Detecting Misbehaving Nodes in Mobile Ad hoc Networks

by

Nan Kang

Thesis

submitted in partial fulfillment of the requirements for

the Degree of Master of Science (Computer Science)

Acadia University

SPRING Convocation 2011

This thesis by Nan Kang was defended successfully in an oral examination on DATE OF DEFENCE.

The examining committee for the thesis was:

_____
Dr. Nancy Clarke, Chair

_____
Dr. Nauman Aslam, External Reader

_____
Dr. Haiyi Zhang, Internal Reader

_____
Dr. Elhadi Shakshuki, Supervisor

_____
Dr. Danny Silver, Director

This thesis is accepted in its present form by the Division of Research and Graduate Studies as satisfying the thesis requirements for the degree Master of Science (Computer Science).

…….

This thesis by Nan Kang was defended successfully in an oral examination on Feb 10th, 2011.

The examining committee for the thesis was:

Dr. Nancy Clarke, Chair

Dr. Nauman Aslam, External Reader

Dr. Haiyi Zhang, Internal Reader (additional internal readers are listed here)

Dr. Elhadi Shakshuki, Supervisor

Dr. Danny Silver, Director

This thesis is accepted in its present form by the Division of Research and Graduate Studies as satisfying the thesis requirements for the degree Master of Science (Computer Science).

I, Nan Kang, grant permission to the University Librarian at Acadia University to reproduce, loan or distribute copies of my thesis in microform, paper or electronic formats on a non-profit basis. I, however, retain the copyright in my thesis.

_____
Author

_____
Supervisor

_____
Date

Table of Contents

Glossary

ACK: ACKnowledgement

AACK: Adaptive ACKnowledgement

CA: Certificate Authority

CGSR: Clusterhead Gateway Switch Routing

DoS: Denial of Service

DSDV: Destination-Sequenced Distance Vector

DSA: Digital Signature Algorithm

DSR: Dynamic Source Routing

DSS: Digital Signature Standard

EAACK: Enhanced Adaptive ACKnowledgement

IDS: Intrusion Detection System

MAC: Message Authentication Code

MANET: Mobile Ad hoc NETwork

MRA: Misbehaviour Report Authentication

NS2: Network Simulator 2

PDR: Packet Delivery Ratio

S-ACK: Secure ACKnowledgement

RAODV: Reliable Ad hoc On-demand Distance Vector

RO: Routing Overhead

RREQ: Route REQuest

RREP: Route REPly

SENCAST: Scalable Protocol for Unicasting and Multicasting in a Large Ad hoc Emergency Network

WRP: Wireless Routing Protocol

WSN: Wireless Sensor Network

ZHLS: Zone-based Hierarchical Link State

List of Figures

List of Tables

Dedicated to my beloved Mother and Father

Abstract

There has been a tremendous growth in the use of wireless communication in the past few decades. Mobile Ad hoc NETwork (MANET) is one of the most important one among various wireless communication mechanisms. Its unique infrastructureless network and self-configuring capability makes it ideal for many mission critical applications. However, these characteristics also make MANET vulnerable to passive and active attacks due to its open medium, changing topology and lack of centralized monitoring. In this research, we propose a new intrusion detection system specially designed for mobile ad hoc networks. Our proposed scheme introduces a combination of a hybrid acknowledgement scheme as well as digital signature techniques. By adopting such techniques, our goal is to design and implement an efficient and secure intrusion detection system called EAACK (Enhanced Adaptive ACKnowledgement) for mobile ad hoc networks that is capable of detecting misbehaving nodes in mobile ad hoc networks and address many challenges in this area. We also implemented the proposed scheme along with other contemporary research work by simulation. In the end of the research, we compare the results and evaluate the performance of our proposed scheme. The results indicate positive performances in various test scenarios when comparing to existing approaches such as Watchdog, TWOACK and AACK.

Acknowledgement

First of all, a million thanks to Dr. Elhadi M. Shakshuki, who constantly gives me precious suggestions and guidance during my research and study in Acadia University. He encouraged and helped me all the time during my research of this thesis project.

Also, I would like to thank the faculty and staff of the Jodrey School of Computer Science at Acadia University, for teaching and helping me to prepare for the real world challenges.

Finally, I would like to thank my parents, who have always been there with their warm support and care.

1 INTRODUCTION

The past few decades witnessed a tremendous growth in the use of wireless networks. With the reduced cost and great improvement of the wireless transmission bandwidth, wireless networks have been able to replace wired networks in many settings. One of the major advantages of wireless network is its ability to allow the data transmission among users in a common area while remains mobility. However, this mobility is largely dependent on the range of the transmitters. That means the participants will not be able to remain connected when the distances between them are beyond the coverage ranges of the transmitters. Mobile Ad hoc NETworks (MANETs) solved this problem by allowing intermediate participants to relay data transmission while still maintaining mobility. For this unique advantage, MANETs have been widely spreaded to many mission critical applications. As a result, security is now a major concern in many MANETs applications.

When it comes to secure MANETs, one of the biggest challenges is all of the factors that must be accounted for: infrastructureless networks, dynamic topologies, resource limitations and limited physical protections. These characteristics make MANETs vulnerable to both active and passive attacks [60]. With regard to these challenges, in this research, we propose a new security mechanism that adopts intrusion detection and cryptography techniques.

1.1 *Problem Definition and Research Objectives*

Research has been conducted in the past few decades to alleviate the security threats and challenges. Each approach has its own strengths and weaknesses. Some concentrates on message authentication and cryptography techniques, but they suffer from the late detection of attacks due

to their lack of misbehaviour report and reaction schemes. Others propose intrusion detection systems to provide fast detections, but most of such proposals spend less effort on protecting the communication itself. There rarely are any security mechanisms which combine the benefits of intrusion detection system and cryptography techniques.

In this research, firstly, we investigate the existing approaches to MANETs security mechanisms on both intrusion detection and cryptography. Secondly, by extending the work of AACK [56] and TWOACK [34], we propose a new intrusion detection mechanism equipped with cryptography techniques. Thirdly, we implement such mechanism and compare its performance against other existing approaches in software simulation environment.

The major research objective of this thesis is to develop a robust intrusion detection systems specifically designed for MANETs. The proposed system demonstrates higher malicious behaviour detection rates in certain circumstances with minimal effect on network performance. Two research works [26][27] based on this thesis project have been published in related conference proceedings.

1.2 *Contributions*

The major contributions of this thesis are:

- An investigation on contemporary research work in MANETs intrusion detection system and cryptography.
- Proposal of a new intrusion detection system equipped with cryptography protection scheme that is specifically designed for MANETs.
- A performance evaluation with network simulation that compares the proposed approach and other popular MANETs intrusion detection systems.

1.3 *Thesis Structure*

The remainder of this thesis is organized as follows: Chapter 2 gives a review of background information. Chapter 3 describes the problem statement and literature review. Chapter 4 concentrates on describing the proposed scheme in details. In Chapter 5, we describe the simulation settings and performance evaluation. Chapter 6 gives conclusion of this research and potential future work.

2 BACKGROUND

2.1 *Mobile Ad hoc Networks*

Ad hoc is a Latin phrase which means “for this purpose”. Mobile Ad hoc NETwork (MANET) is a collection of mobile nodes that communicate with each other via bi-directional wireless links either directly or indirectly. Each node is equipped with both a wireless transmitter and a receiver. There are two types of mobile ad hoc networks, namely single-hop and multi-hop. For single-hop network, nodes are free to directly communicate with any other nodes in their own radio range and there are no intermediate nodes. Examples include the pairing of bluetooth headset and smartphone. The dotted line in the Figure 1 indicates the communication range of a node. In this case, for example, node A and node C are both inside each other’s communication range and thus they can communicate with each other within a single hop network. For multi-hop network, the cooperation of intermediate nodes is required when nodes need to communicate with other nodes that are out of their radio range. Nodes will rely on other nodes to transmit packets to a destination node. Denoted in Figure 1, the radio range of node A and D are marked as circle 1 and 4 while the radio range of node B and C are marked as circle 2 and 3 respectively. As node C is outside of the communication range of node B, they need to rely on other nodes’ cooperation, to relay the message. In this case, it can be either node A or node D. More description of MANET and its related researches can be found in [6][16][46][47].

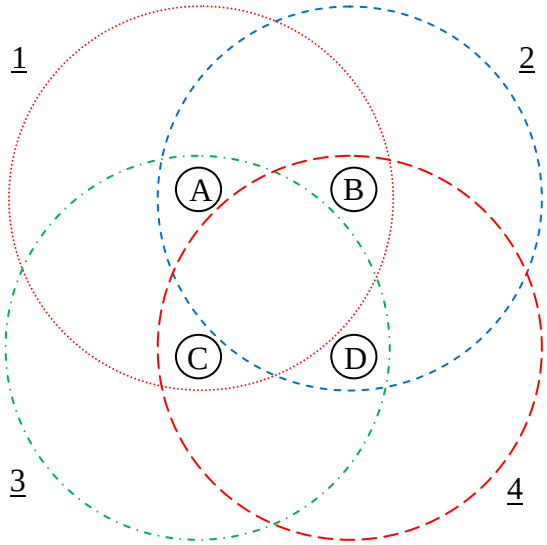

Figure 1 Mobile Ad Hoc Network

In contrast to traditional wireless or wired networks, MANET is a decentralized, self-configuring and self-organizing network that does not require a pre-existing infrastructure or a central station, thus the mobile nodes inside the network are free to move randomly. MANET is capable of dynamically constructing a short lived and self-configuring network without the support of a centralized network infrastructure. Minimal configuration and quick deployment make ad hoc networks suitable for using in emergency circumstances where an infrastructure is unavailable or unfeasible to the installed like natural or human-induced disasters, military conflicts and medical emergency situations [40]. In fact, MANET was originally developed for military purposes [61]. Li *et al.* [33] identifies the main characteristics of MANET as follows:

- *Autonomous*: Each node in MANET is autonomous and is capable of self-configuring and self-organizing.
- *Distributed*: MANET is distributed in its operation of functionalities, such as routing host configuration and security.

- *Multi-hop*: Multi-hop routing is required when the source node and destination node of a transmission are not within the same radio communication range.
- *Dynamic topology*: Nodes can be added and removed from network at any time. The network topology is constantly changing.
- *Thin terminal*: Nodes in MANET are usually light weighted with limited computing power and battery reservation.

In recent years, the applications of MANET have been extended to commercial and daily use. Especially under the current trend of mobile computing, ad-hoc network has become more and more popular among mobile devices. Ranging from the popular bluetooth devices to the ad-hoc mode in WiFi network, MANET has become an important way of sharing information between various computing devices.

2.2 *MANETs Routing Protocols*

Routing is the act of moving information across a network from a source to a destination. Routing protocols are protocols that implement routing algorithms [73].

There have been a lot of routing protocols proposed and implemented for the MANETs. Sun [57] did a very good review of the routing protocols in MANETs. In this section, we give a brief overview of all the major routing protocols in MANETs. Of all the existing protocols, we will concentrate on Dynamic Source Routing (DSR) protocol as it is the one we used in our research.

2.2.1 Overview of MANETs Routing Protocols

Of all the routing protocols that have been proposed for MANETs, based on their routing algorithms, they can be categorized to three types: proactive, reactive and hybrid.

For proactive routing protocols, the routing protocol finds paths to every other individual node in the network whether there is a packet sending request or not, and update these paths frequently after a certain period of time. They react to topology changes regardless of whether there is a change or not. The constant update of routing information can potentially waste a great amount of bandwidth. They are also called table-driven method. Examples include Clusterhead Gateway Switch Routing protocol (CGSR) [51], Destination-Sequenced Distance Vector routing protocol (DSDV) [47] and Wireless Routing Protocol (WRP) [39] .

For reactive routing protocols, the routing protocol does not require constant update of paths. On the other hand, it only updates when there is demand for data transmission. This significantly reduces the routing overhead when the network traffic is light and the network topology does not change dramatically, since there is no need to update routing table frequently when there is no traffic. Examples of reactive routing protocols include: A Scalable Protocol for Unicasting and Multicasting in a Large Ad hoc Emergency Network (SENCAST) [51], Reliable Ad hoc On-demand Distance Vector routing protocol (RAODV) [48] and Dynamic Source Routing protocol (DSR) [22].

Hybrid routing protocol is a combination of proactive and reactive routing protocol. Zone-based Hierarchical Link State (ZHLS) [21] routing protocol is a typical example. According to ZHLS routing protocol, the entire network is divided into several non-overlapping zones. If the source and destination nodes are within the same zone, ZHLS will work as a passive routing protocol. Otherwise, ZHLS works as a reactive routing protocol because it needs a location search to find the zone ID of the destination node. A graph classification of MANET routing protocols can be found in Figure 2 (next page).

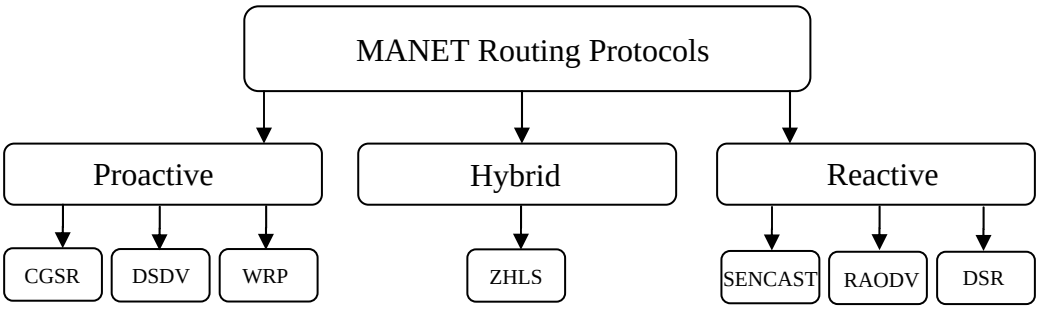

Figure 2 MANET Routing Protocols

In our research, we will concentrate on one of the reactive routing protocols: DSR. The reason being we need the source node to be able to know the identity of every other intermediate node in the network without consuming a great amount of routing overhead. This can be achieved by adopting routing protocols such as DSR. Furthermore, some of the intrusion detection techniques we investigated in this research are restricted to DSR protocol. In order to better compare the performances between different schemes, it is best to keep the routing protocols consistent. In next section, we will describe DSR protocols in more details.

2.2.2 DSR Protocol in Detail

Dynamic Source Routing (DSR) protocol is specifically designed for multi-hop MANETs. DSR allows the network to be completely self-organizing and self-configuring, without the need for any existing network infrastructure or administration [22]. It avoids the need of constantly updating routing information in the intermediate nodes while provides loop-free routing by adopting source routing technique. The DSR protocol is consisted of two mechanisms, namely route discovery and route maintenance.

- *Route Discovery* is a mechanism called whenever the source node **S** planning to send a packet to the destination node **D**. It is only used when there are no existing routes between node **S** and node **D**.
- *Route Maintenance* is used to detect when there are topology changes within the network, and thus make the stored routes between node **S** and node **D** no longer exists. When such scenarios are detected, node **S** can attempt another route to node **D**, or otherwise invoke *Route Discovery* to find a new route.

When the source node has a packet ready to be sent, it first searches the local knowledge base and see if there exists a route to the destination node. If there is, this route will be used; otherwise, the source route will broadcast a Route REQuest (RREQ) message to all the neighbours within its communication range. Upon receiving this RREQ message, each neighbour will append their addresses to the message and broadcast this new message to their neighbours. If any node receives the same RREQ message more than once, it will simply ignore it. When the RREQ message finally arrives the destination node, the destination node will initiate a Route REPly (RREP) message and send this message back to the source node by reversing the route in the RREQ message.

DSR routing protocol has been widely adopted in various implementations including Watchdog [36], TWOACK [34] and AACK [56]. Its reactive on-demand routing discovery scheme greatly reduces the network overhead and thus make it ideal for lightweight network like MANET.

2.3 *Attacks against MANETs*

The advantages of MANET made it more and more popular in mission critical applications over the past few years. Unlike traditional wired network, security for wireless networks is much harder. In wireless networks, radio links are vulnerable to remote attacks, while wired network requires physical access. Among wireless networks, MANET presents additional security threats for the following reasons:

- Due to its sparse distribution and physical condition, mobile nodes are much more vulnerable to capture or compromise.
- The packet transmission in MANETs depends on the cooperation of all nodes on the route. Certain number of compromised nodes may disrupt the entire network.
- Due to its infrastructureless network, traditional security schemes like authentication server are no longer available. All nodes need to handle security by themselves.

As a result, MANET security has become one of the major concerns for researchers. For example, most of the routing protocols for MANETs assume that every node in the network is cooperative and not malicious [66]. However, for multi-hop transmissions in MANETs, several nodes relay packet sent by the source node until it reaches its final destination. Each node on the route from the source node to the destination node acts as a router. Thus, the success of packet distribution highly depends on the cooperation of all nodes involved. Only one compromised node can fail the entire transmission. If the amount of compromised nodes exceeds certain threshold, it is likely to cause a network partition and thus cause a failure of the entire network.

The attacks on MANETs can be divided into two categories, namely passive attacks and active attacks. For passive attacks, packets containing secret information might be eavesdropped,

but the network operation is not disrupted. Examples include eavesdropping, traffic analysis and monitoring. Active attacks, including injecting packets to invalid destinations into the network, deleting packets, modifying the contents of packets, and impersonating other nodes violate availability, integrity, authentication, and non-repudiation [28]. Examples include jamming, spoofing, modification, replaying and Denial of Service (DoS).

Attacks to MANET can also be categorized according to the network layer they are carried on. Figure 3 is a good illustration of the classification of attacks based on network layers.

