SECURITY AND PRIVACY IN VEHICULAR NETWORKS

(ARAÇ AĞLARINDA GÜVENLİK VE MAHREMİYET)

by

Ali Osman BAYRAK, B.S.

Thesis

Submitted in Partial Fulfillment
of the Requirements
for the Degree of

MASTER OF SCIENCE

in

COMPUTER ENGINEERING

in the

INSTITUTE OF SCIENCE AND ENGINEERING

of

GALATASARAY UNIVERSITY

October 2010

SECURITY AND PRIVACY IN VEHICULAR NETWORKS

(ARAÇ AĞLARINDA GÜVENLİK VE MAHREMİYET)

by

Ali Osman BAYRAK, B.S.

Thesis

Submitted in Partial Fulfillment
of the Requirements
for the Degree of

MASTER OF SCIENCE

Date of Submission : September 14, 2010

Date of Defense Examination : October 14, 2010

Supervisor : Asst. Prof. Dr. Tankut ACARMAN

Committee Members : Assoc. Prof. Dr. Temel ÖNCAN

: Asst. Prof. Dr. Murat AKIN

Acknowledgements

First of all, I would like to express my sincere gratitude to Asst. Prof. Dr. Tankut Acarman for his invaluable guidance and helping during the preparation of this dissertation.

I would also like to thank TÜBİTAK – Ulusal Elektronik ve Kriptoloji Araştırma Enstitüsü and my collegues at work for their great support and understandings.

Finally, I am grateful to my lovely wife Çiğdem Sevim Bayrak for her incredible patience and her endless support.

Ali Osman Bayrak,
İstanbul, October 14th, 2010

Table of Contents

Acknowledgements ii

Table of Contents iii

List of Figures v

List of Tables vi

Abstract vii

Résumé viii

Özet x

1 Introduction 1

2 VANET Security 4

2.1 Security Threats 4

2.2 Security Requirements 7

2.3 Background 8

2.4 Security of Mix-Zones against Traffic Flow Analysis 10

3 System Model 15

3.1 System Authorities 15

3.2 Identification 15

3.3 Tamper Proof Device (TPD) 15

3.4 Secure Communication 16

4 Secure and Privacy Protecting Protocol (S3P) 18

4.1 Certificate Management 18

4.2 Secure and Private Communication 20

4.3 Certificate Revocation 23

4.4 Communication Overhead....26
5 Protocol Evaluation....28
5.1 Evaluation against Security Requirements....28
5.2 Formal Protocol Verification with ProVerif....29
5.3 Performance Evaluation of S3P....33
5.3.1 Message Loss Ratio....34
5.3.2 Message Delay....35
6 Conclusion....37
References....38
Biographical Sketch....40

List of Figures

Figure 1.1 – VANET Infrastructure 2
Figure 2.1 - Bogus information attack 5
Figure 2.2 - Vehicle tracking scenario 6
Figure 4.1 – Safety message generation, verification and identity disclosure. 21
Figure 4.2 - Certificate revocation for node N 24
Figure 4.3 - Reference PKI based packet structure 26
Figure 4.4 - S3P packet format with RSA-2048 26
Figure 4.5 - S3P packet format with ECDSA 27
Figure 4.6 - S3P-Symmetric packet format 27
Figure 5.1 - Average message loss ratio 34
Figure 5.2 - Average message delay 35

List of Tables

Table 2.1 - Connection probabilities by types.....12
Table 2.2 - Connection closing probabilities 13
Table 2.3 - Probabilities of successfully tracking a vehicle from entry point to target..14
Table 2.4 - Probability of successfully tracking a vehicle after a mix-zone 14
Table 4.1 – Notations and descriptions 18

Abstract

The wide deployment of wireless technologies and sharply dropping costs of electronic components lead the idea of communicating vehicles in order to provide safer and efficient driving conditions. Vehicles increase their awareness of their environment by communicating with each other and with roadside infrastructure therefore increased safety and optimized traffic is achieved. Taking into consideration the benefits offered from vehicular communications and the large number of vehicles (hundreds of millions worldwide), vehicular ad hoc networks (VANETs) are likely become the most relevant realization of mobile ad hoc networks.

