwarding.

Similarly, the Express protocol [41] also relies on the accessibility of a dedicated banker node called the Reliable Clearance Centre – RCC, which assigns credits and remuneration to nodes based on reports it has received from the network on nodes' activities. Based on these reports, the RCC - who acts as Credit Manager and Digital Certificate Issuer, judges the cooperativeness of nodes and secures the micro-payment by assigning the appropriate amount of remuneration to each node. In Express, public key-based digital certificates are used as methodology to identify the intermediate nodes to which packets should be forwarded. Appropriate incentives and fines are provided by the RCC such that rational nodes do not prefer to misbehave.

In [42], an Incentive-Compatible Opportunistic Routing (ICOR) is proposed, which advocates the use of incentives - in the form of credits - as a stimulus for packets forwarding, by encouraging each user node to honestly participate in routing operation. When a source node initiates a session with a destination node, any intermediate node involved in the routing path is expected to receive a payment – in the form of real money or transfer of credit - granted by the source node in recognition of the service provided. This payment is not immediate, but rather is accumulated for the duration of the session, and the node is expected to receive the total payment in the entire session. As above, ICOR also relies on the accessibility of a banker node (so-called Routing Decision Maker - RDM) that acts as a credit clearance.

The Coupons scheme [43] focuses on data sharing through opportunistic contact and does not address the issue of incentives in its general form as most schemes do, but from an application scenario perspective. Every node is assigned a unique ID, then shares a coupon as it comes in contact with an immediate surrounding neighbor, building an ordered list of unique IDs appended to a message, which in turn, determines the forwarding path to the destination.

The Fair Incentive Protocol (FIP) [44] also provides incentives for nodes to faithfully forward packets. a Third Trusted Party (called Trusted Credit Clearance Service - TCCS) is used to run the credits clearance service, where credits are considered as virtual currency. In order to be allocated a credit, an intermediate node must have received an authorization from the destination node, in the form of a receipt. The security of messages is enforced by some cryptographic operations involving some secure short signature schemes.

E2-SCAN [45] is a network layered security protocol designed to monitor the routing and packet forwarding activities at each node. It uses a packet a drop detection algorithm mechanism similar to the watchdog technique [32] for node’s monitoring, but with a distinct collaborative monitoring mechanism, and a secret sharing techniques [50] as alternative for trust mechanism to enhance the collaboration among nodes.

2.1.1 Limitations of Credit-Based Systems

Most credit-based systems require tamper resistant hardware for storing credits with the message. Typically, secure hardware or Third Trusted Party is required for the management of credits. If that is not available, a centralized 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 node, thus, is an impractical assumption. Additionally, most credit-based systems suffer from what is known as locality problems, i.e. nodes in different locations would have different chances of earning virtual money, hence, such a model lacks fairness. In most cases, nodes that are present at the edges would have less chances of earning credits as their chances of lying on a route are lesser than the chances of nodes lying in the center.

Instead of providing message security, credit-based systems would actually work towards promoting a healthy environment in MANETs. Since these systems directly provide economic incentives, they are more suitable for applications involving E-commerce. Involving direct monetary incentives makes it more generic target for malevolent agents, hence, credit-based systems may not be suitable candidates for providing message security in 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. These are suitable for MANETs as the nodes act on the basis of a mutual *trust* that the peer nodes would act benevolently. 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.

Reputation-based systems typically focus on the role of individual nodes, which are expected to monitor the neighbouring nodes traffic and to keep track of each others reputation, with the goal to detect and eventually exclude uncooperative nodes from the networks. To this effect, the system assigns each node a reputation value and punishes the nodes with bad reputation values.

Although reputation systems bear some similarities with the credit-based systems, they are not directly economic in nature, though indirectly they may lead to monetary advantages. For instance, a node with better reputation may get an advantage in terms of forwarding its message earlier than a node with lower reputation.

Most of the above-cited representative reputation-based schemes for MANETs [21], [22], [23], [32], [51]-[55], to name a few, attempt to identify misbehaving nodes and isolate them from the network. A good recent survey of attacks and defence mechanisms for reputation-based systems is available in [56]. In reputation-based systems, it is assumed that a set of trusted nodes can be used to detect and assess the misbehaviour of selfish nodes and deny them participation in the network operations. Typically, nodes are motivated to participate as relays in data forwarding because of their fear that if detected, they will be punished. Without delving into cataloguing them, one can classify them into the following categories:

1. Global reputation systems: In these systems, each node knows the reputation value of every other node in the network. This is achieved by exchange indirect reputation message among the network. The reputation value is updated based on

both the local and global reputation information. CONFIDANT [21] and CORE [22] are examples of global reputation systems.

2. Local reputation systems: In these systems, each node only keeps the reputation value of its neighbouring nodes. Instead of distributing reputation value or information periodically, the local reputation systems usually update reputation value based on its own observation, i.e. based on local reputation information only. OCEAN [57] and LARS [58] are popular examples of local reputation systems.

CONFIDANT [21] is a reputation-based scheme that is capable of detecting misleading nodes by means of observation and inform other nodes of this behaviour through reports sent around the network. A monitor is embedded in each node for observations purpose. This scheme also relies on the existence of a trust relationship between nodes, which is based on passive observation of all packets within a one hop neighbourhood. Similarly to CORE [12], CONFIDANT also implements a punishment mechanism that isolates misbehaving nodes by not answering to their requests; the main difference being that CONFIDANT is capable of generating some additional traffic for reputation propagation, with the side effect that the produced overhead may be heavy, causing a burden to the network. In addition, CONFIDANT relies on the assumption that all nodes are initially authenticated before their deployment.

CORE [22] is a reputation-based system that lays stress on network level selfishness. Each node keeps track of other nodes' reputation computed based on information monitored and provided by other nodes. A punishment mechanism is used to isolate misbehaving nodes by not serving their requests. Whenever a neighbor's reputation falls down a predefined threshold, service provision to the misleading node is interrupted. By doing so, there is no advantage for a node to misbehave because any resource utilization will be forbidden. However, CORE is vulnerable in the presence of collisions or directional antennas since the watchdog is not able to properly monitor the neighborhood of a node.

OCEAN [57] is a reputation-based protocol designed on top of DSR, which resides between the network and MAC layers of the protocol stack. In OCEAN, it is assumed that each node knows its neighbours and maintains a reputation value (in the form of a rating) for each one. To do this, a Neighbor-Watch is implemented for monitoring the behaviour of a neighbour node, and it reports to a Route-Ranker, which itself maintains the rating of the neighbour node. It some sense, these two components are used to detect and punish selfish behaviour. In a negative sense, OCEAN is sensitive to parameter settings and does not punish misbehaving nodes as severely as other reputation systems that use full-blown reputation information.

LARS [58] is a reputation-based scheme designed to mitigate misbehavior and enforce cooperation in MANET. LARS relies on a trust model in which the trust in a node is associated with its reputation value. The reputation of a node is obtained from direct observation and there is no exchange of second hand reputation information. Different from global reputation systems, LARS uses a local reputation only, i.e. each node only keeps the reputation values of all its one-hop neighbors. When an uncooperative node is identified, its k-hop neighbors become aware of the misbehavior, where k is a parameter that is adaptive to the security requirement of the network.

2.2.1 Limitations of Reputation-Based Systems

In all reputation systems, each node receives a feedback on what other nodes think of it. This mechanism can be either direct, i.e. based on reputation table broadcasts such as in CONFIDANT [21], or indirectly by observing the positive recommendation about other nodes, such as in CORE [22]. This may lead to a grunge war by the node which receives a negative feedback about itself.

The system proposed in this thesis partially addresses this problem, by providing an on-demand reputation system. The proposed system also discourages using promiscuous modes, and uses active acknowledgement instead of passive acknowledgements and promotes the use of directional antennas to enhance security. This is to ensure that a node's feedback remains hidden from a node unless it makes efforts to snoop on other nodes. This decreases the probability of a grunge war.

Some challenges faced by existing reputation schemes for MANETs are: (1) It is difficult to distinguish which node has sent or has not sent a message because the data forwarding cannot be observed during the store-carry-and-forward process, (2) Effective and efficient propagation of the reputation is still an issue, (3) Full-time monitoring of nodes is not guarantee, (4) Traffic overhead and wrong accusation spreading may occur in such systems, to name a few.

2.3 TFT-Based Systems

Tit-for-Tat (TTF)-based schemes are incentive schemes that use a TFT strategy to reward (resp. punish) good behaved nodes (resp. bad behaved nodes). Typically, a node lowers its service to its neighbour if it detects a bad behaviour of the neighbour and fully cooperates with its neighbour if a good behaviour of this neighbour is detected. In TFT schemes, every node forwards as much traffic for a neighbour as the neighbour forwards for it. Representative TFT-based incentive protocols for MANETs are described in [47], [48], [59], [60], [61]. We here describe the two most recent ones, which are the Incentive-Aware Routing (IAR) [47] and CompactPSH [48].

The Incentive-Aware Routing (IAR) [47] is the first practical TFT-based incentive mechanism for Delay Tolerant Networks (DTNs), thus applicable to MANETs. In this protocol (so-called TFT-for-DTN), TFT uses a DTN routing to optimize the routes when all nodes in a DTN are cooperative, as well as a selfish DTN routing - which allows selfish nodes to optimize their own individual performances while conforming to TFT constraints. Every node uses a kind of OSPF (Open Shortest Path First)-based link state routing to keep track of the information about links in the network. Thus, TFT-for-DTN is designed to restrain selfish behavior on the network layer.

CompactPSH [48] is a TFT-based incentive scheme for Peer-to-peer networks, a-fortiori for MANETs that allows peers to establish both direct and indirect reciprocity by finding intermediate peers for enabling trade and thus capitalizing more resources. This is achieved through private and share history information (via a Bloom filter-based algorithm) while keeping messaging overhead lower than other traditional incentive schemes.

In general, TFT-based schemes face bootstrapping problems or suffer from exploitation.

2.4 Cryptography-Based Systems

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, to name a few. Representative and popular cryptography-based schemes are ARIADNE [24], ARAN [62], EndairALoc [63], SEAD [29], TSAODV [64].

ARIADNE [24] is an example of a cryptographic method based on-demand protocols such as the Dynamic Source Routing (DSR). ARIADNE has been designed to be effective against attacks such as Denial of Service (DoS) attacks in ad-hoc networks. The advantages of ARIADNE 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. ARIADNE uses symmetric key cryptography, authorizes route requests with a MAC and key combination. It also makes use of per-hop hashing techniques.

ARAN (Authentication Routing for Ad-hoc Networks) [62] is an on-demand, ad-hoc routing protocol. It uses certificates to ensure authentication, message integrity, and non-repudiation of routing messages in an ad hoc networking environment. ARAN makes use of cryptographic certificates with the assistance of a certification server and/or certification authority to offer route security. Because of its reliance on logical route metrics and certificates, ARAN is immune to modification, impersonation as well as the fabrication of malicious routing messages. This protocol is ideally utilized in environments where mobile nodes can be pre-certified (ex. On campus) but remain untrusted and where nodes are originally unknown to each other and cannot be pre-certified.

EndairALoc [63] is based on the EndairA routing protocol, which is derived from the ARIADNE protocol. EndairALoc was proposed due to a new active-1-1 attack that both the Secure Routing Protocol (SRP) and ARIADNE [24] could not protect against. In an active-1-1 attack, the attacker compromises one node and owns another node. EndairALoc uses symmetric key encryption and location information to see which nodes have been tampered with. It was established that EndairALoc could not resist a proposed man-in-the-middle attack.

SEAD [29] is a secure Ad-Hoc network routing protocol based on the design of Destination-Sequenced Distance-Vector routing protocol (DSDV). Securing routing using SEAD requires the use of an efficient one-way hash chains for node's authentication purpose. Each node in SEAD at first constructs a one-way hash chain using a one-way hash function, then uses a specific single next element from its hash chain and attaches this element to each routing update. Distribution of authentic hash chain element, neighbor authentication through broadcast mechanisms and inheritance of distance vector protocol does not scale SEAD for large networks in terms of energy efficiency.

TSAODV [64] is an extension of the SAODV protocol that uses a cryptographic-based approach for intrusion detection and a trust-based mechanism, to offer more resilience to attacks from malicious nodes authenticated by the network, while promoting collaboration among cooperative nodes and penalizing selfish nodes.

2.4.1 Limitations of Cryptographic Methods

A key limitation of cryptography-based methods is that they assume an effective key distribution mechanism. In systems that are dynamic such as MANETs, such assumption is not practical. Key distribution is a non-trivial problem in MANETs because 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, a few numbers of algorithms [14] have been proposed that use encryption based on the method parts themselves. These systems, though not as strong against attacks as other cryptographic methods, are flexible.

In this thesis, we implement an encryption technique in conjunction with a trust-based reputation system and DSR multi-path routing to provide a secure routing method for MANETs.

2.5 Motivation and Benefits of Our Proposed Mechanism

2.5.1 Motivation

From the above-mentioned discussion on the various representative credit-based, reputation-based TFT-based and cryptography-based schemes, which have been proposed in the literature for securing routing of messages in MANETs, it had appeared that cryptography-based solutions are much better than other schemes since they can deal with message modification and fabrication attacks. However, these methods cannot be effective against packet dropping attacks and they have the inherent problem of key-distribution associated with them.

On the other hand, stand-alone reputation-based systems are insecure as they are vulnerable to problems of multiple co-operating mischievous nodes. In fact, multiple nodes may collude or cooperate together to compromise the integrity and the “Web-of-trust” type security provided by these types of systems.

Finally, the multi-path routing concept has been traditionally used in wired networks for providing various Quality Of Service (QoS) guarantees [25]. Several works have been done [26] on using multiple paths to secure the message transfer. But most of them are based on just routing different parts of the message using different paths [26]. These algorithms also depend on finding k disjoint paths between the source and destination nodes, where a message is divided into n parts, where n > k [26]. Also, there may not be enough redundancy in the network and a ‘vital node’ may be present in the network, which exists in all valid paths. In such a scenario, these algorithms would not work even if the vital node is a trusted node and can allow access to the complete message. A sample topology of such a case is shown in Fig. 2.1.

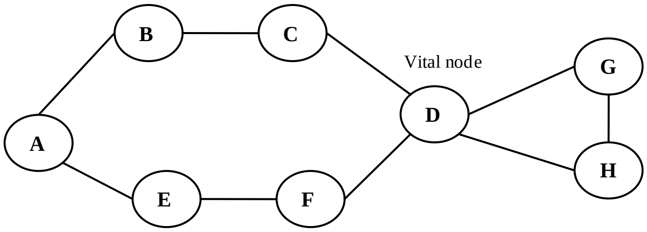

Figure 2.1: No disjoint path to node G [15].

In the topology in Fig. 2.1, even if it is assumed that node D has a non-malicious intent, no secured routing can be realized from node A to node G since no disjoint path can be found from node A to node G. It can be easily inferred that in order for the algorithm to work, the sender (node A) may consider increasing the trust level of node D so that the communication can take place, depending on whether a compromise security is needed to achieve message delivery.

In most multi-path algorithms, it is often trivial to get access to at least some parts of the messages. Neither of these algorithms takes into account the possibility of having some nodes being more malicious than others. For instance, this can happen due to a node's intention or simply because certain nodes may have more computational power than others, which would allow them to perform more successful brute force attacks than other nodes. Consequently, these algorithms can make the message parts available indiscriminately, thus will tend to be vulnerable to brute force attacks, especially in the case a large part of the message had happen to be available to a compromised node. It is important to consider the trustworthiness of the nodes as design feature of such algorithms, 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. Moreover, doing so would also help in limiting the data access to nodes more likely to carry out brute force attacks or other type of attacks if the trust level of such node is low enough.

We believe that a combination of soft message encryption, trust establishment and multi-path routing can be implemented over DSR to provide a pragmatic approach to security in MANETs. This thesis proposes such a combination. We improve a recently proposed method of message security in MANET (so called Trust-based multipath DSR (TB-MDSR)) [15], by introducing a recently proposed trust model [16] that makes multi-path routing flexible enough to avoid non-trusted routes that may use brute force attacks to decrypt messages travelling through the network en route to their destinations. The proposed scheme is composed of a soft-encryption methodology, a trust establishment model, and a DSR-based multipath routing technique. The rationale on using the soft-encryption technique and a novel trust model is described in the sequel.

2.5.2 Message Encryption

Most encryption-based security mechanisms are based on securing the key exchange. In such case, a-priori negotiations are required for key exchanges in dynamic ad-hoc networks. Pre-shared keys may also used in networks which are less dynamic in nature [12]. In [18], [28], the authors introduced a distributed approach to message encryption, in which multiple nodes collaborate to act as a certification authority [18] and each part of the message is involved in encrypting the whole message itself. This way, the problem of key exchange can be avoided since the message parts are themselves used as keys. This type of encryption technique has been adopted in [15]. The same applies in this thesis.

2.5.3 Trust Establishment

The authors of paper [18] stated that all trust establishment protocols depend on a Central Trust Authority. In [6], the authors presented a distributed trust model based on the concept of peer recommendation, where trust is defined as a subjective entity which is transitive in nature under certain predefined conditions. The notion of trust is generalized in the sense that different nodes are given subjective, discrete and dynamic trust values to their peers based on repeated interactions. The authors in [18], [65] presented a trust management model for MANETs, which can provide continuous and normalized trust levels to each node, depending on the benevolent behaviour shown by the nodes. They used the DSR protocol [12] to achieve the routing of messages via the trusted nodes only [66]. Our benchmark method, i.e. the Trust-Based Multi-path DSR (so-called TB-MDSR) method [15] uses a variation of this trust management model.