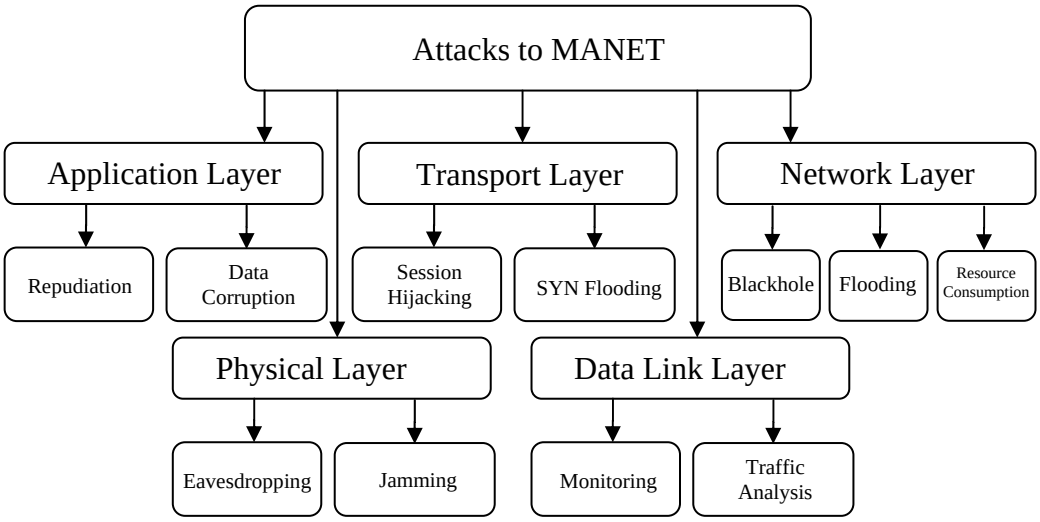

Figure 3 Classes of MANET attacks

2.3.1 Physical Layer Attacks

Due to the nature of MANET and all wireless networks, network signals are broadcasted over the airwaves and thus they can be easily captured by malicious attackers [28][41]. Eavesdropping is one of the attacks that can be accomplished in physical layer. By intercepting radio signals over the air, attackers can easily retrieve the conversations between target nodes. Furthermore, radio signals can be easily interfered as well. Providing the malicious attacker is equipped with a powerful transmitter, a malicious signal can be easily generated to overwhelm

the target signal and thus disrupt the communications. One of the most famous attacks of such type is jamming attacks.

2.3.2 Link Layer Attacks

In MANET, attacks may target the link layer by corrupting the cooperation of link layer's protocols [60]. MANET has an open medium peer to peer network infrastructure. The communication between each two nodes is maintained by link layer protocols. One of the most important attacks to the link layers in MANET is traffic analysis. It provides the attacker with the ability to detect the functionalities of communication participants. As a matter of fact, this kind of attack is widely adopted among various network types. For example, the work presented in [54] is an implementation of traffic analysis attacks to Wireless Sensor Networks (WSNs).

2.3.3 Transport Layer Attacks

Transport layer protocols in MANET are responsible for setting up of end-to-end connection, end-to-end reliable delivery of packets, flow control, congestion control, clearing of end-to-end connection [60]. As a result, similar to traditional TCP network, the transport layers of MANETs are vulnerable to SYN flooding attacks [62] or session hijacking attacks.

2.3.4 Application Layer Attacks

Application layer attacks mainly include computer virus or worms. Like traditional TCP/IP network, attackers can compromise mobile nodes and deploy virus or worms payload. These malicious programs can be transferred to other mobile nodes via application layer communications. Attackers can break down the entire network by spreading malicious programs

to other nodes in the network. Researchers have investigated the worm attacks in both UDP and TCP based MANETs [1][11][12].

2.3.5 Network Layer Attacks

In OSI [74] network model, network layer is responsible for extending the network communication from one hop neighbouring nodes to all the other nodes in MANET. When multi-hop communication is needed, nodes strongly depend on each other to relay data packets. This makes network layer protocols of MANET vulnerable to various attacks.

The network layer protocols can be mainly divided into three phases, namely routing discovery, routing maintenance and data forwarding. We have already discussed the routing discovery and routing maintenance phases in Section 2.2.2. In our research, we mainly concentrate on the attacks to the data forwarding phases. As discussed before, most of the routing protocols for MANETs assume that every node in the network is cooperative and not malicious [66]. This leaves the attackers with the opportunities to achieve different attacks during the data forwarding phase. General attacks including dropping packets quietly, contaminating data content, replay or even flooding data packets.

The type of attack we mainly focus on in this research is called packet dropping attack. As discussed before, when it comes to multi-hop communication in MANETs, the source node totally depends on intermediate nodes to forward packets to the destination node. If the intermediate nodes decide to not forward other packets or dropping other packets, they can easily block the network communication. There are many reasons for nodes to drop packets in MANET. They can be mainly divided into two major categories.

- *Unintended misbehaviour:* Mobile nodes can drop packets when they are overloaded in terms of computational power and memory space. As we know that wireless channels are considered as unreliable [4], this misbehaviour can also be caused by unstable network or network collision.
- *Intended misbehaviour:* This kind of misbehaviour is usually caused by selfish or malicious nodes. For selfish nodes, they conduct such misbehaviour to preserve its battery consumption due to the limited energy and network bandwidth of mobile nodes. For malicious nodes, they agree to participate in the routing process but refuse to forward data packets. The purpose of such malicious nodes is to disrupt the network communication and limit network connectivity. Examples include black hole and grey hole attacks.

A general categorization of packet dropping attacks is demonstrated in Figure 4.

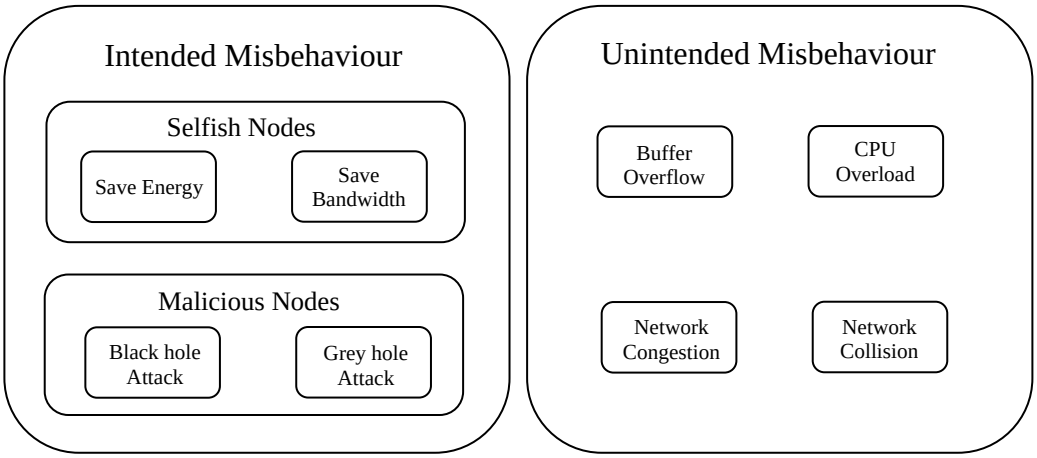

Figure 4 Packet Dropping Attack

2.4 *Intrusion Detection System*

If the attacks discussed in the previous sections cannot be prevented or detected effectively, MANETs will not be able to perform well in the mission critical area. To alleviate the effects of such selfish or malicious nodes in MANETs, some proactive approaches such as cryptography and authentication were introduced to the area [17][18][42][69]. However, all of these security mechanisms suffer from the late detection issue; this flaw leaves the attackers with plenty of time to affect the network performance. Furthermore, due to the uniqueness of MANET infrastructure, such as open medium, rapid changing of topology and lack of centralized monitoring, prevention technique alone is no longer suitable to protect MANET from outside attackers; therefore, Intrusion Detection System (IDS) should be added to enhance its security. If we can detect the attackers as soon as they enter the network, we will be able to completely eliminate their attempts of doing any damage to the network. IDS in MANETs can act as a second layer of defence, and it is a great complement to the existing proactive protection scheme. A more thorough literature review of IDSs in MANETs will be presented in Section 3.2.

2.5 *Cryptography*

2.5.1 A Brief History of Cryptography

By definition, *Cryptography* is the study of mathematical techniques related to aspects of information security such as confidentiality, data integrity, entity authentication, and data origin authentication [37]. As a way to protect communication security, cryptography technique has a long and fascinating history. The Kahn's book [25] completed in 1963 presented the most important history of the development of cryptography techniques in human society. From 4,000 years ago by the Egyptians, to the two world wars in the twentieth century, the evolution of

cryptography techniques indicates the pursuing of mankind's attempts to ensure information security.

With the development of Internet, the problem of how to keep secrets during communication has become more important than ever. Unlike traditional paper based communication protocols, now that all the information is shared digitally over wired or wireless network, new security challenges have emerged. For example, digital document can be easily copied for thousands of times within seconds. The change of communication media also brings up new challenges to traditional signature authentication mechanisms.

Many researchers and scientists have contributed countless time and effort in this area in the past few decades. The most significant advancement among them is believed to be in 1976 when Diffie and Hellman published the paper "New Directions in Cryptography" [14]. In this research work, the concept of public key cryptography was first introduced to the computer security field. Even they did not propose any practical implementation; this concept has since then attracted many research interests.

Two years later, in 1978, Rivest and his colleagues proposed the first practical public-key encryption and signature scheme, which is now referred to as RSA [50]. In the 1980s, much more advancement was developed in public-key encryption, but none of them has rendered RSA as insecure. On the other hand, another important class of powerful public-key schemes was proposed by EIGamal in 1985, which are also based on discrete logarithm problem [37]. Many encryption schemes were later proposed based on EIGamal's work. In 1994, the Digital Signature Standard (DSA) [72] scheme is one of the most important implementation among them. In this research, we will mainly concentrate on investigating DSA and RSA schemes in MANETs.

2.5.2 Cryptography Goals in MANETs

Unlike traditional wired or wireless network, the unique characteristics of MANETs bring new requirements to security. According to Jie *et al.*, security in MANETs is defined as a combination of processes, procedures, and systems used to ensure *confidentiality*, *authentication*, *integrity*, *availability*, and *non-repudiation* [37].

- *Confidentiality:* The purpose of confidentiality is to prevent unauthorized third-parties from accessing disclosed information. Such attack can occur either within or after communication. For example, due to the physical distribution of mobile nodes in MANETs, attackers can easily capture and compromise one node to access its stored unencrypted information.
- *Authentication:* Authentication is to ensure the identities of communication parties. This can prevent impersonation attack. Unlike traditional centralized network, a central authority is not feasible in MANETs. Menezes *et al.* proposed to use message authentication code for authentication.
- *Integrity:* Integrity is to guarantee that the data is not manipulated by unauthorized parties. Unauthorized manipulation includes insertion, deletion and substitution. Attackers usually contaminate data integrity by replay attack where they intercept, modify and re-send data packets. Hash functions are often used to protect data integrity.
- *Availability:* Availability is to ensure the mobile nodes are always available to legitimate users. Due to the lack of physical protection, nodes in MANETs can easily be captured or altered. Attackers can easily achieve Denial of Service (DoS)

attack in MANETs. Prevention scheme to such attack including dynamically finding alternative routes.

- *Non-repudiation:* The goal of non-repudiation is to prevent one from denying previous commitments or actions. By providing a signature for the message, the sender cannot later deny or alter the data packets it sent. Digital signatures are often used to provide non-repudiation schemes.

2.5.3 Overview of Cryptographic Techniques

Among many cryptographic techniques, it is a very difficult decision to decide which one should be used. Many elements need to be taken into consideration, including network metric, node hardware, network infrastructure, etc. Jie *et al.* has a very thorough survey on contemporary cryptographic techniques that have been implemented on MANETs [9].

Cryptographic techniques in MANETs are generally divided into three categories: *symmetric cryptography*, *asymmetric cryptography* and *threshold cryptography* [37].

- *Symmetric Cryptography:* In this case, the encryption key and decryption key are usually closely related, identical in most cases. This scheme requires the communication parties to exchange keys before communication, but it is less computational intensive than *asymmetric cryptography*. Existed implementations for MANETs include SEAD [17], SAODV [64], LEAP [71] and ARIADNE [18].
- *Asymmetric Cryptography:* This category is best known as public-key cryptography where there is a pair of public/private keys. Public keys are accessible to all everyone while private keys are kept secret. The well-known RSA/DSA schemes are a good example of such category. Examples include Kaya [30], ARAN [52] and LHAP [70].

- *Threshold Cryptography:* The idea of *threshold cryptography* is to distribute many keys to a certain amount of parties. In order to decrypt an encrypted secret message, a number of parties exceeding a threshold are required. This idea of shared secret come from the research work by Adi Shamir in 1979 [55]. The work presented in IKM [65] is a good example of such implementation in MANETs.

In fact, threshold cryptography is a kind of asymmetric cryptography. The secret is encrypted with a public key, and private keys are shared among different parties. There are also many subtypes of cryptographic technique. It's unfeasible to describe all of them in this research work. In order to provide the user with a necessary background on cryptographic technique, we will only concentrate on symmetric-key and asymmetric-key cryptography.

Symmetric-key Encryption

In symmetric-key, the encryption key and decryption key are usually identical. These keys serve as a shared secret between the two or more communication parties. As a result, the secure communication is possible only when all communication participants exchanged this shared secret through a secure channel in advance. The process is demonstrated in Figure 5.

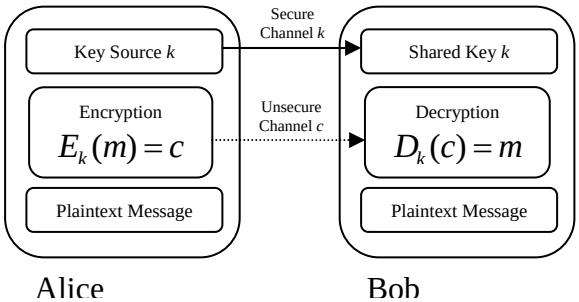

Figure 5 Two Parties Communication Using Symmetric-key Encryption

As demonstrated in Figure 5, in this case, Alice is the sender and Bob is the receiver. To be able to communicate through the unsecured channel, Alice and Bob have to exchange the shared secret key *k* through a secure channel first. Due to the open medium in MANETs, attackers can easily capture one node and duplicate multiple malicious nodes. In the case of symmetric-key encryption, all nodes shared the same secret key. Compromised one node could well lead to a collapse of the entire network [27].

Public-key Encryption

For public-key encryption, the encryption key (public key) and decryption key (private key) are generally different [37]. Receiver holds both the public key and private key. It is designed so that it is mathematically infeasible to deduce the private key solely based on the public key. Thus, the public key is safe to be revealed to sender via an unsecured channel. This process is demonstrated in Figure 6.

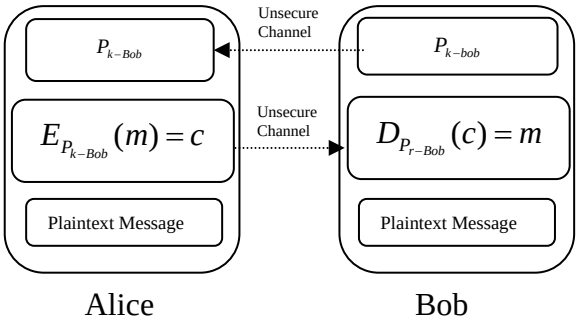

Figure 6 Two Parties Communication Using public-key encryption

As demonstrated in Figure 6, Alice is the sender party and Bob is the receiver party. In order to keep the message *m* secret, Bob first reveal its public key P_{k-Bob} to Alice through an unsecure channel. Note that symmetric-key scheme requires key distributed via a secure channel

while public-key scheme does not. Upon receiving the public key P_{k-Bob} from Bob, Alice uses this public key to encrypt the secret message *m* and get the cipher text *c*, which can be described as:

$$E_{P_{k-Bob}}(m) = c \tag{1}$$

Then, the cipher text can be safely transmitted via an unsecure channel to Bob without revealing the secret *m*. When receives the cipher text *c*, Bob applies its private key P_{r-Bob} to decipher *c* and get the plain text message *m*, which can be described as:

$$E_{P_{r-Bob}}(c) = m \tag{2}$$

In summary, the advantages of *public-key* cryptography in MANETs are that 1) key distribution does not require secure channel 2) one compromised nodes won't take down the entire network as each nodes hold a different pair of keys.

2.5.4 Digital Signature

A digital signature is an electronic analogue of a written signature; the digital signature can be used to provide assurance that claimed signatory signed the information [72]. It can be generalized as a piece of data string that associates a digital message with its original creator. Furthermore, digital signature can also be used to ensure the message was not modified or contaminated after it was signed (i.e., to detect the integrity of the signed data). The purpose of a digital signature is to provide a means for an entity to bind its identity to a piece of information held by the entity into a tag called a signature [37].

Digital signature schemes can be mainly divided into the following two categories:

- *Digital Signature with Appendix:* This scheme requires the original message as input to the verification algorithm. Example includes Digital Signature Algorithm (DSA) [72].
- *Digital Signature with Message Recovery:* This scheme does not require the original message in verification algorithm. It is capable of recover the message based on the signature itself. RSA [50] algorithm is an example of such scheme.

The mathematical descriptions of these schemes are omitted in this thesis. However, interested readers can refer to [37]. In this section, we describe the digital signature with appendix as an example of typical digital signature scheme. The general process of two parties' communication with digital signature is depicted in Figure 7. As demonstrated in Figure 7, every message *m* must be put through a pre-agreed hash function *H* to get a fixed-length message digest, which can be described as:

$$H(m) = d \tag{3}$$

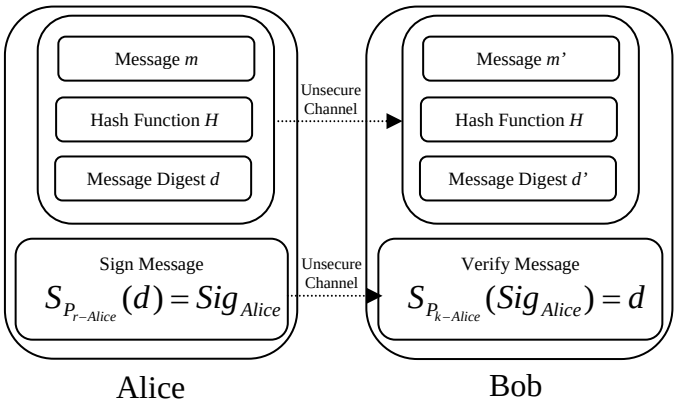

Figure 7 Two Parties Communication Using DSA

The sender Alice then apply its private key $P_{r-Alice}$ on the computed message digest *d* to get a signature Sig_{Alice} bind to message *m* and Alice's private key:

$$S_{P_{r-Alice}}(d) = Sig_{Alice} \tag{4}$$

To assure the validity of digital signature, Alice must keep her private key $P_{r-Alice}$ a secret without revealing to anyone else. Otherwise, the digital signature scheme can be easily penetrated when the malicious attacker Eve intercepts the message and easily forges malicious messages with Alice's signature and send them to Bob.

