TRUST BASED KEY MANAGEMENT FRAMEWORK FOR MANET

A Thesis submitted to

Cochin University of Science and Technology

in partial fulfillment of the requirements

for the award of the degree of

Doctor of Philosophy

Under the Faculty of Technology

By

SAJU P JOHN
(Reg No. 3804)

Under the Guidance of

Dr. PHILIP SAMUEL

Department of Computer Science

Cochin University of Science and Technology

Cochin - 682 022, Kerala, India

November 2016

TRUST BASED KEY MANAGEMENT FRAMEWORK FOR MANET

Ph.D. Thesis under the Faculty of Technology

Author

SAJU P JOHN
Research Scholar
Department of Computer Science
Cochin University of Science and Technology
Kochi - 682022
Email: sajupjohn33@gmail.com

Supervising Guide

Dr PHILIP SAMUEL
Information Technology Division, School of Engineering
Cochin University of Science and Technology
Kochi - 682022
Email: philipcusat@gmail.com

Department of Computer Science

Cochin University of Science and Technology

Cochin -682022

CERTIFICATE

Certified that the work presented in this thesis entitled **"Trust based key management framework for MANET”** is a bonafide work done by Mr. Saju P. John, under my guidance in the Department of Computer Science, Cochin University of Science and Technology and that this work has not been included in any other thesis submitted previously for the award of any degree.

Kochi Dr. Philip Samuel

29th November 2016 (Supervising Guide)

Department of Computer Science

Cochin University of Science and Technology

Cochin -682022

CERTIFICATE

Certified that the thesis entitled "Trust based key management framework for MANET” work done by Mr. Saju P. John, has incorporated all the relevant corrections and modifications suggested by the audience during the pre-synopsis and recommended by the doctoral committee of the candidate.

Kochi

29th November 2016

Dr. Philip Samuel

(Supervising Guide)

DECLARATION

I hereby declare that the work presented in this thesis entitled **"Trust based key management framework for MANET"** is based on the original work done by me under the guidance of Dr. Philip Samuel, Associate Professor, Information Technology Division, School of Engineering, Cochin University of Science and Technology and has not been included in any other thesis submitted previously for the award of any degree.

Kochi

29th November 2016 Saju P John

ACKNOWLEDGMENTS

At the outset, I thank **God Almighty** for providing me the great opportunity to do this research work and complete it within the stipulated time.

I have great pleasure in expressing my profound gratitude to my guide, **Dr. Philip Samuel**, for the motivation and guidance provided throughout this research work. I have been extremely lucky to have a guide who cared so much about my work, and who responded to my questions and queries so promptly. I appreciate his approachability and the trust bestowed on my research efforts. Consistent support, patient hearing and sincere encouragement provided throughout the period are gratefully acknowledged.

I wish to express my sincere gratitude to **Dr. Sumam Mary Idicula**, Head, Department of Computer Science, Cochin University of Science and Technology, for her guidance, help and encouragement throughout the period of this work.

I thank the doctoral committee members, all the faculty members of Department of Computer Science and Information Technology Division, School of Engineering, Cochin University of Science and Technology for the cooperation extended towards me. The excellent computing and library facilities offered by the department were really supportive for carrying out my thesis work. The support of all the non teaching staff of Department of Computer Science, Cochin University of Science and Technology throughout the period is gratefully acknowledged.

I have no words to express my deep sense of gratitude to my colleagues and well-wishers at Thejus Engineering College, for helping me to complete this work. Special thanks to **Sri C C Thampi**, Chairman, Thejus Engineering College, Principal and all heads of departments for their valuable cooperation and encouragement during this period. My sincere thanks to all teaching & non teaching staff of department of computer science, Thejus Engineering College for their cordial relations, sincere co-operation and valuable help, especially to **Dr Gylson Thomas**, Dean PG Studies, **Prof. Jose T P**, Professor Mathematics, **Mr Serin V Simpson**, Assistant Professor, Computer Science Department, **Mr Sreejith U S**, System Manager, who helped a lot during different stages of this research work.

I am blessed to have the unconditional everlasting love from my parents **John & Sosamma** and my parents in law **Abraham & Annie** whose encouragement and support always kept me overcoming hard times. I dedicate this accomplishment to them. I also thank my adorable brothers **Soju & Siju** and sister in law **Raji** for helping me during the difficult situations.

Finally, but most importantly, I would like to thank my dearest wife, **Niji**, for being so understanding and for putting up with me through the toughest moments of my life. I thank God for enlightening my life with her presence. Without her sacrifices, love and support, this thesis would not have been possible. I also dedicate this Ph.D. thesis to my two lovely children, **Nevin and Aiden** who are the pride and joy of my life.

Saju P John

CONTENTS

CHAPTER 2: LITERATURE REVIEW

CHAPTER 3: SELF-ORGANIZED KEY MANAGEMENT FOR TRUSTED CERTIFICATE EXCHANGE AND REVOCATION IN MANET

ABSTRACT

Wireless ad hoc network is a collection of mobile nodes dynamically forming a temporary network without a centralized administration. This kind of network has been applied for both civilian and military purposes. However, security in wireless ad hoc networks is hard to achieve due to the vulnerability of the links, limited physical protection of the nodes, and the absence of a certification authority or centralized management point. Consequently, novel approaches are necessary to address the security problem and to cooperate with the properties of wireless ad hoc network. Similar to other distributed systems, security in wireless ad hoc networks usually relies on the use of different key management mechanisms. The compromise of the node breaks down the whole security system.

In this work, we present a security frame work based on trust for key management in mobile adhoc networks. Nodes originally trust-worthy in the network may be compromised after the attacks. These malicious nodes can harm the authentication service by signing false certificates. Hence, adequate measure is essential to protect the network security. The dissertation research provides new understanding of a trust based framework for key generation, key distribution based on certificate exchange, trusted source routing and detection of malicious node by certificate revocation. In addition, we propose trust management mechanism based on the reputation parameters and also proposed a trust prediction model for predicting the future trust of a node. A combinatorial scheme and prediction based node movement technique for effective key management in cluster based MANET is also proposed. Our trust based framework is able to discover and isolate malicious nodes in the network. Finally, we perform security and performance evaluation on the proposed solution through simulations.

Preface

The characteristics of self-organization and wireless medium make Mobile Ad hoc Networks (MANET) easy to set up and thus attractive to users. MANET has several advantages compared to traditional wireless networks. These include ease of deployment, speed of deployment and decreased dependency on a fixed infrastructure. Security is one of the most indispensable research areas and plays a central role in determining the success of civilian and commercial mobile ad hoc networks. Unfortunately, security solutions that have been proposed for wired networks are not directly inheritable into the MANET, because of the variant attack patterns and the new types of adversary models. In other words, mobile nodes struggle to enlist trusted intermediaries for communication with various destinations, because trusted intermediaries are a prerequisite for keeping those communications alive and free from active attacks.

Several security based routing protocols have been proposed to assist a mobile node to discover a secure path to the destination. Several cryptographic mechanisms are used to achieve the objective of the secure routing protocol. It is also noted that the functionality of secure routing protocols relies heavily on the existence of a robust key management service. Key management is responsible for initializing and distributing keys between nodes in a secured manner, and also responsible for revoking the keys when a node capture attack occurs.

Key management and secure routing protocols are only designed to defend against predefined active attacks and also to act as a prevention system. The mobile nodes should be designed to support and defend against selectively misbehaving nodes or emerging attacks. To ensure the security of the intermediary nodes and to act as a detection-reaction system for MANET, a trust management mechanism is also needed.

The thesis, presented in eight chapters deals with the work carried out in designing a trust based framework for secure key Management in MANET.

Chapter 1 -- Introduces the area of mobile adhoc networks, its applications, design challenges, security threats, and the need of trust management mechanism. The research problem identification and objectives of the research work is also included.

Chapter 2 – is a systematic survey on existing key management techniques, certificate exchange mechanisms and certificate revocation mechanisms based on both trust based, non trust based in the literature are also given. Taxonomy and a comparison based on various criteria of the surveyed mechanisms are presented.

Chapter 3 – Presented the proposed method, Self-Organized Key Management for Trusted Certificate Exchange and Revocation for MANET. The proposed scheme is simulated and performance comparisons with the basic methodology are exhibited.

Chapter 4 – Discussion of Trust Prediction Model based on accusations for certificate exchange and revocation with simulation results and performance comparisons with existing approach.

Chapter 5 – Discussion of the effect of M-OLSR Protocol in the proposed framework for certificate exchange and revocation in Mobile Ad Hoc Network with simulation results and performance comparisons with existing methodology.

Chapter 6 – Discussion of the cluster based combinatorial scheme for key management in Mobile Ad Hoc Networks with simulation results and performance comparisons with existing methodology.

Chapter 7 - Discussion of Prediction based clustering system for Distributed Hierarchical Key Management in Mobile Ad Hoc Networks with simulation results and performance comparisons with existing methodology

Chapter 8 – Concludes the thesis and mentions possible future research directions.

Some of the results have been published in international journals and in the proceedings of various international conferences, the details of which are given at the end of this thesis report.

GLOSSARY OF SYMBOLS AND ABBREVIATIONS

SYMBOLS

R_{ij}	- The adjacency matrix of the network
S(i)	- The set of nodes that are connected to the i^{th} node
n	- The total number of nodes
δ	- A constant
kpu_d	- Public key of Destination
kpu_s	- Public key of Source
T(S)	- Set of nodes certified for kpu_s
REQ_{cert}	- Certificate request message
REP_{cert}	- Certificate reply message
C_{self}	- Self-signed certificate
ID_D	- The identity value of Destination
CP_c	- The cumulative count of right sending control packets
TCP_c	- The aggregate of all control packets from time 0 to t
DP_c	- The cumulative count of right sending data packets
TDP_c	- The aggregate of all data packets from time 0 to t

ABBREVIATIONS

MANET	- Mobile Ad hoc Network (MANET)
DoS	- Denial of service (DoS)
AODV	- Ad hoc On Demand Vector routing
DSR	- Dynamic Source Routing
CA	- Certificate authority
KEK	- Key Encryption Key
DH	- Diffie-Hellman
HMAC	- Hash function based message authentication code
PGP	- Pretty good privacy
CRL	- Certificate revocation list
TTP	- Trusted third party
PKI	- Public key infrastructure
BL	- Black List
ADP	- Attack Detection Packets
WL	- Warning list
ET	- Expiry time
RWREQ	- Renewal request packet
ADP	- Attack Detection Packets
AWC	- Adaptive Weighted Cluster
CH	- Cluster head

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

CONTENTS

1.1 Mobile Ad hoc Network (MANET)

In Mobile Ad Hoc Network (MANET), a collection of nodes having wireless in nature are formed as a transitory/short-lived network not having any fixed infrastructure (as shown in fig.1.1). In MANET all the nodes can move freely and capable to organize themselves. Each node has dual functionality as router and host where the topology may be changing suddenly [1]. Ad hoc networking is used wherever the infrastructure is little or without any physical communication or the existing infrastructure is costly or problematic to use. It lets the devices to preserve connections to the network and also to add or remove a device/node.

There are different arrangement of uses for MANETs, running from expansive scale, portable, profoundly dynamic systems, to little and static systems which are having restricted force sources. Notwithstanding the legacy applications that move from customary framework environment into the specially appointed environment, an extraordinary course of action of new administrations will be made for the new environment. It comprises Military Battlefield, Sensor Networks, Commercial Sector, Medical Service and Personal Area Network [2].

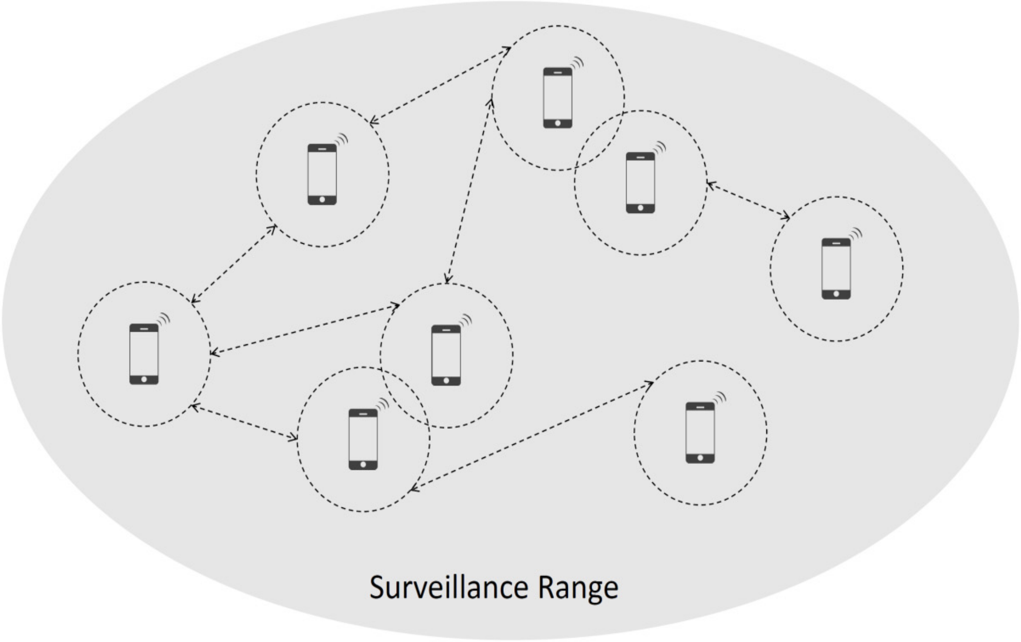

Fig 1.1 Ad hoc Wireless Network (MANET)

Mobile ad hoc networks are vulnerable to attacks compared to the wired networks. The wireless links between the mobile nodes are not secured for communication without imposing proper security measures. A quick and cost effective deployment is required for ad hoc wireless networks. The limited power supply causes denial-of-service attacks issue [3]. The trust relationship among nodes may be disturbed by the Dynamic topology and changeable nodes membership. If some nodes are detected as compromised, it also disturbs the trust. Distributed and adaptive security mechanisms can protect this dynamic behavior [4].

Since the self organization and maintenance properties are built into the ad hoc networks makes it defenseless against attacks. The following are the different challenges and security issues in MANET [5].

- ***Availability*:** Should withstand survivability paying little respect to DoS attacks like in physical and media access control layer assailant utilizes jamming techniques for obstruct with communication on physical channel. On network layer the attacker can intrude on the routing protocol. On higher layers, the attacker could cut down abnormal state services, e.g., key management service.
- ***Confidentiality*:** Should shield certain data which is not to be uncovered to unauthorized elements.
- ***Integrity*:** Transmitted Message ought to be honest to goodness and ought to never be adulterated.
- ***Authentication*:** Empowers a node to shield the qualities of the peer node it is imparting, without which an attacker would copy a node, in this manner

accomplishing unauthorized admission to asset and sensitive data and snooping with operation of different nodes.

- ***Non-Repudiation*****:** Shields that the source of a data ought not to dismiss having sent the data.

1.2 Attacks on MANETs

MANETs are inclined to a few sorts of attacks, which can essentially be ordered into two structures as per the way of the attacks as; Active attacks and passive attacks.

- **Active attacks** – Under such attacks, the attacker means to bring about jamming, transmitting fake routing data or interfere with nodes from giving services. A few cases of active attacks are Black Hole Attacks [6] and Flooding Attacks.
- **Passive attacks** – Under such attacks, the attacker tries to pick up control access over the network [7]. A passive attack does not disrupt the operation of the network, the advisory snoops the data exchanged in the network without altering it. Here the requirements of confidentiality can be violated if an advisory is also able to interpret the data gathered through snooping.

The blend of passive attacks, active attacks, and physical attacks utilized by the malicious client/clients to seize or degenerate network and takes control over the node is known as Node capture attack [8]. The malicious client might actuate replicated or tainted data into the node which can affect the entire network/link to be malfunctioning. These node capture attacks happen because of the uncalled for consideration of the wireless nodes and the high cost of fool-proof hardware in portable devices [9].

To set up or start up a node capture attack, the intruder arranges all data about the nodes or network by eavesdropping on message exchanges. The intruder can know about the composition of the network, regardless of the possibility that message payloads are encrypted, as it can extract the data if the nodes are compromised. Once an adequate measure of passive attacks and active attacks has occurred, the intruder can physically capture nodes [10].

The threats which are included because of compromised (captured) node are significantly more serious than the attacks from outside the network. As mobile nodes are autonomous and can join or leave any network voluntarily, it is difficult to monitor such nodes continually. The mobility of nodes makes this malicious node to continually change the attack target and perform malicious attacks on distinctive networks. There is a more prominent risk on authentication [11] as a message navigating numerous links between a source and destination node is compromised if any of the crossed links in the route gets to be unreliable. These attacks are further classified into four major categories Figure 1.2 which are described as follows:

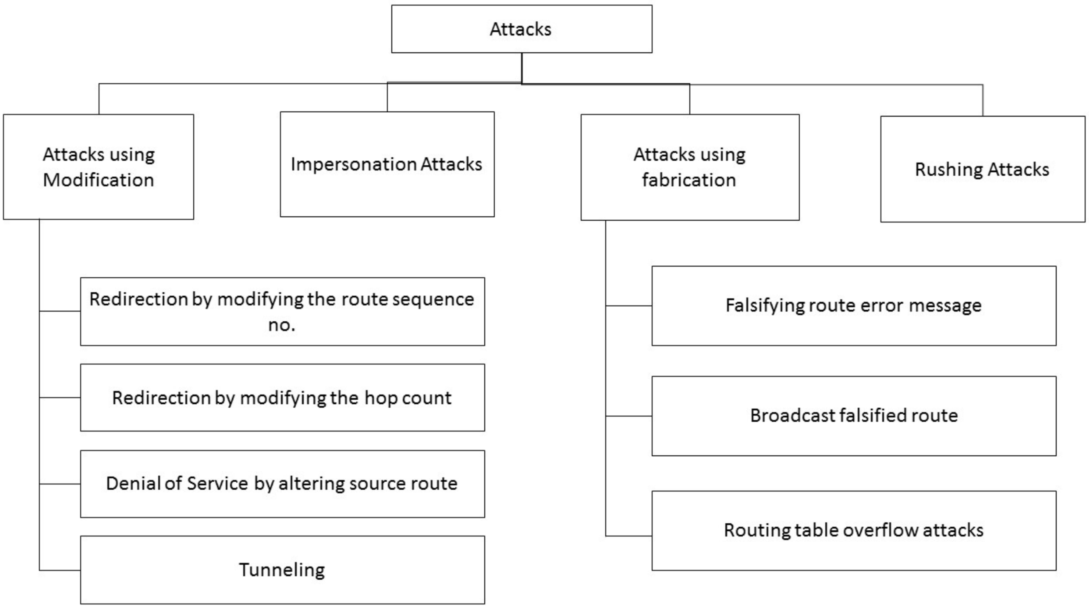
Attacks
Attacks using Modification
Impersonation Attacks
Attacks using fabrication
Rushing Attacks
Redirection by modifying the route sequence no.
Redirection by modifying the hop count
Denial of Service by altering source route
Tunneling
Falsifying route error message
Broadcast falsified route
Routing table overflow attacks

Fig 1.2 Attacks in MANET

- **Attacks using modification**
  - Redirection by changing the route sequence number: with a specific end goal to locate the best route to the destination, nodes dependably relies on the metric values, for example, sequence no, hop count, delay and so on. Normally source will select the path having minimum number of Hops. In this attack, malicious node can redirect the traffic by advertizing the best hop-count value.
  - Redirection by modifying the hop count: Here, in this attack packet traffic can be occupied to any compromised node by changing the hop count metrics to a smaller value.
  - Denial of Service by altering source route: Denial of Service attacks go for the complete obliteration of the routing function. Through modification, an attacker can bring about network traffic to be dropped, redirected to an alternate destination or to a more extended route to reach the destination that causes superfluous communication delay.
  - Tunneling: A tunneling attack is where two or more nodes may collaborate to encapsulate and exchange messages between them along existing data routes. This endeavor gives the chance to a node or nodes to hamper typical stream of messages making a virtual vertex cut in the network that is controlled by the two colluding attackers.
- **Impersonation attacks:** Impersonation attacks are also known as “Spoofing”. In this attack, malicious node changes its IP address or MAC address in the active

packets and utilizes the location of another node. Through spoofing any insidious node can change network topology or seclude any node from rest of the network.

- **Attacks using fabrication**
    - Falsifying route error message: This sort of attack is more unmistakable in On-demand routing protocol, which utilizes path maintenance to recuperate the broken links. At whatever point a node changes its location, the nearest node sends an error message to alternate nodes this route is does not exist. By sending this kind of error message any node can be easily isolated.
    - Broadcast falsified routes: In this sort of attacks attacker misuse the routing data from the packet header and changes the routing path. This will change the route cache of neighboring node.
    - Routing table overflow attacks: In this sort of attack, the attacker endeavors to make routes to non-existing routes. If enough routes have been made, no new routes can be entered in the routing table.
- **Rushing attacks:** This kind of attack is applicable on On-Demand Routing protocol. In On-Demand routing protocol one and only route demand packet is sent to discover path to destination node [12]. This property is being used in rushing attacks by forwarding the RREQ Packets all the more as often as possible than alternate nodes so that the route including the attacker will be found [13].

1.2.1 Attacks against Routing

Routing is a standout amongst the most critical services in the network; in this manner it is additionally one of the primary focuses to which attackers lead their malicious practices [14]. In the mobile ad hoc networks, attacks against routing are generally classified into two categories: attacks on routing protocols and attacks on packet forwarding/delivery [15]. Attacks on routing protocols expect to obstruct the propagation of the routing data to the victim regardless of the possibility that there are a few routes from the victim to different destinations. Attacks on packet forwarding try to disturb the packet delivery along a predefined path.

The fundamental impacts brought by the attacks against routing protocols incorporate network partition, routing loop, resource hardship and route hijack. There are some attacks against routing that have been concentrated on and understood [16-19]:

- Imitating another node to spoof route message.
- Publicizing a false route metric to distort the topology.
- Sending a route message with wrong sequence number to smother other legitimate route messages.
- Flooding Route Discover exorbitantly as a DoS attack.
- Altering a Route Reply message to infuse a false route.
- Producing fake Route Error to disturb a working route.
- Stifling Route Error to misdirect others.

As a result of the mobility and always showing signs of change topology of the mobile ad hoc networks, it is exceptionally hard to validate all the route messages [15]. There are some more complex routing attacks, which include Wormhole attacks [20] [21] [22] [23], Rushing attacks [24] and Sybil attacks [25].

The second category of attacks against routing is attacks on packet forwarding/delivery, which are difficult to distinguish and counteracted [15]. There are two primary attack systems in this sort: one is self-centeredness, in which the malicious node specifically drops route messages that are expected to forward so as to spare it own battery power; the other is denial-of-service, in which the adversary conveys overpowering network traffic to the victim to fumes its battery power.

Aside from the attacks prevailing in MANETs, there are an variety of threats which are separated into two classifications [26, 27]: threats to network mechanism and threats to security mechanism. The following are few attacks based on routing mechanism [22]:

- **Black Hole**

  The black hole attack is briefly introduced in [29]. In this attack, a malicious node utilizes the routing protocol to advertise itself as having the shortest path to the node whose packets it needs to intercept.

- **Worm Hole**

  In a wormhole attack, two malicious collaborating nodes which are joined through a private network, can record packets at one location in the network and burrow them to another location through the private network and retransmits them into the network [30].

- **Routing Table Overflow**

  In a routing table overflow attack the attacker endeavors to make routes to nonexistent nodes. The objective is to make enough routes to keep new routes from being made or to overpower the protocol implementation. [31].

- **Sleep Deprivation**

  The sleep deprivation is briefly presented in [32]. Generally, this attack is useful just in ad hoc networks, where battery life is a basic parameter. Battery powered devices attempt to preserve energy by transmitting just when totally important. An attacker can endeavor to consume batteries by asking for routes, or by sending pointless packets to the node using, for example, a black hole attack.

- **Location Disclosure & Impersonation attacks**

A location disclosure attack can uncover something about the locations of nodes or the structure of the network. The information gained might reveal which different nodes are adjacent to the target, or the physical location of a node [33].

- **Denial of Service and Exhaustive attack**

These attacks are among the most noticeable sorts of attacks. In denial of service (DoS) attacks the adversary averts or forbids the typical use or management of network facilities or functionality. DoS attacks can be dispatched at any layer of an ad hoc network to fumes node resources [34].

1.3 Security techniques for MANET

To preserve the security of MANETs from attacks, a routing protocol must fulfill the accompanying arrangement of prerequisites, to guarantee appropriate working of the path from source to destination in vicinity of malicious nodes [26],

- Authorized nodes ought to perform route computation and discovery.
- Minimal introduction of network topology
- Detection of spoofed routing messages
- Detection of created routing messages
- Detection of changed routing messages
- Avoiding development of routing loops
- Prevent redirection of routes from shortest path

A number of secure routing protocols [35] have been as of late built up that fit in with the greater part of the prerequisites. These protocols utilize an assortment of cryptographic devices for securing the vulnerabilities in diverse routing protocols. The routing protocols for MANETs can be characterized into two primary classifications:

- Proactive or table-driven routing protocols
- Reactive or on-demand routing protocols

In table-driven nodes exchange routing data intermittently to keep up a steady route in every node for each other node in the network, as in Distance Vector Routing Protocol (SEAD), discussed in [36]. While in on-demand, a node starts a Route Request mechanism called Route Discovery at whatever point it needs to achieve a destination and the routes are made in like manner for single time use. The most widely recognized protocols that implement this mechanism are AODV (Ad hoc On Demand Vector routing) [37] and DSR (Dynamic Source Routing) [38]. The table-driven ad hoc routing methodology is like the connectionless methodology of sending packets, with no respect to when and how every now and again such routes are desired. It depends on a hidden routing table update mechanism that includes the constant propagation of routing information. This is not the case, in any case, for on-demand routing protocols. At the point when a node utilizing an on-demand protocol wants a route to another destination, it will need to hold up until such a route can be discovered. On the other hand, in light of the fact that routing data is continually spread and kept up in table-driven routing protocols, a route to each other node in the ad hoc network is constantly accessible, paying little respect to regardless of whether it is required. This sort of protocols keeps up new arrangements of destinations and their routes by intermittently conveying routing tables all through the network. This mechanism will be having several advantages and disadvantages. The main disadvantages of such mechanisms are respective amount of data for maintenance and slow reaction on restructuring and failures.

Then again, the reactive protocols discover a route on demand by flooding the network with route request packets. The main disadvantages of such algorithms are [39] high latency time in route finding and excessive flooding can lead to network clogging.

So the best approach to check the security [40] is Prevention, Detection and Reaction. Attempt to build the challenges for the attacker to enter the framework yet interruption free framework is not practical, so the identification segment assume an essential part to identify the attacker so that appropriate move can be made to maintain a strategic distance from steady adverse impacts.

Prevention can be accomplished by secure adhoc routing protocols that keep the attackers structure introducing off base routing states at different nodes. These protocols utilize distinctive cryptographic primitives,

- HMAC (Hashed Message authentication codes)
- Digital Signature
- Hash Chain

Once a malicious node is recognized sure activities are activated to shield the network from future attacks dispatched by this node the response segment is identified with the prevention action part in the security framework. Once numerous nodes in a nearby neighborhood have come to agreement that one of their neighbors is malicious, they aggregately revoke the certificate of the malicious node. The malicious node is isolated in the network as it can't take part in the routing or packet sending operations later on. The pathrater permits every node to keep up its own rating for each other node it thinks around. A node gradually increases the rating of well behaved nodes over time, however significantly diminishes the rating of a malicious node that is identified by its watchdog. In light of rating source dependably chooses the path with the highest average rating.

Message encryption is the science and specialty of changing a message into a hidden variant which no unauthorized individual can read, however which can be recouped in its unique structure by an expected beneficiary. The procedure of encryption and decryption are governed by keys, which are little measure of information utilized by the cryptographic algorithms. There are two sorts of encryption techniques: symmetric key and asymmetric key. Symmetric key cryptosystem utilizes the same key (the secret key) for encryption and decryption of a message, where as asymmetric key cryptosystems utilize one key (the public key) to encrypt a message and another key (the private key) to decrypt it. Public and private keys are connected in a manner that just the general public key can be utilized to encrypt messages and just the comparing private key can be utilized for decrypting reason. Indeed, if attacker includes a public key, it is basically difficult to retrieve the private key. Symmetric key algorithms are typically speedier to execute electronically than the asymmetric key algorithms.

The procedure of encryption just guarantees the confidentiality of the message being sent. Digital signature is a procedure by which one can accomplish the other security objectives like message trustworthiness, authentication and non-repudiation. In this, the sender utilizes a signing algorithm and its private key to sign the message. The message and the signature are sent to the recipient. The recipient gets the message and the signature and applies the confirming algorithm on the message-signature pair. The check algorithm requires a verification key, which is a public key gave by the signer, to confirm the document. After check if the outcome is genuine, the message is acknowledged; else, it is rejected. Hashing can be utilized for the digital signature prepare particularly when the message is long. In this, the message is gone through an algorithm called cryptographic hash function or one-way hash function before signing. It is an algorithm which makes a compacted picture of the message as a hash esteem (or message digest) which is normally much littler than the message and one of a kind to it [41]. Any change to the message will create an alternate hash result about notwithstanding when the same hash function is utilized. Both digital signature and encryption mechanisms are key-based methodologies. Key distribution and management is accordingly at the focal point of these mechanisms.

1.3.1 Intrusion Detection System

Intrusion detection is not another idea in the network research. Intrusion Detection System (or IDS) by and large identifies unwanted manipulations to systems. Each node in the mobile ad hoc networks takes an interest in the intrusion detection and reaction exercises by recognizing indications of intrusion conduct locally and freely, which are performed by the implicit IDS operators. Be that as it may, the neighboring nodes can impart their investigation results to one another and coordinate in a broader extent. The cooperation between nodes by and large happens when a sure node distinguishes a peculiarity however does not have enough confirmation to make sense of what sort of intrusion it fits in with. In this situation, the node that has recognized the peculiarity requires different nodes in the communication range to perform searches to their security logs in order to track the conceivable hints of the intruder [40].

There are different IDS systems which has some specific features, some of them are given blow

- Cluster based voting
- Neighbor-monitoring
- Trust building

1.3.2 Key Management

Cryptographic schemes, for example, digital signatures, are utilized to secure both routing information and data traffic. These schemes for the most part require a key management service. A public key framework is adopted as a result of its predominance in distributing keys, accomplishing integrity, non-repudiation, authenticate every node and establish a shared secret session key. In this, every node has a public/private key pair. Public keys can be distributed to different nodes, while private keys ought to be kept confidential to individual nodes. There is a trusted element called a certification authority (CA) for key management. The CA has a public/private key pair, with its public key known to every node [26].

1.3.3 Certification system

Although a large number of methods to detect various kinds of attacks have been developed for MANETs, only detecting and blocking attacks in each node is not enough to maintain network security because attackers can freely move and repeatedly launch attacks against different nodes. To reduce the damage from attacks, attackers must be immediately removed from the network after detection of the first attack; this can be achieved by using a certification system. In networks employing a certification system, nodes cannot communicate with each other without a valid certification. In other words, any attacker cannot exist in the network once its malicious behavior has been detected by others and its certification has been revoked accordingly by the system [42].

Shielding legitimate nodes from malicious attacks must be considered in MANETs. This is achievable through the utilization of a key management scheme which serves as a method for passing on trust in a public key base. These certificates are marked by the Certificate Authority (CA) of the network, which is a trusted outsider that is in charge of issuing and revoking certificates. The mechanism performed by the CA assumes a critical part in upgrading network security. It digitally signs a legitimate certificate for every node to guarantee that nodes can communicate with one another in the network. In such networks, a certificate revocation scheme which invalidates attackers' certificates is fundamental in keeping the network secured. An attacker's certificate can be effectively revoked by the CA if there are sufficient accusations showing that it is an attacker [43].

1.4 Key Management techniques

Cryptographic algorithms are security primitives that are generally utilized for the reasons of authentication, confidentiality, integrity, and non-repudiation. Most cryptographic systems require a hidden secure, vigorous, and effective key management framework. Key management is a focal piece of any protected communication and is the weakest purpose of framework security and the protocol design [44].

A cryptographic key is the core part of the cryptographic operations. In the event that the key was compromised, the encrypted information would be unveiled. The secrecy of the symmetric key and private key must always be guaranteed locally. The Key

Encryption Key (KEK) approach [45] could be utilized at nearby has to secure the secrecy of keys. To break the cycle (use key to encrypt the data, and use key to encrypt key) some non-cryptographic approaches need to be used, e.g. smart card, or biometric identity, such as fingerprint, etc.

Key distribution and key agreement over an unreliable channel are at high hazard and suffer from potential attacks. In the traditional digital envelop approach, a session key is created at one side and is encrypted by the public-key algorithm. At that point it is conveyed and recouped at the flip side. In the Diffie-Hellman (DH) scheme [45], the communication parties at both sides exchange some public information and produce a session key on both ends.

A few upgraded DH schemes have been invented to counter man-in-the-middle attacks. However, in MANETs, the lack of a central control facility, the limited computing resources, dynamic network topology, and the difficulty of network synchronization all contribute to the complexity of key management protocols.

Key integrity and ownership ought to be shielded from advanced key attacks. Digital signatures, hash functions, and the hash function based message authentication code (HMAC) [46] are techniques utilized for data authentication and/or integrity purposes. Likewise, the public key is secured by the public-key certificate, in which a trusted entity called the certification authority (CA) in PKI vouches for the binding of the public key with the owner's identity. In systems without a TTP, the public-key certificate is vouched for by peer nodes in a distributed way, for example, pretty good privacy (PGP) [45]. In some distributed methodologies, the system secret is distributed to a subset or the greater part of the network has in light of threshold cryptography. Clearly, a certificate can't demonstrate whether a entity is "good" or "bad". On the other hand, it can demonstrate ownership for key. Certificates are fundamentally utilized for key authentication.

A cryptographic key could be compromised or revealed after a sure time of use. Since the key ought to never again be usable after its disclosure, some mechanism is required to implement this rule. In PKI, this should be possible certainly or expressly. The certificate contains the lifetime of validity - it is not helpful after expiration. Be that as it may, at times, the private key could be revealed during the valid period, in which case the CA needs to revoke a certificate expressly and tell the network by adding it onto the certificate revocation list (CRL) to keep its use.

Key management for large dynamic groups is a troublesome issue as a result of scalability and security. Each time a new member is added or an old member is evicted from the group, the group key must be changed to ensure backward and forward security. Backward security means that new members cannot determine any past group key and discover the previous group communication messages. Forward security means that evicted members cannot determine any future group key and discover the subsequent group communication information. The group key management should also be able to resist against colluded members.

In MANET, key management can be classified into two kinds [47],

- The first one is based on a centralized or distributed trusted third party (TTP). The TTP is responsible for issuing, revoking, renewing, and providing keying material to nodes participating in the network where the key management process is performed using threshold cryptography. In the (*m; k*) threshold cryptography, a secret key is divided into *m* shares according to a random polynomial and kept by *m* legitimate nodes, which we call share holders. Later, a new node needs to collect *k* shares from the response of *k* nodes (among *m* nodes) based on Lagrange interpolation and generates the original secret key as a legitimate node.

- The second kind of key management is the self-organized key management schemes which can permit nodes to produce their own keying material, issue public-key certificates to other nodes in the network based on their knowledge. Certificates are put away and distributed by the nodes. Every node keeps up a neighborhood certificate repository that contains a limited number of certificates chose by the node as per a suitable algorithm. Public-key authentication is performed by means of chains of certificates.

1.4.1 Issues

The key management service must guarantee that the created keys are safely distributed to their owners. Any key that must be kept secret must be distributed so that confidentiality, validness and integrity are not damaged. For example at whatever point symmetric keys are connected, both or the greater part of the parties included must get the key safely. In public-key cryptography the key distribution mechanism must ensure that private keys are conveyed just to authorize parties. The distribution of public keys need not safeguard confidentiality, but rather the integrity and credibility of the keys should in any case be guaranteed. [48]

The traditional public key infrastructure (PKI)-supported approach works well in wired networks, but it is inadequate for the wireless ad hoc environment. In general, PKI-based approaches require a global trusted certificate authority (CA) to provide certificates for the nodes of the network, and the certificates can be verified using the CA's public key. However, ad hoc networks do not possess such an infrastructure characteristics. Even if the service node can be defined, maintaining such a centralized server and keeping its availability to all the nodes in such a dynamic network is not feasible. Moreover, the service node is prone to single point of failure, i.e., by only damaging the service node, the whole network would be paralyzed. Therefore , traditional key management schemes can not be applied directly, and a distributed key management approach is needed in securing ad hoc networks. [49]

1.5 Self-Organized key management

The entirely self-organized mobile ad hoc network is devoid of any kind of online or offline authority. The end-users generated this network in ad hoc mode. As the relationship among the users is not recognized priorly, the user does not share common keys with their nodes. Thus without depending on the common offline trusted authority (TTP), users have to build security relationships among themselves following the formation of network. The authority-based MANET holds up the applications which insist the utility of offline authority. The nodes related to the authority-based ad hoc networks include priorly established relationships when compared to fully self-organized ad hoc networks. The trusted authority is in charge of offering the cryptographic keying material and set of system parameters for every node before formation of the network. Each node will turn out to be self authority and further distributes the certificates to the nodes in transmission range following the formation of the network. [40]

Self-Organized key management technique is categorized into following two groups.

1) Virtual CA (Certificate Authority): This technique considers that there exists a certificate authority called trusted third party (TTP). Virtual CA offers high level

guarantees and does not necessitate warm-up time. Several virtual CA approaches employ threshold cryptography to securely distribute the CA's functionality over multiple nodes [50]. CA functionality is distributed in such a manner that an adversary must compromise a certain fraction of the key shares to compromise the virtual CA itself. At the same time, an end user need only access a subset of the distributed CA nodes to get certification services.

2) Web-of-Trust: This group does not necessitate CA and which reveals that it is more flexible. In web-of-trust, no hierarchy exists such as CA and users. Users issue public-key certificates to different users by self judgment. A user can depend on other user's public key in the event that it is ensured by his trusted user. [51]

1.5.1 Issues

- The main issue concerned with Virtual CA group is related to selection of CA and overcoming attacks in CA which is caused by malicious users. If there is any attack occurs to the certificate authority, who is the in charge of certificate exchange and revocation, a single point of failure will happen which will adversely affects the network resilience.
- Web-of-Trust approach is affected by recurrent communication and more memory spaces as it should gather public-key certificates in advance. Also it needs more time to gather all the certificates in the network due to that reason of exchanging the repository among moving users in periodical manner. Here all the nodes will come together and a hand shaking process will happen by

exchanging certificates. The delay for establishing web of trust will be a major concern while designing a system with this methodology.

1.6 Certificate chaining approach and its Issues

At the point when two nodes desires to interact in secured way, then they exchange public keys with one another utilizing the system that confirms and sign packet in every hop of the network. This strategy is termed as certificate chaining which includes the signing of the key exchange packets by every hop and verification of the signature by the following hop. The value of this methodology is that it allows the transmission of the public keys to the destination in the secured way. [52]

Essential Functions of Certificate Chaining Approach are,

- **Mitigating the Certificate and Private Key Compromise**

Upon compromise of the private key/certificate, the malicious attacker uses these certificates to start man-in-middle attacks. This can be averted utilizing the certificate chaining approach as a part of which each node ensures the trustworthiness of the certificate.

- **Setting Model for Future Extension**

For mitigating attacks on availability criteria, the certificate chaining approach assumes a noteworthy part. For instance, the trust management system utilizes certificate chaining approach for recognizing the flooding attacks.

For the most part the certificate chaining methodology is fitting for self-organized MANET that allows the users to create, gather, distribute and revoke their own public keys without the assistance of trusted authority [53]. The certificate chaining mechanism is having several limitations even though it achieves security and addresses several types of attacks. The authentication is the major concern addressed by the chaining approach, but the delay inquired while this chaining process will affect the performance of model.

1.6.1 Limitations of Certificate Chaining

The existing certificate chaining approaches exhibit the following limitations.

- No certification to the public keys authentication. Absolutely the certificate chaining among two nodes are conceivably not established.
- There is a necessity of broad time until the web-of-trust is set up among one another.
- The anticipated results in this scheme won't be exact since it is not in light of TTP. The nodes act as individual CA and subsequently the certificate chain will rely on upon the nodes honesty concerned with the formation. [54]

1.7 Certificate exchange and revocation

The way of MANET makes it defenseless against attacks. The challenges in MANET securities are confidentiality, integrity, legitimacy, accessibility and non reputability [5]. Out of these, legitimacy is the most central issue in the MANETs, so one of the generally utilized authentication mechanisms as a part of network is the certificate exchange/revocation.

Fundamental difficulties confronting ad hoc wireless networks are nature of service and security. One of the fundamental issues to think about in as a certificate-based scheme is the protected distribution of the public keys to every one of the nodes in the network [42]. The utilization of symmetric-key cryptography in certificate exchange/revocation has much littler computational overhead than that connected with digital certificates or threshold cryptography. Along these lines, the usage of threshold cryptography for the design of MANETs security schemes has produced some interest.

The certificate revocation revokes the certification of attackers in a brief timeframe with a little measure of working traffic furthermore it gives the authority to isolate any malicious nodes or recapture the nodes which turn up to its best state after any attack or failure [43]. Furthermore it ensures routing information in MANETs utilized as a part of emergency and salvage operations.

Here in the certificate revocation the nodes are grouped into ordinary nodes which are profoundly trusted, cautioned nodes with sketchy trust, and attacker nodes which can't be trusted. The CA keeps up both a Black List (BL) and a Warning List. At the point when the CA (certificate authority) gets an ADP (Attack Detection Packets) from an accuser, the accused node is viewed as an attacker and is immediately enlisted in the BL (Black List). The BL incorporates nodes which are named attackers and have had their certificates revoked [55].

The approached scheme can viably diminish the revocation time and communication overhead. Be that as it may, If there are a lot of ordinary nodes around the malicious nodes, the scheme will be exceptionally productive if not the proficiency degrades. To take care of this issue the approached scheme discharges the nodes from the WL (warning list) in view of a threshold with a specific end goal to build the number of ordinary nodes in the network and if any accusations found from any given node, then the nodes are weighted in view of the reliability of the accuser, the higher the dependability of a node, the more noteworthy the heaviness of its accusations, and the other way around. What's more, the node's certificate is revoked if the estimation of the whole of accusation weights against the given node is more prominent than a configurable threshold [56]. The certificate revocation has the following advantages,

- Becomes simple to decrease the attacks of the malicious nodes in the network by the assistance of certificate revocation method.
- When the node's Expiry time (ET) slipped by, the node broadcasts a renewal request packet to its neighbors in the certificate revocation system. [42]
- Cluster-based certificate revocation scheme contains the black list (BL), when the certificate authority gets an ADP (Attack Detection Packets) from an accuser, and immediately enrolled in the BL. Consequently this lessens the attacks in the network. [43]
- By the certificate revocation procedure the nodes will have the capacity to check the validity of the certificates, since they have the public keys of the CAs (certificate authority) which issued them.
- The certificate revocation scheme gives a system of measuring the reliability of MANETs nodes in view of the conduct profiles of. [55]

Also, the limitations of certificate revocation are:

- A node is designated with more than one key share by fusing excess into the network because of which there may be a redundancy issue in the network.
- When the number of malicious nodes is more, the CA (certificate authority) is no more capable to detect any new attackers in light of the fact that the greater part of the typical nodes in the network are presently recorded in the WL (warning list). [43]

- As the threshold esteem expands the mobility likewise increments which in turn diminishes the detection time drops furthermore when the threshold worth is less the nodes are allowed to release from the WL (warning list) until the threshold condition is fulfilled.
- A significant disadvantage is that with certificate lifetimes regularly measured in years, even a little revocation rate might lead to extensive records and don't scale exceptionally well. [56]
- Lack of secure limits makes the mobile ad hoc network defenseless to the attacks. Because of this mobile ad hoc network experiences all weather attacks.

1.8 Motivation

In MANET, when a node is compromised it tends to reveal the other node’s key information and corrupts the whole network. The scalable method of cryptographic key management (SMOCK) [57] proposes a method to deal with such node compromise attacks. Though this scheme achieves controllable resilience against node compromise by defining required benchmark resilience, it posses two major drawbacks: (i) Centralized offline servers for revoking/refreshing keys and create new keys for the new nodes. (ii) Increase in nodes ultimately increases the public-private key pairs. Also, most of the existing cryptographic techniques which are used to prevent attacks by unauthorized intruders become baseless during node capture attack.

When a cluster based key management scheme, which uses Adaptive Weighted Cluster (AWC) technique [58], is used to overcome the above two drawbacks, the following disadvantages occur. They are: (i) The details of the mobile nodes are always gathered before joining or starting the clustering process, which produces congestion and drain the cluster head (CH) and (ii) The overhead induced by Adaptive Weighted Cluster (AWC) technique is very high.

Furthermore, security in MANET is more challenging due to problems related to key exchange. It is necessary to secure the exchanges in MANETs for assuring the development of services in the network. The self organized MANET is visualized as a key communication technology enabler for application such as network driven fighting, catastrophe alleviation operations, crisis circumstances, shrewd transportation systems and so on.

The current key management system [59] to adapt to getting into misbehaving node does not keep users from making virtual identifiers or from taking the identity of individuals that don't participate in the network. Additionally investigation of more advanced load-balancing/data management schemes for public-key management is not taken care of. Thus, there is a necessity of solid self-certified key generation and certificate exchange mechanisms alongside some trusted model.

The certificate exchange method offers the nodes to authenticate themselves with the individuals in the network before they get joined and begins accessing the network resources. In order to upgrade the unwavering quality of certificate exchange protocol, existing certificate exchange protocol utilizes Multi-path Technique. The various paths utilized for certificate exchange ought to be secured and dependable.

An active attacker, then again, can change control packets or send inaccurate control packets to compromise the integrity of the routing protocol. For instance, an intruder node might broadcast its HELLO messages indicating neighbors that don't exist. A replay attack can likewise happen when an attacking node listens to packets and after that broadcasts the same packets. This attack is conceivable notwithstanding when the packets are encrypted. The issue of certificate revocation where there is no online access to trusted authorities is a challenging issue. In particular, the demonstration of ensuring the certificate exactness is all the more difficult as the malicious users can mishandle certification system.

Johann van der Merwe et al. [53] have proposed a Trustworthy key management for mobile ad hoc networks (AdHocTKM). They utilized threshold cryptography and certificate chaining technique that integrates the self-certified public keys and self-certificates to yield a key management service. They proposed a threshold self-certified public keying technique that allows cooperation among a single entity and a distributed authority for an implicit self-certified public key, without the authority gaining knowledge of the corresponding private key. Several issues are there for the existing key management techniques and enhancements are required for eliminating the same. A good key management scheme is essential for a secure routing protocol.

The following are the issues in the existing key management methods in MANETs,

- Lack of an authentic key generation and distribution.
- Trust of the intermediate node cannot be ensured.
- Node authentication mechanism is not addressed.
- Lack of a secured path selection.
- Improper handling of malicious nodes.
- Failure in identifying the invalid accusations during certificate revocation process.
- Attacks on trusted certificate authority.
- The increased number of public/private key pair usage.

- Delay in certificate exchange mechanism.
- Problems due to the mobility in cluster based approach.

The following are the need for establishing trust in MANETs,

- For addressing the above issues a secure key management mechanism is necessary.
- Key management and secure routing protocols are to be designed to shield against predefined active attacks furthermore to go about as a prevention system.
- The mobile node should be designed to support and defend against selectively misbehaving nodes or emerging attacks.
- To ensure the security of the intermediate nodes and for acting as a detection reaction system for MANETs a trust management mechanism is highly essential.

1.9 Problem Statement

The research problem formulated is to design a trust based key management framework for MANET.

1.10 Research Objectives

In order to design a trust based key management framework for MANET, we have identified the following research objectives.

- Develop a trusted certificate exchange and revocation mechanism

After the generation of public/private key pairs, multi-path certificate exchange technique is employed where public key of the nodes are certified by different nodes. The authentication is also performed mutually. A trust management mechanism should be incorporated for certificate revocation of the malicious nodes.