As a wireless communication technology, VANET is highly vulnerable to abuses and attacks. In order to ensure proper operation of safety-related applications the security of safety messages should be guaranteed even in the presence of persistent attackers. An adversary may inject a false information in order to mislead the target vehicles or with tampering the on board unit, implement an impersonation attack. He may also, by recording the messages of a target vehicle, track the vehicle's location and collect private information about the vehicle.

In this work, we study the security and privacy aspects of vehicular ad hoc networks. We first analyze the security requirements of VANET. We then propose a new scheme for secure and privacy protecting vehicular communications. We finally evaluate the proposed protocol against the previously defined requirements. According to our results, the proposed protocol fulfils all the security requirements and provides easy to manage and implement structures.

Résumé

Le large déploiement de technologies sans fil et de prix brusquement tombants de composantes électroniques mène l'idée de véhicules communicants pour fournir des conditions de conduite plus sûres et efficaces. Les véhicules augmentent leur conscience de leur environnement en communiquant l'un avec l'autre et avec l'infrastructure d'accotement de route c'est la raison pour laquelle la sécurité augmentée et a la circulation optimisée sont accompli. Le fait de prendre en considération les avantages offerts par des communications des véhicules et du grand nombre de véhicules (des centaines de millions dans le monde entier), les réseaux véhicule (VANETs) est probablement devenu la réalisation la plus pertinente de réseaux ad hoc mobiles.

Comme une technologie de communication sans fil, VANET est extrêmement vulnérable aux abus et aux attaques. Pour garantir l'opération nécessaire d'applications concernant la sécurité la sécurité de messages de sécurité devrait être garantie même en présence des attaquants persistants. Un adversaire peut injecter des renseignements faux pour induire les véhicules prévus en erreur ou avec le fait de tripoter le matériel sur véhicule, exécuter une attaque d'imitation. Il peut aussi, en enregistrant les messages d'un véhicule prévu, pister l'endroit du véhicule et recueillir des renseignements privés sur le véhicule.

Dans ce travail, nous étudions la sécurité et les aspects de vie privée de réseaux de véhicules. Nous analysons d'abord les exigences de sécurité de VANET. L'efficacité du mécanisme Mix-Zone, qui est déjà proposé par certains articles dans la littérature, est analysée par une simulation par rapport one par rapport à l'analyse des flux de la circulation impliquant des services d'info-divertissement. Nous proposons alors un nouveau projet pour les communications des véhicules. Nous évaluons finalement le

protocole proposé contre les exigences auparavant définies. Selon nos résultats, le protocole proposé réalise toutes les exigences de sécurité et fournit des structures faciles à diriger et exécuter.

Özet

Kablosuz ağ uygulamalarının yaygınlaşması ve elektronik cihaz maliyetlerinin önemli şekilde düşüşü, daha güvenli ve etkili sürüş koşullarına ulaşma amacına yönelik olarak birbirleriyle haberleşen araçlar fikrine önayak olmuştur. Araçların birbirleriyle ve yol üzerinde bulunan baz istasyonları ile haberleşebilmeleri, çevresel farkındalıklarını artıracak ve böylece daha güvenli ve araç trafiğini en iyi hale getirebilecektir. Araç ağlarının sağlayacağı yararlar ve araç sayısının çokluğu (dünya üzerindeki yüzmilyonlarca araç) göz önüne alındığında, araç ağlarının, mobil ad-hoc ağlarıyla ilgili en geniş uygulaması olacağı ortaya çıkmaktadır.

Araç ağlarının kablosuz haberleşme teknolojisini kullanıyor olması, bu ağları kötüye kullanma ve saldırlara açık hale getirmektedir. Saldırı altında dahi araç ağlarının güvenlik uygulamalarının düzgün işleyebilmeleri için mesaj güvenliği sağlanmalıdır. Bir saldırgan, araçları yanlış yönlendirmek amacıyla sisteme sahte mesajlar gönderebilir ya da araç üzerindeki cihazları kurcalayarak kimlik hırsızlığı yani bir başkasının yerine geçme saldırısını gerçekleyebilir. Bununla birlikte saldırgan, hedefindeki aracın haberleşmesini dinleyerek, aracın yerini takip edebilir ve kullanıcısının kişisel bilgilerini toplayabilir.

