People's Democratic Republic of Algeria

Ministry of Higher Education and Scientific Research

UNIVERSITY MOHAMED KHIDER OF BISKRA

Faculty of Exact Sciences, Science of Nature and Life

Computer Science Department

"Order number............"

THESIS

Submitted in fulfilment of the requirements for the degree Doctor of Science

Option: Artificial Intelligence and Distributed Systems

Presented by: **Maqbol Ahmed** Directed by: **Pr. Kazar Okba**

TITLE THESIS

Sécurité Basée Agent Mobile Dans Les Réseaux Sans Fils

Defended before a jury composed of:

President :	Pr. Benmohamed Mohamed	Professor	University of Constantine 2
Rapporter:	Pr. Kazar Okba	Professor	University of Biskra
Examiners :	Pr. Zerhouni Said Noureddine	Professor	University of Besançon – France
	Dr. Benharzalla Saber	Lecturer	University of Batna 2
	Dr. Hamadi Bennoui	Lecturer	University of Biskra
	Dr. Rezeg Khaled	Lecturer	University of Biskra

BISKRA 07/05/2017

Acknowledgements

First, I want to thank almighty ALLAH for giving me the will, patience and health to develop this work although the circumstances surrounding us.

Then I would like to express my gratitude and thanks to Pr. Kazar Okba–my academic advisor – for identification and refinement of my research ideas. He simply expressed his willingness for the supervision, this was rather unusual response that is not expected from a professor of his caliber. As a result, unfortunately, he had to work harder to improve my research capabilities and paper writing skills. However, his confidence in my intellectual capabilities always encouraged me. His dedication to the profession is immaculate and inspirational for all his PhD students. I wish and pray for even a brighter future for his career.

Special thank you to Prof. Benmohamed Mohamed, a professor at the University of Constantine 2, who has done me the honor of chairing this jury. I would also like to thank Pr. Zerhouni Said Noureddine, a professor at the University of Besançon – France, Dr. Rezeg Khaled, Lecturer at the University of Biskra, Dr. Hamadi Bennoui, Lecturer at the University of Biskra, and Dr. Benharzallah Saber, Lecturer at the University of Batna 2 for having accepted to judge this work.

I would also like to thank all the members of the Department of Computer Science. In particular, I would like to thank Pr. Mohamed Chaouki Babahenini, Head of Department of Computer Science to facilitate all the administrative procedures throughout the years of studies.

I take this opportunity to extend my sincere thanks to Algeria's government and people for the hospitality and warm reception throughout the years of studies, where we did not feel that we are strangers, but we live as if we were among our people and our families.

I am also grateful to all my family, and I especially wish to thank my mother and brothers Omer and Djamal for their ongoing encouragement and support throughout my studies. I hope that I can use the skills that I have obtained here to make you proud in the future.

Finally, I keep a special place for all my friends. I would like to express my deep gratitude to them because they have constantly helped me and created appropriate conditions for me to complete this thesis. And everyone else I have met in the duration of the course. I also thank anyone who has helped me directly or indirectly to complete this project.

الإهداء

إلى والدتي حفظها الله..

إلى روح والدي رحمه الله...

إلى إخواني عمر وجمال وإلى جميع أفراد أسرتي..

إلى بلدي يمن الإيمان والحكمة..

إلى بلدي الثاني بلد المليون ونصف المليون شهيد جزائر العزة والكرامة..

أهدي هذا العمل المتواضع...

ملخص

أصبحت الشبكات اللاسلكية وتحديدا شبكات الـ ad hoc أكثر شعبية اليوم بسبب مُميزاتها مثل تنقل المستخدم الذي يستطيع الاتصال في أي مكان وفي أي وقت، ومع أي شخص، والخ، وتتميز شبكات المحمول ad hoc بطوبولوجيا متحركة، وسيط الاتصالات لاسلكي، التعاون الموزع، انعدام البنيات الأساسية ونقاط الإدارة، موارد محدودة، اتصالات متقطعة، غياب سلطة التصديق وقابلية الإصابة الفيزيائية للعقدة، ونتيجة لهذه الخصوصيات، فإن الأمن يواجه تحديات في تحقيق الأهداف الأمنية مثل السرية، التحقق من الهوية، الكمال، التوفر، مراقبة الدخول وعدم الإنكار.

في هذه الأطروحة، نقدم **" النموذج الأمني القائم على الوكيل المتنقل للشبكات المتنقلة ad hoc"**. هنا نطبق مفهوم رئيس المجموعة القائم على تقسيم الشبكة إلى مجموعات، حيث أن انتخاب رئيس كتلة المجموعة مبني على الثقة وكفاءة موارد العقدة، ونقترح دالة تتكون من خمس عوامل لتقييم موارد العقدة وكذلك صيغة رياضية لتقدير ثقة العقدة.

في أول مساهمة، استخدمنا ثلاثة وكلاء: وكيل العقدة (Node Agent) حيث يدير موارد العقدة اعتماداً على القدرة والشروط المقترحة. وكيل المراقبة (Monitor Agent) الذي يعتبر ممثلاً للكتلة، وهو ينشئ وكيل/ وكلاء التفتيش (Inspector Agent) ويُرسله إلى كافة عقد المجموعة بغرض التفتيش الفجائي، بينما في المساهمة الثانية، أضفنا وكيل النقل (Transporter Agent)، ووكيل السفير(Ambassador Agent) الذي يُنشيء بواسطة وكيل المراقبة ويُرسل إلى كل عقد المجموعة، وهو يعتبر مثل نظام كشف أو منع التسلل على مستوى العقدة.

واستناداً إلى النتائج التي تم الحصول عليها، يمكننا أن نلخص أن تنفيذ نموذجنا يلبي تقريباً الأهداف الرئيسية للأمن. ومع ذلك فإن الحل الذي اقترحناه وحققناه ليس الحل السحري لمشكلة الأمن، حيث لم نستطع تغطية كل النقاط المتعلقة بالموضوع ولكن نأمل أن نكون قد فتحنا الأبواب للدارسات المستقبلية من أجل تطوير وتحسين هذا الحل.

الكلمات المُفتاحية: نظام متعدد الوكلاء، الوكيل المتحرك، الأمن، الشبكات اللاسلكية، شبكات المحمول ad hoc.

Abstract

Wireless Networks, especially ad hoc networks are becoming more and more popular today due to its advantages such as user mobility who enables the communication anywhere, anytime, with anyone, etc.

Mobile Ad Hoc Networks are characterize by dynamic nature of network topology, radio communication medium, distributed cooperation, lack of a pre-existing infrastructure or management point, resource constrained, the sporadic nature of connectivity, an absence of a certification authority, and physical vulnerability of node. As a result of these specificities, the security of mobile ad hoc network pose both challenges and opportunities in achieving security goals, such as confidentiality, authentication, integrity, availability, access control, and non-repudiation.

In this thesis, we introduce ***"Security Model based on Mobile Agent for Mobile Ad Hoc Networks***". We apply the concept of dominating set based clustering for partitioning network into clusters. The cluster head elected based on both the trust and resources ability of the node. We propose an optimization function of five parameters to evaluate node resources and the formula to estimate the trust of the node.

In the first contribution, we use three agents: Node Agent (NA) manages node resources depending on the capacity and the proposed conditions. Monitor Agent (MoA) who considered a representative of the cluster, it creates Inspector Agent (IA) and sends it to all nodes of the cluster for surprise inspections. While in the second contribution, we added a Transporter Agent (TA), an ambassador agent (AmA) that creates by Monitor Agent and sending to all nodes in the cluster. The Ambassador Agent is like local IDS and IPS (Intrusion Detection System and Intrusion Prevention System) at the node.

Based on the obtained results, we can summarize that the implementation of our model satisfy almost the main objectives of the security. However, the solution we propose and realize is certainly not the miracle solution to the security problem in ad hoc networks, we cannot cover all the points of our subject but we hope to have succeeded in opening ports to studies in future to develop or make improvements to this solution.

Keywords**:** *Multi-Agent System, Mobile Agent, Security, Wireless Network, Mobile Ad-hoc Networks.*

Résumé

Les réseaux sans fil en particulier les réseaux ad hoc sont de plus en plus populaires aujourd'hui en raison de ses avantages tels que la mobilité des utilisateurs qui permet la communication n'importe où, n'importe quand, avec n'importe qui, etc.

Les réseaux ad hoc mobiles sont caractérisés par la nature dynamique de la topologie du réseau, du support de communication radio, de la coopération distribuée, de l'absence d'une infrastructure préexistante ou d'un point de gestion, des ressources limitées, de la connectivité sporadique, de l'absence d'autorité de certification et de la vulnérabilité physique du nœud. En raison de ces spécificités, la sécurité du réseau ad hoc mobile pose à la fois des défis et des opportunités pour atteindre les objectifs de sécurité, tels que la confidentialité, l'authentification, l'intégrité, la disponibilité, le contrôle d'accès et la non-répudiation.

Dans cette thèse, nous présentons le «***Modèle de Sécurité basé sur l'Agent Mobile pour les Réseaux Mobiles Ad Hoc***». Nous appliquons le concept de l'ensemble dominant basé sur le regroupement pour le partitionnement réseau en clusters. Le chef du cluster est élu en fonction de la confiance et des ressources du nœud. Nous proposons une fonction d'optimisation de cinq paramètres pour évaluer les ressources des nœuds et la formule pour estimer la confiance du nœud.

Dans la première contribution, nous utilisons trois agents : Agent Nœud (NA) qui gère les ressources des nœuds en fonction de la capacité et des conditions proposées. Agent Monitor (MoA) qui a considéré un représentant du cluster, il crée Agent Inspecteur (IA) et l'envoie à tous les nœuds du cluster pour les inspections surprise. Dans la deuxième contribution, nous avons ajouté un Agent Transporteur (TA), un agent ambassadeur (AmA) qui a créé par l'agent Monitor et l'envoyé à tous les nœuds du cluster. L'agent Ambassadeur est considère comme l'IDS et l'IPS local (Système de Détection d'Intrusion et Système de Prévention d'Intrusion) au nœud.

À la base des résultats obtenus, on peut résumer que la mise en œuvre de notre modèle satisfait presque les principaux objectifs de la sécurité. Toutefois, la solution que nous proposons et réaliser n'est sûrement pas la solution miracle au problème de sécurité dans réseaux ad hoc, nous ne pouvons pas couvrir tous les points de notre sujet mais nous espérons avoir réussi à ouvrir des ports à des études en futur pour développer ou faire des amélioration à cette solution.

Mots-clés : Système Multi-Agents, Agent Mobile, Sécurité, Réseau sans fil, Réseau Mobile Ad-hoc.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ACRONYMS

1G:	First-Generation Mobile Systems
2G:	Second-Generation Mobile Systems
3G:	Third-Generation Mobile Systems
4G:	Fourth-Generation Mobile Systems
AAA:	Authentication, Authorization, Accounting
ACA:	Agent Actuator
ACO:	Ant Colony Optimization
AmA:	Ambassador Agent
AmAB:	Ambassador Agent B
AmAD:	Ambassador Agent D
AMPS:	Advanced Mobile Phones Service
AnA:	Analyser Agent
AODV:	Ad hoc On-Demand Distance Vector Routing
AP:	Access Point
ASDK:	Aglets Software Development Kit
ATDSR:	Agent-Based Trusted Dynamic Source Routing
BS :	Base Station
BSS:	Basic Service Set
CBRP:	Cluster Based Routing Protocol
CDMA :	Code Division Multiple Access
CDPD:	Cellular Digital Packet Data
CEO:	Chief Executive Officer
CH:	Cluster Head
CL:	CPU Load
C_{ni}:	Capacity of the node
CRC:	Cyclic Redundancy Check
CT	Confidence Table
DM:	Detection Module
DMV:	Detection of Malicious Vehicle
DMZ:	Demilitarized Zone
DN:	Degree Node
DoS:	Denial-of-Service
DSR:	Dynamic Source Routing
EAP:	Extensible Authentication Protocol
EDGE:	Enhanced Data for Global Evolution
EL:	Energy Level
ETSI:	European Telecommunications Standards Institute
FDMA:	Frequency Division Multiple Access
GPRS:	General Packet Radio System
GPS:	Global Positioning System
GSM:	Global System for Mobile Communications
HD:	High Definition
HMAC:	Keyed-Hash Message Authentication Code
IA:	Inspector Agent
IDE:	Integrated Development Environment
IDS:	Intrusion Detection System
IPS	Intrusion Prevention System
IrDA:	Infrared Data Association
IV:	Initialization Vector
JDK:	Java Development Kit
JVM:	Java Virtual Machine
LMDS:	Local Multipoint Distribution Service
LPI:	Low Probability of Intercept
LTE:	Long Term Evolution
MA:	Mobile Agent
ManA:	Manager Agent
MANET:	Mobile Ad Hoc Network
MAS:	Multi-Agent System
MIC:	Message Integrity Check
MN:	Mesh Node
MoA:	Monitor Agent
MPR:	Multipoint Relay
MS:	Mobile Station
NA:	Node Agent
NAA:	Node Agent A
NAB:	Node Agent B
NAD:	Node Agent D
NFC:	Near Field Communications
NMT:	Nordic Mobile Telephones
NTT:	Nippon Telephone and Telegraph
OA:	Ontology Agent
OBU:	On Board Unit
OLSR:	Optimized Link State Routing Protocol
PCC:	Proof-Carrying Code Institute
PDA:	Personal Digital Assistant
PKI:	Public Key Infrastructure
PKM:	Privacy Key Management

PM:	Prevention Module
PRAC:	Partial Result Authentication Codes
PRE:	Partial Result Encapsulation
PSK:	Pre-Shared Key
PSTN:	Public Switched Telephone Network
RADIUS:	Remote Authentication Dial-In User Service
RC4:	Rivest Cipher 4
RFID:	Radio Frequency Identification
RM:	Registration Module
RoA:	Routing Agent
POS:	Personal Operating Space
SA:	Service Agreement
SHA-1:	Secure Hash Algorithm 1
SSID:	Service Set Identifier
TA:	Transporter Agent
TAB	Transporter Agent B
TACS:	Total Access Communication Systems
TC:	Topology Control
TCL:	Tool Command Language
TDMA:	Time-Division Multiple Access
TKIP:	Temporal Key Integrity Protocol
TL:	Trust Level
TMTO:	Time Memory Trade Off
UMTS:	Universal Mobile Telecommunications Systems
UWB:	Ultra-Wideband
VANET:	Vehicular Ad Hoc Network
VoIP:	Voice over Internet Protocol
VPN:	Virtual Private Network
WCDM:	Wideband Code Division Multiple Access
WEP:	Wired Equivalent Privacy
WLAN:	Wireless Local Area Network
WMAN:	Wireless Metropolitan Area Network
WMN:	Wireless Mesh Network
WPA:	Wi-Fi Protected Access
WPAN:	Wireless Personal Area Network
WSN:	Wireless Sensor Network
WWAN:	Wireless Wide Area Network

Chapter 1

Introduction

Introduction inhibit pleasure,
kill the joy of anticipation,
frustrate curiosity.

Harper Lee

This chapter gives an overview over the thesis. It especially constraints the work into a specific application domain, and gives the limits and boundaries of the elaborations. The chapter closes with a structure of the work.

1.1 Overview

Wireless networks are becoming more and more popular today. This popularity is due to its advantages compared to wireline networks, as, user mobility who enables (communication anywhere, anytime, with anyone), fast and simple installation, flexibility, scalability and relatively low price.

Wireless networking has been of various sizes, such as Wireless Personal Area Networks (WPANs), Wireless Local Area Networks (WLANs), Wireless Metropolitan Area Networks (WMANs), and Wireless Wide Area Networks (WWANs). These wireless networks can be of different formations, such as cellular networks, ad hoc networks, mesh networks, and can be domain specific networks, such as vehicular communication networks and sensor networks. As a result, Wireless networks technologies differ from one type to another in many of the characteristics.

Unfortunately, the security aspects of these technologies were lax to begin with and have improved only marginally. They are still rife with security design flaws and weak built-in security mechanisms. In fact, security is an essential service for wireless network communications. The success of these technologies will depend on people's confidence in its security. However, the specificities of wireless network pose both challenges and opportunities in achieving security goals, such as confidentiality, authentication, integrity, availability, access control, and non-repudiation.

The mobile agent has received considerable attention in recent years for its wide applications in various areas of computing technology. This has led to deal more efficiently and elegantly with the dynamic, heterogeneous, and open environment, which is today's wireless network.

A mobile agent is a software program with mobility that can be sent out from a node into a wireless network and roam among the nodes in the wireless network. It can be executed on those nodes to finish its task on behalf of its owner. The direction of a mobile agent can be multi-way, from any node to another within that network. The migration of a mobile agent can occur multiple times before it comes back to its home node with the computation results. In addition, not only the application logic of a mobile agent is transferred between nodes, but also the application state can be transferred from one node to another. The transferring of a mobile agent state facilitates it in working autonomously to travel between one or more remote nodes [1].

A lot of research has been dedicated to address the security problems in a mobile agent system. This research differs in its aim, emphasis, base, and technique. Some work are towards building the foundations for the security of a mobile agent system; some propose security mechanisms following different approaches; some focus on introducing security mechanisms into the architectures of mobile code systems; and others implement real applications with security features. However, there has been limited research dedicated to provide an intuitive formal framework for a secure mobile agent system, including formal modeling of mobility, communication, and execution [2].

1.2 Problem Definition

As we know, Wireless Networks are becoming more and more popular at the office, home, hotel, coffee shops, airports, train stations, and many other places. However, the bad news that wireless networks are a major target for attackers. The security concerns in wireless networking remains a serious impediment to widespread adoption. The underlying radio communication medium for wireless networks is a big vulnerability. As a result, the channel can be eavesdropped by placing an antenna at an appropriate location, an attacker can overhear the information that the victim transmits or receives. In addition, the data can be altered where an attacker can try to modify the content of the message exchanged between (wireless) parties. The channel can be jammed, notably in order to perpetrate a DoS attack by transmitting at the same time the victim transmits or receives data, an attacker can make it impossible for the victim to communicate. Moreover, the radio channel can be overused where the radio spectrum being a shared resource, there is a risk that a wireless operator or a user makes an excessive use of it.

On the other hand, security in wireless ad hoc network is more difficult to achieve due to lack of a pre-existing infrastructure or management point, unreliable communication, intermittent connection, node mobility, and dynamic topology, the limited physical protection of each of the nodes, the sporadic nature of connectivity, and the absence of a certification authority. In addition, wireless (mobile) devices usually have limited bandwidth, storage space, and processing capacities. It is harder to reinforce security in wireless networks than in wired networks.

Furthermore, wireless sensor networks (WSNs) have many unique features that differ from wireless ad hoc networks. Firstly, a large-scale sensor network and typical sensor nodes are limited in their energy, computation, and communication capabilities. Secondly, wireless sensor nodes may be dispersed over a wide geographical area, presenting the additional risk of physical attack. Thirdly, sensor networks interact with people and environments, posing new security risks.

In the light of the above mentioned, wireless ad hoc networks are target for all the threats that occur in wireless networks, i.e., masqueraded identities, authorization violations, eavesdropping, data loss, modified and falsified data units, repudiation of communication processes and sabotage [3].

The attacks can be classified according to different criteria, such as the domain of the attackers, or the techniques used in attacks. Normally, network security attacks are divided into passive and active attacks. The risks associated with wireless networks can be the result of one or more of these attacks.

In particular, attacks in ad hoc networks can cause congestion, propagate incorrect routing information, prevent services from working properly, or shut them down completely. Nodes that perform active attacks with the aim of damaging other nodes by causing network outage are considered malicious, also referred to as compromised, whereas nodes that perform passive attacks with the aim of saving battery life for their own communications are considered selfish. A selfish node affects the normal operation of the network by not participating in the routing protocols or by not forwarding packets as in the so-called black hole attack [4].

Another classification [5] of attacks (Attacks against secure routing) as, internal attacks and external attacks. Internal attacks are more severe attacks, since malicious nodes have already been authorized and are thus protected with the security mechanisms the network and its services offer. These kind of malicious parties are called compromised nodes. They may operate as a group using standard security protection to protect their attacks, compromising the security of the whole ad hoc network [3].

Internal attacks are especially relevant in ad hoc networks, which are operating in hostile environments like enemy battlefields. It is this threat of internal attacks that makes ad hoc security an extremely challenging field. Realize that attacks launched from internal attackers are much harder to detect for two important reasons. First, if a node determines that the routing information that it has received is invalid, it is difficult for it to conclude whether the information that it has received became invalid because of changes in the network topology or because the sending node was compromised. Second, a compromised node would still arguably be able to generate valid signatures using its private keys, thus making it even harder to use cryptography to detect that it has been compromised [6].

External attacks on routing can be divided into two categories: passive and active [5]. Passive attacks involve unauthorized listening to the routing packets. The attack might be an attempt to gain routing information from which the attacker could extrapolate data about the positions of each node in relation to the others. In a passive attack, the attacker does not disrupt the operation of a routing protocol but only at tempts to discover valuable information by listening to the routed traffic.

Active attacks on the network from outside sources are meant to degrade or prevent message flow between the nodes. Active external attacks on the ad hoc routing protocol can collectively be described as denial-of-service attacks, causing a degradation or complete halt in communication between nodes. One type of attack involves insertion of extraneous packets into the network in order to cause congestion. A more subtle method of attack involves intercepting a routing packet, modifying its contents, and sending it back into the network. Alternatively, the attacker can choose not to modify the packet‘s contents but rather to replay it back to the network at different times, introducing outdated routing information to the nodes. The goal of this form of attack is to confuse the routing nodes with conflicting information, delaying packets or preventing them from reaching their destination.

The mobile agent has received considerable attention in recent years for its wide applications in various areas of computing technology. This has led to deal more efficiently and elegantly with the dynamic, heterogeneous, and open environment like the security of ad hoc networks. Nevertheless, the mobile agent technology has many security threats because the mobile code generated by a party will be transferred and run in an environment controlled by the other party. Several security issues arise in various areas to mobile agent computing, including authentication, authorization (or access control), intrusion detection, etc. Malicious agents, platforms and third parties could attack a mobile agent system. Also, since mobile agents have unique characteristics such as mobility of a mobile agent, security issues become more complicated in mobile agent systems.

It is clear that the problem of security is very complicated. Presently techniques are available to address some of these problems, such as cryptography, virtual private networks, 802.1x, Firewalls, etc. Furthermore, the recent advances in encryption, public key exchange, digital signatures and the development of related standards have set a foundation for the flourishing usage of mobile and wireless technologies in many areas such as e-commerce.

However, security in a network goes way beyond encryption of data. It must include the security of computer systems and networks, at all levels, top to bottom. It is imperative to design network protocols with security considered at all layers as well as to arm the networks systems and elements with well designed, comprehensive, and integrated attack defeating policies and devices. A foolproof prevention of attacks is challenging because at best the defensive system and application software may also contain unknown weaknesses and bugs. Thus, early warning systems (i.e. intrusion detection systems) as components of a comprehensive security system are required in order to prime the execution of countermeasures.

This is the background of our contribution to implement new concepts to enhance the security of the wireless network by improving the security of the mobile agent. A complete security solution should include three components of prevention, detection, and reaction. It must provide security properties of authentication, confidentiality, non-repudiation, integrity, and availability. It should be adaptive in order to trade-off service performance and security performance under resource limitation [7].

1.3 Limitations of Scope

There are numerous security attacks that can be mounted on any nodes in a wireless networks. Although to treat topic of security in wireless networks, it seem a very large and complex issue. However, the scope of this thesis will be specifically aiming to propose a *Security Model based Mobile Agent for Mobile Ad Hoc Networks*.

1.4 Main Contributions

The essential contribution of this thesis is to propose a Security Model based Mobile Agent for Mobile Ad Hoc Networks. This model based on network organization at three levels (node level, cluster level, and network level) for hierarchical management of the security services. We apply the concept of dominating set based clustering for partitioning network into clusters. The cluster head elected based on both the trust and resources ability of the node. Therefore, we have proposed an optimization function of five parameters to evaluate node resources and the formula to estimate the trust ability of the node. This trust is inspired by control all operation of the node (i.e., the reception, the transmission, the behavior, etc.).

In the first contribution, we define three agent types. The Node Agent (NA) manages the use of node resources. The Monitor Agent (MoA) that is responsible for all operations within the cluster and outside with counterparts. The Monitor Agent creates the Inspector Agent (IA), which travails from node to another to examine the actions history of each node agent to detect any suspect behavior, and returns to MoA with report shows the status of each node in the cluster.

In the second contribution, we proposed four types of agents. Node Agent (NA), Monitor Agent (MA), Ambassador Agent (AmA), and Transporter Agent (TA). The Monitor Agent creates the Ambassador Agents and sending to all nodes in the cluster. The Ambassador Agent is like local IDS and IPS (Intrusion Detection System and Intrusion Prevention System) at the node.

The third contribution is considered a hybrid approach for the two preceding. The Monitor Agent created in the most trusted with best resources node to control the communication inside and outside the cluster. In this framework, we propose an architecture of the mobile agent, which acquires the ability to react with unpredictable behavior and achieve security goals.

Finally, we can summarize the objective of our model satisfy almost the main purposes of the security: *Authentication*, where we used the Monitor Agent after the election process as trusted site. *Confidentiality*, we used the mechanism of cryptography symmetric inside the cluster and asymmetric outside the cluster. *Availability*, the Monitor Agent checks the presence of Nodes by it sends a message or by Inspector Agent. *Integrity*, to realize the integrity we use the hash value for verifying the data sent through the nodes of network. In our model, we used Secure Hash Algorithm 1 (SHA-1). *Non-repudiation*, the repudiation cannot appears in our model because Node Agent(s) or/and Ambassador Agent(s) records all sends and receives operations. In addition, the Inspector Agent has the ability to detect any repudiation through analysis and comparison.

1.5 Originality of Intended Work

Resolve the security issues are very important in mobile ad hoc networks, which will allows spread rapidly. Various security solutions have been proposed to protect these networks from vulnerabilities, which an attacker could take advantage of causing a disruption to the normal operation of the network or breaching the confidentiality of a user’s data. However, the solutions that have been proposed suffer from drawbacks. For example, there is no generalized framework that can be adapted to different types of mobile ad hoc network and application, i.e. although these approaches have been able to respond to a set of security requirements, they remain effective only in a specific context related to the assumptions and restrictive requirements that were issued during the design. In addition, most of these security schemes either provide protection to agents from agents/host or host from agent/external parties but not both. Moreover, most of these works are still only at the theoretical level and have not been implemented on the reality, and so on. The proposed model attempts to be cost effective, prevent various attacks, and try to avoid all the disadvantages of previous approaches.

1.6 Research Methodology

- **Background Research and Literature Review**

Research possible security threats and vulnerabilities associated with mobile ad hoc networks (MANETs). Then investigate current solutions that attempt to address these vulnerabilities and find the weaknesses in these solutions. The research will then continue by exploring the current types of security available in an attempt to find a more suitable solution.

- **Design Proposed Security Solution**

From the information gathered in the literature review, a solution will be put forward which addresses the needs of security services by eliminating the vulnerabilities, which used via the attackers and improving the security of previously proposed solutions.

- **Write and Submit Research Papers**

Using the information gathered in the literature review and the model of the proposed security solution and it has been published.

- **Create a Security Solution**

Create a security solution within the prototype environment. The agents modelled and the behavior the agents in relation to each other and the full model programmed. Security solution created using the Aglets accompanied with Java Development Kit (JDK 7) and the NetBeans IDE version 8.0.2. All these tools are installed on a computer running windows 7 system and equipped by Intel core i7 processor.

- **Test Prototype**

The prototype tested to see if the model operates correctly. After the test that carried out with a variety of scenarios, which examined in the literature review. We observe how the model reacts of the attacks, where it prevented or mitigated successfully.

- **Evaluation of Results**

The results of the prototype will be gathered and evaluated to see if any flaws are discovered and then possible improvements can be made to the design.

- **Thesis Write-up**

1.7 Structure of Dissertation

In the *second chapter*, we presented " A brief background of wireless networks", begins by answering the question "What is a Wireless Network?", next, it presents the categories of Wireless Networks, such as, infrastructure-based networks, infrastructureless networks, Wireless Local-Area Networks (WLANs), Wireless Personal-Area Networks (WPANs), Wireless Metropolitan-Area Networks (WMANs), Wireless Wide Area Networks (WWANs), and essential technology in each category.

The purpose of *chapter three* is to present rich details of "Security in Wireless Networks", this part discusses the concept, the security goals, attacks on wireless networks, challenges, security aspects, including, Service Set Identifier (SSID), authentication, the WEP Protocol, 802.1x, IEEE 802.11i, Virtual Private Networks (VPNs), firewalls, etc. We also discuss the variety of security mechanisms such as protocols, algorithms and key management. Although the covered topic may not be an exhaustive representation of all the security issues in wireless networks, we represent a rich and useful sample of the strategy and content.

The concept of mobile agent, their properties, advantages for using in wireless networks, and various aspects of security in the mobile agent discussed in the *chapter four*. It also dedicated in shedding light on the works developed which are used for the security based mobile agent in ad hoc networks and finished this part by giving our discussion and some critics.

The *five chapter* which are devoted to the proposal of our model that called ***"Security Model based Mobile Agent for Ad Hoc Networks***", that goal is to improve security in mobile ad hoc networks. This model based on network organization at three levels (node level, cluster level, and network level) for hierarchical management of the security services. We propose an optimization function of five parameters to evaluate node resources and the formula to estimate the trust ability of the node, a network topology based on the concept of clusters with the mobile agent technology. We define several types of agents, in this framework, we propose an architecture of the mobile agent, which acquires the ability to react with unpredictable behavior and achieve security goals. We gave class diagrams, and communication protocol of our Model.

The *six chapter* describes the implementation of our model, where we used a platform for developing mobile agents published by IBM called Aglets, accompanied with java development kit (JDK 7) and the NetBeans IDE version 8.0.2. All these tools are installed on a computer running windows 7 system and equipped by Intel core i7 processor. For testing the implementation, we used ad hoc network include four machines, such as every machine is configured to run the Aglet Agents. The tests demonstrate a feasibility the model developed to increase the level of security in the ad hoc network, without effect it performance.

Finally, in *chapter seven*, we are finishing through a conclusion with few prospects that come to crown this work, showing the strong points and weakness, and defining the future work, we intend to lead and that go in the direction of improving security in ad hoc networks.

Chapter 2

Overview of Wireless Networks

> We are going to turn, in the distant future, into a race of people who possess extraordinary communication with one another.
>
> Francais Ford Coppola

This chapter gives an overview of Wireless Networks. It begins by answering the question what is a Wireless Network?. Next, it presents the categories of Wireless Networks, such as, Infrastructure-based Networks, Infrastructureless Networks, Wireless Local-Area Networks (WLANs), Wireless Personal-Area Networks (WPANs), Wireless Metropolitan-Area Networks (WMANs), Wireless Wide Area Networks (WWANs), and essential technology in each category.

2.1 What is a Wireless Network?

A wireless network is, as its name suggests, a network in which at least two devices (computer, PDA, printer, router, etc.) can communicate without physical wires [8]. It use radio waves rather than cables to broadcast network traffic and transmit data. Wireless networks can be operated in two different modes. Ad-hoc mode consists of at least two wireless stations where no access point is involved in their communication. In the infrastructure mode, communication between nodes are routed through an Access Point (AP), which is analogous to the base station in a cellular network.

2.2 Classification of Wireless Networks

A number of different wireless networks exist and can be categorized in various ways depending on the criteria chosen for their classification, such as network architecture and communication coverage area [9].

Based on Network Architecture, where wireless networks can be divided into two broad categories based on how the network is constructed: Infrastructure-based networks, Infrastructureless networks. Based on Communication Coverage Area, where, as with wired networks, wireless networks can be categorized into different types of networks based on the distances over which the data are transmitted: Wireless Wide-Area Networks (WWANs), Wireless Metropolitan-Area Networks (WMANs), Wireless Local-Area Networks (WLANs), Wireless Personal-Area Networks (WPANs) [10].

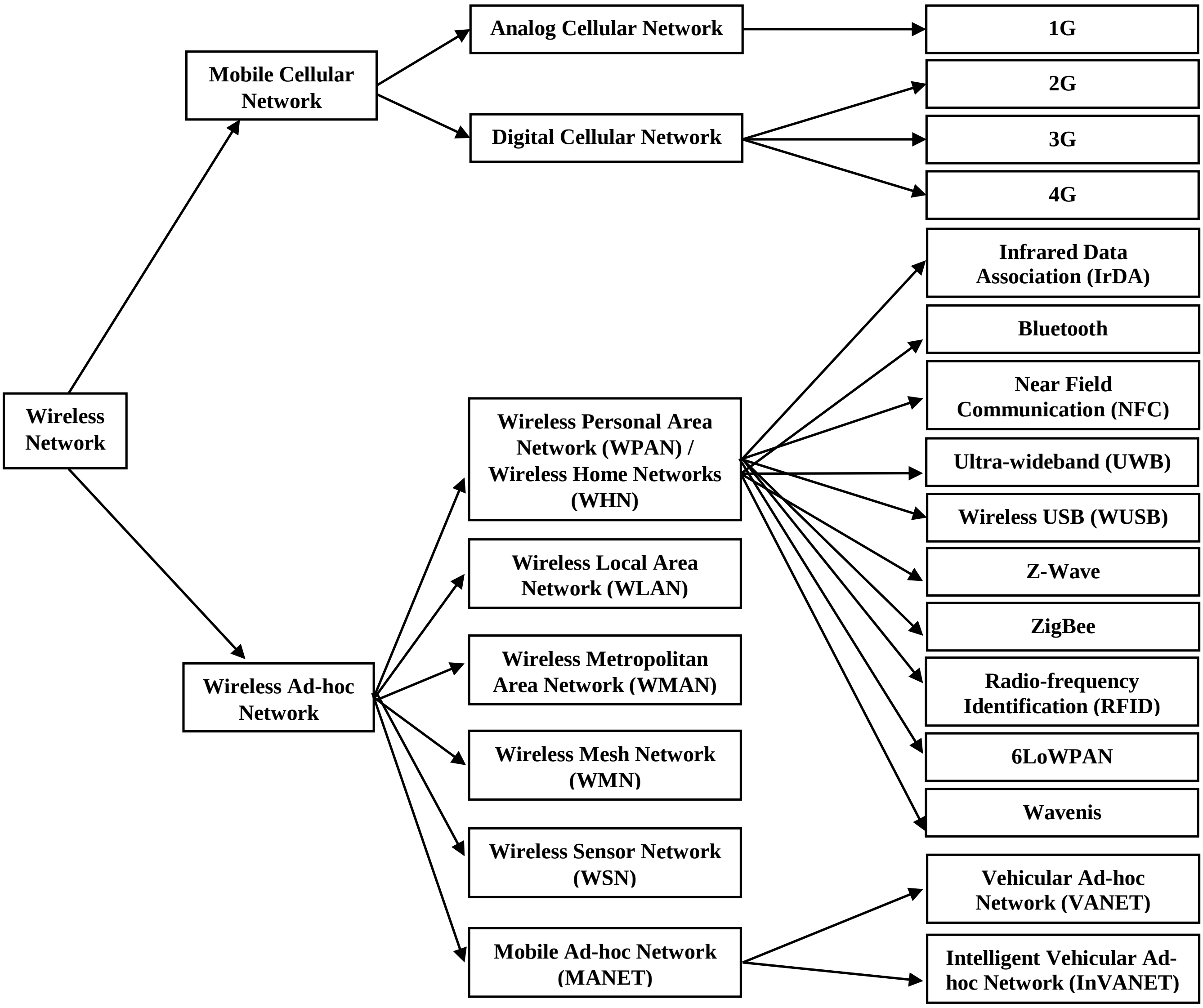

Wireless Network
Mobile Cellular Network
Analog Cellular Network
Digital Cellular Network
1G
2G
3G
4G
Wireless Ad-hoc Network
Wireless Personal Area Network (WPAN) / Wireless Home Networks (WHN)
Wireless Local Area Network (WLAN)
Wireless Metropolitan Area Network (WMAN)
Wireless Mesh Network (WMN)
Wireless Sensor Network (WSN)
Mobile Ad-hoc Network (MANET)
Infrared Data Association (IrDA)
Bluetooth
Near Field Communication (NFC)
Ultra-wideband (UWB)
Wireless USB (WUSB)
Z-Wave
ZigBee
Radio-frequency Identification (RFID)
6LoWPAN
Wavenis
Vehicular Ad-hoc Network (VANET)
Intelligent Vehicular Ad-hoc Network (InVANET)

Figure 2.1: Presents a Classification of Wireless Networks and Technologies

2.2.1 Mobile Cellular Networks

Wireless Cellular Networks (infrastructure-based network) is a network that has a pre-constructed infrastructure that is made of a fixed network structure (typically, wired network nodes and gateways). Network services are delivered via these pre-constructed infrastructures. [9]. A brief overview on each generation is given below.

