Securing Data Dissemination in Vehicular ad hoc Networks

PhD Thesis

Hamza Aldabbas

This thesis is submitted in partial fulfillment of the requirements

for the degree of Doctor of Philosophy

Software Technology Research Laboratory

De Montfort University

Leicester - United Kingdom

November 2012

Dedication

To My Beloved **Father** Dr. Muflih Aldabbas without whom this PhD dream would not be a reality. For his endless support, encouragement and love all the way through my life.

To my **Mother** for her endless love, support, encouragement, and continual prayers.

To my **Family**; sisters, brothers, nieces, and nephews for their love.

Abstract

Vehicular *ad hoc* networks (VANETs) are a subclass of mobile *ad hoc* networks (MANETs) in which the mobile nodes are vehicles; these vehicles are autonomous systems connected by wireless communication on a peer-to-peer basis. They are self-organized, self-configured and self-controlled infrastructure-less networks. This kind of network has the advantage of being able to be set-up and deployed anywhere and anytime because it has no infrastructure set-up and no central administration. Distributing information between these vehicles over long ranges in such networks, however, is a very challenging task, since sharing information always has a risk attached to it especially when the information is confidential. The disclosure of such information to anyone else other than the intended parties could be extremely damaging, particularly in military applications where controlling the dissemination of messages is essential.

This thesis therefore provides a review of the issue of security in VANET and MANET; it also surveys existing solutions for dissemination control. It highlights a particular area not adequately addressed until now: controlling information flow in VANETs. This thesis contributes a policy-based framework to control the dissemination of messages communicated between nodes in order to ensure that message remains confidential not only during transmission, but also after it has been communicated to another peer, and to keep the message contents private to an originator-defined subset of nodes in the VANET.

This thesis presents a novel framework to control data dissemination in vehicle *ad hoc* networks in which policies are attached to messages as they are sent between peers. This

is done by automatically attaching policies along with messages to specify how the information can be used by the receiver, so as to prevent disclosure of the messages other than consistent with the requirements of the originator. These requirements are represented as a set of policy rules that explicitly instructs recipients how the information contained in messages can be disseminated to other nodes in order to avoid unintended disclosure.

This thesis describes the data dissemination policy language used in this work; and further describes the policy rules in order to be a suitable and understandable language for the framework to ensure the confidentiality requirement of the originator. This thesis also contributes a policy conflict resolution that allows the originator to be asked for up-to-date policies and preferences.

The framework was evaluated using the Network Simulator (NS-2) to provide and check whether the privacy and confidentiality of the originators' messages were met. A policy-based agent protocol and a new packet structure were implemented in this work to manage and enforce the policies attached to packets at every node in the VANET. Some case studies are presented in this thesis to show how data dissemination can be controlled based on the policy of the originator. The results of these case studies show the feasibility of our research to control the data dissemination between nodes in VANETs. NS-2 is also used to test the performance of the proposed policy-based agent protocol and demonstrate its effectiveness using various network performance metrics (average delay and overhead).

Declaration

I declare that the work described in this thesis is original work undertaken by me for the degree of Doctor of Philosophy at the Software Technology Research Laboratory (STRL) at De Montfort University, United Kingdom.

No part of the material described in this thesis has been submitted for any award of any other degree or qualification in this or any other university or college of advanced education.

This thesis is written by me and produced using LaTeX.

Hamza Aldabbas

Publications

1. Ensuring Data Confidentiality and Privacy in Mobile Ad hoc Networks, Hamza Aldabbas, Helge Janicke, Radwan Abu Jassar, and Tariq Alwadan. In proceedings of Third International Conference on Wireless and Mobile Networks (WiMoNe-3), (LNICST, Springer), Bangalore-India, January 2 - 4, 2012. [link]

2. Data Confidentiality in Mobile Ad hoc Networks, Hamza Aldabbas, Tariq Alwadan, Helge Janicke, Ali Al-Bayatti, in International Journal of Wireless Mobile Networks (IJWMN), Vol. 4, No. 1, February 2012. [pdf]

3. Dynamic Policy Management in Mobile Grid Environments, Tariq Alwadan, Hamza Aldabbas, Helge Janicke, Omar Aldabbas, in International Journal of Computer Networks Communications (IJCNC), Vol.4, No.2, March 2012. [pdf]

4. Controlling Data Dissemination, Helge Janicke, Mohammad Sarrab and Hamza Aldabbas. In proceedings of the 4th International Workshop on Autonomous and Spontaneous Security (SETOP) 2011 in Data Privacy Management and Autonomous Spontaneous Security (LNCS, Springer), Leuven (Belgium) September 15-16, 2011. [link]

5. New Framework for Dynamic Policy Management in Grid Environments, Tariq Alwadan, Helge Janicke, Omar Aldabbas and Hamza Aldabbas. Proceedings of

CoNeCo 2011 in Communications in Computer and Information Science, Ankara, Turkey (LNCS, Springer) 2011. [link]

Acknowledgement

In the name of **Allah**, the Most Merciful and the Most Gracious, I give praise and thanks to Him for supporting me with the strength to complete this research. Without Him, none of this work would have been possible.

Studying at De Montfort University was the most educational experience I have ever had in my life. To me, this is certainly related to the high standard of the facilities offered to students in general and for the research students in particular, I found the surrounding environment very friendly, and on top of that the excellence of the supervision I found in the STRL department.

In this context I would like to show my profound appreciation and lasting gratitude to my first supervisor **Dr. Helge Janicke** for his guidance and support during my study, his wide knowledge has been of great value to me. His understanding, encouragement and patience during the many discussions we had which have lead to this thesis. Without his guidance and support I would never have been able to organize and complete my thesis in the constrained time. The only thing I can tell him is thank you. I would also like to express my sincere gratitude to **Prof. Hussein Zedan**, the head of the STRL, for his valuable comments, guidance, suggestions, his scientific and morale support through the many difficulties I faced during my study was crucial to my success, the only thing I can say to him is thank you for everything.

Also I would to express my thanks to my family in Jordan, especially my father **Dr. Muflih Aldabbas** who was behind everything I have achieved in this life, and to my

lovely **mother** Jamelah Alhadidi for her care and support, and to my dearest brothers and sisters Bilal, Dr. Ahmad, Dr. Radwan, Dr. Mohammad, Huda, and Hana'. And of course my special thanks goes to my second family in England **Dr. Michael Tully**, **Francoise**, and **Catarina** for their help and their support to improve my English language, being here with you is the same feeling when I be with my family, the only thing I can say to you is thank you for everything. My special thanks also goes to my cousin **Dr. Omer Aldabbas** and his wife **Dr. Mai Alfawair** for their help, concern, and encouragement through all these years.

Furthermore, I would like to express my appreciation to the friends I met here in De Montfort University, Dr. Raed Kanaan, Dr. Maher Abuzeid, Ayman Al omar, Dr. Ma'en Aljezawi, Dr. Ibrahim Naimi, Mustafa Al-khawaldeh, Ghazi Alqarutti, Mohammad Kharabsheh, Mohammad Alshra'ah Muneer Nusir, Abdelruhman Alshebeeb and Dr. Bassam Alzoubi, special thanks goes to my friends in STRL Dr. Tariq Alwad'an, Dr. Murad Magableh and Dr. Khalid Aldrawhish. Our friendship started here and will continue forever.

Last but not least I would like to thank all my colleagues at the Software Technology Research Laboratory (**STRL**) for their constant support and making STRL such a friendly environment for the research. It may not be possible to mention all of you here, but your help and support were highly appreciated.

Contents

List of Figures

List of Tables

Listings

Chapter 1

Introduction

1.1 Background

Vehicle *ad hoc* networks (VANETs) are autonomous systems consisting of a number of mobile nodes communicating between themselves by wireless communication on a peer-to-peer basis. They are self-organized, self-configured and self-controlled infrastructure-less networks. These nodes can communicate with each other without any pre-planned or base station. These networks are therefore particularly useful to those who need to communicate in situations where no fixed wired infrastructures are available. Disseminating information securely between these nodes in such networks, however, is a challenging task, particularly when the information is confidential. Revealing such information to anyone else other than the intended nodes could be highly damaging, especially in military applications where controlling the dissemination of messages is essential.

Intra-vehicle communication brings essential changes to telecommunications and data networking. The manager of the second biggest automobile manufacturer Toyota expects vehicles on sale by 2012 to include Azure-based Smart Center technology (Microsoft's cloud architecture). Toyota and Microsoft have declared a 12 million dollar joint investment on including Microsoft's Azure cloud platform in upcoming Toyota vehicles for

better telematics [5]. Vehicular *ad hoc* networks (VANETs) are a new emerging network technology derived from *ad hoc* networks; vehicles are free to move and organise themselves arbitrarily, whilst they can exchange information between themselves and Road Side Units (RSUs). This promising technology for future smart vehicle systems and Intelligent Transportation Systems (ITS) has the potential to increase road safety. VANETs can also be used to enhance passenger comfort by providing services such as exchanging traffic information, weather information, interactive communication and offering internet access. Compared to the limited resources available in traditional *ad hoc* networks, vehicles can store and process large amounts of information. These data will be obtained via the vehicles' sensors and may also include drivers' personal information. Both travelling vehicles' drivers and passengers today can have access to the sensor data (dash-board), location information (GPS), traffic information, etc.

1.2 Problem statement

The key challenges in designing VANETs come from its decentralised nature, self-organisation, and self-management, since the opportunity of vehicles movement is very high. In addition, all communications are carried out through wireless medium in short-range communication. These unique characteristics present security issues for VANETs, so there have been concerted efforts by the research community in message encryption, digital signature, and key management [6, 7, 8]. Many challenges, especially those related to data confidentiality, however, remain to be solved. A key concern of data confidentiality is that individuals should be able to keep and control access to their personal information by choosing to which entities information should be disclosed in a discretionary way. In addition to access controls, there should also be provision to control the flow of private information.

Existing approaches in security have been applied to VANETs: traditional crypto-

graphic solutions, for example, are using public key certificates to maintain trust, in which a Trusted Third Party (TTP) or Certificate Authority (CA) certifies the identity associated with a public key of each communicated entity; therefore they can provide end-to-end secure communication channels. These approaches are mainly focused on message confidentiality, integrity and non-repudiation; they do not consider, however, controlling the message dissemination after it being sent to recipients; thus the management of data confidentiality, privacy concerns and how these certified entities act is left to the application layer [9]. Point-to-Point communication is secured using the previous traditional methods described a above. This does not prevent, however, unwanted dissemination to specific nodes in the network. Currently, there is no way of tracking dissemination and limiting it effectively in VANETs. Privacy is defined in VANETs as protecting the use of some highly sensitive, private or secure information (for example, the identity and location of vehicles) from being shared or being disclosed to unwanted party(ies) without their permission, this does not take into account policies as to what data can be shared or not, and with whom and for what purpose.

While privacy means preventing the identity and the location of the nodes from being disclosed to any other entities, confidentiality means keeping the secrecy of the exchanged data from being revealed to those who have not permission to access it. This means to protect or restrict secret information from being disclosed to others by controlling the data dissemination.

Most research on privacy issues in VANETs addressed location privacy [10] and 'big brother' scenarios [6] where vehicles location can be tracked by an untrusted third party. The CARAVAN scheme [11] allows vehicles to maintain privacy by forming groups in which the group leader acts as a proxy on behalf of all members of the group with a random silent period to mitigate tracking of vehicles. Others [12, 13] addressed the same problem by using pseudonyms to hide the relationship between identity and the location. Although pseudonyms are significant in the overall security of VANETs and are advan-

tageous for protecting the identity of users, these solutions do not provide full solution to privacy concerns as they cannot control the dissemination of information. Indeed for many application-level services the knowledge of the senders' identity is paramount to their function. Hence, pseudonyms could only be one part of a privacy solution, but the need remains for more comprehensive solution(s) allowing originators of information control over its dissemination.

There is currently no monitoring approach for controlling the data dissemination in the VANET or any such network, also there is no mechanism to support scalable originator interaction. This interaction is considered an essential part in any flexible and reliable security scheme because what might be considered to be secure at particular time could be considered insecure afterwards. The interaction with the policy decision approach enables the originator to control the security requirement to the related information or to modify the way that data disseminated to different nodes in the network, even after the data has been released. Managing the access to secret information sent between nodes in the VANET is currently not possible. To our knowledge, none of the related work addressed the issue of controlling the information flow in VANETs.

1.3 Research Question

Distributing information between nodes in VANETs is a very challenging task, since sharing information always has a risk attached to it, especially when the information is confidential. The disclosure of such information to anyone else other than the intended nodes could be extremely damaging.

Data dissemination takes place from a source to a target therefore the information in a source will be disseminated directly or indirectly to the target depending if the target is adjacent to the source or not. When private information is sent from the source to intended list of targets only, thus the research question is

- **How can we prevent information from being leaked to unintended entity(ies)?**

In order to answer this research question, a set of sub questions were formulated:

1. How to keep message contents private to an originator-defined subset of nodes in the VANET.
2. How to define an originator-defined subset of nodes, including set-up and maintenance cost.
3. How to enforce the privacy and confidentiality requirement of the originator.
4. How to represent the originator data dissemination requirements as a set of rules.
5. How to control the dissemination of messages while the nodes are communicating between each other in the network.
6. How to accommodate changes of the security requirement to the related information by referring back to the originator.

1.4 Contributions

Controlling data dissemination is an important component in a security system; therefore mechanisms must be used to prevent any possible message compromise on VANETs. In this thesis, we reviewed the main security issues and existing solutions in VANET, in particular the area of security of VANET which has not been hitherto widely researched. We addressed the dissemination control and trust management problem in VANETs in order to ensure the originator's data dissemination requirements not only during transmission but also after it has been sent to another node. Therefore, this thesis contains the following original contributions:

- **A policy-based framework to control the data dissemination in VANETs**. This is built on the automatic communication of policies between nodes and draws on concepts developed for other forms of networks e.g. [14]. The purpose of this framework is to keep data confidential not only during communication, but also after it has been transmitted to another node, ensuring that the contents of messages are kept secret to an originator-defined subset of peers in VANETs. Therefore in our research we devised a framework to provide a prevention component which governs the dissemination of an originator's messages, so that they cannot be accidently disclosed to unwanted third parties. This is accomplished by using policies of the originators to control the access to their messages, and to ensure that these polices will be enforced upon intended recipients. The framework is introduced in Chapter 4 and has been published in [15, 16, 17].

- **A data dissemination policy language**, that specifies the data dissemination security actions to be considered in the network to keep the message secure. It is represented as a set of policy rules that declares the data dissemination requirement based on the originator of the message. The data dissemination policy works as the reference that controls the flow of the messages while the nodes are communicating between each other in the network. The data dissemination policy should specify high-level requirements into low-level policy rules whose enforcement can be fully automated and understood for the framework. In this work we provided a suitable data dissemination policy to be used for the framework in which the originator of the message retains control over its dissemination. The framework and the data dissemination policy language been used are introduced in Chapter 4 and Chapter 5 respectively and have been published in [15, 16, 17].

- **An interaction mechanism to query the originator for its up-to-date policy**. In addition of presenting a controlling dissemination mechanism in VANETs by the

use of policy-based framework to ensure that information is not disclosed to unwanted parties, we also presented an interaction mechanism by returning to the originator for its up-to-date policy. Since changing policies is an important requirement in policy-based systems, because what is considered to be secure now can be insecure afterwards. This interaction mechanism is introduced in Chapter 5.

- **Evaluating our framework using the Network Simulator (NS-2) through some case studies to provide and check whether the privacy and confidentiality of the originator are met**. We used NS-2 agent to implement our policy-based framework together with policy rules attached to packets at every node in the VANET; we built a new agent protocol and a new packet structure to suit this protocol in NS-2. The new policy-based agent protocol is derived from an existing class in NS-2. The implementation and evaluation are introduced in Chapter 6 and Chapter 7 respectively and have been published in [15, 16].

1.5 Research methodology

The research methodology which used in this thesis is a typical computer science research technique (constructive research method) [18], where the new contributions are developed via a new framework, theory, model or algorithm. The proposed approach is composed of seven work packages. One addresses the research background and the research project requirements. Five are scientific research work packages. The last work package focuses on writing up the thesis.

- Work package 1: Research background.

  A structured literature survey was conducted in order to understand the approaches which are related to the research question, and to understand the problem domain. The review used the following data base digital resources: IEEE Xplore, Springer-

Link, ACM Digital Library and CiteSeer.

- Work package 2: Framework.

  This work package concentrated on the design of the framework. This work package clearly described all the components of the proposed framework and showed how these components interact with each other to achieve the research objectives. In this work package the work was split into four tasks:

  1. Defining the role of each component in the proposed framework.
  2. Policy model for controlling the packets in the VANET.
  3. The originator interaction.
  4. Designing a computational model for VANETs to describe the interactivity process between these communicating entities.

- Work package 3: Theory of data dissemination.

  This work package concentrated on the development of a mechanism for controlling data dissemination which addresses the possible communication between nodes in the network. Data dissemination takes place from a source to a target therefore the information in a source will be disseminated directly or indirectly to the target depending if the target is adjacent to the source or not. This work package was split into two tasks.

  1. Direct communication between nodes.
  2. Indirect communication between nodes.

- Work package 4: Algorithm development.

  This work package concentrated on providing a new algorithm to control the data dissemination from source to destination in VANETs using flexible security mech-

anisms that can change the control of the security requirement to the related information by referring back to the originator. This work package was split into five tasks.

1. Development of an algorithm to read the policy from the sending node.
2. Development of a policy manager algorithm that does as a decision maker deciding point to receive/send the information from/into the system based on policies conditions installed .
3. Development of an algorithm for attaching policy with the message in the sending function.
4. Development of an algorithm for splitting the message from the policy attached in receiving process.
5. Development of an algorithm to write the policy to the receiving node once the packet is received.

- Work package 5: Data dissemination policy and the originator interaction.

  This work package concentrated on how to represent the originator data dissemination requirements in terms of policies. The objective is to show how the originator interacts with the controlling mechanism, and to describe the data dissemination policy language used in the system. The research in this work package was split into two tasks:

  1. Development of the data dissemination policy.
  2. The originator feedback.

- WP 6: Implementing and evaluating the research through some case studies.

  This work package illustrated the practical phase of the proposed research. This work package concentrated on how to build the policy agent in NS-2, create pack-

ets structure to suit that protocol, choose the encryption, choose the decryption algorithm and the hash function. This work package also presented a conclusion which been obtained from the simulation results which been carried out in the evaluation process through some case studies. Some of the potential future works for this research was mentioned to motivate further investigation in the field of data dissemination in VANETs.

- WP 7: Writing up.

  Writing up of the thesis which is based on the results of all work packages.

1.6 Thesis Outline

The thesis is structured as follows:

- **Chapter 2** presents an introduction of the wireless *ad hoc* networks and mobile *ad hoc* networks (MANETs), it also describes the characteristics, challenges, vulnerabilities of mobile ad hoc network, and then it illustrates the various advantages of MANET and numerate the applications of MANET. This chapter also presents an introduction of the vehicle *ad hoc* networks (VANETs), history and background, also it describes the characteristics, challenges, vulnerabilities of VANETs.

- **Chapter 3** reviews the security in vehicle *ad hoc* networks, it also defines the security concepts and requirements, it presents an overview of the network security, it also presents an overview of the cryptography background and finally presents the related work in privacy and confidentiality issues in VANETs and MANETs.

- **Chapter 4** presents a novel policy-based framework to control the dissemination of data communicated between nodes in VANETs by attaching originator policies to messages as they are sent, this chapter also gives a motivating example drawn from the military domain, where the impact of confidentiality breach is self-evidently

crucial. This chapter provides a brief description of the data dissemination requirement and describes the data dissemination policy. Finally this chapter describes how the components in the framework interact between each other.

- **Chapter 5** links between the framework and data dissemination policy in VANETs at a high level in order to address the research question (described in Section 1.3). This chapter describes the data dissemination policy language used in this work; it also describes the policy rules modified from previously published work [17, 19] in order to be a suitable and understandable language for the framework to ensure the originator confidentiality requirement. Finally this chapter describes the data dissemination policy conflict rules and when the originator should be asked for its up-to-date policy.

- **Chapter 6** outlines the network simulator chosen to implement and evaluate this framework. This chapter discusses the implementation process which been carried out in this work starting from building the policy agent, creating packets structure, choosing the encryption, the decryption algorithms and the hash function. All this was implemented using NS-2 simulator which is a real network environment simulator.

- **Chapter 7** the policy-based framework was evaluated through some case studies. There are many network simulation tools available to evaluate the performance of the proposed mechanisms and protocols for simulating both the wireless and wired networks. Most of the research in the *ad hoc* networks has been evaluated using the program NS-2 [4]. Similarly we used NS-2 simulator to evaluate our policy-based framework to check whether the privacy and confidentiality requirements of the originator are met or not. Therefore, four case studies are presented in this chapter to evaluate our policy-based approach via NS-2.

- **Chapter 8** Conclusion and Future Work. This chapter includes the final results,

conclusion of our research and a sight on future work.

Chapter 2

Background on MANETs and VANETs

Objectives:

- Present an introduction of the wireless networks, MANETs, and VANETs .
- Highlight the characteristics, challenges, and vulnerabilities of MANET and VANET.
- Highlight the various advantages and the applications of MANET.
- Present the history and background of the Vehicular *ad hoc* networks (VANETs).
- Highlight the recent wireless communication technologies for VANETs.

2.1 Introduction

Recently, ad hoc networks received extensive attention in both industrial and military applications, because of the striking property of creating a network while moving from one place to another and not requiring any pre-designed infrastructure. This chapter therefore presents an introduction of wireless networks and its two types: infrastructure and

infrastructure less in Section 2.2, concentrating on the second type; then an introduction to mobile ad hoc networks (MANETs) in Section 2.3, with its characteristics described in Section 2.3.1 : constrained resources, infrastructure less, low and variable bandwidth, dynamic topology, multi-hop communications, limited device security, limited physical security, and short range connectivity.

These unique characteristics in MANETs present appreciable challenges, therefore Section 2.3.2 describes the vulnerabilities and challenges of MANET: lack of secure boundaries, restricted power supply, unreliability, lack of centralized management facility, threats from compromised nodes, and scalability. Section 2.3.3 mentions the advantages of MANETs.

There are many applications of MANETs therefore Section 2.3.4 presents these applications: home networks, enterprise networks, military applications, emergency response networks, sensor networks, and Vehicular ad hoc Networks (VANETs). Section 2.4 presents an introduction to VANETs, and describes the modern vehicles' components.

Section 2.4.1 presents history and background of VANETs by reviewing these projects and consortiums in VANET: PROMETHEUS project (program for European traffic with highest efficiency and unprecedented safety), DRIVE project (dedicated road drive infrastructure for vehicle safety in Europe), C2CCC (car2car communication consortium), and IEEE 802.11p task group.

Section 2.4.2 presents an overview of recent wireless communication technologies for VANETs: Wi-Fi, WiMAX and DSRC (Dedicated short range communication). Section 2.4.2.3 presents an overview of P1609 IEEE standards.

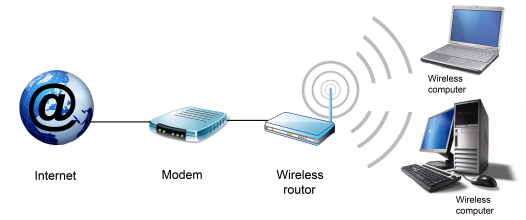
Section 2.4.3 presents characteristics of VANETs: dynamic topology, random disconnection, mobility modelling, computational power, and variable density. These unique characteristics present appreciable challenges in designing VANETs; Section 2.4.4, therefore, presents these challenges: Medium Access Control Protocols (MAC), mobility management, data dissemination and security.

2.2 Introduction to wireless networks

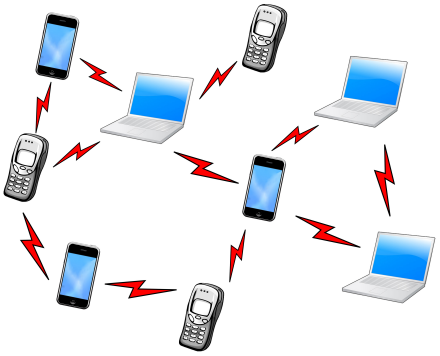
In the past few decades wireless networks have become increasingly popular, due to the wide availability and rapid introduction of wireless transceivers into a variety of computing devices such as PDAs, laptop and desktop computers. Wireless communication brings essential changes to telecommunications and data networking. Air is used as the transmission medium, allowing great flexibility; networks can be deployed quickly where cabling is difficult. Good performance and low prices encourage progressively more home users and companies to choose these new kinds of networks. Wireless communications could replace wired communications in many situations. Travelling users today have access to the Internet at many places like their offices, homes, and even at public places like airports, conferences, shopping centres, hotels, and libraries.

Wireless LAN networks can be classified into two categories. The first and most common is infrastructure networks with fixed and wired gateways (wireless network built on-top of a wired network). In this kind of network mobile nodes connect to a network via an AP (Access Point) within its coverage range in a single hop communication technique. The second type of wireless network is the infrastructure-less mobile network, commonly known as mobile *ad hoc* network (MANET). Figure 2.1 shows examples of the two kinds; in MANET nodes can communicate directly, operating both as router and host, sending and receiving packets of data to and from other nodes in the network. MANETs are thus termed multi-hop networks.

One advantage of wireless is the ability to transmit data among users in a common area while remaining mobile. However, the distance between participants is limited by the range of transmitters or their proximity to wireless access points. On the other hand mobile *ad hoc* wireless networks (MANETs) solve this problem by allowing out of range nodes to route data through intermediate nodes.



[Infrastructure-based wireless network.]



[Mobile *ad hoc* wireless network.]

Figure 2.1: Example of Infrastructure and Infrastructure-less wireless networks

2.3 Mobile wireless *ad hoc* networks (MANETs)

Mobile *ad hoc* networks are autonomous systems which consist of a number of mobile nodes that communicate between themselves using wireless transmission. They are thus self-organized, self-configured and self-controlled infrastructure-less. This kind of network has the advantage of being able to be set up and deployed anywhere and anytime because it has a simple infrastructure setup and no central administration .

Mobile *ad hoc* networks (MANETs) are case of wireless *ad hoc* networks, progressively more popular and successful in the marketplace of wireless technology. Examples include Bluetooth and Wireless Local Area Networks (WLANs).

These networks are particularly useful to those mobile users who need to communicate in situations where no fixed wired infrastructures are available. Obvious examples are the military or the emergency services: one clear situation might be a fire fighter who needs to connect to an ambulance. In such situations a collection of mobile nodes with wireless network interface can form a transitory network [20]. Recently, *ad hoc* networks received

extensive attention in both industrial and military applications, because of the striking property of creating a network while moving from one place to another and it does not require any pre designed infrastructure.

2.3.1 The Characteristics of MANET

A mobile *ad hoc* network (MANET) is an independent system of mobile nodes linked by wireless connections. These nodes are free to move arbitrarily; therefore, the topology of wireless networks may change swiftly and in an unpredictable manner.
Generally, direct communication in MANETs is possible only between adjacent nodes. Thus, communication between distant nodes is established using multiple-hop. Since the locations may change dynamically, as a consequence the interconnections between the adjacent nodes may change continually. Each mobile node functions as both host and router, relaying data packets from one node to another. MANETs have many characteristics that make them distinguishable from other wireless and wired networks [21, 22, 23, 24, 25] which are in detail :

- **Constrained Resources**: Most MANET devices are small handheld devices like personal digital assistants (PDAs), laptops and cell phones. These devices have limitations because of their restricted battery-capacity, small processing power and storage facilities. Energy consumption is an important criterion when designing the MANET.

- **Infrastructure-less(Autonomous)**: MANETs are based on the teamwork between independent peer-to-peer nodes that communicate with each other. Without any pre-planned arrangement or base station, all nodes have the same role in the network. There are no pre-set roles like router, server or gateways for the nodes participating in the network.

- **Low and Variable Bandwidth**: Wireless links which connect the MANET nodes

have lower bandwidth than wired links. The effects of interference, congestion and noise are more significant.

- **Dynamic Topology**: MANET nodes can move arbitrarily; thus the nodes can dynamically enter and leave the network, continually change their links and topologies. This leads to frequent changes in the routing information.

- **Multi-hop communications**: The communication in MANET between any two nodes is performed by numerous intermediary nodes whose functions are to relay data-packets from one point to another. *Ad hoc* networks require multi-hop communications, for example, in Figure 2.2, nodes A and D must engage the help of nodes B and C to relay data-packets between them in order to communicate.

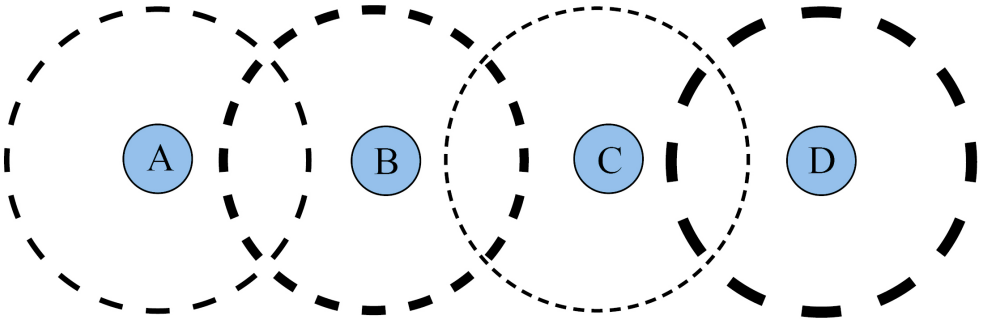


Figure 2.2: Mobile *ad hoc* network of four nodes, using the transmission range of nodes B and C in order to communicate between node A and node D

- **Limited Device Security**: MANETs devices are usually small and can be transported from one place to another. Unfortunately, as a result these devices can be easily lost, stolen or damaged.

- **Limited Physical Layer Security**: MANETs are in general more vulnerable to physical layer's attacks than wired networks; the possibility of spoofing, eavesdropping, jamming and denial of service (DoS) attacks should be carefully considered.

However the self-administration nature of MANET makes them more robust against single failure points.

- **Short Range Connectivity**: MANETs rely on radio frequency (RF) technology to connect, which is in general considered to be short range communication. For that reason, the nodes that want to communicate directly need to be in the close frequency range of each other.

2.3.2 The vulnerabilities and challenges of MANET

The key challenges in designing MANETs result from the decentralised nature and lack of central infrastructure like a base station, access point or server. In addition to that, all communications are carried out through the wireless medium. These unique characteristics present appreciable challenges for MANETs such as [26, 27, 28, 29]:

- **Lack of Secure Boundaries**: In comparison with wired networks where the devices must have a physical access to the network medium, mobile *ad hoc* networks have no apparent secure boundary. There is no need for attackers to have physical access to the network; once the attackers are in the transmission range of any other devices, then they can join and communicate with other devices.

- **Restricted Power Supply**: In contrast to wired networks where the nodes can get their electrical supply directly from the power points, MANETs nodes are generally operated by small batteries with limited lifetime. Nodes are therefore less likely to be able to operate intensive computations, which makes them vulnerable to a denial-of-service attack (DoS). This can be done by sending additional routing packets to a targeted node, in order to be executed by the targeted node in an attempt to exhaust its battery.

- **Unreliability**: Due to the limited battery supply and mobility in MANETs, the mobile devices cannot be assured as being reliable to serve communication participants; some nodes may behave in a 'selfish' manner when it finds that there is only limited power supply.

- **Lack of Centralized Management Facility**: The lack of centralized management makes the detection of attacks complicated. Mobile *ad hoc* networks are highly dynamic and large scale therefore they cannot be easily monitored. Also benign (non-malignant) failures in the mobile *ad hoc* network are fairly common, for example, transmission destructions and packet dropping. As a result, malicious failures will be more difficult to discover.

- **Threats From Compromised Nodes**: Due to the movement of the nodes in *ad hoc* networks, it can be challenging to detect the malicious attack carried out by a compromised node, particularly in a large scale *ad hoc* network.

- **Scalability**: In MANETs nodes entering and leaving the network cause frequent changes to the network topology; the network may consist of hundreds to thousands of nodes; the routing protocols configurations and key management services therefore have to be adjusted to fit these new conditions.

2.3.3 Advantages of mobile *ad hoc* networks

Mobile *ad hoc* networks (MANETs) have particular advantages over the conventional networks. Some of these advantages are:

1. Increasing mobility and flexibility, as MANETs can be initiated and terminated in a very short time.

2. More robust than traditional wireless networks, as MANETs do not rely on centralised base station.

3. More economical than traditional networks, as MANETs eliminate the cost deployment of infrastructure.

4. Reducing the power consumption of devices by using multi-hop sending mechanism; all nodes can be relay stations receiving and sending packets to the goal destination, rather than sending data packets over one long hop.

5. *Ad hoc* networks can be used to enlarge the coverage area of an access point. By this method, a few users are connected to a single access point providing connections to another outside of range users. Figure 2.3 shows how the *ad hoc* fashion can do this.

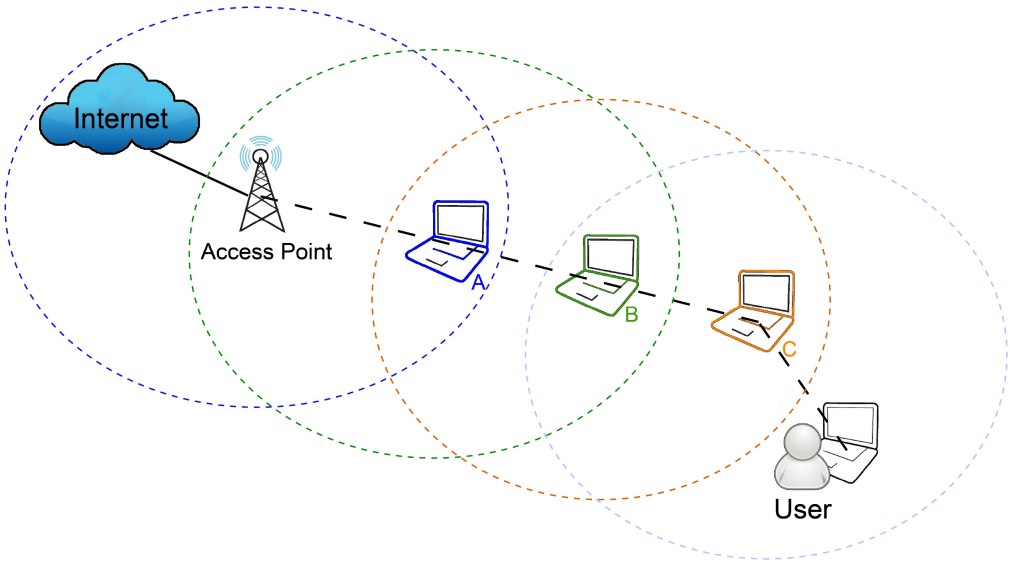


Figure 2.3: Using *ad hoc* to extend coverage

2.3.4 Applications of mobile *ad hoc* networks

There are many applications of mobile *ad hoc* networks; these have been listed in [20] [30] [22] [31] [32] [33]:

1. **Home Network and Enterprise Network**: One use of MANET is in some home environments, such as home wireless networks, smart homes and personal area

networks (PAN) which we can make communication between smart household appliances, in comparison with fixed wireless network, wireless *ad hoc* devices can move in free manner and they organise themselves in an arbitrary type. Roaming can be carried out while the devices are communicating with each other, which is suitable to businesses demand such as in office wireless networks, conferences, meeting rooms and networks at construction areas.

2. **Military Applications**: Mobile *ad hoc* network can be valuable to soldiers in order to establish communication for tactical campaigns; setting up a fixed infrastructure in enemy areas or in hostile lands may not be possible in such conditions. Whereas, MANETs can offer the required communication promptly and quickly. The coordination of military objects moving at high speeds, such as fleets of airplanes or warships is another application in this area.

3. **Emergency Response Network**: Mobile *ad hoc* network can be used to supply emergency management services applications, for example in disaster recovery, fire fighting, search and rescue operations where the whole communication infrastructure has been demolished or is unavailable. Deploying MANETs in these places can set up an infrastructure quickly.

4. **Sensor Network**: Wireless sensor networks can be deployed in *ad hoc* mode to assist monitoring and controlling physical surroundings from distant places with sufficient accuracy. These sensors might be equipped with a selection of components (processor, radio transceiver, actuator, micro-controller, and energy source) in order to measure several physical attributes like motion, temperature, moisture, atmospheric pressure, sound, vibration, pollution and velocity.

   Sensor networks are used in military applications such as battlefield observation; equipment ammunition; targeting; and nuclear, biological and chemical attack detection and reconnaissance. It is also commonly used in many manufacturing and

civilian applications, such as monitoring product quality, controlling machines, healthcare applications, home automation control(smart home), and traffic control.