In this thesis, we use a different trust management model [16]. The key difference between the trust model proposed in [15] and that used in our work stems from the capacity of our proposed trust model to resolve few drawbacks of the trust model in [15] as follows. In the trust model proposed in [15], (1) there is no provision on how the model can be customized according to the pervasive networking environments' different applications, (2) even though the trust value computation is based on direct observation and recommendations, it is unclear whether the trust model design [15] is based on comparisons between realistic pervasive networking environments' security/privacy requirements and the theoretical concepts of pervasive computing, (3) there is no provision of a mechanism to avoid misjudgments due to outdated trust values when dealing with trust value updates, (4) there is no mechanism provided for ensuring that false recommendations are eliminated from the set of recommendations when computing the trust values.

The above-mentioned concerns are directly or indirectly related to the practicality the trust model in [15]. It has been demonstrated [16] that our targeted trust model can be used to resolve these issues thanks to its intrinsic weighting method that can be used to capture the effect of time on the current behavior of nodes, thus rending the trust model dynamic.

Chapter 3 : Trust-Enhanced Secure MultiPath Routing

This chapter constitutes the main contribution of this thesis. Here, we describe our proposed enhanced message security protocol (in the form of a routing strategy) which is composed of three interconnected parts: (1) the chosen trust model along with a trust defined strategy, the message encryption and routing, and finally the algorithm to select secure routes.

3.1 Trust Management Model

3.1.1 Definition of Trust

In this thesis, the same definition of trust used in [15] is reported in our work, i.e. the trust that node A places in a node B is the strength of node A's belief that (1) node B will behave without malicious intent, and (2) the service or interaction that node B provides will satisfy node A's request.

Trust has been used in various ways as a solution for enhancing security in pervasive networks environments, including MANETs [67]. Using trust in such context, nodes can be reliant to run trust computations and guide their behaviors. To this effect, a method should be designed to evaluate the level of trust between nodes, while reflecting the relationships between them. The trust value is thereby used to determine the level at which a node can trust another one. The mechanism that deals with the evaluation, collection, and propagation of trust is referred to as *trust management*.

3.1.2 Trust Management Model

Most trust management schemes are based on the principle of *recommendations* as a means for enhancing the trust evaluation method, assuming that all recommendations are qualified as “honest’ and ‘accurate’ - i.e. recommendations are not false, nor inaccurate - which is not always guarantee [68]. The trust model used in [15] is a variation of the trust models introduced in [17], [65], which uses the above-mentioned principle. In this thesis, we replace that trust model by a recently proposed probabilistic trust model [16] for the reasons explained earlier in Section 2.5.3. This novel trust model, which also follows the aforementioned principle, is described next.

3.1.2.1 Trust Management Scheme Architecture

The architecture of our targeted trust management model is depicted in Fig. 3.1. We refer the reader to paper [16] for its detailed description. Here, we focus only on its main functionalities, by describing the most important trust model’s features that have helped addressing the few concerns raised in Section 2.5.3.

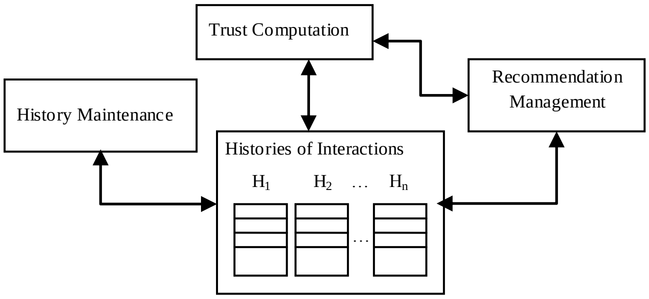

Figure 3.1: Trust Model Architecture [16]

In the architecture shown in Fig. 3.1, the History of Interactions (*HI*) Module stores records on interactions between nodes in a suitable data structure. The history of interactions embedded in a node *A* about a node *B*, denoted as $H_A(B)$, is a list $H_A(B)$ =

{H_1, ... , H_n}, kept at node A, where each entry H_i represents the trust record of a single interaction with node *B*; H_i is defined by the triple $H_i = <e_i, s_i, d_i>$, where e_i is the evaluation of the interaction, s_i is the type of interaction provided and d_i is the time the interaction had happened. During direct or indirect computation, the *HI* Module is maintained and updated by the History Maintenance Module. The functioning of the trust management model follows.

3.1.2.2 Trust Management Scheme Operations

In the architecture depicted in Fig. 3.1, trust computation takes place prior to each interaction occurring between nodes. The Trust Computation (TC) module selects the desired entry in the H_I module, then decides whether to pursue with direct or indirect computation to evaluate trust values. This decision is guided by computing a certain level of confidence that a node has in the trust evaluation of another node. This in turn depends on the amount of contextual information gathered on the interactions. If the confidence level is low, it is concluded that the confidence in the current trust information is inadequate for running a direct trust computation, thus, an indirect trust computation is invoked.

Prior to running the indirect trust computation, more information is required such as the recommendations obtained from peers (recommenders) and the trustworthiness of these recommenders by the TC module prior to accepting and collecting their recommendations. This additional information is gathered through interactions between the TC module and the Recommendation Management module. Once the above judgments on recommenders and their recommendations are completed, the indirect trust computation is activated by the TC module and run with the help of accepted "trusted" recommendations, (from the Recommendation Management module), in addition to records of known experience on history of interactions it has received from the HI module. During this process, an iterative filtering method [16] is invoked as a second judgment level by the TC module to filter out among honest recommenders while ensuring that their recommendations are still as much accurate as possible. In the design of this iterative filtering method, a threshold value between [0, 1] is used as a decision factor to accept or discard a recommendation from a given node.

The major feature of the targeted trust model lies in the way that the TC module computes trust values while reflecting as accurately as possible the node's behaviours. In fact, our trust model considers that the results of recent interactions are more important than those of older interactions because they represent the current behaviours of a node. To accurately reflect the node's behaviours, a weighting method [16] is implemented to assign weights to records of interactions based on when these interactions have occurred.

3.1.2.3 Trust Levels Assignment

In the benchmark scheme's trust model (so-called TB-MDSR) [15], the trust level assigned to a node is a combination of direct interaction with its neighbours and the recommendations from its peers. A node assigns a direct trust level to its neighbour on the basis of acknowledgements received. If the neighbour 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 of a node is then combined with the trust recommendation from its peers and a final trust level is assigned to the node. 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.

In our targeted trust model [16], the direct trust computation is performed when two nodes that are attempting to interact with each other have no experience so far with each other. In this case, the trust computation is based on the direct observation, derived either from personal identification or from the identity information embodied in nodes. If the information gathered so far by a node is not enough for running the direct trust computation, then the indirect trust computation is invoked, which requires that a node use not only its own experience of interaction, but also recommendations from other peers (nodes) to run a trust computation. In this case, the trust level assigned to a node will be a combination of direct interaction with its neighbours and the recommendations from its peers.

It should be noticed that the above-mentioned decision on running an indirect trust computation is built upon computing a level confidence (denoted *Conf*) that a node has in the trust evaluation of another node, which is obtained by determining the variance of a given beta distribution [16]. More precisely, *Conf* is calculated as follows [16]:

$$Conf = 1 - \frac{(n_s + 1)(n_u + 1)}{(n_s + n_u + 2)^2 (n_s + n_u + 3)} \qquad (1)$$

where n_s the number of previous satisfying interactions with B, and n_u the number of unsatisfying interactions with B, have been used as the two parameters in the beta distribution [16] to represent the observations of the interaction between nodes A and B. If *Conf* is low, the confidence in the current trust information is inadequate for running a direct trust computation and more information is needed. The indirect trust computation is then invoked and run with the help of recommendations as already stated.

When a node, say A, computes the trust value of another node, say B, with which it interacts, the notation $T_A(B)$ is used to represent the probability that a satisfying interaction can be provided by *B*. When computing the values of n_s and n_u, it is assumed that the desired type of future interactions is identical to that of previous interactions.

3.1.2.3.1 Direct Trust computation

In this case, the estimated value of $T_A(B)$ is obtained by computing the expected value of a probability distribution function of the beta distribution as follows [16]:

$$T_A(B) = \frac{n_s + 1}{(n_s + 1) + (n_u + 1)} = \frac{n_s + 1}{n_s + n_u + 2} \qquad (2)$$

where the values of *n*s and *n*u are obtained by searching the entries in *HA*(*B*), the history of interactions.

3.1.2.3.2 Indirect Trust Computation

In this case, let *i* be the number of accepted recommendations, and n_s^m and n_u^m be respectively the number of satisfying interactions and that of unsatisfying interactions, calculated based on the recommendation provided by a device D_m. Let n_s^r (resp. n_u^r) be the total number of satisfying interactions (respectively of unsatisfying interactions) in the recommendations, then $T_A(B)$ is estimated as follows [16]:

$$T_A(B) = E(f(x;\alpha,\beta)) = \frac{\alpha}{\alpha+\beta}$$

$$= \frac{n_s + n_s^r + 1}{(n_s + n_s^r + 1) + (n_u + n_u^r + 1)} = \frac{n_s + \sum_{k=1}^{i} n_s^k + 1}{n_s + n_u + \sum_{k=1}^{i} n_s^k + \sum_{j=1}^{i} n_u^j + 2} \qquad (3)$$

In this context, satisfying or unsatisfying interactions are identified in the trust computation process by means of the aforementioned weighted method [16]. More precisely, in our targeted trust model, some weights are assigned to records of interactions based on when they have occurred. More weights are assigned to recent interactions to illustrate their importance compared to older ones. Under the assumption that the current time is t_{cur}, each interaction record is assigned a weight *WT* according to the time it had happened, i.e. [16]

$$H_A^m(B) \Rightarrow WT_m = w^{t_{cur} - d_m} \qquad (4)$$

where *w* is a weighting factor in the interval [0,1], and (3) there exists *n* records of satisfying interactions (resp. *m* records of unsatisfying interactions) in $H_A(B)$, the weighted n_s and n_u are computed as follows [16]:

$$n_s = \sum_{i=1}^{n} WT_i = \sum_{i=1}^{n} w^{t_{cur} - d_i} \qquad (5)$$

$$n_u = \sum_{j=1}^{m} WT_j = \sum_{j=1}^{m} w^{t_{cur} - d_j} \qquad (6)$$

On the other hand, accepted recommendations are obtained using the above-mentioned iterative filtering technique [16]. For the sake of simplicity, a stepwise description of this technique is as follows:

- **Step 1:** Node *A* broadcasts the request for recommendations about node *B*. Let *i* be the number of accepted recommendations collected from honest (trustworthy) recommenders. Recommendations are denoted as pairs (n_s^1, n_u^1),

(n_s^2, n_u^2), … , (n_s^i, n_u^i), where n_s^R (resp. n_u^R) represents the number of satisfying interactions (resp. the number of unsatisfying interactions) in the device *R*'s recommendation.

- **Step 2:** Based on each recommendation (n_s^R, n_u^R), compute a trust value *T_R(B)*. This value represents the trust opinion of node R on node B.
- **Step 3:** Calculate the average trust value $T_{ave}(B)$ using $T_{ave}(B) = \frac{1}{i}\sum_{R=1}^{i} T_R(B)$
- **Step 4***:* Evaluate the inequality $|T_{ave}(B) - T_R(B)| > S$, where *S* is a predefined threshold in the interval [0, 1]. If that inequality holds, then, the recommendation (n_s^R, n_u^R) is false and is filtered out. Otherwise, the recommendation (n_s^R, n_u^R) is considered.
- **Step 5:** Repeat Steps 2 to 4 until all false recommendations are all filtered out.

It should be noticed that the trust recommendations are also piggybacked on DSR routing packets.

3.1.2.3.3 Trust Level Normalization

Using both trust management models in [15] and [16], trust levels that have been assigned to nodes are afterwards normalized to integer values using a standard method. By doing so, each node is given an integer trust value lying between [-1, 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 nodes. A trust level of -1 means 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.

Any normalization technique can be used. In this thesis, we use the following formulae for converting trust values from range (y_{max}, y_{min}) to (x_{max}, x_{min}):

$$x = x_{min} + ((y - y_{min}) * ((x_{max} - x_{min}) / (y_{max} - y_{min}))) \qquad (7)$$

Let's assume that we know the maximum trust value t_{max} and minimum trust value t_{min} in the network and in our case, $x_{max} = 4$ and $x_{min} = -1$. Considering the actual trust value as t and the normalized trust value *as* t_{norm}. Using Equation (8) will yield

$$T_{norm} = -1 + ((t - t_{min}) * (5 / (t_{max} - t_{min}))) \qquad (8)$$

In our case, we used $t_{min} = -80$, $t_{max} = 28$, which were obtained via simulation observations.

3.1.2.3.4 When a New Node Joins the Network

If a new node joins the network, it sends a HELLO packet to its neighbours. The neighbours would assign an initial trust value of 0 to the node. The trustworthiness of the node can be increased, if the node shows benevolent behaviour. Similarly, when a node leaves the network, it would no longer respond to the messages. The neighbour may conclude that the network has lost its connectivity or the node has exited the network. In this scenario, the network would delete the node from its network table and would broadcast this information to the other nodes in the network. The nodes would then delete this table from their route cache.

If on the other hand, the node sends a message to the node and it does respond with an acknowledgement, the node may assume that the node has stopped acting benevolently and may assign it a trust level of -1. Even in this scenario, the node would instruct that the node in question should not be used for message forwarding and hence should be dropped from the network Cache.

3.2 Message Encryption and Routing

In both the TB-MDSR scheme [15] and our enhanced scheme, we use the message encryption method introduced in [14], where a *4n* bits message is divided into 4 parts which are *n* bits long. Let's denote these parts by *a, b, c* and *d*. We define the bit operation *XOR* on a bit vector *k* and *l* such that

$$\text{if } k = \{k_1, k_2, k_3 .... .k_n\} \text{ and } l = \{l_1, l_2, l_3, ... l_n\} \text{ then}$$

$$k\ XOR\ l = \{k_1\ XOR\ l_1, k_2\ XOR\ l_2, k_3\ XOR\ l_3, ..., k_n\ XOR\ l_n\} \qquad (9)$$

We divide a *4n*-bit message into 4 n-bit parts (a, b, c and d) and encrypt them as follows:

$$a' = a\ XOR\ c \qquad (10)$$

$$b' = b\ XOR\ d \qquad (11)$$

$$c' = c\ XOR\ b \qquad (12)$$

$$d' = d\ XOR\ a\ XOR\ b \qquad (13)$$

The parts *a', b', c'* and *d'* are then 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 next.

3.3 Trust Defined Strategy

In both the TB-MDSR scheme [15] and our enhanced scheme, we design a trust defined strategy to secure routing as the policy in which a node with a trust level of say *X*, is given at most *X* parts of the packet to forward. Doing so will limit the possibility of a brute force decryption of the message being transmitted. For instance, if the nodes are assigned four levels of trust (trust values of 1 to 4), excluding no trust (trust level of 0) and complete distrusts (trust level of -1), the message would be divided into 4 parts. Consequently, the following observations apply:

1. 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.
2. 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 is 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 and assuming the remaining byte (out of 256 possibilities through brute force), it can find the entire message.
3. Using a similar process as in Step 2, a node with a trust level of 2 can be sure of finding $2^8 * 2^8$ possible messages.
4. Similarly, a node with a trust level of 1 can be sure of finding $2^8*2^8*2^8$ possible messages.
5. A node with a trust level of 0 is not given any part of the message. These are nodes that are acting as sink and are not forwarding any message or nodes that mangle the messages before forwarding.
6. A node with trust level of -1 is a certified malicious node. All packets received from these nodes are dropped immediately. Measures are taken to limit any promiscuous access of message parts by this node.

It follows that 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 are decrypted as follows [14]:

$$a = b' \text{ XOR } d' \quad (14)$$

$$b = a' \text{ XOR } b' \text{ XOR } c' \text{ XOR } d' \quad (15)$$

$$c = a' \text{ XOR } b' \text{ XOR } d' \quad (16)$$

$$d = a' \text{ XOR } c' \text{ XOR } d' \quad (17)$$

3.4 Implementation Using DSR Protocol

3.4.1 Overview of DSR

In this Section, we provide some details of the operation of the DSR protocol that can allow the reader to understand how DSR operations are embedded within our proposed message security scheme.

The DSR protocol [12] is based on source routing, i.e. the node originating each packet (source node) determines an ordered list of intermediate nodes through which the packet must pass en route to the destination. The source node learns the complete, ordered sequence of network hops necessary to reach the destination, and each packet to be routed carries this list of hops in its header. Thus, intermediate nodes are not required to maintain up-to-date routing information in order to route the packets that they forward.

Typically, the DSR protocol is made of two complement mechanisms, referred to as Route Discovery – in which a node S wishing to send a packet to a destination D obtains a source route to D and Route Maintenance – in which a source node (packet's originator) S detects if the network topology has changed such that it can no longer use its route to the destination D because some of the nodes listed on the route have moved out of range of each other. In this process, route discovery works by flooding a request through the network in a controlled manner, to seek for a route to some destination; route maintenance is used to ensure that broken links to next hops are rapidly detected. Each node is requested to maintain a Route Cache of source routes it has learned or overheard in order to reduce the cost of route discovery. This basic operation of DSR is captured in Fig. 3.2.