Next, Alice is able to send the message *m* along with the signature Sig_{Alice} to Bob via an unsecured channel. Bob first compute the received message *m'* with hash function *H* and get the message digest *d'*, which can be described in:

$$H(m') = d' \tag{5}$$

Bob can verify the signature by applying Alice's public key $P_{k-Alice}$ on Sig_{Alice}, as in:

$$S_{P_{k-Alice}}(Sig_{Alice}) = d \tag{6}$$

If $d = d'$, then it is safe to claim that the message *m'* transmitted through an unsecured channel is indeed sent from Alice and intact.

A detailed literature review of cryptography techniques used in MANETs will be presented in Section 3.4.

3 PROBLEM STATEMENT AND LITERATURE REVIEW

In this chapter, we define the problem statement in the context of a literature review. It will be divided into three sections. In the first section, we describe and discuss the mechanisms and problems of the Watchdog and Pathrater scheme which is one of the most popular IDSs in MANETs. In the second section, we explore related research work in the research field of ad hoc networks. Last but not least, we discuss some of the limitations of current IDSs designed for MANETs.

3.1 *Problem Statement*

The Watchdog and Pathrater scheme is one of the most important IDSs designed for MANETs. Many implementations of IDSs in MANETs are based on this scheme. Our research work concentrates on solving three of the six weaknesses of Watchdog and Pathrater scheme. In the following sections, we first describe the Watchdog and Pathrater scheme in detail, and then we identify our research problems.

3.1.1 Watchdog and Pathrater

In 2000, Marti *et al*. [36] proposed two techniques, Watchdog and Pathrater, which improve the network throughput with the existence of selfish or misbehaving nodes. It has served as the basis for many MANET IDSs proposed in the past few years. The scheme consists of two parts, namely Watchdog and Pathrater. Watchdog works as an intrusion detection system in MANETs, and is responsible for detecting misbehaving or selfish mobile nodes. Pathrater, on the other hand, is proposed to respond to these reported misbehaving nodes and help the routing protocol to avoid them. It has been observed that the combination of Watchdog and Pathrater schemes significantly improve the network throughput in the presence of malicious nodes [36].

The idea of Watchdog is straightforward; it detects misbehaving or selfish nodes by promiscuously listening to its next hop's transmission. If Watchdog overhears that the next node fails to forward the packet, it will increase its failure counter. Once the failure counter of certain nodes exceeds a predefined threshold, these nodes will be reported as misbehaving. In this case, Pathrater will cooperate with routing protocols to avoid these nodes in future transmission. This act is achieved by finding another route from source node to destination node which excludes these reported nodes.

Both the advantage and disadvantage of Watchdog scheme are discussed in [36]. The major advantage of Watchdog is that it is capable of improving network throughput in the presence of misbehaving nodes. It also exceeds other schemes in terms of its ability to detect misbehaving nodes rather than links. On the other hand, the disadvantages of Watchdog are that it may fail to detect the misbehaving node in the presence of 1) ambiguous collisions, 2) receiver collisions, 3) limited transmission power, 4) false misbehaviour report, 5) collusion, and 6) partial dropping. We discuss these disadvantages of MANET with further details in the next section.

3.1.2 Disadvantages of Watchdog

Watchdog is capable of detecting misbehaving nodes and improve network throughput. However, it suffers from the fact that it fails to detect misbehaving nodes in the presence of the following six scenarios. To make this easier for understanding, we describe each scenario using an example.

- *Ambiguous Collisions:* The collisions prevent node A from overhearing node B forward Packet 1 to node C because node A is trying to receive Packet 2 sent from node S at the same time, as shown in Figure 8 (next page).

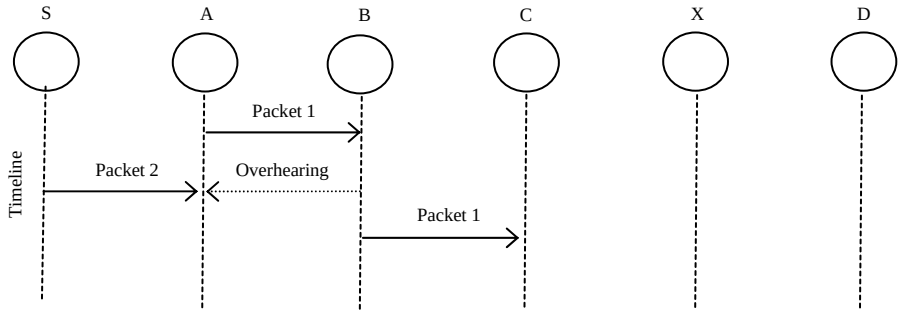

Figure 8 Ambiguous Collisions

- *Receiver Collisions:* In this collision, node A assures that node B has successfully forwarded Packet 1 to node C, but fails to detect that node C did not receive Packet 1 due to a collision with Packet 2 from node X. This scenario is demonstrated in Figure 9 (next page).
- *Limited Transmission Power:* In order to conserve its battery energy, selfish node B limits its transmission power so that it can be overheard by node A while too weak to be received by node C, as demonstrated in Figure 10 (next page).
- *False Misbehaviour Report:* Although node B successfully forwarded Packet 1 to node C, and node A overhears this transmission, node A still report node B as misbehaving. The process is demonstrated in Figure 11 (next page). Due to the open medium of nodes in MANETs, mobile nodes are physically accessible to attackers. Attackers can easily capture and compromise some nodes and achieve this kind of attack.
- *Collusion:* In this scenario, two or more consecutive misbehaving node cooperates to mount a sophisticated attack. This kind of attack is studied by Johnson at CMU [23].

- *Partial Dropping:* The misbehaving node can circumvent the Watchdog by dropping packets at lower rate than the minimum misbehaving threshold predefined by the scheme.

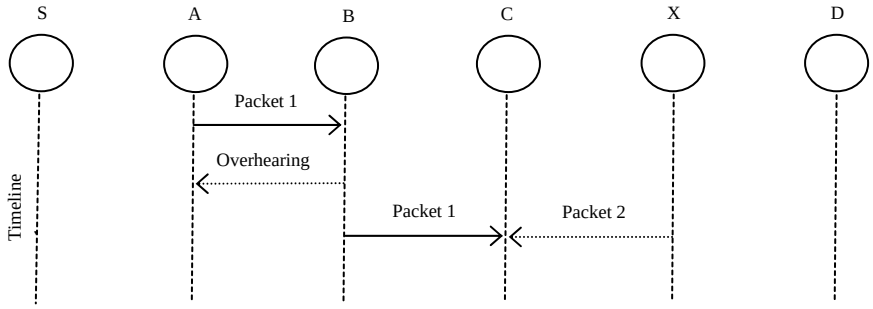

Figure 9 Receiver Collisions

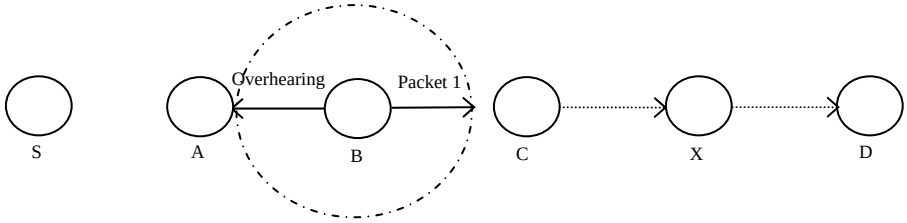

Figure 10 Limited Transmission Power

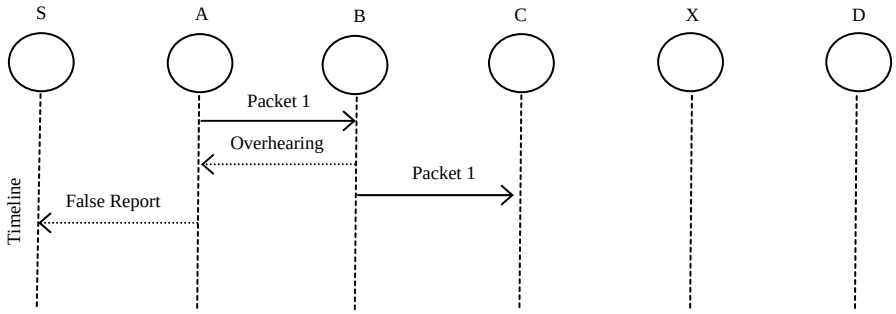

Figure 11 False Misbehaviour Report

3.1.3 Research Problem Statement

Due to the popularity of Watchdog scheme, many intrusion detection systems designed for MANETs are either based on Watchdog or use Watchdog as a performance evaluation metric. Due to time limitations, we decide to propose an intrusion detection system that specifically concentrates on solving three of the six weaknesses of Watchdog scheme, namely: receiver collisions, limited transmission power and false misbehaviour report.

Furthermore, we extend our research to include cryptography techniques during transmission. In our proposed acknowledgement based IDS, it is vital to ensure the acknowledgement packet is not contaminated or forged by malicious parties. For this reason, we intend to integrate digital signature technique to our proposed scheme. In the next section, we present a literature review of other's work related to our research. By investigating the advantages and disadvantages of these works, we clarify the uniqueness of our proposed scheme.

3.2 *Intrusion Detection Systems in MANETs*

Intrusion Detection System is generally considered as the second layer of the security system. Most of the time, it is used as a complement to existing prevention techniques which are considered as the first layer security system [57]. For traditional wired or wireless network, there have been countless implementations of IDSs. As in traditional network, there exists a centralized infrastructure where security scheme can be easily deployed to devices like switches, routers or gateways. Such benefits are not available in MANETs. Moreover, the medium of MANETs is wide open and lack of physical protection, thus malicious attackers have as much accesses as legitimated users do. Last but not the least, due to the fact that mobile nodes are allowed to move randomly, it is hard to distinguish normal or misbehaving behaviour in the

context of MANETs. All of these facts render developing IDSs for MANETs a challenging and unique research work.

Many researchers have devoted their time and effort on developing suitable IDS for MANETs [17][18][45][63][69]. Jie *et al*. presented a very thorough survey on existing IDSs in MANETs [3]. There are many ways to classify these works. For example, [4] classified them into reputation-based and incentive-based IDSs. Ping *et al*. suggested categorizing them based on their detection algorithms, namely, anomaly, misuse and specification-based detection. Depending on the detection algorithms used, IDS can be classified into three major categories:

- *Specification-based detection:* The IDS monitors the behaviour of a certain protocols or programs and compare it with set of pre-defined specification that describes the normal behaviour of a program or protocol. This type of IDS is capable of detecting previously unknown attacks with a low false positive rate.
- *Signature-based detection:* The IDS keeps track of known attacks or misbehaviour and compare them with the captured data traffic. However, it is vulnerable to unknown attacks. Different signature-based IDSs separate each other by their abnormal behaviour detection algorithms; the most common techniques include expert system [35], pattern recognition [15] and state transition analysis [49].
- *Anomaly-based detection:* The anomaly-based IDSs compare captured data traffic with the pre-deployed normal profiles. With appropriate matching algorithms, it reports any activities that deviate from the baseline to the administrator. This is suitable for detecting unknown attacks, but it comes with a high false positive rates.

On the other hand, Jie *et al.* proposed to classify them based on network architectures. The optimal IDS for a specific MANET are strongly related to the network structures [7]. The network architecture of MANETs can be divided into either flat or multi-layer. In a flat MANETs, all the mobile nodes are considered equal. Their functionalities and behaviours are identical. On the other hand, in a multi-layer MANETs, the entire network is divided into clusters. Nodes are free to directly communicate with other nodes within the same cluster. However, communication among different clusters must go through their cluster heads. This type of MANETs is generally implemented for military purpose.

In the following sections, we discuss IDSs for MANETs based on their structures.

3.2.1 Stand-alone Intrusion Detection Systems

In such architecture, IDSs are deployed on individual nodes to detect misbehaviours. No cooperation between different nodes exists nor is required. Nodes carry stand-alone IDSs independently. Due to the fact that no alert information is exchanged, nodes do not share any information with others. Obviously the advantage of such IDS is that it greatly eliminates overheads, which are the key fact of power consumption in MANETs. The disadvantage is the lack of information sharing greatly limits the effectiveness of such schemes. For this reason, such IDS architecture is not a popular choice among researchers. Despite this, stand-alone IDS may be useful in the situation when not all the nodes are capable of running IDS.

3.2.2 Distributed and Cooperative Intrusion Detection Systems

As the nature of MANETs is distributive and cooperative, it is a straightforward idea to develop IDSs based on a distributed and cooperative architecture. Zhang *et al.* suggested that intrusion detection and response systems should be both distributed and cooperative to suit the

needs of MANETs[66]. In this architecture, each node in the network participates in intrusion detection and is responsible for detecting misbehaviours independently and locally. Furthermore, a collaborative detection of neighbour nodes can be called to globally investigate a potential attack. Many IDSs designed for MANETs are based on this architecture; examples include [24][34][36][56][59][66]. Our proposed scheme can be classified in such category as well. The obvious benefit of distributed and cooperative detection is the effectiveness brought by global information sharing and cooperation. On the contrary, depending on the design features, constant information sharing in such IDSs consumes considerate amount of network throughput.

3.2.3 Hierarchical Intrusion Detection Systems

Hierarchical intrusion detection systems are mainly designed for multi-layered MANETs where the networks are divided into clusters. It can be viewed as an extension to the distributed and cooperative IDS architecture. Each cluster has a cluster head which acts like a switch, router or gateway in traditional wired network. Similar to the distributed and cooperative IDS architecture, each cluster node is equipped with an IDS agent and is responsible for detecting local misbehaviours. The cluster head node, on the other hand, is responsible for monitoring network packets delivered across clustered and react to misbehaviours in a global scale. This type of IDSs was adopted by Sterne *et al.* [59], Sun *et al.* [58] and Parker *et al.* [43].

More detailed classification and information about IDS can be referred to in [5][13][38][61].

3.3 *Sample Intrusion Detection Systems in MANETs*

As discussed in Section 3.1, the Watchdog scheme is one of the most important implementations of IDS for MANETs. Most research is either based or related to this work. Various attempts have been made to solve the six weaknesses of the Watchdog scheme.

As one attempt to solve the collusion problems of Watchdog, Patcha *et al.* [44] proposed an extension to Watchdog scheme. In their proposed research work, they suggest a method of only trusting the node that formed the network in the early stage. All the watchdog nodes are selected among these trusted nodes to prevent false report. Each watchdog node keeps track of two thresholds. The first one is the SUSPEND_THRESHOLD, which records the node's misbehaving times. The other one is the ACCEPTANCE_THRESHOLD, which measures the node's normal behaviour times. Neighbours of the watchdog nodes are classified based on these two thresholds. This scheme solves the collusion problems of Watchdog but remains vulnerable to all the other weaknesses.

In 2004, Parker *et al.* [43] proposed another approach to improve the Watchdog scheme. As discussed in Section 3.1.1, in Watchdog scheme, nodes only overhear its next node on the route. In the scheme proposed by Parker and his co-workers, the proposed solution let nodes overhear all other nodes in its proximity range instead of just one. On the other hand, this scheme has two response mechanisms, namely active and passive. In active response, the decision is done by the cluster head by initiating a voting procedure. If the majority decides certain nodes are misbehaving, the alert will be broadcasted to the entire network. The reported nodes will be blocked from network. With passive response, on the contrary, each node makes its own decisions. Eventually the misbehaving nodes will be blocked as well.

Balakrishnan *et al.* [34] proposed a scheme called TWOACK, which is an acknowledgement based protocol. It aims to solve the receiver collision and limited transmission power problem of Watchdog. TWOACK is neither an extension of Watchdog nor a Watchdog based scheme. Unlike Watchdog, it does not rely on overhearing other nodes to detect misbehaviour. On the contrary, it detects misbehaving nodes by acknowledging every data packets transmitted over every three consecutive nodes along the path from source to destination. When a node forwards a data packet to its next hop, it verifies the arrival of this packet by requiring acknowledgement from the node that is two hops away from itself down the route. This mechanism is achieved by a special type of acknowledgement packet called TWOACK, which stores a fixed route of two hops in the opposite direction to the data packet. This scheme can be added into a source routing protocol such as DSR [22]. The process is demonstrated in Figure 12. Node A send Packet 1 to node B and node B forward this packet to node C. Upon receiving Packet 1, node C is obliged to generate a TWOACK packet which contains the route from node C to node A and sends it back to node A. The receiving of this TOWACK packet at node A suggests Packet 1 is successfully forwarded to node C. Otherwise, if node A does not receive this TWOACK packet within a certain time period, it suspects node B or node C to be misbehaving. The same process is carried out along the entire route from the source node S to the destination node D.

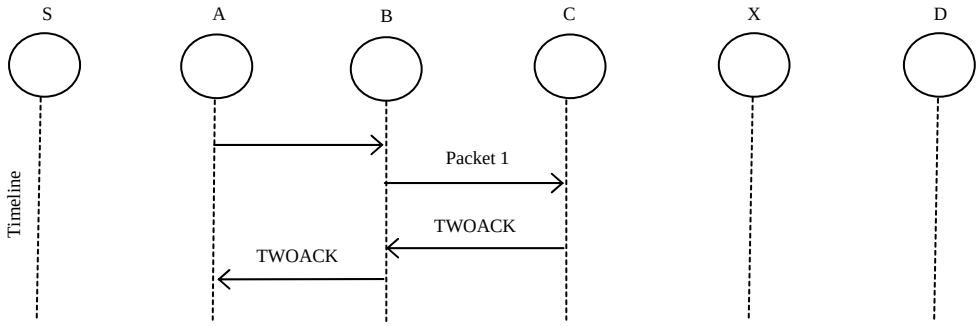

Figure 12 TWOACK Scheme

The disadvantage of TWOACK lies in the fact that every forwarded packet must be acknowledged. This adds considerable amount of unwanted overhead to the network. To address this issue, Balakrishnan and his colleagues also proposed S-TWOACK (Selective TWOACK) scheme to reduce the extra network traffic. The idea behind S-TWOACK is to send one acknowledgement packet for a number of data packets. Compared to TWOACK, S-TWOACK reduced the unwanted extra network traffic. However, as a trade-off, the network packet delivery ratio is inevitably harmed as it takes longer to detect misbehaving node with S-TWOACK. Furthermore, instead of detecting misbehaving nodes, S-TWOACK is only capable of detecting misbehaving links.