Bu tez çalışmasında, araç ağlarında güvenlik ve mahremiyet konuları ele alınmıştır. Öncelikle araç ağlarının güvenlik gereksinimleri analiz edilmiş, daha sonra güvenli ve mahremiyeti koruyan bir araç ağları haberleşme protokolü ortaya konulmuştur. Son olarak daha önceden tanımlanan gereksinimler kullanılarak, ortaya konan protokol

analiz edilmiştir. Sonuç olarak, ortaya konan protokol tüm güvenlik gereksinimlerini sağlamakla birlikte yönetilebilirliği ve uygulaması kolay olarak değerlendirilmiştir.

1 Introduction

The wide deployment of wireless technologies and sharply dropping costs of electronic components lead the idea of communicating vehicles in order to provide safer and efficient driving conditions. Vehicles increase their awareness of their environment by communicating with each other and with roadside infrastructure therefore increased safety and optimized traffic is achieved. Taking into consideration the benefits offered from vehicular communications and the large number of vehicles (hundreds of millions worldwide), vehicular ad hoc networks (VANETs) are likely become the most relevant realization of mobile ad hoc networks.

VANET consist of vehicles and road side units (RSUs) as network nodes and enables inter-vehicle communications (IVC) along with the road side to vehicle communications (RVC) as seen in Figure 1.1. As the majority of the VANET nodes will consist of vehicles, the network dynamics will be characterized by high speeds, persistent mobility and mostly short-lived connections (e.g., vehicles on different directions on high ways). Vehicles in VANET are equipped with various sensors, an on board unit (OBU) for information processing, and a tamper-resistant hardware security module for housing the cryptographic keys and operations.

IVC and RVC applications fall into two categories: Safety-related applications and infotainment applications. The idea behind the safety related applications is that vehicles generate messages about safety related events, such as accidents, road conditions, and their own behavior like emergency breaking. These safety related messages are broadcasted periodically and neighboring vehicles and RSUs are informed. The period of these safety related messages is expected to be between 100 and 300 milliseconds.

