Design and Evaluation of Security Mechanism for Routing in MANETs

Elliptic Curve Diffie-Hellman cryptography mechanism to secure Dynamic Source Routing protocol (DSR) in Mobile Ad Hoc Network (MANET)

Sultan H. Almotiri

Submitted for the Degree of

Doctor of Philosophy

School of Computing, Informatics and Media

University of Bradford

2013

Acknowledgement

First of all, I want to thank Allah Almighty for giving me the health and the courage to finish my thesis.

No words can be enough to state my gratitude to my supervisor, Prof. Irfan U. Awan, Professor of Computer Science at Bradford University. He helped, motivated and supported me all through my research and my thesis write-up.

I am truly indebted and thankful to Dr. D. Holton for his support. My thanks also go to all staff members at Bradford University for making things easy for me.

Furthermore, I would like to extend my sincere thanks and gratitude to all my friends and especially Mohammed Al-Ghamdi, Abdulallah Al-Garni, and Ali Sebaa for their support and encouragement during my time at Bradford University. A very special thank you goes to my friend Jamal Al-Orabi who accompanied me throughout my stay in Bradford and gave me feedback and support during my studies.

Last, but not least, I would like to thank my closest family members. To my mother, I say thank you for her love and support. Without her prayers, I could not have finished my studies. I also would like to thank my father for encouraging me and my brothers for their love and support during my studies in Bradford. They have always helped me to strive towards excellence. My wife Hanaa deserves special mention for her patience and support. She has been a pillar throughout my research and my thesis write-up. Also, big thank you to my little girl Shahd who always brought a smile on my face and helped me relax during my research.

Abstract

Ensuring trustworthiness through mobile nodes is a serious issue. Indeed, securing the routing protocols in Mobile Ad Hoc Network (MANET) is of paramount importance. A key exchange cryptography technique is one such protocol. Trust relationship between mobile nodes is essential. Without it, security will be further threatened. The absence of infrastructure and a dynamic topology changing reduce the performance of security and trust in mobile networks.

Current proposed security solutions cannot cope with eavesdroppers and misbehaving mobile nodes. Practically, designing a key exchange cryptography system is very challenging. Some key exchanges have been proposed which cause decrease in power, memory and bandwidth and increase in computational processing for each mobile node in the network consequently leading to a high overhead. Some of the trust models have been investigated to calculate the level of trust based on recommendations or reputations. These might be the cause of internal malicious attacks.

Our contribution is to provide trustworthy communications among the mobile nodes in the network in order to discourage untrustworthy mobile nodes from participating in the network to gain services.

As a result, we have presented an Elliptic Curve Diffie-Hellman key exchange and trust framework mechanism for securing the communication between mobile nodes. Since our proposed model uses a small key and less calculation, it leads to a reduction in memory and bandwidth without compromising on security level. Another advantage

of the trust framework model is to detect and eliminate any kind of distrust route that contain any malicious node or suspects its behavior.

List of Figures

Figure 2.1: Mobile Ad Hoc Network (MANET) 8

Figure 2.2: Mobile Ad Hoc routing range coverage 11

Figure 2.3 Categorization of Mobile Ad hoc Routing Protocols 13

Figure 2.4: Route Discovery in AODV 16

Figure 2.5: Route Maintenance in AODV 17

Figure 2.6: Route Discover in DSR 19

Figure 2.7: Route Maintenance in DSR 20

Figure 2.8:Classification of active attacks on MANET routing protocols 25

Figure 2.9: Example of impersonation attack 26

Figure 2.10: Example of route sequence number modification attack 27

Figure 2.11: Example of a fabrication attack 28

Figure 2.12: Example of Denial of service attack 29

Figure 2.13: Example of wormhole attack 30

Figure 3.1: Symmetric key algorithm 36

Figure 3.2: Asymmetric key algorithm 37

Figure 3.3: Digital Signature 40

Figure 3.4: X.509 Digital certificate format 43

Figure 5.1: A catalogue of elliptic curves. 57

Figure 5.2: ECC diffie-hellman key exchange 64

Figure 5.3: 10 mobile nodes with 6 UDP links 70

Figure 5.4: 10 mobile nodes with 6 UDP links 71

Figure 5.5: 10 mobile nodes with 6 TCP links 72

Figure 5.6: 10 mobile nodes with 6 TCP links 73

Figure 5.7: 20 mobile nodes with 6 UDP links 74

Figure 5.8: 20 mobile nodes with 6 UDP links 75

Figure 5.9: 20 mobile nodes with 6 TCP links 76

Figure 5.10: 20 mobile nodes with 6 TCP links 77

Figure 5.11: 50 mobile nodes with 10 UDP links 78

Figure 5.12: 50 mobile nodes with 10 UDP links 78

Figure 5.13: 50 mobile nodes with 10 TCP links 80

Figure 5.14: 50 mobile nodes with 10 TCP links 80

Figure 5.15: TCP handshake 81

Figure 5.16: SYN attack 82

Figure 5.17: Session hijacking attack 83

Figure 5.18: Overhead 85

Figure 5.19 Packet Delivery Fractions 86

Figure 5.20 Average Latency 87

Figure 6.1: An example of the trust value mechanism 96

Figure 6.2: Trust DSR data packet header format (RREQ) 99

Figure 6.3: Trust DSR data packet header format (RREP) 100

Figure 6.4: Trust Model and ECDH on DSR routing protocol 101

Figure 6.5: CONFIDANT Architecture 103

Figure 6.6: Packet Delivery Rate (PDR) (in Low malicious nodes) 106

Figure 6.7: Packet Overhead (in Low malicious nodes) 107

Figure 6.8: Average Latency (in Low malicious nodes) 107

Figure 6.9: Malicious mobile node detecting (in Low malicious nodes) 108

Figure 6.10: Packet Delivery Ratio (PDR) (in High malicious node) 109

Figure 6.11: Packet Overhead (in High malicious node) 110

Figure 6.12: Average Latency (in High malicious nodes) 110

Figure 6.13: Malicious mobile node detecting (in High malicious nodes) 111

Figure 6.14: Trust Values for 50 nodes (in High malicious nodes) 112

List of Tables

Table 3.1: Some public key encryption schemes 41

Table 5.1: Keys sizes comparison with RSA, DH and ECC 58

Table 5.2 Parameters 66

Table 5.3 Scenarios Parameters 84

Table 6.1: Trust value level 95

Table 6.2: Example of the trust value mechanism 96

Table 6.3: Trust DSR and DSR algorithm 97

Table 6.4: Examples of ways to increase or decrease the trust value 98

Table 6.5: Parameters 104

List of Acronyms

ACK	Acknowledgement
AODV	Ad hoc On-Demand Distance Vector
ARAN	Authenticated Routing for Ad hoc Networks
Bid	Broadcast identifier
CA	Certificate Authorities
CONFIDANT	Cooperation Of Nodes: Fairness In Dynamic Ad-hoc Networks
CORE	A Collaborative Reputation Mechanism to enforce node cooperation in Mobile Ad hoc Networks
CRL	Certificate Revocation list
DH	Diffie-Hellman
Did	Destination identifier
DLP	Discrete logarithm problem
DoS	Denial-of-service
DSDV	Destination-Sequenced Distance-Vector
DSeq	Destination Sequence number
DSR	Dynamic Source Routing
ECC	Elliptic curve cryptography
ECCS	elliptic curve cryptography system
ECDH	Elliptic Curve Diffie-Hellman
ECDSA	Elliptic Curve Digital Signature Algorithm
ECIES	Elliptic Curve Integrated Encryption Scheme

HMAC	Hashed Message Authentication Code
IDS	Intrusion Detection Systems
KMS	Key Management System
MAC	Message Authentication Code
MANET	Mobile Ad Hoc Network
OLSR	Optimized Link State Routing Protocol
PDF	Packet Delivery Fraction
PGP	Pretty Good Privacy
PKI	Public Key Infrastructure
QoS	Quality of Service
RERR	Reoute Error
RDP	Route Discovery Packet
RREP	Route Reply
RREQ	Route Request
RSA	Rivest, Shamir and Adleman
SAR	Security Aware Routing
Sid	Source identifier
SRP	Secured Routing Protocol
SSeq	Source Sequence number
TARP	Trust Aware Routing Protocol
TCP	Transmission Control Protocol
TESLA	Timed Efficient Stream Loss-tolerant Authentication
TIK	TESLA with Instant Key disclosure
TTL	Time To Live

TTP	Trusted Third Party
TV-ECDH	Trust Value using Elliptic Curve Diffie-Hellman
UDP	User Datagram Protocol
WiMAX	Worldwide Interoperability for Microwave Access
WLAN	Wireless Local Area Network
WoT	Web of Trust

Table of Contents

Acknowledgement i

Abstract iii

List of Figures v

List of Tables viii

List of Acronyms ix

Table of Contents xii

Chapter 1: Introduction 1

1.1 Introduction 1

1.2 Research Motivation 3

1.3 Objectives 4

1.4 Contribution 4

1.5 Thesis Structure 5

1.6 Publications 6

Chapter 2: Background …………………………………….. 7

2.1 Mobile Ad Hoc Network (MANET) ……….......... 7

2.1.1 MANET Applications 8

2.1.2 Characteristics and challenges of MANET... 9

2.2 Routing Protocols in MANET ………………… 10

2.2.1 Challenges in Routing 11

2.2.2 Classification of Routing Protocols 12

2.2.3 AODV 14

2.2.4 DSR 17

2.3 Security in MANET ……………………………. 21

2.3.1 Security objectives …………….................. 21

2.3.2 Attacks ……………………………………. 22

2.3.2.1 Passive Attacks ………………………. 23

2.3.2.1.1 Eavesdropping ………………… 23

2.3.2.1.2 Traffic Analysis ……………….. 24

2.3.2.2 Active Attacks ………………………... 24

2.3.2.2.1 Impersonation Attacks ……….. 25

2.3.2.2.2 Modification Attacks ………….. 26

2.3.2.2.3 Fabrication Attacks ……………. 28

2.3.2.2.4 Denial of Service ……………… 28

2.3.2.2.5 Wormhole Attack ……………… 29

2.3.3 Cryptography ……………………………... 30

2.3.4 Trust Model ……………………………….. 31

2.4 Summary ………………………………………. 31

Chapter 3: Cryptography in Routing Protocol …………... 32

3.1 Introduction ……………………………………. 32

3.2 Background ……………………………………. 32

3.3 Cryptography Types …………………………....... 34

3.3.1 Symmetric Cryptography ………………… 34

3.3.2 Asymmetric Cryptography ………………... 36

3.3.2.1 Public key ……………………………. 38

3.3.2.2 Digital Signature ……………………... 39

3.3.2.3 Public key encryption schemes ……… 40

3.3.3 Digital certificates ………………………… 41

3.4 Hash Function ………………………………….. 43

3.5 Key Management ………………………………. 45

3.5.1 Key Exchange …………………………….. 47

3.5.2 Key Agreement and Group Keying ………. 48

3.6 Summary ……………………………………….. 48

Chapter 4: Security ……………………………………… 49

4.1 Security Examples ……………………………... 49

4.1.1 ARAN …………………………………….. 49

4.1.2 Ariadne …………………………………… 51

4.1.2.1 TESLA protocol ………………….. 52

4.2 Summary ……………………………………….. 55

Chapter 5: Elliptic Curve Cryptography in DSR

Routing Protocol ……………………………..... 56

5.1 Introduction …………………………………… 56

5.2 Elliptic Curve Cryptography (ECC) 58

5.2.1 Secure Routing Comparisons 60

5.2.1.1 ARIADNE 61

5.3 Diffie-Hellman Key Exchange 61

5.4 ECC Diffie-Hellman Key Exchange 62

5.5 ECDH Experiment in MANET 64

5.5.1 Goals 64

5.5.2 Simulation Setup 65

5.5.3 Parameters 65

5.5.4 Metrics 66

5.5.4.1 Packet Delivery Fraction (PDF) 67

5.5.4.2 Average End-to-End Delay (Delay) 67

5.5.4.3 Network Throughput 67

5.5.4.4 Normalized Routing Load 68

5.5.5 Performance Metrics 68

5.5.2.1 Analysis using Performance Metrics ... 68

5.5.6 Simulations Results 69

5.5.6.1 10 mobile nodes having 6 UDP links..... 69

5.5.6.2 10 mobile nodes having 6 TCP links...... 71

5.5.6.3 20 mobile nodes having 6 UDP links..... 73

5.5.6.4 20 mobile nodes having 6 TCP links...... 75

5.5.6.5 50 mobile nodes having 10 UDP links... 77

5.5.6.6 50 mobile nodes having 10 TCP links.... 79

5.6 Security handshake attacks ……………………… 80

5.6.1 Performance metrics……………………… 83

5.6.2 Simulation Results ………………………... 84

5.6.2.1 10 malicious mobile nodes …………... 85

5.7 Summary 87

Chapter 6: Trust Routing in MANET for securing DSR Protocol …………………………………... 89

6.1 Trust in Ad-hoc Network ………………........... 89

6.2 Trusts in routing protocols 90

6.2.1 Security aware routing (SAR) 90

6.2.2Trust-aware Routing Protocol (TARP) 91

6.3 Trust Calculation in Routing 92

6.4 Trust Route in DSR ……………………........... 94

6.5 Trust Models in MANETs ……………………… 100

6.5.1CONFIDANT 102

6.6 TV-ECDH Experiment in MANET ……………... 103

6.6.1 Goals 103

6.6.2 Simulation Setup 103

6.6.3 Parameters 104

6.6.4 Performance Results ……………………… 105

6.7 Simulation Results ………………………………. 105

6.7.1 Simulation results for 5 malicious nodes (low) ………………………………………. 105

6.7.2 Simulation results for 25 malicious nodes (high) ……………………………………… 109

6.8 Summary …………………................................... 112

Chapter 7: Conclusion and Directions for Further Research 114

7.1 Conclusion ……………………………........ 114

7.2 Future Work ……………………………......... 116

Bibliography ………………………………………………….. 117

CHAPTER 1: INTROUDCTION

1.1 Introduction

It is a fact that new technologies play important roles in our lives. The last ten years have seen a speedy growth in the use of wireless mobile technologies. We are increasingly using wireless links instead of traditional cable in our homes, offices and public places. This increase in use of such technologies poses a challenge in terms of security. Indeed, exchange of important information between devices such as PDA, mobile phones, laptops and tablets wirelessly is subject to intrusion from unidentified and untrusted network participants. This leads to the need for special security measures for wireless networks.

Security solutions are available for infrastructure based wireless networks such as (WiMAX) IEEE 802.16 and (WLAN) IEEE 802.11 [1]. This is not an issue for Mobile Ad Hoc Network (MANET) without infrastructure. We therefore have a security challenge that requires a secure communication design.

The definition of MANET is a collection of mobile nodes communicate wirelessly and moving dynamically for forming a provisional wireless network. This kind of network does not need any kind of infrastructure based network. The purpose of each mobile node is to work as a router to forward and receive packets from other nodes in the network.

MANET applications can be used in efficient and dynamic communication such as emergency operations, military and remote areas. MANET is based on special characteristics which are different from traditional infrastructure wireless based; for example, mobility, bandwidth limitations and variable ability links, energy limitations, and physical security limitations.

The Routing layer is one of the important layers in any MANET network. This layer is for all time required discovering the route to transmit packets correctly among origin and target mobile nodes. Designing a routing protocol in MANET is a major challenge. According to [2], routing protocols categorized into three groups: Proactive, Reactive and Hybrid protocols.

The Development of a good security solution is the second step in designing routing protocol. It is crucial to know the possible forms of attacks. In MANET, broadcasting wireless medium inherently signifies that attacks may come from any direction and in different network layers. A secure routing protocol has to provide these services: *availability, confidentiality, integrity, authentication* and *non-repudiation* [3].

An important kind of security is cryptography. This technique uses mathematicl science to encrypt or decrypt the data between two communications parties. By using the encryption techniques we can be sure that the information is transmitted securely. The technique works in the following manner: it encrypts a plain text to cipher text and transmits it to the target mobile node to decrypt cipher text to as it is plain text. There are three types of cryptography: Symmetric key (Secret key) - which uses one key for both encryption and decryption mechanisms; Asymmetric key (Public key) - which uses one key for encryption mechanism and the other key for decryption mechanism; and hash function which uses an arithmetic conversion to encrypt of the information. Cryptography can give confidentiality; Hash function can give integrity. In fact, none of these types of security will work without trust [4]. Trust model is a kind of mechanism to protect the routes from untrustable mobile nodes in the network. There are three kinds of trust models: direct trust, hierarchical trust and web of trust.

Elliptic Curve Cryptography (ECC) is defined as a process on the public key mechanism that depend on an algebraic formation of the elliptic curve over finite fields [5], or with large prime numbers. The main advantage of using ECC is using a smaller key size compared to other encryption techniques such RSA

1.2 Research Motivation

A Mobile ad hoc network works without any infrastructure base. The mobile nodes have to work as routes to communicate with each other. Routing layer such as DSR, AODV, DSDV "*Destination-Sequenced Distance-Vector*", OLSE, etc. is the most important for researchers. Routing security has to be trustworthy, speedy and secure during information exchange between mobile nodes in Mobile Ad Hoc Network. Security mechanisms such as Ariadne have been proposed [6, 7]. Ariadne relies only on TESLA (Timed Efficient Stream Loss-tolerant Authentication) which is a kind of symmetric cryptography. TESLA is an efficient authentication. However, it needs the receiver to buffer packets through one disclosure delay before it can authenticate them. Many attacks such as malicious attacks or selfish attacks try to disturb the network operations by modifying, dropping, altering, fabricating or injecting packets to consume the network resources.

Securing the channel among the source and the destination mobile nodes in the network from misbehaving attacks in an efficient manner is of paramount importance. This project proposes to solve this security issue in a way that saves power consumption and memory as well.

1.3 Objectives

This research aims to improve the integrity, availability, reliability, confidentiality, non-repudiation of Mobile Ad hoc Network (MANET) communications and the data in the upcoming future. Moreover, because of nodes mobility and changing environment, the proposed mechanisms have been designed to be scalable.

1.4 Contribution

As mentioned above, security is a major concern in routing protocols in MANET. Therefore, this kind of network is very vulnerable to attacks compared to wired networks. We are aiming to design an overall security rule by implementing the security requirements that predict, detect and solve the vulnerabilities.

To achieve the security requirements and targets and defeat any attacks, we need to have a set of efficient secure mechanisms. Our research shows that cryptography mechanisms are essential security management tools.

Our contribution to the research topic can be split into the following two areas:

1. Our proposal is to exchange keys by using Elliptic Curve Diffie-hellman (ECDH). This is because ECDH is perfect in flexibility to node capture, has excellent scalability, low memory and

bandwidth requirements, and low communications overhead. For example, an eavesdropper can know part of the key but he/she cannot compute the secret key. In addition, these public keys remain unchanged over network lifetime and could be used again for key exchange with different mobile nodes.

2. In our trust model stage is to evaluate the experience of the trust vector by monitoring the node participation, node forwarding packet and node dropping packets. Our scheme helps in achieving authentication with minimal overhead.

1.5 Thesis structure

This thesis consists of 6 chapters. This chapter is a short introduction about Mobile Ad Hoc Network (MANET).

Chapter 2 presents background information on Mobile Ad Hoc Network (MANET), by describing its characteristics, applications, routing protocols and security issues.

Chapter 3 presents the area of cryptography and includes a description of the different types of cryptography. It also describes the Hash function and Key Management. The chapter ends with a brief summary.

Chapter 4 focuses on the Elliptic Curve Cryptography in DSR Routing Protocol. The first two sections describe the Elliptic Curve Cryptography. In the third section, presents and analyzes the Diffie-Hellman Key Exchange. The fourth section focuses on the Elliptic Curve Cryptography Diffie-Hellman Key Exchange. Then moves on to describe the ECDH Experiment in MANET and discuss simulation results on self-created scenarios using performance metrics. Fifth section presents the security handshake attacks. Finally chapter ends with a short summary.

Chapter 5 discusses trust in ad-hoc networks. It analyses trust in routing protocols focusing specifically on security aware protocols and trust-aware routing protocols. It examines trust computation in routing and looks at a novel method of message security using trust table multi-path routine. The final part of this chapter consists of a short conclusion.

The last chapter of this thesis consists of a summary of the data presented in the first five chapters. The thesis ends with directions for further research.

1.6 Publications

Paper published in PGNet conference 11th Annual Postgraduate Symposium on the Convergence of Telecommunications, Networking and Broadcasting (2010)

1. S. Almotiri and I. Awan, Trust Routing in MANET for Securing DSR Routing Protocol, PGNet (2010)

CHAPTER 2: BACKGROUND

This chapter focuses on the Mobile Ad Hoc Network (MANET). By drawing on current research in the field, it aims to provide a concise picture of MANETs. In the first section, it describes the characteristics and applications of MANETs based on [2, 8-12]. In the next section, it discusses the routing protocols of MANET, following Das, Hu, Johnson, Macker, Misra, Perkins, Tseng, and Xing 2006. The third part of this chapter focuses on the security issues of MANET. This section is largely based on papers by [7, 13-26]. And it concludes with a brief summary.

2.1 Mobile Ad Hoc Network (MANET)

Mobile Ad Hoc Networks (MANETs) is defined as a set of mobile nodes working wirelessly and dynamically forming a provisional wireless network. This kind of wireless network does not rely on any network infrastructure unlike traditional mobile wireless network – for example, sensor networks, emergency rescue, etc. Node mobility may change the network topology frequently at unpredicted times. Without an infrastructure, nodes have to cooperate to provide the necessary network functionality. Every mobile node in the network has a dual function as (a) a host: to send and receive packets among mobile nodes in the network and (b) as a router: to do a route discovery and route maintenance. Transmission domain for a mobile node could detect other mobile node transmission domain with difficulty, as illustrated in Figure 2.1 [8]. There are many benefits to the “Mobile Ad Hoc architecture, such as self-reconfiguration and adaptability to highly variable mobile characteristics such as power and transmission conditions, traffic distributions, and load balancing” [27],2009, p.25) . Some of the possible ideas proposed to solve problems include “distributed MAC and dynamic routing, wireless service location protocol, wireless dynamic host configuration protocol, distributed admission call control, and quality-of-service (QoS)" [27] (2009, p.25), based on routing mechanisms [12].

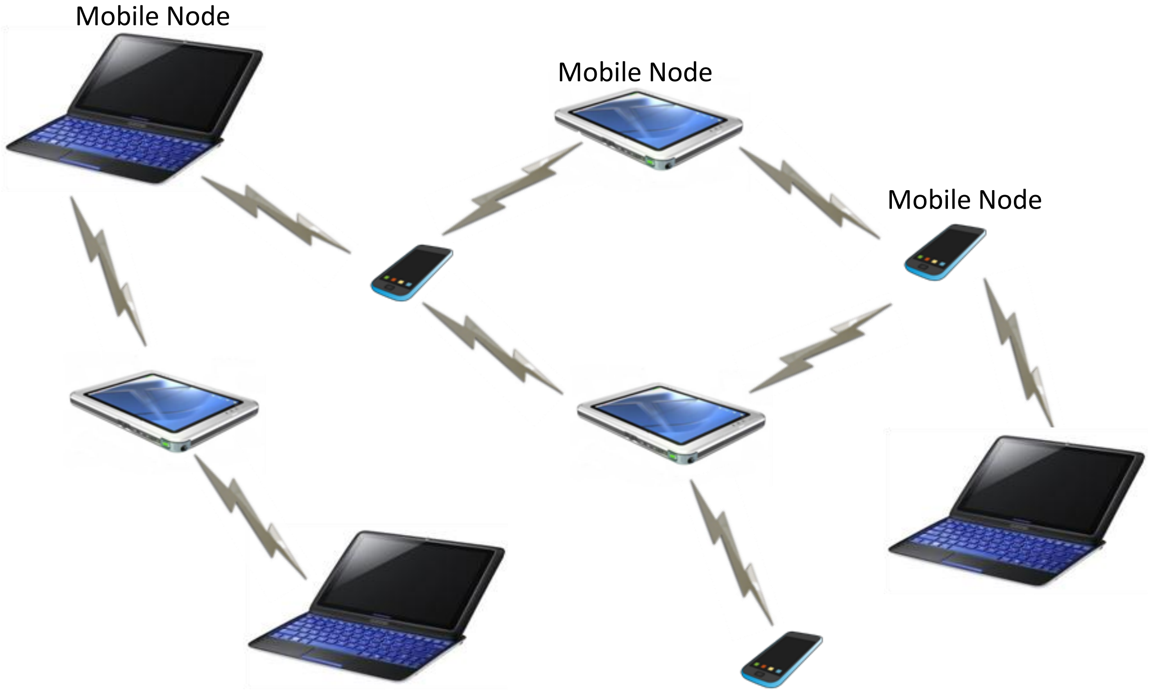


Figure 2.1: Mobile Ad Hoc Network (MANET)

2.1.1 MANET Applications

Some MANET examples are used in emergency or disaster operations, military networks and remote area. In a particular situation of any mobile ad hoc network applications cannot be based on central well-organized communication. Another application of MANET is to conduct field studies in remote locations. Lately, many researchers have been including the use of MANET applications in their vehicles to make it efficiently available with the vehicular environment. For example, distributing recent traffic information to all vehicles including traffic accidents [18].

2.1.2 Characteristics and challenges of MANET

There are two categories of wireless networks: (a) infrastructure networks and (b) wireless ad hoc networks. Entire wireless network infrastructure works with the variety of networks supporting tools such as a base station and wireless access points. Wireless ad hoc networks are therefore different from wireless infrastructure based networks. According to [2], these networks have several important characteristics:. The paragraphs below, based on [2] (1999, p.3-4), further explain these characteristics:

1. *Dynamic topology (Mobility)*: Mobile nodes are ultimate to move around expeditiously. Consequently, the topology of a network, that classically multi hops, can be modified many times at random and rapidly, and could contain both one way and two ways connections.
2. *Bandwidth limitations and variable ability links*: Wireless links have considerable minimum loads than other similar versions. Beside, the actual throughput of wireless is frequently lower than a radio’s ultimate transmission average even if we include the effects of multiple accesses, waning, and the noise. Only one impact for the low to the average connections abilities is that congestion is the ordinary standard rather than the exception, that is, total application demand will possibly approach or exceed the network ability regularly. Mobile ad hoc users have to request services as in based network infrastructure. This is due to the fact it is an extension of based network infrastructure. Those requests

usually will rise as multi-media computing and cooperative networking applications rise.

3. *Energy limitation*: The foundation in such network is power or other inexhaustible ways for their energy. Saving power is the essential design in a MANET system.
4. *Physical security limitation*: In general, Mobile ad hoc networks are additionally prone to threats in physical security than wired network. That will increase the chances for some kinds of attacks such as *eavesdropping*, *denial of service* and *spoofing* and these should be studied. To decrease the security threats in wireless networks, link security mechanisms are functional. In MANET which the no infrastructure based network has extra force to the single point of failure in the infrastructure based network.

2.2 Routing Protocols in MANET

A routing process is constantly required to discover a path to send the packets correctly between the initiator and the target mobile nodes (Figure 2.2) because (a) in MANETs, there is no infrastructure support as is the case with wireless networks and (b) a target mobile node could be outside the initiator area.

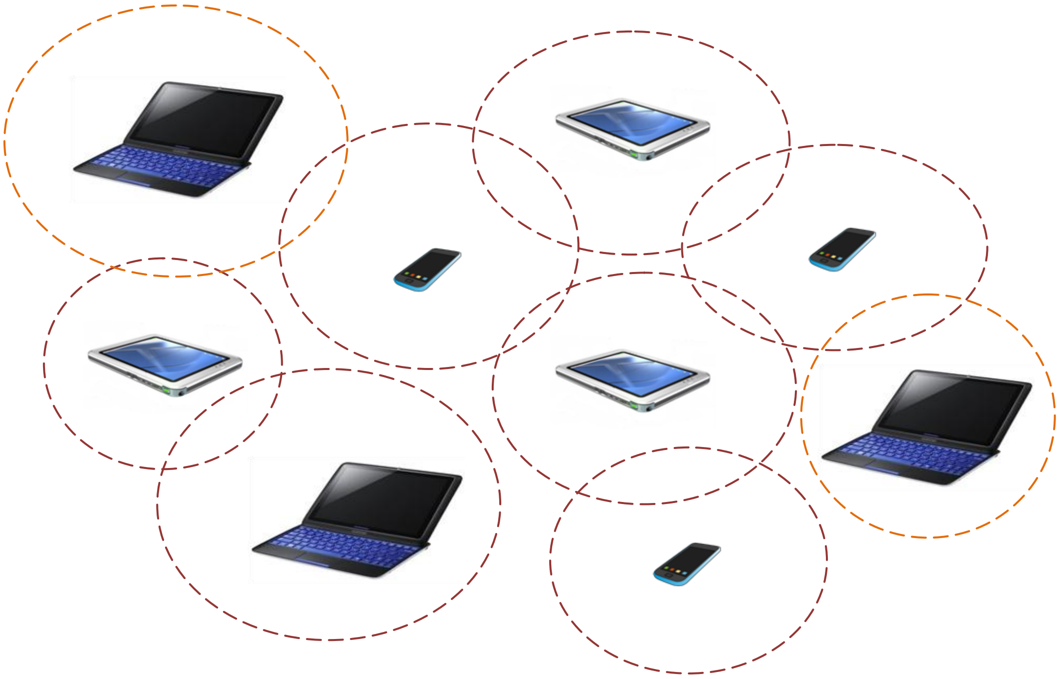


Figure 2.2: Mobile Ad Hoc routing range coverage

2.2.1 Challenges in Routing

The design of a routing protocol for MANET is a major challenge [2, 13]. Moreover, determining a packet route requires a node. This node needs to know at least the availability information to its neighbours. In this section, we will illustrate some of the challenges:

1. Distributed network: MANET is a distributed wireless network with no infrastructure, implying that decentralized authority is required to maintain the status of the mobile nodes.
2. Dynamic topology*:* The nodes are mobile and therefore the network is self-organizing. This can lead to changes in the topology of the wireless network over time. Therefore, routing protocols designed for such wireless networks must also be adaptive to the topology changes.