Based on TWOACK, Sheltami *et al.* [56] proposed a scheme called AACK (Adaptive ACKnowledgement). Similar to TWOACK and S-TWOACK, AACK focuses on solving the receiver collision and limited transmission power problem of Watchdog. The AACK is a network layer acknowledgement-based scheme which can be considered as a combination of TWOACK and end-to-end acknowledgement scheme. Compared to TWOACK and S-TWOACK, AACK significantly reduced network overhead while maintaining better performance [56].

In order to reduce the network overhead in TWOACK scheme. AACK adopted an adaptive mechanism that is switchable between two schemes. One is called ACK and the other one is called TACK. The AACK mechanism starts at ACK mode by default. ACK mode is basically a simple implementation of an end-to-end acknowledgement scheme. As demonstrated in Figure 13 (next page), in ACK mode, the source node S sends out Packet 1 without any overhead except for one bit of flag indicating the packet type (i.e. AACK or TACK). All the intermediate nodes between source node and destination node need to forward this packet. For the destination node, upon receiving Packet 1, it generates an ACK packet and sends it back to

the source node S in the opposite direction of route. If node S receives this acknowledgement within a certain period of time, the data transmission is successful. Otherwise, node S will switch to TACK mode where it works exactly like TWOACK scheme. This adaptive scheme greatly reduces the overhead compared to TWOACK scheme while maintaining the same detection performance. Aside from the advantages of AACK compared to TWOACK, AACK scheme still suffers from the fact that it fails to detect misbehaving nodes in the presence of false misbehaviour report in the network. Our proposed scheme is an improvement based on AACK mechanism, which is capable of solving the false misbehaviour report problem.

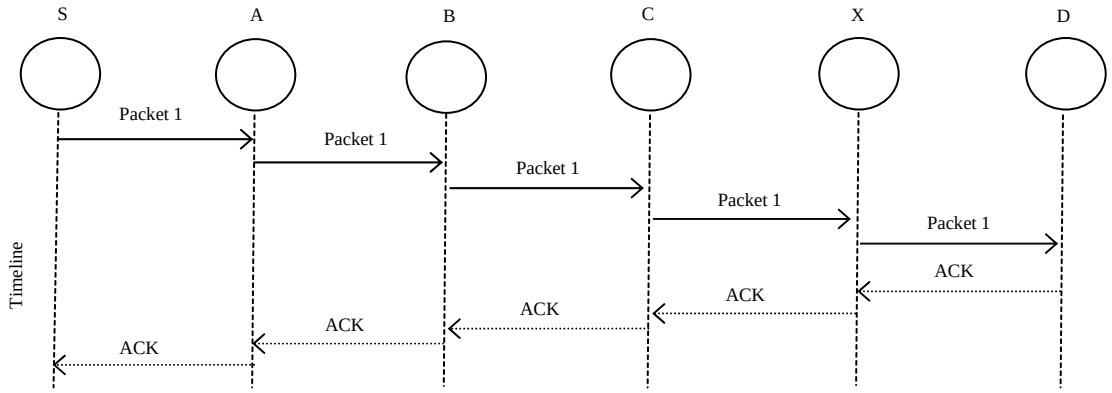

Figure 13 ACK Scheme

3.4 *Cryptography Techniques in MANET*

We presented a brief overview of MANET cryptography techniques in Section 2.5.3. A more detailed literature review will be provided in this section. As we discussed before, cryptography techniques in MANETs can be mainly divided in to three categories: symmetric cryptography, asymmetric cryptography and threshold cryptography [37]. In fact, among all these three categories, threshold cryptography is a unique one. It is based on Shamir's paper in 1979 [55]. Rather than a stand-alone category, the threshold cryptography is generally adopted in

combination with the other two techniques. Thus, in the following sections, we will mainly present the literature review with regarding to the first two categories rather than three.

3.4.1 Symmetric Cryptography in MANETs

As we discussed in Section 2, compare to other cryptography schemes, symmetric cryptography is far less computational intensive. This factor is of great importance due to the hardware limitation of mobile nodes. This efficiency factor makes symmetric cryptography the most popular choice for most security MANETs schemes. Symmetric cryptography mainly includes the following techniques [9]:

- *Random Nonce:* The random nonce is usually either a time stamp or a random number. The purpose of random nonce is to keep the packets fresh and prevent it from the replay attack [29]. Random nonce is typically generated by a pseudo random generator. The security level of such scheme depends on the design of this random generator, which is usually the weakest point of the security scheme. This cryptography technique is used in many MANETs security schemes such as ARAN [52], ARIADNE [18], SOLSR [10] and SPAAR [8].
- *Message Authentication Code (MAC):* MAC is calculated using a hash function and a secret key. Typical hash functions adopted in MANETs are MD5 or SHA-1. It can also be used to examine the completeness of the unencrypted data packets by calculating MAC with the secret key. Typical security schemes that uses MAC includes SOLSR [10] and the work proposed by Huang *et al.*. [19].
- *Hash Chain:* The hash chain scheme was first proposed by Lamport *et al.* [32] in 1981. The idea is to continuously apply a hash chain function to a message string. Due to the one-way nature of hash function, it is computationally impossible to do

reverse computing. Thus, cryptography goal is achieved. As it is straightforward and computationally less intensive, hash chain technique is widely adopted by various MANETs security schemes, examples include SAODV[64], ARIADNE[18], LHAP [70], LEAP [71] and SEAD[17].

It should be noticed that most MANETs security schemes adopts more than one cryptography techniques. This is usually due to trade-off between security level and performance concern. After reviewing symmetric cryptography techniques, we continue discussing some of the asymmetric cryptography techniques used in MANETs.

3.4.2 Asymmetric Cryptography in MANETs

As discussed before, asymmetric cryptography technique is also known as public key cryptography, where there is a pair of public/private keys. Public keys are readily accessible to all users while private keys are kept in secret. Knowing the public key, it is computationally infeasible to deduce the according private key. To encrypt a packet, sender encrypts it with the receiver's private key. To decrypt a packet, the receiver decrypts it with its own private key. Compare to symmetric cryptography, asymmetric cryptography's advantage is that only the private key needs to be kept secret. There is no need to develop complex key exchanging protocol. However, the disadvantages are that the key generation, encryption/decryption process requires much more computational power than symmetric cryptography. The cryptography technique used in asymmetric cryptography can be mainly divided into the following categories:

- *Certificate Authority (CA):* In public key cryptography, CA is usually considered as a part of great importance. It is responsible for issuing digital certificates for use by other parties. These digital certificates are used for certifying the ownership of public keys. Important as it is, as there is no fixed infrastructure of

MANETs, there exists a debate on whether it is even suitable to deploy CA in MANETs. Existing approaches that adopt CA include ARAN [52] and Kaya [30].

- *Digital Signature based on RSA/DSA:* Discussed in Section 2.5, digital signature using RSA/DSA is popular among security schemes. They are mainly used for message authentication and integrity check. Due to the fact that digital signature based on RSA/DSA is more expensive to compute compared to symmetric cryptographies like hash function, digital signature is not widely adopted by MANETs security schemes. The only implementation of digital signature in MANETs is only performed once in bootstrapping a TESLA key chain in LHAP [70] scheme. In our proposed scheme, we use digital signature within an adaptive scheme. In this way, we are able to ensure message authentication and integrity without consuming too much computational power.
- *Identity-based Cryptography:* This cryptography technique was first proposed by Rivest *et al.* in 1978 [50]. It uses the identity of users as the public keys. This scheme requires a certain amount of parties work together to decrypt a secret, which requires considerate amount of overhead. Example implementation in MANETs includes IKM [65] and AC-PKI [67].

4 DESIGN OF PROPOSED SCHEME

4.1 *Overview*

As we discussed in the previous chapters, the misbehaving nodes in MANETs can degrade the network performance or even break down the entire network by dropping packets. Packet dropping can be caused by various factors including selfish node, malicious node or even hardware problems like buffer overflow. In this research, we concentrate on selfish node and malicious node. The reason being they are more likely to cause significant harm to the network performance.

Selfish nodes drop packets to conserve their battery power, while malicious nodes generally drop packets because they are compromised by attackers. Due to the fact that the output of both types is to drop packets, we use malicious node to refer to both selfish and malicious nodes in the rest of this thesis. Discussed in Section 2.3, there exist two types of packet dropping attack, namely black hole attack and grey hole attack. Both attacks are conducted in network layer and can be categorized as a type of Denial of Service (DoS) attack. In the black hole attack, malicious nodes drop all the data packets they receive. On the other hand, in the grey hole attack, malicious nodes are smart enough to drop only part of the data packets they receive. Grey hole attackers may successfully avoid being detected by adjusting their packet dropping rates to the detection threshold. In general, gray hole attacks are more difficult to be detected.

In this research, we concentrate on detecting black hole attacks. In general, black hole attacks are capable of affecting the network performance in a much larger margin. With enough compromised nodes, black hole attackers can easily harm the availability and connectivity of the

entire network, and cause network divisions in some circumstances. Furthermore, as an important factor of IDS, false alarms are more common under grey hole attacks.

An early solution against packet dropping attacks in MANETs is the Watchdog scheme [36]. As discussed in Section 3.1.1, Watchdog is one of the most popular IDSs in MANETs, and many other researches are either based upon or closely related to this technique. It is based on the concept of letting nodes overhear the node in its next hop along the route. With almost no network overhead, this concept is simple and clear and it poses positive performance in the presence of malicious nodes. However, as pointed out by its author in [36], it suffers from the problem that it fails to detect malicious nodes when any of these six circumstances exist: 1) ambiguous collisions, 2) receiver collisions, 3) limited transmission power, 4) false misbehaviour report, 5) collusion and 6) partial dropping. Detailed information regarding these six weaknesses is discussed in Section 3.1.2. In this research work, we concentrate on resolving three of the six weaknesses of Watchdog. They are receiver collisions, limited transmission power and false misbehaviour report:

- *Receiver Collisions*: While monitored by the nodes at its previous hop, the attackers send out the packet exactly when the intended receiver is busy with other transmissions. In this case, the attackers cheated the Watchdog scheme while in fact dropped the packet.
- *Limited Transmission Power*: The attackers can control the compromised nodes to limit its transmission power so that it is strong enough to be overheard by Watchdog nodes, but too weak to be received by the destination nodes.
- *False Misbehaviour Report*: The attackers compromise the Watchdog nodes and control it to send out false misbehaviour report. Due to the open medium in

MANETs, attackers can easily capture and compromise nodes to achieve such an attack.

Balakrishnan *et al.* [34] proposed an acknowledgement-based approach to verify the delivery of packets in every three consecutive nodes along the route. TWOACK successfully solved the receiver collision and limited transmission power problem in Watchdog. However, the acknowledgement scheme in TWOACK requires considerable extra network overhead, as the acknowledgement process is required on each transmission. Later after TWOACK, Sheltami *et al.* [56] proposed another scheme called AACK, which combines an adaptive scheme with TWOACK. With introduction of these hybrid schemes, AACK successfully reduce the overhead problem in TWOACK. Nevertheless, AACK is still vulnerable to false misbehaviour report attack. On the other hand, both schemes are vulnerable when the attackers are smart enough to forge acknowledgement packets.

In this research, we propose a new mechanism called Enhanced Adaptive ACKnowledgement (EAACK). Although evolved from AACK, EAACK differs from AACK in the fact that it is not only capable of detecting false misbehaviour attack, but also is able to ensure authentication and packet integrity. These enhancements are brought by the introduction of a new scheme called Message Receiving Authentication (MRA) and digital signature. Details of these two improvements are presented in the Section 4.4.

4.2 *Assumptions*

In our research, we make the following assumptions:

- All of the schemes we investigated and compared in simulation is based on DSR [22] routing protocol. The reason being Watchdog requires DSR routing protocol.

In order to evaluate performances, it only makes sense to compare all schemes on the same routing protocol.

- We always assume that the nodes in MANETs are capable of committing bi-directional communication. Due to the nature of MANETs, this assumption is generally valid in most scenarios.
- Even though malicious nodes drop data packets, they still participate in the routing activity. Furthermore, malicious nodes always try to conceal themselves from being detected.
- During any packet transmission, source node and destination node are not malicious. Because if they are malicious, it is no longer a packet dropping attack, which will be considered in our future research.
- Cooperation attack with multiple malicious nodes is not investigated in this work. This possible attack will be investigated in future research.
- Each node in MANETs is pre-distributed with a public key set which contains all the other nodes' public key.

4.3 *Network Behaviours*

Typically, in order to evaluate the performance of an IDS, we need to design two types of nodes. One type is the regular nodes; they represent the normal behaviour of MANETs. The other type is the malicious nodes; they are responsible for the attacks on the network. By studying their behaviours, readers can have a better understanding of our proposed scheme. In this section, we first give a brief description on the type of data packets in EAACK. This is followed by a study on node behaviours in our proposed scheme.

4.3.1 Packet Description

Two categories of packets are implemented in the EAACK approach: data packets and acknowledgement packets.

Data Packets

The proposed EAACK scheme can be mainly divided into three parts, namely ACK (ACKnowledgement), S-ACK (Secure-ACKnowledgement) and MRA (Misbehaviour Report Authentication). Details of these three parts are discussed in Section 4.4. Three different types of data packets are assigned to each part. Take the normal data packet into account, there are overall four types of packets in the process of data communication. To distinguish between them, we mark each type of packets with a packet flag. This flag information is stored in the header of DSR header. According to the internet draft of DSR [22], there are six bits reserved in DSR header. In our proposed scheme, we use two of the six bits to store the packet flag. In this case, we define general data packet as “00”, ACK data packet as “01”, S-ACK data packet as “10” and MRA packet as “11”. All data packets format is listed in Table 1. In fact, general data packet and MRA data packet are not currently available in EAACK, but we reserved it for future work and potential test work.

Table 1 Data Packet Types and Data Packet Flags

Packet Type	General Data Pkt	ACK	S-ACK	MRA
Packet Flag	00	01	10	11

Acknowledgement Packets

As an acknowledgement based IDS mechanism for MANETs, EAACK have three types of acknowledgement packets:

- *ACK Acknowledgement Packet*: It contains the received data packet's packet ID and signed by the destination node. Destination node send out ACK acknowledgement packet to source node to acknowledge it that the packet has been successfully received. Details of ACK scheme can be found in Section 4.4.1.
- *S-ACK Acknowledgement Packet*: This packet contains hashed value of the received data and signed by the receiver that is two hops away from the data packet sender. Detail of S-ACK description can be referred to in Section 4.4.2.
- *MRA Acknowledgement Packet*: Similar to S-ACK acknowledgement packet, this packet contains hashed value of the received data packets and signed by the destination node. Details are discussed in Section 4.4.3.

4.3.2 Regular Node Model

In order to implement the proposed EAACK mechanism, the regular node's behaviours must be adjusted. In this section, we will discuss these modifications in details. We use Network Simulator 2 (NS2) as our tools for network simulation. According to NS2 [75], all the regular nodes in MANETs can be divided into three categories based on their operations:

- *Source Node*: The originator of one packet transmission.
- *Forwarder Node*: In multi-hop MANETs, forwarder node is responsible for receiving and forwarding the packet to its next hop along the route.
- *Destination Node*: The final destination of one packet transmission.

A typical scenario describing the three nodes is shown in Figure 14. In this figure, Node S and node D are the source node and destination node respectively. F1, F2, F3 and Fx are the forwarder nodes.

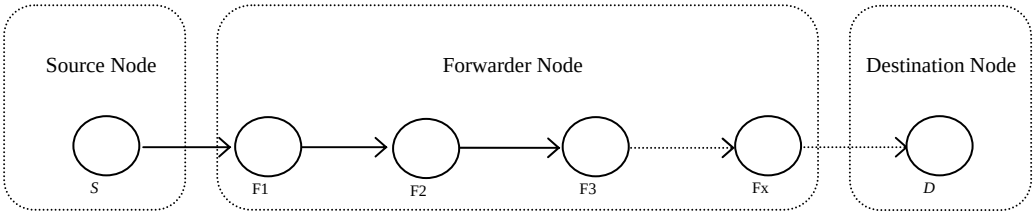

Figure 14 Type of Nodes

All of these three types of node will participate in our proposed EAACK scheme. As a result, some modifications are necessary to accommodate our scheme. The details will be presented below:

Source Node

Aside from sending data packet to destination nodes, source nodes have two other responsibilities.

- *Switching Schemes*: As discussed before, overall three schemes are included in EAACK, namely ACK, S-ACK and MRA. Only source nodes are capable of switching between schemes. Source nodes switch schemes by sending out data packet accordingly. For example, to switch to S-ACK mode, source nodes send out S-ACK data packet.
- *Verifying Packets*: In ACK scheme, source nodes are responsible for verifying the ACK acknowledgement packet sent from destination node. In S-ACK scheme,

source nodes are required to verify the S-ACK acknowledgement packet sent from the node that is two hops away from itself along the route. Similar to these two, source nodes are also responsible for verifying the MRA acknowledgement packets sent from destination node.