5. **Vehicular *Ad hoc* Network (VANET)**: It is a subclass of mobile *ad hoc* networks (MANETs), where the mobile nodes are vehicles; today vehicles are becoming "computer networks on wheels", these vehicles are free to move and organise themselves arbitrarily, which they can exchange information between themselves and Road Side Units (RSUs), in order to increase safety in the roads by warning the drivers about ongoing hazard situations, and increasing the responsiveness of their surroundings and make them more vigilant.

   In another aspect inter-vehicle communication(IVC) can be used to enhance passenger comfort and traffic system such as exchanging traffic information, weather information, petrol station, restaurants location and price information, and providing the interactive communication like offering access to the Internet.

2.4 *Vehicular ad hoc networks (VANETs)*

Vehicular *ad hoc* network is a new emerging network technology derived from *ad hoc* networks, which can provide wireless communication services between vehicles and adjacent road side units; it is a promising technology for future smart vehicle systems and intelligent transportation systems (ITS).

In VANETs, each vehicle in the system as in Figure 3.1 has a computing device, a short-range wireless interface, event data recorder (EDR), front and rear sensors and a GPS (Global Positioning System) device which is progressively more becoming common in vehicles today, in order to provide vehicles' location, speed, current time and direction.

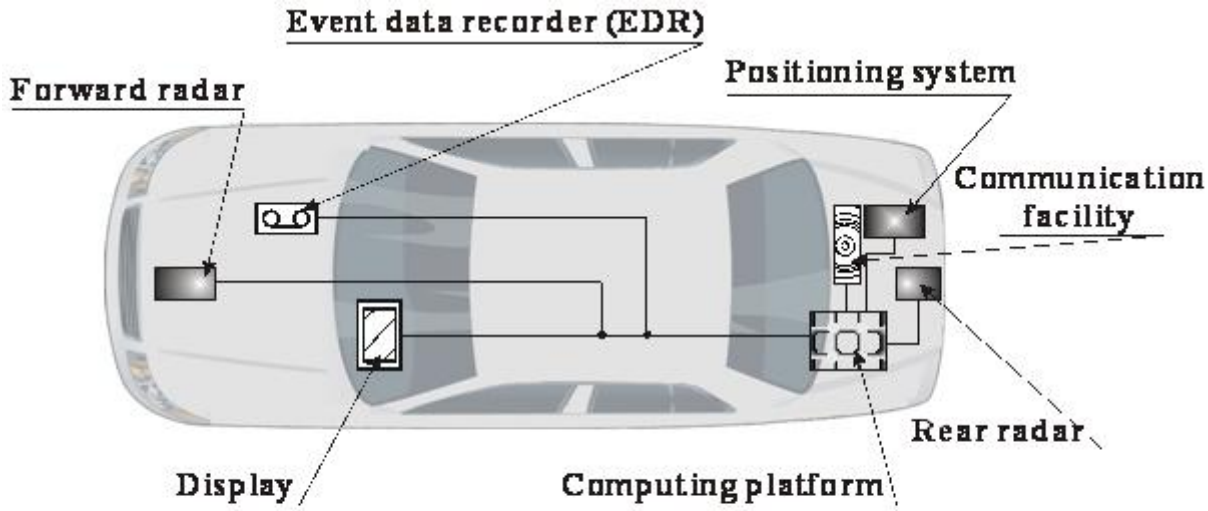


Figure 2.4: A modern vehicle is a network of sensors/actuators on wheels [1].

2.4.1 History and background:

The idea of inter vehicle communication (IVC) has gained considerable interest in the last few decades, which includes vehicle-to-vehicle (V2V), and vehicle-to-infrastructure (V2I) communications. In Europe for examples PROMETHEUS (program for European traffic with highest efficiency and unprecedented safety) project was created during 1987-1995 by eighteen European car manufacturers, incorporating more than forty research institutions in addition to state authorities; the main purpose of the PROMETHEUS project was automated driving (adaptive cruise control) for private cars. The next project DRIVE (dedicated road drive infrastructure for vehicle safety in Europe) was created during 1988-1994; the main purpose of the DRIVE project was to improve traffic efficiency and safety considering road-side infrastructure [1]. These projects led substantial progresses in European road transport; however the deployment of inter vehicle communication was not adequate enough to deploy, because of the need of a suitable wireless communication

technology.

When new wireless technologies have emerged, to support the revolution of vehicular *ad hoc* networks, the number of academic and industrial interests in VANETs has increased. and many efforts moved from the pure research stage to the experimental and execution stage. As a result a non-profit organization called C2CCC (car2car communication consortium) was created by Audi, BMW, Daimler Chrysler, Fiat, Renault, and Volkswagen. After that IEEE 802.11p task group was formed which is focused on providing wireless access technology for vehicular environment; in accordance with the official IEEE 802.11p working group project timelines, the standard is scheduled to be published in December 2010. Recently, Toyota and Microsoft have declared a 12 million dollar joint investment on including Microsoft's Azure cloud platform in upcoming Toyota vehicles for better telematics [5].

The main goal of these projects and consortiums are to increase road safety, increasing transportation efficiency, and reducing the impact of transportation on the environment.

2.4.2 Wireless communication technologies for VANETs

In recent years various wireless network technologies have been developed to offer different services, increased coverage area and data rates. In this introduction we will describe in overview:

2.4.2.1 Wi-Fi

(abbreviation of Wireless Fidelity) is a class of wireless (LAN) devices; the technology is based on the IEEE 802.11 standards [34]. Today, Wi-Fi devices can be found in many desktop computers, smart phones, printers, and indeed all modern laptops and (PDAs) are equipped with Wi-Fi technology. Wi-Fi's original purpose was mobile computing devices (for example laptops in LANs), but is now progressively more used for more purposes, including VoIP phones, games, and televisions and DVD players. Wi-Fi today is more

commonly used to provide an Internet LAN connection to Wi-Fi enabled devices like a computers, smart phones or PDAs. The above functions require the device to be within range of an access point.

The most common Wi-Fi standard IEEE 802.11g has a data transfer rate of around 54 Mbps; the range indoors is a maximum 150 feet (approximately 45 meters) and double that outdoors though, this depends on the conditions, like obstacles, power and weather. In Wi-Fi both 802.11b and 802.11g are using 2.4 GHz under the speed of 11 Mbps and 54 Mbps respectively, while 802.11n operates in both 2.4 and 5 GHz with theoretical speed 600 Mbps [35].

In Wi-Fi MAC (Media Access Controller) users are competing when they are connected to Wi-Fi access point, and users therefore have different levels of bandwidth. Wi-Fi however is short range (tens of meters) can be encrypted with WEP(Wired Equivalent Privacy) or WPA and WPA2 (Wi-Fi Protected Access encryption).

2.4.2.2 WiMAX

(Worldwide Interoperability of Microwave Access) is based on the IEEE 802.16 standard (also called Broadband Wireless Access). WiMax was formed in 2001 by the WiMax Forum, in order to endorse WiMax as a standard [36].

WiMax was described as a standard based technology for use as "last mile" broadband delivery rather than using wires. WiMax was planned to be used to link Wi-Fi hotspots together. WiMax 802.16 operates at range of 10-66 GHz and is classified as fixed wireless broadband; later, in 2004 802.16a was updated and operates at lower frequency range 2-11 GHz and is classified as fixed wireless broadband as well; finally in 2005 mobile wireless broadband was created under 802.16 e which operates at frequency range of 2-6 GHz [37].

WiMax technology has an advantage which is not affected by obstacles like buildings. This makes WiMax especially useful and cost-effective for countryside homes where set-

ting a traditional wire would be more difficult and very expensive.

WiMax is equipped with stronger encryption than Wi-Fi, and typically suffers less interference. WiMax speed in theory delivers up to 70 Mbps, and range coverage 112 Km. These numbers changes depends on the conditions, like obstacles, power and weather, expected values is 10 Mbps in 2 Km coverage area.

2.4.2.3 DSRC

In 1999, Dedicated Short-Range Communication (DSRC) spectrum was allocated by the U.S. Federal Communication Commission (FCC), for intra-vehicle communication at 5.9 GHz. The original goal was to make public safety applications possible in order to rescue lives and increase of quality of traffic flow [38] [39], but it is now increasingly used for comfort applications. In order to decrease the cost and support DSRC development, they permitted the private services as well. DSRC supports vehicle speeds up to 120 mile/hour, and the transmission range is between 300m and up to 1000m. This will enable operations related to the improvement of traffic flow, highway safety, and other intelligent transport system (ITS) applications.

DSRC spectrum is divided into seven 10 MHz wide channels as shown in Figure 2.5, the Channel 178 (control channel) is confined to safety communications only. The two channels at the edges of the spectrum are kept back for future advanced accident avoidance applications and high-powered public safety usages. The four channels (service channels) are left for both safety and non safety usage [2].

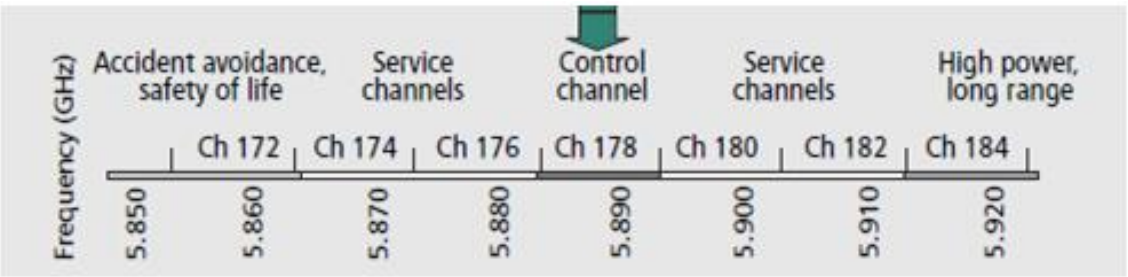


Figure 2.5: DSRC channel arrangement [2]

IEEE 802.11p basically is based on IEEE 802.11a; both of them operate in the 5.8/5.9-GHz band, IEEE 802.11a had been modified to cope with vehicular environment. IEEE 1609 working group endorsed all DSRC communication stack between the data link layer and applications, IEEE 802.11p is founded on an orthogonal frequency-division multiplexing (OFDM) PHY layer; however it uses 10-MHz channels in contrast to the 20-MHz channels for IEEE 802.11a. Therefore, data rates can range from 3 to 27 Mb/sec.

In 2006, The IEEE 1609 working group had completed the standards IEEE P1609.1, P1609.2 and P1609.4 for vehicular networks, and they released them for trial use [40, 41, 42]. A fourth standard, P1609.3 was released in 2007 [43], we give an overview for each standard:

- P1609.1 is the standard for Wireless Access for Vehicular Environments (WAVE) -Resource Manager, which specifies the services and interfaces of the WAVE resource manager application (enabling applications at remote sites to communicate with onboard units OBUs), describes the data and management services offered within the WAVE architecture, and defines the message data format. It also provides access for applications to the other architecture.

- P1609.2 is the Standard for Wireless Access in Vehicular Environments (WAVE) -Security Services for Applications and Management Messages which defines security, secure message formatting, processing, and message exchange.

- P1609.3 is the Standard for Wireless Access in Vehicular Environments (WAVE)-Networking Services which specifies transport layer and network layer services within a WAVE system, it also specifies Wave Short Messages, providing an efficient WAVE-specific alternative to IPv6 (Internet Protocol version 6). Further, this standard defines the Management Information Base (MIB) for the WAVE protocol stack.

- P1609.4 is the Standard for Wireless Access in Vehicular Environments (WAVE) - Multi-channel Operation which provides enhancements to the IEEE 802.11p medium access control (MAC) and physical layer (PHY) to support multi-channel wireless radio operations.

2.4.3 Characteristics of VANETs

VANETs have similarities with MANETs like low and variable bandwidth, short range connectivity, infrastructure-less, and self-organisation, but can be distinguished from MANETs by the unique characteristics such as high mobility and unreliable channels. These created research challenges such as routing protocols, data broadcasting, security issues. Most the routing protocols that have been used in MANETs cannot be applied in VANETs, because they suffered from poor performances caused by the fast movement in vehicles.
The most important differences between them is that vehicles in VANETs can move randomly but still predictably (restricted by geography of roads), even if they move at much higher speeds than traditional MANETs. Vehicles in VANETs are also have much higher power than in MANETs [44, 45, 1]. At the end of this section a comparison between the characteristics of MANETs and VANETs is provided as shown in the Table 2.1.

- **High and Dynamic Topology**: Because of the high speed and random of movement in vehicles, the topology of VANETs changes rapidly [46], for instance, assuming that all vehicles have the same transmission range which is 300 meters, a link can be formed between any two vehicles if the distance between them is less than 300 meters. In the worst possible scenario, if there are two vehicles driving in opposite directions, with the speed of 60 miles/hour (26.6 meters/second) consequently, the connection will last only for at most 11.2 seconds.

- **Random disconnection(frequent fragmentation) in network scale**: The vehicles in VANETs are free to move, hence they can dynamically enter or leave the network.

Consequently, the connectivity in VANETs would change frequently [47] which it will affect the network structure services, for example, in a low vehicles traffic density case, where there are two vehicles that need to communicate with each other, and there was only one vehicle in between them, if this vehicle changed its direction to another road, this will cause disconnection between these two vehicles, as well consider the obstacles (for example buildings, trees) that exist in the urban and crowded areas which they can prevent wireless signals, therefore the need to sustain the wireless connection must be improved by deploying more road side units or several relay nodes along the roads.

- **Mobility modelling and prediction**: Mobility and prediction model plays a significant role when designing protocols in VANETs, because of the high mobility of vehicles, high speed of vehicles and dynamic topology. Generally, we can predict the future position of vehicles if we know their speed and road maps, because the vehicles are restricted to pre-built high ways, roads, and streets [48].

- **High energy and computational power**: There are a common characteristic in VANETs which make them are distinguished from other networks; vehicles can have large energy, adequate storage, and high processor, powerful wireless transceivers and high data rate because nodes in VANETs are vehicles instead of small handheld devices as in MANETs.

- **Potentially large-scale and variable density**: In traditional wireless network the nodes number can be restricted or can be expected, in VANETs however the nodes number can be much larger and cannot be predicted, for example, assume an urban and crowded area with thousands of vehicles and a plenty of roads and streets, where the vehicles are located close to each other in the same area, and consider the case where vehicles are driving at period in the morning and evening of the greatest burden upon the channels of transportation in the same time (rush hour), in addition

VANETs can be extended in large areas as far as the road is available. All these facts increase the large-scale probability in VANETs [49].

Characteristic	MANET	VANET
Constrained Resource	✓	×
Topology	Dynamic	More Dynamic than MANET
Mobility Prediction	×	✓
Multi-hop	✓	✓
Limited Device Security	✓	×
Limited Physical Security	✓	✓
Short Range Connectivity	✓	✓
Infrastructure less	✓	✓
Low and Variable Bandwidth	✓	✓

Table 2.1: Comparison between characteristics of MANETs and VANETs.

2.4.4 The Challenges of VANETs

The key challenges in designing VANETs come from the decentralised nature, self-organisation, and self-management, since the opportunity of vehicle movement is very high. On top of that all communications are carried out through the short-range communication. These unique characteristics present appreciable challenges for VANETs such as:

2.4.4.1 Medium Access Control Protocols (MAC)

Designing MAC protocols in VANETs should be given more importance, due to the fast changes in topology and type of services; since the circulation messages in vehicles control channel are divided into two main types, they are classified based on how they generated [50]

- Periodic messages (beaconing safety messages) which are generated in order to make vehicles responsive to their environment by sending the vehicle's current sta-

tus to nearby vehicles like speed, direction, position. This type of broadcast messages can be used in connection with safety applications to make all vehicles benefit from the messages' content in order to avoid urgent or dangerous situations before they arise for example into intersection, collision warning, and blind merge warning.

- Event-driven messages which are emergency messages sent to other vehicles depending on the unsafe situations that have been discovered by sending vehicle's location, event type and the time. Generally, they are used in public safety application, for example, in approaching emergency vehicle warnings, emergency vehicle signal pre-emption, SOS services, post crash warnings, safety recall notices, emergency electronic brake lights. Hence, event-driven messages have to be given much higher priority than periodic and comfort messages, the industry and research community should give more attention in MAC layer to propose mechanisms and standards to control the handle of services and distinguish between them, to reduce the medium access delay, to efficient allocate shared channel access, and to increase reliability which is significant in case of safety application.

Katragadda *et al* [51] investigated reusing channels problem in VANETS. They introduced a novel location channel access (LCA) protocol in MANETs, which is suited for vehicle communication. Any vehicle is assigned to a channel in a dynamic manner, based on its geographical location and without using a central station.

The hidden terminals problem is the major limiting performance factor in VANETs. It happens when there are two nodes which they are out of radio range of each other, and are hidden to one another. Therefore they may access the medium at the same time, making the receivers channels experience a data collision [52]. A new (MAC) architecture protocol therefore emanated from the European project Car TALK2000 which is called ADHOC MAC [53]. The protocol uses a Basic Channel (BCH) in order to provide nodes

with the overlapping segments of information. The information on the Basic Channel (BCH) solves the hidden terminals problem by giving the nodes ability to receive information of all nodes in its second hop range. ADHOC-MAC is suitable for VANETs which use a Dynamic TDMA (time division multiple access) mechanism capable of providing immediate access, variable-bandwidth, and reliable channels, required for quality of service delivery.

The IEEE 802.11p task group is making a new PHY/MAC revision of the standard (802.11p), which is known as Wireless Access in Vehicular Environments (WAVE), in connection with MAC operations, WAVE uses CSMA/CA (Carrier Sense Multiple Access / Collision Avoidance) technique, whereas four channels for services and one channel for controlling the transmission [54].

2.4.4.2 Mobility Management

Besides the infrastructure less inter vehicular communication, VANETs applications may be extended by accessing internet services. The access is provided by Internet gateways (IGWs) installed along the Roadside Units (RUs). However, the Internet integration requires a respective mobility support. Since, vehicles in VANETs are highly mobile, they change their internet gateways (IGWs) frequently while getting Internet access services [55]; it is valuable therefore to have some mobility management schemes that give consideration to a vehicle's mobility characteristic.

Mobility management has to meet the following requirements seamless communication (making least disruption to the ongoing services of the roaming vehicle' users) irrespective of their current location; minimizing handover latency, supporting IP V6 and scalable and efficient mechanisms in terms of overhead, since VANETs can become very large including possibly thousands of vehicles. Seamless communication is considered crucial in terms of QOS to ubiquitous computing application particularly in real time services [56].

Due to the fact that *ad hoc* routing protocols do not support Mobile IP v6, these protocols are not suitable to provide mobility management in VANETs. So M. Bechler and L. Wolf [55] proposed a mobility management protocol called (MMIP6) relying on the principles of mobile IPv4 (32-bit), but designed to support IPv6 (128-bit) based on mobile nodes organised in *ad hoc* networks, MMIP6 uses foreign agents (FAs) which are installed at the internet gateways (IGWs), in order to hide the multi-hop capability of the VANET, and the vehicles appear as common mobile nodes.

2.4.4.3 Data Dissemination

In comparison with other networks, VANETs generally use a combination of broadcast, multicast, unicast dissemination messages between the vehicles, depending on the type of packets that we need to send. Vehicle can broadcast massages to all vehicles in all directions (one-all), or can be directed to a group of vehicles or one vehicle behind it (one-many).

Each vehicle broadcasts information about itself and the other vehicles it has knowledge about. In the meantime other vehicles receive this information and update their stored information correspondingly; during this period the receiving vehicles postpone their broadcasting information to the next period [57].

The security issue also is a real challenge in this context, since disseminating information securely between these nodes in such networks is a challenging task, particularly when the information is confidential. Revealing such information to anyone else other than the intended nodes could be highly damaging, especially in military applications where keeping the message secret from adversaries is essential [17, 16]. Therefore, any further research should consider controlling the data dissemination from source to destination in VANET.

The leakage of confidential information may cause real damage. To avoid this a mandatory access control mechanism used in Trusted Solaris Sun Microsystems [58] to

determine which information is accessible by users. Sometimes however the discretionary access mechanism is more reliable and can protect the confidentiality and the privacy of the information communicated by nodes in VANET. The discretionary access mechanism is more suitable as it does not involve the source level of administration and instead it confers upon the originators of data the discretion about to whom their information can be distributed.

Data dissemination takes place from a source to a target therefore the information in a source will be disseminated directly or indirectly to the target depending if the target is adjacent to the source or not. If a private information sent from the source (A) to intended list of targets only, therefore how can we prevent it from being leaked to undesirable entity(ies)?

The first technique that comes to mind is using encryption mechanisms or any type of access control mechanisms. These are very feasible approaches; however they have a limitation where the originator cannot update the restrictions which been made. These security methods are focused only on controlling the release of information; no restrictions however are placed on the dissemination of that information and thus these methods are insufficient to protect the originator confidentiality.

Therefore, the objective of this thesis is to control the data dissemination from source to destination in VANETs by automatically attaching policies along with messages to specify how the information can be used by the receiver, so as to prevent disclosure of the messages other than consistent with the requirements of the originator.

2.4.4.4 Security

Two reasons have come together to make the topic of security is important. Firstly, the explosive development in computer systems and connecting them by networks has increased the reliance of individuals on both the information stored and the information communicated via these systems. This has, however, increased the need to protect data

and resources from disclosure to other entities, and to protect such these systems from network attacks. Secondly, the cryptography mechanisms have developed increasingly and they are available to be enforced in these systems, leading to have the proper way of encrypting and decrypting data to the intended entities securely [3]. Since, security is an essential component in VANET, the striking features of Vehicular *ad hoc* network raise both challenges and opportunities in achieving security, unlike other traditional networks (wired) where nodes must have physical access to the network line or communicate through several lines of protection like firewalls and gateways. VANET uses the wireless medium so attacks on a wireless network can come from all directions and target any node. It gives high opportunity to be attacked if does not has certain security measurements. Consequently, link attack ranging from passive attack to active attack, message replay, message leakage, message contamination and message distortion can occur. All these mean that VANET does not have a clear line of defence, and every node must be arranged for the different kind of attacks [29].

Therefore, in order to achieve high survivability and scalability, VANETs should have a distributed architecture with no central administration, and of course the high mobility nature in VANETs should be considered, since prior trust cannot be counted upon in such networks; any intended solution to the security aspects therefore, should be adaptive 'on the fly' to these changes and should have the ability to deal with large networks as in VANET it may consist of hundreds or even thousands of mobile nodes.

The distinctive characteristics of VANETs bring a new set of essential challenges to security design such as open peer-to-peer network architecture, sharing of the wireless medium, large-scale density, the high relevance of vehicle geographic location and dynamic network topology. These challenges noticeably make the looking for security solutions that perform both data protection and applicable network performance are required.

Distributing information between vehicles in VANET over long ranges in such networks, however, is a very challenging task, since sharing information always has a risk

attached to it especially when the information is confidential.

2.5 Summary

This chapter presented a review of wireless *ad hoc* networks and mobile *ad hoc* networks (MANETs); it also described the characteristics, challenges, vulnerabilities of mobile *ad hoc* network, and then it illustrated the various advantages of MANET and numerated the applications of MANET. This chapter also presented an introduction of the vehicle *ad hoc* networks (VANETs), history and background, also it described the characteristics, challenges, vulnerabilities of vehicle *ad hoc* network, Finally, this chapter provided a comparison between characteristics of MANETs and VANETs as described in the Table 2.1.

Although VANETs are interesting for many on road applications, they nevertheless have several challenges, as shown in Section 2.4.4. Each of these challenges can be considered as a separate research area needing intensive investigation. Researchers investigated the security issues in both MANETs and VANETs and they proposed many solutions; the next chapter will investigate these issues by discussing the security requirements, security attacks, and security mechanisms used in the literature to make a secure communication between the entities.

Chapter 3

Review of security in VANETs and MANETs

Objectives:

- Define the basic security concepts and requirements.
- Present the access control models.
- Present an overview of the network security.
- Present an overview of the cryptography background.
- Present the related work in privacy and confidentiality in (VANETs) and (MANETs).

3.1 Introduction

Normally in addressing network security, three significant issues need to be considered in the system: security requirements, security attacks and security mechanisms. Security

requirements take account of the functionality required to provide a secure networking system, whereas the security attacks include the techniques that might be carried out to break these requirements. Finally, the security mechanisms are the fundamental elements used to enforce the security requirements. Section 3.2 therefore presents the security requirements: authentication, authorisation, access control, privacy, confidentiality, availability, survivability, data integrity, and non-repudiation. Section 3.4.1 presents the access control models: Discretionary Access Control (DAC), Mandatory Access Control (MAC), and Role-Based Access Control (RBAC).

Attacks on VANETs can be divided into two types: passive and active attacks. Section 3.3 therefore presents both types of attacks: passive attacks are hard to detect because they are based on 'snooping' on transmitted packets between entities, whereas in active attacks the attacker tries to change or destroy the data being transmitted within the network. External active attack and internal active attacks are also described in Section 3.3. Section 3.4 presents a set of security mechanisms which can be used to enforce the security requirement: cryptography, digital signature, access control, authentication, traffic padding, notarization, and routing control.

Section 3.5 presents an overview of the cryptographic background to understand work already done on securing VANETs, as well as the recent research. Two main types of cryptographic algorithms are used in cryptography: symmetric key algorithms presented in Section 3.5.1 in which sender and receiver both use the same key (secret key) for encryption and decryption, whereas in asymmetric key algorithms presented in Section 3.5.2, the sender and the receiver uses two different keys for encryption and decryption. Public Key Infrastructure (PKI), Digital signature, and Digital Certificate will also be discussed in detail in Sections 3.5.2 and 3.5.3 respectively.

Section 3.6 presents a critical review of the security issues in both MANETs and VANETs. It also provides a survey of existing solutions in VANET to highlight a particular area, not been addressed up to now: controlling the information flow in VANETs,

aimed to provide an architecture (to be described in Chapter 4) that allows the policy-based framework to control the dissemination of data communicated between nodes. This is to ensure that data remains confidential not only during transmission but also after it has been communicated to another peer.

3.2 Security Requirements

The security requirements are specified by standards of several organisations such as the International Telecommunications Union (ITU-T), which defines the security requirement as a set of services provided by the system which ensures the adequate security level for data communication, by giving specific protection to system resources. ITU-T, in their recommendation X.800 and X.805 defines these requirements as follows [21, 59, 26, 3, 60]:

- **Authentication:** Authentication verifies the identity of each vehicle in VANET and its eligibility to access the network. This means that vehicles in VANETs are required to verify the identities of the communicating entities in the network, in order to ensure that they are communicating with the correct entity (vehicle or road side unit). Thereafter, vehicle reactions to events such as car crashes and road congestion warnings etc.) should be based on authenticated messages, hence, the need to identify the senders of these messages is required. This is an essential and difficult requirement to satisfy. If the authentication stage was not fulfilled, no further requirements would be properly implemented. For example, if two entities are using symmetric-key encryption for securing the communication and one of these entities become compromised caused by the lack of authentication, then all encrypted material such as the shared key and the encryption algorithm will be available to that adversary entity. Techniques to securely authenticate vehicles are essential to the operation of VANETs.

- **Authorisation and Access Control:** Each vehicle in VANET is required to have access to shared resources, services and personal information on the network. In addition, vehicles should be capable of restricting each other from accessing their private information. There are many techniques that can be used for access control such as Discretionary Access Control (DAC), Mandatory Access Control (MAC) and Role Based Access Control (RBAC) (to be discussed in Section 3.4.1). Traditionally authorisation policies are related to auditing techniques to track resource usage and deduce statistics about nodes in the network.

- **Privacy and confidentiality:** Each vehicle has to secure both the information that is exchanged between it and others, and secure the location information and the data stored on these vehicles. Privacy means preventing the identity and the location of the vehicle from being disclosed to any other entities, while confidentiality means keeping the secrecy of the exchanged data from being revealed to those who do not have permission to access it. Data confidentiality can be applied by using any encryption techniques based on secure key management system. In contrast, protecting the users' privacy in VANET such as driver-id, the license plate, position, and travelling routes needs something more than encrypting the data, indeed sophisticated mechanisms are required to conceal those users' attributes such as using a pseudonym technique.

- **Availability and survivability:** The network services and applications in VANET should be accessible when needed, even in the presence of faults or malicious attack such as denial-of-service attack (DoS), while survivability means the capability of the network to restore its normal services under such these conditions. These two requirements should be supported in VANET.

- **Data integrity:** The data transmitted between vehicles in VANET should be received by the intended entities without been tampered with or changed by unautho-

rised modification. This requirement is essential especially in military, banking and aircraft control systems, where data modification would cause potential damage.

- **Non-repudiation:** This ensures that vehicles in VANET when sending or receiving data-packets should not be able to deny their responsibilities of those actions. This requirement is essential especially when disputes are investigated to determine the entity which misbehaved. Digital signature techniques are used to achieve this requirement to prove that the message was received from or sent by the alleged vehicle.

3.3 Security Attacks

Attacks on VANETs can be divided into two types, namely, passive and active attacks [3, 22]. Passive attack are based on 'snooping' upon transmitted packets between entities; the goal of the attacker is to acquire data that is being sent without modifying it, but not to stop the operation of the network, and thereby breaching the confidentiality requirement. Passive attacks are hard to detect because the data packets are sent and received normally and neither the sender nor receiver is aware that the attacker has read the packet or has intercepted the traffic pattern. Therefore, it is more important to prevent such this attack rather than to detect it; the prevention mechanisms involved use encryption algorithms to encrypt the data being transmitted, thereby preventing attackers from acquiring any useful information from the data overheard.

In contrast, in active attacks the attacker tries to change or destroy the data being transmitted in the network, thereby interrupting the normal operations of the network. Active attacks can be divided into two types, external and internal attacks. External attacks can be executed by nodes from outside the network. This kind of attack can be prevented easily by using authorisation and access control mechanisms. By contrast, internal attacks are very difficult to prevent and can cause severe damage, because they come from

malicious nodes who are already authorised inside the network. The security architecture proposed for VANET should therefore provide a comprehensive end-to-end security solution in order to prevent/detect data leaks. This work identifies the security requirements in VANETs, their objectives, and the methods by which they could be applied to VANETs, therefore a set of security mechanisms needs to be defined. Cryptography is one of the most powerful tools that can be used to achieve most of the security requirements, such as peer entity authentication, data origin authentication, data confidentiality, and data integrity as shown in Figure 3.1. The next section will show some security mechanisms that are needed to understand the work that has been done to manage and secure VANETs.

3.4 Security Mechanisms

These are the security mechanisms as they are defined in X.800 [3]:

- Cryptography (Encipherment): In this mechanism data is transformed or encrypted into a not understandable format at the sender side, by using mathematical algorithms based on one or two encryption keys, and then it is decrypted to readable format again at the receiver side.
- Digital Signature: In this mechanism extra data are added to the message to give the receiver a 'guarantee' that the data come from a legitimate sender, and was not altered in transmission (integrity).
- Access Control: A mechanism to enforce access rights to resources.
- Authentication Exchange: A mechanism destined to ensure the identity of an entity.
- Traffic Padding: A mechanism destined to frustrate traffic analysis attempts by adding extra bits into gaps in data packets.

- Notarization: A trusted third party (certificate authority) which is trusted by all parties to facilitate interactions to assure certain properties of data exchange.

- Routing Control: a mechanism used to select special securing routes for specific data and enable routing changes accordingly, particularly when a break of security is suspected.

3.4.1 Access Control

Protecting resources and information from unauthorised access is an important cornerstone in any information security system, this can be done by controlling how these resources and information can be accessed, otherwise unauthorized access or disclosure of confidential information especially in military systems would be an extremely damaging and fatal. So the need for access control arose because it is the first line of defence against unauthorized access to network resources and information. The purpose of using access control is to give the ability to control, monitor, restrict, and protect the confidentiality of resources and to define how users (subjects) can interact with other systems or resources and information (objects); the subject can be a user, program, or process that accesses an object, where the object can be a computer, database, or file [61]. Access control models had been divided into three models based on the mechanisms of setting the access to these objects; each model type has a different method to control accessing objects by subjects. This section explains these different models as we describe them in below:

1. Discretionary Access Control

   Each resource (object) in Discretionary Access Control (DAC) has an owner who specifies and controls of which users (subjects) can access his resource (object), and states the permission type the subjects may have on this object. In this kind of access control model the access is restricted to the subjects based on the authorisation granted by the initial owner of this object. The initial owner of an object

is the subject who created it [62]. It is called Discretionary Access Control (DAC) because of the access is based on the discretion of the owner (subject); the user in this model is allowed to specify the type of access to his object.

Access control lists (ACLs) is a form of Discretionary Access Control (DAC) which has been used in various operating systems such as Microsoft Windows, Linux, and Macintosh systems, the properties of any file in these systems have an options that allow you to control and choose which users can get an access to this resource and what the permissions may they have.

2. Mandatory Access Control

   Subjects and data owners do not have an option to specify who can access their resources, the administrator makes that instead. Both users (subjects) in Mandatory Access Control (MAC) model have a security clearance (secret, top secret, confidential, and so on), and also data (objects) classified similarly to security clearance, these security clearances are stored in security labels, which are given to subjects and objects [62].

   In Mandatory Access Control (MAC) For example, a user (subject) may have a security clearance of secret, and the data (object) to which user has been requested has a security clearance of top secret, then the user will be rejected to access this data because his security clearance (secret) is lower than the classification of the data (top secret), in order to get access to such a resource the subject must have a security clearance which is equal or higher than the security clearance of the object.

   This type of access control model has been used in applications where classification of information and confidentiality is essential, especially in military system where accessing the information is allowed for a specified set. Mandatory Access Control (MAC) model used in Unix systems, and recently SE Linux which was developed by the National Security Agency (NSA) [63].

3. Role-Based Access Control

   In role-based access control (RBAC) model the subject will be given an access to the object based on his role or functional position (position assigned to a particular person or thing), this model is also called nondiscretionary access control, because allocating a user to a role is obligatory. This means that user does not have the choice to specify what role he will be given.

   Role-based access control (RBAC) model is more complex than Discretionary Access Control (DAC), instead of specifying the access control at the object level with Access Control List (ACLs) by the subject, the administrator in (RBAC) is required to transform the policies into permission as soon as setting (ACLs). Using (RBAC) model in such these companies where the members of staff can come and leave the company in a dramatic manner is a paramount system, better than using (DAC) and (MAC) models. For example, if an x is an employee assigned to contractor role after that x left the company, then y become his replacement in this way the new replacement employee can be easily mapped to this role by the system administrator [64].

As we see from Figure 3.1 the confidentiality requirement can be solved by using encryption and routing control mechanisms, otherwise disclosing private information by a malicious node (inside the network) to unauthorised nodes will cause a fatal problem and data will be leaked. Therefore, encipherment tools (to be described in Section 3.5) are widely used in security systems and solve part of the problem by encrypting data exchanged between entities. Using a mechanism based on access control to ensure confidentiality, however, has still not been used, so this work intends to use access control mechanism especially Discretionary Access Control (DAC) to ensure data confidentiality and privacy in VANETs.

	Mechanism				
Requirement	Encipherment	Digital Signature	Access Control	Data Integrity	Routing Control
Authentication	Y	Y			
Access control			Y		
Confidentiality	Y				Y
Data integrity	Y	Y		Y	

Figure 3.1: Relationship between security requirements and mechanisms [3]

Most of the previous security solutions used in VANET focused on conventional cryptographic techniques which are the most powerful tools that can be used to achieve most of the security requirements such as authentication, data confidentiality, data integrity and non-repudiation. The next section, therefore, will give an overview of the cryptographic background to understand work already done on securing VANETs and MANETs.

3.5 Cryptographic Background

Cryptography [3, 59] is the science of encoding in cipher using specific mathematics and algorithms to encrypt and decrypt data in order to ensure secrecy and/or authenticity of messages. Using cryptography data are transformed or encrypted to a format incomprehensible to third parties at the sender side by using mathematical algorithms based on one or two encryption keys. It is then decrypted to a readable format again at the receiver side. This enables nodes to transmit secret information through insecure networks, so that it cannot be read by any node except the intended node. The main goals of cryptography are to ensure confidentiality, integrity, authentication and non-repudiation security requirements.

In cryptography, the input to an encryption algorithm or the output of a decryption algorithm is called plaintext. Before data are sent from one node to another, the plaintext is converted into an unintelligible form which called ciphertext by the process of encryption using certain algorithms or functions. The intended receiver can then decipher/decrypt the

ciphertext back into original text (plaintext) by the process of decryption. Mathematically, if M represents the plaintext message and C represents the ciphertext message as shown in Listing 3.1, we can say then:

Listing 3.1: Encryption and decryption formulas

```
Encryption :: E(M)= C
Decryption :: D(C)= M
```

The encryption and decryption algorithms are based on keys, which are small amounts of information used by the cryptographic functions. Keys must be distributed and kept secure to ensure security of the system; this is why they are called secret keys. The security of administering the keys in cryptography science is called key management. Two main types of cryptographic algorithms are used in cryptography: symmetric key algorithms, where sender and receiver both use the same key (secret key) for encryption and decryption, whereas in asymmetric key algorithms, sender and receiver uses two different keys for encryption and decryption. These two algorithms will be discussed in the following Section 3.5.1 and 3.5.2 respectively. Digital signature, digital certificate, Public Key Infrastructure (PKI) also will be discussed in following Sections 3.5.2 and 3.5.3 respectively.