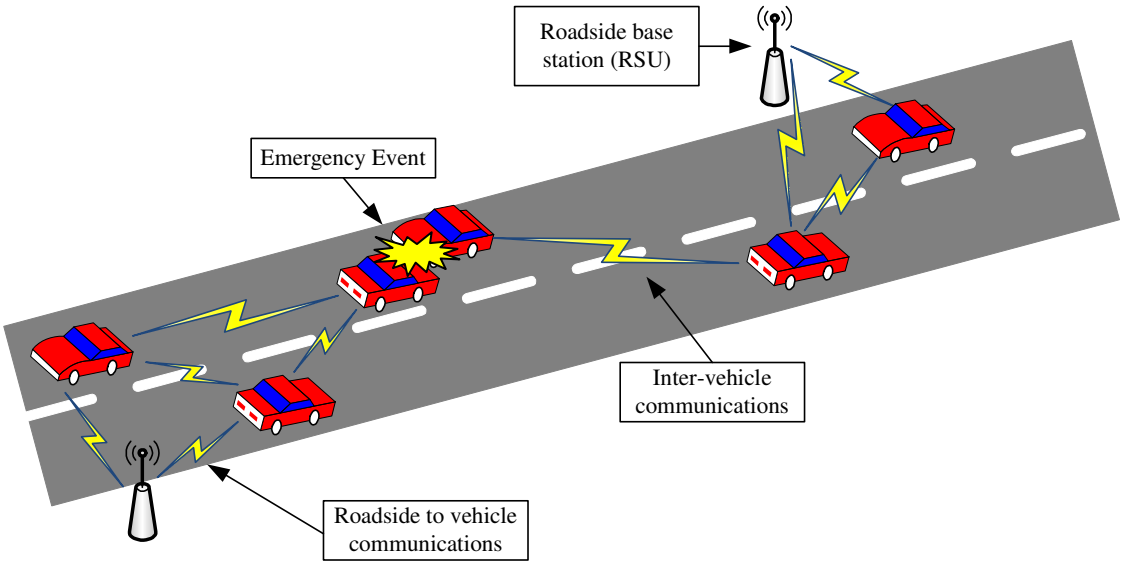

Figure 1.1 – VANET Infrastructure

As infotainment stands for information and entertainment, infotainment applications focuses on non safety related operations such as internet access, toll collection, and location-based services. These services enable passengers to access web pages, internet services (e.g., internet radio or television, e-mail, online games), to find nearest gas station or to pay the high way usage fee.

As a wireless communication technology, VANET is highly vulnerable to abuses and attacks. In order to ensure proper operation of safety-related applications the security of safety messages should be guaranteed even in the presence of persistent attackers. An adversary may inject a false information in order to mislead the target vehicles or with tampering the on board unit, implement an impersonation attack. He may also, by recording the messages of a target vehicle, track the vehicle's location and collect private information about the vehicle.

In this work, we study the security and privacy aspects of vehicular ad hoc networks. We first analyze the security requirements of VANET. We then propose a new scheme

for secure and privacy protecting vehicular communications. We finally evaluate the proposed protocol against the previously defined requirements.

The rest of the document is organized as follows. In section 2, we analyze the security requirements of a vehicular communication system. Section 3 includes detailed design of a novel protocol providing secure and privacy protecting communications. In section 4, we evaluate the protocol given in section3 against the requirements detailed in section2. In section 5, we conclude our study by commenting the obtained results.

2 VANET Security

2.1 Security Threats

In this section, security threats against the vehicular communications are examined. Attacks can be implemented either from inside or outside. Insider adversary is an authenticated member of the VANET while the outsider is considered as an intruder who uses vulnerabilities in the network-specific protocols. As infotainment services and their network-specific protocols differ, we only consider attacks on safety related communications. Threats against safety related communications can be categorized as follows:

1. **Bogus information:** Attacker injects false information in the network to affect the behavior of other drivers. In such case, attacker may have a single or multiple target vehicles and by launching this attack, misleads the targets about the road conditions (e.g., traffic jam, accident, closed roads). As the misleading information can be generated by explicitly or by falsifying the sensor devices the attack can be implemented by either an insider adversary or an outsider adversary. In Figure 2.1, as an example of a *bogus information attack*, attackers A1 and A3 disseminate false information to affect the decision of other vehicles (in this case V) and thus clear the way of attacker A2.