Forwarder Node

Forwarder nodes work according to type of received data packets and each data packets are designed for a different scheme in EAACK. There are overall three schemes in our proposed scheme, we discuss forwarder nodes' behaviour based on this classification:

- *ACK*: Upon receiving ACK packets, whether they are data packets or acknowledgement packets, forwarder nodes simply forward these packets to its next hop.
- *S-ACK*: As we will further discuss in Section 4.4, S-ACK is based on TWOACK [34] mechanism, where acknowledgement is done from nodes two hops away. To discuss how the forwarder node works in S-ACK mode, we need to divide route into several three consecutive nodes groups. For example, as demonstrated in Figure 14, F1, F2 and F3 is a three consecutive nodes group. In this case, when F2 receives the S-ACK data packets, as it is only one hop away from the sender F1, it simply forwards the S-ACK data packets to F3. When F3, who is two hops away from the data packets sender, receives the S-ACK data packets, it is responsible for generating an S-ACK acknowledgement packet and sends it back to F1.
- *MRA*: Forwarder nodes have no privileges to modify MRA packet, so it simply forward it according to the route.

Destination Node

Similar to forwarder node, destination node also responds based on the type of packets it receives. For this reason, we continue discussing its behaviour according to the type of packets it receives:

- *ACK*: When an ACK packets arrives the destination node, it is required to generate an ACK acknowledgement packet and send it back to the source node by reversing the existing route.
- *S-ACK*: Upon receiving of an S-ACK data packets, the destination node works the same way as the forward node does. In addition, it is responsible of generating an ACK acknowledgement packet and sends it back to the source node.
- *MRA*: When the destination node receives an MRA data packet, it searches its local memory and compare if this packet has already been received. Either way, an MRA acknowledgement packet will be sent back to the source node indicating whether the data packet was previous received or not.

4.3.3 Malicious Node Model

In this research, all misbehaving nodes are described as malicious nodes regardless of whether they are selfish or compromised. The reason being they achieve the same packet dropping attack to the network. As discussed in Section 4.2, we assume all malicious nodes will cooperate in the routing discovery period but drop data and acknowledgement packets whenever possible. The target of our proposed scheme EAACK is to solve the three problems of Watchdog technique, namely receiver collision, limited transmission power and misbehaviour report. In addition to that, EAACK also address the potential attack when the attackers are smart enough to

forge acknowledgement packets. We describe the malicious node model behaviour according to the type of nodes as discussed in Section 4.3.2. According to Section 4.2, we made the assumption that the source node and destination node are not malicious. In this section, we only discuss the malicious behaviour of the forwarder node.

In Section 3.1.2, we discussed the detail of receiver collision and limited transmission power attacks against Watchdog technique. The concepts between these two attacks are different, but they share the same result of dropping the intended data packet. To simulate these two misbehaviours, we modified part of the mobile nodes in the network so that whenever a data packet received, it will be dropped. Furthermore, regarding the simulation of misbehaviour report and forged acknowledgement packet, we designed three scenarios to evaluate the performance of our proposed scheme against all the attacks mentioned above.

Malicious Scenario 1

To test the performance of our proposed scheme under the receiver collision and limited transmission power attack, we designed a scenario. In this scenario, the malicious node drops any data packet and acknowledgement packet whenever one arrives. Basically, this is a simple packet dropping attack. Meanwhile, the nodes are modified to achieve receiver collisions, as discussed in previous chapters.

Malicious Scenario 2

This scenario is designed to test the performance of EAACK when the attackers generate false misbehaviour report in S-ACK scheme. Figure 15 (next page) demonstrates an example of such scenario.

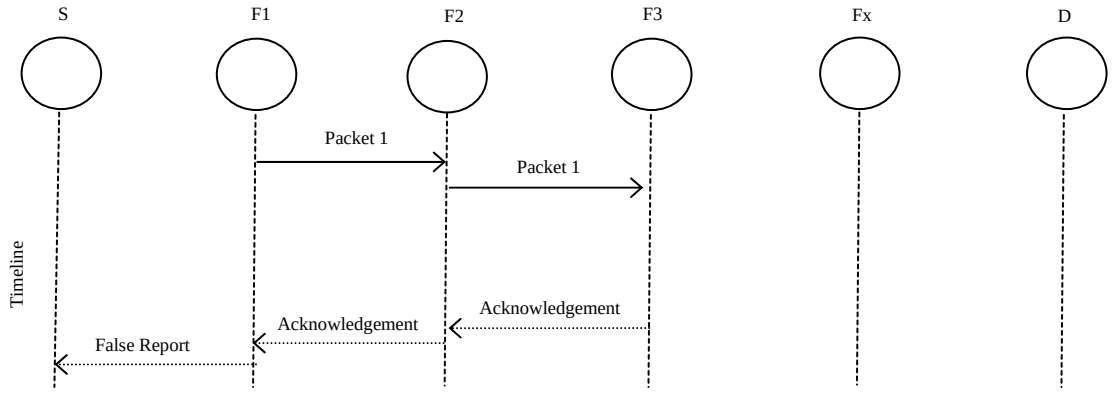

Figure 15 Malicious Nodes Scenario 2

As shown in Figure 15, Packet 1 was forwarded to F3 from F2. As F3 is the second hop node from F1, it is required to generate an acknowledgement packet and send it back to F1. This action is carried in AACK, TWOACK and our proposed scheme EAACK. Then, F3 generates an acknowledgement packet and sends it back to F2, follows by F2 forwarding this acknowledgement packet to F1. In Scenario 2, instead of confirming receiving this packet, F1 drops it and falsely report F2 and F3 as misbehaving. By repeatedly doing this, the smart attackers can break down the entire network by dropping the acknowledgement packet and report other nodes as misbehaving.

Malicious Scenario 3

In the case when the attackers are smart enough to forge acknowledgement packets, all acknowledgements based IDS, including AACK and TWOACK, we discussed in this thesis will be rendered in danger. This scenario is discussed in Figure 16 (next page). In this case, F2 is the malicious node. As shown in the figure, F1 forward Packet 1 to F2. Instead of forwarding this packet to F3, F2 drops the packet. To protect itself from being reported, F2 forges an acknowledgement packet and sends it to F1, claiming this acknowledgement packet is from F3.

By doing this, attackers can achieve a black-hole packet dropping attack without being detected by mechanisms like TWOACK or AACK.

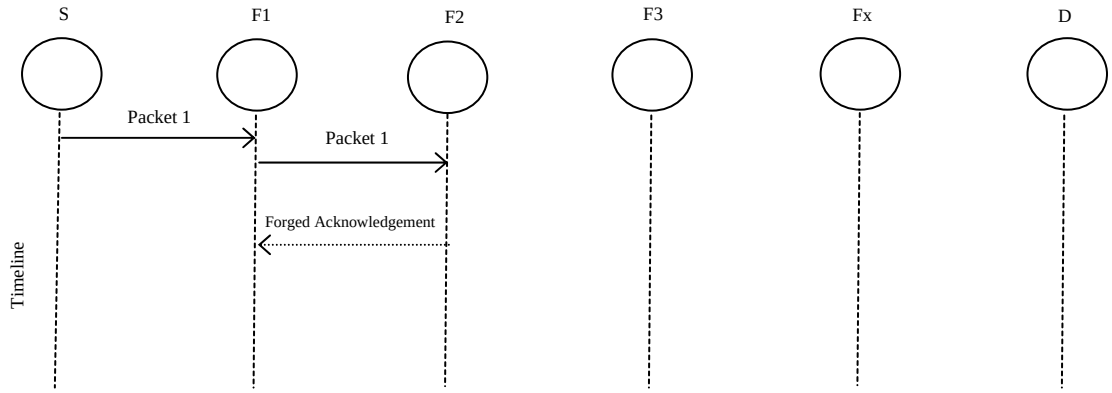

Figure 16 Malicious Nodes Scenario 3

In the next section, we discuss the details of our proposed scheme EAACK, and how it reacts to these three attack scenarios.

4.4 *Scheme Descriptions*

In this section, we present detailed descriptions of our proposed scheme EAACK. As discussed before, EAACK mechanism can be divided to three schemes, namely, ACK, S-ACK and MRA. A basic flowchart describing EAACK scheme is demonstrated in Figure 17 (next page).

4.4.1 ACK

ACK is simply an end-to-end acknowledgment scheme. This simple scheme is included in EAACK as a part of the hybrid scheme. The introduction of ACK brings extremely low network overhead during packet transmission in MANETs. Due to the fact that nodes in MANETs are generally limited by its energy and computational power, to preserve the life cycle

of mobile nodes, it is vital to keep battery energy consumption at a minimal level. ACK is an ideal solution to this problem for two reasons:

- *Low Network Overhead*: ACK packet is no difference compared to ordinary data packets other than one bit in the packet header which indicates its packet type. Furthermore, it does not require constant acknowledgement packet exchange like TWOACK.
- *Low Memory Consumption*: In ACK, the forwarder nodes are not required to store the packet information. Compare to TWOACK, it barely consumes any memory space in all nodes.

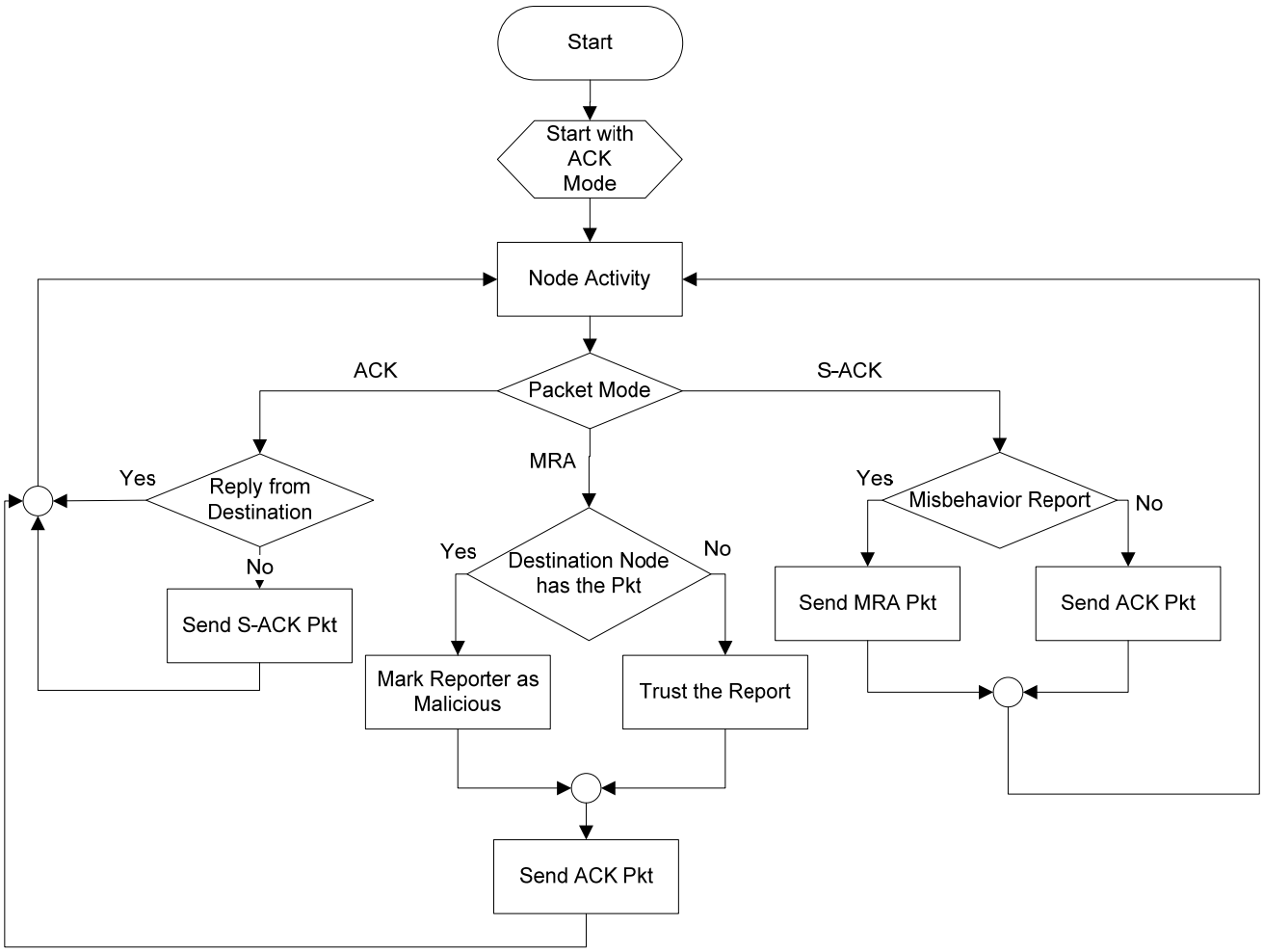

Figure 17 EAACK Scheme

To better show how ACK works, we will describe it in an example. As demonstrated in Figure 18 (next page), in ACK mode, source node S first searches its local knowledgebase and sees if there exists a route from node S to node D. If not, it will initiate a DSR routing request and find the route, denoted as R_{SD}. Then node S sends out an ACK data packet P_{ad1} to a forwarder node F1 along the route specified by R_{SD}. Upon sending out P_{ad1}, source node S stores a hash value of this packet h_{adp1} along with its sending time t_s. Starting from F1, the rest of the forwarder nodes (i.e. F1, F2, F3, etc.) simply forward the received data packet to their next hop. Providing there are no malicious nodes along the route, the destination node D will eventually receive this packet. Then, node D generates an ACK acknowledgement Packet P_{ak1} which contains a hash value of the received packet, denoted as h_{adp1}', and a digital signature $\Pr_d(h_{adp1}')$. Packet P_{ak1} is sent to source node S along the reversed order of route R_{SD}. When node S receives this packet P_{ak1}' (denoted as P_{ak1}' because it has not been authenticated yet), it verifies the signature $\Pr_d(h_{adp1}')$ with the node D's public key Pk_d, which is pre-distributed to each nodes in the network, as in:

$$Pk_d(\Pr_d(h_{adp1}')) = h_{adp1}' \tag{7}$$

If $h_{adp1}' == h_{adp1}$, it indicates the acknowledgement packet P_{ak1}' is indeed sent from node D to acknowledge that it has already received the complete and uncontaminated data packet P_{ad1}. Up until this point, the packet transmission is successful and complete.

On the other hand, after a certain period of time t_{out}, if node S still does not receive the correct acknowledgement packet from Node D, it suggests potential misbehaving node along the

route. In this case, source node will switch to S-ACK mode to detect the misbehaving nodes. As discussed in Section 4.3.2, source node switch to S-ACK mode by sending out an S-ACK packet.

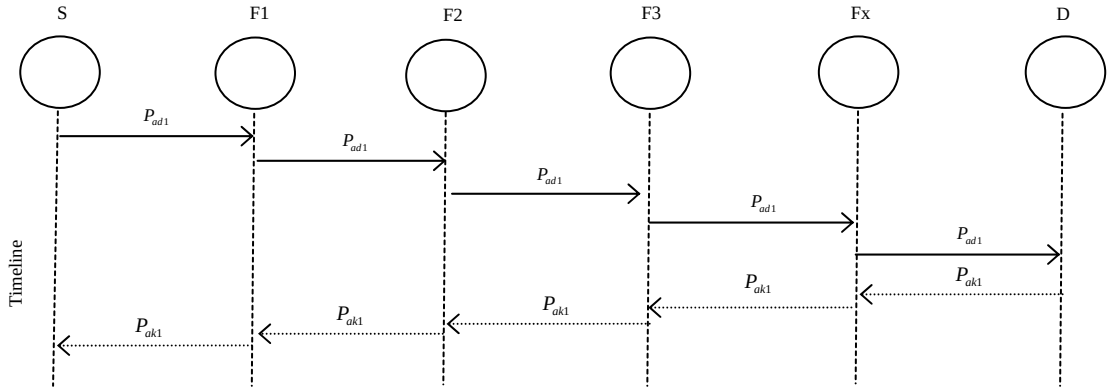

Figure 18 ACK Scheme

4.4.2 S-ACK

In S-ACK mode, every three consecutive nodes work in a group to detect misbehaving nodes. The intension of including S-ACK in EAACK mechanism is to detect misbehaving nodes in the presence of receiver collision and limited transmission power, which are the two major weaknesses of Watchdog technique. This scheme is an extension to TWOACK. We extended it to include digital signature scheme. By doing this, we can eliminate the possible attack that the attackers are smart enough to forge acknowledgement packets.

Again, we will describe this scheme with the help of an example. As shown in Figure 19 (next page), we take node F1, F2 and F3 as a group of three consecutive nodes. The concept of S-ACK is that that the third node (i.e. F3) which is two hops away from the first node (i.e. F1) is required to generate an acknowledgement packet (i.e. P_{sak1}) and send it back to the first node upon receiving a data packet (i.e. P_{sad1}). At this point, we have no clue whether if any of the F1, F2 or F3 is malicious or not. S-ACK reacts differently to different situation, so we divide our

discussions to the following possible scenarios. Please note that due to the fact that the potential scenarios discussed in this section are all closely related to S-ACK scheme, the detecting processes share the same pattern. Thus, we only concentrate on describing the scenarios when there are no malicious nodes along the route. For other scenarios, they are all based on the same scheme; we only get into details when S-ACK reacts differently.