3.5.1 Symmetric Key Algorithms

Symmetric Key Algorithms [3, 59] are those kinds of cryptographic algorithms based on the existence of a shared key (agreed between the participants' nodes) in both the sender and receiver sides. The key used in such symmetric encryption/decryption algorithm is required to be exchanged through a secured channel. Both participants' nodes must share the same key before starting to communicate; this key can be used in both encryption and decryption processes (K) and it must be maintained secret to protect the communication

afterwards. Symmetric key cryptography is the process where both sender and the receiver use the same secret key to encrypt and decrypt. An example is depicted in Figure 3.2 where Alice ciphers the plain text message (m) using the shared secret key (k), as a result the plaintext is changed to a ciphertext (c). Bob wants to receive the message sent from Alice in a readable format, thus he deciphers the received ciphertext (c) using the same secret key (K) which is been used in the encryption algorithm at Alice's side to change it back again to a readable format (m).

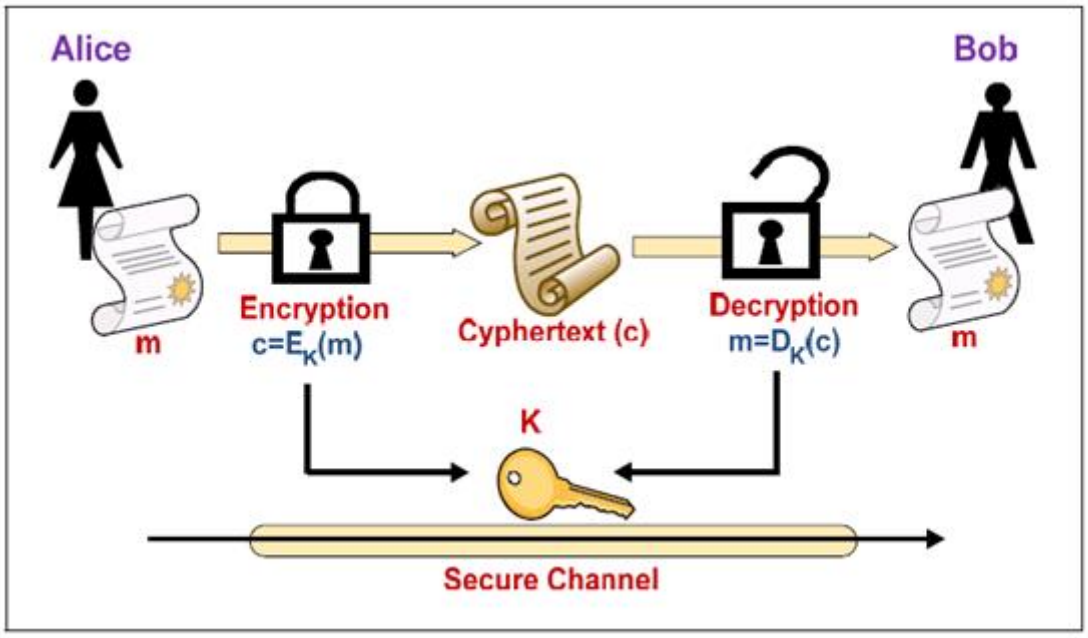


Figure 3.2: Symmetric key scheme [3]

Symmetric-key algorithms can be divided into two types: stream ciphers and block ciphers. Stream ciphers encrypt a byte of the plaintext message one at a time, whereas block ciphers encrypt a number of bytes as a single unit. Blocks of 64 bits have been previously used. Currently, however, AES (Advanced Encryption Standard) has been approved by National Institute of Standards and Technology (NIST) in 2001; it uses 128-bit blocks which replaces the commonly used Data Encryption Standard (DES) [65, 66].

Generally, symmetric cryptography is much faster to execute than asymmetric cryptography. Because symmetric key algorithms require a secret key to be shared between

the participants' nodes, however, any other node which 'knows' the shared secret key can decipher the messages sent in the network. The drawback of symmetric-key algorithms is thus that if the shared secret key is compromised, all messages can be deciphered which can make the whole system susceptible to attack. Therefore, the secret key in such a cryptography type needs to be altered frequently and stored securely during the key distribution process. Data integrity and non-repudiation requirements are solved by hash functions and digital signatures respectively. Key-management issues are solved by RSA (Rivest, Shamir and Adleman) encryption and by DH (Diffie-Hellman) key agreement algorithm [66].

3.5.2 Asymmetric Key Algorithms

Asymmetric Key Algorithms [3, 59] are those kinds of cryptographic algorithms in which encryption and decryption are carried out using two different keys, one of which is referred to as the public key and the other is referred to as the private key. Asymmetric key algorithm is also termed a public key cryptography using two keys. One key is used for ciphering and the other one is used for deciphering. The decryption key is kept secret, therefore, it is termed the "private key", whereas the encryption key is known to all participants' nodes to be able to send encrypted messages, therefore it is termed the "public key". Every node that has the public key can send encrypted messages to the node that possesses the private key, but message encrypted with the public key can be decrypted only with the corresponding private key. Both keys are related mathematically; the private key however, cannot be derived from the public key. The key management issue in symmetric key algorithm solved by public key cryptography (asymmetric key) after the idea of asymmetric algorithms was first published in 1976 by Diffie and Hellman [67].

An asymmetric key encryption scheme is depicted in Figure 3.3. At the start, both Alice and Bob should have an authenticated pair of public and private keys. If Alice wants to send a ciphered message m to Bob, she needs to know Bob's public key (PK[Bob]) in

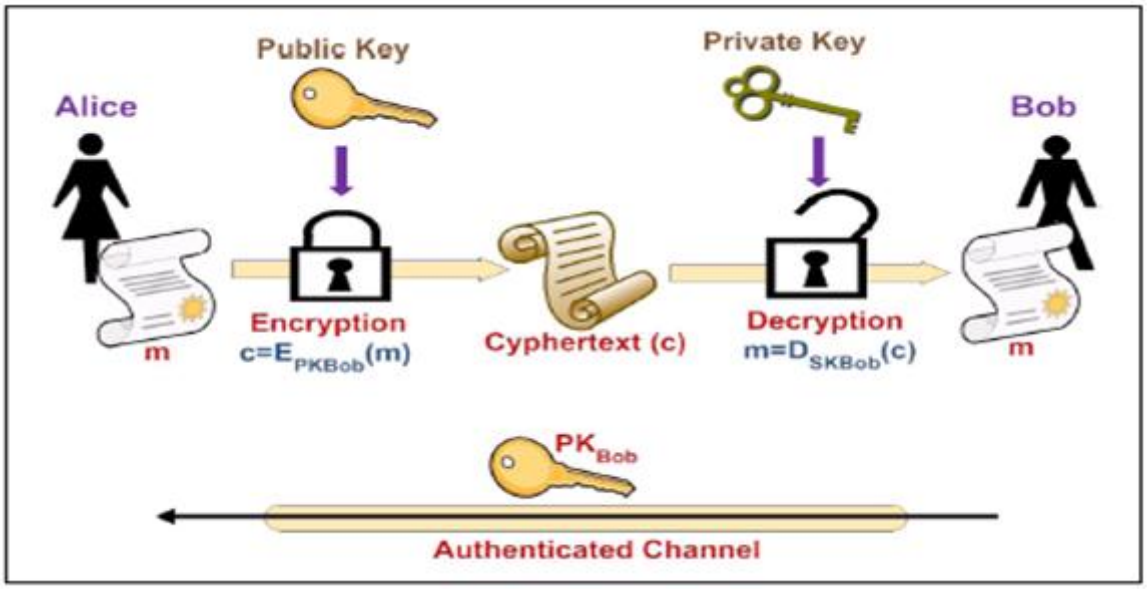


Figure 3.3: Asymmetric key scheme [3]

order to encrypt the message m and change it to a ciphertext (c). Bob is able to decrypt this ciphertext (c) using his private key (SK[Bob]) which is secret and known only to him. Public key cryptography can be divided into two subtypes which are:

- Public key encryption: a form of cryptographic system in which encryption and decryption are performed using two different keys, one to encrypt the plaintext, and another one to decrypt the ciphertext. Neither key will do both functions. When a message has been encrypted with a receiver's public key, it can be decrypted by only that receiver which has the correspondent private key. In this way the confidentiality requirement can be ensured.

- Digital signature: an approach to authenticate the identity of the sender which enables the sender of a message to attach a piece of code that functions as a signature. The signature is created by calculating the hash of the message and encrypting the message with the sender's private key. The sender's signature guarantees the source and integrity of the message sent to other nodes. Therefore, any message signed with the sender's private key can be taken to mean that the message has not been

tampered with. In this way the authenticity, integrity and non-repudiation requirements can be ensured [68].

The main problem when using public-key cryptography is how to prove that a certain public key is genuine (belongs to the claimed node) or not, and has not been tampered with or changed by an unauthorised third party. This problem is solved using a public-key infrastructure (PKI) approach, which is an arrangement that matches public keys with respective nodes identities *via* a one or more group(s) of third parties, which is termed as certificate authority (CA) to authorise the ownership of key pairs [69].

Rivest, Shamir and Adleman [70] proposed a novel algorithm for obtaining digital signatures and public-key cryptosystems in 1978 which was termed afterwards as RSA. This is an example of public key cryptography based on the integer factorisation difficulty, in RSA (m) plaintext message can be encrypted or ciphertext can be decrypted using the following formula as shown in Listing 3.2:

Listing 3.2: RSA encryption and decryption formulas

```
c=m^e mod n
m=c^d mod n
```

One of the advantages of using public key cryptography [71, 72] is to provide a technique for implementing digital signatures. Digital signatures give a guarantee to the receiver of a particular message that it has been sent from a node of authenticated identity, and also to ensure that the content of message is received to the intended node without it having been tampered with or changed by unauthorised modification. Digital signatures thus ensure authentication and data integrity system requirements. A digital signature also ensures non-repudiation requirement, in which the sender should not be able to deny its responsibilities of some actions. This requirement is essential especially when disputes are investigated to determine which node misbehaved. Therefore, digital signature technique is used to achieve this requirement to prove that the message was received from or

sent by the alleged node.

A digital signature acts as the traditional handwritten signature. The handwritten signature however, can be imitated, whereas a digital signature is better than handwritten because it is harder to be counterfeited. It also certifies that the content of the message is received intact as well as the identity of the sender is authenticated.

As depicted in Figure 3.4 as a replacement of encrypting message using the receiver node's public key, digital signature encrypts the message using the sender's node's private key. Therefore, the same message can be decrypted using the sender's public key, so that tells the receiver node that the message originated from that sender. As depicted in Figure 3.4, if Alice wants to send an encrypted message m to Bob signed by Alice's identity, she calculates the hash digest of the message m using a specified hash function. Alice then encrypts this digest using her private key (SK[Alice]) to produce the signature and sends it with the message to Bob. When the message received at Bob's end, he recalculates the hash digest of the received message using the same hash function which was implemented at Alice's side and compares it with the hash digest generated from decrypting the signature using Alice's public key of (PK[Alice]). If both digests match that means the message m must have been created from Alice and it has not been changed or tampered with during transmission.

3.5.3 Digital Certificate

The Digital Certificate is an electronic document used for establishing the credentials of a node (i.e. certify the identities of nodes) which combines a digital signature to match between the public key and the nodes' identification to verify the nodes' identities. It is issued and certified by one or more certification authorities (CAs) [73, 74]. In public cryptographic system nodes need to make sure that they are ciphering to legitimate identities of nodes. The important of digital certificates comes from protecting the network from the man-in-the-middle attack scenario. The man-in-the-middle attack is a potential threat

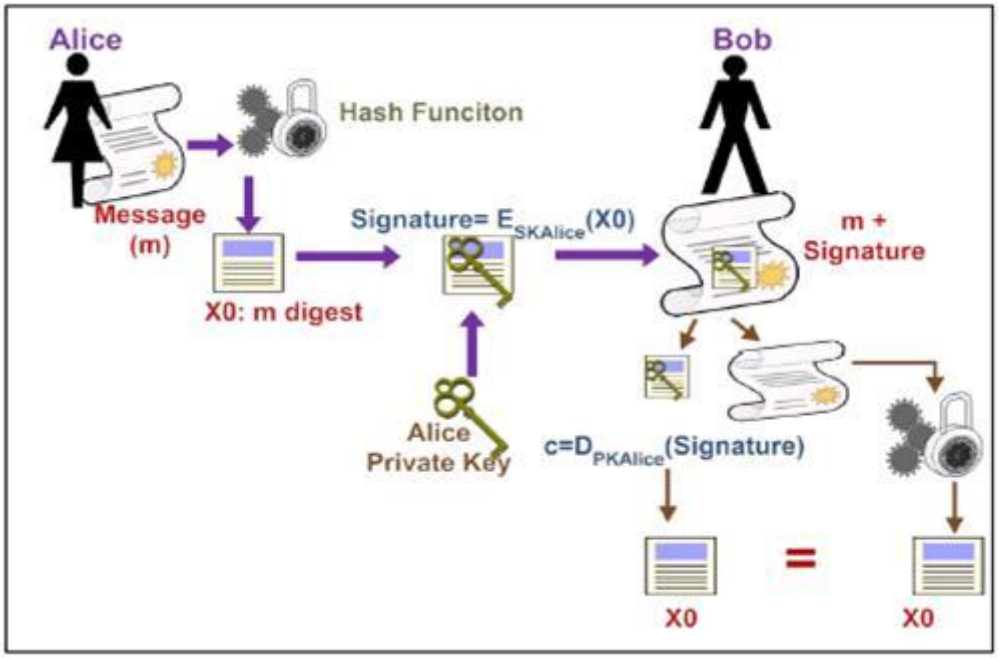


Figure 3.4: Digital Signature example [3]

in such environments where keys exchanged between nodes and servers can enable the attacker to insert, read, and modify messages sent among two victim nodes without either node being aware of the connection they have used has been compromised. In this type of attack the attacker makes autonomous links with the victims to play with messages sent between them. Victim nodes believe that they are communicating directly and securely between each other, when in fact the entire connection is managed by the attacker [3].

For instance, if Alice wants to send a message to Bob securely, she will ask for Bob's public key. If Emma (the attacker) can find the public key of Bob and be able to intercept the messages sent between Alice and Bob, the man-in-the-middle attack can be mounted. First, Emma will impersonate the identity of Bob and send her public key to Alice as if it were Bob's public key. This will make Alice believe that it belongs to Bob and she will use it to encrypt the message and then send it back to Bob. This encrypted message will be intercepted by Emma [75].

This time Emma ciphers the message using her private key, keeps a copy of it and

re-ciphers it using the correct public key of Bob. Once the message is received by Bob, he will believe that it was sent by Alice. This scenario shows simply the need for some method of ensuring that Alice and Bob have genuinely used each other's public keys and not the attacker's public key. If not, they will remain vulnerable to such an attack. Digital certificates therefore are used to prevent this kind of attack happening. They are like the traditional identification cards such as passports and drivers' licenses which can verify the identities of their owners'. Similar to traditional identification cards which are issued by identified government authorities, digital Certificates in MANETs and VANETs are also issued by trusted third parties. A digital Certificate verifies the identity of the node but instead of including a photo and a signature of the certificate's owner, digital certificates bind the owner's public key to the owner's private key. Therefore, digital certificates contain node identification, serial number, expiry date, public key, and digital signature of the certification authority (CA) which issued the certificate. This signature in the certificate act as attestation by the certificate's signer that the information of node and the public key belong together [76].

In order to make a digital signature, a certification authority (CA) employs its private key to digitally sign each certificate it issues. The CA creates a message digest from the certificate using a specified hash function, and then encrypts this digest with its private key, and inserts the digital signature inside the certificate. When the certificate is received at the node, the node recalculates the hash digest of the received certificate using the same hash function which was implemented by the CA, and then compares it with the hash digest generated from decrypting the certificate using the CA's public key to verify the certificate's integrity. If both digests match, that means the certificate must have been created from the CA and has not been changed or tampered with during transmission. If they do not match then the certificate is not original or has been issued from a non certified authority [73].

3.6 Related Work

Vehicular ad hoc network (VANET) is an emerging new technology that promises to be an aid to road safety and efficiency of users of vehicles. It is derived from mobile ad hoc network (MANET), which can provide wireless communication services between vehicles and adjacent road side units (RSU). It is a promising technology for future 'smart' vehicle systems and intelligent transportation systems (ITS). VANETs promises to enhance passenger comfort by providing services such as exchanging traffic information, weather information, interactive communication and offering internet access.

Nowadays, vehicles manufacturers are progressively trying to improve safety and optimize traffic by utilizing the information technology (IT) in the vehicles industry [5]; using this technology, vehicles hopefully will have enhanced awareness of their environment through communication with other vehicles and/or with roadside units. Vehicles today are becoming "computers on wheels"; or rather "computer networks on wheels": modern cars may have many interconnected processors; short range wireless interfaces, event data recorders (EDRs) which are similar to the "black boxes" used in aviation. In addition to EDRs modern cars in general are also fitted with front and rear sensors and a GPS (Global Positioning System) device in order to provide vehicles' location, speed, current time and direction.

These networks are therefore particularly useful to those mobile users who need to communicate in situations where no fixed wired infrastructures are available. However, the salient feature of creating a network 'on the fly' without requiring any prearranged infrastructure gave both MANETs and VANETs an appreciated interest in both industrial and military systems. The key challenges in MANETs and VANETs design come from the decentralised nature, self-organisation, self-management, and also the fact that all communications are carried over wireless links in short-range communication [25, 47, 46, 48, 49]. These unique characteristics present appreciable challenges for both MANETs

and VANETs [26, 27, 28, 29].

3.6.1 Industrial Projects

The idea of Inter Vehicle Communication (IVC) gained considerable interest in the last few decades. In Europe for examples PROMETHEUS (program for European traffic with highest efficiency and unprecedented safety) project was created during (1987-1995) by eighteen European car manufacturers the main purpose of PROMETHEUS project was automated driving (adaptive cruise control) for private cars. The next project DRIVE (dedicated road drive infrastructure for vehicle safety in Europe) was created during (1988-1994), the main purpose of the DRIVE project was to improve traffic efficiency and safety considering road-side infrastructure, then a non-profit organization called C2CCC (car2car communication consortium) was created to consider the security issues [77]. While, the early previous projects mainly focused on the feasibility of VANET, increasing the road safety, increasing the transportation efficiency, and reducing the impact of transportation on the environment. Recently some projects in Europe considered the security issues namely the Network on Wheels (NoW) Germany's nationally funded project, which started in June 2004 to look at potential attacks on such networks and devising methods and mechanisms to protect them [78], SEVECOM (Secure Vehicular Communications) initiated in 2006 to define the security architecture of such networks and to propose a roadmap for the deployment of security functions in these networks [79], PRECIOSA (Privacy Enabled Capability in Co-operative Systems and Safety Applications) lunched in 2008 to define an approach for the privacy evaluation of co-operative systems in terms of communication privacy and data storage privacy and investigate specific challenges for privacy [80], EVITA(E-Safety Vehicle Intrusion Protected Applications) co-funded project initiated in 2008 by the European Union within the Seventh Framework Programme for research and technological development, the objective of EVITA is to design, verify, and prototype an architecture for automotive on-board networks where

security-relevant components are protected against tampering and sensitive data are protected against compromise [81], OVERSEE (open Vehicular Secure Platform) lunched in 2010 to contribute to the efficiency and safety of road transport by developing the OVERSEE platform, which will provide a secure, standardized, and generic communication and application platform for vehicles [82], members of these projects have worked together to combine and extend their results in PRESERVE project (Preparing Secure Vehicle-to-x communication systems) [1] to design, implement, and test a secure and scalable vehicular security subsystem for realistic deployment scenarios.

3.6.2 Academic Research

In comparison with wired networks where the devices must have a physical access to the network medium, mobile *ad hoc* networks and vehicular *ad hoc* network have no apparent secure boundary. There is no need for the attackers to have a physical access to the network; once the attackers are in the transmission range of any other devices, then they can join and communicate with other devices. According to the nature of mobility in *ad hoc* networks, liberty to join, moving outside and inside the networks makes MANETs and VANETs vulnerable to attacks, which can result from any device in the same transmission range [25]. In comparison with wired networks where the nodes can get electrical supply directly from the power points, in MANETs nodes are generally operated by small batteries with limited lifetime. This makes nodes unable to perform intensive computations over prolonged periods of time. An attacker on the other hand is typically able to provide sufficient power-supply and thus must be assumed to be able to perform intensive computations [83], meaning that attack and defence in these networks is not equally matched. The lack of centralized management in MANETs and VANETs makes detection of attacks difficult, since they are highly dynamic and large scale therefore they cannot be easily monitored; benign (non-malignant) failures in MANETs and VANETs are also fairly common, for example transmission destructions and packet dropping. As a result,

malicious failures will be more difficult to discover. Since security is an essential component in a hostile environment, these unique characteristics of MANETs and VANETs raise challenges that security requirements must address [84, 85].

There has been appreciable work by the research community [84, 86, 87, 88, 89] in message encryption, digital signature, and key management. Many challenges particularly related to the privacy and data confidentiality of originator, however, remain to be solved. These available approaches which have been used in MANETs and VANETs such as access control, digital signature, and encryption focused only in securing the channel during the transmission, however how these nodes act after and use this information has been mostly neglected.

Existing approaches in security have been applied to VANETs: traditional cryptography solutions for example, are using Public Key Infrastructure (PKI) to ensure authentication, confidentiality, privacy, non repudiation and integrity requirements in VANET communication [90]. Other researchers [91, 92], further developed this approach to counter the security threats more effectively. Wasef *et al* [93] proposed a mechanism for mitigating the effect of DOS attacks in VANETs, this mechanism complements the Public Key Infrastructure solutions to secure VANET. At the beginning they set some security requirements: authentication, privacy, non repudiation, access control and availability, however confidentiality was not been taken into consideration in their work; as a result they only addressed the availability, authentication and non repudiation requirement, some issues related to privacy and how to protect the location of vehicles against legitimate insiders in traditional certificate-based PKI was not addressed.

As conventional security solutions found in the literature depend on centralized infrastructure to manage security tasks such as key assignment and management, they may not suit VANET because of its high mobility and random disconnection. Yeh *et al* [94] proposed dynamic establishment of secure communications in VANET (DESCV) based on decentralized Inter Vehicle Communication (IVC), without using fixed infras-

tructure (RSUs). Other work suggested that vehicles should be connected to road side units (RSUs), for example Lin *et al* [95] proposed a social-tier-assisted packet forwarding protocol (STAP) to achieve receiver-location privacy preservation in VANETs, however this solution increased the cost deployment of RSUs infrastructure.

Solutions using PKI therefore, can provide end-to-end secure communication channels, these approaches are mainly focused on message confidentiality, integrity and non-repudiation; they do not consider, however, controlling the message dissemination after it being sent to recipients; thus the management of data confidentiality, privacy concerns and how these certified entities act is left to the application layer [9].

In addition, a few academic papers have been published by Raya's group *et al* to provide a general survey of crucial security issues, giving an overview of challenges, adversaries, attacks, properties of VANET, and useful security mechanisms to design robust solutions [6, 7]. In later research [96] they proposed a secure architecture in VANET to address these issues.

Fuentes *et al* [97] and Mishra *et al* [98] reviewed the security developments in VANETs, and analysed the security mechanisms previously proposed to achieve the security requirements. Since the confidentiality requirement is of particular interest to this work, three main solutions have been proposed: the first one uses Road Side Units (RSUs) to control a region. Verma and Huang [99] proposed a framework called Secure Group Communication (SeGCom), to provide support for V2I communication. In their work, they assumed that RSUs are connected to each other, to share the information of vehicles, and they also assumed that roads are partitioned into multiple segments of equal length and each segment is monitored by a RSU, so if any vehicle 'wants' to enter a specific region, it should register within that RSU. Once the registration (which involves mutual authentication) has been processed, the RSU sends a symmetric key (shared key) to the vehicle, and it is used to encrypt the communication among vehicles in that region. In VANETs however the number of nodes can be large and unpredictable therefore this can

cause overhead in the RSU.

The second solution is based on establishing self-organising geographical regions [100]. Any vehicle can become a member of a group depending on its location. However a group leader is needed (e.g. the most centered vehicle). The leader role is in charge of creating and delivering the symmetric key. In contrast to the previous solution, this mechanism allows a group to have a longer communication period (it is not constrained by the range of the RSU).

The last solution to make a group communication is based on Attribute-Based Encryption (ABE) by Huang *et al* [101]; they proposed a Situation-Aware Trust (SAT) Architecture for vehicular networks containing three components, one of which was an attribute based policy control model for VANETs to address a number of trust situations and application scenarios on-road. In their work, they assumed that each vehicle has a set of attributes which can be classified as dynamic and static attributes, depending on whether the attributes change frequently or stay the same during the time period. Vehicles that satisfy a set of descriptive attributes form a group which is called a policy group. For example, a policy group can be a group of vehicles which have the same attributes, common interests, security or service requirements, or environmental restriction (for example street name, time, driving direction, etc). The idea of policy group is that it is organised automatically without depending on a trust party to manage the group.

Some researchers proposed a number of solutions to improve authentication, privacy, non repudiation [102, 103, 104, 105]; but here, too ensuing the data confidentiality requirement in VANET however has been given relatively less attention than the other requirements. Other work has been proposed in protecting the transmission among nodes from attacks [6, 7, 106], however, how these nodes deal with this confidential information after being received is mostly neglected. Some other researchers proposed detection algorithms [107, 108] to discover the misbehaving nodes but still a prevention framework is more important to prevent the leakage of that information than detecting it after being

disclosed.

Existing approaches to security of MANETs and VANETs include traditional cryptographic solutions using public key certificates [109, 110] to maintain trust, in which a Trusted Third Party (TTP) or Certificate Authority (CA) certifies the identity associated with a public key of each communicated entity, Almomani and Zedan [111] proposed a comprehensive, top-down, end-to-end security solution for MANET based upon a well defined architecture and exploiting two of the ITU-T recommendations: X.800, and X.805. Such approaches can therefore, provide end-to-end secure communication channels. These approaches mainly focused on message confidentiality, integrity and non-repudiation, they do not consider however controlling the message dissemination of the communicated entities, and how these certified entities act is left to the application layer [9]. Therefore, Al-Bayatti *et al* [112] proposed behaviour detection algorithm combined with threshold cryptography digital certificates to satisfy prevention and detection to securely manage Mobile Ad hoc Network of Networks (MANoNs), whereas Zhou and Haas [87] studied the security threats, vulnerabilities and challenges which faces the *ad hoc* network. In their work [87] they protected the packets sent between nodes by choosing the secure routing path to the destination node based on the redundancies routes between nodes to maintain the availability requirement. This is because all key-based cryptographic approaches such as digital signature need a proper and secure key management scheme to bind between the public and private keys to the nodes in the network; Zhou and Haas subsequently used replication and new cryptographic technique (threshold cryptography) [113, 114] to build a secure key management process to achieve the trust between a set of servers in ad hoc networks by distributing trust among aggregation of nodes to certify nodes are trustworthy.

Securing the routing in MANETs has also been given much attention by the researchers; many approaches, therefore, have been proposed to deal with external attack. Sirios and Kent [115] proposed an approach to protect the packet sent to multi receivers by

using keyed one-way hash function supported by windowed sequence number to ensure data integrity.

Controlling data dissemination in communication systems such as VANETs is difficult to achieve. The framework that we propose has some relation with secure routing protocols, as it determines the sharing of information. We consider, however, policies operating at a higher level in the protocol stack (to be described in Chapter 4) where application specific trust decisions can be made. Public Key Infrastructure (PKI) and cryptography are achieving a kind of a quasi-trust before communication is started. How the nodes act after that, however, is a controversial issue as untrusted nodes cannot be predicted without establishing tracing techniques to ensure that they are not misbehaving whilst participating in the VANET.

In an analogous context of commercial and medical environments, individuals also demand that their personal information such as their names, addresses, phone numbers, national insurance numbers, credit card details, passwords, or date of birth (DOB) are transmitted confidentially. In particular they need assurance that these sensitive data have been securely communicated to the appropriate persons or organisations and to no others. Therefore, Pearson and Mont [14] employed a clever idea of sticking policies with data to control how the personal information should be processed, handled, shared with other specified parties.

As we see from Figure 3.1 confidentiality requirements can be solved by using encryption and routing control mechanisms. Traditional encryption tools being commonly used in security systems however they solve one part of the problem by encrypting data exchanged between nodes using the public key of the destination node and then decrypting the packet by the destination's private key but how the destination behave after is neglected (described in Section 3.5). Using a mechanism which employs access control to ensure confidentiality is a real possibility, which we use in particular Discretionary Access Control (DAC) to control data dissemination, thus ensuring data confidentiality and

privacy of the originator node in VANETs.

3.7 Summary

To gain more understanding of the problem domain and requirements, this chapter highlighted the network security concepts: security requirements, security attacks and security mechanisms, it also presented an overview of the cryptography background, and presented the related work in privacy and confidentiality issues in both VANETs and MANETs. Finally, this chapter presented some of the previous work (State of the Art) on securing VANETs and MANETs to which we relate our proposed policy-based framework and algorithm charts in Chapter 4 and Chapter 7 respectively.

Chapter 4

Framework

Objectives:

- Provide general overview of the proposed framework.
- Describe the framework.
- Show how the framework components interact.

4.1 Introduction

The problem of controlling data dissemination in VANETs (described in Section 2.4.4.3) is challenging, especially when the information is intended to be kept secret and is central to this thesis, as enunciated in the research question in Section 1.3 "how can we prevent secret information from being leaked to undesirable entity(ies)?". Information disclosed to anyone else other than the intended nodes is likely to allow a breach of privacy and confidentiality, especially important in hostile environments where keeping the message confidential from an enemy is essential.

This chapter therefore presents a novel policy-based framework to control the dissemination of data communicated between nodes in VANETs by attaching originator policies to messages as they are sent (published in [15, 16, 17]). Our framework differs from previous approaches (described in Section 3.6) since it takes into consideration the originator confidentiality requirements which is attached as a set of policy rules along with messages to ensure message confidentiality is maintained not only during transmission to the intended node(s), but to keep the message contents private to an originator-defined subset of nodes in the VANET, thus preventing the destination node from forwarding the message to unwanted recipients.

The remainder of this chapter is structured as follows. Section 4.2 gives a motivating example drawn from the military domain, where the impact of this form of confidentiality breach is self-evidently crucial. Section 4.3 provides a brief description of the security requirements specification for the originator and describes the data dissemination policy. Section 4.4 gives a general overview of the proposed framework. Finally Section 4.6 describes how its components interact between each other.

4.2 Motivating Example

Protecting a message sent in wireless networks such as in VANETs is difficult and crucially important, for example, in military contexts where member armies of an alliance want to share tactical mission information exclusively between themselves but not with other coalition members.

Taking an contemporary campaign as an example, consider three vehicles A,B,C in Figure 4.1, where vehicles A and B respectively belong to Country 1 and Country 2 armies, while C belongs to the Country 3 army. The Second Lieutenant in Vehicle A receives a command by radio from a commanding aircraft determining the target and the time of a mission.

Now, vehicle A wants to send a tactical message for the mission that says "we are going to start the mission at 8:30 am" to vehicle B; vehicle A, however, does not want vehicle B to send the message to vehicle C, because the latter is not trusted by A, and any breach in security may jeopardise the mission. So the general question becomes how can vehicle A ensure that vehicle B does not send the message to vehicle C?

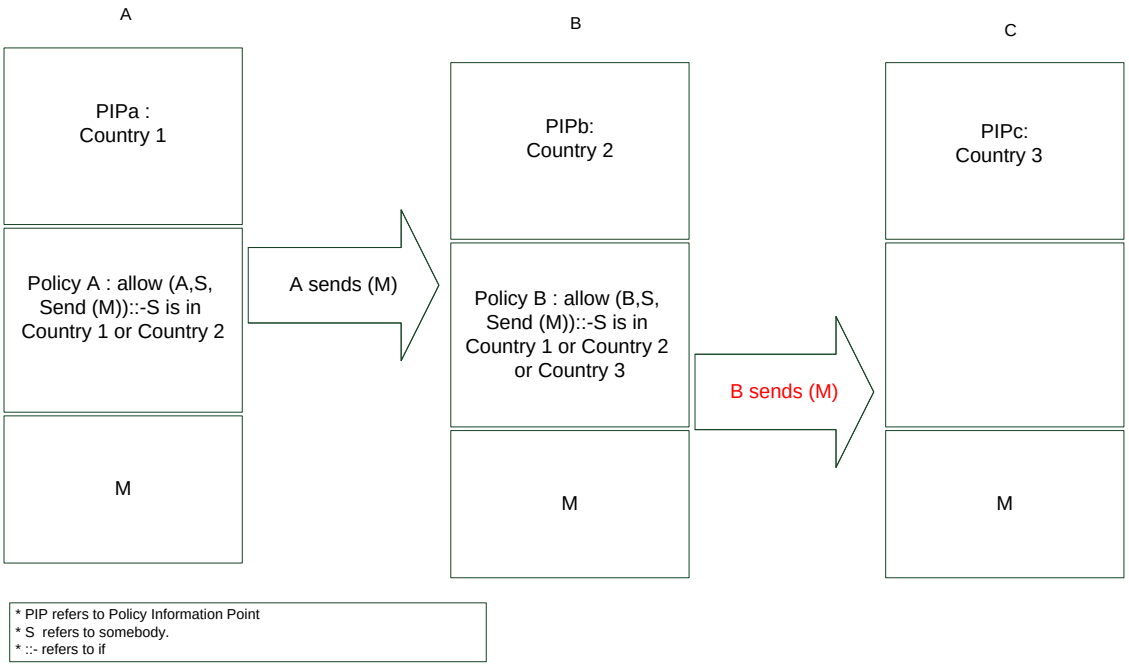


Figure 4.1: Vehicle B disclose the message to C

Vehicle A sends the message (M) to vehicle B; vehicle B now ‘knows’ the message (M)(as shown Figure 4.1). However, depending on its policy, vehicle B could send the message (M) and disclose it to vehicle C, thus breaching confidentiality of the message.

The framework addresses this problem by empowering (allowing) the originator of a message to specify the security requirements to be automatically applied and enforced on all the communicated entities in the network. This is done by attaching the policy of the originator (A) to the message (M) to control access to it, by defining who is allowed to access the message. In this way the policy of vehicle A attached to the message (M), tells vehicle B to which vehicles can the message (M) be sent (i.e. only Country 1 or Country

2 army units can receive the message) as described in the process 2 in Figure 4.2.