- Develop a trusted route discovery/path selection mechanism

The routes are selected based on different trusts calculated dynamically. Among the obtained routes, source selects a path which is having more certifiers of the destination node. After selecting the path, source and destination certifies their public keys each other.

- Develop a trust prediction mechanism of intermediate nodes

The nodes in the network are validated using different trust management technique. The trust is to be calculated based on the past performance and Trust Prediction Model.

- Develop a mechanism to isolate malicious nodes based on trust

Based on the calculated trust value, each node is going to be assessed with the threshold value and the isolation of node is performed by using certificate revocation list.

- Develop a cluster based combinatorial scheme/prediction based clustering

Here a cluster based combinatorial scheme / prediction based clustering is used for effective key management in MANET.

1.11 Thesis Overview

The rest of the thesis is organized into 7 chapters.

Chapter 2 – provides a systematic overview on existing key management techniques, certificate exchange mechanisms and certificate revocation mechanisms in view of both trust based, non trust situated in the writing are likewise given. A scientific classification and a correlation taking into account different criteria of the overviewed literature/methods are exhibited.

Chapter 3 – Presented the proposed method, Self-Organized Key Management for Trusted Certificate Exchange and Revocation for MANET. The proposed scheme is simulated and performance comparisons with the fundamental methodology are exhibited.

Chapter 4 – Discussion of Trust Prediction Model in light of accusations for certificate exchange and revocation with simulation results and execution correlations with existing methodology.

Chapter 5 – Discussion of the effect of M-OLSR Protocol in the proposed framework for certificate exchange and revocation in Mobile Ad Hoc Network with simulation results and performance comparisons with existing methodology.

Chapter 6 – Discussion of the cluster based combinatorial scheme for key management in Mobile Ad Hoc Networks with simulation results and performance comparisons with existing methodology.

Chapter 7 - Discussion of Prediction based clustering system for Distributed Hierarchical Key Management in Mobile Ad Hoc Networks with simulation results and performance comparisons with existing methodology

Chapter 8 – Concludes the thesis and specifies conceivable future research directions.

CHAPTER 2

LITERATURE REVIEW

CONTENTS

2.1 Introduction

Mobile Adhoc Network (MANET) is a self-sorting out network in which the nodes are allowed to move subjectively and arrange themselves [1]. These networks are utilized as a part of uses ranges from large-scale and exceptionally dynamic networks, to little and static networks [2]. Mobile adhoc networks are effortlessly influenced because of different attacks like active attacks, passive attacks and so on., When the attacker cause jamming, transmit fake routing information or disturb nodes from giving services, it is said to be active attacks. In passive attack, the attacker desires to pick up control access over the network. To minimize the attacks, one should remove the attackers immediately after detecting the first attack. This can be done by using a certification system.

2.1.1 Key Management

The presence of cryptographic keys acts as a proof of trustworthiness. Therefore, a proper key-management service is very much needed to ensure that the nodes are legitimate members of the network and are equipped with the necessary keys whenever needed. Key-management services are generally needed for application layer security and protection of the network layer. Key management schemes for the application layer can assume an already running network service. Schemes for the network layer routing information cannot. Keys are a prerequisite to bootstrap a protected network service. The classification is illustrated in figure 2.1.

The key management schemes are divided into two methods, contributory method and distributive method. In contributory methods, all the nodes participate together to achieve security. Here, there is no trusted third party for key generation and maintenance. In distributive method, a trusted third party will be there, who is the in charge of key generation and transport. We can ensure security by using certificate based and ID based approach. Different trust management mechanisms can be incorporated together with the certificate exchange for improving the security of the key management scheme.

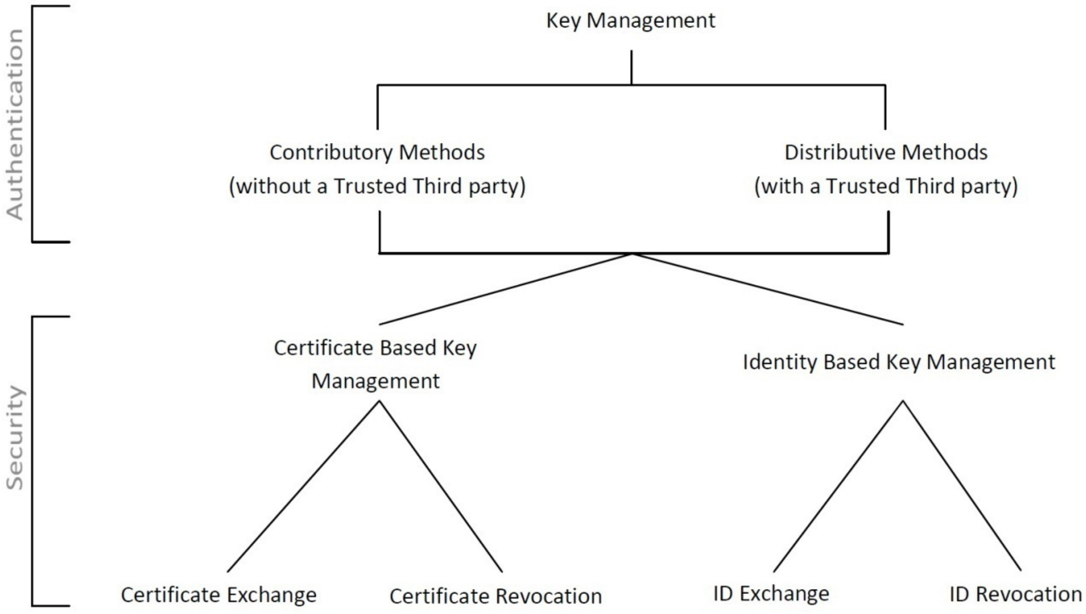
Authentication
Security
Key Management
Contributory Methods (without a Trusted Third party)
Distributive Methods (with a Trusted Third party)
Certificate Based Key Management
Identity Based Key Management
Certificate Exchange
Certificate Revocation
ID Exchange
ID Revocation

Fig 2.1 Key Management Mechanisms

2.1.2 Certificate Distribution and Exchange

During transmission, each node in the adhoc network produces a public/private key pair. As the node creates this key pair by its own, the node must authenticate with a few individuals in the network before joining and accessing the network resources. This authentication is performed by certificate exchange.

The certificates are produced by any outside resources, for example, server or Certificate Authority (CA). Certificate Authority (CA) is a trusted outsider in charge of issuing and revoking certificates [43]. CA signs a valid certificate digitally for every node. In the certificate exchange system, the nodes authenticate themselves with the individuals before they join and begin accessing the network resources [60]. In self organized environment, the certificate distribution and exchange plays a major role for ensuring availability and security.

2.1.3 Certificate Revocation

Among confidentiality, integrity, authenticity, availability and non-reputability [5], authenticity is the primary issue in MANET. Certificate Revocation is the commonly used authentication mechanisms. When the malicious behavior is detected, the certification of the attacker must be revoked [42]. Certificate revocation invalidates the certificates of the attacker for maintaining the network secured [43]. Here, the nodes are classified into normal nodes, which are exceedingly trusted, warned nodes with faulty trust, and attacker nodes, which can't be trusted. The CA keeps hint of Black List (BL) and a Warning List (WL). At the point when CA gets an Attack Detection Packets (ADP) from an accuser, the accused node is considered as an attacker and enrolled in BL [53].

The certificate revocation strategy has a few difficulties that are discussed as follows. At the point when a node is assigned with more than one key share, redundancy issue might happen in the network. Certificate lifetimes are typically measured in years. Here, a little revocation rate might lead to considerable records and can't scale well [56]. Attacks are brought on because of the nonattendance of secure limits in mobile adhoc network. Because of this, MANET experiences every all weather attack.

A cluster based certificate revocation scheme has limitation in certificate accusation and recovery mechanism where the number of nodes fit for charging malicious nodes diminishes after some time so that malicious nodes can never again be revoked in an auspicious way. Subsequently, these mechanisms are overviewed to design a system for improving the viability and proficiency of the scheme by utilizing a threshold based way to deal with restore a node's accusation capacity and to guarantee adequate typical nodes to accuse malicious nodes in MANETs [43].

This chapter makes a survey of various certificate exchange and revocation mechanisms. In the following sections, we have classified the key management methods into two types, namely, contributory and distributive methods, the certificate revocation mechanisms into two categories, namely voting-based mechanism and trust-based mechanism, and certificate distribution and exchange into trust-based and non-trust based methods. In section 2.5, the advantages and disadvantages are examined to compare the existing mechanisms in key management, certificate exchange and revocation. Finally, section 2.6 gives the summary of this survey work. Several existing works under these categories are discussed below.

2.2 Existing Key Management Methods

To accomplish the high security in MANET, diverse Key Management schemes are utilized. Utilizing and managing keys for security is a significant task in MANET due its energy compelled operations, limited physical security, variable limit links and dynamic topology. Secure communication is a critical test in ad hoc networks. The untrustworthy wireless medium in MANET is a risk for Secure Data Transmission. The communication in mobile ad hoc networks contains two stages, the route discovery and the data transmission. In an adverse situation, both Phases are defenseless against an assortment of attacks, one way to counter security attacks would be to cryptographically ensure and authenticate all control and data traffic. Key management is an essential piece of any protected communication structure. Most secure communication protocols depend on a protected, hearty, and productive key management system. The key is a bit of data information for cryptography algorithms. Ensuring security in key generation and exchange plays a major role in MANET.

Key management methods can be classified into two types, namely

(i) Contributory Methods

(ii) Distributive Methods

2.2.1 Contributory Methods

Contributory schemes are described by the absence of a trusted third party in charge of generation and distribution of the cryptographic keys. Instead, all conveying parties coordinate to establish (i.e., "agree" upon) a secret symmetric key. The number of members ranges from two parties (establishing a pair wise key) to numerous parties (establishing a group key). Despite the fact that not as a matter of course designed on account of ad hoc networks, naturally the contributory methodology of coordinated effort and self-association might appear to fit the way of ad hoc networks. A portion of the contributory schemes concentrated on here depend on a centralized entity, others don't.

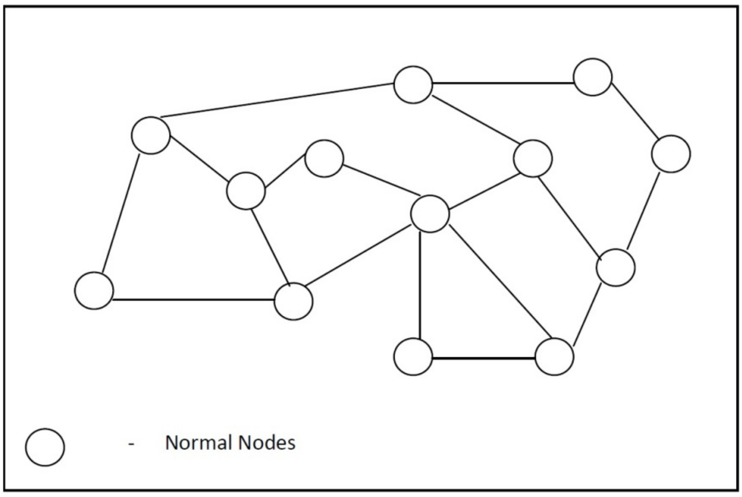

Fig 2.2 Contributory approach in key management

Figure 2.2 shows the general network architecture of a contributory key management approach. It does not use a trusted third party in it. In contributory schemes, the key is a result of a collaborative effort of more nodes.

The question of key exchange was one of the first problems addressed by a cryptographic protocol. This was prior to the invention of public key cryptography. The Diffie-Hellman key agreement protocol [61] was the first practical method for establishing a shared secret over an unsecured communication channel. The point is to agree on a key that two parties can use for a symmetric encryption, in such a way that an eavesdropper cannot obtain the key. It applies to just two parties. Assurance of routing messages with pairwise keys requires different signatures for every conceivable beneficiary that scales incompletely. This scheme does not tackle MIM (Man in Middle Attack) vulnerability and not reach out to more than two parties.

As an extension to Diffie-Hellman, the group key agreement scheme is proposed by Burmester and Desmedt (B-D) [62]. Dependable multicasting is hard in wired networks, and considerably additionally challenging in ad hoc networks. Changes in-group enrollment requires a restart of the key-agreement technique. In an ad hoc network with moving nodes, there is no probability for establishment of a group key by B-D and maintenance of later changes in-group participation. Group changes can bring about delay and interruption. B-D likewise demands an already running routing protocol or stand out hop neighbors. This implies, the key-agreement schemes rely on upon an already established routing foundation. In any case, the framework can't be established before the keys have been set up.

For reducing the complexity of existing algorithms Hypercube and Octopus (H&O) [63], has proposed a method which minimizes the number of rounds by arranging the nodes in a hypercube. H&O contains two protocols, to be specific, Hypercube and Octopus. Hypercube expect the number of members is a force of 2. Octopus extends the

Hypercube to permit a self-assertive number of nodes. H&O is helpless against MIM attacks as authentication is absent. Byzantine or defective nodes might block fruitful key agreement. Changes in group enrollment require rekeying. It is left for the nodes to choose when re-keying is required. H&O depends on a basic communication system to offer a reliable node-requesting perspective to all group individuals. H&O is unsuitable for network layer security in ad hoc networks.

As an extension to the H&O, a password Authenticated mechanism is proposed [64] which is the stand out of the contributory systems designed for ad hoc networks. It is often referred to as the H&O method stretched out with secret key authentication. This method expect that all the legal members get a secret word offline. During the pair wise D-H key agreements of the H&O protocols, the nodes must demonstrate the learning of the secret key. The secret word is utilized to encrypt the public quality and a starting test in a test reaction protocol. This scheme duplicates the number of messages and expands the computational many-sided quality when contrasted with H&O. It solves the vulnerability of H&O to MIM attacks at the cost of scalability. The scheme acquires the lacks of H&O in regards to the trustworthiness of an already established communication base and node-requesting scheme. In this manner, it is not fitting for network layer security in mobile ad hoc networks.

CLIQUES [65] is another protocol suite that extends the generic D-H protocol to bolster the dynamic group operations. A group controller synchronizing the key agreement system is required. This scheme is computationally proficient. The designers did not consider the security properties like authentication while concentrating on group changes. CLIQUES depends upon the solid multicast and the accessibility of a steady perspective of node ordering.

2.2.2 Distributive Methods

Distributive schemes include one or more trusted entities and involve both public key systems and symmetric systems. Really ad hoc networks require the trusted entity to be established during network initialization.

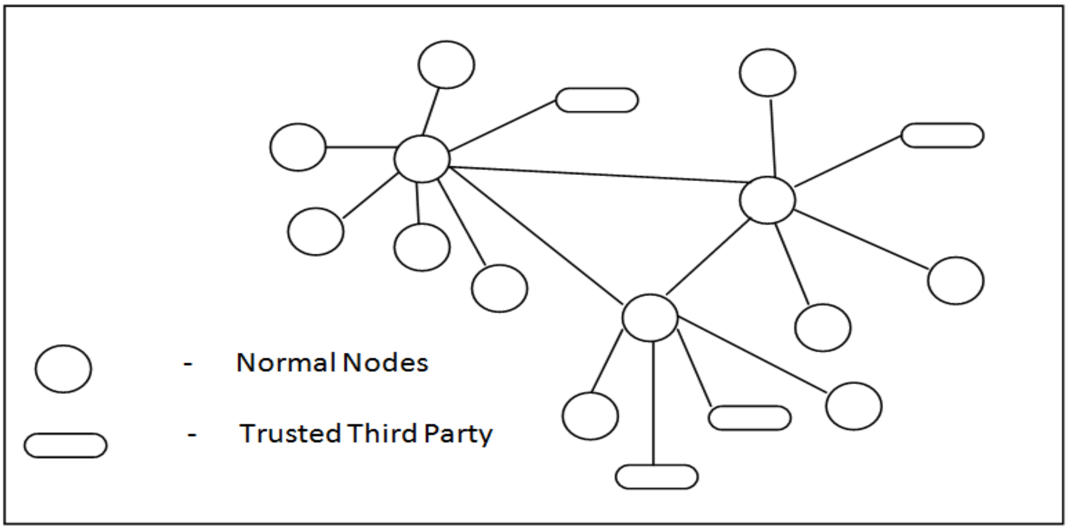

Fig 2.3 Distributive Approach in key management

Figure 2.3 shows the existence of the trusted third party in the distributive scheme. Distributive schemes may be centralized, however can likewise be distributed. In the latter, every node produces a key and tries to distribute it to others.

By Yi, Naldurg, and Kravets et al [66] have proposed Mobile Certificate Authority (MOCA) which is a decentralized key management scheme. In this scheme, a certificate service is distributed to MOCA nodes. MOCA nodes are picked in view of heterogeneity if the nodes are physically more secure and computationally all the more effective. It presents a practical key management framework for ad hoc wireless networks using PKI. It clarifies the necessity and the problem of providing a PKI framework for ad hoc network. It also identifies the requirements for such a framework and provides some insights into the configuration of such security services in ad hoc networks.

In Secure and Efficient Key Management (SEKM) [67], it is easier for a node to request service from a well maintained group rather than from multiple "independent" service providers, which may be spread in large area. The servers of MOCA structure a multicast group to effectively update the secret shares and certificates. A node broadcasts a certificate request to the CA server group. The server that first gets the request, produces an partial signature, and advances the request to additional servers. The additional servers are utilized for redundancy as a part of case some are lost or debased. SEKM does not discuss about how a server can let it know is the first to get the refresh request and start the forwarding. SEKM has the same components as MOCA. The required number of servers still must be reached, and the partial signatures returned. Be that as it may, this system neglects to work under different server groups in large networks including partitioned network.

Ubiquitous Security Support (UBIQ) [68] is a fully distributed threshold CA scheme. Like MOCA, and SEKM, UBIQ also relies on a threshold signature system with a secret sharing of the private CA key. In addition, all nodes get a share of the private CA key. Every entity holds a secret share and different entities in a nearby neighborhood together give complete services. Confined certification schemes are utilized to empower pervasive services. This scheme operates well at the network with breakages. The solution is completely decentralized to operate in a large-scale network. The limitation in the rescue operations scenario is the possible requirement of human involvement.

Composite Key Management (COMP) [50] joins partially distributed threshold CA of MOCA together with the self organized key management scheme. Certificate-chaining method and expanded accessibility of CA is the specialty of the proposed scheme. Results shows the effectiveness of composite key management under stressful scenarios where the existing approaches fail to work. COMP assumes a level of trust transitivity. This technique uses the highest confidence certificate chain that does not fully exploit the information contained in a certification graph. COMP can provide flexible, modular, and adaptive key management services for mobile ad hoc networks.

Efficient and robust key management scheme [69] is a hierarchical scheme based on threshold cryptography to address both security and efficiency issues of key management and certification service in MANET. Key management scheme provides various parts of MANET the flexibility of selecting appropriate security configurations, according to the risks faced. It also offers the adaptivity to cope with rapidly changing environments. This technique maintains a large number of nodes and issue certificates with different levels of assurance. This scheme can isolate the compromised regions and provide stronger protection to the Global Secret Key (GSK) compared to flat-structured schemes.

Scalable means of cryptographic key management (SMOCK) [57] is an independent public key-management scheme, which obtains irrelevant communication overhead for authentication, and offers greatest service accessibility. Here a combinatorial design of public-private key pairs is made which furnishes every node with additional security of more than one key pair to encrypt and decrypt messages. A combination method is used and a pair of public keys is used to encrypt the data, the pair

ID is going to be assigned during the initialization process. Before the communication, the ID to be forwarded to the destination and depends upon this, the encryption process will be carried out. At the receiver end the decryption will be done by using the private keys of the corresponding public keys. The information about which public key should use for encryption is received from the pair ID communicated by the two parties. Here each node is going to store all the public keys and a set of private keys (depends upon the combination). This reduces the number of keys stored in each node and enhances the traditional public/private pair key generation. This configuration helps in acquiring higher security as far as nodes and storage space. The scheme likewise accomplishes controllable resilience against node compromise by characterizing required benchmark resilience. Be that as it may, this technique has two noteworthy disadvantages. It utilizes the centralized offline servers for revoking/invigorating keys and to make new keys for the new nodes. The second downside is that, the increment in nodes eventually expands the public-private key pairs (yet relatively in low extent than traditional methodology).

ID-based multiple secrets key management (IMKM) [70] protocol is a comprehensive solution for inter and intra-cluster key management, including key revocation, key update, and group key agreement. IMKM requires that cluster heads (CHs) participate in the construction of the key, in order to establish a (t, n) threshold sharing of the master secret key. The advantages of using a distributed method lie in its efficiency and flexibility in updating CHs' share keys. This method does not require the exchange or signing of any additional messages when the network is within security tolerance. To address security concerns, it updates the CHs' share keys when CHs are evicted and the number of revoked CHs reaches a predefined threshold.

2.3 Existing Certificate Distribution and Exchange Methods

The certificate exchange system offers the nodes to authenticate themselves with the individuals in the network before they some assistance with getting joined and begin another communication. Nodes with a valid certificate can participate in a communication. Initially the certificates will be distributed to all normal nodes. The newly joining nodes will get the certificate after an initial verification.

Certificate distribution and exchange method is broadly classified into two types. They are

(i) Trust based methods

(ii) Non-trust based methods

2.3.1 Trust Based Methods

It requires a trust based mechanism to exchange the certificates among nodes. It is to authenticate the nodes before allocating the certificate to it. Mostly it carries a centralized architecture, in order to monitor the participant nodes in the network. Distributed approach can also be applied based on the routing scenario. A trusted authority is required to monitor the certificate distribution as well as allocation. That authority will collect the trust value of a node before allocating the certificate to it. The nodes with a good amount of trust value are considered as normal nodes.

This trust may be calculated dynamically during the path selection of the protocol. Different paths may be obtained by using the existing algorithms and the trust of each path is going to be calculated. A threshold trust value is fixed to identify the normal nodes. If any node does not acquire a minimum threshold trust value, then that node is considered as malicious node and steps are initiated for isolating that particular node.

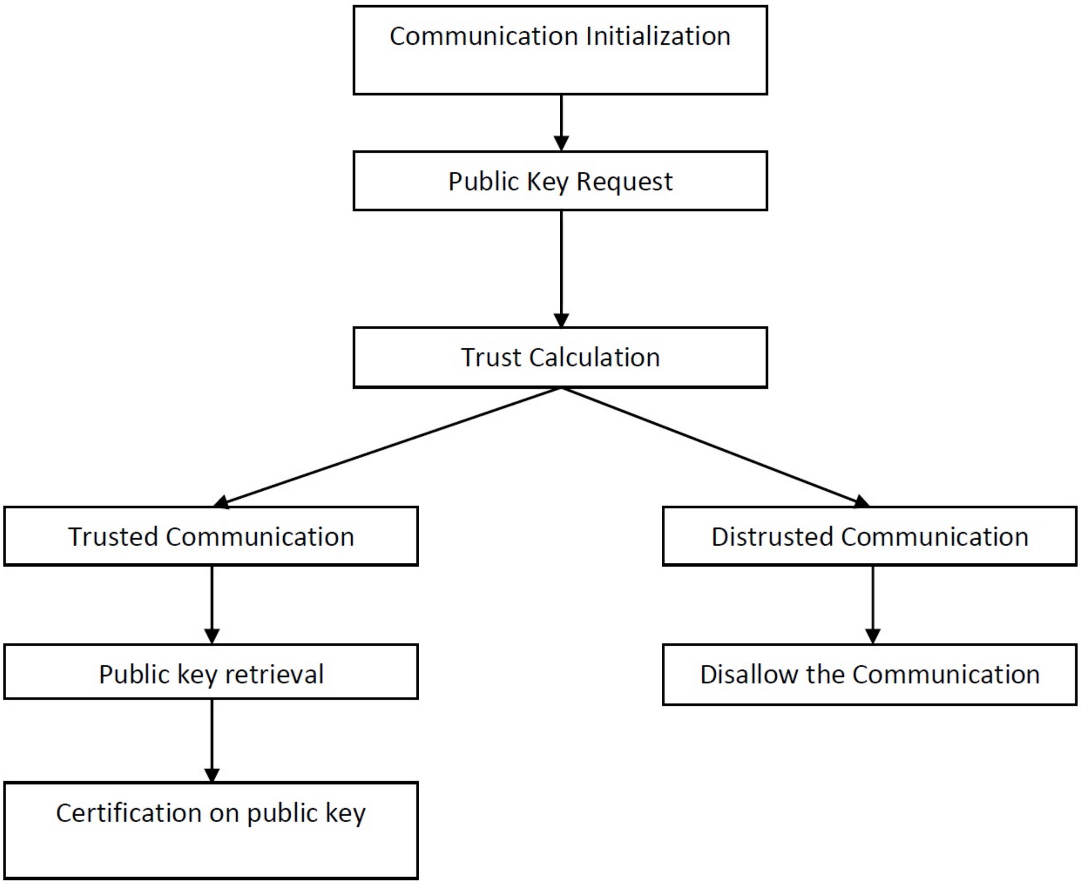

Fig 2.4 Certificate Exchange –Trust Based approach

Figure 2.4 illustrates the trust based certificate exchange mechanism. The trust value will be calculated before issuing certificates. If the trust value is minimal, then the communication will be marked as distrusted communication.

The design focuses on a truly ad hoc networking environment where geographical size of the network, numbers of network members and mobility of the members is all unknown before deployment. The process of development of the protocol and the application to system design are developed to assure information security and potential evidential retention for forensic purposes. Threshold encryption key management is utilized and simulation results show that security within the network can be increased by requiring more servers to collaborate to produce a certificate for a new member, or by requiring a higher trust threshold along the certificate request chain. The cost such as time, processor use and battery use in mobile devices of information management is also considered here. When the number of servers increases, longer certificate chain is required. An increase in the chain length may increase the likelihood of the chain encountering a malicious node. Therefore, a failure occurs in the network [71].

A key exchange protocol [72] integrated with a routing protocol is lightweight, efficient and alleviates the routing-security interdependency cycle. This routing protocol establishes a path between source and destination by reactive routing protocols. Initially, a route request message is broadcasted to discover the route to the destination. The key exchange protocol uses this approach to retrieve the public keys of the nodes. Source node floods a certificate request to find a certificate of a public key. The receiving node replies either by the target node or by an intermediate node. It tries to distribute spurious certificates and causes routing disruption. Multi-path certificate exchange and trust-based certification are used for providing robustness and reliability. It is resistant to isolated attack launched by malicious nodes that may introduce spurious certificates. When sufficient level of trust exists among some nodes before the network deployment, it performs well against cooperative attacks. When MPKTV value is 0.5, any reply is accepted. Further due to the increase in number of attackers, the probability of accepting corrupted public key increases.

In an efficient public key management scheme [73] for fully self-organized mobile ad hoc networks, the operations of creating, storing, distributing, and revoking nodes' public keys are carried out locally by the nodes themselves. This method improves the process of building the local certificate repositories of nodes. An authentication solution based on the web of trust concept is combined with an element of routing based on the multipoint relay concept to introduce the optimized link state routing protocol. It offers good tradeoff among security, overhead and flexibility, considerable reduction in resource consumption whereas performing the certificate verification process. However, this mechanism increases the delay of issued certificates.

Ad hoc Trust Framework (ATF) [74] support ADOPT's robustness and efficiency. ADOPT is deployed as a trust-aware application that provides feedback to ATF. ATF calculates the trustworthiness of the functions of the peer nodes. ATF also helps ADOPT for improving its performance by quickly locating valid certificate status information. The TrustSpan algorithm reduces the overhead produced by ATF. It can also identify and use the trusted routes to propagate the sensitive information like accusations of the third party. ATF adds limited overhead when compared to its efficiency in noticing and isolating the malicious and selfish nodes. As it can quickly locate a genuine response by using ATF's information, the reliability of ADOPT is increased. Lastly, the optimized caching policies based on mobility, connectivity, capacity, and trustworthiness is discussed. However, the overhead due to the requested recommendations is largely increased, when the number of invalid responses is maximized.

To obtain resilience and efficient path discovery among Peer-to-Peer trusted PKI's [75] for issuing entities, a virtual hierarchical architecture is used. As the execution time is less, it is suitable for MANETs. A virtual hierarchy in a Peer-to-Peer PKI is established based on the trustworthiness of the participating neighbors. The upward approach is used to build the hierarchical structure, that is, from the leaves to the root. In addition, it does not require to issue new certificates among PKI entities, facilitates the certification path discovery process and the maximum path length can be adapted to the characteristics of the users with limited processing and storage capacity. However, the effectiveness of this technique is only analyzed theoretically. There is no practical implementation to prove the results.

2.3.2 Non-trust based Methods

The trust value of the node is used to make the communication authentic and secure. So we are helpless to perform the certificate exchange process without the assistance of a trust mechanism. But, it is possible to eliminate the overhead of trust computation by replacing it with some alternate techniques. Such techniques must guarantee a secure communication in all means. ID based exchange as well as certificate chaining approaches are the good alternatives to the trust based certificate exchange. Here the process itself offers the security without a trust management mechanism.

Figure 2.5 illustrates the non trust based certificate exchange mechanism. Here the public keys can collect either from an existing certifier (intermediate node) or from the destination itself.

A certificate based authentication mechanism [76] is used to contradict the effect of black hole attack. Nodes authenticate each other by issuing certificates to neighboring nodes and generating public key without the need of any online-centralized authority. After the route establishment process of On Demand Multicast Routing Protocol (ODMRP), this scheme has certification phase and authentication phase. All certificates issued are stored in the repositories of the issuer and the certificate subject. The certificates are exchanged between the neighboring nodes periodically. By utilizing this, nodes collect certificates in their repositories at a low communication cost in light of the fact that the exchanges are performed locally in one hop. This mechanism can be applied for securing the network from other routing attacks by changing the security parameters in accordance with the nature of the attacks. When the number of attackers is increased, the packet delivery ratio is greatly reduced due to loss of packets in the black hole nodes.

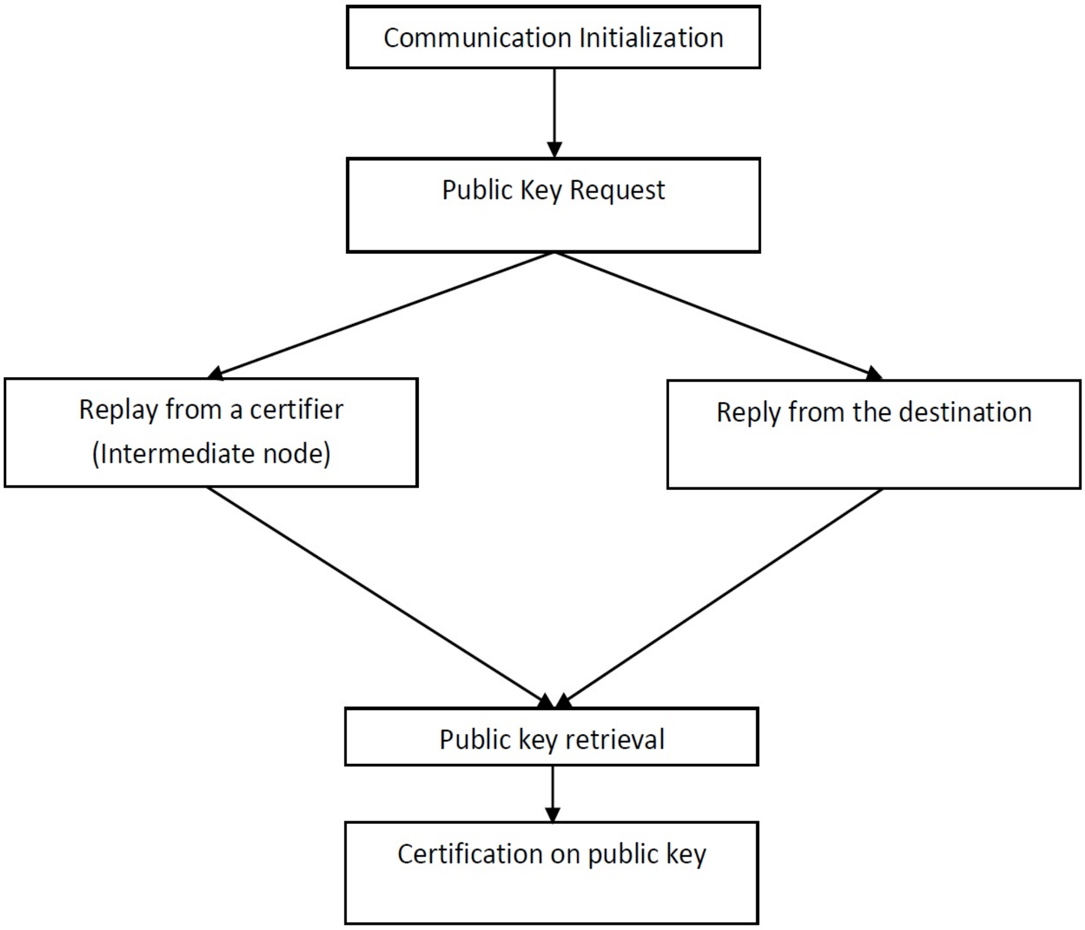
Communication Initialization
Public Key Request
Replay from a certifier (Intermediate node)
Reply from the destination
Public key retrieval
Certification on public key

Fig 2.5 Certificate Exchange – Non-Trust Based approach

A completely self-organized public-key management system [59] permits users to create their public private key pairs, issue certificates and perform authentication paying little heed to the network partitions and with no centralized services. The two users in a mobile ad hoc network can perform key authentication construct just in light of their nearby information, regardless of the fact that security is performed in a self-organized way. With the vicinity of neighborhood repository development algorithm and a little communication overhead, it accomplishes superior on an extensive variety of certificate graphs. Nodes can exploit mobility to encourage authentication and to distinguish conflicting and false certificates. This methodology does not require any trusted authority, not even in the system initialization stage. Nonetheless, the detection and the determination of conflicting certificates are excluded in this mechanism. A few parameters like delay, throughput are not discussed in this mechanism.

A novel key distribution scheme [77] for MANETs exploits the routing base to viably chain peer nodes together. Keying material propagates along these virtual chains through a message handing-off mechanism. It results in a key distribution scheme with low implementation complexity, preferably suited for stationary ad hoc networks and MANETs with low to high mobility. It utilizes mobility as a guide to fuel the rate of bootstrapping the routing security, yet rather than existing schemes does not get to be subject to mobility. The key spread happens totally on-demand; security affiliations are just established as required by the routing protocol. The communication and computational overhead of this methodology has immaterial effect on network execution. In any case, when the mobility builds, route failure might happen.

Here, the node mobility is considered and the major improvements related to the number of elected cluster heads are given for creating the PKI council. Here, the certification authority functions are distributed for a reduced set of mobile nodes to serve for keys management. For selecting the council members, the two solutions are made. 1)

A set of nodes that makes the council of PKI is designed. The members are randomly chosen. They remain the same until the network exists. Hence, it is considered as fixed-members architecture. 2) The network is organized as clusters and each cluster has a cluster-head. The council of PKI is made up of the cluster heads present in the network form. Hence, it is denoted as cluster-based architecture. These two architectures are compared and concluded that the clustered architecture provides a better result and is well suited to the dynamic environment. However, it is not focused on aspects of the choice of the threshold parameter values and the council member's number [48].

The Tseng model and the Capkun model [78] are merged to improve the overall performance and offer less overhead with high security. This model authenticates the nodes via 4G services to facilitate the communication after the nodes becoming the part of certificate chain-based groups. The nodes have logins and passwords through server before joining the MANET that forms the basis of verifiable identities for getting certificates. Nodes generate private and public keys by built-in PKI techniques. For issuing certificate, the servers sign these public keys. Each node needs a certificate for joining the certificate chain based group. A long chain of certificates leads to group formation. The session last until the expiry time of either of the node's certificate. This process saves the bandwidth and increases efficiency. However, when the time period is very limited, it leads to extra burden of entity verification messages. When the time period is large, security related issues might occur. This produces lack of performance of the protocol in terms of delay and security related parameters. This tradeoff becomes a major drawback of this approach.

2.4 Existing Certificate Revocation Methods

In MANET, the certificates are used to make the communication more secure. Only the nodes with a valid certificate can participate in the communication. Thus the certificate of an attacker node has to be revoked once it listed as a malicious node.

The certificate revocation methods are classified based on the mechanisms used to revoke the certificates. They are

(i) Voting Based Method

(ii) Trust Based Method

2.4.1 Voting based Methods

The voting based mechanism is defined as the means of revoking a malicious attacker's certificate through votes from valid neighboring nodes. Voting-based scheme, allows all nodes in the network to vote (accuse) against malicious nodes. Each node monitors the behavior of its neighbors. An authorized node will collect the accusations from the normal nodes. The accusations will be treated as a valid one only if the authorized node is able to get same accusation from different nodes. The malicious nodes can accuse legitimate nodes. Thus, it is important to identify valid accusations. In the case of invalid accusations, the accuser node will be treated as a malicious node. The nodes can vote with variable weight. The weight is ascertained from a node's dependability which is received from its past conduct. The higher its unwavering quality is, the more prominent its weight will be. The certificate of a suspicious node can be revoked when the total of the weights of the votes against the node comes to or surpasses a predefined threshold. Thusly, the exactness of certificate revocation can be moved forward. Then again, since all nodes are required to participate during each vote, the communication overhead required to exchange voting information is entirely high. That will thusly expand the time expected to revoke the certificate.

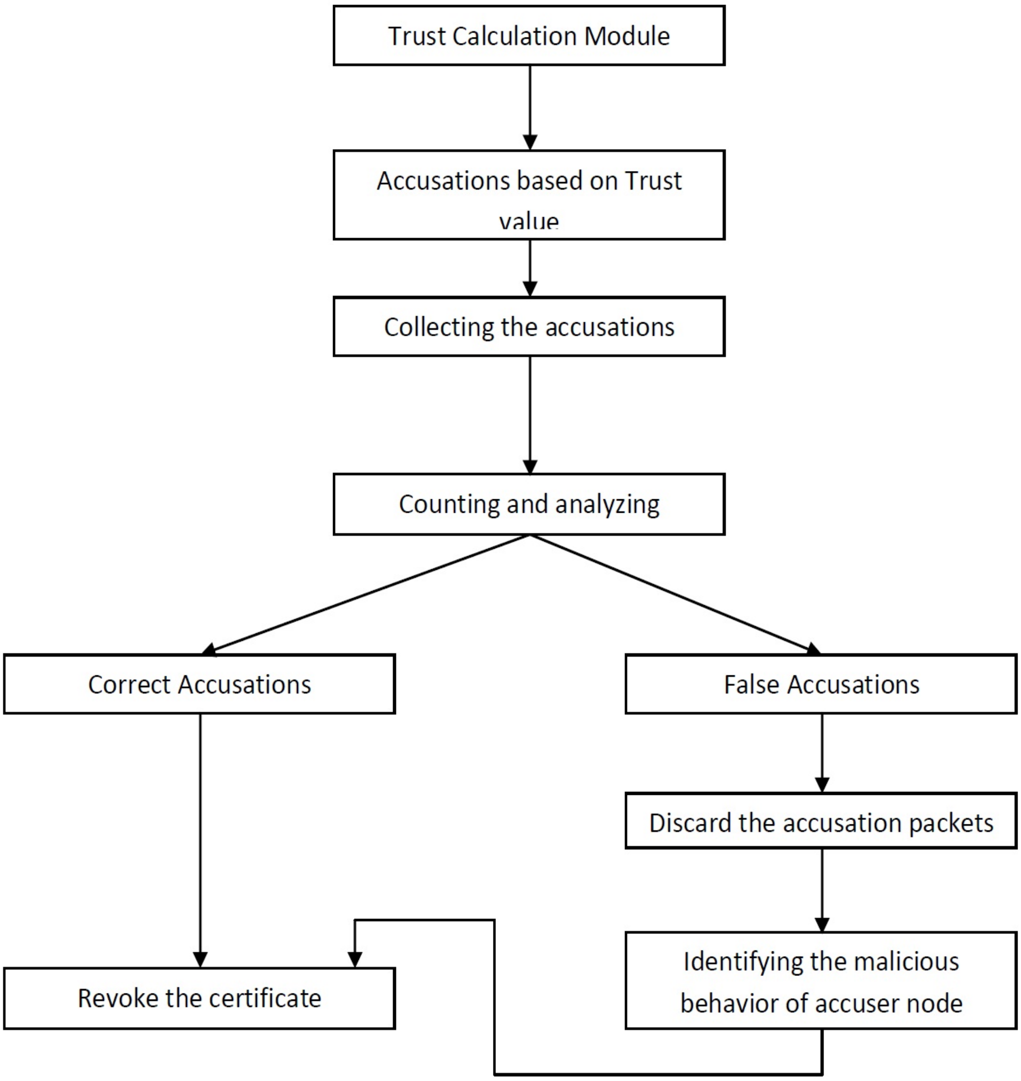
Trust Calculation Module
Accusations based on Trust value
Collecting the accusations
Counting and analyzing
Correct Accusations
False Accusations
Discard the accusation packets
Revoke the certificate
Identifying the malicious behavior of accuser node

Fig 2.6 Certificate Revocation – Voting Based approach

Figure 2.6 illustrates the voting based certificate revocation mechanism. Here the votes can treat as the accusations from normal nodes. An accusation will consider as a valid accusation, only if the collecting node is able to get enough number of accusations about a same node. Certificate of the accused node will be revoked in the case of valid accusation. Otherwise the accuser node will be marked as a malicious node.

In this voting-based scheme, all nodes in the network are permitted to vote. Like URSA, no CA exists in the network, and instead every node screens the conduct of its neighbors. Here, the nodes vote with variable weight. The weight is ascertained from a dependability of the node that is gotten from its past conduct. The higher unwavering quality can bring about more prominent weight. The certificate of a suspicious node can be revoked when the entirety of the weights of the votes against the node comes to or surpasses a predefined threshold. Consequently, the exactness of certificate revocation strategy can be progressed. Then again, since all nodes are required to participate during each vote, the communication overhead required to exchange voting information is entirely high, along these lines expanding the time expected to revoke the certificate [55].

URSA [79] utilizes a voting-based mechanism to expel nodes. The certificates of recently joining nodes are issued by their neighbors. The certificate of an attacker is revoked in light of votes from its neighbors. In URSA, every node per-frames one-hop checking, and exchanges screen information with its neighboring nodes. At the point when the number of negative votes surpasses a foreordained number, the certificate of the accused node will be revoked. Since nodes can't communicate with others without valid certificates, revoking the certificate of a voted node infers disconnection of that node from network exercises. Deciding the threshold, on the other hand, remains a test. If it is much larger than the network degree, nodes that dispatch attacks can't be revoked and can be progressively continued speaking with different nodes. Another basic issue is that URSA does not address false accusations from malicious nodes. While URSA does not require any exceptional gear, for example, Certificate Authorities (CA), the operational cost is still high.

T.R.Panke [80] has improved their proposed clustering-based certificate revocation scheme which considers quick certificate revocation. At the point when the number of normal nodes gradually diminished, a threshold-based mechanism is utilized to restore the accusation function of nodes in the WL. Despite the fact that a centralized CA oversees certificates for every one of the nodes in the network, cluster development is decentralized and performed autonomously. Every CM has a place with two unique clusters so as to give robustness against changes in topology because of mobility. The protocol does not address security analysis part.

The procedure of revoking malicious Certificates is discussed to revoke a malicious attacker's certificate, there is a need to consider three stages accusing, verifying, and notifying. The revocation procedure begins at detecting the presence of attacks from the attacker node. The false accusation of a malicious node against a legitimate node to the CA will degrade the accuracy and robustness of our scheme. To address this problem, one of the aims of constructing clusters is to enable the CH to detect false accusation and restore the falsely accused node within its cluster. This certificate revocation process tends to take a long time in detecting malicious node [81].

By ordering nodes into clusters, this scheme permits every Cluster Head (CH) to distinguish false accusation by a Cluster Member (CM) inside of the cluster. Node clustering gives a way to alleviate false accusations. CHs always screen their CMs and look for false accusations. Just normal nodes having high unwavering quality are permitted to wind up a CH. Nodes with the exception of CHs join the two unique clusters of which CHs exist in the transmission range of them. By developing such clusters, each CH can know about false accusations against any CMs since each CH knows which CM executes attacks or not, on account of the greater part of the attacks by a CM can be identified by any node, obviously including the CH, inside of the transmission range of the CM.

The motivation behind why every node with the exception of CH has a place with two distinct clusters is to diminish the danger of having no CH because of dynamic node development. To keep up clusters, CH and CMs as often as possible affirm their presence by trading messages, i.e., the CH intermittently broadcasts CH Hello packets to the CMs inside of its transmission range, and every CM answers to the CH with the CM Hello packet.

A newly joining node becomes CH at a constant rate. A node, which has decided not to become a CH itself, will look for other CH nodes in the area. If there are more than two CHs near the node, it will attempt to join two of these clusters by randomly selecting two of their CHs and sending each of them a CM Hello packet. Otherwise, the joining node declares itself as a CH and broadcasts CH Hello packets. When a CM leaves the cluster, it needs to invoke a similar procedure to find out new CHs. If the CM receives no CH Hello packet from its CH for a certain period of time, the CM considers itself having departed from the cluster, and tries to find and join a new cluster [42]. This certification revocation method outperforms all the other existing revocation mechanisms since it uses a vindication capability mechanism to identify the malicious behavior of a node.

2.4.2 Trust based Methods

Establishing trust among the nodes in an ad hoc network is critical from a security point of view as the nodes go about as self securing devices to ensure themselves with no infrastructural support. Besides, nodes route packets to a destination through intermediate nodes. Thus, nodes need confirmation to depend on different nodes in the network and this is accomplished by establishing trust relationships among the nodes. The certificate of a node will be revoked in view of the trust value. A good trust model ought to be adaptable; it ought to have a broad adversary control mechanism and ought to establish trust among the nodes. On the other hand, establishing trust relationships among the nodes includes communication overheads. A good trust based model must have the capacity to address the issues identified with the communication overhead. Dependable nodes ought to get by in the network, while nodes which don't give a good nature of service or are malicious ought to be distinguished rapidly and expelled from the network.

Figure 2.7 illustrates the trust based certificate revocation mechanism. Here the revocation happens purely based on trust value. The revocation can be done either by a neighbor node or by a trusted third party.

An enhanced distributed certificate authority scheme [5] give data integrity by making the network more secure from both inside and outside attacks. It makes utilization of Shamir's secret sharing scheme along to a redundancy strategy to backing certificate renewal and revocation. In this strategy, the malicious nodes are recognized by the monitoring so as to trust mechanism the conduct hop by hop. This scheme builds the integrity of the network and gives the network nodes to be more mobile. Be that as it may, this method lessens the general throughput significantly.