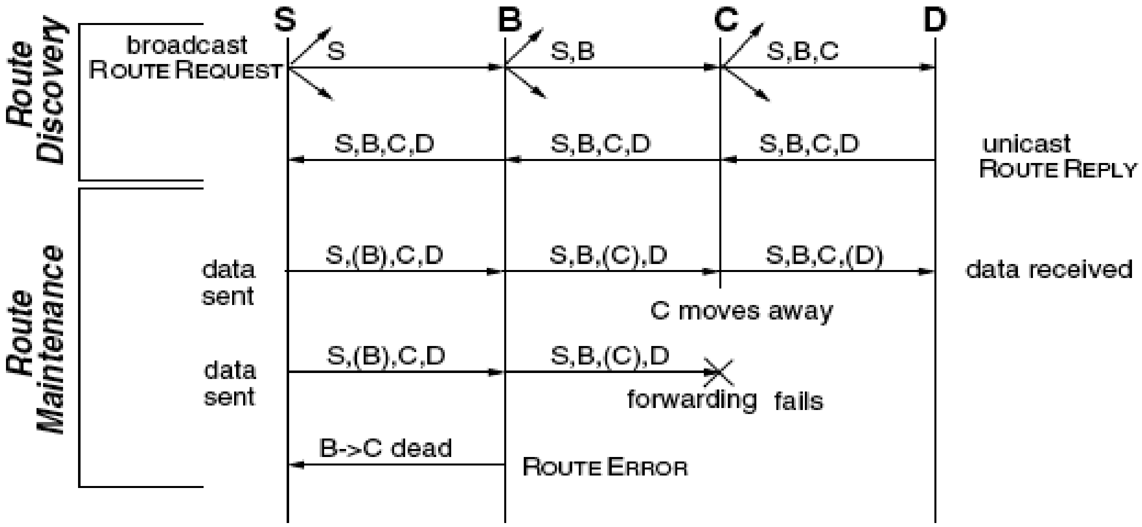

Figure 3.2: Basic Operation of the DSR Protocol [1].

3.4.2 Route Selection

In both the TB-MDSR scheme [15] and our enhanced scheme, DSR was used as underlying routing protocol, based on the following route selection strategy: 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 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. To understand this latest point, let's consider in the network presented in Fig. 3.3.

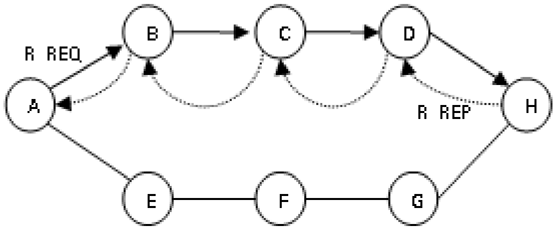

Figure 3.3: DSR routing protocol example

In Fig. 3.3, 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 node *C*, it appends to it a trust value in the packet for *D*. Similarly, *B* appends the trust value of *C*. Node *A* can also explicitly request the trust values of nodes lying in the path from other nodes in the network, by sending a recommendation request. This is explained in detail in [5] and is omitted from here to maintain the brevity of this Section. The algorithm used for selecting routes to securely route the data from source to destination is shown in Fig. 3.4.

```
Arrange the paths $P = \{P_1, P_2, P_3.... P_n\}$ in increasing order of path length
Initialize Count $C_i$ for all nodes = 0
Select the smallest path from P{
    Select next smallest path
    If ( for all selected nodes i $C_i <= T_i$ ){
        if ( four paths selected )
            break out of loop;
        else
            continue;
    }
    if (all paths exhausted )
        wait for another path
    }
if (no paths left)
Print ("It is not possible to route the message securely")
```

Figure 3.4: Algorithm to select secure routes [15].

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 (Fig. 3.4) 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 count. This Step is to ensure 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 numbers 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 achieved by the already selected paths.
4. This process is repeated until secured routes are found.
5. If no secured route is 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 the 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 assumes to have failed and a failure message is displayed.

This 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. For simulation purpose, we have assumed that m is equal to 4. It should be noticed that secured routes can also be found in a more effective way, for instance, 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 our proposed algorithm is computationally effective enough for the desired purpose.

3.4.3 Theoretical Foundation

In our simulation study, in order to compare the above-mentioned algorithm (Fig. 3.4) applied to the TB-MDSR scheme [15] and our proposed enhanced scheme, some security-related performance parameters have been considered, namely the *trust compromise* and the *route selection time*. Here, security was measured on the basis of access violation of the trust defined strategy presented in Section 3.3. In this Section, some evidence (in the form of theoretical proofs) of the behaviour of the proposed algorithms are presented, which are validated through the obtained simulation results presented later in Chapter 4.

The *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.

The *Trust compromise* is measured as the total sum of access violation 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

$$\textit{Trust Compromise} = \Sigma_{s \in S} (N_s - T_s), \textit{ where } N_s > T_s \qquad (18)$$

The total trust compromise is calculated for all the paths selected for routing. Since the normal DSR uses a single path to route a message, only one path is considered in DSR. It is clear that the more the trust compromise for a path set is, the lesser the message security will be. Ideally, the trust compromise of a path should be zero to make sure that only minimal access is given to the peer nodes to a message. The more the trust compromise is, the higher is the probability of a message to be compromised and broken by a malicious agent.

The summing up of individual trust compromises models the cooperation that may be taking place between malicious nodes. For instance, 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 cooperating. Similarly, the higher the aggregated trust compromise is, the higher the chances are for a message to be broken into by the compromising and cooperating malicious agents.

Theorem 1 [15]: *The trust compromise of the selected routes for soft encryption and trust based, multi-path routing is always equal to zero.*

Theorem 2: *All generic multi-path algorithms use a static and equal trust levels for all the nodes present in a network*

Proof: 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. Now, we modify our algorithm slightly and assume to divide the message into n parts, out of which at least m are required to decrypt the message. In addition, if we assume the trust level of each node is constant and equal to 1, i.e. $T_i = 1$. 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. We can then conclude that all the generic and pure multi-path algorithms use a static and equal trust level for all the nodes in a network. Consequently, 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.

Theorem 3: *Two sectors of a network can communicate, even when there is just one connecting node, given the vital node has the highest trust level.*

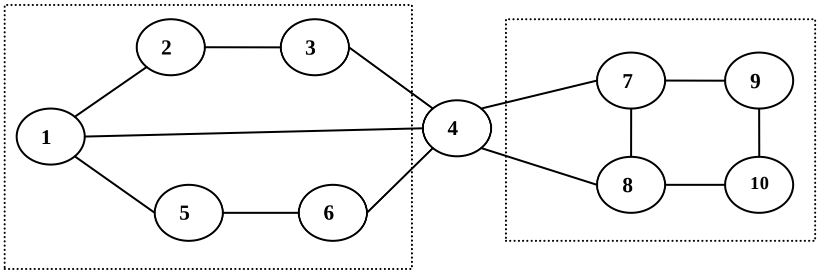

Figure 3.5: Illustration of Theorem 3.

Proof: Let us consider the network given Fig. 3.5. The two rectangles demark the 2 sectors of a network. It should be noted that here node 4 is the *vital* node. Now, if node 1 has to communicate with node 9, the multi-paths in between the nodes consist of a common node 4. If node 4 has a 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), the secured route can then be established.

Chapter 4 : Performance Evaluation

The intent of this Chapter is to evaluate by simulation the Enhanced Trust-Based Multipath DSR scheme (ETB-MDSR) proposed in this thesis, and compare it against (1) our benchmark scheme (referred here as Trust-Based Multipath DSR scheme [15] (so-called TB-MDSR), (2) the Traditional Multipath Routing using 2-disjoint paths (so-called M-DSR), (3) the normal DSR (so-called Normal DSR), using the performance metrics stated in Section 3.4.3, namely the trust compromise and the route selection time. A total of four algorithms are thus implemented and simulated.

The simulation experiments are conducted using the Global Mobile Information System Simulator (GloMoSim) [69], a scalable simulation environment for large wireless and wired communication systems, using a discrete event simulation language called PARSEC [70].

GloMoSim implements all the seven layers of the OSI reference model [71] and can be customizable and assessable at all layers. It also supports various pre-compiled models and protocols at various layers including the DSR algorithm at the network layer, which is used as underlying routing protocol for all message security schemes studied in this thesis.

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, traffic such as TCPLIB, CBR (Constant Bit Rate) and HTTP are supported. It is at the application layer that our proposed soft-encryption method using various message parts is implemented.

Finally, in this thesis, the implementations of the above-mentioned schemes have been realized as a Java framework within the implemented protocols. It is assumed that the trust levels of various nodes are readily 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 is 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.

4.1 Simulation Setup and Scenarios

The number of nodes is varied in various terrain dimensions. To maintain connectivity, radio transmission power is varied accordingly. The MAC protocol used is IEEE 802.11 [72]. Nodes are placed uniformly throughout the terrain and simulations are allowed to run for 600 seconds. The main simulation parameters are captured in Table 4.1.

Parameter	**Setting**
Number of total nodes	10~100
Speed of nodes	5~50 (meters/sec)
Terrain dimension	1000x1000, 1500x1500, 2000 x 2000 (meters)
Traffic Type	CBR
Simulation Time	600 (sec)
Percentage of malicious nodes	10% to 50% of the total nodes in the network

Table 4.1: Simulation Parameters.

Four simulation scenarios are considered:

- Varying the number of nodes under a fixed mobility scenario.
- Varying the maximum speed of nodes using a fixed number of nodes.
- Varying the percentage of malicious nodes using a fixed number of nodes and a fixed mobility scenario.
- Varying the terrain dimension using a fixed number of nodes, a fixed mobility scenario and a fixed percentage of malicious nodes.

4.2 Simulation Results

In this Section, we compare the normal DSR, M-DSR, TB-MDSR, and ETB-MDSR schemes on the basis of the performance metrics stated in Section 3.4.3, i.e. the trust compromise and the route selection time, under the aforementioned simulation scenarios. The results that are obtained validate the theoretical proofs (theorems) provided in Section 3.4.3. Their justification follows.

4.2.1 Varying the number of nodes under one fixed mobility scenario

In this scenario, the terrain dimension is fixed to 1000 m x1000 m and the maximum speed of nodes is fixed (these values could be reset as needed).

4.2.1.1 Effect of the network size on the route selection time

The network size is varied and we study the impact of this variation on the *route selection time* for the four studied algorithms, for a given proportion of malicious nodes present in the network. The results are depicted in Fig. 4.1, Fig. 4.2, Fig. 4.3, Fig. 4.4 and Fig. 4.5.

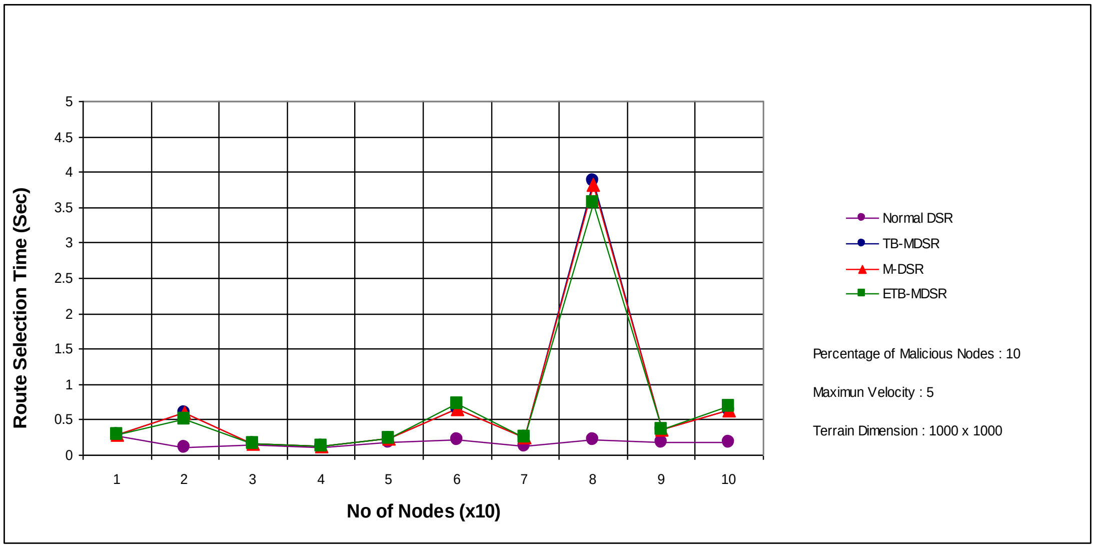

Figure 4.1: Route selection time when varying the network size, under one fixed mobility scenario, with 10% of malicious nodes present in the network.

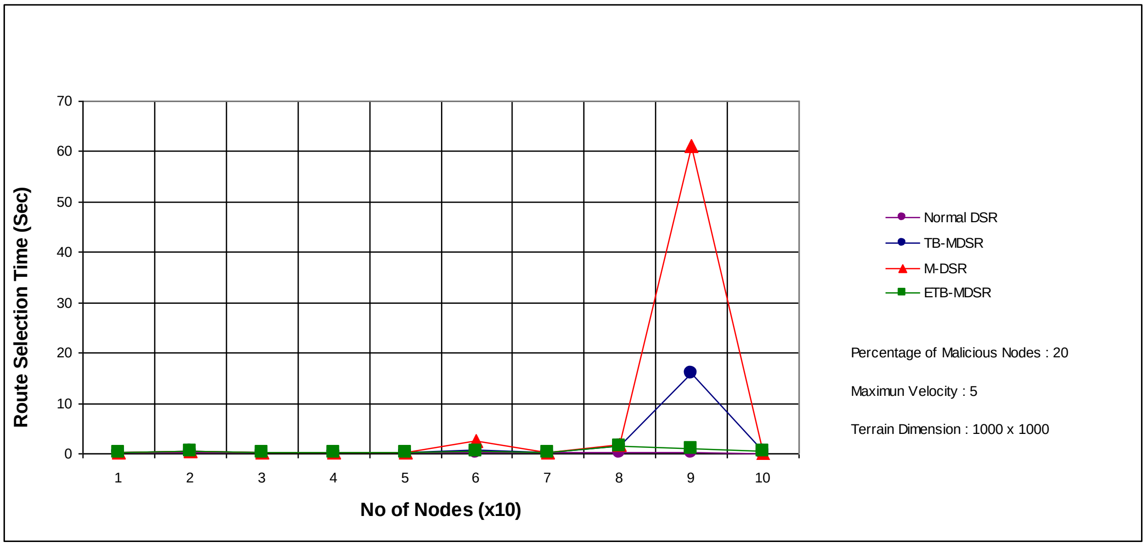

Figure 4.2: Route selection time when varying the network size, under one fixed mobility scenario, with 20% of malicious nodes present in the network.

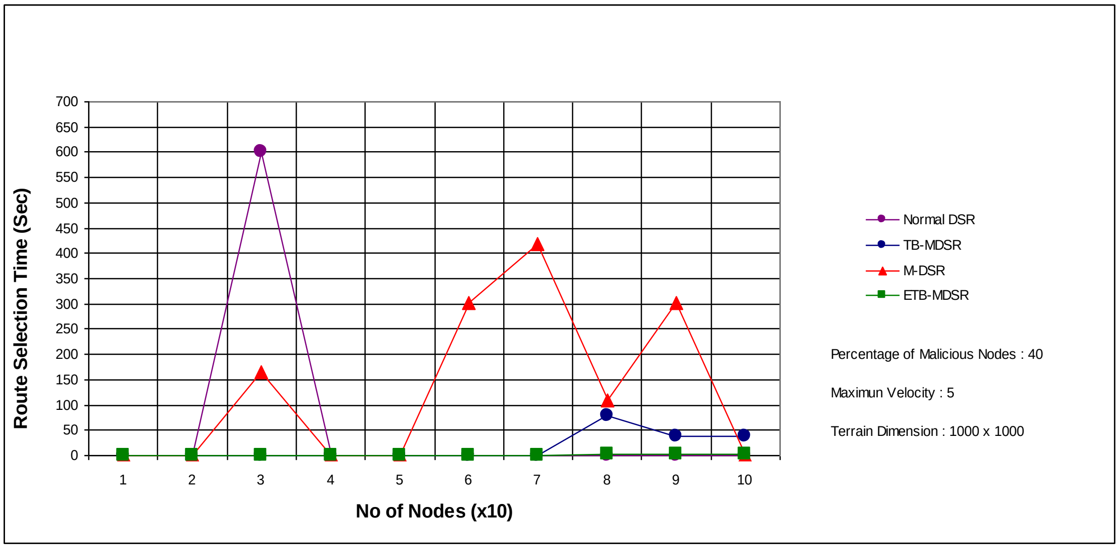

Figure 4.3: Route selection time when varying the network size, under one fixed mobility scenario, with 40% of malicious nodes present in the network.

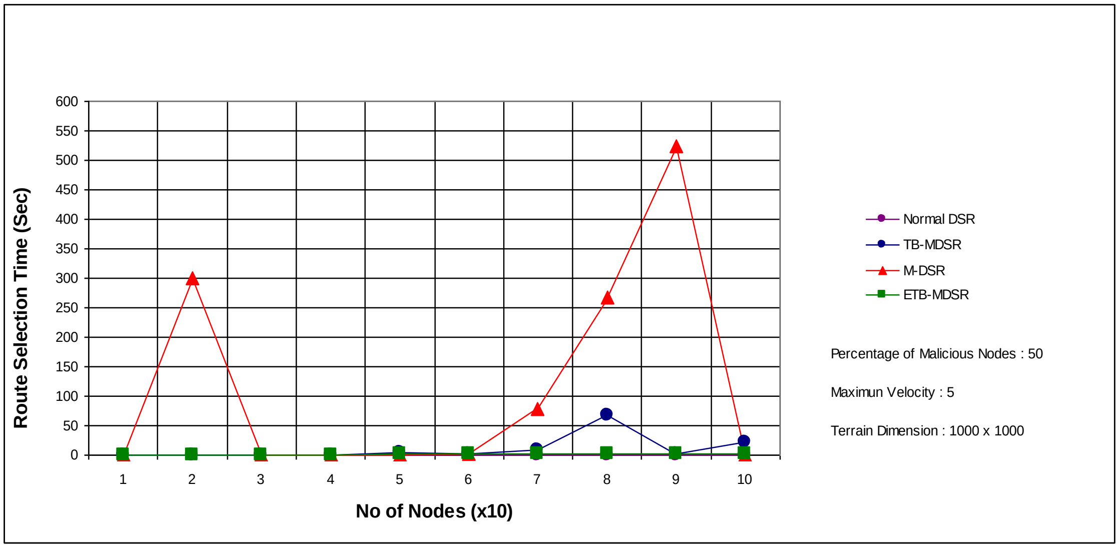

Figure 4.4: Route selection time when varying the network size, under one fixed mobility scenario, with 50% of malicious nodes present in the network.