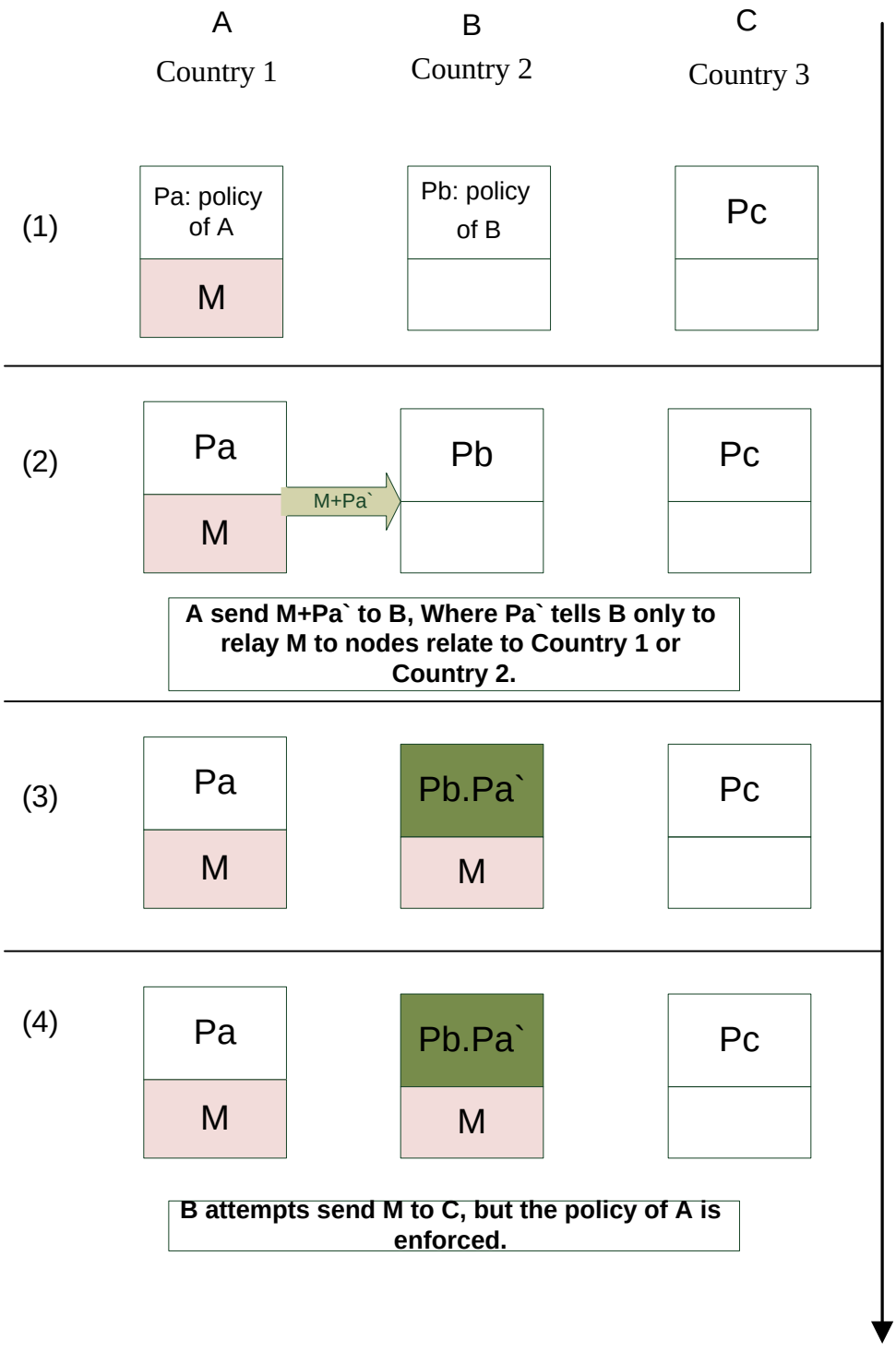


Figure 4.2: Prevention of disclosing the message (M) to vehicle C

Vehicle A sends the message (M) with the policy of A attached to it. The policy in-

structs vehicle B to send the message (M) to any vehicle if it relates to Country 1 or Country 2. The sender node assembles the packet depending on the protocol agreed among nodes in the network, and it normally contains: source address, destination address, message, the length of the message (size), and flags. In our work the packet contains a specific slot for the policy and some other slots to suit our NS-2 agent (The agent and the packet structure are to be respectively described in Sections 6.3 and 6.4). When vehicle B receives the packet, the message and the policy will both be extracted (as shown in the process 3 in Figure 4.2). Vehicle B now 'knows' the message (M) in addition it 'knows' the policy of A (inbound policy), as depicted in Listing 4.1 and its own policy (policy of B) as depicted in Listing 4.2.

Listing 4.1: Inbound policy at node B

```
Policy A : allow (A,S, Send (M)) if S relates to Country 1 or
    Country 2 where S is refer to somebody.
```

Listing 4.2: policy of B

```
Policy B : allow (B,S, Send (M)) if S relates to Country 1 or
    Country 2 or Country 3 where S is refer to somebody.
```

After that if vehicle B tries to reveal the message (M) (just received from vehicle A) to vehicle C (as shown in the process 4 in Figure 4.2), vehicle B will check its outbound policy depicted in Listing 4.3. Because the policy of A (the originator) is more dominant (important) then it is enforced, thus the message (M) is prevented from being disclosed to vehicle C.

Listing 4.3: Outbound policy at node B

```
allow (B,S, Send (M)) if S relates to Country 1 or Country 2 where S
    is refer to somebody.
```

4.3 Security Requirements Specification

Originators of messages have some concerns about their privacy and confidentiality requirements to retain the data secret from some nodes in the network. In this work the originators can specify the required rules that a node (s) must possess with regarding to sensitive message disseminated in VANET . In our motivating example as in Figure 4.2 (described in Section 4.2), this is the requirement that node A tells node B to send the message (M) to Country 1 or Country 2 nodes; the secret message (M) must not be disclosed to any other nodes such as those of Country 3 army members. The originators should provide the message (M), type of the message (top secret, secret, unclassified) and the destination which will be formally expressed in the data dissemination policy, to enable our policy-based framework to control the message flow between nodes in VANETs.

The data dissemination policy (to be described in Chapter 5) specifies the security requirements of disseminating data in VANETs which determined by the originator, as a set of rules that control how messages can be securely disseminated to other destination(s) without be disclosed to unwanted node(s) in the network. The data dissemination policy is designed to protect the message confidentiality as expressed by the originator as a set of policy rules which they be able to be understood by the framework.

4.4 Proposed Policy-based Framework

In this thesis we provided a policy-based framework that addresses this problem (described previously in Section 4.2) by automatically attaching policies to the messages that identify how the information can be used by the receiver, thus limiting the relay of messages based on the originator's confidentiality requirements.

Figure 4.3 presents the proposed framework, where policies are used to enforce access control to such information sent by the originator to other entities in the system.

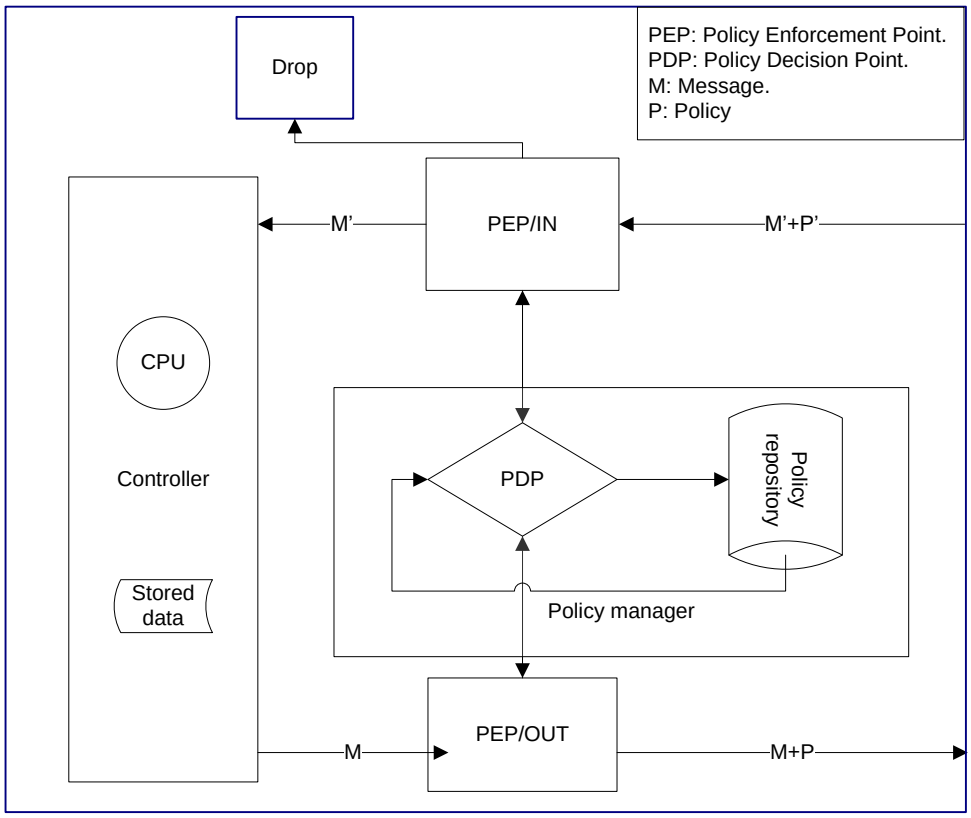


Figure 4.3: The proposed framework

Figure 4.3 shows the proposed framework, where the originator attaches policies that express the information system's security requirements at a high level of abstraction.

The overall framework composed of four components as they are shown in Figure 4.3:

1. policy enforcement point (PEP/OUT): This component executes and enforces policy decisions in the sender node, it is installed at the transmitter interface that does merge system's policy with the message sent to others nodes.

2. policy enforcement point (PEP/IN): This component executes and enforces policy decisions in the receiver node, this component installed at the receiving interface that does splitting the message from the policy attached.

3. policy manager: This component is composed of three sub components:

   - The policy decision point (PDP), which handles the requests come from PEP/OUT

in the sending process to determine whether the message (M) is allowed to be send to a specific destination(s) or not, this request processed by looking into its policy repository.

- The policy repository, where the policy rules and condition can be stored. A policy is a set of rules that express how information contained in the message (M) can be disseminated to other destination(s). A key part of a policy specification is that it associates rules with nodes' identities. For example the following policy rules could be present in the policy of vehicle A by assuming that there are three types of messages (Top secret, Secret, Unclassified) to be sent at vehicle A:

  + Top secret $\rightarrow$ Country 1
  - Top secret $\rightarrow$ Country 2
  - Top secret $\rightarrow$ Country 3
  + Secret $\rightarrow$ Country 1
  + Secret $\rightarrow$ Country 2
  - Secret $\rightarrow$ Country 3
  + Unclassified $\rightarrow$ Country 1
  + Unclassified $\rightarrow$ Country 2
  + Unclassified $\rightarrow$ Country 3

  This means that the policy of vehicle A allows the Top secret messages to be disseminated only to Country 1 vehicles, whereas Secret messages can be disseminated to both Country 1 and Country 2 vehicles, Finally Unclassified messages can be disseminated to all vehicles.

  Hence, a positive data dissemination (+) represents a send permission and a negative data dissemination (-) represents a denial send permission. The data dissemination policy consists of a list of policy rules which specifies the restrictions on the possible paths of the data dissemination (to be described in

Section 5.3).

- The policy conflict detection and resolution point, which handles and solves the policy rule 'clashes' which might happen at some points when the receiver's policy 'clashes' with the originator's policy; hence the receiver has its own policy and the policy just received from the originator, therefore this node needs to take a proper action to resolve this conflict (to be discussed in detail in Section 5.5).

  In Figure 4.2 for example, assume at Time 1 that node A sent a Secret message (M) to node B along with its policy (disallow sending a Secret type of message from source to destination if the destination is in the Country 3 group).

  So the policy rules are:

  + Secret $\rightarrow$ Country 1

  + Secret $\rightarrow$ Country 2

  - Secret $\rightarrow$ Country 3

  At Time 2: The node B received the message. But node B has this policy rule (allow sending a Secret type of message from source to destination if the destination is in the Country 3 group)

  + Secret $\rightarrow$ Country 1

  + Secret $\rightarrow$ Country 2

  + Secret $\rightarrow$ Country 3

  Therefore whenever Node B wants to send the same message to any node inside the coalition members, there will be a conflict, between 'disallow' and 'allow' sending the message of type Secret to a node in Country 3 group; we therefore implemented our work to make the originator's policy to be dominant, which as a result, disallows the flow of the message (see Section 5.5 for different cases of policy conflict rules).

4. The controller that processes and stores the information received from the other components, this component composed of two parts:

    - central processing unit (CPU).
    - data store where the message can be saved.

The framework as depicted in Figure 4.4 intercepts all messages arriving for the application layer of the node. Each incoming packet is expected to carry a policy together with the application layer message. The Ingress Policy Enforcement Point (PEP/IN) splits the message (M') from its policy (P') and queries the Policy Decision Point (PDP) whether the message should be passed to the Application Layer for processing or not. This functionality is similar to a traditional firewall; if the PDP decides that the message should not be processed it returns a 'Deny' to the Ingress PEP/IN; if the message can be processed according to the policy, it will respond 'Permit' to the ingress PEP/IN. Depending on the policy-model used the PDP may retain or alter state information, such as attributes that can have an effect on future policy decisions. In this thesis the only state retained by the PDP is the policy, which change based on the received messages and their dissemination policies. The PEP/IN passes the inbound policy to the PDP. The PDP merges the policy with its own policy and assigns a unique label l_m to the ingress message M'. This label is then used by the framework to trace the flow of data contained in the message M' using the label l_m.

When this node wants to send a message (M), the Egress PEP/OUT handles the message and queries the PDP whether the message can be sent to the intended recipient or not. The policies that are being checked by the PDP are traditional access policies present on the node, as well as all polices (inbound policy) that match the label l_m of the received message (M). If the PDP responds with 'PERMIT' the message (M) is sent with the outbound policy (P).

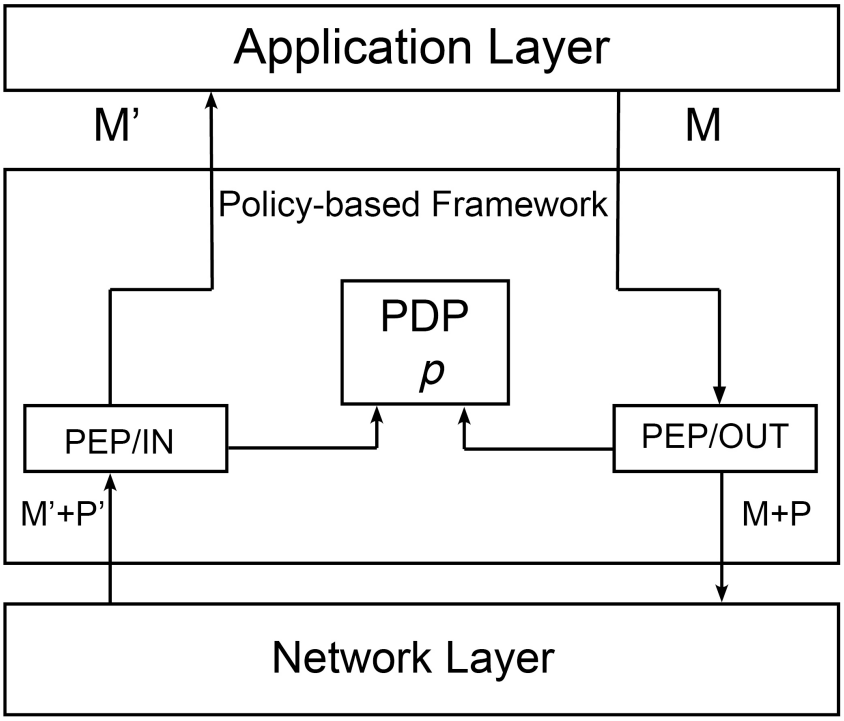


Figure 4.4: Conceptual model where our policy-based framework added between the Network Layer and the Application Layer

4.5 Assumptions

We assumed that our framework will be implemented in every entity in the communicated systems so that all nodes in the system behave homogeneously. To provide a complete security package, the proposed framework should be implemented in conjunction with other security services based on Public Key Infrastructure (PKI), for instance, integrated with the approach proposed in CARAVAN scheme to ensure privacy [11], or other approaches to provide integrity and confidentiality [96], or to provide authentication, authorization, and non-repudiation [7]. Also the proposed framework should be implemented on a classical certificate-based system such as the X.509 standard where CA (Certificate Authority) is used for message authentication [116].

In this work, nodes in the system are organised into different groups. The group-

id (GID) is defined and distributed by the CA (Certificate Authority). Since the X.509 standard is the most commonly used in trust systems, this policy-based framework is therefore built on using X.509 certificates which contain attributes such as the name of the issuing authority, the node identity, the public key of the and a validity period (as described in Section 3.5.3).

We currently assume that all nodes in our system are trusted to correctly enforce the policies that are attached to the message and provide a communication system that includes a policy processing layer dealing with inbound and outbound policies. Our proposed framework will be applied at the upper layer as shown in Figures 4.4.

4.6 Computational Model

The main contribution of this thesis is linking between policy- based control and messages dissemination in VANETs at the high level as depicted in Figure 4.4, we therefore use a simple policy language (to be described in Chapter 5).

To illustrate how these components communicate, Figures 4.5 and 4.6 present the proposed computational model and the sequence diagram respectively, which show how the message (as in Figure 4.2 described in the motivating example in Section 4.2) will be sent and received between the layers and the policy-based framework in the system. Our framework will be applied in the upper layers as seen in Figure 4.5. The following steps show how the packet can be processed by each component in the framework starting from the originator side until it is received on the recipient side:

1. Originator Side (A): The application layer processes a request to PEP/OUT querying to send the message (M) to node B (as shown in 4.6 sending part and in the process 1 in Figure 4.5). The PEP/OUT send a request to the PDP of node A to check its policy repository where the policy rules of A are stored (as shown in the process 2 in Figure 4.5), to check whether the message (M) can be send to the

destination or not. The PDP starts by looking up to find these policy rules which matched to the message (M), and retrieves them (as shown in the process 3 and 4 in Figure 4.5). Then the PDP replies to the PEP/OUT of node A by either giving permission to send 'Permit' or disallowing to send 'Deny' (as shown in the process 5 in Figure 4.5). In this case, as in Figure 4.2, the policy rules for the Secret type of message in vehicle A are as follows:

+ Secret $\rightarrow$ Country 1

+ Secret $\rightarrow$ Country 2

- Secret $\rightarrow$ Country 3

Based on these policy rules the PDP of node A sends a 'Permit' result in this case to the PEP/OUT to send the message (M). Then PEP/OUT of node A retrieves the message (M) from the data store in the controller component (as shown in the process 6 in Figure 4.5), and then sends the message (M) + Policy of A (P) to the adjacent node which is B.

2. Receiver Side (B): When node B receives the Packet [M'+ Policy of A (P')] through PEP/IN of B (as shown in 4.6 receiving part and in the process 8 in Figure 4.5). The PEP/IN splits the message (M') from its policy (P'), then it sends a request to the PDP of node B to check the inbound policy to decide whether the message should be accepted and passed to the Application Layer or not (as shown in the process 9, 10 and 11 in Figure 4.5). The PDP replies to the PEP/IN of node B by either giving permission to receive 'Permit' or reject to receive 'Deny' (as shown in the process 12 in Figure 4.5). In case of 'Permit' as in Figure 4.2 the PEP/IN passes the message (M) to the Application Layer (as shown in the process 13 in Figure 4.5), and the PEP/IN passes the inbound policy to the PDP. The PDP merges the policy with its own policy which leads to update the outbound policy of node B that matches the message (M). The policy rules for the Secret type of message in vehicle

A, as in Figure 4.2 are as follows:

+ Secret $\rightarrow$ Country 1

+ Secret $\rightarrow$ Country 2

+ Secret $\rightarrow$ Country 3

The inbound policy rules are as follows:

+ Secret $\rightarrow$ Country 1

+ Secret $\rightarrow$ Country 2

- Secret $\rightarrow$ Country 3

When node B processes a request to send the message (M) to node C, the PDP of node B process the request by looking up into its policy repository (outbound policy) where the policy rules of node B and the inbound policy of node A (P') are stored in node B, thereafter the PDP of node B needs to make a decision whether it is allowed to send the message (M) to node C or not, because there is a policy conflict between the policy rules of both node A and node B as shown above.

Therefore, whenever node B wants to send the message (M) to any node inside Country 3 members, there will be a conflict between 'Deny' and 'Permit' sending the message of type secret to a node in Country 3 group as the policy rule shown above, therefore we implemented our work to make the originator's policy to be dominant in order to disallow the flow of the message (see Section 5.5 for different cases of policy confliction rules). In this case, however, the PDP of node B decides to send a 'Deny' result to PEP/OUT of node B to not send the message (M), because the policy rule of the outbound policy of node B as shown below is not complied with (adjacent node C is not a member of the allowed group of the originator node A).

As a result the outbound policy rules of node B are as follows:

+ Secret $\rightarrow$ Country 1

+ Secret $\rightarrow$ Country 2

- Secret $\rightarrow$ Country 3

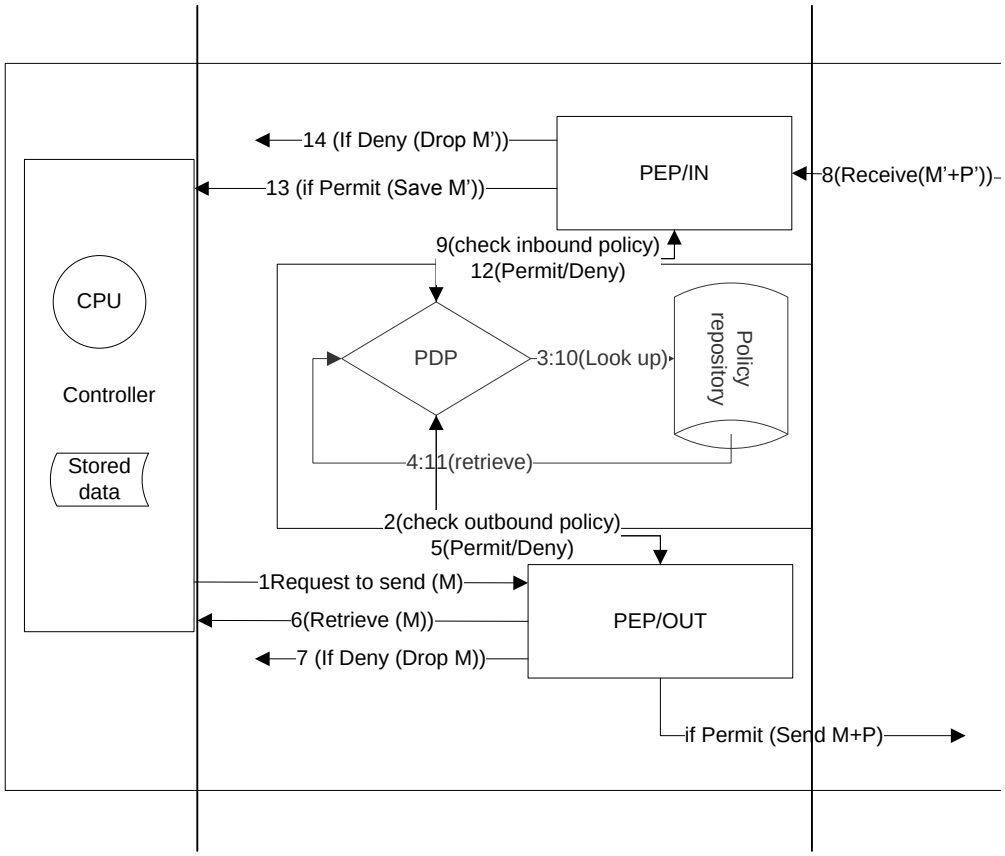


Figure 4.5: Computational model for our proposed framework

4.7 Summary

This chapter presented a novel policy-based framework to control the dissemination of data communicated between nodes in VANETs by attaching originator policies to messages as they are sent. The policy-based framework addresses the data dissemination problem in VANETs by automatically attaching policy rules to the messages that identify

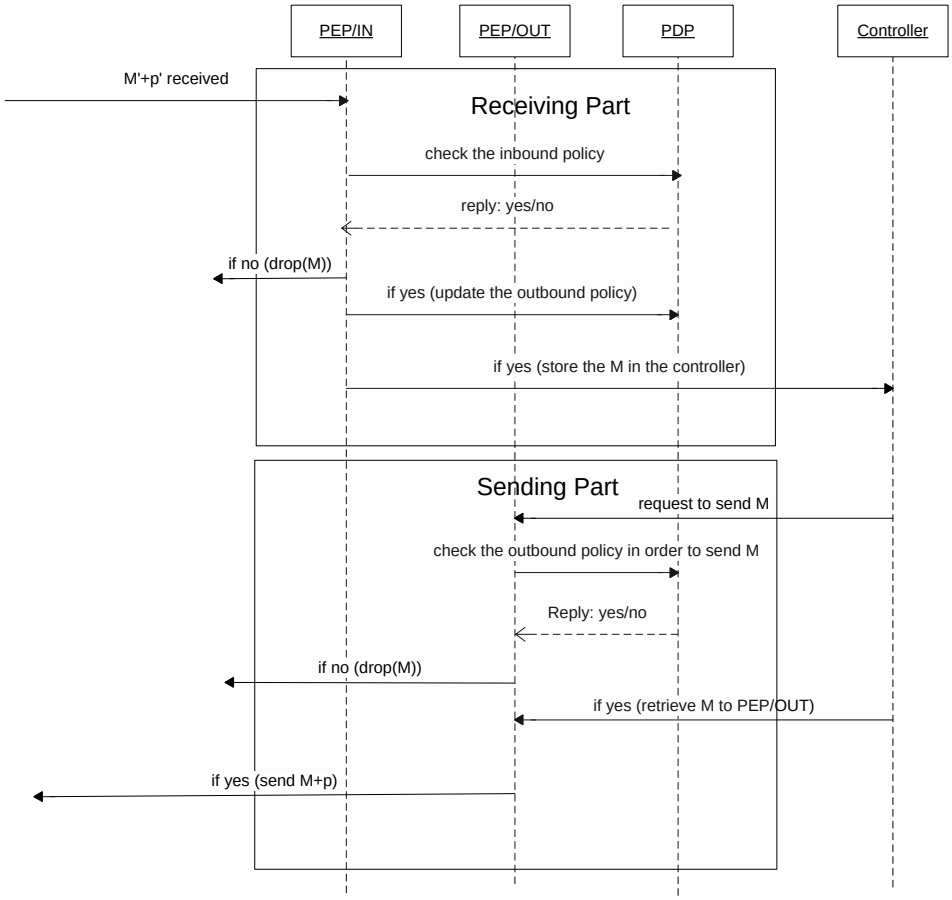


Figure 4.6: Sequence diagram

how the information can be used by the receiver, to keep the message contents private to an originator-defined subset of nodes in the VANET, thus preventing the destination node from forwarding the message to unwanted recipients based on the originators confidentiality requirements. This chapter also described how the policy-based framework components interact between each other, it also presented a brief description of the security requirements specification and data dissemination policy rules (to be described in Chapter 5).

By presenting the novel policy-based framework to control the dissemination of data communicated between nodes in VANETs by attaching originator policies to messages as they are sent, this chapter therefore addressed the main research question as articulated in Section 1.3 "how can we prevent information from being leaked to undesirable entity(ies)?" and the other sub research questions "how to keep message contents private to an originator-defined subset of nodes in the VANET" and "how to control the dissemination of messages while the nodes are communicating between each other in the network".

Our assumption that the framework is implemented in all nodes within the system limits the applicability of the approach in that it assumes some level of cooperation between the nodes to which the data can be disseminated. The presented framework will not prevent the wrongful dissemination of data by malicious nodes that have been trusted by the originator. It prevents, however, cooperating nodes from inadvertently publishing information that the originator would not want to release.

Chapter 5

Data Dissemination Policy and the Originator Interaction

Objectives:

- Define data dissemination requirements.
- Define data policy rule.
- Define data dissemination policy language.
- Define data dissemination policy conflict.
- Define the originator interaction.

5.1 Introduction

In Section 4.4 a policy-based framework was described that addresses the problem of secure data dissemination in VANETs by automatically attaching policies along with messages to specify how the information can be used by the receiver, so as to prevent

disclosure of the messages other than consistent with the requirements of the originator. Section 5.2 now describes these requirements as a set of policy rules (to be described in Section 5.3) that explicitly instructs recipients how the information contained in messages can be disseminated to other nodes.

This chapter is considered as one of our contributions which links between the framework and data dissemination policy in VANETs at a high level in order to address the research question (described in Section 1.3). Section 5.4 describes the data dissemination policy language used in this work; it also describes the policy rules modified from previously published work [17, 19] in order to be a suitable and understandable language for the framework to ensure the originator confidentiality requirement. Finally Section 5.5 describes the data dissemination policy conflict rules and when the originator should be asked for its up-to-date policy.

Our data dissemination policy differs from the current policy languages (to be described in Section 5.4) since it takes into consideration the originator high-level requirements as low-level policy rules whose enforcement can be fully automated and understood for the framework (described in Section 4.4) in order to solve the research question.

5.2 Data Dissemination Requirements

Originators of messages in VANETs require that their privacy and confidentiality requirements are maintained not only during transmission to the intended node(s), but to keep the message contents private to an originator-defined subset of nodes, thus preventing the destination node from forwarding the message to unwanted recipients. In this work the originators specify the policy rules that a node(s) must possess with regard to their sensitive message dissemination. Data dissemination requirements are type of messages that restrict the action to allow or disallow message flowing to specific destination(s). In our motivating example as in Figure 4.2 (described in Section 4.2), the originator (A) of mes-

sage (M) provide the specification of the desired behaviour that a node (B) must possess with respect to a particular message flow; this is the requirement that node A tells node B to send the message (M) to Country 1 or Country 2 nodes only.

Data dissemination requirements are the originators' interest how the data can be disseminated to a specific node(s). When the originators send messages they should provide the message (M), type of the message (top secret, secret, unclassified) and the destination (these requirements to be explained in Section 5.4.2). In order to enable our policy-based framework to control the message flow to destinations, in this work we organised the nodes into different groups (for example group-id1, group-id2, group-id3). The policy rules are therefore intended to identify the different type of messages that can hold sensitive information and to identify the destinations to which a particular message can be forwarded (to determine whether the destination is allowed to receive the sensitive information or not), depending on the group-id that the destinations relate to.

5.2.1 Example of Data Dissemination Requirements

In Figure 5.1 we show an example of six nodes, assuming that each node in the system has a group-id number, thus classifying the nodes in our work into different groups, in this case: group-id 1, group-id 2, and group-id 3 (the group id is defined and distributed by the CA (Certificate Authority) based on X.509 standard). The first group contains node 0, node 2, node 4 and node 5, whereas node 1 and node 3 are respectively in group-id 2 and group-id 3.

As depicted in Figure 5.1 assume that the originator (node 0) requires Top secret messages to be disseminated only to nodes in group-id 1, whereas Secret messages can be disseminated to both group-id 1 and group-id 2 nodes, Finally Unclassified messages can be disseminated to all nodes in group-id 1, group-id 2, and group-id 3.

1. Assume at Time 1 that node 0 'wants' to send a Top secret message (M1) to node

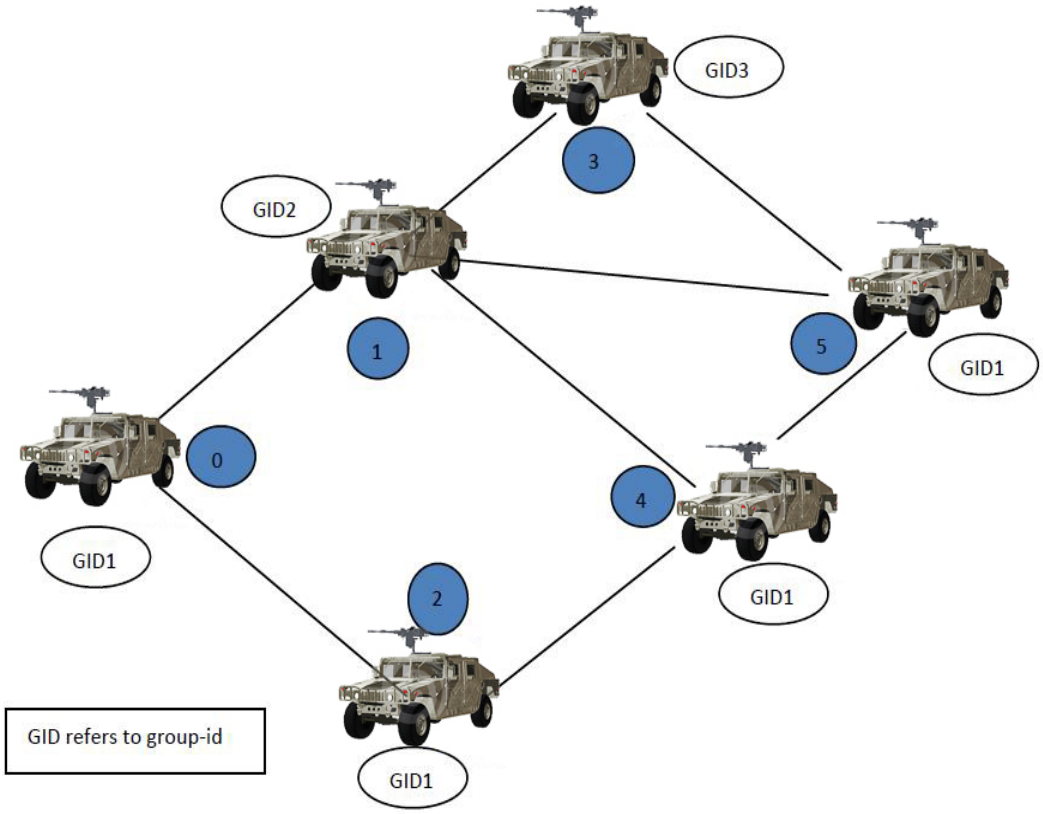


Figure 5.1: Example to illustrate organising nodes into groups

2, along with its policy rules specified by the originator's (node 0) message requirements:

- allow sending a Top secret type of message from source to destination if the destination is in the group-id 1.
- disallow sending a Top secret type of message from source to destination if the destination is in the group-id 2 or the group-id 3.

2. Assume at Time 2 that node 0 'wants' to send a Secret message (M2) to node 1 along with its policy rules specified by the originator's (node 0) message requirements:

   - allow sending a Secret type of message from source to destination if the des-

tination is in the group-id 1 or group-id 2.

- disallow sending a Secret type of message from source to destination if the destination is in the group-id 3.

3. Assume at Time 3 that node 0 also 'wants' to send a Unclassified message (M3) to node 1 along with its policy rules specified by the originator's (node 0) message requirements:

   - allow sending a Unclassified type of message from source to destination if the destination is in the group-id 1 or group-id 2 or group-id 3.

So how can the originator (node 0) specify these data dissemination requirements as a set of rules that control how messages can be securely disseminated to other destination(s) without being disclosed to unwanted node(s) in the network?. Therefore, the data dissemination policy rule format should be designed to protect the message confidentiality as expressed by the originator as a set of policy rules which they be able to be understood by the framework without any ambiguity. The question above is to be answered in Section 5.4.2 as we describe the translation of the originator high-level requirements into low-level concrete policy rules.

5.3 Data Dissemination Policy

In information technology [117] a policy means "a predetermined action pattern that is repeated by an entity whenever certain system conditions appear". The IETF (Internet Engineering Task Force) defines a policy as a explicit goal, course, or method of action to guide and to choose present and future decisions [118]. Finally and most relevant to this work the term policy can be defined as a set of rules to allow the originators to administer, manage, and control access to messages. The data dissemination policy is

group of regulations, rules, written by the originators of the messages to control how their messages can be disseminated and used by other recipients.

The data dissemination policy reflects the originator data dissemination requirements which concentrate on the type of messages that the originator 'wants' to send, and the destinations to which the message can flow, to specify how the data can be disseminated between the different nodes.

This section describes how to specify the data dissemination policy which should reflects the originators' requirements of the message in a precise manner without ambiguity.

Data dissemination policy defines which messages are allowed or disallowed to be sent to a specific node(s), and in which cases that attempt to send a message to specific destination(s) the originator should be asked. The data dissemination policy rule consists of the following three components:

- Action **A**.
- Type of Message **T**.
- Destination **D**.

$$\text{A T} \quad \rightarrow \quad \text{D}$$

Possible actions are (+) for allowing to send the message, (-) for disallowing to send the message, and (?) for asking the originator for its up-to-date policy to allow or disallow data dissemination.

- A positive send form is represented by + T $\rightarrow$ D (+ symbolize an action, T as type of message and D as the destination), the (+) symbol is used to allow sending the message from source to the destination.
- A negative send form is represented by - T $\rightarrow$ D. Similar to the positive syntax, but is used to deny sending a specific type of message from source to destination.

Therefore the (-) symbol is used to prevent the leaking of the message from source to destination.

- The originator decision form is represented by ? T $\rightarrow$ D. It is similar to the previous two syntaxes, however, the (?) symbol is used in the case of the originator need to be asked for its up-to-date policy to decide whether a particular message can be sent from source to destination or not.

Hence, a positive data dissemination represents a send permission and a negative data dissemination represents a denial send permission. The data dissemination policy consists of a list of policy rules which specifies the restrictions on the possible paths of the data dissemination, and when the originator of message should be asked in case of updating the policy.

Our policy language is similar to the ACL (Access Control List) used in systems files in most of the Unix and Unix-like operating systems (for example, Linux, Solaris). ACL is a list of permissions attached to the object (in our case the message) this ACL should specify which subjects (nodes) are allowed to access/receive the message, in addition it also instructs the receivers to which nodes they are allowed to forward the object (message).

For instance, if a file has an ACL that contains (Bob, edit), this would give Bob permission to edit the file.

Whereas in our case, if the originator node attaches this data dissemination policy to a message:

+ Secret $\rightarrow$ GID 1

This would give nodes permission to send/receive the object (message) to and from the nodes in group-id 1.

The data dissemination policy should specify high-level requirements into low-level

policy rules whose enforcement can be fully automated and understood for the framework.

In this work we provide a suitable data dissemination policy to be used for the framework in which the originator of the message retains control over its dissemination provided that receivers of the message are trusted to enforce the policy.