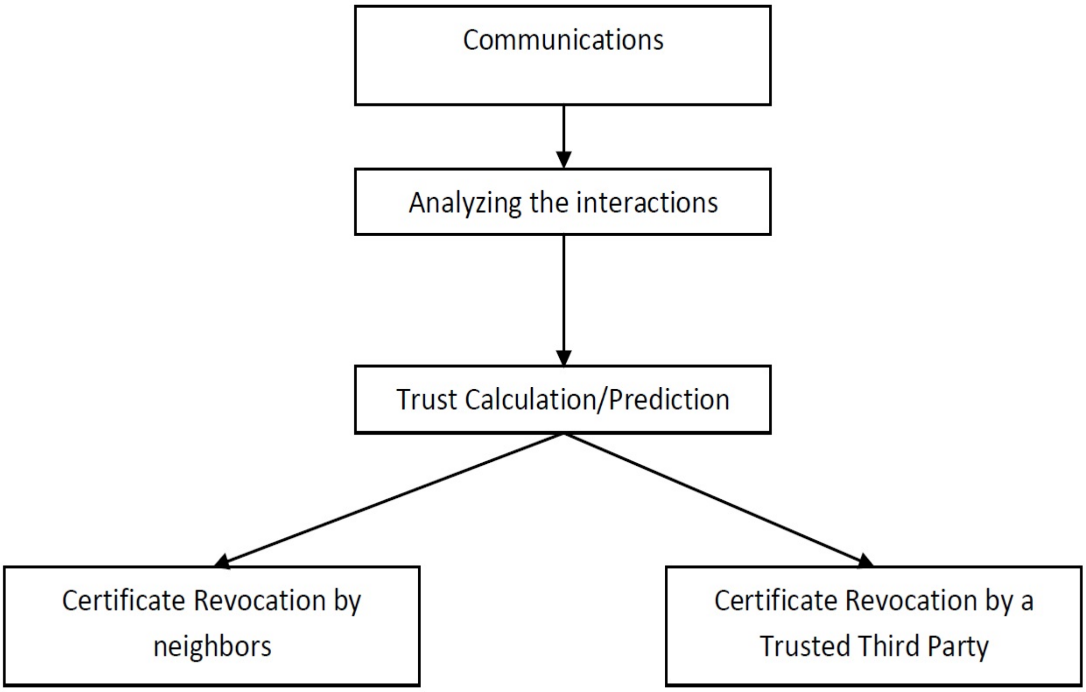
Communications
Analyzing the interactions
Trust Calculation/Prediction
Certificate Revocation by neighbors
Certificate Revocation by a Trusted Third Party

Fig 2.7 Certificate Revocation – Trust Based approach

A distributed trust model for certificate revocation [82] permits trust to be constructed after some time as the number of interactions between nodes increment. Besides, trust in a node is characterized as far as its potential for maliciousness and nature of the service it gives. Trust in nodes where there is practically no history of interactions is dictated by proposals from different nodes. If the nodes are narrow minded, trust is acquired by an exchange of portfolios. The rate of helpful communication is enhanced by presenting setting particular trust connections among the nodes. The intrusion detection system of every node joined with the trust associations with alternate nodes viably removes malicious nodes in the network. Thus, the malicious nodes are viably uprooted and the expulsion of honest nodes from the network is minimized. A few parameters like accessibility and nature of service are excluded for the determination of trust. Bayesian model causes execution overheads.

This revocation scheme [56] is utilized for the distribution of revocation information in mobile ad hoc networks (MANETs). This scheme can be executed in conjunction with the transcendent routing protocols in ad hoc networks. At that point it gives an itemized security examination somewhat taking into account the utilization of formal techniques. It basically concentrates on revocation of certificates and IDs used to secure routing information in MANETs utilized as a part of crisis and rescue operations. The revocation records should thusly be particular to the network. They are established with the guide of trusted gateways reporting the identity of the nodes to a focal trusted entity. To minimize overhead, the revocation records are distributed alongside the routing messages. It offers limited robustness to fluctuating network availability.

In the decentralized suicide based approach [83], while the certificate revocation can be done with an accusation, the certificate of the accused node and the accuser's certificate are revoked. No less than one node needs to give up itself to expel an attacker from the network. This system significantly lessens both the time required to remove a node and the communication overhead of the certificate revocation methodology. On the other hand, inferable from its suicide-based procedure, the use of this methodology is limited. This scheme does not give a mechanism to separate erroneously accused legitimate nodes from appropriately accused malicious nodes. Thusly, the exactness is degraded.

DICTATE[84] utilizes a number of CA to productively perform the publication and revocation of certificates. CA screens node conduct so as to recognize attacks and share the certificate information with one another. In the event that a CA distinguishes a malicious node, the certificate of the node is revoked by the CA and its information is shared among other CA, there by isolating the node from the network. Then again, the deployment of an adequate number of CA is not a simple task in MANETs.

Cooperation Of Nodes: Fairness In Dynamic Ad-hoc Network (CONFIDANT) is a reputation system going for adapting to trouble making in MANET [85] [86] [87]. The thought is to recognize the got into misbehaved nodes and confine them from communication by not utilizing them for routing and sending and by not permitting the acted mischievously nodes to utilize it to forward packets. CONFIDANT remains for Cooperation Of Nodes: Fairness In Dynamic Ad-hoc Network. It normally fills in as an augmentation to on demand routing protocols.

2.5 Classification of existing Key Management & Certificate Exchange/Revocation Methods

S. No	Approach	Category	Metrics	Advantages	Disadvantages
Key Management					
1	New Directions in Cryptography [17]	Contributory Method	Packet Overhead	Basic approach for key management	Vulnerable to MIM attack
2	A Secure and Efficient Conference Key Distribution System [62]	Contributory Method	Packet Overhead	Secure against any type of attack and solve DL problem	Not used in real time applications
3	Communication Complexity of Group Key Distribution [63]	Contributory Method	Number of exchange, number of rounds and no. of messages	Minimizes the number of rounds	It cannot be adopted for network layer security. Authentication is not discussed.
4	Key Agreement in Adhoc Networks [64]	Contributory Method	Number of rounds	Eliminates the MIM attacks	Not appropriate for network layer security in MANET
5	CLIQUES: A New Approach to Group Key Agreement [65]	Contributory Method	Lifespan and security of particular key	Computationally efficient	Do not provide authentication.

6	MOCA: Mobile Certificate Authority for Wireless Ad Hoc Networks [66]	Distributive Method	Packet overhead, certificate delay, number of CREP	Less vulnerable to attacks	Bandwidth wastage
7	Secure and Efficient Key Management in Mobile Ad Hoc Networks [67]	Distributive Method	Average Number of Hops, average delay, server rate, convergence time and Computation time	Efficiently update the secret shares and certificates	Fails to work under multiple server groups in large networks
8	Providing Robust and Ubiquitous Security Support for Mobile Ad-Hoc Networks [68]	Distributive Method	Success ratio, average delay, and average number of failures	Decentralized to operate in large network	Need human intervention
9	Composite Key Management for Ad Hoc Networks [50]	Distributive Method	Success ratio, mobility, crypto Threshold, communication overhead, average confidence value	Flexible, modular, and adaptive key management service	Has limited robustness and scalability
10	Efficient and Robust Key Management for Large Mobile Ad Hoc Networks [69]	Distributive Method	success rate, error rate, average retries of certification renewals	Maintain large number of nodes. Adaptive to cope with rapidly changing network	Susceptible to some attacks
11	SMOCK: A scalable method of cryptographic key management for mission critical wireless ad-hoc networks [57]	Distributive Method	Memory storage, Delay	Reduction in the number of key pairs.	Increase in nodes ultimately increases the public-private key pairs.
12	Securing Cluster-Based Ad Hoc Networks with Distributed Authorities [70]	Distributive Method	Memory storage, Delay	Address the problem of varying link qualities.	Bandwidth overhead

Certificate Distribution and Exchange					
13	Black Hole Attack Prevention in Multicast Routing Protocols using Certificate Chaining [76]	Non-trust based Scheme	Packet delivery ratio, Average end-to-end delay	Secure the network from routing attacks by varying the security parameters based on attacks.	The packet delivery ratio is reduced due to loss of packets, When the attackers increases,
14	Secure Key Deployment and Exchange Protocol for MANET Information Management [71]	Trust based Scheme	Mobility, Speed, certificate issuance ratio	Security inside the network can be increased with more servers	Increased servers can cause longer certificate chain that may increase the likelihood of the chain encountering a malicious node. A failure occurs in the network.
15	A Multi-Path Certification Protocol [72]	Trust based Scheme	Valid PK acceptance rate, Corrupted PK acceptance rate, Delay	Resistant to isolated attack by malicious nodes that may introduce spurious certificates	When MPKTV value is 0.5, any reply is accepted. Due to more attackers, the probability of accepting corrupted public key increases.
16	Self-Organized Public-Key Management [88]	Non-trust based Scheme	Average shortest path, mobility	Mobility facilitate authentication and detect the inconsistent and false certificates.	Parameters like delay, throughput are not discussed
17	Key Distribution based on Message Relaying [77]	Non-trust based Scheme	Packet delivery ratio, packet end-to-end delay	Communication and computational overhead has less impact on performance.	When the mobility increases, route failure may occur
18	Efficient Public Key Certificate Management [73]	Trust based Scheme	Number of nodes in graph, rate of certificates in repository, clustering	Good tradeoff among security, overhead and flexibility, considerable	The delay of issued certificates are increased

			coefficient, maximum length of chains, time consumption	reduction in resource consumption	
19	Integrating a Trust Framework with a Distributed Certificate Validation Scheme [74]	Trust based Scheme	Communication overhead, detection time, number of responses and trusted paths, Cerberus function gain, roundtrip delay	ATF adds limited overhead for detecting and isolating malicious and selfish nodes	Overhead due to the requested recommendations is increased, when there is more invalid responses
20	Certificate Path Discovery by Constructing Virtual Hierarchy to Administer Trust Relationship using Peer to Peer PKI [75]	Trust based Scheme	-	Formation of trust relationship to establish the hierarchy and not to issue new certificates or to adjust the trust points.	No practical implementation to prove the results.
21	A Distributed Key Management Scheme using council architecture [48]	Non-trust based Scheme	Delivery delay of a certificate, certificate delivery fraction, response time of PKI	Delay is reduced with increased efficiency	Aspects of the choice of the threshold parameter values and the council member's number are not focused
22	Certificate Chain based Authentication using 4th Generation Technologies [78]	Non-trust based Scheme	Cost of external messages, number of nodes	Saves the bandwidth and increases efficiency.	Limited time period causes extra burden of entity verification messages. Larger time period causes security related issues
Certificate Revocation Methods					
23	A localized certificate revocation scheme [55]	voting-based scheme	Communication overhead, communication complexity	Accuracy of certificate revocation technique can be improved	As all nodes participate in voting, communication overhead required is high, thus increasing the time

					needed to revoke the certificate.
24	URSA: Ubiquitous and Robust Access Control [79]	voting-based scheme	Success ratio, number, average number of retries, average delay, normalized overhead	Effectively enforces access control in the highly dynamic network	Operational cost is still high.
25	Certificate Revocation to Cope with False Accusations [42]	voting-based scheme	Ratio of revoked attackers, control packet traffic, node density, attack success count,	It promptly removes the attackers with low operating traffic even in the presence of malicious nodes carrying out false accusations.	The reliability and accuracy is low when compared to other schemes
26	Clustering Based Certificate Revocation Scheme for Malicious node [43]	voting-based scheme	-	Rapidly revoke attacker's certificates and recover falsely accused certificates.	There is no security analysis.
27	Cluster-Based Certificate Revocation with Vindication Capability [81]	voting-based scheme	Revocation time, number of warned nodes, node speed, node density	Effective and efficient to guarantee secure communications	Detection of malicious node takes longer time
28	Suicide for the Common Good: a New Strategy for Credential Revocation in Self-Organizing Systems [83]	-	Overhead, delay	Has fully decentralized, low communication and storage overhead, fast removal of misbehaving nodes	There is no mechanism to differentiate falsely accused legitimate nodes from properly accused malicious nodes. So, accuracy is degraded.
29	DICTATE: Distributed Certification Authority With Probabilistic Freshness [84]	-	Freshness degree, speed, freshness load	Robust against various attacks in MANET	Deployment of a sufficient number of CA is a complex task
30	An Enhanced Distributed Certificate	Trust based scheme	Control overhead, average end-to-	Integrity of the network increases	Overall throughput is considerably

	Authority Scheme for Authentication [5]		end delay, average packet delivery ratio		reduced
31	Trust Model for Certificate Revocation [82]	Trust based scheme	Number of false positives, speed, number of true positives	Malicious nodes are effectively removed and the removal of innocent nodes from the network is minimized.	Parameters like availability and quality of service are not considered in the derivation of trust. Bayesian model causes performance overheads.
32	Scalable Revocation in Hybrid Ad Hoc Networks The SHARL Scheme [56]	Trust based scheme	Load, delay, overhead	As the revocation lists are distributed along with the routing messages, the overhead is minimized	It offers limited robustness for varying network connectivity.

2.6 Design Considerations of the Thesis

A secure framework suitable for military and tactical applications based on trust is to be developed. The trusted system should address the following applications,

- Key management
- Secure source routing
- Malicious node detection.

While designing such a system, the following mechanisms should be considered in the framework.

- Independent key generation mechanisms
- Key distribution based on certificate/ID exchange
- Reducing the number of public/private key pair used
- Battery power and memory of a mobile node should be preserved

- Path establishment based on trust
- Trust calculation based on the connectivity and past performance of a node
- Future trust prediction of a node
- Malicious node detection
- Isolation of a malicious node

The proposed framework should possess the following performance criteria.

- Overall network resilience should be improved
- Detection rate of a malicious node to be improved
- The above improvements should be done without affecting the basic performance factors like delay, throughput, packet drop and overhead.

2.7 Summary

In this section, several key management schemes, clustering techniques used for effective key management and certificate Revocation schemes have been discussed. Some of them are having some disadvantages. To overcome them, effective techniques must be developed. In this research thesis, some limitations stated below are considered and overcome by proposing effective techniques.

The scalable method of cryptographic key management (SMOCK) [57] proposes a method to deal with such node compromise attacks. It has certain main drawbacks such as over dependent on centralized server and increase in key-pair when node increases.

The existing Weighted Clustering Algorithm (WCA) [89] induces increased overhead. Also, the details of the mobile nodes are gathered always before joining or starting the clustering process, which produces Congestion and drain the CH. From these issues, it is known that an effective clustering technique for key management must be developed to reduce the overhead and congestion.

The proposed technique [59] to cope with misbehaving node does not prevent users from creating virtual identifiers or from stealing the identity of people that do not participate in the network. Also, exploration of more sophisticated load-balancing/data management schemes for public-key management is not handled. In [51], the author has not discussed the authentication parameters.

In [53], the proposed technique lags certificate revocation methodology. Also, authentication parameters are discussed in detail. In [60], the authors have only assumed that every node in a MANET first generates a public/private key pair. From these existing works, it is known that strong self-certified key generation and certificate exchange mechanisms along with some trusted model must be developed.

The existing certificate chaining mechanisms did not provide assurance to the public keys authentication. Certainly the certificate chaining among two nodes are possibly not established. There is a requirement of extensive time until the web-of-trust is set up among each other. The predicted results in this scheme will not be precise since it is not based on TTP. The nodes acts as individual CA and consequently the certificate chain will depend on the nodes honesty concerned with the formation.

Even though the existing certificate revocation schemes have handled the situation that no on-line access to trusted authorities in MANET. They provide a large amount of operational traffic and a long revocation time, because the opinion of every node in the network is needed for each node to decide whether to revoke the certificate of the malicious node or not. Also, most of the existing revocation schemes did not provide trust management along with certificate revocation mechanism. Therefore, it is evident that a certificate revocation mechanism integrated with the underlying routing protocol and the trust Management must be developed.

A security framework based on trust is proposed to address the above issues. The framework consists of trusted certificate exchange and revocation, trusted route discovery/path selection, trust prediction of intermediate node and isolation of malicious node in various multipath/clustered routing protocols.

CHAPTER 3

SELF-ORGANIZED KEY MANAGEMENT FOR TRUSTED CERTIFICATE EXCHANGE AND REVOCATION IN MANET

CONTENTS

3.1 Overview

A security framework based on trust for effective key management in MANET is proposed. The proposed architecture consists of the coordinator node, servers and ordinary mobile nodes. The coordinator node acts as a mediator for transmitting the message among the servers and mobile nodes. Each node generates its own public/private key pairs using server-signed public keying technique [53]. The coordinator node helps in generating the publicly-recoverable public key for any node N_i without the knowledge of the subsequent private key. The coordinator node acts as a distributed trusted authority. It combines the shares of (t+1) servers for computing signature parameter. The nodes in the network are validated using the trust management mechanism. The trust value is computed using the Eigen Vector Reputation Centrality [90]. Then multi-path certificate exchange technique is employed where public key of the nodes are certified by different nodes [60]. This certificate exchange system offers the nodes to authenticate themselves with the individuals in the network before they some assistance with getting joined and begins accessing the network resources. As a consequence of numerous autonomous certifications, the confidence assigned to the certificates is higher. At the point when the source node needs to forward the data packet to D, it disposes of the malicious nodes in that path and sidesteps the data through different nodes in substitute chose path towards the destination utilizing multipath technique and source performs the certificate revocation process for safeguarding against the malicious nodes. The proposed scheme is simulated and performance comparisons with the fundamental methodology are displayed.

3.2 Proposed Architecture

Our architecture consists of the coordinator node, servers and normal mobile nodes. The coordinator node will act as the distributed trusted authority.

Fig 3.1 shows the proposed architecture of the self-organized key management technique of the MANET. It includes the ordinary nodes (N_1, N_2, ….. N_{10}), where N_4 is chosen as the coordinator node (N_c). There are 4 servers { z_1, $z_{2\ ...}$ z_n }. The coordinator acts as a mediator node for transmitting messages from normal mobile nodes to the servers.

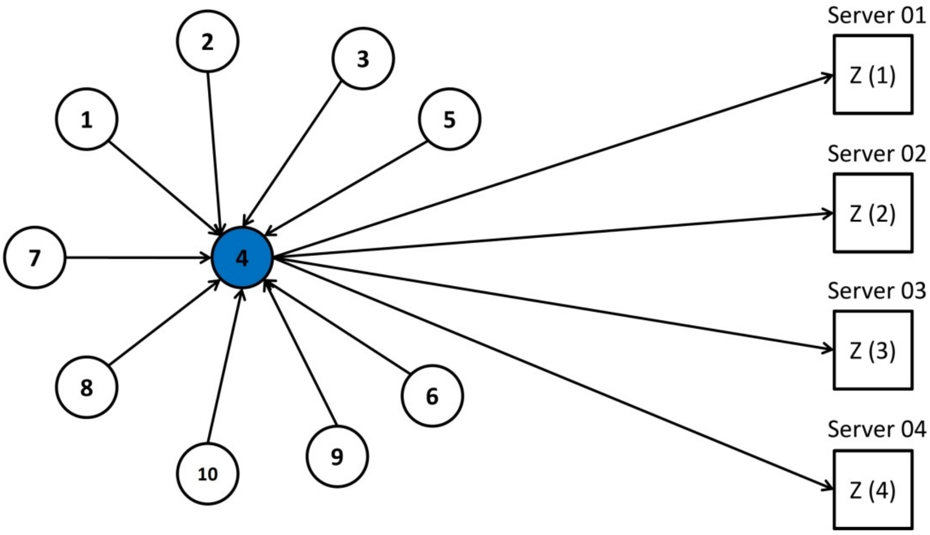

Fig 3.1 Architecture of the Self-Organized Key Management Technique

The proposed technique includes four phases which are described in the following section.

3.2.1 Creation of Public/Private Key Pairs

This phase involves the generation of public/private key pairs (K_{pu}, K_{pr}) using server-signed public keying technique. This technique allows the users to generate their own public/private key pairs. The system consists of n servers (z_1, $z_{2\ ...}$ z_n) and a trusted coordinator node. The coordinator node sends a secret value to the corresponding node which requires a public/private key generation. The receiver node sends back a newly generated secret value for getting the signature parameter.

The coordinator node acts as a distributed trusted authority. It combines the shares of (t+1) servers for computing signature parameter. Consider that coordinator node selects the prime numbers x, λ with $\lambda / x-1$, a generator d of a multiplicative subgroup of Z_x^* with order λ. Let h [ ] denotes a one-way hash function and N_i denotes any node in the network.

The coordinator node publishes x, λ, d and h.

The steps involves in the generation of public/private key pairs are as follows.

Step 1: After selecting its random number $k'_{Ni} \in_R Z_\lambda^*$, coordinator node N_c computes the secret value (using Eqn-3.1).

$$C'_{Ni} = d^{(k'_{Ni})} \tag{3.1}$$

This computed secret value C'_{Ni} is transmitted to N_i.

Step 2: After selecting its random number $q \in_R Z_\lambda^*$, N_i computes (since λ is publically known, Z_λ^* can be manipulated and any element from Z_λ^* can be selected, here we say, q can be selected for N_i) the secret value (using Eq-3.2).

$$\begin{aligned} C_{Ni} &= C'_{Ni} d^q \\ &= d^{(k'_{Ni})} d^q \text{, using 3.1} \\ &= d^{(k'_{Ni})+q} \end{aligned} \tag{3.2}$$

After the above computation of secret value C_{Ni}, N_i then transmits its own ID and secret (ID_{Ni}, C_{Ni}) again to N_c.

(Here the selected q is only known to N_i)

Step 3: The coordinator node N_c forwards the (ID_{Ni}, C_{Ni}) to each server.

Step 4: Each server computes the hash function of (ID$_{Ni}$, C$_{Ni}$), which is represented as h[(ID$_{Ni}$, C$_{Ni}$)]. Then it computes its threshold signature, i.e. Sign$_i$ [Z$_i$, h[(ID$_{Ni}$, C$_{Ni}$)] to coordinator node.

Step 5: Coordinator node collects all the *t+1* shares from the servers and computes the signature parameter Sign$_{Ni}$ and forwards both Sign$_{Ni}$ and Sign$_i$ [h[(ID$_{Ni}$, C$_{Ni}$)] to N$_i$.

$$\text{Sign}_{Ni} = \text{h }[(\text{ID}_{Ni}, \text{C}_{Ni})] + \text{k'}_{Ni} \qquad (3.3)$$

Step 6: Then N$_i$ computes the private key k$_{pr}$

$$\text{k}_{pr} = \text{Sign}_{Ni} + \text{q} \qquad (3.4)$$

(once N$_i$ computes k$_{pr}$ = Sign$_{Ni}$ + q as q is known only by N$_i$, it becomes secret to others)

The tuple (C$_{Ni}$, k$_{pr}$) can be viewed as the signature of the DTA on ID$_{Ni}$.

Step 7: After verifying the signature, N$_i$ computes the corresponding public key k$_{pu}$ and publishes C$_{Ni}$ and ID$_{Ni}$. k$_{pu}$ of N$_i$ is publicly verified by decrypting Sign$_l$ [h[(ID$_{Ni}$, C$_{Ni}$)] using the public key of the coordinator node; comparing the decrypted hash value to [h (ID$_{Ni}$, C$_{Ni}$)] and evaluates Eq-3.5

$$\begin{aligned} k_{pu} &= d^{(k_{pr})} \\ &= d^{sign_{N_i}+q}\text{, using (3.4)} \\ &= d^{sign_{N_i}} \cdot d^{q} \\ &= d^{h[ID_{Ni},C_{Ni}]+k'_{Ni}} \cdot d^{q}\text{, using (3.3)} \\ &= d^{h[ID_{Ni},C_{Ni}]} \cdot d^{k'_{Ni}+q} \\ &= d^{h[ID_{Ni},C_{Ni}]} \cdot C_{Ni}\text{, using (3.2)} \end{aligned}$$

$$\text{Thus, } k_{pu} = d^{(k_{pr})} = d^{h[ID_{Ni},C_{Ni}]} \cdot C_{Ni} \qquad (3.5)$$

The above approach assists the users to fully control the security settings of the system.

3.2.2 Trust Management Mechanism

The trust value is going to be calculated by using two factors, the centrality score and the recent satisfaction index termed as combined trust (CT_{ij}). If the trust value is higher than a threshold minimum then the node is going to be treated as a trusted node, otherwise it will be considered as malicious node. The trust value will lie between zero and one.

3.2.2.1 Computation of Centrality Score

The trust management mechanism is employed in order to validate the nodes in the network. The trust value is computed using the Eigen Vector Reputation Centrality Mechanism [90]. Each node deployed in the network computes the Eigen vector centrality (EVC_i) of its neighbors for exhibiting the reputation and level of confidence on each neighbor.

Let n_i and n_j be the adjacent nodes. The centrality for the ith node is relative to the total of the scores of all nodes that are linked to it.

$$EVC_i = \frac{1}{\delta}\sum_{j=1}^{n} R_{ij} EVC_j \qquad (3.6)$$

Where, R_{ij} is the adjacency matrix. R_{ij} will be 1, if there is a direct link between ith node and jth node and R_{ij} will be '0', if there is no direct link. Here 'n' is the total number of nodes and δ is a constant. The centrality value will be the total link scores of all nodes connected to a particular node. Which is nothing but,

$$EVC_i = \frac{1}{\delta}\sum_{j \in \mathrm{S(i)}} EVC_j \qquad (3.7)$$

Where, S(i) is the set of nodes that are linked to the ith node.

3.2.2.2 Calculation of Trust Based on Recent Satisfaction Index

A node N processes F(i, j) and E(i, j). F(i, j) is defined as the percentage of packets originated from n_i, which were forwarded by n_j over the aggregate number of packets offered to n_j. E(i, j) is defined as the rate of packets that were lapsed over the aggregate number of packets offered to node j. Every node periodically computes its connectivity rating (recent satisfaction index (RSI)) with each of its immediate neighbor nodes utilizing the above computed rates.

$$RSI_{ij} = F(i, j) - E(i, j) \qquad (3.8)$$

RSI_{ij} is normalized into the direct reputation of node j. The trust value calculated will lie in between the previous trust value and the recent satisfaction index. So it is a convex function and we denote it by Tr_{ij}.

$$Tr_{ij} = Tr_{ij\text{-}pr} * \eta + RSI_{ij} * (1- \eta) \qquad (3.9)$$

where, $Tr_{ij\text{-}pr}$ = is the previous trust value calculated.

η = a constant that shows level of confidence. The value of η denotes different confidence level. If the connectivity of a node less than a particular limit, the value of η decreases multiplicatively to β for calculating the Tr_{ij}.

3.2.2.3 Calculation of Combined Trust

The combined trust is or normalized trust is going to be calculated by using the two factors computed in section 3.2.2.1 and 3.2.2.2.

$$CT_{ij} = EVC_i * \frac{Tr_{ij}}{(f(t)_{\max}(Tr_{ij}))} \qquad (3.10)$$

$f(t)_{max}$ is the function that reports about the maximum Tr_{ij} over time t. The normalized trust will lie between 0 and 1, since it is derived by using a convex function.

CT_{min} represents the minimum threshold value of trust. The trust value depends on the eigen vector centrality score and the recent satisfaction index. A normal node must have a trust value CT_{ij}, higher than the threshold minimum CT_{min}, of the network (i.e. $CT_{ij} > CT_{min}$). The trust value is computed dynamically and depends upon this the route is going to be selected. The routes that contained nodes having trust value less than threshold will not be considered for path selection. The node revocation mechanism will be initiated to isolate the malicious nodes from the network.

3.2.2.4 Application Level Trust Optimization

Here we have calculated the trust value by using Eigen vector centrality method by considering the total connectivity scores and the recent satisfaction index about a node by its neighbors. The trust value is normalized between 0 and 1 as a continuous real number by using appropriate constants which shows the level of confidence as explained in 3.2.2. The threshold is fixed based on the number of nodes and the adjacency matrix. Also the threshold depends upon the kind of security service required for particular applications that we are using.

Here we propose a concept of application level trust optimization allowing an application to optimize the trust and fix a threshold to classify the nodes whether it is malicious or not. For misbehaving node detection application a threshold is fixed and it may be dynamically vary depends upon the network environment (Eg: increasing the population of misbehaving nodes).For the survivability management application the minimum best level of trust and the drop dead trust is fixed to complete the mission. For the source routing application a path trust is maintained for ensuring maximum delivery ratio and minimum delay.

So we can say that the threshold is application dependant rather platform dependant. Applications like tactical operations and military objects coordination the need of maintaining high trust for each node is essential. So the threshold may be fixed slightly higher compared to applications like smart city / office setup. So the threshold will vary depends upon the application that we are using. In our simulation we have fixed 0.5 as the minimum threshold. Setting the threshold value dynamically is a new research area that has to be explored.

3.2.2.5 Analysis of Trust aggregation

In this section we have presented the analysis of the combined trust calculated and we prove that the trust value of a normal node will always be greater than the trust value of a malicious node.

Theorem:

The trust value of a normal node will always be greater than the trust value of a malicious node.

Definition 01:

The level of trust is measured by a continuous real number referred to as trust value between 0 and 1. A node x_i is considered to be malicious, if the combined trust value calculated is below a threshold. Let the threshold be 0.5. x_i is considered to be normal if the combined trust value calculated is above the threshold. So, as per the assumption the combined trust value of a malicious node will lie between 0 & 0.5 and that of a normal node will lie between 0.5 and 1.

Assumption 01:

A malicious node always drops more number of packets compared to a normal node.

Claim:

Let x$_i$ is a node x$_j$ represents the node linked to x$_i$. Then the trust calculated,

$$CT_{ij} > CT_{ij}^{m}$$

Where, CT_{ij} is the combined trust calculated for normal node and CT_{ij}^{m} is the combined trust calculated for malicious node.

Proof:

Let us prove this by contradiction.

Assume that,

$$CT_{ij} < CT_{ij}^{m}$$

From Equation 3.10,

$$CT_{ij} = EVC_i * \frac{Tr_{ij}}{f(t)_{max}(Tr_{ij})}$$

Where, EVC_i is the centrality score calculated by the total link score connected to node x$_i$, normalized to a value between 0 and 1. Tr_{ij} is the trust value calculated for node x$_i$ based on the recent satisfaction index and $f(t)_{max}(Tr_{ij})$ is a function that reports the maximum trust value over time t.

Substituting:

$$EVC_i * \frac{Tr_{ij}}{f(t)_{max}(Tr_{ij})} < EVC_i^{m} * \frac{Tr_{ij}^{m}}{f(t)_{max}(Tr_{ij}^{m})}$$

As per definition 1 the combined trust value of normal node will always be greater that the assumed threshold 0.5. The combined trust value is calculated by using two metric, the centrality score EVC_i and the recent satisfaction index Tr_{ij}. The centrality score used in the trust calculation for selecting a high connectivity normal node for routing. Only the value of Tr_{ij} is the deciding factor to realize whether a node is malicious or not. The value of EVC_i for a node in normal malicious condition will be same. So,

$$EVC_i = EVC_i^m$$

From equation 3.9,

$$Tr_{ij} = Tr_{ij-pr} * \eta + RSI_{ij}(1-\eta)$$

For a node x_i, the previous trust value (Tr_{ij-pr}) will be same in both scenarios. η is a constant which shows the level of confidence. So, the combined trust value purely depends upon the Recent Satisfaction Index (RSI_{ij}). So,

$$RSI_{ij} < RSI_{ij}^m$$

From equation 3.8,

$$RSI_{ij} = F(i,j) - E(i,j)$$

Where, F(i, j) is defined as the percentage of packets originated from x_i, which were forwarded by x_j over the aggregate number of packets offered to x_j and E(i, j) is defined as the rate of packets that were dropped over the aggregate number of packets offered to x_j.

$$F(i,j) - E(i,j) < F(i,j)^m - E(i,j)^m$$

In both scenarios, $F(i,j)$ can be considered as same. So,

$$E(i,j) > E(i,j)^m$$

As per the assumption 01, it will be like $\mathrm{E}(i,j) < E(i,j)^m$ so it is a contradiction. Hence we can say that, the trust value of a normal node will always be greater than the trust value of a malicious node.

3.2.3 Certificate Exchange Technique

The certificate exchange method offers the nodes to authenticate themselves with the individuals in the network before they some assistance with getting joined and begin another communication. With a specific end goal to improve the unwavering quality of certificate exchange protocol, Multi-path Technique is used. During the multi-path certificate exchange, the public key of a node is certified by the diverse nodes. As an aftereffect of various autonomous certifications, the certainty doled out to the certificates is higher. Also, the authentication is performed commonly. Table 3.1 presents the notations used in the certificate exchange technique.

Notations	*Representation*
S	*Source Node*
D	*Destination Node*
N_i	*intermediate nodes*
kpu_d	*public key of D*
kpu_s	*public key of*
$T(S)$	*set of nodes certified for* kpu_s
REQ_{cert}	*certificate request message*
REP_{cert}	*certificate reply message*
C_{self}	*self-signed certificate*
ID_D	*the identity value of D*
CL	*certificate list*

Table 3.1 Notations used in certificate exchange technique

When S receives kpu_d, it issues a certificate for that public key. Consequently, D issues a certificate for kpu_s. Each node in T(S) contains its public key certified by S since the authentication is mutual. The steps involved in the certificate exchange process are as follows.

Step 01: S broadcasts REQ_{cert} containing ID_D and T(S) for D's certificates.

$$\text{S} \xrightarrow{REQ_{Cert} + ID_D} \text{Neighbor nodes}$$

This REQ_{cert} is sent with a minimum time to live (TTL_{min}) for minimizing the communication overhead of the protocol.

Step 02: When N_i receives the REQ_{cert}, it verifies kpu_s and checks its own CL.

If, (N_i has no certificate for D) | | (N_i has already replied to the REQ_{cert})

Then, N_i forwards the REQ_{cert} to its neighbor nodes

Else, N_i feedbacks REP_{cert} to S that contains the certificate of kpu_d signed by N_i

Step 03: If, N_i is unaware of S, Then, N_i constructs a C_{self} and notifies S that it wants to make a certificate exchange which is performed via a multiple node-disjoint paths.

Step 04: If, N_i already has a route to D in its cache, Then, N_i informs D that S has requested its $kpu_{d.}$ D responds to query and requests a certificate for kpu_s.

Since N_i and D can authenticate each other, the communication among the D and N_i is made secured using N_i's signature. Hence there is no possibility for any node to corrupt the certificate of S which is issued by N_i.

Step 05: If, D is unaware of adequate number of nodes Then, D replies to REQ_{cert} itself.

Step 06: S repeats the above process by increasing the TTL value until it obtains the minimum number of certificates for kpu_d.

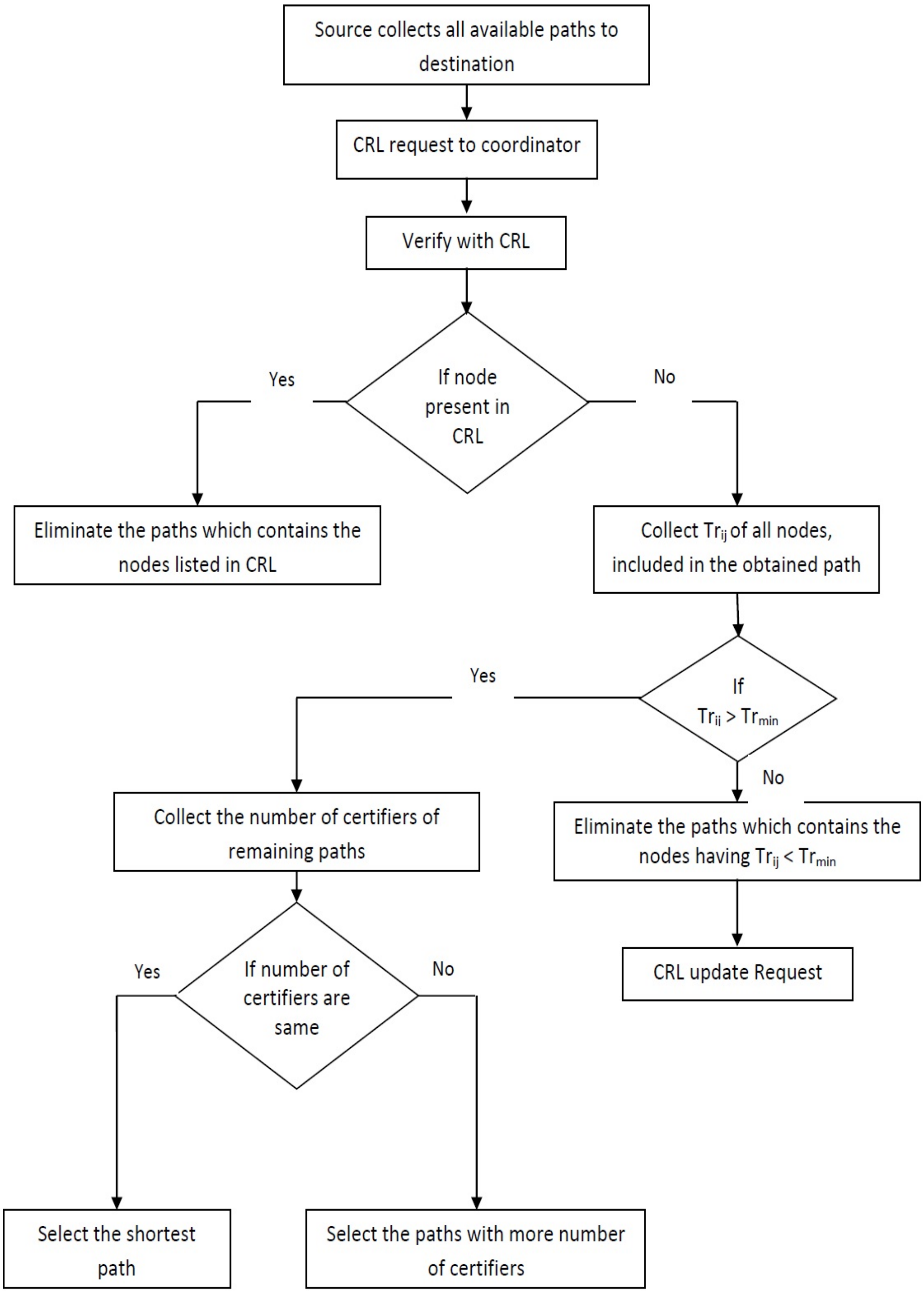
Source collects all available paths to destination
CRL request to coordinator
Verify with CRL
If node present in CRL
Yes
No
Eliminate the paths which contains the nodes listed in CRL
Collect Tr_{ij} of all nodes, included in the obtained path
If $Tr_{ij} > Tr_{min}$
Yes
No
Collect the number of certifiers of remaining paths
Eliminate the paths which contains the nodes having $Tr_{ij} < Tr_{min}$
CRL update Request
If number of certifiers are same
Yes
No
Select the shortest path
Select the paths with more number of certifiers

Fig 3.2 Route Discovery and Path Selection

Step 07: S then calculates the trust value CT_{ij} of the nodes included in the all offered paths.

Step 08: S considers only those paths, which are free from malicious nodes. S performs the certificate revocation process for defending against the malicious nodes.

Step 09: Among the obtained paths, source selects a path which is having more certifiers of the destination node D, as explained in fig 3.2.

Step 10: S then forwards the first packet to D that contains the set of nodes that has offered the certificates for kpu_d.

Step 11: Once they have exchanged their public keys, S and D issue certificates for each other.Due to multiple independent certifications, the confidence assigned to these certificates is higher. For example, consider the Figure 3.3.

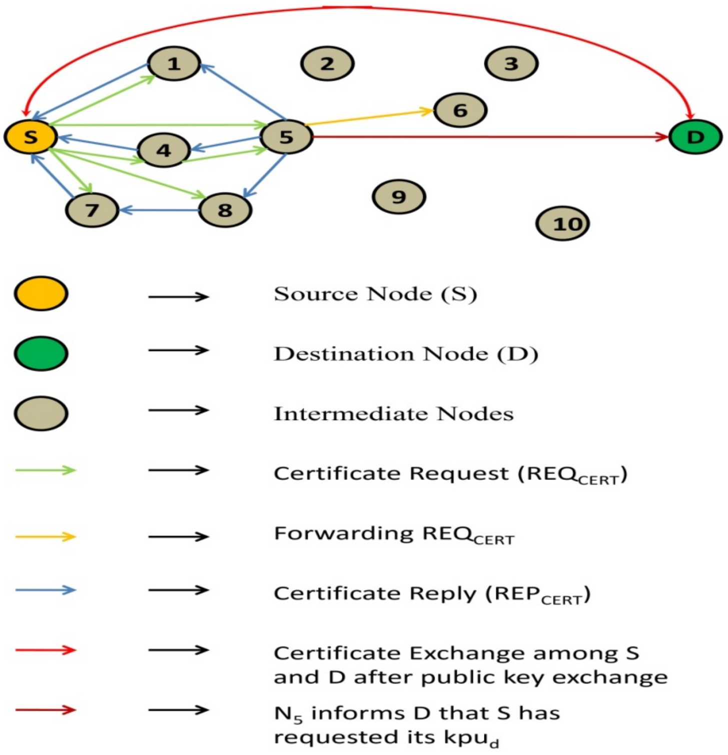
1
2
3
S
6
5
D
4
9
7
8
10
Source Node (S)
Destination Node (D)
Intermediate Nodes
Certificate Request (REQ$_{CERT}$)
Forwarding REQ$_{CERT}$
Certificate Reply (REP$_{CERT}$)
Certificate Exchange among S and D after public key exchange
N$_5$ informs D that S has requested its kpu$_d$

Fig 3.3 Certificate Exchange Technique

We demonstrate our certificate exchange mechanism by considering N_5. S broadcasts the REQ_{cert} to its neighbor nodes. When N_5 receives the message, it checks its CL. If N_5 does not know D or it has already sent the REP_{cert}, then it just forwards it to next node N_6. Otherwise, N_5 replies with REP_{cert} that contains the certificate of kpu_d signed by N_5 to S. When N_5 is not aware of S, then N_5 constructs a C_{self} and notifies S that it wants to make a certificate exchange via multiple node-disjoint paths. i.e. through (N_5-N_1-S) & (N_5-N_4-S) & (N_5-N_8-N_7-S). If N_5 already has a route to D in its cache, then it informs D that S has requested its kpu_d and it responds to query and requests a certificate for kpu_s. If D is unaware of adequate number of nodes, it replies to REQ_{cert} itself. S repeats the above process by increasing the TTL value until it obtains the minimum number of certificates for kpu_d.

3.2.4 Certificate Revocation Technique

Each node is going to revoke its public key certification of the malicious nodes depends upon the updated CRL. The source collects the trust value, Tr_{ij} of the participant nodes before every communication. Based on the trust value, it lists the malicious nodes before communication. The nodes having a trust value lower than Tr_{min} can be marked as malicious nodes. Source will select the communication path based only on the latest retrieved CRL from the coordinator node. If it could find all the listed nodes in the CRL, then it will continue the communication.

If the list contains some nodes that are not listed in the CRL, then source requests a CRL update to coordinator node before the communication. The coordinator node will be responsible for CRL update and revocation. This process initially takes the following assumptions.

Let CRL be the certificate revocation list regarding nodes in MANET.

Let R_{REQ} be the revocation initialization request.

Let R_{REP} be the revocation initialization reply.

Let ID_s be the source ID.

The steps involved in this technique are as follows:

Step 1: Source transmits R_{REQ} signed by source itself for initiating the CRL update. The R_{REQ} includes ID of source node.

$$\text{Source} \xrightarrow{Sign_s\{R_{REQ}\}} \text{Coordinator Node}$$

Step 2: Upon receiving $R_{REQ,}$ coordinator node replies with R_{REP} message signed by coordinator node itself that contains the ID of source node$_{.}$

$$\text{Source} \xleftarrow{Sign_c\{R_{REP}\}} \text{Coordinator Node}$$

Step 3:

If, Signature verification fails,

Then, Message is discarded

Else

$$\text{Source} \xrightarrow{Sign_s\{Rp\text{^}RF\}} \text{Coordinator Node}$$

$$\text{Source} \xleftarrow{Sign_c\{CRL\}} \text{Coordinator Node}$$

If signature verification fails, the message is discarded. Otherwise source starts to transmit the report and refresh (RP^RF) message to coordinator node. RP^RF contains ID_s and detected node ID's. Coordinator node then registers the reported nodes and replies with updated CRL (CRL with incremented CRL number).

Case 01:

If, CRL is not received by source node in spite of continuous service demand of the coordinator node

Then, Source repeats the above process by increasing the TTL value until obtains the updated CRL

Source appends the CRL into the existing routing messages which is utilized by the certificate exchange mechanism. The revocation information appended with routing message is required to be verified only at the time of changes in the revocation list number.

Case 02:

If, the routing message comes from revoked nodes, then the message is discarded.

The node which receives the updated CRL will check the new list for getting the details of newly added nodes to the CRL. If it could find a certified public key of any malicious nodes in its own certified list, then it revokes that certificate before next communication.

3.3 Overall Algorithm

The entire process of the proposed framework is described using the following algorithm.

Step 01 – Network Architecture

The architecture for self-organized key management technique is constructed such that it includes a coordinator node, servers and normal mobile nodes. The coordinator acts as a mediator node for transmitting messages from normal mobile nodes to the servers. The coordinator node will collect the shares for generating the public/private key pair. The public/private key pair is generated by the node itself.

Step 02 – Public/Private Key Generation

Each mobile node generates its own public/private key pairs using server-signed public keying technique. The coordinator node helps in generating the publicly-recoverable public key for any node N_i without the knowledge of the subsequent private key. The coordinator node acts as a distributed trusted authority. It combines the shares of (t+1) servers for computing signature parameter.

Step 03 – Trust Calculation

The nodes in the network are validated using the trust management technique named as Eigen Vector Reputation Centrality technique.

Step 04 – Multi-Path Certificate Exchange

After the generation of public/private key pairs, multi-path certificate exchange technique is employed where public key of the nodes are certified by different nodes. The authentication is also performed mutually.

Step 05 – Malicious Node Detection

Source will collect the certifiers of the destination node and all possible paths to the destination. It then collects the trust value $CT_{ij,}$ of the nodes included in the all offered paths. CT_{min} represents the minimum threshold value of trust. The minimum threshold value depends on the total number of nodes and the adjacency matrix of the network. A normal node must have a trust value CT_{ij}, higher than the threshold minimum CT_{min}, of the network (i.e. $CT_{ij} > CT_{min}$).

Step 06 – Security Considerations

Source considers only those paths, which are free from malicious nodes. It will select the communication path based only on the latest retrieved CRL from the coordinator node.

Source performs step 8, 9 and 10 for defending against the malicious nodes.

Step 07 – Path Selection

Among the obtained paths, source selects a path which is having more certifiers of the destination node. After selecting the path, source and destination certifies their public keys each other.

Step 08 – CRL Update

The coordinator node performs the certificate revocation mechanism for defending against the malicious nodes. Based on the trust value, source identifies the malicious nodes before communication. The source node checks the CRL list and identifies if any new node showing malicious behavior. If so source requests for a CRL update to the coordinator node. The coordinator node updates the CRL and sends back the list to the source for certificate revocation process.

Step 09 – CRL Distribution

Source appends the CRL into the existing routing messages which is utilized by the certificate exchange mechanism. The revocation information appended with routing message is required to be verified only at the time of changes in the revocation list number.

Step 10 – Certificate Revocation

Each node is going to revoke its public key certification of the malicious nodes depends upon the updated CRL. The coordinator node takes the responsibility to update the CRL depends upon the entry in CRL, each node is going to eliminate the nodes from that node to the destination node. Since the CRL update happens dynamically, the security and the detection ratio of malicious node will be high.

3.4 Simulation Results

3.4.1 Simulation Model and Parameters

Simulations were performed utilizing Network Simulator (NS-2), especially well known in the ad hoc networking group. The MAC layer protocol IEEE 802.11 with a data rate of 11 Mbps is utilized as a part of all simulations. The transmission range is set to 250m. The propagation model is Two Ray Ground. The aggregate number of nodes is set to 100 nodes in 1000m x1000m network territory. In our simulation, the minimal speed is 5 m/s. The source-destination pairs are spread randomly over the network. The ns-2 constant bit rate (CBR) traffic generator is utilized to set up the association designs with distinctive irregular seeds. Every node has one CBR traffic association with a solitary unique destination. Sources start time is consistently distributed over the initial 60 seconds of the simulation time. We change the load value as 50,100,150,200 and 250Kb.

The size of certificates was likewise set to 512 bytes. The aggregate number of connections in the network was set to 20 connections. The Ad Hoc On-demand Multipath Distance Vector (AOMDV) routing protocol was decided for the simulations. The simulation results are the normal of 10 runs. The proposed system was effectively incorporated into the AOMDV protocol's route discovery mechanism.

In the simulation, attacks are simulated where the attacker nodes send spurious certificates to the nodes which have requested for those certificates. These attacks can be isolated attacks where each attacker guarantees an alternate public key. In any case, the attackers might likewise dispatch an agreeable attack where a group of attackers collude and send certifications for the same public key that is spurious. Both these sorts of attacks-isolated and intrigue are simulated.