- ***First-Generation Mobile Systems (1G)***

The first-generation mobile system started in the 1980s was based on analogue transmission techniques. At that time, there was no worldwide coordination for the development of technical standards for the system [11]. As a result, various standard systems were developed worldwide: Advanced Mobile Phones Service (AMPS) in the United States, Nordic Mobile Telephones (NMT) in Europe, Total Access Communication Systems (TACS) in the United Kingdom, Nippon Telephone and Telegraph (NTT) in Japan, and so on [12].

- ***Second-Generation Mobile Systems (2G)***

The term “Second-Generation Mobile Systems” is a generic term referring to a range of digital cellular technologies [13]. Compared with first-generation systems, second-generation (2G) systems use digital multiple-access technology, such as time-division multiple access (TDMA) and Code Division Multiple Access (CDMA) [9].

The *Global System for Mobile Communications (GSM)* is by far the most successful 2G cellular technology and the first cellular technology to achieve market acceptance on a global scale. By 1991, GSM was the first commercially operated digital cellular system with Radiolinja in Finland. The GSM system operates at various radio frequencies, with most them operating at 900 MHz and/or 1800 MHz. In the US and Canada, the operation is at 850 MHz and/or 1900 MHz [14]. GSM uses a combination of time division multiple access (TDMA) and frequency division multiple access (FDMA) to divide physical resources among multiple users [15]. The original GSM has evolved into a family of standards (3GPP) that includes several technologies, which have been designed to be coexisting as complementary systems. These are the cases of GPRS, EDGE, WCDMA (FDD and TDD) and TD-SCDMA [16].

General Packet Radio Service (GPRS) is referred to by many as a 2.5G technology an evolution from 2G Global System for Mobile Communications (GSM) technology and an interim phase toward 3G high-speed services including multimedia traffic with different QoS requirements. The European Telecommunications Standards Institute (ETSI) conducted GPRS standardization efforts during the mid and late 1990s [17]. It was planned to evolve the existing GSM network, together with the idea of the 3G network, which was conceived of circuit-switched services, into a mixed voice and data system that could share some of the network elements that were to be deployed for 3G purposes. With this objective, and trying to reuse as much as possible the existing infrastructure, it was conceived the General Packet Radio System (GPRS) for GSM. The changes introduced affected at different levels and network elements, but none of them modified the air interface [16].

The limitation of the GPRS network such as the data rates, led to ETSI standardization organization defined a new family of data services, built on the existing structure of GPRS. This new family of data services was initially named Enhanced Data rates for GSM Evolution, and subsequently renamed *Enhanced Data for Global evolution (EDGE).* Nevertheless, the disadvantage with EDGE is that the data rates offered are not necessarily available throughout the cell. If EDGE is to be offered with complete coverage, the amount of cells will increase dramatically [18].

❖ *Third-Generation Mobile Systems (3G)*

In effect, 2G systems are limited in terms of maximum data rate. While this fact is not a limiting factor for the voice quality offered, it makes 2G systems practically useless for the increased requirements of future mobile applications [19]. The goals of the third-generation (3G) wireless networks are to solve problems of 2G and 2.5G systems and offer wireless Internet services at a wide scale, extending the scope of 2G wireless networks from simple voice telephony to complex data applications including voice over Internet protocol (VoIP), video conferencing over IP, Web browsing, and multimedia services [20].

Universal Mobile Telecommunications Systems (UMTS) is the generic name for the third generation of GSM cellular radio mobile systems. More precisely, UMTS is the European vision of International Mobile Telecommunications 2000 (IMT-2000). It represents the ITU-T initiative to conceive, design and implement 3G mobile systems for the year 2000 and beyond, promising data rates or bandwidth up to 2 Mbps [21]. The key benefits of UMTS include improvements in quality and services, incorporated broadband and sophisticated multimedia services, flexibility of future service creation and introduction, and ubiquitous service portability. Three G networks become a reality when they meet the UMTS standards and offer true ubiquity of an IP packet-switched backbone that can deliver any communication service anywhere [22].

❖ ***Fourth-Generation Mobile Systems (4G)***

The ever-increasing growth of user demands, the limitations of 3G and the emergence of new mobile broadband technologies on the market have prompted researchers and industries to a thorough reflection on 4G. [23]. 4G is a loose term for the fourth generation of cellular communications, offering speeds that are about 10 times faster than they are on current third-generation, or 3G, networks. Its higher data speeds could make smartphones much more comparable to PCs, giving them better multimedia and gaming capabilities [24].

2.2.2 Wireless Local-Area Networks (WLANs)

A wireless local area network (WLAN) is a data communications system implemented as an extension or as an alternative to a wired LAN. Using a variety of technologies including narrowband radio, spread spectrum, and infrared, wireless LANs transmit and receive data through the air, minimizing the need for wired connections [25].

WLANs can function in two primary modes of operation “ad-hoc mode” (also referred to as “peer-to-peer mode”) and “infrastructure mode.” Each one functions in a slightly different way and each has advantages within specific situations. Some networks actually make use of both modes and are thus called “hybrid mode” networks [26].

In “infrastructure mode”, two stations exchanging data can communicate only through an access point. An access point (AP) is a device that accepts wireless signals from multiple nodes and retransmits them to the rest of the network. Access points may also be known as base stations. Access points for use on small office or home networks often include routing functions. As such, they may also be called wireless routers or wireless gateways [27].

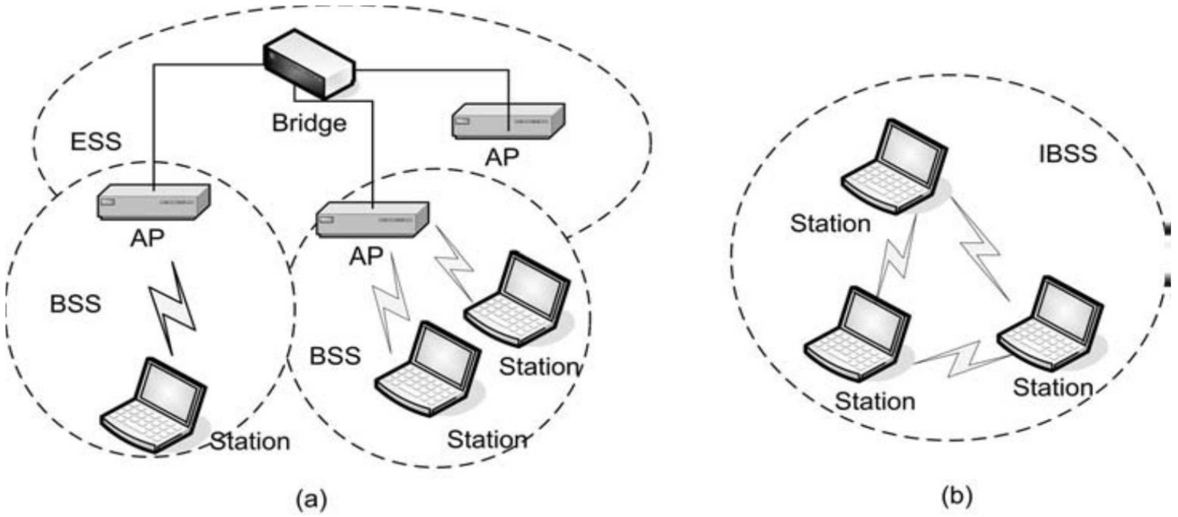


Figure 2.2: Illustration of WLAN Configuration: (a) ESS Composed of Infrastructure BSS, and (b) IBSS [28]

In "ad hoc" mode, wireless devices can communicate with each other directly and do not use an access point. This is so called an Independent BSS (IBSS). These independent networks usually do not require any administration or pre-configuration [29]. However, an ad hoc arrangement would not work well for a WLAN with many users or whose users are spread out over a wide area, or where obstacles could stand in the way of signals between stations.

Several Standards exist for WLANs, such as IEEE 802.11, a, b, c, d,...., ac, ad af, and the last standard (In January 2016) is 802.11ah to be called Wi-Fi HaLow.

2.2.3 Wireless Personal Area Networks (WPANs)

A Wireless Personal Area Network (WPAN) is a computer network used for wireless communication among devices (including telephones and personal digital assistants) close to a person. The reach of a WPAN is typically a few meters. WPANs can be used for communication among the personal devices themselves (intrapersonal communication), or for connecting to a higher-level network or the Internet (an uplink) [30].

The main design goal of WPANs is to allow devices that are in close proximity to communicate and exchange information with each other, either stationary or moving. WPANs can be used to replace cables between computers and their peripherals; to share multimedia content amongst devices; to build an infrastructure for sensor networking applications; or to establish various location aware services. The operating range for these devices is within a Personal Operating Space (POS) of up to 10 meters in all directions, and envelops a stationary or a mobile person. The concept of a POS can also be extended to devices that are not attached to a person, like peripherals such as printers, scanners, digital cameras, etc. [31].

Several technologies exist for WPANs, such as Bluetooth, ZigBee, Z-Wave, RFID, Infrared Data Association (IrDA), and Ultra-Wideband (UWB), to mention a few. Each of these technologies has its own particular strengths and weaknesses in the way it addresses the challenge of delivering easy to use.

2.2.4 Wireless Metropolitan Area Networks (WMANs)

Wireless metropolitan area networks (WMANs) enable users to establish wireless connections between multiple locations within a metropolitan area (for example, between multiple office buildings in a city or on a university campus), without the high cost of laying fiber or copper cabling and leasing lines. In addition, WMANs can serve as backups for wired networks, should the primary leased lines for wired networks become unavailable. WMANs use either radio waves or infrared light to transmit data. Broadband wireless access networks, which provide users with high-speed access to the Internet, are in increasing demand [32].

WMAN technologies must satisfy a wide range of service requirements such as broadband access capability, reliability, scalability, security, quality of services, manageability, and cost effectiveness, all at a time when multimedia traffic is exploding at exponential rates [21].

Finally, one main disadvantage of a WMAN presents a very large area for a hacker to attempt a break in. Like any wireless access network, a wireless access point is a tempting target for someone to hack into a secure network. A WMAN is simply a larger WAN footprint, and thus presents a larger opportunity for a hacker [32].

2.2.5 Wireless Wide Area Networks (WWANs)

A Wireless Wide Area Network (WWAN) is a class of WAN technologies that uses mostly cellular and satellite infrastructures to enable interconnectivity over a WAN via several services [33], such as LTE, WiMAX (often called a WMAN), UMTS, CDMA2000, GSM, Cellular Digital Packet Data (CDPD) and Mobitex to transfer data. It can also use Local Multipoint Distribution Service (LMDS) or Wi-Fi to provide Internet access. These technologies are offered regionally, nationwide, or even globally and are provided by a wireless service provider. WWAN connectivity allows a user with a laptop and a WWAN card to surf the web, check email, or connect to a virtual private network (VPN) from anywhere within the regional boundaries of cellular service. Various computers can have integrated WWAN capabilities [34].

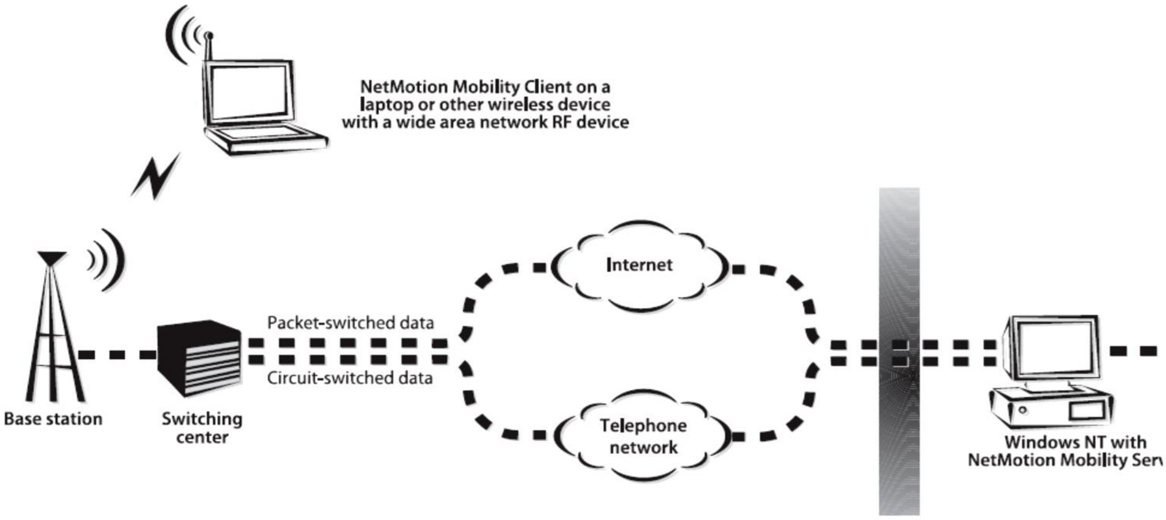


Figure 2.3: Components of a Wireless WAN [35**]**

Wireless connectivity can be installed based on one of two models: where the receiver is fixed in position, or where the receiver is mobile. Within these models, there are different solutions and options that can be applied.

The two key solutions are the use of satellite and mobile phone technology. Satellite technology is portable, though it requires an element of configuration at each site to point the dish at the satellite. Satellite wireless links often work downstream only, from the satellite to the receiver, rather than downstream and upstream. In some cases, the upstream connection is achieved with a normal telephone line or mobile phone connection [36].

Mobile computing devices are getting smaller, cheaper and more powerful. At the same time, the amount of information available today is growing astronomically. The demand for connecting mobile devices to content rich networks is also rising quickly, and it would seem that WWAN technology is the perfect answer. However, today's wireless WANs have limitations in several areas, including: security, performance, application persistence, roaming, central management, etc. [35].

2.2.6 Wireless Mesh Networks (WMNs)

A wireless mesh network (WMN) [37] is a particular type of mobile ad hoc network (MANET), which aims to provide ubiquitous high bandwidth access for a large number of users. WMN has the ability of self-organization, self-discovering, self-healing, and self-configuration. A WMN is a consisting of two parts: mesh backbone and mesh clients as shown in Fig. 2.4.

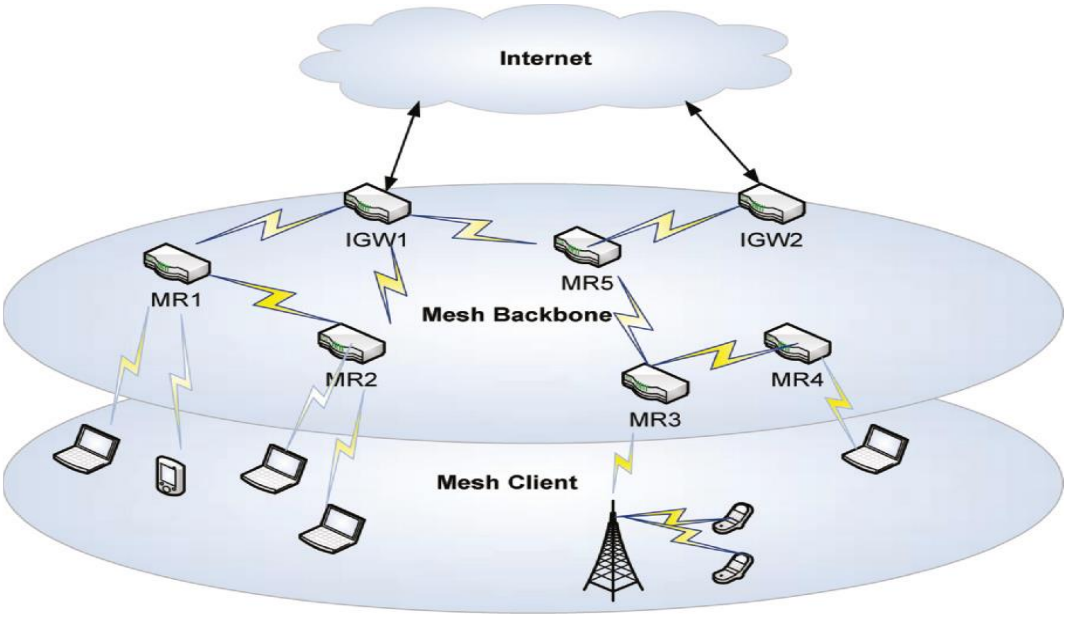


Figure 2.4: A typical Architecture of a WMN

The primary advantages of a WMN lie in its inherent fault tolerance against network failures, simplicity of setting up a network, and the broadband capability. Unlike cellular networks where the failure of a single Base Station (BS) leading to unavailability of communication services over a large geographical area, WMNs provide high fault tolerance even when a number of nodes fail [38].

2.2.7 Mobile Ad hoc Networks (MANETs)

A mobile ad hoc network (MANET) is a collection of wireless mobile nodes (or routers) dynamically forming a temporary network without the use of any existing network infrastructure or centralized administration [39]. In this environment, nodes are free to move randomly and organize themselves arbitrarily; thus, the network's wireless topology may change rapidly and unpredictably [40]. Nodes in ad hoc networks are computing and communication devices, which can be laptop computers, PDAs, mobile phones, or even sensors.

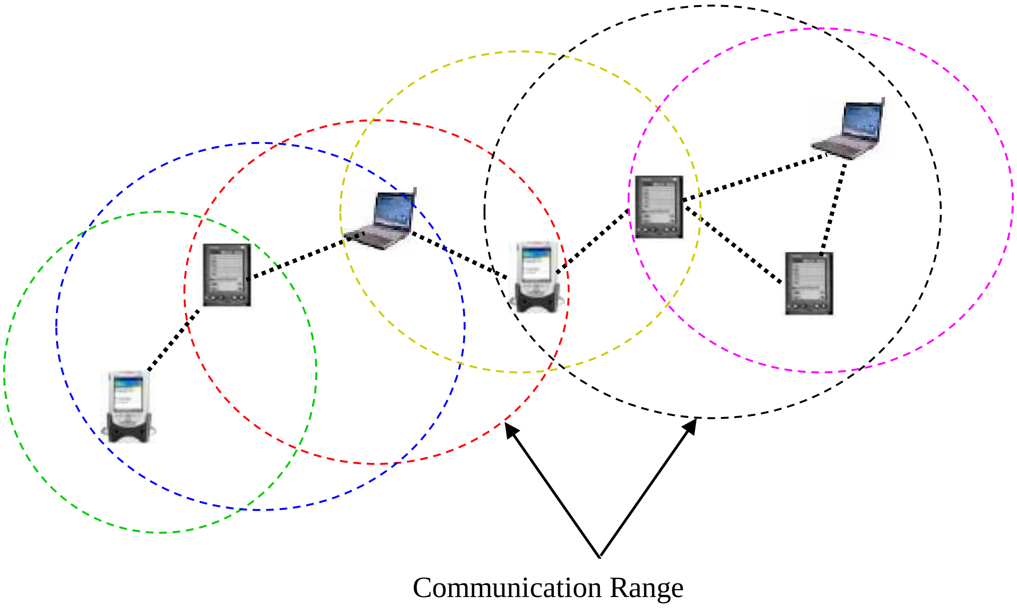


Figure 2.5: Illustration of Mobile Ad Hoc Networks

Nodes in a MANET do not have a priori knowledge of the network topology. They have to discover it. A node will find its local topology by broadcasting its presence, and listening to broadcast announcements from its neighbors. As time goes on, each node gets to know about all other nodes and finds one or more ways to reach them. End-to-end communication in a MANET does not rely on any underlying static network infrastructure but requires routing via several intermediate nodes [40].

Recently, MANETs have attracted considerable attentions due to the new developments in wireless technology and standards. There are numerous possibilities for applications of the ad hoc concept in the networking world. Variety of services and applications were developed ranging from tactical military networks, through different commercial and educational applications, sensor networks, to location-aware services. Fast deployment and easy establishment of functionalities, autonomous or relayed communications, cooperativeness and emerging areas of nomadic and ubiquitous computing, as well as improved IP-based networking in dynamic autonomous wireless environments promote ad hoc networks as a desirable wireless access technology. Ad hoc networking also gets momentum in emergency communication in catastrophic disaster areas and during terrorist attacks. They participate in collaborative and distributed computing and mesh (infrastructure relayed) and hybrid (integrated cellular and ad hoc) wireless networks [41].

2.2.8 Wireless Sensor Networks (WSNs)

Recent advances in miniaturization, low-cost and low-power circuit design, and wireless communications have led to the development of low-cost, low power, and tiny communication devices, called sensors. Like nodes (or computers, laptops, etc.) in traditional wireless networks, such as mobile ad hoc networks, the sensors have data storage, processing, and communication capabilities. Unlike those nodes, the sensors have an extra functionality related to their sensing capability [42].

Generally, a sensor node refers to any device that is capable to sense its environment. Wireless Sensor Network (WSN) as a technology is a collection of sensor devices that co-operate with each other. A WSN may comprise even thousands of autonomic and self-organizing nodes that combine environmental sensing, data processing, and wireless multi-hop ad-hoc networking [43].

The main goal of a WSN is to collect data from the environment and send it to a reporting site where the data can be observed and analyzed. Wireless sensor devices also respond to queries sent from a “control site” to perform specific instructions or provide sensing samples. Finally, wireless sensor devices can be equipped with actuators to “act” upon certain conditions. These networks are sometimes more specifically referred as Wireless Sensor and Actuator Networks [44].

Compared to traditional wireless networks, such as mobile ad hoc wireless networks, WSNs have several inherent characteristics. First, the sensors are very tiny and hence more susceptible to hardware failure. It is worth mentioning that battery power (or energy) is the most crucial resource, and hence sensors can also fail due to low energy. Second, the sensors are deployed in a field with high density to extend the network lifetime. Indeed, using a large number of sensors facilitates multi-hop communication between them, and hence the sensors can save their energy by transmitting or forwarding their sensed data through short distances. Third, the network topology may change very frequently as sensors join and/or leave the network. Thus, protocols designed for WSNs should account for all these features, which are inherent to these types of networks so they remain operational as longer as possible [42].

Wireless sensor networks (WSNs) are used in a variety of applications, such as environment monitoring, health care, natural disaster prediction and relief, precision agriculture, security, manufacturing, home appliances and entertainment, transportation, food safety, military operations, water quality monitoring, intelligent transportation, smart grid communications, and so forth [45, 46, 47].

2.3 Conclusion

In this chapter, we have given a simple overview on the concept of wireless networks. In fact, the field of wireless networks is very broad and difficult to present to all the details. We tried to give a brief overview of the types of wireless networks, such as, Infrastructure-based networks, Infrastructureless networks, Wireless local-area networks (WLANs), Wireless personal-area networks (WPANs), Wireless metropolitan-area networks (WMANs), Wireless wide area networks (WWANs), and essential technology in each category.

Chapter 3

Security in Wireless Networks

To say a system is secure because no
one is attacking it is very dangerous

(Microsoft Founder, Bill Gates)

In this chapter, the security of wireless networks is addressed. First, answering the question "What Is Security?", next, we present security goals; attacks on wireless networks, the main security characteristics, various issues and challenges in various wireless networks, especially ad hoc networks. We also discussed the variety of security mechanisms such as protocols, algorithms and key management. Although the covered topic may not be an exhaustive representation of all the security issues in wireless networks, we do represent a rich and useful sample of the strategy and content.

3.1 Introduction

As we have seen previously, Wireless networks are becoming more and more popular at the office, home, a hotel, coffee shops, airports, train stations, and many other places. However, the bad news that wireless networks are a major target for attackers. One of the biggest challenges today is to make sure that the appropriate tools and mechanisms are used to protect data in-transit across wireless networks. In addition, the wireless infrastructure needs to be protected against attacks targeted to the wireless networking devices [48].

Wireless networks face more security challenges than their wired counterparts. This is partly due to the nature of the wireless medium as transmitted signals can travel through the walls, ceilings, and windows of buildings up to thousands of feet outside of the building walls. Moreover, since the wireless medium is airwaves, a shared medium allows anyone within certain distance or proximity to intrude into the network and sniff the traffic. Further, the risks of using a shared medium is increasing with the advent of available hacking tools that can be found freely from hacker’s Web sites [49].

On the other hand, wireless (mobile) devices usually have limited bandwidth, storage space, and processing capacities. It is harder to reinforce security in wireless networks than in wired networks. Compared with infrastructure-based wireless networks, security management for wireless ad hoc networks is more challenging due to unreliable communication, intermittent connection, node mobility, and dynamic topology. A complete security solution should include three components of prevention, detection, and reaction. It provides security properties of authentication, confidentiality, non-repudiation, integrity, and availability. It should be adaptive in order to trade-off service performance and security performance under resource limitation [40]. Wireless networks use several techniques to provide secure transfer of voice, data, or video (As we will see later).

3.2 Terminology

Within the security community, some words have specific meanings. Common security vocabulary [50] includes the following:

- **Vulnerability**: A defect or weakness in the feasibility, design, implementation, operation, or maintenance of a system.
- **Threat**: An adversary who is capable and motivated to exploit a vulnerability.

- **Attack**: The use or exploitation of a vulnerability. This term is neither malicious nor benevolent. A bad person may attack a system, and a good person may attack a problem.
- **Exploit**: The instantiation of a vulnerability; something that can be used for an attack. A single vulnerability may lead to multiple exploits, but not every vulnerability may have an exploit (e.g., theoretical vulnerabilities).
- **Target**: The person, company, or system that is directly vulnerable and impacted by the exploit. Some exploits have multiple impacts, with both primary (main) targets and secondary (incidental) targets.
- **Defender**: The person or process that mitigates or prevents an attack.
- **Compromise**: The successful exploitation of a target by an attacker.
- **Risk**: A qualitative assessment describing the likelihood of an attacker/threat using an exploit to successfully bypass a defender, attack a vulnerability, and compromise a system.
- **Hacker**: The term hacker is closely associated with computer security. In contrast to the common terminology, hacker has conflicting definitions. This term originally referred to people with highly technical skills. Later it became associated with technical malcontents and online anarchists. Today, people with virtually no technical skills are called hackers if they use a computer.

3.3 What Is Security?

Evidently, the notion of security has many facets, which might depend on the point of view of a specific investigation, the levels of abstraction under consideration, or even social agreements or personal opinions. In any case, it appears demanding to treat security in computing systems as a comprehensive property that takes care of many aspects with mutual impacts. It is necessary to find a definition of security common to an asset, a service, an infrastructure and an infosphere for any concerned owner. We can find several definitions of security.

According to [51] security is, freedom from danger, safety; freedom from fear or anxiety. Another definition in [52] Security is the sum of all measures taken to prevent loss of any kind. Loss can occur because of user error, defects in code, malicious acts, hardware failure, and acts of nature. With holistic computer security, a number of methods are used to prevent these events, but it is primarily focused on preventing user error and malicious acts.

Network security is defined as the protection of networks and their services from unauthorized access, modification, destruction, or disclosure. It provides assurance that the network performs its critical functions correctly and that there are no harmful side effects [53].

The security of wireless systems can be divided into four sections [54]:

- Security of the application: This means the security of user applications and standard applications such as e-mail.
- Security of the devices: How to protect the physical device in case it is lost or stolen.
- Security of the wireless communication: How to protect messages in transit.
- Security of the server that connects to the Internet or other wired network: After this server, the information goes to a network with the usual security problems of a wired network (not discussed here).

Security consists in assessing threats, vulnerabilities and attacks. It also involves estimating the cost of the threats and the probabilities of the attacks given the vulnerabilities. Once these points assessed, security encompasses developing appropriate safeguards and countermeasures, and implementing the ones for which the certain price of the defense is worth spending compared to the uncertain loss that a potential threat implies. To achieve this, security properties or requirements should be defined [55].

3.4 Security Goals

Five major security goals known as security services and can be used as security requirements. These goals are discussed as follows [5, 40, 56, 57, 58].

- **Authentication**: Authentication is the process to verify the identity of the sender of a communication. Without authentication, malicious attackers can access resource, gain-sensitive information, and interfere with the operation of other nodes very easily. In a wireless network, this procedure is commonly done both at the network layer and by higher layer protocols that the application uses. We can perform authentication with either a public key or a secret key. The simplest form of authentication is the transmission of a shared password between the entities wishing to authenticate with each other.
- **Confidentiality**: Confidentiality means certain information is only accessible to authorized recipients. Encryption is used to fulfill this goal. With an active attack, it is

possible to decrypt any form of encrypted data; thus, confidentiality is primarily considered a protection against passive attacks. In the physical world, ensuring confidentiality can be accomplished by simply securing the physical area. However, as evidenced by bank robberies and military invasions, threats exist to the security of the physical realm that can compromise security and confidentiality. The moment electronic means of communication were introduced, many new possible avenues of disclosing the information within these communications were created.

- **Integrity**: Prevents unauthorized changes to the data. Only authorized parties are able to modify the data. Modification includes changing status, deleting, creating, and delaying or replaying of the transmitted messages. Without integrity, attackers can easily corrupt and modify the data and therefore cause mobile devices to make wrong decisions based on the corrupted data.
- **Availability**: As defined in an information security context, ensures that access data or computing resources needed by appropriate personnel is both reliable and available in a timely manner. Redundancy, fault tolerance, reliability, failover, backups, recovery, resilience, and load balancing are the network design concepts used to assure availability. If systems aren’t available, then integrity and confidentiality won’t matter.
- **Non-repudiation**: Non-repudiation means that the sender of a message cannot later deny sending the information and the receiver cannot deny the reception. This can be useful while detecting and isolating compromised nodes. Any node that receives an erroneous message can accuse the sender with proof and, thus, convince other nodes about the compromised node.

3.5 Attacks on Wireless Networks

While listening on the wireless radio link is the obvious attack, other attacks also exist. These attacks can be classified according to different criteria, such as the domain of the attackers, or the techniques used in attacks. This section summarizes some of these attacks. The list of attacks provided here is by no means a comprehensive list of possible attacks but provides a broad view of the attacks that need to be addressed which will motivate the subsequent chapters discussing approaches to defending against such attacks. Normally, network security attacks are divided into passive and active attacks. Another classification [5] of attacks (Attacks against secure routing) as, internal attacks and external attacks.

Internal attacks are especially relevant in ad hoc networks, which are operating in hostile environments like enemy battlefields. These attacks are more severe attacks, because of two important reasons. First, if a node determines that the routing information that it has received is invalid, it is difficult for it to conclude whether the information that it has received became invalid because of changes in the network topology or because the sending node was compromised. Second, a compromised node would still arguably be able to generate valid signatures using its private keys, thus making it even harder to use cryptography to detect that it has been compromised [59].

External attacks on routing can be divided into two categories: passive and active [5]. Passive attacks involve unauthorized "listening" to the routing packets. The attack might be an attempt to gain routing information from which the attacker could extrapolate data about the positions of each node in relation to the others.

Active attacks on the network from outside sources are meant to degrade or prevent message flow between the nodes. Active external attacks on the ad hoc routing protocol can collectively be described as denial-of-service attacks, causing a degradation or complete halt in communication between nodes. One type of attack involves insertion of extraneous packets into the network in order to cause congestion. A more subtle method of attack involves intercepting a routing packet, modifying its contents, and sending it back into the network. Alternatively, the attacker can choose not to modify the packet's contents but rather to replay it back to the network at different times, introducing outdated routing information to the nodes.

3.5.1 Passive Attacks

An attack is called passive when an unauthorized person obtains access to a resource without changing its content [60]. In a passive attack, an unauthorized node monitors and aims to find out information about the network. The attackers do not otherwise need to communicate with the network. Hence, they do not disrupt communications or cause any direct damage to the network [61]. The goal of the opponent is to obtain information that is being transmitted [62]. Examples of passive attacks are eavesdropping and traffic analysis.

- **Eavesdropping**

Eavesdropping also known as, disclosure is a very easy passive attack in the radio environment [63]. This attack consist of the unauthorized interception of network communication and the disclosure of the exchanged information. This can be performed in several different layers, for example, in the network layer by sniffing into the exchanged packets or in the physical layer by physically wiretapping the access medium (cabling or wireless medium) [64].

In this attack, the intruder listens to things he or she is not supposed to listen it. This information could contain, for example, the session key used for encrypting data during the session. This kind of attack means that the intruder can get information that is at times strictly confidential [58].

- **Traffic Analysis**

This is a subtle form of passive attack. The objective of an adversary launching this attack is to extract information about the characteristics of transmission. This could include information about the amount of data transmitted, identity of communicating nodes, or their locations [63]. It is possible that at times for the intruder knowing the location and identity of the communicating device or user is enough. An intruder might only require information like a message has been sent, who is sending the message to whom, and at the frequency or size of the message [58].

3.5.2 Active Attacks

An attack is called active when making unauthorized changes are made to messages and data flows or files [60]. It is quite difficult to prevent active attacks absolutely, because to do so would require physical protection of all communications facilities and paths at all times [62]. Active attacks may take the form of one of the four following types, either singly or as a combination:

- **Masquerade**

An intruder pretends to be a trusted user and thereby gains certain unauthorized privileges. Such an attack is possible if the intruder captures information about the user like the authentication data, simply the username and the password. Masquerading includes the use of spoofing, rogue APs, and redirection attacks [47].

- **Replay**

A replay attack occurs when the attacker is able to capture traffic from one party and replay it to another, causing the targeted party to perform actions as if the traffic had been received from a legitimate sender. Replay attacks are often coupled with other attacks, such as man-in-the-middle attacks or denial-of-service attacks [65].

- **Message Modification**

The attacker alters a legitimate message by deleting, adding, modifying or rearranging the contents [60].

- **Denial-of-Service (DoS)**

The attacker prevents or prohibits normal usage of the management of the communication medium [60]. The denial-of-service attack is one of the simplest attacks to implement in wireless networks, an attacker attempts through some means to reduce the ability of a network or server to provide service to legitimate users. For example, an entity may suppress all messages directed to a particular destination (e.g., the security audit service). Another form of service denial is the disruption of an entire network, either by disabling the network or by overloading it with messages to degrade performance [62]. Sabotage is also a form of DoS attack. A DoS attack termed as sabotage could also mean the destruction of the system itself [58]. DoS attacks are possible at various layers, namely, physical layer, MAC layer, and network layer, and on the applications executing in such networks. For example, jamming of radio frequencies could be done at the physical layer similarly; violation of medium access control rules could lead to denial of service at the link layer [63].

- **Man-in-the-Middle Threat**

Man-in-the-middle (MitM) attacks occur when the attacker manages to position themselves between the legitimate parties to a conversation. The attacker spoofs the opposite legitimate party so that all parties believe they are actually talking to the expected other, legitimate parties. A MitM attack allows the attacker to eavesdrop on the conversation between the parties, or to actively intervene in the conversation to achieve some illegitimate end [65].

3.5.3 Software-Based Attacks

Malicious software, malicious code, or malware, is software that enters a computer system without the owner’s knowledge or consent. Malicious mobile code deals with *viruses, worms, Trojan horses,* and similar problems of rogue code that might compromise security policy. Malware is a general term that refers to a wide variety of damaging or annoying software. One way to classify malware is by primary objective. The three primary objectives of malware are to infect a computer system, conceal the malware’s malicious actions, or bring profit from the actions that it performs [57, 66].

3.5.4 Attacks in Wireless Ad Hoc Networks

Wireless ad hoc networks are target for all the threats that occur in networks, the existence of the wireless transmission links and dynamic network topology contributes considerably towards increasing the threat potential [41]. Attacks in ad hoc networks can cause congestion, propagate incorrect routing information, prevent services from working properly, or shut them down completely. Nodes that perform active attacks with the aim of damaging other nodes by causing network outage are considered to be malicious, also referred to as compromised, whereas nodes that perform passive attacks with the aim of saving battery life for their own communications are considered to be selfish [4]. This section summarizes some of these attacks. The list of attacks provided here is by no means a comprehensive list of possible attacks but provides a broad view of the attacks that need to be addressed which will motivate the subsequent chapters discussing approaches to defending against such attacks. [5, 19, 40, 41, 61, 63, 64, 67, 68, 69, 70].