3. Power efficient*:* Since the mobile nodes in an ad hoc network usually run on portable batteries and are deployed in aggressive terrains, they have stringent power requirements. Hence, the protocols must be designed to save battery time.
4. Network size: The ability to enable commercial applications such as voice transmission in conference, meetings, etc., is an attractive feature of ad hoc networks. However, the delay concernes in the underlying protocols places a strict upper bound on the size of the network.
5. Security: Security in MANET is extremely necessary in such scenarios such as a battlefield. The five objectives of security are: *availability*, *confidentiality*, *integrity authenticity* and *non-repudiation* [24] - are hard to obtain in MANET, because every mobile node in the network participates evenly in routing packets.

2.2.2 Classification of Routing Protocols

We can generally divide routing protocols in MANET into three groups [2] (Figure 2.3). The following paragraphs are based on [13]

1. Proactive: Each mobile node in the wireless network preserves a complete routing information for every mobile node in the wireless network by periodically updating the routing table even before it is needed [28]. Consequently, no delay to discover a path throughout the wireless network when a mobile node needs to transmit data packets. This type of routing protocol almost works similar way as routing protocols for wired networks. The thing is proactive protocols are not appropriate for a large network, as they need to maintaining mobile node entries for every mobile node in the routing table. That is a reason for causing more overhead in the routing table which leads to consumption of the bandwidth.

2. Reactive: Such a kind of routing, a mobile node maintains routes to send information to the target mobile node. These routes will expire after certain time if there is no communication between initiator and target mobile nodes. This protocol will look for a route if a mobile node needs to send information to another.

3. Hybrid: Such a kind of routing protocols merges the features from the two types that have been explained above. Mobile nodes will be based on a specific geographical area or within a range from a concerned mobile node in the routing area and use proactive protocols. The communication among these mobile nodes in different areas will depend on reactive or proactive routing protocols.

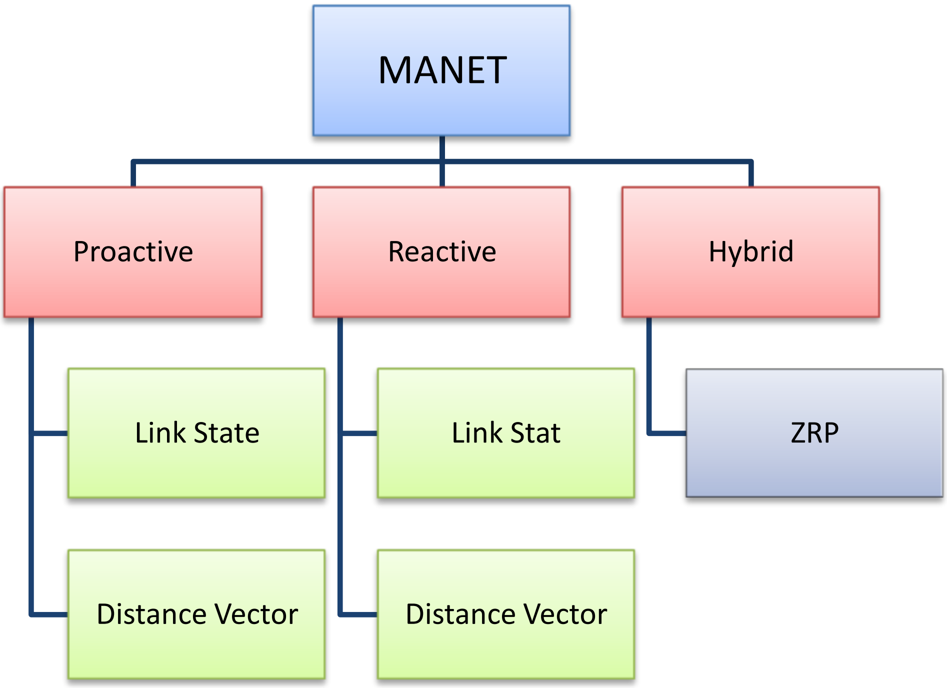


Figure 2.3 Categorization of Mobile Ad hoc Routing Protocols

The remainders of section 2.2 presents survey of the most two popular routing protocols used in (MANET): Dynamic Source Routing protocol (DSR) [29, 30] and Ad hoc On-demand Distance Vector routing protocol (AODV) [31].

2.2.3 Ad hoc On-demand Distance Vector (AODV):

The AODV adopts different approaches on the basis of a function. To find routes, the AODV routing protocol [31] based on a reactive routing protocol. Beside that it uses a proactive routing if the AODV wants to discover the latest route. In order to find the latest routes it uses the route discovery and destination sequence numbers. The Route Discovery and Route Maintenance of AODV are described below, as discussed in Das (2003).

Route Discovery

In this step, Route Request (RREQ) packet is transferred by the initiator mobile node. According to [31], the fields in Route Request packet are *source identifier* (SId), *destination identifier* (DId), *source sequence number* (SSeq), *destination sequence number* (DSeq), *broadcast identifier* (BId), *Time To Live* (TTL). When the intermediate mobile node received the RREQ packet, it has two possibilities: (a) broadcast the RREQ packet to the others if the intermediate mobile node didn’t have the route to the destination or (b) return to the source mobile node a Route Reply (RREP) packet if there is up to date route to the destination mobile node in its own cache. By using the (Sid) and (BId) as a pair, it will check if a specific RREQ has been received or not. This is done to prevent duplicates. While broadcasting the RREQ packet, each intermediate mobile node accesses the prior node’s address and its (BId). The node also keeps a timer connected with each access. This is done as a try to delete a RREQ packet as long as the RREP has not been received before RREQ packet expired.

[31] shows the Route Replay (RREP) packet stored the information about the previous mobile node once it has been received to transmit the packet to the next hop of the destination mobile node. By doing this, every mobile node includes just the next hop information; where in fact in the source routing, every intermediate mobile node on the route to the destination is stored.

Figure 2.4 shows a case of route discovery in AODV [31]. In the paragraph below, it explains how this mechanism works (for a more detailed explanation, refer to [31]).

Assume that mobile node called A needs to transmit a packet to mobile node called G but it does not have a route in his own cache. Consequently, G will start a route discovery mechanism by broadcasting (RREQ) packets to all neighbours next to mobile node A, which are in this case B, C and D.

All fields which described above are added in the Route Request (RREQ) packet. When RREQ packet arrives to mobile nodes B, C and D, these mobile nodes directly look to their route's caches for any available route. If there is no available route, then they will broadcast the RREQ packet to their neighbours; alternatively, there will be a comparison in the RREQ packet and the route cache for the DSeq. It returns toward the source mobile node a route replay (RREP) packet with the route to the destination if the DSeq in the RREQ packet is higher. As shown in Figure 2.4, mobile node C has a route to mobile node G in its own route cache and its DSeq is higher comparing with the RREQ packet. Accordingly, it transmits the RREP backwards to the source mobile node A. By achieving that, mobile node A has been stored the route (A-C-F-G). A RREP packet is sent backward from the destination mobile node to the source as well. There will be an update in the intermediate mobile node's routing table, which is in the route between the source and the destination mobile nodes, with the most recent DSeq in RREP packet.

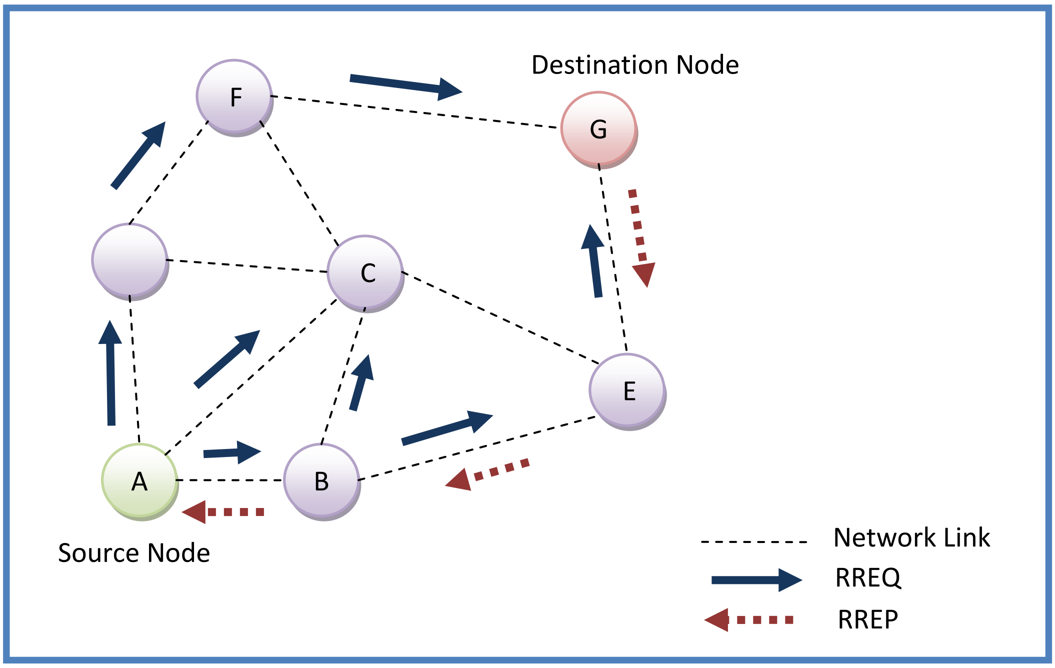


Figure 2.4: Route Discovery in AODV

Route Maintenance

Route Maintenance is illustrated in Figure 2.5. As described by [32], at any time a mobile node detect a link disconnect through the link layer acknowledgements or HELLO messages, it will notify the source and ends mobile nodes by broadcasting a Route Error (RERR) packet. As shown in the figure 2.5 below, A RERR packets will be transmitted if there is disconnect between mobile nodes C and F on the route A-C-F-G by them to inform the source and the destination mobile nodes.

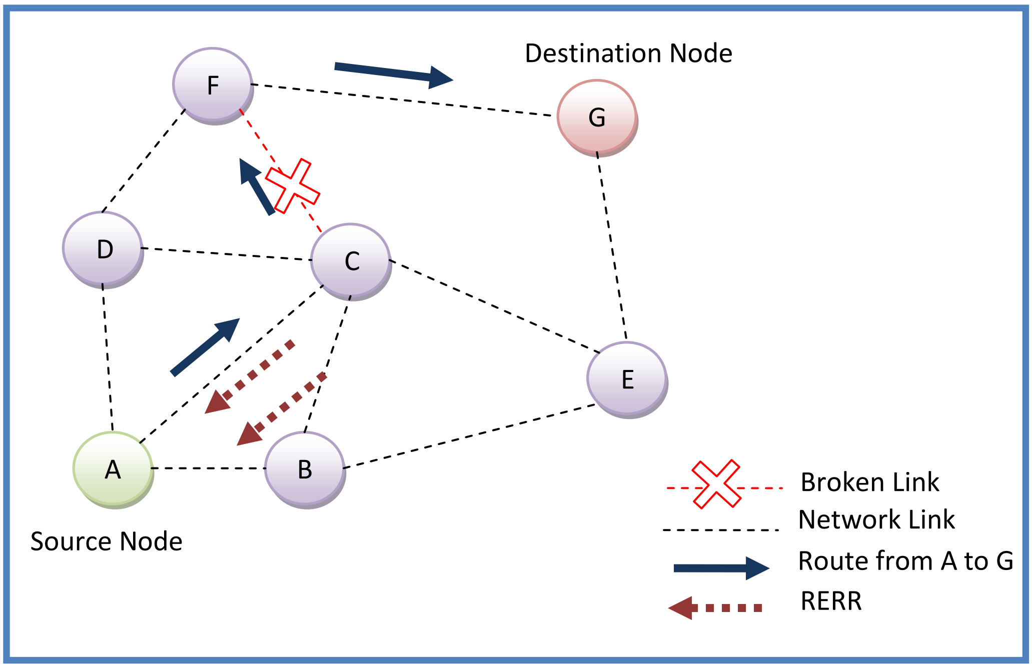


Figure 2.5: Route Maintenance in AODV

2.2.4 Dynamic Source Routing protocol (DSR)

Dynamic Source Routing Protocol [33] is an on-demand routing built on the idea of *source routing*. Johnson (2007: pp. 9-16) describes the DSR in details. The DSR works as follows. In the source routing, a sender mobile node has the complete route in the packet header that the packet must drive to get to the destination mobile node. Specifically, each mobile node in the route only sends the packet to its next hop, which has been allocated in the header. This happens with any check of its routing table while in table driven routing protocols. Moreover, the mobile node does not have to regularly transmit their routing table to the neighbouring mobile nodes. This will save lots of wireless network bandwidth. The two parts of the DSR procedure are described below, as in [33]:

Route Discovery

The source mobile node looks for a route to the destination mobile node via transmitting (RREQ) packets to the all neighbours. Every neighbour who has received that packet will transmit to other neighbour, if RREQ has not been received before, if the Time to Live (TTL) field counter is higher than 0, else if it is not the destination mobile node of that RREQ packet, or if it has not been transmitted the RREQ packet to its neighbours. In addition, the mobile node can know if it has been received a specific (RREQ) packet before, by using *Request ids.* Every mobile node will update its table of RREQ packets which has been received recently. The table of RREQs include the *initiator* and *request id* fields. If the mobile node received two RREQs with same *initiator* and *request id*, then it will transmit just the first RREQ and reject the other. Furthermore, this technique prevents any routing loops in the wireless network. As soon as RREQ packet arrives to the destination mobile node it will transmit back a (RREP) packet to the same route it has been come from which include the traced route to the destination mobile node. An example has been shown in Figure 2.6. If mobile node S needs to transmit information to mobile node D, it will use the route discovery procedure and transmits a (RREQ) packet to its neighbours C, E, and A. In this example Mobile node E can receives more than RREQ packets from C and A. By the Route Discovery procedure it will drop both of the packets which have been received from mobile node S earlier. As soon as mobile node D received the RREQ packet, it will insert its address and the traced route, and then will send back (RREP) packet on the same route to the source mobile node.

The destination mobile node sends the best route to the source mobile node (the first route that has been received) and saves the other routes in the route cache the other routes to be used in the future. A *route cache* is present in each mobile node and it is updated regularly. Thus, at any time a mobile node receives a (RREQ) and finds a route to the destination mobile node in its own cache, it will send back a (RREP) packet to the source mobile node without broadcasting it further.

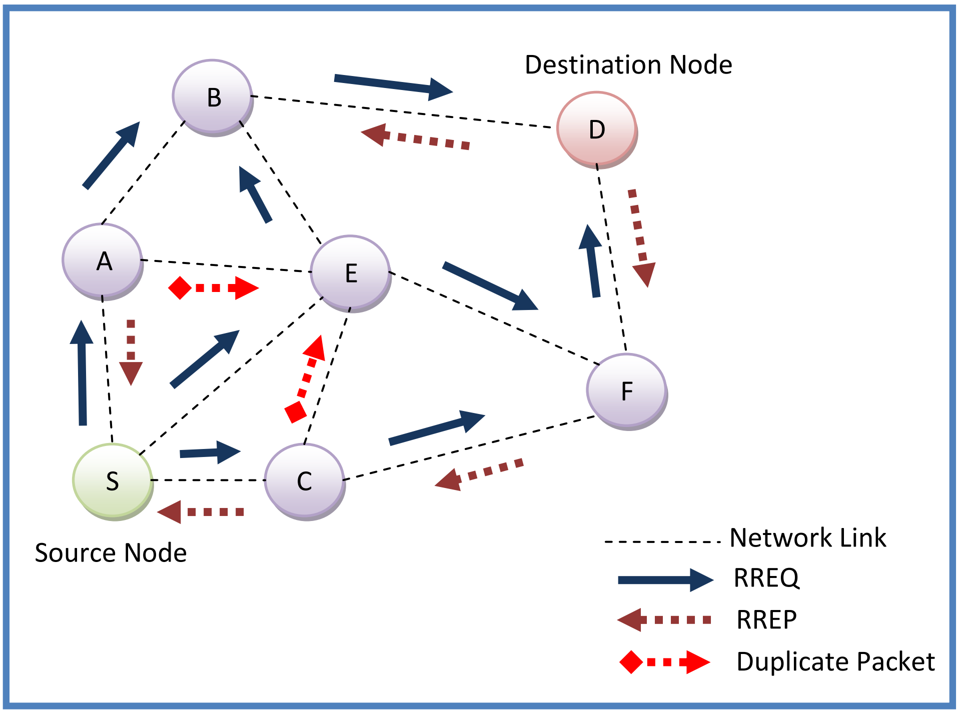


Figure 2.6: Route Discover in DSR

Route Maintenance

The route maintenance is achieved as soon as there is a broken link among two nodes. According to Reference [33], there are two ways by which a node can find a broken link: (a) by negative controller the link or; (b) by active controller the link. The example in Figure 2.7 shows a route error (RERR) sent back from mobile node B to notify the source mobile node S that the link (B-D) is broken. Thereafter, the source mobile node will restart the route discovery procedure to look for a new route. In addition, it eliminates any route it might have in its own cache to such destination.

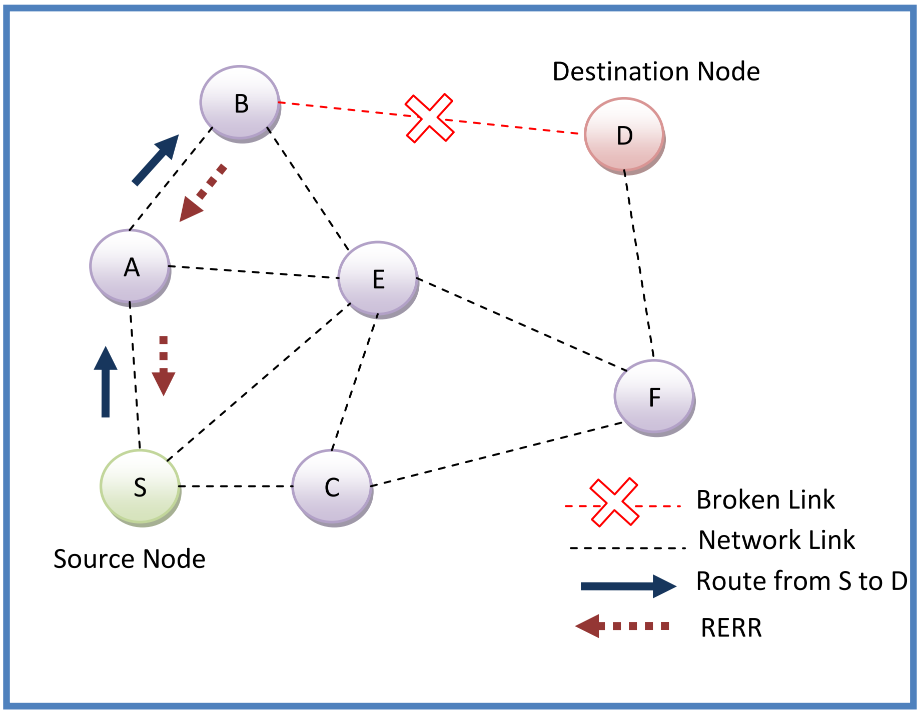


Figure 2.7: Route Maintenance in DSR

Dynamic source Routing (DSR) protocol gets important advantages from the source routing because the intermediate mobile nodes do not require: (a) to keep updating the route information to direct the packets that has been received and (b) to have every routing advertisement packets. But, a problem arises because the network size increases. This is because the routing overhead indeed rises as every packet has to take the whole route along with it to the destination. The authors in [34] suggest that one way of decreasing the propagation delay is through the use of route caches. In addition, as long as there is a link broken the (RERR) packet broadcasts immediately to the source mobile node, in order to start a route discovery procedure. Some researchers proposed some improvements to DSR, such as *non- propagating* route request, which will avoid from re-broadcasting, and *gratuitous* route replies, once the mobile node find a packet with its own address been in the header, that mobile node will transmit a RREP to the source mobile node by passing the previous hops [30].

2.3 Security in MANET

To develop good security solutions, we should know the possible form of attacks. In MANET, broadcasting wireless medium naturally indicates that attacks possibly will come from any route and from different layers. One of the main security obstructions in MANET is the absence of fixed infrastructure support, which makes it impossible to use existing trusted nodes.

2.3.1 Security objectives:

It is crucial to secure the routing protocols in MANETs. Different security services have been looked at: *availability, confidentiality, integrity, authentication* and *non-repudiation* [3, 14, 15]. These are discussed below, based on [16],

Availability Assure the permanence of the services of the network in spite of attacks. A *Denial-of-Service* (DoS) is a potential threat at any layer of a mobile ad hoc network. On the media access control (MAC) layer, an attacker could jam the physical communication channels. On the network layer disruption of the routing operation may result in a partition of the network, rendering certain nodes inaccessible. On higher levels, an attacker could minimize high-level services such as key management service.

Confidentiality guarantees that particular information is not at all revealed to unauthorized nodes. It is of paramount importance to strategic or tactical military communications. Routing information has to stay also confidential in some situations, because the information may be valuable for attackers to trace their targets.

Integrity ensures that a message that is on the way to the destination is never corrupted. Channel noise or malicious attacks on the network could corrupt the message.

Authentication is to ensure the peer mobile node's identity. With no authentication, any attacker might masquerade as a normal mobile node, so they going to get a benefit of accessing to the sensitive information.

Non-repudiation guarantees that the initiator of any message cannot deny that it is the real initiator. Non-repudiation is important for discovering and removing of compromised mobile nodes.

Networking environment in wireless schemes makes the routing protocols vulnerable to attacks. Such attacks are varied and can variety from passive attacks like eavesdropping to active attacks such as impersonation, message replay, message dropping, network partitioning, etc... Eavesdropping is a threat to confidentiality and active attacks are threats to *availability, integrity, authentication* and *non-repudiation*. Mobile nodes roaming in a mobile ad hoc environment with bad physical security are quite vulnerable and they may be compromised. Compromised mobile nodes can next be used as starting points to initiate attacks against the routing protocols.

2.3.2 Attacks

There are various ways of categorizing attacks in the MANETs – for example, *passive* and *active* attacks which focus on the behavior of the attacks themselves; *external* and *internal* attacks which focus on the source of the attacks; *mobile* and *wired* attacks which look on the processing capabilities of the attackers and finally, *single* and *multiple* attacks which relate to the number of the attackers. In this section, it focuses on passive and active attacks. These are the main problems for MANETs.

2.3.2.1 Passive Attacks

Passive attacks "are launched to steal valuable information in the targeted networks and to detect such attack is difficult because neither the system resources nor the critical network functions are physically affected to prove the intrusions" [17]2004, p.149). Eavesdropping and traffic analysis attacks are examples of such attacks.

2.3.2.1.1 Eavesdropping

Eavesdropping attack is a method of collecting information by spying on broadcasted data on authentic network [19]. Although eavesdrop privately overhears the communication, the information remains intact. However, privacy is endangering. This kind of attack is very easy for the malicious mobile node to carry out as it relates to the tradition wired network. Eavesdropping attack in mobile ad hoc network works in the following way [18]:

(a) It shares the wireless medium by working in the promiscuous mode. This allows a network device to interrupt and read every packet in the network that received.

(b) The attacker mobile node interrupts the communication. This can easily be achieved since each MANET node is transceiver fitted in the communication range. A malicious mobile node can decode the information to aim the authorized mobile node on the wireless network.

(c) The malicious mobile node can then remain the sensitive information, alter the route or modify the routing table with wrong information.

Such process can pose a serious threat to the wireless network resource and decrease the performance of the wireless network.

2.3.2.1.2 Traffic Analysis

Traffic analysis remains a serious but subtle security attack. Traffic analysis works in the following way: adversaries attempt to discover the identities of the parties in the communication. Once they have achieved this, they can analyze the traffic to find out the wireless network traffic style and route the changes in the traffic style [20]. Such an attack involving leakage of information can have serious implications in security sensitive scenarios.

2.3.2.2 Active Attacks

Active attack, as discussed in [21], involves information interception, alteration, or fabrication, in this way will disrupt the standard operations of a MANET. Figure 2.8

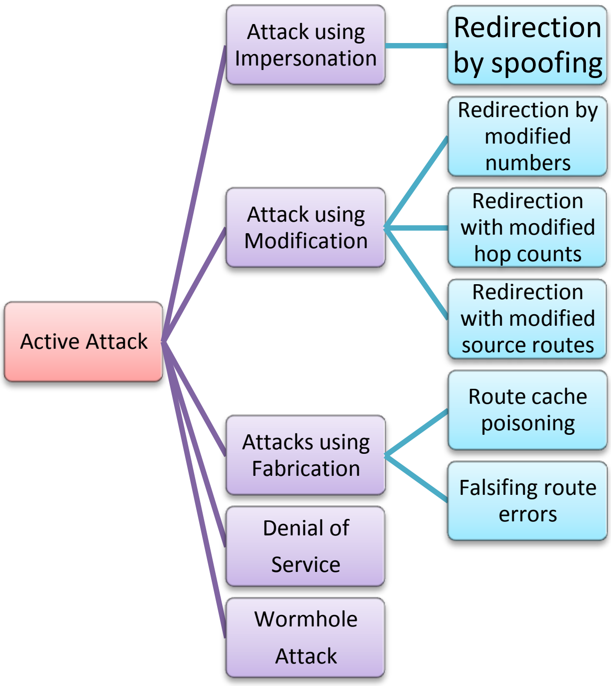

Active Attack
Attack using Impersonation
Redirection by spoofing
Attack using Modification
Redirection by modified numbers
Redirection with modified hop counts
Redirection with modified source routes
Attacks using Fabrication
Route cache poisoning
Falsifing route errors
Denial of Service
Wormhole Attack

Figure 2.8: Classification of active attacks on MANET routing protocols

2.3.2.2.1 Impersonation Attacks

This type of attacks violates authenticity and confidentiality in a network. A malicious node can impersonate or spoof the address of another node to modify the image of the network topology as perceived by another node. An example is described in Figure 2.9 below [22].

Mobile node S needs to send a data to mobile node D and starts the Route Discovery procedure. The malicious mobile node M, closer to node S than node D, impersonates node D as D'. It forwards a (RREP) to mobile node S. Without checking the authenticity of the RREP, mobile node S accepts the route which is in the RREP packet and starts to send data to the malicious node. This type of attacks can cause a routing loop within the network.

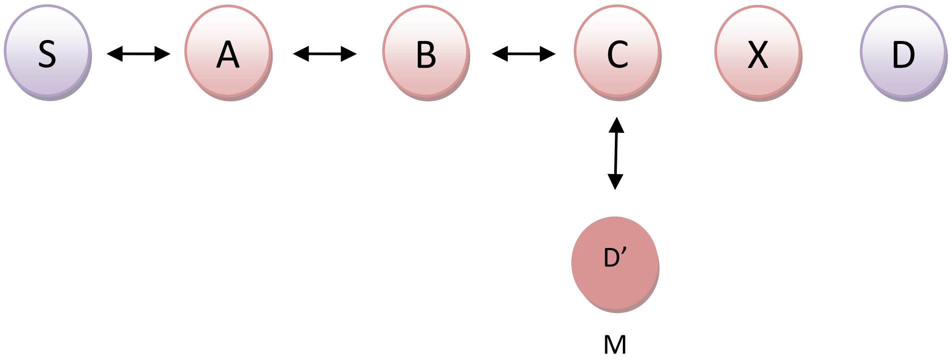


Figure 2.9: Example of impersonation attack

2.3.2.2.2 Modification Attacks

In this type of attacks, some of the protocol fields of the messages passed between the mobile nodes are modified, thereby resulting in traffic subversion, or redirection attacks. The following sections discuss some of these attacks:

1. *Modification of route sequence numbers*: This attack is possible against the AODV protocol. The malicious mobile node can alter the sequence number in the RREQ packets or RREP packets in order to make the route fresh. In Figure 2.10, malicious mobile node M receives the route request RREQ from mobile node B that originates from mobile node S and is prepared for mobile node D. M forwards a RREP to B with a large destination sequence number for D than the value last announced by D. The mobile node S accepts the RREP and then sends the data to D through M. When the legitimate RREP from D gets to S, if the destination number is less than the one announced by M, it will be discarded as a stale route. The situation will not be corrected until a valid RREP with higher sequence number than that of M gets to S.

2. *Modification of hop count:* This type of attacks is possible against the AODV protocol where a malicious mobile node can increase the chance that it is included in a recently generated route by resetting the hop count's field of a RREQ packet to a lower number or even zero. Similar to route modification attack with sequence number, the hop count's field in the routing packets is modified to attract data traffic.
3. *Modification of source route*: This attack is possible against DSR which uses source routes and works as illustrated in Figure 2.10. In Figure 2.10, it is assumed that the shortest path exists from S to D. It is also assumed that mobile nodes C and D cannot listen to each other, mobile nodes B and C cannot listen to each other, and M is a malicious mobile node trying the *denial-of-service* attack. Assume S forwards a data packet to D with the route (S-A-B-C-X-D). If M intercepts this packet, it removes X from the list and forwards it to C. C will attempt to forward this packet to D which is not possible since C cannot listen to D. Thus M has successfully launched a DoS attack on D.

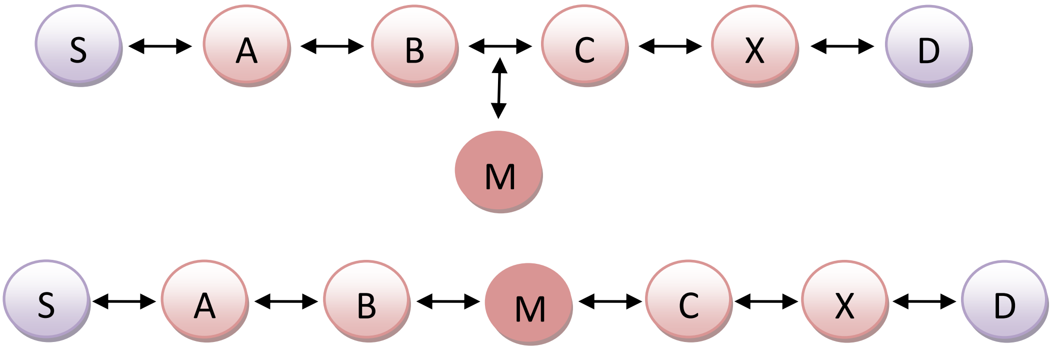


Figure 2.10: Example of route sequence number modification attack

2.3.2.2.3 Fabrication Attacks

In this type of attacks, a malicious mobile node attempts to inject fake messages or routing packets to disrupt the routing mechanism. These attacks are difficult to detect in a MANET since the routing packets appear to be legitimate packets to the nodes processing them. Attacks by fabrication are discussed in [22, 23]. Figure 2.11 exemplifies this type of attacks (Huang and Lee 2004). Mobile node S needs to send data packet to mobile node D. So it broadcasts RREQs in order to discover the route to mobile node D. Malicious mobile node M pretends to have a cached route to the destination D, as well as backward RREP to the source mobile node S. The source S, without checking the validity of the RREP, accepts the RREP and starts to send data through M. Furthermore, malicious nodes can fabricate RERR to advertise a link break to a certain node in a MANET with AODV or DSR protocols.