Our simulation settings and parameters are summarized in table 3.2

No. of Nodes	100
Area Size	1000 X 1000
Mac	802.11
Radio Range	250m
Simulation Time	100 to 500 sec
Routing Protocol	AOMDV
Traffic Source	CBR
Packet Size	512
Speed	5m/s
Pause time	5 seconds
Load	1000 Kb.
No. of attackers	1 to 10

Table 3.2 Simulation Settings for SOKMTC

The rate of attacker nodes is altered as 10% of the aggregate number of nodes in the network (ie) 10 attackers. Node initialization at the network bootstrapping stage is likewise simulated. It is demonstrated that every node has effectively executed the initialization venture by exchanging imperative number of certificates with the honest nodes in the network. Starting trust value of 0.75 is assigned to a node that is authenticated during the initialization step, while other nodes are expected to have a trust value of 0.5. The full trust value is thought to be 1. The beginning trust value is picked more than half of the full trust value and different nodes trust values are picked half of the full trust value.

The following are the assumptions used for the proposed framework,

- A malicious node can compromise the key, create packet drop attack, routing overflow attack etc.
- A trusted node will be having a trust value greater than 0.5.
- A threshold trust is fixed as 0.5.
- Malicious node will be having a trust value less than 0.5
- Trust value 1 refers to full trust and 0 refers to complete distrust.
- The coordinator node is the in charge of CRL update.
- Protocol used is AOMDV.

3.4.2 Performance Metrics

We compare the proposed Self-organized Key Management for Trusted Certificate Exchange and Revocation (SOKMTC) technique with On-demand Self-Organized Public Key Management (SOPKM) scheme [88], Ad hoc on-demand trusted-path distance vector (AOTDV) routing protocol [91] and Ad Hoc On-demand Multipath Distance Vector (AOMDV) routing protocol. We select SOPKM and AOTDV among the existing works, since it is the latest work which deals self-organized key management along with certificate chains and simulated in NS-2.

We evaluate mainly the performance according to the following metrics [91]:

- **Average end-to-end Delay:**

  The normal time taken by the data packets from sources to destinations, including support delays during a route discovery, lining delays at interface lines, retransmission delays at MAC layer and propagation time.

- **Packet Delivery Ratio:**

  The portion of the data packets delivered to destination nodes to those sent by source nodes.

- **Packet Drop:**

  It is the number of packets dropped during the transmission.

- **Misdetection Ratio:**

  The proportion of the number of nodes whose conduct (malicious or generous) is not recognized accurately to the genuine number of such nodes in the network.

- **Routing packet overhead**:

  The number of control packets (including route request/reply/update) for establishing connection over a period of time.

- **Resilience against Node Capture:**

  The fraction of communications compromised to the total number of communications by a capture of x-nodes.

3.4.3 Results

Varying Number of Attackers

The number of attackers is increased from 1 to 10 and the performances of the techniques are measured in terms of Delivery Ratio, Misdetection and Resilience.

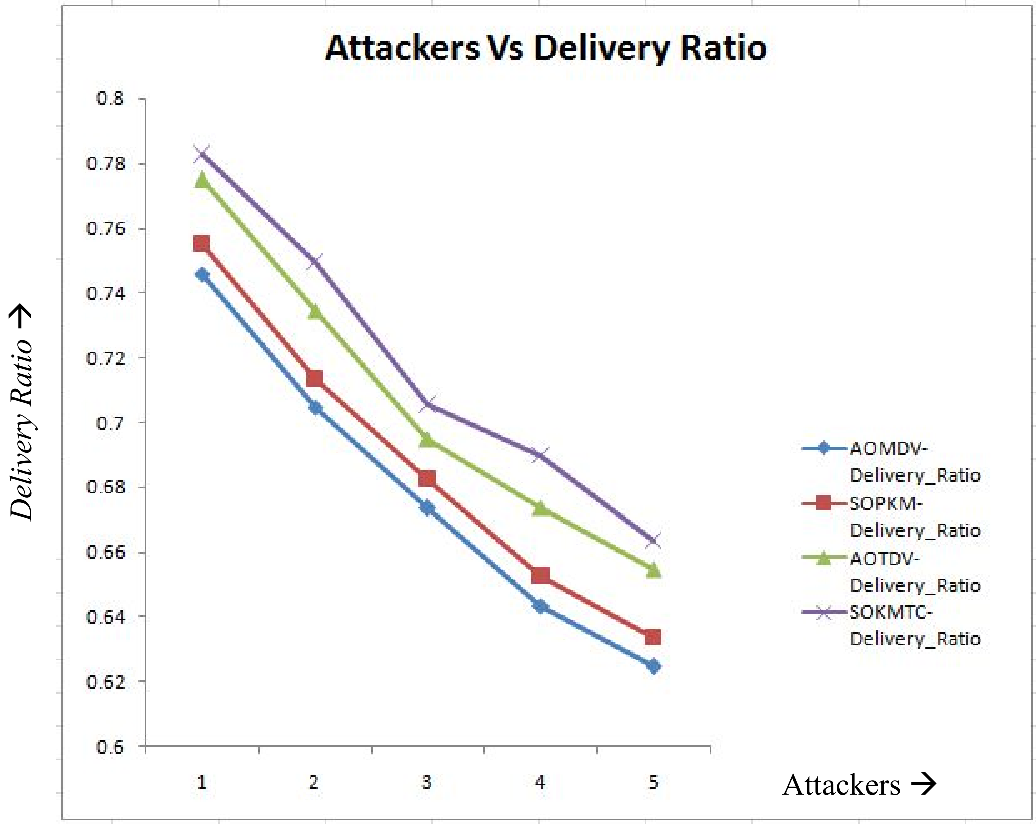

Fig 3.4 Packet Delivery Ratio

Figure 3.4 shows the average Packet delivery Ratio of the schemes, when the attackers are increased from 1 to 5. We can see that the delivery ratio decreased linearly as the attacker increases. But, the delivery ratio of our proposed SOKMTC is greater than the existing schemes. The delivery ratio is high, because the trusted certificate exchange and revocation mechanism identifies the malicious nodes dynamically and eliminates the same immediately after the detection.

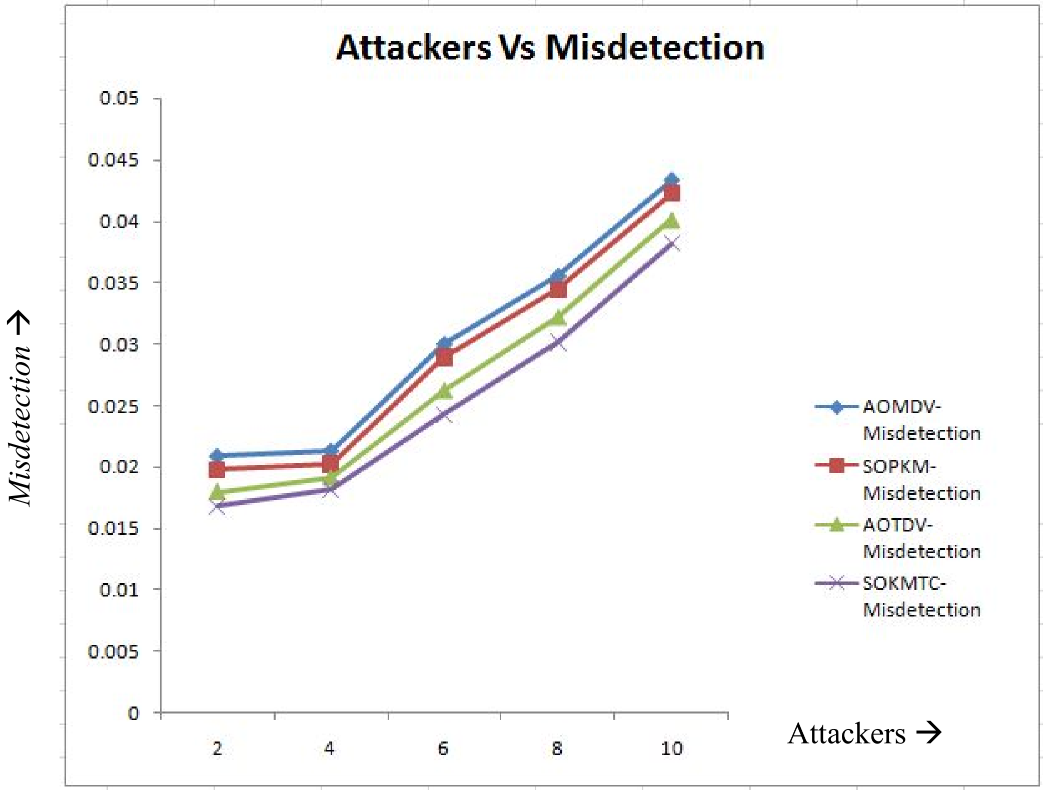

Fig 3.5 Misdetection Ratio

The ratio of the number of nodes whose behavior is not identified correctly to the actual number of such nodes in the network is shown in figure 3.5. Our proposed method is capable to detect more malicious nodes while comparing to the existing methods. The misdetection ratio is less, that means the framework can successfully detect the malicious node in time itself and able to eliminate dynamically.

The result of fraction of compromised communications is shown in figure 3.6. Because of the trusted mechanism, the number of compromised communications is less in SOKMTC. Hence the proposed SOKMTC is more resilient than the existing mechanisms. Here the malicious nodes are identified immediately when their behavior becomes malevolent. So the ability to defend against attacks of a network is improved, that means the communications where the attacker node involved is very less.

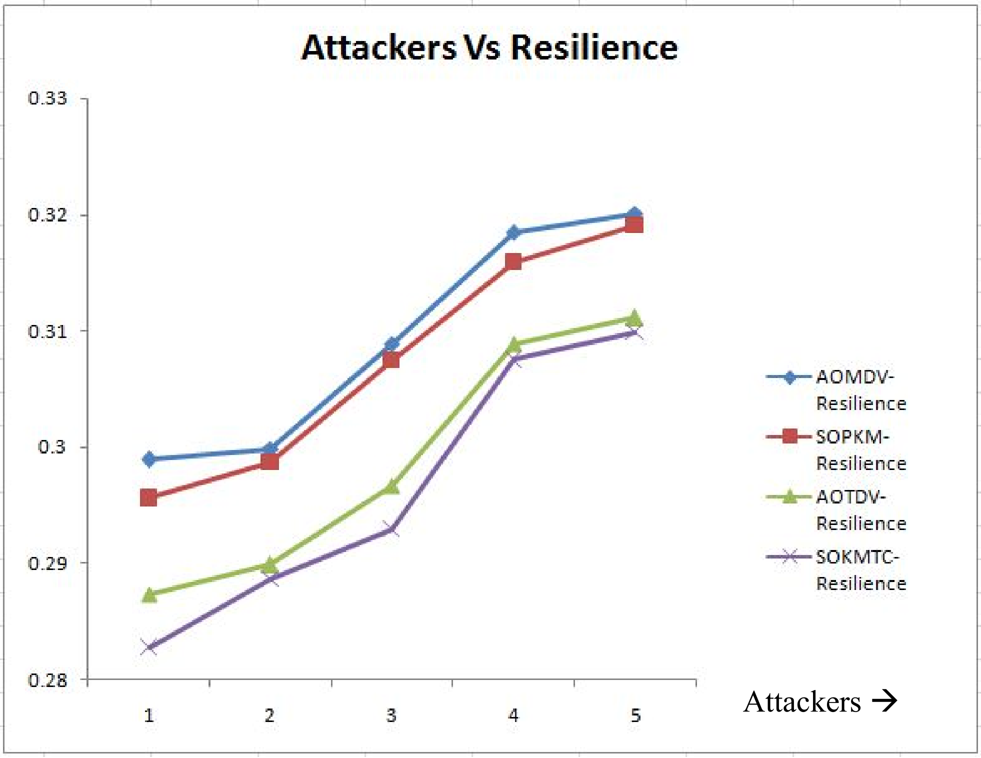

Fig 3.6 Resilience against Node Capture

Varying Number of Nodes

The CBR data packets and control packets dropped due to the attackers, presented in figures 3.7. As the number of attacker increases, more data packets are dropped. But SOKMTC has less packet drops when compared to other schemes. The dropping of packets is less for the proposed method, since the framework ensure to select the trust path, having more certifiers for communication.

Figure 3.8 depict the delay involved in the communication by each pair of source and destinations. The number of nodes is varied from 10 to 50, and corresponding delay for the four schemes are measured. The proposed method outperforms the existing methods in case of delay. The delay is very less for the proposed method since the framework selects the trusted path, thus the path breakage problem will not affect communication.

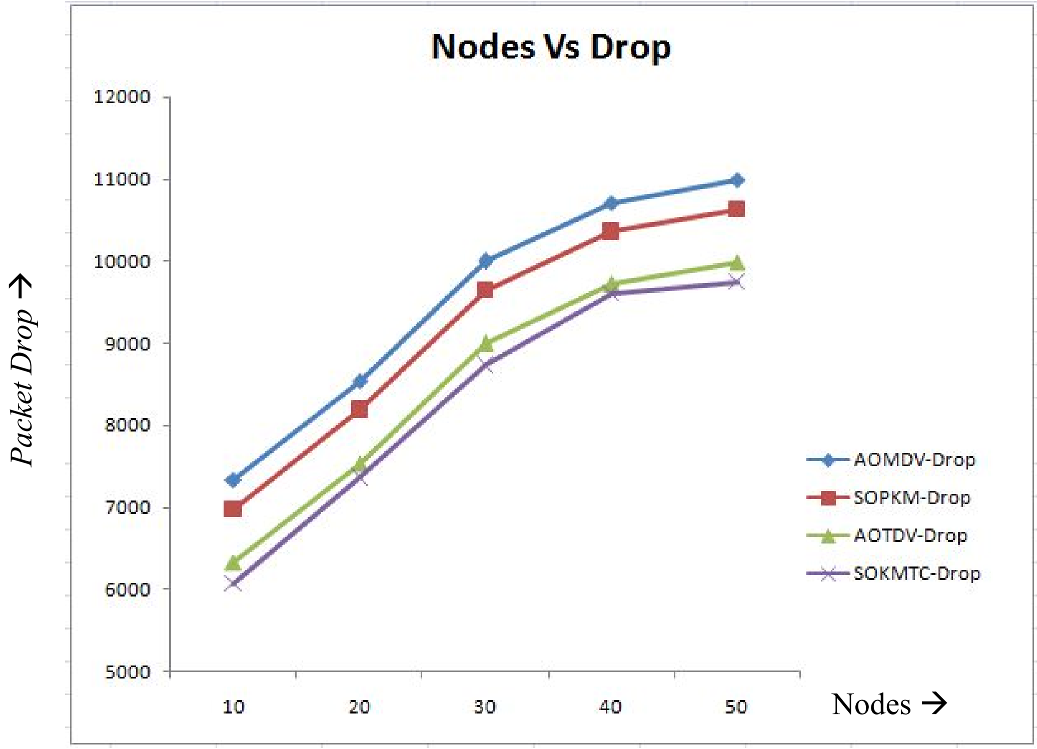

Fig 3.7 Packet Drop

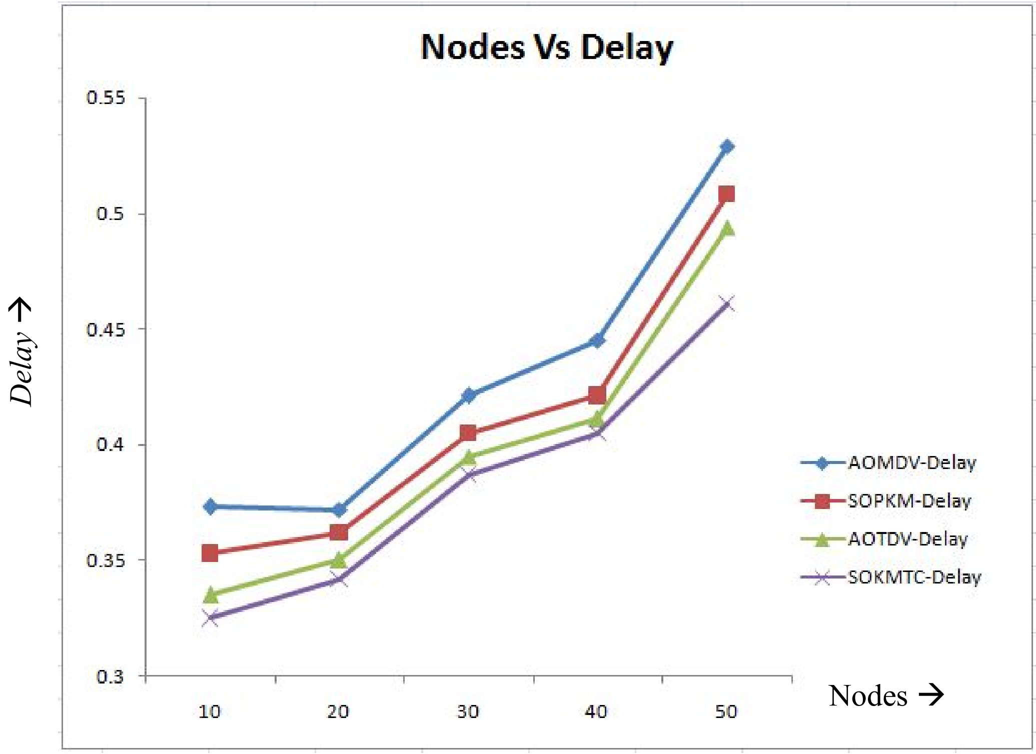

Fig 3.8 Average end-to-end Delay

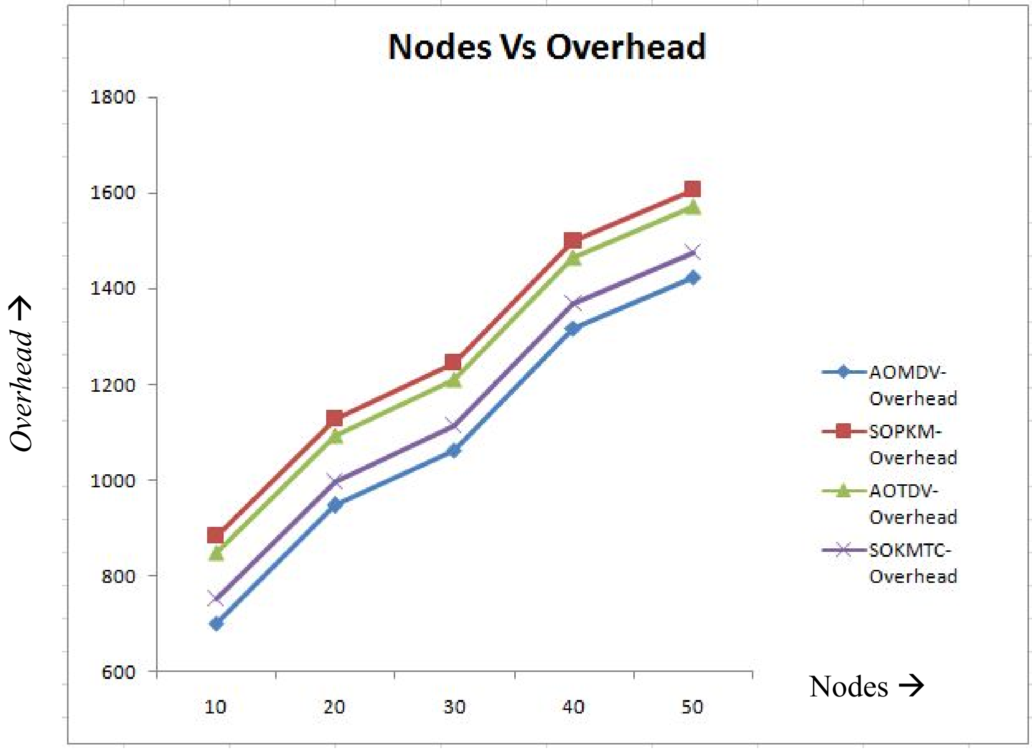

Fig 3.9 Routing packet overhead

Figure 3.9 shows the Routing packet overhead of the schemes, when the nodes are increased from 10 to 50. We can see that the overhead of our proposed SOKMTC is greater than the basic AOMDV since the proposed method contains the trust management mechanism for certificate exchange and revocation, but it is lesser than both other schemes.

3.5 Conclusion

In this chapter, we have proposed a framework based on trust for effective key management in MANET, with trusted certificate exchange and revocation. The proposed architecture consists of the coordinator node, servers and ordinary mobile nodes. The coordinator acts as mediator for transmitting the message among the servers and mobile nodes. Each node generates its own public/private key pairs using server-signed public keying technique. The coordinator node helps in generating the publicly-recoverable public key for any node N_i without the knowledge of the subsequent private key. The nodes that issued the certificates are validated using the trust management mechanism. The trust value is computed using the Eigen Vector Reputation Centrality. Then multi-path certificate exchange technique is employed where public key of the nodes are certified by different nodes. As a result of multiple independent certifications, the confidence assigned to the certificates is higher. The source performs certificate revocation process for defending against the malicious nodes. By simulation results, we have shown that the proposed approach enhances the security against node capture attacks and improves the packet delivery ratio and detection rate. The approach also achieves better resilience, reduced delay and packet drop.

CHAPTER 4

TRUST PREDICTION MODEL FOR CERTIFICATE EXCHANGE AND REVOCATION IN MANET

CONTENTS

4.1 Overview

In this chapter, a trust prediction model based on accusation is proposed for certificate exchange and revocation in MANET. The trust value is computed from three distinct sorts of trust measures, such as, verifiable trust, current trust and route trust. Node's verifiable trust is calculated by the node's one hop neighbors based on the packet forwarding ratio over a particular time period. A node's current trust can be computed from the node's verifiable trust in light of the fuzzy logic rules prediction strategy. Route trust can be computed by intermediate nodes trust values along the route. It utilizes a trust based accusation scheme to overcome with the malicious nodes. Those nodes that issued the certificates are validated utilizing the Trust Prediction Model. The source node discards the malicious nodes in the data sending path and sidesteps the data through different nodes in interchange chose path. Source performs certificate revocation process for guarding against the malicious nodes in light of the trust values.

4.2 Trust Prediction Model

In this framework we are using a prediction mechanism to predict the trust of each node. Here, we are going to calculate the trust by using a prediction technique based on two factors [92] [93]. The mathematical function that we have defined, predicts the present trust level of a node by using the following,

1. The past behavior of the node called the node's verifiable trust
2. The current capability level of the node.

Here each node is going to calculate or predict its trust, depends upon the past interaction with its neighbors and the current capability level in terms of battery power, memory usage etc. The communications with the one hop neighbors is going to be treated as the direct interactions. The trust in a particular time domain will lie between a range 0 and 1. The value 0 means, complete distrust and 1 means full trust. We are going to define a threshold value between 0 and 1, below which a node is going to be treated as malicious node otherwise it will be treated as trusted node. Here for eliminating the malicious nodes and identifying the trusted path, each node calculates trust by using three parameters, which are verifiable trust, current trust and route trust.

4.2.1 Node's Verifiable Trust

Node's verifiable trust is calculated as the proportion of the number of packets successfully forwarded correctly to the number of packets expected to be forwarded. A packet is said to be forwarded correctly, not only transmits the packets to the next hop, but also forward the same without any alteration in the data packets. Suppose, the sender monitors the forwarding of packets, when it finds any illegal modification to the packets, the forwarding ratio of the neighbor will diminish. Here the forwarding ratio for a particular time period *t* is called sending ratio, denoted by SR(t).

A node n_i maintains a trust table, which contains the trust of its neighboring node assessed by $n_{i,}$ depends upon the successful number of packet forwarded to the total number of packets offered by n_i. The verifiable trust is calculated by using the rate of successful packet transmission to the total number of packets offered over a period of time. Here we are considering the direct interactions with the nodes. Each node is calculating its neighbor's verifiable trust depends upon the history of successful packet forwarding ratio over a period of time. When packet dropping occurs during transmission the sending ratio diminishes.

At time t, SR(t) is computed over a period of time 0 to t as,

$$\mathrm{SR(t)} = \frac{\mathrm{P_c}}{\mathrm{TP_c}} \tag{4.1}$$

Where, P_c is the cumulative count of correct forwarding packets and TP_C is the total number of all packets offered.

In the adhoc network scenario, all the packets can be divided into two types, control packets and data packets. The control packets are used to establish accurate routes in the network. The integrity of the control packets forwarded from one node to another should be ensured. So the sending ratio of packets is divided into two parts, the control packet sending ratio and the data packet sending ratio. The control packet sending ratio over a period of time is denoted by CSR(t) and the data packet sending ratio over a period of time is denoted by DSR(t).

$$\mathrm{CSR(t)} = \frac{\mathrm{CP_c}}{\mathrm{TCP_c}} \tag{4.2}$$

Where, $\mathrm{CP_c}$ is the cumulative count of correct forwarding of control packets and $\mathrm{TCP_C}$ is the total number of all control packets offered.

$$\mathrm{DSR(t)} = \frac{\mathrm{DP_c}}{\mathrm{TDP_c}} \tag{4.3}$$

Where, $\mathrm{DP_c}$ is the cumulative count of correct forwarding of data packets and $\mathrm{TDP_C}$ is the total number of all data packets offered.

For calculating the CSR(t) and DSR(t), each node should able to identify, whether the packet offered to neighbor is successfully forwarded or not. For this, during the route discovery process by using the route request and route reply mechanism, the cumulative count of the successful packet forwarding ratio of the control packets is calculated. To compute the data packet sending ratio, an acknowledgement mechanism is going to be used. The packets are buffered until receives an acknowledgement regarding the successful packet forwarding by its neighbors.

The two factors that we have calculated, the conrol packet sending ratio and the data packet sending ratio are assigned different weight value, in order to determine a node’s trust. The weight values are assigned in order to obtain a trust value in between 0 and 1. The value 0 is referred complete distrust and 1 referred absolute trust. Here for a particular time interval between 0 to t, the verifiable trust of jth node assessed by ith node (VT_{ij}), is calculated by using the following formula:

$$\mathrm{VT}_{ij}(t) = w_1 \times \mathrm{CSR}_{ij}(t) + w_2 \times \mathrm{DSR}_{ij}(t) \tag{4.4}$$

This is the model of a convex function, where $w_1$and w_2 are two non-zero convex parameters and $w_1 + w_2$ will be equal to 1. So the trust value will lie between a range between zero and one.

4.2.2 Node’s Current Trust

The current trust of a node is predicted by using its verifiable trust and the current capacity called the capability level depends upon the memory usage and the battery power of a node. Let VT(t) represents a node's verifiable trust level at time t, calculated depends upon the successful packet forwarding rate. Let C(t) represents for the node's capacity level on offer services at time t, which incorporates the leftover usage proportion of battery, neighborhood memory, CPU cycle, and data transfer capacity. Let VT(t+1) represents the node's trust level at time t + 1. We define the following function:

Assume the fuzzy membership function of VT(t) or VT(t + 1) consists of four fuzzy sets:

VeryLow	(VL - malicious node)
Low	(L - low trustworthy node)
Medial	(M - trustworthy node)
High	(H - high trustworthy node),

Expect the fuzzy part function of C(t) additionally comprises of four fuzzy sets:

VeryLow	(VL - can't bear to give services)
Low	(L - low capacity level)
Medial	(M - medium capacity level)
High	(H - high capacity level)

The mapping function between VT(t) × C(t) will lead to VT(t+1), that is nothing but the current trust. That is, by using the node's verifiable trust, the capability level, the current trust of the node is going to be predicted by using the above fuzzy membership function. The mathematical function defined below, predicts the present trust of a node by using its past behavior and its current capacity level to offer the service.

Assume VL = 0, L = 1, M = 2 and H =3, now we define the function as follows,

$$f(x,y)=\begin{cases}0 \text{ if either } x=0 \text{ or } y=0\\ x \text{ if eiher } y=2 \text{ or } y=3\\ x-y \text{ if } x\neq 0 \text{ and } y=1\end{cases}$$

Where, f(x) is the fuzzy member function of the node's verifiable trust, f(y) is the fuzzy member function of the capacity level of a particular node. f(x,y) is the function that predicts node's current trust depends upon the node's verifiable trust and capacity level.

At last, every node owns a current trust table based on the predicted trust value. The values of the fuzzy set members will be between 0 and 1. A threshold is defined below which it is treated as malicious and includes in the black list. Each node is going to maintain a trust table, which contains the trust of all its neighboring nodes. A sample trust table for the node n_i is given below:

Node ID	**NT_{ij}**	**Block List**	**Threshold**
n_j	0.94	0	0.50
n_k	0.87	0	
n_l	0.17	1	
…	…	…	

Table 4.1 Trust Table

Here node ID is the unique identifier of node n_i's neighbors; NT_{ij} is the trust value about the neighbors predicted by node n_i. Block list indicator indicates, whether a node is malicious or not. 0 indicates that the trust value of the particular node is below a minimum threshold and routes through that nodes cannot be able to select for secure communication. 1 indicates the trust value is above a minimum threshold and we can select the paths through this node for secure communication. A threshold value, termed as the black list trust threshold (BT), is utilized to distinguish malicious nodes. If the value of BT is below the defined threshold, then that node is going to be treated as malicious; otherwise it is going to be treated as trusted node.

4.2.3 Route Trust

At time t, the trust of a route denoted by $NT_{path}(t)$ is calculated as the continued product of node trust values in that route. For example, let S is the source node, D is the destination node, and let the path from S to D is {S → I → J → D}. The route trust is calculated as follows,

$$\mathrm{NT_{SD}} = \mathrm{NT_{SI}} * \mathrm{NT_{IJ}} \tag{4.8}$$

Where, NT_{SI} is the predicted trust of node I stored in node S and NT_{IJ} is the predicted trust of node J stored in node I. Here we are not considering the trust of node D stored by node J for calculating the route trust from S to D, since D is the destination node. We are going to calculate the route trust of all possible paths. Paths does not have minimum threshold trust will be eliminated from the obtained paths. In general,

$$\mathrm{NT_{SD}}(t) = \prod(\{\mathrm{NT_{ij}}(t) | n_i, n_j \in P \text{ and } n_i \rightarrow n_j\}) \tag{4.9}$$

Where, P is the set of nodes contained in the particular path. While calculating the route trust there is no need to include the trust of destination D, assessed by its upstream node (say R). The trust NT_{RD} means the predicted trust for the destination D stored in the node R. We are not considering this for the route trust calculation, since D is the destination.

4.3 Certificate Exchange Technique and Route Discovery

The certificate exchange technique helps the nodes to authenticate themselves with the members in the network before they get joined and start a new communication. In order to enhance the reliability of certificate exchange protocol, Multi-path Technique is utilized. During the multi-path certificate exchange, the public key of a node is certified by the different nodes. As a result of multiple independent certifications, the confidence assigned to the certificates is higher. Moreover, the authentication is performed mutually. The certificate exchange mechanism is explained in section 3.2.3.

4.4 Path Selection

Route discovery process results a set of routes to the destination. An On-demand clustering will apply to the nodes in the obtained paths. Such a cluster includes all nodes in the obtained paths as well as the one-hop neighbors of those nodes. Source considers only those paths, which are free from malicious nodes. It will select the communication path based only on the latest retrieved CRL from the coordinator node. Source will do the following for finding the best path among the obtained routing routes.

Step 01: ***CRL Verification***

The source node checks the vicinity of already identified malicious nodes in the acquired paths. Source will send a CRL request to coordinator node. At that point it will check the nodes with the CRL. In the event that it could locate a malicious node in any path, then that path will be avoided from the arrangement of significant routes.

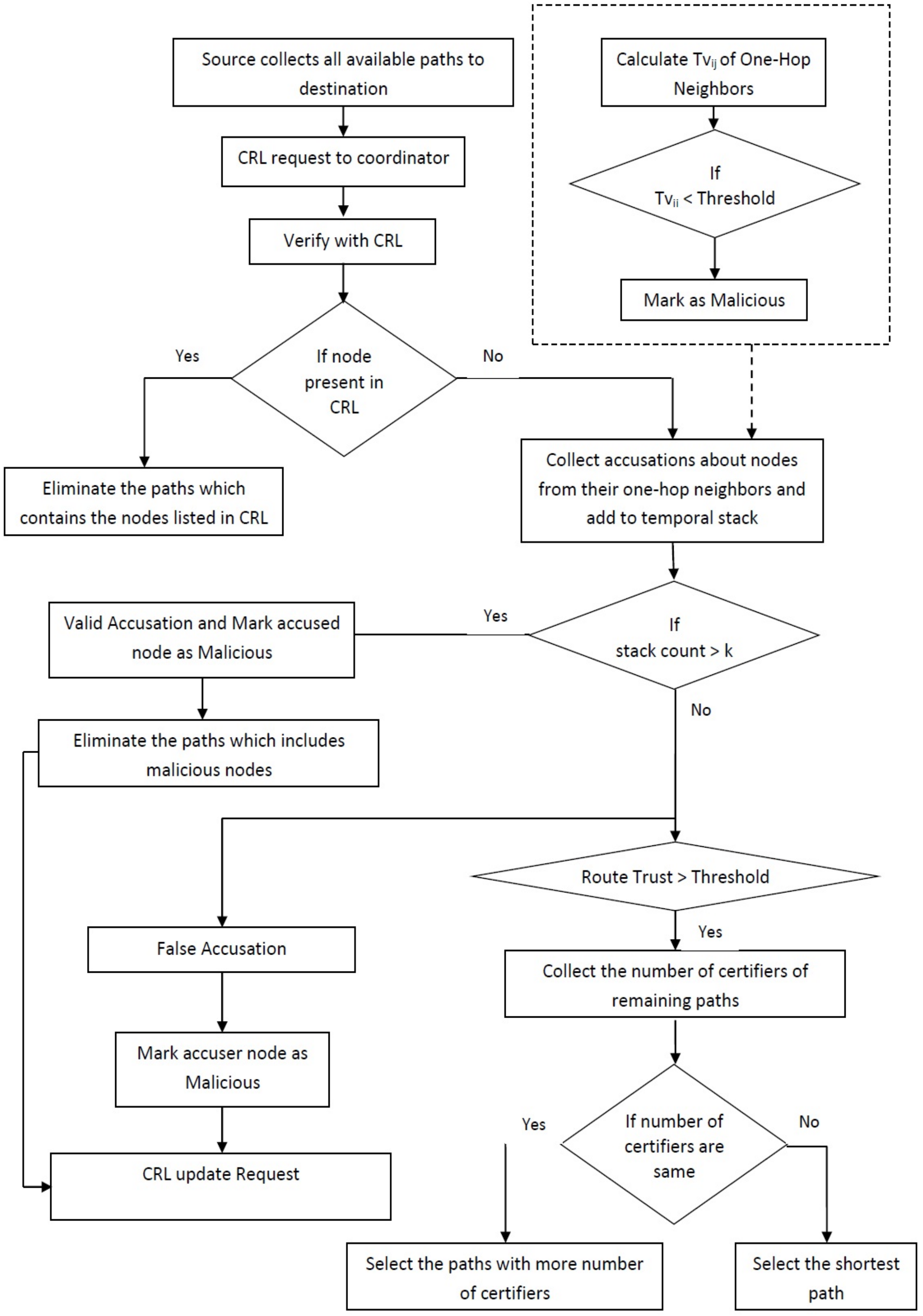
Source collects all available paths to destination
CRL request to coordinator
Verify with CRL
If node present in CRL
Yes
No
Eliminate the paths which contains the nodes listed in CRL
Calculate Tv_{ij} of One-Hop Neighbors
If Tv_{ii} < Threshold
Mark as Malicious
Collect accusations about nodes from their one-hop neighbors and add to temporal stack
Valid Accusation and Mark accused node as Malicious
Yes
If stack count > k
No
Eliminate the paths which includes malicious nodes
Route Trust > Threshold
False Accusation
Yes
Collect the number of certifiers of remaining paths
Mark accuser node as Malicious
Yes
If number of certifiers are same
No
CRL update Request
Select the paths with more number of certifiers
Select the shortest path

Fig 4.1 Route Discovery and Path Selection

Step 02: ***Collecting Accusations***

The next step is to gather the accusations about the nodes in the extensive routes from their one-hop neighbors as clarified in the section 4.4.1. The accusations are absolutely taking into account the computed trust value NT_{ij}. Every node keeps up a trust table with the list of malicious nodes (block list).

Step 03: ***Identifying Malicious Nodes***

Accusations around a node from its one hop neighbors will be put away in a transient stack, if the accuser node is not in the most recent retrieved CRL. A threshold "k" is utilized to locate the false accusations. If the stack count is littler than the dynamically ascertained threshold value k, it will be dealt with as a false accusation; and the accuser node will be counted as malicious node. Otherwise the accused node will be added to CRL.

Step 04: ***CRL update Request***

The path which contains such a malicious node won't be considered as a routing route. The recently distinguished malicious node must be added into the CRL. Therefore, source will send a CRL update request to the coordinator node. The coordinator node will reply with the updated CRL as clarified in section 4.2.5.

Step 05: ***Route Trust Computation***

The remaining paths can be considered for the routing reason. The source node will figure the route trust as clarified in the section 4.2.2.3. Source will choose the routes with a base threshold value of trust. The routes with a lower trust value than the threshold will be prohibited from the impressive paths. The nodes that are not having the minimum threshold is going to be isolated from the network by using the CRL update mechanism.

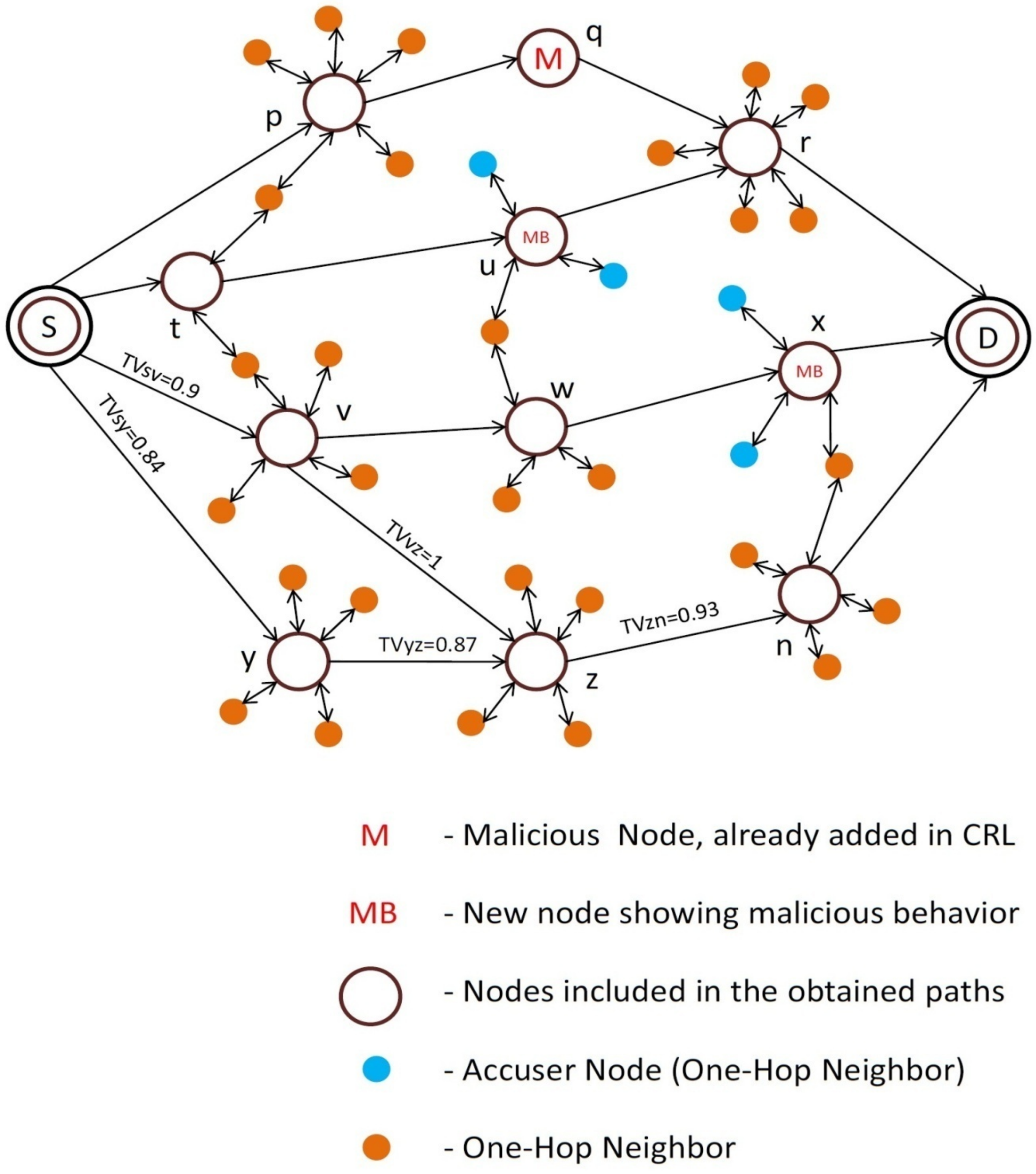
M
q
p
r
MB
u
t
S
D
x
MB
w
v
TVsv=0.9
TVsy=0.84
TVvz=1
TVyz=0.87
TVzn=0.93
y
z
n
M - Malicious Node, already added in CRL
MB - New node showing malicious behavior
○ - Nodes included in the obtained paths
● - Accuser Node (One-Hop Neighbor)
● - One-Hop Neighbor

Fig 4.2 Path Selection

Step 06: ***Comparison Based on Certifier Count***

In the event that there are "l" routes meet the required route trust limit, source node will choose the route with more certifiers of the destination node.

Step 07: ***Shortest Path Selection***

If the number of certifiers is equivalent then route with shortest hop count will be chosen.

Step 08: ***Certificate Exchange***

In the wake of selecting the path, source and destination certifies their public keys one another as clarified in section 4.3.

For example, consider the figure 4.2. We demonstrate our path selection approach by considering the network structure as in the figure. S and D indicate source node and destination node respectively. We can see 5 possible routing routes to the destination from source.

Path 01: {s-p-q-r-d}

Path 02: {s-t-u-r-d}

Path 03: {s-v-w-x-d}

Path 04: {s-v-z-n-d}

Path 05: {s-y-z-n-d}

Initially the source will verify the nodes with the latest retrieved CRL. Thus, source will identify the malicious node ‘q’. Then, source will exclude Path 01 from the list of considerable paths.

After the verification with the CRL, source gathers the accusations from the one-hop neighbors of the remaining nodes. Here, u and x are already recognized as malicious nodes by their one-hop neighbors. So source will get valid accusations from their neighbors. At that point source will offer request to coordinator node to add u and x into CRL. At that point, source will reject Path 02 and Path 03 from the list of significant paths.

Source will figures the route trust of remaining paths. Route trust indicates a joint likelihood at which packets will be sent If they are sent along the routing path. The repetition trust of Path 04 and Path 05 can be figured utilizing the following equation.

$NT_{path\ 04}(t) = NT_{sv} \times NT_{vz} \times NT_{zn} = 0.9 \times 1 \times 0.93 = 0.837$

$NT_{path\ 05}(t) = NT_{sy} \times NT_{yz} \times NT_{zn} = 0.84 \times 0.87 \times 0.93 = 0.680$

Let us assume that the base threshold value is settled as 0.4. It is clear that, both Path 04 and Path 05 can use as routing route. At that point source node will choose the route with more certifiers of the destination node. If the number of certifiers is equivalent then the route with least hop count will be chosen. In the wake of selecting the path, source and destination guarantees their public keys one another.

4.4.1 Accusation

It is utilizing a trust based accusation scheme to defend against the malicious nodes. Every node processes the trust value NT_{ij} of their one-hop neighbors and keeping up a trust table inside of it. The trust table contains the ID of neighbors, trust value and the block list. Here we are utilizing a threshold value for the trust. If the trust value is below the threshold, then the node will be stamped as the malicious node and that will be added to the block list. At whatever point a source node requests the trust details of a specific node, the one-hop neighbors will reply with an accusation packet, if that node is there in their block list.

The source node gathers the accusations from the virtual cluster members. Accusations around a node from its one hop neighbors will be put away in a transient stack if the accuser node is not in the most recent retrieved CRL. A threshold "k" is utilized to locate the false accusations.

$$\mathrm{k} = \frac{\text{Total number of one−hop neighbors} \times 78}{100} \tag{4.10}$$

In the event that the stack count is less than the dynamically figured threshold value k, it will be treated with as a false accusation; and the accuser node will be counted as malicious node. Otherwise the accused node will be added to CRL.

4.5 Certificate Revocation Technique

Every node is going to revoke its public key certification of the malicious nodes depends upon the updated CRL. After the route discovery, source gathers the Attack Accusations, about the nodes incorporated into the got paths, from their one-hop neighbors. Taking into account the accusations, it lists the malicious nodes before communication. Source will choose the communication path with respect to the most recent retrieved CRL from the coordinator node. If the listed nodes are already added in the CRL, then it will proceed with the communication.

Otherwise a CRL update is required, then source requests a CRL update to coordinator node before the communication. This procedure at first takes the following presumptions.

Let CRL be the certificate revocation list with respect to nodes in MANET.

Let R_{REQ} be the revocation initialization request.

Let R_{REP} be the revocation initialization reply.

Let ID_s be the source ID.

The steps involved in this scheme are as follows:

Step 1: Source transmits R_{REQ} signed by source itself for initiating the CRL update. The R_{REQ} includes ID of source node.

$$\text{Source} \xrightarrow{Sign_s\{R_{REQ}\}} \text{Coordinator Node} \tag{4.11}$$

Step 2: Upon receiving $R_{REQ,}$ coordinator node replies with R_{REP} message signed by coordinator node itself that contains the ID of source node.

$$\text{Source} \xleftarrow{Sign_c\{R_{REP}\}} \text{Coordinator Node} \tag{4.12}$$

Step 3: If signature verification fails,

Then, the message is discarded

Else

$$\text{Source} \xrightarrow{Sign_s\{Rp\text{^}RF\}} \text{Coordinator Node} \tag{4.13}$$

$$\text{Source} \xleftarrow{Sign_c\{CRL\}} \text{Coordinator Node} \tag{4.14}$$

If signature verification falls flat, the message is disposed of. Otherwise source begins to transmit the report and refresh (RP^RF) message to coordinator node. RP^RF contains IDs and distinguished node ID's. Coordinator node then registers the reported nodes and answers with updated CRL (CRL with increased CRL number).

Case 01:

In the event that CRL is not got by source node regardless of consistent service demand of the coordinator node. At that point, source rehashes the above procedure by expanding the TTL value until gets the updated CRL.

Source appends the CRL into the current routing messages which is used by the certificate exchange mechanism. The revocation information affixed with routing message is required to be checked just at the season of changes in the revocation list number.

Case 02:

In the event that the routing message originates from revoked nodes, then the message is disposed of. The node which gets the updated CRL will check the new list for getting the points of interest of recently added nodes to the CRL. In the event that it could locate a certified public key of any malicious nodes in its own certified list, then it revokes that certificate before next communication.

4.6 Overall Algorithm

The entire process of the proposed framework is described using the following algorithm.

Step 01 – Network Architecture

The architecture is developed such that it incorporates a coordinator node, servers and normal mobile nodes. The coordinator goes about as a mediator node for transmitting messages from normal mobile nodes to the servers.

Step 02 – Public/Private Key Generation

Every mobile node produces its own public/private key pairs utilizing server-signed public keying system. The coordinator node helps in producing the publicly-recoverable public key for any node Ni without the information of the ensuing private key. The coordinator node goes about as a distributed trusted authority. It consolidates the shares of (t+1) servers for registering signature parameter.

Step 03 – Trust Calculation

The nodes in the network are validated using the trust management technique named as Trust Prediction Model.

Step 04 – Multi-Path Certificate Exchange

After the generation of public/private key pairs, multi-path certificate exchange method is utilized where public key of the nodes are certified by diverse nodes. The authentication is additionally performed commonly.

Step 05 – Malicious Node Detection

Each node will maintain a Trust Table for every one-hop neighbor. It contains the Trust Value NT_{ij}, that node v_i has about its neighbors. The Block List Threshold (BT) is used to detect malicious nodes. If a node's trust value is smaller than n, standing on the evaluating node's point of view, it will be treated as a malicious node and remarked in the evaluating node's local Trust Table.

Step 06 – Route Discovery

Source will collect the certifiers of the destination node and all possible paths to the destination.