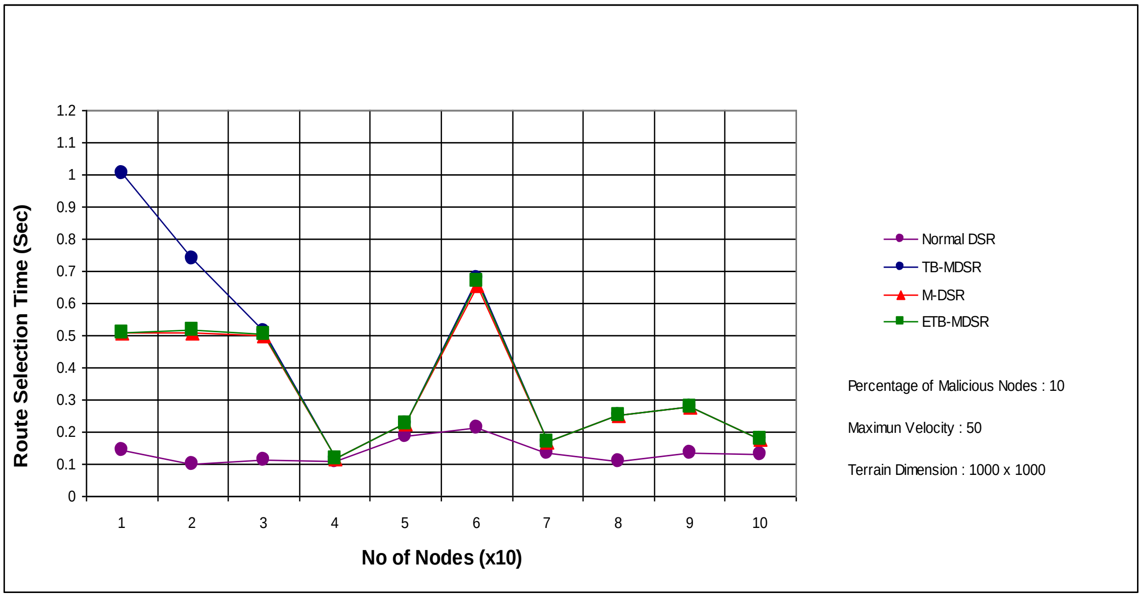

Figure 4.5: Route selection time when varying the network size, under one fixed mobility scenario and a node's maximum speed of 50, with 10% of malicious nodes present in the network.

In Fig. 4.1, Fig. 4.2, Fig. 4.3 and Fig. 4.4, it can be observed that for the class of schemes for which no security constraint apply (i.e. normal DSR and Disjoint Multipath Routing (M-DSR)), the normal DSR always yields the minimum route selection time compared to M-DSR. This can be justified by the fact that the normal DSR selects the first path it receives as the path set for routing whereas the M-DSR scheme may have to wait for a longer period of time before two candidate paths become available for selection as routing path set. A node may even end up not receiving a disjoint path at all in case a 'vital' node exists or if there is not enough redundancy in the network.

For the case of schemes dealing with security constraints (i.e. the Trust-Based Multipath DSR (TB-MDSR) and our Enhanced Trust-based Multipath DSR (ETB-

MDSR)), it can be observed that ETB-MDSR generally takes less time in finding the path set for routing compared to TB-MDSR. This difference in time is even more pronounced when the percentage of malicious nodes in the network increases. This might be justified by the fact that in the trust model employed in ETB-MDSR, a node is not only capable of judging the trustworthiness of another node it interacts with (hence can detect malicious nodes), but is also able to make a better use of the received recommendations from its peers while selecting appropriately these peers. These are achieved respectively through the intrinsic iterative filtering technique and the weighting methods that are used in the trust model in ETB-MDSR. As a result, our targeted trust model (used in ETB-MDSR) reacts better in terms of node's behaviour compared to the trust model used in TB-MDSR, especially when a node has little or enough experience of interactions with other nodes, and when changes dynamically occur in the proportion of malicious nodes (as well as the proportion of nodes) in the network.

Independent of the proportion of malicious nodes in the network, the TB-MDSR scheme would tend to generally take the longest time in route selection compared to the M-DSR scheme (as shown in Fig. 4.5) because the TB-MDSR scheme would require trusted paths, which may take a longer time to come. But this is not always true as illustrated in Fig. 4.1, Fig. 4.2, Fig. 4.3 and Fig. 4.4, where it can be observed that the route selection time of the M-DSR scheme is larger than that of the TB-MDSR scheme, and is even equal or close to that of the normal DSR in some cases. This might be justified by the fact that in the simulated scenarios, most of the nodes (or all the nodes) of the path received first in the TB-MDSR scheme were trusted nodes, which annihilates the time that should have been taken to find these nodes.

Finally, in Fig. 4.1, Fig. 4.2, Fig. 4.3, Fig. 4.4 and Fig. 4.5, it can be observed that the route selection time of the normal DSR is minimal compared to that of all other schemes. Moreover, there are few cases where the route selection times of the ETB-MSDR and normal DSR schemes are equal. This might be due to the fact that in those cases, all the nodes in the path received first in the ETB-DSR scheme have already been qualified as trusted by the intrinsic trust model, which has this capability even when changes dynamically occurred in the proportion of malicious nodes in the network. This same path has then been used as selected routing path by the normal DSR scheme.

4.2.1.2 Effect of the network size on the trust compromise

The network size is varied and we study the impact of this variation on the *trust compromise* for the studied four algorithms, for a given proportion of malicious nodes present in the network. The results are depicted in Fig. 4.6, Fig. 4.7, and Fig. 4.8.

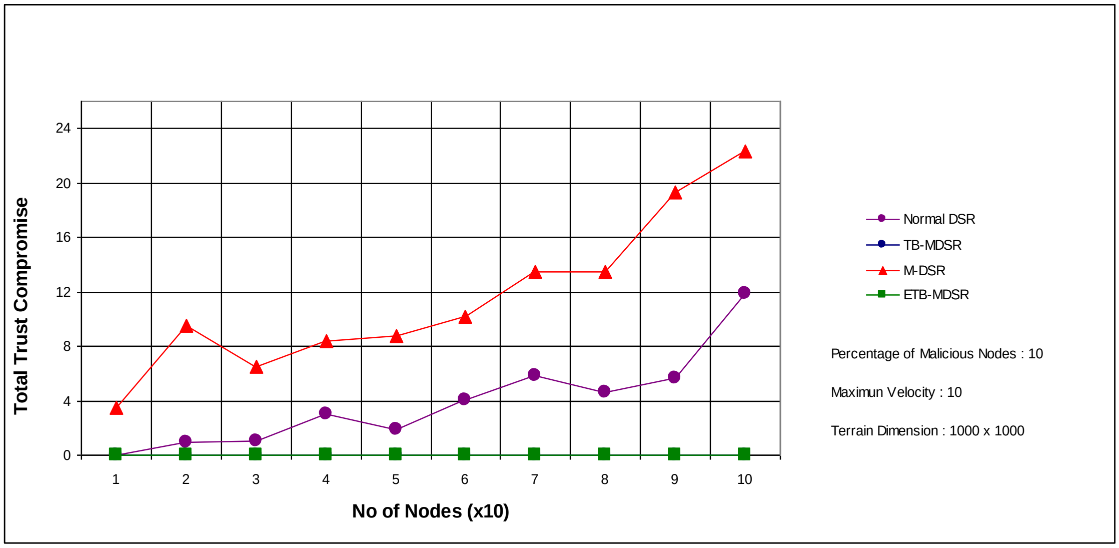

Figure 4.6: Trust compromise when varying the network size, under one fixed mobility scenario, with 10% of malicious nodes present in the network.

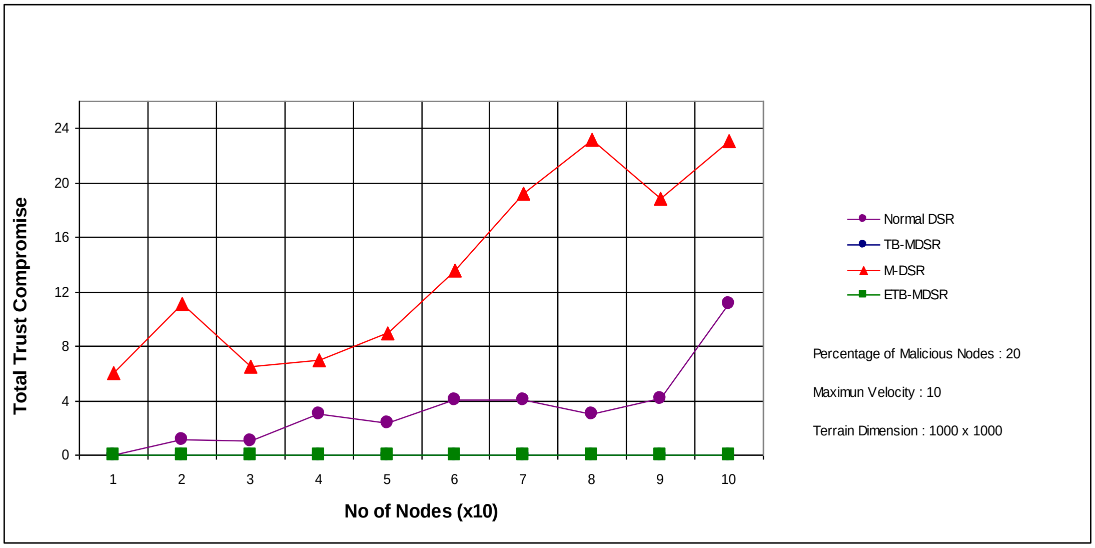

Figure 4.7: Trust compromise when varying the network size, under one fixed mobility scenario, with 20% of malicious nodes present in the network.

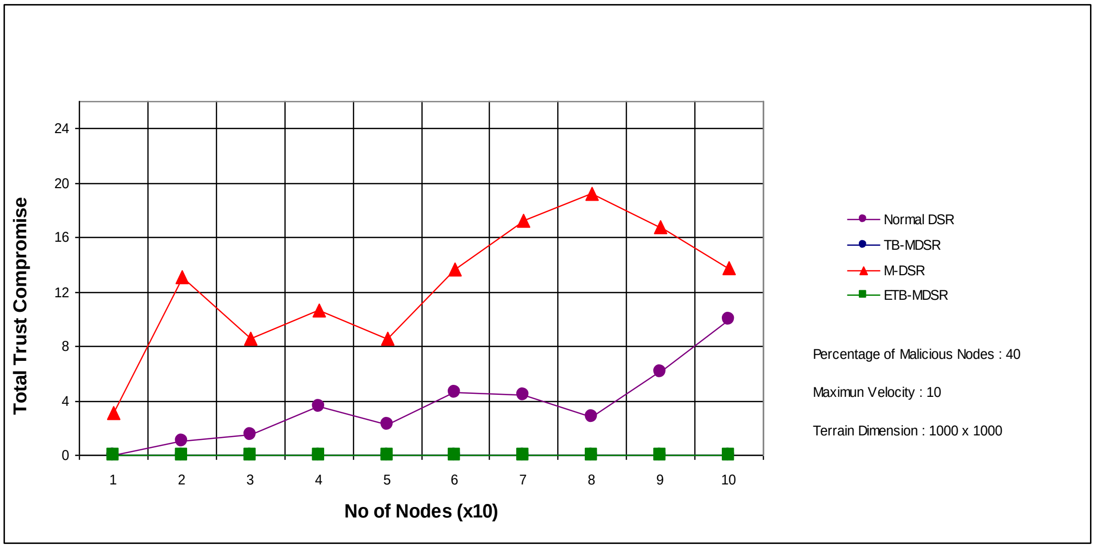

Figure 4.8: Trust compromise when varying the network size, under one fixed mobility scenario, with 40% of malicious nodes present in the network.

In Fig. 4.6, Fig. 4.7, and Fig. 4.8, it is observed that independent of the proportion of malicious nodes in the network, both the TB-MDSR and ETB-MDSR schemes have a trust compromise of 0 in all cases. This can be justified by the fact that in both schemes, the routing path is selected in such a way that no node in the path receives more parts of a message than its trust level.

Since M-DSR sends a message along different disjoint paths, its trust compromise will be equal to 0 when all the involved nodes have a trust level greater than or equal to 2. But since that may not necessarily be the case, the trust compromise for the M-DSR scheme varies between 3 and 23 in the results shown in Fig. 4.6, between 6 and 24 in the results shown Fig. 4.7, and between 3 and 14 in the results shown in Fig. 4.8.

The normal DSR scheme routes the message using the first path it gets, without any security constraint. Thus, its message security is minimal compared to all other algorithms. The trust compromise for the normal DSR varies between 0 and 12 in the results shown in Fig. 4.6 and Fig. 4.7, and between 0 and 10 in the results in Fig. 4.8.

4.2.2 Varying the maximum speed of nodes with one fixed number of nodes

In this scenario, the number of nodes is fixed to 100, and the terrain dimension is fixed to 1000 m x 1000 m (these values could be reset as needed).

4.2.2.1 Effect of the maximum speed of nodes on the route selection time

The maximum speed of nodes is varied and we study the impact of this variation on the *route selection time* for the studied four algorithms, for a given proportion of malicious nodes in the network. The results are depicted in Fig. 4.9, Fig. 4.10, Fig. 4.11, and Fig. 4.12.

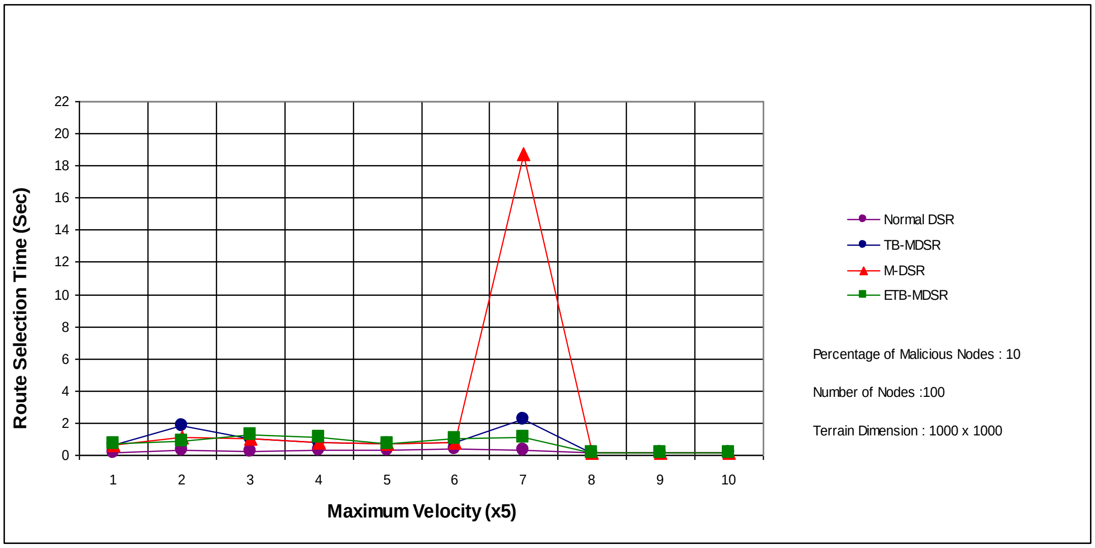

Figure 4.9: Route selection time when varying the maximum speed of nodes, using one fixed number of nodes, with 10% of malicious nodes present in the network.

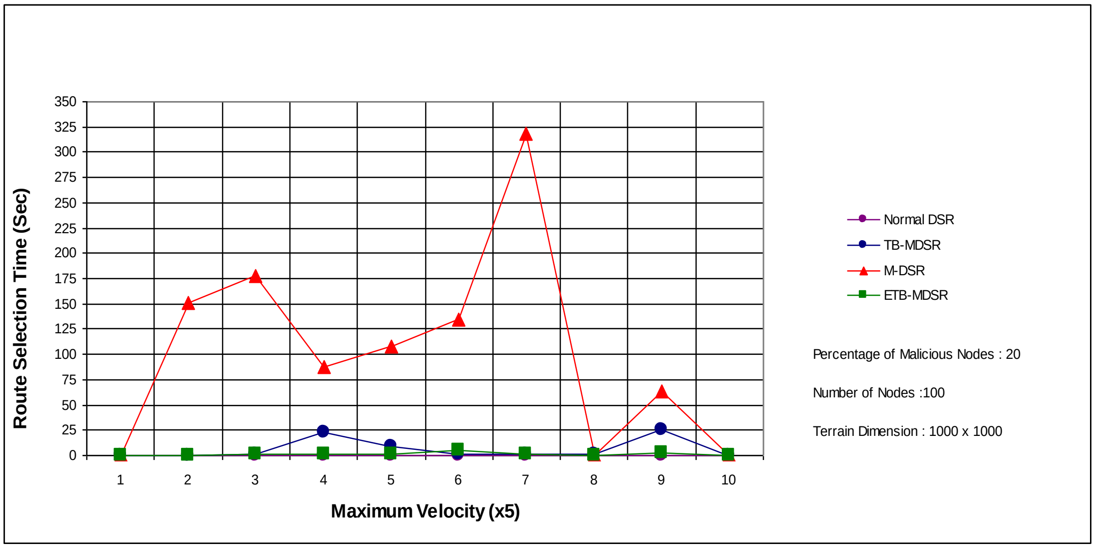

Figure 4.10: Route selection time when varying the maximum speed of nodes, using one fixed number of nodes, with 20% of malicious nodes present in the network.