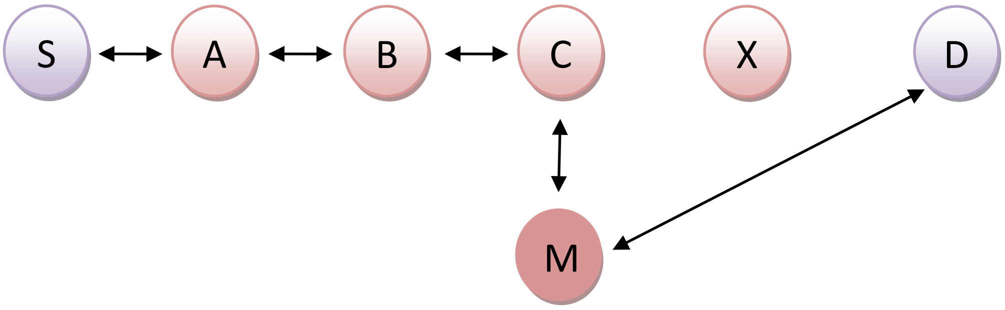


Figure 2.11: Example of a fabrication attack

2.3.2.2.4 Denial of Service

According to [24], *Denial-of-service* (DoS) attacks might be appeared from different layers. Denial of Service is a kind of attack in which no access to the system(s) is gained, but rather availability to network services is affected.

For example, suppose that mobile node D listen to mobile node C, and mobile node B listen to mobile node M. Source mobile node S sends data for destination D using the route (S-A-M-B-C-D). Mobile node A forward packets to mobile node M, and mobile node M alter the route (S-A-M-B-C-D) in (S-A-M-B-D), Mobile node B will send backward a route Error RRER (broken link) to source mobile node S. As a result, malicious mobile node M will drop the route error RRER packet. As shown in figure 2.12.

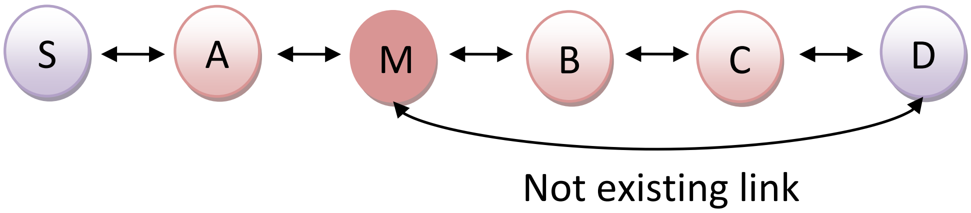


Figure 2.12: Example of Denial of service attack

2.3.2.2.5 Wormhole Attack

The wormhole attack [25] is a severe type of attacks in which two malicious nodes can forward packets through a private "*tunnel*" in the network as shown in Figure 2.13.

For example, M and N are two malicious mobile nodes which link through a private connection. Each packet that M receives from the wireless network is forwarded through "*wormhole*" to mobile node N, and vice versa. This kind of attack will interrupt the routing protocols via short looping to the normal flow of routing packets. Such a type of attack is difficult to detect in a network, and may severely damage communication among the nodes. Such an attack can be prevented by using *packet leashes* [25], which authenticate the timing information in the packets to detect faked packets in the network.

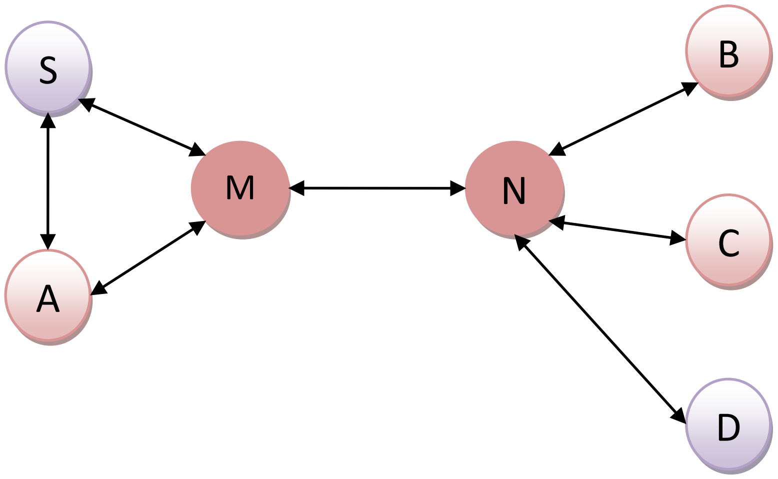


Figure 2.13: Example of wormhole attack

2.3.3 Cryptography

Encryption requires two communication parties to possess two objects; firstly an encryption algorithm and secondly an encryption key. Encryption algorithms are generally publicly known and available for security, meaning a non-secret agreement can be made as to the algorithm to use. This leaves the encryption key which either must remain secret when exchanged or a method must be used where publicly disclosing the key does not allow an attacker to decrypt messages that are encrypted with the key. A full encryption key management system (KMS) will perform more than mere key exchange. It will allow for creation, distribution, updating if necessary and destruction of the key. Further description details are analysed in chapter 4.

2.3.4 Trust Model

Trust can be defined as confident dependence between mobile nodes in the network. There are three types of trust models in MANETs: a) Reputation trust model, b) Recommendation trust model, and c) combination of Direct and recommendation trust model. Further description about trust, trust calculations and trust model are analyzed in Chapter 6.

2.4 Summary

This chapter provides an overview of the MANETs which are formed due to the absence of any infrastructure and work wirelessly and moving freely to perform a provisional wireless communication among the mobile nodes. Furthermore, it explains the characters and the challenges that affect MANET and gives a brief description of the routing protocols and their challenges, focusing on two routing protocols AODV and DSR. This chapter also discusses security issues in MANETs. It classifies the attacks that are possible against the existing routing protocols. An understanding of these attacks and their impacts on the routing mechanism will help researchers in designing secure routing protocols. Cryptography is a main factor to protect and secure the important information among the parties.

CHAPTER 3: CRYPTOGRAPGHY IN ROUTING PROTOCOLS

This chapter focuses on cryptography. After a short introduction and some background information on the topic, It discusses the different types of cryptography analysis following [35-47]. The fourth section, based on [48] and [49], describes the Hash function. In Section 3.5, it discusses Key Management using materials from [50]. The chapter ends with a brief summary.

3.1 Introduction

Cryptography is a technique to store and send information among whose can read and treat. This could be described as a scientific way to save the data by encrypting it to make it difficult to read for later decryption only by authorized parties. [35] (2003, p.659) argues that “Cryptography is an effective way of protecting sensitive information as it is stored on media or transmitted through untrusted network communication paths”. According to [51], cryptography techniques also enable secure communication in both wired and wireless networks. Symmetric key cryptography has computation efficiency, but it also has weaknesses in the management of secret key. Asymmetric key cryptography is widely used because of its simplicity in key distribution. Nevertheless, this technique depends on a centralized infrastructure and is resource expensive.

3.2 Background

According to [52] (1996, p.2) and [35] (2003, p.669), cryptography has a number of objectives. These are:

1. ***Confidentiality***: It stops all but those allowed to have access to the content of the data. Confidentiality can be achieved through various means. These range from physical to mathematical methods. [35] (2003, p.669) gives the following example to illustrate this point. Suppose the data to be secured, for instance the key, is stored in an accessible place only by allowed users. By using the encryption the data can be encrypted to only who has the appropriate keys and also identify the encryption algorithm. Only this user is capable to decrypt the message. Encryption can be categorising as: Symmetric key (Section 3.3.1) and Asymmetric key (Section 3.3.2).
2. ***Integrity***: It stops not permitted users from modifying the messages. Data integrity can only be guaranteed through the detection of all unauthorized manipulation of information, such as elimination or injection. To achieve the data integrity there is one cryptographic approach which is digital signatures (Section 3.3.2.2) and hash functions (Section 3.4).
3. ***Authentication***: This authentication is categorized into two types: *entity authentication* and *data origin authentication* [36]. In the first type, a user accessing any communication session has to identify himself to the other users. In the second type, any data to be sent has to be authenticated as regards to its data content or time of sending.
4. ***Non-repudiation***: It stops any entity from rejecting any past actions. The other meaning is a process whereby the sender who sends the data is given proof of delivery. With regards to the receiver, he/she is made aware of the identity of the sender; therefore neither can later reject to have processed the information. For instance, if there is a user who affords electronic order should not be able to deny it later. One of the cryptographic approaches to achieve non-repudiation is digital signatures (Section 3.3.2.2).

5. ***Availability***: The service should be available all the time. It must be robust for two reasons: (a) to withstand network failures and (b) to resist (DoS) attacks.

In cryptography, unique data transmitted from one user to another is called plain text [36]. This plain text is transformed to cipher text by encryption's procedure which is a particular algorithm or function. The authentic receiver can decrypt (decode) the cipher text which reveres into plain text by decryption's procedure. Mathematically, if M appears as a plain text of a message and C appears as a cipher text of a message, subsequently:

$$\text{Encryption} :: E(M) = C \quad \text{and} \quad \text{Decryption} :: D(M) = M,$$

Encryption and decryption procedures are administrated by keys. These keys are small size of data used by cryptography algorithms. In addition, these keys have to be saved to guarantee the security of the system. This kind of keys is called secret key. Additionally, the key management is a term used to refer to the secure administration of cryptography keys.

Cryptography algorithms are of two main types: (a) *symmetric key* that apply one key for encryption and also for decryption, and (b) *asymmetric key* that used two distinct keys for encryption and decryption [35]. In the following sections, I discuss these two Cryptography algorithms, *digital signature*, *digital certificates*, *Public Key Infrastructure* (PKI) and *Web of Trust* (WoT) models.

3.3 Cryptography Types

3.3.1 Symmetric Cryptography

In traditional cryptography, symmetric key algorithms are based on the use of the shared key - by both parties, that is, the sender and the receiver.– This has been arranged previously for exchange between them, for instance, via a secure communication channel [35].

According to [35], this shared key, which has been exchanged, is used for encryption and decryption. In the symmetric key, the sender and the receiver apply a private key (K) to encrypt and decrypt. Symmetric encryption, as discussed by [36], is illustrated in Figure 3.1. [36] describes this process in the following way: there are two users (called Alice and Bob). User Alice can encrypt the plain text ***m*** by the shared key ***k*** and transform it to a cipher text ***c***. User Bob can decrypt the cipher text ***c*** by the same shared key ***k*** that has been used for encryption and re-transform it to plain text ***m***.

Symmetric key algorithms are of two kinds: (a) *stream ciphers* and (b) *block ciphers*. These differ in the way in which they encrypt the messages. “Block ciphers operate on blocks of plain text and cipher text – usually of 64 bits but sometime longer. Stream ciphers operate on stream of plain text and cipher text one bit or byte (sometimes even one 32 bit word) at a time” [37]1996, p.189). Nechvatal et al. (2000) have reached the 128-bit block cipher and it is a standard of AES (Advanced Encryption Standard) algorithm which has been officially agreed by NIST (National Institute of Standards and Technology) in October 2000 [53].

[35] explains how symmetric key algorithms are efficient in that they can usually execute electronically quickly. But they also have certain limitations since they require a private key to be shared among the sender and receiver. When a connection between mobile nodes needs to be recognized, all of the sender and the receiver pair should share a key. This, in turn, creates a non-scalable system. Use of the same key among

more than two mobile nodes will result in a breach of security. This, in turn, makes the whole system vulnerable.

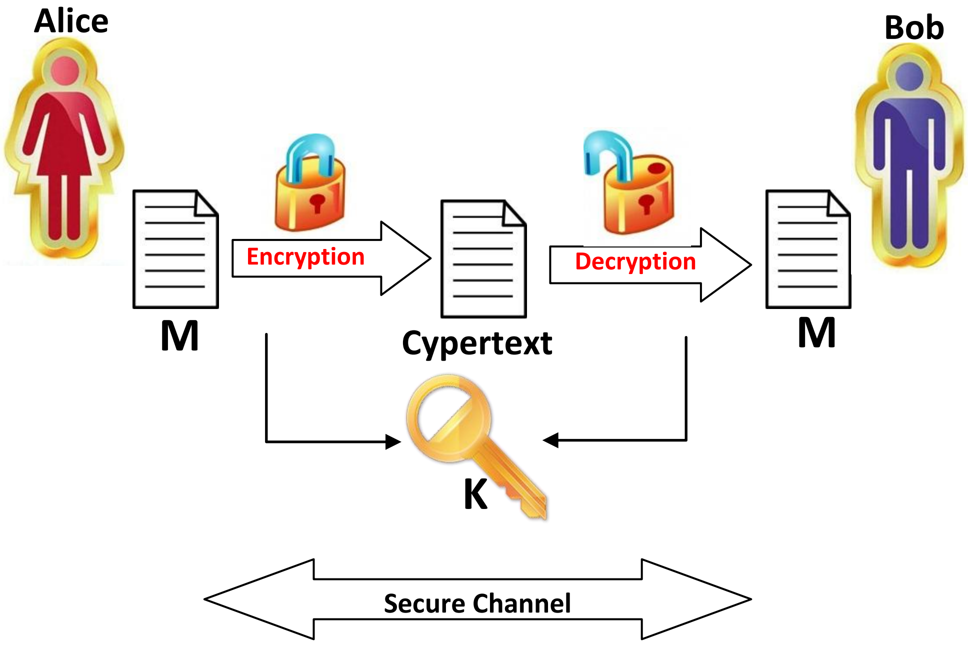


Figure 3.1: Symmetric key algorithm

3.3.2 Asymmetric Cryptography

There are problems in key management in the symmetric key. These have been solved in Asymmetric key (public key), proposed by a Whitfield Diffie and Martin Hellman in 1976. Public key can be defined as a format of cryptography where any user can have a pair of keys, which are called public and private keys. The procedure is to keep the private key secret, besides the public key can be widely distributed to all users in the network. Both keys are related mathematically. However, it is impossible to derive the private key is impossible from the public key in any way. Any message encrypted by public key only will be decrypted by corresponding private key [38].

Figure 3.2 shows the public key scheme clearly. [38] explains the process in the following way. Firstly, both users Alice and Bob must have two different keys which are the public and the private keys. If the user Alice needs to transmit an encrypted message ***M*** to the user Bob, she first of all gets Bob's public key (PK_{Bob}) and this key should be authenticated. Bob's public key can encrypt Alice's message ***M*** and transform it to cipher text ***C***. Then, the user Bob can decrypt the cipher text by identical private key (SK_{Bob}) which just the user Bob knows of.

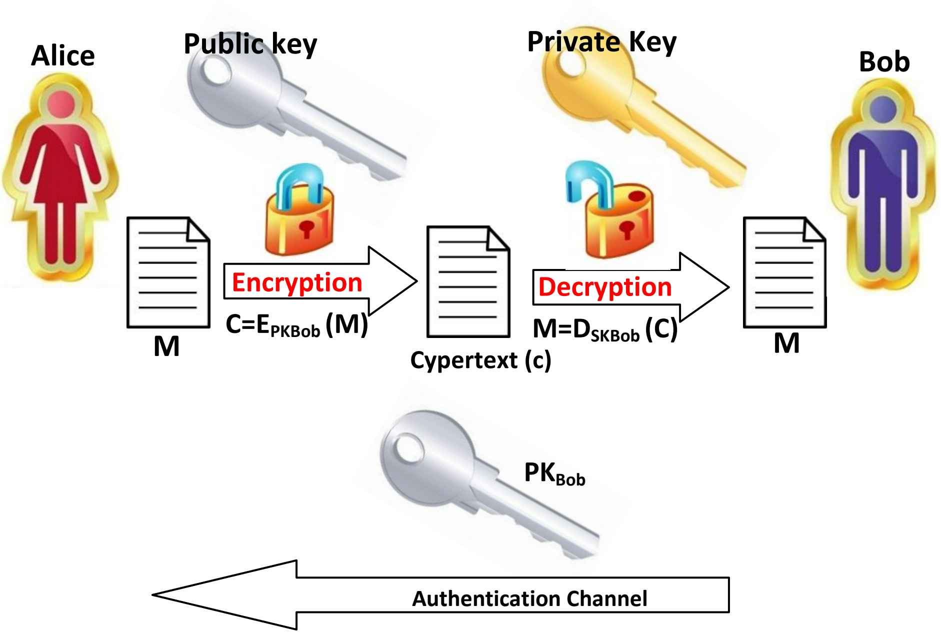


Figure 3.2: Asymmetric key algorithm

The two major sections of public key are [54]:

- **Public key encryption** — to ensure confidentiality, a message should be encrypted with a recipient's public key which cannot be decrypted by anyone except by the recipient possessing the corresponding private key.

- **Digital signatures** — to guarantee authenticity, integrity and non-repudiation, a message signed with a sender's private key can be confirmed by everyone who has admission to the sender's public key, thus confirming that the sender signed it and that the message not been altered.

3.3.2.1 Public key

The main issue with the public key is to confirm it is genuine, and has not been altered or exchanged by any malicious user [39]. The use of *Public Key infrastructure* (PKI), in which one or more third parties, called *Certificate Authorities* (CA), confirm the ownership of the key pairs is one way of overcoming this problem. Another approach, used by *Pretty Good Privacy* (PGP), is the *Web of Trust* (WoT) technique to guarantee the authenticity the pairs of key.

“A very popular example of public key cryptography is the RSA system developed by Rivest, Shamir and Adleman, which is based on the integer factorization problem” [39] 1996, p.89). In RSA mechanism, to encrypt any plain text ***m*** or decrypt any cipher text ***c***, the next mathematic computations are executed:

$$C = M^{e} \bmod N$$

$$M = C^{d} \bmod N$$

“A major benefit of public key cryptography is that it provides a method for employing digital signatures” [40]2008, p.35).These : (a) allow the receiver of the data to confirm the authenticity of the origin for the data; (b) maintain the data complete and undamaged; and (c) prevent the sender from claiming that user did not transmit the message. Therefore, public key digital signatures preserve the authentication, the data integrity and the non-repudiation.

3.3.2.2 Digital Signature

Digital signature is based on the same principle as the handwritten signature. However, the handwritten signature can be forged easily. The digital signature is more advanced than the handwritten signature because it is impossible to emulate it. In addition, it asserts to the contents of the data and the identity of the signer.

Figure 3.3 below, based on [39] (1996: pp.22-23), illustrates the way in which digital signatures are generated. The data is encrypted with the sender‘s private key and not with someone else’s public key. For example, if the data can be decrypted with the sender A’s public key, so it must have been created by that sender A.

Figure 3.3 shows that user Alice needs to transmit a message ***m*** to user Bob which is signed by her (Alice). [41] (1995, p.19-22] explains the various stages in this process. User Alice applies the hash digest of ***m*** and her private key to generate the signature. Firstly, she will use a hash function on ***m*** and calculate the hash digest. Secondly, she will encrypt this digest by her private key (SK_{Alice}) then transmit it with the message to the user Bob. Then, Bob recalculates the digest by using the same hash function on ***m*** which has been received and match it with the digest outcome from decrypting the signature by the Alice’s public key (PK_{Alice}). As a result, if both digests are equal, then ***m*** originates from Alice and has not been modified.

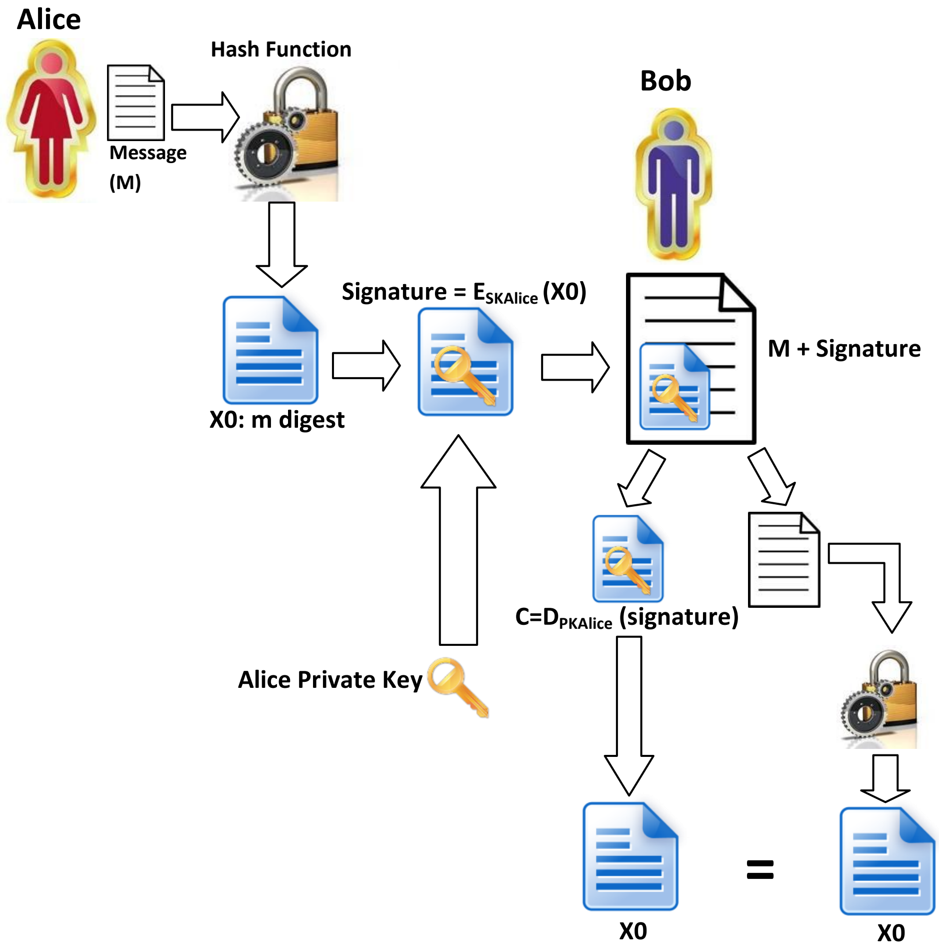


Figure 3.3: Digital Signature

3.3.2.3 Public key encryption schemes

Many of the public key techniques are related to mathematical issues, as listed in Table 3.1 based on [42] (2008, p.88).

Public key encryption scheme	mathematical issue
RSA	(a) integer factorization problem (b) RSA problem
Rabin	(a) integer factorization problem (b) square roots modulo composite n
ElGamal	(a) discrete logarithm problem (b) Diffie-Hellman problem
generalized ElGamal	(a) generalized discrete logarithm problem (b) generalized Diffie-Hellman problem
McEliece	(a) linear code decoding problem

Table 3.1: Some public key encryption schemes

3.3.3 Digital certificates

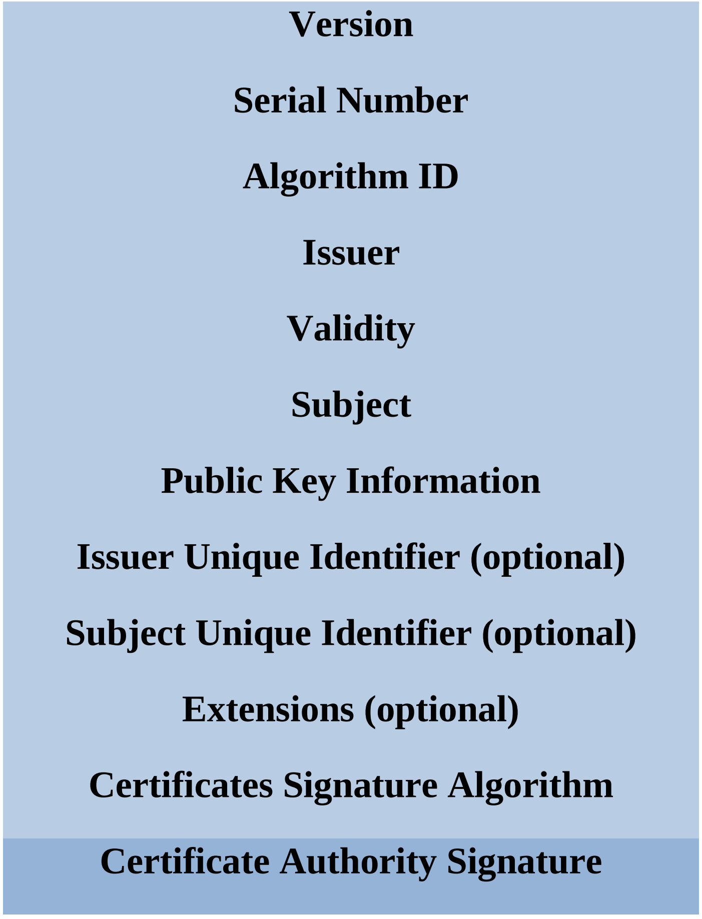
An important aspect of public cryptography is that users want to ascertain that they are encrypting to the accurate identity. This is because in “an environment where it is safe to exchange keys freely via public servers, man-in-the-middle attacks are a potential threat” [43] 2002, p.67). This type of attack is dangerous in that not only can it read the messages between two parties but it can also insert and modify them at will. This process can happen without the parties’ knowledge that the connection among them has been compromised. In this way, the attacker has to be capable to monitor and interrupt these messages passing among the two victims. The following paragraphs explain this process further by referring to Figure 3.3 above and also on [44].

Suppose Alice needs to transmit a secure message to Bob, she then asks for Bob’s public key. Suppose there is a third party called Emma who can interrupt the messages among Alice and Bob. If Emma is capable to get Bob’s public key, this will allow the *man-in-the-middle* attack to start. The attack will work in this manner: firstly, Emma will act as Bob’s identity and transmit her public key to Alice instead of the public key of Bob. Then, as soon as Alice receives Emma’s public key, she will think that it actually belongs to Bob’s public key and she will apply it to encrypt the message and then transmit it back to Bob. Again this encrypted message is interrupted by Emma.

Now, Emma can decrypt the message by her private key. She saves a copy of the message and re-encrypts it by using the Bob’s public key. When this message is received by Bob, he will think that it was transmitted by Alice (Wu 2007). “This example shows the need for Alice and Bob to have some way of ensuring that they are truly using each other's public keys rather than the public key of an attacker. Otherwise, such attacks would be generally possible, in principle, against any message sent using public-key technology” [48]2009, p.1-24).

Man-in-the middle attacks, such as the one described above, can be prevented by the use of digital certificates. “A digital certificate is an electronic document which incorporates a digital signature to bind together a public key with an identity-information, such as the name of a person or an organization and their address” [46]2008, p.209). It is used to check that public key applies to a singular entity. [47] explains that in a model of *public key infrastructure* (PKI) system, the signature will be of a *Trusted Third Party* (TTP) known as *Certificate Authority* (CA).

X.509 is a commonly used as a standard for clarifying digital certificates subsequent the PKI system. It is released as ITU recommendation ITU-T X.509 [55]. The X.509’s format, which is version 3, digital certificate, is shown in Figure 3.4 according to [55] .



Version

Serial Number

Algorithm ID

Issuer

Validity

Subject

Public Key Information

Issuer Unique Identifier (optional)

Subject Unique Identifier (optional)

Extensions (optional)

Certificates Signature Algorithm

Certificate Authority Signature

Figure 3.4: X.509 Digital certificate format

3.4 Hash Function

Hash function is frequently known as a unidirectional hash function. It is considered as one of the essential primal in the present area of cryptography [48]. [48] 2009, p.13) explains that a hash function $\boldsymbol{H}$ is a conversion by taking a message $\boldsymbol{m}$ and return it as a fixed size chain known as hash value $\boldsymbol{h}$. This conversion is represented as the equation $\boldsymbol{H}(.)$, to be exact, [$\boldsymbol{h} = \boldsymbol{H}(\boldsymbol{m})$]. Hash functions only with this attribute have many common mathematical uses. Those used in cryptography has several additional attributes [49] 2008). These are explored in the following paragraphs. There is no size limitation in the cryptographic hash function input. Compared with the output, it has to be of limited size. In addition, hash function has to be simple to calculate.

Additionally, the hash function must be unidirectional and collision free. If it is difficult to reverse, that is, it is one way hash function [56]. A hash function can be weakly collision free or strongly collision free [57]. We have a weakly collision free hash function, if given a message x, it is mathematically impossible to apply finding a message y 6= x such that ***H***(x) = ***H***(y). On the other hand, we have a strongly collision free hash function if it is mathematically impossible to apply finding x and y messages such that ***H***(x) = ***H***(y).

The hash value appears in brief with respect to the longer message from which it was calculated; this value is known as message digest. A message digest, therefore, can be understood as a digital fingerprint of the major message. Examples of famous hash functions are MD-family (MD2 [58], MD4 [59, 60], and MD5 [61], RipeMD-family (RIPEMD-128 [62], RIPEMD-160 [62], RIPEMD-256 [62], RIPEMD-320 [62], and SHA-family (SHA-1 [63], [64], SHA-256 [64], and SHA-384 [64], SHA-512 [64].

According to [42], the majority of cryptography functions apply compression functions to compress the information. The compression function works in the following way: it gets a limited size of input and gives a shorter limited size of output. For a specific compression function, we can describe the hash function as repeating the compression function till the whole message has been processed. “In this process, a message of arbitrary length is broken into blocks whose length depends on the compression function, and padded so that the size of the message is a multiple of the block size” [42] 2008, p.130). For instance, SHA-1 block size and RIPEMD-160 block size are 512 bits.