- **Impersonation**

We talked to this attack under name Masquerade, impersonation attacks are launched by using other node's identity, such as MAC or IP address. Impersonation attacks sometimes are the first step for most attacks, and are used to launch further, more attacks that are sophisticated. Spoofing occurs when a malicious node misrepresents its identity by altering its MAC or IP address in order to alter the vision of the network topology that a benign node can gather. Impersonation attacks constitute a serious security risk at all levels of ad hoc networking. If proper authentication of parties is not supported, compromised nodes may be able to join the network undetectably, send false routing in formation, and masquerade as some other trusted node.

- **Location Disclosure**

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

- **Trust Attacks**

Trust is a privilege associated between the identities of the user with particular trust level. Therefore, a trust hierarchy is an explicit representation of trust levels that reflects organizational privileges. It associates a number with each privilege level, which reflects the security, importance, or capabilities of the mobile node and of the paths. Inside or outside nodes can initiate attacks on the trust hierarchy, if they try to impersonate anyone else and obtain higher-level privileges.

- **Black Hole Attack**

Black hole attacks cause packets to disappear from the network without any trace. In this attack, a malicious node uses the routing protocol to advertise itself as having the shortest path to the node whose packets it wants to intercept. The attacker will then receive the traffic destined for other nodes and can then choose to drop the packets to perform a denial-of-service attack, or alternatively use its place on the route as the first step in a man-in-the-middle attack by redirecting the packets to nodes pretending to be the destination.

Black hole attacks have other variants. In one variant, the data traffic is forwarded to a non-existent or another malicious node, where the data are dropped. This forwarding behavior before the actual sinking will make the detection of sinking behavior hard. Other than sinking, there are numerous ways by which a malicious node can coerce benign nodes to drop incoming traffic. Essentially, the malicious node achieves this by disrupting the benign route between source and destination. Route disruption can be enforced using spoofing and modification of routing messages.

- **Wormhole**

Wormhole attack: a malicious node can eavesdrop or receive data packets at a point and transfer them to another malicious node, which is at another part of the network, through an out-of-band channel. The second malicious node then replays the packets. This makes all the nodes that can hear the transmissions by the second malicious node believe that the node that sent the packets to the first malicious node is their single-hop neighbor and they are receiving the packets directly from it.

- **Sleep Deprivation**

Sleep deprivation (sometimes called Flooding Attack): Usually, this attack is practical only in ad hoc networks where battery life is a critical parameter. Battery powered devices try to conserve energy by transmitting only when absolutely necessary. An attacker can attempt to consume batteries by requesting routes or by forwarding unnecessary packets to the node using, for example, a black hole attack.

- **Rushing Attacks**

Rushing attacks: to target certain routing protocols that choose routes on what message arrives first a rushed malicious route message may block legitimate messages that arrive later. In fact, even the secure routing protocols were shown to be vulnerable to this particular rushing attack. In case of a typical on-demand ad hoc routing protocol, a node that intends to discover a route to a given destination floods the target network with RREQ packets. In order to keep the impact of the flood as minimal as possible, the nodes in conventional routing protocols forward only the request that arrives first from each Route Discovery. This particular mode of route discovery operation is exploited by the rushing attack.

- **Routing Table Overflow**

In this attack, the attacker attempts to create routes to nonexistent nodes. The goal is to create enough routes to prevent new routes from being created or to overwhelm the protocol implementation. An attacker can disrupt a proactive network simply by sending excessive route advertisements to the routers in the network. Reactive protocols, on the other hand, do not collect routing data in advance.

- **Cache Poisoning**

Corrupting the routing information (tables) stored locally in the nodes is referred to as cache poisoning. In route cache poisoning attacks, attackers take advantage of the promiscuous mode of routing table updating, where a node overhearing any packet may add the routing information contained in that packet header to its own route cache, even if that node is not on the path.

- **Byzantine Attack**

Byzantine attack: A compromised intermediate node works alone, or a set of compromised intermediate nodes works in collusion and carry out attacks such as creating routing loops, forwarding packets through no optimal paths, or selectively dropping packets, which results in disruption or degradation of the routing services.

- **Session Hijacking Attack**

The attacker in a session hijacking scenario exploits the unprotected session following its initial setup. The attacker forges the IP address of the victim node, computes the sequence number expected by the target, and then launches a DoS attack against the victim. By so doing, the attacker pretends to impersonate the victim node and maintains communicating with the target over the already established TCP session.

- **Network Partitioning**

A malicious node can delete routes to isolate a part of the network and render the nodes in the isolated network sector unreachable. The above is referred to as network partitioning. Statistically, detection of network partitioning is trivial. However, the malicious node can thwart detection by exhibiting low layer attacks such as channel jamming, MAC flooding.

- **Link Withholding Attacks**

In this attack, a malicious node does not advertise the information about the links to specific nodes or group of nodes, which may result in losing the links to these nodes. Mounting such attacks is difficult in the dynamic environment of MANET but is more relevant in case of sensor networks.

- **Repudiation Attacks**

In the network layer, firewalls can be installed to keep packets in or keep packets out. In the transport layer, entire connections can be encrypted, end-to-end. However, these solutions do not solve the authentication or nonrepudiation problems in general. Repudiation refers to a denial of participation in all or part of the communication.

- **Jamming**

Jamming is a well-known attack on wireless communication, it interferes with the radio frequencies that are used by nodes in a network, and which causes the message to be corrupted or lost. A small number of randomly distributed jamming nodes can disrupt the entire network, and cause all the nodes in the network to be out of service. There are many low cost devices available in the market, which can be used for jamming purpose [71]. The most common types of this form of signal jamming are random noise and pulse.

- **Cryptanalysis**

Different ciphers and cryptographic mechanisms are vulnerable to different Cryptanalysis attacks that exploit mathematical findings and shortcuts that break or decrease the security of a cipher. Cryptanalysis can either support brute-force attacks by reducing the size of the probable key-space or make key-search unnecessary by providing alternative ways of deciphering the ciphertext. A special kind of Cryptanalysis attack is the side channel attack, exploiting weaknesses in the physical implementation of the cipher.

- **Link Spoofing Attack**

In a link spoofing attack, a malicious node advertises fake links with non-neighbors to disrupt the routing operations. For example, in the OLSR protocol, an attacker can advertise a fake link with a target's two-hop neighbors. This causes the target node to select the malicious node to be its multipoint relay. As a multipoint relay node, a malicious node can then manipulate the data or routing traffic, for example, modifying or dropping the routing traffic or performing other types of DoS attacks.

3.5.4.1 Attacks on VANET

VANET being a special case of MANET, all the vulnerabilities of MANET may be considered here also. We will discuss some specific attacks on VANETs in this section [72, 73, 74, 75, 76]

- **Sybil Attacks**

Sybil attack is the creation of multiple fake nodes broadcasting false information. In Sybil attack, a vehicle install with On Board Unit (OBU) sends multiple copies of messages to other vehicle and each message contains a different fabricated identity. The problem arises when malicious vehicle is able to pretend as multiple vehicle and reinforce false data, to tell other vehicles that there is jam ahead, and force them to take alternate route. Sybil attack depends on how cheaply identities can be generated, the degree to which the system accepts inputs from entities that do not have a chain of trust linking them to a trusted entity, and whether the system treats all entities identically. This attack achieve two main goals: selfishness and denial-of-service.

- **Privacy Violation**

The inferences on driver's personal data could be made, and thus violating his or her privacy. The vulnerability lies in the periodic and frequent vehicular network traffic: Safety and traffic management messages, transaction based communications (e.g., automated payments).

- **In the Case of an Accident**

In the worst case, colluding attackers can clone each other, but this would require retrieving the security material and having full trust between the attackers. In cases where liability is involved, drivers may be tempted to cheat with some information that can determine the location of their car at a given time.

- **Hidden Vehicle**

Here fabrication happens on positioning information. It follows the basic safety messaging protocol described; a vehicle broadcasting warnings will listen for feedback from its neighbors and stop its broadcasts if it realizes that at least one of these neighbors is better positioned for warning other vehicles. The hidden vehicle attack consists in deceiving vehicle A into believing that the attacker B is better placed for forwarding the warning message, thus leading to silencing A and making it hidden to other vehicles. This ultimately stops the dissemination of the warning message, hence causing a DoS.

- **Tunnel**

Since GPS signals disappear in tunnels, an attacker may exploit this temporary loss of positioning information to inject false data once the vehicle leaves the tunnel and before it receives an authentic position update. An area jammer from the attacker, which results in the same effects, can also replace the physical tunnel in this example.

- **Sinkhole Attack**

In sinkhole attack, an intruder attracts surrounding nodes with unfaithful routing information, and then performs selective forwarding or alters the data passing through it. The attacking node tries to offer a very attractive link e.g. to a gateway. Therefore, a lot of traffic bypasses this node. Besides simple traffic analysis, other attacks like selective forwarding or denial of service that can be combined with the sinkhole attack.

3.5.4.2 Attacks on Sensor Networks

Wireless sensor networks are a subclass of wireless networks in general, so most kinds of attacks that can be directed at wireless networks can be directed at WSN. There are various ways to classify attacks on WSNs such as by attacker locations (outside and inside attacks), network layers (including attacks on physical, link, network, transportation, and application layers.), Passive or Active attack, and the purpose of the attacks. WSN breed a whole new set of attacks that can be classified into “ sensor-level attacks” and “ laptop-level attacks” [77]. In the following, we describe the various attacks a few detail as stated in [71, 77, 78, 79, 80, 81, 82, 83].

- **Collision**

An adversary may strategically cause collisions in specific packets such as ACK control messages. A possible result of such collisions is the costly exponential back off in certain media access control protocols. Whenever collision occurs, the nodes should retransmit packets affected by collision, thus leading to multiple retransmissions.

- **Selective Forwarding Attack**

Many sensor network routing protocols are based on the assumption that participating nodes will faithfully forward received packets. In a selective forwarding attack, compromised or malicious nodes may selectively forward some packets while dropping other packets. Selective forwarding attacks can take many forms. An adversary can drop packets arbitrarily, attempt to give unreasonable priority to its own messages, or misdirect traffic flows.

- **Hello Flood Attack**

It is an attack by utilizing unidirectional connections between nodes. Many protocols require nodes to broadcast Hello packets to announce themselves to their neighbors, and a node receiving such a packet may assume that it is within (normal) radio range of the sender. This assumption may be false because of the well-known unidirectional problem in ad hoc networks. An attacker may use a high-powered transmitter to trick a large area of nodes into believing they are neighbors of that transmitting node. If the attacker falsely broadcasts a superior route to the base station, all of these nodes will attempt transmission to the attacking node, despite many being out of radio range in reality. In this type of attack, all nodes will be responding to HELLO floods and wasting the energies.

- **Resource Depletion Attack**

In this type of attack, a malicious node tries to deplete resources of other nodes in a network. The typical resources that are targeted are battery power, bandwidth, and computational power. Attacks could be in the form of unnecessary requests for routes, very frequent generation of beacon packets, or forwarding of stale packets to other nodes.

- **Desynchronization**

It is a case of disruption of an existing connection. An adversary attempts to disrupt the communication between two legitimate nodes by repeatedly forging messages to these nodes. If timed correctly, an attacker may degrade or even prevent the ability of the end hosts to exchange data successfully, thus causing them instead to waste energy by attempting to recover from errors, which never really existed.

- **Node Replication Attack**

An attacker attempts to add a node to an existing WSN by replicating (i.e., copying) the node identifier of an already existing node in the network. A node replicated and joined to the network in this manner can potentially cause severe disruption in message communication in the WSN by corrupting the packets and forwarding them to wrong routes. This may also lead to network partitioning and communication of false sensor readings, it is possible to copy the cryptographic keys and use these keys for message communication from the replicated node.

- **Clock Unsynchronization**

Time synchronization is a critical building block in distributed WSN. Time unsynchronization can disrupt sleep scheduling. An attacker node can send a falsified synchronization message to its neighbor during this time exchange period. This will make other nodes calculate an incorrect phase offset and skew.

- **Injection Attack**

After the attacker has clandestinely intruded into the WSN network, he may impersonate a few of the sensor nodes (or even sink nodes) and may inject malicious data into the network. The malicious data might be false advertisement of neighbor-node information to other nodes, leading to impersonation of sink nodes and aggregation of all data.

- **Node Compromise**

Node compromise is one of the most common and detrimental attacks in WSN. As sensors can be deployed in harsh environments such as a battlefield, ocean bed, or the edge of an active volcano. This enables an adversary to steal cryptographic information, view and alter their programming, and damage or replace their hardware. Tamper-resistant packaging and camouflaging to prevent the attacker from easily locating motes are possibilities.

- **Misdirection**

By forwarding messages along wrong paths, an attacker misdirects them, perhaps by advertising false routing updates. An attacker could inflict DoS on a particular sender by diverting only traffic originating from the victim node. A receiver could likewise be denied service if the attacker diverts traffic away from the node. An attacker can also forge a source address when sending a request, so that the response will return to the victim. This could be done to confuse the victim or to flood it, if a service provides a mechanism for traffic amplification.

3.6 Security Attack Countermeasures

Attacks on wireless networks become more sophisticated, the demand for new security solutions is continually increasing. Hence, an array of new security schemes have been designed and implemented. Most of these schemes have been designed to provide solutions on a layer-by-layer basis rather than on a per-attack basis; in doing so, they have left a gap between layers that may lead to cross-layer attacks. In this section, we will explain some of the countermeasures to security attacks [5, 41, 61, 63, 71, 80, 84].

One defense technique is to use *tamper-resistant hardware* in sensor nodes. However, the cost of current tamper-resistant hardware is too high to be installed in each sensor node. Effective key management schemes can reduce the damage of node tampering. Another intrinsic deterrent to tampering is the physical distribution of the network, that is, the geographic separation of individual nodes. Protection measures outside the scope of the WSN may be sufficient to discourage tampering attacks. Another approach is to prevent detection of the nodes. Camouflaging the packaging, hiding the device, and using Low Probability of Intercept (LPI) radio techniques are among the possibilities.

Detection of a *collision* in wireless networks with one's own transmission is difficult. Error-correcting codes can be used to provide certain defense to the collision attack. However, error-correcting codes would not work if many bits received were corrupted.

One defense against *flooding attack* is to limit the number of connections that a node can request (in a period of time). Some researchers propose to use client puzzles to defend against DoS attack. Another approach describe principles for stateless connection management. This approach securely stores the server's state in all messages, requiring the client to return it with every future response. The need for protocols that create traffic asymmetries, such as area multicast, should be carefully weighed against their potential to allow traffic amplification attacks. A final strategy is to provide a way to detect the source of the flooding using a trace back mechanism. Existing schemes are IP-based and are appropriate for the Internet's scale and structure.

Using multiple disjoint routing paths and diversity coding can mitigate the effect of the *Selective Forwarding attack*. These defenses lessen the probability that a message will encounter an adversary along all routes to the destination. Diversity coding sends encoded messages along multiple paths so that the originals can be reconstructed to conceal message loss, without the cost of full duplication.

For *Misdirection attack*, untrusted adversaries should authenticate routing updates to prevent malicious modification. A freshness mechanism can protect against replay attacks, while cryptographic integrity checks protect against unauthorized modification of a message while in transit.

Because a malicious node utilizing a large transmission power to generate asymmetric links between it and other legitimate nodes, one intuitive defense against such an attack causes a *HELLO flood attack* is to verify the bi-directionality of a link between two "neighboring" nodes. Authentication is also a possible solution. Nodes can use a trusted third party to verify the authenticity of each of their neighbors before forwarding messages to them.

Impersonation threats are mitigated by applying strong authentication mechanisms. Authentication provides a party to be able to trust the origin of data it receives or stores. It usually is performed in every layer by application of digital signatures, keyed fingerprints over routing messages, different information (configuration or status), or exchanged payload data of the used services. Digital signatures and public-key cryptography requires relatively significant computation power and secure key management, which is inappropriate for wireless ad hoc network capabilities. Lighter solutions include keyed hash functions or a priori negotiated and certified keys and session identifiers.

In order to guard against *location disclosure attacks*, the technique preferred is the use of pseudo identities. However, this gives rise to another problem of maintaining and updating the list of identities that the node is using.

To prevent *trust attacks*, stronger access control mechanisms are required (Authentication, Authorization, and Accounting or AAA). In order to force the nodes and users to respect the trust hierarchy, cryptographic techniques, e.g., encryption, public key certificates, shared secrets, etc., can be employed.

There have been some proposals recently to protect networks from *wormhole attacks* by detecting such attacks. The authors introduce the concept of leashes to detect wormhole attacks. A leash is any information added to a packet in order to restrict the distance that the packet is allowed to travel. A leash is associated with each hop. Thus, each transmission of a packet requires a new leash. Two types of leashes are considered, namely geographical leashes and temporal leashes. A geographical leash is intended to limit the distance between the transmitter and the receiver of a packet. A temporal leash provides an upper bound on the lifetime of a packet. *Another approach for detecting wormhole attacks*. In this case, the authors assume the presence of directional antennae. The approach here is based on the use of packet arrival direction to detect that packets are arriving from the proper neighbors. Such information is possible due to the use of directional antennae. This information about the direction of packet arrival is expected to lead to accurate information about the set of neighbors of a node. As a result, wormhole attacks can be detected since such attacks emanate from false neighbors.

Some researchers envisaged a simple, yet effective, mechanism to prevent the resource consumption attacks (*sleep deprivation*), particularly in MANETs that use AODV as the routing protocol. In this mechanism, every node monitors and computes the respective RREQ rates of its neighbors. If the RREQ rate of a neighbor is found to exceed a threshold defined a priori, the node blacklists the neighbor and drops further RREQs from that particular neighbor.

A simple way to address the *rushing attack* is to allow for randomized selection of route request messages. Thus, every node is expected to collect a threshold number of route requests. Following this, the node can randomly choose to forward a route request from among the received requests.

Authentication of nodes is one of the major approaches against such *Byzantine attacks*. Authentication and integrity of packets are generally done using cryptographic techniques such as PKI.

To detect a *link spoofing attack*, some researchers proposed a location information-based detection method using cryptography with a Global Positioning System (GPS) and a time stamp. This approach requires each node to advertise its position obtained by the GPS and the time stamp to enable each node to obtain the location information of the other nodes. This approach detects the link spoofing by calculating the distance between the two nodes that claim to be neighbors and checking the likelihood that the link is based on a maximum transmission range. The main drawback of this approach is that it might not work in a situation where not all MANET nodes are equipped with a GPS.

In order to protect MANETs from *replay attacks*, the solution based on time stamps and asymmetric encryption. The solution simply compares the current time with the time stamp embedded in the received control messages from other nodes. If the time stamp in a received control packet deviates much from the current time, the receiving node considers it a possible replay attack.

The order to prevent *Link withholding attacks*, the authors proposed a detection technique based on the observation of both a TC message and a HELLO message generated by the Multipoint Relay (MPR) nodes. If a node does not hear a TC message from its MPR node regularly but hears only a HELLO message, a node judges that the MPR node is suspicious and can avoid the attack by selecting one or more additional MPR nodes. Similarly, some researchers has proposed an Intrusion Detection System (IDS) to detect TC link and message withholding in the OLSR protocol.

For mitigation of *Repudiation attacks* at application layer, there is a host of techniques that can be employed. In a network with a firewall installed, the firewall can provide access control, user authentication, packet filtering, and a logging and accounting service. Application layer firewalls can effectively prevent many attacks, and application-specific modules.

The most common defense against *jamming* attacks is the use of spread-spectrum communication. In frequency hopping, a device transmits a signal on a frequency for a short period of time, changes to a different frequency, and repeats. The transmitter and receiver must be coordinated. Direct sequence spreads the signal over a wide band, using a pseudorandom bit stream. A receiver must know the spreading code to distinguish the signal from noise. Frequency-hopping schemes are somewhat resistant to interference from an attacker who does not know the hopping sequence.

3.7 Cellular Network Security

This section present the security of the cellular network. In fact, the security of the cellular network is the security of each aspect of the network, that is, radio access network, core network, Internet connection, and PSTN connection [85].

3.7.1 Security in First-Generation Mobile Systems (1G)

In fact, there is no security provision in the first generation of cellular communication networks, because the first generation of cellular communications used analogue signal, which is difficult to provide security services. Hence, the security issues in cellular communications have been addressed only from the second generation of cellular communications with digitalized implementations [76].

3.7.2 Security in Second-Generation Mobile Systems (2G)

The decisions to move from an analog system to use of a digital system led to a significant improvement in the security of the system. The use of a digital system is only one of the many security provisions that were designed into the second generation [86].

3.7.2.1 Security Mechanisms in GSM

GSM is a prominent example of cellular networks. A GSM network is composed of several functional entities, with specified functions and interfaces. GSM security is composed of three classes of protection [60]:

Subscriber identity protection. For privacy issues, transmitting a subscriber identity in plain on a radio link must be avoided;

Network access control by means of SIM cards. The major functionality of the SIM is to securely hold and manage confidential information to allow the GSM network to formally identify a subscriber's identity;

Radio communication encryption between a MN and the BTS. Eavesdropping on radio *communication* being significantly easier than landline communication, it is absolutely vital to protect the radio link.

3.7.2.1.1 Security Flaws in GSM

The GSM security architecture provides a reasonable level of protection, but it has some deficiencies. One main problem with the GSM security architecture is that it provides only unilateral authentication, where the subscriber is authenticated and the visited network operator is not. This means that someone can set up a fake base station and implement a man-in-the-middle attack.

Another problem is that the GSM security architecture does not provide integrity protection services for communications and signaling over the wireless interface. Although, it is true that modifying messages on-the-fly in a wireless channel is quite challenging, if the communication between the mobile phone and the visited network takes place through a fake base station, then the attacker does not need to carry out the modifications in the wireless channel, but it can implement the attack within the fake base station. In addition, as a stream cipher is used for encryption, the attacker can easily manipulate individual bits in encrypted messages without decrypting them. Of course, if the messages carry parts of a voice communication, then the attacker can only achieve some distortion, but it is very unlikely that it can alter the true content of the communication in an unnoticeable way. It can still attack the signaling information. Moreover, besides voice communications, cellular networks are increasingly used for data communications, where flipping a single bit in a message can have devastating consequences. Additional reasons for a new design include the short length of the encryption key (practically 54 bits only), and the weaknesses discovered in the commonly used implementation of the A3 and A8 algorithms, which, under specific conditions, allow an attacker to compromise the long-term secret key of the subscriber and clone her SIM card [87].

In brief, major GSM security flaws find their origin in the lack of any form of mutual authentication, in the possible yet unfortunate plain text transmission of secrets and in cryptanalytic weaknesses of the A3, A5 and A8 ciphers. The 3GPP community has identified these flaws and provisions have been added to the UMTS standard [60].

3.7.3 Security in Third Generation Mobile System (3G)

In fact, the prominent improvements of the 3rd generation of cellular communication networks over the 2nd generation ones include the improved security algorithms, and different radio frequency ranges providing larger communication bandwidth, and improved security architecture (mutual authentication versus one-way authentication). This means that a network has to authenticate itself to the mobile users, apart from the users needing to authenticate themselves to the network [76]. The whole 3G security was designed based on three fundamental principles [88]:

- The security for 3G will build on the security features of 2G systems. Some of the robust features of 2G systems will be retained;
- The 3G security will improve on the security of the 2G systems. Some security holes and disadvantages of 2G systems will be addressed and corrected in 3Gsystems;
- 3G security will offer new features and will secure new services offered by 3G.

3.7.4 Security in Fourth Generation Mobile System (4G)

Security turns out to be one of the major problems in fourth generation (4G) of mobile networks that arise at different interfaces when trying to realize such a heterogeneous system by integrating the existing wireless and mobile systems. Indeed, current wireless systems use very different and difficult to combine proprietary security mechanisms, typically relying on the associated user and infrastructure management means. It is generally impossible to apply a security policy to a system consisting of different heterogeneous subsystems [89].

- **Vulnerabilities of Service Provider Networks**

A 4G system encompassing different technologies has to support complex management mechanisms (control systems, signaling, etc.), which considerably add to the system complexity and thus represent a major vulnerability per se. This is especially true for a multi-provider and thus multi-authority environment where a mutual preliminary user network trust does not necessarily exist and must be established by some means. The serving network protection is one of the critical points to ensure service continuity and investment in new infrastructures.

- **User Vulnerabilities**

The user device is vulnerable to attacks by other devices involved in the provision of the consumed services (impostors, data modifications, data sniffing, man-in-the-middle) and by devices consuming services provided by the user device (denial of service, abuse). Connected to multiple interfaces over several providers the device is naturally multi-homed. It is exposed potentially to all attacks over the established connections, including malicious code intrusion (viruses, spyware, and worms). A 4G user needs a particular protection to ensure his/her anonymity and an offer-consistent and verifiable billing. Without any protection, in an international multi-provider 4G environment, a user can be an easy target for both price fraud (charging wrong prices, charging incorrect usage) and user tracking.

- **Heterogeneous Security**

The security solutions proposed by the wireless technologies are limited to the identified needs. They are thus different from technology to technology reflecting its expected usage. Very often, they fail to fulfill the security requirements, typically because of conceptual or implementational flaws. However, even if their implementation is correct, their scope is naturally wrong: as access security, they aim to provide link security, but ultimately providers need service access security and users need personal data security. How can the defined security policy for the entire system be applied and enforced to all system entities given that the available solutions are different, potentially flawed, and limited to system parts?.

For instance, if the security policy identifies link encryption as a necessary confidentiality implementation, how can this be universally activated and with which keys and properties? How can we guarantee an adequate, comparable strength of the different encryption mechanisms? What to do with the technologies that do not provide link encryption? The security policy must consider these cases and provide answers to such questions.

3.8 Security in WLANs

WLANs need to support all security applications in daily work, life, and entertainment, such as downloading large volumes of confidential data, streaming High Definition (HD) video and audio in a confidential CEO meeting. A number of IEEE 802.11 standards specify security requirements of WLAN. In this section, we will explain briefly some of the security aspects of the WLAN.

3.8.1 WLAN Security Aspects

Considering that it does not stop at the physical boundaries or perimeters of a wired network, wireless communication has significant implications on the security aspects of modern networking environment. WLAN technology has, precisely for that reason, built in the following mechanisms, which are meant to enhance the level of security for wireless data communication.

3.8.1.1 Service Set Identifier (SSID)

Network access control can be implemented using an SSID associated with a single AP or a group of APs [90]. An SSID is a unique identifier of up to 32 characters that is attached to the data sent over a wireless LAN and acts as a password when a wireless LAN device tries to connect to an AP [91].

3.8.1.2 Device Authentication

The 802.11 specification provides two modes of authentication: open authentication and shared key authentication.

Open Authentication is a null authentication algorithm. It involves sending a challenge, but the AP will grant any request for authentication. It is simple and easy, mainly due to 802.11 compliancy with handheld devices that do not have the CPU capabilities required for complex authentication algorithms [92].

Shared Key Authentication is the second authentication mode specified in the 802.11 standard. This involves a shared secret key to authenticate the wireless LAN adapter to the AP. The shared−key authentication approach provides a better degree of authentication than the open system approach [91].

3.8.1.3 MAC Address Filtering

A client computer can be identified by the unique MAC address of its 802.11 network card, while an AP or group of APs can be identified by an SSID [93]. MAC address filtering provides improved security, but it is best suited to small networks where the MAC address list can be efficiently managed [90].

3.8.1.4 Wired Equivalent Privacy (WEP)

The first security scheme provided in the series of IEEE 802.11 standards is Wired Equivalent Privacy (WEP), specified as part of the 802.11b Wi-Fi standard. WEP was originally designed to provide security for WLANs [94]. The principal aim of the security implemented in 802.11 is to address the need for authentication and confidentiality. The protocol is based on a symmetric key, shared between the AP and the station. WEP aims at covering the lack of physical security akin to WLANs with security mechanisms based on cryptography. WEP suffers from various design flaws and some exposure in the underlying cryptographic techniques that seriously undermine its security claims [31].

3.8.1.4.1 WEP's Security Problems

Researchers have found several security flaws in the WEP that severely undermine its encryption and authentication capabilities. We can say the WEP protocol presents some significant weaknesses [95].

- Lack of Proper Integrity Protection: WEP uses CRC-32 for integrity protection. CRC-32 is not a keyed integrity algorithm and is highly susceptible to collisions, and thus is ineffective as an integrity algorithm.
- WEP's use of RC4 has several serious flaws: In summary, RC4 itself has some vulnerabilities that an attacker might exploit, and WEP's design of reusing the key with an IV increases the potential for related keys. The 802.11 protocols also provide opportunities for the attacker to gain access to cleartext and the corresponding ciphertext.
- Lack of Replay Protection: WEP does not support replay protection. Thus even if confidentiality and message integrity were effective, an adversary can replay previously sent packets. The receiver would have no way of telling that the packet is legitimate or a replay.
- Lack of Mutual Authentication and Key Management: The shared-key authentication mechanism only allows the responder to authenticate the initiator, but not vice versa. It

does not define any specifications for key management and merely recommends the shared use of group keys. The distribution of shared group keys are difficult, if not impossible, to control, and the fact that any member of a group can pretend to be another member of the group [96].

3.8.1.5 802.1x

Because of the security gaps in the WEP encryption specification, which previously mentioned, the industry developed authentication methods based on 802.1x specification, which was originally designed for wired networks [97]. It is a standard for port-based network access control, and offers an effective framework for authenticating and controlling user traffic, and for keys periodical refresh [98]. After successful authentication, a virtual port is opened on the access point for network access, while communications are blocked if authentication fails [99]. Many wireless network equipment manufacturers and software developers have adopted yet another IEEE standard, 802.1x, to add another layer of security to their networks [100].

The advantages of 802.1x authentication includes the following (1) every user in the network can be identified and authenticated; (2) it supports extensible authentication technologies, such as token cards, certificate/smart cards, and one-time passwords; and (3) it supports key managements, including key management and key reproduction.

However, even though 802.1x was aimed at improving user authentication of the original 802.11 security, some researchers are found flaws, which an attacker could launch several attacks, such as session hijacking and man-in-the-middle attacks. These security problems underline the weaknesses within the 802.1x design [97].

3.8.1.6 WPA

The Wi-Fi Alliance developed Wi-Fi Protected Access (WPA) as a means to provide enhanced protection from targeted attacks, which is based on those components of the 802.11i standard that are stable and may be deployed on existing 802.11 network and client equipment with a software upgrade [90]. Two optional authentication mechanisms are offered to WPA users according to different scenarios, 802.1x authentication framework together with extensible authentication protocol (EAP) for enterprise WLAN security (Enterprise mode), or simpler pre-shared key (PSK) authentication for the home or small office network which does not have an authentication server (Consumer mode).

WPA uses Temporal Key Integrity Protocol (TKIP). TKIP scrambles the keys using a hashing algorithm and adds an integrity-checking feature that verifies that the keys haven't been tampered with. WPA improves on WEP by increasing the IV from 24 bits to 48. Rollover has also been eliminated, which means key reuse is less likely to occur. WPA also avoids another weakness of WEP by using a different secret key for each packet [101]. Besides the extended size of keys, some other methods are applied to generate keys dynamically and prevent the repetition of the same traffic keys, which is another critical issue to improve the security level. Another improvement in WPA is message integrity. WPA addressed a Message Integrity Check (MIC) that is known as Michael. Michael is designed to detect invalid packets and can even take measures to prevent attacks. A complete picture of WPA is depicted, involving the mechanisms of confidentiality, authentication, and integrity [102].

3.8.1.6.1 Weaknesses in WPA

Although WPA is much more secure than WEP, it suffers from some drawbacks [102]. Since the security depends on the secrecy of all the packet keys, if a packet key is exposed to the attacker, the MIC key can be found easily. Furthermore, if two packet keys with the same IV are disclosed, the attacker can do anything in the duration of the current temporal key, though the temporal key has its own lifetime and is replaced frequently.

Besides the problem discussed above, in PSK, the administrator is allowed to specify a password to be known by all users for access to the AP. It is obvious that this method is vulnerable to an offline dictionary attack. In addition, DoS attack is also effectual for WPA. The malicious users can launch this attack only by initiate the access requests with a short interval.

3.8.1.7 IEEE 802.11i

The IEEE 802.11i known as WPA2, is an additional specification to provide enhanced WLANs' security, WPA2 defines data confidentiality, mutual authentication, and key management protocols [76].

The main advantages of the WPA2 standard can be listed as follows:

- Providing more excellent security by using advanced encryption algorithms;
- Using stronger key management policies;
- Protecting against the man-in-the-middle attacks by using the two-way authentication process;

- Providing improved message integrity by using Cipher Block Chaining Message Authentication Code.

The comparison of WEP, WPA, and WPA2 can be summarized in the following table.

Security Protocol	**WEP**	**WPA**	**WPA2**
Major Component	IV	TKIP	CCMP
Stream Cipher	RC4	RC4	AES
Key Size	40 bit	128 bit (encryption) and 64 bit (authentication)	128 bit
IV Size	24 bit	48 bit	48 bit
Key Management	Not Available	IEEE 802.1x/EAP	IEEE 802.1x/EAP/CCMP
Data Integrity	CRC-32	MIC	CBC-MAC

Table 3.1: The comparison of WLAN Security Protocols

3.8.1.7.1 Weakness of IEEE 802.11i

Although WPA2 is designed to cover up for the weaknesses of WEP, it still has its own drawbacks [102] [76]. First, WPA2 is costly. Due to the requirements of the implementation of the advanced properties designed in WPA2 (the change in devices is meaningful).

Second, WPA2 is suffers from other new types of DoS attack derived from some particular features, such as reflection attack for 4-Way Handshake authentication, RSN Information Element Poisoning and 4-Way Handshake Blocking, and so forth.

Third, WPA2 is also prone to attacks such as security level rollback attack, reflection attack, and Time Memory Trade Off (TMTO) attack. Specifically, when Pre-RSNA and RSNA algorithms are both used in a single WLAN, an adversary can launch a security level rollback attack, avoiding authentication and disclosing the default keys.

3.8.1.8 VPNs

Virtual private networks (VPNs) are typically used in TCP/IP−based networks to secure communication between remote users and a private network. Using VPN to establish such connectivity guarantees that the remote user is authenticated and all data over the Internet is transmitted in encrypted form [91].

The same VPN technologies can also be used for secure wireless access in WLANs. In this scenario, the untrusted network is the wireless network. The APs are now configured for open access with no WEP encryption, but wireless access is isolated from the enterprise network by the VPN server. The APs can be connected together via a virtual LAN or LAN that is deployed in the Demilitarized Zone and connected to the VPN server. The APs should still be configured in closed mode with SSIDs for network segmentation. Authentication and full encryption over the wireless network is provided through VPN servers that also act as firewalls and gateways to the internal private network [90].

The VPN approach has a number of advantages [90]. First, already deployed on many enterprise networks. Second, scalable to a large number of 802.11 clients. Thirdly, low administration requirements for 802.11 APs and clients. Fourthly, traffic to the internal network is isolated until VPN authentication is performed, WEP key and MAC address list management is not needed because of security measures created by the VPN channel itself. Finally, addresses general remote access with a consistent user interface in different locations such as at home, at work, and in an airport.

A drawback to current VPN solutions is the lack of support for multicasting, which is a technique used to deliver data efficiently in real time from one source to many users over a network. Although the standard of 802.11i can guarantee the same security requirements as the wireless VPNs, the vulnerabilities in the implementations of the 802.11i standard could still make it less trustworthy. Another issue related to roaming between wireless networks is not completely transparent. Users receive a log-on dialog when roaming between VPN servers on a network or when the client system resumes from standby mode. In addition, in the case of point-to-point wireless links it is easier and more economical to deploy a network-to-network VPN than 802.11i-based defenses, including the RADIUS server and user credentials database, while using 802.11i with PSK and no 802.11x is not a good security solution for a high throughput network-to-network link. When using a VPN solution, it is still recommended that client computers be equipped with personal firewall protection to provide increased security, including the protection against attacks by nearby wireless client systems [90],[76].

3.8.1.9 Public Key Infrastructure (PKI)

PKI (Public Key Infrastructure), simply defined, is an infrastructure that allows the creation of a trusted method for providing privacy, authentication, integrity, and nonrepudiation in online communications between two parties [103]. For applications requiring higher levels of security, WLANs can integrate PKI for authentication to secure network transactions because it provides strong authentication through user certificates.

Furthermore, the user can use the same certificates in application-level security, such as signing and encrypting messages. Wireless PKI, handsets, and smart cards that integrate with WLANs are available from third-party manufactures [54].