5.4 Data Dissemination Policy Language

The data dissemination policy specifies the data dissemination security actions to be considered in the network to keep the message secure. It is represented as a set of policy rules that declares the data dissemination requirement based on the originator of the message. The data dissemination policy works as the reference that controls the flow of the messages while the nodes are communicating between each other in the network.

There are many policy languages exist such as Authorisation Specification Language (ASL), XACML (eXtensible Access Control Markup Language), and Ponder. Unfortunately, none of these policy languages ensure the message confidentiality requirement while messages flow between nodes in *ad hoc* networks. These policy languages cannot enforce any control on the information flow once this information has been received by a node. Hence, these policy languages concentrate on controlling the access at specific resources located on central or distributed nodes, they are not intended however to control the data dissemination between nodes.

- ASL (Authorisation Specification Language): This language developed by Woo and Lam [119] investigated logic-based languages for the specification of security policies. Their language requires a strong mathematical background, which makes it complex to use and implement.

- XACML (eXtensible Access Control Markup Language): This is an XML-based language for access control developed by a project of Sun Microsystems, then standardised by OASIS (Organization for the Advancement of Structured Information

Standards)[120]. The language supports role based access control, in addition to that, it is based on XML representation; which means the policy is not really aimed at human interpretation, there is no formal semantics for the language itself, which makes policy analysis difficult and hard to understand.

- Ponder: it is a declarative, object-oriented language developed by the policy group at Imperial College for specifying security policies in distributed systems. Ponder can be used for firewalls, operating systems, and databases. It supports both Authentication (deals with verification of the identity of nodes) and Access control requirements, it does not support, however, the message confidentiality requirement and it also been recently withdrawn [121].

Our data dissemination policy language has various advantages over other policy languages to control and manage the access to resources. For example, the data dissemination policy can be deployed easily and without any ambiguity by various different nodes. Since it is aimed for a specific purpose, it focuses on access control for a particular resource which is the message, based on the originator requirements. It has a powerful combining logic capability which makes it ideal for network systems. As a result of implementing our policy rules with the framework to check whether the privacy and confidentiality of the originator are met in VANETs, these policy rules are attached to messages to specify how the information can be used by the recipients, it showed that these policy rules are expressive enough to ensure and implement the message originator requirements and to distribute enforcement policies in the network efficiently (to be described in Chapter 7). Finally, it is compatible with the Network Simulator (NS-2) which makes it to be understandable for the policy agent. Since it is considered as a low-level policy language which makes it perfectly run by C++ programming language supported in our NS-2 policy-based agent protocol (to be described in Section 6.2).

5.4.1 Syntax

The syntax of the data dissemination policy language is depicted in Listing 5.1. The policy definition is introduced by the key word policy and three identifiers as follows:

- <$ACTION$ >which can be either (+) to represent positive data dissemination, (-) to represent negative data dissemination, or the originator decision (?).
- <$STRING$ >which can be either "TOP SECRET" or "SECRET" or "UNCLASSIFIED" or "GID".
- <ID >is used for the destination group-id number.

Listing 5.1: The data dissemination policy syntax

```
Policy= (<ACTION> <STRING>  >>>  <STRING><ID> )
        <ACTION>=   "+" | "-" | "?"
        <STRING>= "SECRET","TOP SECRET","UNCLASSIFIED", "GID"
        < ID>= <LETTER>
        <LETTER>=  "0" _ "9"
```

5.4.2 Semantics of Data Dissemination Policy Rules

The semantics of the data dissemination policy declare the desired flow of the message in the network and how the destination can deal with the message afterwards depending on the originator policy. Data dissemination policy rules also determine when the the originator should be asked for its up-to-date policy about a particular message flow.

Allowed data dissemination rule example: Assume these parameters passed by the originator as in the list 5.2

Listing 5.2: Allowed data dissemination rule

```
Action A= +
Type of message T = Secret
Destination D = GID 1
```

Then the data dissemination policy rule is

+ Secret $\rightarrow$ GID 1

Hence, the information contained in the message classified as a secret message type is allowed to flow to the nodes inside the group-id 1 only. Whenever this rule exists in a node, it should be applied.

Disallowed data dissemination rule example: Assume these parameters passed by the originator as in the list 5.3

Listing 5.3: Disallowed data dissemination rule

```
Action A= -
Type of message T = Secret
Destination D = GID 2
```

Then the data dissemination policy rule is

- Secret $\rightarrow$ GID 2

Hence, the information contained in the message which classified as a secret message type is disallowed to flow to any node inside the group-id 2. Whenever this rule exists in a node, it should be applied.

The originator decision rule example: Assume these parameters passed by the orig-

inator as in the list 5.4

Listing 5.4: The originator interaction

```
Action A= ?
Type of message T = Top secret
Destination D = GID 3
```

Then the data dissemination policy rule is

? Top secret $\rightarrow$ GID 3

According to this rule the originator will be asked for its up-to-date policy, in order to allow or disallow the data dissemination from source to any node inside the group-id 3, since the information contained in the message which classified as a top secret message type requires the originator's up-to-date policy.

At the end of this section, according to the example of data dissemination requirements requested by the message originator (node 0) (as explained in Section 5.2.1) the following rules are therefore reflect these requirements as a set of policy rules at node 0:

+ Top secret $\rightarrow$ GID 1

- Top secret $\rightarrow$ GID 2

- Top secret $\rightarrow$ GID 3

+ Secret $\rightarrow$ GID 1

+ Secret $\rightarrow$ GID 2

- Secret $\rightarrow$ GID 3

+ Unclassified $\rightarrow$ GID 1

+ Unclassified $\rightarrow$ GID 2

+ Unclassified $\rightarrow$ GID 3

These data dissemination policy rules now avoid any ambiguity which can happen at

the receiver side and make the specification more precisely, its also understandable for the framework.

5.5 Data Dissemination Policy Conflict Rules

In this section the conflicts between the data dissemination policy rules have been addressed using the conflict keyword. This is in case of a conflict that may have occurred between 'allow' and 'disallow' in the flow for one type of message at the node as previously described in the motivating example in Chapter 5.2.1. The data dissemination policy conflict rules handle and solve the policy 'clashes' which might happen at some points when the receiver's policy 'clashes' with the originator's policy rule; hence the receiver has its own policy and the policy just received from the originator. A proper action, therefore, has to be done in this case where in this work we implemented the originator policy to be the highest priority over any other policy. Listing 5.5 represents the data dissemination policy with conflict syntax to show how the data dissemination policy specify which action has to be taken in the case of policy conflict.

Listing 5.5: The data dissemination policy with conflict syntax

```
Policy= (<CONFLICT>)* (<ACTION> <STRING>  >>>  <STRING><ID>)
                  <ACTION>="+" | "-" | "?"
                  <STRING>= "SECRET","TOP SECRET","UNCLASSIFIED","GID
                      "
                  < ID>= < LETTER>
                  <LETTER>=  "0" _ "9"
                  <CONFLICT>= "Conflict:"(("+-")  >>>  ("+" )
                                          ("-+ ") >>>  ("-" )
                                          ("+?")  >>>  ("+" )
                                          ("-?")  >>>  ("-" )
                                          ("?+")  >>>  ("?" )
                                          ("?-")  >>>  ("?" ))
```

We used <CONFLICT >syntax to deal with the data dissemination policy rules conflicts as follows. There are six cases where the conflict can occur at the node wherever the receiver's policy conflicts with the sender's policy of a specified message as follows:

- **Case 1**:

  At Time 1: Assume that sender A sends a message (M) to node B attached with these policy rules:

  + Top secret $\rightarrow$ GID 1

  + Top secret $\rightarrow$ GID 2

  + Top secret $\rightarrow$ GID 3

  At Time 2: The node B received the message (M). But node B has these policy rules:

  - Top secret $\rightarrow$ GID 1

  + Top secret $\rightarrow$ GID 2

  + Top secret $\rightarrow$ GID 3

  Therefore whenever Node B wants to send the message (M) to any node inside the group-id 1, there will be a conflict, the conflict is between allow and disallow of sending the message of type Top secret to a node in group-id 1, therefore we implemented our work to make the originator's policy to be the dominant (+- >+) which, as a result allows the flow of the message.

- **Case 2**:

  At Time 1: Assume that sender A sends a message (M) to node B attached with these policy rules:

- Secret $\rightarrow$ GID 1

+ Secret $\rightarrow$ GID 2

+ Secret $\rightarrow$ GID 3

At Time 2: The node B received the message (M). But node B has these policy rules:

+ Secret $\rightarrow$ GID 1

+ Secret $\rightarrow$ GID 2

+ Secret $\rightarrow$ GID 3

Therefore whenever Node B wants to send the message (M) to any node inside the group-id 1, there will be a conflict, the conflict is between 'disallow' and 'allow' sending the message of type Secret to a node in group-id 1, therefore we implemented our work to make the originator's policy to be the dominant (-+ >-) which, as a result disallows the flow of the message.

- **Case 3**:

  At Time 1: Assume that sender A sends a message (M) to node B attached with these policy rules:

  + Top secret $\rightarrow$ GID 1

  + Top secret $\rightarrow$ GID 2

  + Top secret $\rightarrow$ GID 3

  At Time 2: The node B received the message (M). But node B has these policy rules:

  + Top secret $\rightarrow$ GID 1

  ? Top secret $\rightarrow$ GID 2

  + Top secret $\rightarrow$ GID 3

Therefore whenever Node B wants to send the message (M) to any node inside the group-id 2, there will be a conflict, the conflict is between allow and ask the originator, of sending the message of type Top secret to a node in group-id 2, therefore we implemented our work to make the originator's policy to be the dominant (+? >+) which, as a result allows the flow of the message.

- **Case 4**:

  At Time 1: Assume that sender A sends a message (M) to node B attached with these policy rules:

  - Top secret $\rightarrow$ GID 1

  + Top secret $\rightarrow$ GID 2

  + Top secret $\rightarrow$ GID 3

  At Time 2: The node B received the message (M). But node B has these policy rules:

  ? Top secret $\rightarrow$ GID 1

  + Top secret $\rightarrow$ GID 2

  + Top secret $\rightarrow$ GID 3

  Therefore whenever Node B wants to send the message (M) to any node inside the group-id 1, there will be a conflict, the conflict is between ask the originator and 'disallow' of sending the message of type Top secret to a node in group-id 1, therefore we implemented our work to make the originator's policy to be the prominent (-? >-) which, as a result disallows the flow of the message.

5.5.1 The Originator Interaction

Since scalability is an important requirement in policy-based systems, this thesis presented both a prevention mechanism for VANETs using policy-based framework

to ensure that information is not disclosed to unwanted nodes, together with an interaction mechanism by referring back to the originator for updating the policy to meet the scalability requirement. A data dissemination policy language therefore should be developed to support the originator interaction.

Policy rules should be dynamic and can be altered in response to the originator requirement update, because what might be considered to be secure now could be considered later insecure depending on the originator requirements. Therefore, the following two cases show when the originator should be asked for its up-to-date policy rules to decide whether the data can be disseminated or not during the communication.

- **Case 5**:

  At Time 1: Assume that sender A sends a message (M) to node B attached with these policy rules:

  + Unclassified $\rightarrow$ GID 1

  + Unclassified $\rightarrow$ GID 2

  ? Unclassified $\rightarrow$ GID 3

  At Time 2: The node B received the message (M). But node B has these policy rules:

  + Unclassified $\rightarrow$ GID 1

  + Unclassified $\rightarrow$ GID 2

  + Unclassified $\rightarrow$ GID 3

  Therefore whenever Node B wants to send the message (M) to any node inside the group-id 3, there will be a conflict, the conflict is between ask the originator and 'allow' sending the message of type Unclassified to a node in group-id 3, therefore we implemented our work to make the originator's policy to be the prominent (?+

>?) which as a result to ask the originator for its up-to-date policy.

- **Case 6**:

  At Time 1: Assume that sender A sends a message (M) to node B attached with these policy rules:

  + Top secret $\rightarrow$ GID 1

  ? Top secret $\rightarrow$ GID 2

  + Top secret $\rightarrow$ GID 3

  At Time 2: The node B received the message (M). But node B has these policy rules:

  + Top secret $\rightarrow$ GID 1

  - Top secret $\rightarrow$ GID 2

  + Top secret $\rightarrow$ GID 3

  Therefore whenever Node B wants to send the message (M) to any node inside the group-id 2, there will be a conflict, the conflict is between ask the originator and disallow of sending the message of type Top secret to a node in group-id 2, therefore we implemented our work to make the originator's policy to be the prominent (?->?) which as a result to ask the originator for its up-to-date policy.

  Because of the variable but potentially high speed and random movement of vehicles, the topology of VANETs can change rapidly, and since the vehicles in VANETs are free to move, they can dynamically enter or leave the network. In consequence, the connectivity in VANETs may change frequently, therefore affecting originator availability during interaction with other receivers (disconnected or 'disappeared' from the network): this is the only limitation of this interaction mechanism.

5.6 Summary

This chapter described the originator requirements as a set of policy rules that explicitly instructs recipients how the information contained in messages must be disseminated to other nodes. This chapter introduced a suitable data dissemination policy to be used for the framework in which the originator of the message retains control over its dissemination provided that receivers of the message are trusted to enforce the policy. This chapter linked between the policy-based framework (described in Section 4.4) and data dissemination policy, in order to solve the research question, it also described the data dissemination policy language and the policy rules to be a suitable and understandable language for the framework. Finally it described the data dissemination policy rules conflict and when the originator should be asked for its up-to-date policy.

By presenting the data dissemination policy which takes into account the originator high-level requirements as low-level policy rules whose enforcement can be fully automated and understood for the framework, this chapter addressed the main research question as articulated in Section 1.3 "how can we prevent information from being leaked to undesirable entity(ies)?" and the other sub research questions "how to keep message contents private to an originator-defined subset of nodes in the VANET", "how to enforce the privacy and confidentiality requirement of the originator" and "how to represent the originator data dissemination requirements as a set of rules".

Chapter 6

Implementation

Objectives:

- Give an introduction about the Network Simulator (NS-2).
- Describe the NS-2 structure and components
- Present the implementation of our policy-based agent protocol.

6.1 Introduction

There are many network simulation tools available to implement the proposed framework and protocols for simulating both the wireless and wired networks. Since most of the research in *ad hoc* networks has been implemented using Network Simulator (NS-2) [4].

This chapter therefore, implements and simulates the policy-based framework (described in Chapter 4) as an NS-2 agent protocol with a suitable packet structure, and takes into consideration the originator high-level requirements as low-level policy rules (described in Chapter 5) whose enforcement can be fully automated and understood for the framework in order to solve the research question.

Section 6.2 provides a general overview of NS-2, and describes its structure, and illustrates why we chose the NS-2 in this work after discussing the usage of it in many research papers based on a previously published survey [4]. In order to ensure the message privacy and confidentiality requirements of the originator (controlling data dissemination), Section 6.3 and Section 6.4 therefore present the implementation of our policy-based agent protocol and a new packet structure to suit that protocol in NS-2. The new policy-based agent protocol is derived from an existing class in NS-2.

6.2 The Network Simulator (NS-2)

The Network Simulator (NS-2) is a real network environment simulator, is an open-source discrete event and object-oriented simulator intended mainly for networking research. NS-2 now considered as a reliable simulation tool for computer communication networks both in academia and industry. It was developed by the University of California at Berkeley, University of Southern California's Information Sciences Institute (USC/ISI), Lawrence Berkeley National Laboratory (LBNL) and Xerox Palo Alto Research Center (PARC)under the VINT (Virtual InterNetwork Testbed) project [122, 123]. Its main sponsors are the Defence Advanced Research Projects Agency (DARPA) and the National Science Foundation (NSF).

There are several versions of NS-2, the latest one is termed NS-3.12, it was released on 31 August 2011 [124]. This release is mainly a maintenance release, but contains a few new features and many bugs fixed. In our work, however, we used version NS-2.26 Allinone which was released on 19 October 2005 [125]. The simulator is installed on the Cygwin environment under the Microsoft Windows operating system (xp service pack 3). Cygwin is "a collection of tools which provide a Linux look and feel environment for Windows, Cygwin acts as a Linux API (Application Programming Interface) layer providing substantial Linux API functionality" [126].

There are many network simulation tools available to evaluate the performance of the proposed mechanisms and protocols for simulating both the wireless and wired networks. Most of the research in the *ad hoc* networks has been evaluated using the program NS-2. Other programs which have been used include Global Mobile Information System Simulation Library (GloMoSim),OPNET Modeler, QualNet, MATLAB, and CSIM. [127, 128, 129, 130]. In some cases, however, the researchers decided their work should be simulated using self-developed code [4].

Kurkowski *et al* [4] study analysed the 2000-2005 proceedings of the ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc). Their study showed that NS-2 is the most used of all simulators in MANET research: "35 of the 80 simulation papers that state the simulator used in the simulation study used NS-2 (approximately 44 percent)" as depicted in Figure 6.1 which shows the percentage of the academic papers produced by NS-2 compared with other simulators such as (GloMoSim), OPNET Modeler, QualNet, MATLAB, and CSIM. A more recent study by Abuarqoub *et al* [131], confirms that NS-2 is still the most commonly used simulator in the field of *ad hoc* networks. Because NS-2 is the most popular network simulator used till now [132], we decided therefore to use it in our work to check whether the policy-based framework ensures the message privacy and confidentiality requirements of the originator or not, and to demonstrate its effectiveness.

The free source feature in NS-2 increasingly encourages the research community to use it as a potential simulation tool which the researcher can modify and extend the source code. In addition to that NS-2 can support the simulation of the TCP, routing and security protocols for both wired and wireless networks .

The NS-2 simulator employs the clever idea of using two programming languages, C++ language to implement the low level simulation mechanisms, while the higher level activities (configurations setup, parameters) are implemented with Objective Tool Command Language (OTCL), an object-oriented extension of TCL (Tool Command Lan-

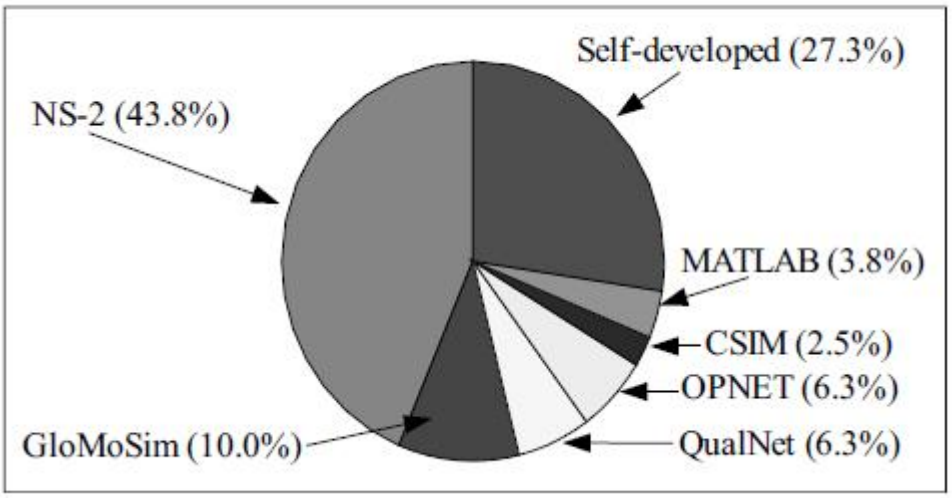


Figure 6.1: Simulator usage from MobiHoc survey [4]

guage); used to execute the script for the user's command. NS-2 has a rich library of network and protocol objects. Therefore NS-2 maintains two class hierarchies, the compiled C++ and the interpreted OTCL, with one-to-one correspondence relationships between C++ and OTCL classes as shown in the figure 6.2.

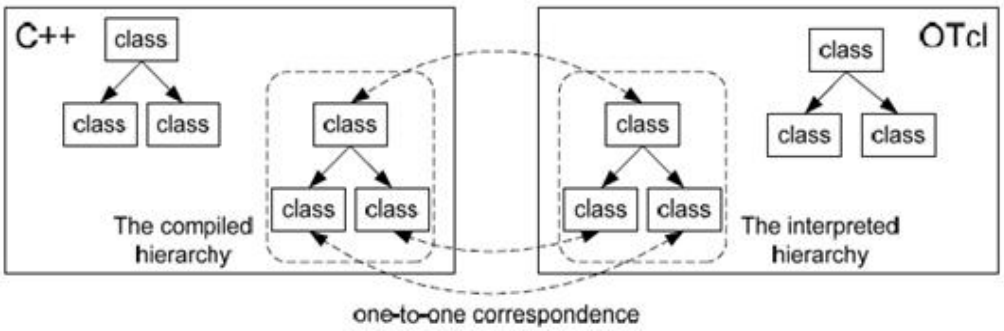


Figure 6.2: One-to-One Correspondence Relationship Between C++ and OTCL Classes

Using the compiled C++ hierarchy we can obtain efficient simulation and faster execution times. This is especially useful for the detailed protocols to reduce the packet and the processing time. Then in the OTCL script (the interpreted hierarchy) specified by the user, we can specify the network topologies, protocols and applications that we want to simulate (whose behaviour is already defined in the compiled hierarchy); we also can

specify the form of the output that we want to achieve from the simulator. The OTCL can derive benefit from the objects compiled in C++ through an OTCL linkage (using tclCL: a Tcl/C++ interface) that creates a matching of each OTCL object for each of the C++ (as shown in Listing 6.1). Therefore, from the user's point of view, NS-2 is an OTCL interpreter that takes an OTCL script as input, and generates a trace file as output [133, 75] as shown in the figure 6.3.

Listing 6.1: OTCl linkage between Tcl and C++

```
static class policyClass : public TclClass {
public:
     policyClass() : TclClass("Agent/policy") {}
     TclObject* create(int argc, const char*const* argv) {
     return (new policy());  }
} class_policy;
policy::policy():Agent(PT_POL),r(this)//PT_POL is added to packet.h
{bind("addr_",&addr);
  bind("dest_",&dst);
  }
```

NS uses two programming languages because the simulator has to deal with two different sorts of situations: on the one hand, simulating detailed protocols and applications requires a systems programming language such as C++ which can efficiently manage bytes, packet headers, and implement algorithms that run over large data sets. The turn-around time (run simulation, find bug, fix bug, recompile, re-run) is important, however, the run-time of these tasks is much more important. Even thought C++ requires appreciable time needed (to re-compile it after any change in code), it is still very fast to run (run on machine codes).

On the other hand, researchers are always interested in changing the network parameters, the topology configurations and scenarios. In these cases, however, iteration time factor is more important (such as change the model and re-run). Because the configura-

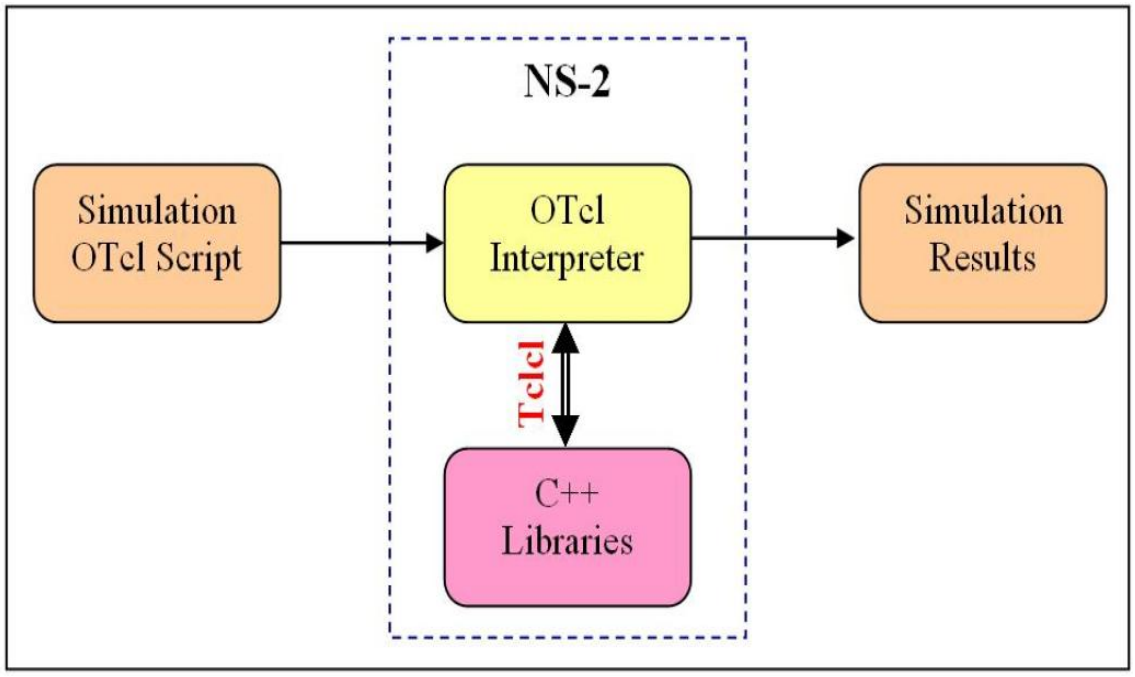


Figure 6.3: Schematic structure for NS-2

tion runs once (at the beginning of the simulation), run-time factor in this part of the task is less important. OTCL therefore, runs slower than C++ but the code can be changed quickly which makes it perfect for configuring the simulation parameters in an interactive manner with the user.

6.2.1 NAM file

NAM (Network Animator) is the default component of the NS-2 bundle and it is the only standard Graphical User Interface (GUI) in NS-2 [134]. Once the Tcl code is compiled, a trace file and NAM file are created, and hence the simulation is run at the packet level, NAM allows us to obtain detailed graphical results visualizing packet flows, queue length, packet drops, etc. In order to run the NAM file in the interface we need to define these commands in the Tcl file as described in Listing 6.2:

Listing 6.2: Tcl file example to create trace object

```
# Initialize Global Variables
# create simulator instance
set ns [new Simulator]
# setup topography object
set topo [new Topography]
# create trace object for ns and nam
set nf [open policy-namfile.nam w]
$ns namtrace-all $nf
set nc [open Policy-tracefile.tr w]
```

NAM interface allows the user to see the simulation while it is 'running'. NAM can visualise the network topology and display packet flows and queues. In addition, the time line scale feature in NAM and time actuator allow the user to go to any point and move forward or backward to a specific point in the simulation, and to increase or decrease the simulation running speed.

6.3 Agent

Agents in NS-2 represent endpoints where packets are assembled or consumed, they are used to implement protocols at different layers. The class Agent in NS-2 has an implementation part in OTcl and another part in C++. The C++ implementation is in ns/agent.cc and ns/agent.h, whereas the OTcl implementation is in ns/tcl/lib/ns-agent.tcl [122].

Our policy-based agent protocol is derived from the existing Agent class in NS-2 as shown in Listing 6.3. The class agent (called policy in the simulation) supports sending, receiving, forwarding packets, searching for adjacent, encryption, decryption, read from policy file, write to policy file, and policy check. The following member functions are implemented by the C++ policy class as shown in Listing 6.3. In sending process [Packet* allocpkt()] is used to allocate a new packet and a pointer from packet header type to assign values to fields in the packet header such as target, sequence, ack, size, org, stime, rtime,

data, policydata, hashvalue; whereas, fields such as addr, dst are assigned to values by using pointer from the ip header type (to be described in Section 6.4).

In the receiving process [recv (Packet*, Handler*)] is used to receive the packet by the agent, it is invoked whenever nodes are allowed to receive the packet after applying the send command in the simulation.

Listing 6.3: Our policy-based agent protocol

```
#include <stdio.h>
#include <stdlib.h>
#include"agent.h"
#include "ip.h"
#include "address.h"
#include "tclcl.h"
#include "packet.h"
#include <fstream>
#include <ctype.h>
#include <vector>
#include <iostream>
#include <sstream>
#include <string>
class policy : public Agent {
 public:
    policy::policy();
    virtual int command(int argc, const char*const* argv);
    void sendit();
    int* search_adjacent(int);
    void recv(Packet* P, Handler* );
    void encryption(char out[]);
    void decryption(char out[]);
    void writeto_policyfile(int);
    void readfrom_policyfile(int);
    void readfrom_policyfile1(int);
```

```
    void readnode_groupid();
    int Getgruopid(int);
    bool policy_check(int);
    int Getgruopid1(int);
    bool policy_check1(int, string);
    bool target_found(int);
    string get_Permit_group(int, string);
    void trasform2dim_1dim();
    void trasform1dim_2dim();
    string myitoa(int value, int base);
    string stringarray[45];
    string rcv_string1dim[100];
    string rcv_string2dim[18][5];
    int rec_permit[2];
    void init();
   };
```

6.4 Packet

A new packet structure is built as defined in Listing 6.4 to suit the policy-based agent protocol. From this packet structure two packets can be created and used in the simulation:

Listing 6.4: Our Packet structure

```
struct polic {
  int addr;
  int org;
  int dst;
  int target;
  int size;
  char ack;
  int seq;
```

```
    double stime;
    double rtime;
    char data[128];
    std::string policydata[100];
    unsigned int hashvalue;
          static int offset_;
          inline static int& offset() { return offset_; }
          inline static polic* access(const Packet* pol) {
                  return (polic*) pol->access(offset_);}
};
```

1. The first packet is called (pkt) which used to send the data attached with its policy rules from the source node to the destination node, this packet contains these fields:

   (a) The source address of the sending node (addr).

   (b) The originator address of the message (org).

   (c) The destination address to where the packet can be received as a relay node or as final destination node (dst).

   (d) The target node to which should receive the message (target).

   (e) Size of the packet (size), which is set to 200 bytes.

   (f) Acknowledgement flag (ack) to decide whether the packet can be sent to direct adjacent, or to a target which is far from the source node. This flag can carry three different values:

   - 0: means the target is direct adjacent to the source node.
   - 1: means the packet been received to the target and the target acknowledged that by set this flag to 1.
   - 2: means the target is far from the source so the flag is set to 2 by the source node to indicate to the receiver to forward the packet to the target

node.

(g) The sequence of the packets (seq).

(h) Sent time for the packet (stime).

(i) Receive time for the packet (rtime).

(j) The data to be sent which are messages in our case (data).

(k) The policy of each node to be attached in the packet to specify how the message can be used by the recipient node (policydata).

(l) The hash value which calculated at the sender node (hashvalue).

Listing 6.5 shows how the packet (pkt) can be created at the sender node in our policy-based agent protocol, it also shows how the fields of the packet (pkt) can be assigned (assembled) to different values in the simulation before is sent.

Listing 6.5: How the PKT can be created at the sender node

```
Packet* pkt = allocpkt();
hdr_ip* ih = hdr_ip::access(pkt);
        ns_addr_t w;
        ns_addr_t w2;
        w.addr_ = addr ;
        w2.addr_ = dst;
        ih->src_ =w;
        ih->dst_ =w2;


polic* eh = polic::access(pkt);
eh->stime = Scheduler::instance().clock();
eh->org=addr;
int tcl_target=atoi(argv[5]);
eh->target=tcl_target;
if(eh->target==dst){
```

```
eh->ak= 0;}
else{ eh->ak= 2;
                }
sequence++;
eh->seq= sequence;
hdr_cmn::access(pkt)->size()=200;
strcpy(eh->data,argv[4]);
trasform2dim_1dim();
memcpy (eh->policydata,Permit_group1dim,(RECORDS*FIELDS));
eh->hashvalue = hashing(eh->data,(unsigned int)strlen(eh->
    data));
encryption(eh->data);
send(pkt, 0);
```

2. The second packet is called packet return (pkt ret) which used for acknowledging the originator node that the data been received either successfully or been tampered with while in its way, by comparing the hash value attached with the message by the hash value calculated at the receiver node. This packet (pkt ret) is used only between the adjacent communication between source and destination nodes and it contains these fields:

   (a) The address of the node which has received the packet (addr).

   (b) The originator address of the message (org).

   (c) The destination address to where the packet originally come from (dst).

   (d) Size of the packet (size) which is set to 50 bytes.

   (e) Acknowledgement flag (ack) to tell the originator that the packet is received; it does not, however, give more information whether the data is been successfully received or been tampered with while in its way . This flag can carry one value only:

- 1: means the packet been received to the target and the target acknowledged that by set this flag to 1 in the packet return.

(f) The sequence of the packets (seq).

(g) Sent time for the packet (stime).

(h) Receive time for the packet (rtime).

(i) The data (data) to be sent is the authentication result to tell the originator more about whether the packet is received successfully (accepted because both hash values are equal) or was tampered with while in its way (message dropped because there is message integrity violation).

Listing 6.6 shows how the packet (pkt ret) can be created at the receiver node in our policy-based agent protocol, it also shows how the fields of the packet (pkt ret) can be assigned (assembled) to different values.

Listing 6.6: How to create packet return at the receiver node

```
void policy::recv(Packet* d, Handler* )
{
 Packet* pktret = allocpkt(); //def  new packet called pktret
 polic *eh1 = polic::access(pktret); //this one def a header
     from your header and connect it to the packet
 hdr_ip* ih = hdr_ip::access(pktret);
 hdr_ip* ih2 = hdr_ip::access(d);
 polic *eh2=polic::access(d);

 if(newhash==eh2->hashvalue)
 ns_addr_t w;
 ns_addr_t w2;
 ih->src_=ih2->dst_;
 ih->dst_=ih2->src_;
```

```
hdr_cmn::access(pktret)->size()=50;
eh1->rtime = recv_time;
eh1->stime = Scheduler::instance().clock();
strcpy(eh1->data, authenticate_result);
eh1->ak= 1;
send(pktret, 0);
}
```

These fields defined in Listing 6.4 can be used by any creating packet process in the agent, however not all of them must be used by any particular packet (for example, pkt ret do not use the fields policydata, hashvalue, target).

6.5 Security Design

In order to meet the message privacy and confidentiality requirements in our work, we need to check the security functions feasibility in NS-2 along with our policy-based agent. The purpose of introducing a simple encryption algorithm is only to illustrate a possibility of implementing a security function in NS-2. In this work, we also built a new packet structure to suit our protocol. The new policy-based protocol agent was derived from an existing class in NS-2 (as described in Listing 6.1) by adding encryption and decryption algorithms to secure the data field in the packet. Furthermore, message digest generation function (hash function) was implemented to ensure the integrity requirement of data.

6.6 Summary

We chose the NS-2 in this work because of the free source feature which increasingly encourages the research community to use it as a potential simulation tool, allows the researcher to modify and extend the source code. The contribution of this chapter is to

implement and simulate the policy-based framework (described in Chapter 4) as an NS-2 agent protocol with a suitable packet structure, and takes into consideration the originator high-level requirements as low-level policy rules (described in Chapter 5) whose enforcement can be fully automated and understood for the framework in order to solve the research question. In this chapter, therefore we implemented a new agent protocol and a new packet structure to suit that protocol in NS-2 (respectively described in Sections 6.3 and 6.4). The new policy-based protocol agent was derived from an existing class in NS-2 by adding encryption and decryption algorithms to secure the data field in the packet. Furthermore, a message digest generation function (hash function) was also implemented within NS-2 to ensure the integrity requirement of data.

Chapter 7

Evaluation through case study

Objectives:

- Present case studies to show how data dissemination will be controlled.
- Demonstrate the importance of the proposed research.
- Evaluate the proposed research through case studies.
- Discuss the limitation(s) of the proposed research.

7.1 Introduction

In the previous chapter we discussed the implementation characteristics of NS-2 to implement and to simulate our policy-based framework as an NS-2 agent protocol with a suitable packet structure. This chapter examines the results of the simulation when applying the originator message requirements as policy rules (described in Chapter 5) attached to packets at every node in the VANET to solve the research question; this evaluation is done by creating different network topologies using Tcl (Tool Command Language) in NS-2 to represent/simulate some nodes communicating between themselves, to check

whether the policy-based agent protocol achieves the originator goal to keep message contents private to a originator-defined subset of nodes in the VANET or not.

Network Simulator (NS-2) is a real network environment simulator, which is used to test the performance of the proposed policy-based protocol and demonstrate its effectiveness using various network performance metrics (average delay and overhead). This chapter also presents four case studies to show how data dissemination can be controlled based on the policy of the originator. These case studies have been provided to show how the policy-based framework components interact together to control the data dissemination between nodes within the NS-2 Simulation. The results of these case studies are intended to show the feasibility of our research to control the data dissemination between nodes in VANETs and to demonstrate how it works.

This chapter evaluates our policy-based framework depending on these success criteria:

- Feasibility of the implementation.
- Ensuring the message confidentiality requirement based on the originator data dissemination policy rules.
- Proving the ability of the originator of the message to control the data dissemination between nodes within NS-2 simulation.
- Deciding when the originator should be asked for its up-to-date data dissemination policy.
- Average delay performance.