3.5 Key Management

Cryptography works as a security system to preserve the *confidentiality*, the *integrity*, and the *authentication* as long as the keys are not compromised whatsoever. [35] explains how the capture, modification, corruption or disclosure of the keys to unauthorized individuals can lead to the whole cryptosystem becoming compromised. Cryptography depends on a trust model. Trust works at the level of individuals – they trust everyone in the network to defend their keys and also trust the administrator who maintains the keys – and at the level of administrators – they also trust the server that saves, maintains, and deploys the keys.

Key management is the essential section of any secure communication, as shown by [45]. The majority of cryptography systems depend on some underlying secure, strong, and effective key management system. Key management covers several aspects including key creation, storage, deploying, updating, cancellation, and certificate service, according to security policies [65]. If any key is exposed, the encrypted data would not be protected from malicious attacks. The privacy of the symmetric key and private key has to be guaranteed assuredly. In fact, both Key distribution and key agreement over unsecure channel formally are in a high chance of risk and also vulnerable from feasible attacks. In the classical digital envelop way, one side produces a session key and encrypts it with the public key algorithm. After such a generation and encryption, the other side receives and recovers it. In the Diffie-Hellman (DH) system, as elaborated in [51] (2005, p.3), communication on both sides shares some public information and creates a session key on both sides. A number of difficult key exchange or distribution protocols and frameworks have been designed and made. Nevertheless, the calculation load and complexity of these protocols have been limited by (MANETs).

According to [45], this is due to a number of factors including the node's lack of resource availability, mobility and network synchronization complexity. Strong key attacks remain a threat. It is therefore essential to protect Key integrity and ownership.

Various techniques such as *Digital Signature*, *Message Digest* and *Hashed Message Authentication Code* (HMAC) are used to authenticate data or to maintain the integrity. As shown by [46]2008, p.484), "public key is protected by public-key certificate where a trusted entity known as the certification authority (CA) in PKI vouches the binding of the public key with owner's identity". In methods where there is no trusted third party (TTP), public key certificate has to be ensured in a different way. Certification is then achieved through peer nodes in a distributed method, such as pretty good privacy (PGP). Clearly, the purpose of key authentication is that the certificate can prove the ownership of the key rather than decide whether it is good or not. After certain valid period of usage, the key might be compromised or disposed of. As the key should not be used again after it has been discovered, some techniques are required to revoke the compromised key in the period where it has not yet expired. As explained by [51] (2005, p.5), the certificate "contains the lifetime of validity. If the key is expired, it is not useful. However, the private key maybe is able to be disclosed during the valid period. In this case, certificate authority (CA) needs to revoke this certificate explicitly and notify the network by using the certificate revocation list (CRL) to prevent its invalid usage". Key maintenance is a very important factor in securing a communications network. It is used to encrypt and decrypt messages. The keys also have to meet certain conditions and fulfill some specific functions: (a) distribution: securely to the right entities and updated continuously; (b) protection: when being transmitted and stored on each workstation and server; (c) generation, destruction and proper recovery: these need to happen on demand by authorized individuals.

The keys must be stored securely before and after distribution. As soon as the key is distributed, it does not exist to be found in any location. It needs a secure place to be stored and applied only in the controlled method. Indeed, the "keys, the algorithm that will use the key, configurations, and parameters are stored in a module that also needs to be protected. If an attacker were able to obtain these components, she could masquerade as another user and decrypt, read, and re-encrypt messages that were not intended for her" [35]2003, p.557). [66] (2005, p.32) further explain that a "Key Management System (KMS) creates, distributes, and manages these certificates. Thus, the KMS is at the heart of the network's defenses. A KMS provides high service availability in highly partitioned networks, requires minimal pre-configuration during the network deployment phase, and can accommodate new nodes joining the network".

3.5.1 Key Exchange

"Key exchange is the most primitive form of key management. People wishing to communicate over an insecure channel must exchange a cryptographic key" [50] 2003, p.1). The initial version of the key management was the physical key exchange, if it can be explained as key management in any way. Generally, key exchange is the majority of "inconvenient" [50] 2003, p.1) method of generating a secure relationship between two communicating entities. But it is essential to use this method in some cases. Indeed, according to [50] (2003, p.1), in "some ad hoc networking scenarios, it is NOT inconvenient" but it is actually a requirement. "Thus, for small personal area networks or similar scenarios, physical key exchange must be both logical and convenient" (Lehane et al. 2003, p.1).

3.5.2 Key Agreement and Group Keying

"Group keying allows multiparty secure communications, and hence provides group level authentication and security" [50] 2003, p.2). Nevertheless, giving the keying information for single members of the group (for example, to permit the members to communicate in privacy in the presence of other members of the group) needs other predetermined key agreements. In fact, networks might design where group membership does not exist, especially in a wide range national network. Essentially, any group key agreement is of restricted use in a non group oriented network, like a civilian network that lots of mobile nodes select to transmit but some require continuous confidentiality. For this reason, a "public key infrastructure is better suited to this scenario" [50] 2003, p.2).

3.6 Summary:

This chapter provides an overview of the cryptography in routing protocols. Furthermore, it gives an introduction and a background by state the cryptography objectives. This chapter also discusses the cryptography types which include the Symmetric cryptography, Asymmetric cryptography and Digital certificates. Indeed, in the chapter, we give a description of a hash function and key management and how a group keying gives the level of authentication.

CHAPTER 4: SECURITY

This chapter introduces applications of secured routing protocols ARAN and Ariadne in some interesting examples.

4.1 Security Examples

4.1.1 ARAN

Authenticated Routing for Ad hoc Networks (ARAN) was proposed by [23]. Authors present in their research the probable security uses against the routing protocols in MANET. Particularly, the attacks which are used against the AODV [32] and DSR [29]. The route discovery stage in ARAN routing protocol is depend on AODV and DSR, but the explained attacks are decreased by amount of extensions to the protocol.

Route discovery

ARAN needs both of the Route Requests (RREQs) and Route Replies (RREPs) to be signed. According to [23], this is done in the interest of (a) authentication: to prevent spoofing of node identity; (b) integrity: to ensure about the packet not been adjusted from the time when generated; and (c) non-repudiation: to capture any internals malicious mobile node for instances, mobile nodes has a genuine certificate and the identical key pair). [23] (2002, p.5) explain this process in detail. A summary of their explanation follows.   Suppose mobile node S needs to establish a route to mobile node D. Just one route to D is the intermediate mobile nodes A and B. Consequently, the route should be (S-A-B-D). Firstly, mobile node S starts a *route discovery packet* (RDP), similar to the (RREQ), and transmits it to the all mobile nodes in the network.

The sender S will sign the message. At the same time as the packet is received by the intermediate mobile node A, it will check the accuracy of the packet such as certificates, addresses and signatures. The intermediate mobile node will sign on it and attach its own certificate if it is accurate; and so on with next mobile node to check the accuracy of the packet. Once the packet comes correctly to the intermediate mobile node it will delete the previous mobile node's certificate and signature and replaced with its own certificate and signature. Then transmit the packet to the next mobile node and will be checked through the route to the destination. If the RDP packet arrived to the destination mobile node D, it will create and send backward a RREP packet through the reverse route to the source mobile node.

For the RREP packet procedure it is the same as explained in RDP packet for checking the certificate and signature. ARAN protocol instead of using the hop count to select a route it will select the fastest RDP packet travel from source to destination. This procedure will avoid the attacks whereas any cooperative malicious mobile node will show every route appear as a shorter route if it is travel via them. This kind of attack called "tunneling" attack; the malicious mobile node will cover the RREQ packet and transmit it to nearest malicious mobile node to the destination. Once the other malicious mobile node receives it will uncover it and transmit it to the destination. In this stage, the route request appears to the destination with extremely low hop count, as it has a high probability to be chosen if the hop count is used to evaluate the route quality.

4.1.2 Ariadne

Ariadne is a secured routing protocol proposed by [6] and [7], and is based on DSR routing protocol. It depends on symmetric cryptography (Hu et al. 2002). It ensures the authentication and the integrity of the routing packets:

1. Destination mobile node of a route discovery procedure can validate the source node.
2. Source mobile node can validate every intermediate mobile node present on the route to the destination in the RREP messages; also can ensure that no intermediate mobile node is removed from the mobile node list in RREQ or RREP packets.

The authentication in Ariadne for the routing packets can be in three ways. It can use any of the following schemes [6]2002, p.21):

1. “Shared keys between all pairs of nodes”,
2. “Shared keys between communicating nodes combined with broadcast authentication”,
3. “Digital signatures”.

Authors assume that there exists a key distribution scheme for each authentication scheme. The next section discusses the use of Ariadne with *Timed Efficient Stream Loss-tolerant Authentication* (TESLA) [26], a scheme that request time synchronization. Synchronization can be avoided if pair-wise shared keys are used. Additional protocol optimizations can also be achieved by broadcast authentication such as TESLA. Ariadne needs a mechanism to enable each node to share a secret key (i.e., K_{SD} between source and destination). A TESLA key for each node in the network must be securely set up for each node in the network.

4.1.2.1 TESLA broadcast authentication protocol

As mentioned above and discussed in [26], Ariadne uses the TESLA for authenticating routing packets. It is efficient in that it attaches just the Message Authentication Code (MAC) to a routing packet in order to achieve the broadcast authentication. A MAC can provide point-to-point authentication between two nodes using the same shared key. However, the receiving mobile nodes require the MAC key to authenticate the message for broadcasting communication. This is a vulnerability that may allow any receiving node to forge packets and impersonate the sender. TESLA solves this issue by relying on the time synchronization and delayed key detection.

In order to utilize TESLA, all sending nodes generate a unidirectional key chain by choosing an initial TESLA key K_N and repeatedly applying a unidirectional hash function H on this initial value. The equation is "$K_i = H[K_{i+1}] = H^{N-i}[K_N]$" [7] 2005, p.23). The mobile node request the equation to the arriving value to verify the total is equal to the previous received key. This will authenticate all received values on the one way chain. For example, in order to authenticate K_i, we use the equation "$K_j = H^{i-j}[K_i]$" [7] 2005, p.23) to compute the value of K_j. If this value matches the previously received value of Kj, then K_i is authenticated.

Each sending node decides a schedule to detect all keys of its unidirectional key chain, in sequentially K_0, K_1,..., K_N. An easy and basic key detection schedule is the time at which K_0 is detected, and the time t is the key detection interval. TESLA based on a receiving mobile node to check which keys a sending mobile node might have already detected. To do this, a receiving node calculates the time synchronization among nodes. For example, allow D be the highest difference between two mobile nodes; D value have to be known by every node. For sending a packet, the sender mobile node chooses a key K_i from its own unidirectional key chain, uses the key to make a MAC value. This MAC value is attached to the packet. On receiving a packet authenticated with TESLA, the receiving node checks if the K_i has been disclosed by verifying t_r <= (T_0 + i*t −D) – which is called TESLA condition. If this inequality is true, the K_i has not been detected – if not the key might have already detected and the attacker might have the packet been faked.

The authors in [7] (2005, p.23) explain that if this verification is fully successful, the receiving mobile node will save the packet. Moreover, it stays for the sender mobile node to distribute key K_i. When the receiver mobile node has the K_i, it first authenticates K_i by applying the equation "$K_j = H^{i-j}[K_i]$" [7] (2005, p.23). After that, it authenticates the saved packets and it is authenticated with a key K_j, where j <= i.

Route Discovery

In Ariadne, the basic RREQ message contains eight fields [6] (as shown below). Their functions are to provide: (a) authentication and (b) integrity to the routing protocol.

The authors in [6] (2002: p.28) show that the mobile node verify its local table for source mobile node address and the identifier values as the mobile node received RREQ that is not the destination. If it received the same RREQ before, the node will reject it. That stage works in the same way as DSR protocol works. In addition, the mobile node verifies the time period in the RREQ. Its validity depends on these terms: (a) time period not be so large, and (b) the key identical to it has not been detected. Otherwise, the mobile node will reject the packet. If the packet met the TESLA conditions, (a) the mobile node will change the RREQ by attaching its address in the mobile node request's list ,(b) it will exchange the hash chain with "**H [A, hash chain]**" [6] 2002: p.28), and (c) attach the MAC of the complete RREQ to MAC's list. The mobile node needs the TESLA key to calculate the MAC. At the end, the mobile node will retransmit the changed RREQ.

As the destination mobile node has the RREQ, it will verify the TESLA condition, which equal to:

"H [h_n, H [h_{n-1} , H [. . . , H [h_1 , MAC_{Ksd} (initiator, target, id, time interval)]..]]]", [6] (2002: p.28). If the destination mobile node defines the RREQ packet is valid, it backwards a RREP packet to the source mobile node.

Route Maintenance

TESLA handles the authentication of RERR messages in a way similar to how the RREQ messages are handled [7]. These are briefly explained here with reference to [7]. To avoid the insertion of wrong route errors (RERRs) at the wireless network from any mobile node that have not seen the broken link, every mobile node going back on the same route to the source mobile node just transmit the RERR packet. This leads to a delay in TESLA authentication. So every node on the returning route saves the error without sending it awaiting it is authenticated. Finally, the mobile node that saw the broken link detects the key and transmits it back to the same route, which will allow mobile nodes on that route to validate the saved error packets

Ariadne is secure against the wormhole attacks just in its advanced edition that uses the TIK (TESLA with Instant Key disclosure) protocol

[25]. The authors in [25] (2003, p.4-5) show how the TIK protocol lets for extremely accurate time synchronization among the mobile nodes of the wireless network. It can also disclose any exception in the routing traffic flows in the wireless network.

4.2 Summary

In this chapter, two secured routing protocols for MANETs have been briefly presented. Their applications in some examples have also been explained.

CHAPTER 5: ELLIPTIC CURVE CRYPTOGRAPGHY IN DSR ROUTING PROTOCOL

This chapter discusses Elliptic Curve Cryptography in DSR Routing Protocol. The first two sections describe the Elliptic Curve Cryptography by drawing on the works of [5-7, 36, 40, 67-85] . In the third section, Its present and analyze the Diffie-Hellman Key Exchange [69, 74]. The fourth section focuses on the Elliptic Curve Cryptography Diffie-Hellman Key Exchange. Its then move on to describe the ECDH Experiment in MANET [2, 86-88] and discuss simulation results on self-created scenarios using performance metrics. In the fifth section it presents the security handshake attacks. The chapter ends with a short summary.

5.1 Introduction

Elliptic Curve Cryptography (ECC) is defined as a process of the public key mechanism which depends on the algebraic formation of elliptic curve over a finite field [5] Figure 5.1. The technique of elliptic curve, based on cryptography, was proposed only by N. Koblitz and V. Miller in the year 1985. According to Kumar and Anil (2011, p. 544), "public-key cryptography is based on the intractability of certain mathematical problems". In the beginning public keys are secure considering the difficulty to the large real number consisting of two or more large prime factors. "For elliptic-curve-based protocols, it is widely assumed that finding the discrete logarithm of a random elliptic curve element with respect to a publicly-known base point is infeasible" [80] 1986, p.418).

The Elliptic Curve size defines the difficulty of the problem. The main advantage of Elliptic Curve Cryptography is a smaller key size. This leads to a reduction in memory and communication demands that an ECC group could offer the same level of security as an RSA technique afforded based on a big modulus and identically larger key. For example, 256 bit key size in ECC should give as same security as 3072 bit key size in RSA as shown in table 5.1 [83]. For modern cryptography objectives, an *elliptic curve* is a standard curve which contains the points which meet the equation proposed by [80].

$$y^2 = x^3 + ax + b,$$

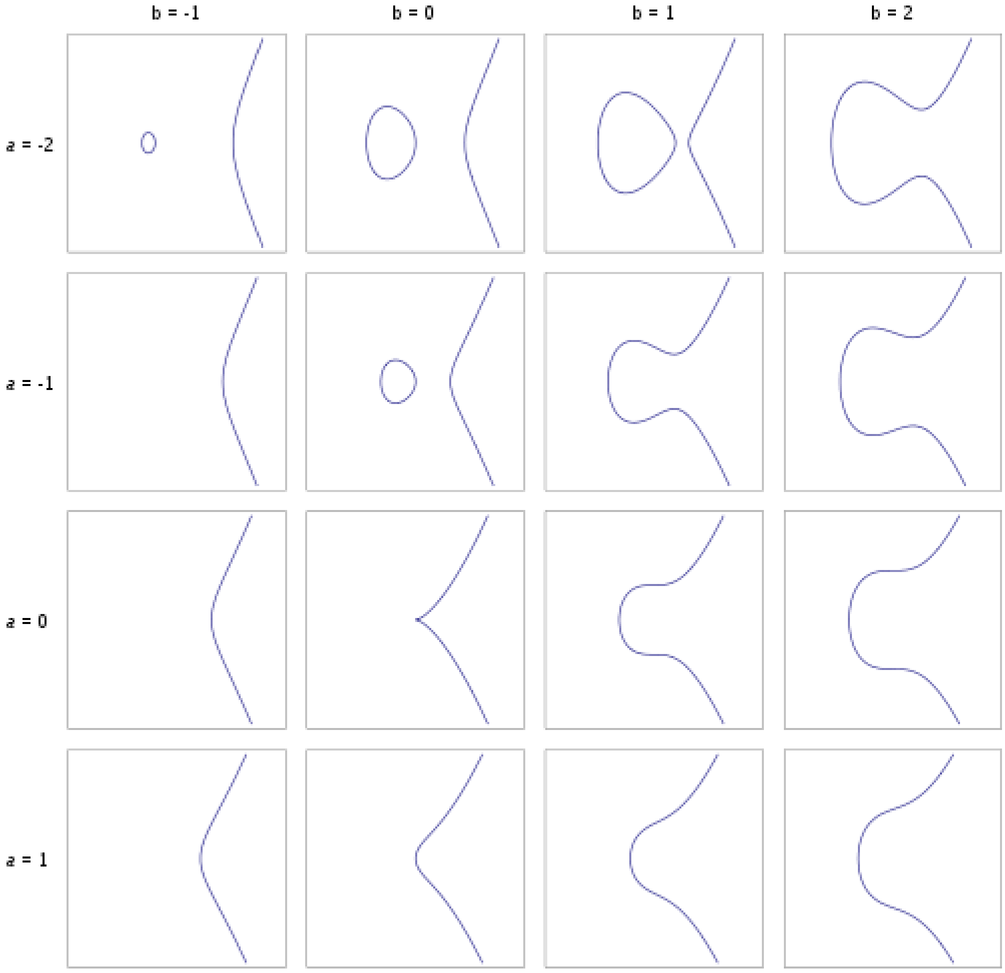


Figure 5.1: A catalogue of elliptic curves.

Symmetric Algorithm (bits)	RSA and DH (bits)	ECC (bits)
56	512	112
80	1024	160
112	2048	224
128	3072	256
192	7680	384
256	15360	521

Table 5.1: Keys sizes comparison with RSA, DH and ECC

ECC has been verified as lightweight computationally, comparable to RSA. Elliptic Curve Cryptographic mechanism can offer the same standard of security as other cryptosystems such as RSA which provide signatures by using Elliptic Curve DSA (ECDSA) [76], key establishment by using Elliptic Curve Diffie-Hellman (ECDH) and asymmetric encryption by using Elliptic Curve Integrated Encryption Scheme (ECIES) [78].

5.2 Elliptic Curve Cryptography (ECC)

Researchers like [84] have looked at the implementation of the Elliptic Curve cryptography (ECC) for Mobile Ad Hoc Network.

The benefits offered by ECC [36, 82, 84] such as size and efficiency, have made it the cryptographic option for wireless networks and network devices. The elliptic curve cryptography system (ECCS) is a cryptography process of using a *discrete logarithm problem* (DLP) through the points on the elliptic curve [67]. ECC is generally defined over two finite fields: the main finite field F_p containing p factors and the special finite field containing 2^m factors. Cryptographic schemes are based on ECC which rely on scalar multiplication of elliptic curve points, as explained in [71]. Given a real number k and a point $P \varepsilon E(F_p)$, the scalar multiplication simply is the method of adding point P to itself k times. As an outcome of this scalar multiplication is stand for k times P or ($k*P$). [69] shows how the scalar multiplication for the elliptic curve numbers can be calculated professionally by using the addition base together with the double and add algorithm or only one of it differs.

The great numbers of public key cryptography such as RSA or DH utilize even real number or polynomial mathematical with huge numbers or polynomials. This poses a major problem: it will force a considerable load in storing and processing for the keys and also for the messages. This will, in turn, result in lower speed and consumption of more bandwidth. Indeed, another solution is to apply an elliptic curve which offers two important aspects: the same security and smaller key sizes. An ECC is defined as an equation with two factors x and y, with coefficients [70]. In view of the cubic elliptic curve in the form of $y^2 = x^3 + ax + b$, as x, y, a, and b are all integers.

The elliptic curve over integers is also problematic [68]. It can affect: (a) speed – the calculations can be slow; (b) accuracy – inaccurate results can be obtained due to rounding error and also unlimited fields. As a result, cryptography mechanisms need to be rapid and accurate arithmetically. In the cryptography mechanisms proposed by [73], elliptic curves cryptography applies curves whose factors and coefficients are limited over two finite fields. There are two families commonly used, as explained by [75]:

- prime curves $E_p(a,b)$ defined over Z_p

  This type uses real numbers modulo as a prime. It is best to use in application softwares, which does not require extending bit-fiddling processes required by the binary curves.

- binary curves $E_2{}^m(a,b)$ defined over $GF(2^n)$

  This type uses polynomials and binary coefficients. It is best to use in hardware, which can get fewer logic gates to generate a cryptography system in contrast with prime curves.

5.2.1 Secure Routing Comparisons

A number of secure routings have been proposed. For example, [72] proposed on demand routing protocol ARAN for MANET environment which utilize certificates to guarantee the authentication, the integrity and the non-repudiation of routing protocol messages. This protocol uses public key cryptography to overwhelm the attacks and ensures secured routing for the managed-open and open ad hoc networking environments.

A secured routing protocol, SRP, was proposed by Papadimitratos and Haas [81]. It ensures secured communication in the open, collaborative and highly dynamic ad hoc networking environment. SRP respond to malicious behaviour in a timely manner and ensures comprehensive secure communication. Ariadne [6] prevents a wide range of attacks to ensure secure routing in an ad hoc networking environment. This protocol uses highly efficient symmetric cryptography that makes it more proficient and prohibits attackers from tampering with uncompromised routes. The problem with this protocol is that it does not safeguard against passive attackers.

Zhou and Haas [89] have used effective key management to ensure secured routing over ad hoc networking environment. [79] have used misbehaviour detection schemes to secure ad hoc networks. The problem with this scheme is that it does not guarantee to have two main security parameters viz. integrity and authentication of routing packets. Johnson et al. (2002) proposed to use symmetric cryptography for secured routing over ad hoc networking environments. This can be implemented using one way hash chains. Zapata and Asokan [85] proposed a secured routing protocol that can make the use of asymmetric cryptography to authenticate participating mobile nodes and also uses one way hash chains to ensure secured routing over wireless ad hoc environment.

5.2.1.1 ARIADNE

Ariadne, proposed by [7] and discussed in Chapter 4, is based on DSR routing protocol and relies only on TESLA which is a kind of Symmetric Cryptography. TESLA is a proficient authentication that requires wide time synchronization. It works in the following way: first, it checks route authenticity and then it checks that no nodes are missing on RREQ message. It is vulnerable to any attacker during the route discovery process.

5.3 Diffie-Hellman Key Exchange

The sender and receiver Diffie-Hellman algorithms were proposed by Diffie and Hellman in 1976 [74]. The Diffie-Hellman algorithm, as described in Chapter 3, relies on the complexity of calculating the discrete logarithms. It presumes that all users who participate in the network define the prime number p and a primitive root g of p that necessary ($g < p$). Below is an example of Diffie-Hellman exchanges protocol [90] 1998, p.1): Alice and Bob are to exchange some information to each other by applying a public key cryptography technique:

1. The users in public select a cyclic group G and a creator x of G.
2. The two users Alice and Bob can select private keys a and b, as a and b are arbitrary real numbers.
3. Now user Alice calculates x^a, and also user Bob calculates x^b, then they will exchange x^a and x^b values through an insecure network.
4. While receiving these values between them, both users Alice and Bob calculate the value xab by applying their keys and indeed is $x^{ab} = (x^a)^b = (x^b)^a$ values are equals.

An eavesdropper (Chapter 2) who intercepts the message will have to get the value x^{ab} from x, x^a, and x^b to be able to decrypt it. This is called the discrete logarithm problem (DLP).

There are advantages to using ECDH key exchange in Mobile Ad Hoc Networks. These include: perfect flexibility to node capture, excellent scalability, low memory and bandwidth, and low communication overhead.

5.4 ECC Diffie-Hellman Key Exchange

As described in Section 5.2, ECC requests are extremely smaller than traditional public key cryptography systems, however maintaining an equivalent level of security is a main concern. The use of Elliptic curve permits faster encryption and decryption. The key exchange among users *A* and *B* by applying elliptic curve Diffie-Hellman (ECDH) can be accomplished as follows [84] (2006, p.2244), as shown in figure 5.2:

1- User *A* choose a real number which is A’s private key n_A which is less than *p*. Then user *A* produces a public key from this equation:

$P_A = n_A * G$; (public key is a point in $Ep(a, b)$).

2- User *B* will do the same by selecting a private key n_B and calculate the public key from the equation:

$P_B = n_B * G$.

3- User *A* executes a scalar multiplication to achieve the shared secret key from this calculation:

$K = n_A * P_B$.

4- User *B* also execute a scalar multiplication to achieve the shared secret key from this calculation:

$K = n_B * P_A$.

[84] (2006, p.2244) show how the two calculations in steps 3 and step 4 generate the same outcome that because

$$n_A * P_B = n_A * (n_B * G) = n_B * (n_A * G) = n_B * P_A.$$

Both parties obtained equals values for *K* as $E(F_\mathrm{p})$ is a commutative group. Every single run of the ECDH protocol needs both user A and user B to transmit two messages (exchanging the ephemeral public keys) and to execute overall of four scalar multiplications. The two parties could compute the first two scalar multiplications at the same time and the other two thereafter. “It is also possible to recalculate a pair of ephemeral keys when the parties are idling to speed up subsequent protocol runs” [91] 2009, p.118). Indeed, the secret key *K* is a point in the ECC. An eavesdropper knows only n_A and n_B but is not able to compute the secret from that. Since the two users A and B public keys are probably not to be changed over network lifetime and could be used again for key exchange with different communications partners, it is possible to recompute them offline before sensor deployment. This is illustrated below. The drawback of using ECDH is the intensive computation from the cryptographic processes which will affect the energy consumption.

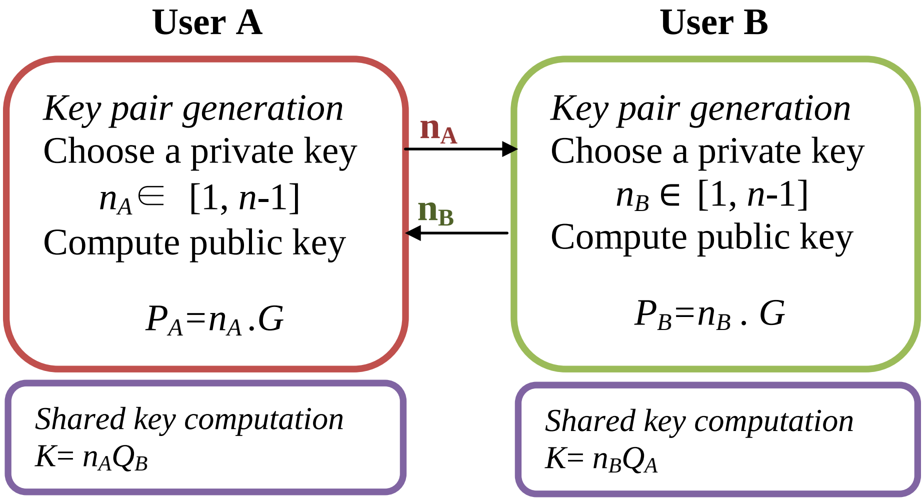


Figure 5.2: ECC diffie-hellman key exchange

5.5 ECDH Experiment in MANET

5.5.1 Goals

One of the main goals is to determine the effects in the Performance analysis of ECDH extension on the performance metrics which is described in 5.5.3. In the experiments the normal DSR routing protocol has been used as a reference. Our goal is also to compare the ECDH with other existing secure routing protocol to illustrate the difference between them. Instead of simulating ECDH protocol in two different experiments one with DSR and one with Ariadne we have joined them in one experiment to achieve good results. The results indicate that ECDH routing protocol performs better than Ariadne and DSR routing protocols. This is because the proposed scheme contains memory effectiveness and powerful security advantages with less complication.

5.5.2 Simulation Setup

Various performance metrics have been used in different network scenarios providing the modifications in DSR, ECDH and Ariadne routing protocols. In all scenarios DSR routing protocol is used as a reference. Indeed, Ariadne has been compared to distinguish the difference of security that has been used in ECDH routing protocol in MANET.

In the second section of the experiment, we introduced the misbehaving mobile nodes in the network which do not forward packets to the other mobile nodes.

The simulator has been implemented on Network Simulator 2 (NS2.34)[92], a simulator for MANET. The experiments in this chapter were run 9 times. In addition, the confidence level of the intervals is 95.70%

5.5.3 Parameters

The set of parameters for the simulations are shown in the Table 5.2.

Parameter	Value
Area	670m x 670m, 1500m x 300m
Speed	1 to 10 m/s, 0 to 5 m/s
Radio Range	250 m
Movement	Random waypoint model
MAC	802.11
Application	UDP, CBR
Packet Size	512 Bytes
Simulation Time	500s, 600s
Number of Nodes	10, 20, 50 nodes
Pause Time	100 s
Simulation runs	7, 9 times

Table 5.2: Parameters

5.5.4 Metrics

RFC 2501 illustrates the amount of quantitative and qualitative performance metrics that can be applied to analyze the performance of MANET routing protocols as well as the secure routing [2]. Metrics that have been used to analyze the performance of proposed on-demand routing protocol (DSR), the proposed secure routing (ECDH), and an existing secure routing protocol (Ariadne) are the packet delivery fraction (PDF), average end to end delay, network throughput and normalized routing load. In the sections below, PDF has been used as quantitative metrics for pattern analysis and performance evaluation as mentioned in the secure routing protocols. This metric determines the completeness and correctness of the secure routing protocol, as discussed by [86].