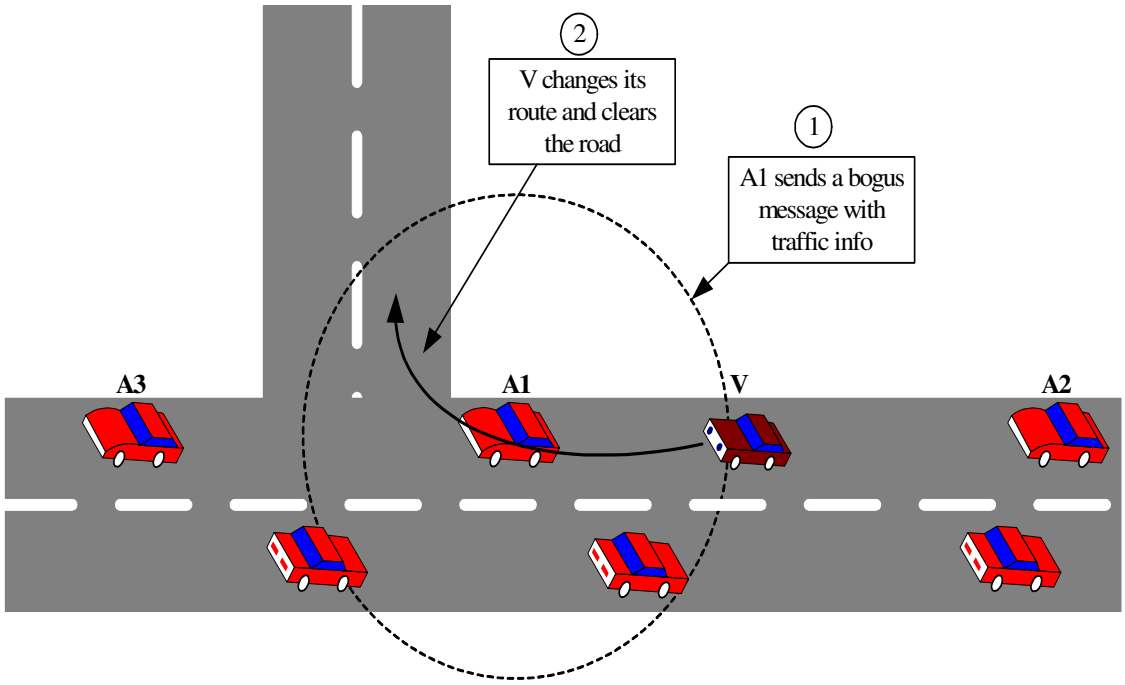

Figure 2.1 - Bogus information attack

2. **Masquerading (Identity Theft):** Adversary pretends to be another vehicle by using false identities to implement an impersonation attack. Masquerading allows adversary to hide its real identity while to be considered by the network as a legitimate node.

3. **Replay attack:** Adversary replays previously saved safety related messages which are generated by legitimate vehicles and misleads the neighboring nodes. As these messages are generated by valid network nodes, receiving vehicles believe that these messages are authenticated.

4. **Modification attack:** The message is altered during or after transmission. The adversary may wish to change the source or content of the message in terms of the position or time information that had been sent and saved in its device to escape from the consequence of a criminal/car accident event.

5. **Vehicle tracking:** Since wireless communication is on an openly shared medium, an adversary can easily eavesdrop on any traffic. After the adversary intercepts a significant amount of messages in a certain region, the adversary may trace a vehicle in terms of its physical position and

moving patterns simply through information analysis. An example is illustrated in Figure 2.2. In this example, an attacker implements his own antennas to receive safety messages from its target and tracks his target's movements.

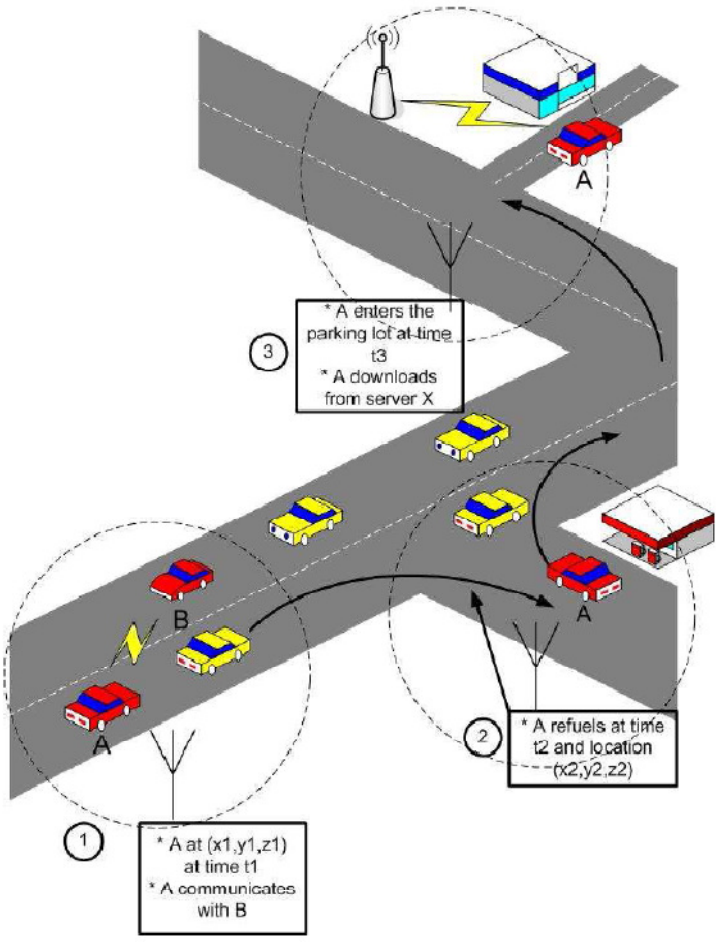

Figure 2.2 - Vehicle tracking scenario

6. **Denial-of-Service (DoS) attack:** The adversary may want to bring down the VANET or even cause an accident. Example attacks include channel jamming and aggressive injection of dummy messages.