3.8.1.10 Smart Card

A smart card is a portable and tamper-resistant computer. It provides data security, data integrity, and personal privacy and supports mobility [89]. In wireless networks, smart cards provide the added feature of authentication, although they also add another layer of complexity. These devices are beneficial in situations that require authentication beyond simple usernames and passwords. Furthermore, they are portable, and hence users can securely access their networks from various locations. Organizations can use smart cards in a two-factor authentication by combining it with biometrics [54].

3.8.1.11 Biometrics

Biometrics uses methods for unique recognition of humans based upon one or more intrinsic physical or behavioral traits. In computer science, particularly, biometrics is used as a form of identity access management and access control. It is also used to identify individuals in groups that are under surveillance [104].

For higher levels of security, biometrics can be integrated with wireless laptops, wireless smart card, or other wireless devices and used to authenticate the user to access the wireless network. The advantage of modern biometric technology is that it is very convenient and provides for higher security than most other forms of authentication [103]. Biometric authentication combined with encryption is being used to improve the security of encryption systems. To make brute force attacks obsolete, a biometric key with a person’s unique personal identification can be added to or can replace the normal encryption key. Biometric encryption also makes key management unnecessary because the encryption key becomes a unique physical characteristic of a person and is hard to break [54].

3.8.1.12 Firewalls

A firewall can be defined as a collection of components (hardware and/or software) that is placed between two networks. The following properties exist, all traffic in either direction must pass through the firewall, only traffic authorized by the local security policy will be allowed to pass, the firewall itself is immune to penetration [105]. Implementing personal firewall software on client computers can provide some protection against attacks, especially for clients accessing public WLANs. Organizations can set up these personal firewalls to be centrally or individually managed [47]. In effect, a firewall divides a network into a more-trusted zone internal to the firewall, and a less-trusted zone external to the firewall. This is useful if you do not want external users to access a particular host or service within your site. A firewall may also impose restrictions on outgoing traffic, to prevent certain attacks and to limit losses if an adversary succeeds in getting access inside the firewall. Firewalls may be used to create multiple zones of trust, such as a hierarchy of increasingly trusted zones. A common arrangement involves three zones of trust: the internal network, the demilitarized zone (DMZ), and the rest of the Internet [106].

3.8.1.12.1 Weaknesses of Firewalls

According to [107], the weaknesses of a software firewall are, may slow down system applications since it is installed on the system itself and requires more memory and disk space, may also prove costly because such a firewall has to be purchased separately for each computer on the network. It maybe unwieldy to remove from the system, such firewalls cannot be configured to mask IP addresses (they only close unused ports and monitor traffic to and from open ports), and may not be capable of fast reaction. While the weaknesses of a hardware firewall are, they treat outgoing traffic as safe and may fail if a malware is attempting to connect to the internet from within, they may be more complex to configure, they are more expensive, and takes up more physical space with its added wiring.

3.8.1.13 Wireless IDS

An intrusion detection system (IDS) is a device or software attempting to perform network intrusion detection and stop possible incidents/ attacks by gathering and analyzing data. These wireless IDSs can recognize patterns of known attacks, identify abnormal network activity, and detect policy violations for WLANs by monitoring and analyzing network, user, and system activities. Also, like traditional signature based IDSs and anomaly-based IDSs, wireless IDSs can generate intrusion alters according to either the predefined signatures or the observed abnormal network behavior [76].

3.9 Security in WPAN

Although a wireless PAN will generally, have a more limited range than a WLAN, ensuring security will remain an important implementation issue, and a number of challenges exist that make this task difficult.

First, access to the wireless medium is open to all, including potential adversaries that don’t even have to be physically close to the sensor field. Second, the network may consist of many nodes, which are expected to operate with little human intervention for prolonged periods of time. Furthermore, some or all of the nodes may be mobile. As a result, attacks and disruptions more difficult to detect. Third, most of the nodes in a WPAN or a sensor network have limited energy at their disposal and the chips computational power is limited; this restricts the choice of cryptographic techniques that can be applied to ensure that privacy and integrity are adequately supported. Finally, actual nodes may be subject to damage, or even physical capture and subsequent subversion by a hostile adversary.

Consequently, sophisticated techniques are required to monitor and detect possible intrusions and, if necessary, launch appropriate countermeasures. From the networking perspective, security threats may occur at different layers of the ISO/OSI model [108]. Bluetooth, ZigBee and NFC (Near Field Communications) have emerged as the key WPAN technologies. They are however subject to the usual range of security vulnerabilities found in wireless LANs.

3.10 Wireless MAN Security

Wireless Metropolitan Area Networks (WMANs) provide wireless communications at acceptable bandwidth over much larger geographical areas compared to WLANs. WMANs use WiMAX technologies to provide Mobile Stations (MS) communications with Base Stations connected to backbone networks and the Internet. Also known as, the “last mile” technology, WiMAX was designed and developed to have relatively long communication range, that fits wonderfully in urban areas. WMANs support all secure applications that can run over the Internet [76].

3.10.1 Security in WiMAX

Given the fact that WiMAX is intended for wide area coverage, reliable security features and many complex security mechanisms are adopted for authentication and confidential data transfer [4]. WiMax security has two goals; one is to provide privacy across the wireless network and the other is to provide access control to the network. Privacy is accomplished by encrypting connections between the subscriber station and the base station. The base station protects against unauthorized access by enforcing encryption of service flows across the network. A Privacy and Key Management (PKM) protocol is used by the base station to control the distribution of keying data to subscriber stations. This allows the subscriber and base stations to synchronize keying data. Digital-certificate-based subscriber station authentication is included in the PKM to provide access control [109].

3.11 Wireless WAN Security

Some of the problems related to Wireless WAN security are a result of inherent vulnerabilities in the TCP/IP protocols and services, while others are a result of host configuration and access controls that are poorly implemented or too complex to administer. Additionally, the role and importance of system administration is often shortchanged in job descriptions, resulting in many administrators' being, at best, part-time and poorly prepared. WWAN security also presents a variety of challenges depending on the access networks used; and when user connect via different means such as a mixture of Wireless LANs and Cellular data networks, the challenges compound [110].

3.12 Security in Wireless Mesh Networks

The potential of wireless mesh networking cannot be exploited without considering and adequately addressing the involved security issues. The nodes within a wireless mesh network function as routers relaying packets to other nodes. The number of nodes in a wireless mesh network increases, therefore more locations where insidious persons can view the data. In addition, if software permits nodes to be added without centralized control, a mechanism is required to ensure the node is legitimate and not a PC operated by a hacker [111].

3.12.1 Why Security is Important in WMNs

Security plays a critical role in wireless networks. In WMNs, security becomes even more critical, for the following reasons [112].

- Multihop wireless network security: Many security schemes for wireless networks are focused on one-hop communications. The multihop architecture renders these schemes insufficient to protect a WMN from being attacked.
- Multitier security: In WMNs, security is needed for wireless access from mesh clients to mesh routers and also for wireless connectivity among mesh routers. Mesh routers usually belong to a service provider, while mesh clients can be any users. Such features make the security issue different from that in any other wireless network such as wireless LANs or mobile ad hoc networks. The security mechanism for communications among mesh routers must be different from that in the wireless access part.

- Multisystem security: For the benefits of better wireless services, WMNs usually involve interoperation of multiple wireless networks such as IEEE 802.11, IEEE 802.16, IEEE 802.15 based wireless networks. Both security architecture and schemes are much different from one system to another.

3.12.2 WMN Specific Security Challenges

- The shared nature of wireless medium, the absence of globally trusted central controller, and the lack of physical protection of mesh routers pose the main challenges for securing WMNs [113].
- An authentication mechanism is usually implemented with the help of Public Key Infrastructure (PKI), which requires a globally trusted entity to issue certificates. However, it is impractical to maintain a globally trusted entity in WMNs.
- The mesh routers are located outdoor, usually on rooftops or traffic light poles. Therefore, it is much easier for attackers to capture the mesh routers and get full control of the device. If a router is fully controlled by attackers, the attacks can be launched from that router and the information sent by the attackers will be regarded as authenticated by other routers.

Other specific challenges for WMNs include [38]:

- WMN may be dynamic because of changes in both its topology and its membership. Any security with a static configuration would not suffice.
- Mesh routers and mesh clients hold significantly different characteristics such as mobility and power constraints. As a result, the same security solution may not work for both mesh routers and mesh clients.
- There are also issues introduced by MN (Mesh Node) belonging to different authorities, such as selfish and greedy behavior, and trust management.

3.13 Security in Ad Hoc Networks

Compared with infrastructure-based wireless networks, security management for wireless ad hoc networks is more challenging due to unreliable communication, intermittent connection, node mobility, and dynamic topology [40]. As a result, security design in ad hoc mobile networks has to face the lack of clear line of defense. Each node in an ad hoc network may function as a router and forward packets for other peer nodes, unlike wired networks that have dedicated routers. The wireless channel is accessible to both legitimate network users and malicious attackers. There is no well-defined place where traffic monitoring or access control mechanisms can be deployed. This makes the separation of inside from outside network domain obscure [114].

As all nodes are expected to cooperate, no a priori classification or security association can be made, and nodes are free to form independent sub-domains. An additional problem with compromised nodes is the potential Byzantine failures wherein a set of nodes could be compromised such that innocent and malicious behavior cannot be distinguished. Malicious nodes can advertise nonexistent links, provide incorrect link state information, create new routing messages, and flood other nodes with routing traffic, thus causing Byzantine failures on the system. The wireless links between nodes are highly susceptible to link attacks, which include active interfering, leakage of secret information, eavesdropping, data tampering, impersonation, message replay, message distortion, and denial-of-service (DoS). The presence of even a small number of malicious nodes could result in repeatedly compromised routes. As a result, the network nodes would have to rely on cycles of timeout and new route discoveries to communicate [115].

Moreover, different applications have different security requirements. The complexity and diversity of the field has led to a multitude of proposals, which focus on different parts of the problem domain. They vary between trust and key management, secure routing and intrusion detection, availability and cryptographic protocols [41]. A complete security solution should include three components of prevention, detection, and reaction. It must provide security properties of authentication, confidentiality, non-repudiation, integrity, and availability. It should be adaptive in order to trade-off service performance and security performance under resource limitation [40].

3.13.1 Security Challenges

Security is an important issue for ad hoc networks, especially for those in security-sensitive environments. The unique characteristics of ad hoc networks have posed nontrivial challenges to security designs.

- Resource Constraints: The wireless devices usually have limited bandwidth, memory and processing power [40]. For this reasons, security mechanisms for ad hoc networks must be lightweight in terms of communication overhead, computation complexity, and storage overhead [89].

- Open Shared Medium: Use of open shared medium makes an ad hoc network susceptible to attacks such as eavesdropping, signal jamming, impersonation, message distortion, message injection, and cause other problems [89].
- Absence of infrastructure, frequent change of topology and node mobility: Ad hoc networks operate independently of any infrastructure, which makes inapplicable any classical solutions based on certification authorities and on-line servers [41]. The connectivity among the nodes may vary with time due to node departures, node arrivals, and the mobility of nodes. This emphasizes the need for secure solutions to be adaptive to dynamic topology [40].
- Need to cooperate: In the absence of a router, each participant may have to relay packets to the other network nodes. Consequently, if one of these participants decides either with a "selfish" behavior or with a voluntarily malicious goal, not to relay the packets, it is the network operation, which is affected, and its effectiveness is reduced [60].
- Scalability: Due to the limited memory and processing power on mobile devices, the scalability is a key problem when we consider a large network size [40].
- Auto-configuration: Auto-configuration seems to be an essential functionality, since it enables the integration of nodes into a network without requiring human intervention. This mechanism constitutes a target choice for malicious nodes [60].

3.13.2 Security in Vehicular Ad Hoc Networks (VANETs)

A modern vehicle can be considered as a network of sensors/actuators on wheels. VANET is a special kind of Mobile Ad-hoc Network (MANET) where vehicles equipped with the technologies are the key constituents [74]. VANETs constitute all types of ad hoc networks formed by the use of short-range radios installed in private (personal consumer) and public (public transport and law enforcement authorities) vehicles [61]. Without security, a VANET system is vulnerable to a number of attacks such as propagation of false warning messages and suppression of actual warning messages, thereby causing accidents. This makes security a factor of paramount importance in building such networks.

3.13.3 Security in WSN

Security is very important for many sensor network applications, such as military target tracking and security monitoring. Providing security to small sensor nodes is challenging, due to the limited capabilities of sensor nodes in terms of computation, communication, and energy.

3.13.3.1 Security Goals

Security goals in sensor networks depend on the need to know what we are going to protect. In addition to the security goals referred to in section 3.4 such as, Authentication, Confidentiality, Integrity, Availability, Non-repudiation. We determine other security goals in sensor networks: Scalability, Privacy, Flexibility, Battery, life Transmission range, Bandwidth, Memory, Prior deployment knowledge, Resistance, Revocation, and Resilience [116].

- **Scalability:** Efficiency demands that sensor networks utilize a scalable secure technique to allow for the variations in size typical of such a network.
- **Privacy:** Privacy is one of the key primitives for securing a sensor network in terms of disclosure of identity. To ensure privacy of the nodes, an anonymous communication protocol can be an effective tool for communication in the network without disclosing the IDs of the nodes.
- **Flexibility:** Secure techniques should be able to function well in any kind of environments and support dynamic deployment of nodes, i.e., the techniques should be useful in multiple applications and allow for adding nodes at any time. One of the challenges in developing sensor networks is to provide high-security features with limited resources.
- **Prior deployment knowledge:** As the nodes in sensor, networks are deployed randomly and dynamically, it is not possible to maintain the knowledge of every placement. A secure protocol should not, therefore, be aware of where nodes are deployed when initializing keys in the network.
- **Resistance:** An adversary might attack the network by compromising a few nodes in the network and then replicating those nodes back into the network. A secure protocol must resist node replication to guard against such attacks.
- **Revocation:** If an adversary invades a sensor network, the secure protocol should provide an efficient way to revoke compromised nodes, a lightweight method that does not use much of the network’s already limited capacity for communication.
- **Resilience:** If a node within a sensor network is captured, the secure protocol should ensure that the secret information about other nodes is not revealed. Resilience also means conveniently making new inserted sensors to join secure communications.

3.13.3.2 Security Considerations in Wireless Sensor Networks

Can security measures and cryptographic protocols in wireless sensor networks be considered in the same way as for other types of networks? There is some consensus that the answer seems to be “no”, because of some special features or considerations of sensor networks make it particularly challenging to provide these security services for sensor networks [117].

- The network infrastructure of a WSN is made up of small, cheap nodes may be deployed in public or hostile locations (such as public buildings or forward battle areas) in many applications. Furthermore, the large number of nodes that are deployed implies that each sensor node must be low cost, which makes it difficult for manufacturers to make them tamper resistant. Special secure memory devices would be needed to prevent the attacker from reading the memory. Moreover, heterogeneous nature of sensor nodes is an additional limitation that prevents one security solution [118].
- Traditional security mechanisms that have high overheads are not suitable for resource-constrained WSNs. Many security mechanisms are computationally expensive or require communication with other nodes or “remote” devices (e.g., for authorization purposes), thereby leading to energy overheads. Small sensor devices are also constrained in their available memory and storage capacities [83].
- In addition to the limitations of nodes, sensor networks provide all the limitations of a mobile ad hoc network. Security solutions should be decentralized and nodes must collaborate to achieve security [83].
- When in-network processing is to be performed, intermediate nodes need to access and modify the information contained in packets; hence, a larger number of parties is involved in end-to-end information transfers [117].
- The finite energy budget of sensor nodes opens up a particularly attractive line of attacks: to force victim sensor nodes to exhaust their energy budget quickly and to die. An additional challenge is that attackers can have much more energy at their disposal than the sensor nodes [117].

3.14 Security Mechanisms

A variety of security mechanisms such as protocols, algorithms and key management has been invented to counter malicious attacks. A first line of defense (traditional security mechanisms) include authentication, access control, encryption, and digital signature. As a second line of defense, intrusion detection systems and cooperation enforcement mechanisms implemented in MANET can also help to defend against attacks or enforce cooperation, reducing selfish node behavior. This section gives an overview of basic concepts concerning security mechanisms [67], [41].

- **Preventive mechanism**: The conventional authentication and encryption schemes are based on cryptography, which includes asymmetric and symmetric cryptography.
- **Reactive mechanism:** A number of malicious attacks could bypass the preventive mechanisms due to its design, implementation, or restrictions. An intrusion detection system provides a second line of defense. There are widely used to detect misuse and anomalies. In practice, both approaches can be combined to be more effective against attacks.

3.14.1 Cryptographic Issues

Preventive security controls are often protocols that utilize cryptography. Cryptography analyzes and develops methods for transforming of unsecured plaintext into ciphertext that can not be read by unauthorized entities. There are two main applications of cryptography: data encryption and data signing.

3.14.2 Cryptographic Primitives

Three types of cryptographic primitives are used in order to authenticate the content of messages exchanged among nodes: Keyed-Hash Message Authentication Code (HMAC), digital signature, and one-way HMAC key chain.

HMAC, it is applicable in case when nodes share a secret symmetric key which allow them to generate and verify a message authenticator h_k (.), using a cryptographic one-way function.

Digital signature is based on asymmetric key cryptography and involves much more computational overhead in signing/decrypting and verifying/encrypting operations. It is sensitive to DoS attacks because of possibility of bogus signatures.

One-way HMAC key chain provides a cryptographic one-way function $f(x)$, designed to make the input x invisible. When applied repeatedly on the input, a chain of outputs $f^i(x)$ is obtained. The reverse order of generation is used to authenticate messages. A message with an HMAC using $f^i(x)$ as the key is proven to be authentic when the sender reveals $f^{i-1}(x)$.

3.14.3 Cryptographic Algorithms

Cryptographic algorithms perform a mathematical transformation of input data (e.g., data, keys) to output data to conceal it. They may use one or both mentioned security applications and have to be embedded into a semantic context, which usually occurs as a part of a cryptographic protocol. A cryptographic protocol is a procedural instruction for a series of processing steps and message exchanges between multiple entities, aiming to achieve specific security objectives. Cryptographic algorithms can be classified according to the number of used different keys into:

- Hash algorithms, use no key;
- Secret-key cryptography, use one key (symmetric algorithms);
- Public-key cryptography, use two different keys for encryption and decryption or signing and signature check (asymmetric algorithms).

3.14.4 Key Management

Keys are an essential component of security because they allow us to read otherwise unintelligible messages and to sign documents, among other things. Cryptographic protocols use keys to authenticate entities and grant access to guarded information to those who exhibit their knowledge of the keys. Therefore, it is imperative that keys be securely generated and distributed to appropriate entities.

Secret keys are shared between communicating entities. A secret key can be generated by one party and distributed to another entity, through either direct physical contact or a secure channel. The key can also be negotiated among entities, in which case key generation and distribution are accomplished simultaneously.

In public-key cryptography, a public key is made public, while the corresponding private key is kept secret. A public-key certificate certifies the binding between a public key and an entity. Certificates are signed bindings by a trusted party whose public key is known beforehand. Public-key certificates can be generated and distributed through a central server (similar to publishing phone numbers in a phone book) or a network of nodes that provides such services (similar to distributing cell phone numbers by the word of mouth), or a combination of the two. Public-key cryptography is often used to distribute secret keys [5].

3.14.5 Intrusion Detection Systems (IDS)

In fact, the intrusion defined as any set of actions that attempt to compromise the integrity, confidentiality, or availability of a resource [40] and Intrusion detection can be defined as the automated detection and subsequent generation of an alarm if an intrusion is taking place. An IDS is a defense mechanism that continuously monitors the network for unusual activity and detects adverse activities [115].

According to [115] three intrusion detection techniques are used: anomaly detection, signature or misuse detection, and specification-based detection.

- The first technique is anomaly-based intrusion detection, which profiles the symptoms of normal behaviors of the system such as usage frequency of commands, CPU usage for programs and the like. It detects intrusions as anomalies, i.e. deviations from the normal behaviors [40].
- In misuse detection, decisions are made based on an intrusive process by defining legal or illegal behavior on the basis of observed behavior. The misuse detection technique involves analyzing the collected data for specific behavior patterns known to be consistent with specific attacks. These behavior patterns are called signatures [63].
- The last technique is specification-based intrusion detection. In this approach, assumes the existence of a precise protocol specification. Malicious behavior is detected by comparing the protocol traffic with the protocol specification. The detectors typically build precise models of expected behavior (e.g. by using state machines) based on the protocol specifications and then compare the observed behavior in the network against the model [63].

3.14.5.1 Architectures for IDS in Wireless Ad Hoc Networks

As we know, the nature of wireless ad hoc networks makes them very vulnerable to attack. To tackle these challenges, several possible IDS architectures exist including standalone IDS, distributed and cooperative IDS and hierarchical IDS [69].

3.14.5.1.1 Standalone IDS

Each node has its own IDS and detects attacks independently in this architecture. There is no cooperation between nodes and all decisions are based on information collected by individual nodes.

3.14.5.1.2 Distributed and Cooperative IDS

In this architecture, each node has an IDS agent and makes local detection decisions by itself. At the same time, all the nodes participate in a global detection process. This architecture is more suitable for a flat network configuration than a cluster-based multilayered one.

3.14.5.1.3 Hierarchical IDS

In this architecture, each node has its own IDS agent responsible for local intrusion detection. At the same time, the IDS agent of the cluster head is responsible for both local and global intrusion detection. Total network coverage is assured by activating global agents in every cluster head. However, the clustering also adds possible points of attack and overhead and complexity in the creation and maintenance of clusters.

3.14.5.1.4 Mobile Agent for IDS

Mobile-agent-based IDS can be considered either a distributed and cooperative intrusion detection technique or it can be used in combination with hierarchical IDS. An agent is mobile due to its ability to move through the network, interact with nodes, and collect information from them. The intrusion-detection tasks are distributed and assigned to these mobile agents. Each mobile agent is assigned a specific task and acts upon the information it collects along its moving path.

There are many advantages of using mobile agents. First, power consumption of the network is reduced because the tasks are distributed and each node holds only some of the tasks and not all of them. Secondly, the overall system fault tolerance increases because the IDS tasks are distributed to different parts of the network; when some agents are destroyed or parts of the network are separated, the other agents can remain functional. Third, as the mobile agent may be platform independent, the IDS can run under different operating system environments. Furthermore, when distributed mobile agents replace a central processing unit, the computational load is divided between machines and the network load is reduced. However, these mobile agents still need to be run in a secure module on each node in order to protect themselves on remote hosts.

3.14.5.2 Architecture for Intrusion Detection in MANET

In order to create an architecture there are some fundamental questions that need to be considered. A key question that will have to be answered first is related to the roles that a node can have in a MANET intrusion detection architecture. These roles include the following [63]:

- Self-Detection: A node may run a detector that focuses on monitoring the node itself to see whether it is behaving as expected. It could do this for example by monitoring the messages that the node itself is sending to other nodes. Since this detector has perfect knowledge of the state of the node, it can detect malicious behavior without false positives.
- Local Detection: A node may run a detector that detects attacks based on evidence available locally. This may include evidence from packets received by the node, forwarded by the node (as part of the routing process), or packets that have been observed by the node going through the wireless link (by eavesdropping on the link).
- Data Collection: Since a number of attacks in a MANET environment cannot be detected locally with a high degree of certainty by a single node, it is necessary for nodes to collect intrusion detection evidence and share it with other nodes. This data is then shared either with everybody or with some subset of the nodes, depending on the specific IDS architecture used.

The next key question is related to the number of nodes that need to be part of the intrusion detection architecture as well as the role of each. If our goal is to detect all the attacks for all possible mobility scenarios. If we can utilize only a subset of the total number of nodes in the intrusion detection process, it is important to decide what role each node needs to play in the intrusion detection process. There is no general answer to this question that is well suited for all situations. The answer depends on several parameters, such as:

- The degree of certainty required for the detection. If we need to detect intrusions under almost all conditions with minimal false positives then a very large percentage of nodes will need to execute the IDS.
- The mobility scenario expected for the specific application and mission. For example, in a military environment, units tend to move together and be close to each other for a long period of time. In such cases a small number of nodes running IDS placed within a unit may be able to detect most intrusions.
- Mode of monitoring. If nodes are capable of monitoring in the promiscuous mode whereby each node can receive and analyze any packet transmitted by any of its neighbors, then each node can monitor a larger portion of the traffic. If this is not

possible, then a node might be restricted to monitoring traffic that it relays. In this case, each node can monitor a relatively smaller portion of the traffic.

- The connectivity environment. In a flat area where there are few obstructions, most nodes usually have multiple neighbors that can observe their behavior. In that case, a small percentage of nodes may have visibility of most of the traffic and therefore be able to detect most attacks.
- The capabilities of nodes. It may only be possible for a small subset of the nodes to run IDS due to limitations on resources such as power, CPU processing, storage, and bandwidth. In that case, there may not be a choice and the nodes that have available resources and the necessary capabilities will have to execute the IDS functions.
- Number of nodes that need protection. It may be critical to protect only a small subset of the nodes that either store important data or run critical servers. The specific mission or application may be willing to tolerate attacks against a number of other nodes. In that case, it is probably more important to place nodes running IDS around the important nodes. A few additional nodes running IDS scattered throughout the rest of the network might then suffice.
- Percentage of compromised nodes that can be tolerated. Certain applications can continue functioning at an acceptable level even after some percentage of the nodes has been compromised. If that is the case then a smaller percentage of IDS nodes may be sufficient to ensure that no more than the acceptable percentage of nodes has been compromised.

3.15 Conclusion

In fact, to obtain a security in wireless networks equivalent to that provided by wired networks is a very hard task. Many constraints must be overcome in order to benefit from an ad hoc network: access to the radio channel, mobility and energy management, etc.

Although the covered topic may not be an exhaustive representation of all the security issues in wireless networks, But we have tried to give an overview of security in wireless networks with more detail with regard to ad hoc networks security. we have explained the security goals, types of attacks, security attack countermeasures, aspects and mechanisms of security, etc. in general, we do represent a rich and useful sample of the strategy and content within wireless network security. In the next chapter, we will present the mobile agent paradigm and the contribution of it to ad hoc networks security.

Chapter 4

Mobile Agent Security

And trust no agent; for beauty is a witch against
whose charms faith melteth into blood.

W. Shakespeare (deliberately taken out of context)

This chapter gives an overview about the concept and the security issues related to the mobile agent paradigm such as security threats and requirements. It gives the main solutions for keeping a mobile agent platform secure against a malicious mobile agent. Similarly, it presents a set of solutions for ensuring the security of mobile agents against illegitimate platforms. We finished the chapter by reviewing and discussing some relevant previous research efforts for the security in ad hoc networks based mobile agent.

4.1 Introduction

The mobile agent has received considerable attention in recent years for its wide applications in various areas of computing technology. This has led to deal more efficiently and elegantly with the dynamic, heterogeneous, and open environment, which is today's wireless network.

A mobile agent is an active entity that can act within a distributed system of agent places on behalf of its user, following a given task. A place provides an abstract representation of a host and its services. The agent can autonomously migrate from one place in the network to another during its execution. While it computes, it is able to observe its environment and to adapt dynamically to changes. It can continue its computations asynchronously even if the user that has started it is (temporarily) not connected to the distributed system any more, i.e. the mobile agent paradigm is able to support mobile computing quite naturally. By moving the agent to the host on which data resides, communication latency may be reduced in many cases. Furthermore, by processing the data and sending only the relevant results, the consumption of bandwidth and/or the connection time can be reduced, which is again an advantage for mobile computing [119].

Although mobile agent technology extends the capabilities of traditional distributed network applications such as the client server model, there is an increase in the security requirements. Mobile agent–based systems are subject to several security threats. Indeed, since mobile agents migrate through open and insecure networks and are executed on hosts of uncertain trust, security is a major concern [64].

4.2 Overview of Mobile Agent

Mobile agents provide a new programming paradigm and a very new scenario to develop complex applications. In some cases, this technology is one of the few available (often the only) to implement applications with special requirements, such as wireless networks.

4.2.1 Definition

There are many definitions of an agent. The major reason for this variance is due to the exponential growth of diversity and functionality. According to the most accepted definition of software agents, agents are simply computer systems that are capable of autonomous action in some environment in order to meet their design objectives. An agent will typically sense its environment, and will have available a repertoire of actions that can be executed to modify the environment, which may appear to respond non-deterministically to the execution of these actions [120].

Mobile agents (MAs) add to regular agents the ability to travel to multiple locations in the network, by saving their state and restoring it in the new host. As they travel, they work on behalf of the user, such as collecting information or delivering requests. This mobility greatly enhances the productivity of each computing element in the network and creates a powerful computing environment [121].

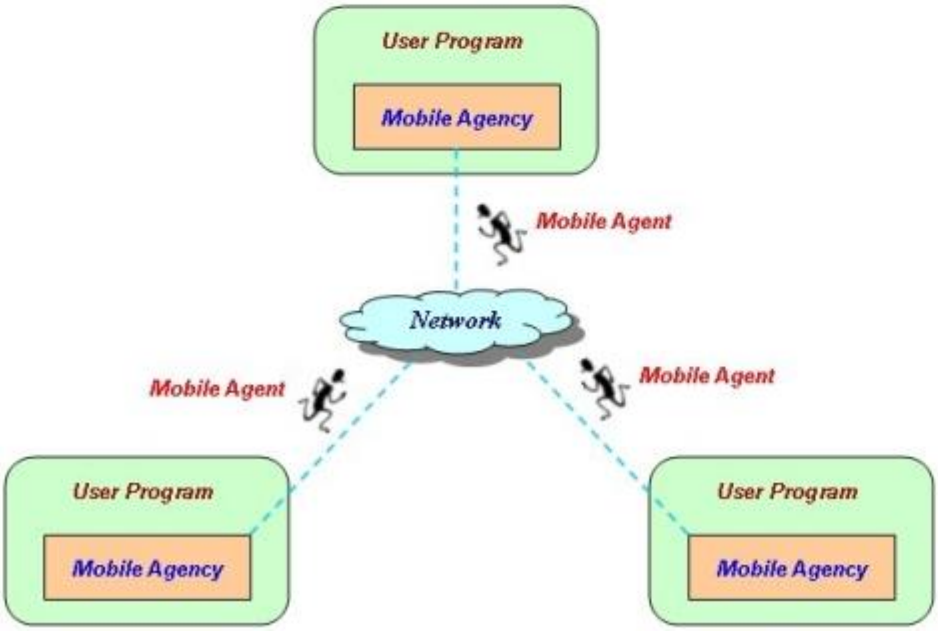


Figure 4.1: Shows the Mobile Agent Concept

Mobile agents require a software infrastructure that provides them security and data protection. This infrastructure includes protocols, rules for safe mobility, and directions and directories with information about all available hosts [122].

4.2.2 Structure of Mobile Agents

Mobile agents consist of three components: *code, data, and execution state*. The code contains the logic of the agent, and all agents of the same type use the same code. The code must be separated from the code of the agency so that it can be transferred alone to another agency, and the code must be identifiable and readable for an agency.

The second component of an agent is data. This term corresponds to the values of the agent’s instance variables if we assume an agent to be an instance of a class in object-oriented languages. The data is sometimes also called the object state. It is important to note that not all data items an agent can access are part of its object state.

The third component is the execution state. The difference between object and execution state information is that the agent itself directly controls the elements of the object state, whereas the processor and the operating system usually control execution state information [123].

4.2.3 Properties of Mobile Agents

The mobile agent paradigm provides some very interesting properties. There are several property but we will mention the most important from our point of view [119, 122, 124].

- Mobility: Transport itself from host to host within a network. This is the most distinguishing property from other kinds of agents. Note that a moving agent will carry its identity, execution state, and program code so that it can be authenticated and hence can resume its execution on the destination site after the move.
- Autonomy: Agents are capable of operating without the direct intervention of humans or others, and have some kind of control over their actions and internal state.
- Asynchrony: A mobile agent does not need a permanent connection to its owner, i.e. the device of the owner; it performs its task asynchronously.
- Intelligence: Interact with, learn from the environment, and make decisions. A most advanced agent should be able to decide its action based on its knowledge and the information it gets en route, and thus be able to generate new knowledge from its experience.
- Recursion: Create child agents for subtasks if necessary. An important concept is agent cloning. The agent can clone itself, that is, create a new mobile agent that is a copy of the parent.
- Collaboration: Cooperate and negotiate with other agents. Complicated tasks can be carried out by collaboration of a group of agents.

4.2.4 Why Are Mobile Agents a Good Idea?

In this section, we want to describe some major advantages of mobile agents and try to explain why they will meet the requirements of future distributed systems [119, 122, 123, 125].

- **Reduction in Network Load**

Mobile agents are useful when reducing the flow of raw data in the network. When very large volumes of data are stored at remote hosts, that data should be processed in its locality rather than transferred over the network.

- **Delegation of Tasks**

A user can employ a mobile agent as a representative to which the user may delegate tasks. Autonomous mobile agents aim at taking care of entire tasks and working without permanent contact and control. As a result, the user can devote time and attention to other, more important things.

- **Adapt Dynamically**

Mobile agents can examine their execution environment, and adapt dynamically to changes. For example, if the host signals shutdown, the agent can pick up and go to another host to continue its work.

- **Protocol Encapsulation**

Protocols enable components of a distributed system to communicate and co-ordinate their activities. Mobile agents permit new protocols to be installed automatically, and when a protocol is upgraded, only the mobile agent has to be altered.

- **Robustness and Fault Tolerance**

The potential of a mobile agent to react dynamically to unfavorable situations makes it easier to build robust and fault-tolerant distributed systems. This is because of their mobility.

- **Reduction in Network Latency**

By transferring an agent across the network to the source of data to process it there, the communication bandwidth and communication latency can be reduced.

- **Naturally Heterogeneous Property**

Network computing is fundamentally heterogeneous, often from both hardware and software perspectives. Mobile agents are preferably implemented in an interpretive language; they are independent of platforms and networks.

4.3 Mobile Agent Security

Although the mobile agent technology has many significant benefits to be applied to a wide range, it also brings significant new security threats because the mobile code generated by a party will be transferred and run in an environment controlled by the other party. Several security issues arise in various areas to mobile agent computing, including authentication, authorization (or access control), intrusion detection, etc. Malicious agents, platforms and third parties could attack a mobile agent system. In addition, mobile agents have characteristics such as mobility. Therefore, security issues become more complicated in mobile agent systems. This section introduces the concept and structure of mobile agent security, discusses various attacks and countermeasures of mobile agent systems [126].

4.3.1 Security Threats of Mobile Agent

Since mobile agents migrate through open and insecure networks and are executed on hosts of uncertain trust, security is a major concern. The security threats in mobile agent systems can be divided into three categories [64]: threats of malicious agents, threats of malicious hosts, and threats during migration, as shown in Figure 4.2.

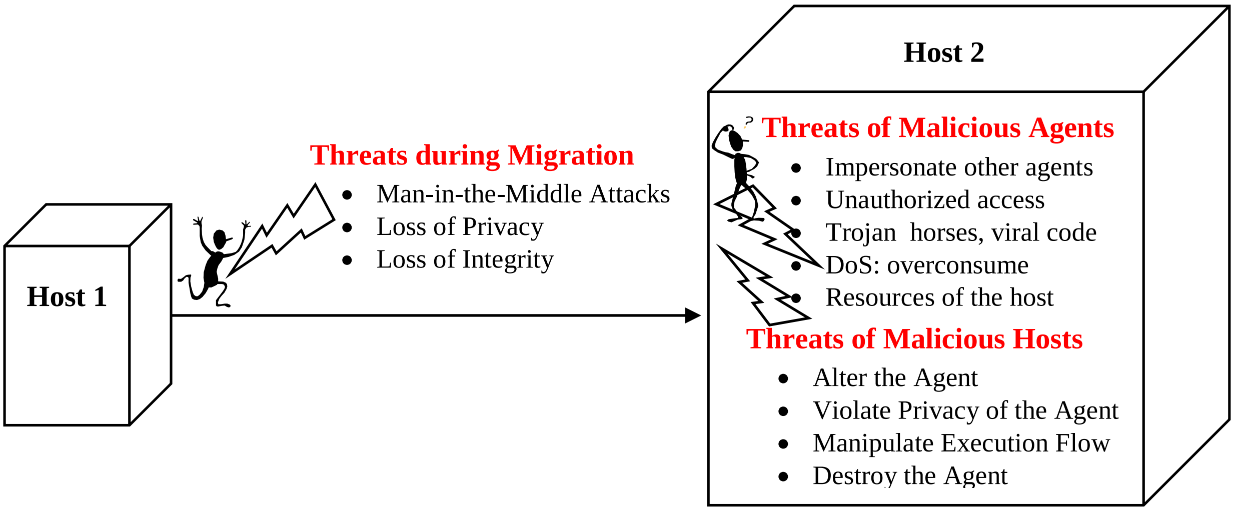


Figure 4.2: Basic Threat Model in Mobile Agent Technologies

4.3.1.1 Threats of Malicious Agents

A malicious agent is a potential security threat for the hosts within the network. Such agents may attempt to impersonate a legitimate agent in order to gain unauthorized access to a particular host. They may eavesdrop the execution host, for example, through a hidden Trojan horse, in order to transfer confidential information to another host controlled by the attacker.