5.5.4.1 Packet Delivery Fraction (PDF)

[88] describe the ratio of number of the packets received (DPR) at the destinations over the number of the packets sent (DPS) by the sources as shown below.

$$\boxed{\text{PDF} = \text{Total DPR} / \text{Total DPS X } 100}$$

5.5.4.2 Average End-to-End Delay (Delay)

It measures the average time engaged in delivery of the packets from source to destination [88]. This can be computed by adding each delay for each succeeded packet delivery and after that, dividing the total by the number of succeeded received packets, as shown below.

$$\boxed{\text{Delay} = \sum (\text{Time Received} - \text{Time Sent})/\text{Total Packets Received}}$$

5.5.4.3 Network Throughput

"A network throughput is the average rate at which message is successfully delivered between a destination node (receiver) and source node (sender)" [88], p.35). It is also referred to as a ratio of the number of data received from its sender to the time the last packet reaches its destination. The unit of measurement of Throughput is bits per second (bps). A high level of throughput is a requirement in any network; it is required that the throughput is at high-level. There are some factors that affect the throughput for instance, topology changes, energy limitation, bandwidth limitation and unreliable communication.

- ***Throughput Vs Goodput:***

Goodput has been used as one of the performance metrics which is the total number of correct and uncorrupted packets delivered to destination. In contrast to Throughput, loss and retransmission packet has been considered. Packet loss, which can be happen because of link errors, unreachable mobile nodes or the intermediate mobile nodes drops them, can affect the Goodput. Thus, we use the total number of dropped packets in the network as a performance metric.

5.5.4.4 Normalized Routing Load

“The normalized routing load is determined as the ratio of all routing control packets sent by all nodes in the network over the number of received data packets at the destination nodes” [88]2012, p.35). Alternatively, it is the overall numbers of routing packets sent divided by the overall number of data packets received, as shown below.

Normalized Routing Load = Total Routing Packets Sent/Total Data Packets Received

5.5.5 Performance Metrics

In the sections below, it presents and analyses different simulation results using performance metrics. We have evaluated the performance by selecting several network scenarios.

5.5.5.1 Analysis using Performance Metrics

The random waypoint model has been used as a mobility model [87]. A wide simulation model including the scenario of 10, 20 and 50 mobile nodes has been used to measure the performance for DSR, ECDH and Ariadne. We simulated by using NS 2.34 simulator tools. And the packet size is 512 bytes. The equal scenario has been used for all protocols to match the results. Both ECDH and Ariadne protocols share the same on-demand behaviour, (which is based on DSR routing protocol). The difference between both protocol techniques can point to a considerable gap in performance. This is analyzed using packet delivery fraction in consideration of speed and pause time changing. The performance results that look close to the existing ones give better security than the others.

5.5.6 Simulations Results

5.5.6.1 10 mobile nodes having 6 UDP links

Area considered is 670 × 670 and the simulation running time is 500 seconds over pattern analysis of 10 mobile nodes by using both UDP and TCP links with consideration to varying speed and pause time. Figure 5.3 shows the packet deliveries Fraction based on the parameter of speed. The performance has been assessed for all protocols: DSR, ECDH and Ariadne by using 10 mobile nodes and 6 UDP links. In addition, the speed starts from 1 (m/s) to 10 (m/s). The packet delivery fraction PDF rates, calculated by using the received and the dropped packets, and the result rates are from 99.29% to 99.71%. The outcome shows that DSR in all t only at one point of time, as the same time ECDH and Ariadne give same PDF rates. In another way, DSR protocol performs better than ECDH and Ariadne protocols in “low mobility” case.

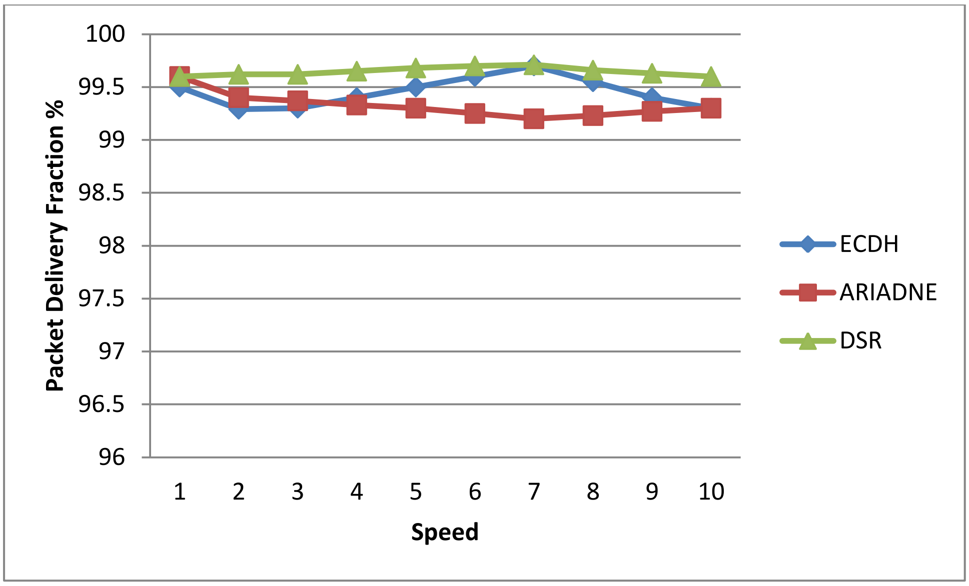


Figure 5.3: 10 mobile nodes with 6 UDP links

In Figure 5.4, the PDF has been assessed for all protocols: DSR, ECDH and Ariadne based on the parameter of pause time without changing the number of mobile nodes also UDP links. In addition, the pause time starts from 100s to 500s. The PDF rates were calculated by using received and dropped packets. The result rates were from 99.31% to 99.94%. In this simulation scenario, the perception here is that DSR gives fewer PDF rates than ECDH and Ariadne protocol when pause time ranges from 100s to 300s. However, ECDH and Ariadne protocols give approximately same PDR values, DSR and ECDH perform better than Ariadne when pause time is between 300 and 500s and Ariadne does better than DSR and ECDH when pause time is more than 500s. This enhanced the performance results that ECDH routing protocol launch between the destination mobile node which is receiving a ROUTE REQUEST and sending a ROUTE REPLY.

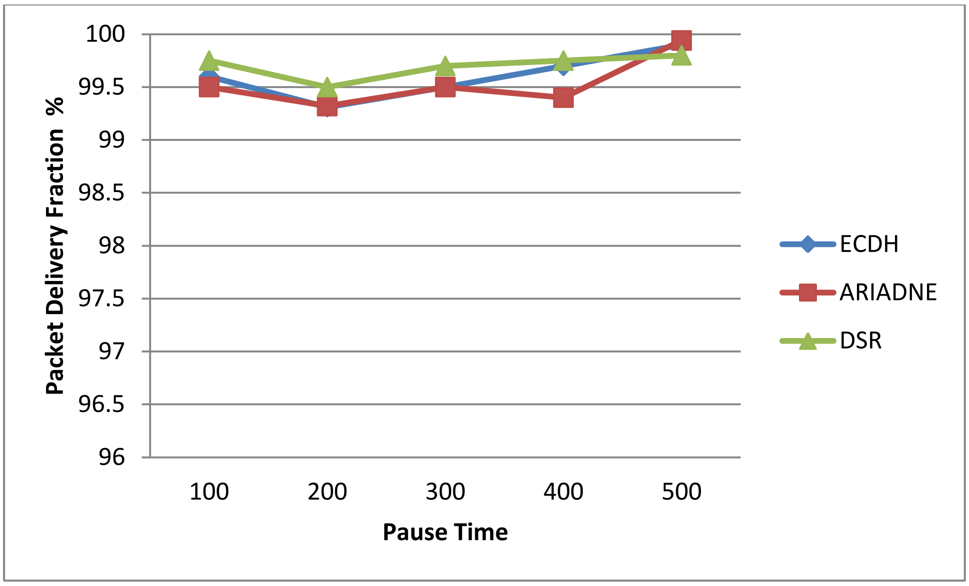


Figure 5.4: 10 mobile nodes with 6 UDP links

5.5.6.2 10 mobile nodes having 6 TCP links

Figure 5.5 shows the packet delivery fraction by using the parameter speed for all protocols: DSR, ECDH and Ariadne. The outcomes are based on 10 mobile nodes and 6 TCP links. In addition, the speed starts from 1 m/s to 10 m/s. The PDF rates are calculated by using received and dropped packets, and the results rates are from 97.61% to 98.12%. It shows that in "low mobility" case, Ariadne protocol gives just about the same PDF rates as ECDH protocol. That is because our ECDH protocol gives the same security power as Ariadne. DSR also gives lower PDF values. However, in "high mobility" case, Ariadne performs better than DSR and ECDH protocols.

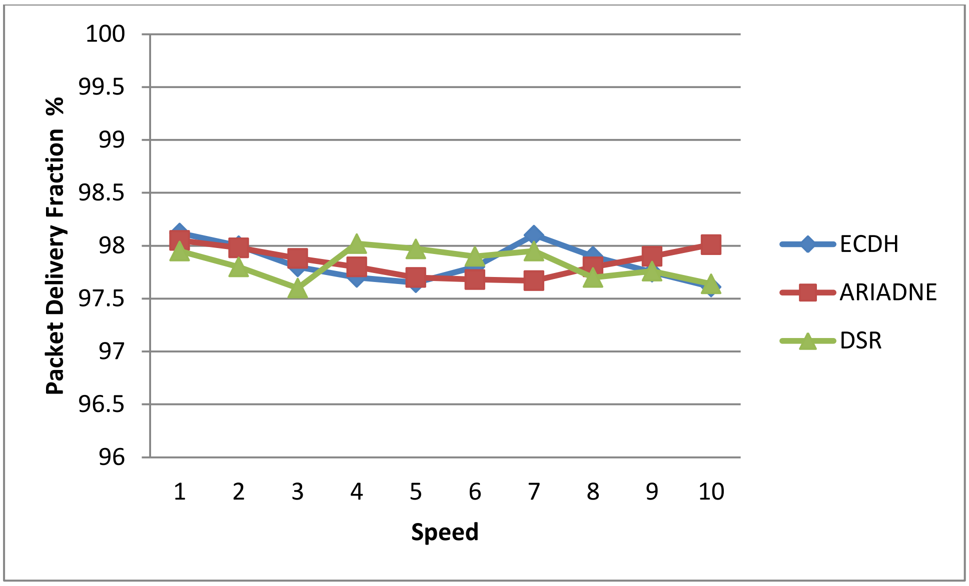


Figure 5.5: 10 mobile nodes with 6 TCP links

In Figure 5.6, the packet delivery fraction has been assessed using the parameter pause time based on 10 mobile nodes having 6 TCP links. In addition, the pause time begins from 100s to 500s. The PDF rates are calculated by using received and dropped packets. The results rates are from 96.41% to 98%. The perception here is that DSR gives low PDF and it increases when the pause time is increasing. However, Ariadne protocol better than DSR and ECDH as pause time is less but ECDH does better than DSR and Ariadne as the pause time is high as the ECDH trying to provide security as Ariadne.

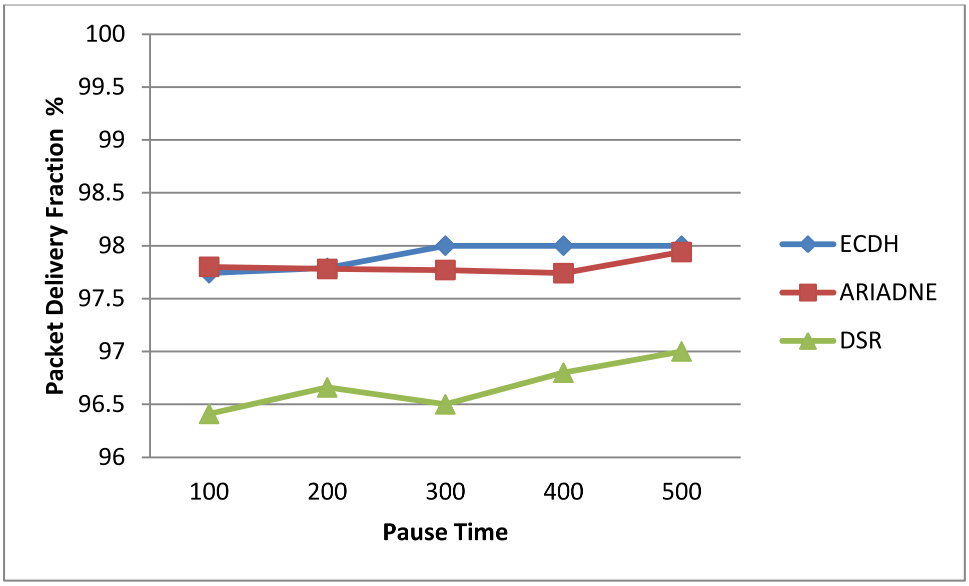


Figure 5.6: 10 mobile nodes with 6 TCP links

5.5.6.3 20 mobile nodes using 6 UDP links

Area considered is 750 × 750 and simulation run time is 500s over pattern analysis of 20 mobile nodes by using UDP and TCP links both with consideration to varying speed and pause time. Figure 5.7 illustrates the packet delivery fraction depends on the parameter “speed”. This performance has been assessed for DSR, ECDH and Ariadne protocols by using 20 mobile nodes using 6 UDP links. In addition, the speed begins from 1 m/s to 10 m/s. The PDF rates are calculated by using received and dropped packets. The results rates are from 96.92% to 98.74%. It shows that DSR in all speeds is steady. Indeed. At a single point of time, DSR, ECDH and Ariadne protocols give same PDF rates. Otherwise, ECDH protocol does better than DSR and Ariadne in “low mobility” case. This is because the number broken links were decreased for ECDH compared to Ariadne.

In Figure 5.8, the PDF has been assessed for DSR, ECDH and Ariadne protocols depending on the parameter of pause time without changing the number of mobile nodes or UDP links. In addition, the pause time begins from 100s to 500s. The PDF rates, calculated by using received and dropped packets, the results rates are from 95.09% to 98.94%. In this simulation scenario, the perception is DSR and ECDH protocols nearly give the equal PDF rates. Furthermore, DSR and ECDH perform better than Ariadne in all the cases. This is because that ECDH routing protocol is greatly efficient in discovering and maintaining routes between mobile nodes for delivering the data packets even with mobility.

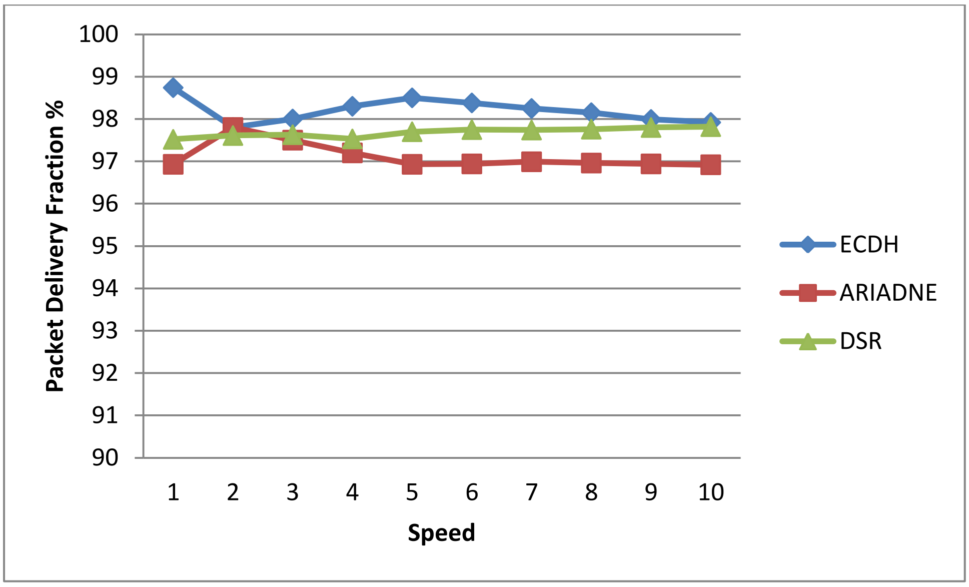


Figure 5.7: 20 mobile nodes with 6 UDP links

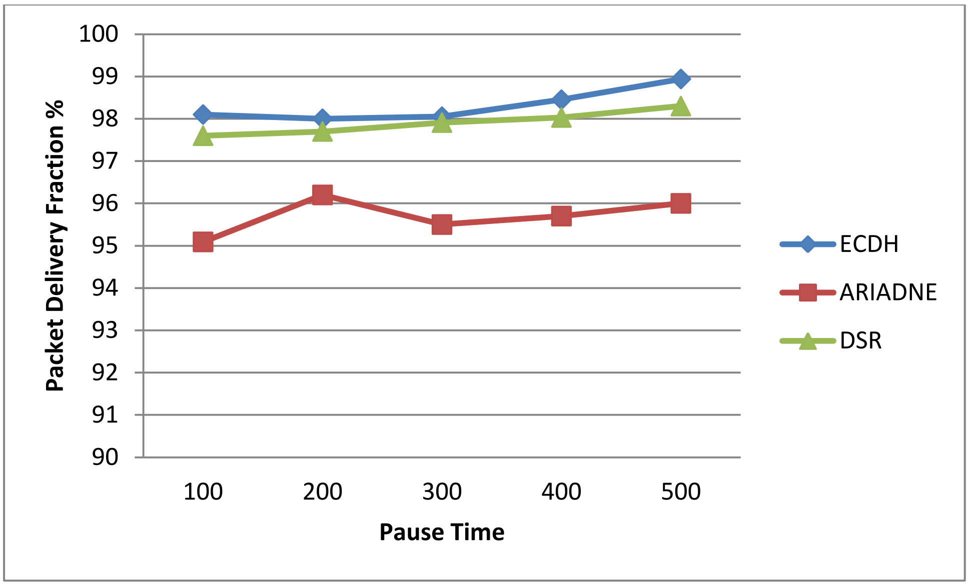


Figure 5.8: 20 mobile nodes with 6 UDP links

5.5.6.4 20 mobile nodes having 6 TCP links

Figure 5.9 presents the packet delivery fraction depending on the parameter speed for all protocols: DSR, ECDH and Ariadne. The outcomes are based on 20 mobile nodes and 6 TCP links. In addition, the speed starts from 1 m/s to 10 m/s. The PDF rates, calculated by using received and dropped packets, the outcomes are from 97.81% to 98.52%. It shows that in “low mobility” case, DSR protocol gives higher PDF rates than ECDH and Ariadne protocols in the beginning of the experiment. However, it gives lower rates than ECDH and Ariadne in just one point. Besides, ECDH and Ariadne protocols give roughly the same PDF rates as ECDH protocol in the beginning of the scenario, at the middle and at the end stage only. DSR and ECDH protocols approximately give equal results and also perform better than Ariadne. Otherwise, DSR does better than ECDH protocol in “high mobility” case.

In Figure 5.10, the PDF has been assessed based on the parameter pause time on 20 mobile nodes using 6 TCP links. Additionally, the pause time starts from 100s to 500s. The PDF rates are calculated by using received and dropped packets. The outcomes rates are between 97.23% and 98.34%. The perception is that the ECDH protocol performs better than DSR and Ariadne protocols as the pause time is low as the Route Discovery in ROUTE REPLY has a shorter time in ECDH routing protocol. Otherwise DSR does better than ECDH and Ariadne protocols as the pause time is high.

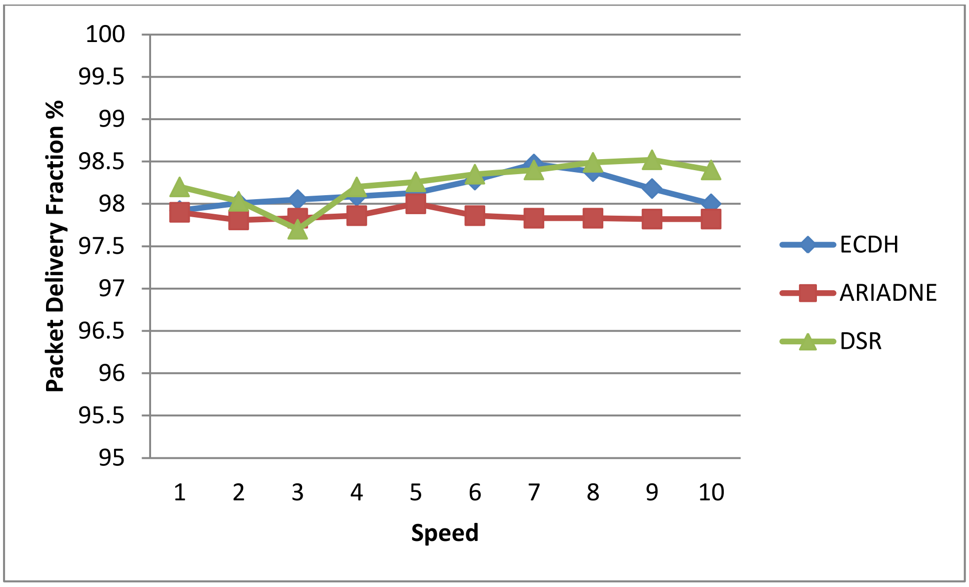


Figure 5.9: 20 mobile nodes with 6 TCP links

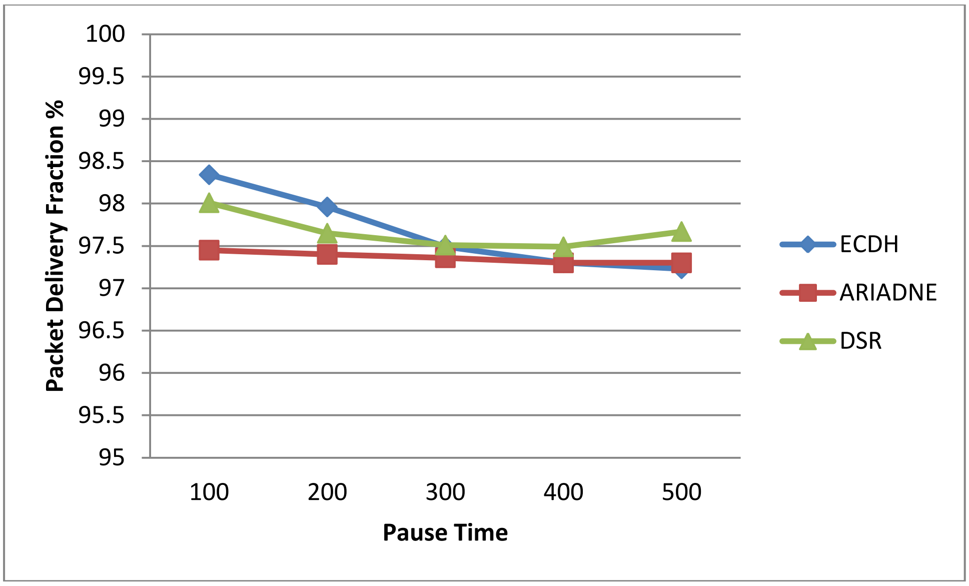


Figure 5.10: 20 mobile nodes with 6 TCP links

5.5.6.5 50 mobile nodes having 10 UDP links

In this simulation scenario, area considered is 1000 × 1000 and run time is 700 seconds over pattern analysis of 50 mobile nodes using both UDP and TCP links with consideration to varying speed and pause time. Figure 5.11 illustrates the PDF by using the parameter “speed”. This performance has been assessed for all protocols: DSR, ECDH and Ariadne using 50 mobile nodes and 10 UDP links. In addition, the speed starts from 1 m/s to 10 m/s. The PDF rates are calculated by using received and dropped packets. The outcome rates are between 89.04% and 95.60%. The DSR protocol performs better than ECDH and also ECDH protocol performs better than Ariadne protocol as route cashing in ECDH can further reduce Route Discovery time unlike Ariadne. In Figure 5.12, the PDF has been also assessed for DSR, ECDH and Ariadne protocols by using the parameter pause time without changing the number of mobile nodes and the UDP links. Additionally, the pause time begins from 100s to 650s. The PDF rates are calculated by using received and dropped packets. The results rates are from 88.95% to

95.26%. In this simulation scenario, the perception is the same as mentioned above: DSR does better than ECDH and ECDH does better than Ariadne. The decrease in Ariadne is because the losses of data packets during the communication due to broken links and the node mobility.

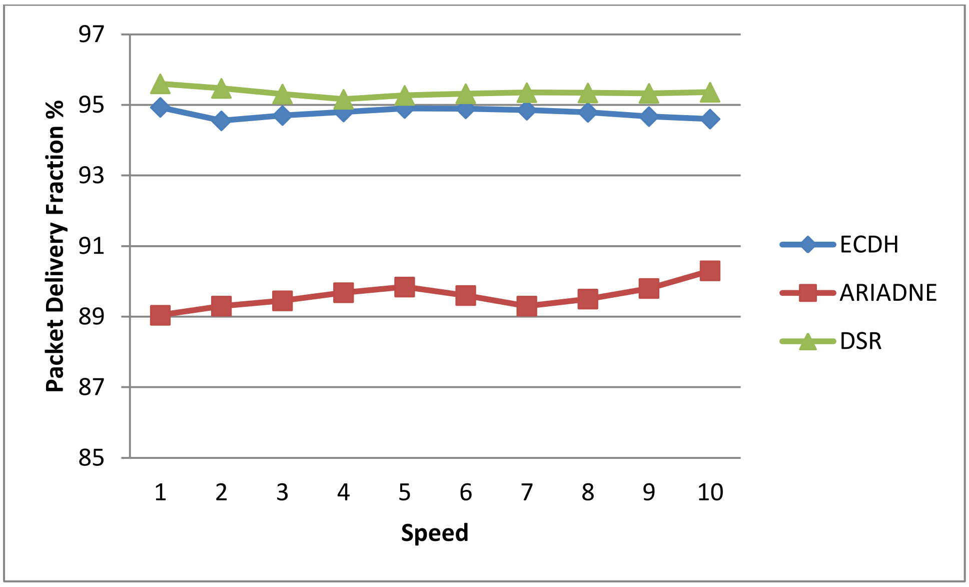


Figure 5.11: 50 mobile nodes with 10 UDP links

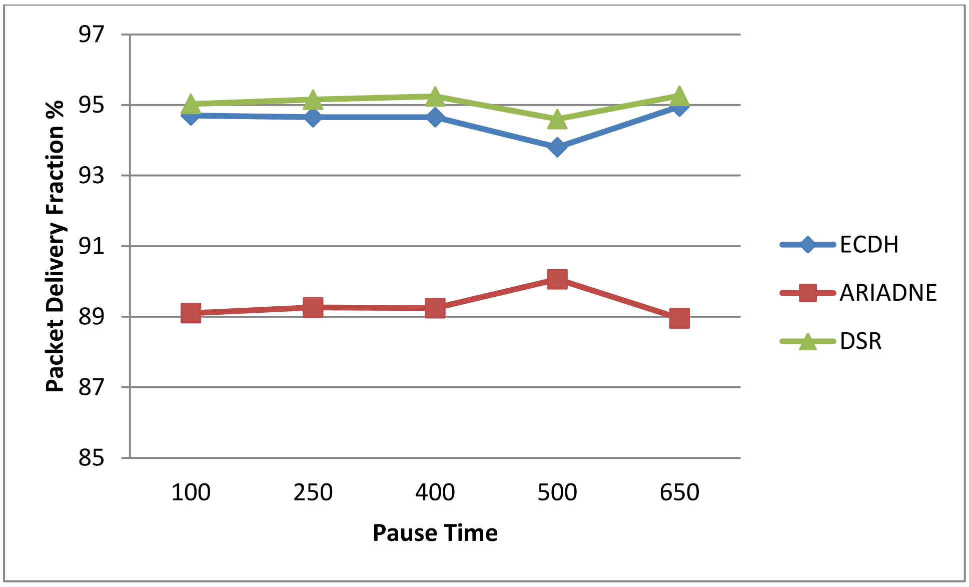


Figure 5.12: 50 mobile nodes with 10 UDP links

5.5.6.6 50 mobile nodes having 10 TCP links

Figure 5.13 shows the PDF by using the parameter of speed for all protocols: DSR, ECDH and Ariadne. The outcomes are based on 50 mobile nodes and 10 TCP links. Furthermore, the speed starts from 1 m/s to 10 m/s. The PDF rates are calculated by using received and dropped packets. The outcomes are between 91.71% and 96.58%. In "low mobility" case, ECDH approximately gives the same PDF rates as the Ariadne protocol. However, DSR protocol gives lower PDF values in same "low mobility". However, in "high mobility" case, ECDH does better than DSR and Ariadne protocols. In Figure 5.14, the PDF has been assessed by the parameter of pause time based on 50 mobile nodes having 10 TCP links. Furthermore, the pause time begins from 100s to 650s. The PDF rates are calculated by using received and dropped packets. The results are between 92.70% and 97.04%. In this perception the DSR protocol does better than ECDH and Ariadne while ECDH performs better than Ariadne when the pause time is low. The three protocols give approximately the same PDF rates when pause time is high. Those routes in ECDH are maintained among mobile nodes in the network that need to communicate. This has decreased the overhead of route maintenance.