Section 7.6 describes some of the simulation environment parameters and illustrates why we chose the NS-2 in this work. The thesis shows two types of algorithms for applying the originator requirements as numbers (to be described in Section 7.2) in the files of the nodes (Algorithm 1) whereas the other one as policy rules (Algorithm 2) (to be

described in Section 7.3, 7.4, and 7.5). Finally, Section 7.8 presents the result and discussion.

7.2 Case Study (1) for Algorithm 1

The thesis shows two types of algorithms for applying the originator requirements as numbers (to be described in this section) in the files of the nodes (Algorithm 1) whereas the other one as policy rules (Algorithm 2) (to be described in Section 7.3, 7.4, and 7.5). In Figure 7.1 we show an example of six nodes, assuming that each node in the system has a group-id number, means we are classifying the nodes in our work into different groups, which in our case three groups: group-id 1, group-id 2, and group-id 3. The first group contains node 0, node 1 and node 5. Whereas group-id 2 contains node 2 and node 4. Finally, group-id 3 contains only node 3.

Listing 7.1: Tcl Command for the case study (1): Part A

```
$ns at 1.0 "$u(0) 0 send start_the_mission_A 1"
```

In this case study assume that node 0 'wants' to send a message (start the mission A) to node 1 by executing this command in Tcl as shown in Listing 7.1 with a condition or a requirement attached; the originator of message (node 0) provide the specification of the desired behaviour that node 1 must possess with respect to this particular message flow; the requirement that node 0 tells node 1 to send the message (start the mission A) to nodes only nodes exist in the group-id specified in the policy file at file0.txt in node 0, and we call this group-id in this situation a permitted group as shown in the algorithm chart in Figure 7.2. If file0.txt as in the example has 1 that means only nodes in the group-id 1 can receive the packet. Then node 0 will start searching for the adjacent nodes in the range. In this example node 0 will find node 1 and node 2. dest1=n1, dest2=n2. Now node 0 will check if dest1 in the permitted group or not, and do the same for dest2 also. In the

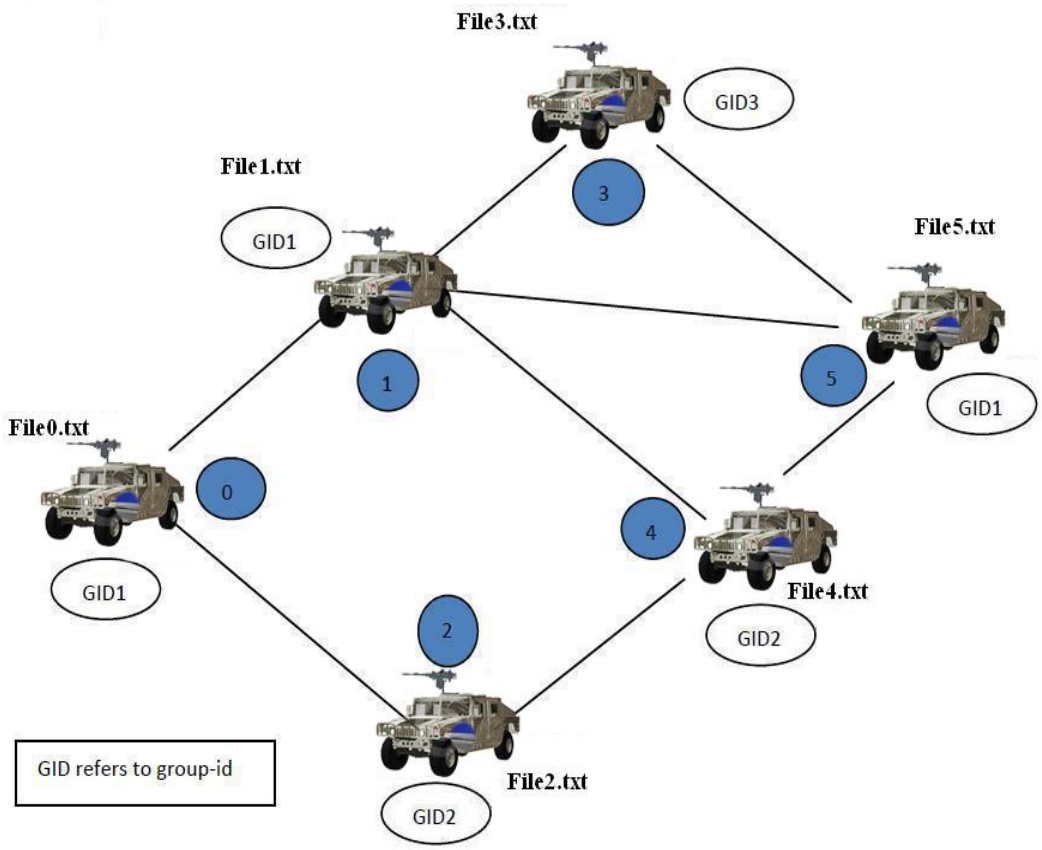


Figure 7.1: Example to illustrate organising nodes into groups for the case study (1)

algorithm chart this is depicted as Getgroupid (dest) process and checks if group-id equals the permitted group or not. In this example it will be yes for node 1 as node 1 is in the group-id 1. Node 0 will send to node 1 not only the packet it also sends its policy which existed in file0.txt, whereas node 1 will create a packet handler to receive the packet; once it has received the policy of node 1 will be updated according to policy of node 0 and deletes its old policy because it is the originator policy.

The result of the simulation for this case study is shown in Listing 7.2

Listing 7.2: The result of the simulation for the case study 1: Part A

```
Index 0  0 Contains integers : 0   Index 0  1 Contains integers : 1
Index 1  0 Contains integers : 1   Index 1  1 Contains integers : 1
Index 2  0 Contains integers : 2   Index 2  1 Contains integers : 2
```

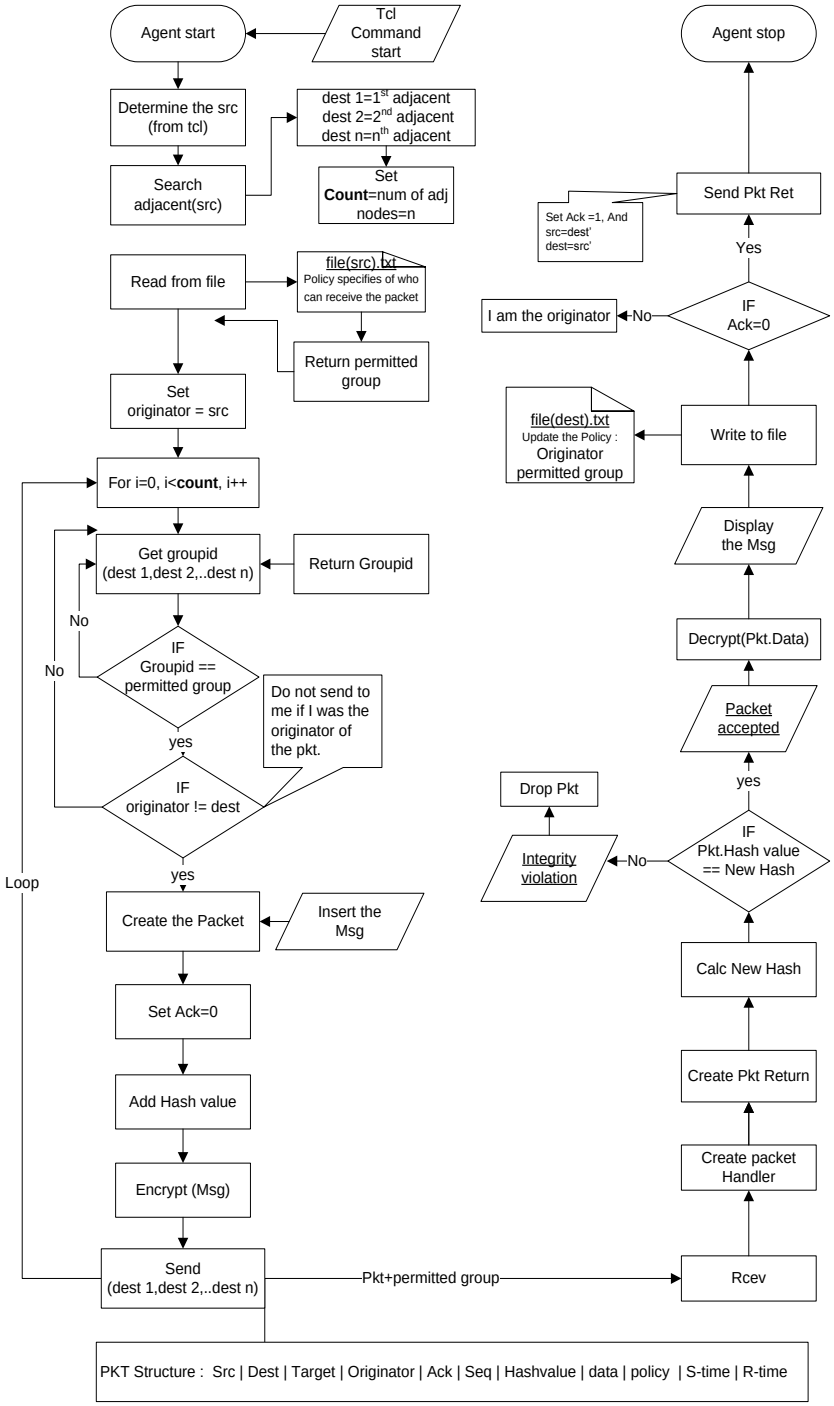


Figure 7.2: Algorithm One: Send and Receive Chart Algorithm

```
Index 3  0 Contains integers : 3   Index 3  1 Contains integers : 3
Index 4  0 Contains integers : 4   Index 4  1 Contains integers : 2
Index 5  0 Contains integers : 5   Index 5  1 Contains integers : 1

Opened node0.txt for reading.
Permit_group array has in Index 0 Integers : 1
The adjacent nodes in search adj function of node 0 is 1
The adjacent nodes in search adj function of node 0 is 2

********************sending + check policy
the dest is  1 in groupid: 1
seq value in the header 1
Message sent start_the_mission_A with hashing 32673884
The Packet Sent successfully, the source is 0 The destination is 1
   at time 1

getgroup id function return -1 OR the policycheck function return
   false for sending to node 2

The packet received at dest node 1 from 0
at time 1.0036
dest is 1

Opened node1.txt for writing.

newhash has: 32673884
The encrypted data is: vwduwbwkhbplvvlrqbD
The original data after decr has: start_the_mission_A
The hashvalue has 32673884
Message_Accepted
send packet return
I am node 1 Thanks, the Packet received successfully from node 0
```

Now assume that file1.txt as in the example has 3 that means only nodes in the group-id 3 can receive the message, if node 1 at a later time 'wants' to send the same message (start the mission A) to node 3 by executing this command in Tcl as shown in Listing 7.3. Let us see whether the message is allowed to be sent or not depends on the policy rules which node 1 has already received from node 0. According to the original policy rule of node 1, the message can be sent to node 3 because node 3 is in group-id 3, however this does not agree with the policy rules of the originator of the message (node 0). So the old policy is deleted and the new one is taken into consideration. The result of this simulation is shown in Listing 7.4.

Listing 7.3: Tcl Command for the case study (1): Part B

```
$ns at 2.0 "$u(1) 1 send start_the_mission_A 3"
```

Listing 7.4: The result of the simulation for the case study 1: Part B

```
Opened groupid.txt for reading.
Index 0  0 Contains integers : 0   Index 0  1 Contains integers : 1
Index 1  0 Contains integers : 1   Index 1  1 Contains integers : 1
Index 2  0 Contains integers : 2   Index 2  1 Contains integers : 2
Index 3  0 Contains integers : 3   Index 3  1 Contains integers : 3
Index 4  0 Contains integers : 4   Index 4  1 Contains integers : 2
Index 5  0 Contains integers : 5   Index 5  1 Contains integers : 1

Opened node1.txt for reading.

Permit_group array has in Index 0 Integers : 1

The adjacent nodes in search adj function of node 1 is 0
The adjacent nodes in search adj function of node 1 is 3
The adjacent nodes in search adj function of node 1 is 4
The adjacent nodes in search adj function of node 1 is 5
```

```
getgroup id function return -1 OR the policycheck function return
    false for sending to node 0
org equal dest

getgroup id function return -1 OR the policycheck function return
    false for sending to node 3
Reject to send
```

Whenever Node 1 ‘wants’ to send the message (start the mission A) to any node inside the group-id 3, there will be a conflict, therefore we implemented our work to make the originators policy to be the dominant which, as a result disallows the flow of the message, and this can be seen from Listing 7.4, the message is rejected from being sent to node 3.

7.3 Case Study (2) for Algorithm 2

This section shows the second type of algorithm for applying the originator requirements as policy rules (Algorithm 2). As similar to Figure 7.1 in case study 1, the network topology in case study 2 is shown in Figure 7.3, we show an example of six nodes, assuming that each node in the system has a group-id number, means we are classifying the nodes in our work into different groups, which in our case three groups: group-id 1, group-id 2, and group-id 3. The first group contains node 0, node 2 and node 5. Whereas group-id 2 contains node 1 and node 4 . Finally, group-id 3 contains only node 3.

In this case study assume that node 0 ‘wants’ to send a secret message (start the mission 1 at 8 am) to node 2 by executing this command in Tcl as shown in Listing 7.5 with these conditions or requirements attached to the message:

Listing 7.5: Tcl Command for the case study (2): Part A

```
$ns at 1.0 "$u(0) 0 send secret start_the_mission1_at8am 2"
```

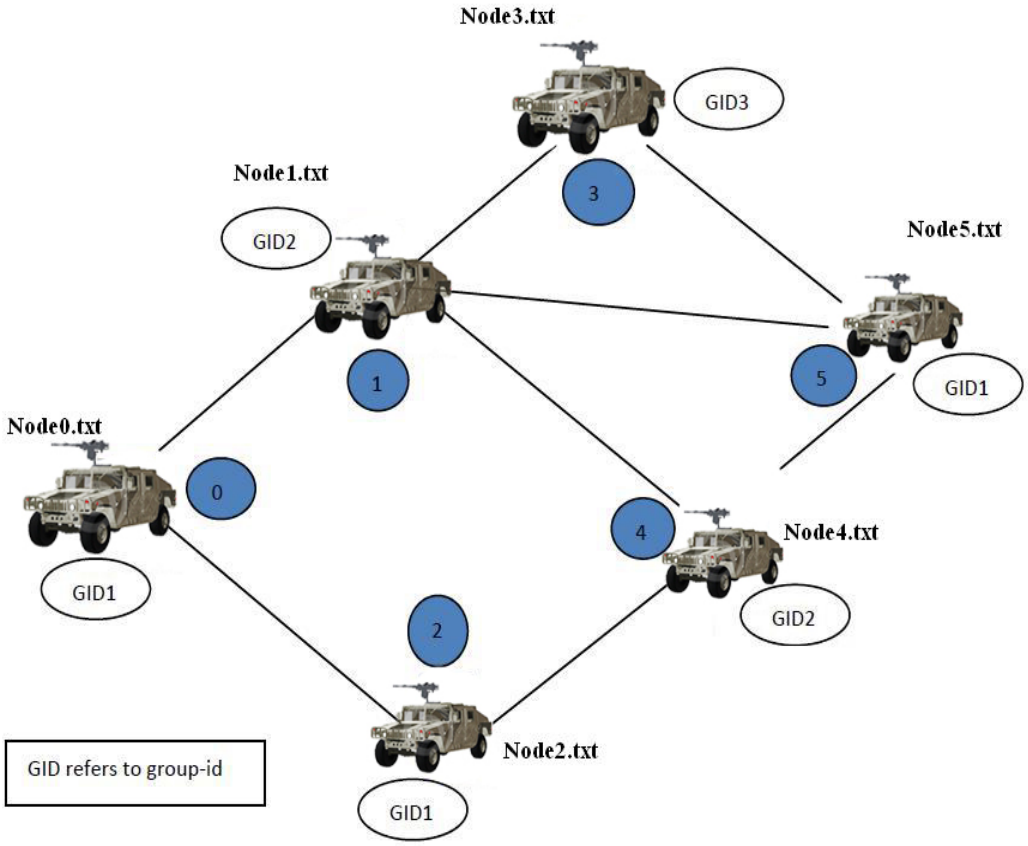


Figure 7.3: Example to illustrate organising nodes into groups for the case study (2)

- allow sending a Top secret type of message from source to destination if the destination is in the group-id 1.
- disallow sending a Top secret type of message from source to destination if the destination is in the group-id 2 or group-id 3.
- allow sending a Secret type of message from source to destination if the destination is in the group-id 1.
- disallow sending a Secret type of message from source to destination if the destination is in the group-id 2 or group-id 3.
- allow sending a Unclassified type of message from source to destination if the destination is in the group-id 1 or group-id 2 or group-id 3.

These conditions are specified in the policy file at node0.txt in node 0, and we call these conditions data dissemination policy rules as shown in the algorithm chart in Figure 7.8. So the node0.txt as in the example has the following rules:

+ Top secret → GID 1

- Top secret → GID 2

- Top secret → GID 3

+ Secret → GID 1

- Secret → GID 2

- Secret → GID 3

+ Unclassified → GID 1

+ Unclassified → GID 2

+ Unclassified → GID 3

That means only nodes in the group-id 1 can receive the Secret message. Then node 0 will start searching for the adjacent nodes in the range. In this example node 0 will find node 1 and node 2. dest1=n1, dest2=n2. Now node 0 will check its policy check function for the dest1 as shown in the algorithm chart in Figure 7.8. If the function returns true the message will be sent otherwise it will not be sent, and do the same for dest2 also. In the algorithm chart this is depicted as policy-check (gid, priority) process and checks if group-id of the node is satisfying the policy requirement or not. In this example it will be yes for node 2 as node 2 is in the group-id 1. Node 0 will send to node 2 not only the packet it also sends its policy which existed in node0.txt, whereas node 2 will create a packet handler to receive the packet, once it received the policy of node 2 will be updated according to policy of node 0 by appending policy of node 0 to the file of node 2 (node2.txt).

The result of the simulation for this case study is shown in Listing 7.6

Listing 7.6: The result of the simulation for the case study 2: Part A

```
Opened groupid.txt for reading.
```

```
Node 0 relate to groupid : 1
Node 1 relate to groupid : 2
Node 2 relate to groupid : 1
Node 3 relate to groupid : 3
Node 4 relate to groupid : 2
Node 5 relate to groupid : 1

Opened node0.txt for reading.
+   topsecret   ->   gid   1
-   topsecret   ->   gid   2
-   topsecret   ->   gid   3
+   secret   ->   gid   1
-   secret   ->   gid   2
-   secret   ->   gid   3
+   unclassified   ->   gid   1
+   unclassified   ->   gid   2
+   unclassified   ->   gid   3

The adjacent nodes in search adj function of node 0 is 1
The adjacent nodes in search adj function of node 0 is 2
----------------------1
The Msg has secret priority
Calling The policy Check Function To Decide To Where The Packet
    secret Can Be Routed To Reach The Target
The policy check function for sending to node 1 return False
----------------------2
The Msg has secret priority
Calling The policy Check Function To Decide To Where The Packet
    secret Can Be Routed To Reach The Target
**********************sending + Check policy-->>True
the dest is  2 in groupid: 1
 The message we send is start_the_mission1_at8am with hashing
```

```
     752783452
The Packet Sent successfully, the source is 0 The destination is 2
   at time 1
The packet recieved at dest node 2 from 0
at time 1.0036
dest is 2
Opened node2.txt for writing.
+   topsecret   ->   gid   1
-   topsecret   ->   gid   2
-   topsecret   ->   gid   3
+   secret   ->   gid   1
-   secret   ->   gid   2
-   secret   ->   gid   3
+   unclassified   ->   gid   1
+   unclassified   ->   gid   2
+   unclassified   ->   gid   3

newhash has: 752783452
The encrypted data is: wxevxcxlicqmwwmsr5cex<eq
The original data after decr is: start_the_mission1_at8am
The hashvalue is 752783452
Message_Accepted
send packet return
I am node 2 Thanks, the Packet received successfully from node 0
```

Now, assume that node 2 has these policy rules at node2.txt:

+ Top secret	→	GID 1
- Top secret	→	GID 2
- Top secret	→	GID 3
+ Secret	→	GID 1
+ Secret	→	GID 2
- Secret	→	GID 3
+ Unclassified	→	GID 1
+ Unclassified	→	GID 2
+ Unclassified	→	GID 3

If node 2 at a later time 'wants' to send the same Secret message (start the mission 1 at 8am) to node 4 by executing this command in Tcl as shown in Listing 7.7. Let us see whether the message is allowed to be sent to node 4 or not depends on the policy rules which node 2 has already received from node 0. According to the original policy rules of node 2, Secret message can be sent to node 4 because node 4 is in group-id 2, however this does not agree with the policy rules of the originator of the message (node 0). The result of this simulation is shown in Listing 7.8.

Listing 7.7: Tcl Command for the case study (2): Part B

```
$ns at 1.0 "$u(2) 2 send secret start_the_mission1_at8am 4"
```

Listing 7.8: The result of the simulation for the case study 2: Part B

```
Opened groupid.txt for reading.
Node 0 relate to groupid : 1
Node 1 relate to groupid : 2
Node 2 relate to groupid : 1
Node 3 relate to groupid : 3
Node 4 relate to groupid : 2
Node 5 relate to groupid : 1
```

```
Opened node2.txt for reading.
+   topsecret   ->   gid   1
-   topsecret   ->   gid   2
-   topsecret   ->   gid   3
+   secret   ->   gid   1
+   secret   ->   gid   2
-   secret   ->   gid   3
+   unclassified   ->   gid   1
+   unclassified   ->   gid   2
+   unclassified   ->   gid   3
+   topsecret   ->   gid   1
-   topsecret   ->   gid   2
-   topsecret   ->   gid   3
+   secret   ->   gid   1
-   secret   ->   gid   2
-   secret   ->   gid   3
+   unclassified   ->   gid   1
+   unclassified   ->   gid   2
+   unclassified   ->   gid   3

The adjacent nodes in search adj function of node 2 is 0
The adjacent nodes in search adj function of node 2 is 4
---------------------1
The Msg has secret priority
Calling The policy Check Function To Decide To Where The Packet
    secret Can Be Routed To Reach The Target
*********************sending + Check policy-->>True
the dest is  0 in groupid: 1
org equal dest
---------------------2
The Msg has secret priority
```

```
Calling The policy Check Function To Decide To Where The Packet
    secret Can Be Routed To Reach The Target
The policy check function for sending to node 4 return False
the dest is  4 in groupid: 2
Reject to send
```

Whenever Node 2 'wants' to send the message (start the mission1 at 8am) to any node inside the group-id 2, there will be a conflict, the conflict is between 'disallow' and 'allow' sending the message of type Secret to a node in group-id 2, therefore we implemented our work to make the originator's policy to be the dominant (as described in Section 5.5) which, as a result disallows the flow of the message, and this can be seen from Listing 7.8, the message is rejected from being sent to node 4.

7.4 Case Study (3) for Algorithm 2

The network topology in this case study is shown in Figure 7.4, we show an example of six nodes, assuming that each node in the system has a group-id number, means we are classifying the nodes in our work into different groups, which in our case three groups: group-id 1, group-id 2, and group-id 3. The first group contains node 0, node 2, node 4 and node 5. Whereas group-id 2 contains node 1. Finally, group-id 3 contains only node 3. In this case study assume that node 0 'wants' to send a Top secret message (start the mission 2 at 9 am) to node 4 by executing this command in Tcl as shown in Listing 7.9 with these conditions or requirements attached to the message:

Listing 7.9: Tcl Command for the case study (3): Part A

```
$ns at 2.0 "$u(0) 0 send topsecret start_the_mission2_at9am 4"
```

- allow sending a Top secret type of message from source to destination if the destination is in the group-id 1.

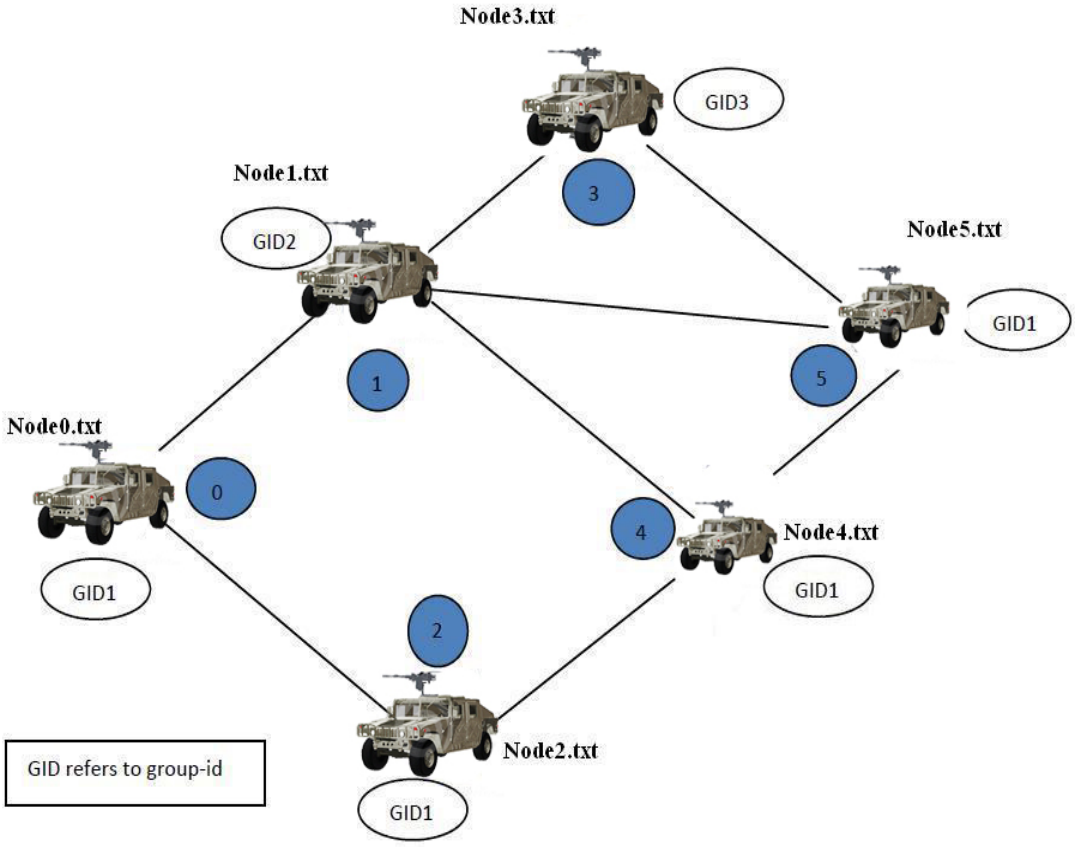


Figure 7.4: Example to illustrate organising nodes into groups for the case study (3)

- disallow sending a Top secret type of message from source to destination if the destination is in the group-id 2 or group-id 3.
- allow sending a Secret type of message from source to destination if the destination is in the group-id 1 or group-id 2.
- disallow sending a Secret type of message from source to destination if the destination is in the group-id 3.
- allow sending a Unclassified type of message from source to destination if the destination is in the group-id 1 or group-id 2 or group-id 3.

These conditions are specified in the policy file at node0.txt in node 0 as follows:

+ Top secret	→	GID 1
- Top secret	→	GID 2
- Top secret	→	GID 3
+ Secret	→	GID 1
+ Secret	→	GID 2
- Secret	→	GID 3
+ Unclassified	→	GID 1
+ Unclassified	→	GID 2
+ Unclassified	→	GID 3

That means only nodes in the group-id 1 can receive the Top secret message. Then node 0 will start searching for the adjacent nodes in the range. In this example, node 0 will find node 1 and node 2. dest1=n1, dest2=n2. Now node 0 will check its policy check function for the dest1 as shown in the algorithm chart in Figure 7.8. If the function returns true, the message will be sent otherwise it will not be sent, and do the same for dest2 also. In the algorithm chart this is depicted as policy-check (gid, priority) process and checks if group-id of the node is satisfying the policy requirement or not. In this example it will be yes for node 2 as node 2 is in the group-id 1. Node 0 will set the ack flag to 2 which tells node 2 to forward the packet to the target (node 4). Node 0 will send to node 2 not only the packet it also sends its policy which existed in node0.txt. When node 2 receives the packet to forward it, but it cannot read the message because it is encrypted.

Node 2 will start searching for the adjacent nodes in the range. In this example, node 2 will find node 0 and node 4. dest1=n1, dest2=n2. Now node 2 will check its policy check function for the dest1 as shown in the algorithm chart in Figure 7.8. If the function returns true, the message will be sent otherwise it will not be sent, however node 0 is the originator, so it will not be sent to node 0, and do the same check for dest2 also. In the algorithm chart this is depicted as policy-check (gid, priority) process and checks if

group-id of the node is satisfying the policy requirement or not. In this example it will be yes for node 4 as node 4 is in the group-id 1. Node 2 send to node 4 not only the packet it also sends its policy which existed in node0.txt, once it received the policy of node 4 will be updated according to policy of node 0 by appending policy of node 0 to the file of node 4 (node4.txt).

The result of the simulation for this case study is shown in Listing 7.10

Listing 7.10: The result of the simulation for the case study 3: Part A

```
Opened groupid.txt for reading.
Node 0 relate to groupid : 1
Node 1 relate to groupid : 2
Node 2 relate to groupid : 1
Node 3 relate to groupid : 3
Node 4 relate to groupid : 1
Node 5 relate to groupid : 1

Opened node0.txt for reading.
+   topsecret   ->   gid   1
-   topsecret   ->   gid   2
-   topsecret   ->   gid   3
+   secret   ->   gid   1
-   secret   ->   gid   2
-   secret   ->   gid   3
+   unclassified   ->   gid   1
+   unclassified   ->   gid   2
+   unclassified   ->   gid   3

The adjacent nodes in search adj function of node 0 is 1
The adjacent nodes in search adj function of node 0 is 2
--------------------1
The Msg has topsecret priority
```

```
Calling The policy Check Function To Decide To Where The Packet Top
    secret Can Be Routed To Reach The Target
The policy check function for sending to node 1 return False
----------------------2
The Msg has topsecret priority
Calling The policy Check Function To Decide To Where The Packet Top
    secret Can Be Routed To Reach The Target
**********************sending + Check policy-->>True
the dest is  2 in groupid: 1
The message we send is start_the_mission2_at9am with hashing
    754487388
The Packet Sent successfully, the source is 0 The destination is 2
    at time 2
******************************
stop_flag==0, Will be Forwarded by node 2
eh2->target!=temdest, target is 4
The adjacent nodes in search adj function of node 2 is 0
The adjacent nodes in search adj function of node 2 is 4
**************frwd
org equal dest
the packet will be forwarded to node 4 by node 2
The Packet Sent successfully, the source is 2 The destination is 4
    at time 2.0036
******************************
The target receives the pkt
The packet received at dest node 4 from 2
at time 2.0072


Opened node4.txt for writing.
+   topsecret   ->   gid   1
-   topsecret   ->   gid   2
-   topsecret   ->   gid   3
```

```
+   secret    ->   gid   1
-   secret    ->   gid   2
-   secret    ->   gid   3
+   unclassified    ->   gid   1
+   unclassified    ->   gid   2
+   unclassified    ->   gid   3

newhash has: 754487388
The encrypted data is: wxevxcxlicqmwwmsr6cex=eq
The original data after decr is: start_the_mission2_at9am
The hashvalue is754487388
Message_Accepted
```

Now, assume that node 4 has these policy rules at node4.txt:

- Top secret → GID 1
+ Top secret → GID 2
- Top secret → GID 3
+ Secret → GID 1
+ Secret → GID 2
- Secret → GID 3
+ Unclassified → GID 1
+ Unclassified → GID 2
+ Unclassified → GID 3

If node 4 at a later time 'wants' to send the same Top secret message (start the mission 2 at 9am) to node 5 by executing this command in Tcl as shown in Listing 7.11. Let us see whether the message is allowed to be sent to node 5 or not depends on the policy rules which node 4 has already received from node 0. According to the original policy rule of node 4, Top secret message cannot be sent to node 5 because node 5 is in group-id 1, however this does not agree with the policy rules of the originator of the message (node

0). The result of this simulation is shown in Listing 7.12.

Listing 7.11: Tcl Command for the case study (3): Part B

```
$ns at 3.0 "$u(4) 4 send topsecret start_the_mission2_at9am 5"
```

Listing 7.12: The result of the simulation for the case study 3: Part B

```
Opened groupid.txt for reading.
Node 0 relate to groupid : 1
Node 1 relate to groupid : 2
Node 2 relate to groupid : 1
Node 3 relate to groupid : 3
Node 4 relate to groupid : 1
Node 5 relate to groupid : 1

Opened node4.txt for reading.
-   topsecret   ->   gid   1
+   topsecret   ->   gid   2
-   topsecret   ->   gid   3
+   secret   ->   gid   1
+   secret   ->   gid   2
-   secret   ->   gid   3
+   unclassified   ->   gid   1
+   unclassified   ->   gid   2
+   unclassified   ->   gid   3
+   topsecret   ->   gid   1
-   topsecret   ->   gid   2
-   topsecret   ->   gid   3
+   secret   ->   gid   1
-   secret   ->   gid   2
-   secret   ->   gid   3
+   unclassified   ->   gid   1
+   unclassified   ->   gid   2
+   unclassified   ->   gid   3
```

```
The adjacent nodes in search adj function of node 4 is 1
The adjacent nodes in search adj function of node 4 is 2
The adjacent nodes in search adj function of node 4 is 5
----------------------1
The Msg has topsecret priority
Calling The policy Check Function To Decide To Where The Packet
   topsecret Can Be Routed To Reach The Target
The policy check function for sending to node 1 return False
----------------------2
The Msg has topsecret priority
Calling The policy Check Function To Decide To Where The Packet
   topsecret Can Be Routed To Reach The Target
org equal dest
----------------------3
The Msg has topsecret priority
Calling The policy Check Function To Decide To Where The Packet
   topsecret Can Be Routed To Reach The Target
**********************sending + Check policy-->>True
the dest is  5 in groupid: 1
The message we send is start_the_mission2_at9am with hashing
   754487388
The Packet Sent successfully, the source is 4 The destination is 5
   at time 3

The packet recieved at dest node 5 from 4
at time 3.0036
dest is 5

Opened node5.txt for writing.
+   topsecret   ->   gid   1
+   topsecret   ->   gid   2
```

```
-   topsecret   ->   gid   3
+   secret   ->   gid   1
+   secret   ->   gid   2
-   secret   ->   gid   3
+   unclassified   ->   gid   1
+   unclassified   ->   gid   2
+   unclassified   ->   gid   3
+   topsecret   ->   gid   1
-   topsecret   ->   gid   2
-   topsecret   ->   gid   3
+   secret   ->   gid   1
-   secret   ->   gid   2
-   secret   ->   gid   3
+   unclassified   ->   gid   1
+   unclassified   ->   gid   2
+   unclassified   ->   gid   3

newhash has: 754487388
 The encrypted data is: wxevxcxlicqmwwmsr6cex=eq
 The original data after decr is: start_the_mission2_at9am
The hashvalue is754487388
Message_Accepted

send packet return
I am node 5 Thanks, the Packet received successfully from node 4
```

Whenever Node 4 ‘wants’ to send the Top secret message (start the mission 2 at 9am) to any node inside the group-id 1, there will be a conflict, the conflict is between ‘allow’ and ‘disallow’ sending the message of type Top secret to a node in group-id 1, therefore we implemented our work to make the originator’s policy to be the dominant (as described in Section 5.5) which, as a result allows the flow of the message, and this can be seen from

Listing 7.12, the message is sent to node 5.

7.5 Case Study (4) for Algorithm 2

The network topology in this case study is shown in Figure 7.5, we show an example of six nodes, assuming that each node in the system has a group-id number, means we are classifying the nodes in our work into different groups, which in our case three groups: group-id 1, group-id 2, and group-id 3. The first group contains node 0, node 1, node 4 and node 5. Whereas group-id 2 contains node 2. Finally, group-id 3 contains only node 3. In this case study assume that node 0 'wants' to send a Top secret message (start the mission 3 at 10 am) to node 1 by executing this command in Tcl as shown in Listing 7.13 with these conditions or requirements attached to the message:

Listing 7.13: Tcl Command for the case study (4): Part A

```
$ns at 1.0 "$u(0) 0 send topsecret start_the_mission3_at10am 1"
```

- allow sending a Top secret type of message from source to destination if the destination is in the group-id 1 or group-id 2.
- ask the originator for its up-to-date policy before trying to send Top secret type of message from source to destination if the destination is in the group-id 3.
- allow sending a Secret type of message from source to destination if the destination is in the group-id 1 or group-id 2.
- disallow sending a Secret type of message from source to destination if the destination is in the group-id 3.
- allow sending a Unclassified type of message from source to destination if the destination is in the group-id 1 or group-id 2 or group-id 3.