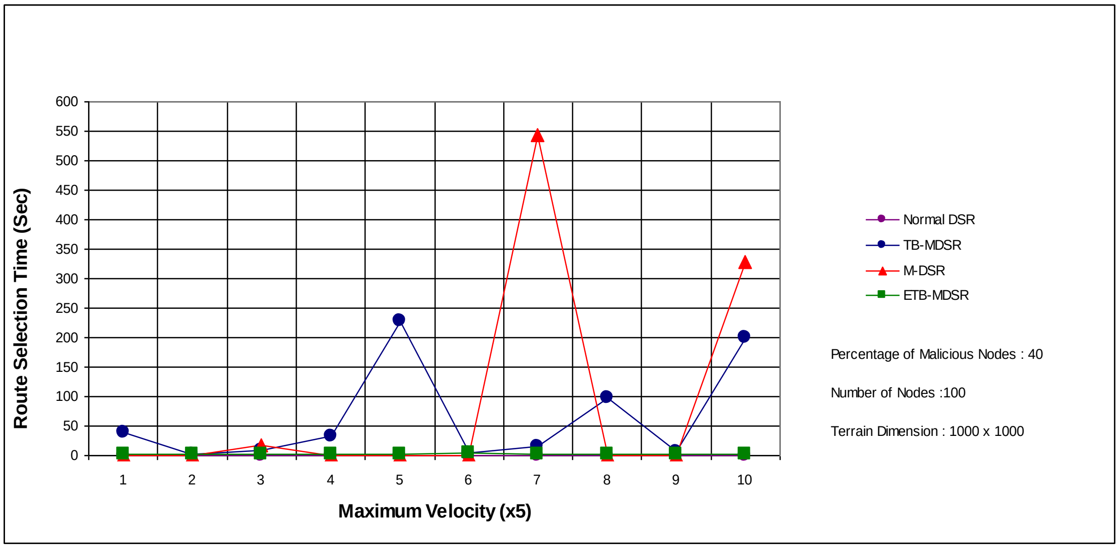

Figure 4.11: Route selection time when varying the maximum speed of nodes, using one fixed number of nodes, with 40% of malicious nodes present in the network.

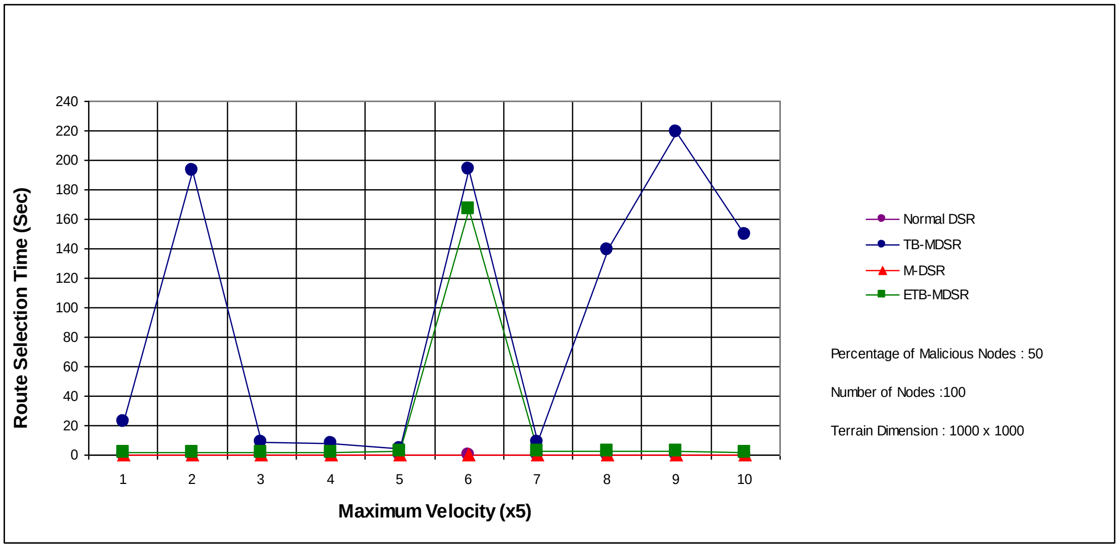

Figure 4.12: Route selection time when varying the maximum speed of nodes, using one fixed number of nodes, with 50% of malicious nodes present in the network.

In Fig. 4.9, Fig. 4.10, Fig. 4.11, and Fig. 4.12, it is observed that independent of the maximum speed of nodes, the normal DSR always yields the minimum route selection time compared to all other schemes. The reason might be that the normal DSR uses the first path it receives as the routing selection path whereas other schemes required an additional procedure for the selection of disjoint paths for routing the message parts (case of M-DSR), or for the selection of trusted paths to the same (case of TB-MDSR and ETB-MDSR), which may take longer time to be realized or abort.

It is also found that in most cases, the ETB-MDSR scheme yields a better route selection time compared to the TB-MDSR scheme. In few cases where this is not true (for instance, in Fig. 4.9 when the number of nodes is in the range [15, 30]), the difference between the route selection times of the ETB-MDSR and TB-MDSR schemes is very small (less than 1%). In our simulations, we have observed that in these cases, the computation of the trust value of a node using both trust models may have occurred in a context where the targeted node has enough experience of interactions with its peers, and the recommendations from these peers are accurate enough (meaning that the number of malicious nodes is kept minimal) so as to have facilitated a quick choice of the trusted routing path in the case of the TB-MDSR scheme compared to the ETB-MDSR scheme. In is also observed that when the proportion of malicious nodes in the network increases, there is less experience of interactions among existing nodes, and thus, the above-mentioned cases become rare.

It is also observed that the trust-based schemes TB-MDSR and ETB-MDSR can alternatively consume more time in finding the routing path set than the M-DSR scheme (case of Fig. 4.12), or can use less or equal amount of time to find the routing path set than the M-DSR (case of Fig. 4.10). The former may be due to the fact that the trust-based schemes would generally require longer time to find the trusted routing path. The later may be due to the fact that most (or all) of the nodes in the path received first in the trust-based schemes were already trusted, reducing or eliminating the time required for choosing the trusted routing path.

4.2.2.2 Effect of the maximum speed of nodes on the trust compromise

The maximum speed of nodes is varied and we study the impact of this variation on the *trust compromise* for the studied four algorithms, for a given proportion of malicious nodes in the network. The results are depicted in Fig. 4.13, Fig. 4.14, Fig. 4.15 and Fig. 4.16.

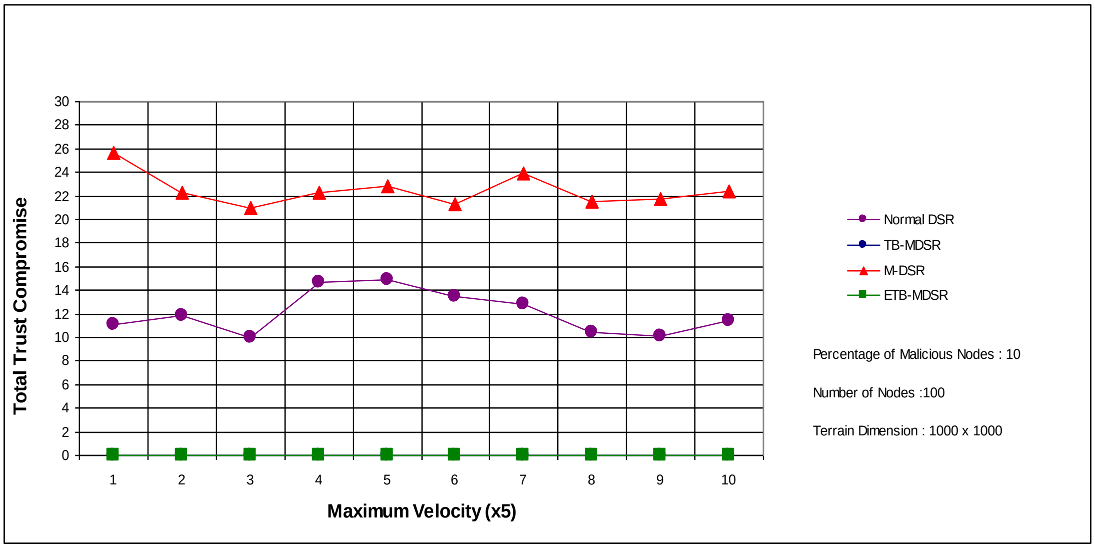

Figure 4.13: Trust compromise when varying the maximum speed of nodes, using one fixed number of nodes, with 10% of malicious nodes present in the network.

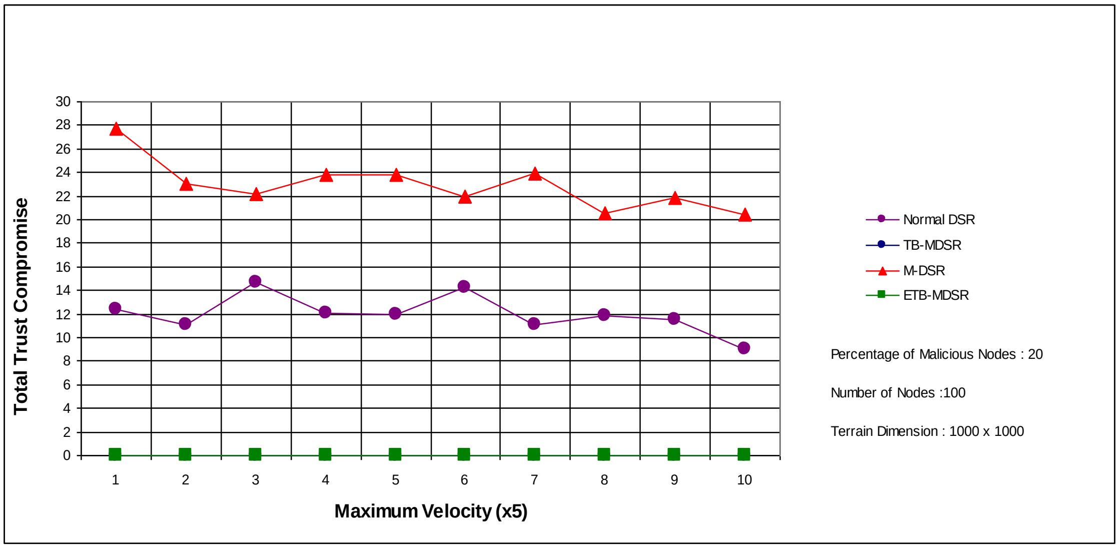

Figure 4.14: Trust compromise when varying the maximum speed of nodes, using one fixed number of nodes, with 20% of malicious nodes present in the network.

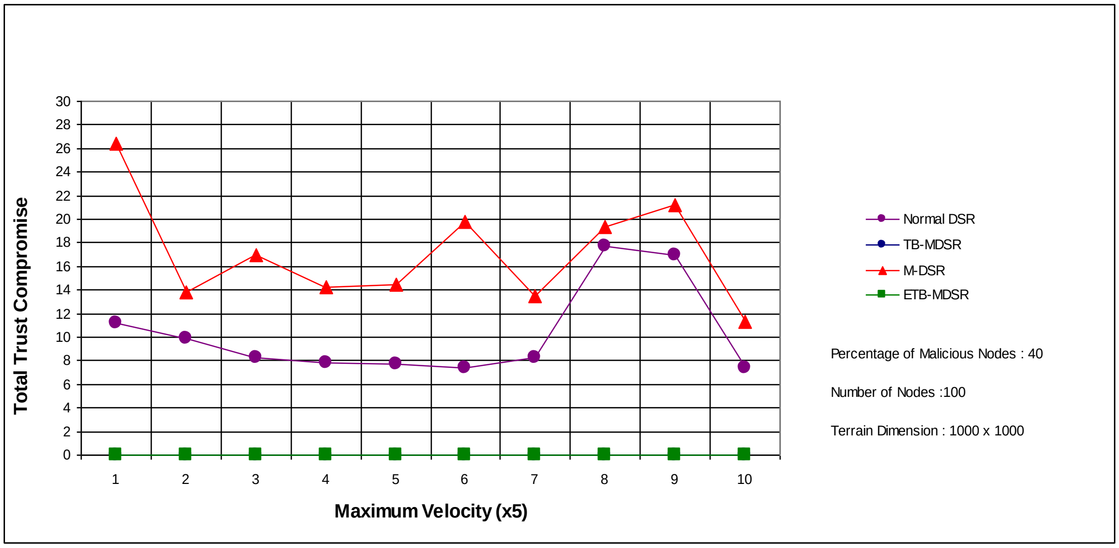

Figure 4.15: Trust compromise when varying the maximum speed of nodes, using one fixed number of nodes, with 40% of malicious nodes present in the network.

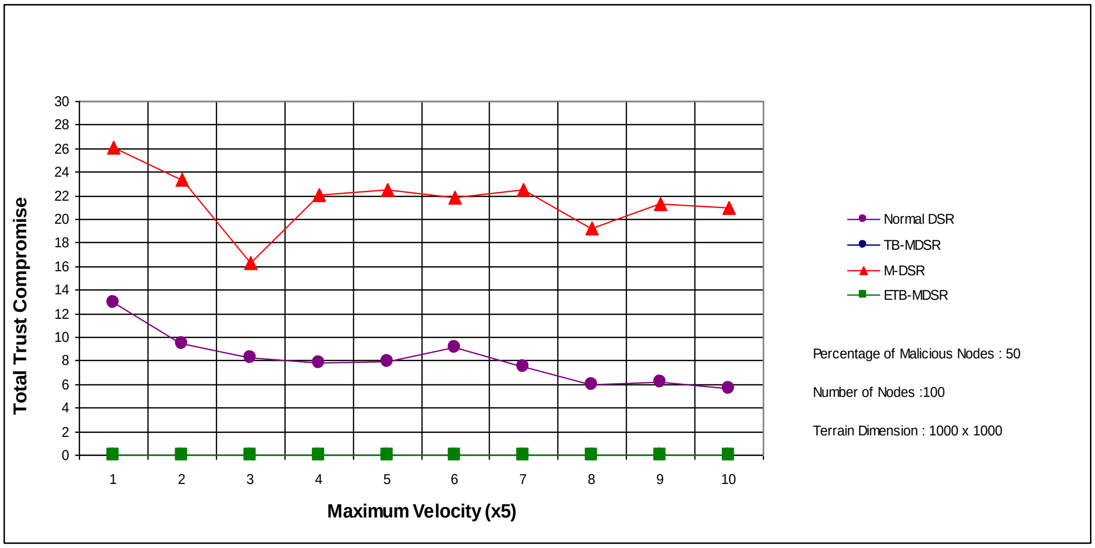

Figure 4.16: Trust compromise when varying the maximum speed of nodes, using one fixed number of nodes, with 50% of malicious nodes present in the network.

In Fig. 4.13, Fig. 4.14, Fig. 4.15 and Fig. 4.16, it can be observed that independent of the chosen maximum speed of nodes in the network, both the TB-MDSR and ETB-MDSR schemes have a trust compromise of 0 in all cases. This result is in agreement with Theorem 1 stated in Section 3.4.3.

In Fig. 4.13, Fig. 4.14, Fig. 4.15 and Fig. 4.16, it is also observed that independent of the chosen maximum speed of nodes in the network, the normal DSR scheme yields a better message security compared to the M-DSR scheme. In fact, since trust values are assumed to be assigned randomly to the nodes in the network, including nodes in selected routing path (only one path for the normal DSR scheme and 2-disjoint paths for the M-DSR scheme), it is clear that both schemes will yield a minimal message security compared to the TB-MDSR and the ETB-DSR schemes. In the studied scenario, for a given maximum speed of nodes, and independent of the percentage of malicious nodes in the network, the total trust compromise using the normal DSR is less than the total trust compromise when using the M-DSR scheme, in all cases.

4.2.3 Varying the percentage of malicious nodes with one fixed number of nodes and one fixed mobility scenario

In this scenario, the mobility scenario is fixed, the number of nodes is fixed to 100, and the terrain dimension is fixed to 1000 m x 1000 m.

4.2.3.1 Effect of the percentage of malicious nodes on the route selection time

The percentage of malicious nodes is varied and we study the impact of this variation on the *route selection time* for the studied four algorithms, for a given maximum speed of nodes. The results are depicted in Fig. 4.17, Fig. 4.18 and Fig. 4.19.