Step 07 – Security Considerations

Source considers just those paths, which are free from malicious nodes. It will choose the communication path construct just with respect to the most recent retrieved CRL from the coordinator node. Source performs step 8 to 12 for guarding against the malicious nodes.

Step 08 – Collecting Accusations

Subsequent to performing the route discovery handle, the source will gather the Attack Accusations, about the nodes incorporated into the acquired paths, from their one-hop neighbors. Accusations around a node from its one hop neighbors will be put away in a transient stack if the accuser node is not in the most recent retrieved CRL. A threshold "k" is utilized to locate the false accusations. If the stack count is littler than the dynamically figured threshold value k, it will be dealt with as a false accusation; and the accuser node will be counted as malicious node. Otherwise the accused node will be added to CRL.

Step 09 – Path Selection

After the route discovery process, if one or more routes are found, then source will register the Route Trust. If there are "l" routes meet the required route trust limit, source node will choose the route with more certifiers of the destination node. If the number of certifiers is equivalent then the route with least hop count will be chosen. Subsequent to selecting the path, source and destination ensures their public keys one another.

Step 10 – CRL Update

The coordinator node performs the CRL update mechanism for shielding against the malicious nodes. In light of the Attack Accusations, source distinguishes the malicious nodes before communication. The source node checks the CRL list and distinguishes if any new node showing malicious behavior. In the event that so source requests for a CRL update to the coordinator node. The coordinator node updates the CRL and sends back the list to the source for certificate revocation process.

Step 11 – CRL Distribution

Source appends the CRL into the current routing messages which is used by the certificate exchange mechanism. The revocation information annexed with routing message is required to be confirmed just at the season of changes in the revocation list number. During the path selection process the source node requests the CRL and depends upon entry in the CRL, those routes will be discarded where malicious nodes involves.

Step 12 – Certificate Revocation

Each node is going to revoke its public key certification of the malicious nodes depends upon the updated CRL.

4.7 Simulation Results

4.7.1 Simulation Model and Parameters

Simulations were performed utilizing Network Simulator (NS-2), especially mainstream in the ad hoc networking group. The MAC layer protocol IEEE 802.11 with a data rate of 11 Mbps is utilized as a part of all simulations. The transmission range is set to 250m. The propagation model is Two Ray Ground. The aggregate number of nodes is set to 100 nodes in 1000m x1000m network range.

In our simulation, the minimal node speed is 5 m/s. The constant bit rate (CBR) traffic generator is utilized to set up the association designs with diverse arbitrary seeds. The measure of certificates was likewise set to 512 bytes. The aggregate number of connections in the network was set to 20 connections. The simulation results are the normal of 10 runs.

In the simulation, attacks are simulated where the attacker nodes send spurious certificates to the nodes which have requested for those certificates. These attacks can be isolated attacks where each attacker confirms an alternate public key. Be that as it may, the attackers might likewise dispatch an agreeable attack where a group of attackers intrigue and send certifications for the same public key that is spurious.

Both these sorts of attacks-isolated and plot are simulated. The rate of attacker nodes is altered as 10% of the aggregate number of nodes in the network (ie) 10 attackers. Node initialization at the network bootstrapping stage is additionally simulated. It is demonstrated that every node has effectively executed the initialization venture by exchanging imperative number of certificates with the honest nodes in the network. Our simulation settings and parameters are compressed in 4.3.

Table 4.2 Simulation Settings for TPMCER

No. of Nodes	100
Area Size	1000 X 1000
Mac	802.11
Radio Range	250m
Simulation Time	100 to 500 sec
Traffic Source	CBR
Packet Size	512
Speed	5m/s
Pause time	5 seconds
Load	1000 Kb.
No. of attackers	1 to 10

The following are the assumptions used for the proposed framework,

- A malicious node can compromise the key, create packet drop attack, routing overflow attack etc.
- A trusted node will be having a trust value greater than 0.45.
- A threshold trust is fixed as 0.45.
- Malicious node will be having a trust value less than 0.45
- Trust value 1 refers to full trust and 0 refers to complete distrust.

- The coordinator node is the in charge of CRL update.
- Accusation threshold is fixed as 78%.
- Protocol used is AOMDV.

4.7.2 Performance Metrics

We compare the proposed Trust Prediction Model for Certificate Exchange and Revocation (TPMCER) technique with Trusted Certificate Exchange and Revocation (SOKMTC) technique proposed in chapter 3.

We evaluate mainly the performance according to the following metrics [91]:

- **Average end-to-end Delay:** the normal time taken by the data packets from sources to destinations, including buffer delays during a route discovery, lining delays at interface lines, retransmission delays at MAC layer and propagation time.
- **Packet Delivery Ratio:** or packet throughput, the fraction of the data packets conveyed to destination nodes to those sent by source nodes.
- **Packet Drop** It is the number of packets dropped during the transmission.
- **Misdetection Ratio:** the proportion of the number of nodes whose behavior (malicious or considerate) is not recognized effectively to the real number of such nodes in the network.
- **Routing packet overhead**: the number of control packets (including route request/reply/update) for establishing connection over a period of time.
- **Resilience against Node Capture:** the fraction of communications compromised to the total number of communications by a capture of x-nodes.

4.7.3 Results

Varying Number of Attackers

The number of attackers is increased from 1 to 10 and the performances of the techniques are measured in terms of Delivery Ratio, Misdetection and Resilience.

Figure 4.3 shows the average Packet delivery Ratio of the schemes, when the attackers are increased from 1 to 5. We can see that the delivery ratio decreased linearly as the attacker increases. But, the delivery ratio of our proposed TPMCER is greater than SOKMTC.

The ratio of the number of nodes whose behavior is not identified correctly to the actual number of such nodes in the network is shown in figure 4.4. Our proposed method is capable to detect more malicious nodes while comparing with SOKMTC.

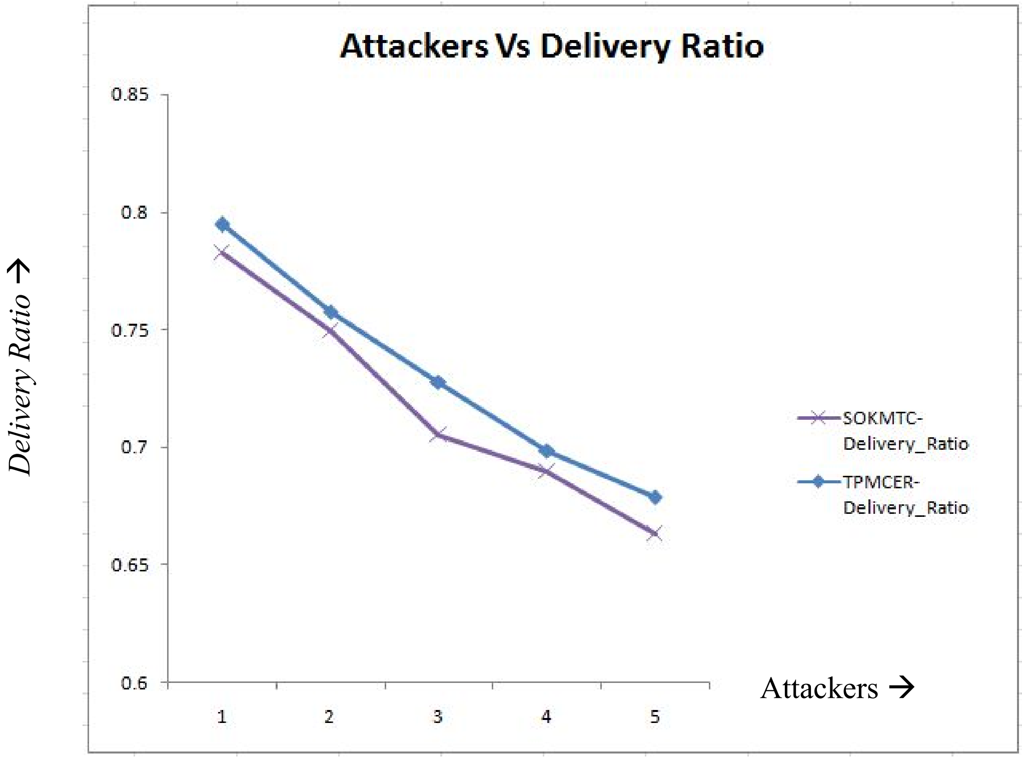

Fig 4.3 Packet Delivery Ratio

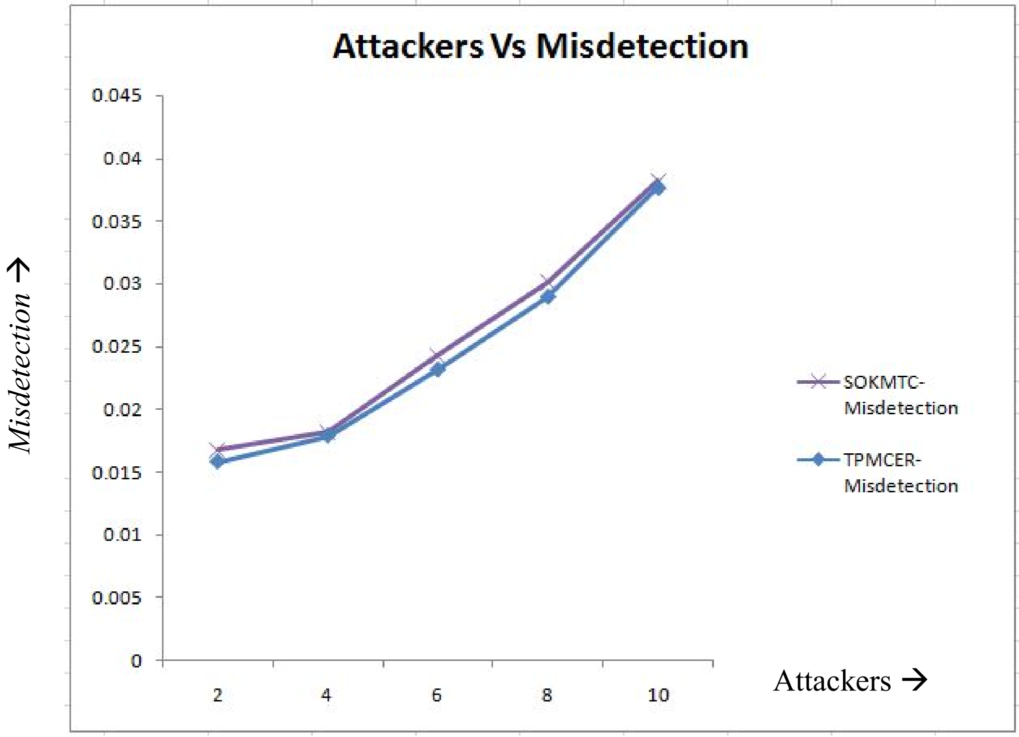

Fig 4.4 Misdetection Ratio

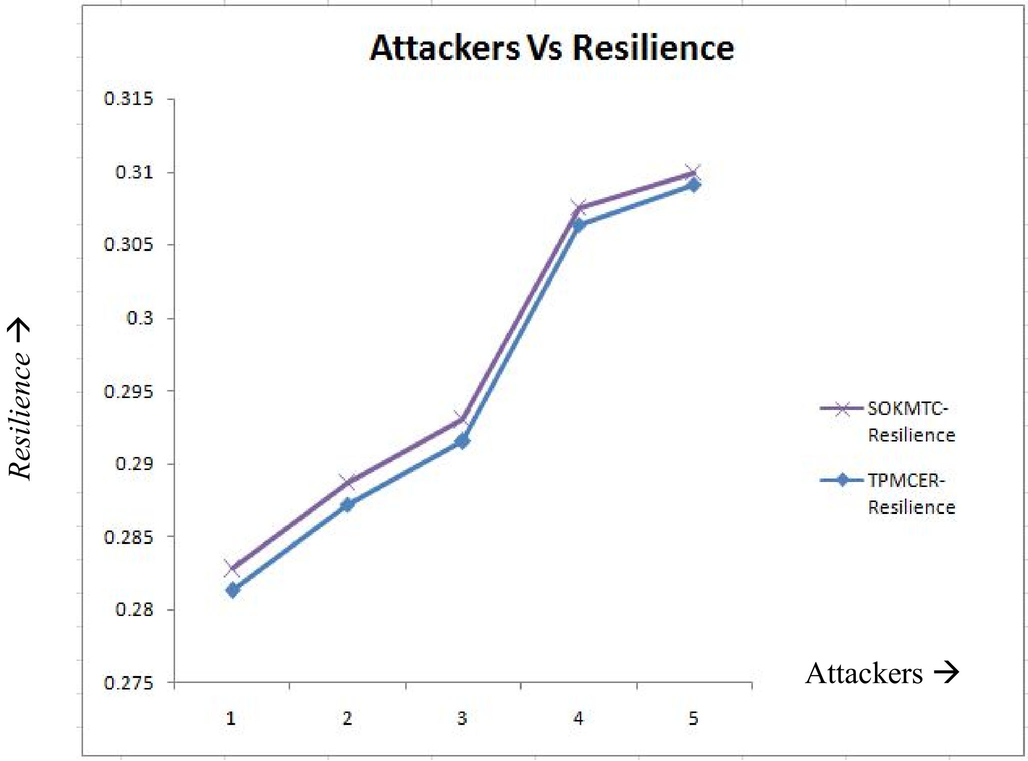

Fig 4.5 Resilience against Node Capture

The result of fraction of compromised communications is shown in figure 4.5. Because of the trust prediction mechanism, the number of compromised communications is less in TPMCER. Hence the proposed TPMCER is more resilient than SOKMTC.

Varying Number of Nodes

The CBR data packets and control packets dropped due to the attackers, presented in figures 4.6. As the number of attacker increases, more data packets are dropped. But TPMCER has less packet drops when compared to SOKMTC.

Figure 4.7 depict the delay involved in the communication by each pair of source and destinations. The number of nodes is varied from 10 to 50, and corresponding delay for the TPMCER and SOKMTC are measured. The proposed method outperforms the SOKMTC in case of delay.

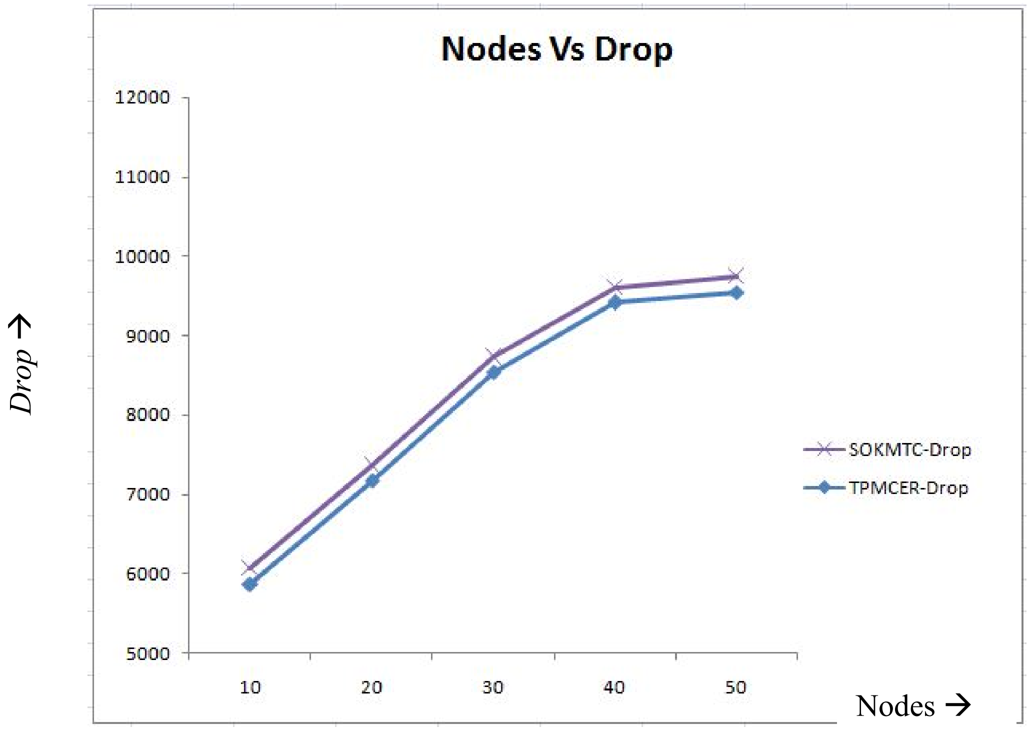

Fig 4.6 Packet Drop

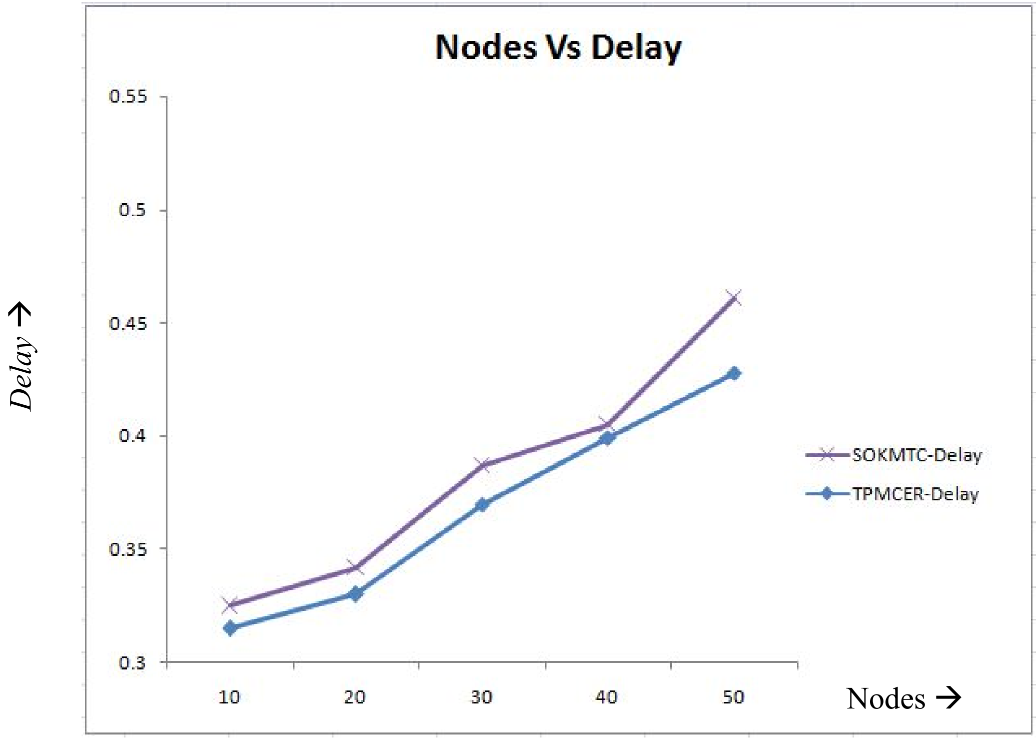

Fig 4.7 Average end-to-end Delay

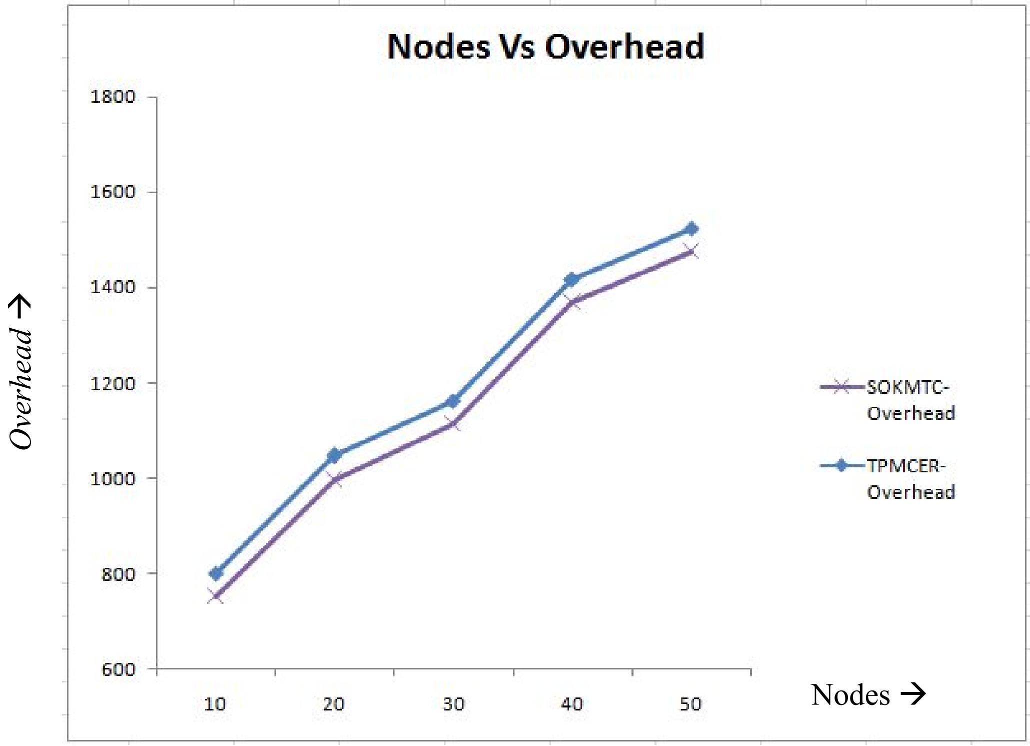

Fig 4.8 Routing packet overhead

Figure 4.8 shows the Routing packet overhead of the schemes, when the nodes are increased from 10 to 50. We can see that the overhead of our proposed TPMCER is greater than the SOKMTC since the proposed method contains the trust prediction mechanism for certificate exchange and revocation, but it is more resilient than other schemes.

4.8 Conclusion

In this chapter, we have proposed a trust prediction model for certificate exchange and revocation based on accusations in mobile ad hoc network. The proposed architecture comprises of the coordinator node, servers and customary mobile nodes. The nodes that issued the certificates are validated utilizing the trust prediction model. The trust value is computed from the three unique sorts of trust, for example, verifiable trust, current trust and route trust. At that point multi-path certificate exchange procedure is utilized where public key of the nodes are certified by diverse nodes. The source performs certificate revocation process for safeguarding against the malicious nodes. By simulation results, we have demonstrated that the proposed approach upgrades the security against node capture attacks and enhances the packet delivery ratio and detection rate. The approach also achieves better resilience reduced delay and packet drop even though the packet overhead is high.

CHAPTER 5

SECURE MULTIPATH ROUTING PROTOCOL FOR CERTIFICATE EXCHANGE IN MANET

CONTENTS

5.1 Overview

This chapter studies the impact of multipath optimized link state routing protocol. The trust management, key generation, certificate exchange and revocation mechanisms explained in chapter 3 are consolidated in the M-OLSR protocol. Here the architecture comprises of normal nodes and shareholder nodes. A random shift mechanism is utilized to choose the coordinator node among shareholder nodes. The coordinator node is in charge of the maintenance of CRL. Here, we are also addressing the link failure problem in source routing. Trusted re-computation of routes is introduced when link failure occurs. The proposed scheme is simulated and compared with the existing methodologies are presented.

5.2 Multipath Optimized link state routing (M-OLSR)

The basic optimized link state routing protocol (OLSR) is a proactive routing protocol, where the routes are going to be updated every time and maintained in the routing table [94] [95]. It is a link state protocol in which each node is going to send HELLO and topology control (TC) messages periodically to show its existence in the network. The route discovery process is carried out by identifying a set of designated nodes called multi point relay (MPR) nodes. So the flooding of the link state information is not needed because of the introduction of these MPR nodes.

Figure 5.1 shows normal broadcasting situation, where the packets will be sent to all the one hop neighbors. Due to the flooding of link state information, the overhead is very high in the normal scenario. In OLSR, as explained above a set of designated nodes are assigned as Multi Point Relay (MPR) nodes, only these nodes are participating in the route discovery process.

Figure 5.2 demonstrates the MPR flooding situation utilized as a part of OLSR protocol, where the broadcasting of packets is just done by MPR nodes. It diminishes the number of copy retransmissions while sending a broadcast packet. It additionally limits the arrangement of nodes retransmitting a packet from all nodes (normal flooding) to a subset of all nodes. The measure of this subset depends on the topology of the network [96]. Here by using the MPR flooding strategy the overhead of the routing protocol can be reduced and by avoiding the unnecessary data forwarding, the capability level of a particular node can be maintained.

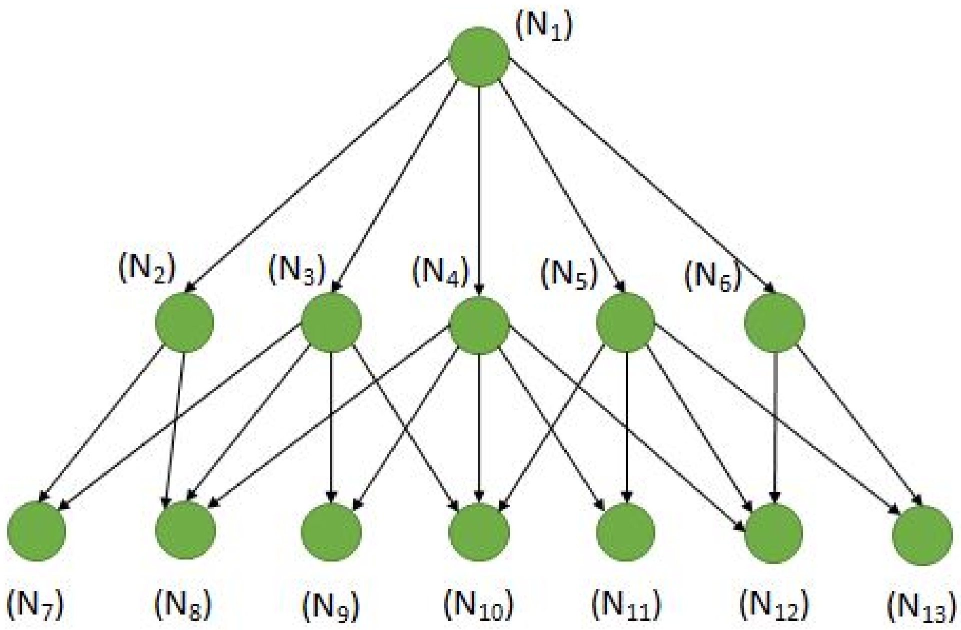

Fig 5.1 Regular Flooding

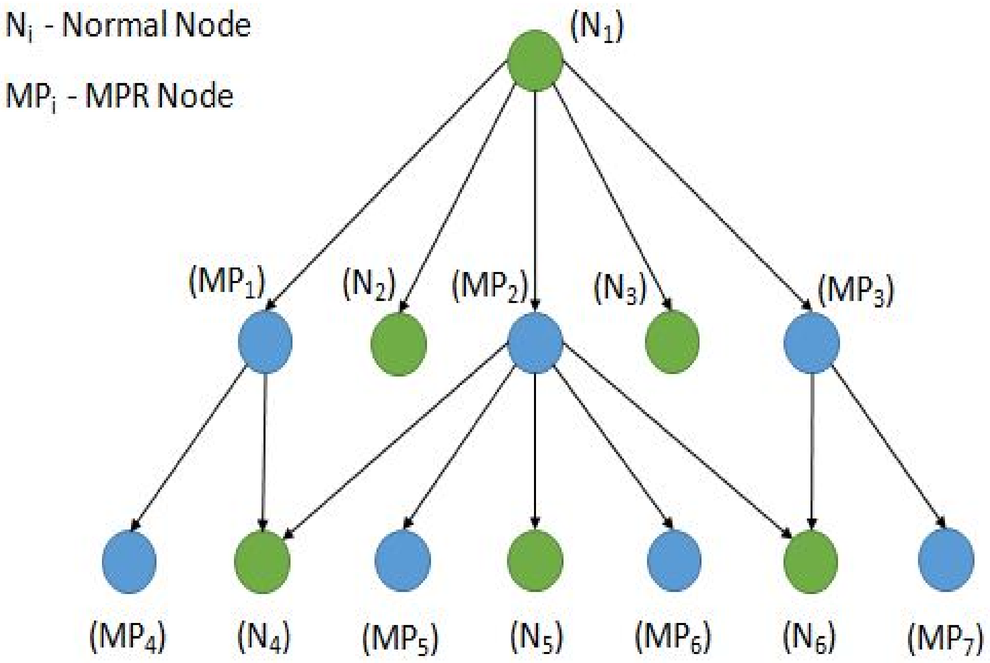

Fig 5.2 MPR Flooding

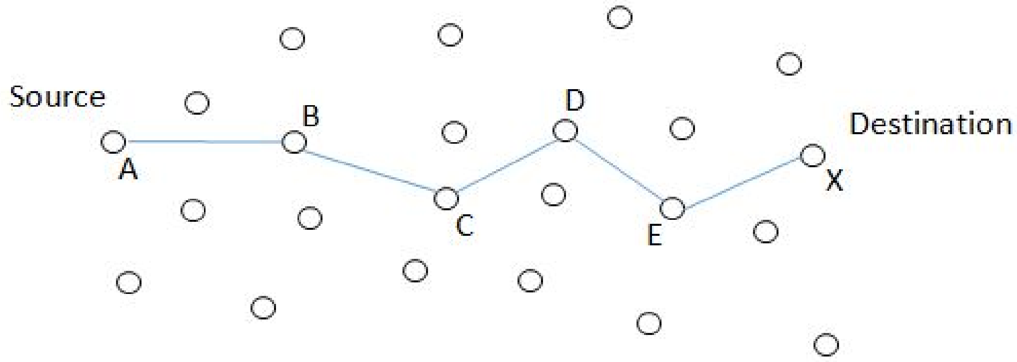

Figure 5.3 Building a route in OLSR

In OLSR the route is going to be identified with the assistance of TC messages. Figure 5.3 demonstrates the route computation in OLSR protocol. In the event that A needs to discover a path to node X, it first discover the pair [A, B], then [B, C], then [C, D], then [D, E] and [E, X]. So the path is made sense of. It is A-B-C-D-E-X. In OLSR, routes are controlled by nodes every time they get another Topology Control messages (TC or HELLO). The routes to all the conceivable destinations are saved in the routing table [94].

The M-OLSR can be viewed as a sort of hybrid multipath routing protocol which consolidates the proactive and reactive components. It sends out HELLO and TC messages occasionally to recognize the network topology, much the same as OLSR. But, M-OLSR does not always keep a routing table. It is going to identify multiple paths for data forwarding and from these multiple paths the shortest path is going to be selected for data forwarding. During the data transmission, when a link failure occurs, a dynamic route re-computation process is also takes place in M-OLSR. The major functionality of M-OLSR has two sections: topology detecting and route computation.

The topology sensing mechanism is used to identify the one hop neighbor and which one hop neighbor can be assigned the MPR status. This sensing is done by using two control messages as explained earlier. The route computation is done by using the Multipath Dijkstra Algorithm [97,98] to compute the multiple path received from the information by the topology sensing process.

The topology detecting and route computation make it possible to discover different paths from source to destination. The path should be identified in such a way that there will not be any loop and from the obtained paths the protocol should be able to identify the shortest one. The protocol should be able to identify the path break during the data transmission. So route recovery, when path break occurs and loop detection are also addressed in M-OLSR multipath routing protocol. The route recovery can successfully decrease the packet loss. For M-OLSR, an on-demand scheme is utilized to maintain multiple routes from the computation of various routes for every possible destination. The multiple paths are going to be identified by using multipath dijkstra algorithm [94].

A route recovery mechanism is used to address the disadvantage of the source routing. Before an intermediate node tries to forward a packet to the next hop as indicated by the source route, the node first checks whether the following hop in the source route is one of its neighbors (by checking the neighbor set). Provided that this is true, the packet is sent normally. If not, it is understood that, the next hop is no longer accessible. At that point the node will re-compute the route by using the dynamic route re-computation mechanism and forward the packet by utilizing the newly identified route. The route re-computation mechanism is not included in any of the multi path protocols, where after selecting one path, it is going to be used for communication. But in M-OLSR the path break during the communication is also addressing.

In Fig. 5.4 we show an illustration of route recovery. Node S is attempting to send packets to D. The original multiple paths we have are S->P->Q->D and S->R->T->W->X->D. Be that as it may, node W moves out of the transmission range of node T and makes the second path distracted.

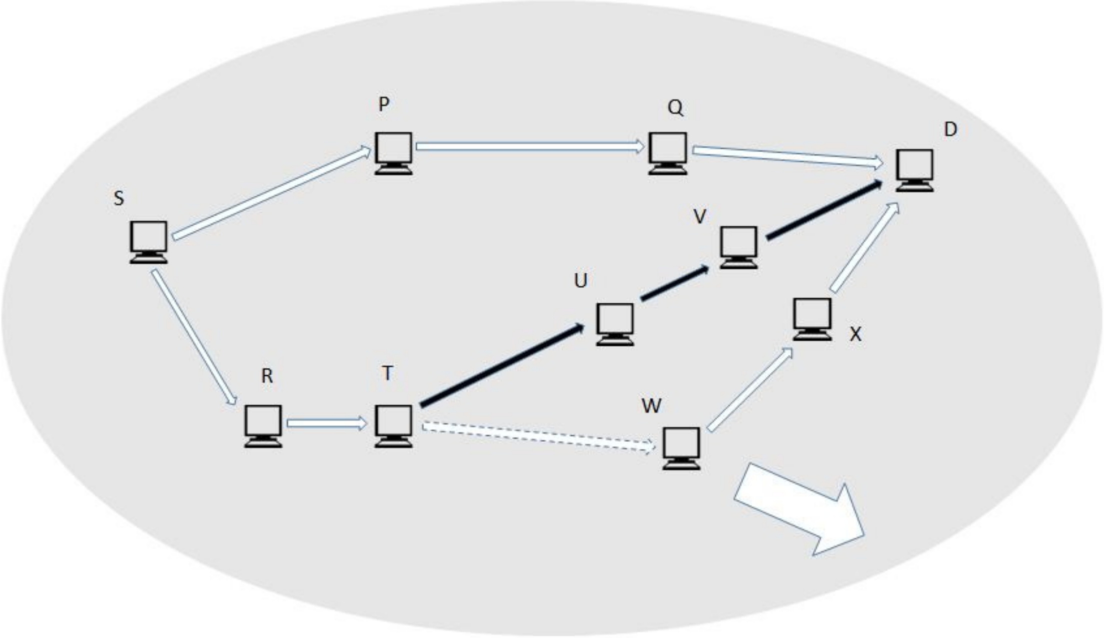

Figure 5.4 Route Recovery in MOLSR

The source node S is not able to recognize the link failure immediately (in light of the delay and long interim of TC messages) and continues sending the packets along the path and every one of these packets are dropped during this period if just the source routing is utilized. With route recovery, when the packet arrives, node T will first check if node W is still one of its neighbors, before sending the packet as indicated by the source route. If not, node T will re-compute the route to node D, and acquire T->U->V->D. At that point the following packets will be sent through the new path [94].

5.3 Security extensions to M-OLSR Protocol

Several security enhancements is proposed in the multipath protocol. A mechanism to address wormhole attack is presented [99], where the attack is going to be identified neighborhood sensing process itself, where the HELLO and the topology control based on acknowledgement is employed. The dynamic nature of nodes to become malicious cannot be able to detect by using this mechanism. During the MPR detection scenario the nodes are going to be assessed regarding their behavior depends upon the control message dialog delivery.

A property based intrusion detection mechanism is proposed [100], based on the availability property of a nodes. Here, if a route exists from a mobile node to another then the protocol should be able to identify it. Several security properties are defined and the protocol checks whether these security properties are violating or not. By doing this, intrusion is going to detect and those paths are no longer going to be considered for path selection.

A security scheme based on the shared secret key based algorithm [101] is proposed to enhance the security of the protocol. Here Shamir secret key algorithm has applied for securing the protocol. The secret shares are going to be distributed among the nodes and a minimum of shares are going to be retrieved, combined to generate the share.

An extension is proposed for improving the quality of service of routing protocol [102]. Here the different classes of flows; control flows, delay flows, bandwidth flows, best effort flows are addressing for enhancing the quality of service provided by the protocol.

A reputation based mechanism is used to address the node isolation problem [103] [104]. Here the reputation of each node is calculated before data transmission. The node isolation attack allows at least one node to prevent a specific node from receiving data packets, from other nodes that are more than two hops away.

For enhancing security in M-OLSR trust management mechanism should be incorporated. Here we are proposing a trust based route recovery mechanism and a timestamp exchange mechanism to defend against link failure and reply attacks.

5.4 Proposed Architecture

The proposed architecture consists of normal nodes, MPR nodes and shareholder nodes. Shareholder nodes are specially designated MPR nodes. Figure 5.5 shows the proposed architecture of the Secure Multipath Key Management technique. It includes normal nodes (N_1, N_2 …), OLSR multipoint relay nodes (MP_1, MP_2 …), number of designated MPRs as share holder nodes (SH_1, SH_2 …) and a coordinator node (SH_C).

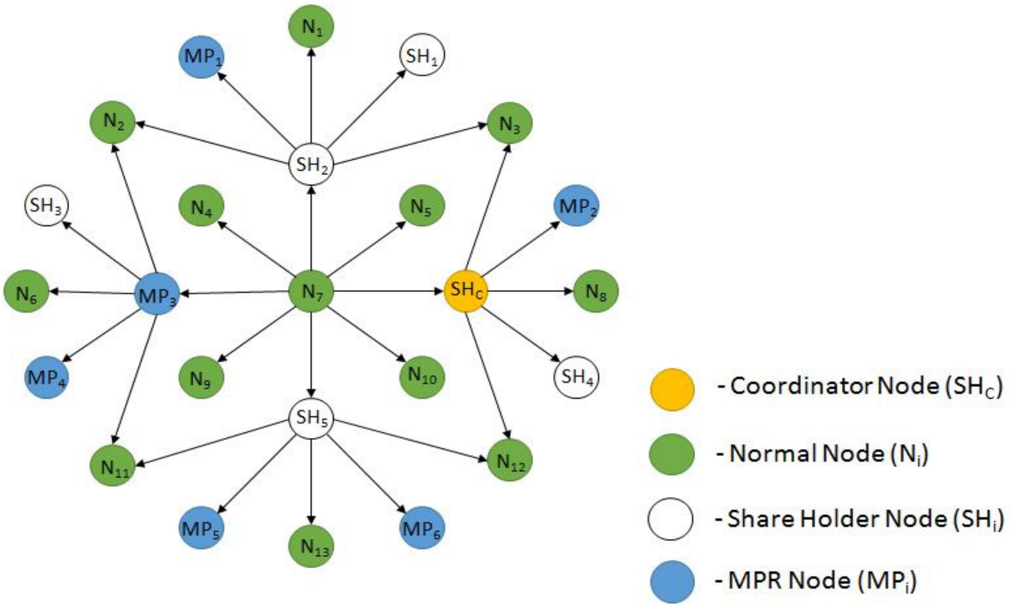

Figure 5.5 Architecture of Secure Multipath Key Management Technique

A random shift mechanism among the SH nodes is used to select the coordinator node. The coordinator node is responsible for the maintenance of CRL. When a new node enters into the network, the coordinator node sends a secret a value to the corresponding node which requires a public/private key pair generation. The receiver node sends back a newly generated secret value for getting the signature parameter. The coordinator node acts as a distributed trusted authority. The coordinator node has to identify minimum number of shareholder nodes for collecting the partial shares. It combines the shares of minimum ‘n’ shareholder nodes for generating the signature parameter. By using this signature parameter, each node can generate its own public/private key pair as explained in section 3.2.1. The shareholder identification process by the coordinator is combined with the neighborhood sensing mechanism of M-OLSR as explained in section 5.4.1.

The multiple paths are going to be identified by using the M-OLSR protocol route identification process. Here we are incorporating the eigen vector centrality trust calculation mechanism (explained in section 3.2.2) with the multipath route discovery process to ensure trusted path selection. Here we are proposing a trust based route recovery mechanism (explained in 5.4.2) and a timestamp exchange mechanism (explained in 5.4.3) to defend against link failure and replay attacks.

5.4.1 Shareholder Identification Process

The HELLO and TC messages are used to identify the link stability and the MPR nodes in the network. Let N_{in} represents the node entering into the network (say inward node), it sends the HELLO message to handshake with the coordinator node. Once the coordinator node identifies the existence of a new node, it sends hello messages to its one hop neighbors. The multi point relay mechanism is used to identify the shareholder nodes.

1) Set the reserved bit in HELLO messages so that every node is able to identify the existence of shareholder nodes and their ability to offer service.
2) As HELLO message can only identify SHi within one hop, to collect minimum of n SHs, every MPR uses the TC messages.
3) The reserved bit in TC messages is set to designate the number of nodes in MPR group.
4) SH_C sends share requests to the identified SH nodes.
5) SH nodes reply with the partial shares.
6) After receiving ‘n’ number of partial shares, it verifies the validity by secret sharing mechanism.
7) After the reception of ‘n’ valid shares, SH_C extracts the partial shares.
8) SH_C forwards the partial signature to N_i for generating the public/private key pair.

5.4.2 Trust Based Route Recovery Mechanism

A trust based route recovery mechanism is incorporated in the M-OLSR multipath route recovery methodology. After certifying the paths, during the transmission due to mobility or link failure, a particular path may become unavailable. The source node may not be able to detect link failure at that time due to delay in sending TC messages. In M-OLSR before an intermediate node tries to forward a packet to the next hop as per the route, the node first checks, whether the neighbor node is valid or not. If the neighbor node is not valid the node will take its best effort to re-compute the route by using the same trusted source routing mechanism and forward the packet by using the new route. The proposed solution enhances the existing M-OLSR protocol to defend against different kinds of security attacks. Figure 5.6 demonstrates the same.

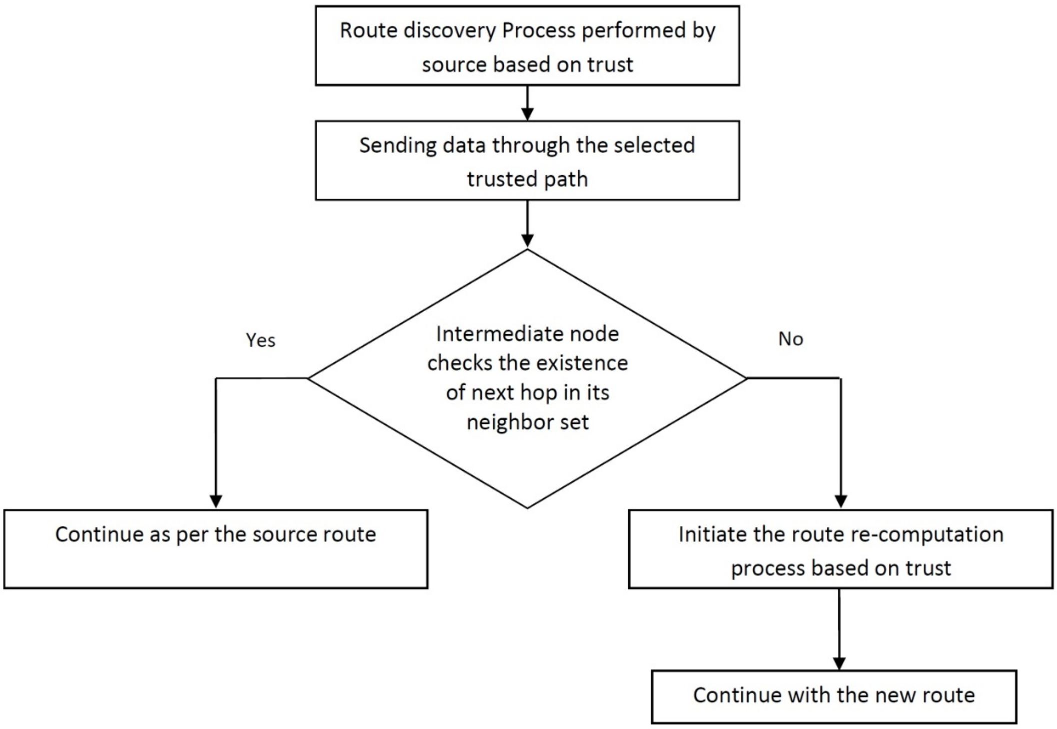
Route discovery Process performed by source based on trust
Sending data through the selected trusted path
Intermediate node checks the existence of next hop in its neighbor set
Yes
No
Continue as per the source route
Initiate the route re-computation process based on trust
Continue with the new route

Figure 5.6 Trusted Route Re-computation

5.4.3 Timestamp Exchange Mechanism

A replay attack happens when an attacker node listens to the signed traffic packets and afterward re-broadcasts the same packets later on [105]. This can be anticipated utilizing timestamp exchange mechanism. This is on the grounds that, the timestamp exchange happens among the neighbors that have no enlisted timestamp of one another. We consider the scenario of timestamp exchange among two neighbor nodes n_x and n_y which is explained as follows. Let IP_x be the IP address of $n_{x,}$ IP_y be the IP address of n_y TS_x be the time stamp of n_x, TS_y be the timestamp of n_y, and a secure hashing algorithm is used to produce the message digest. When n_x receives a signed message from n_y where n_x is not having any registered time value, it initiates the timestamp exchange process.

n_x broadcasts the test message (TM) to n_y as a message digest. The message digest contains the IP_y, R_x, Shared Key (K_{sh}) and TS_x.

$$n_x \xrightarrow{TM} n_y \qquad (5.3)$$

R_x is the random number used to generate the message digest. Upon receiving TM, n_y generates digest of its IP address, the received random number, the shared key K_{sh} and the timestamp TS_x i.e. $d(IP_{y,} R_x, K_{sh,} TS_x)$. The node n_y calculates the time required to receive a packet from n_x (i.e. d_x) by using the time stamp received from n_x and the time when the TM receives. Then it generates random number, R_y and transmits the reply message to n_x.

$$n_x \xleftarrow{REPLY} n_y \qquad (5.4)$$

The reply message contains the message digest of the IP_x, R_y, and TS_y. When n_x receives the reply message from n_y, it initially validates the data using the shared key K_{sh}. If d (IP_x, R_y, TS_y) is verifiable, then, TS_y is utilized to generate time difference among n_x and n_y and that will be recorded as d_y, thus completing the timestamp exchange process. While receiving the data packets, the receiver node will compute the timestamp difference which is nothing but the time taken to reach a packet from source to destination, using the equation 5.5.

$$TS_{diff} = TS_{pr} - TS_{at} \qquad (5.5)$$

Where, TS_{pr} is the packet reception time at the receiver node and TS_{at} is the appended timestamp with the packet by the source node. The computed TS_{diff} will be vefified with $d_{(source\ node)}$, computed during the timestamp exchange process. The difference can be allowed to certain extend called time stamp slack. It is represented by using the equation 5.6.

$$TS_{slack} = TS_{diff} - d_i \qquad (5.6)$$

Where, TS_{slack} is the slack in the calculated timestamp variation. Certain extend of variation in timestamp is allowed and if it is more, it is assumed as a replay attack and the receiver is going to drop the packet, which in turn reduces the route trust.

5.5 Simulation Results

5.5.1 Simulation Model and Parameters

Simulations were performed utilizing Network Simulator (NS-2), especially famous in the ad hoc networking group. The MAC layer protocol IEEE 802.11 with a data rate of 11 Mbps is utilized as a part of all simulations. The transmission range is set to 250m. The propagation model is Two Ray Ground. The aggregate number of nodes is set to 100 nodes in 1000m x1000m network region. In our simulation, the minimal speed is 5 m/s.

The source-destination pairs are spread randomly over the network. The ns-2 constant bit-rate (CBR) traffic generator is utilized to set up the association designs with diverse random seeds. Every node has one CBR traffic association with a solitary unique destination. Sources start time is consistently distributed over the initial 60 seconds of the simulation time. We fluctuate the load value as 50,100,150,200 and 250Kb. The measure of certificates was additionally set to 512 bytes. The aggregate number of connections in the network was set to 7 connections.