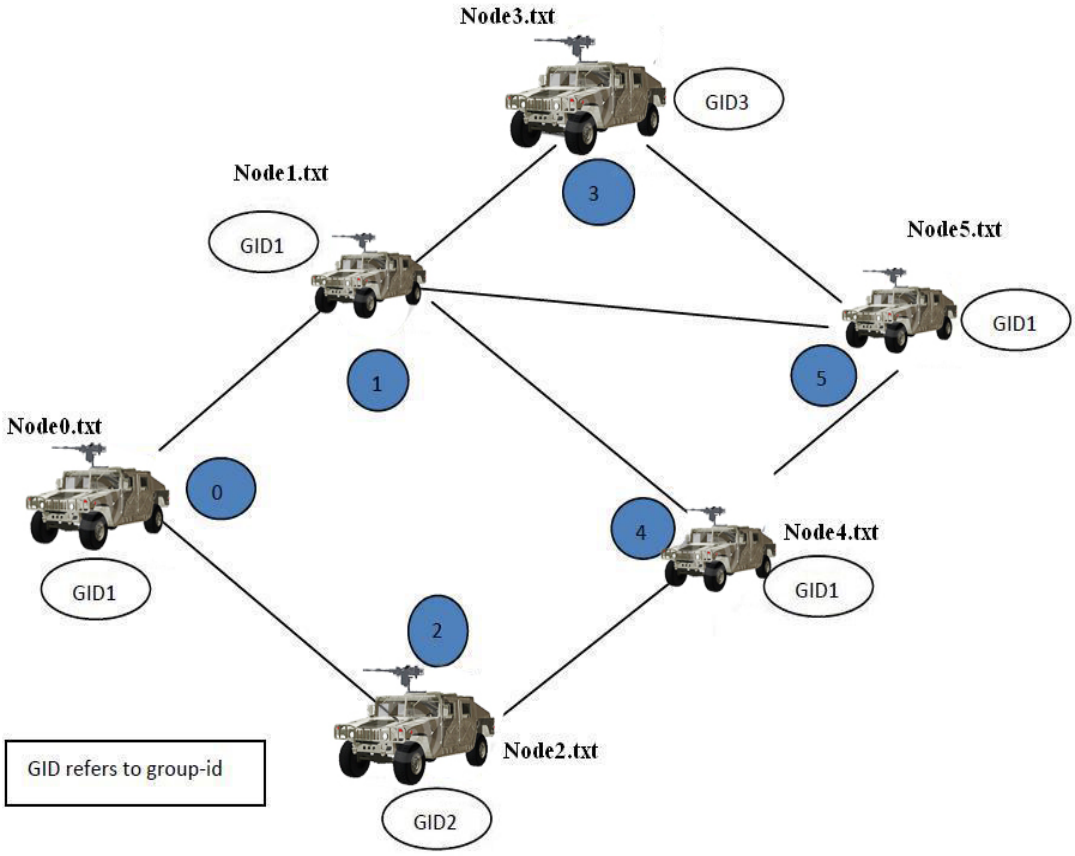


Figure 7.5: Example to illustrate organising nodes into groups for the case study (4)

These conditions are specified in the policy file at node0.txt in node 0 as follows:

+ Top secret → GID 1

+ Top secret → GID 2

? Top secret → GID 3

+ Secret → GID 1

+ Secret → GID 2

- Secret → GID 3

+ Unclassified → GID 1

+ Unclassified → GID 2

+ Unclassified → GID 3

That means only nodes in the group-id 1 and group-id 2 can receive the Top secret message. Then node 0 will start searching for the adjacent nodes in the range. In this example, node 0 will find node 1 and node 2. dest1=n1, dest2=n2. Now node 0 will check its policy check function for the dest1 as shown in the algorithm chart in Figure 7.8. If the function returns true, the message will be sent otherwise it will not be sent, and do the same for dest2 also. In the algorithm chart this is depicted as policy-check (gid, priority) process and checks if group-id of the node is satisfying the policy requirement or not. In this example it will be yes for both node 1 and node 2 as they are in the group-id 1 and group-id 2 respectively. Node 1, however, is the target, so node 0 will send to node 1 not only the packet it also sends its policy which existed in node0.txt, whereas node 1 will create a packet handler to receive the packet, once it received the policy of node 1 will be updated according to policy of node 0 by appending policy of node 0 to the file of node 1 (node1.txt).

The result of the simulation for this case study is shown in Listing 7.14

Listing 7.14: The result of the simulation for the case study 4: Part A

```
Opened groupid.txt for reading.
Node 0 relate to groupid : 1
Node 1 relate to groupid : 1
Node 2 relate to groupid : 2
Node 3 relate to groupid : 3
Node 4 relate to groupid : 1
Node 5 relate to groupid : 1

Opened node0.txt for reading.
+   topsecret   ->   gid   1
+   topsecret   ->   gid   2
?   topsecret   ->   gid   3
+   secret   ->   gid   1
+   secret   ->   gid   2
```

```
-   secret   ->   gid   3
+   unclassified   ->   gid   1
+   unclassified   ->   gid   2
+   unclassified   ->   gid   3

The adjacent nodes in search adj function of node 0 is 1
The adjacent nodes in search adj function of node 0 is 2
----------------------1
The Msg has topsecret priority
Calling The policy Check Function To Decide To Where The Packet
   topsecret Can Be Routed To Reach The Target
**********************sending + Check policy-->>True
the dest is  1 in groupid: 1
 The message we send is start_the_mission3_at10am with hashing
    1387958364
The Packet Sent successfully, the source is 0 The destination is 1
   at time 1

The packet recieved at dest node 1 from 0
at time 1.0036
dest is 1

Opened node1.txt for writing.
+   topsecret   ->   gid   1
+   topsecret   ->   gid   2
?   topsecret   ->   gid   3
+   secret   ->   gid   1
+   secret   ->   gid   2
-   secret   ->   gid   3
+   unclassified   ->   gid   1
+   unclassified   ->   gid   2
+   unclassified   ->   gid   3
```

```
newhash has: 1387958364
The encrypted data is: wxevxcxlicqmwwmsr7cex54eq
The original data after decr is: start_the_mission3_at10am
The hashvalue is 1387958364
Message_Accepted
send packet return
I am node 1 Thanks, the Packet received successfully from node 0
```

Now, assume that node 1 has these policy rules at node1.txt:

+ Top secret → GID 1
+ Top secret → GID 2
+ Top secret → GID 3
+ Secret → GID 1
+ Secret → GID 2
+ Secret → GID 3
+ Unclassified → GID 1
+ Unclassified → GID 2
+ Unclassified → GID 3

If node 1 at a later time 'wants' to send the same Top secret message (start the mission 3 at 10am) to node 3 by executing this command in Tcl as shown in Listing 7.15. Let us see whether the message is allowed to be sent to node 3 or not depends on the policy rules which node 1 has already received from node 0. According to the original policy rule of node 1, Top secret message can be sent to node 3 because node 3 is in group-id 3, however this does not agree with the policy rules of the originator of the message (node 0). The result of this simulation is shown in Listing 7.12.

Listing 7.15: Tcl Command for the case study (4): Part B

```
$ns at 2.0 "$u(1) 1 send topsecret start_the_mission3_at10am 3"
```

Listing 7.16: The result of the simulation for the case study 4: Part B

```
Opened groupid.txt for reading.
Opened groupid.txt for reading.
Node 0 relate to groupid : 1
Node 1 relate to groupid : 1
Node 2 relate to groupid : 2
Node 3 relate to groupid : 3
Node 4 relate to groupid : 1
Node 5 relate to groupid : 1

Opened node1.txt for reading.
+  topsecret -> gid 1
+  topsecret -> gid 2
+  topsecret -> gid 3
+  secret -> gid 1
+  secret -> gid 2
+  secret -> gid 3
+  unclassified -> gid 1
+  unclassified -> gid 2
+  unclassified -> gid 3
+  topsecret   ->   gid   1
+  topsecret   ->   gid   2
?  topsecret   ->   gid   3
+  secret   ->   gid   1
+  secret   ->   gid   2
-  secret   ->   gid   3
+  unclassified   ->   gid   1
+  unclassified   ->   gid   2
+  unclassified   ->   gid   3

The adjacent nodes in search adj function of node 1 is 0
The adjacent nodes in search adj function of node 1 is 3
```

```
The adjacent nodes in search adj function of node 1 is 4
The adjacent nodes in search adj function of node 1 is 5
---------------------1
The Msg has topsecret priority
Calling The policy Check Function To Decide To Where The Packet
   topsecret Can Be Routed To Reach The Target
**********************sending + Check policy-->>True
the dest is  0 in groupid: 1
org equal dest
---------------------2
The Msg has topsecret priority
Calling The policy Check Function To Decide To Where The Packet
   topsecret Can Be Routed To Reach The Target
**********************sending + Check policy-->>True
the dest is  3 in groupid: 3

ask the originator for its up-to-date policy
```

Whenever Node 1 'wants' to send the Top secret message (start the mission 3 at 10am) to any node inside the group-id 3, there will be a conflict, the conflict is between ask the originator and 'allow' sending the message of type Top secret to a node in group-id 3, therefore we implemented our work to make the originator's policy to be the dominant (as described in Section5.5.1) which, as a result to ask the originator for its up-to-date policy, and this can be seen from Listing 7.16.

7.6 Simulation Environment and Parameters

Most of the research in *ad hoc* networks has been evaluated just as has been implemented using the Network Simulator (NS-2) [4]. Similarly we used NS-2 simulator to evaluate our policy-based protocol to check whether the privacy and confidentiality requirements

of the originator are met. Because NS-2 is an object-oriented network simulator, with the back end of the simulator written in C++ to implement the protocols and to extend the NS-2 library, whereas the front end of NS-2 is written in Tcl (Tool Command Language) with the OTCL interpreter, it is simple to create and control the simulation environment, including the selection of output data.

In order to implement the policy-based agent protocol certain simulation scenario must be defined. The details of the simulation which has been done and the results of implementing the policy-based agent protocol are described in this chapter. The simulation was conducted under Microsoft Windows operating system (xp service pack 3) using Cygwin environment.

In this work we modified the Tcl script file for mobile *ad hoc* wireless as in Listing 7.17, which is the one provided by NS-2 to suit the simulation environment of the security mechanism. In this file, we defined multi types of networks in the topology and specified which nodes are related to a specific type (for example organised nodes into three different groups). A mobile node consists of network components and parameters such as radio propagation (TwoRayGround), Antenna type(OmniAntenna, Directional, Bi-directional), interface queue(Queue/DropTail/PriQueue), Link Layer (LL), MAC layer type (Mac/802.11) and the wireless channel through which nodes transmit and from which they receive signals. Additionally, we need to define other parameters such as type of *ad hoc* routing protocol used by mobile nodes, the number of nodes simulated, and dimension of the topography. In addition to that, in the TCL file we linked the agent class (defined in C++) that manages and enforces the policies attached to packets at every node in the network.

Listing 7.17: Example of parameters options

```
set val(chan) Channel/WirelessChannel
set val(prop) Propagation/TwoRayGround
set val(netif) Phy/WirelessPhy
```

```
set val(mac) Mac/802_11
set val(ifq) Queue/DropTail/PriQueue
set val(ll) LL
set val(ant) Antenna/OmniAntenna
set val(x) 4000 ;# X dimension of the topography
set val(y) 4000 ;# Y dimension of the topography
set val(ifqlen) 100 ;# max packet in ifq
set val(seed) 0.0
set val(adhocRouting) AODV
set val(nn) 40 ;# how many nodes are simulated
set val(stop) 1000.0 ;# simulation time
```

Every time after running the TCL file, a trace file is generated. In general, the total size of all the traces files generated while experimenting the policy-based approach is approximately 20 MB. Afterwards, for interest these files were analysed to find the average delay (to be described in Section 7.8). Therefore, to take out unwanted lines and discard the rest of the generated trace file we used the AWK utility at the analysis stage to filter the files content. AWK is a language used to process files of text. A file is considered as a sequence of records, and each line is considered as a record. Each line is divided into a sequence of fields, so the first word in a line is considered as the first field, the second word as the second field, and so on. The AWK program is consisting of a set of actions to be taken against textual data. AWK reads the trace file line by line. A line is scanned for each pattern in the program, and for each pattern that matches, the associated action is executed [135]. An AWK program is a series of pattern action pairs, written as Condition Action.

Currently, there is no a specific scenario (benchmark) to test a protocol created by NS-2 in both MANETs and VANETs [4]. The research community in *ad hoc* networks needs a way to standardise some simulation scenarios to evaluate/compare the performance of the protocols, and to ensure that these protocols are accurately tested. In order to generate

results are near to the real world scenarios, simulations were run with various parameter values as shown in Table 7.1: number of nodes varying from 20 to 80, and node speed varying from 0 to 80 km/s. These nodes form the VANETs, moving about over an area of 4000m * 4000m (the length and width of the topology) for 1000 seconds of simulated time. A 4000m space was chosen because it is four times the transmission range of 1000 metres, which allows the possibility of a reasonable number of nodes between the source and destination nodes.

Number of nodes	20, 40, 60, 80
Network area	4000m*4000m
Radio range	1000 m
Speed	0 ,10, 20, 30, 40, 50, 60, 70, 80
Total simulation time	1000s
Antenna model	Omni Antenna
MAC Protocol	IEEE 802.11

Table 7.1: Simulation Parameters

7.7 Simulation Results

This section will show the results of implementing the policy-based agent protocol in the system. NS-2 simulations have been carried out to evaluate the performance of the proposed protocol in the predefined scenario. The parameters used for simulation are shown in Table 7.1.

We simulated our policy-based agent protocol with a variable number of UDP (User Datagram Protocol) agents simultaneously to check what happen if all agents are started in the simulation and how the time necessary for a packet to be transmitted across a network from source to destination will be affected. In Figure 7.6 we measured the delay time versus number of CBR (Constant Bit Rate) traffics which are depicted on the y-axis and x-axis respectively. The result of this figure shows that as the number of CBR traffic

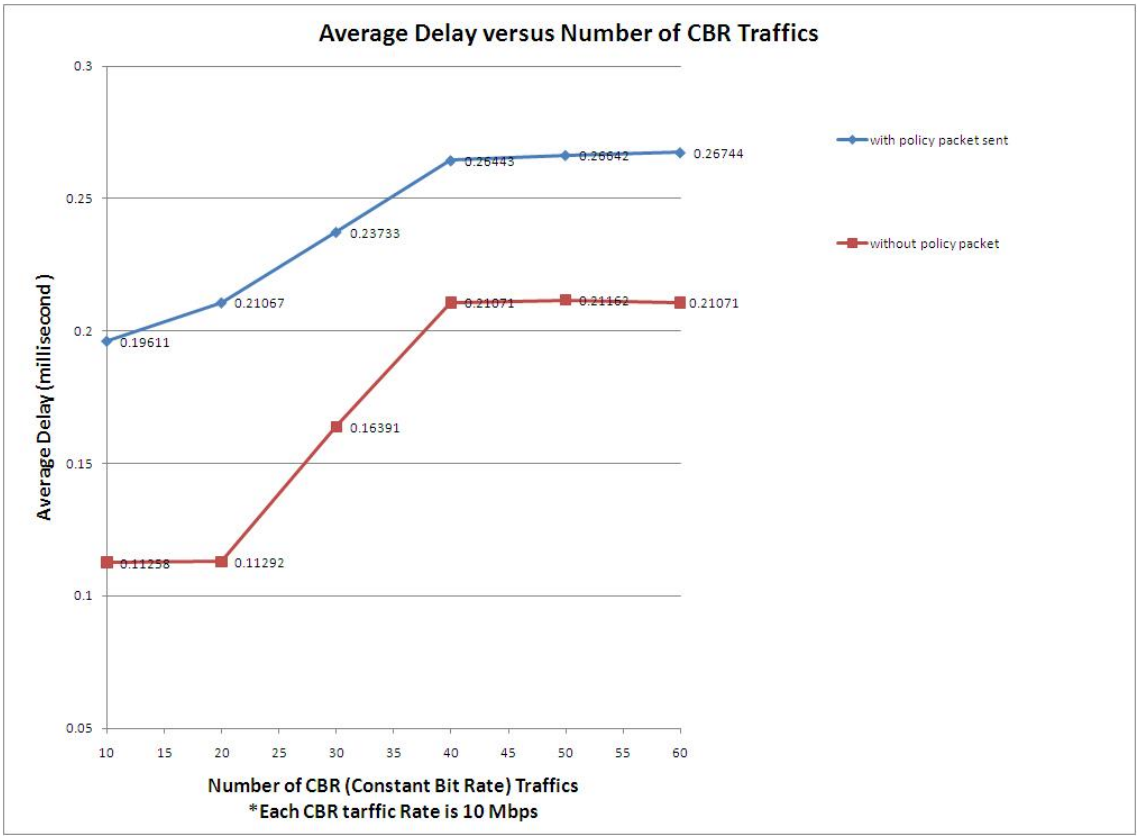


Figure 7.6: Average Delay versus Number of CBR Traffics

increases, the delay time of both agents increases. We started with 10 CBR traffic, 20, 30, 40, 50 and 60 with and without our policy-based agent to be started at different sources and destinations to measure the average of the delay time between them.

In this set of experiments we varied the number of CBR traffics, started from 10, 20, 30, 40, 50 and 60 but we fixed the rate of the CBR packet size to 10 Mbps to determine the effect of the traffic on the policy-based agent protocol. As we increased the number of CBR traffics in the network, the average delay increased from 0.11258 to 0.21071 milliseconds without the policy added to packets, whereas with the policy added to packets the average delay increased from 0.19611 to 0.26744 milliseconds: that is 0.08353 change on the average delay when the number of CBR traffics was set to 10, and also 0.05673 change on the average delay when the number of CBR traffics was set to 60. This can be plainly seen from Figure 7.6.

In conclusion, even though the average delay of the policy-based agent protocol in the experiments that include more CBR traffics is higher than the delay in the experiments that includes the same number of CBR traffics without the policy added to the protocol, the increase is still very small (less than 0.08 and 0.07 milliseconds) with variable number of CBR traffics. Hence, multi CBR traffic does not significantly cause the delivery of the policy-based agent protocol packets to be delayed: it is clear from Figure 7.6 that there is a delay introduced by sending additional information that contains the policies, but not to a degree that would yield the system unusable.

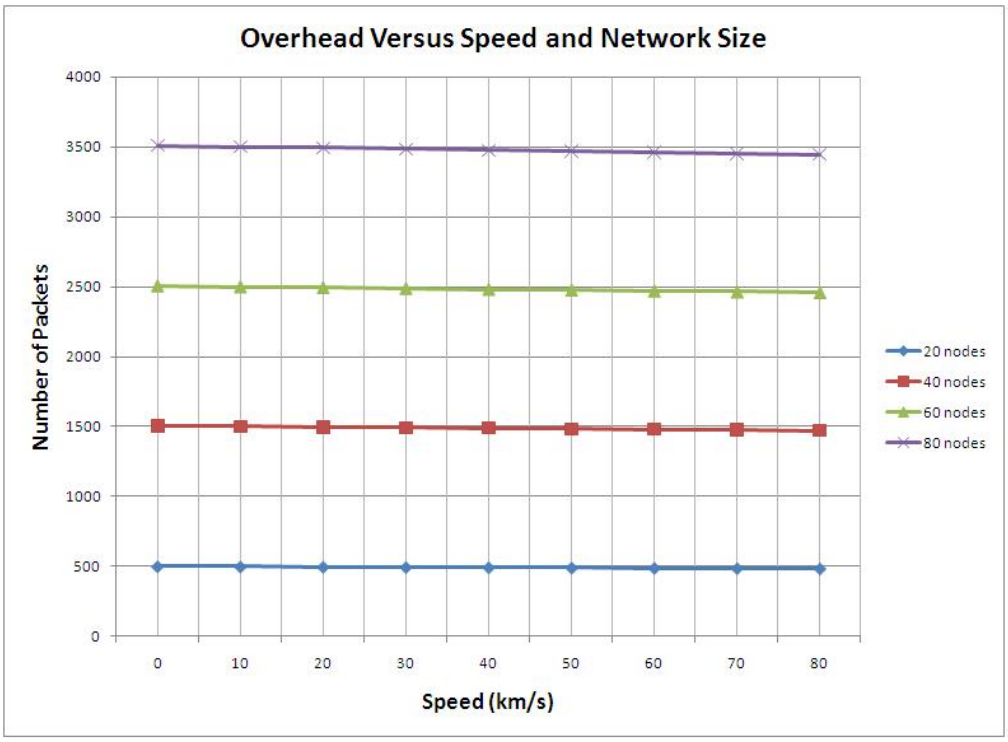


Figure 7.7: Overhead Versus Speed and Network Size

Similarly to average delay, overhead is considered an essential to any network system. Overhead consists of the number of packets generated by this policy-based agent protocol, of which there are two types in this work: packets, and return packets. The overhead is calculated versus speed of nodes and network size. As the node speed increases, the overhead remains almost unchanged if any thing slightly decreasing when the speed in-

creases. This is shown in Figure 7.7 which illustrates the increase in overhead caused by increasing network size.

7.8 Result and Discussion

In this work we used the Network Simulator (NS-2) which is a real network environment simulator to evaluate the policy-based framework. The results of the evaluation through the case studies (described in Section 7.2, 7.3, 7.4, and 7.5) supported the research that is presented in this thesis, they showed that only intended nodes can receive the packet which has been sent by the source based on the originator message requirement. We simulated multi variable number of nodes where the originator node 'wants' to disseminate the packet to a specific group(s) of nodes. The results of these case studies showed the feasibility of our policy-based framework to control the data dissemination between nodes in VANETs and demonstrated how it works. In addition of these results (explained in Section 7.2, 7.3, 7.4, and 7.5), our result from the tracing file and the NAM (Network Animator) also showed that only nodes in the permitted group can receive the packet because of the restriction which has issued from the originator nodes which represented as policy rules (described in Chapter 5) attached to packets at every node in the VANET to solve the research question.

In this work, a policy-based framework was described that addresses the problem of secure data dissemination in VANETs by automatically attaching policies along with messages to specify how the information can be used by the receiver, so as to prevent disclosure of the messages other than consistent with the requirements of the originator. Section 5.2 described these requirements as a set of policy rules (described in Section 5.3) that explicitly instructs recipients how the information contained in messages must be disseminated to other nodes.

In this work, we implemented a new agent protocol and a new packet structure to suit

this protocol in NS-2 (respectively described in Sections 6.3 and 6.4). The new policy-based protocol agent was derived from an existing class in NS-2 by adding encryption and decryption algorithms to secure the data field in the packet. Furthermore, a message digest generation function (hash function) was also implemented to ensure the integrity requirement of data.

We assumed that all nodes in the system are trusted to enforce the policy attached within the packet, and the encryption, digital signature, and the keys management have already been done securely. So in this work we address the stage after the processes mentioned above. So implicitly that means if a node is trusted to receive a certain information (in the clear) our framework assumes that it also will be trusted to protect this information. In the domains discussed above this appears to be a reasonable assumption that however does not protect against malicious nodes in the network that have infiltrated the system and (wrongly) gained the trust of message originators. Similarly the system does not prevent out-of-band communications and assumes the data is communicated using the provided infrastructure. The protection here is that automated services that provide e.g. situational awareness are trust-enabled to limit the dissemination of information.

In this work, we presented a novel policy-based framework to control the dissemination of data communicated between nodes in VANETs by attaching originator policies to messages as they are sent (published in [15, 16, 17]). Our framework differs from previous approaches (described in Section 3.6) since it takes into consideration the originator data dissemination requirements which is attached as a set of policy rules along with messages to ensure message confidentiality is maintained not only during transmission to the intended node(s), but to keep the message contents private to an originator-defined subset of nodes in the VANET, thus preventing the destination node from forwarding the message to unwanted recipients.

In this work, we highlighted the special considerations for security in *ad hoc* networks and provided an extensive overview of related work and the state of the art in this area. To

our knowledge, none of the related work addressed the issue of controlling the information flow in VANETs. We presented a scenario drawn from the military domain (described in Section 4.2), where the impact of confidentiality breach is evident and a real risk. We provided a framework that addresses this problem by automatically attaching policies to the messages that identify how the information can be used by the receiver, thus limiting the relay of messages based on the originators' data dissemination requirements. We currently assume that all nodes in our system are trusted to correctly enforce the policies that are attached to the message and provide a communication system that includes a policy processing layer dealing with the merging of existing and received policies. However, the assumption that all are trusted is strong. In future work (to be described in Section 8.3) we shall relax this assumption by providing traceability, *viz.* water-marking messages in such a way that they are identified as compromised, they will then lead to a dynamic adjustment of the originator policy.

7.9 Summary

The chapter presented four case studies to show how data dissemination can be controlled based on the policy of the originator. These case studies (examples) have been provided to show how the policy-based framework components interact together to control the data dissemination between nodes within the NS-2 Simulation. The results of these case studies showed the feasibility of our policy-based framework to control the data dissemination between nodes in VANETs and demonstrated how it works.

The chapter presented four case studies that describe how information can be disseminated and controlled based on the originator data dissemination policy rules, it also examined the result of applying the originator message requirements as policy rules (described in Chapter 5) attached to packets at every node in the VANET to solve the research question; this evaluation was done by creating different network topologies using Tcl (Tool

Command Language) in NS-2 to represent/simulate some nodes communicating between themselves, to check whether the policy-based agent protocol achieves the originator goal to keep message contents private to a originator-defined subset of nodes in the VANET or not.

As a result, we have answered the main research question that says "how can we prevent information from being leaked to undesirable entity(ies)?" and at the same time, we answered the following sub research questions: "how to keep message contents private to an originator-defined subset of nodes in the VANET", "how to control the dissemination of messages while the nodes are communicating between each other in the network", "how to enforce the privacy and confidentiality requirement of the originator", "how to represent the originator data dissemination requirements as a set of rules" and "how to accommodate changes of the security requirement to the related information by referring back to the originator". Based on the evaluation and case studies all research questions have been answered. We have proofed our research contributions including: a policy-based framework to control the data dissemination in VANETs, a data dissemination policy language that specifies and supports the originator data dissemination requirements to be considered in the network to keep the message secure, and an interaction mechanism to query the originator for its up-to-date policy.

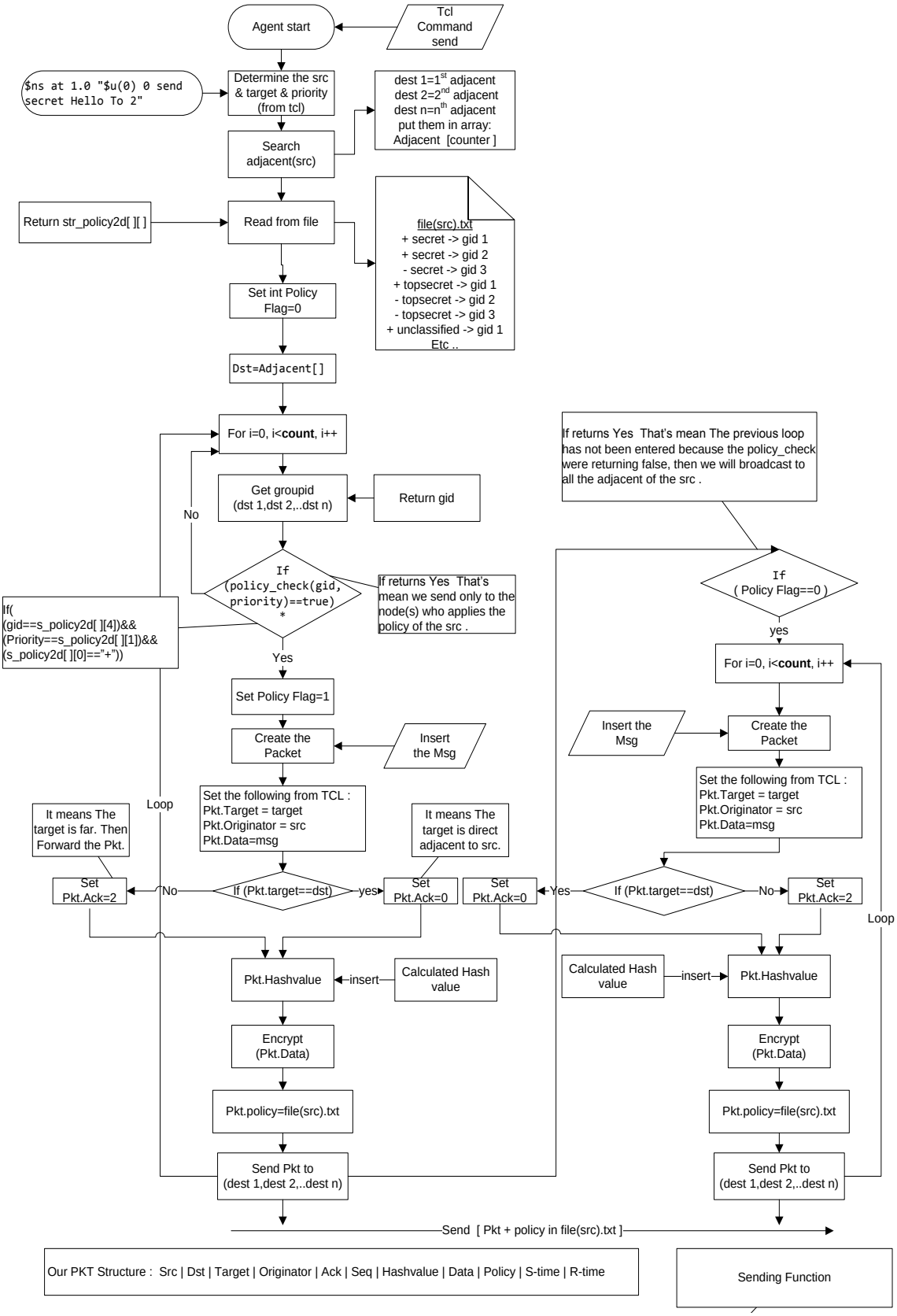


Figure 7.8: Algorithm two: Sending Chart Algorithm and packet structure

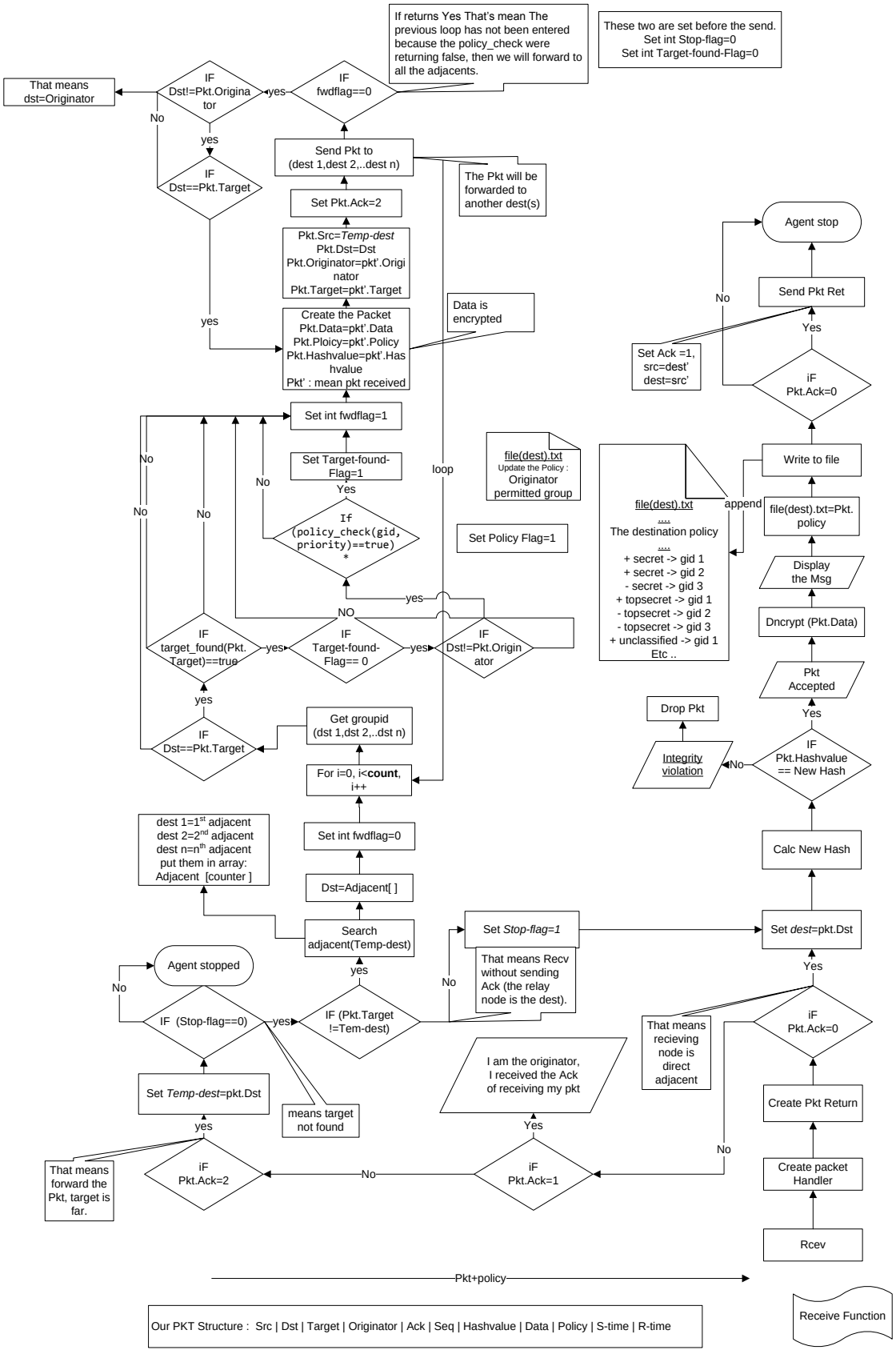


Figure 7.9: Algorithm two: Receiving and forward Chart Algorithm

Chapter 8

Conclusion and Future Work

Objectives:

- Summary of this research.
- Highlight the original contributions to knowledge.
- General overview of future research.

8.1 Summary

This thesis provided a review of the issue of security in VANET and MANET, and also surveyed existing solutions. It highlighted a particular area not previously addressed: controlling the information flow in VANETs. This thesis therefore aimed to provide a policy-based framework (described in Chapter 4) to control the dissemination of data communicated between nodes, in order to ensure that data remains confidential not only during transmission but also after having been communicated to other nodes, and to keep the message contents private to an originator-defined subset of nodes in VANETs. This

is done by automatically attaching policies along with messages to specify how the information can be used by the receiver, so as to prevent disclosure of the messages other than consistent with the requirements of the originator. Section 5.2 described these requirements as a set of policy rules (described in Section 5.3) that explicitly instructs recipients how the information contained in messages can be disseminated to other nodes.

This thesis presented both a controlling data dissemination mechanism for VANETs using policy-based framework to ensure that information is not disclosed to unwanted nodes, together with an interaction mechanism by referring back to the originator for up-to-date policy. A data dissemination policy language was developed (described in Chapter 5) that supports and expresses the policy-based framework.

Our data dissemination policy differed from the policy languages exist such as Authorisation Specification Language (ASL), XACML (eXtensible Access Control Markup Language), and Ponder (described in Section 5.4), since our language took into consideration the originator high-level requirements as low-level policy rules whose enforcement can be fully automated and understood for the framework (described in Section 4.4) in order to solve the research question.

Whereas, those policy languages exist cannot enforce any control on the information flow once this information has been received by a node. Hence, these policy languages concentrate on controlling the access at specific resources located on central or distributed nodes, they are not intended however to control the data dissemination between nodes.

Our data dissemination policy language had various advantages over other policy languages to control the information flow, as it showed that our policy languages are expressive enough to ensure and implement the message originator requirements and to distribute enforcement policies in the network efficiently (described in Chapter 7.8). Finally, it is compatible with the Network Simulator (NS-2) which makes it to be understandable for the policy agent. Since it is considered as a low-level policy language which makes it perfectly run by C++ programming language supported in our NS-2 policy-based agent

protocol (described in Section 6.2).

The thesis then presents a novel framework for controlling data dissemination in VANETs in which privacy policies are attached to messages as they are sent between nodes. Our framework differs from previous approaches (described in Section 3.6) since it takes into consideration the originator confidentiality requirements which is attached as a set of policy rules along with messages to ensure message confidentiality is maintained. The framework was evaluated using the Network Simulator (NS-2) to provide and check whether the privacy and confidentiality of the originator were met. NS-2 agent (described in Chapter 6) has been implemented to manage and enforce the policies attached to packets at every node in the VANET.