Additionally, they may cause denial-of-service attacks on the executing host if proper precautions are not taken by consuming the bandwidth or the resources of the host. In addition, part of the agent's execution code may be destructive code, such as viral code [64]. We further classify malicious agents according to the target they attack [123, 126].

- **Attacking the Hosting Agency**

The most obvious example of a malicious agent is one that consumes resources of the hosting environment in an improper way. For this reason, the agency eventually is not able to provide its usual service to other agents. Such attacks are therefore called denial-of-service attacks. In a less severe case, the agent merely wants to annoy the agency's administrator by opening windows on its screen. In this case, the agent is authorized but does not comply with the unwritten rules of a benevolently behaving agent.

The second type of attack to the hosting agency is when an agent tries to gain unauthorized access to the agency. If it succeeds, not only the privacy and integrity of this agency may be detrimentally affected, but also other mobile agents would be attacked from a variety of aspects, which in turn will have a negative impact on the entire mobile agent system.

A malicious mobile agent may also claim itself as another agent on a mobile agent platform. Such masquerading action can also be called "faking". The results of masquerading include unauthorized access and even damage to platform resources, leaking confidential secrets, and ruining the established trust and reputation of the legitimate agent.

- **Attacking Other Agents**

A malicious agent that wants to attack other agents currently residing at the same agency has several possibilities.

First, a malicious agent would gain full control of the referenced agent, could invoke methods outside the agent's own life-cycle model, and could modify accessible object variables. Consequently, no agent must have access to any other agent on the programming language level. Second, a malicious agent can mask its identity to cheat other agents and gain sensitive information from them or to use services on behalf of the betrayed agents without paying for them. Third, a malicious agent could initiate denial of service attacks on other agents, for example, by sending thousands of spam messages. The attacked agent is not able to work properly and the agent owner suffers from these attacks. Finally, an agent rejects the result of a communication with another agent. This can be done intentionally or unintentionally. In either case, there will be a quarrel about this, and the agency should prevent that by logging all agent activities.

4.3.1.2 Threats of Malicious Hosts

Mobile agents are extremely vulnerable to attacks from malicious hosts since the execution of the agent relies on the host. Since the host has access to the code, data, and state of an agent on execution time, a malicious host may alter, tamper, or manipulate the code, data, and state of the agent [64, 123].

- **Attacking Other Agencies**

Attacks against other agencies are directed at the communication link between agencies. Using passive attacks, such as eavesdropping, where in which the adversary monitors the communication link between two agencies and captures agents to extract useful information from the agent's state or code. This might result in a leakage of sensitive information.

Another form of attack is traffic analysis. Here, the adversary attempts to find patterns in the communication between two agencies, which might allow the adversary to derive certain assumptions based on these patterns. Active attacks include security threats in which an agency tries to manipulate agent code or data while it is transmitted between agencies. The most common examples of this kind of attack are alterations and impersonation. Sometimes malicious agencies attack an agent to cause another agency to malfunction.

- **Attacking Agents**

Attacks against agents involve malicious agencies that try to tamper with an agent's code or data. Unfortunately, this type of attack is much more difficult to prevent than malicious agent attacks. The general problem is that a mobile agent must disclose its information about code and data if it wants to be executed. We distinguish this type of attack with regard to the type of information that is targeted.

- Modify Mobile Agent's Code: The mobile agent's code has to be readable by a guest Host. This characteristic makes the attack of leaking out/modifying mobile agent's code unavoidable. A malicious Host could read and remember the instruction going to be executed and might infer the rest of the program based on that knowledge. Thus, the Host could get to know the strategy and purpose of the mobile agent.
- Modify Mobile Agent's Data: This could be very dangerous too. Since some data are security sensitive, such as security keys, electronic cash, or social security numbers, it

may cause leak of privacy or loss of money. If a malicious Host knows the physical location of data, it may modify the data in accordance with the semantics of data. Therefore it can result in very severe consequences.

- Modify Mobile Agent's Execution Flow: If a malicious Host knows a mobile agent's code, data and the physical location of its program counter, it can infer what instruction will be executed next. Moreover, it can deduce the state of that mobile agent. Then it might change the execution flow according to its will to achieve its goal.
- Denial of Service: A malicious Host can simply not execute the mobile agent migrating to it or put the agent into waiting list and thus cause delay to that mobile agent.
- Masquerade: A malicious Host may disguise itself as a Host to which a mobile agent will migrate to or even as the home Host when the mobile agent returns. If it succeeds, it can get the secrets of the mobile agent by cheating and at the same time hurt the reputation of the original Host.
- Modify the Interaction between a Mobile Agent and other Parties: A malicious host may eavesdrop on the interaction between a mobile agent and other parties like another agent or another Host. From the information it gets, it may infer some secrets about the mobile agent and the third party.

4.3.1.3 Threats during Migration

These threats are related to logical attacks on mobile agents during their transmission from one host to another, such as man-in-the middle attacks. For this reason, they can be considered a special case of malicious host threats.

In the following sections, we attempt to provide a taxonomy of the solutions proposed in the literature in order to help the reader understand the aim of each proposed solution, the assumptions it is based on, and the practicality of each particular implementation.

4.3.2 Security Mechanisms in Mobile Agent

Security mechanisms are mechanisms designed to prevent, detect or recover from security attacks. Several mechanisms have been proposed in the literature to confront security threats of mobile agent systems. These are generally divided into two basic categories: host protection and agent protection mechanisms. We analyze these two categories.

4.3.2.1 Host Protection

The primary issue in the security of mobile agent systems is to protect host against malicious attacks launched by the agents. We now consider the problem of host how can be protected against malicious agents. Actually, this problem is played down and regarded as almost solved in large parts of the literature. In fact, the problem of malicious agents seems to be better understood than the reverse problem of malicious hosts. The main ideas of protecting a host from attacks should include the following three aspects. First, a safe environment should be provided for execution of any alien program, which includes both software-based fault isolation and safe-code interpretation. Second, the safety properties of any alien code should be checked before being executed on the platform. Third, the security should be ensured through signed code and path histories [126].

- **Sandboxing**

Sandboxing is a software technique used to protect mobile agent platform (host) from malicious mobile agents. In an execution environment (platform), local code is executed with full permission and has access to crucial system resources [127]. This technique is includes the following elements [123]:

- Each Java class is loaded from a specific code source that is specified by a URL (which is called codebase). If the code is signed, the code source also includes information about the signer.
- A permission is a specific action that a code is allowed to perform. In Java, permissions have a type (a class name), a name, and an action.
- A protection domain is an association of code sources and a set of permissions.
- Policy files are used to define protection domains. They can be plain text files in which you define which permissions a code loaded from some URL will have.
- Keystores contain certificates that can be used to verify signed code. The agent authorization process defines which permissions a mobile agent should have.

Actually, code is executed in a sort of “sand box”, in a distinct domain, where very few things can be damaged. A single identifier associated to each domain checks access to memory and other resources. This approach has been adopted for Java applets distribution. The problem is that the creation of a “sand box” means setting some limitations to the code, restrictions that, to some particular applications, can be too strict [128]. Another problem of this technique is that it increases the execution time of legitimate remote code but this can be overcome by combining Code Signing and Sandboxing, as will be explained later [127].

- **Code Signing**

A fundamental technique to protect the code is to sign it with a digital signature. Digital signature is a means ensuring the confirmation of the code's authenticity, its origin and its integrity. Usually who signs the code is the agent's creator or user; hence, digital signature is considered as an indication of the authority under which the agent operates [128].

Code signing involves public key cryptography, which relies on a pair of keys associated with an entity. One key is kept private by the entity and the other is made publicly available. The agent code, signature, and public key certificate can then be forwarded to a recipient, who can easily verify the source and authenticity of the code. In fact, the meaning of a signature may be different depending on the policy associated with the signature scheme and the party who signs [129].

There are two main drawbacks of the Code Signing approach. Firstly, this technique assumes that all the entities on the trusted list are trustworthy and that they are incorruptible. Mobile code from such a producer is granted full privileges. If the mobile agent is malicious, it can use those privileges not only to directly cause harm to the executing platform but also to open a door for other malicious agents by changing the acceptance policy on the platform. Moreover, the effects of the malicious agent attack may only occur later, which makes it impossible to establish a connection between the attack and the attacker. Such attacks are referred to as "delayed attacks". Secondly, this technique is overly restrictive towards agents that are coming from untrustworthy entities, as they do not run at all [127]. The approach that combines Code Signing and Sandboxing described in the next alleviates these drawbacks.

- **Code Signing and Sandboxing Combined**

According to [127], Java JDK 1.1 combines the advantages of both Code Signing and Sandboxing. If the code consumer trusts the signer of the code, then the code will run as if it were local code, that is, with full privileges being granted to it. On the other hand, if the code consumer does not trust the signer of the code then the code will run inside a Sandbox as in JDK 1.0.

The main advantage of this approach is that it enables the execution of the mobile code produced by untrustworthy entities. However, this method still suffers from the same drawback as Code Signing, that is, malicious code that is deemed trustworthy can cause damage and even change the acceptance policy.

The security policy is the set of rules for granting programs permission to access various platform resources. The “black-and-white” policy only allows the platform to label programs as completely trusted or untrusted, as the case in JDK 1.1. The combination of Code Signing and Sandboxing implemented in JDK 1.2 (Java 2) incorporates fine-grained access control and follows a “shades-of-grey” policy. This policy is more flexible than the “black-and-white” policy, as it allows a user to assign any degree of partial trust to a code, rather than just “trusted” and “untrusted”. There is a whole spectrum of privileges that can be granted to the code. In JDK 1.2 all code is subjected to the same security policy, regardless of being labelled as local or remote. The run-time system partitions code into individual groups called protection domains in such a way that all programs inside the same domain are granted the same set of permissions. The end-user can authorize certain protection domains to access the majority of resources that are available at the executing host while other protection domains may be restricted to the Sandbox environment. In between these two, there are different subsets of privileges that can be granted to different protection domains, based on whether they are local or remote, authorized or not, and even based on the key that is used for the signature. Although this scheme is much more flexible than the one in JDK 1.1, it still suffers from the same problem, that an end user can grant full privileges to malicious mobile code, jeopardizing the security of the executing platform.

- **Safe Code Interpretation**

Agent systems are often developed using an interpreted script or programming language. The main motivation for doing this is to support agent platforms on heterogeneous computer systems. Moreover, the higher conceptual level of abstraction provided by an interpretative environment can facilitate the development of the agent's code [129].

The main idea of this security policy is that commands that could be harmful to the platform be made safe or refused by an agent. A dangerous command, for example, is that which considers the execution of a common string of data as a program’s fragment. The most commonly known language security interpreter is probably Safe TCL, used in the first development of the Agent TCL system. It is based on the concept of “padded cell”, referring just to such access isolation and control technique. After this first interpretation, a second “safe” interpreter examines the code, before being executed by TCL interpreter, and points out possible harmful commands to platform. Various safe interpreters can be implemented, to create various kinds of approaches [128].

- **Proof-Carrying Code**

This technique implies that the code's producer formally proves that the code he wrote is safe, i.e. it conforms to the security features previously agreed on with the code's user and, therefore, it can safely be installed and executed [128]. The code consumer publishes a safety policy that describes properties that any mobile code has to comply with by using an extension of first-order logic, receives the PCC, validates the proof that is part of the PCC, and loads the code. This check must be done only once, even if the code is going to be executed several times. Afterward, the code can be executed without any additional checking [123].

The PCC involves low-cost static program checking after which the program can be executed without any expensive run-time checking. In addition, PCC is considered "tamper-proof" as any modification done to the code or the proof will be detected. These advantages make the Proof Carrying Code technique useful not only for mobile agents but also for other applications such as active networks and extensible operating systems [127].

Nevertheless, some kind of security policy formalism must in fact be established, as well as automatic support for proof generation and a technique to limit the several proofs that can occur [128]. Experiments showed that it can become even larger than the code that it has to prove and, in the worst case, can be exponentially larger than the size of the program [123]. In addition, the technique is tied to the hardware and operating environment of the code consumer, which may limit its applicability [129].

- **State Appraisal**

The "State Appraisal" is a technique to ensure that an agent has not become malicious or modified because of its state alterations at an untrustworthy platform. A mobile agent is roaming with carries the following information: code, static data, collected data, and execution state.

In this technique the author, who creates the mobile agent, produces a state appraisal function. This function calculates the maximum set of safe permissions that the agent could request from the host platform, depending on the agent's current state. Similarly, the sender, who sends the agent to act on his behalf, produces another state appraisal function that determines the set of permissions to be requested by the agent, depending on its current state and on the task to be completed. Subsequently, the sender packages the code with these state appraisal functions. If both the author and the sender sign the agent, their appraisal functions will be protected against malicious modifications. Upon receipt, the target platform checks and verifies the correct state of the incoming agent. Depending on the result of the verification process, the platform can determine what privileges should be granted to this incoming agent given its current state [127].

However, this approach also has some drawbacks. The main problem with this technique is that it is not easy to formulate appropriate security properties for the mobile agent and to obtain a state appraisal function that guarantees those properties. It is not clear how well the theory will hold up in practice, since the state space for an agent could be quite large, and while appraisal functions for obvious attacks may be easily formulated, more subtle attacks may be significantly harder to foresee and detect [129]. Even in specific application domains, the decisive issue is whether it is possible to find suitable appraisal functions that can distinguish normal results from deceptive alternatives [123].

- **Path Histories**

Path History is helpful for the security of mobile agent platforms that a mobile agent maintains a record of the platforms it has already visited [126].

The “Path History” is constructed in the following way. Each visited platform in the mobile agent’s travel life adds a signed record to the Path History. This record should contain the current platform’s identity together with the identity of the next platform to be visited in the mobile agent’s travel path. Moreover, in order to prevent tampering, each platform should include the previous record in the message digest that it is signing. After executing the agent, the current platform should send the agent together with the complete Path History to the next platform. Depending on the information in the Path History, the new platform can decide whether to run the agent and what privileges should be granted to the agent [127].

A drawback of this approach is that the size of the path history increases with the number of hops, and in the same manner, the time for verification also increases. Some researchers believe a major drawback of this approach that each agency already must have a sense of trust; in addition, some technique must be available to determine whether it can trust another agency. However, path histories as a concept have already influenced some other techniques [123].

4.3.2.2 Agent Protection

In the previous section, we presented several techniques for protecting mobile agent platforms (Protection of Host) against malicious mobile agents. In order to improve the security of mobile agents against the attacks that are launched by malicious platforms, many security techniques have been suggested. In this section, we explore these techniques.

- **Execution Tracing**

Execution Tracing enables detection of any possible misbehavior by a platform, that is, improper modification of the mobile agent code, state, and execution flow. This technique assumes that all the involved parties own a public and private key that can be used for digital signatures, in order to identify involved parties. Different parties, such as users and platform owners, communicate by using signed messages. A platform that receives the agent and agrees to execute it produces the associated trace during the agent's execution. The message that an execution platform attaches to the mobile agent typically contains information such as the unique identifier of the message, the identity of the sender, the timestamp, the fingerprint of the trace, the final state and the trusted third party (which could later be used to resolve disputes). Later, the owner of the agent may suspect that certain platform cheated while executing the agent. If this is the case, the owner will ask the suspicious platform to reproduce the trace. Finally, the agent's owner validates the execution of the agent by comparing the fingerprint of the reproduced trace against the fingerprint of the trace that is originally supplied by the suspicious platform.

Execution Tracing has some limitations, such as the potential large size and number of logs to be retained. Another limitation of this technique is that the owner platform needs to wait until it obtains suspicious results in order to run the verification process. In addition, this technique is considered too difficult to use in the case of multi-threaded agents.

A new version of the Execution Tracing technique proposed, which modifies the original technique by assigning the trace verification process to a trusted third party, the verification server, instead of depending on the agent's owner. Execution tracing with a verification server does not wait until a suspicion is raised in order to run the verification process. The verification here is compulsory and this is an advantage over the original Execution Tracing technique where the verification process is triggered only by suspicious results. However, Execution Tracing with a verification server still suffers from the same limitation as the original technique. Additionally, each platform chooses a verification server and that might encourage and facilitate a possible malicious collaboration between a platform and the server [127].

- **Obfuscated Code**

Code obfuscation [130] aims at generating executable agents, which cannot be attacked by reading, or manipulating their code i.e., an agent is a black box if its code and data cannot be read or modified at any time. This technique is based on transforming the agent code in such a way that it is functionally identical to the original one, but it is impossible to understand it.

There are different useful obfuscating transformations. Layout Obfuscation tries to remove or modify some information in the code, such as comments and debugging information, without affecting the executable part of the code. Data Obfuscation concentrates on obfuscating the data and data structures in the code without modifying the code itself. Control Obfuscation tries to alter the control flow in the code without modifying the computing part of the code. Preventive Obfuscation concentrates on protecting the code from decompilators and debuggers [127].

In fact, there are many techniques for improving the effectiveness of these approaches. However, the major drawbacks of these techniques is that there is no known method or algorithm for providing black box protection. Computing with encrypted functions is cited as an example, but serious reservations about the limited range of input specifications that apply are raised. A time limited black box implies that code or data of an agent cannot be read or modified within a known time interval and that after the interval, the attacks do not have effects [131].

- **Co-Operating Agents**

The Co-Operating Agent technique [127] distributes critical tasks of a single mobile agent between two co-operating agents. Each of the two cooperating agents executes the tasks in one of two disjoint sets of platforms. The co-operating agents share the same data and exchange information in a secret way. The Co-Operating Agent technique reduces the possibility of the shared data being pilfered by a single host. Each agent records and verifies the route of its co-operating agent. When the agent travels from one platform to another, it uses an authenticated communication channel to pass information to its co-operating agent.

However, this technique has some drawbacks. The first one being the complexity of defining subgroups of platforms that will not collaborate with each other to attack the application. The second drawback is the need to establish a secure authenticated channel between the agent and its co-operators, which may not be possible to provide in all scenarios. Besides, this technique undermines the agent's autonomy, for it requires the agent to interact with other agents in order to carry out its tasks [130].

- **Environmental Key Generation**

The environmental Key Generation describes a scheme for allowing an agent to take predefined action when some environmental condition is true [129].

This can be achieved by sending a mobile agent carrying an encrypted message. The encrypted message may include some data and/or executable code. Neither can the mobile agent precisely predict its own execution at the receiver platform, nor can the platform foresee the incoming agent task. The agent will wait at the receiving platform for some environmental condition to occur [127].

The basic scenario is as follows: The agent has a cipher-text message and a method to search the environment for the data needed to generate the secret key for decryption. When the information is found, the agent can generate the key and decipher the message. Without this key, the agent has no idea about the content and the semantic of the encrypted message; that is, the agent is clueless.

Therefore, we see, the general idea is to give the agent only hash values of some information and let the agent compare this hash value with computed hash values at the remote agency. If they match, the hash value is used as a key to decrypt additional information or code that should be processed now.

Thus, the goals of this method are (1) to protect the intention of the agent by not giving it full knowledge about its task and (2) to protect further against actions by using encryption. It is obvious that the hosting agency can still attack the agent after it has decrypted the message, but to do this, it must be executed. An analysis a priori, for example, by a dictionary attack, is very costly [123].

In fact, this technique has some limitations. The receiving platform could act maliciously against the incoming agent. When the environmental condition is met and the activation key is generated, the platform could modify the agent to perform a different function, for example, to print out the executable code instead of running it [127] (i.e., this technique is that it protects data and code but does not protect the behavior of agent.). Another problem with this technique is that decrypting pieces of code at runtime implies that it must be allowed to create code dynamically, which might be prohibited by the hosting agency and/or the underlying execution environment [123].

- **Partial Result Encapsulation**

Partial Result Encapsulation (PRE) is a detection technique tampering by malicious hosts that aims to discover any possible security breaches on an agent during its execution at different platforms. PRE is used to encapsulate the results of an agent's action, at each platform visited for subsequent verification. The verification can be done either when the agent returns to the point of origin or at intermediate points [131].

The PRE technique has different implementations. In certain scenarios, the encapsulation can be done by the agent, platform or by a trusted third party. To meet certain security requirements such as integrity, accountability, and privacy of the agent, PRE makes use of different cryptographic primitives, such as encryption, digital signatures, authentication codes, and hash functions [127].

Another method for an agent to encapsulate result information is to use Partial Result Authentication Codes (PRAC), which are cryptographic checksums formed using secret key cryptography (i.e., message authentication codes). This technique requires the agent and its originator to maintain or incrementally generate a list of secret keys used in the PRAC computation [129].

The PRAC technique has a number of limitations [129]. The most serious occurs when a malicious platform retains copies of the original keys or key generating functions of an agent. If the agent revisits the platform or visits another platform conspiring with it, a previous partial result entry or series of entries could be modified without the possibility of detection. Since the PRAC is oriented towards integrity and not confidentiality, the accumulated set of partial results can also be viewed by any platform visited, although applying sliding key or other forms of encryption easily resolve this.

Some researchers devised a platform-oriented technique for encapsulating partial results, which reformulated and improved on the PRAC technique. A variant of this technique, which uses message authentication codes in lieu of digital signatures, is also described.

- **Computing with Encrypted Functions**

This technique represents a software solution for protecting a mobile agent from a malicious executing platform during its itinerary. This is a cryptographic solution to achieve integrity and privacy of the mobile agent. Achieving privacy means that the mobile agent can conceal its program (code) when it is executed remotely in an untrusted environment [127]. Supposing that a mobile agent has to execute a certain function *f*, then f is encrypted to obtain *E(f)*, and a program is created that implements *E(f)*. Platforms execute *E(f)* on a cleartext input value x, without knowing what function they actually computed. The execution yields *E(f(x))*, and this value can only be decrypted by the agent owner to obtain the desired result *f(x)* [130].

Although the idea is straightforward, the trick is to find appropriate encryption schemes that can transform arbitrary functions as intended [129]. In addition, this technique protects the mobile agent's integrity and privacy [127], however, does not prevent denial of service, replay, experimental extraction, and other forms of attack against the agent.

- **Detection of Denial of Services**

This method in order to enable detection of any Denial of Services' attack (DoS) on the agent. DoS attack includes preventing the agent from accomplishing its task, preventing the agent from migrating to its next destination, and destroying the agent. The method is based on the usage of undeniable proofs, e.g, digital signature. When agent's owner suspects that the agent suffered from DoS attack, e.g. the agent did not return back after a certain threshold period of waiting time. The owner then asks all visited platforms in agent's itinerary to introduce the undeniable proof in order to judge that the visited platform did not launch DoS attack against the agent and correctly dispatch the agent [127].

4.4 Approaches based Mobile Agent Security for Ad Hoc Networks

The different security mechanisms have been invented to counter malicious attacks. A first line of defense includes encryption, access control, authentication, and digital signature. As a second line of defense, intrusion detection systems and cooperation enforcement mechanisms implemented in MANET can also help to defend against attacks or enforce cooperation, reduction of selfish node behavior.

In this section, we will briefly explain some study to achieve security in ad hoc networks, which based mobile agent paradigm that is a new solution and it has important advantages such as, distribution of management code, robustness, dynamically changing network policies, data collection, re-activeness, decentralization, network monitoring, and so on.

The several directions of progressing research in ad hoc networks are based on security challenges that cover various classes of security attacks and how ad hoc network can defend against those attacks. Various other approaches are proposed in the last few years based on existing mechanism. One of this mechanism discussed in [132], where the network is splitted into a power two number grid clusters, respecting to the available battery level. A node in the cluster is elected to be the cluster head and the rest nodes become cluster members. In each cluster there is a dedicated mobile agent consist of four modules: Registration Module (RM), Service Agreement (SA), Detection Module (DM) and Prevention Module (PM).

All the node in the cluster including the cluster head have to be register with mobile agent, and the MA store the list of all cluster nodes in the RM. The Detection Module of the mobile agent analyse the packets exchanged between nodes, if any mismatch is found, the MA informs the CH to drop the packet and to block the node. The communication inter-cluster is possible with the same supervision of the MA, but the packets have to be transmitted from CH to the other CH.

A new approach called securing DSR with mobile agents in wireless ad hoc networks proposed in [133]. The authors try to secure Dynamic Source Routing (DSR) protocol of an ad hoc network by using mobile agents. There are three types of mobile agents used in this routing protocol: discovery/reply of mobile agent, maintenance of mobile agent, update/approve for symmetric key mobile agent. Hybrid encryption technique (symmetric key encryption/public key encryption) is used to improve performance; where symmetric keys are used to encrypt routing data to authenticate and authorize node-sending data, while, public keys are used for the exchange of symmetric keys between nodes.

The distributed trust based framework presented in [134] to protect the agents and the host platforms against threats of the environment like the kill of the agents while visiting some hosts, the authors propose a threat model, where they assume that any node in the network can be malicious node, which kill or misrouting the arrived agent. Due to the nature of MANET, nodes can only have an opinion about its neighbours, this opinion in defined as the degree of trust between nodes. The authors define a model of trust as a reputation system, where they defined three concepts: belief (how much trustworthy a host is) or disbelief (how much suspected a host is) as well as uncertainty, this expressed mathematically as: $b + d + u=1$. Here b, d, u designate belief, disbelief and uncertainty respectively. They claimed that nodes could detect all the malicious nodes and eventually prevent themselves and their agents from network hostilities.

Another work proposed new approach in [135], where mobile agents collect information about the nodes of a cluster by visiting them one by one, until it returns to the cluster head. The cluster head to process of key deactivation, common leader election, use this information and key serving nodes selection one-way hash function protects the code of the mobile agent against any malicious modification. A secret key of cluster nodes is generated based on a distributed private key generation scheme, used for validate the identity of the cluster head and cluster's members. The paper presents the behaviour of the proposed protocol against several scenarios like: Masquerading, Eavesdropping, Unauthorized access and alteration, and Denial of service. The simulation of the proposed protocol is carried on using the ns-2 simulator because it is very used in such problems. The authors claim that the proposed schema is effective and provides a high packet delivery ratio and low delay compared to the cluster based routing protocol CBRP

The authors of [136] define a new composite key management technique for key management in ad hoc network. A network is partitioned into clusters based on the dominator concept, in each cluster a node considered the most trusted and active is elected as a cluster head. A fuzzy logic controller calculates the degree of trust of nodes, which represent the degree of belief about the future behavior of other entities. In addition to the public key, each node has also a private key generate by a specific cluster called the Primary Key Generation, which are a number of cluster head with high value of trust.

Agent based trusted on-demand routing protocol for mobile ad-hoc networks is presented in [137], authors propose a protocol called ATDSR. It selects the most trusted as well as the minimum hop count route from different possible routes with minimal overhead in terms of extra messages and time delay. This protocol uses a Multi-Agent System (MAS) that consists of two types of agents that cooperate with each other to achieve the required task; specifically Monitoring Agent (MoA) and Routing Agent (RoA). MoA is responsible for monitoring its hosting node behavior in the routing process and then computing the trust value for this node. RoA is responsible for using the trust information and finding out the trust worthiest route for a particular destination.

The authors proposed a clustering protocol and trusted model for enhancing the security in the network [138], the clustering protocol organizes the network into one-hop disjoint clusters and elects the most qualified, trustworthy node as a cluster head. Each node computes its trust level using self and recommendation evidences of its one-hop neighbors.

Another work proposed new approach in [139]; the main idea of this approach is based on dummy agent. It is proposes a secured mechanism to manage the security of the mobile agent for MANET. Before transferring the mobile agent to a node, a source station it sends a dummy agent to check if the node is malicious or not. in case the node is intrusion, the source station transfer another agent called supervisor for it inform about the intrusions or kills the mobile agents at wireless node before intruder infects to it.

Among the works that Addressed a security of ad hoc network mentioned in [140]. This approach based on the key management, where benefit from the advantages of mobile agents to use them in the process of exchanging of private key and network topology between the nodes. A mobile agent travels from node to another according to least-visited-neighbor-first algorithm, when a new node wants to join the network, it created a new mobile agent carrying the public and the secret key, and a set of nodes will cooperate with each other to authenticate the new one using their mobile agents.

This approach [141] incorporates agents and data mining techniques to prevent anomaly intrusion in mobile ad hoc networks. Home agent is present in each system and it collects the data about its system from application layer to routing layer. This approach provides security solution to current node, neighboring node, and global networks, where it monitors its own system and its environment dynamically, it uses classifier construction to find out the local anomaly and it provides the same type of solution throughout the global networks.

The author in [142] develops an Intrusion Detection System as a combination of the rule based and the behaviour based scheme; to defend the security of the network against the major security attacks. This IDS works as two phases system: A), Initialization Set up and Learning Phase. A collector agent collect a raw data from the network to be stored in the primary database, this data are filtered and processed then organized as atomic events used for treating simple network attacks, the events can be combined to create complex attack rules stored in the attack library. In other side, monitor agents are deployed on the network according to the clustered node selection algorithm. B) Agent Deployment and Intrusion Detection: in this phases a set of agents are initialized and organized as an hierarchical system, this system represents the proposed IDS in this paper, the main type of agents used in it are: *Network Monitoring Agents*: exist in few nodes of the cluster, their role is to collect the information necessary for IDS function. Host Monitoring Agents: a host-monitoring agent monitors each node. *Decision Making Agents*: it is role is to make decision for each node based on individual threshold threat level assigned. It can cooperate with the learning module to make critical decisions. *Database Agents*: database agents are of three types: the primary database used in phase “A” for learning; the host information database and the network information database to store information about the host and the network respectively. *Communication Agents*: this agent is part of the host and network IDS, its role is to read information from any arrived agent and if any new attack is found it will be added to the attack database by the database agent. *Alert Agents*: is used to notify the learning module by any node whenever a new attack is found.

The Ant Colony Optimization (ACO) is a swarm intelligence used in the paper [143] for developing a real time routing protocol, where the (ACO) is used as Identification Agent and Target Agent. In the initialization of network phase, ACO flooded in the network as Identification Agent to identify all authenticated members in order to process handshake. When an authenticated node of a group receives the message from unknown node, it initiates the mobile agent to collect security information of the unknown node. The Message Digest 5 hash function h=H (M) generates hash value, which is used to create message digest authenticated node. The authenticated node generates the digital signature (d_{sign} = $(H\ (M))^d$ mod n). If the unknown node is an authenticated node of the group, if the generated H (M) by the receiver and the decrypted H (M) of digital signature d^{sign} is equal, then the receiver accepts the data; otherwise rejects the data and informs the sender that the data is altered through by generating route error packet. This process is repeated in every hop of the noded is joint path between source and destination.

The authors of this paper [144] propose a new IDS architecture for MANET based on system multi agent, for protecting this type of networks against intrusions and attacks. The network in this proposition is structured as a set of clusters, inside it there are five types of agents consist the multi agent detection system, these agents are: Sensor Agent (SA): it captures network raw traffic and formats in a predefined format, then send it to the Agent Analyser (AnA). Manager Agent: The manager agent can ask other agents for local information related to suspicious activity, in this case one or more agents analyser located in different nodes in cluster can provide local information to the initiator. Ontology Agent (OA): for treating unknown attacks. Agent actuator (ACA): for treating known attacks. Agent Analyser (AnA): The agent analyser analyse the formatted data and compare the analysis result by applies rules of detection recorded in his database.

4.4.1 Discussion

In general, these approaches presented above on the security of ad hoc networks based mobile agent are certainly interesting, but it is suffered from one or more of the following limitations:

- There is no generalized framework that can be adapted to different types of network and application, i.e. although these approaches have been able to respond to a set of security requirements, they remain effective only in a specific context related to the assumptions and restrictive requirements that were issued during the design**.**
- These approaches does not interest on the different concerns of security in a system based on mobile agents. Indeed, the majority of approaches interested in specifying security policies to control the behavior of mobile agents and their access to resources.
- Most of these security schemes either provide protection to agents from agents/host or host from agent/external parties but not both.
- Most of these works are still only at the theoretical level, have not been implemented on the reality, and do not provide flexibility to users for specifying their desired security policies under a given attack scenario.
- In fact, there is not universal solution to the problem of malicious host. Only partial solutions have been proposed. In addition, most of these security mechanisms aimed detection rather than preventing attacks from malicious hosts.
- Finally, we noticed also that the majority of these works limit the representation of the mobile agent to a simple object or process. Such representation is devoid of all necessary concepts to express autonomy, intelligence, and the cognitive aspect of the agent (such as beliefs, knowledge and skills).

4.5 Conclusion

In this chapter, we gave overview of mobile agent, such as, definition, structure, propriety and why the mobile agent use or benefit of the mobile agent. We also described the security threats of mobile agent, security mechanisms in mobile agent, such as, Sandboxing, Code Signing, State Appraisal, Execution tracing, Obfuscated Code, Environmental Key Generation,.....etc.

In fact, the problem of malicious hosts attacking an agent is by far the most difficult to solve. Although achieving a complete solution is considered impossible, mechanisms have been presented that mitigate several problems. However, the security mechanisms used in existing mobile agent systems are usually hard to be generalized and implemented. In addition, most existing mobile agent systems do not have formal methods as their bases. Therefore, it is not easy to analyze their features, and verify their security properties and consistency. It is still an open research question: Which of all these mechanisms can be combined to later form some kind of general security solution?.

Chapter 5

Security Model based on Mobile Agent for Ad Hoc Networks

> No one can build his security upon
> the nobleness of another person.
>
> Willa Cather

We present in this chapter our novel and effective model that we have devised and titled "Security Model based on Mobile Agent for Mobile Ad Hoc Networks". Our model relies on the concept of dominating set based clustering for partitioning network into clusters and mobile agent. The cluster head elected based on both the trust and resources ability of the node.

In the first contribution, we define three agent types. The Node Agent (NA) manages the use of node resources. The Monitor Agent (MoA) that is responsible for all operations within the cluster and outside with counterparts. The Monitor Agent creates the Inspector Agent (IA), which travails from node to another to examine the actions history of each node agent to detect any suspect behaviour, and returns to MoA with report shows the status of each node in the cluster. In the second contribution, we proposed four types of agents. Node Agent (NA), Monitor Agent (MA), Ambassador Agent (AmA), and Transporter Agent (TA). The Monitor Agent creates the Ambassador Agents and sending to all nodes in the cluster. The Ambassador Agent is like local IDS and IPS (Intrusion Detection System and Intrusion Prevention System) at the node. The third contribution is considered a hybrid approach for the two preceding. The Monitor Agent created in the most trusted with best resources node to control the communication inside and outside the cluster. We gave the architecture interne of mobile agent, class diagrams, and communication protocol.

5.1 Introduction

The concept of ad hoc networks is very promising as a new mode of telecommunication to complement and extend the existing communication systems. Ad hoc networks consist of terminals are generally small sizes, where resource constraints in terms of memory, batteries and can move randomly and at any speed (no existing infrastructure or centralized administration). These networks evolve in increasingly dynamic environments, unpredictable and hostile and therefore generate an important security issue whose resolution is a big challenge. This challenge explained by the importance of the use of mobile agents in distributed applications to take full advantage of the benefits of their strengths.

In fact, we discussed in the previous chapter why we used the mobile agent paradigm to achieve the expected security services. Studies previously conducted, which focused on the importance of the use of mobile agents to solve the problem of security in ad hoc networks explained that the benefits of the approach to mobile agent protection for their robustness, easy maintenance, and low cost. However, these solutions a lack of global protection solution to all security concerns.

In this chapter, we propose a security model for mobile ad hoc networks based mobile agent, where the network is consisting of a set of nodes, each node has node agent for resources estimation of the node and communicate with others agents. The network is divided into a set of clusters; each cluster must elect a node to be the head cluster (Monitor Agent). This monitor agent controls the communication inside cluster by collecting and analysing the data from the others nodes, it creates mobile agents to perform certain tasks.

In the following, we present the network organization, the internal architecture of the Node Agent, Inspector Agent, Ambassador Agent, Monitor Agent, and Manager Agent. In addition, the sequence diagrams and classes, the property of this model, and a conclusion.