In the simulation, attacks are simulated where the attacker nodes send spurious certificates to the nodes which have requested for those certificates. These attacks can be isolated attacks where each attacker guarantees an alternate public key. Nonetheless, the attackers might likewise dispatch a helpful attack where a group of attackers connive and send certifications for the same public key that is spurious. Both these sorts of attacks-isolated and conspiracy are simulated. The rate of attacker nodes is settled as 10% of the aggregate number of nodes in the network (ie) 10 attackers. Our simulation settings and parameters are summarized in table 5.1

Table 5.1 Simulation Settings for SMRP

No. of Nodes	100
Area Size	1000 X 1000
Mac	802.11
Radio Range	250m
Simulation Time	10 to 50 sec
Routing Protocol	M-OLSR
Traffic Source	CBR
Packet Size	512
Speed	5m/s
Pause time	5 seconds
No. of attackers	1 to 10

The following are the assumptions used for the proposed framework,

- A malicious node can compromise the key, create packet drop attack, routing overflow attack etc.

- A trusted node will be having a trust value greater than 0.5.
- A threshold trust is fixed as 0.5.
- Malicious node will be having a trust value less than 0.5
- Trust value 1 refers to full trust and 0 refers to complete distrust.
- The designated shareholder node will be the coordinator node.
- The coordinator node is the in charge of CRL update.
- Protocol used is m-OLSR.

5.5.2 Performance Metrics

We compare the proposed Self-Organized Key Management with Trusted Certificate using OLSR (SMRP) technique with Multipath Optimized link state routing (M-OLSR) and Self Organized Key management for trusted certificate exchange and revocation (SOKMTC) scheme. We evaluate mainly the performance according to the following metrics [91]:

- **Average end-to-end Delay:**

  The normal time taken by the data packets from sources to destinations, including support delays during a route discovery, lining delays at interface lines, retransmission delays at MAC layer and propagation time.

- **Packet Delivery Ratio:**

  The amount of the data packets delivered to destination nodes to those sent by source nodes.

- **Packet Drop:**

  It is the number of packets dropped during the transmission.

- **Misdetection Ratio:**

  The proportion of the number of nodes whose conduct (malicious or generous) is not recognized accurately to the genuine number of such nodes in the network.

- **Routing packet overhead**:

  The number of control packets (including route request/reply/update) for establishing connection over a period of time.

- **Resilience against Node Capture:**

  The fraction of communications compromised to the total number of communications by a capture of x-nodes.

5.5.3 Results

Varying Number of Attackers

The number of attackers is increased from 1 to 10 and the performances of the techniques are measured in terms of Delivery Ratio, Misdetection and Resilience.

Figure 5.7 shows the average Packet delivery Ratio of the schemes, when the attackers are increased from 1 to 5. We can see that the delivery ratio decreased linearly as the attacker increases. But, the delivery ratio of our proposed SMRP is greater than the existing M-OLSR and the proposed SOKMTC. The delivery ratio is high because the trusted route discovery is happening in the proposed method. During when a path break occurs, the path re-calculation is happening dynamically. So we can be able to address the path breakage effectively so as to improve the delivery ratio.

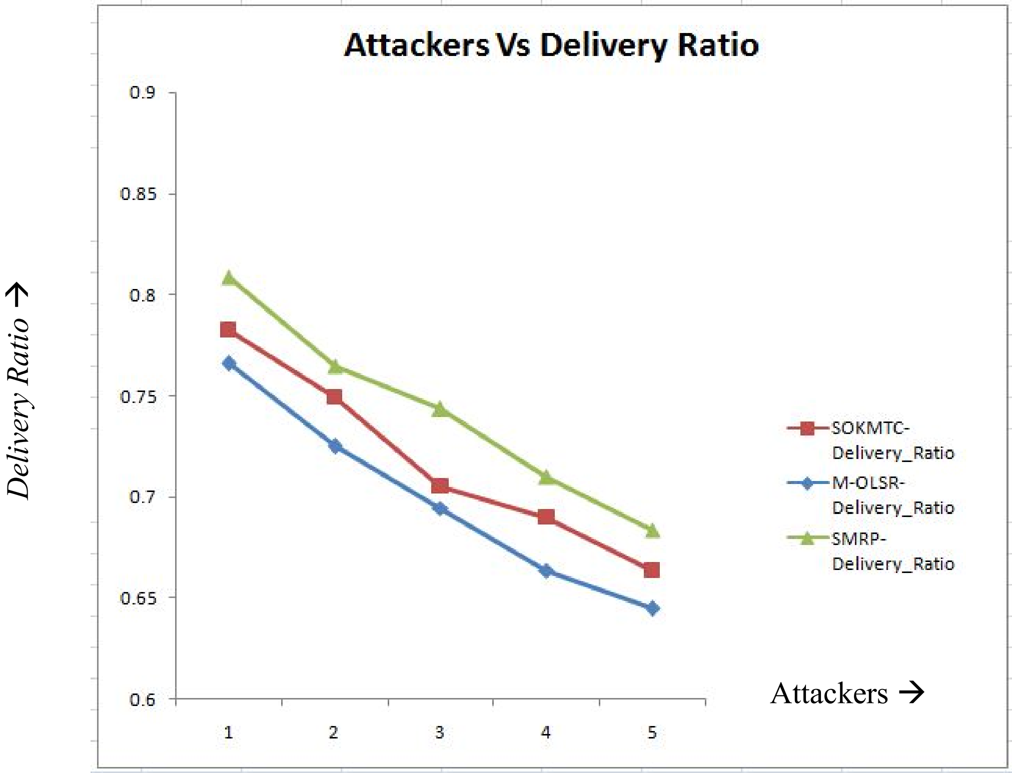

Fig 5.7 Packet Delivery Ratio

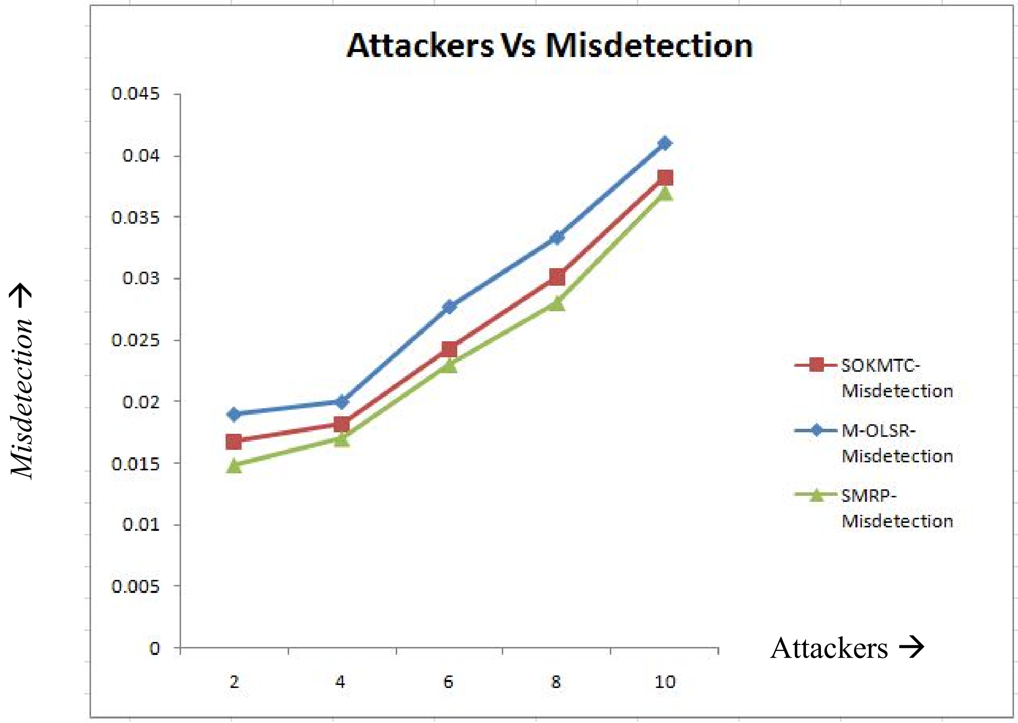

Fig 5.8 Misdetection Ratio

The ratio of the number of nodes whose behavior is not identified correctly to the actual number of such nodes in the network is shown in figure 5.8. Our proposed method is capable to detect more malicious nodes while comparing to the other methods. Due to the trusted certificate exchange and revocation mechanism, only the trusted nodes are participating in communication. The malicious nodes are isolated dynamically and such nodes are not going to participate in the rest of the network communication. Also the timestamp exchange mechanism addresses the replay attack.

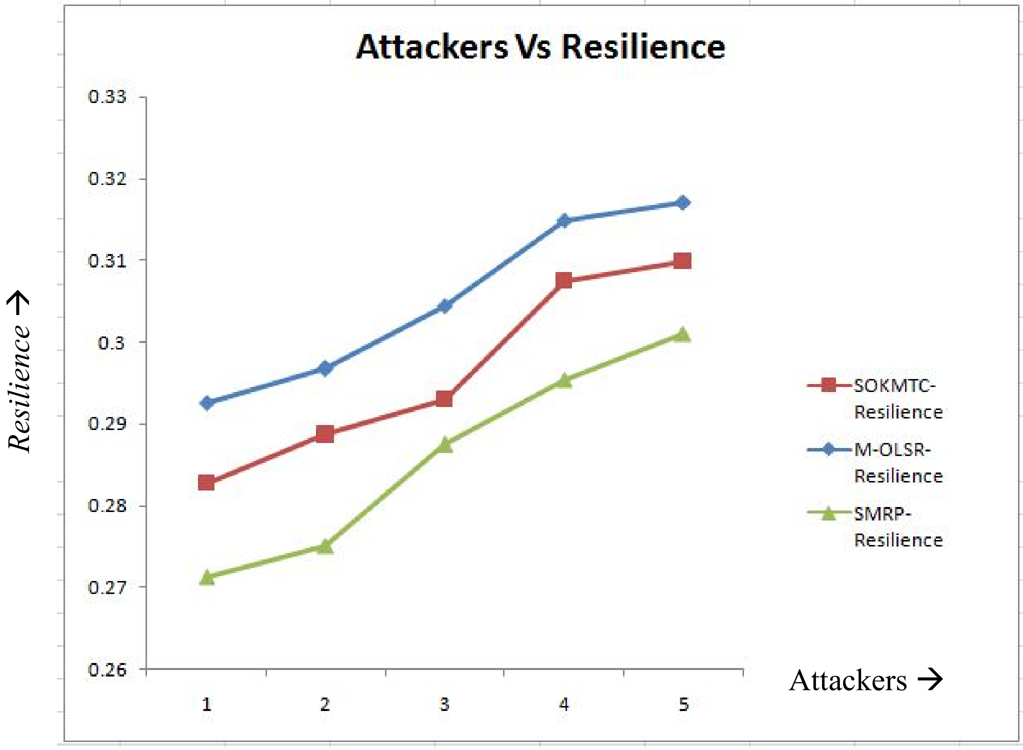

Fig 5.9 Resilience against Node Capture

The result of fraction of compromised communications is shown in figure 5.9. Because of the trusted mechanism, the number of compromised communications is less in SMRP. The certificate exchange and revocation based on the M-OLSR protocol improves the security. Hence the proposed SMRP is more resilient than the other mechanisms.

Varying Number of Nodes

The CBR data packets and control packets dropped due to the attackers, presented in figures 5.10. As the number of attacker increases, more data packets are dropped. But SMRP has less packet drops when compared to other schemes. When the number of nodes increases, depends upon the connection, more path breakage may happen. Here the dynamic route recalculation mechanism allows new path generation during the data transfer. This reduces the packet drop, there by improves the security.

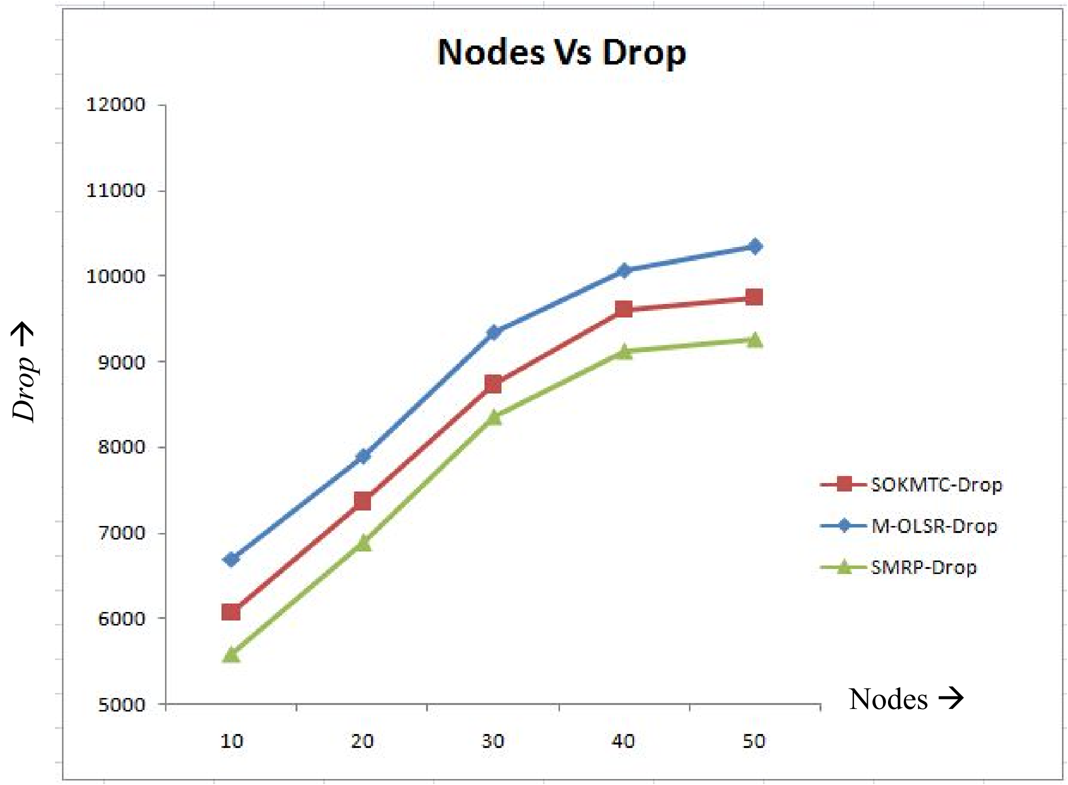

Fig 5.10 Packet Drop

Figure 5.11 depict the delay involved in the communication by each pair of source and destinations. The number of nodes is varied from 10 to 50, and corresponding delay for the three schemes are measured. The proposed method outperforms the other methods in case of delay.

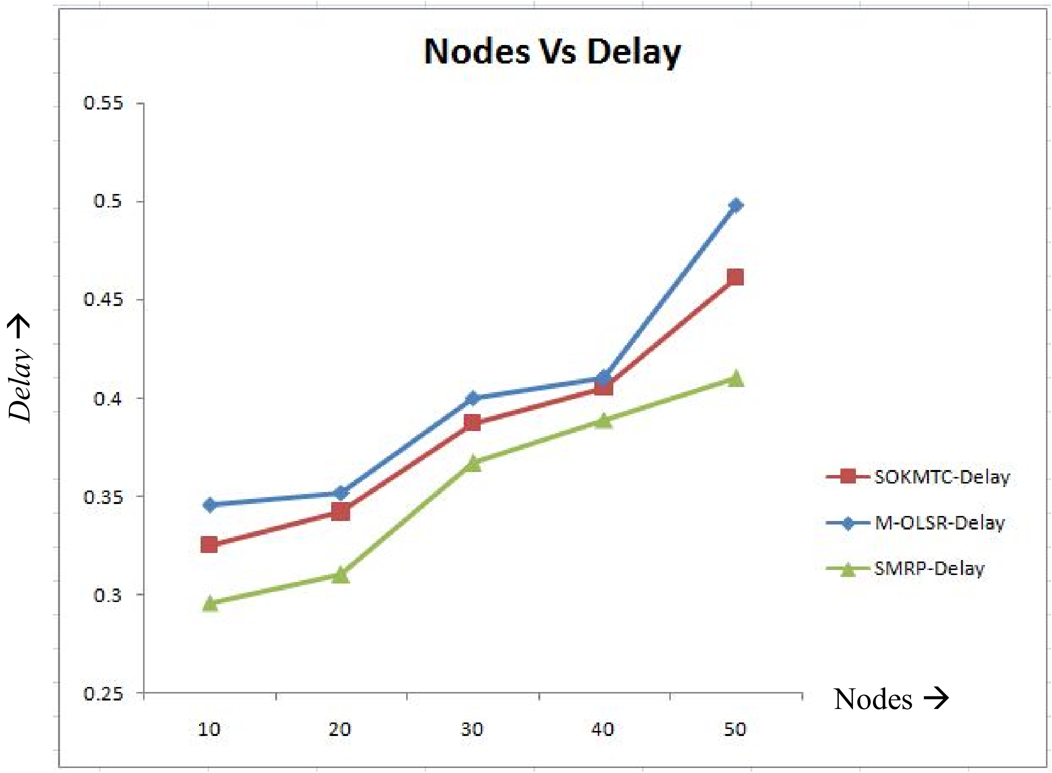

Fig 5.11 Average end-to-end Delay

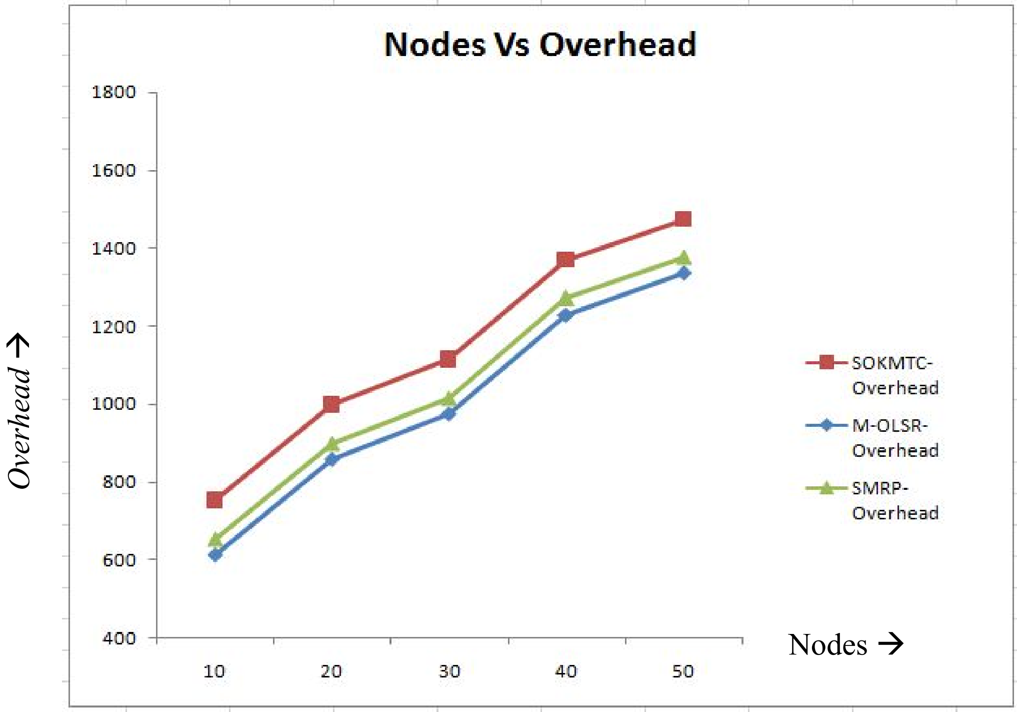

Fig 5.12 Routing packet overhead

Figure 5.12 shows the Routing packet overhead of the schemes, when the nodes are increased from 10 to 50. We can see that the overhead of our proposed SMRP is greater than the basic M-OLSR since the proposed method contains the trust management mechanism for certificate exchange and revocation, but it is lesser than SOKMTC.

5.6 Conclusion

In this chapter, we have proposed an M-OLSR based framework for certificate exchange and revocation in MANET. Here, we are also addressing the link failure problem in source routing. Trusted re-computation of routes is introduced when link failure occurs. In order to defend the replay attacks, the timestamp exchange mechanism is utilized in our proposed strategy. By simulation results, we have demonstrated that the framework gives more security than the existing method. The approach also achieves better resilience, good detection ratio and better performance in terms of delay and packet drop.

CHAPTER 6

A DISTRIBUTED HIERARCHICAL KEY MANAGEMENT SCHEME FOR MOBILE AD HOC NETWORKS

CONTENTS

6.1 Overview

In this chapter, a Distributed hierarchical key management scheme for mobile ad hoc networks (DHKM) has been proposed using a stable and power efficient cluster management system. It incorporates a trust management mechanism based on verifiable trust. A cluster based approach is used to reduce the storage overhead of every node. Every cluster head has the public key of its member nodes and act as a router while inter cluster communication happening. The communication of nodes between two unique clusters happens through their CH. A cluster based method is used to reduce the two limitations i.e., the over dependency on centralized server and increase in key-pair when node increases which SMOCK posses. The clustering system chooses a CH utilizing an adaptive weight clustering technique. This strategy additionally discusses about the impacts of node mobility between clusters. The need of every node to store all public keys is reduced along these lines minimizing the storage overhead on every node. The next sections explain different existing clustering techniques and the scalable key management system followed by our proposed system.

6.2 Associativity based cluster formation and cluster management in ad hoc networks [106]

The associativity based cluster formation scheme generates clusters as per the associativity of each nodes with the other mobile nodes. The protocol creates clusters in such a way that the clusters will be stable over a particular amount of time. The highest value of associativity means, the nodes are having high stability. These nodes are treated as cluster heads (CH).

A cluster head periodically sends beacon message and serves to advertise the presence of the cluster. Whenever a node enters into the network it waits for a random amount of time within this period, if it receives a beacon message from a CH, it joins a cluster by sending a joining request. Otherwise it invokes a cluster formation procedure. It contains three steps, neighbor identification, cluster controlling and cluster head selection. In neighbor identification, each node is going to send a HELLO packet and the neighbors are going to reply with an acknowledgement which determines the number of neighborhood nodes.

In the next phase, the associativity of nodes is going to be calculated. For calculating the same, a control message mechanism is used. Each node is going to send a control message to its neighbors. The nodes are going to reply together with the details of the route from a source to destination. If the reply is not received within a minimum threshold time, that reply message is going to be discarded. The cumulative associativity value is calculated. In the next phase, the cluster head is going to be selected depends upon the associativity value.

For maintaining the cluster, every node occasionally sends an alive message. Nodes that hear this checks their list of neighbors. If the details of the node are not there in the neighbor list, it is going to be added. The cluster heads are also going to be maintained a cluster head table, which contains the information about the cluster heads. The cluster head re-election process is done when the associative value of cluster head falls down to the threshold.

6.3 An Adaptive Weighted Cluster Based Routing (AWCBRP) Protocol for Mobile Adhoc Networks [58]

Here the clusters are formed based on the associativity and cluster head is going to be selected depends upon different stability parameters of the node. Every cluster chooses a CH to deal with the cluster and organize with different clusters. The CH determination is performed by assigning a weight value in view of power level, connectivity and stability. Figure 6.1 demonstrates the clustered adhoc network.

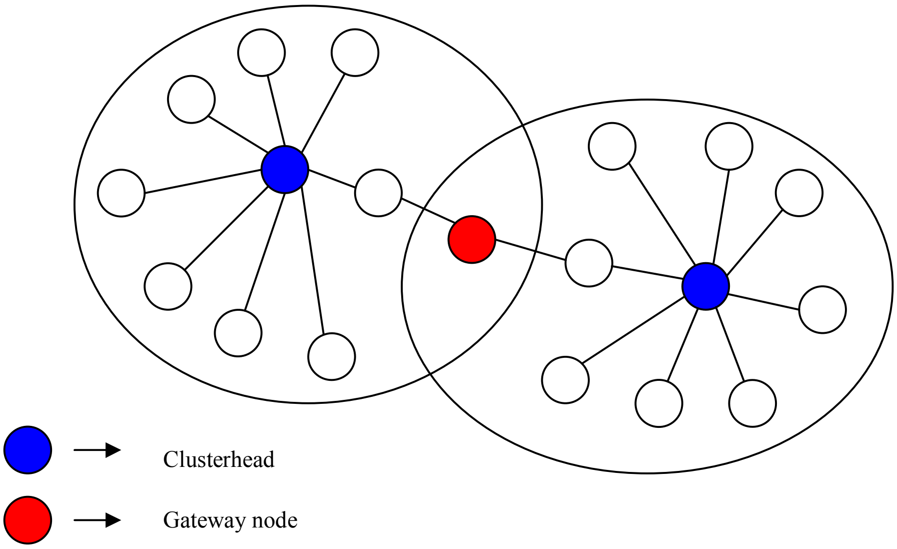

Fig 6.1 Clustered network

The node with the highest connectivity is been the cluster head. Once a node becomes a cluster head, then it is going to advertise its existence beacon messages. Nodes which are members of other clusters can register under the new cluster head. This will reduce the cluster size and also the network functioning will become smooth.

The stability of the cluster highly depends upon the node mobility. If the node mobility is high, it is very difficult to maintain the cluster. So a clustering algorithm should address the problems due to node mobility. The capacity of the cluster head should be analyzed periodically and if it founds that the stability reduces below a threshold, then the cluster size should be reduced and the election process for identifying a new cluster head should initiate.

In Cluster Based Routing Architecture, each cluster will be having a maximum capacity. Depends upon this value, the new nodes will be added to that cluster. Periodically the cluster heads capacity to serve the nodes is also monitored before a new node enters into the cluster. During the communication, if the destination is within the cluster, the cryptographic information is going to be collected from the cluster head and the route establishment is happening within the cluster. If the destination is outside the cluster, then the cluster head of the source node sends messages to other cluster heads to identify the cluster in which the destination node resides. Depending upon that information the communication is happening, where the cluster heads of the two clusters are act as routers for establishing path. The delay will be high for the inter-cluster communication.

6.4 SMOCK: A Scalable Method of Cryptographic Key Management for Mission-Critical Wireless Ad-Hoc Networks [57]

In SMOCK, a pair wise key mechanism is used to exchange communication securely. The key pool of such a group consists of a set of private–public key pairs. It is maintained by an offline-trusted server. Each key pair consists of two mathematically related keys. To support secure communication in the group, each member is loaded with all public keys of the group and assigned a distinct subset of private keys.

SMOCK uses the isometric key allocation algorithms to achieve the objectives. For a given network, the parameter a and b is calculated where ‘a’ is the no of public keys and ‘b’ is the no of private keys stored in each node. Then, the method that is used to allocate distinct private-key sets to users to achieve secure communication between each pair of users is discussed. By observing the tradeoff between memory usage and resilience against break-ins, an algorithm is presented to fully utilize memory space to achieve better resilience. The value of a and b affects the complexity of encryption and decryption. Therefore, it must be small.

When new nodes are deployed, the detailed protocols used for initialization, communication and bootstrapping are specified. The initialization phase is performed before deployment. Since communication and bootstrapping are online procedures, they have to be very efficient in terms of communication overhead.

Figure 6.2 illustrates the protocol that is used between sender and receiver to provide secure communication. The initialization phase assigns keys and identifications to each node. A node's identification (ID) indicates the subset of private keys that the node carries. If two nodes want to exchange a secure message, each node needs to know the ID of the other. From the ID, a node can infer which private keys the other node has, and it can encrypt the message with the corresponding public keys. Node IDs do not have to form a contiguous range. After key allocation, each node knows the private keys assigned to it, and all of the public keys.

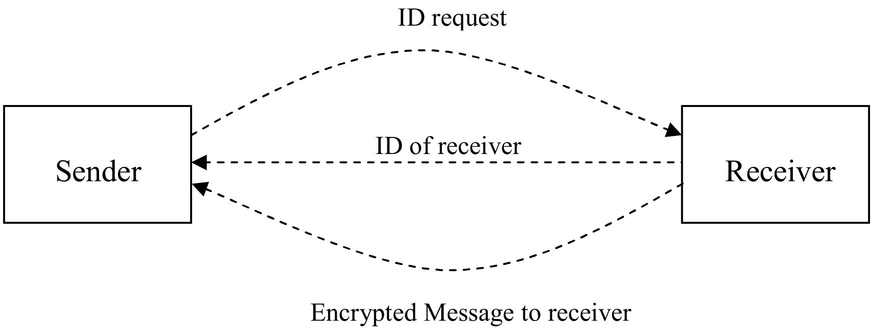

Fig 6.2 Secure communication protocol between Sender and Receiver

In SMOCK, since cryptographic keys are generated and maintained by the central offline trusted servers, the power to revoke keys and create the new ones is left in the hands of central servers. A key revocation message must be spread to all of those who might potentially hold it, and as rapidly as possible. Therefore, key revocation in

SMOCK relies on message broadcasting, where the revocation messages are signed and pushed by the central servers.

6.5 Proposed Work

We are proposing a cluster based approach based on trust to avoid the problems faced by the self certified method. The proposed method achieves high ratio of malicious node detection, reducing the memory space utilization of each node. There by increasing the performance and security of the communication.

6.5.1 Clustering Technique

The grouping of nodes is done by an associatively construct cluster formation scheme based with respect to the spatio-temporal stability [106] in which the separation between the nodes as indicated by the time period is considered for cluster formation (as discussed in section 6.2). After the cluster formation, the need of cluster head is definitive. An adaptive weight cluster (AWC) strategy [58] is achieved an effective power level, stable and higher connectivity based cluster head. In this system, the cluster head is framed considering the weighted entirety of the power level (*PL*) of the node, connectivity index (*CI*) between the nodes and stability record (*SR*) of the nodes (as discussed in section 6.3). The three factors are normalized by using three weight values to form the combined metric (CM). By using this combined metric, the cluster head is going to be identified.

$$CM = w1 * PL + w2 * CI + w3 * SR \tag{6.1}$$

($w1$, $w2$ and $w3$ - weighted factors)

The power level of the node is calculated by using a centralized algorithm and the connectivity variable is acquired as for link disjoint and joint values. Taking into account the entropy model, the stability values of the nodes are figured. This is calculated by taking the entropy of nodes as for node distances. Since nodes are mobile in nature, their development might saddle the cluster topology. To guarantee a legitimate communication between the nodes and in addition between the clusters, the techniques utilized as a part of AWC are utilized.

In this system, the connectivity element gives the information of joining or separating of any node, which at last yields node developments around the network. Alongside it, the power level component gives the list of nodes, which are at the border of any cluster. The border list made from such a border nodes decides the cluster link information as well as uncovers the information identified with movement of border nodes.

The consolidated weight value helps in giving a stable, imposing and a proficient cluster head. Aside from these variables, detection of node mobility and the regular topology changes are broke down. The cluster head chose can withstand a higher stability in the network and can expand the number of its individuals, in this way giving adaptable topology.

6.5.2 Key Management

A self-contained public key-management scheme is performed by a scalable means of cryptographic key management (SMOCK) [57], which acquires negligible communication overhead for authentication and offers maximum service availability. Here, a combinatorial design of public-private key pairs is created which provides each node with extra protection of more than one key pair to encrypt and decrypt messages. This format helps in earning higher stability in terms of nodes and storage space. The scheme also achieves controllable resilience against node compromise by defining required benchmark resilience and higher availability. However, this possesses two major drawbacks;

- Central trusted authority for revoking/refreshing keys and make new keys for the new nodes.
- Increase in nodes eventually builds the public-private key pairs (however relatively in low extent than traditional methodology)

A solution is introduced to enhance the above two drawbacks by increasing the fault-tolerance and reducing the overhead for the central trusted authority as well as individual nodes.

Selection of Private – Public Key Pairs

The selections of key pairs are dependable on two factors; memory and protection for key exposure upon attacks or node compromise. The private keys and the public keys are stored in the memory of every individual node. Increase in nodes increases the memory slot allocated to memory space (as per the traditional public-key-management scenario of $n+1$ key). Thus to enhance the memory in each node, reduction of keys with respect to pair-wise key distribution schemes are necessary.

When using pair-wise key distribution the network is opened to vulnerabilities by attackers. Reduction of keys or pairing of keys makes node to compromise their keys as well as keys of their neighbors to attacker. Therefore, need of a capable pair wise adjustments should be executed.

With respect to the above two constraints, the private and public keys for a node are determined using the method proposed by Wenbo He et.al. [57]. The number of nodes is considered as "N". Let P$_x$ be the public key and P$_y$ be the private key. As per SMOCK, each node knows all P$_x$ values but only posses a unique combination of P$_y$ values.

The private-key combination pattern is afflicted to the node ID. This can be explained with a scenario where in node A wants to transmit a message to a node B. Here the node A initially collects B's id to deduce public key values of B. Then, A will encrypt the message with the public-key set that correlates with the private keys owned by B. Thus only B can open the message as the public-private key combinations are inherited only by it.

The value of P$_x$ and P$_y$ is determined using the Binomial coefficient also known as a combination or combinatorial number. It is the number of combinations of r items that can be selected from a set of n items $C(n,r)$. Consider there are 21 nodes (assume the exact number of key combination to be created for all the 21 nodes without excess keys) in a network. With Binomial coefficient the result of $C(7,2)$, $C(7,5)$ and $C(21,1)$ all produces 21. Thus, there can be three types of public-private key combinations,

- A set of 7 public keys and 2 private keys
- A set of 7 public keys and 5 private keys
- A set of 21 public keys and 1 private key.

The third option is basically is the traditional public-key-management scenario of $n+1$ key. This is the least value of combination available. The first and second values are having a total pair of 9 and 12 keys in there node memory, which is comparatively very small value when compared with traditional public-key-management scenario of 22 keys. The first scenario of appropriate memory allocation can be obtained efficiently by using first and the second key combination but when taking the scenario of node compromise , the second value of 12 keys are more vulnerable as each node contains 5 private keys. Therefore one private key value shares its key with 15 nodes (calculated by $\frac{Py * N}{Px}$), which is highly vulnerable when subjected to any attack. The first key combination of 9 keys not only provides an efficient memory management but also is comparatively less vulnerable as it shares its single key only with 6 nodes. Similarly, the nodes are provided with public-private key combination. When number of nodes increases, the combinations are predicted based on the minimum value of Py, as increase in Py increases the sharing of private keys among the nodes.

6.5.3 Key Distribution

In this section, the effective key distribution of "key pairs" along every cluster is resolved. After the beginning cluster formation, every cluster head send the information of its cluster individuals to the central trusted authority. The central trusted authority generates the public keys and private key combination as appeared in table 6.1 and 6.2. In the wake of acquiring the information, the central trusted authority distributes the produced key mix in an extraordinary way.

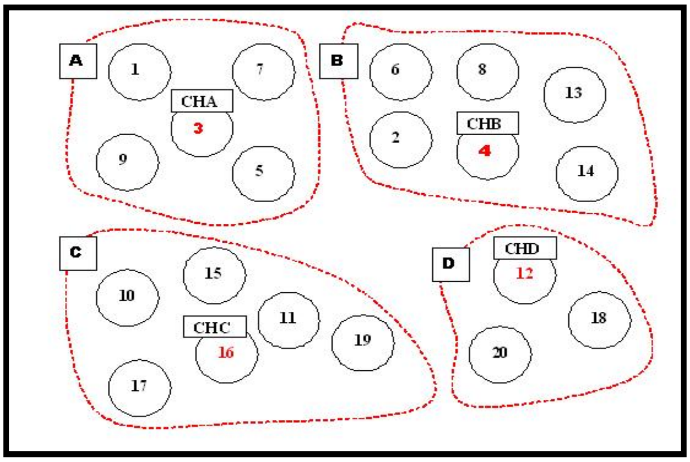

Fig 6.3 Clustering Phase

Consider a case of 20 nodes in a network as appeared in the figure 6.3 grouped along with 4 clusters with each having a cluster head. The central trusted authority (CTA) creates the blend of keys and directs the required keys to the cluster head as indicated by the cluster head information. One designated cluster head is going to be as central trusted authority. A random shift mechanism is incorporated for changing the CTA. Consider the public key produced in the central trusted authority as appeared in table 6.2.

Node	Private Key Set held by the node
1	P_y 1, P_y 2
2	P_y 1, P_y 3
3	P_y 1, P_y 4
4	P_y 2, P_y 3
5	P_y 2, P_y 4
6	P_y 2, P_y 5
7	P_y 3, P_y 4
8	P_y 3, P_y 5
9	P_y 3, P_y 6
10	P_y 4, P_y 5
11	P_y 4, P_y 6
12	P_y 4, P_y 7
13	P_y 5, P_y 6
14	P_y 5, P_y 7
15	P_y 5, P_y 1
16	P_y 6, P_y 7
17	P_y 6, P_y 1
18	P_y 6, P_y 2
19	P_y 7, P_y 1
20	P_y 7, P_y 2

Table 6.1 Private key allocation by the central trusted authority

Public Key	Nodes
P_x 1	1, 2, 3, 15, 17, 19
P_x 2	1, 4, 5, 6, 18, 20
P_x 3	2, 4, 7, 8, 9
P_x 4	3, 5, 7, 10, 11, 12
P_x 5	6, 8, 10, 13, 14, 15
P_x 6	9, 11, 13, 16, 17, 18
P_x 7	12, 14, 16, 19, 20

Table 6.2 Public Key Generation

The distribution of the key combination is done via cluster head and it holds the copy of all the public keys from its member nodes. Therefore, each node contains the public keys of its cluster member only. This is shown in the table 6.3 below.

Clusters	Cluster Members	CH	Public Keys held in cluster
Cluster 01	1, 3, 5, 7, 9	3	P_x 1, P_x 2, P_x 3, P_x 4, P_x 6
Cluster 02	2, 4, 6, 8, 13, 14	4	P_x 1, P_x 2, P_x 3, P_x 5, P_x 6, P_x 7
Cluster 03	10, 11, 15, 16, 17, 19	16	P_x 1, P_x 4, P_x 5, P_x 6, P_x 7
Cluster 04	12, 18, 20	12	P_x 2, P_x 4, P_x 6, P_x 7

Table 6.3 Public Key Distribution in Each Cluster Head

Here, two types of secure communication are carried out;

- Communication between the nodes inside a cluster.(Intra Cluster)
- Communication of nodes between two different clusters (Inter Cluster)

If two nodes need to exchange a protected message, it should know the ID of one another. Accordingly the source requests the ID of the destination. In the wake of getting the destination node's ID, source will derive the private key information of the destination node. Taking into account the deduced information, source will check the accessibility of relating public keys in its own particular cluster head. If that it is not accessible in CH, then CH will send a public key request to every single other CHs.

As appeared in table 6.3, the cluster part just posses the public keys which are identified with its cluster part nodes. In a secured communication of nodes inside of a cluster, each node posses the public keys of their part nodes. In the event that a node needs to communicate with another node inside of a cluster, he procures the public key and sends the encrypted message through it.

Otherwise, nodes of two distinct clusters communicate through the cluster heads. In such a circumstance, the cluster head acts as a router. If that node 1 needs to communicate with node 2, which is set in another cluster (Figure 6.4), it sends the ID request to its cluster head (CHA) which posses every single public key of its individuals. CHA advances this request to all other cluster heads. The cluster head which contains node 2 (CHB) thus illuminates the node 2 and gathers the ID from it. CHB sends back the ID as a reply to CHA. Subsequent to inducing the private key information from the got ID, node 1 gets the comparing public keys of node 2 from other cluster heads by means of CHA. Subsequently, the need of every node having all public key is lessened and simply need to store the public keys of its cluster individuals.

6.5.4 Trust Management Mechanism

The trust management mechanism is utilized with a specific end goal to validate the nodes in the network. The trust value is computed using the node’s verifiable trust. This model considers just the direct trust. Direct trust is the direct information of neighbors and simple to get. Watching a node's behaviors is a successful mechanism to figure out if this node can be trusted. The trust is going to be calculated by using the node's verifiable trust mechanism explained in 4.2.1.

6.5.5 ID Revocation

Each node is going to calculate the verifiable trust of its neighbors. This is going to be stored in the trust table. If the trust value is below a particular threshold, then the node is going to inform the corresponding cluster head. One cluster head is going to designated as the central trusted authority, who is the in charge of CRL maintenance. The cluster head is going to report to the corresponding coordinator node regarding the malicious behavior.

The certificate revocation list is going to be updated, when some nodes show malicious behavior. During the path selection, the node involved in those paths, whose trust value below the threshold is not allowed to participate in that. Each cluster head is going to check the trust of other cluster head depends upon their past performance. If the trust value is below a threshold, then that cluster head has to be isolated and new cluster head selection should be initiated.

6.5.6 Effect of Node Mobility

The cluster head updates due to mobility plays an important role. If a node moves from one cluster to another cluster, both the cluster heads (cluster in which the node leaves and the cluster in which the node joins) need to know details of the moving node. Consider node 19 of cluster C moves to cluster D (Figure 6.3), then both the cluster heads (CHC and CHD) needs to know about the movement.

Here, when node 19 moves from cluster C to cluster D, CHD gets node 19's previous cluster details. With these details, CHD gains the Public keys of node 19 from its previous cluster head (CHC). Before submitting the public key details, the CHC checks for similar public key information (checks for nodes related to Px7 and Px1) , if no related nodes are found, it sends the public key and delete the details stored in it and if there exist any related nodes, it will only send the public key without deleting the key details. Similarly, when CHD, receives the public key, it will check for any existence of similar public key (checks for Px7 and Px1 in Cluster D).

6.5.7 Overall Algorithm

The entire process of the proposed technique is described using the following algorithm.

Step 01 – Cluster Formation

The grouping of nodes into cluster follows an associatively based cluster formation scheme based on the spatio-temporal stability. The distance between the nodes according to the time period is considered for cluster formation.

Step 02 – Cluster Head Selection

An adaptive weight cluster (AWC) technique is used to attain an efficient power level, stable and higher connectivity based cluster head.

Step 03 – Selection of Public/Private Key (Px/Py) Pairs

In order to reduce the total number of keys, it uses the combination of multiple private keys instead of a single private key. The value of *Px* and *Py* is determined using the Binomial coefficient also known as a combination or combinatorial number. It is the number of combinations of r items that can be selected from a set of *n* items *C*(*n*, *r*).

Step 04 – Key Generation

After the beginning cluster formation, every cluster head sends the information of its cluster individuals to the central trusted authority. The central trusted authority creates the public and private key blends. The Px and Py are created regarding the information acquired from its part nodes by the central trusted authority. In the wake of acquiring the information, the central trusted authority distributes the created key blend in an uncommon way.

Step 05 – Key Distribution

The distribution of the key combination is done via cluster head and it holds the copy of all the public keys from its member nodes. So each node contains the public keys of its cluster member only.

Step 06 – ID Allocation

A node’s ID can be generated by the node itself, after getting its private key pairs. A node’s identification is a good indicator to show what subset of private keys the node carries. From the ID, a node can infer which private keys the other node has.

Step 07 – Node Mobility Considerations

If a node moves from one cluster to another cluster, both the cluster heads need to know details of the moving node. The cluster head must contain the public keys of its cluster members. Thus the new CH (the cluster in which the node joins) will request the public keys of the newly arrived node from the old CH (cluster in which the node leaves).

Step 08 – ID Based Communication

If two nodes want to exchange a secure message, each needs to know the ID of each other. Thus the source requests the ID of the destination. After getting the destination node's ID, source will infer the private key information of the destination node. Based on the inferred information, source will check the availability of corresponding public keys in its own cluster head. If it is not available in CH, then CH will send a public key request to all other CHs.

Step 09 – Message Encryption/Decryption

Based on the obtained private key information, source can encrypt the message with the corresponding public keys. The message can be decrypted using the private key pairs held by the destination.

Step 10 – Trust Calculation

The nodes in the network are validated using the trust management technique. The trust values of the cluster members are calculated by the corresponding CH. CH also calculates the trust value of other CH nodes. Trust management mechanism is explained in section 4.2.1

Step 11 – Identifying the malicious nodes

If the trust value of a node is smaller than the black list trust threshold, then that node will be regarded as malicious nodes.

Step 12 – Defending against malicious nodes

An ID revocation list will be maintained based on the trust value. Whenever CH finds any misbehaving nodes, it will request an 'ID revocation list update' to the central trusted authority (CTA). A random shift mechanism among the cluster heads is used to select CTA. The updated list will be broadcasted to all CHs. CH will verify the communicating entities with the list during every communication.

6.6 Simulation Results

6.6.1 Simulation Model and Parameters

Network Simulator (NS2) is utilized to simulate the proposed algorithm. In this simulation, the channel limit of mobile hosts is set to the same value: 2 Mbps. The distributed coordination function (DCF) of IEEE 802.11 for wireless LANs is utilized as the MAC layer protocol. It has the functionality to inform the network layer about link breakage. In this simulation, mobile nodes move in a 1000 meter x 1000 meter network region for 50 seconds simulation time. The number of nodes has been changed as 10, 20, 30,..50. Expect every node moves autonomously with the same normal speed. All nodes have the same transmission range of 250 meters. In this simulation, the minimal speed is 5 m/s and maximal speed is 10 m/s. The simulated traffic is Constant Bit Rate (CBR). The number of attackers is changed as 1 to 10. The simulation settings and parameters are summarized in table 6.4

No. of Nodes	10,20,30,…50
Area Size	1000 X 1000
Mac	802.11
Radio Range	250m
Simulation Time	50 sec
Traffic Source	CBR
Packet Size	512
Speed	5m/s t 10m/s
Misbehaving Nodes	1 to 10

Table 6.4 Simulation Settings for DHKM

The following are the assumptions used for the proposed framework,

- A malicious node can compromise the key, create packet drop attack, routing overflow attack etc.
- A trusted node will be having a trust value greater than 0.5.
- A threshold trust is fixed as 0.5.
- Malicious node will be having a trust value less than 0.5
- Trust value 1 refers to full trust and 0 refers to complete distrust.
- The designated cluster head will be the coordinator node.
- The coordinator node is the in charge of CRL update.
- Protocol used is AWCBRP.

6.6.2 Performance Metrics

We compare the proposed Distributed Hierarchical Key Management Scheme for Mobile Ad hoc Networks (DHKM) with scalable method of cryptographic key management (SMOCK) and Cluster based routing protocol (CBRP).

We evaluate mainly the performance according to the following metrics [91]:

- **Average end-to-end Delay:** the normal time taken by the data packets from sources to destinations, including buffer delays during a route discovery, lining delays at interface lines, retransmission delays at MAC layer and propagation time.
- **Packet Delivery Ratio:** or packet throughput, the fraction of the data packets conveyed to destination nodes to those sent by source nodes.
- **Packet Drop** It is the number of packets dropped during the transmission.

- **Misdetection Ratio:** the proportion of the number of nodes whose behavior (malicious or considerate) is not recognized effectively to the real number of such nodes in the network.
- **Routing packet overhead**: the number of control packets (including route request/reply/update) for establishing connection over a period of time.
- **Resilience against Node Capture:** the fraction of communications compromised to the total number of communications by a capture of x-nodes.

6.6.3 Results

Varying Number of Attackers

The number of attackers is increased from 1 to 10 and the performances of the techniques are measured in terms of Delivery Ratio, Misdetection and Resilience.