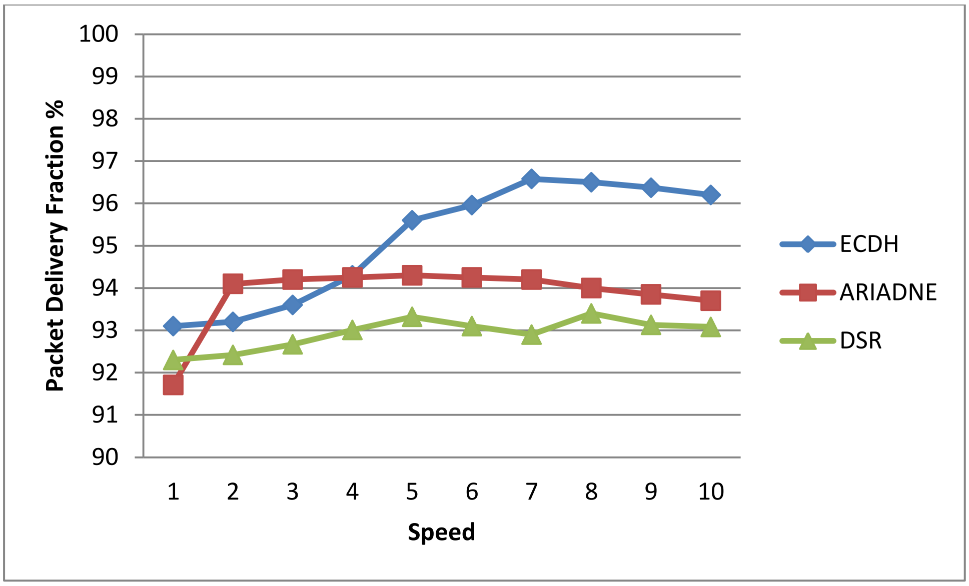


Figure 5.13: 50 mobile nodes with 10 TCP links

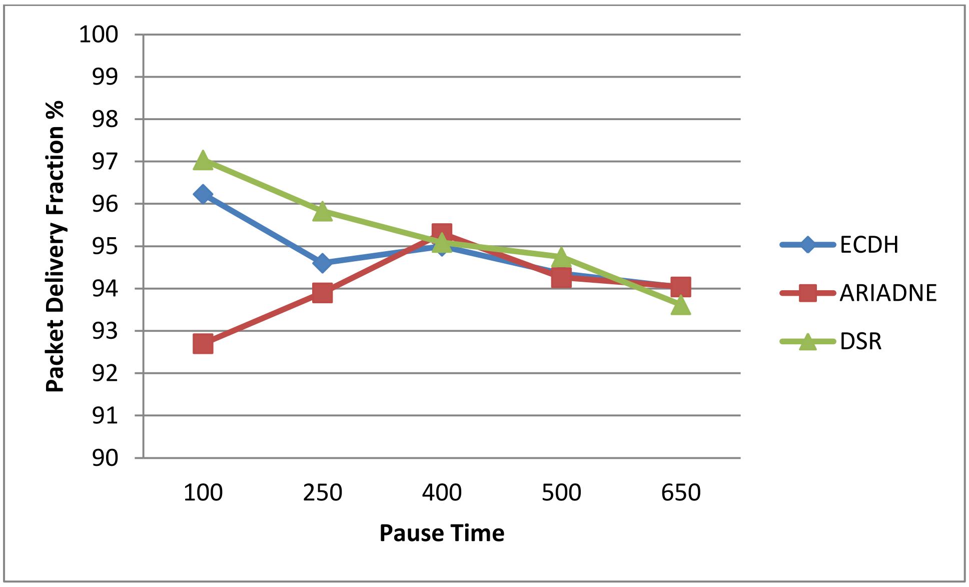


Figure 5.14: 50 mobile nodes with 10 TCP links

5.6 Security handshake attacks

The following paragraphs describe two types of security handshake attacks: SYN flooding attack and session hijacking. As explained and illustrated in the figures below, these two kinds of attack differ in the way in which they are carried out.

1. *SYN flooding attack*: This is a kind of attack that allows many halves to open TCP connections and keep them open without completing the handshake procedure, as shown in Figure 5.15. The attack takes place in the following manner. Firstly, mobile node A transmits a packet SYN to mobile node B which is the Synchronize, and sequence number = X. Secondly, node B sends a packet SYN, and ACK to node A which is synchronize acknowledge, sequence number = P, and acknowledge number = X+1. Finally, node A sends a packet ACK to node B which is acknowledge, sequence number equal to X+1, and acknowledge number equal to P+1. As a result of this procedure, the handshake is now completed.

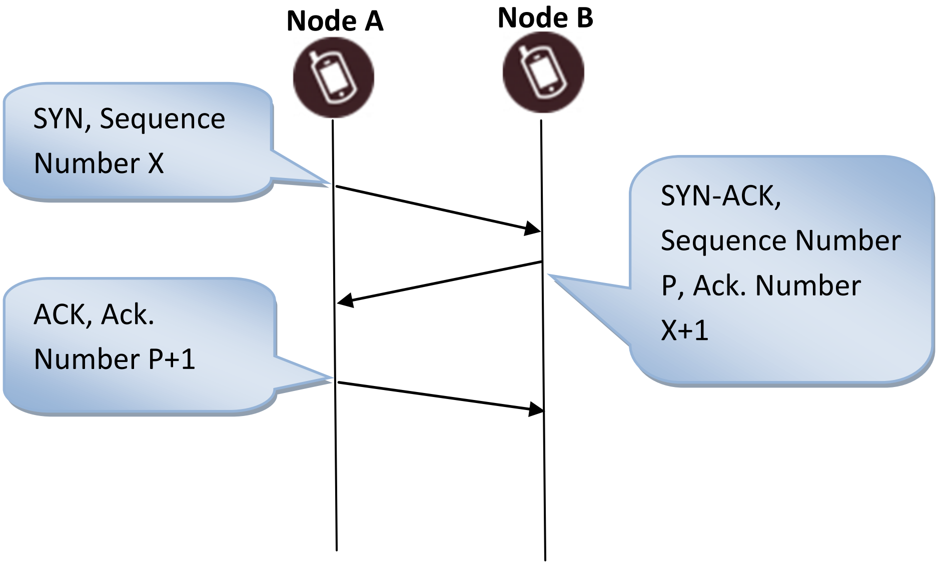


Figure 5.15: TCP handshake

Through the SYN attacks, the malicious mobile node transmits a huge number of SYN packets to the target mobile node. This deceives the return SYN packets addresses. Then the SYN-ACK packets will send from the target mobile node after receiving the SYN packets from the malicious mobile node. In next stage, the target mobile node remains for the reply of ACK packet. With no getting the ACK packets, the semi opened connection will remain for the acknowledgment of the handshake; all that will affect and overflow the buffer. This is illustrated in Figure 5.16 below.

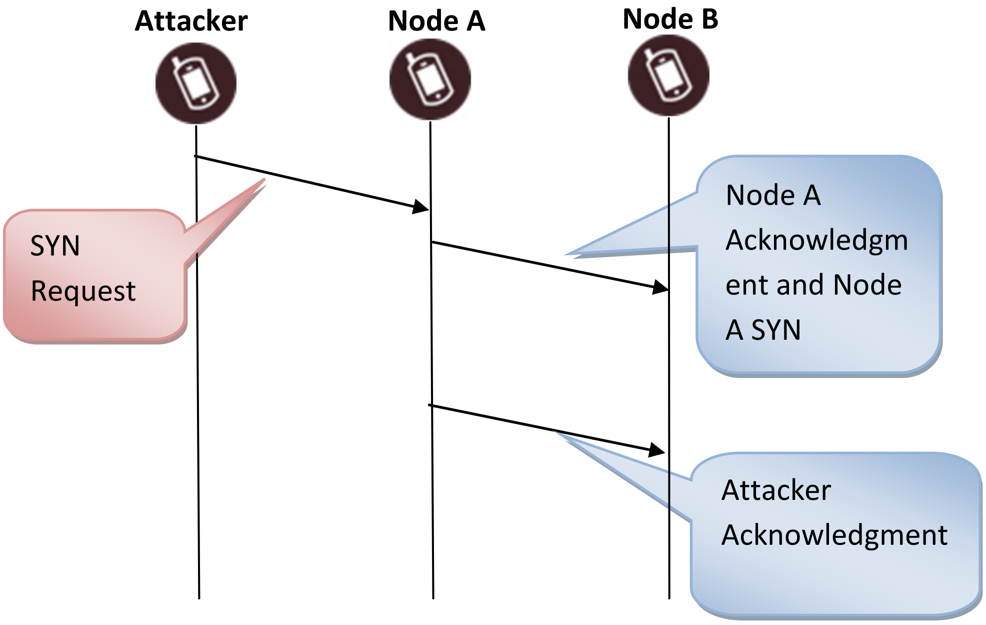


Figure 5.16: SYN attack

2. *Session Hijacking*: it gets the benefit that the majority of the connections are secured (via giving the certificates) at the session establish, except the next stage. "The TCP session hijacking attacks, the attacker spoofs the victim's IP address, determines the correct sequence number that is expected by the target, and then performs a DoS attack on the victim" [93] 2010, p.210) . Consequently, the attacker will pretend to be the victim and carry on the session with the destination node. This type of attack is illustrated in the figure 5.17 below.

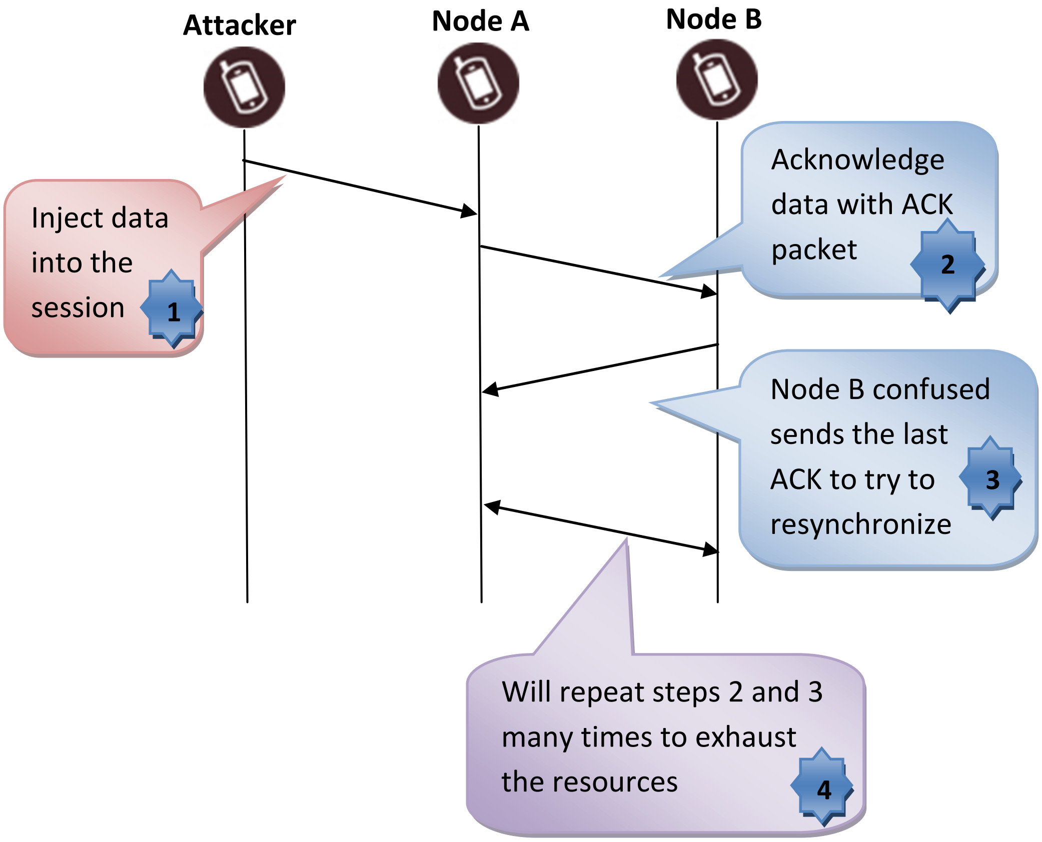

Attacker
Node A
Node B
Inject data into the session
1
Acknowledge data with ACK packet
2
Node B confused sends the last ACK to try to resynchronize
3
Will repeat steps 2 and 3 many times to exhaust the resources
4

Figure 5.17: Session hijacking attack

5.6.1 Performance metrics

In this section, it looks at the set-up for the experiments performed. We have simulated the experiment in NS-2. All mobile nodes in our simulation scenario moving depend on the *Random Waypoint* model; a mobile node begins at an arbitrary location, remains for the duration known as pause time, and then selects another location and goes there by the speed between 0 and 5 m/s. it has used a space size 1500m X 300m to raise the amount of hops in the routes. The average values of these 7 simulation runs are then calculated for the two metrics in malicious environment. The three protocols: DSR, Ariadne and ECDH were run on the same movements and the same communication scenarios, as shown in Table 5.2 and 5.3. It calculated three metrics for each simulations run:

1. *Packet Overhead*: is number of routing packets broadcasts; for instance, a (RREP) transmitted through three hops will take it as three packets in this metric.
2. *Packet Delivery fraction*: is total fraction of application level packet sent that was in fact received at the intended destination mobile node.
3. *Average Latency*: is average time ended from when the packet is initially sent to when it is initially received at its destination.

Number of nodes	**50 nodes**
Speed	0 to 5 m/s
Area	1500m X 300m
Nodes connections	20
Number of malicious nodes	10 nodes
Packet size	512 bytes
Initial RREQ timeout	2 seconds
Maximum RREQ timeout	40 seconds

Table 5.3 Scenarios Parameters

5.6.2 Simulation Results

In the sections below, it presents the simulation results for 10 malicious nodes. The figures below show the simulation result for each of the three metrics (packet overhead, packet delivery fraction and average latency) mentioned above.

5.6.2.1 10 malicious mobile nodes

Figure 5.18 shows the overhead. ECDH protocol has constantly lower overhead than Ariadne protocol. This is achieved by decreasing the quantity of Route Error (RERR) packets that have been sent. In addition, because of TESLA the overhead is much lower than for the DSR and ECDH protocol. Figure 5.18 also illustrates that ECDH authentication gets much less overhead than Ariadne protocol.

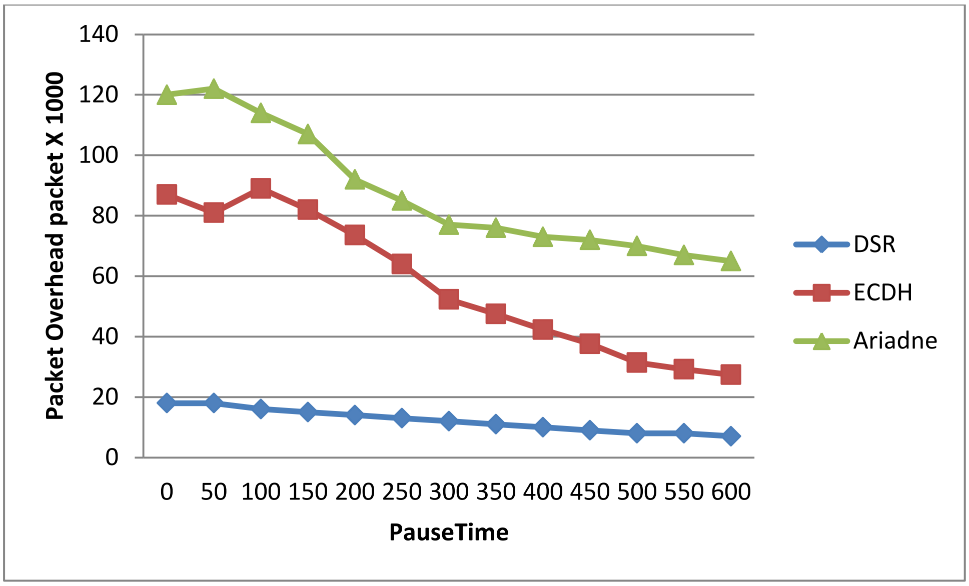


Figure 5.18: Overhead

Figure 5.19 illustrates the packet delivery fraction (PDF) for all protocols. The data show that the delivery packet ratio in DSR and Ariadne is lower than ECDH at higher levels of mobility. This is due to the fact that Route Discovery in a Route Reply will have a shorter time.

Of particular interest here is the result for ECDH. Surprisingly, ECDH actually outperforms Ariadne and DSR at lower and higher level of mobility. This enhanced the performance outcomes from the average delay that ECDH introduces among the target when receiving the (RREQ) packet and sending the (RREP) packet.

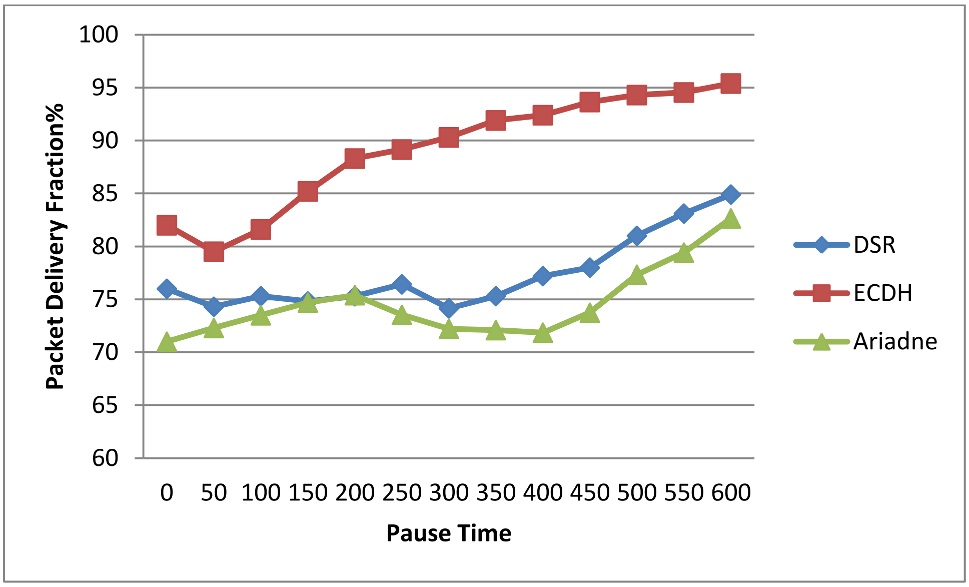


Figure 5.19 Packet Delivery Fractions

Figure 5.20 shows the average latency for all protocols. In general average latency, ECDH protocol does better than Ariadne protocol. This is due to the decrease in the number of broken links for ECDH protocols (compared to the Ariadne protocol). However, DSR has a better average latency than both ECDH and Ariadne. The results show that after the packet forwarding is disabled by a number of malicious mobile nodes of the MANET, the overall network performance hardly deteriorates. To correct this, some kind of selfishness behavior has to take into consideration while designing a secure routing protocol.

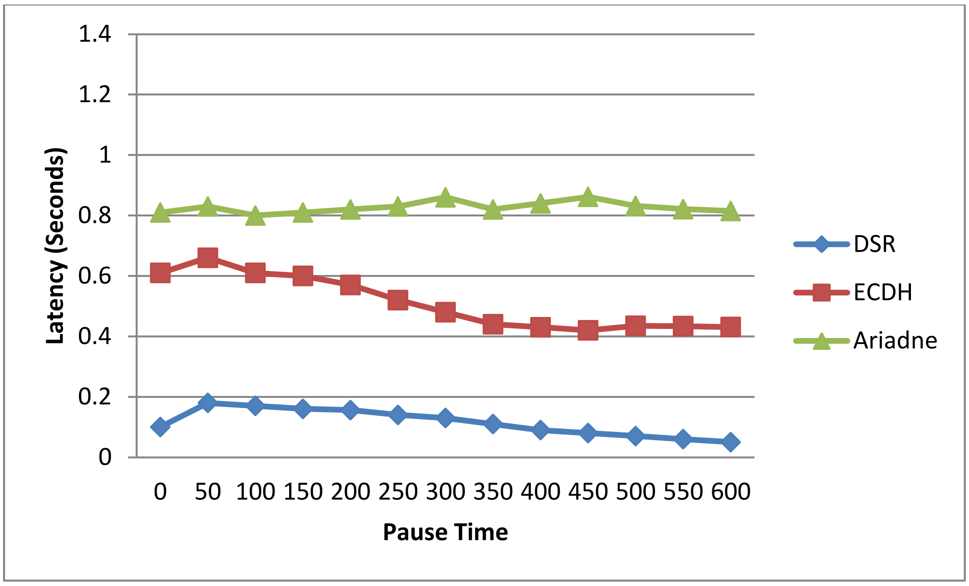


Figure 5.20 Average Latency

To summaries, EDCH, Ariadne and DSR differ in terms of their results for packet overhead, packet delivery ratio and average latency. The experiments show that DSR has the lowest packet overhead while Ariadne has the highest packet overhead. EDCH consistently has the highest packet delivery fraction. DSR shows the best results for average latency.

5.7 Summary

This chapter has focused on the comparative study and performance analysis of two important secure routing based on DSR routing protocol based on packet delivery fractions which it has been drawn in our research.

A significant part of this chapter has been devoted to the analysis of data from selective several network scenarios. The outcomes have been presented in figures. It has been shown that ECDH protocol is better in performance compared with DSR and Ariadne protocols in a normal network environment. However, ECDH protocol provides better security as it achieves better in malicious environment. Nevertheless, in handshake attacks ECDH protocol shows better results that in DSR and Ariadne protocols.

In the future MANET's denser mediums will be used with increasing applications. It can therefore be said that in terms of packet delivery, ECDH is a better choice for routing in the malicious environment. It should be noted that the research presented in Section 5.5 is ongoing as some aspects of the research are still being investigated. Further research will consider the performance of other metrics like delay, throughput, and node lifetime in wireless network environments.

CHAPTER 6: TRUST ROUTING IN MANET FOR SECURING ECDH PROTOCOL

This chapter describes a new authentication service and trust level attached in every packet to make the routing in Mobile Ad-Hoc Network secure. Efficient procedure of Mobile Ad Hoc Networks depends on suitable maintenance of routing information in a distributed network. Since the routing protocol is vulnerable from malicious mobile nodes attacks, our prime focus is on securing them. This has been achieved through the introduction of the security method for routing protocols in MANET. Our scheme which assists to achieve the authentication with minimum overheads has been developed to work better with DSR routing protocol. Not only does it prevent attacks from external intruders but it also detects misbehaviours of the wireless network nodes at the same time. The combination of the scheme in the routing protocol has guaranteed that the performance is not altered considerably.

Section 6.1 below describes trust in ad-hoc networks. The next section analyses trust in routing protocols focusing specifically on security aware protocols and trust-aware routing protocols. Section 6.3 examines trust computation in routing. Section 6.4 looks at a new method of security using trust table multi path routine. The final part of this chapter consists of a short conclusion.

6.1 Trust in Ad-hoc Network

Trust, is a featured and important part in any network environment, is defined in terms of the confident reliance of one entity on the other [94]. Trust management deals with: (a) the establishment of justification for placing trust; and (b) the modification of dependencies so as to mitigate the associated risks. Dynamic trust management is concerned with these issues under changing circumstances. Authentication is the important application of trust in network systems. Trust has the potential to solve further problems than the traditional cryptographic security. For example, Trust can help in deciding the quality of the nodes and the quality of their services, and provide the corresponding access control.

In [85, 95] the authors have proposed security systems to secure MANET. These systems use digital signatures or one way hash algorithms but totally ignoring the trust relationships models among mobile nodes in MANET. While these systems can afford more secure solutions to routing, they actually decrease the effectiveness of routing discovery. The reduction in efficiency is due to the significant time and performance consuming procedure of the complex computation in each operation [96, 97].

6.2 Trusts in routing protocols

6.2.1 Security aware routing (SAR)

Security aware routing (SAR) presents a system that combines the security levels into the routing techniques. SAR protocol classifies mobile nodes and clearly describes the trust values for every classification. According to [98], "“quality of protection” and “security attributes” to the route metrics have to be clarified. Specification is essential as some applications need not just the shortest routes but as well secure ones”. The SAR protocol depends on every on demand ad hoc routing protocols such as DSR and AODV. The SAR protocol has two major objectives: (a) discovery for the routes which include security levels and (b) protection of information passing through so that security levels cannot be modified. SAR attempts to use classical symmetric key so as to offer a higher level of security in MANET. Although the use of SAR presents with a higher level of security in MANET, attacks are still possible. [99] (2007, p.8) describes some of the drawbacks in using SAR, e.g.:

1. Some nodes misbehave provided that they have the exact key.
2. If a malicious node somehow restores the exact key, the protocol will remain open for all attacks.
3. High power consumption because the encryption and the decryption are used at every hop.

6.2.2Trust-aware Routing Protocol (TARP)

[100] proposed TARP protocol for determinable for securing the route discovery and diffusion of the trust levels and security attributes as metrics. TARP is performed over Dynamic Source Routing (DSR) protocol. Trust-Aware Routing Protocol “TARP” [100] 2006, p.135) is one of the mechanisms which focus on some factors on security mobile ad hoc network trusted availability and quality of trust, unlike other mechanisms which focus on the shortest path.

TARP has been evaluated on two important attributes: (a) the *battery power* and (b) the *software configuration*. However, the other parameters, i.e., *hardware configuration*, *credit history*, *exposure*, and *organization hierarchy*, which may affect the trust metric, have not been evaluated.

TARP mechanism has 6 steps to create a trust route between source and destination node. Firstly, the source will send N_Request to the neighbors when the source has data to send; asking for Attribute Number (ANs). The neighbors will send N_Replay including AN to the source;

this step is called “One Hop Check”. Secondly, the source will check ANs up to confirm whether they are matching or not. This is done by sending FN_Request to neighbors in different paths asking for Trust Numbers (TNs). The neighbors will replay by sending FN_Replay including TN. Thirdly; the third and the fourth steps are on demand route discovery. Finally, the fifth and the sixth steps are TARP localized route maintenance.

However, the second step is not a dynamic solution to the trust problem [101]. Indeed, for instance, if the source mobile node request is above 60% Trust Number (TN) for the credit history, power and RAM, in addition to a neighbour mobile node claimed 90% TN for credit history, 95% for power and 58% RAM, that mobile node will truly not be trusted as the neighbour mobile node requesting above 90% for credit history and 95% for power. It can be assumed that there is a failing in the TARP work which is the neighbour node. It has to use the same encryption algorithm as the source node, otherwise the packet will be dropped [102].

6.3 Trust Calculation in Routing

When we assess the experience of a trust value [103], it is essential to measure the amount of out coming packets genuinely sent by the neighbouring mobile node [103]. To understand this, we have to monitor any mobile node that participates in the packet forwarding. Monitoring can be achieved by putting every mobile node in the *promiscuous* procedure for the all time even the mobile node sending control or data packets. Upon discovering that its instant neighbour mobile nodes are transmitting the packet, the mobile node verifies packet integrity. This is done to guarantee the packet has not been changed by another malicious mobile node. If it detects that the neighbour mobile node has succeeded the integrity check, the outcome packet counter of this neighbour mobile node have to be increased. However, failure in succeeding the integrity check or in cooperating to forward the packets it is assumed to, results in its equivalent forwarding counter not to be changed. After some time, the value of experience will be very low on consideration of malicious behaviour.

Investigation of trust value knowledge uses the link layer acknowledgements that implicate the MAC protocol which gives feedback of the succeeded transmission delivery for data packets. That will enable the MAC layer to perform an easy calculation. A number of mobile nodes in MANET cannot obtain the experience and knowledge of trust vector directly, except through recommendation from others. That restriction is due to the transmission range of MANET which is usually about 100 or 200 meters. In fact, we require the trust value to be easy to broadcast among mobile nodes in the network - avoiding annoying overhead in the assessment in recommendations of the trust value.

It proposed an appropriate system that depends on route discovery procedure by expanding the trust values of mobile nodes over the Route Request packets. In fact, trust values are evaluated based on direct experience among the mobile nodes. The mobile node before transmitting the RREQ packet to the neighbours has to insert its trust value regarding the one hop neighbour mobile nodes and then broadcasted to them. Therefore, the RREQ packet will distribute the trust values. For instance, mobile node S sends a RREQ packet to mobile nodes A, B, and C. The mobile node A will attach its trust value about mobile node E to the RREQ packet and then broadcast it. Mobile node B will also attach its trust value about mobile node F to the RREQ and so on. Then the source mobile node will select the high trust values in the route, any low trust value in any route will be dropped.

The essential problem with some of routing protocols that been used to trust all mobile nodes and assume that they all behave well. However some of those mobile nodes which been trusted may be not behaving well. "Most ad hoc network routing protocols become inefficient and show dropped performance while dealing with large number of misbehaving nodes" [104] 2010, p.12).

6.4 Trust Route in DSR

We proposed a new technique of security using trust table multi path routing such as "*trust-aware routing framework for wireless sensor networks*" (TARF)[105]. This mechanism makes it hard for malicious mobile nodes to have access to the data packet. If they do, they will be dropped. We are avoiding non trustable routes that might bring force attacks. They might decrypt packets if they get access to sufficient sections of these messages.

[106] recommended the use of cryptographic mechanisms. This kind of techniques includes several complex encryption/decryption algorithms. Other mechanisms of trust have been proposed in the literature. For example, [107] tried to establish a trust management by using the concept of weight.

Our working assumption is that every mobile node makes a trust table and saves the all the trust values for its one hop mobile nodes. The trust value is set between 0 and 1. And we can assign for well behaved (normal) mobile node a trust value >= 0.5, whereas the malicious mobile node trust value can be marked < 0.5 as shown an examples of normal and malicious behavior in table 6.4. In our model, we do not consider the path trust value calculation that will reduce the overhead and also the delay. Once the DSR chooses the shortest route among the source and the destination, it will look to the lowest trust and destination. It will look to the lowest trust value in the path if its value is less than 0.5 (<0.5). This means in that path there is a malicious node. We will then go for the next path from the DSR cache as seen in Table 6.1. Figure 6.1 and Table 6.2 show an example for this mechanism. Based on the behavior (table 6.4) of the mobile node in the network it can decrease or increase the trust values by 0.1 as described in this equation:

For increment: New Trust Value = Old Trust Value + 0.1

For decrement: New Trust Value = Old Trust Value – 0.1

Trust value level	*Range*	*Action*
Low Trust Value (Malicious mobile Node)	< 0.5 to 0	Not trusted
High Trust Value (Trusted mobile Node)	>= 0.5 to 1	Trusted

Table 6.1: Trust value level

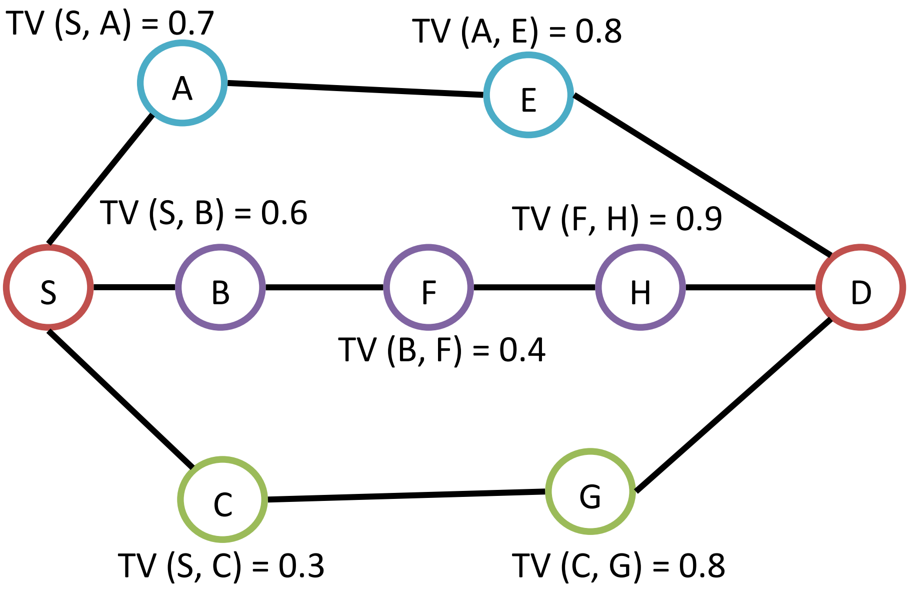


Figure 6.1: An example of the trust value mechanism

Route	*Neighbor ID*	*Trust Value*	*Trust Level*
Route 1	A	0.7	High
	E	0.5	High
Route 2	B	0.6	High
	F	0.4	Low
	H	0.9	High
Route 3	C	0.3	Low
	G	0.8	High

Table 6.2: Example of the trust value mechanism

Our trust model differs from the other models proposed so far e.g., [106-110] . Previous models make a calculation, which causes a delay and an overhead. Our algorithm is shown in Table 6.3 by adding Trust Value filed to the packet in the RREQ and RREP packets.