5.2 The Proposed Model (Contribution 1)

In fact, we divide the network into clusters where each cluster has a dominator. The construction of clusters are a distributed manner, a self-organisable, and under the proposed conditions. There are three kind of agents: Node Agent, Monitor Agent, and Inspector Agent.

5.2.1 Organization of the Network

In our model, the security is carried out in three levels (Node Level, Cluster Level, and Network Level) by a set of agents, which communicate with each other a secure manner.

5.2.1.1 Node Level

The node agent is installed in each node to estimate available resources to better manage the resources of the node (battery, degree node, CPU and memory,) in order to satisfy application security requirements. We take into account the parameters *TL, EL, DN, CL, and ML,* to calculate the full capacity C_{ni} of the terminal, whereas:

$$C_{ni} = f\,(TL, EL, DN, CL, ML)$$

Knowing that:

TL: Trust Level

EL: Energy Level

DN: Degree Node (i.e. the highest number of neighbors)

CL: CPU Load

ML: Memory Load

To simplify our Model, we assume that these parameters are independent and we introduce the following equation to measure the capacity of a node:

$$C_{ni} = aTL + bEL + cDN + dCL + eML$$

Where:

a, b, c, d, e : are the security management parameters to favor a resource or a terminal compared to the other depending on the role of the agent will play or the proposed conditions, while: a + b + c + d + e =1.

5.2.1.2 Cluster Level

This level describes the interactions between nodes within the same group to manage local of various security features. Here the node takes the state: Member or Cluster Head.

5.2.1.3 Network Level

A network organizes as a set of clusters; each cluster is a set of nodes. We proposed a mechanism of mobile agent to manage security interactions between different clusters. The following figure illustrates the general architecture of our Model.

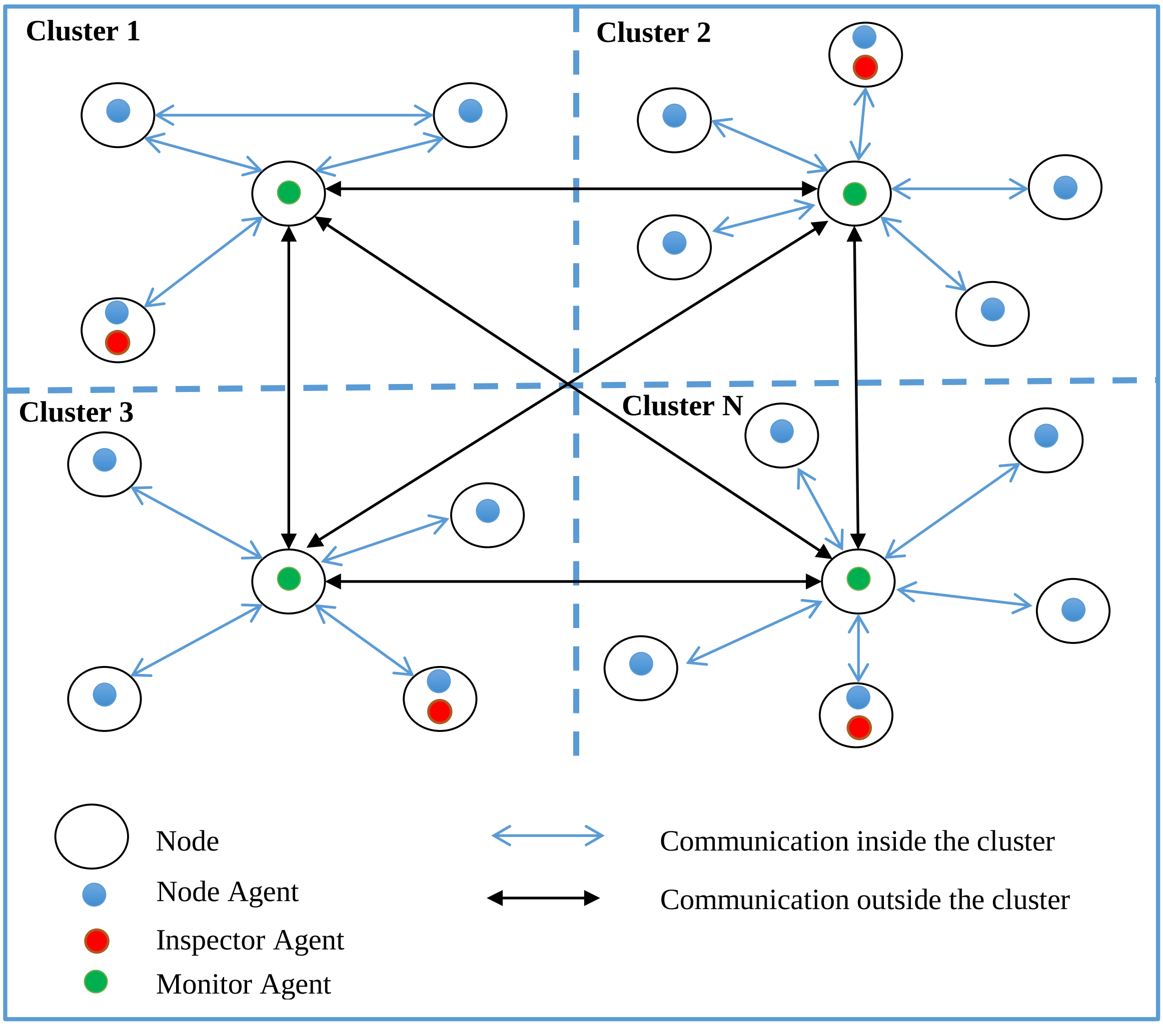


Fig 5.1: Our proposed architecture for security Model based Mobile Agent in MANETs

5.2.2 Algorithm to Construct Clusters

Several algorithms have been proposed in the literature for the construction and maintenance of groups. We will try to offer an algorithm to construct clusters according to the optimization function that we present before.

- Each node agent calculates the capacity of node (C_{ni}), create a message containing its routing table and send it to all adjacent nodes agents.
- After the authentication process, the exchange of information, groups of nodes agents make an election between them.

- Each group is a limited number of node agent, the node agent has the great ability elected as Monitor Agent, while others consider Members.
- In the initial state, we will give to a limited number of nodes a value *fully trusted* of the trust level, and the Monitor Agent after the election become a little movement.
- Each member agent can leave its group and enter another group after the approval of the Monitor Agent.

5.2.3 Modeling of the Trust Level

The aggregation of mistake and malicious behavior generated by the node is an important element in the elaboration of a final decision as estimating the trust that can grant to an entity (node). Let P = {p_1, p_2,….., p_i,….,p_n} the set of parameters involved in the evaluation of the trust. For example, a message/agent dropped, a message altered, a message delayed, a message repeated, and wrong password, etc. Let W_i is represent the weight assigned to the parameter P_i. The introduction of a weighting of different parameters to aggregate proposed. The Trust (T) formula is as follows:

$$T = 100 - \sum_{i=1}^{n} |P_i| * W_i$$

Where: $|P_i|$ is the number of occurrence of the error P_i. The following algorithm shows how to determine the level of trust.

```
Function getleveloftrust (int T)
{If 80 ≤ T ≤ 100 then
  Trust_Level: = 'Fully Trusted'
Else If 60 ≤ T < 80 then
  Trust_Level: = 'Normal'
Else If 40 ≤ T < 60 then
  Trust_Level: = 'Average'
Else If 20 ≤ T < 40 then
  Trust_Level: = 'Low'
Else If T < 20 then
  Trust_Level: = 'Not Trusted'
Return Trust_Level
}
```

Algorithm 1: Determine the Level of Trust

The definition of the trust parameters and weightings made by the network administrator. These two operations very linked to the service security criteria. If the ratio between the number of success operations and the number of all operations is greater than a defined threshold, we update the value of trust level by adding the average of weights assigned to mistakes and malicious behaviors. Therefore, the Trust Level will increase as follows:

$$T = T + \frac{\sum W_i}{|W_i|}$$

5.2.4 Architecture of the Mobile Agent

In our Model, there are three agents: Node Agent (NA), Inspector Agent (IA), and Monitor Agent (MoA). We present in the following the internal architecture of these agents. This architecture based on components where every component implements some functions of the agent.

5.2.4.1 Architecture of Node Agent

The node agent is installed in each node, it maintains routing table that represented by conceptual data structures with the necessary information. The structure of the routing table is the following:

Neighbor_ID	Cluster_ID	State	C_{ni}	Threshold
@ IP_N	@ IP_C	M/H	%	%

Table 5.1: Structure of the Routing Table

Knowing that:

Neighbor_ID: Is an identifier of a neighbor, we use the IP address of a node to identify it.

Cluster_ID: Is an identifier of a cluster, we use the IP address and the name of a cluster head.

State: This field designs the state of a node agent it may be a member or a cluster head.

C_{ni}: Represents the capacity of the node that calculated by the node agent.

Threshold: Represents the degree of capacity, if the capacity of a terminal reaches a constant value, it is necessary to inform others to reduce their load, or can be removed or replaced.

The figure 5.2 shows the architecture of the node agent, the main components that allow the agent implementation are the following:

Security component: This component has a function is to ensure the security agent against all malicious access, protect all information that is sent to other agents by well-defined mechanisms as symmetric and asymmetric key, etc.

Reception component: The reception's role is to receive information from other agents for evaluation or communication between them (e.g. information on the node resources, routing table, etc.).

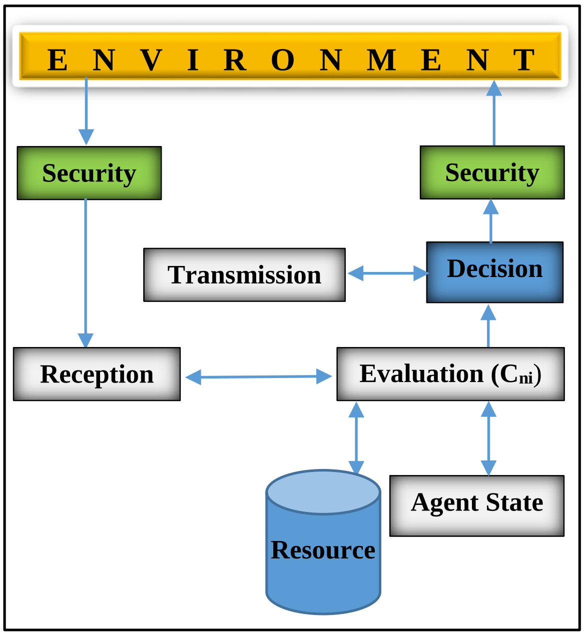


Fig 5.2: Architecture of Node Agent

Evaluation component: An agent is evaluate the resources of the node according to the optimization function that previously presented and proposed conditions.

Transmission: The transmission's role is to send message to other agents.

Decision component: It allows the agent to select the action to perform.

5.2.4.2 Architecture of Monitor Agent

The monitor agent is created in the node that called cluster head. This agent is the most important among other agents, where it is responsible for all operations within the cluster and outside with counterparts. The monitor agent maintains a table of confidence it contains the necessary information for the trustworthiness and authentication of each node in the cluster. The structure of the confidence table is:

Node	ID	Trust Level	Public Key
1			
2			
..			
..			
N			

Table 5.2: Structure of the Confidence Table

Knowing that:

ID: Is an identifier of a node, where each node has a unique identifier.

Trust Level: This field takes the following properties (Not Trusted, Low, Average, Normal, and Fully Trusted).

Public Key: A key generated by Monitor Agent.

The same status in the architecture of node agent, which based on components for simplicity, adaptability, evolution, and code reuse, etc., where every component implements some functions of the agent. The architecture of Monitor Agent is as follows:

Collection component: A collection component collects activity information (for instance, the process of sending and receiving agents, messages, agent log files, etc.), either from its node or from other agents. Those data are gives as an input analyzer component.

Analyzer component: An analyzer implement a verification policy, which is a set of rules defined for a set of events related to the node system or/and agent system (e.g. changes in the execution context or behavior).

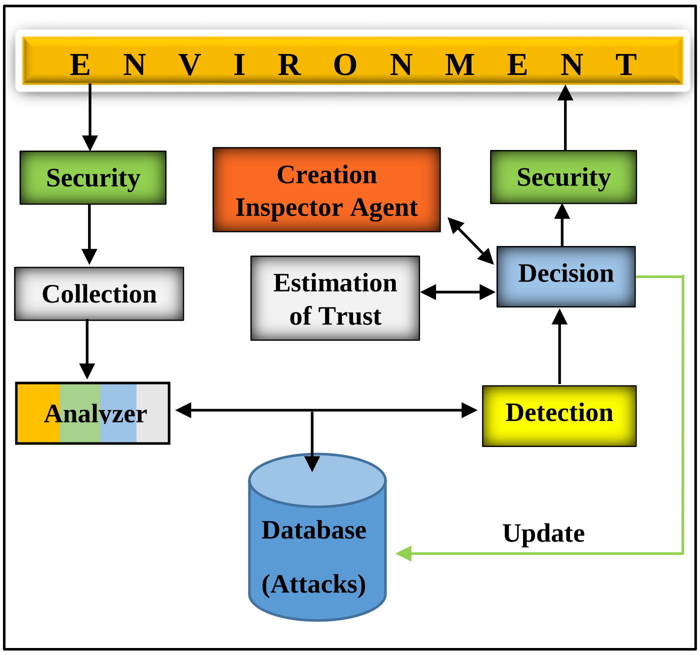


Fig 5.3: Architecture of Monitor Agent

Detection component: Their goal is a classification and detection. It uses results provided by analyzer component to detect the type of intrusions. It includes both a misuse detection, an anomaly detection, and specification detection. The procedure of a misuse detection used to determine the exact types of attacks. An anomaly detection procedure used to detect new or unknown attacks. Specification detection is a procedure where we defined a set of constraints that describe the correct operation of our Model. The execution of the Model should respect the defined constraints.

Estimation component: This component evaluates the trust level of a node by the formula that is already proposed. Therefore, the agent takes a decision (e.g.it will elect as cluster head, exclusion of the cluster and the network, or attempt to repair if it is possible).

5.2.4.3 Architecture of Inspector Agent

The Inspector Agent (IA) is created periodically by the Monitor Agent, it roles is to inspect each node locally and send the results to the monitor agent. Therefore, it travails from node to another to examine the actions history of each node agent to detect any suspect behaviour.

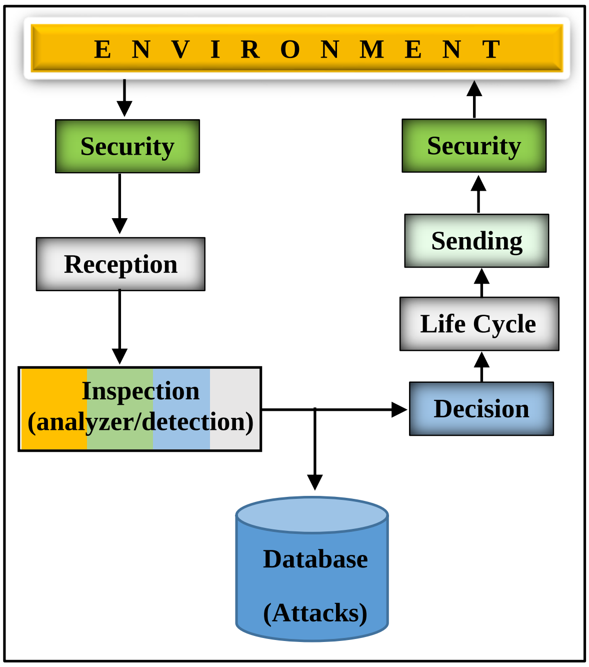


Fig 5.4: Architecture of Inspector Agent

If the node agent is not trusted, the inspector agent can compare its history action with the history actions of its communication partners. The life cycle of an Inspector agent initialized to be active, waiting, suspended, move, and dispose.

5.2.5 Class Diagrams of Model

Here we show the class diagram of our Model, which contains a Node Agent, an Inspector Agent, and a Monitor Agent.

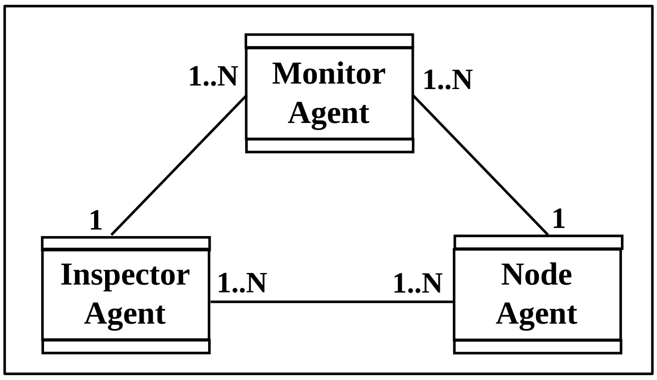


Fig 5.5: Class Diagrams of our Model

5.2.6 Communication Protocol

When a source node wants to send a message to the destination node, it creates a message contains information about source and destination node, table 3 shows the whole structure of this message:

Node	ID_SN	ID_DN	*Data*	Hash
1				
2				
..				
N				

Table 5.3: Structure of the Message

Knowing that:

ID_SN: Is the unique identifier of the Source Node.

ID_DN: Is the unique identifier of the Destination Node.

Data: Is the continent of the message.

Hash: Hash value is useful for verifying the integrity of data sent through the nodes of network. The hash value of sent data of the source node must be compared between the hash value of received data of the destination node to determine whether the data was altered. In our Model, we used Secure Hash Algorithm 1 (SHA-1).

After the formation of the clusters, the node agent of cluster head change its state to Monitor Agent of the cluster. The Monitor Agent creates the inspector agent(s) and sending in different times to all nodes in the cluster. The role of inspector agent is to move from node to another, in each node it collects, analyses, and inspects the behavior of node agent to detect any malicious actions. In other words, the inspector agent works like an IDS at the node level.

When a source node A wants to send data to a destination node B. There are two cases, the first one, where node A and B are in the same cluster. The process of sequence diagram shown in Fig 6: Sequence Diagram (a).

1. Node Agent A (NAA) requests from its Monitor Agent, is the node B trust?
2. Monitor Agent (MoA) sends the trust level of the node B to the Node Agent A, if the trust level of the node B is greater than 'Not Trusted'.
3. NAA encrypted a Message, if the node B is neighbor of node A, we use low technic of encryption ( ), if the node B is not neighbor of A but in the same cluster, we use medium technic of encryption ( ).

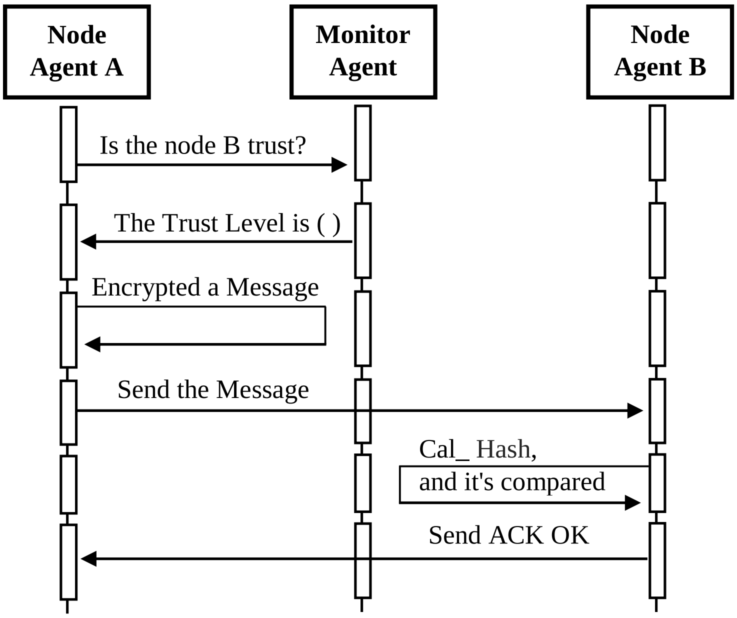


Fig 5.6: Sequence Diagram (a)

4. NAA sends the message to Node Agent B (NAB).
5. NAB accepts the message, calculates the *Hash* to verify the integrity of the message.
6. If the hash is equal then, it is sends to the (NAA) ACK OK. Otherwise, it sends ACK not OK and alert message to Monitor Agent.
7. MoA sends Inspector Agent (IA) suddenly to all nodes of cluster in different times to detect any suspect behaviour. In case MoA receives an alert message, it sends IA directly.

The second case (b), the source node A and destination node D are in different cluster. The process of sequence diagram shown in Fig 7: Sequence Diagram (b)

1. The NAA requests from the Monitor Agent of cluster which continent the node A, is the node D trust?

2. The MoA of the node A searches on the node D in its Confidence Table (CT) and didn't find it. As a result, it sends a request to all its counterparts, the MoA of node D response it and sends to it the trust level of the node D, if it is greater than 'Not Trusted', which resends the response to MoA of the node A.
3. The NAA encrypts message, it sends to its MoA, while the MoA of the node A resends this message to MoA of the node D.
4. The Node Agent D (NAD) accepts the message, calculates the *Hash* to verify the integrity of the message.
5. If the hash is equal then, it is sends to the MoA of the node D ACK OK; the MoA of the node D resends it to Mo A of the node A until it reaches the NAA. Otherwise, it sends ACK not OK and alert message to its Monitor Agent, which resends this warning to all its counterparts in the network.

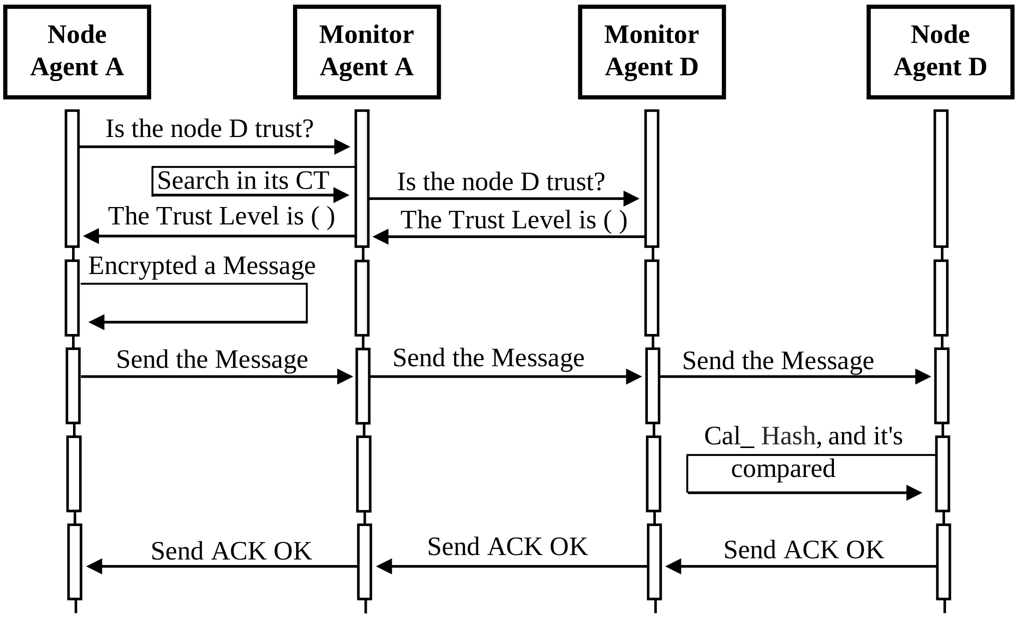


Fig 5.7: Sequence Diagram (b)

5.3 Another Proposition (Contribution 2)

Another proposal will be presented in this section. This proposal is similar to the previous contribution with the changes in the concept of Multi Agent System. The organization of the network is itself like the previous with formulas that calculate the node capability, as well as trust. Here, we have four agents: Node Agent (NA), Monitor Agent (MoA), Ambassador Agent (AmA), and Transporter Agent (TA).

After the formation of the clusters, the node agent of cluster head change its state to Monitor Agent of the cluster. The Monitor Agent creates the Ambassador Agents and sending to all nodes in the cluster. The AmA has almost the same degree of MoA but at the node level, i.e. the Ambassador Agent is like local IDS and IPS (intrusion detection system and Intrusion Prevention System) at the node. Figure 5.8 shows the general architecture of our proposal (Contribution 2).

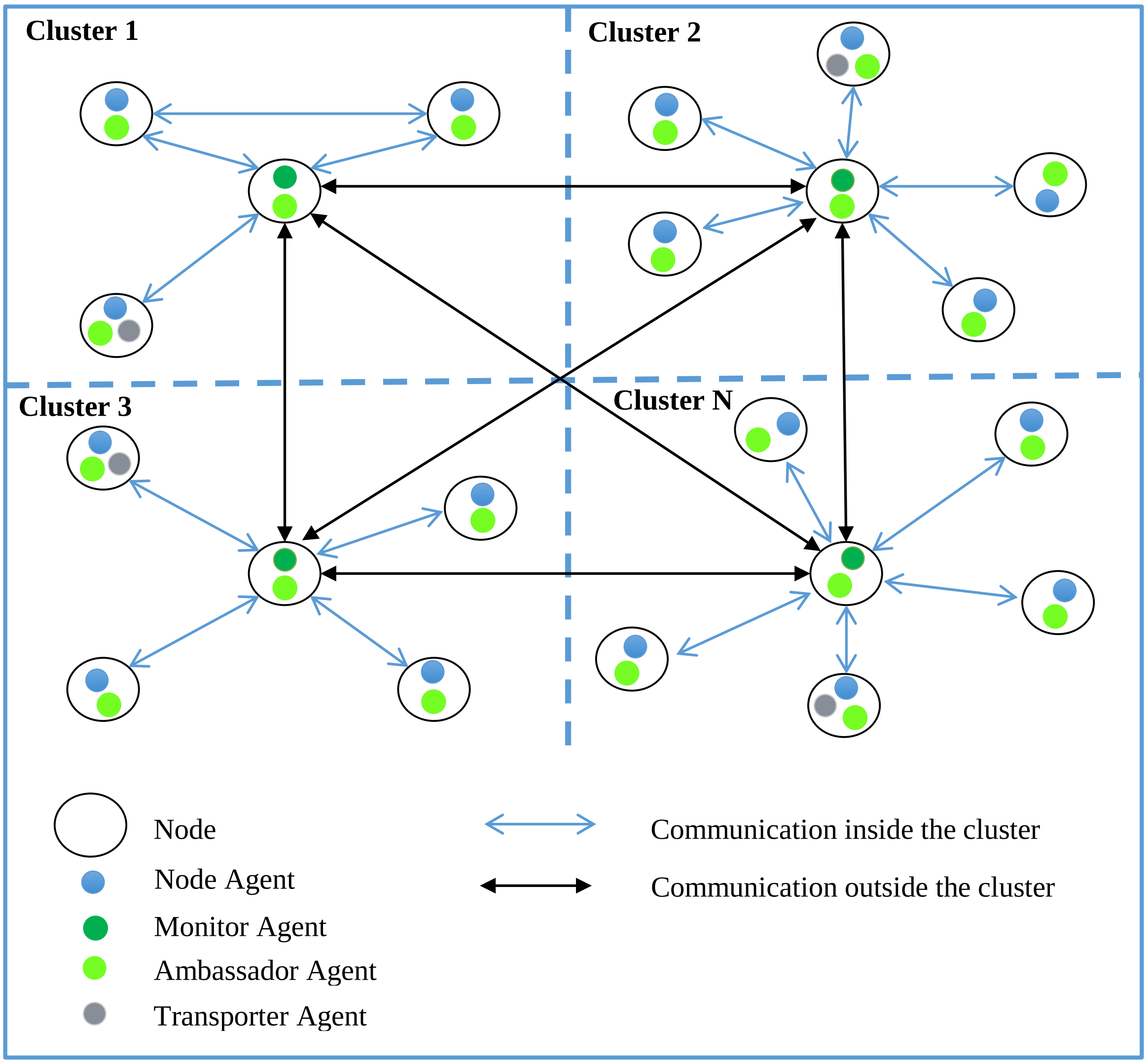


Fig 5.8: Our Proposed Architecture for Security Model based Mobile Agent in MANETs (Contribution 2)

When Node Agent wants to send information to another node, it creates a Transporter Agent (TA) and send it with encrypted information. The agent analyze its data to take a decision. The life cycle of a TA initialized to be active, waiting, suspended, move, and dispose. The following figure illustrates the general architecture of our Model (contribution 2).

5.3.1 Communication Protocol (Contribution 2)

When a source node B wants to send data to a destination node D. There are three cases, the first one, where node B and D are neighbors in the same cluster. To facilitate of explanation, we will neglect the contact between ambassador agent and node agent. The process of sequence diagram shown in Fig 5.9: Sequence Diagram (c).

1. Ambassador Agent B (AmAB) request from the Ambassador Agent D (AmAD), is the node D trust?
2. Ambassador Agent D (AmAD) sends the trust level of the node D to the Ambassador Agent B (AmAB) if the trust level of the node D is greater than ‘Not Trusted’.
3. Ambassador Agent B (AmAB) gives the permission to the Node Agent B (NAB) for sends its data.
4. Node Agent B (NAB) creates Transporter Agent B (TAB) attaches by a Message Signature (MS) [ID_node B, ID_node D, ID_Transporter Agent B (TAB), and Hash] and it gives them to the AmAB, which resends to the AmAD.

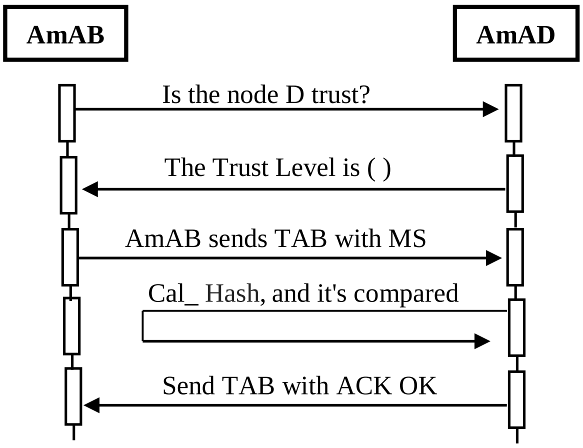


Figure 5.9: Sequence Diagram (c)

5. Ambassador Agent D (AmAD) accepted the Transporter Agent B (TAB), calculated the hash that arrived by it and compares, if the hash is equal then, it is gives to the (TAB) ACK OK, and the (TAB) is returns to node B. Otherwise, it will reject the (TAB) and sends alert message to its control agent.

The second case (d), the source node B and destination node D are not neighbors, but at the same cluster, the process of sequence diagram was shown in Fig 5.10: Sequence Diagram (d).

1. The (AmAB) request from its Monitor Agent (MoA), is the node D trust?
2. The Monitor Agent (MoA) sends the trust level of the node D to the (AmAB) if the trust level of the node D is greater than ‘Not Trusted’.

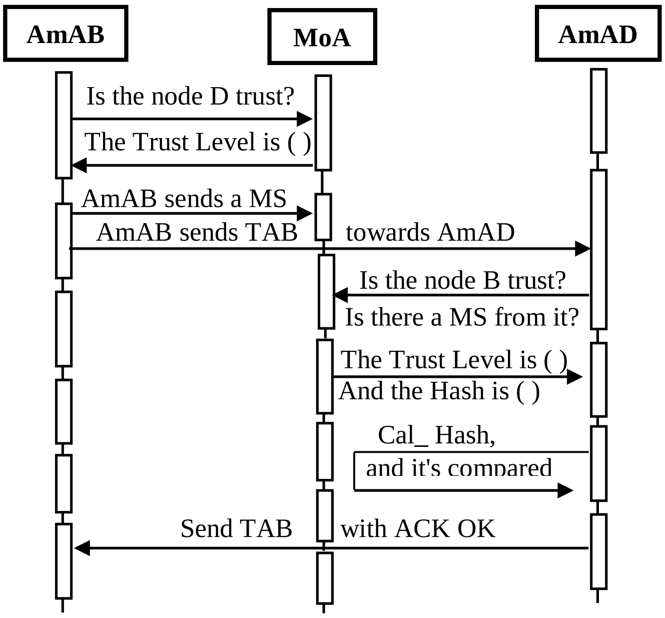


Figure 5.10: Sequence Diagram (d)

3. Ambassador Agent B (AmAB) sends to its Monitor Agent (MoA) a Message Signature (MS) [ID_node B, ID_node D, ID_Transporter Agent B (TAB), and Hash].
4. The (AmAB) sends its Transporter Agent B (TAB) towards node D.
5. The (AmAD) request from its MoA, is the node B trust?, and is there a (MS) from it?
6. The Monitor Agent of the node D sends the trust level of node B and the hash to the (AmAD).
7. The (AmAD) accepted the (TAB), calculated the hash, which arrived by it and compares, if the hash is equal then, it is gives to the (TAB) ACK OK, and the (TAB) is returns to node B. Otherwise, it will reject the (TAB) and sends alert message to its MoA.

The third case (f), the source node B into cluster while the destination node D into another cluster, the process of sequence diagram was shown in Fig 5.11: Sequence Diagram (f).

1. The (AmAB) request from its MoA, is the node D trust?
2. The Monitor Agent of the node B searches on the node D in its Confidence Table (CT), it did not find.
3. The Monitor Agent of the node B sends a request to all its counterparts, a Monitor Agent of the node D response it and sends to it the trust level of the node D, if it is greater than ‘Not Trusted’, which resends the response to (AmAB).
4. The (AmAB) sends to its MoA a (MS), while the MoA resends a MS to MoA of the node D.

5. The (AmAB) sends its TAB towards AmAD attaches by a (MS).
6. The (AmAD) request from its MoA, is the node B trust? and is there a (MS) from it?.

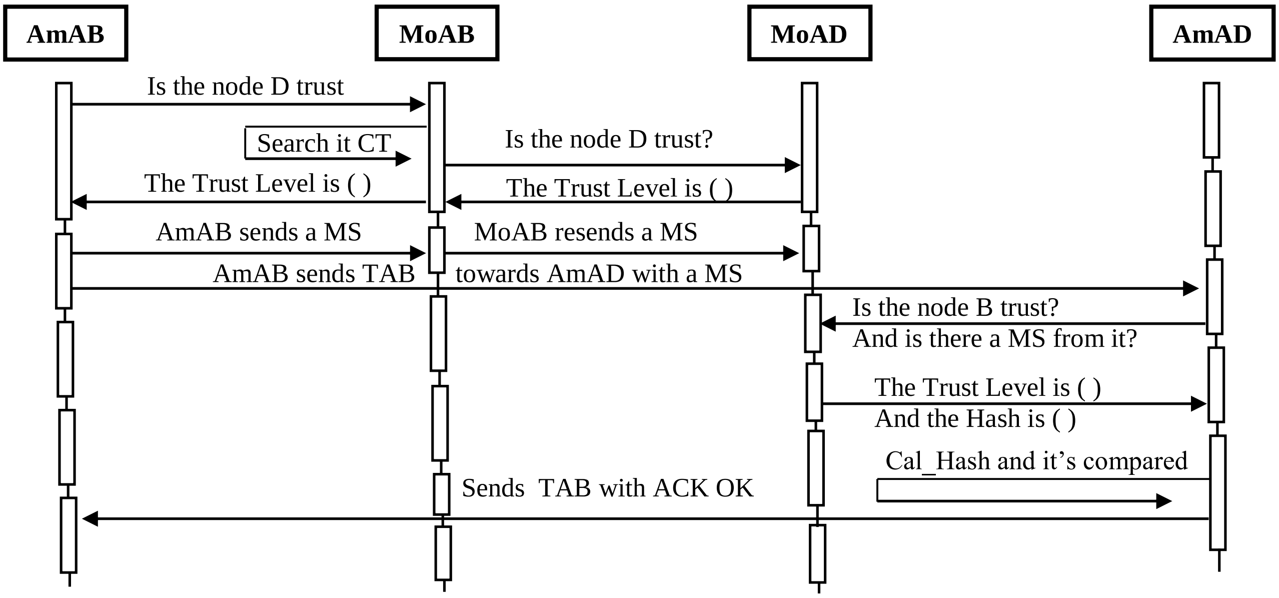


Figure 5.11: Sequence Diagram (f)

7. The MoA of the node D sends the trust level of node B and the hash value to AmAD.
8. The (AmAD) accepted the (TAB), calculated the hash, which arrived by it and compares, if the hash is equal then, it is gives to the (TAB) ACK OK, and the (TAB) is returns to node B. Otherwise, it will reject the (TAB) and sends alert message to its MoA.