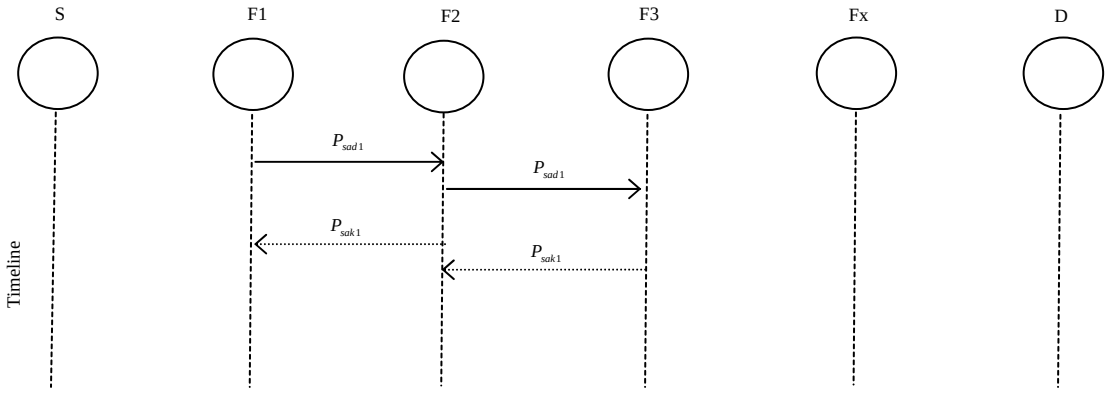

Figure 19 S-ACK Scheme

None is Malicious

In this scenario, similar to ACK scheme, F1 first searches its local knowledge base to see if there is a route $R_{F_1F_3}$ from F1 to F3. If not, it will initiate a DSR routing request to find a new route to F3. Then F1 send out an S-ACK data packet P_{sad1} to F3 along the route $R_{F_1F_3}$, which in our case is F1-F2-F3. Upon sending out P_{sad1}, source node S stores a hash value of this data packet h_{sadp1} together with its sending time t_s. Next, F2 receives this data packet P_{sad1} and simply forwards it without doing anything else. When F3 receives this data packet P_{sad1}, it generates an S-ACK acknowledgement packet P_{sak1} which contains a hash value of the received data packet, denoted by h_{sakp1}', and a digital signature $\Pr_{F_3}(h_{sakp1}')$. Then F3 send P_{sak1} back to F1 along the

reverse route of $R_{F_1F_3}$. When F1 receives this acknowledgement packet P_{sak1}, it verifies the digital signature $\Pr_{F_3}(h_{sakp1}')$ by computing it against F3's public key Pk_{F_3}. Formally, it is described as follows:

$$Pk_{F_3}(\Pr_{F_3}(h_{sakp1}')) = h_{adp1}' \quad (8)$$

As we made the assumption that there are no malicious nodes among F1, F2 and F3, then we shall find $h_{sakp1}' == h_{sakp1}$, which indicates the packet transmission from F1 to F3 is successful and the route is safe.

F1 is Malicious

TWOACK and AACK assume that F1 cannot be malicious as it successfully participates in the packet transmission with its previous two nodes. This assumption is valid to most general packet dropping attacks. But when the attackers are designed to participate in packet transmission whenever they are not the sender node in the three nodes cooperation; such assumption is not valid anymore. This is because the attackers can still harm the network by falsely reporting other nodes as malicious. As discussed in Section 3.1.2, this is a false misbehaviour attack. This type of attack can actually contribute much more damage to the network compared to regular packet dropping attack. By constantly reporting other normal nodes as malicious, the entire network can soon be taken apart because of the lack of available nodes.

To address this false misbehaviour problem, we added special schemes in EAACK to handle this situation. In TWOACK, when a malicious report is generated by the sender node of the three consecutive nodes, the sender node send this report back to the source node and the

source node will notify other nodes that the reported nodes are malicious. Details of IDS response system can be found in Section 4.4.4. Unlike TWOACK, in EAACK, when the source node receives such misbehaviour report, instead of instantly trusting the report, MRA scheme is initiated to verify the correctness of such malicious report. Details of MRA scheme will be discussed in Section 4.4.3. This scenario is demonstrated in Figure 20.

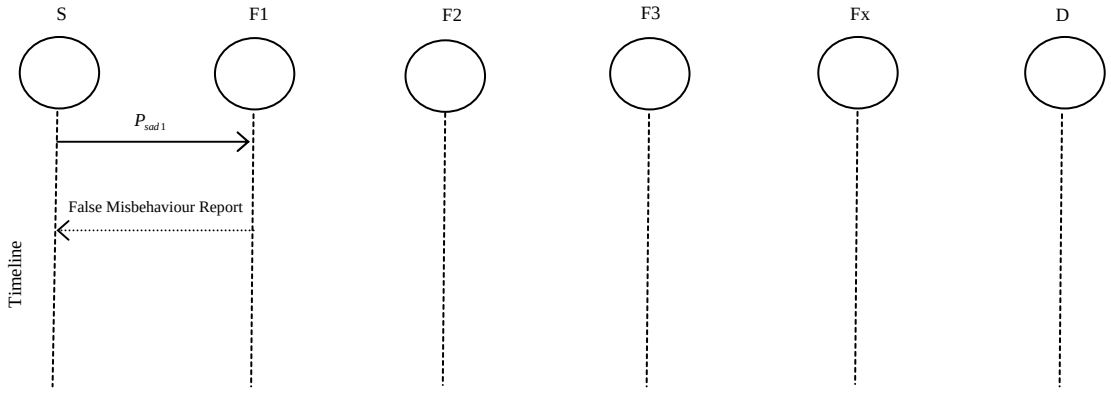

Figure 20 S-ACK: F1 is malicious

F2 is Malicious

When F2 is malicious, F2 drops all S-ACK data packets forwarded to it. As demonstrated in Figure 21 (next page), F1 forwards P_{sad1} to F2, but F2 drops this packet without forwarding it to F3. If this is the case, acknowledgement scheme like TWOACK and S-ACK can easily detect F2 and F3 as malicious link as no acknowledgement packets are received by F1. However, when the attackers are smart enough to forge these acknowledgement packets, TWOACK scheme is not able to detect this attack. For S-ACK, as we discussed, F3 is required to provide a digital signature along with the acknowledgement packet. This prevents the attackers from being able to forging these acknowledgement packets.

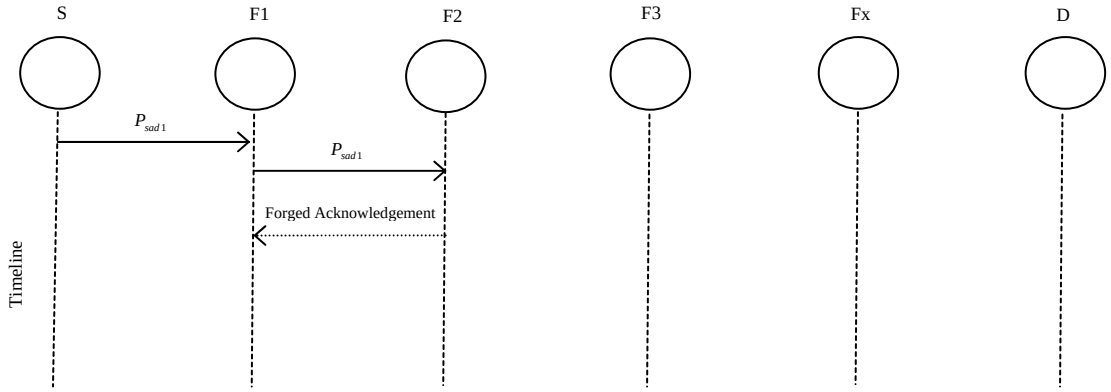

Figure 21 S-ACK: F2 is Malicious

F3 is Malicious

In this case, as demonstrated in Figure 22, if F3 refuses to send back acknowledgement packets, it will be marked as malicious. As far as this research is concerned, F3 needs to cooperate in the acknowledgement process in order to concede itself from being detected.

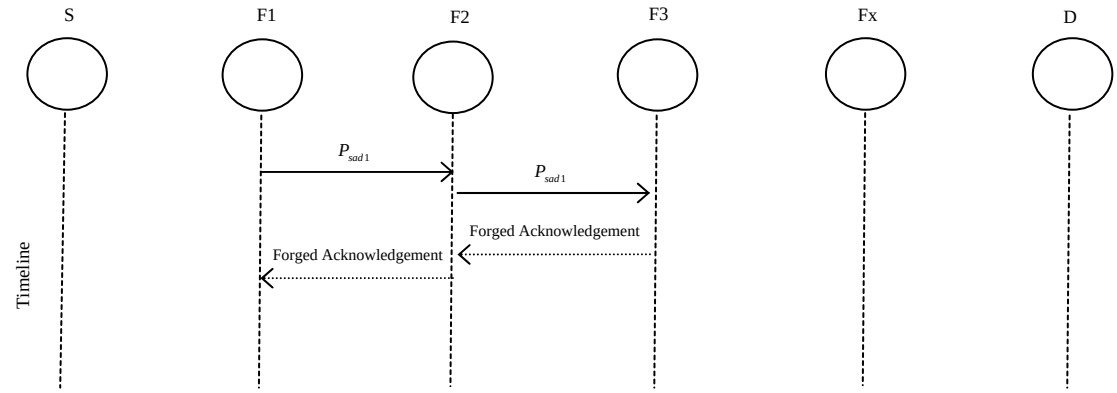

Figure 22 S-ACK: F3 is Malicious

4.4.3 MRA

The Misbehaviour Report Authentication (MRA) is designed to resolve the false misbehaviour report attack. The false misbehaviour attack, discussed in Section 4.3.3, is a smart attack when the malicious nodes send out a misbehaviour report that falsely reports other nodes as malicious. This attack can cause much more serious problem than regular packet dropping attack. By falsely report other nodes as malicious, such attack can break down the entire network. To address this attack, we propose to include MRA scheme in EAACK.

The concept of MRA scheme is to authenticate whether the destination node has received the reported missing packet. To initiate MRA scheme, the source node sends out an MRA packet to the destination node through a different route. If an alternative route does not exist in the source node's knowledge base, it will initiate a DSR routing request to find a new route. Due to the nature of ad hoc network, it is common to have multiple routes from one node to another. By adopting an alternative route to the destination node, we circumvent the false misbehaviour reporter. The MRA packet contains the ID of the packet that has been reported dropped. When the destination node receives the MRA packet, it is required to search its local knowledgebase and see if there is a match of the reported packet. If it already existed, then we can conclude that the reported dropped packets have been received and whoever sends out this misbehaviour report is malicious. Otherwise, the misbehaviour report is trusted. The destination node acknowledges the source node by sending back an MRA acknowledgement packet.

By adopting MRA scheme, we eliminate the possible attack when the attackers send out false misbehaviour report and try to break down the network by falsely marking other nodes as malicious.

4.4.4 Response System

As an IDS designed specifically for MANETs, our proposed mechanism EAACK needs a response system to respond to misbehaviours detected by the detection system, in which case, are the combination of ACK, S-ACK and MRA. As the MANETs we discuss in this thesis have a flat topology, all nodes within the network are equal and shall be capable of detection and response.

In our proposed scheme, when a misbehaviour report is authenticated and verified by the MRA scheme, the source node is responsible for sending out alarm to all the other nodes in the same network. This alarm contains the malicious nodes' ID and signed by the source node. When other nodes receive this alarm, providing the signature is properly authenticated, the reported malicious node will be directly banned from accessing the network. The reason we did not implement the popular threshold scheme is that we believe our proposed mechanism is secure compared to other mechanisms we investigated through this research. Furthermore, this is also a compliment to the extra energy consumption we spent on digital signatures.

5 SIMULATION AND PERFORMANCE EVALUATION

In this chapter, we describe our methodology on evaluating the performance of our proposed mechanism EAACK by simulation. To better evaluate the performance of EAACK, we implemented Watchdog, TWOACK, AACK and EAACK in the same network settings. Through testing the performance of each mechanism in various scenarios, we present a valuable performance evaluation.

5.1 *Introduction to Simulation*

In this section, we present a brief overview on two important tools we used in our research, namely the Network Simulator 2 (NS2) and the Botan cryptography library.

5.1.1 Simulation Environment

In order to conduct proposed simulation, we deployed NS2 and Botan library on Ubuntu 9.10 with GCC-4.3 as its default compiler. The system is running on a laptop equipped with Core 2 Duo T7250 CPU and 3GB RAM.

The software version of NS2 in our simulation was 2.34, it came with a wireless extension by the CMU Monarch project. The version of selected Botan crypto library is 1.8.11. These are the most up-to-date packages when we began our simulation experiment.

5.1.2 Overview of NS2

Network Simulator 2 (NS2) is a discrete event simulator targeted at networking research. It provides substantial support for simulation of TCP, routing, and multicast protocols over wired and wireless (local and satellite) networks [75]. Starting from 1989, NS2 was first developed as a variant of REAL network simulator [76]. After that, NS 2 development was supported by many

organizations like DARPA, Xerox PARC, UCB and USC/ISI. It has long been considered as a great simulation tool for researchers. According to [31], a survey of simulation-based papers in ACM's international symposium on Mobile Ad Hoc networking and computing (MobiHoc) 2000-2004 shows that over 44% of the papers used NS2 as their simulation tool. A detailed description of NS2 can be found in [77]. Furthermore, various tutorials to NS2 are available over the Internet, examples including [78] and [79].

NS2 can be mainly divided into two parts based on their functionalities. The first part is written in TCL [79] language, it is mainly responsible for setting up the simulation configuration and running user's customized scripts. The other part is written in C++ language, it is responsible for running core simulation engine, agents and event schedulers. The control parameters and functions of the C++ compiled objects are exposed to the OTcl interpreter via OTcl linkage. For every OTcl object invoked in the interpreter hierarchy, there is a mirrored object created in the C++ hierarchy [75].

The major advantage of such dual design is efficiency. Compare to scripting language TCL, C++ is a much more efficient programming language. To ensure the efficiency of NS2, it is ideal to write the simulation engine in C++. On the other hand, research simulations are usually exposed to constant parameter changing. TCL as a scripting language provides an easy and convenient way to change simulation configuration parameters without the need of constantly recompiling the whole NS2 package. The compilation and validation process of NS2 on our experiment workstation takes roughly 40 minutes for each run. Therefore, it is best to implement NS2 in this hybrid style.

5.1.3 Overview on Botan Cryptography Library

In order to implement digital signature technique in our proposed simulation, we introduced Botan Crypto library [80] to our research. Botan Cryptography library is a C++ crypto library released under BSD-license. It provides applications with the ability to use a number of cryptographic algorithms, including DSA and RSA. In this research, we implemented both DSA and RSA scheme in our proposed mechanism. They are both capable of generating digital signatures. We included both of them to compare the performance differences.

5.2 *Simulation Configurations*

5.2.1 Network Simulator

We chose to run our simulation on the default network settings in NS 2.34. The reason being the default configuration is likely to bring more typical results and makes it easy to compare the simulation results with others work. Our simulation contains 50 nodes randomly distributed in a flat space with the size of 670x670m. The maximum hops allowed in this configuration are four and the physical layer and 802.11 MAC layer are included in the wireless extensions of NS2. To simulate different network topology, for each mechanism we run three distinct network scenario and calculate the average performance. The moving speed of a mobile node is set between the range of 1 to 20m/s. The UDP traffic with Constant Bit Rate (CBR) is implemented through a packet size of 512 bytes. A list of the configuration parameters' setting is demonstrated in Table 2.

Table 2 Simulation Settings

Parameter Name	Value
Channel Type	Wireless Channel
Propagation Model	TwoRayGround
Mac Protocol	802.11 CSMA/CA
Dimension of Topology (x)	670m
Dimension of Topology (y)	670m
Antenna Model	Omni-directional Antenna
Number of Nodes	50
Simulation Time	1000 second
Agent Trace	ON
Router Trace	ON
Node Moving Speed	1-20m/s
CBR Packet Size	512 Bytes
Transmission Range	250 meter
Routing Protocol	DSR
Watchdog Timeout	0.1 second
Watchdog and TWOACK Threshold	40 packets
AACK and EAACK threshold	30 packets

5.2.2 Botan Crypto Library

Thanks to the powerful open source Botan cryptography library, we are able to conveniently implement DSA and RSA algorithms through well documented API. A detailed documentation on Botan reference can be found in [81].

We implemented both DSA and RSA algorithms in our proposed IDS mechanism. As discussed in previous chapters, mobile nodes in MANETs are generally limited to its limited computational and battery power. To implement asymmetric cryptography in such environment, it is vital to choose the most efficient way to do so. By comparing their performances, we plan to find out the most optimum digital signature scheme for MANETs.

In our cryptography implementation, we generated a 1024 bit DSA key and a 1024 bit RSA key. A pair of private/public key file is generated for each node in the network. Typical size of public key and private key file is 654 bytes and 509 bytes with a 1024 bit DSA key respectfully. On the other hand, the size of public key and private key file with 1024 bit RSA key is 272 bytes and 916 bytes respectfully. The reason why we chose this key size is that NIST specified in Digital Signature Standard (DSS) [72] that the minimum key length for DSA is 1024 bit. In order to make the performance comparison consistent and valid, we decided to adopt this key size. Due to the fact that DSA algorithm in Botan adopted SHA-1 as their hash function, the resulting signature file for arbitrary length of packet is always 89 bytes. On the other hand, the digital signature generated by RSA is 131 bytes.

In terms of computational complexity and memory, we did a research on popular mobile sensors. One of the most popular sensors on the market is Tmote Sky [82]. This sensor is equipped with a TI MSP430F1611 8MHZ CPU and overall 1070 KB of memory space. We believe it is capable of handling 1024 bit DSA with respect to both memory space and

computational power. Regarding memory space in our proposed mechanism, for node, the pair of public/private key together consumes around 1100 bytes for both DSA and RSA scheme. This adds up to about 55KB in total for all 50 nodes in the simulation. Regarding the computational power, the capability of mobile sensors to handle asymmetric cryptography has been proven through various research works as we discussed in Chapter 2.

5.3 *Performance Metrics*

To measure the performance of our proposed schemes, we introduce the following two performance metrics:

- *Packet Delivery Ratio (PDR)*: PDR defines the ratio of the number of packets received by the destination node and the number of packets sent by the source node. PDR indicates the detection rates of our proposed IDS systems. High PDR is likely to indicate higher detection rates of IDS, as malicious nodes will be circumvented if detected correctly.
- *Routing Overhead (RO)*: RO defines the ratio of the amount of routing-related transmissions (RREQ, RREP, RERR, ACK acknowledgement, S-ACK acknowledgement, MRA acknowledgement) in bytes to the amount of data transmissions in bytes in a network. It is a reasonable metric on how efficient our proposed IDS mechanism is.

In our simulation, we record each packet transmission's detail (including both acknowledgement packet and data packet) in a trace log file. The trace file includes information like sending time, packet type, packet size and etc. Depending on the packet type, the trace format of each type of packet varies. Detailed information on trace file format can be found at

[83]. To better evaluate the performance displayed by the trace file, we created a script that parse the trace log file and calculate PDR and RO.