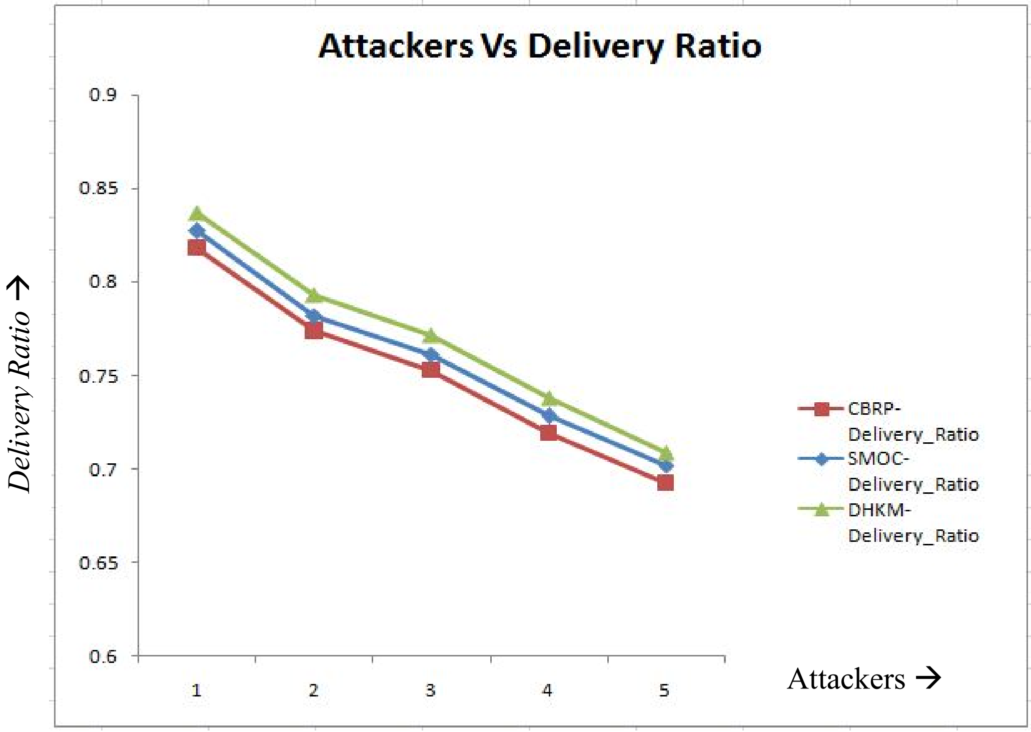

Fig 6.4 Packet Delivery Ratio

Figure 6.4 shows the average Packet delivery Ratio of the schemes, when the attackers are increased from 1 to 5. We can see that the delivery ratio decreased linearly as the number of attackers increase. But, the delivery ratio of our proposed DHKM is greater than the other existing schemes. The packet delivery ratio is high because of the trusted ID exchange and revocation. The centralized trusted authority is going to update the ID revocation table and depends upon this table, the route selection happens.

The ratio of the number of nodes whose behavior is not identified correctly to the actual number of such nodes in the network is shown in figure 6.5. Our proposed method is capable to detect more malicious nodes while comparing to the existing methods. A trusted security mechanism has been included and the malicious nodes are isolated depends upon the past performance of the node. That's why the misdetection ratio is very less compared to the existing schemes.

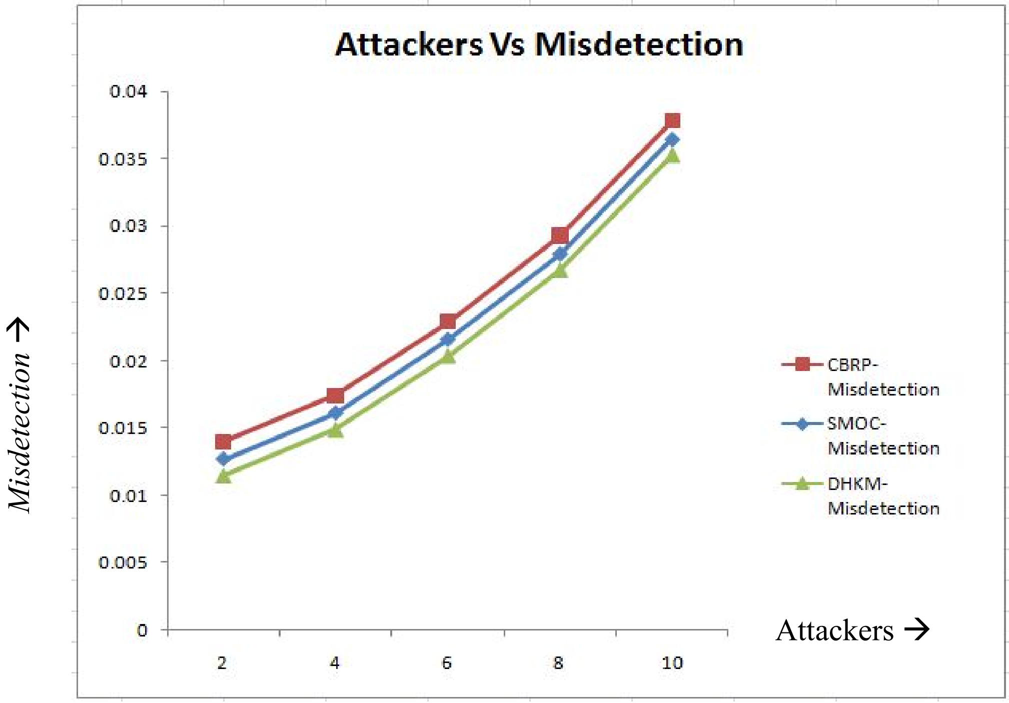

Fig 6.5 Misdetection Ratio

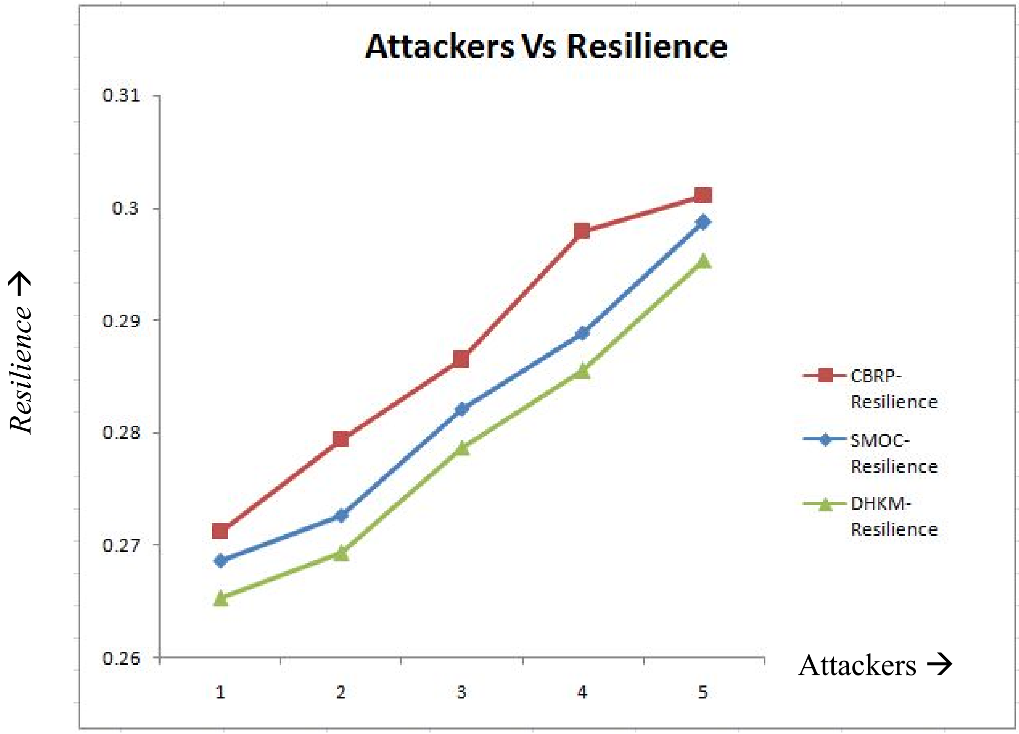

Fig 6.6 Resilience against Node Capture

The result of fraction of compromised communications is shown in figure 6.6. Because of the trusted mechanism, the number of compromised communications is less in DHKM. Hence the proposed DHKM is more resilient than the existing mechanisms.

Varying Number of Nodes

The CBR data packets and control packets dropped due to the attackers, presented in figures 6.7. As the number of attacker increases, more data packets are dropped. But DHKM has less packet drops when compared to other schemes.

Figure 6.8 depict the delay involved in the communication by each pair of source and destinations. The number of nodes is varied from 10 to 50, and corresponding delay for the three schemes are measured. The proposed method outperforms the existing methods in case of delay.

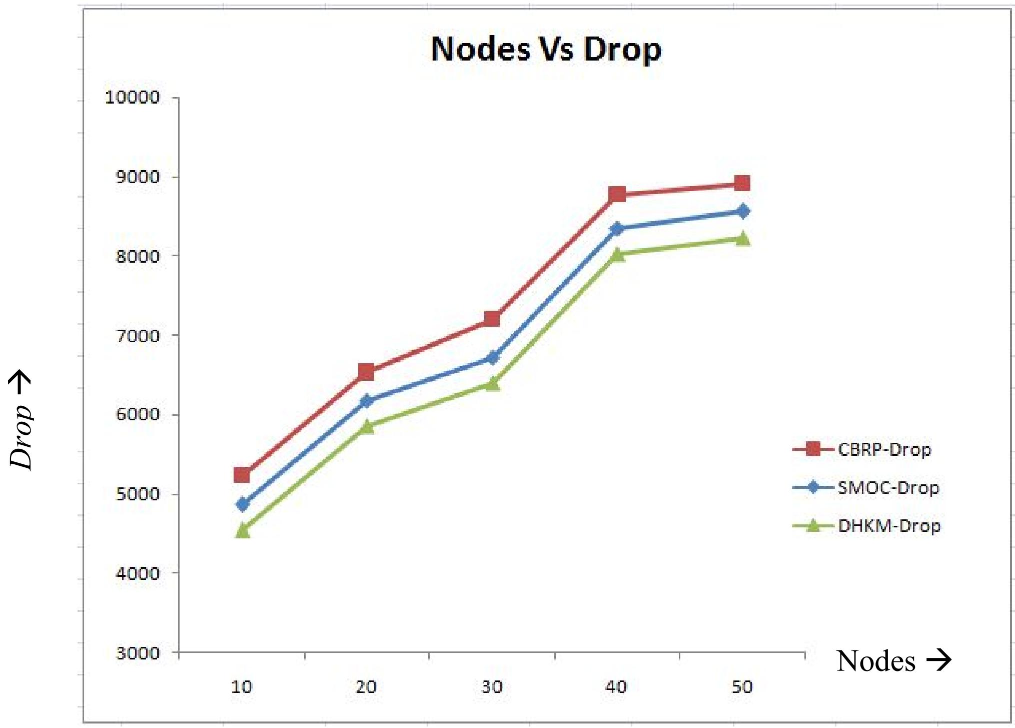

Fig 6.7 Packet Drop

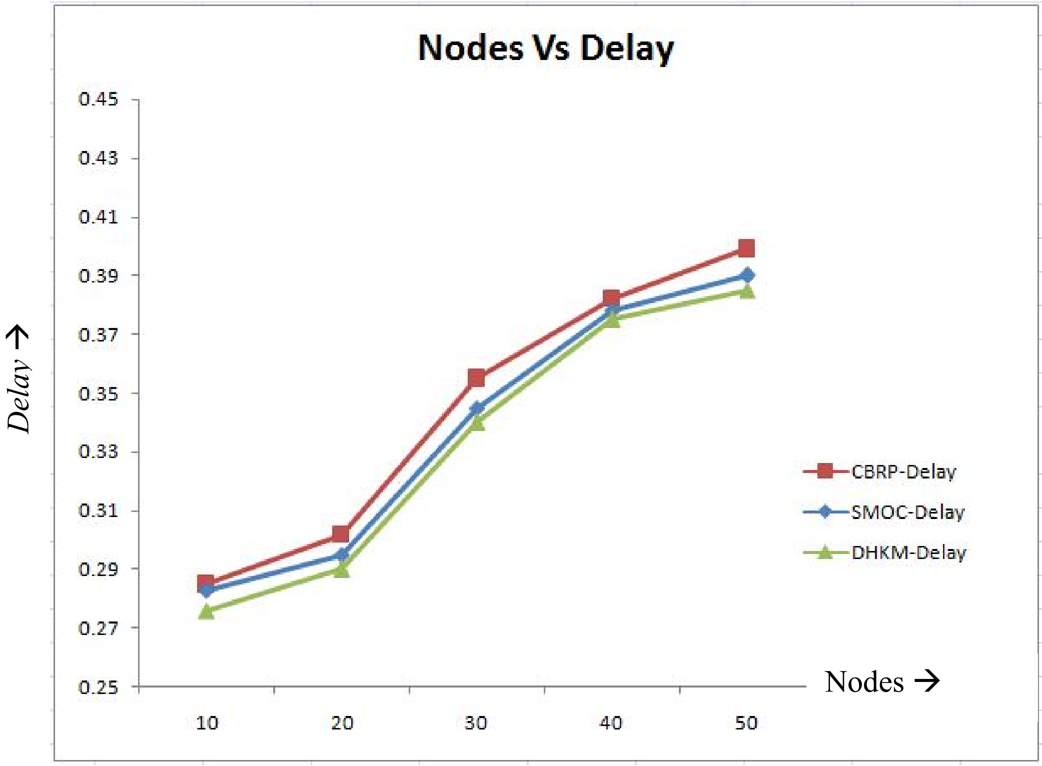

Fig 6.8 Average end-to-end Delay

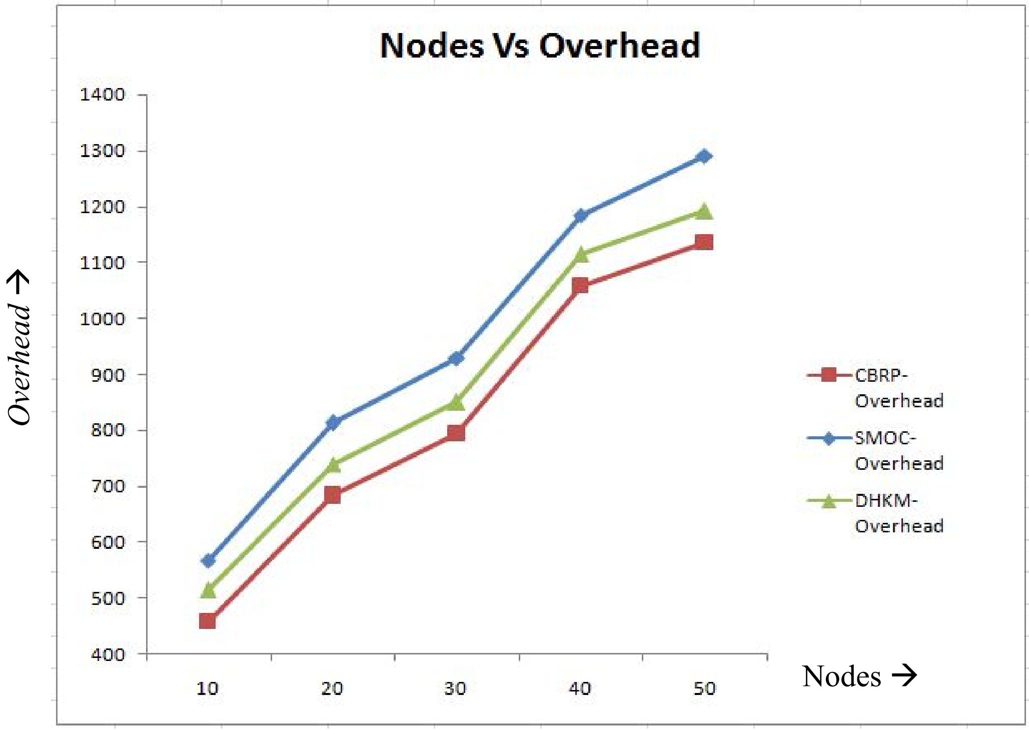

Fig 6.9 Routing packet overhead

Figure 6.9 shows the routing packet overhead of the schemes, when the nodes are increased from 10 to 50. We can see that the overhead of our proposed DHKM is greater than the basic CBRP since the proposed method contains the trust management mechanism for certificate exchange and revocation, but it is lesser than SMOCK.

6.7 Conclusion

In this chapter, a scalable method of cryptographic key management (SMOCK) is enhanced. A clustering based technique is presented to reduce the two drawbacks; to over dependent on centralized server and increase in key-pair when node increases (proportionally less compared to traditional approach) which SMOCK posses. The clustering technique used here to select a CH, is an adaptive weight clustering method. The CH is stored with public keys of all its member nodes. The communication of nodes between two different clusters happens through their CH. Our method also discusses about the effects of node mobility between clusters. A Trust management system based on the verifiable trust is incorporated and the malicious nodes are isolated by using black list threshold. By Simulation results, it is shown that this proposed scheme achieves better delivery ratio and resilience with reduced delay and overhead. The method also achieves better detection ratio as we have incorporated trust management mechanism.

CHAPTER 7

PREDICTIVE CLUSTER BASED DISTRIBUTED HIERARCHICAL KEY MANAGEMENT SCHEME FOR MANET

CONTENTS

7.1 Overview

In this chapter, an improved progressive key management scheme utilizing a stable and power proficient cluster management procedure has been proposed. The mobility prediction strategy is combined in the proposed hierarchical key management scheme. The method predicts the node movement and sends information if there should arise an occurrence of cluster movement. The consolidated metric for prediction is evaluated taking into account route expiration time and node velocity. Every cluster head holds the public key of its part nodes just and go about as a router when managing nodes of other cluster individuals. Utilizing this procedure, the overhead on centralized server is decreased. Also, the need of every node putting away all public keys is lessened consequently minimizing the stockpiling overhead on every node. By Simulation results, it is demonstrated this scheme accomplishes better delivery ratio and resilience with lessened delay and overhead.

7.2 Prediction Technique

Here we are going to predict the mobility based on the route expiration time and link stability of individual links. A probabilistic approach is used to find out the link availability. Link stability is going to be calculated based on the transmission range and distance travelled. The combined metric of these two will be the mobility prediction metric.

7.2.1 Based on Route Expiration Time

The $\mathbf{T}_{RE}$ is the minimum time selected from a set of link expiration times (T_{LE})s designed for the sake of sufficient path. The time period between nodes is given as T_{LE}. Hence, the minimum value of T_{LE} attained in each path and the maximum number of T_{RE} is selected which is representing the reliable routing path [107].

$$\mathbf{T}_{RE} = \text{Min}\ (T_{LEs}) \tag{7.1}$$

Thus for the feasible path T_{RE} is the maximum value among T_{LE} s. This link availability is going to be predicted based on the probability distribution theory (T_{pr}).

Global positioning system is the method used to achieve the principle of T_{LE} which is for estimating future disconnection time with the help of two neighbors in motion. It determines the movement parameters of two neighboring nodes. The assumptions made are as follows:

- Signal strength of free space propagation model is exclusively depended on the distance to the transmitter.
- GPS clock helps all nodes to synchronize themselves with their clock values.

By having knowledge of the motion parameters of two nodes, the time period for the nodes can be calculated. These parameters obtained from GPS include speed, direction, and radio range.

On the continuously available time for an active link between two nodes the link expiration time at time T_0 with a given prediction T_{pr}, the link availability is defined as,

$$L\,(T_{pr}) = P\,\{T_0 \text{ to } T_0 + T_{pr} \mid \text{Available at } T_0\} \qquad (7.2)$$

Here if we denote link availability of a node by LA, then it is seen that LA follows an exponential distribution $LA(x) = \lambda e^{-\lambda x}$, for $x \geq 0$.

Hence from équation (7.2) it follows that $L\,(T_{pr}) = P\{LA > T_0 + T_{pt} / LA > T_0\}$

$= P\{LA > T_{pt}\}$, by Memory less property of exponentiel distribution

$= \int_{T_{pr}}^{\infty} \lambda e^{-\lambda x} dx$, as Link avaialablity follows an exponentiel distribution

$$= \lambda \left[\frac{e^{-\lambda x}}{-\lambda} \right]_{T_{pr}}^{\infty} = -1\left[e^{-\lambda x}\right]_{T_{pr}}^{\infty} = -1\left[0 - e^{-\lambda T_{pr}}\right] = e^{-\lambda T_{pr}} \qquad (7.3)$$

Also we have $L\,(T_{pr}) = P\{LA > T_0 + T_{pr} / LA > T_0 + T_{pr}\}$

$= P\{LA > T_{pt}\}$, by Memory less property of exponentiel distribution

$= 1 - P\{LA \leq T_{pt}\}$, by complément probablity (7.4)

$$= 1 - \int_{0}^{\mathrm{T_{pr}}} \lambda \mathrm{e}^{-\lambda x} dx$$

$$= 1 - \lambda \left[\frac{e^{-\lambda x}}{-\lambda} \right]_{0}^{\mathrm{T_{pr}}} = 1 + 1\left[e^{-\lambda x} \right]_{0}^{\mathrm{T_{pr}}} = 1 + 1\left[e^{-\lambda \mathrm{T_{pr}}} - 1 \right] = e^{-\lambda \mathrm{T_{pr}}} \qquad (7.5)$$

We usually denote $\mathrm{P}\{\mathrm{LA} \le \mathrm{T_{pt}}\}$ as the distribution function $\mathrm{F(T_{pr})}$

Thus From (7.4), we obtain $\mathrm{L}\ (\mathrm{T}_{pr}) = 1 - \mathrm{F(T_{pr})}$ (7.6)

This indicates the probability of link availability existing from T_0 to $T_0 + T_{pr}$.

The calculation of $\mathrm{L}\ (\mathrm{T}_{pr})$ can be divided into two parts:

$L_1\ (T_{pr})$: the link availability when the velocities of the two nodes keep unchanged between T_0 and $T_0 + T_{pr}$,

$L_2\ (T_{pr})$: the one for the other cases

$$\text{(i.e.)}\ \ L\ (T_{pr}) = L_1\ (T_{pr}) + L_2\ (T_{pr}) \qquad (7.7)$$

Calculation of $L_1\ (T_{pr})$, which is equal to the probability that the epochs from t_0 onwards for the two nodes are longer than T_{pr} because T_{pr} is an accurate prediction if the movements of the two nodes keep unchanged. Since node movements are independent of each other and exponential distribution is memory less, $L_1(T_{pr})$ is given by

$L_1(T_{pr})$ = link availability of the first node × link availability of the second node

= (link availability of the first node)^2 (using the fact that the probability that the epochs from t_0 onwards for the two nodes are longer than T_{pr} and because T_{pr} is an accurate prediction if the movements of the two nodes keep unchanged)

$= [L\ (T_{pr})]^2$

$= \left(e^{-\lambda T_{pr}}\right)^2$ (by using the fact that nodes' movements follows an exponential distribution and exponential distribution is 'memory less-see equation (7.3 or 7.5))

$= e^{-2\lambda Tpr}$

We can also see by equation (7.6) that $L\ (T_{pr}) = [1 - F(T_{pr})]^2$

Thus $L_1\ (T_{pr}) = [1 - E\ (T_{pr})]^2 = e^{-2\lambda Tpr}$ (7.8)

Where, E and F are probability functions. However, it is complicated to give an accurate calculation for $L_2\ (T_{pr})$ because of the difficulties in learning changes in link status caused by changes in a node's movement. Here we are only considering the link availability when the velocities of two nodes keep unchanged.

7.2.2 Based on Link Stability

Link stability in terms of link expiration time is defined as maximum time connectivity between any two neighbor nodes. For calculating the link expiration time, it is assumed that motion parameters of any two neighbors are known [108].

Let n_1 and n_2 be two nodes within the transmission range *r* and (x_1, y_1) and (x_2, y_2)be the coordinate for node n_1 and n_2 with velocity v_1 and v_2 and direction θ_1 and θ_2 respectively.

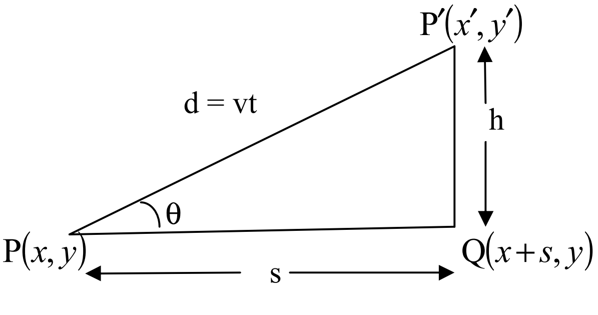

Fig 7.1 Distance Calculation

After a time interval t the new coordinate will be (x_1', y_1') for n_1 and (x_2', y_2') for n_2. For a time t, let d_1 and d_2 be the distance traveled by node n_1 and n_2. d_1 and d_2 are calculated using the following formula: distance = velocity * time

$$d_1 = v_1 t \tag{7.9}$$

$$d_2 = v_2 t \tag{7.10}$$

Referring the figure above, new coordinates (with respect to old coordinates) can be calculated as

$$x_1' = x_1 + s_1 = x_1 + d_1 \cos\theta_1 = x_1 + v_1 t \cos\theta_1 \tag{7.11}$$

$$y_1' = y_1 + h_1 = y_1 + d_1 \sin\theta_1 = y_2 + v_1 t \sin\theta_1 \tag{7.12}$$

$$x_2' = x_2 + s_2 = x_2 + d_2 \cos\theta_2 = x_2 + v_2 t \cos\theta_2 \tag{7.13}$$

$$y_2' = y_2 + h_2 = y_2 + d_2 \sin\theta_2 = y_2 + v_2 t \sin\theta_2 \tag{7.14}$$

Distance *D* between two nodes at time *t* can be obtained from:

$$D = \sqrt{(x_1' - x_2')^2 + (y_1' - y_2')^2}$$

$$= \sqrt{\{[x_1 + v_1 t \cos\theta_1] - [x_2 + v_2 t \cos\theta_2]\}^2 + \{[y_1 + v_1 t \sin\theta_1] - [y_2 + v_2 t \sin\theta_2)]\}^2}$$

$$= \sqrt{\{[x_1 - x_2] + t[v_1 \cos\theta_1 - v_2 \cos\theta_2]\}^2 + \{[y_1 - y_2] + t[v_1 \sin\theta_1 - v_2 \sin\theta_2]\}^2} \tag{7.15}$$

When the distance between two nodes becomes larger than the transmission range the nodes will be disconnected. For transmission range r, link stability L_{stab} between any two nodes overtime period t can be calculated by:

$$L_{stab} = r/D \tag{7.16}$$

L_{stab} is the link stability of individual links between any two nodes and for a path, it is a concave parameter(as L_{stab} tends to zero either when transmission range r too small or distance between two nodes becomes too large) and it is same as the minimum link stability along the path.

Finally, the combined metric for mobility prediction is given by

$$CP = L_l\ (T_{Pr}) + L_{stab} \tag{7.17}$$

7.3 Weighted Clustering Algorithm [58]

7.3.1 Cluster Formation

Let n be the maximum allowable number of members of cluster. Let c, a counter maintained by each node. When a new node joins a cluster then the cluster head is going to check with the counter value, if c is less than n, then the counter value is going to be incremented. If c is greater than or equal to n then, that node cannot be able to join in the cluster [109].

Each node maintains a table, which contains the information about its neighbors. Each cluster heads are also going to maintain a table, containing the details of all other cluster heads.

Each node is going to calculate a weight value depends upon four parameters. The degree of difference (δ_i), Sum of distance to its neighbors (d_n), Average speed of every node (t_c) and the remaining battery power (p). The cluster head is going to be selected depends upon the weight value.

The degree of difference, $\delta_i = |d_i - N|$, where d_i is the number of neighboring nodes within the transmission range and N is the maximum cluster size. The sum of distance of every node is calculated by $D_n = \Sigma$ dst (n, n'), which is the sum of distance from a node to its neighbors. Average speed of every node is calculated by using, Avsi$=\frac{1}{t}\sum_{t=1}^{T}\sqrt{(x_t-x_{t-1})^2+(y_t-y_{t-1})^2}$, where x_t, y_t are the coordinates of node at time t.

The remaining battery power P is calculated by considering how much battery power has been consumed, the different kinds of roles like cluster heads or ordinary nodes. The weighted sum of these four factors are going to taken into account for calculating the combined metric of node weight.

$$W_n = (w1*\delta_i) + (w2*d_n) + (w3*\text{Avsi}) - (w4*P_n) \qquad (7.18)$$

HELLO messages are used to update the node and cluster tables. Each hello message contains the state of the node which is periodically exchanged between CHs or between each CH and its members. Before considering the cluster maintenance procedure, it is necessary to describe the process by which the node is able to compute its weight and several metrics under consideration. Depending upon the weight values of the node, the cluster head is elected.

The node with the smallest $\mathbf{W}_n$ is elected as a cluster-head. All the neighbors of the chosen cluster-head are no more allowed to participate in the election procedure.

All the above calculations are repeated for remaining nodes which is not yet elected as a cluster-head or assigned to a cluster.

In order to update the node_tables and CH_tables, node periodically calculates its weights and sends hello messages to its members and to the neighboring CHs. CH monitors the communication channel whether it hears any HELLO message or leave message. When the CH receives a leave message, it updates the node_table and broadcasts a HELLO message to its members and to its neighboring cluster heads. When the CH receives a HELLO message from a neighboring CH, it updates the CH_table. If

HELLO's source is a node member, CH updates a node_table and verifies the weight. In case of lowest weight, the CH must invoke the re-election procedure.

7.4 Effect of Node Mobility

In the event that a node moves starting with one cluster then onto the next cluster, both the cluster heads (cluster in which the node leaves and the cluster in which the node joins) need to know details of the moving node. Consider node 19 of cluster C moves to cluster D (Figure 6.4), then both the cluster heads (CHC and CHD) needs to think about the movement.

Depends upon the mobility prediction, the corresponding updates are going to be done in each cluster heads, before the movements of the nodes. By using this mobility prediction technique, we can be able to predict the movements and depends upon that, the corresponding changes are going to be made in each cluster.

7.5 Simulation Results

7.5.1 Simulation Model and Parameters

Network Simulator (NS2) is utilized to simulate the proposed algorithm. In the simulation, the channel limit of mobile hosts is set to the same value: 2 Mbps. The distributed coordination function (DCF) of IEEE 802.11 is utilized for wireless LANs as the MAC layer protocol. It has the functionality to inform the network layer about link breakage.

In the simulation, mobile nodes move in a 1000 meter x 1000 meter network region for 50 seconds simulation time. The number of nodes are fluctuated as 10,20,30,..50. Accept that every node moves autonomously with the same normal speed. All nodes have the same transmission range of 250 meters. In the simulation, the minimal speed is 5 m/s and maximal speed is 10 m/s. The simulated traffic is Constant Bit Rate (CBR). The no. of attackers are fluctuated as 1 to 10.

The simulation settings and parameters are summarized in table 7.4

No. of Nodes	10,20,30,…50
Area Size	1000 X 1000
Mac	802.11
Radio Range	250m
Simulation Time	50 sec
Traffic Source	CBR
Packet Size	512
Speed	5m/s t 10m/s
Misbehaving Nodes	1 to 10

Table 7.1 Simulation Settings for PCTEKM

The following are the assumptions used for the proposed framework,

- A malicious node can compromise the key, create packet drop attack, routing overflow attack etc.
- A trusted node will be having a trust value greater than 0.5.
- A threshold trust is fixed as 0.5.
- Malicious node will be having a trust value less than 0.5
- Trust value 1 refers to full trust and 0 refers to complete distrust.
- The designated cluster head will be the coordinator node.

- The coordinator node is the in charge of CRL update.
- Protocol used is AWCBRP.
- Nodes are moving with different velocities, ranging 5m/s to10m/s.

7.5.2 Performance Metrics

We compare the proposed Predictive Clustering Technique for Effective Key Management in Mobile Ad Hoc Networks (PCTEKM) with Distributed Hierarchical Key Management Scheme for Mobile Ad hoc Networks (DHKM) proposed in chapter 6.

We evaluate mainly the performance according to the following metrics [91]:

- **Average end-to-end Delay:** the normal time taken by the data packets from sources to destinations, including buffer delays during a route discovery, lining delays at interface lines, retransmission delays at MAC layer and propagation time.
- **Packet Delivery Ratio:** or packet throughput, the fraction of the data packets conveyed to destination nodes to those sent by source nodes.
- **Packet Drop** It is the number of packets dropped during the transmission.
- **Misdetection Ratio:** the proportion of the number of nodes whose behavior (malicious or considerate) is not recognized effectively to the real number of such nodes in the network.
- **Routing packet overhead**: the number of control packets (including route request/reply/update) for establishing connection over a period of time.
- **Resilience against Node Capture:** the fraction of communications compromised to the total number of communications by a capture of x-nodes.

7.5.3 Results

Varying Number of Attackers

The number of attackers is increased from 1 to 10 and the performances of the techniques are measured in terms of Delivery Ratio, Misdetection and Resilience. Since the prediction methodology for identifying the node movement, the performance of the proposed method will improve and the security will be high. The trust calculation and the isolation of malicious nodes are also incorporated.

Figure 7.2 shows the average Packet delivery Ratio of the schemes, when the attackers are increased from 1 to 5. We can see that the delivery ratio decreased linearly as the attacker increases. But, the delivery ratio of our proposed PCTEKM is greater than DHKM. Here also the trusted mechanism for the ID exchange and revocation is used.

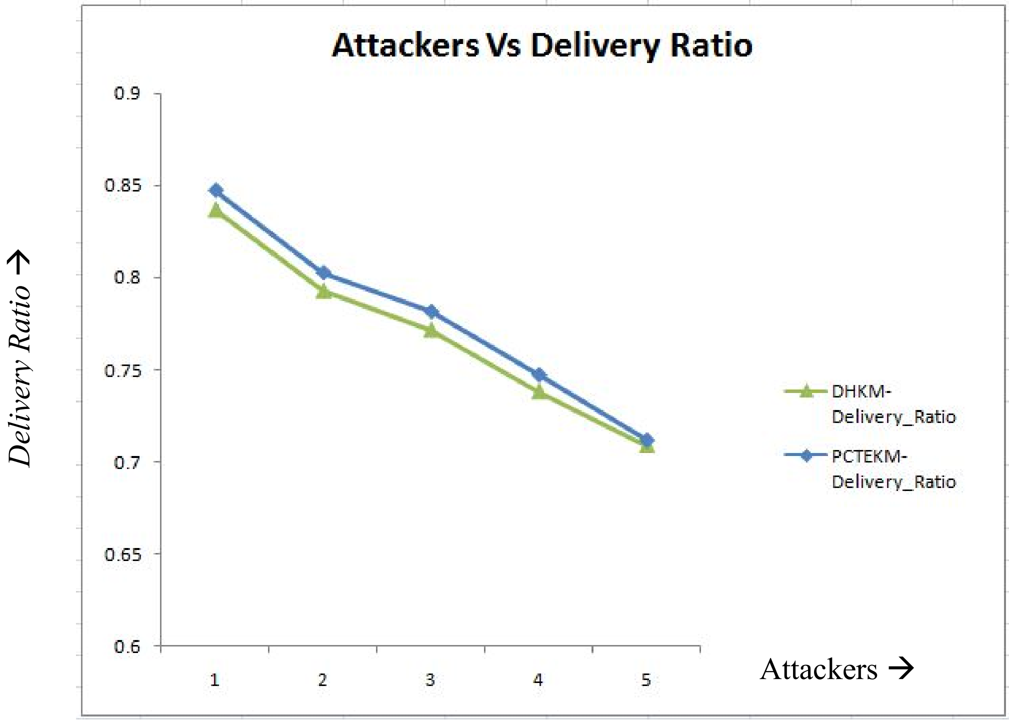

Fig 7.2 Packet Delivery Ratio

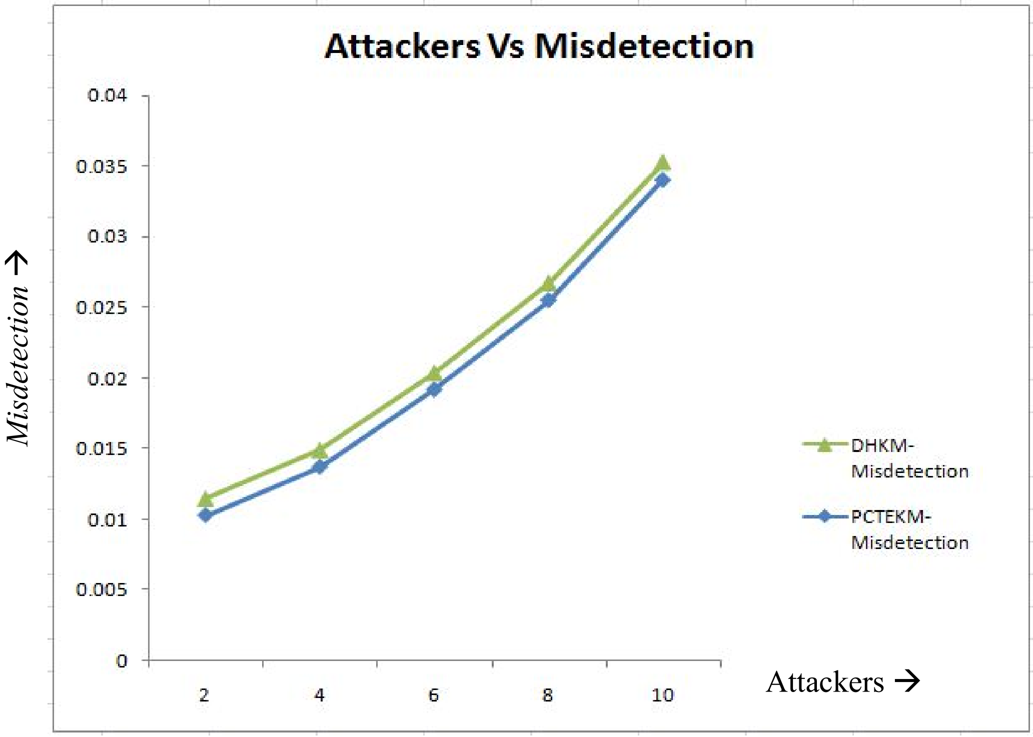

Fig 7.3 Misdetection Ratio

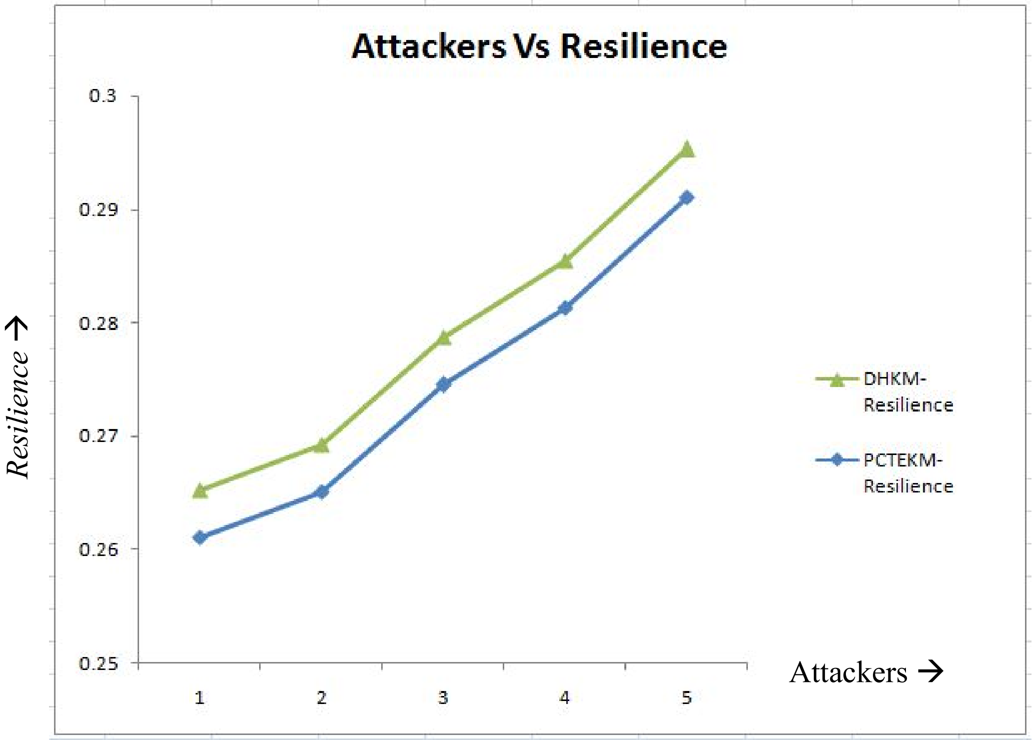

Fig 7.4 Resilience against Node Capture

The ratio of the number of nodes whose behavior is not identified correctly to the actual number of such nodes in the network is shown in figure 7.3. Our proposed method is capable to detect more malicious nodes while comparing with DHKM.

The result of fraction of compromised communications is shown in figure 7.4. Because of the trust prediction mechanism, the number of compromised communications is less in PCTEKM. Hence the proposed PCTEKM is more resilient than DHKM.

Varying Number of Nodes

The CBR data packets and control packets dropped due to the attackers, presented in figures 7.5. As the number of attacker increases, more data packets are dropped. But PCTEKM has less packet drops when compared to DHKM. The cluster head updates are done depends upon the predicted mobility value, there by improves the reliability of clustering algorithm.

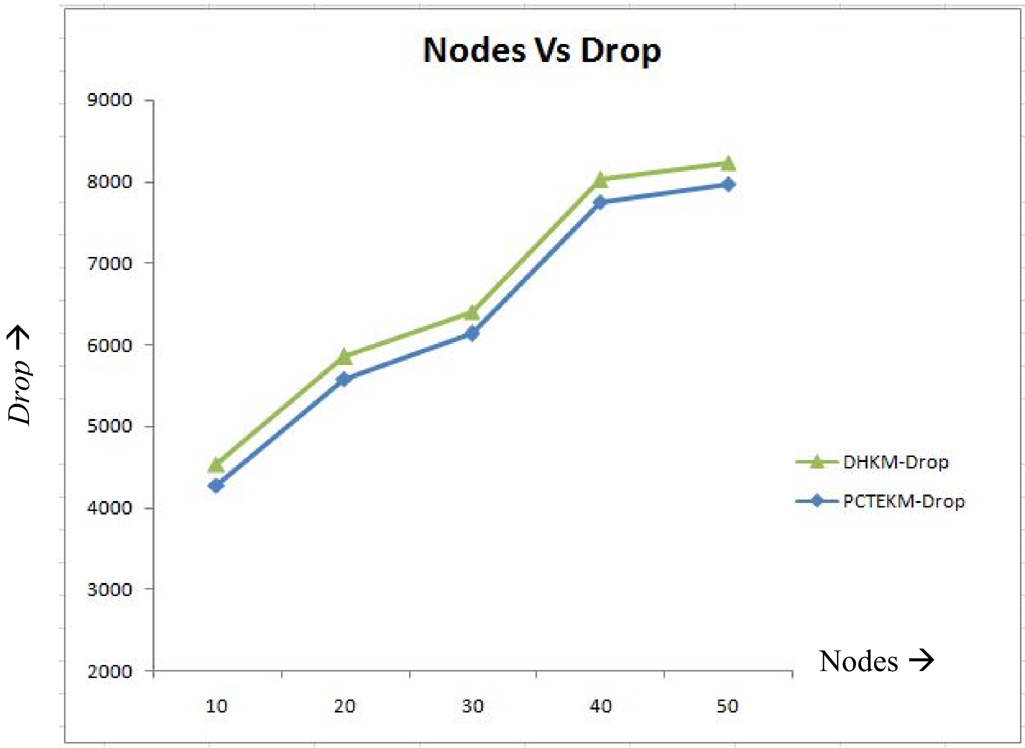

Fig 7.5 Packet Drop

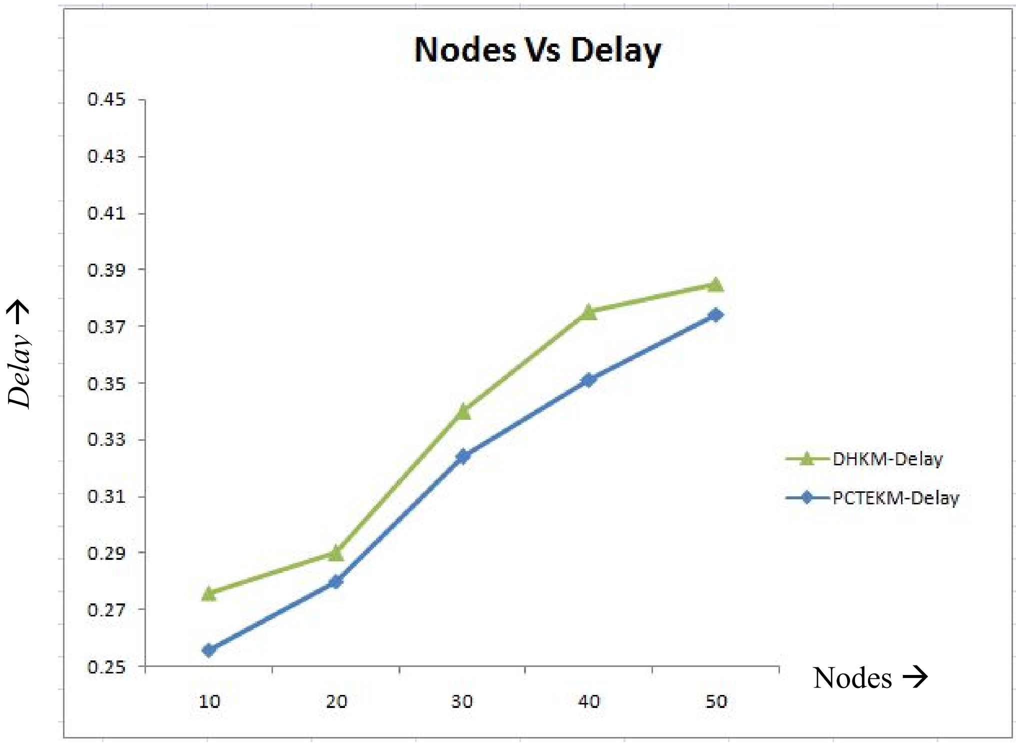

Fig 7.6 Average end-to-end Delay

Figure 7.6 depicts the delay involved in the communication by each pair of source and destinations. The number of nodes is varied from 10 to 50, and corresponding delay for the PCTEKM and DHKM are measured. The proposed method outperforms the DHKM in case of delay. Because of the mobility prediction, the cluster heads are updated proactively and the delay will be less compared to existing methodology. Because of the updates the network resilience also improves.

Figure 7.7 shows the Routing packet overhead of the schemes, when the nodes are increased from 10 to 50. We can see that the overhead of our proposed PCTEKM is greater than the DHKM since the proposed method contains the mobility prediction based on the past performance of the other nodes.

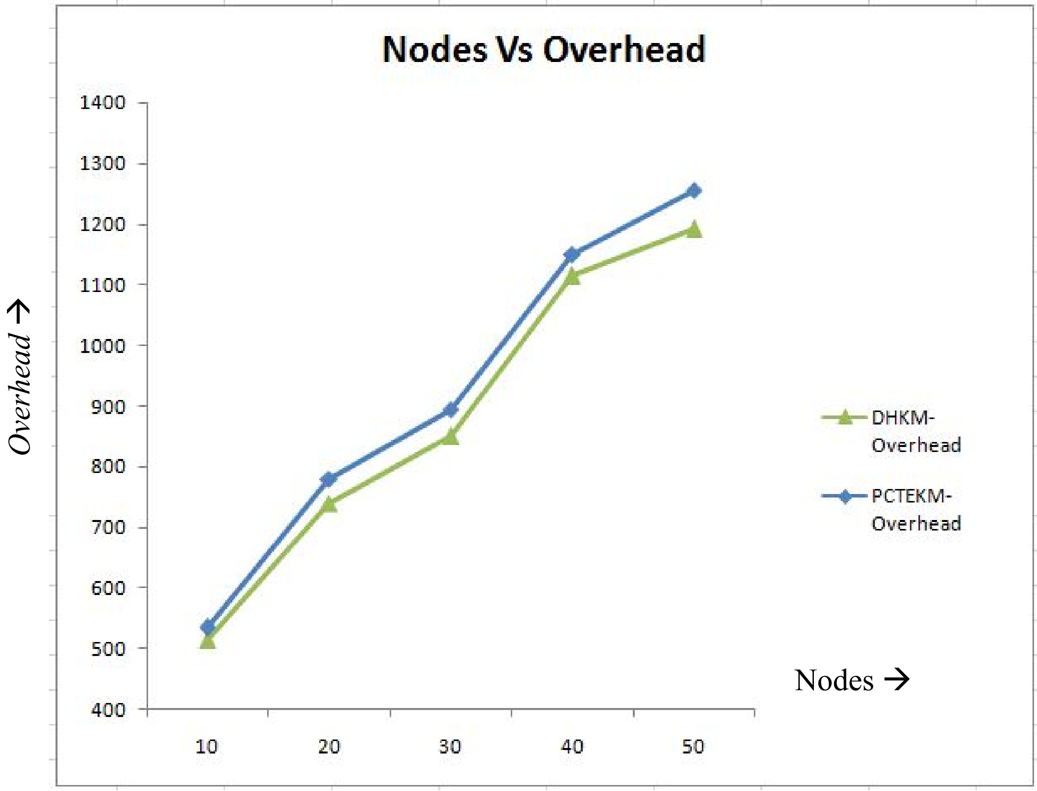

Fig 7.7 Routing packet overhead

7.6 Conclusion

In this chapter, our proposed framework (distributed hierarchical key management scheme) has been enhanced by a predictive clustering technique. A prediction based clustering technique is used to reduce the delay and packet drop due to node mobility.