8.2 Contributions

In this thesis, we reviewed the main security issues and existing solutions in VANETs, in particular the area of security of VANET which had not been previously researched. We addressed the dissemination control problem in VANETs in order to ensure the originator's data dissemination requirements. The main contributions to knowledge in this thesis are summarised as follows:

- **A policy-based framework to control the data dissemination in VANETs**. We devised a framework to provide a prevention component which governs the privacy of an originator's messages, so that they cannot be disclosed to unwanted parties. This work was accomplished by using policies of the originators to control the access to their messages, and to ensure that these polices will be enforced upon intended recipients. This framework was described in Chapter 4 and has been published in [15, 16, 17].

- **A data dissemination policy language that supports and expresses the policy-based framework**. It specifies the data dissemination security actions to be consid-

ered in the network to keep the message secure. It is represented as a set of policy rules that declares the data dissemination requirement based on the originator of the message. The data dissemination policy works as the reference that controls the flow of the messages while the nodes are communicating between each other in the network. The data dissemination policy should specify high-level requirements into low-level policy rules whose enforcement can be fully automated and understood for the framework. In this work we provided a suitable data dissemination policy to be used for the framework in which the originator of the message retains control over its dissemination. The framework and the data dissemination policy language been used are introduced in Chapter 4 and Chapter 5 respectively and have been published in [15, 16, 17].

- **An interaction mechanism by returning to the originator for its up-to-date policy**. In addition to presenting a controlling dissemination mechanism in VANETs by the use of policy-based framework to ensure that information is not disclosed to unwanted parties, we also presented an interaction mechanism by returning to the originator for its up-to-date policy, since changing policies is an important requirement in policy based systems. This interaction mechanism is described in Chapter 5.

- **Evaluation of the framework using the Network Simulator (NS-2) through some case studies to check whether the privacy and confidentiality of the originator are met**. We used NS-2 agent to implement our policy-based framework together with policy rules attached to packets at every node in the VANET; we built a new agent protocol and a new packet structure to suit this protocol in NS-2. The new policy-based agent protocol is derived from an existing class in NS-2. The implementation and evaluation are introduced in Chapter 6 and Chapter 7 respectively and have been published in [15, 16].

8.3 Future Work

Trust is an important component in a security system; therefore mechanisms must be used to prevent any possible message compromise on VANETs, in order to satisfy a set of security requirements: privacy, authentication, integrity, availability, and non-repudiation.

Typically, the sharing of information is done on some notion of trust and if that trust is broken then it is important to know who leaked the information so as to know whom to trust in the future. This thesis presented a prevention mechanism for VANETs by the use of policy-based framework to ensure that information is not disclosed to unwanted parties, however it does not provide a detection mechanism should the information be leaked by a member(s) of the communicated parties.

A data tracing mechanism that enables the detection of trust-breaches by member(s) of this subset would seem to be therefore essential. This detection component will be used to differentiate between normal and malicious entities. This component will be used by the originator if the message is leaked. In this way the originator will be able to detect the untrustworthy entity in VANETs.

One such detection mechanism could be based on Boneh and Shaw codes [136]; the main objective of this mechanism is to explain how a copy of a message(M) leaked can be effectively traced back to the vehicle (or vehicles) who disclosed that copy. The process of data tracing includes two types of techniques: The first technique (fingerprinting) for generating secure codes with each copy of a message sent can be uniquely fingerprinted and the second technique (watermarking) is to embed these fingerprints into the messages such that they are difficult to perceive.

We recommend any future research in this scope should consider techniques to trace messages by using watermarking and fingerprinting. Fingerprinting mechanism is used to create multiple unique copies of the same message which can be identified with the help of the unique code (binary code) embedded in each message. The embedding of these

unique codes is done by using the watermarking mechanism [137].

In future work, each vehicle in VANETs should get a unique copy of the message which makes it possible to trace back a copy of the message leaked to a particular vehicle. After each message has been embedded with a unique fingerprint, tracing the vehicle which leaked the message should be simple by decoding the fingerprint on the revealed copy. It is common, however, for the untrustworthy entities to make some changes to the messages to such a degree that either the fingerprints are removed or changed to make the copy of a message untraceable. Kankanhalli and Hau in [138] indicated that the untrustworthy entities generally change the messages to create a similar copy by changing a part of the text, and manage to catch various legal copies, compare them and overwrite on the original fingerprint in order to make it impossible to trace.

User collusion makes data tracing a challenging issue. Collusion can take place as in Figure 8.1 when two nodes of Country 2 with different fingerprinted copies compare their copies. This comparison can show the differences between two or more similar copies and hence showing the fingerprints, which can be modified to hide their identities or to frame other nodes in the network [139]. This also needs addressing in future work.

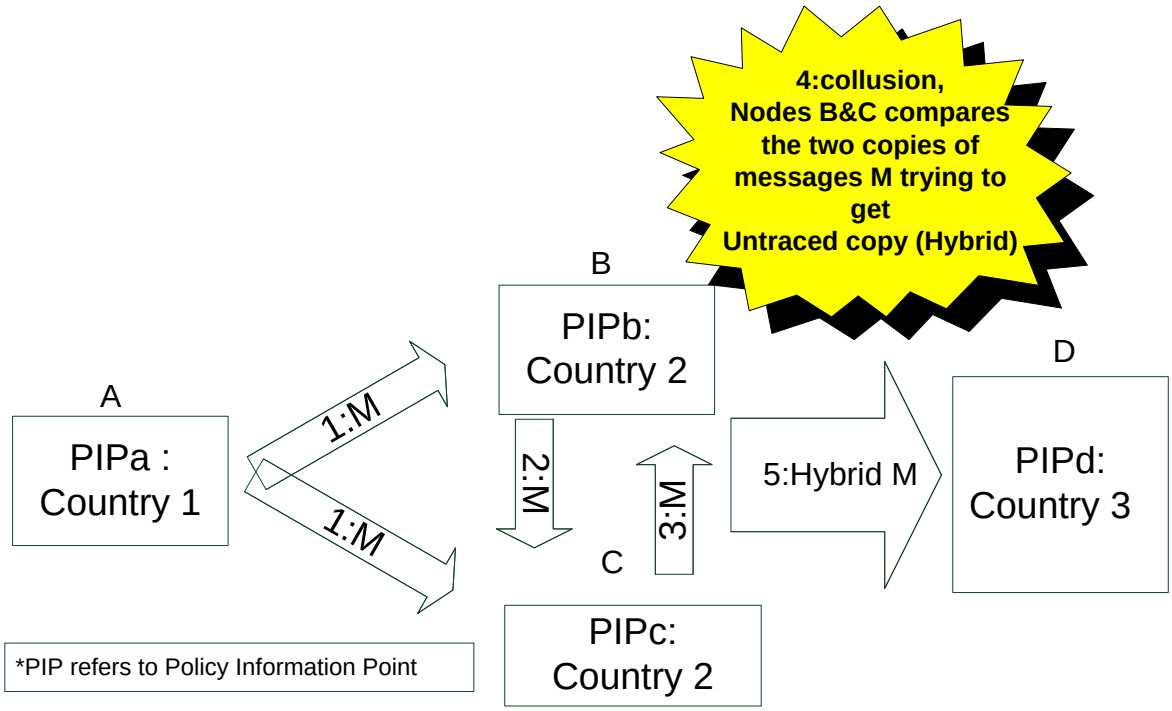


Figure 8.1: Collusion between B and C

Bibliography

[1] S. Olariu and M.C. Weigle. *Vehicular Networks from Theory to Practice*. Chapman & Hall, USA, 2009. (document), 2.4, 2.4.1, 2.4.3, 3.6.1

[2] D. Jiang, V. Taliwal, A. Meier, W. Holfelder, and R. Herrtwich. Design of 5.9 GHz DSRC-based vehicular safety communication. *Wireless Communications, IEEE*, 13(5):36–43, 2006. (document), 2.4.2.3, 2.5

[3] William Stallings. *Cryptography and Network Security: Principles and Practice*. Pearson Education, 4rd edition, USA, 2005. (document), 2.4.4.4, 3.2, 3.3, 3.4, 3.1, 3.5, 3.5.1, 3.2, 3.5.2, 3.3, 3.4, 3.5.3

[4] Stuart Kurkowski, Tracy Camp, and Michael Colagrosso. Manet simulation studies: the incredibles. *ACM SIGMOBILE Mobile Computing and Communications Review*, 9:50–61, 2005. (document), 1.6, 6.1, 6.2, 6.1, 7.6, 7.6

[5] John Oates. Toyota and microsoft ink e-car deal in a cloud of telematics. http://www.theregister.co.uk/2011/04/07/microsoft_toyota/, 7th April 2011. Accessed July 8, 2011. 1.1, 2.4.1, 3.6

[6] Maxim Raya and Jean-Pierre Hubaux. The security of vehicular ad hoc networks. In Kenneth P. Laberteaux, Hannes Hartenstein, David B. Johnson, and Raja Sengupta, editors, *Vehicular Ad Hoc Networks*, pages 93–94. ACM, 2005. 1.2, 3.6.2

[7] M. Raya and J.P. Hubaux. Securing vehicular ad hoc networks. *Journal of Computer Security*, 15(1):39–68, 2007. 1.2, 3.6.2, 4.5

[8] Klaus Ploessl, Thomas Nowey, and Christian Mletzko. Towards a security architecture for vehicular ad hoc networks. In *ARES*, pages 374–381. IEEE Computer Society, 2006. 1.2

[9] Matt Blaze, Joan Feigenbaum, and Jack Lacy. Decentralized trust management. In *Proceedings of the 1996 IEEE Symposium on Security and Privacy*, pages 164–173. IEEE Computer Society Press, 1996. 1.2, 3.6.2

[10] Krishna Sampigethaya, Mingyan Li, Leping Huang, and Radha Poovendran. Amoeba: Robust location privacy scheme for vanet. *IEEE Journal on Selected Areas in Communications*, 25(8):1569–1589, 2007. 1.2

[11] Krishna Sampigethaya, Leping Huang, Mingyan Li, Radha Poovendran, Kanta Matsuura, and Kaoru Sezaki. Caravan: Providing location privacy for vanet. In *in Embedded Security in Cars (ESCAR)*. Citeseer. 1.2, 4.5

[12] F. Dotzer. Privacy issues in vehicular ad hoc networks. In *Privacy Enhancing Technologies*, pages 197–209. Springer, 2006. 1.2

[13] M. Gerlach and F. Guttler. Privacy in VANETs using changing pseudonyms-ideal and real. In *Vehicular Technology Conference, 2007. VTC2007-Spring. IEEE 65th*, pages 2521–2525. IEEE, Dublin, 2007. 1.2

[14] S. Pearson and M. Casassa-Mont. Sticky policies: An approach for privacy management across multiple parties. *Computer Society*, (99):60–68, 2011. 1.4, 3.6.2

[15] H. Aldabbas, H. Janicke, R. AbuJassar, and T. Alwada'n. Ensuring data confidentiality and privacy in mobile ad hoc networks. pages 490–499. Springer, 2012. 1.4, 4.1, 7.8, 8.2

[16] H. Aldabbas, T. Alwada'n, H. Janicke, and A. Al-Bayatti. Data confidentiality in mobile ad hoc networks. *International Journal of Wireless and Mobile Networks (IJWMN), 4, (1), pp. 225-236*, 2012. 1.4, 2.4.4.3, 4.1, 7.8, 8.2

[17] H. Janicke, M. Sarrab, and H. Aldabbas. Controlling data dissemination. pages 303–309. Springer, 2012. 1.4, 1.6, 2.4.4.3, 4.1, 5.1, 7.8, 8.2

[18] E.B. Wilson. *An introduction to scientific research*. Dover Publications, USA, 1991. 1.5

[19] Mohamed Sarrab. *Policy-Based Runtime Verification of Information Flow*. PhD thesis, De Montfort University, March 2011. 1.6, 5.1

[20] Subir Kumar Sarkar, T. G. Basavaraju, and C. Puttamadappa. *Ad Hoc Mobile Wireless Networks: Principles, Protocols and Applications*. Auerbach Publications, Boston, MA, USA, 2007. 2.3, 2.3.4

[21] Jameela Al-Jaroodi. Security issues at the network layer in wireless mobile ad hoc networks at the network layer. Technical report, Faculty of Computer Science and Engineering, University of Nebraska-Lincoln, Nebraska, USA, 2002. 2.3.1, 3.2

[22] C. Siva Ram Murthy and B.S. Manoj. *Ad Hoc Wireless Networks: Architectures and Protocols*. Prentice Hall PTR, Upper Saddle River, NJ, USA, 2004. 2.3.1, 2.3.4, 3.3

[23] C.K. Toh. Maximum battery life routing to support ubiquitous mobile computing in wireless ad hoc networks. *IEEE communications Magazine*, 39(6):138–147, 2001. 2.3.1

[24] R. Chadha and L. Kant. *Policy-driven mobile ad hoc network management*. Wiley-IEEE Press, USA, 2007. 2.3.1

[25] M. Carvalho. Security in mobile ad hoc networks. *Security Privacy, IEEE*, 6(2):72 –75, 2008. 2.3.1, 3.6, 3.6.2

[26] W. Li and A. Joshi. Security issues in mobile ad hoc networks-a survey. *White House Papers Graduate Research In Informatics at Sussex*, 17:1–23, 2004. 2.3.2, 3.2, 3.6

[27] Panagiotis Papadimitratos and Zygmunt J. Haas. *Securing mobile ad hoc networks*, pages 551–567. CRC Press, Inc., Boca Raton, FL, USA, 2003. 2.3.2, 3.6

[28] A. Mishra and K.M. Nadkarni. Security in wireless ad hoc networks. In *The handbook of ad hoc wireless networks*, pages 499–549. CRC Press, Inc., 2003. 2.3.2, 3.6

[29] Y. Zhang and W. Lee. Security in mobile ad-hoc networks. *Ad Hoc Networks*, pages 249–268, 2005. 2.3.2, 2.4.4.4, 3.6

[30] A. Bharathidasan and V.A.S. Ponduru. Sensor networks: An overview. Technical report, Department of Computer Science, University of California, Davis, USA, 2002. 2.3.4

[31] S. Yousefi, S. Bastani, and M. Fathy. On the performance of safety message dissemination in vehicular ad hoc networks. In *IEEE: Proceedings of 4th European Conference on Universal Multiservice Networks ECUMNŠ2007*, pages 377–390, 2007. 2.3.4

[32] Xavier Carcelle, Tuan Dangand, and Catherine Devic. *Ad-Hoc Networking*, volume 212/2006 of *IFIP International Federation for Information Processing*, chapter Wireless Networks in industrial environments: State of the art and Issues, pages 141–156. Springer, Santiago, Chile, 2006. 2.3.4

[33] J. Yick, B. Mukherjee, and D. Ghosal. Wireless sensor network survey. *Computer Networks*, 52(12):2292–2330, 2008. 2.3.4

[34] W. Lehr and L.W. McKnight. Wireless Internet access: 3G vs. WiFi. *Telecommunications Policy, Research Program on Internet and Telecoms Convergence, Massachusetts Institute of Technology (MIT)*, 27(5-6):351–370, 2003. 2.4.2.1

[35] M. Gast. *802.11 wireless networks: the definitive guide*. O'Reilly Media, USA, 2005. 2.4.2.1

[36] Jeffrey G. Andrews, Arunabha Ghosh, and Rias Muhamed. *Fundamentals of WiMAX: Understanding Broadband Wireless Networking (Prentice Hall Communications Engineering and Emerging Technologies Series)*. Prentice Hall PTR, Upper Saddle River, NJ, USA, 2007. 2.4.2.2

[37] A. Ghosh, D.R. Wolter, J.G. Andrews, and R. Chen. Broadband wireless access with wimax/802.16: current performance benchmarks and future potential. *Communications Magazine, IEEE*, 43(2):129–136, 2005. 2.4.2.2

[38] S. Biswas, R. Tatchikou, and F. Dion. Vehicle-to-vehicle wireless communication protocols for enhancing highway traffic safety. *Communications Magazine, IEEE*, 44(1):74–82, 2006. 2.4.2.3

[39] F. Bai and H. Krishnan. Reliability analysis of DSRC wireless communication for vehicle safety applications. In *Intelligent Transportation Systems Conference, 2006. ITSC'06. IEEE*, pages 355–362. IEEE, 2006. 2.4.2.3

[40] IEEE P1609.1, Trial-Use Standard for Wireless Access in Vehicular Environments (WAVE)-Resource Manager, 2006. 2.4.2.3

[41] IEEE P1609.2, Trial-Use Standard for Wireless Access in Vehicular Environments (WAVE)-Security Services for Applications and Management Messages, 2006. 2.4.2.3

[42] IEEE P1609.4, Trial-Use Standard for Wireless Access in Vehicular Environments (WAVE) Multi-Channel Operation , 2006. 2.4.2.3

[43] IEEE P1609.3, Trial-Use Standard for Wireless Access in Vehicular Environments (WAVE)-Networking Services, 2007. 2.4.2.3

[44] Sudip Misra, Isaac Woungang, and Subhas Chandra Misra. *Guide to Wireless Ad Hoc Networks*. Springer-Verlag New York Inc, 2009. 2.4.3

[45] S. Manui and M. Kakkasageri. Issues in mobile ad hoc networks for vehicular communication. *IETE (INSTITUTE OF ELECTRONICS & TELECOMMUNICATION) Technical Review*, 25(2):59, 2008. 2.4.3

[46] F. Li and Y. Wang. Routing in vehicular ad hoc networks: A survey. *Vehicular Technology Magazine, IEEE*, 2(2):12–22, 2008. 2.4.3, 3.6

[47] N. Wisitpongphan, F. Bai, P. Mudalige, and O.K. Tonguz. On the routing problem in disconnected vehicular ad-hoc networks. In *INFOCOM 2007. 26th IEEE International Conference on Computer Communications*, pages 2291–2295. IEEE, 2007. 2.4.3, 3.6

[48] M. Fiore, J. Harri, F. Filali, and C. Bonnet. Vehicular mobility simulation for VANETs. In *Simulation Symposium, 2007. ANSS'07. 40th Annual*, pages 301–309. IEEE, 2007. 2.4.3, 3.6

[49] Moritz Killat, Felix Schmidt-Eisenlohr, Hannes Hartenstein, Christian Rossel, Peter Vortisch, Silja Assenmacher, and Fritz Busch. Enabling efficient and accurate large-scale simulations of vanets for vehicular traffic management. In *Proceedings*

of the fourth ACM international workshop on Vehicular ad hoc networks, VANET '07, pages 29–38, New York, NY, USA, 2007. ACM. 2.4.3, 3.6

[50] Saleh Yousefi, Mahmoud Siadat Mousavi, and Mahmood Fathy. Vehicular ad hoc networks (vanets): Challenges and perspectives. In *2006 6th International Conference on ITS Telecommunications Proceedings*, pages 761 –766, june 2006. 2.4.4.1

[51] S. Katragadda, CNS Ganesh Murthy, R. Rao, S. Mohan Kumar, and R. Sachin. A decentralized location-based channel access protocol for inter-vehicle communication. In *Vehicular Technology Conference, 2003. VTC 2003-Spring. The 57th IEEE Semiannual*, volume 3, pages 1831–1835. IEEE, 2003. 2.4.4.1

[52] K. Sjoberg, E. Uhlemann, and E.G. Strom. How severe is the hidden terminal problem in vanets when using csma and stdma. In *Vehicular Technology Conference (VTC Fall), 2011 IEEE*, pages 1–5. IEEE, 2011. 2.4.4.1

[53] Flaminio Borgonovo, Antonio Capone, Matteo Cesana, and Luigi Fratta. Adhoc mac: New mac architecture for ad hoc networks providing efficient and reliable point-to-point and broadcast services. *Wireless Networks*, 10(4):359–366, 2004. 2.4.4.1

[54] S. Eichler. Performance evaluation of the IEEE 802.11 p WAVE communication standard. In *2007 IEEE 66th Vehicular Technology Conference, 2007. VTC-2007 Fall*, pages 2199–2203, 2007. 2.4.4.1

[55] M. Bechler and L. Wolf. Mobility management for vehicular ad hoc networks. In *Vehicular Technology Conference, 2005. VTC 2005-Spring. 2005 IEEE 61st*, volume 4, pages 2294–2298. IEEE, 2005. 2.4.4.2

[56] Jong Min Lee 0001, Myoungju Yu, Young Hun Yoo, and Seong Gon Choi. A new scheme of global mobility management for inter-vanets handover of vehicles in

v2v/v2i network environments. In Jinhwa Kim, Dursun Delen, Jinsoo Park, Franz Ko, and Yun Ji Na, editors, *Networked Computing and Advanced Information Management*, pages 114–119. IEEE Computer Society, 2008. 2.4.4.2

[57] T. Nadeem, P. Shankar, and L. Iftode. A comparative study of data dissemination models for vanets. In *2006. 3rd Annual International Conference on Mobile and Ubiquitous Systems-Workshops*, pages 1–10. IEEE, 2006. 2.4.4.3

[58] Technical Overview Sun Microsystems. Trusted solaristm 8 operating environment. http://www.sun.com/solaris/. Accessed December 12, 2011. 2.4.4.3

[59] A.J. Menezes, P.C. Van Oorschot, and S.A. Vanstone. *Handbook of applied cryptography*. CRC, 1997. 3.2, 3.5, 3.5.1, 3.5.2

[60] F. Xing and W. Wang. Understanding Dynamic Denial of Service Attacks in Mobile Ad Hoc Networks. In *Military Communications Conference, 2006. MILCOM 2006. IEEE*, pages 1–7. IEEE, 2007. 3.2

[61] S. Harris. *CISSP all-in-one exam guide*. McGraw-Hill Osborne Media, 2007. 3.4.1

[62] Ravi Sandhu and Qamar Munawer. How to do discretionary access control using roles. In *Proceedings of the third ACM workshop on Role-based access control*, RBAC '98, pages 47–54, New York, NY, USA, 1998. ACM. 1, 2

[63] NSA Peter Loscocco. Integrating Flexible Support for Security Policies into the Linux Operating System. In *Proceedings of the FREENIX Track: 2001 USENIX Annual Technical Conference, June 25-30, 2001, Boston, Massachusetts, USA*, page 29. USENIX Association, 2001. 2

[64] D.F. Ferraiolo, R. Sandhu, S. Gavrila, D.R. Kuhn, and R. Chandramouli. Proposed NIST standard for role-based access control. *ACM Transactions on Information and System Security (TISSEC)*, 4(3):224–274, 2001. 3

[65] R.J. Robles and M.K. Choi. Symmetric-key encryption for wireless internet scada. *Security Technology*, pages 289–297, 2009. 3.5.1

[66] Types of symmetric algorithms. http://www.encryptionanddecryption.com/algorithms/symmetric_algorithms.html. Accessed January 27, 2012. 3.5.1

[67] W. Diffie and M. Hellman. New directions in cryptography. *Information Theory, IEEE Transactions*, 22(6):644–654, 1976. 3.5.2

[68] R. Hunt. Pki and digital certification infrastructure. In *2001. Proceedings. Ninth IEEE International Conference on Networks*, pages 234 – 239, oct. 2001. 3.5.2

[69] E. Palomar, J.M.E. Tapiador, J.C. Hernández-Castro, and A. Ribagorda. 17 cooperative security in peer-to-peer and mobile ad hoc networks. *Cooperative Wireless Communications ed by Yan Zhang, Hsiao-Hwa Chen, Mohsen Guizani*, page 391, 2009. 3.5.2

[70] R. L. Rivest, A. Shamir, and L. Adleman. A method for obtaining digital signatures and public-key cryptosystems. *Commun. ACM*, 21:120–126, February 1978. 3.5.2

[71] R.A. Mollin. *An introduction to cryptography*. CRC Press, 2007. 3.5.2

[72] J. Katz and Y. Lindell. *Introduction to modern cryptography*. Chapman & Hall, 2008. 3.5.2

[73] Digital certificates. http://technet.microsoft.com/en-us/library/cc962029.aspx. Accessed June 15, 2012. 3.5.3

[74] Digital certificates. http://www.webopedia.com/TERM/D/digital_certificate.html. Accessed June 15, 2012. 3.5.3

[75] Iman Musa Almomani. *Security Solutions for Wireless Mobile Ad hoc Networks(WMANET)*. PhD thesis, De Montfort University, August 2007. 3.5.3, 6.2

[76] Ali Hilal Al-Bayatti. *Security management for mobile ad hoc network of networks (MANoN)*. PhD thesis, De Montfort University, February 2009. 3.5.3

[77] R. Baldessari, A. Festag, and M. Lenardi. C2C-C Consortium Requirements for NEMO Route Optimization. *draft-baldessari-c2ccc-nemo-req-01 (work in progress)*, 2007. 3.6.1

[78] Now: Network on wheels. http://dsn.tm.uni-karlsruhe.de/english/projects_now-project.php. Accessed November 10, 2012. 3.6.1

[79] Sevecom: Secure vehicular communication. http://www.sevecom.org/. Accessed November 10, 2012. 3.6.1

[80] Preciosa (privacy enabled capability in co-operative systems and safety applications). http://www.preciosa-project.org/home. Accessed November 10, 2012. 3.6.1

[81] Evita: E-safety vehicle intrusion protected applications. http://www.evita-project.org/. Accessed November 10, 2012. 3.6.1

[82] Oversee: Open vehicular secure platform. https://www.oversee-project.com/index.php?id=2. Accessed November 10, 2012. 3.6.1

[83] E. Mehul and V. Limaye. Security in mobile ad hoc networks. In *Handbook of Research in Mobile Business Second edition: Technical, Methodological and Social Perspectives Ed. Bhuvan Unhelkar*, pages 541–558, 2009. 3.6.2

[84] H. Yang, H. Luo, F. Ye, S. Lu, and L. Zhang. Security in mobile ad hoc networks: challenges and solutions. *Wireless Communications, IEEE*, 11(1):38–47, 2004. 3.6.2

[85] D. Djenouri, L. Khelladi, and N. Badache. A survey of security issues in mobile ad hoc networks. *IEEE communications surveys*, 7(4), 2005. 3.6.2

[86] J.L. Burbank, P.F. Chimento, B.K. Haberman, and W.T. Kasch. Key challenges of military tactical networking and the elusive promise of manet technology. *Communications Magazine, IEEE*, 44(11):39–45, 2006. 3.6.2

[87] L. Zhou and Z.J. Haas. Securing ad hoc networks. *Network, IEEE*, 13(6):24–30, 1999. 3.6.2

[88] J.P. Hubaux, L. Buttyán, and S. Capkun. The quest for security in mobile ad hoc networks. In *Proceedings of the 2nd ACM international symposium on Mobile ad hoc networking & computing*, pages 146–155. ACM, 2001. 3.6.2

[89] S. Capkun, L. Buttyán, and J.P. Hubaux. Self-organized public-key management for mobile ad hoc networks. *Mobile Computing, IEEE Transactions*, 2(1):52–64, 2003. 3.6.2

[90] A.A. Wagan, B.M. Mughal, and H. Hasbullah. Vanet security framework for trusted grouping using tpm hardware: Group formation and message dissemination. In *Information Technology (ITSim), 2010 International Symposium in*, volume 2, pages 607 –611, june 2010. 3.6.2

[91] S. Sumitkumar and R. Vijayan. Enhanced security for information flow in vanet using signcryption and trust level. *International Journal of Computer Applications*, 16(5):13–18, 2011. 3.6.2

[92] C. Hu, T.W. Chim, SM Yiu, L.C.K. Hui, and V.O.K. Li. Efficient hmac-based secure communication for vanets. *Computer Networks*, 2012. 3.6.2

[93] A. Wasef, R. Lu, X. Lin, and X. Shen. Complementing public key infrastructure to secure vehicular ad hoc networks [security and privacy in emerging wireless networks]. *Wireless Communications, IEEE*, 17(5):22–28, 2010. 3.6.2

[94] C.H. Yeh, Y.M. Huang, T.I. Wang, and H.H. Chen. Descv: A secure wireless communication scheme for vehicle ad hoc networking. *Mobile Networks and Applications*, 14(5):611–624, 2009. 3.6.2

[95] X. Lin, R. Lu, X. Liang, and X. Shen. Stap: A social-tier-assisted packet forwarding protocol for achieving receiver-location privacy preservation in vanets. In *INFOCOM, 2011 Proceedings IEEE*, pages 2147–2155. IEEE, 2011. 3.6.2

[96] M. Raya, P. Papadimitratos, and J.-P. Hubaux. Securing vehicular communications. *Wireless Communications, IEEE*, 13(5):8 –15, 2006. 3.6.2, 4.5

[97] J.M. Fuentes, A.I. González-Tablas, and A. Ribagorda. Overview of security issues in vehicular ad-hoc networks. IGI Global, 2010. 3.6.2

[98] Bharati Mishra, Priyadarshini Nayak, Subhashree Behera, and Debasish Jena. Security in vehicular ad hoc networks: a survey. In *Proceedings of the 2011 International Conference on Communication, Computing and Security*, ICCCS '11, pages 590–595, New York, NY, USA, 2011. ACM. 3.6.2

[99] M. Verma and D. Huang. Segcom: secure group communication in vanets. In *Consumer Communications and Networking Conference, 2009. CCNC 2009. 6th IEEE*, pages 1–5. IEEE, 2009. 3.6.2

[100] M. Raya, A. Aziz, and J.P. Hubaux. Efficient secure aggregation in vanets. In *Proceedings of the 3rd international workshop on Vehicular ad hoc networks*, pages 67–75. ACM, 2006. 3.6.2

[101] Dijiang Huang, Xiaoyan Hong, and M. Gerla. Situation-aware trust architecture for vehicular networks. *Communications Magazine, IEEE*, 48(11):128 –135, november 2010. 3.6.2

[102] Yong Hao, Yu Chengcheng, Chi Zhou, and Wei Song. A distributed key management framework with cooperative message authentication in vanets. *Selected Areas in Communications, IEEE Journal*, 29(3):616 –629, march 2011. 3.6.2

[103] Hsin-Te Wu, Wei-Shuo Li, Tung-Shih Su, and Wen-Shyong Hsieh. A novel rsu-based message authentication scheme for vanet. In *Systems and Networks Communications (ICSNC), 2010 Fifth International Conference*, pages 111 –116, aug. 2010. 3.6.2

[104] Xiaohui Liang, Rongxing Lu, Xiaodong Lin, and Xuemin Shen. Ppc: Privacy-preserving chatting in vehicular peer-to-peer networks. In *Vehicular Technology Conference Fall (VTC 2010-Fall), 2010 IEEE 72nd*, pages 1 –5, sept. 2010. 3.6.2

[105] R.J. Hwang, Y.K. Hsiao, and Y.F. Liu. Secure communication scheme of vanet with privacy preserving. In *Parallel and Distributed Systems (ICPADS), 2011 IEEE 17th International Conference*, pages 654–659. IEEE, 2011. 3.6.2

[106] C. Chen, X. Wang, W. Han, and B. Zang. A robust detection of the sybil attack in urban vanets. In *Distributed Computing Systems Workshops, 2009. ICDCS Workshops' 09. 29th IEEE International Conference*, pages 270–276. IEEE, 2009. 3.6.2

[107] M. Ghosh, A. Varghese, A. Gupta, A.A. Kherani, and S.N. Muthaiah. Detecting misbehaviors in vanet with integrated root-cause analysis. *Ad Hoc Networks*, 8(7):778–790, 2010. 3.6.2

[108] S. Ruj, M.A. Cavenaghi, Z. Huang, A. Nayak, and I. Stojmenovic. On data-centric misbehavior detection in vanets. In *Vehicular Technology Conference (VTC Fall), 2011 IEEE*, pages 1–5. IEEE, 2011. 3.6.2

[109] S.T. Li and X. Wang. Enhanced security design for threshold cryptography in ad hoc network. 3.6.2

[110] B. Wu, J. Chen, J. Wu, and M. Cardei. A survey of attacks and countermeasures in mobile ad hoc networks. *Wireless Network Security*, pages 103–135, 2007. 3.6.2

[111] I. Almomani and H. Zedan. End-to-end security solution for wireless mobile ad hoc network (wmanet). 2007. 3.6.2

[112] A.H. Al-Bayatti, H. Zedan, and A. Cau. Security solution for mobile ad hoc network of networks (manon). In *2009. ICNS '09. Fifth International Conference on Networking and Services*, pages 255 –262, april 2009. 3.6.2

[113] Y.G. Desmedt. Threshold cryptography. *European Transactions on Telecommunications*, 5(4):449–458, 1994. 3.6.2

[114] Y. Desmedt and Y. Frankel. Threshold cryptosystems. In *Advances in CryptologyÛCRYPTO'89 Proceedings*, pages 307–315. Springer, 1990. 3.6.2

[115] K.E. Sirois and S.T. Kent. Securing the nimrod routing architecture. In *sndss*, page 74. Published by the IEEE Computer Society, 1997. 3.6.2

[116] P. Wex, J. Breuer, A. Held, T. Leinmuller, and L. Delgrossi. Trust issues for vehicular ad hoc networks. In *Vehicular Technology Conference, 2008. VTC Spring 2008. IEEE*, pages 2800–2804. IEEE, 2008. 4.5

[117] J. Saperia. IETF Wrangles over Policy Definitions. *Network Computing*, page 36, 2002. 5.3

[118] A. Westerinen, J. Schnizlein, J. Strassner, M. Scherling, B. Quinn, S. Herzog, A. Huynh, M. Carlson, J. Perry, and S. Waldbusser. Terminology for policy-based management. *Request for comments*, 3198, 2001. 5.3

[119] T.Y.C. Woo and S.S. Lam. Authorization in distributed systems: A new approach. *Journal of Computer Security*, 2(2):107–136, 1993. 5.4

[120] D. Agrawal, S. Calo, K. Lee, J. Lobo, and D. Verma. *Policy technologies for self-managing systems*. IBM Press, 2008. 5.4

[121] Ponder: A policy language for distributed systems management. http://www-dse.doc.ic.ac.uk/Research/policies/ponder.shtml. Accessed July 24, 2012. 5.4

[122] The network simulator- ns-2 home page. http://www.isi.edu/nsnam/ns//. Accessed December 13, 2011. 6.2, 6.3

[123] M. Małowidzki. Network simulators: A developer's perspective. *Proc. Int. Sym. Performance Evaluation of Computer and Telecommunication Systems (SPECTSŠ04), San Jose, USA*, pages 1–9, 2004. 6.2

[124] Ns-3.12: Network simulator ns-3, home page. http://www.nsnam.org/ns-3-12/. Accessed December 22, 2011. 6.2

[125] Ns-2.29: Network simulator ns-2, home page. http://isi.edu/nsnam/ns/CHANGES.html. Accessed December 22, 2011. 6.2

[126] The cygwin project home page. http://cygwin.com/index.html. Accessed January 4, 2012. 6.2

[127] The Vint Project, U C Berkeley, Xerox Parc, Kevin Fall, and Editor Kannan Varadhan. The ns manual (formerly ns notes and documentation) 1. *Facilities*, 1(3):1–431, 2010. 6.2

[128] Eitan Altman and Tania Jimenez. Ns2 for beginners, lectures notes 2003-2004, university of de los andes, france, 4th December 2003. Accessed November 8, 2011. 6.2

[129] Glomosim: Global mobile information system simulation glomosim, home page. http://pcl.cs.ucla.edu/projects/glomosim/. Accessed December 13, 2011. 6.2

[130] Opnet modeler, home page. http://www.opnet.com/products/modeler/home.html. Accessed December 13, 2011. 6.2

[131] A. Abuarqoub, F. Alfayez, M. Hammoudeh, T. Alsboui, and A. Nisbet. Simulation issues in wireless sensor networks: A survey. In *SENSORCOMM 2012, The Sixth International Conference on Sensor Technologies and Applications*, pages 222–228, 2012. 6.2

[132] David Curren. A survey of simulation in sensor networks. *Architecture*, pages 867–872, 2008. 6.2

[133] Marc Esquius Morote. *Evaluation of MANET Routing Protocols in Realistic Environments*. College of Electronics and Information Engineering, Tongji University, Novembre 2010. Accessed December 28, 2011. 6.2

[134] T. Issariyakul and E. Hossain. *Introduction to network simulator NS2*. Springer Verlag, 2008. 6.2.1

[135] E. Raggi, K. Thomas, T. Parsons, A. Channelle, and S. Vugt. Working with text files. *Beginning Ubuntu Linux*, pages 265–278, 2010. 7.6

[136] D. Boneh and J. Shaw. Collusion-secure fingerprinting for digital data. *Information Theory, IEEE Transactions*, 44(5):1897–1905, 2002. 8.3

[137] T. Van Le, M. Burmester, and J. Hu. Short c-secure fingerprinting codes. *Information Security*, pages 422–427, 2003. 8.3

[138] M.S. Kankanhalli and KF Hau. Watermarking of electronic text documents. *Electronic Commerce Research*, 2(1):169–187, 2002. 8.3

[139] J. Brassil, S. Low, NF Maxemchuk, and L. O'Gorman. Hiding information in document images. In *Proceedings of the 29th Annual Conference on Information Sciences and Systems*, pages 482–489. Citeseer, 1995. 8.3