DSR algorithm	*Trust DSR*
Source mobile node broadcast RREQ to its neighbor RREQ: [$\text{IP}_{\text{destination}}$,$\text{IP}_{\text{source}}$,Seqnum]	Mobile node will judge on the next one hop neighbor and give a trust value on it ∪ Trust Value
Neighbor node mobile checks its routing cache for available route to destination.	
IF fresh route exit THEN reply with RREP to source node	As in RREQ will include the neighbor list with their Trust Values
ELSE, rebroadcast RREQ to the neighbor (add its IP address in the RREQ before rebroadcast)	Before rebroadcast the intermediate node will judge on the next one hop neighbors and give a trust value
Source node waits for more that RREP from the destination RREP:[$\text{IP}_{\text{sourse}}$,$\text{IP}_{\text{destination}}$,Seqnum]	∪ Trust Value
Then, the source will choose the shortest route to the destination	And also based on the trust value, IF, there is a low trust value in the path. THEN, eliminate that path and go to the next shortest path.

Table 6.3: Trust DSR and DSR algorithm

The first step for trusting the nodes is in the Route Discovery figures, as described above. Secondly, while sending the data to the target node, the trust value will increase or decrease based on trustworthiness evaluation.

For instance, CONFIDANT [111] , CORE "*Collaborative Reputation Mechanism to enforce node cooperation in Mobile Ad hoc Networks*" [112] and OCEAN [113] protocols check and evaluate the trustworthiness of the one hop mobile nodes and eliminate any distrust mobile nodes from broadcasting packets. Examples of ways to increase or decrease the trust value are shown in Table 6.4 below.

Increment (+.01)	*Decrement (-0.1)*
common leaving from the network	Uncommon leaving from the network
Standard joining	Unusual joining
Higher Power availability	Lower power availability
Higher bandwidth availability	Lower bandwidth availability

Table 6.4: Examples of ways to increase or decrease the trust value

Our focus will be on direct trust relationship between two nodes. Most of the reputation systems rely on reputation values like a metric of trust. But these present with some drawbacks [114] (2007, p.6-9):

1. Huge caching: Every mobile node maintains public reputation values; consequently saving this information requires a massive caching.
2. Increase in volume of network traffic due to dissemination of reputation information.
3. These reputations information might be modify, replied, forgery or lost by fraud mobile node transmission between the source and mobile nodes in the network.

4. Creation of additional problems upon assignment of an initial reputation value as a new mobile node accesses the wireless network or a mobile node goes to a different location, e.g., time needed at this point to trust the neighbor mobile nodes before they access the wireless network.
5. Inconsistency in reputation due to the fact that in reputation systems, a mobile node might have two or more reputation values based on other mobile node reputation values.

All the mobile nodes over the route of the packet will participate to learn and detect all kinds of changing behavior of the neighbors.

The packet forwarding route of neighbors enables classification of trust metric. Forwarding average of the neighbors is checked and registered for each time, as shown below.

$$Forwarding\ Rate\ = \frac{packet\ forwarded\ by\ node\ B}{packet\ sent\ by\ node\ A\ to\ node\ B}$$

Route Request (RREQ)

<table>
<tr><td>Option type</td><td>Option data length</td><td>Identification</td></tr>
<tr><td colspan="3">Target address</td></tr>
<tr><td colspan="3">Address [1]</td></tr>
<tr><td colspan="3">Address [2]</td></tr>
<tr><td colspan="3">….</td></tr>
<tr><td colspan="3">Address [n]</td></tr>
<tr><td colspan="3">Neighbor List</td></tr>
<tr><td colspan="3">Trust Value</td></tr>
</table>

Figure 6.2: Trust DSR data packet header format (RREQ)

Route Reply (RREP)

	Option type	Option data length	L	Reserved
Address [1]				
Address [2]				
….				
Address [n]				
Neighbor List				
Trust Value				

Figure 6.3: Trust DSR data packet header format (RREP)

6.5 Trust Models in MANETS

The general term “misbehavior” has sometimes been used to describe attacks in MANETs. Indeed, some researchers, for example [115, 116] , have not defined the specific selfish behavior in their mechanisms but have preferred the broad term “misbehavior” to define any kind of attack.

There are different kinds of trust in MANET such as Reputation based model and credit based model. Examples of reputation based models are CONFIDANT[111], CORE[112], and OCEAN[113]. These kinds of mechanisms are based on DSR routing protocol without cryptographic authentication. The assumption in these models is that all mobile nodes in the network are not malicious. Reputations systems are based on the following two types: (1) mobile nodes monitoring other neighbors directly, and (2) other mobile nodes monitoring other neighbors.

A model proposed by Liu et al. (2004) defined and maintained a dynamic trust relationship using trust value between mobile nodes. This is based on some important assumptions. These assumptions deploy processes and intrusion detection systems (IDS) that: (1) detect the malicious mobile node, and (2) report that to other mobile nodes in the network.

Another model proposed by Davis (2004) used a kind of trust management mechanism which depends on a structure hierarchical trust model. Every mobile node in the network must provide an active digital certificate to be trusted. Once this is achieved, it will sum up all weighted accusations. If the result is bigger than a predefined threshold, the mobile node will cancel the certificate which goes against MANET's nature.

Some trust models shared recommendations among mobile nodes to create a reputation database. However, these kinds of models faced some problems such as a large network overhead because of the reputation information exchanged among the mobile nodes and also addressing the possible for malicious recommendations needs to be trusted by trust third party or public key infrastructure.

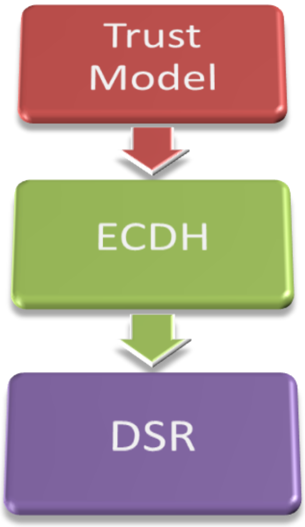


Figure 6.4: Trust Model and ECDH on DSR routing protocol

Our trust model, as mentioned in 6.3, will be built on ECDH routing protocol. This will provide better security against selfish mobile nodes in the MANETs. However, some mechanisms assume that all mobile nodes in the network are good while others support the trust values after the authentication has been made between the mobile nodes in the network. By adding the trust model to the ECDH, it will eliminate the routes that contain selfish nodes between the source and destination mobile nodes. In our experiments, the word “Malicious” is defined as a bad behavior performed by selfish (because of the bandwidth preservation or the power consumption) or erroneous mobile nodes. These nodes do not forward packets, drop packets, lead to hardware failures or incorrect software. The section below describes our experiments. Section 6.8 presents the results for these experiments in low malicious nodes and high malicious nodes.

6.5.1 CONFIDDENCE

Cooperation of Nodes, Fairness in Dynamic Ad-hoc NeTwork is a protocol which proposed by [111]. It is based on direct trust and indirect trust from other mobile nodes sharing behaviour information which is updated by *Bayesian* estimation.

This technique contains four components: 1) the *Monitor*, 2) the *Reputation System*, 3) the *Path Manger*, and 4) *Trust Manager*. Moreover, these components are shown in every mobile node in the network as shown in Figure 6.5.

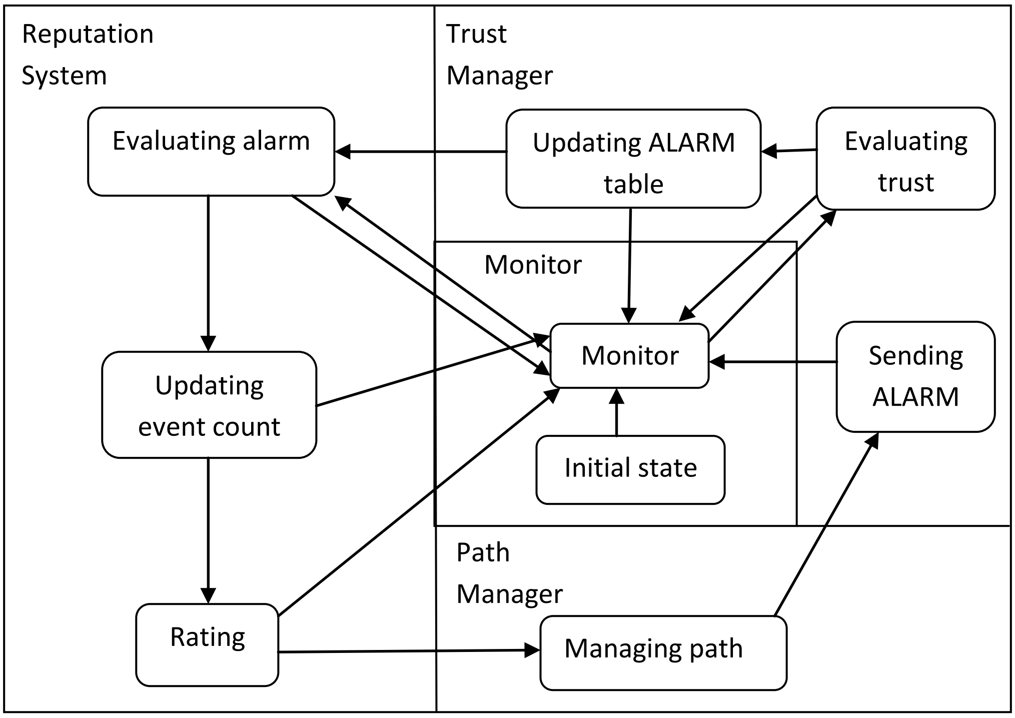


Figure 6.5: CONFIDANT architecture

6.6 TV-ECDH Experiment in MANETs

6.6.1 Goals

Our goal is to evaluate out trust model in the lack of the security in the routing protocols, so as to understand its performance against the attacks. We have been focused on the performance in terms of packets delivery ratio, overhead, average latency and malicious mobile nodes detected in the network.

6.6.2 Simulation Setup

The metrics has been used in two network scenarios providing the modifications in DSR, TV-ECDH and CONFIDANT routing protocols. The first scenario contains a low number of malicious mobile nodes and in the second scenario contains a high number of malicious mobile nodes in the network. In all scenarios DSR routing protocol is used as a reference. Indeed, CONFIDANT has been evaluated to distinguish the difference of security that has been used in TV-ECDH routing protocol in MANET.

The simulator has been implemented on Network Simulator 2 (NS2.34)[92], as it has been used in Chapter 5. The experiments in this chapter were run 6 times. In addition, the confidence level of the intervals is 93.40%.

6.6.3 Parameters

The set of parameters for the simulations are shown in the table 6.2.

Parameter	Value
Area	900m x 900m
Speed	5 m/s
Radio Range	250 m
Movement	Random waypoint model
MAC	802.11
Application	CBR
Packet Size	512 Bytes
Simulation Time	800s
Number of Nodes	50 nodes
Number of Malicious nodes	5 nodes (10%), 25 nodes (50%)
Pause Time	100 s
Node Connections	20
Simulation runs	6 times

Table 6.5: Parameters

6.6.4 Performance Metrics

We have simulated the experiment in NS-2. Each node in our simulation moves is based on Random Waypoint model and works in the following way: the node starts at a random position; waits for duration called pause time, and then chooses a new random location and moves there with a speed 5 m/s. We have used a space size 900m X 900m to increase the number of hops in routes used relative to a square space. The average values of these 6 simulation runs are then calculated for the three levels of malicious environment. The aim of our experiment is to detect the malicious nodes and eliminate that route which contains a malicious mobile node. All protocols were run on identical movement and communication scenarios as presented in Table 6.5. We computed four metrics for each simulations run, as explained below:

1. *Packet Delivery Ratio (PDR)*: the total fraction of application level data packet sent that was actually received at the intended destination node.
2. *Packet Overhead*: the number of transmissions of routing packets; for instance, a ROUTE REPLY sent over three hops would count as three packets in this metric.
3. *Average Latency*: the average time elapsed from when a data packet is first sent to when it is first received at its destination.
4. Malicious mobile nodes detected in the MANET.

6.7 Simulation Results

6.7.1 Simulation results for 5 malicious nodes (low)

Figure 6.6 shows the packet delivery ratio (PDR) for DSR, CONFIDANT and TV-ECDH. The delivery packet ratio in DSR is the lowest because no authentication got in the protocol. On the other hand, the packet delivery ratio for CONFIDANT is slightly higher than those for TV-ECDH and DSR in low mobility. However, TV-ECDH has a better result in high mobility. That’s because the TV-ECDH routing protocol is effective in discovering and maintaining routes for delivering packets, even with high mobility.

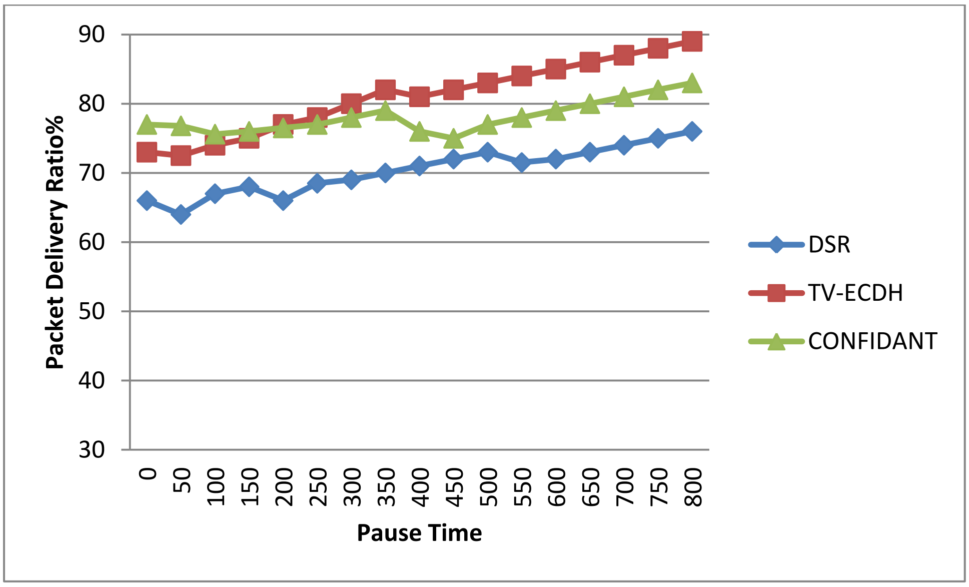


Figure 6.6: Packet Delivery Rate (PDR) (in Low malicious nodes)

Figure 6.7 illustrates the packet overhead. As illustrated by the results, TV-ECDH always gets lower packet overhead during the pause time than CONFIDANT. TV-ECDH is close to DSR routing protocol in high and almost low mobility. The main cause for getting better performance in TV-ECDH routing protocol is the routing decisions based on trust evaluations.

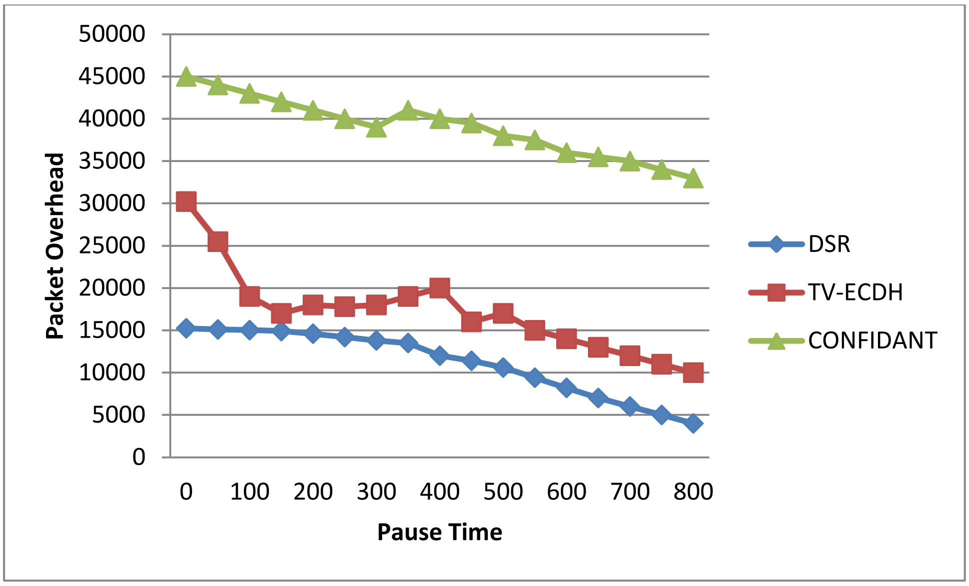


Figure 6.7: Packet Overhead (in Low malicious nodes)

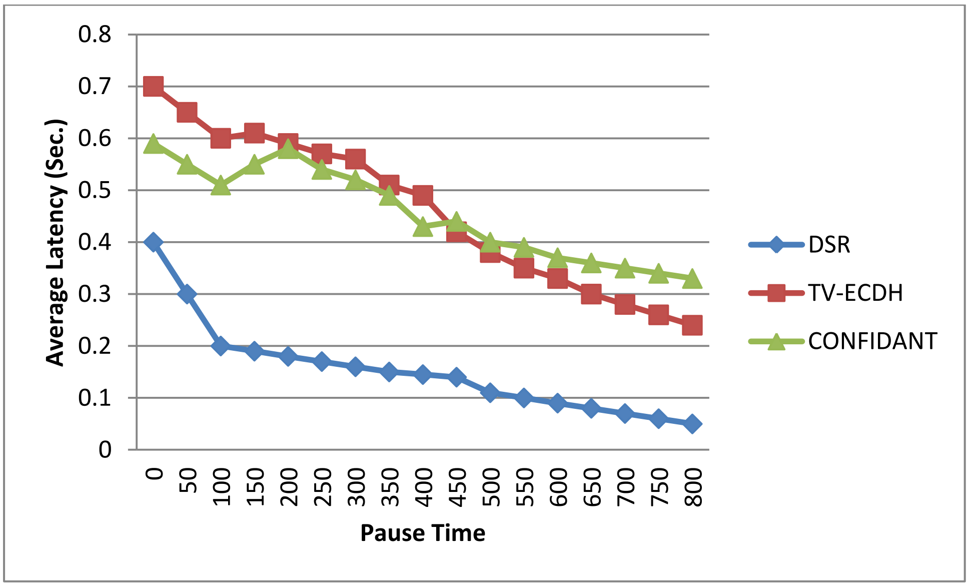


Figure 6.8: Average Latency (in Low malicious nodes)

Figure 6.8 illustrates the results for average latency in low malicious nodes. As shown, the average latency in low malicious mobile nodes in TV-ECDH and CONFIDANT are almost the same. It is to be noted that TV-ECDH slightly outperforms CONFIDANT in high mobility. However, in low mobility CONFIDANT outperforms TV-ECDH. TV-

ECDH in high mobility is able to reduce the average latency by forwarding more packets in less time.

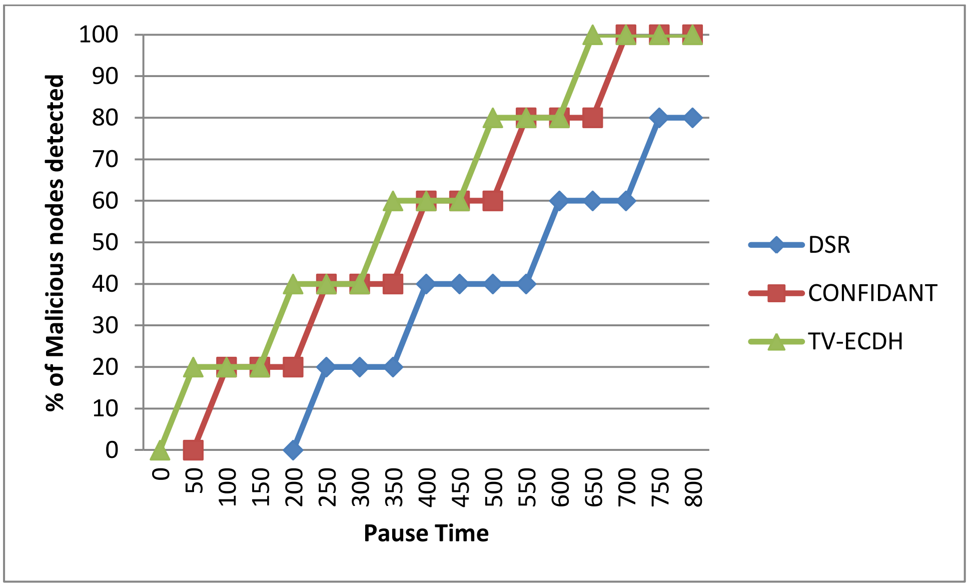


Figure 6.9: Malicious mobile node detecting (in Low malicious nodes)

Figure 6.9 presents the findings for the fourth metric tested: detection of malicious nodes. Figure 5.8 shows the percentage of detecting malicious mobile node for DSR, CONFIDANT and TV-ECDH in low malicious mobile nodes. The results show that TV-ECDH is a little faster to detect malicious nodes than both CONFIDANT and DSR. DSR cannot detect malicious node properly but according to the ROUTE ERRORs the DSR eliminate that route which has the error from the source cache.

In the next section of experiment, it presents the simulation results in high malicious nodes.

6.7.2 Simulation results for 25 malicious nodes (high)

Figure 6.10 shows the PDR in high malicious node environment. This environment is understood as having 50% of mobile nodes in the network as malicious. As shown by the results, DSR got a lower PDR compared to CONFIDANT and TV-ECDH. TV-ECDH has the best packet delivery ratio of all three. Indeed, TV-ECDH outperforms CONFIDANT and DSR in both low and high levels of mobility. CONFIDANT's performance can be explained by the fact that it faced *Bad Mouthing Attacks*. These provide bad or wrong recommendation of other nodes.

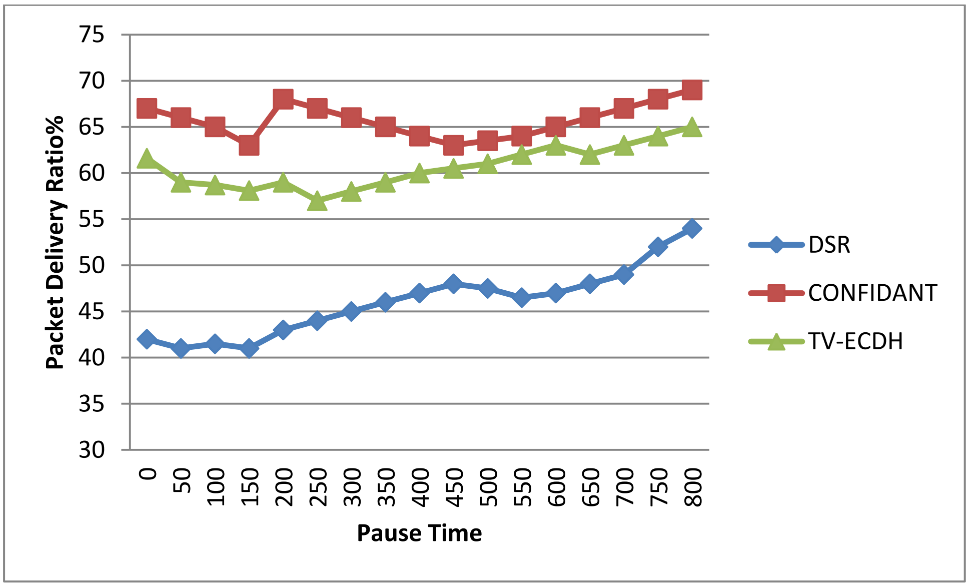


Figure 6.10: Packet Delivery Ratio (PDR) (in High malicious node)

Figure 6.11 shows the packet overhead in high malicious mobile nodes in MANET for DSR, CONFIDANT and TV-ECDH. In the beginning of the scenario DSR performs slightly better than both TV-ECDH and CONFIDANT. But after 200 seconds TV-ECDH outperforms both DSR and CONFIDANT. CONFIDANT consistently has the highest packet overhead in high malicious nodes.

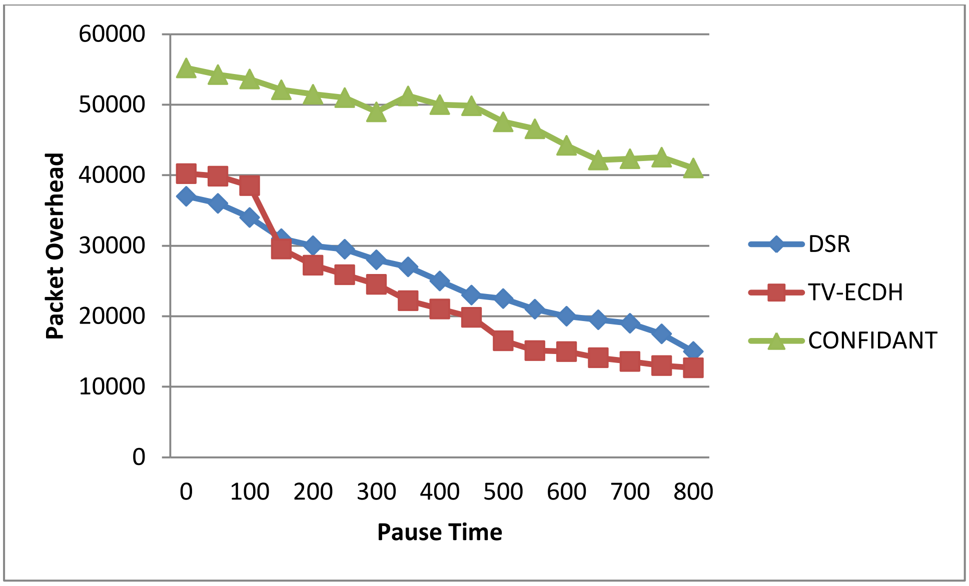


Figure 6.11: Packet Overhead (in High malicious node)

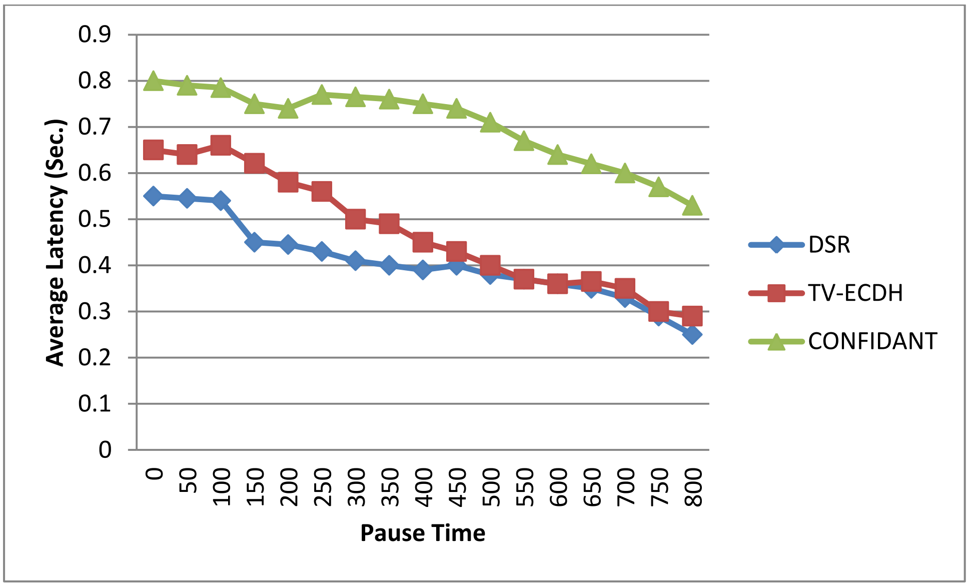


Figure 6.12: Average Latency (in High malicious nodes)

Figure 6.12 shows the results for average latency in high malicious nodes. It can be observed that in terms of average latency, TV-ECDH outperforms CONFIDANT at all pause times and has the same average latency values as DSR in high mobility. However, DSR outperforms both TV-ECDH and CONFIDANT.