Here we are predicting the movement of a node from one cluster to another based on the route expiration time and link stability. So each cluster head is going to predict the movement of its members from one cluster to another. Based on the predicted value, the public/private key pair update is happening in the cluster heads. The cluster heads are updated proactively, based on the combined mobility metric. The overall delay of the proposed system is reduced due to the prediction based clustering technique that we are using. The overhead will be high because of the prediction mechanism, we have incorporated.

The clustering technique used to select a CH, is based on weighted clustering algorithm. The CH is stored with public keys of all its member nodes. The communication of nodes between two different clusters happens through their CH. This method also discusses about the effects of node mobility between clusters. By Simulation results, it is shown that the proposed scheme achieves better delivery ratio and resilience with reduced delay and packet drop. The detection ratio of the proposed method is better than the existing one, as we have incorporated the method to predict the node movement.

CHAPTER 8

CONCLUSION AND FUTURE WORK

CONTENTS

8.1 Summary

The theme of the thesis is centered on one critical part of mobile adhoc networks; the trust based key management. Key management for MANET is a basic issue that has been discussed and solutions for it have been proposed in view of trust management mechanism. The review introduced in the Chapter 2 has offered promising changes over the routine certificate exchange and revocation techniques. Each of the schemes in the literature has its own advantage, disadvantage, limitation, operation criteria, design issues and application. The key management scheme depends upon the application situation for which it is designed. A harmony between the utilization and the accessible resource of power, computation figures out which key Management mechanism is to be deployed. The trusted intermediaries are essential for keeping communications alive and free from attacks. However there is still much work to be done. The proposed trust based mechanisms in this thesis are vital for secure key management and routing in MANETS. The execution results of all the proposed techniques are exhibited utilizing differed simulation situations utilizing network-simulator 2. In this framework different aspects were discussed for establishing trust based key management in mobile adhoc networks.

First we discussed certificate exchange/revocation method of our frame work. In this scheme self-organized key management for trusted certificate exchange and revocation is proposed in which the coordinator goes about as mediator for transmitting the message among the servers and mobile nodes. Every node produces its own public/private key pairs utilizing threshold self-certified public keying strategy. Multi-path certificate exchange method and certificate exchange procedure is utilized. As a result of various autonomous certifications, the confidence assigned to the certificates is higher. At the point when the source node needs to forward the data packet to destination, it discards the malicious nodes in that path and sidesteps the data through different nodes in exchange chose path towards the destination utilizing multipath system and source performs the certificate revocation process for guarding against the malicious nodes.

In the second scheme, a trust prediction model is proposed in view of accusations for certificate exchange and revocation in MANET. The trust value is computed from three distinct sorts of trust values, for example, verifiable trust, current trust and route trust. Node's verifiable trust can be evaluated by method for the node's physical neighbors in light of historical interaction information. A node's current trust can be computed from the node's verifiable trust taking into account the fuzzy logic rules prediction system. Route trust can be computed by intermediate nodes trust values along the route. Trust based accusation scheme is utilized to overcome with the malicious nodes.

In the third scheme, the effect of M-OLSR protocol in the proposed framework is examined. The architecture comprises of normal nodes and shareholder nodes. A random shift mechanism is utilized to choose the coordinator node among shareholder nodes. The coordinator node is in charge of the maintenance of CRL. Here, we are also addressing the link failure problem in source routing. Trusted re-computation of routes is introduced when link failure occurs. The proposed scheme is simulated and execution correlations with the fundamental methodology are displayed. A timestamp exchange mechanism is employed to address the replay attacks.

In the fourth scheme a distributed hierarchical key management scheme has been proposed utilizing a stable and power effective cluster management strategy. Every cluster head has the public key of its part nodes and go about as a router while managing the nodes of other cluster individuals. The communication of nodes between two unique clusters happens through their CH. A cluster based procedure is utilized to decrease the two limitations i.e., to over reliant on centralized server and increment in key-pair when node increments which SMOCK posses. The clustering system chooses a CH utilizing an adaptive weight clustering technique. This technique likewise discusses about the impacts of node mobility between clusters. The need of every node storing all the public keys are reduced therefore minimizing the storage overhead on every node.

In the fifth scheme the proposed system predicts the node movement and proactively sends information in case of any cluster movement. The cluster head is chosen taking into account weight values of the nodes, and low weighted nodes is picked as CH. If a node starts to move starting with one cluster then onto the next cluster, the source cluster head predicts the node mobility and send the source cluster details to the objective cluster head. If a cluster head is prone to move, it can be anticipated and cluster re-election procedure is performed. The node can no more perform its activity when mobility prediction for the cluster-head is more prominent than threshold. New head must be chosen from the accessible individuals. Every cluster head holds the public key of its part nodes just and go about as a router when managing nodes of other cluster individuals.

The fundamental goal of the thesis was to design a trust based key management framework for MANET comprising of, a trusted certificate exchange and revocation, trust prediction, path establishment when link failure occurs, key management for cluster based MANET and prediction based node movement for secure communication. A progressive approach was followed to accomplish this objective.

8.1.1 Comparative Study

In this section we highlight how different chapter's progresses to accomplish different objectives of the thesis and the difference between the outcomes of each proposal made in the thesis.

The proposed works mainly deal with the trust based certificate exchange and the revocation for efficient key management in MANET. The revocation process is carried out by validating the nodes in the network with the help of trust management mechanisms. SOKMTC, proposed in chapter 3 of our thesis uses Eigen vector Reputation Centrality technique for computing the trust value. The system is capable to identify and isolate the nodes which show malicious behavior. The framework maintains a certificate revocation list(CRL) to avoid communications through the already identified malicious nodes. The paths which include nodes in the CRL will be excluded from the considerable paths and also the system considers only the paths which are free from the malicious nodes.

SOKMTC considers the trust value for identifying the malicious nodes. The Eigen vector reputation centrality mechanism is itself capable to identify the misbehavior of nodes. There is a chance that a malicious node can contribute a falsified trust value during the route discovery process. SOKMTC considers the number of certifiers in the path to overcome such attacks. It may fail while handling with the paths having a same number of certifiers. At that time the system will select the shortest path among them. That might be the one which is having the malicious node.

TPMCER, proposed in chapter 4 of our thesis is capable to overcome the above limitation of SOKMTC. Only a better elimination scheme can provide an ideal system. SOKMTC possess a onetime accusation based elimination process. That is the main limitation of SOKMTC. TPMCER overcomes that limitation by possessing a collective accusation based elimination process. This can be explained with the help of following example.

Figure 8.1 represents the basic architecture of SOKMTC framework. Here the source S wants to send data to D. SOKMTC initiates the route discovery process and end up with the following paths,

Path 01: S-A-P-B-C-D

Path 02: S-A-P-B-Q-D

Path 03: S-A-M-B-C-D.

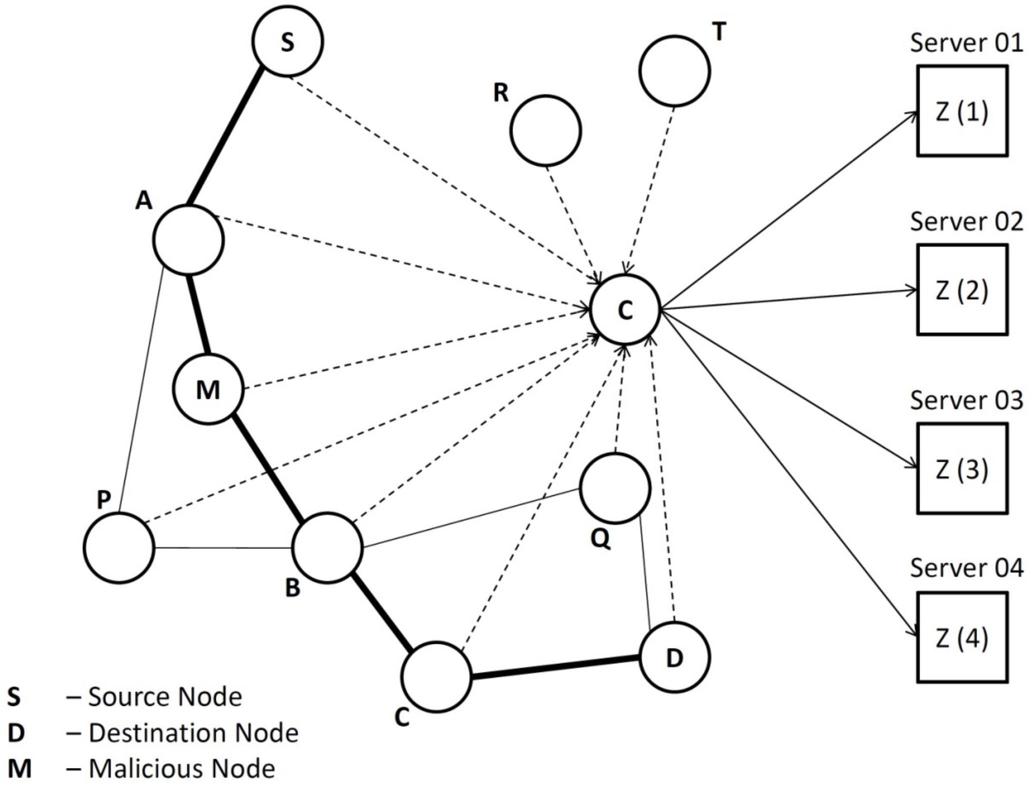

Fig 8.1 Comparing SOKMTC and TPMCER

Assume that path 01 is having minimum threshold trust value and path 02 does not have minimum threshold trust value. Path 03 does not possess the minimum trust value due to the presence of malicious node M. But, malicious node M can falsely claim, that it has trust value greater than the threshold. Then the source will consider the path 01 and path 03 for path selection. After completing the first phase of elimination, SOKMTC checks with the number of certifiers for each path. The path with the more number of certifiers will be selected as the routing path and if the number of certifiers is same, the shortest path will be selected for routing.

Assume that, the number of certifiers is same in our current scenario. Then path 03 will be selected for routing, which contains malicious node M. Thus the system may wrongly select the malicious path due to the falsified entry of trust value by the malicious node.

TPMCER has an accusation based false entry identification technique together with the trust prediction method. An accusation can become legitimate only if the source gets the same accusation from a countable number of nodes. Rather than collecting the trust value from the malicious node, TPMCER collects the accusations from the one hop neighbors. Thus the path {S-A-M-B-C-D} will be excluded before considering the number of certifiers and also the path with legitimate nodes will remain {S-A-P-B-C-D}.

SOKMTC propagates data packets strictly based on obtained source route. Thus it is not capable to identify the link failure as well as a CRL update after the route discovery process. The system will be a failure in both cases. Route re-computation at each intermediate node can become the solution for those two problems. Route re-computation will add additional delay to the process. The reactive behavior of the base protocol will not allow us to incorporate such a solution to the system.

SMRP, proposed in chapter 5 of our thesis always out performs SOKMTC due to its proactive nature. SMRP is capable to perform a route re-computation process at required nodes to overcome the drawbacks of SOKMTC. During data transmission, intermediate nodes will check the existence of next hop in their neighbor set. The existence of next hop in the neighbor set results in the continuation of data transmission as per the source route. Otherwise SMRP will initiate the route re-computation process.

Thus the certificate exchange and data transmission process will be carried out successfully.

SOKMTC, TPMCER & SMRP each node generates public/ private key pair. In traditional public key cryptography each node stores one private key and all the private keys of other nodes. We have to reduce the storage overhead of each node and also need to reduce the number of public-private key pairs used in the applications. DHKM, proposed in chapter 6 of our thesis is capable to reduce the storage overhead of each node by using a cluster based distributed combinatorial key management mechanism.

In DHKM, the cluster head is maintaining all the public keys of its member nodes. The cluster heads acts as routers during the intra/inter cluster communications. Due to high mobility, the delay for updating the cluster head will be high. The PCTEKM, proposed in chapter 7 of our thesis is capable to reduce the overall delay happening due to cluster head update by using efficient predictive clustering technique, where the movement of a node from one cluster to another is predicting beforehand and cluster update process is happening without much delay.

8.1.2 Performance Study

In this section we have highlighted a detailed performance analysis of the proposals made in the thesis.

In chapter 3 we have proposed a self organized trust based framework for establishing secure path in MANET. The framework consists of various components

1. Public/Private key pair generation/Distribution
2. Multipath certificate Exchange

3. Trust Management mechanism
4. Malicious node detection
5. Isolation of Malicious node by certificate revocation.

The proposed system (SOKMTC) was effectively incorporated in to the AOMDV protocol. We have compared the performance of the protocol with the basic AOMDV, On-demand Self-Organized Public Key Management (SOPKM) scheme and Ad hoc on-demand trusted-path distance vector (AOTDV) routing protocol. We demonstrate our protocols resiliency against attacks. Here we address the node capture attack. A malicious node is assumed to drop the packets, sent spurious certificates, creating routing inconsistency by flooding different messages. The performance analysis parameters are divided into quality of service parameters and security parameters. We are considering the Average end-to-end Delay, Packet Delivery Ratio, Packet Drop and Routing packet overhead to determine the quality of service of the protocol Misdetection ratio and Resilience against Node Capture are the security related parameters.

From the simulation results it is clear that the proposed method outperforms the existing mechanisms. The resilience of the protocol is improved, that means the capacity of the protocol to defend against attacks is improved. The detection ratio is improved, that is the capacity to identify the malicious behavior of the node is high. The performance of a key management scheme and certificate exchange based on trust is evaluated by the overall protocol resiliency. The overall protocol resiliency has improved because of the various components that we have used in the framework.

In chapter 4 we have proposed a trust prediction based framework for establishing secure path in MANET by eliminating the malicious node. The framework consists of various components as proposed in chapter 3. But here we have used a trust prediction mechanism to manage the trust. The proposed framework (TPMCER) is compared with our earlier framework (SOKMTC). The performance analysis parameters and settings are same. From the results it is clear that the TPMCER performs better SOKMTC. TPMCER overcomes the drawbacks of SOKMTC. The proposed approach upgrades the security against node capture attacks and enhances the packet delivery ratio and detection rate. The approach also achieves better resilience; reduced delay and packet drop even though the packet overhead is high because of the accusation based scheme and prediction technique that we have incorporated.

In chapter 5 we proposed a secure framework (SMRP) for Multipath Optimized link state routing protocol (M-OLSR). The proposed scheme additionally performs Share Holder Identification Process, Trust based route recovery Mechanism and Timestamp Exchange. The proactive nature of SMRP reduces the overall delay, even though it has an additional delay due to trusted path re-computation. We compare the proposed technique with M-OLSR and SOKMTC. The simulation results strengthen our views about SMRP. SMRP has more overhead while comparing with M-OLSR, but it has reduced overhead with SOKMTC.

In chapter 6 we proposed a cluster based approach based on trust to avoid the problems faced by the self certified method. Here, the selections of key pairs are dependable on two factors; memory and protection for key exposure upon attacks or node compromise. The proposed method achieves high ratio of malicious node detection, reducing the memory space utilization of each node. There by increases the performance and security of the communication. We compare the proposed Distributed Hierarchical Key Management Scheme for Mobile Ad hoc Networks (DHKM) with scalable method of cryptographic key management (SMOCK) and Cluster based routing protocol (CBRP). By Simulation results, it is shown that this proposed scheme achieves better resilience and delivery ratio with reduced delay and overhead. The method also achieves better malicious node detection ratio as we have incorporated trust management mechanism.

In chapter 7, an improved progressive key management scheme utilizing a stable and power efficient cluster management procedure has been proposed. The prediction strategy is fused in the proposed distributed various leveled key management scheme. The proposed method predicts the node movement and updates the details in the cluster head, if there is inter-cluster movement by a node in the near future. The consolidated metric for prediction is evaluated taking into account route expiration time and link stability. We compare the proposed Predictive Clustering Technique for Effective Key Management in Mobile Ad Hoc Networks (PCTEKM) with Distributed Hierarchical Key Management Scheme for Mobile Ad hoc Networks (DHKM) proposed in chapter 6. By Simulation results, it is shown that the proposed scheme achieves better resilience and delivery ratio with reduced delay and overhead. The malicious node detection ratio of the proposed method is better than the existing one, as we have incorporated the method to predict the node movement.

8.2 Contributions

Designed a trust based key management framework for MANET. This framework consists of,

- A trusted certificate exchange and revocation for secure communication in MANET. This method enables trust based secure routing, malicious node detection and isolation dynamically.
- Trust prediction model based on accusations for certificate exchange and revocation. The model helps in high detection ratio of malicious nodes and ensures secure trusted routing.
- Multipath certificate exchange based on M-OLSR protocol. This reduces link failure because of the trusted route re-computation mechanism. Also the malicious node detection rate is high.
- Cluster based hierarchical key management scheme. This scheme minimizes the number of public/private key usage, gives more security than the traditional key management mechanisms, random shift excludes the risk of one point failure in the network.
- Predictive clustering technique for addressing node mobility. The prediction reduces the overall delay in the network and minimizes the number of public/private key pair usage.

8.2.1 Applications of the proposed Framework

The following are the various applications of trust based frame work.

- Establishing communication among group of soldiers for tactical operations.
- Coordination of military objects in battle field.
- Coordinating vehicle mounted nodes.
- Collaborative computing.
- Crowd control.
- Search and rescue.
- Commando operations

The self organized nature of our frame work based on trust is very essential for the above applications. The independent Key generation, Key distribution based on certificate exchange, selecting the path based on trust, Revocation and isolation of malicious nodes mechanisms will help the application to execute in a secured manner.

8.2.2 Limitations of the proposed Framework

The proposed frame work is having the following limitations.

- Dynamic threshold fixation of trust is not considered. It is essential for high mobility environment. A mechanism may be proposed to update the trust dynamically in future.
- Different kind of group attacks and collaborative attacks not considered for analyzing the performance of the system. More kind of attacks can be addressed and mechanisms can be proposed to improve the protocol resilience in future.

- Hybrid routing protocols performance has not considered by using the frame work. A study can be made by using hybrid routing protocol in future.
- The isolation of the node is done by using a list called CRL (Certificate Revocation List). The delay and the cost incurred during the update and circulation of CRL list is affecting the performance. A study based on this can be done in future.

8.3 Future Directions

The framework is designed in AOMDV and M-OLSR protocols in Mobile adhoc network. The M-OLSR based scheme showed better performance, since it contains the route re-computation mechanism and timestamp mechanism. The memory cost for the proactive type protocols will be high, since it has to maintain the routing table. The work can be extended to study the robustness of Wireless Ad Hoc Networks for hybrid routing protocols. A quick response mechanism can be created for proactive protocols to diminish packet drop because of route changes.

We distinguish the behavior of the nodes whether it is malicious or not. The source ascertains trust of every node dynamically and the malicious nodes will be isolated when the trust is underneath a threshold. A study can be directed on the relationship between the normal detection delays and the mobility of the nodes can be made.

The revocation procedure of the nodes is done by the CRL list maintenance and broadcast mechanism. We have not considered the list update cost and the security of the CRL list broadcast is not guaranteed. A study taking into account Certificate Revocation list update cost and the security in broadcasting the CRL list can be made.

Here we are addressed the node capturing attack, where the keys will be compromised and the nodes can start routing attacks like adjusting the sequence no, hop count, false broadcast, routing table overflow attacks and spoofing attacks and so on. More sorts of attacks including group attacks can be contemplated and their relations to the vulnerability of the framework can be determined.

REFERNENCES

[1] Sunil Taneja and Ashwani Kush, “A Survey of Routing Protocols in Mobile Ad Hoc Networks, International Journal of Innovation”, Management and Technology, Vol. 1, No. 3, August 2010 ISSN: 2010-0248, pp. 452-456.

[2] Kuldeep Sharma, Neha Khandelwal, Prabhakar.M , “An Overview Of security Problems in MANET”, Proceedings of the International Conference on Network Protocols (ICNP), 2010, pp. 197-213.

[3] Zaiba Ishrat, “Security issues, challenges & solution in MANET”, International Journal of Computer Science & Technology, ISSN : 0976-8491 (Online) | ISSN : 2229-4333(Print), IJCST Vol. 2, Issue 4, Oct – Dec, 2011, pp. 63-70.

[4] Priyanka Goyal, Vinti Parmar, Rahul Rish, “MANET: Vulnerabilities, Challenges, Attacks, Application”, IJCEM International Journal of Computational Engineering & Management, Vol. 11, January 2011 ISSN (Online): 2230-7893, pp. 32-37.

[5] Rajaram Ayyasamy and Palaniswami Subramani “An Enhanced Distributed Certificate Authority Scheme for Authentication in Mobile Ad-hoc Networks”, The International Arab Journal of Information Technology, Vol. 9, No. 3, May 2012, pp. 291-298.

[6] H. Deng, W. Li, and Dharma P. Aggarwal. "Routing Security in Ad Hoc Networks”. In IEEE Communications Magazine, Special Topics on Security in Telecommunication Networks, Vol. 40, No. 10, October 2002, pp. 70-75.

[7] Jean-Pierre Hubaux, Levente Buttyan and Srdan Capkun. “The Quest for Security in Mobile Ad hoc Networks”. In Proceedings of International Symposium on Mobile Ad Hoc Networking & Computing, ACM Press, 2001, pp. 146-155.

[8] Patrick Tague,Mingyan Li and Radha Poovendran," Mitigation of Control Channel Jamming under Node Capture Attacks" IEEE Transactions on Mobile Computing, Volume 8 , Issue 9, 2009, pp. 1221-1234.

[9] Patrick Tague ,David Slater , Jason Rogers and Radha Poovendran ,"vulnerability of network traffic under node capture attacks using circuit theoretic analysis ",In proceedings of the 27th conference on Computer Communications. IEEE INFOCOM, Washington, April 2008, pp. 161 – 165.

[10] Patrick Tague and Radha Poovendran," Modeling Node Capture Attacks in Wireless Sensor Networks", In proceedings of 46th Annual Allerton Conference on Communication, Control, and Computing, Washington, 23-26 Sept. 2008, pp. 1221-1224.

[11] Han Park and JooSeok Song," An Enhanced Key Management Scheme Based on Key Infection in Wireless Sensor Networks" World Academy of Science, Engineering and Technology 60, 2009, pp. 240-254.

[12] Yih-Chun Hu, D. Johnson, A. Perrig. "Rushing Attacks and Defense in Wireless Ad Hoc Network Routing Protocols". In Proceedings of the ACM Workshop on Wireless Security (WiSe 2003), USA, Sep. 2003, pp. 30-40.

[13] Tomar, P., Suri, PK, & Soni, MK, "A comparative study for secure routing in MANET", International Journal of Computer Applications, Vol. 4(5), 2010, pp. 17-22.

[14] Wenjia Li and Anupam Joshi, "Security Issues in Mobile Ad Hoc Networks - A Survey", Department of Computer Science and Electrical Engineering, University of Maryland, Baltimore County, pp. 135-147.

[15] Yongguang Zhang and Wenke Lee, Security in Mobile Ad-Hoc Networks, in Book Ad Hoc Networks Technologies and Protocols (Chapter 9), Springer, 2005, pp. 249-268.

[16] P. Papadimitratos and Z. J. Hass, Secure Routing for Mobile Ad Hoc Networks, inProceedings of SCS Communication Networks and Distributed Systems Modeling andSimulation Conference (CNDS), San Antonio, TX, January 2002, pp. 193-204.

[17] Y. Hu, A. Perrig and D. Johnson, Ariadne: A Secure On-demand Routing Protocol for Ad Hoc Networks, in Proceedings of ACM MOBICOM'02, Atlanta, Georgia, USA, 2002, pp. 21-38.

[18] K. Sanzgiri, B. Dahill, B. N. Levine, C. Shields, and E. M. Belding-Royer, A Secure Routing Protocol for Ad Hoc Networks, in Proceedings of ICNP'02, 2002, pp. 78-87.

[19] Y. Hu, D. Johnson, and A. Perrig, SEAD: Secure Efficient Distance Vector Routing for Mobile Wireless Ad Hoc Networks, Ad Hoc Networks, July 2003, pp. 175-192.

[20] Y. Hu, A. Perrig and D. Johnson, Packet Leashes: A Defense against Wormhole Attacksin Wireless Ad Hoc Networks, in Proceedings of IEEE INFOCOM'03, 2003, pp. 1976-1986.

[21] Y. Hu, A. Perrig, and D. Johnson. “Packet Leashes: A Defense Against Wormhole Attacks in Wireless Ad Hoc Networks”. Technical report TR01-384, Department of Computer Science, Rice University, Houston, December 2001, pp. 370-380.

[22] Sencun Zhu, Sanjeev Setia, Sushil Jajodia. “LEAP: Efficient Security Mechanisms for Large-Scale Distributed Sensor Networks”. In Proceedings of 10th ACM Conference on Computer and Communications Security (CCS '03), Washington D.C., ACM Press, October, 2003, pp. 62-72.

[23] Yih-Chun Hu, D. Johnson, A. Perrig. "Rushing Attacks and Defense in Wireless Ad Hoc Network Routing Protocols". In Proceedings of the ACM Workshop on Wireless Security (WiSe 2003), San Diego, CA, USA, Sep. 2003, pp. 30-40.

[24] Y. Hu, A. Perrig and D. Johnson, Rushing Attacks and Defense in Wireless Ad Hoc Network Routing Protocols, in Proceedings of ACM MobiCom Workshop - WiSe'03,2003, pp. 30-40.

[25] J. R. Douceur, The Sybil Attack, in Proceedings of the 1st International Workshop on Peer-to-Peer Systems (IPTPS'02), March 2002, LNCS 2429, pp. 251-260.

[26] Anuj K. Gupta, Dr. Harsh Sadawarti, "Secure Routing Techniques for MANETs", International Journal of Computer Theory and Engineering (IJCTE), ISSN: 1793-8201, Article No. 74, Vol.1 No. 4, October 2009, pp. 456-460.

[27] Jean-Pierre Hubaux, Levente Buttyan, Srdjan Capkun. The Quest forsecurity in Mobile Ad Hoc Networks. Proceedings of the 2001 ACMInternational Symposium on Mobile ad Hoc networking & computing, Long Beach, CA. 2001, pp. 146-155.

[28] Y. Hu, A. Perrig and D. Johnson, Ariadne: A Secure On-demand Routing Protocol for Ad Hoc Networks, in Proceedings of ACM MOBICOM'02, 2002, pp. 21-38.

[29] F.Wang, B. Vetter, and S.Wu, "Secure Routing Protocols: Theory and Practice," Technical Report, North Carolina State University, May 1997, pp. 180-187.

[30] Yongguang Zhang, Wenke Lee. Intrusion Detection in Wireless Ad-Hoc Networks. Proceedings of MobiCom 2000, Sixth Annual International Conference on Mobile Computing and Networking, Boston, MA, USA, 6-11 Aug. 2000, pp. 60-70.

[31] H. Li, Z. Chen, X. Qin, C. Li, H. Tan, “Secure Routing in Wired Networks and Wireless Ad Hoc Networks,” Technical Report, Department of Computer Science, University of Kentucky, April 2002, pp. 456-460.

[32] F. Stajano and R. Anderson, “The Resurrecting Duckling: Security Issues for Ad Hoc Wireless Networks,” Security Protocols, 7th International Workshop, LNCS, Springer-Verlag, 1999, pp. 172-182.

[33] N. Ahuja and A. Menon, “Security in Mobile Networks : Ad-hoc and Infrastructure,” Computer and Information Sciences, University of Florida, Dec 2001, pp. 331-341.

[34] M. Jakobsson, W. S, and Y. B, “Stealth Attacks on Ad-Hoc Wireless Networks,” in proc. Vehicular Technology Conf., October, 6-9 2003, pp. 2103-2111.

[35] T.A.Wysocki, A. Dadej, and B. J. Wysocki, Eds. “Secure routing protocols for mobile ad-hoc wireless networks,” inAdvanced Wired and Wireless Networks Springer, 2004, pp. 57-80.

[36] Y.C. Hu, D.B. Johnson and A. Perrig, “SEAD: Secure efficient distance vector routing for mobile wireless ad hoc networks,” IEEE,Proceedings of the Fourth IEEE Workshop on Mobile Computing Systems and Applications (WMCSA’02), 0-7695-1647-5, 2002, pp. 175-192.

[37] Perkins, Charles E., and Elizabeth M. Royer. "Ad-hoc on-demand distance vector routing." In Mobile Computing Systems and Applications, 1999. Proceedings. WMCSA'99. Second IEEE Workshop on, pp. 90-100. IEEE, 1999, pp. 90-100.

[38] Johnson, David B., and David A. Maltz. "Dynamic source routing in ad hoc wireless networks." Kluwer International Series in Engineering and Computer Science (1996), pp. 153-179.

[39] Yang, Hao, Haiyun Luo, Fan Ye, Songwu Lu, and Lixia Zhang. "Security in mobile ad hoc networks: challenges and solutions." Wireless Communications, IEEE 11, no. 1 (2004), pp. 38-47.

[40] Praveen Rai Shubha Singh, “A Review of MANET's Security Aspects and Challenges”, IJCA Special Issue on “Mobile Ad- hoc Networks” MANETs, 2010, pp. 162-166.

[41] K. Sahadevaiah and O.B.V. Ramanaiah, “Self-Organized Public Key Cryptography in Mobile Ad Hoc Networks”, Journal of Ubiquitous Computing and Communication, pp. 161-170.

[42] Kyul Park, Hiroki Nishiyama, Nirwan Ansari and Nei Kato “Certificate Revocation to Cope with False Accusations in Mobile Ad Hoc Networks”, IEEE, 2010, pp. 1-5.

[43] Wei Liu, Hiroki Nishiyama, Nirwan Ansari and Nei Kato “A Study on Certificate Revocation in Mobile Ad Hoc Networks”, IEEE, 2011, pp. 1-5.

[44] Wu, Bing, Jie Wu, and Mihaela Cardei. "A survey of key management in mobile ad hoc networks." Handbook of research on wireless security 2 (2007), pp. 1-23.

[45] Burnett, S. and Paine, S. (2001). RSA Security’s Official Guide to Cryptography, RSA Press, pp. 160-170.

[46] Menezes, A., Oorschot, P., and Vanstone, S. (1996). Handbook of Applied Cryptography, CRC Press, pp. 816-820.

[47] Dahshan, Hisham, and James Irvine. "A robust self-organized public key management for mobile ad hoc networks." Security and Communication Networks 3, no. 1 (2010), pp. 16-30.

[48] Abdelmajid HAJAMI and Mohammed ELKOUTBI, “A Distributed Key Management Scheme for MANET using council architecture”, International

Journal of Advanced Computer Science and Applications (IJACSA), Vol 1 Issue 3, 2010, pp. 213-220.

[49] Deng, Hongmei, Anindo Mukherjee, and Dharma P. Agrawal. "Threshold and identity-based key management and authentication for wireless ad hoc networks." IEEE International Conference on Information Technology: Coding and Computing, Washington DC USA, vol. 1, 2004, pp. 107-111.

[50] Yi, Seung, and Robin Kravets. "Composite Key Management for Ad Hoc Networks." In MobiQuitous, 2004, pp. 52-61.

[51] Hideaki KAWABATA, Yoshiko SUEDA, Osamu MIZUNO, Hiroaki NISHIKAWA and Hiroshi ISHII, " Self-Organized Key Management based on Trust Relationship List", International conference on intelligence in next generation networks (ICIN), 2008, pp. 259-265.

[52] Kitada, Y; Takemori, K; Watanabe, A; Sasase, I. "On Demand Distributed Public Key Management without Considering Routing Tables for Wireless Ad hoc Networks" 6th Asia-Pacific Symposium on Information and Telecommunication Technologies, 2005, pp. 375-380.

[53] Johann van der Merwe, Dawoud Dawoud and Stephen McDonald, " Trustworthy Key Management for Mobile Ad Hoc Networks", In proceedings Southern African Telecommunication Networks and Applications Conference (SATNAC), South Africa, 2004, pp. 136-140.

[54] Hella Kaffel-Ben Ayed, A. Belkhiri, " Toward a Peer-to-Peer PKI for Mobile Ad-Hoc Networks", Cyber Journals: Multidisciplinary Journals in Science and Technology, Journal of Selected Areas in Telecommunications (JSAT), 2011, pp. 51-61.

[55] Genevie`ve Arboit, Claude Cre´peau, Carlton R. Davis and Muthucumaru Maheswaran "A localized certificate revocation scheme for mobile ad hoc networks", Elsevier, 2006, pp. 17-31.

[56] Anne Marie Hegland, Pål Spilling, Øivind Kure and Leif Nilsen "Scalable Revocation in Hybrid Ad Hoc Networks The SHARL Scheme", Journal of Networks, Vol. 3, No. 6, June 2008, pp. 182-192.

[57] Wenbo He, Ying Huang, Ravishankar Sathyam, Klara Nahrstedt and Whay C. Lee, "SMOCK: A Scalable Method of Cryptographic Key Management for Mission-Critical Wireless Ad-Hoc Networks", IEEE Transaction on information forensic and security, Vol. 4, No. 1, 2009, pp. 140-150.

[58] S. Karunakaran and P.Thangaraj,"An adaptive weighted cluster based routing (AWCBRP) protocol for mobile ad-hoc networks", WSEAS Transaction on comunications, Volume 7 , Issue 4, 2008, pp. 248-257.

[59] Srdjan Capkun, Levente Buttya and Jean-Pierre Hubaux, "Self-Organized Public-Key Management for Mobile Ad Hoc Networks", IEEE Transaction on mobile computing, Vol 2, No 1, Jan 2003, pp. 52-64.

[60] Jaydip Sen, "A Robust and Efficient Node Authentication Protocol for Mobile Ad Hoc Networks", Second International Conference on Computational Intelligence, Modelling and Simulation (CIMSiM), Washington DC USA, 2010, pp. 476 – 481.

[61] Whitfield Diffie and Martin E. Hellman, "New Directions in Cryptography", IEEE Trans. Info. Theory, Vol. IT-22, No. 6, Nov. 1976, pp. 644-654.

[62] Mike Burmester and Yvo Desmedt, "A Secure and Efficient Conference Key Distribution System", Proc. EUROCRYPT'94, 1994, pp. 275–286.

[63] Klaus Becker and Uta Wille, "Communication Complexity of Group Key Distribution," Proc. 5th ACM Conf. Comp. and Commun. Security, 1998, pp. 1–6.

[64] N. Asokan and Philip Ginzboorg, "Key Agreement in Ad Hoc Networks", Computer Commun., vol. 23, no. 17, Nov. 2000, pp. 1627-1637.

[65] M. Steiner, G. Tsudik, and M. Waidner, "CLIQUES: A New Approach to Group Key Agreement," Proc. ICDCS'98, 1998, pp. 66-78.

[66] Seung Yi and Robin Kravets, "MOCA: MObile Certificate Authority for Wireless Ad Hoc Networks," Report No. UIUCDCS-R-2004- 2502, UILU-ENG-2004-1805, University of Illinois at Urbana- Champaign, 2002, pp. 1-19.

[67] Bing Wua, Jie Wua, Eduardo B. Fernandeza, Mohammad Ilyasa, Spyros Magliveras, "Secure and Efficient Key Management in Mobile Ad Hoc Networks," Proc. IPDPS'05, 2005, pp. 52-69.

[68] Jiejun Kong, Petros Zerfos, Haiyun Luo, Songwu Lu, Lixia Zhang, "Providing Robust and Ubiquitous Security Support for Mobile Ad-Hoc Networks," Proc. 9th Int'l. Conf. Network Protocols (ICNP'01), Los Angeles, 2001, pp. 251–60.

[69] Bo Zhu, Feng Bao, Robert H. Deng, Mohan S. Kankanhalli, Guilin Wang, "Efficient and Robust Key Management for Large Mobile Ad Hoc Networks," Computer Networks, vol. 48, no. 4, July 2005, pp.657–82.

[70] Lung-Chung Li and Ru-Sheng Liu, "Securing Cluster-Based Ad Hoc Networks with Distributed Authorities", IEEE transactions on wireless communications, vol. 9, no. 10, October 2010, pp. 3072-3081.

[71] Brian Cusack and Alastair Nisbet, "Secure Key Deployment and Exchange Protocol for Manet Information Management", Proceedings of the 10th Australian Digital Forensics Conference, Novotel Langley Hotel, Perth, Western Australia, December, 2012, pp. 1-15.

[72] Aydip Sen, "A Multi-Path Certification Protocol for Mobile Ad Hoc Networks", Poster Paper in Proceedings of the 4th International Conference on Computers &

Devices for Communications (CODEC), Institute of Radio Physics and Electronics, India, December 14 – 16, 2009, pp. 4-15.

[73] P. Caballero-Gil and C. Hern´andez-Goya, “Efficient Public Key Certificate Management for Mobile Ad Hoc Networks”, EURASIP Journal on Wireless Communications and Networking, Vol. 2011, Article 18, January 2011, pp. 1-21.

[74] Giannis F.Marias, Konstantinos Papapanagiotou, Vassileios Tsetsos, Odysseas Sekkas, and Panagiotis Georgiadis, “Integrating a Trust Framework with a Distributed Certificate Validation Scheme for MANETs”, EURASIP Journal on Wireless Communications and Networking, Vol. 2006, Issue 2, April 2006, pp. 1–18.

[75] Sushma Nayak and Ramakrishna M, “Certificate Path Discovery by Constructing Virtual Hierarchy to Administer Trust Relationship using Peer to Peer PKI in MANETs”, International Journal of Information and Electronics Engineering, Vol. 2, No. 2, March 2012, pp. 243-246.

[76] E. A .Mary Anita and V. Vasudevan, “Black Hole Attack Prevention in Multicast Routing Protocols for Mobile Ad hoc networks using Certificate Chaining”, International Journal of Computer Applications, Vol. 1, No. 12, 2010, pp. 21-28.

[77] Johann van der Merwe, Dawoud Dawoud, and Stephen McDonald, “Key Distribution in Mobile Ad Hoc Networks based on Message Relaying”, ESAS'07 Proceedings of the 4th European conference on Security and privacy in ad-hoc and sensor networks, 2007, pp. 87-100.

[78] Azeem Irshad, Wajahat Noshairwan, Muhammad Shafiq, Abdul Wahab Khan, Muhammad Usman and Ehtsham Irshad, “Certificate Chain based Authentication in MANETS using 4th Generation Technologies”, IEEE, 2009, pp. 250-254.

[79] Haiyun Luo, Jiejun Kong, Petros Zerfos, Songwu Lu and Lixia Zhang, “URSA: Ubiquitous and Robust Access Control for Mobile Ad-Hoc Networks”, IEEE/ACM Transactions on Networking, October 2004, pp. 1-16.

[80] T.R.Panke, “Clustering Based Certificate Revocation Scheme for Malicious Nodes in MANET”, International Journal of Scientific and Research Publications, Vol. 3, Issue 5, May 2013, pp. 1-5.

[81] Wei Liu, Hiroki Nishiyama, Nirwan Ansari, Jie Yang, and Nei Kato, “Cluster-based Certificate Revocation with Vindication Capability for Mobile Ad Hoc Networks”, IEEE Transactions On Parallel And Distributed Systems, Vol. 24 , Issue 2, 20 December 2012, pp. 239-249.

[82] Sudha Chinni, Johnson Thomas, Gheorghita Ghinea and Zhengming Shen, “Trust Model for Certificate Revocation in Ad hoc Networks”, Journal on Ad Hoc Networks, Vol. 6, Issue 3, May 2008, pp. 441-457.

[83] Jolyon Clulow and Tyler Moore, “Suicide for the Common Good: a New Strategy for Credential Revocation in Self-Organizing Systems”, ACM SIGOPS Operating Systems, Vol. 40, Issue 3, July 2006, pp. 18-21.

[84] Jun Luo, Jean-Pierre Hubaux and Patrick T. Eugster, “DICTATE: DIstributed CerTification Authority with probabilisTic freshness for Ad Hoc Networks”, IEEE Transactions On Dependable And Secure Computing, Vol. 2, No. 4, October-December 2005, pp. 311-323.

[85] Sonja Buchegger. “Coping with Misbehavior in Mobile Ad-hoc Networks”. Ph.D. Thesis number 2935, EPFL April 2004.

[86] Sonja Buchegger and Jean-Yves Le Boudec. “Nodes Bearing Grudges: Towards Routing Security”, Fairness and Robusteness in Mobile Ad hoc Networks. In Proceedings of the Tenth Ecurfomicro Workshop on Parallel, Distributed and networks based Processing, January 2003, pp. 403-410.

[87] Sonja Buchegger, Jean-Yves Le Boudec. “The Effect of Rumor Spreading in Reputation Systems for Mobile Ad-hoc Networks”. In Proceedings of WiOpt 2003: Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks, Sophia-Antipolis, France. March 2003, pp. 7-50.

[88] Hisham Dahshan and James Irvine, “On Demand Self-Organized Public Key Management for Mobile Ad Hoc Networks”, IEEE 69th Vehicular Technology Conference, VTC Spring, 2009, pp. 3037-3041.

[89] Vieu .V.B., Nasser .N., & Mikou .N. (2006). A Weighted Clustering Algorithm Using Local Cluster-heads Election for QoS in MANETs. IEEE GLOBECOM, pp. 344-356.

[90] Sameh R. Zakhary and Milena Radenkovic, “ Reputation-Based Security Protocol for MANETs in Highly Mobile Disconnection-Prone Environments”, The Seventh International Conference on Wireless On-demand Network Systems and Services, (IEEE/IFIP WONS), Washington DC USA, 2010, pp. 42-46.

[91] X. Li, Z. Jia, P. Zhang, R. Zhang and H. Wang, "Trust-based on-demand multipath routing in mobile ad hoc networks", Vol. 4, Iss. 4, 2010, pp. 212-232.

[92] Hui Xia, Zhiping Jia, Lei Ju, Xin Li, Edwin H. and M. Sha b. "Impact of trust model on on-demand multi-path routing in mobile ad hoc networks". Computer Communications Journel, Elsevier, 2012, pp. 381-402.

[93] Hui Xia, Zhiping Jia, Lei Ju, Xin Li, Edwin H. and M. Sha b. "Trust prediction and trust-based source routing in mobile ad hoc networks", Ad Hoc Networks, Elsevier, 2013, pp. 1078-1091.

[94] Jiazi Yi, Asmaa Adnane, Sylvain David, Benoît Parrein, “ Multipath optimized link state routing for mobile ad hoc networks”, Journal Ad Hoc Networks, Volume 9 Issue 1, 2011, pp. 28-47.

[95] Shushan Zhao, Akshai Aggarwal, Shuping Liu, Huapeng Wu, " Secure Routing Protocol in Proactive Security Approach for Mobile Ad-hoc Networks", IEEE Wireless Communications and Networking Conference (WCNC), 2008, pp. 2627 – 2632.

[96] Jiazi Yi, Asmaa Adnane, Sylvain David and Benoit Parrein. "Multipath optimized link state routing for mobile ad hoc networks", Ad Hoc Networks, Elsevier, November 2009, pp. 28-47.

[97] E. Cizeron and S. Hamma, "A multiple description coding strategy for multi-path in mobile ad hoc networks", International Conference on the Latest Advances in Networks (ICLAN), Paris, France, 2007, pp. 95-107.

[98] J. Yi, E. Cizeron, S. Hamma, B. Parrein, "Simulation and performance analysis of MP-OLSR for mobile ad hoc networks", IEEE WCNC: Wireless Communications and Networking Conference, Las Vegas, USA, 2008, pp. 3101-3012.

[99] Mohamed Amine FERRAG, Mehdi NAFAA, " Securing the OLSR routing protocol for Ad Hoc Detecting and Avoiding Wormhole Attack", Cyber Journals: Multidisciplinary Journals in Science and Technology, Journal of Selected Areas in Telecommunications (JSAT), 2011, pp. 1-8.

[100] Frederic Cuppens, Nora Cuppens-Boulahia, Seila Nuon, Tony Ramard, "Property Based Intrusion Detection to Secure OLSR", Proceedings of the Third International Conference on Wireless and Mobile Communications, (ICWMC), Cambodia, France 2007, pp. 52-61.

[101] Alia Fourati, Khaldoun Al Agha, "A shared secret-based algorithm for securing the OLSR routing protocol", 2005, pp.481-494.

[102] Cedric Adjih, Pascale Minet, Paul Muhlethaler, Emmanuel Baccelli and Thierry Plesse, “ QoS support, security and OSPF interconnection in a MANET using OLSR”, Journal of Telecommunications and Information technology, 2008, pp. 5-8.

[103] K. Urmila Vidhya, M. Mohana Priya, “A novel technique for defending routing attacks in OLSR MANET”, IEEE International Conference on Computational Intelligence and Computing Research, 2010, pp. 12-18.

[104] Uike N, Lohkare V, Deulkar B, and Tayde P, “Mitigation of Flooding Disruption Attacks In OLSR Network”, International Journal of Networking, Volume 2, Issue 1, 2012, pp. 60-63.

[105] Andreas Hafsulund, Andreas Tonneses, Roar Bjorgum Rotvik, Jon Andersson and Oivind Kure, “Secure Extension to the OLSR protocol”, OLSR Interop and Workshop, 2004, pp. 67-75.

[106] Arvind Ramalingam, Sundarpremkumar Subramani and Karthik Perumalsamy (2002), "Associativity based cluster formation and cluster management in ad hoc networks", School of Computer Science and Engineering, Anna University, Chennai, pp. 1-5.

[107] Jiang, S., He, D. and Rao, J. "A prediction-based link availability estimation for mobile ad hoc networks", Proceedings of IEEE Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies, 2001, pp. 956-965.

[108] Mamun-Or-Rashid, M. D. and Hong, C.S. "LSLP: Link stability and lifetime prediction based QoS aware routing for MANET", JCCI 2007, Phoenix Park, Korea, May 2007, pp. 1-5.

[109] Sivavakeesar, S., Pavlou, G., Bohoris, C. and Liotta, A. "Effective management through prediction-based clustering approach in the next-generation ad hoc networks", Proceedings of the IEE ICC, Paris, France, June 2004, pp. 105-121.

List of Publications

1. Saju P John, Philip Samuel, *"Self-organized key management with trusted certificate exchange in MANET"*, **ELSEVIER**, Ain Shams Engineering Journal 6 (2015), pp 161-170.
2. Saju P John, Philip Samuel, *"A Predictive Clustering Technique for Effective Key Management in Mobile Ad Hoc Networks"*, **Taylor & Francis**, Information Security Journal: A Global Perspective, 20:250–260, 2011.
3. Saju P John, Philip Samuel, *"Secure multipath routing protocol certificate exchange for mobile adhoc networks"*, **Praiseworthy Publications**, International Review of Computers & Software (IRECOS), ISSN 1828-6003 Vol 8 N.7 July 2013.
4. Saju P John, Philip Samuel *"A Survey on key Management and certificate exchange in Mobile Ad Hoc Networks",* **IGI Publications,** International Journal of Business data Communication and networking. Volume 10 issue 2 April 2014 doi.10.4018/ ijbdcn. 2014040103.
5. Saju P John, Philip Samuel, *"Trust Prediction Model for Certificate Exchange and Revocation in MANET"*, Paper accepted in: International Journal of Networking and Virtual Organisations (IJNVO), **Inderscience Publishers Ltd.**
6. Saju P John, Philip Samuel, *"A Distributed Hierarchical Key Management Scheme for Mobile Ad hoc Networks"*, **IEEE** International Conference on Information, Networking and Automation (ICINA 2010), Kunming, China.
7. Saju P John, Philip Samuel, *"Secure Multipath Routing with Trusted Certificate Exchange and Revocation in MANET"*, Communicated to: International Journal of Mobile Network Design and Innovation (IJMNDI) **Inderscience Publishers Ltd.**