5.4 Another Proposition (Contribution 3)

Another proposal will be presented in this section. This proposal is considered as a hybrid approach between the two previous (contribution 1 and 2). The organization of the network is itself like the previous with formulas that calculate the node capability, as well as trust. Here, we have four agents: Node Agent (NA), Manager Agent (ManA), Ambassador Agent (AmA), and Inspector Agent (IA).

The node agent that elected cluster head, it transforms itself to *Manager Agent*. This agent is responsible for all operations within the cluster and outside with counterparts. The manager agent maintains a table of confidence it contains the necessary information for the trustworthiness and authentication of each node in the cluster. The figure 5.12 illustrates the general architecture of our proposal (Contribution 3).

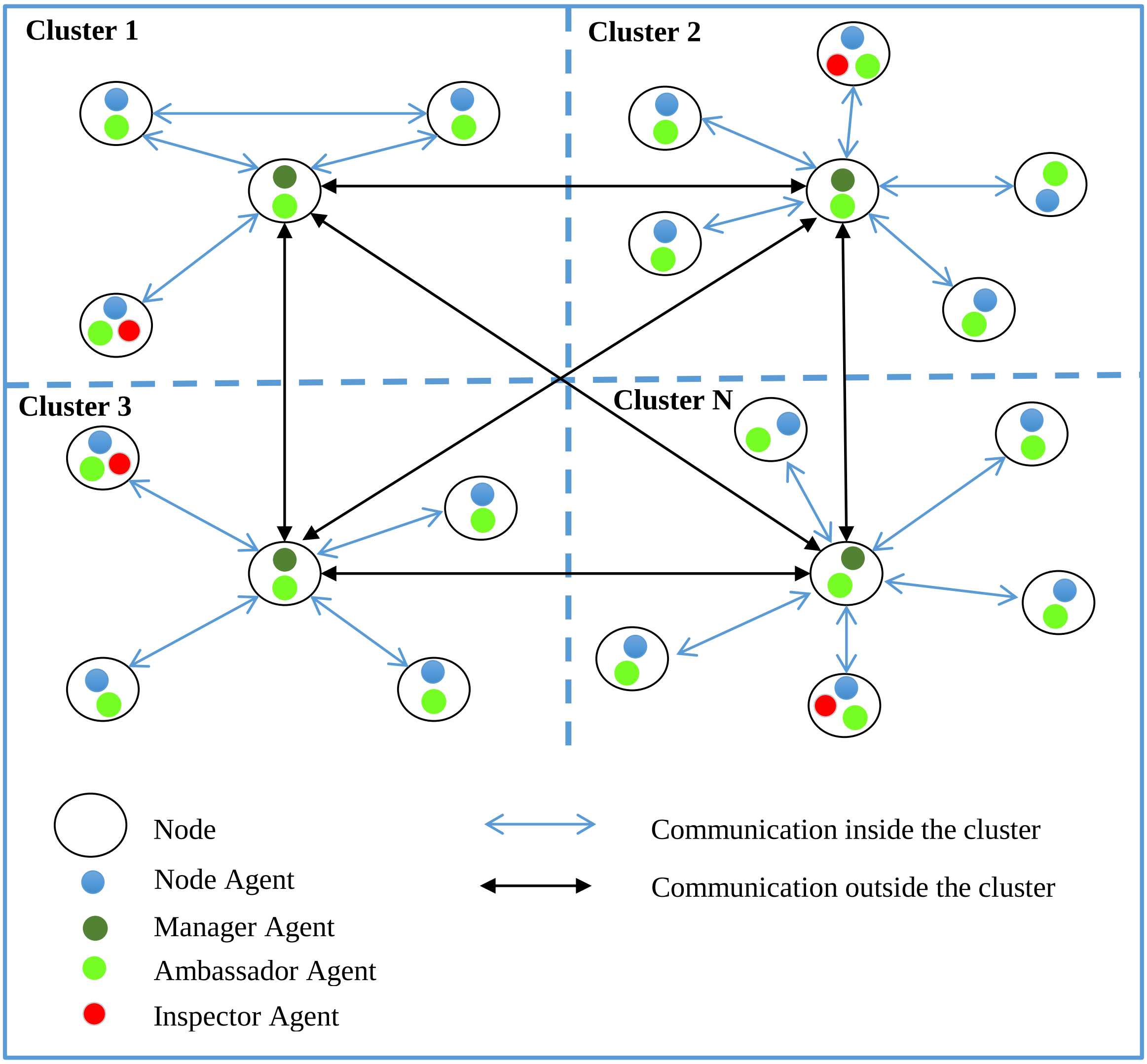


Fig 5.13: Our Proposed Architecture for Security Model based Mobile Agent in MANETs (Contribution 3)

In the following figure, which illustrates the internal architecture of the Manager Agent. A manager component act as a connector between the other components, the regulator and the crossing point obliged to call all of the component method. It also provides primitives change components. The manager agent creates the mobile agent like ambassador agent which sends it to all node in the cluster and the inspector agent that dispatch at specific intervals in order to sudden inspection process. The inspector agent after the return to manager agent carrying with it a report of the status of each node. In the case where there is an attack or an act suspicious, the manager agent launches a warning message to all nodes within and outside of the group. In parallel, it excites a maintenance operation, where it tries to kill the attack, exclusion of the cluster and the network, or repair if it is possible.

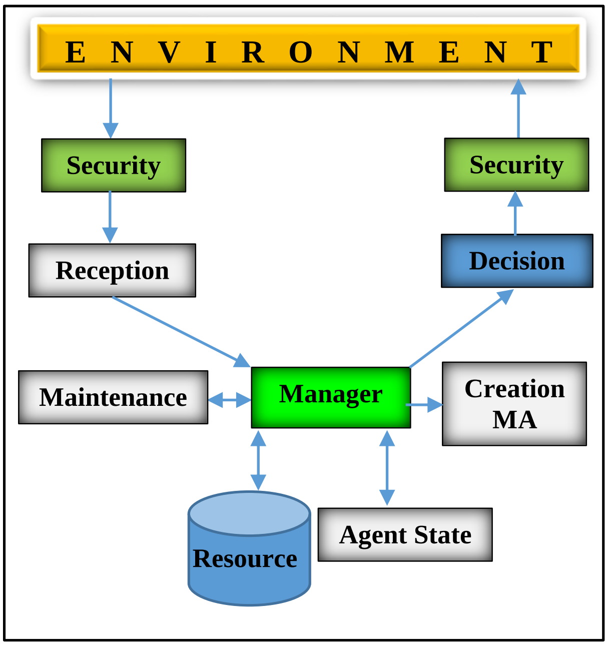


Figure 5.12: Architecture of Manager Agent

5.4.1 Class Diagrams of Model (Contribution 3)

Here we show the class diagram of this approach, which contains a Node Agent, an Ambassador Agent, an Inspector Agent, and a Manager Agent.

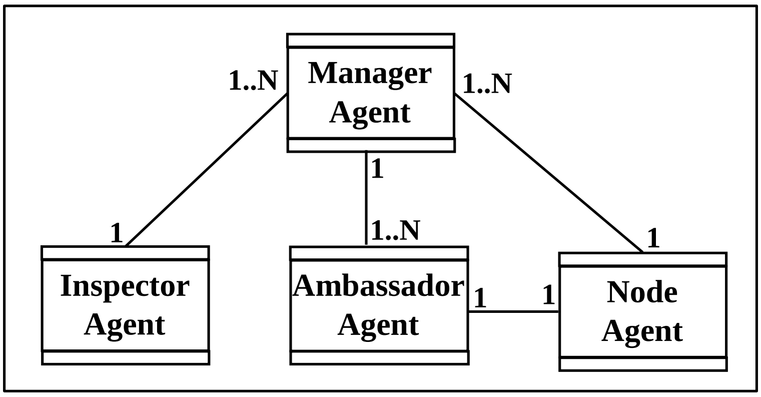


Figure 5.14: Class Diagrams of our Model (Contribution 3)

The communication protocol is the same as the previous. In addition, if Manager Agent receive an alert message from a node, it creates (IA) and sends it urgently for detect the problem and return in order to take appropriate action. In the general case, a Manager Agents of each cluster creates (IA) and sends it at different times to each nodes of cluster in form surprised.

5.5 Model Properties

- Our model is dedicating to improving the security service in ad hoc networks and specifically for secure communication with Quality of Service, which is the main problem in this type of network.
- Exploiting easy network, for this, we compose the network into a set of levels to get a hierarchical view, which will make the deployment of model taking into account the security service faster and easier.
- The tasks allocated to the terminals in an unfair way according to the heterogeneous capacities, this implies an energy saving and take advantage of competence of a node.
- We use the mobile agent technology that came from the field of artificial intelligence who gave significant benefits compared to other technologies and more suitable for wireless environments. For this, we associate with each terminal, an agent, which introduces a degree of intelligence in each node of the network. In addition, the use of the mobile agent technology overcomes the problems associated with the disconnection of nodes (Tolerance to Defects), and many more advantage.
- Network terminals are heterogeneous. Therefore, we break down the security problem between the node, cluster, network, and cooperate among themselves to ensure strong security.
- We have gain the benefits of both paradigms "agent" by the internal architecture and "component" that offer flexibility to our model, and therefore, it can give a good operation for the security function with quality of service.
- In comparison of this model with others works described in the previous chapter. We note that each work deals almost one security goal, while this model treated more security goals.

5.6 Conclusion

In this chapter, we presented the *"Security Model based Mobile Agent for Mobile Ad Hoc Networks*". This model defines which types of security service can be provided in an ad hoc network and some mechanisms used to provide these services. Using cluster topology provides a hierarchical view of the network to best manage the security associated with each level. It takes into consideration more objective security. However, it is very difficult to guarantee certain goals of security because the constraints imposed by this type of network, in particular: the mobility of the nodes, the dynamic and variable topology, and the heterogeneity of the terminals.

The emphasis in this model is on the efficient management of resources in order to improve the security in a network where its topology can vary in a fast and random way. In addition, we have modeled the agent by a component assembly in order to facilitate its implementation, thus benefiting from the advantages of the component paradigm. Furthermore, we presented the class diagram of our model, the communication protocol between agents. Finally, we finished the chapter by the property of this model.

Chapter 6

Experimentation and Performance Evaluation

When I examine myself and my methods of thought, I come to the conclusion that the gift of fantasy has meant more to me than my talent for absorbing positive knowledge.

Albert Einstein

This chapter describes the implementation of our model, where we used a platform for developing mobile agents published by IBM called Aglets, accompanied with java development kit (JDK 7) and the NetBeans IDE version 8.0.2. These tools are installed on a computer running windows 7 system equipped by Intel core i7 processor. For testing the implementation, we used ad hoc network include four machines, such as every machine is configured to run the Aglet Agents. We examined at several scenarios of model, and discuss patterns (and associated implications) for how they could be increased the level of security in the ad hoc network, without effect it performance. Based on the obtained results, we can summarize that the implementation of our model satisfy the main objectives of the security.

6.1 Introduction

After detailing the concepts of our security model based mobile agent for ad hoc networks and discussing the internal architecture of the mobile agent, we describe in this chapter, its implementation has been performed using the agent platform Aglets under the JDK 7 and NetBeans IDE version 8.0.2 environment.

Generally, the program of our model consists of four steps: the first is evaluates node resources such as the Trust Level, CPU load and memory, degree of the node, and an energy level. The second step will be the construction of the groups with the election (i.e. determine the cluster head and members. The third step is operation between agents in or out of the clusters. The last step is the measurement of the efficiency to our model to function properly, control and detection the various attacks, and take appropriate solutions in case to detect any types of attacks.

6.2 Implementation Tools

6.2.1 Programming Language

We chose to implement our model the Java language because Java is an object oriented programming language the high level, easy to use. It is a multi-platform language, which make it a choice of languages for programming a mobile code. It has some characteristics that led us to develop our system by using it, these features are:

- Java overcomes the platforms: Opposite to many compilers, the Java translates source code into the language of a Java Virtual Machine (JVM). The Product code called bytecode is assigned to an interpreter, which reads and executes it. This is the key principle that enables to the mobile agent to adapt in all execution environments based in the implementation of task on an interpretive language such as Java [145]. In brief, Java works in interpreted mode as opposed to compiled languages, it is can run on multiple operating systems, offers the possibility to create simple application of multi tasks suitable for network, and on anywhere platform.
- Java is multithreaded: The using an autonomous agent requires the use of a language for the competition and parallelism. This language must also include a synchronization management system. As we said in the second chapter an agent is a standalone program still waiting, and ready to respond to changes in its environment, therefore, the use of a multi-threaded language be required.

- Java is a strongly typed language, so very safe. It simultaneously offers a relative flexibility. In addition, Java provides us with multiple levels of data protection, as well as a mechanism for handling sophisticated exceptions. Moreover, it was developed in the interests of maximum security. The idea that a program containing errors should not be compiled. Thus, errors are unlikely to escape of the programmer. In fact, the Java program does not need to know all the classes at compile time. This possibility is very useful for dynamic loading of tasks contained in the library of the mobile agent. Furthermore, Java has a rich library that contains many security libraries among which we mention javax.security, javax.CryptoPackage or javax.crypto.

6.2.2 Platform Aglets

The criteria for choosing a mobile agent platform are naturally linked with the needs of the model proposed, and its characteristics. Thus, the platform must answer the following requirements:

- The mobility of the Platform Aglets is flexible and robust.
- The underlying programming language should provide adequate security enough features such as cryptographic functions and hash.
- A free platform.

Aglets is the nearest platform of our criteria, we used it to implement the architecture of the mobile agent. It is a platform of agents developed by Big Blue (IBM) in Java language. Environments are provided on hosts by specialized servers, which understand the aglet transfer protocol ATP and provide security and other services. The Aglet distribution is provided with such a server, called Tahiti. The design of aglets is modeled on that of Java applets. The word "**Aglet**" is a contraction of "**Agent**" and "**Applet**" [125]. Aglets renamed in 1998 ASDK (Aglets Software Development Kit). The ASDK is intended to facilitate the creation of mobile agent.

6.2.2.1 Architecture of Aglets

The aglet object model defines a set of techniques to create mobile agents in an extended type of network. The main elements are [146, 147, 148, 149, 150, 151]:

- Aglet**:** Is a java-based autonomous software agent. Major buzzwords that can be used to characterize an Aglet are written in pure java, light-weight object migration, built with persistent support, event-driven. It visits local and remote hosts, and reacts to events and messages. The Java aglet extends the model of network-mobile code made famous by Java applets. An aglet requires a host Java application, an aglet host, to be running on a computer before it can visit that computer.

- Proxy**:** A proxy is a representative of an aglet. It serves as a shield for the aglet that protects the aglet from direct access to its public methods. The proxy also provides location transparency for the aglet; that is, it can hide the aglet's real location of the aglet.

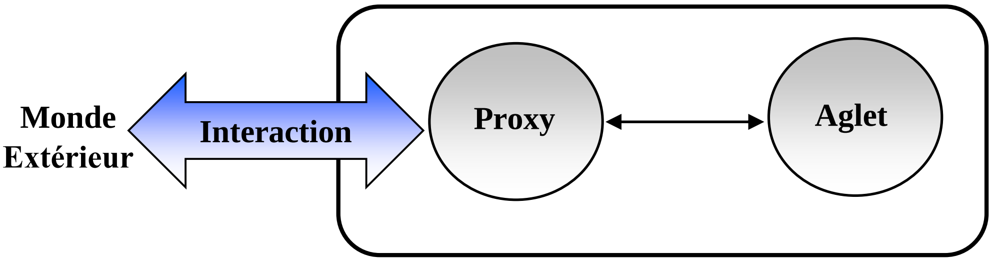


Figure 6.1: Relationship between Aglet and Proxy

- Context: A context is an aglet's workplace; it is a stationary object provides a means for maintaining and managing active aglets in a uniform execution environment where the host system is secured against malicious aglets.
- Host: A host is a machine capable of hosting multiple contexts. The host is generally a node in a network.
- Message: A message is an object exchanged between aglets. It allows for synchronous as well as asynchronous message-passing between aglets. Message-passing can be used by aglets to collaborate and exchange information in a loosely coupled fashion. A message manager allows concurrency control of incoming messages.

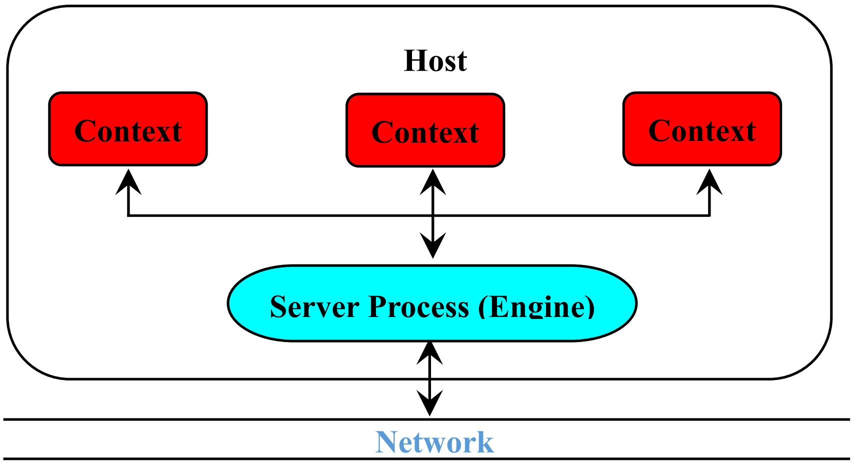


Figure 6.2: Relationship between Host, Server Process, and Contexts

- Itinerary: An itinerary is an aglet's travel plan. It provides a convenient abstraction for non-trivial travel patterns and routing.
- Identifier: An identifier is bound to each aglet. This identifier is globally unique and immutable throughout the lifetime of the aglet.

6.2.2.2 Life-Cycle of Aglets

The types of behavior of the Aglets have been implemented in a way to respond of the main needs of mobile agent. The main operations affecting the life of Aglets are:

- Creation: An aglet is created within a context. The new aglet is assigned an identifier, inserted into the context, and initialized. The aglet starts to execute as soon as it has been initialized.
- Cloning**:** The cloning of an aglet produces an almost identical copy of the original aglet in the same context. The only differences are the assigned identifier and the fact that execution restarts in the new aglet. Note that execution threads are not cloned.
- Dispatching: An aglet from one context to another will remove it from its current context and insert it into the destination context, where it will restart execution. We say that the aglet has been \pushed" into its new context. The retraction of an aglet will \pull" (remove) it from its current context and insert it into the context from which the retraction was requested.
- Retraction: The retraction of an aglet will pull (remove) it from its current context and insert it into the context from which the retraction was requested.

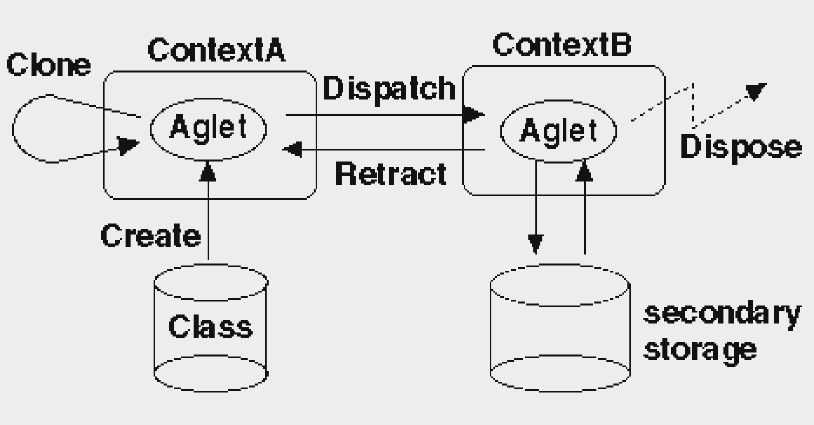


Figure 6.3: Life-Cycle of Aglets

- Disposal**:** The destruction of an Aglet, will stop in its current execution and remove it from its current context.

- Activation/deactivation**:** The deactivation of an aglet is the ability to temporarily halt its execution and store its state in secondary storage. Activation of an aglet will restore it in a context.

6.2.3 Programming Environment

Integrated Development Environment (IDE) is a program combining a text editor, compiler, automatic creation tools, and often debugger. In Java, there are several IDE such as NetBeans (Sun), JCreator or Eclipse (IBM). The IDE that we will use must to be extensible, universal, versatile, free and compliant to the platform Aglets chosen. Our choice fell on NetBeans because it respond all criteria listed.

NetBeans was created in Sun Microsystems initiative. It has all the characteristics necessary for a quality environment, whether to develop in Java, Ruby, C / C ++ or even PHP. NetBeans is under OpenSource license, it can develop and deploy quickly and free Swing GUI applications, Applets, JSP / Servlets, J2EE architectures in a highly customizable environment. NetBeans is a complete environment including all the features of development and all Java-related technologies for quick and visual development of Java applications [152].

6.3 Implementation and Tests

In order to implement our Model we used a platform for developing mobile agents published by IBM called Aglets, accompanied with java development kit (JDK 7) and the NetBeans IDE version 8.0.2. For testing the prototype, we are using ad hoc network consisted of four node (laptop), where every node is configured to run the Aglet Agents.

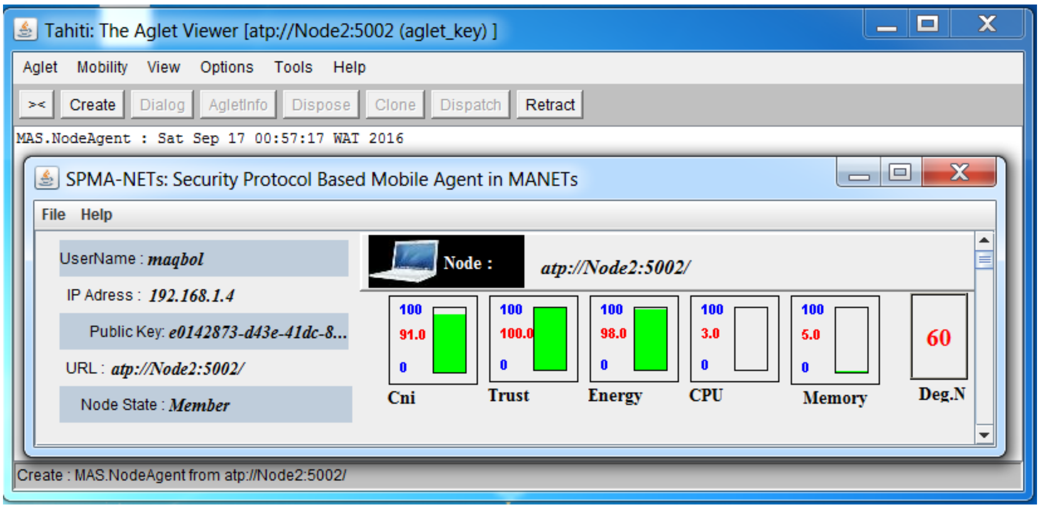
6.3.1 The First Scenario (Contribution 1)

Note1: We assume the values of coefficient (a, b, c, d, e) as follows: a = 0.5, b = 0.3, c = 0.2, d = - 0.05, e = - 0.05.

Note2: In the initial state, we gave the node that called maqbol and URL: atp://Node2:5002, the value =100 (Fully Trusted) of the trust level, while the other nodes takes the value = 59 (Average) of the trust level.

Note3: The degree of node = 60, i.e. the node has three neighbors, while the value = 40, i.e. the node has two neighbors.

The following figure illustrated an example of initial state of the node, the Tahiti of the Aglets platform where the node agent created and calculated the value of C_{ni}.


Tahiti: The Aglet Viewer [atp://Node2:5002 (aglet_key)]
Aglet Mobility View Options Tools Help
>< Create Dialog AgletInfo Dispose Clone Dispatch Retract
SPMA-NETs: Security Protocol Based Mobile Agent in MANETs
File Help
Node :
100 91.0 0 Cni
100 100.0 0 Trust
100 98.0 0 Energy
100 3.0 0 CPU
100 5.0 0 Memory
60 Deg.N

MAS.NodeAgent : Sat Sep 17 00:57:17 WAT 2016

UserName : *maqbol*

IP Adress : *192.168.1.4*

Public Key: *e0142873-d43e-41dc-8...*

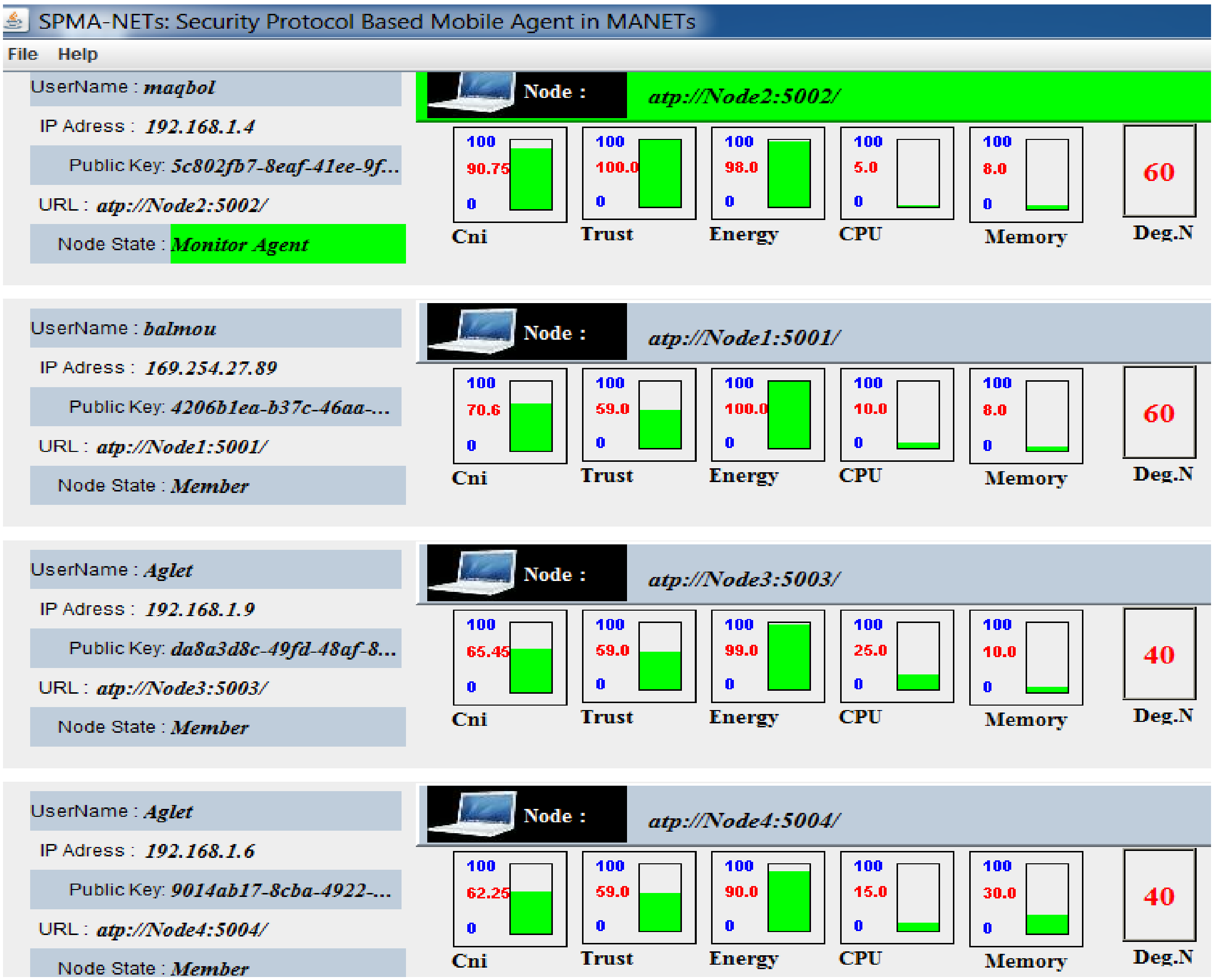
URL : *atp://Node2:5002/*

Node State : *Member*

Node : *atp://Node2:5002/*

Create : MAS.NodeAgent from atp://Node2:5002/

Fig 6.4: Illustrated an Example of Initial State of the Node



UserName : *maqbol*

IP Adress : *192.168.1.4*

Public Key: *5c802fb7-8eaf-41ee-9f...*

URL : *atp://Node2:5002/*

Node State : *Monitor Agent*

Node : *atp://Node2:5002/*

UserName : *balmou*

IP Adress : *169.254.27.89*

Public Key: *4206b1ea-b37c-46aa-...*

URL : *atp://Node1:5001/*

Node State : *Member*

Node : *atp://Node1:5001/*

UserName : *Aglet*

IP Adress : *192.168.1.9*

Public Key: *da8a3d8c-49fd-48af-8...*

URL : *atp://Node3:5003/*

Node State : *Member*

Node : *atp://Node3:5003/*

UserName : *Aglet*

IP Adress : *192.168.1.6*

Public Key: *9014ab17-8cba-4922-...*

URL : *atp://Node4:5004/*

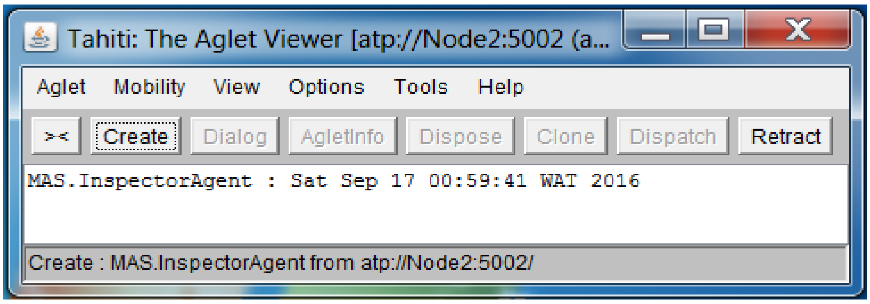
Node State : *Member*

Node : *atp://Node4:5004/*

Fig 6.5: Explained the Election Process

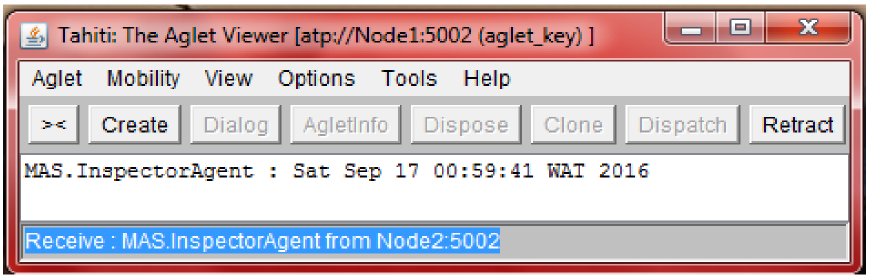
The fig 6.5 at the top indicates the process of election between four ad hoc nodes where the node that called maqbol and URL: atp://Node2:5002 elected as Monitor Agent because it has the more capacity C_{ni} that equal = 90.75%. While others nodes takes the Member states.

After the election process, the Monitor agent creates the Inspector Agent as in fig 6.6, dispatch it to any node, which travels from one node to another for detect any attack or suspicious behavior, and it then returns to the original node (see the following figures respectively.


Tahiti: The Aglet Viewer [atp://Node2:5002 (a...
Aglet Mobility View Options Tools Help
>< Create Dialog AgletInfo Dispose Clone Dispatch Retract

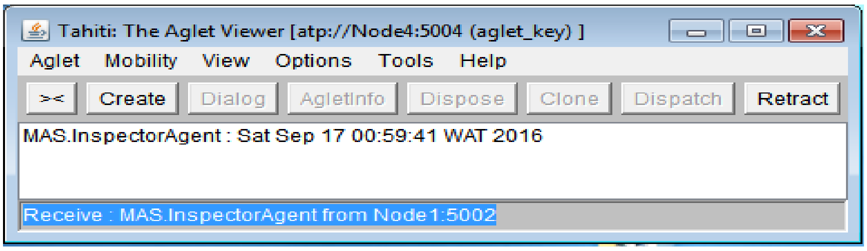
```
MAS.InspectorAgent : Sat Sep 17 00:59:41 WAT 2016
Create : MAS.InspectorAgent from atp://Node2:5002/
```

Fig 6.6: Shows the Creation of Inspector Agent


Tahiti: The Aglet Viewer [atp://Node1:5002 (aglet_key)]
Aglet Mobility View Options Tools Help
>< Create Dialog AgletInfo Dispose Clone Dispatch Retract

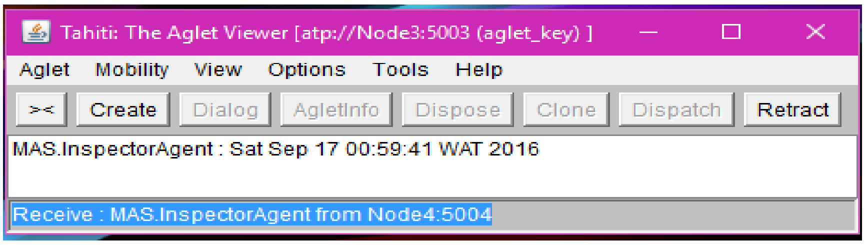
```
MAS.InspectorAgent : Sat Sep 17 00:59:41 WAT 2016
Receive : MAS.InspectorAgent from Node2:5002
```

Fig 6.7: Shows the Travels of Inspector Agent from Node2 to Node1


Tahiti: The Aglet Viewer [atp://Node4:5004 (aglet_key)]
Aglet Mobility View Options Tools Help
>< Create Dialog AgletInfo Dispose Clone Dispatch Retract

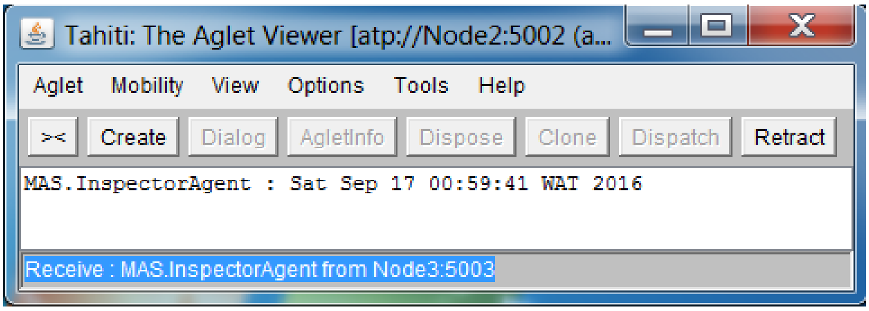
```
MAS.InspectorAgent : Sat Sep 17 00:59:41 WAT 2016
Receive : MAS.InspectorAgent from Node1:5002
```

Fig 6.8: Shows the Travels of Inspector Agent from Node1 to Node4


Tahiti: The Aglet Viewer [atp://Node3:5003 (aglet_key)]
Aglet Mobility View Options Tools Help
>< Create Dialog AgletInfo Dispose Clone Dispatch Retract

```
MAS.InspectorAgent : Sat Sep 17 00:59:41 WAT 2016
Receive : MAS.InspectorAgent from Node4:5004
```

Fig 6.9: Shows the Travels of Inspector Agent from Node4 to Node3

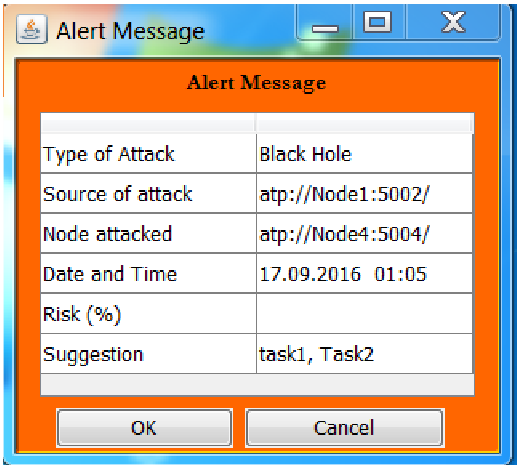


```
MAS.InspectorAgent : Sat Sep 17 00:59:41 WAT 2016
```

Receive : MAS.InspectorAgent from Node3:5003

Fig 6.10: Shows the Returns of Inspector Agent from Node3 to Node2

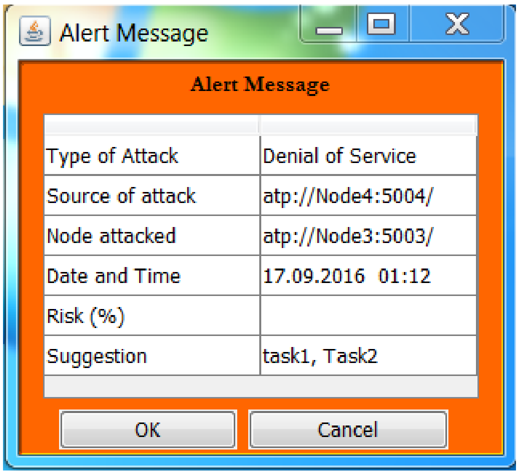
In our experimental results, shows the proposed Model is expected to perform better in all situations. For example, in the first test, we tested our Model that detect Black Hole Attack as in figure 13.