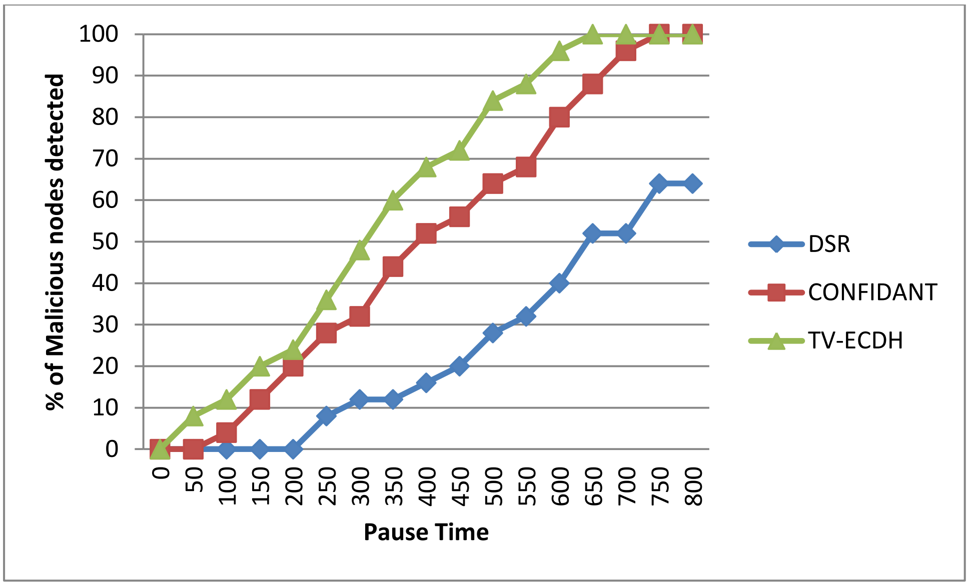


Figure 6.13: Malicious mobile node detecting (in High malicious nodes)

Figure 6.13 shows the percentage of malicious mobile nodes detection with 50% malicious for DSR, CONFIDANT and TV-ECDH. DSR is the slowest to detect malicious mobile nodes. It should be noted that DSR did not detect all the malicious mobile nodes in the experiment. This can be compared to the performance of TV-ECDH and CONFIDANT. TV-ECDH gives the best results in that it can detect malicious nodes faster than CONFIDANT. However, by the end of the experiment, both TV-ECDH and CONFIDANT had detected all the malicious mobile nodes – unlike DSR.

Figure 6.14 shows only an example of TV-ECDH for the trust values for 50 mobile nodes with 50% malicious mobile nodes just to show how the trust values of the mobile nodes vary with the tasks and their performance in our experiment in MANET. It is an initial Trust Values when the experiment starts the simulation. However, these Trust Values change during the simulation time.

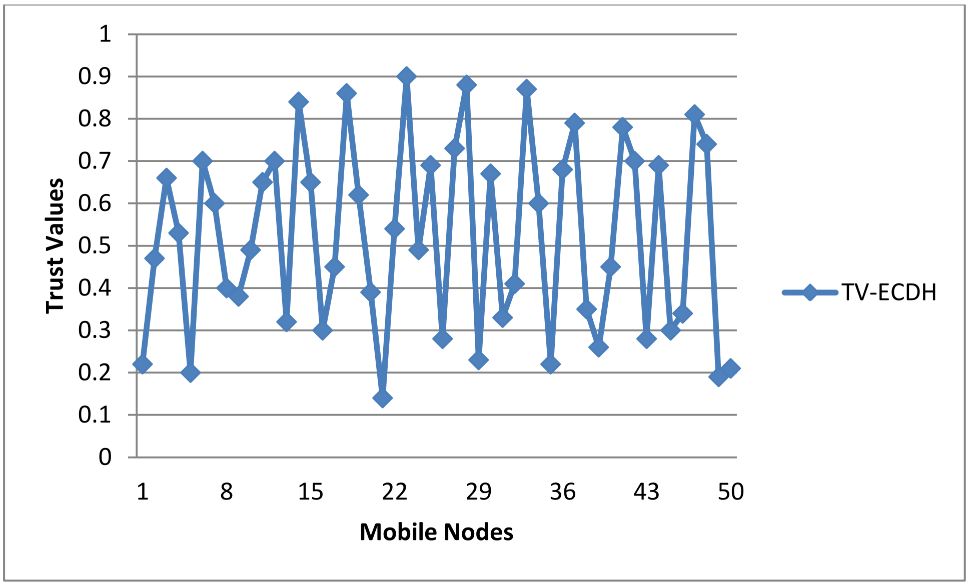


Figure 6.14: Trust Values for 50 nodes (in High malicious nodes)

6.8 Summary

This chapter has explained and focused in detail trust routing in MANET. The concept of trust value can be part of the key management subsystems to implement more flexible and self organized scheme. As a result, the most significant contribution is that any mobile node can issue the certificate of authentication by evaluating the trustworthiness of neighbours. In this chapter, we have put forward a new trust model based on ECDH and has been devoted to the analysis of data from self created network scenario. The outcomes have been presented in figures. Clearly, it has been shown that TV-ECDH is outperform in detecting selfish mobile nodes compared to DSR and CONFIDANT protocols in low (10%) and high (50%) malicious mobile nodes in the network.

Our design enables increase in performance of securing routing information without decrease in security through trust relationships which reduce unnecessary calculations. TV-ECDH model can also catch any kind of selfish behaviour made by any kinds of selfish or erroneous mobile nodes, eliminate that node from the route table, trust the route to that node or drop it from the network by adding our trust value to ECDH protocol which has been implemented in chapter 4 to increase the security in MANET by eliminating them from the routes between the mobile nodes.

CHAPTER 7: CONCLUSION AND DIRECTIONS FOR FURTHER RESEARCH

7.1 Conclusion

As mobile technologies become more and more important, security also becomes a major issue. As technologies develop and improve, so do the types of attacks. As shown in the previous chapters, attacks can be varied and sophisticated and can have disastrous consequences. It is essential to have security measures that can detect and prevent such attacks.

Security routing protocol in (MANET) has become one of the main challenges faced by researchers. Indeed, It is hard to implement security routing protocols in MANETs because of the absence of infrastructure, the limitation of resources (such as power, bandwidth) and the dynamic topology changing.

The variable and diverse nature of attacks also makes it challenging for researchers to devise routing protocols. Indeed, as explained in the earlier chapters, attacks facing the routing protocol can range from active to passive attacks. In passive attacks, the attacker listens to the channel and packets without disturbing the operations of the network. However, in active attacks, the attacker disturbs the operations of the network by modifications, fabrications or alterations. Security routing protocols, therefore, have to be devised to deal with a range of attacks.

In this thesis, we present two solutions to secure the routing protocols in DSR: Elliptic Curve Diffie-Hellman (ECDH) and Trust Routine based on DSR protocol. Our proposed solutions of the key exchange Elliptic Curve Diffie-Hellman (ECDH) work well with DSR routing protocol.

Many secure routing protocols, such as Ariadne, has a very complicated key exchange. These cause a lack of power and bandwidth.

Our contribution is a) to provide trustworthy communications among the mobile nodes in the network, b) to encourage untrustworthy mobile nodes to be trustable, and c) to discourage untrustworthy mobile nodes from participating in the network to gain services.

Our implementation of the ECDH key exchange, as described in earlier chapters, offers three major advantages compared to other secure routing protocols: (a) it is significantly faster, (b) it has small key size, (c) it is more energy efficient, that is, it consumes less energy than the other secure routing protocols. This is an attractive solution as an attacker might capture the public keys P_A and P_B, but the attacker cannot be able to conclude the private key n_A and n_B from P_A and P_B.

We have also proposed the solution for the Trust routing based on DSR protocol. This design also works in a very efficient manner: it increases the security routing performance through trust relationship that reduce the calculations. Some trust models present their trust relationship based on credit history from previous networks such TARP routing protocol. This may affect the trust evaluation. Putting it all together, our results confirm the high performance which has been achieved and simulated on NS2 simulator.

Our proposed solutions have been simulated and evaluated therefore, fare better than the existing secure routing protocols which have been compared in the previous chapters. We have put forward two proposals that not only detect attacks but also protect networks. Our solutions also take efficiency into account: they are fast using small key and save energy consumption.

7.2 Future Work

In the coming years, progress in mobile technologies will happen at a very fast rate. Security measures will have to match the pace of progress in the field. It is very likely that the types and nature of attacks will also develop quickly. There will also be a need for new security routing protocols,

Our scheme, through the proposal is specialized in three most important fast, small key size and energy-efficient solutions, seeks to contribute to the area of security in mobile networks. However, further work is needed in the field. Future designs have to be even more efficient. The steps below have to be followed to ensure a good secure routing protocol design:

(1) Implement a group of attacks against ECDH and Trust model;

(2) calculate the power consumption for ECDH;

(3) Improve and extend our proposed design such as End-to-End Quality of Service (QoS) and power efficient protocol provisions;

(4) Explore other areas in Mobile Ad Hoc Network such as MAC layer and;

(5) Explore other areas in wireless networking, such as MAC layer issues and location.

The constant update and improvement of secure routing protocols should ensure that networks are always protected against malicious attacks and can also face the challenge of new attacks. There will, therefore, always be a need for more research and improvement in security measures in wireless networks.

Bibilography

1 Crow, B.P., Widjaja, I., Kim, L., and Sakai, P.T.: ‘IEEE 802.11 wireless local area networks’, Communications Magazine, IEEE, 1997, 35, (9), pp. 116-126

2 Macker, J.C., S.: ‘Mobile Ad hoc Networking (MANET): Routing Protocol Performance Issues and Evaluation Considerations’, 1999

3 Mishra, A., and Nadkarni, K.M.: ‘Security in wireless ad hoc networks’, in Mohammad, I., and Richard, C.D. (Eds.): ‘The handbook of ad hoc wireless networks’ (CRC Press, Inc., 2003), pp. 499-549

4 Kessler, G.C.: ‘An Overview of Cryptography’, 2010

5 Mastorakis, N.E.: ‘Recent researches in communications and IT : proceedings of the 15th WSEAS International conference on communications, part of the 15th WSEAS CSCC multiconference, proceedings of the 5th International conference on communications and information technology (CIT '11), Corfu Island, Greece, July 14-17, 2011’ (WSEAS, 2011. 2011)

6 Hu, Y.C., Perrig, A., and Johnson, D.B.: ‘Ariadne: A Secure On-Demand Routing Protocol for Ad Hoc Networks’, 2002

7 Hu, Y.C., Perrig, A., and Johnson, D.B.: ‘Ariadne: A secure on-demand routing protocol for ad hoc networks’, Wireless Networks, 2005, 11, (1-2), pp. 21-38

8 Deng, H., Li, W., and Agrawal, D.P.: ‘Routing security in wireless ad hoc networks’, Communications Magazine, IEEE, 2002, 40, (10), pp. 70-75

9 Wood, A.D., and Stankovic, J.A.: ‘Denial of service in sensor networks’, Computer, 2002, 35, (10), pp. 54-62

10 Ilyas, M.: ‘The handbook of ad hoc wireless networks’ (CRC, 2002. 2002)

11 Hoebeke, J., Moerman, I., Dhoedt, B., and Demeester, P.: ‘An overview of mobile ad hoc networks: applications and challenges’, JOURNAL-COMMUNICATIONS NETWORK, 2004, 3, (3), pp. 60-66

12 Xiao, Y., Shen, X., and Du, D.: ‘Wireless network security’ (Springer, 2007. 2007)

13 Misra, S., Woungang, I., and Misra, S.C.: ‘Guide to wireless ad hoc networks’ (Springer, 2009. 2009)

14 Stallings, W.: ‘Network Security Essentials: Applications and Standards, 4/e’ (Pearson Education India, 2003. 2003)

15 Murthy, C.S.R., and Manoj, B.: ‘Ad hoc wireless networks: Architectures and protocols’ (Prentice Hall, 2004. 2004)

16 Ilyas, M., and Mahgoub, I.: ‘Mobile computing handbook’ (CRC, 2004. 2004)

17 Razak, S.A., Furnell, S., and Brooke, P.: ‘Attacks against mobile ad hoc networks routing protocols’, ‘Book Attacks against mobile ad hoc networks routing protocols’ (2004, edn.), pp.

18 Anjum, F., and Mouchtaris, P.: ‘Security for wireless ad hoc networks’ (Wiley-Interscience, 2007. 2007)

19 Padmavathi, D.G., and Shanmugapriya, M.: ‘A survey of attacks, security mechanisms and challenges in wireless sensor networks’, arXiv preprint arXiv:0909.0576, 2009

20 Goyal, P., Batra, S., and Singh, A.: ‘A literature review of security attack in mobile ad-hoc networks’, International Journal of Computer Applications IJCA, 2010, 9, (12), pp. 24-28

21 Wu, B., Chen, J., Wu, J., and Cardei, M.: ‘A survey of attacks and countermeasures in mobile ad hoc networks’, Wireless Network Security, 2007, pp. 103-135

22 Huang, Y., and Lee, W.: ‘Attack analysis and detection for ad hoc routing protocols’, ‘Book Attack analysis and detection for ad hoc routing protocols’ (Springer, 2004, edn.), pp. 125-145

23 Sanzgiri, K., Dahill, B., Levine, B.N., Shields, C., and Belding-Royer, E.M.: ‘A secure routing protocol for ad hoc networks’, ‘Book A secure routing protocol for ad hoc networks’ (IEEE, 2002, edn.), pp. 78-87

24 Xing, F., and Wang, W.: ‘Understanding dynamic denial of service attacks in mobile ad hoc networks’, ‘Book Understanding

dynamic denial of service attacks in mobile ad hoc networks' (IEEE, 2006, edn.), pp. 1-7

25 Hu, Y.C., Perrig, A., and Johnson, D.B.: 'Packet leashes: a defense against wormhole attacks in wireless networks', 'Book Packet leashes: a defense against wormhole attacks in wireless networks' (IEEE, 2003, edn.), pp. 1976-1986

26 Perrig, A., Canetti, R., Song, D., and Tygar, J.: 'Efficient and secure source authentication for multicast', 'Book Efficient and secure source authentication for multicast' (2001, edn.), pp. 35-46

27 Wuthnow, M., Shih, J., and Stafford, M.: 'IMS: A New Model for Blending Applications' (Auerbach Publications, 2009. 2009)

28 Perkins, C.E.: 'Ad hoc networking' (Addison-Wesley, 2001. 2001)

29 Johnson, D.B., and Maltz, D.A.: 'Dynamic source routing in ad hoc wireless networks', Mobile computing, 1996, pp. 153-181

30 Johnson, D.B.: 'The dynamic source routing protocol for mobile ad hoc networks', draft-ietf-manet-dsr-09. txt, 2003

31 Das, S.R., Belding-Royer, E.M., and Perkins, C.E.: 'Ad hoc on-demand distance vector (AODV) routing', 2003

32 Perkins, C.E., and Royer, E.M.: 'Ad-hoc on-demand distance vector routing', 'Book Ad-hoc on-demand distance vector routing' (IEEE, 1999, edn.), pp. 90-100

33 Johnson, D., Hu, Y., and Maltz, D.: ‘The dynamic source routing protocol (DSR) for mobile ad hoc networks for IPv4’, RFC4728, 2007, pp. 2-100

34 Hu, Y.C., and Johnson, D.B.: ‘Implicit source routes for on-demand ad hoc network routing’, ‘Book Implicit source routes for on-demand ad hoc network routing’ (ACM, 2001, edn.), pp. 1-10

35 Harris, S.: ‘CISSP Certification All-in-One Exam Guide, Emeryville’, ‘Book CISSP Certification All-in-One Exam Guide, Emeryville’ (CA: McGraw-Hill/Osborne, 2003, edn.), pp.

36 William, S.: ‘Cryptography and Network Security, 4/e’ (Pearson Education India, 2006. 2006)

37 Schneier, B.: ‘Applied cryptography : protocols, algorithms and source code in C’ (Wiley, 1996, 2nd ed. edn. 1996)

38 Kahate, A.: ‘Cryptography and network security’ (Tata McGraw-Hill Education, 2003. 2003)

39 Menezes, A.J., Van Oorschot, P.C., and Vanstone, S.A.: ‘Handbook of applied cryptography’ (CRC, 1996. 1996)

40 Nagpal, R.: ‘Ecommerce - Legal Issues’, 2008

41 Zimmermann, P.R.: ‘The official PGP user's guide’ (1995. 1995)

42 Patel, D.R.: ‘Information Security Theory and Practice’, ‘Book Information Security Theory and Practice’ (Prentice-Hall of India Private Limited, 2008, edn.), pp.

43 Drira, K., Martelli, A., and Villemur, T.: ‘Cooperative Environments for Distributed Systems Engineering: The Distributed Systems Environment Report’ (Springer, 2002. 2002)

44 Jeong, J., Chung, M.Y., and Choo, H.: ‘Integrated OTP-based user authentication scheme using smart cards in home networks’, ‘Book Integrated OTP-based user authentication scheme using smart cards in home networks’ (IEEE, 2008, edn.), pp. 294-294

45 Wu, B., Wu, J., Fernandez, E.B., Ilyas, M., and Magliveras, S.: ‘Secure and efficient key management in mobile ad hoc networks’, Journal of Network and Computer Applications, 2007, 30, (3), pp. 937-954

46 Zhang, Y., Zheng, J., and Ma, M.: ‘Handbook of research on wireless security’ (Information Science Reference-Imprint of: IGI Publishing, 2008. 2008)

47 Ferilli, S.: ‘Automatic digital document processing and management : problems, algorithms and techniques’ (Springer, 2011. 2011)

48 Bagad, V.S.a.D., I. A.: ‘Information Security ’, 2009, pp. 1-24

49 Bagad, I.A.D.V.S.: ‘Cryptography And Network Security’ (Technical Publications, 2008. 2008)

50 Lehane, B., Doyle, L., and O'Mahony, D.: ‘Shared RSA key generation in a mobile ad hoc network’, ‘Book Shared RSA key generation in a mobile ad hoc network’ (IEEE, 2003, edn.), pp. 814-819

51 Wu, B., Wu, J., and Fern, E.B.: 'Secure and Efficient Key Management in Mobile Ad Hoc Networks', 2005

52 Schneier, B.: 'Applied cryptography: Protocols, algorithms, and source code in C', 'Book Applied cryptography: Protocols, algorithms, and source code in C' (Wiley (New York), 1996, edn.), pp.

53 Nechvatal, J., Barker, E., Bassham, L., Burr, W., and Dworkin, M.: 'Report on the development of the Advanced Encryption Standard (AES)', 'Book Report on the development of the Advanced Encryption Standard (AES)' (DTIC Document, 2000, edn.), pp.

54 Stallings, W.: 'Cryptography and network security : principles and practice' (Prentice Hall, 2011, 5th edn. 2011)

55 Farrell, S., Kause, T., and Mononen, T.: 'Internet X. 509 Public Key Infrastructure Certificate Management Protocol (CMP): IETF RFC 2510', 'Book Internet X. 509 Public Key Infrastructure Certificate Management Protocol (CMP): IETF RFC 2510' (IETF Network working Group, 2005, edn.), pp.

56 Gibson, J.: 'Discrete logarithm hash function that is collision free and one way', Computers and Digital Techniques, IEE Proceedings E, 1991, 138, (6), pp. 407-410

57 Markovski, S., Gligoroski, D., and Bakeva, V.: 'Quasigroups and Hash Functions', 6th ICDMA, Bankso, 2001, pp. 43-50

58 Kaliski, B.: 'The MD2 message-digest algorithm', RFC 1319, Internet Activites Board, Internet Privacy Task Force ,April 1992, available at: http://tools.ietf.org/html/rfc1319.html

59 Rivest, R.L.: 'THE MD4 MESSAGE DIGEST ALGORITHM', Lecture Notes in Computer Science, 1991, 537, pp. 0303, available at : http://tools.ietf.org/html/rfc1320.html

60 Rivest, R.: 'The MD4 Message-Digest Algorithm', IETF Netwrok Working Group, RFC 1320' (MIT and RSA Data Security, Inc, 1992), available at: http://tools.ietf.org/html/rfc1320.html

61 Rivest, R.: 'The MD5 message-digest algorithm', IETF Netwrok Working Group, RFC 1321, April 1992, available at: http://tools.ietf.org/html/rfc1321.html

62 Dobbertin, H., Bosselaers, A., and Preneel, B.: 'RIPEMD-160: A strengthened version of RIPEMD', Fast Software Encryption, LNCS 1039, Springer-Verlag, April 1996, pp. 71-82

63 NIST, F.P.U.B.: '180-1: Secure Hash Standard', Federal Information Processing Standards, National Bureau of standards, U.S. department of commerce, Washington D.C. ,April 1995.

64 Pub, NIST Fips.: '180-2: Secure Hash Standard (SHS)', Technical report, National Institute of Standards and Technology, US Department of Commerce, DRAFT, 2004

65 Barker, E., Barker, W., Burr, W., Polk, W., and Smid, M.: 'NIST Special Publication 800-57', NIST Special Publication, 2007, 800, (57), pp. 1-142

66 Hadjichristofi, G.C., Adams, W.J., and Davis IV, N.J.: 'A framework for key management in mobile ad hoc networks', 'Book A

framework for key management in mobile ad hoc networks' (IEEE, 2005, edn.), pp. 568-573

67 Al-Kayali, A.K.M.: 'Elliptic Curve Cryptography and Smart Cards', GIAC Security Essentials Certification (GSEC), SANS Institute Reading Room Site, February 2004, available at: http://www.sans.org/reading_room/whitepapers/vpns/elliptic-curve-cryptography-smart-cards_1378

68 Anoop, M.: 'Elliptic Curve Cryptography: An Implementation Tutorial', Tata Elxsi Ltd, Thiruvananthapuram, India, 2007, available at: http://hosteddocs.ittoolsbox.com/AN1.5.07.pdf

69 Brown, D.: 'Standards for Efficient Cryptography 1 (SEC 1)', Standards for Efficient Cryptography, Certicom Research, 2009, available at: http://www.secg.org/secg-docs.html

70 Connell, I.: 'Elliptic curve handbook', McGill University, Montreal, 1996, available at: http://www.math.mcgill.ca/connell/public/ECH1/

71 Constantinescu, N.: 'Elliptic curve cryptosystems and scalar multiplication', Annals of the University of Craiova-Mathematics and Computer Science Series, 2010, pp. 27-34

72 Dahill, B., Levine, B., Royer, E., and Shields, C.: 'ARAN: A secure Routing Protocolfor Ad Hoc Networks', (UMass Tech Report 02-21), 2002

73 Darade, P.L.: 'Knapsack Based ECC with Encryption and Decryption', IJETAE, April 2012

74 Diffie, W., and Hellman, M.: ‘New directions in cryptography’, Information Theory, IEEE Transactions on, 1976, pp. 644-654

75 Hua, A., and Chang, S.L.: ‘Proceedings of the 9th International Conference on Algorithms and Architectures for Parallel Processing’, ICA3PP, Taipei, Taiwan, Springer , June 2009

76 Johnson, D., Menezes, A., and Vanstone, S.: ‘The elliptic curve digital signature algorithm (ECDSA)’, International Journal of Information Security, 2001, pp. 36-63

77 Kumar, R., and Anil, A.: ‘Implementation of Elliptical Curve Cryptography’, International Journal of Computer Science Issues, July 2011

78 Laurendeau, C., and Barbeau, M.: ‘Threats to Security in DSRC/WAVE’, Ad-Hoc, Mobile, and Wireless Networks, 2006, pp. 266-279

79 Marti, S., Giuli, T.J., Lai, K., and Baker, M.: ‘Mitigating routing misbehavior in mobile ad hoc networks’, 6th Annual ACM/IEEE International Conference on Mobile Computing and Networking, 2000, pp. 255-265

80 Miller, V.: ‘Use of elliptic curves in cryptography’, In Advances in Cryptology- CRYPTO '85, Springer, 1986, pp. 417-426

81 Papadimitratos, P., and Haas, Z.J.: ‘Secure routing for mobile ad hoc networks’, Proceedings of the SCS Communication Netwroks and Distributed Systems Modeling and Simulation Conference (CNDS 2002) January 2002, pp. 193-204

82 Rabah, K.: ‘Theory and Implementation of Elliptic Curve Cryptography’, Journal of Applied Sciences (Pakistan), 2005, pp. 604-633

83 Shanmugalakshmi, R., and Prabu, M.: ‘Research Issues on Elliptic Curve Cryptography and its applications’, IJCSNS International Journal of Computer Science and Network Security, June 2009,

84 Wang, Y., Ramamurthy, B., and Zou, X.: ‘The performance of elliptic curve based group Diffie-Hellman protocols for secure group communication over ad hoc networks’, IEEE international Conference on Communication, 2006, pp. 2243-2248

85 Zapata, M.G., and Asokan, N.: ‘Securing ad hoc routing protocols’, ACM Workshop on Wireless Security (WiSe), September 2002, pp. 1-10

86 Datorkommunikation, A.F., and Jacobson, A.: ‘Metrics in Ad Hoc Networks’, Master's Thesis, Tekniska Univeristy, 2000

87 Hyytiä, E., Koskinen, H., Lassila, P., Penttinen, A., Roszik, J., and Virtamo, J.: ‘Random waypoint model in wireless networks’, Networks and Algorithms: complexity in Physics and Computer Science, Helsinki, June 2005

88 Taneja, S., and Kush, A.: ‘Energy Efficient, Secure and Stable Routing Protocol for MANET’, Global Journal of Computer Science and Technology, May 2012

89 Zhou, L., and Haas, Z.J.: ‘Securing ad hoc networks’, iEEE Network Magazine , December 1999, pp. 24-30

90 Boneh, D.: ‘The decision diffie-hellman problem’, 3rd Algorithmic Number Theory Symposium, Lecture Notes in Computer Science, Springer, 1998, pp. 48-63

91 Lederer, C., Mader, R., Koschuch, M., Großschädl, J., Szekely, A., and Tillich, S.: ‘Energy-efficient implementation of ECDH key exchange for wireless sensor networks’, Information Security Theory and Practice. Smart Devices, Pervasive Systems, and Ubiquitous Networks, 2009, pp. 112-127

92 Network Simulator 2, 1989, : http://www.isi.edu/nsnam/ns

93 Pathan, A.S.K.: ‘Security of self-organizing networks: MANET, WSN, WMN, VANET’, CRC press, Auerbach Publication, USA, 2010

94 Hollick, M., Martinovic, I., Krop, T., and Rimac, I.: ‘A survey on dependable routing in sensor networks, ad hoc networks, and cellular networks’, 30th EUROMICRO Conference, IEEE Computer Society Press, Los Alamitos, September 2004, pp. 495-502

95 Maltz, D.B.J.D.A., and Broch, J.: ‘DSR: The dynamic source routing protocol for multi-hop wireless ad hoc networks’, Computer Science Department Carnegie Mellon University Pittsburgh, PA, 2001, pp. 15213-13891

96 Kim, J., and Tsudik, G.: ‘SRDP: Secure route discovery for dynamic source routing in MANETs’, Ad Hoc Networks, August 2009, pp. 1097-1109

97 Mamoun, M.H.: ‘A Secure DSR Routing Protocol for MANET’, Journal of Convergence Information Technology, 2009

98 Yi, S., Naldurg, P., and Kravets, R.: ‘Integrating quality of protection into ad hoc routing protocols’, 6th World Multi-Conference on Systemics, Cybernetics and informatics (SCI'02) , August 2002

99 Deshpande, V.S.: ‘Security in Ad-Hoc Routing Protocols’, IEEE Workshop on Mobile Computing Systems and Applications, 2007, pp. 158-163

100 Abusalah, L., Khokhar, A., BenBrahim, G., and ElHajj, W.: ‘TARP: trust-aware routing protocol’, ACM International Wireless Communication and Mobile Computing Proceeding - IWCMC, Vancouver, Canada, August 2006, pp. 135-140

101 Abusalah, L., and Khokhar, A.: ‘TARP Performance in a Mobile World’, 50th Annual IEEE Globecom International Conference, Washington, DC, November 2007, pp. 700-704

102 Samian, N., and Maarof, M.A.: ‘Securing MANET Routing Protocol using Trust Mechanism’, Normalia Postgraduate Annual Research Seminar, July 2007

103 Gong, W., You, Z., Chen, D., Zhao, X., Gu, M., and Lam, K.Y.: ‘Trust based routing for misbehavior detection in ad hoc networks’, Journal of Networks, May 2010, pp. 551-558

104 Ukey, A.S.A., and Chawla, M.: ‘Detection of packet dropping attack using improved acknowledgement based scheme in manet’, International Journal of Computer Science Issues, 2010, pp. 12-17

105 Zhan, G., Shi, W., and Deng, J.: 'Tarf: A trust-aware routing framework for wireless sensor networks', 7th European Conference on Wireless Sensor Networks (EWSN'10), Springer, 2010, pp. 65-80

106 Balakrishnan, V., Varadharajan, V., Tupakula, U.K., and Lues, P.: 'Trust and recommendations in mobile ad hoc networks', 3rd International Conference on Netwroking and services (ICNS'07), Athens, Greece, 2007, pp. 64-69

107 Pirzada, A.A., Datta, A., and McDonald, C.: 'Trust-based routing for ad-hoc wireless networks', 12th IEEE International Conference on NEtwroks (ICON'04), November 2004, pp. 326-330

108 Pirzada, A.A., and McDonald, C.: 'Establishing trust in pure ad-hoc networks', 27th Australian Conference on Computer Science, Dunedin, New Zealand, 2004, pp. 47-54

109 Yong, C., Chuanhe, H., and Wenming, S.: 'Trusted dynamic source routing protocol', International Conference on Wireless Communication, Networking and Mobile Computing, 2007, pp. 1632-1636

110 Garg, K., and Misra, M.: 'Trust based multi path DSR protocol', International Conference on Availability, Reliability and Security, 2010, pp. 204-209

111 Buchegger, S., and Le Boudec, J.Y.: 'Performance analysis of the CONFIDANT protocol', 3rd ACM Symposium on Mobile Ad Hoc Netwroking and Computing, 2002, pp. 226-236

112 Molva, R., and Michiardi, P.: ‘Core: a collaborative reputation mechanism to enforce node cooperation in mobile ad hoc networks’, Institute EurecomResearch Report RR-02-062, December 2001

113 Bansal, S., and Baker, M.: ‘Observation-based cooperation enforcement in ad hoc networks’, Technical repor, Computer Science Department , Standford University, July 2003

114 Hoffman, K., Zage, D., and Nita-Rotaru, C.: ‘A Survey of attacks on Reputation Systems’, Technical reports, Paper 1677, Computer Science Department, Purdue University, 2007

115 Theodorakopoulos, G., and Baras, J.S.: ‘On trust models and trust evaluation metrics for ad hoc networks’, Selected Areas in Communications, IEEE Journal on, 2006, pp. 318-328

116 Gordon, R., and Dawoud, D.: ‘A Hybrid Trust Model for Mobile Ad Hoc Networks’,Netwroking and Communications, KwaZulu-Natal University, July 2009