5.4 *Results Comparison*

In Section 4.3.3, we proposed three malicious scenarios corresponding to three different kinds of attacks to MANETs. In this thesis, we implemented our simulation based on such scenarios. This way, we can better evaluate the performance of our proposed mechanism in certain type of attacks. To refresh these three scenarios, we give a brief review of them:

- *Malicious Scenario 1*: It is simple packet dropping attack. Malicious nodes simply drop data packets by achieving receiver collision attacks whenever they receive one. It is targeted to test the performance of our scheme against the two weaknesses of Watchdog, namely, limited transmission power and receiver collisions.
- *Malicious Scenario 2*: This scenario is designed to test EAACK's performance when the attackers are smart enough to send out false misbehavior report. In this case, whenever malicious nodes receive packets, they drop it and send back a malicious report to the source node.
- *Malicious Scenario 3*: This scenario is designed to test the proposed mechanism's ability to test forged acknowledgement packet. Watchdog technique is not applicable to this case, because it's not an acknowledgement-based IDS.

For each individual scenario, we run three simulations and calculate the average values as results. In the following sections we discuss the simulation results with respect to these three malicious scenarios.

5.4.1 Malicious Scenario 1

Malicious nodes in this scenario drop all data packets that pass through it. This is achieved through receiver collision attack, where the malicious nodes overhear its next hop's transmission and only forward packets when its next hop is busy. The simulation result based on PDR is shown in Figure 23.

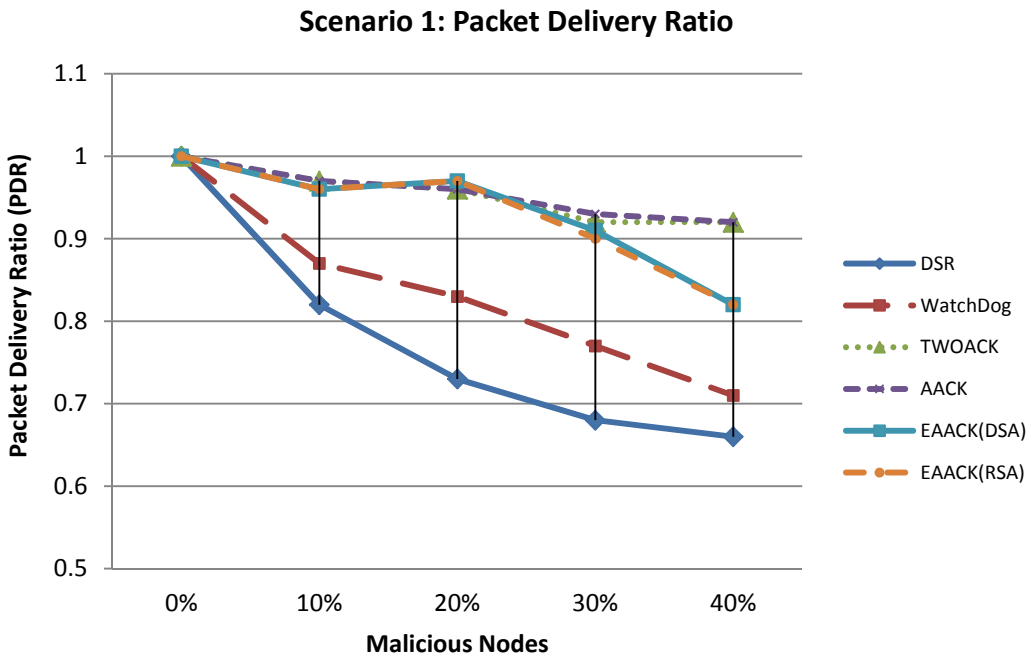

Figure 23 Simulation Scenario 1: Packet Delivery Ratio

We observe that all IDSs' performances are identical when there are no malicious nodes in the network. When the malicious nodes ratio rise up to 10%, all the two-hop acknowledgement based IDSs all outperforms the Watchdog scheme. After the malicious nodes ratio goes over 20%, the PDR by our proposed scheme surpassed Watchdog by 21%. The advantage remains when the malicious nodes reaching 30% and 40%. All two-hop acknowledgement IDS performs better than Watchdog. This is easy to understand as the

malicious nodes in this scenario are designed for the weaknesses of Watchdog. As long as the IDS does not only rely on adjacent nodes, their PDR performances are identical.

In the routing overhead test, Watchdog scheme achieves significant low result. The result is demonstrated in Figure 24. This is because it detects misbehavior by overhearing and thus eliminates the need for acknowledgement packet overhead. For the other tested IDSs, overall AACK achieves the lowest overhead due to its hybrid scheme. EAACK scheme incurs less overhead than TWOACK when malicious nodes are less than 30%. This is because EAACK uses the same hybrid scheme as AACK, this eliminates the requirement for doing acknowledgement in every packet transmission. However, when malicious nodes exceeded 30%, the extra overhead brought by digital signature eventually overcomes this advantage.

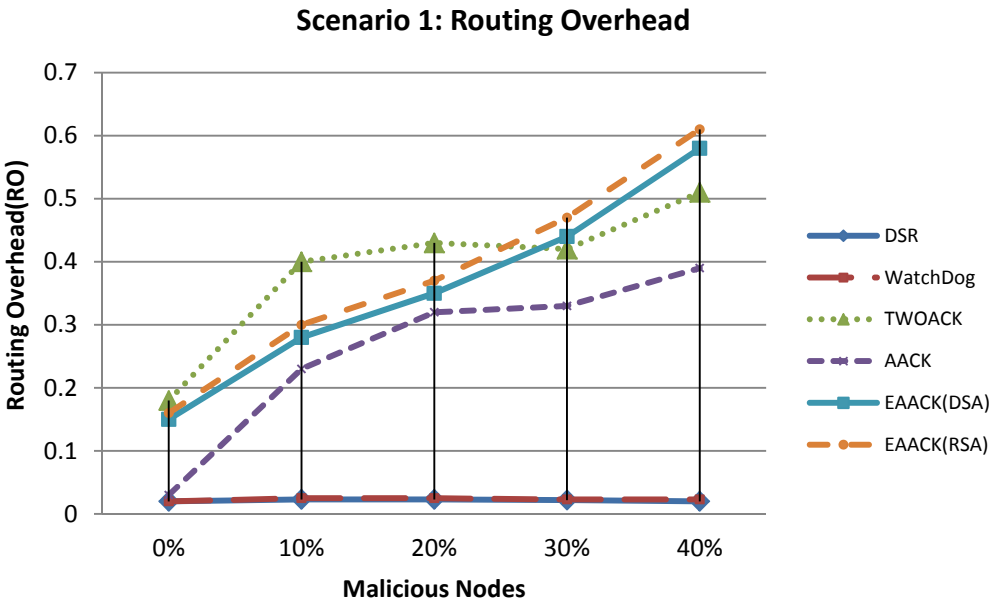

Figure 24 Simulation Scenario 1: Routing Overhead

In conclusion, our proposed scheme EAACK, despite its cryptography scheme, performs better than Watchdog in terms of PDR. Although consumes more routing overhead than

TWOACK when malicious nodes exceeds 30%, it is still a decent choice when malicious nodes are less than 30%.

5.4.2 Malicious Scenario 2

In this scenario, we set all malicious nodes to be smart enough to generate false misbehavior report and send it back to the source node. The PDR result comparison is demonstrated in Figure 25.

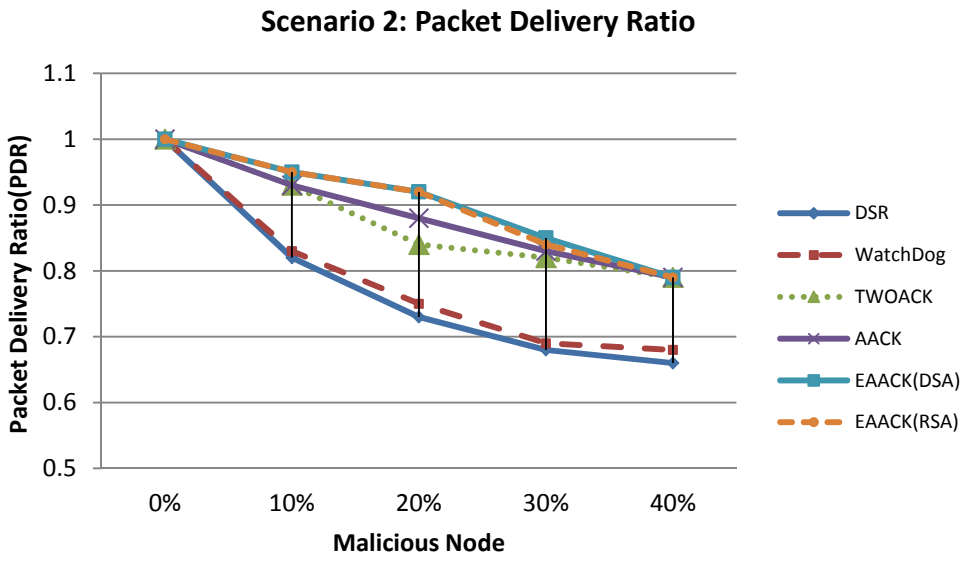

Figure 25 Simulation Scenario 2: Packet Delivery Ratio

As we can observe from Figure 25, when the malicious node is at 10%, our proposed EAACK performs 2% better than AACK and TWOACK. When malicious node reaches 20%, EAACK extended this advantage to 7% and maintain a packet delivery rate of 92%. When the malicious node rate arrives 30%, EAACK still maintains a 5% lead. We believe this is caused by the fact that neither TWOACK nor AACK has a report verification scheme; they simply chose to believe whatever misbehavior report they receive. The introduction of MRA scheme to EAACK makes it perform better when the malicious nodes are smart enough to generate false

misbehavior report. When the malicious node rate is 40%, EAACK's performance drops to the same level as TWOACK and AACK. We believe this is caused by the fact that there are not enough good nodes in the network that allow MRA to always find an alternative route to the destination node. In such case, it simply accepts the misbehavior report. Similar to scenario 1, our proposed mechanism EAACK tend to increase its routing overhead with the increase of malicious nodes, as shown in Figure 26.

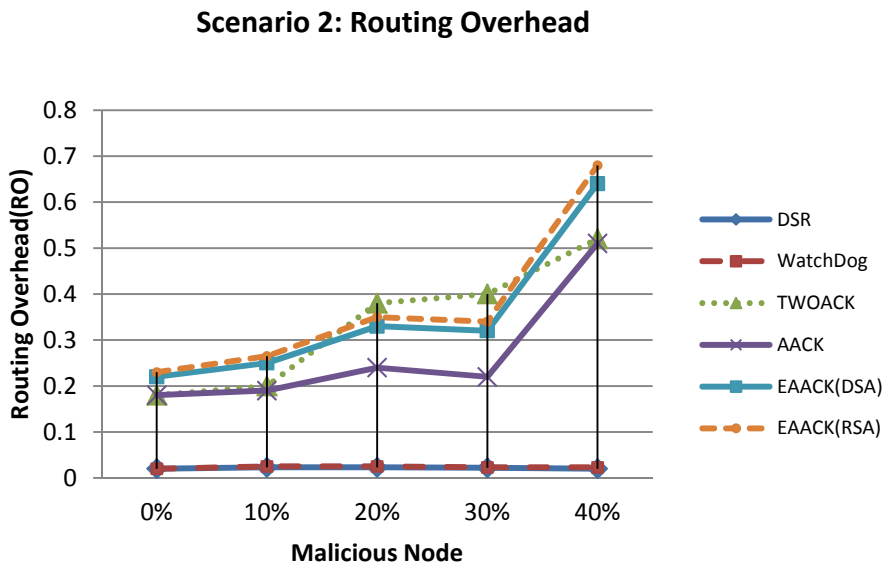

Figure 26 Simulation Scenario 2: Routing Overhead

5.4.3 Malicious Scenario 3

In this scenario, attacker nodes are modified to be able to send out forged acknowledgement packet. Attacker nodes achieve such attack to drop data packets while protecting itself from detection scheme. TWOACK, AACK and EAACK are all based on acknowledgement detection scheme, thus it is important to investigate their performance against such attack. The simulation result regarding PDR is shown in Figure 27 (next page).

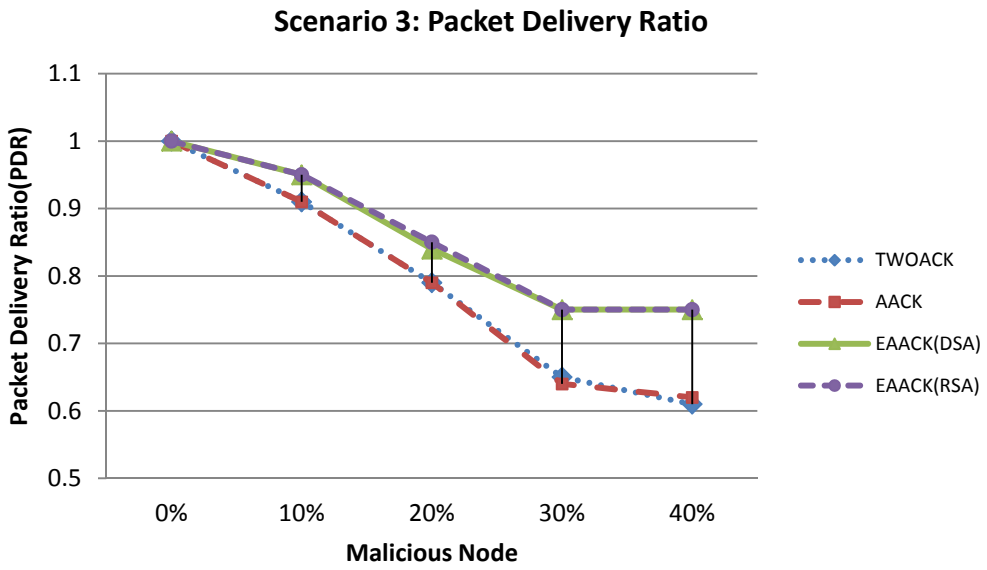

Figure 27 Simulation Scenario 3: Packet Delivery Ratio

In this scenario, AACK and TWOACK's performances are identical. As they are both vulnerable to forged acknowledgement scheme, all acknowledgements are trusted without doubt. On the other hand, our proposed EAACK scheme adopt DSA/RSA scheme to provide digital signature authentication. This approach successfully improved the PDR performance of IDS when facing forged acknowledgement attack. As shown in Figure 27, even when malicious nodes climbed up to 40%, our proposed EAACK schemes, despite the digital signature technique used, remain a PDR of 77%. On the other hand, similar to scenario 2, EAACK consumes the most routing overhead when malicious nodes exceeds 20%, as indicated in Figure 28 (next page).

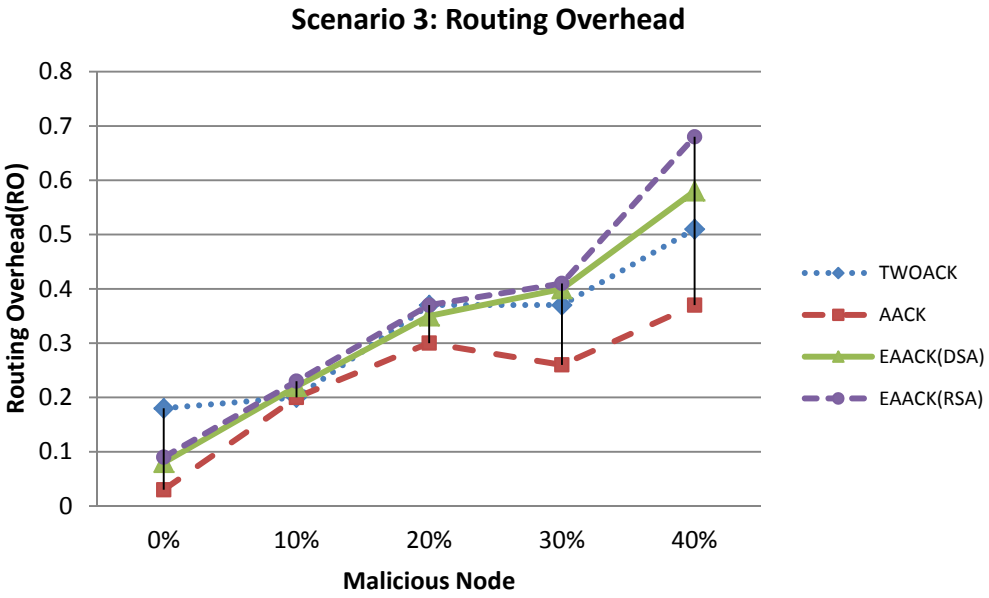

Figure 28 Simulation Scenario 3: Routing Overhead

6 CONCLUSIONS AND FUTURE WORK

In this chapter, we summarize the contributions of our research work as well as the potential future work in this research topic.

6.1 *Conclusions*

Packet dropping attack has been one of the major threats to MANETs. In order to prevent and eliminate packet dropping attack, various approaches have been proposed. In this research, we did a broad literature review on different approaches to prevent packet dropping attack. Acknowledgement based IDS is one of most important techniques against packet dropping attacks. By comparing and categorizing different acknowledgement based schemes, we analyzed the advantage and disadvantage of three popular schemes existed in the field, namely Watchdog, TWOACK and AACK. However, none of the existing approaches address the problem when the attackers are smart enough to forge acknowledgement packet or send out false acknowledgement. In this research, we proposed a hybrid scheme called EAACK to address these two problems. EAACK stands for Enhanced Adaptive ACKnowledgement mechanism. It is an enhancement on Adaptive ACKnowledgement scheme (AACK) [56]. We extended AACK to a new level where EAACK is capable of detecting forged acknowledgement packet or false misbehaviour report. The performance of our proposed schemes has been tested and compared with other schemes through simulation. The results show that EAACK has the highest packet delivery ratio among all three test scenarios. We adopted a hybrid scheme to reduce the overhead to a minimum level. Even though EAACK produces a considerable amount of network overhead in some scenarios, we believe our proposed scheme is valuable when security is of top concern.