Type of Attack	Black Hole
Source of attack	atp://Node1:5002/
Node attacked	atp://Node4:5004/
Date and Time	17.09.2016 01:05
Risk (%)	
Suggestion	task1, Task2

Fig 6.11: Illustrates the Detection of Black Hole Attack

In the second test, it is succeed to detect Denial of Service attack as in figure 14. The Inspector Agent sends or delivers to the Monitor Agent an alert message contains the necessary information such as: Type of Attack, Source of attack, Node attacked, Date and Time, Percentage of Risk, and Suggestion.

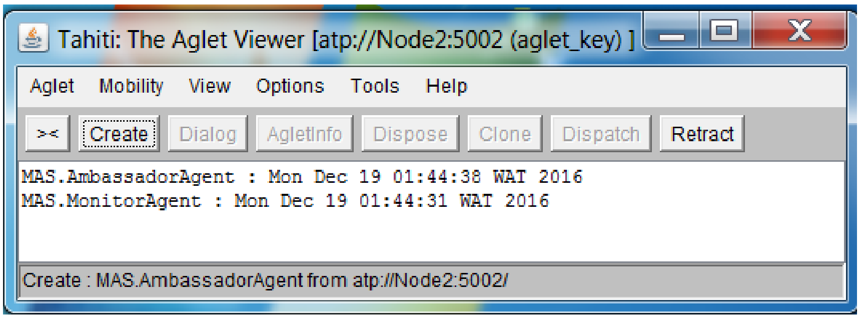


Type of Attack	Denial of Service
Source of attack	atp://Node4:5004/
Node attacked	atp://Node3:5003/
Date and Time	17.09.2016 01:12
Risk (%)	
Suggestion	task1, Task2

Fig 6.12: Illustrates the Detection of Denial of Service Attack

6.3.2 The Second Scenario (Contribution 2)

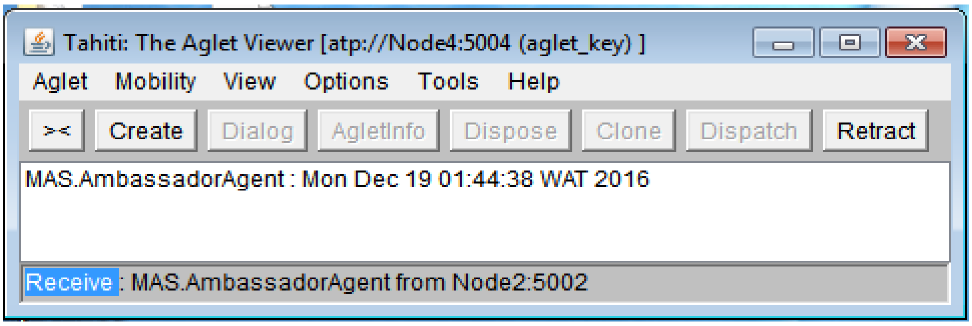
The hypothesis is same as the previous that we have seen in the first scenario. The node agent evaluates the resources of the node. After the election process to determine the monitor agent and the members, each monitor agent creates the Ambassador Agent (s) as in fig 6.13, dispatch it to any node inside of cluster, which reside in node for detect any attack or suspicious behaviour.


Tahiti: The Aglet Viewer [atp://Node2:5002 (aglet_key)]
Aglet Mobility View Options Tools Help
>< Create Dialog AgletInfo Dispose Clone Dispatch Retract

```
MAS.AmbassadorAgent : Mon Dec 19 01:44:38 WAT 2016
MAS.MonitorAgent : Mon Dec 19 01:44:31 WAT 2016
Create : MAS.AmbassadorAgent from atp://Node2:5002/
```

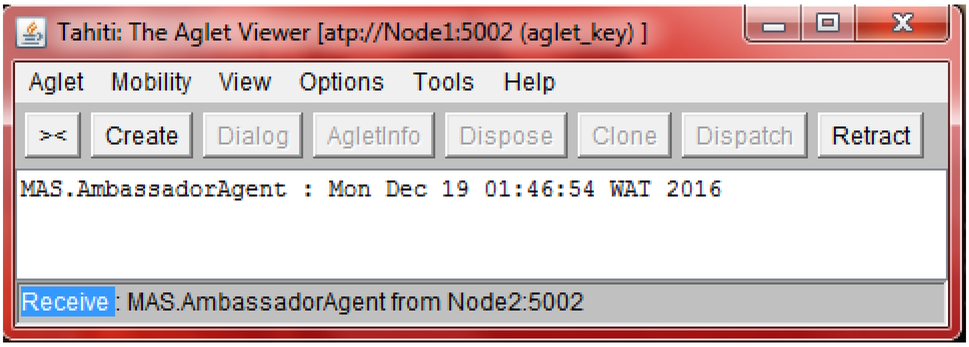
Fig 6.13: Shows the Creation of Ambassador Agent

The following figures 6.14, 6.15 illustrate respectively the reception of the ambassador agents, which are sent almost simultaneously by Monitor Agent, i.e. those nodes that have the URL(s): atp://Node4:5004 and atp://Node1:5002 receive ambassador agents that sent by node that has the URL: atp://Node2:5002 (Monitor Agent).


Tahiti: The Aglet Viewer [atp://Node4:5004 (aglet_key)]
Aglet Mobility View Options Tools Help
>< Create Dialog AgletInfo Dispose Clone Dispatch Retract

```
MAS.AmbassadorAgent : Mon Dec 19 01:44:38 WAT 2016
Receive : MAS.AmbassadorAgent from Node2:5002
```

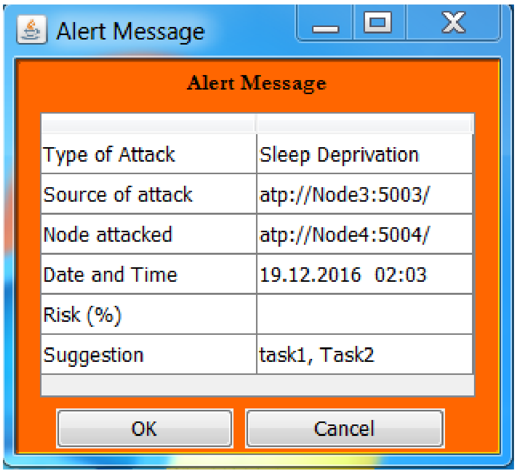
Fig 6.14: Shows Node4 Receives Ambassador Agent From Node2


Tahiti: The Aglet Viewer [atp://Node1:5002 (aglet_key)]
Aglet Mobility View Options Tools Help
>< Create Dialog AgletInfo Dispose Clone Dispatch Retract

```
MAS.AmbassadorAgent : Mon Dec 19 01:46:54 WAT 2016
Receive : MAS.AmbassadorAgent from Node2:5002
```

Fig 6.15: Shows Node1 Receives Ambassador Agent From Node2

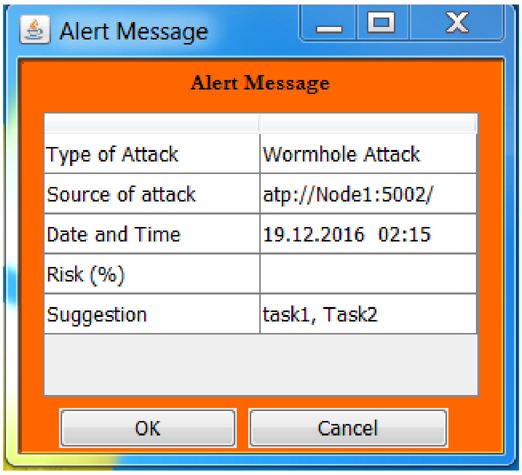
In this scenario, and after several experiments, this model could to discover these following attacks (see figure 6.16, 6.17).



Alert Message

Type of Attack	Sleep Deprivation
Source of attack	atp://Node3:5003/
Node attacked	atp://Node4:5004/
Date and Time	19.12.2016 02:03
Risk (%)	
Suggestion	task1, Task2

Fig 6.16: Illustrates the Detection of Sleep Deprivation Attack



Alert Message

Type of Attack	Wormhole Attack
Source of attack	atp://Node1:5002/
Date and Time	19.12.2016 02:15
Risk (%)	
Suggestion	task1, Task2

Fig 6.17: Illustrates the Detection of Wormhole Attack

6.4 Conclusion

In this chapter, we have specified development tools so justified their selection criteria in order to help developers for the implementation of our security Model. In order to prove the validity of the proposed Model, implementation and experimentation work has been conducted using the Aglet platform. The results show that our model raises the level of security in a Mobile Ad Hoc Networks, without affecting it performance; however, there are some issues, which may addressed in future works.

Chapter 7

Conclusion and Outlook

I have offended God and mankind because my work didn't reach the quality it should have

Leonardo da Vinci

Many security mechanisms have been proposed for wireless networks. Although these mechanisms could have responded a set of security requirements, they remain only effective in a specific context related to the assumptions and restrictive requirements that were issued during the design. The main subject of this thesis is to provide a solution to problems related to security in wireless networks and especially of the ad hoc networks. We tried to understand the behavior and the specifics of this kind of networking, as well as the various problems associated with security management in this environment.

The mobile agent system is a very promising paradigm that has already established its presence in many applications including wireless networks. At the same time, this technology has introduced some security problems and emphasized some existing security issues. In this thesis, we surveyed the main issues in the security of mobile agents. We considered both the mobile agent and the agent platform points of view. We discussed the security threats and requirements that need to be met in order to alleviate those threats.

We presented the most important techniques for providing security in mobile agent systems. Some of those techniques, for example Sandboxing, Code Signing, safe Code Interpretation, State Appraisal, etc. Furthermore, we presented the security techniques of the mobile agent against the attacks that are launched by malicious platforms. Some of those techniques, for example Execution Tracing, Co-Operating Agents, Obfuscated Code, Partial Result Encapsulation, etc. None of the existing techniques provides an optimal solution for all scenarios. In any case, more research is needed in the future to warrant sufficient trust in mobile agent technology by a wide range of users.

In this thesis, we present our novel and effective model that we have devised and titled "*Security Model based on Mobile Agent for Mobile Ad Hoc Networks"* that aims to improve the level of security. This model used on a network topology based on the concept of clusters and mobile agent technology. i.e. the network organization at three levels (node level, cluster level, and network level) for hierarchical management of the security services. We are used an optimization function of five parameters to evaluate node resources and the formula to estimate the trust ability of the node.

In the first contribution, we define three agent types. The Node Agent (NA) manages the use of node resources. The Monitor Agent created in the most trusted with best resources node to control the communication inside and outside the cluster. The Monitor Agent (MoA) creates the Inspector Agent (IA), which travails from node to another to examine the actions history of each node agent to detect any suspect behavior, and returns to MoA with report shows the status of each node in the cluster.

In the second contribution, we proposed four types of agents. Node Agent (NA), Monitor Agent (MoA), Ambassador Agent (AmA), and Transporter Agent (TA). The Monitor Agent creates the Ambassador Agents and sending to all nodes in the cluster. The Ambassador Agent is like local IDS and IPS (Intrusion Detection System and Intrusion Prevention System) at the node.

The third contribution is considered a hybrid approach for the two preceding. We gave the architecture interne of mobile agent, class diagrams, and communication protocol for our model.

To implement the proposed model, we choose to use the Aglet platform, because it is appropriate for developing mobile agent. Based on the obtained results, we can summarise that the implementation of our Model satisfy the main objectives of the security.

- **Authentication**: Where we used the Monitor Agent after the election process as trusted site.
- **Confidentiality**: We used the mechanism of cryptography symmetric inside the cluster and asymmetric outside the cluster.
- **Availability**: The Monitor Agent checks the presence of Nodes by it sends a message or by Inspector Agent.

- **Integrity**: To realize the integrity we use the hash value for verifying the data sent through the nodes of network. In our model, we used Secure Hash Algorithm 1 (SHA-1).
- **Non-repudiation**: The repudiation cannot appears in our model because Node Agent(s) or/and Ambassador Agent(s) records all sends and receives operations. In addition, the Inspector Agent has the ability to detect any repudiation through analysis and comparison.

However, the solution we propose and realize is certainly not the magic solution to the security problem in ad hoc networks, we can not cover all the points of our theme but we hope to have succeeded in opening ports to studies in future to develop or make improvements to this solution. It is clear that there are improvements that remain in order to achieve a more acceptable level of security.

- The evaluation of any security parameters is one of the perspectives of our work and it would be interesting to carry out this study under true real conditions. There are a number of parameters, which we would like to explore, for example, response time, error rate, detection rate, etc.
- We have tried to find a general framework that can be adapted to different types of network and application, i.e. although these approaches have been able to respond to a set of security requirements, they remain effective only in a specific context related to the assumptions and restrictive requirements that were issued during the design.
- We have made a comparison study between this model and other work to prove the effectiveness of this model or not. Thus, each approach (contribution) needs to be improved as an independent approach.
- Finally, we addressed some attack especially the attack of masquerade, which appears if an agent pretend to be a very trustful entity for wining a main position in the network with evil intent.

References

[1]: Mahmoodi, Maryam, and Mohammad Mahmoodi Varnamkhasti. "A secure communication in mobile agent system." arXiv preprint arXiv:1402.0886 (2014).

[2] : Ma, Lu, and Jeffrey JP Tsai. *Security modeling and analysis of mobile agent systems*. Vol. 5. World Scientific, 2006.

[3]: Gavrilovska, Liljana, and Ramjee Prasad. *Ad hoc networking towards seamless communications*. Heidelberg: Springer, 2006.

[4]: Stavroulakis, Peter, and Mark Stamp, eds. *Handbook of information and communication security*. Springer Science & Business Media, 2010.

[5]: Ilyas, Mohammad, ed. *The handbook of ad hoc wireless networks*. CRC press, 2002.

[6]: Chandra, Praphul. *Bulletproof Wireless Security: GSM, UMTS, 802.11, and Ad Hoc Security*. Elsevier, 2011.

[7]: Misra, Sudip, Isaac Zhang, and Subhas Chandra Misra, eds. *Guide to wireless ad hoc networks*. Springer Science & Business Media, 2009.

[8] : Lemainque, Fabrice. *Tout sur les réseaux sans fil*. Dunod, 2009.

[9] : Li (Xiangyang.). Wireless *Ad Hoc and Sensor Networks: Theory and Applications*. Cambridge University Press, 2008.

[10]: Makaya, Christian, and Samuel Pierre, eds. *Emerging Wireless Networks: Concepts, Techniques and Applications*. CRC Press, 2012.

[11]: Mishra, Ajay R. *Fundamentals of cellular network planning and optimisation: 2G/2.5 G/3G... evolution to 4G*. John Wiley & Sons, 2004.

[12]: Stavroulakis, Peter. Interference *analysis and reduction for wireless systems*. Artech House, 2003.

[13]: Chandra, Praphul, and David Lide. *Wi-Fi Telephony: Challenges and solutions for voice over WLANs*. Newnes, 2011.

[14]: Mishra, Ajay R., ed. *Advanced cellular network planning and optimisation: 2G/2.5 G/3G... evolution to 4G*. John Wiley & Sons, 2007.

[15]: Burbank, Jack L., et al. *Wireless Networking: Understanding Internetworking Challenges*. John Wiley & Sons, 2013.

[16]: Gomez, Gerardo, and Rafael Sanchez. *End-to-End Quality of Service over Cellular Networks: Data Services Performance and Optimization in 2G/3G*. England, John Wiley and Sons (2005).

[17]: Ganz, Aura, Zvi Ganz, and Kitti Wongthavarawat. *Multimedia Wireless Networks: Technologies,* Standards *and QoS*. Pearson Education, 2003.

[18]: Heikki...[et al] Kaaranen. *UMTS Networks: Architecture, Mobility and Services*. Wiley, 2005.

[19]: Nicopolitidis, Petros, et al. *Wireless networks*. John Wiley & Sons, Inc., 2003.

[20]: Bidgoli, Hossein. *The Handbook of Computer Networks*. Wiley Publishing, 2007.

[21]: Ghetie, Joseph. *Fixed-Mobile Wireless Networks Convergence*. Cambridge University Press, 2008.

[22]: Vacca, John R., and Michael Foreword By-Erbschloe. *Wireless Broadband Networks Handbook*. McGraw-Hill Professional, 2001.

[23]: Fitzek, Frank HP, and Marcos D. Katz, eds. *Cognitive wireless networks: concepts, methodologies and visions inspiring the age of enlightenment of wireless communications*. Springer Science & Business Media, 2007.

[24]: https://www.wired.com/2010/06/wired-explains-4g/. Last visited: January 07, 2015.

[25]: Muller, Nathan J. *Wireless A to Z*. McGraw-Hill, 2003.

[26]: Wall, David. *Managing and securing a Cisco structured wireless-aware network*. Syngress, 2004.

[27]: Dean, Tamara. *Network+ guide to networks*. Cengage Learning, 2012.

[28]: Lee, Byeong Gi, and Sunghyun Choi. *Broadband wireless access and local networks: mobile WiMAX and WiFi*. Artech House, 2008.

[29]: Chuah, Mooi Choo, and Qinqing Zhang. *Design and Performance of 3G Wireless Networks and wireless LANS*. Springer Science & Business Media, 2005.

[30]: Han, Zhu, Husheng Li, and Wotao Yin. *Compressive sensing for wireless networks*. Cambridge University Press, 2013.

[31]: de Morais Cordeiro, Carlos, and Dharma Prakash Agrawal. *Ad hoc and sensor networks: theory and applications*. World Scientific, 2011.

[32]: http://web.mst.edu/~mobildat/WMAN/index.html. Last visited: January 25, 2015.

[33]: Soyinka, Wale. *Wireless Network Administration A Beginner's Guide*. McGraw Hill Professional, 2010.

[34]: http://en.wikipedia.org/wiki/Wireless_WAN. Last visited: January 05, 2015.

[35]: http://www.afn.org/~afn48922/downs/wireless/wireless_wan.pdf. Last visited: June 18, 2015.

[36]: http://fulloutputproject.wikispaces.com/file/view/wan.pdf. Last visited: January 20, 2015.

[37]: Misra, Sudip, Subhas Chandra Misra, and Isaac Woungang, eds. *Guide to wireless mesh networks*. Springer, 2009.

[38]: Zhang, Yan, Jijun Luo, and Honglin Hu, eds. *Wireless mesh networking: architectures, protocols and standards*. CRC Press, 2006.

[39]: Sarkar, Subir Kumar, T. G. Basavaraju, and C. Puttamadappa. *Ad hoc mobile wireless networks: principles, protocols and applications*. CRC Press, 2007.

[40]: Misra, Sudip, Isaac Zhang, and Subhas Chandra Misra, eds. *Guide to wireless ad hoc networks*. Springer Science & Business Media, 2009.

[41]: Gavrilovska, Liljana, and Ramjee Prasad. *Ad hoc networking towards seamless communications*. Heidelberg: Springer, 2006.

[42]: Ammari, Habib M. *Challenges and Opportunities of Connected K-covered Wireless Sensor Networks: From* Sensor *Deployment to Data Gathering*. Vol. 215. Springer, 2009.

[43]: Suhonen, Jukka, et al. *Low-power wireless sensor networks: protocols, services and applications*. Springer Science & Business Media, 2012.

[44]: Labrador, Miguel A., and Pedro M. Wightman. *Topology Control in Wireless Sensor Networks: with a companion simulation tool for teaching and research*. Springer Science & Business Media, 2009.

[45]: Swami, Ananthram, et al., eds. *Wireless Sensor Networks: Signal Processing and Communications*. John Wiley & Sons, 2007.

[46]: Raghavendra, Cauligi S., Krishna M. Sivalingam, and Taieb Znati, eds. *Wireless sensor networks*. Springer, 2006.

[47]: Boukerche, Azzedine, ed. *Handbook of algorithms for wireless networking and mobile computing*. CRC Press, 2005.

[48]: Santos, Omar. *End-to-end Network Security: Defense-in-depth*. Pearson Education, 2007.

[49] Xiao, Yang, et al. "Wireless network security." EURASIP Journal on Wireless Communications and Networking 2009.1 (2009): 1-3.

[50]: Krawetz, Neal. "Introduction to Network Security (Networking Series), Charles River Media." Inc., Rockland, MA (2006).

[51]: Maiwald, Eric. *Network security: a beginner's guide*. McGraw-Hill Professional, 2001.

[52]: Strebe, Matthew. *Network Security Foundations: Technology Fundamentals for IT Success*. John Wiley & Sons, 2006.

[53]: McCabe, James D. *Network analysis, architecture, and design*. Morgan Kaufmann, 2010.

[54]: Ilyas, Mohammad, and Syed A. Ahson, eds. *Handbook of wireless local area networks: Applications, Technology, Security, and Standards*. CRC Press, 2005.

[55]: Fathi, Hanane, Shyam S. Chakraborty, and Ramjee Prasad. *Voice over IP in wireless heterogeneous networks: Signaling, mobility and security*. Springer Science & Business Media, 2008.

[56]: Andersson, Christoffer. *GPRS and 3G wireless applications: professional developer's guide*. Vol. 19. John Wiley & Sons, 2002.

[57]: Wenstrom, Michael. *Managing Cisco Network Security*. Cisco Press, 2001.

[58]: Prasad, R., and N. Prasad. *802.11 WLANs and IP Networking: Security, QoS, and Mobility*.–USA MA Norwood: Artech House. (2005).

[59]: Chandra, Praphul. *Bulletproof Wireless Security: GSM, UMTS, 802.11, and Ad Hoc Security*. Elsevier, 2011.

[60]: Chaouchi, Hakima, and Maryline Laurent. *Wireless and mobile networks security: security basics, security in on-the-shelf and emerging technologies*. (2009): 667.

[61]: Pathan, Al-Sakib Khan, ed. *Security of self-organizing networks: MANET, WSN, WMN, VANET*. CRC press, 2016.

[62]: Stallings, William. *Network security essentials: applications and standards*. Pearson Education India, 2007.

[63]: Anjum, Farooq, and Petros Mouchtaris. *Security for wireless ad hoc networks*. John Wiley & Sons, 2007.

[64]: Douligeris, Christos, and Dimitrios N. Serpanos. *Network security: current status and future directions*. John Wiley & Sons, 2007.

[65]: Kempf, James. *Wireless Internet Security: Architecture and Protocols*. Cambridge University Press, 2008.

[66]: Ciampa, Mark. *Security+ guide to network security fundamentals*. Cengage Learning, 2012.

[67]: Yang, Xiao, Shen Xuemin, and Du Ding-Zhu. *Wireless Network Security*. (2007).

[68]: Gurtov, Andrei. *Host identity protocol (HIP): towards the secure mobile internet*. Vol. 21. John Wiley & Sons, 2008.

[69]: Cayirci, Erdal, and Chunming Rong. *Security in wireless ad hoc and sensor networks*. John Wiley & Sons, 2008.

[70]: Mohapatra, Prasant, and Srikanth Krishnamurthy, eds. *AD HOC NETWORKS: technologies and protocols*. Springer Science & Business Media, 2004.

[71]: Li, Yingshu, and My T. Thai. *Wireless sensor networks and applications*. Springer Science & Business Media, 2008.

[72]: Dak, Ahmad Yusri, Saadiah Yahya, and Murizah Kassim. "A literature survey on security challenges in VANETs." International Journal of Computer Theory and Engineering 4.6 (2012): 1007.

[73]: Samara, Ghassan, Wafaa AH Al-Salihy, and R. Sures. "Security analysis of vehicular ad hoc nerworks (vanet)." Network Applications Protocols and Services (NETAPPS), 2010 Second International Conference on. IEEE, 2010.

[74]: Khan, Shafiullah, and A. Khan Pathan. *Wireless networks and security*. Berlin: Springer, 2013.

[75]: Ertaul, Levent, and Sridevi Mullapudi. "The Security Problems of Vehicular Ad Hoc Networks (VANETs) and Proposed Solutions in Securing their Operations." ICWN. 2009.

[76]: Chen, Lei, Jiahuang Ji, and Zihong Zhang. "Wireless network security." (2013).

[77]: Patil, Harsh Kupwade, and Stephen A. Szygenda. *Security for wireless sensor networks using identity-based cryptography*. CRC Press, 2012.

[78]: Khan, Shafiullah, Al-Sakib Khan Pathan, and Nabil Ali Alrajeh, eds. *Wireless Sensor Networks: Current Status and Future Trends*. CRC Press, 2016.

[79]: Zhang, Yan, and Paris Kitsos. *Security in RFID and sensor networks*. Auerbach Publications, 2009.

[80]: Ilyas, Mohammad, and Imad Mahgoub, eds. *Handbook of sensor networks: compact wireless and wired sensing systems*. CRC press, 2004.

[81]: Misra, Sudip, Isaac Zhang, and Subhas Chandra Misra, eds. *Guide to wireless sensor networks*. Springer Science & Business Media, 2009.

[82]: Boukerche, Azzedine, ed. *Algorithms and protocols for wireless sensor networks*. Vol. 62. John Wiley & Sons, 2008.

[83]: Dargie, Waltenegus, and Christian Poellabauer. *Fundamentals of wireless sensor networks: theory and practice*. John Wiley & Sons, 2010.

[84]: Xiao, Yang, ed. *Security in sensor networks*. CRC Press, 2016.

[85]: Vacca, John R., ed. *Network and system security*. Elsevier, 2013.

[86]: Chandra, Praphul, et al. *Wireless security: Know it all*. Newnes, 2011.

[87] : Buttyan, Levente, and Jean-Pierre Hubaux. *Security and cooperation in wireless networks: thwarting malicious and selfish behavior in the age of ubiquitous computing*. Cambridge University Press, 2007.

[88]: Imai, Hideki. *Wireless communications security*. Artech House, Inc., 2005.

[89]: Zhang, Yan, Jun Zheng, and Miao Ma, eds. *Handbook of research on wireless security*. IGI Global, 2008.

[90]: Fung, Kwok T. *Network security technologies*. CRC Press, 2004.

[91]: Khan, Jahanzeb, and Anis Khwaja. *Building secure wireless networks with 802.11*. John Wiley & Sons, 2003.

[92]: Krause, Micki, ed. *Information Security Management Handbook on CD-ROM*. Vol. 27. CRC press, 2006.

[93]: Vacca, John R. *Guide to wireless network security*. Springer Science & Business Media, 2006.

[94]: Basagni, Stefano, et al., eds. *Mobile ad hoc networking*. John Wiley & Sons, 2004.

[95]: Hardjono, Thomas, and Lakshminath R. Dondeti. *Security in Wireless LANS and MANS (Artech House Computer Security)*. Artech House, Inc., 2005.

[96]: Schäfer, Günter. *Security in Fixed and Wireless Networks: An Introduction to securing data communications*. John Wiley & Sons, 2004.

[97]: Wu, Shih-Lin, and Yu-Chee Tseng, eds. *Wireless ad hoc networking: personal-area, local-area, and the sensory-area networks*. CRC Press, 2007.

[98]: Prasad, Ramjee, and Luc Deneire. *From WPANs to personal networks: technologies and applications*. (2006): 302.

[99]: Rackley, Steve. *Wireless networking technology: From principles to successful implementation*. Elsevier, 2011.

[100]: Ross, John. The book of wireless: *A painless guide to Wi-Fi and broadband wireless*. No Starch Press, 2008.

[101]: Gregg, Michael. *BUILD YOUR OWN SECURITY LAB: A FIELD GUIDE FOR NETWORK TESTING (With CD)*. John Wiley & Sons, 2008.

[102]: Ma, Jianfeng, Changguang Wang, and Zhuo Ma. *Security Access in Wireless Local Area Networks*. Springer Berlin Heidelberg, 2009.

[103]: Raina, Kapil. *PKI security solutions for the Enterprise: solving HIPAA, E-Paper Act, and other compliance issues*. John Wiley & Sons, 2003.

[104]: Yang, JuCheng. *Biometrics*, 2011.

[105]:http://vxr.es/Hacking%20%20Firewalls%20And%20Networks%20How%20To%20Hack%20Into%20Remote%20Computers.pdf. Last visited: May 03, 2015

[106]: Joshi, James. *Network security: know it all*. Morgan Kaufmann, 2008.

[107]:https://www.certificationkits.com/cisco-certification/ccna-security-certification topics/ccna-security-implement-firewalls-with-sdm/ccna-security-operational-strength-a-weaknesses-of-firewalls/. Last visited: January 12, 2015.

[108]: Misic, Jelena, and Vojislav Misic. *Wireless personal area networks: Performance, interconnection, and security with IEEE 802.15. 4*. Vol. 1. John Wiley & Sons, 2008.

[109]: Mishra, Amitabh. *Security and quality of service in ad hoc wireless networks*. Cambridge University Press, 2008.

[110]: Oppliger, Rolf. *Security technologies for the world wide web*. Artech House, 2003.

[111]: Held, Gilbert. *Wireless mesh networks*. CRC Press, 2005.

[112]: Akyildiz, Ian F., Xudong Wang, and Weilin Wang. "Wireless mesh networks: a survey." Computer networks 47.4 (2005): 445-487.

[113]: Leung, Kin K., and Ekram Hossain. *Wireless Mesh Networks: Architectures, Protocols, Services and Applications*. Springer-Verlag New York, Inc., 2007.

[114]: Yang, Hao, et al. "Security in mobile ad hoc networks: challenges and solutions." IEEE wireless communications 11.1 (2004): 38-47.

[115]: Agrawal, Dharma P., and Qing-An Zeng. *Introduction to wireless and mobile systems*. Cengage Learning, 2015.

[116]: Zhang, Yan, Laurence T. Yang, and Jiming Chen, eds. *RFID and sensor networks: architectures, protocols, security, and integrations*. CRC Press, 2009.

[117]: Karl, Holger, and Andreas Willig. *Protocols and architectures for wireless sensor networks*. John Wiley & Sons, 2007.

[118]: Raghavendra, Cauligi S., Krishna M. Sivalingam, and Taieb Znati, eds. *Wireless sensor networks*. Springer, 2006.

[119]: Baumann, Joachim. *Mobile agents: control algorithms*. Springer, 2006.

[120]: Wooldridge, Michael. *An introduction to multiagent systems*. John Wiley & Sons, 2009.

[121]: http://www.mobilec.org/overview.php. Last visited: March 10, 2016.

[122]: Bagchi, Susmit. *Ubiquitous Multimedia and Mobile Agents: Models and Implementations*. IGI Global (701 E. Chocolate Avenue, Hershey, Pennsylvania, 17033, USA), 2012.

[123]: Braun, Peter, and Wilhelm R. Rossak. *Mobile agents: Basic concepts, mobility models, and the tracy toolkit*. Elsevier, 2005.

[124]: Cao, Jiannong, and Sajal Kumar Das. *Mobile agents in networking and distributed computing*. Vol. 3. John Wiley & Sons, 2012.

[125]: Grimshaw, D. "CPS 720 Artificial Intelligence topics with agents." (2001).

[126]: Ma, Lu, and Jeffrey JP Tsai. *Security modeling and analysis of mobile agent systems*. Vol. 5. World Scientific, 2006.

[127]: Alfalayleh, Mousa. "Mobile agent security| NOVA. The University of Newcastle's Digital Repository." (2009).

[128]: Genco, A. L. E. S. S. A. N. D. R. O. *Mobile agents: principles of operation and applications*. Vol. 6. WIT Press, 2008.

[129]: Jansen, Wayne, and Tom Karygiannis. *Mobile agent security*. No. NIST-SP-800-19. NATIONAL INST OF STANDARDS AND TECHNOLOGY GAITHERSBURG MD, 1998.

[130] : Garrigues Olivella, Carles, and Sergi Robles Martínez. *Contributions to mobile agent protection from malicious hosts*. Universitat Autònoma de Barcelona,, 2008.

[131]: Vijil, E. C., and Kanwal Rekhi. "Security issues in mobile agents." Indian Institute of Bombay, Mumbai (2002).

[132]: Pattanayak, Binod Kumar, and Mamata Rath. "A MOBILE AGENT BASED INTRUSION DETECTION SYSTEM ARCHITECTURE FOR MOBILE AD HOC NETWORKS." Journal of Computer Science 10.6 (2014): 970.

[133]: Abosamra, Ahmed, Mohamed Hashem, and Gamal Darwish. "Securing DSR with mobile agents in wireless ad hoc networks." Egyptian Informatics Journal12.1 (2011): 29-36.

[134]: Chowdhury, Chandreyee, and Sarmistha Neogy. "Securing Mobile Agents in MANET against attacks using Trust." International Journal of Network Security & Its Applications 3.6 (2011): 259.

[135]: Lakshmi, R. Pushpa, and A. Vincent Antony Kumar. "Mobile agent based clustering and maintenance using secure routing protocol for mobile ad hoc network." International Journal of Physical Sciences 8.17 (2013): 793-802.

[136]: PushpaLakshmi, R., A. Vincent Antony Kumar, and R. Rahul. "Mobile agent based composite key management scheme for MANET." Emerging Trends in Electrical and Computer Technology (ICETECT), 2011 International Conference on. IEEE, 2011.

[137]: Halim, Islam Tharwat A., et al. "Agent-based trusted on-demand routing protocol for mobile ad-hoc networks." 2010 6th International Conference on Wireless Communications Networking and Mobile Computing (WiCOM). IEEE, 2010.

[138]: Chatterjee, Pushpita, et al. "A trust enhanced secure clustering framework for wireless ad hoc networks." Wireless Networks 20.7 (2014): 1669-1684.

[139]: Jain, Y., and R. Ahirwar. "Secure Mobile Agent Based IDS for MANET." International Journal of Computer Science and Information Technologies 3.4 (2012): 4798-4805.

[140]: Zhu, Lina, Yi Zhang, and Li Feng. "Distributed Key Management in Ad hoc Network based on Mobile Agent." Intelligent Information Technology Application, 2008. IITA'08. Second International Symposium on. Vol. 1. IEEE, 2008.

[141]: Esfandi, Abolfazl. "Efficient anomaly intrusion detection system in adhoc networks by mobile agents." Computer Science and Information Technology (ICCSIT), 2010 3rd IEEE International Conference on. Vol. 7. IEEE, 2010.

[142]: Krishnan, Deepa. "A Distributed Self-Adaptive Intrusion Detection System for Mobile Ad-hoc Networks Using Tamper Evident Mobile Agents." Procedia Computer Science 46 (2015): 1203-1208.

[143]: http://www.arpnjournals.com/jeas/research_papers/rp_2015/jeas_0615_2182.pdf. Last visited: February 13, 2016.

[144]: Chadli, Sara, et al. "Modeling an Intelligent Architecture of Intrusion Detection System for MANETs." Proceedings of the Mediterranean Conference on Information & Communication Technologies 2015. Springer International Publishing, 2016.

[145] : Hacini, Salima. Sécurité des Systèmes d'Information: Mise en œuvre de la confiance et de l'adaptabilité pour la protection de l'agent mobile. Diss. Université d'Alger, 2008.

[146]: Pandey, Mr Rajesh, Mr Nidheesh Sharma, and Mr Ramratan Rathore. "Aglets (A Java Based Mobile Agent) And Its Security Issue." International Journal of Emerging Trends & Technology in Computer Science (IJETTCS) 2.4 (2013).

[147]: Lange, Danny B., and Mitsuru Oshima. "Mobile agents with Java: the Aglet API." World Wide Web 1.3 (1998): 111-121.

[148]: Aoued, B. E. T. T. A. H. A. R. "Les Aglets d'IBM." Université de Montréal, BETA08036508, Cours IFT6802–H2003.

[149]: Vigna, Giovanni, ed. *Mobile agents and security*. Vol. 1419. Springer, 2003.

[150]: http://www.engr.uconn.edu/~steve/Cse298300/finalagents.ppt. Last visited: February 15, 2016.

[151]: Lange, Danny B., et al. "Aglets: Programming mobile agents in Java." International Conference on Worldwide Computing and Its Applications. Springer Berlin Heidelberg, 1997.

[152]: http://www.memoireonline.com/02/09/1930/Integration-de-protocoles-de-securite-pour-la-communication-inter-agents-dans-la-plate-forme-Aglets.html. Last visited: September 20, 2016.