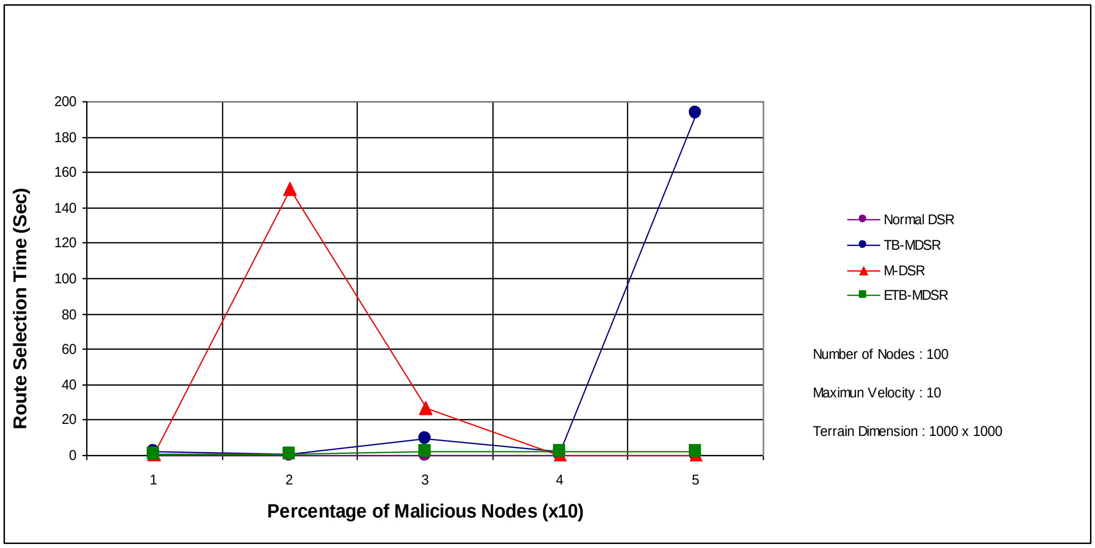

Figure 4.17: Route selection time when varying the percentage of malicious nodes in the network, under one fixed mobility scenario, using one fixed number of nodes and a node's maximum speed of 10.

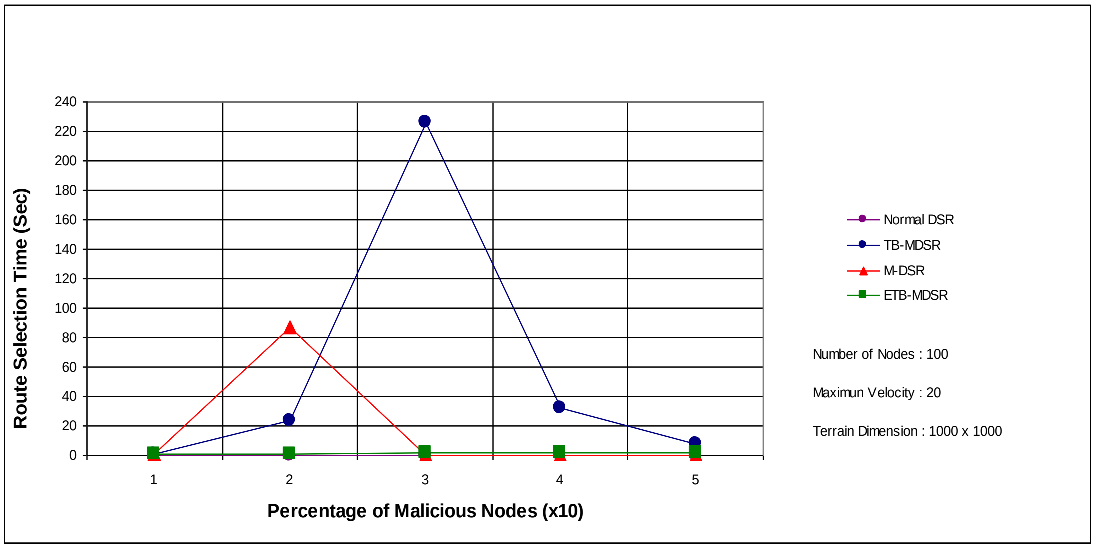

Figure 4.18: Route selection time when varying the percentage of malicious nodes in the network, under one fixed mobility scenario, using one fixed number of nodes and a node's maximum speed of 20.

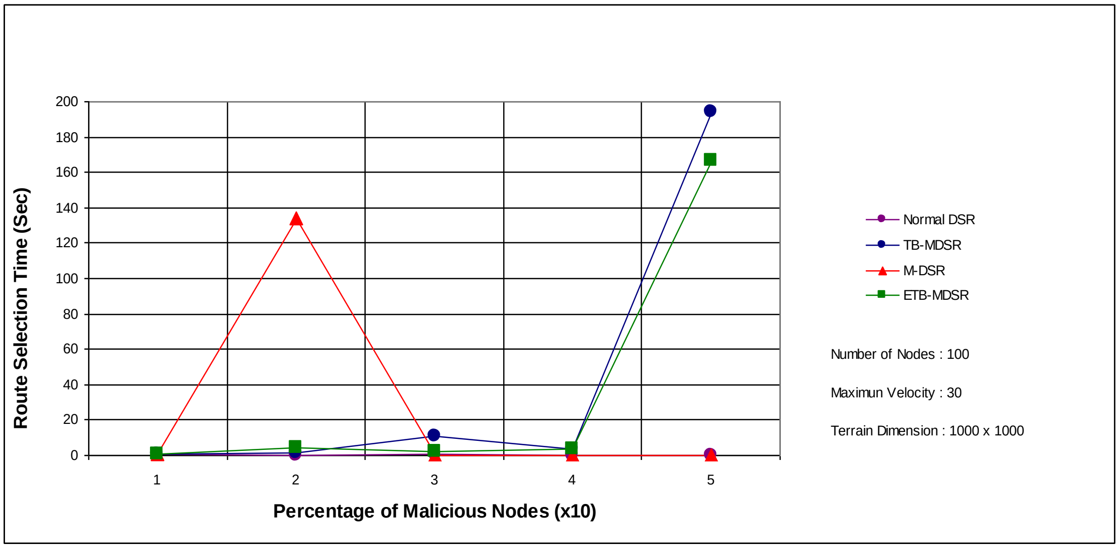

Figure 4.19: Route selection time when varying the percentage of malicious nodes in the network, under one fixed mobility scenario, using one fixed number of nodes and a node's maximum speed of 30.

In Fig. 4.17, Fig. 4.18 and Fig. 4.19, it is observed that independent of the percentage of malicious nodes in the network, the normal DSR takes the minimum time to select the routing path set compared to all other schemes, which is similar to the observation made earlier in previous scenarios.

In Fig. 4.17, Fig. 4.18 and Fig. 4.19, it is also observed that in all cases studied, the ETB-MDSR scheme takes less time to select the trusted routing path (to route the message parts) compared to the TB-MDSR scheme when the percentage of malicious nodes increases in the network. Also, in few cases, the route selection times of the ETB-MDSR and normal DSR schemes are equal or very close. These might be justified by the fact that the weighting method of the targeted trust model has the capability to adjust the trust computation process, leading to a quick choice of trusted nodes in the selected routing path.

4.2.3.2 Effect of the percentage of malicious nodes on the trust compromise

The percentage of malicious nodes is varied and we study the impact of this variation on the *trust compromise* for the studied four algorithms, for a given maximum speed of nodes. The results are depicted in Fig. 4.20, Fig. 4.21 and Fig. 4.22.

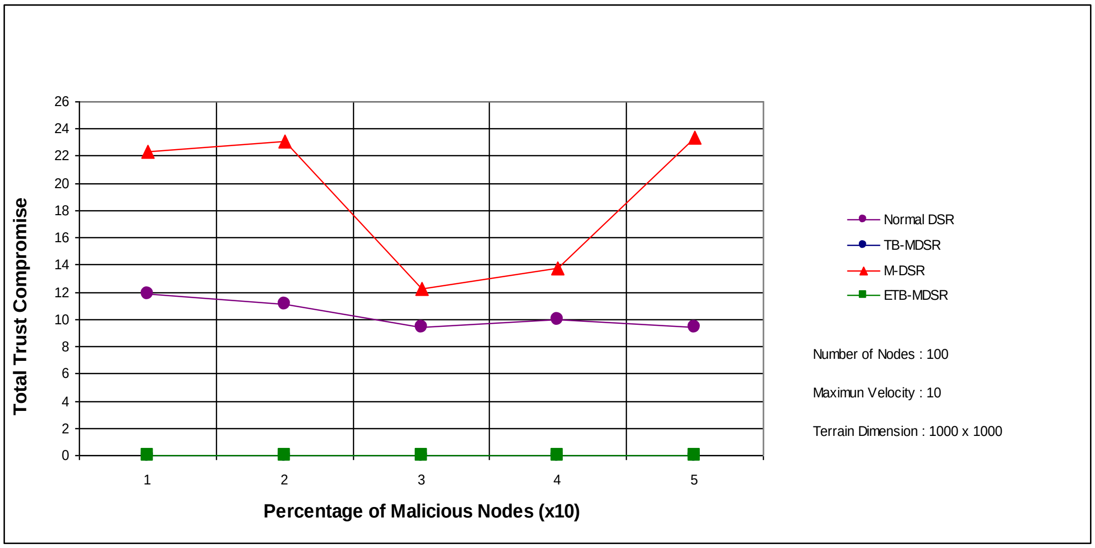

Figure 4.20: Trust compromise when varying the percentage of malicious nodes in the network, under one fixed mobility scenario, using one fixed number of nodes and a node's maximum speed of 10.

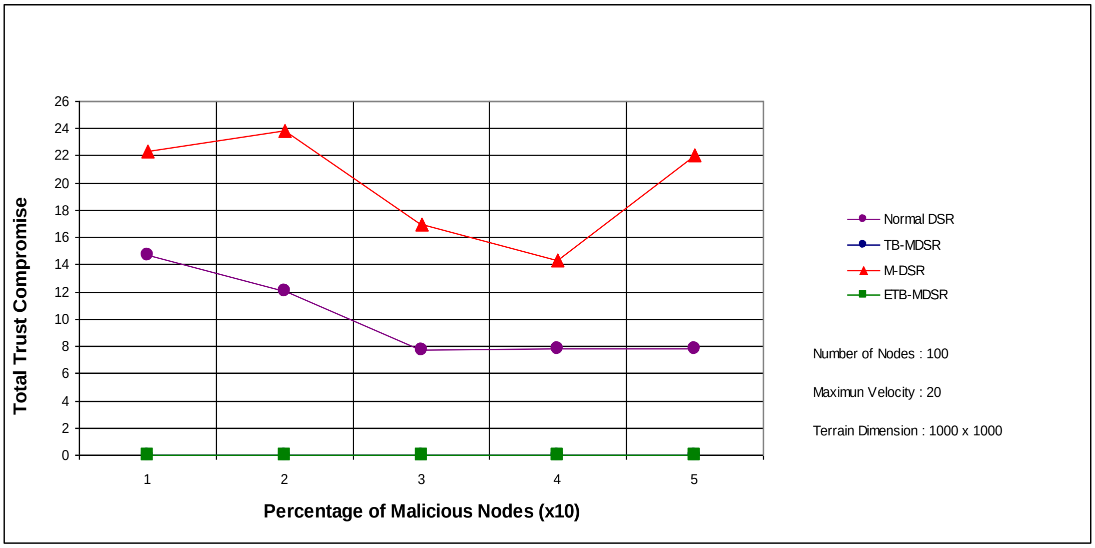

Figure 4.21: Trust compromise when varying the percentage of malicious nodes in the network, under one fixed mobility scenario, using one fixed number of nodes and a node's maximum speed of 20.

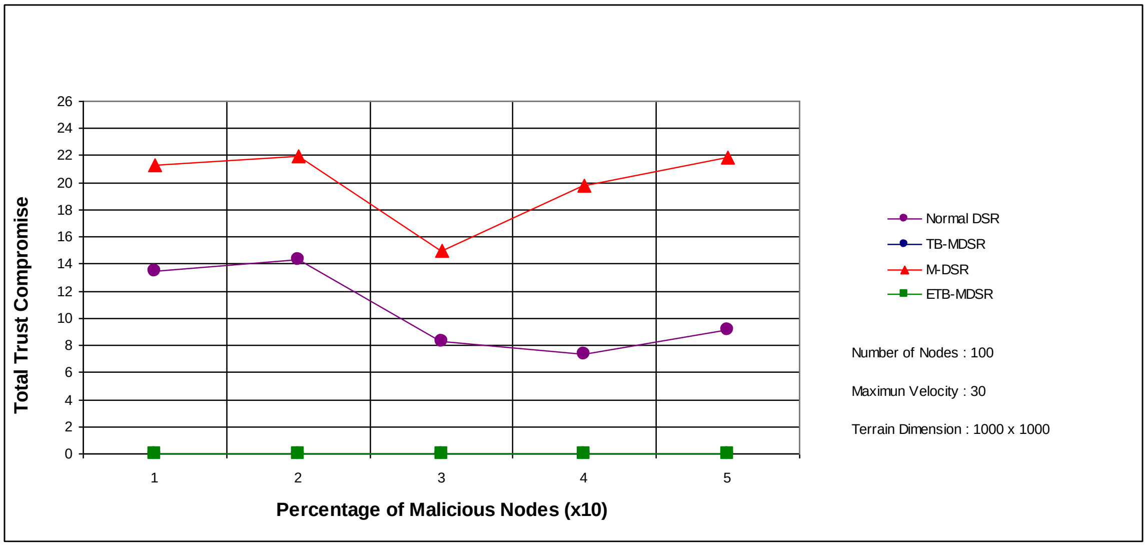

Figure 4.22: Trust compromise when varying the percentage of malicious nodes in the network, under one fixed mobility scenario, using one fixed number of nodes and a node's maximum speed of 30.

In Fig. 4.20, Fig. 4.21 and Fig. 4.22, it is observed that independent of the chosen maximum speed of nodes in the network, both the TB-MDSR and ETB-MDSR schemes have a trust compromise of 0 in all cases. This result is again in agreement with Theorem 1 stated in Section 3.4.3.

In Fig. 4.20, Fig. 4.21 and Fig. 4.22, it is also found that for the studied scenario, independent of the chosen maximum speed of nodes in the network, the message security of the normal DSR scheme is much better compared to that of the M-DSR scheme, no matter the proportion of malicious nodes in the network. This is due to the fact that for a given percentage of malicious nodes, the total trust compromise using the normal DSR is less than the total trust compromise when using the M-DSR scheme, in all cases.

4.2.4 Varying the terrain dimension with one fixed number of nodes, one fixed mobility scenario, and one fixed percentage of malicious nodes

In this scenario, the mobility scenario is fixed, the number of nodes is fixed to 100 and the maximum speed of nodes is fixed to 10.

4.2.4.1 Effect of the terrain dimension on the route selection time

The terrain dimension is varied and we study the impact of this variation on the *route selection time* for the studied four algorithms, for a given proportion of malicious nodes in the network. The results are shown in Fig. 4.23, Fig. 4.24, Fig. 4.25 and Fig. 4.26.

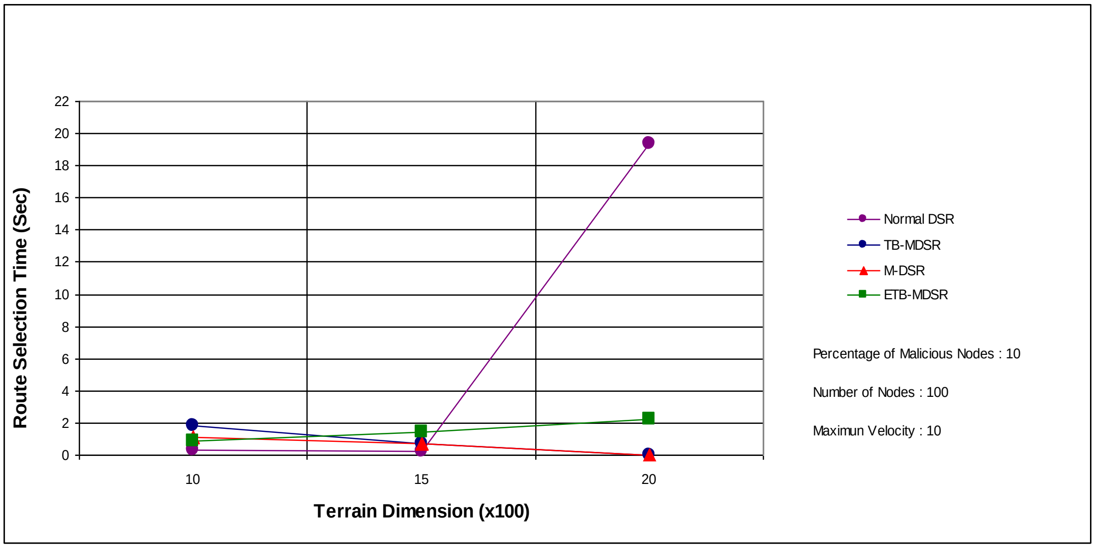

Figure 4.23: Route selection time when varying the terrain dimension, under one fixed mobility scenario and one fixed number of nodes, with 10% of malicious nodes present in the network.