6.2 *Future Work*

As discussed in this thesis, our proposed scheme EAACK achieves highest packet delivery ratio in most cases, but it suffers from extra amount of network overhead. Depending on the number of malicious nodes, the network overhead produced by EAACK can consume up to 70% of entire network traffic in some extreme cases. This is certainly not desirable in an environment when battery energy and computational power is strictly limited.

To increase the merits of our work, we planned to investigate the following issues in the future:

- Investigating the possibility of adopting hybrid cryptography technique to further reduce the network overhead caused by digital signature.
- Experimenting key exchange mechanism to eliminate the requirement of pre-distributed public keys. See Section 4.2.
- Due to the hardware limitation, we were not able to test our proposed scheme on real network environment. We planned to implement our proposed scheme in real life environment and better evaluate the performances in terms of computational efficiency and battery consumption.

REFERENCES

[1] M. Abdelhafez, G. Riley, R. Cole, and N. Phamdo. Modeling and Simulations of TCP MANET Worms. In the Proceedings of the 21st International Workshop on Principles of Advanced and Distributed Simulation, 2007.

[2] P. Albers, O. Camp, J. Percher, B. Jouga, L. M, and R. Puttini. Security in Ad Hoc Networks: A General Intrusion Detection Architecture Enhancing Trust Based Approaches. In the Proceedings of the 1st International Workshop on Wireless Information Systems (WIS '02), pp. 1-12, 2002.

[3] T. Anantvalee and J. Wu. A Survey on Intrusion Detection in Mobile Ad hoc Networks. Wireless Network Security. Springer US, 2007.

[4] T. Anusas-amornkul. On Detection Mechanisms and Their Performance for Packet Dropping Attack in Ad Hoc Network. PHD dissertation, University of Pittsburgh, 2008.

[5] S. Axelsson. Intrusion Detection Systems: A Taxonomy and Survey. Technical Report no. 99-15, Department of Computer Engineering, Chalmers University of Technology, Sweden, 2003.

[6] M. Barbeau and E. Kranakis. Principles of Ad-hoc Networking, Wiley, 2007.

[7] P. Brutch and C. Ko. Challenges in Intrusion Detection for Wireless Ad-hoc Networks. In the Proceedings of Symposium on Applications and the Internet Workshop, pp. 368-373, 2003.

[8] S. Carter and A. Yasinsac. Secure Position Aided Ad hoc Routing Protocol. In Proceedings of the IASTED International Conference on Communications and Computer Networks (CCN02). 2002

[9] J. Chen and J. Wu. A Survey on Cryptography Applied to Secure Mobile Ad Hoc Networks and Wireless Sensor Networks. In Wireless/Mobile Network Security, Springer, 2008.

[10] T. Clausen, C. Adjih, P. Jacquet, A. Laouiti, A. Muhlethaler and D. Raffo. Securing the OLSR Protocol. In Proceeding of IFIP Med-Hoc-Net. 2003

[11] R. Cole. Initial Studies of Worm Propagation in MANETs. In Army Science Conference (ASC), Orlando, FL, USA, 2004.

[12] R. Cole, N. Phamdo, M. A. Rajab, and A. Terzis. Requirements on Worm Mitigation Technologies in MANETs. In the Proceedings of the 19th Workshop on Principles of Advanced and Distributed Simulation (PADS '05), pp. 207–214, Washington, DC, USA, 2005.

[13] H. Debar, M. Dacier, and A.Wespi. A Revised Taxonomy for Intrusion Detection Systems. Annales des Telecommunications, vol. 55, pp. 361-378, 2000.

[14] W. Diffie and M. Hellman. New Directions in Cryptography. In IEEE Transactions on Information Theory, pp. 644-654, 1976.

[15] M. Esposito and C. Mazzariello. Evaluating Pattern Recognition Techniques in Intrusion Detection Systems. The 7th International Workshop on Pattern Recognition in Information Systems, pp. 144-153, 2005.

[16] R. Hekmat. Ad-hoc Networks: Fundamental Properties and Network Topologies, Springer, 2006.

[17] Y. Hu, D. Johnson and A. Perrig. SEAD: Secure Efficient Distance Vector Routing for Mobile Wireless Ad Hoc Networks. In the Proceedings of 4th IEEE Workshop on Mobile Computing Systems and Applications, pp. 3-13, 2002.

[18] Y. Hu, A. Perrig, and D. Johnson. ARIADNE: A Secure On-Demand Routing Protocol for Ad hoc Networks. In the Proceedings of the 8th ACM International Conference on Mobile Computing and Networking (MobiCom'02), pp. 12-23, Atlanta, GA, 2002.

[19] J. Huang, J. Buckingham and R. Han. A Level Key Infrastructure for Secure and Efficient Group Communication in Wireless Sensor Networks. In the Proceedings of 1st International Conference on Security and Privacy for Emerging Areas in Community Network, pp. 249-260, 2005.

[20] M. Ilyas. The Handbook of Ad-hoc Wireless Networks, CRC Press, 2002.

[21] M. Joa-Ng and I. Lu. A Peer-to-Peer Zone-Based Two-Level Link State Routing for Mobile Ad-hoc Networks. In the IEEE Journal on Selected Areas in Communications, vol. 17, no. 8, pp. 1415-1425, 1999.

[22] D. Johnson and D. Maltz. Dynamic Source Routing in Ad hoc Wireless Networks. Mobile Computing, Kluwer Academic Publishers, Chapter 5, pp. 153-181, 1996.

[23] J. Jubin and J. Tornow. The DARPA Packet Radio Network Protocols. In the Proceedings of the IEEE, vol. 75, issue 1, pp. 21-32, 1987.

[24] O. Kachirski and R. Guha. Intrusion Detection Using Multiple Sensors in Wireless Ad hoc Networks. In the Proceedings of the 36th Annual Hawaii International Conference on System Sciences (HICSS '03), pp. 57.1, 2003.

[25] D. Kahn. The Code Breakers. Scribner, 1967.

[26] N. Kang, E. Shakshuki and T. Sheltami. Detecting Misbehaving Nodes in MANETs. The 12th International Conference on Information Integration and Web-based Applications & Services (iiWAS2010), ACM, pp. 216-222, November, 8-10, Paris, France, 2010.

[27] N. Kang, E. Shakshuki and T. Sheltami. Detecting Forged Acknowledgements in MANETs. The 25th International Conference on Advanced Information Networking and Applications (AINA), IEEE Computer Society, Biopolis, Singapore, March 22-25, 2011.

[28] T. Karygiannis and L. Owens. Wireless Network Security-802.11. Bluetooth and Handheld Devices. National Institute of Standards and Technology, Technology Administration, U.S Department of Commerce, Special Publication, pp. 800-848, 2002.

[29] C. Kaufman, R. Perlman and M. Speciner. Network Security Private Communication in a Public World. Prentice Hall PTR, A division of Pearson Education, Inc. 2002.

[30] T. Kaya, G. Lin, G. Noubir, and A. Yilmaz. Secure Multicast Groups on Ad hoc Networks. In the Proceedings of 1st ACM workshop on Security of Ad hoc and Sensor Networks, pp.94-103, 2003.

[31] S. Kurkowski, T. Camp and M. Colagrosso. MANET Simulation Studies: The Current Stated and New Simulation Tools. Technical Report, Department of Math and Computer Science, Colorado School of Mines, MCS-05-02, 2005.

[32] L. Lamport. Password Authentication with Insecure Communication. In the Communications of the ACM, vol. 24, issue 11, pp. 770-772, 1981.

[33] Y. Li and J. Wei. Guidelines on Selecting Intrusion Detection Methods in MANET. In the Proceedings of Information System s Educators Conference, 2004.

[34] K. Liu, J. Deng, P. K. Varshney, and K. Balakrishnan. An Acknowledgment-Based Approach for the Detection of Routing Misbehaviour in MANETs. In the IEEE Transactions on Mobile Computing, vol. 6, pp. 536-550, 2007.

[35] F. Lunt and R. Jagannathan. IDES: The Enhanced Prototype C a Realtime Intrusion-Detection Expert System. Technical Report SRI-CSL-88-12, SRI International, Menlo Park,CA, 1988.

[36] S. Marti, T.J. Giuli, K. Lai, and M. Baker. Mitigating Routing Misbehaviour in Mobile Ad hoc Networks. In the Proceedings of the 6th Annual International Conference on Mobile Computing and Networking (MobiCom '00, ACM), pp. 255-265, Boston, Massachusetts, US, 2000.

[37] A. Menezes, P. van Oorschot, and S. Vanstone. Handbook of Applied Cryptography, CRC Press, 1996.

[38] B. Mukherjee, L. Heberlein and K. Levitt. Network Intrusion Detection. In the Proceedings of IEEE International Conference on Network, vol.8, no.3, pp.26-41, 1994.

[39] Murthy and J.J. Garcia-Luna-Aceves, An Efficient Routing Protocol for Wireless Networks. In ACM Mobile Networks and Applications Journal, vol. 1, no. 2, pp. 183-197, 1996.

[40] N. Nasser and Y. Chen. Enhanced Intrusion Detection Systems for Discovering Malicious Nodes in Mobile Ad hoc NETwork. In the Proceedings of IEEE International Conference on Communication (ICC '07), Glasgow, Scotland, 2007.

[41] R. Nicholas and P. Lekkas. Wireless Security-Models, Threats, and Solutions, McGraw-Hill, Chapter 7, 2002.

[42] J. Parker, J. Undercoffer, J. Pinkston, and A. Joshi. Secure Routing for Mobile Ad hoc Networks. In the Proceedings of the SCS communication Networks and Distributed Systems Modeling and Simulation Conference (CNDS '02), Texas, SA, US, 2002.

[43] J. Parker, J. Undercoffer, J. Pinkston and A. Joshi. On Intrusion Detection and Response for Mobile Ad hoc Networks. In the Proceedings of IEEE International Conference on Performance, Computing, and Communications, pp. 747-752, 2004.

[44] A. Patcha and A. Mishra. Collaborative Security Architecture for Black Hole Attack Prevention in Mobile Ad hoc Networks. In the Proceedings of Radio and Wireless Conference (RAWCON '03), pp. 75-78, 2003.

[45] A. Patwardhan, J. Parker, A. Joshi, M. Iorga, and T. Karygiannis. Secure Routing and Intrusion Detection in Ad hoc Networks. In the Proceedings of 3rd International Conference on Pervasive Computing and Communications, pp. 191-199, 2005.

[46] C. E. Perkins. Ad-hoc Networking. Addison Wesley Professional. 2000.

[47] C. E. Perkins and P. Bhagwat. Highly Dynamic Destination-Sequenced Distance-Vector Routing (DSDV) for Mobile Computers. In ACM Computer Communication Review, vol. 24, pp. 234-244, London, UK, 1994.

[48] C. E. Perkins and E. M. Royer. Ad-hoc On-Demand Distance Vector (AODV) Routing. In the Proceedings of the IEEE Workshop on Mobile Computing Systems and Applications (WMCSA'99), pp. 90-100, New Orleans, LA, 1999.

[49] P.A. Porras and R. Kemmerer. Penetration State Transition Analysis C a Rule-Based Intrusion Detection Approach. The 8th Annual Computer Security Application Conference, pp. 220-229, 1992.

[50] R. Rivest, A. Shamir, and L. Adleman. A Method for Obtaining Digital Signatures and Public-key Cryptosystems. In the Communications of ACM, vol. 21, pp. 120-126, 1978.

[51] E. Royer and C. Toh. A Review of Current Routing Protocols for Ad hoc Mobile Wireless Networks. In IEEE Personal Communications Magazine, vol. 6, no. 2, pp. 46-55, 2002.

[52] K. Sanzgiri, B. Dahill, B. Levine, C. Shields and E. Belding-Royer. A Secure Routing Protocol for Ad Hoc Networks. In the Proceedings of 10th IEEE International Conference on Network Protocols, pp. 78-87, 2002.

[53] S. K. Sarkar, T. Basavaraju, and C. Puttamadappa. Ad Hoc Mobile Wireless Networks, Auerbach Publications, 2008.

[54] E. Shakshuki, N. Kang, X. Xing, and T. Sheltami. Tracking Anonymous Sinks in Wireless Sensor Networks. In the IEEE 23rd International Conference on Advanced Information Networking and Applications, pp. 510-516, Bradford, UK, 2009.

[55] A. Shamir. How to Share a Secret. In the Communications of the ACM, vol. 22, issue 11, 1979.

[56] T. Sheltami, A. Al-Roubaiey, E. Shakshuki, and A. Mahmoud. Video Transmission Enhancement in Presence of Misbehaving Nodes in MANETs. International Journal of Multimedia Systems, Springer, vol. 15, issue 5, pp. 273-282, 2009.

[57] B. Sun. Intrusion Detection in Mobile Ad hoc Networks. Doctoral Dissertation. Texas A&M University, 2004.

[58] B. Sun, K. Wu, and U. Pooch. Alert Aggregation in Mobile Ad Hoc Networks. In the Proceedings of the ACM Workshop on Wireless Security (WiSe'03) in Conjunction with the 9th Annual International Conference on Mobile Computing and Networking (MobiCom'03), pp. 69-78, 2003.

[59] D. Sterne, P. Balasubramanyam, D. Carman, B. Wilson, R. Talpade, C. Ko, R. Balupari, C.-Y. Tseng, T. Bowen, K. Levitt, and J. Rowe. A General Cooperative Intrusion Detection Architecture for MANETs. In the Proceedings of the 3rd IEEE International Workshop on Information Assurance (IWIA '05), pp. 57-70, 2005.

[60] B. Wu, J. Chen, J. Wu, and M. Cardei. A Survey on Attacks and Countermeasures in Mobile Ad hoc Networks. In Wireless/Mobile Network Security, Springer, 2008.

[61] Y. Xiao, X. Shen, and D. Du (Eds.). A Survey on Intrusion Detection in Mobile Ad-hoc Networks. In Wireless/Mobile Network Security, pp. 170-196, 2006.

[62] P. Yi., Y. Hou, Y. Zhong, S. Zhang, and Z. Dai. Flooding Attack and Defence in Ad hoc Networks. In the Journal of Systems Engineering and Electronics, vol 17, issue 2, pp. 410-416, 2006.

[63] M. Zapata and N. Asokan. Securing Ad hoc Routing Protocols. In the ACM Workshop on Wireless Security (WiSe 2002), pp. 1-10, 2002.

[64] M. Zapata. Secure Ad hoc On-Demand Distance Vector Routing. In the ACM SIGMOBILE Mobile Computing and Communications Review, vol. 6, issue 3, 2002.

[65] Y. Zhang, W. Liu, W. Lou and Y. Fang. Securing Mobile Ad hoc Networks with Certificateless Public Keys. In the proceedings of IEEE Transactions on Dependable and Secure Computing, vol. 3, no. 4, pp. 386-399, 2006.

[66] Y. Zhang, W. Lee, and Y. Huang. Intrusion Detection Techniques for Mobile Wireless Networks. In the ACM/Kluwer Wireless Networks Journal (ACM WINET), vol. 9, no. 5, 2003.

[67] Y. Zhang, W. Liu, W. Lou, Y. Fang and Y. Kwon. AC-PKI: Anonymous and Certificateless Public-Key Infrastructure for Mobile Ad Hoc Networks. In the

Proceedings of IEEE International Conferences on Communication, pp. 3515-3519, 2005.

[68] S. Zhong, J. Chen, and Y.R. Yang. Sprite: A Simple, Cheat-proof, Credit-based System for Mobile Ad Hoc Networks. In the Proceedings of InfoCOM '03, pp. 1987-1997, 2003.

[69] L. Zhou and Z. Haas. Securing Ad-hoc Networks. In the IEEE Network Magazine, vol. 13, no. 6, pp. 24-30, 1999.

[70] S. Zhu, S. Xu, S. Setia, and S. Jajodia. LHAP: A Lightweight Hop-by-Hop Authentication Protocol for Ad-Hoc Networks. In the Journal of Ad hoc Networks, vol. 4, no. 5, pp. 567-585, 2006.

[71] S. Zhu, S. Setia, and S. Jajodia. LEAP: Efficient Security Mechanisms for Large-Scale Distributed Sensor Networks. In the Proceedings of 10th ACM Conference on Computer and Communication Security, pp. 62-72, 2003.

[72] Digital Signature Standard (DSS). Federal Information Processing Standards Publication, National Institute of Standards and Technology, Gaithersburg, MD, 2009.

[73] Cisco.com. Routing Basics. Internetworking Technology Handbook. http://docwiki.cisco.com/wiki/Internetworking_Technology_Handbook Accessed: March 14, 2010.

[74] Cisco.com. Open Systems Interconnection (OSI) Protocols. Internetworking Technology Handbook. http://docwiki.cisco.com/wiki/Internetworking_Technology_Handbook Accessed: March 7, 2010.

[75] The Network Simulator – ns-2. http://www.isi.edu/nsnam/ns/ Accessed: January 11, 2010.

[76] REAL 5.0 Overview. http://www.cs.cornell.edu/skeshav/real/overview.html Accessed: May 20, 2010.

[77] The Network Simulator ns-2: Documentation. http://www.isi.edu/nsnam/ns/ns-documentation.html Accessed: February 10, 2010.

[78] Marc Greis' Tutorial for the UCB/LBNL/VINT Network Simulator. http://www.isi.edu/nsnam/ns/tutorial/index.html Accessed: February 18, 2010.

[79] E. Altman and T. Jimenez. NS Simulator for Beginners. Lecture Notes, University de Los Andes, Merida, Venezuela and ESSI, Sophia-Antipolis, France.

[80] Botan, a Friendly C++ Crypto Library. http://botan.randombit.net/ Accessed: October 2, 2010.

[81] BOTAN API Reference. http://files.randombit.net/botan/api.pdf Accessed: October 10, 2010.

[82] TIK WSN Research Group. The Sensor Network Museum – Tmote Sky. http://www.snm.ethz.ch/Projects/TmoteSky Accessed: November 11, 2010.

[83] NS-2 Trace Formats. http://nsnam.isi.edu/nsnam/index.php/NS-2_Trace_Formats Accessed: April 4, 2009.