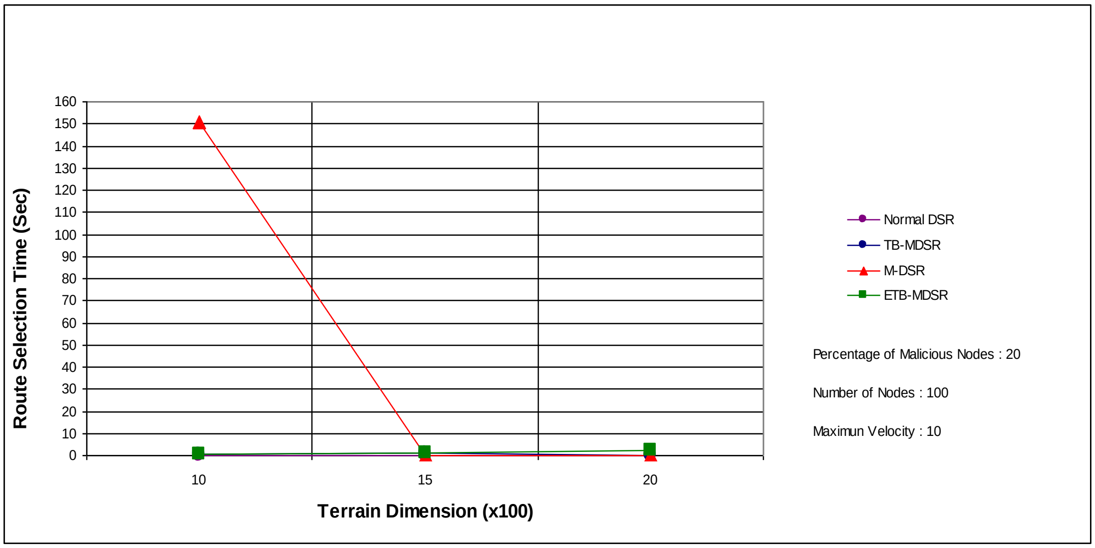

Figure 4.24: Route selection time when varying the terrain dimension, under one fixed mobility scenario and one fixed number of nodes, with 20% of malicious nodes present in the network.

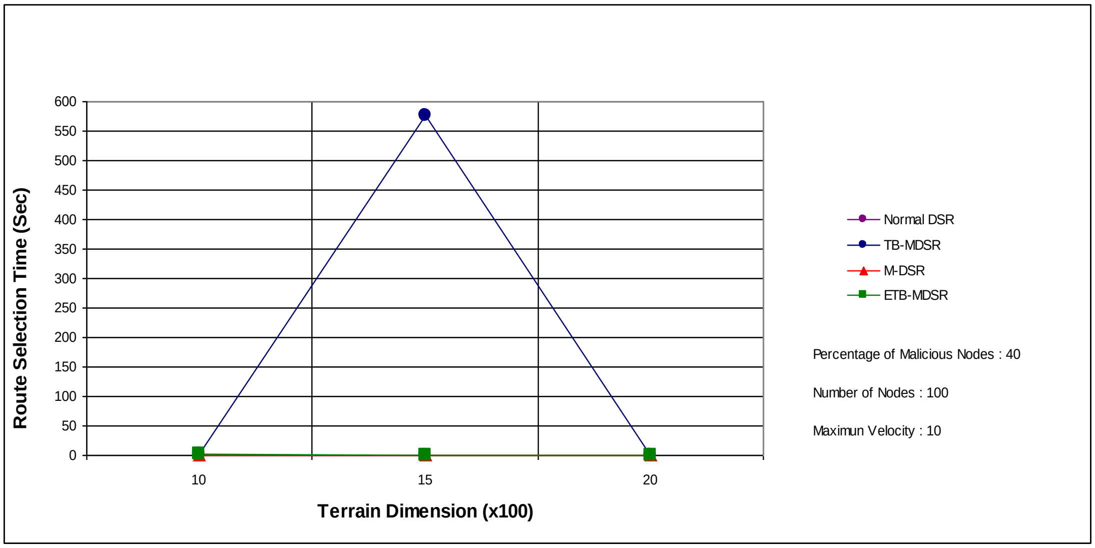

Figure 4.25: Route selection time when varying the terrain dimension, under one fixed mobility scenario and one fixed number of nodes, with 40% of malicious nodes present in the network.

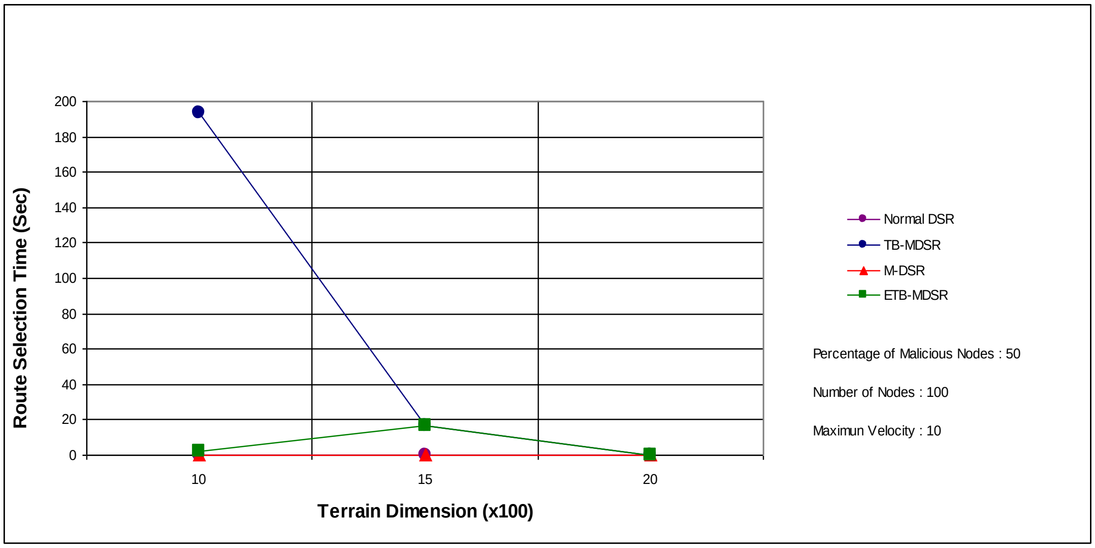

Figure 4.26: Route selection time when varying the terrain dimension, under one fixed mobility scenario and one fixed number of nodes, with 50% of malicious nodes present in the network.

In Fig. 4.23, Fig. 4.24, Fig. 4.25 and Fig. 4.26, it is observed that independent of the chosen terrain dimension, the normal DSR takes the minimum time to select the routing path set compared to all other schemes.

It is also observed (Fig. 4.25 and Fig. 4.26) that the ETB-MDSR scheme takes less time in selecting the trusted routing path compared to the TB-MDSR scheme when the proportion of malicious nodes in the network is important (at least 40% of the total nodes in the network). Below that threshold, this is not necessarily true (as illustrated in Fig. 4.23). In such cases, we have observed that the route selection times of both schemes depend on the placement of the source and the destination nodes in a network as well as on the network topology. As in the previous scenarios, the route selection times of the ETB-MDSR and normal DSR schemes can be equal or close each other in few cases (cases in Fig. 4.24 and Fig. 4.25).

4.2.4.2 Effect of the terrain dimension on the trust compromise

The terrain dimension is varied and we study the impact of this variation on the *trust compromise* for the studied four algorithms, for a given proportion of malicious nodes in the network. The results are depicted in Fig. 4.27, Fig. 4.28 and Fig. 4.29.

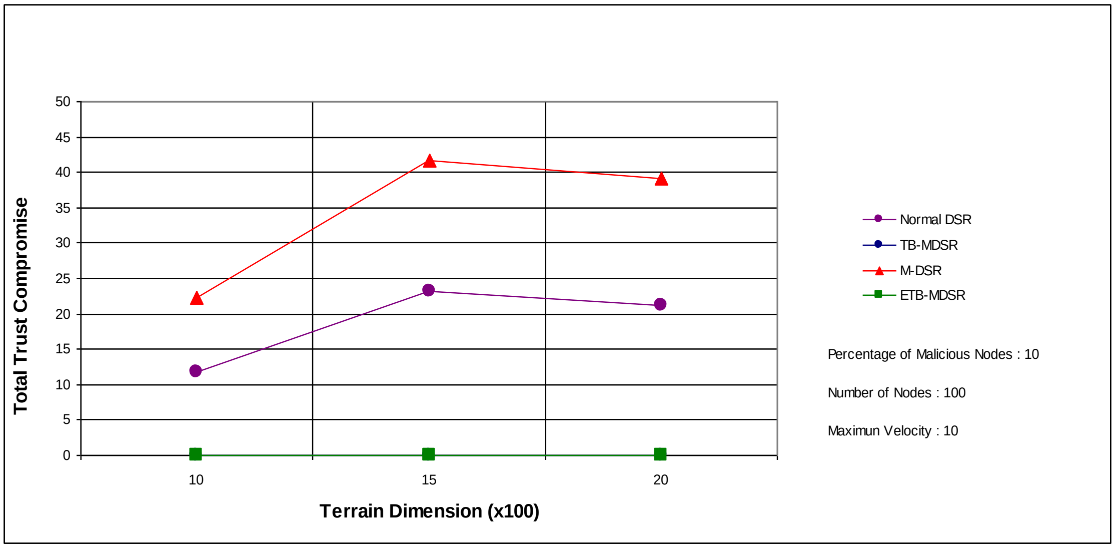

Figure 4.27: Trust compromise when varying the terrain dimension, under one fixed mobility scenario and one fixed number of nodes, with 10% of malicious nodes present in the network.

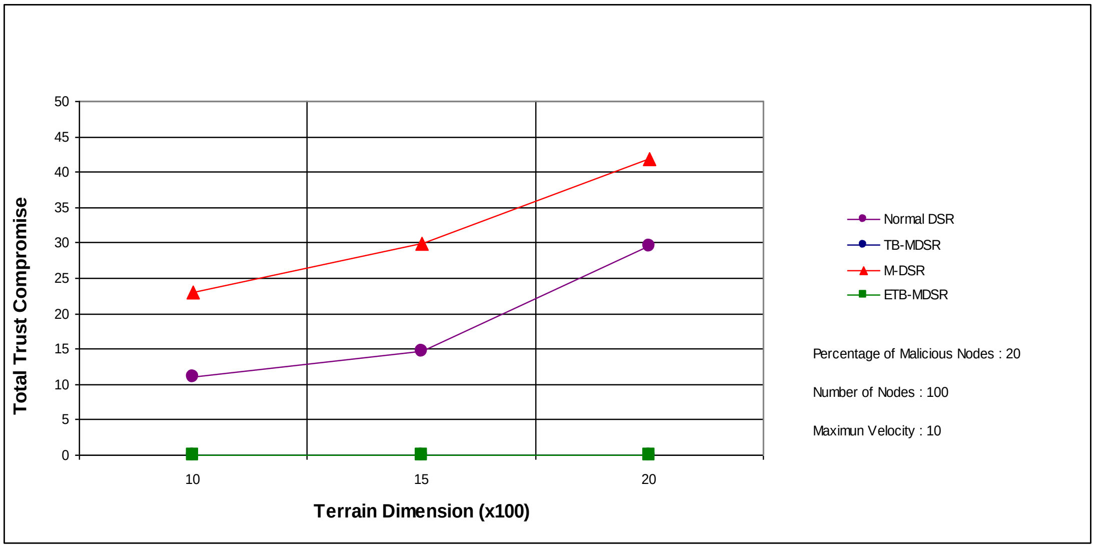

Figure 4.28: Trust compromise when varying the terrain dimension, under one fixed mobility scenario and one fixed number of nodes, with 20% of malicious nodes present in the network.

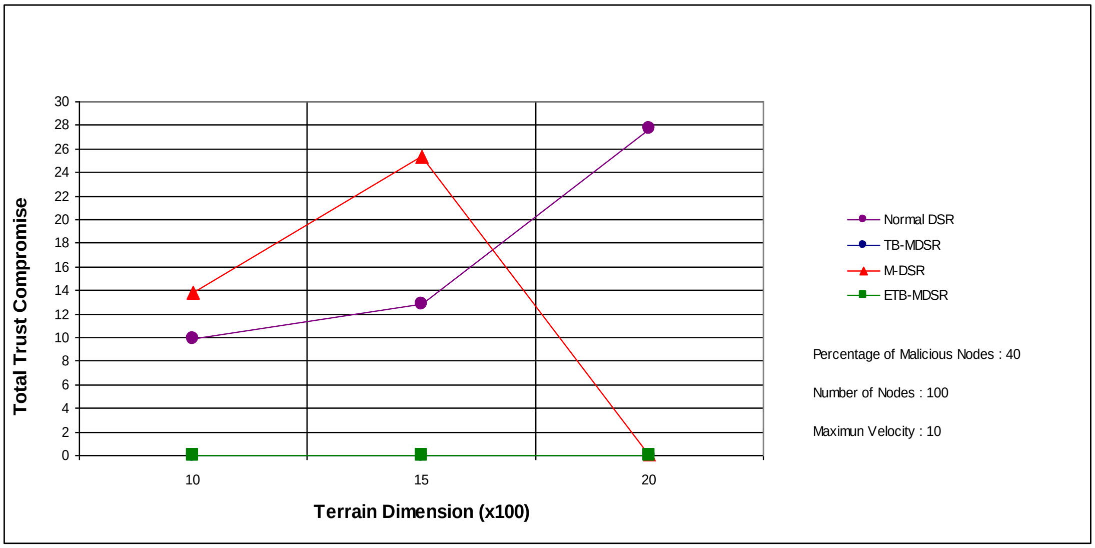

Figure 4.29: Trust compromise when varying the terrain dimension, under one fixed mobility scenario and one fixed number of nodes, with 40% of malicious nodes present in the network.

In Fig. 4.27, Fig. 4.28 and Fig. 4.29, it is observed that independent of the proportion of malicious nodes in the network, the TB-MDSR and ETB-MDSR schemes have a trust compromise of 0 in all cases. This result conforms to Theorem 1 stated in Section 3.4.3.

It is also observed that in general, the message security of the normal DSR scheme is much better than that of the M-DSR scheme (see Fig. 4.27 and Fig. 4.28) when the proportion of malicious nodes in the network is less than 40% of the total nodes in the network. Beyond this threshold, this is not necessarily true (as illustrated in Fig. 4.29). In such cases, we have observed that the decision on which scheme provides the lowest trust compromise depends on the chosen terrain dimension, thereby on the placement of the source and the destination nodes in a network as well as on the network topology.

4.2.5 Summary

In general, from our simulation study, we can conclude that:

- The route selection times of all four algorithms (normal DSR, M-DSR, TB-MDSR, and ETB-MDSR) depend on the placement of the source and the destination nodes in a network as well as on the network topology. In most cases studied, we found that the ETB-MDSR scheme takes less time in selecting the trusted routing path compared to the TB-MDSR scheme.
- The TB-MDSR and ETB-MDSR schemes both have a trust compromise of 0 in all cases. These results are in agreement with the theoretical proofs (Theorem 1) given in Section 3.4.3.
- The ETB-MDSR and TB-MDSR schemes are much more secure than the traditional multi-path DSR routing algorithms.
- Our results show that there is a compromise between message security (i.e. trust compromise) and routing time (i.e. route selection time), which is generally the

case with most of the security algorithms. A balance must be struck between the two to provide a maximum security without causing substantial delay for a user.

Conclusion

In this thesis, we have proposed an enhancement to a recently introduced routing strategy towards message security in MANETs (so-called Enhanced Trust-Based Multipath DSR). First, a comprehensive discussion on prominent works on security in MANETs has been presented, along with their respective advantages and limitations. Second, we have discussed our proposed enhanced message security scheme (ETB-MDSR) and contrast it against our so-called benchmark scheme (TB-MDSR) in terms of design components and features, mainly the trust model components and the trust strategy for route set selection. Thirdly, we have presented a theoretical foundation that captures the behaviours of the designed algorithms. Fourthly, we have evaluated by simulations our proposed ETB-MDSR scheme, and have compared it against (1) our so-called benchmark scheme (i.e. the trust-based multipath DSR (TB-MDSR), (2) the traditional multipath routing using 2-disjoint paths (M-DSR), (3) and the normal DSR, using the trust compromise and the route selection time as performance metrics.

Our results proved that both the TB-MDSR and ETB-MDSR schemes have a trust compromise of 0 in all cases, meaning these schemes are much more secure than the traditional multi-path DSR routing algorithms. Furthermore, the ETB-MDSR scheme takes in general less time in selecting the trusted routing path compared to the TB-MDSR scheme. Our results are found to be in agreement with the theoretical justifications presented in Section 3.4.3, showing that there is a compromise between message security (i.e. trust compromise) and routing time (i.e. route selection time). However, providing a maximum message security without causing substantial delay for a user would require a good balance between trust compromise and route selection time.

In the future, it will be interesting to investigate more efficient algorithms for selecting the trusted routes from a set of routes or to design a strong encryption methodology that would strengthen the whole message security scheme. Finally, the message security schemes discussed in this thesis can be implemented using other MANETs routing algorithms such as AODV [13], TORA [73], to name a few.

References

[1] W. K. Wolterink, "A Content-Based Routing Protocol for Mobile Ad-Hoc Networks Using a Distributed Connected k-Hop Dominating Set as a Backbone", *Masters Thesis, University of Twente,* EEMCS Department, Dec. 7, 2008.

[2] S. Corson, J. Macker, "Mobile Ad hoc Networking (MANET): Routing Protocol Performance Issues and Evaluation Considerations", *RFC 2501*, IETF, 1999

[3] E. Gustafsson, G. Karlsson, "A Literature Survey on Traffic Dispersion", *IEEE Networks*, 11(2): 28-36, Mar. 1997.

[4] Z. Ye, S. V. Krishnamurthy, S. K. Tripathi, "A Framework for Reliable Routing in Mobile Ad Hoc Networks", *IEEE INFOCOM 2003*, San Francisco, USA, Mar., 2003

[5] A. Nasipuri, R. Castaneda, S. R. Das, "Performance of Multipath Routing for On-Demand Protocols in Mobile Ad Hoc Networks", *Mobile Networks and Applications,* 6(4): 339-349, 2001

[6] W. Lou, W. Liu, Y. Fang, "SPREAD: Enhancing Data Confidentiality in Mobile Ad Hoc Networks", *IEEE INFOCOM 2004*, Hong Kong, China, Mar., 2004.

[7] Mishra, A. and Nadkarni, K. M., "Security in Wireless Ad-Hoc Networks", *Chapter 30, The Handbook of Wireless Ad-Hoc Networks, Ilyas, M. (Ed.),* CRC Press, ISBN 0849313325, 2003.

[8] S. Yi and R. Kravets, "Composite Key Management for Ad Hoc Networks", *Proc. of the 1st Annual International Conference on Mobile and Ubiquitous Systems: Networking and Services (MobiQuitous'04),* pp. 52-61, 2004.

[9] Y. Hu, A. Perrig, "A Survey of Secure Wireless Ad Hoc Routing*", IEEE Security and Privacy,* pp. 28-39, 2004

[10] P. Papadimitratos and Z. Haas, "Secure Data Transmission in Mobile Ad Hoc Networks", *Proc. of the 2003 ACM Workshop on Wireless Security*, pp. 41-50, 2003.

[11] C. E. Perkins, P. Bhagwat, “Highly Dynamic Destination-Sequenced Distance-Vector Routing (DSDV) for Mobile Computers”, *Proc. of the Conference on Communications Architectures, Protocols and Applications*, London, UK, Aug. 31 – Sept. 2, pp. 234-244, 1994.

[12] D. B. Johnson, D. A. Maltz, and J. Broch, “DSR: The Dynamic Source Routing Protocol for Multi-hop Wireless Ad Hoc Networks”, *Chapter 5 in Ad Hoc Networking, Charles E. Perkins (Ed.),* Addison-Wesley, pp.139-172, 2001.

[13] C.E. Perkins, E. M. Royer, “Ad-Hoc On Demand Distance Vector Routing”, *Proc. of IEEE WMCSA'99*, New Orleans, LA, pp. 90-100, Feb., 1999.

[14] T. Haniotakis, S. Tragoudas, and C. Kalapodas, “Security Enhancement Through Multiple Path Transmission in Ad Hoc Networks”, *Proc. of IEEE Intern. Conference on Communications*, June, pp. 4187-4191, 2004

[15] P. Narula, S. K. Dhurandher, S. Misra, and I. Woungang, "Security in Mobile Ad-Hoc Networks Using Soft Encryption and Trust Based Multi-path Routing”, *Computer Communications*, vol. 31, pp. 760-769, 2008.

[16] M. K. Denko, T. Sun, “Probabilistic Trust Management in Pervasive Computing”,

Proc. of the IEEE/IFIP Intern. Conference on Embedded and Ubiquitous Computing (EUC’08), pp. 610-615, Dec. 17-20, Shangai, China, 2008

[17] A. Abdul-Rahman, S. Hailes, “A Distributed Trust Model”, *Proc. of the 1997 Workshop on New Security Paradigms,* Langdale, Cumbria, UK, Sept. 23-26, ACM Press, NY, pp. 48-60, 1997.

[18] A. A. Pirzada, and C. McDonald, “Establishing Trust in Pure Ad-Hoc Networks”, *Proc. of the 27th ustralasian Conference on Computer Science*, Dunedin, New Zealand, Estivill-Castro (Ed.), pp. 47-54, 2004.

[19] L. Buttyan and J.-P Hubaux, “Stimulating Cooperation in Self-Organizing Mobile Ad Hoc Networks”, *Technical Report DSC/2001/046, EPFL-DI-ICA,* Aug. 2001.

[20] R. Norcen and A. Uhl, "Encryption of Wavelet-coded Imagery Using Random Permutations", *Proc. of 2004 International Conference on Image Processing*, Oct. 24-27, 2004, vol. 5, pp. 3431- 3434, 2004.

[21] S. Buchegger and J.-Y. LeBoudec, "Performance Analysis of the CONFIDANT Protocol: Cooperation of Nodes - Fairness In Dynamic Ad-Hoc NeTworks", *Proc. of IEEE/ACM Symposium on Mobile Ad Hoc Networking and Computing (MobiHOC),* Lausanne, Switzerland, June 2002.

[22] P. Michiardi and R. Molva, "CORE: A Collaborative Reputation Mechanism to Enforce Node Cooperation in Mobile Ad Hoc Networks", *Proc. of the IFIP TC6/TC11 Sixth Joint Working Conference on Communications and Multimedia Security,* Portoroz, Slovenia, pp. 107-121, 2002.

[23] S. D. Kamvar, M. T. Schlosser, and H. Garcia-Molina, "The EigenTrust Algorithm for Reputation Management in P2P Networks", *Proc. of the Twelfth International World Wide Web Conference*, Budapest, Hungary, pp. 640-651, 2003.

[24] Y.-C. Hu, A. Perrig, and D. B. Johnson, "ARIADNE: A Secure On-Demand Routing Protocol for Ad Hoc Networks", *Proc. of MOBICOM 2002*, Atlanta, Georgia, USA, 2002.

[25] I. Cidon, R. Rom, and Y. Shavitt, "Analysis of Multi-Path Routing", *IEEE/ACM Transactions on. Networking*, vol. 7, No. 6, Dec. , pp. 885-896, 1999.

[26] W. Lou, W. Liu, and Y. Fang, "SPREAD: Enhancing Data Confidentiality in Mobile Ad Hoc Networks", *Proc. of INFOCOM 2004*, March, pp. 2404-2413, 2004.

[27] W. Wang, Y. Zhu, B. Li, "Self-Managed Heterogeneous Certification in Mobile Ad Hoc Networks", *Proc. of IEEE Vehicular Technology Conference (VTC 2003),* pp. 2137-2141, 2003.

[28] P. Papadimitratos, P. and Haas, Z. J., "Secure Routing for Mobile Ad Hoc Networks", *In Proc. of SCS CNDS*, San Antonio, TX, Jan. 27–31, pp. 193–204, 2002.

[29] Hu, Y., Perrig, A. and Johnson, D. B., “Secure Efficient Distance Vector Routing for Mobile Wireless Ad Hoc Networks”, Ad Hoc Networks, Vol. 1, No. 1, Jul. 2003, pp. 175–190.

[30] P. Papadimitratos and Z. J. Haas, “Secure QoS-Aware Route Discovery in Ad Hoc Networks”, *Proc. of 2005 IEEE Sarnoff Symposium*, Princeton, NJ, USA, Apr., pp. 176–179, 2005

[31] P. Papadimitratos, “Secure and Fault-Tolerant Communication in Mobile Ad Hoc Networks”, *Ph.D. Thesis, Cornell University*, Ithaca, NY, Jan. 2005.

[32] S. Marti, T. J. Giuli, K. Lai, and M. Baker, “Mitigating Routing Misbehavior in Mobile Ad Hoc Networks”, *Proc. of 6th MobiCom*, Boston, MA, USA, Aug., pp. 255–265, 2000.

[33] B. Dahill, B. Neil, E. Royer, and C. Sheilds, “A Secure Protocol for Ad Hoc Networks", *Proc. of IEEE Intern. Conference on Network Protocols (ICNP 2002),* Nov. 12-15, Paris, France, 2002.

[34] V. Gupta, S. Krishnamurthy, and M. Faloutsos, “Denial of Service Attacks at the MAC Layer in Wireless Ad Hoc Networks”, *Proc. of IEEE Military Communications Conference (MILCOM 2002),* Anaheim, CA, USA, Oct., 2002.

[35] B. Awerbuch, D. Holmer, C. Nita-Rotaru, and H. Rubens, “An On-Demand Secure Routing Protocol Resilient to Byzantine Failures”, *Proc. of the ACM Workshop on Wireless Security (WiSe),* Sept., Atlanta, GA, USA, 2002.

[36] H. Yang, X. Meng, and S. Lu, “Self-Organized Network Layer Security in Mobile Ad Hoc Networks”, *Proc. of the ACM Workshop on Wireless Security (WiSe)*, Sept., Atlanta, GA, USA, 2002.

[37] Y. Hu, A. Perrig, A., and D. Johnson, “Packet Leashes: A Defense Against Wormhole Attacks in Wireless Networks”, *Proc. of IEEE INFOCOM 2002,* June 23-27, New York, USA, 2002.

[38] I. Woungang, M. K. Denko, "Credit-based Cooperation Enforcement Schemes Tailored to Opportunistic Networks", *Chapter 3 in: M. Denko et al. (Eds.), Mobile Opportunistic Networks: Architectures, Protocols and Applications,* Auerbach Publications, Taylor & Francis Group, Boca Raton, Florida, To appear Dec. 2010.

[39] Y. Zhang, W. Lou, W. Liu and Y. Fang, "A Secure Incentive Protocol for Mobile Ad Hoc Networks", *ACM Wireless Networks*, vol. 13, no. 5, pp. 569-582, October 2007.

[40] H. Zhu, X. Lin, R. Lu, Y. Fan, and X. Shen, "SMART: A Secure Multi-Layer Credit Based Incentive Scheme for Delay-Tolerant Networks", To appear in IEEE Transactions on Vehicular Technology, 2010

[41] H. Janzadeh, K. Fayazbakhsh, M. Dehghan, M. S. Fallah, "A Secure Credit –Based Cooperation Stimulating Mechanism for MANETs Using Hash Chains", *In Future Generation Computer Systems*, vol. 25, Issue 8, Sept., pp. 926-934, 2009.

[42] F. Wu, T. Chen, S. Zhong, L. E. Li and Y. R. Yang, "Incentive-Compatible Opportunistic Routing for Wireless Networks", Proc. of ACM MOBICOM 2008, San Francisco, CA, USA, Sept., 2008.

[43] A. Garyfalos and K. C. Almeroth, "Coupons: A Multilevel Incentive Scheme for Information Dissemination in Mobile Networks", *In IEEE Transactions on Mobile Computing*, vol. 7, No. 6, June 2008.

[44] R. Lu, X. Lin, H. Zhu, C. Zhang, P.H. Ho and X. Shen, "A Novel Fair Incentive Protocol for Mobile Ad Hoc Networks", *Proc. of IEEE WCNC'08,* Las Vegas, Nevada, USA, Mar. 31 – Apr. 3, 2008.

[45] S. K. Dhurandher, S. Misra, S. Ahlawat, N. Gupta, N. Gupta, "E2-SCAN: An Extended Credit Strategy-Based Energy-Efficient Security Scheme for Wireless Ad Hoc Networks", *IET Communications*, U.K., vol. 3, Issue 5, 2009, pp. 809-819.

[46] D. Boneh, M. Franklin, "Identify-Based Encryption from the Weil Pairing", *In Proc. Of CRYPTO'01, LNCS, vol. 2139,* Springer-Verlag, pp. 213-229, 2001.

[47] U. Shevade, H. H. Song, L. Qiu, Y. Zhang, “Incentive-Aware Routing in DTNs”, *In Proc. of the 16th IEEE International Conference on Network Protocols (ICNP 2008)*, Orlando, FL, USA, Oct. 19-22, pp. 238-247, 2008.

[48] T. Bocek, Y. El-khatib, F. Victora Hecht, D. Hausheer, B. Stiller, “CompactPSH: An Efficient Transitive TFT Incentive Scheme for Peer-to-Peer Networks”, *In: M. Younis, C. Chun Tung, Local Computer Networks,* Zurich, Switzerland, pp. 483-490, 2009.

[49] M. Felegyhazi, J.-P. Hubaux, and L. Buttyan, “ìNash Equilibria of Packet Forwarding Strategies in Wireless Ad Hoc Networks”, *IEEE Trans. On Mobile Computing*, 2006.

[50] J. Kong, P. Zerfos, H. Luo, S. Lu, L. Zhang, “Providing Robust and Ubiquitous Security Support for MANET”, *Proc. of IEEE Intern. Conference on Network Protocols (ICNP 2001),* Nov. 11-14, Riverside, CA, USA, pp. 251–260, 2001.

[51] Q. He, D. Wu, and P. Khosla, “SORI: A Secure and Objective Reputation-Based Incentive Scheme for Ad Hoc Networks”, *Proc. of the Wireless Communications and Networking Conference (WCNC 2004),* Atlanta, GA, Mar. 2004.

[52] Y. Zhang and Y. Fang, “A Fine-grained Reputation System for Reliable Service Selection in Peer-to-peer Networks”, *IEEE Trans. on Parallel and Distributed Systems*, vol. 18, no. 8, pp. 1134-1145, August 2007.

[53] Z. Zhang, F. Nait-Abdesselam, P.-H. Ho, and X. Lin, "RADAR: a Reputation-Based Scheme for Detecting Anomalous Nodes in Wireless Mesh Networks", *In Proc. of the Wireless Communications and Networking Conference (WCNC 2008),* Mar. 31-Apr. 3, Las Vegas, Nevada, USA, 2008.

[54] X. Xie, H. Chen, and H. Wu, "A Queuing Model-based Incentive Scheme for Optimal Data Transmission in Wireless Networks with Selfish Nodes", *In Proc. of the 5th IEEE Intern. Conference on Mobile Ad Hoc and Sensor Systems (MASS 2008),* Atlanta, Georgia, USA, Sept- 29- Oct.2, pp. 463-468, 2008.

[55] N. J. Al-Karaki and A. E. Kamal, "Stimulating Node Cooperation in Mobile Ad hoc Networks", *Wireless Personal Communications,* vol. 44, Issue 2, pp. 219-239, Jan. 2008.

[56] K. Hoffman, D. Zage, C. Nita-Rotaru, “A Survey of Attack and Defense Techniques for Reputation Systems”, *ACM Computing Surveys*, vol. 42, Issue 1, Dec. 2009.

[57] S. Bansal, M. Baker, “Observation-based Cooperation Enforcement in Ad hoc Networks”, *Technical Paper, Computer Science Department*, Stanford University, July 2003.

[58] J. Hu, M. Burmester, “LARS: A Locally Aware Reputation System for Mobile Ad Hoc Networks”, *Proc. of 44th Annual Southeast Regional Conference,* pp. 119-123, 2006.

[59] T. Bocek, M. Shann, D. Hausheer and B. Stiller, "Game Theoretical Analysis of Incentives for Large-scale, Fully Decentralized Collaboration Networks", *Proc. of the IEEE International Symposium on Parallel and Distributed Processing (IPDPS 2008),* Apr. 14-18, pp. 1-8, 2008.

[60] J. J. Jaramillo and R. Srikant. DARWIN: Distributed and Adaptive Reputation Mechanism for Wireless Ad-Hoc Networks”, *In Proc. of ACM MobiCom 2007,* pp. 87-98, New York, NY, USA, 2007.

[61] F. Milan, J. J. Jaramillo, and R. Srikant. “Achieving Cooperation in Multi-hop Wireless Networks of Selfish Nodes”, *In Proc. of 2006 Workshop on Game Theory for Communications and Networks,* Pisa, Italy, 2006.

[62] K. Sanzgiri, D. LaFlamme, B. Dahill, B. N. Levine, C. Shields, and E. M. Belding-Royer, "Authenticated Routing for Ad hoc Networks", *IEEE Journal on Selected Areas in Communications,* 23.3 (2005): 598-610.

[63] F. X. Junmo, L. Y. Liu, J. Fu, “Secure Routing for Mobile Ad Hoc Networks”, *In 8th IEEE/ACIS International Conference on Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing (SNPD 2007)*, vol. 3, pp. 314-318, July 30-Aug. 1, 2007

[64] F. De Rango, S. Marano, “Trust-based SAODV Protocol with Intrusion Detection and Incentive Cooperation in MANET”, *In Proc. of the 2009 Intern. Conference on Wireless Communications and Mobile Computing: Connecting the World Wirelessly*, Leipzig, Germany, pp. 1443-1448, 2009.

[65] A. A. Pirzada, A. Datta, C. McDonald, “Propagating Trust in Ad-Hoc Networks for Reliable Routing”, *In Proc. of the 2004 Intern. Workshop on Wireless Ad-Hoc Networking*, May-June, pp. 58-62, Tokyo, Japan, 2004.

[66] A. A. Pirzada, A. Datta, and C. McDonald, "Trust-Based Routing for Ad-Hoc Wireless Networks," In Proc. of the 12th IEEE International Conference on Networks (ICON 2004), pp. 326-330, Nov. 2004.

[67] A. Boukerche, Y. Ren, “A Trust-based System for Ubiquitous and Pervasive Computing Environments”, *Computer Communications*, 31 (2008), pp. 4343-4351.

[68] A. Whitby, A. Josang, J. Indulska, “Filtering out Unfair Ratings in Bayesian Reputation systems”, *In Icfain Journal of Management Research*, vol. 4, Issue 2, pp. 48-64, 2005.

[69] M. Takai, L. Bajaj, R. Ahuja, R. Bargrodia, and M. Gerla, “GloMoSim: A Scalable Network Simulation Environment”, *Technical Report 990027, Department of Computer Science, University of California Los Angeles*, USA, 1999.

[70] R. Bargodia, R. Meyer, M. Takai, Y.-A. Chen, X. Zeng, J. Martin, and H. Y. Song, “PARSEC: A Parallel Simulation Environment for Complex Systems”, *IEEE Computer*, vol. 31, No. 10, Oct. 1998, pp. 77-85.

[71] B. A. Forouzan, “Data Communications and Networking, 2/e”, *McGraw Hill*, ISBN: 0072515848, 963 pages, 2003

[72] 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

[73] E. M. Royer, C. K. Toh, "A Review of Current Routing Protocols for Ad Hoc Mobile Wireless Networks", *IEEE Personal Communications*, vol. 6, Apr., pp. 46-55, 1999.