2.2 Security Requirements

VANET should address to the following requirements in order to provide secure and privacy protecting communications:

Message authentication and integrity: Message receiving nodes need to be ensured that the source of the message in a valid VANET node in order to prevent outsider attackers to inject messages in to the network. It is also required that the message integrity should be provided to guarantee that the message has not been altered during transmission. When digital signatures are used to provide the message origin authentication, they also supply integrity protection.

Message non-repudiation: Sender of a message should not be able to deny having sent the message for liability reasons. In case of an accident, messages sent by involving VANET nodes can be examined and the cause of the accident may be revealed. In this case, message non-repudiation mechanism prevents involving nodes to deny sending the messages.

Entity authentication: Receiver should be ensured that the sender is an authenticated entity of the system and is alive when the message is sent. Entity authentication mechanism prevents an outsider attacker to sent messages over the network. It also provides protection against replay attacks as the message generating node needs to be alive when the message is sent.

Privacy protection: The private information of a system user should be protected against a global adversary. An attacker monitoring the vehicular communications can track a specific vehicle online or offline. In VANET, *anonymity* of a vehicle should be provided by achieving unlinkability between a vehicle's identity and the messages generated by that vehicle. An adversary should not be able to make a link between a message and its generator's identity as well as between two messages generated by the same vehicle.

2.3 Background

There have been some efforts addressing the security challenges in VANET. VANET's security challenges and requirements are analyzed in [1], [2], [3], [4] and [5]. From privacy point of view, these efforts fall into two categories: Pseudonym based solutions and group signature based solutions.

In the first category, privacy protection provided by using pseudonyms precisely public-private key pairs which do not disclose the identity of the communicating node. Each node has a set of pseudonyms given by an authority to sign the safety messages. Pseudonyms are used once in a period of time and never used again. Vehicles change pseudonyms regularly to avoid getting tracked by an attacker. Pseudonym utilization for protecting the privacy proposed in [1], [2], [3], [5], [6], [7], [8], [9], [10], [11], [12]. Despite attracting so much attention, pseudonym-based solutions have several drawbacks.

As mentioned in [6] changing pseudonyms does not guarantee the unlinkability of new and old pseudonyms of a node. One can still link old and new pseudonyms by using spatial and temporal relation between new and old locations. As a solution, in [6] a random silent period between pseudonym changes is proposed. Due to the periodical broadcast of safety messages, this solution does not enhance the privacy protection [7]. To provide better protection of privacy [7] proposed that vehicles form groups and only group leader communicates the outside serving as a gateway. This solution is applicable only to infotainment communications because each node has to broadcast safety messages periodically. To overcome the linkage problem between two pseudonyms successively used, mix-zones, where all vehicles within a region, are proposed in [5], [9], [10], and [11]. Nodes in the mix-zone change pseudonyms at the same time in order to prevent an attacker to link new and old pseudonyms. The level of unlinkability of pseudonyms achieved by this approach is highly dependent to the number of vehicles in the mix-zone. To make difficult to link two pseudonyms successively used, [9] also proposes to encrypt all the safety messages sent by vehicles in the mix-zone.

Despite all efforts to make pseudonym-based solution more effective, this approach still has some disadvantages. To be efficient, for a vehicle, driven two hours per day, needs approximately 43800 public-private key pairs (pseudonyms) per year [1]. These keys and corresponding certificates are generated and signed by an authority for all vehicles registered with it. In order to identify a vehicle from its safety messages, possibly for liability reasons or misbehaving, the authority needs to keep the anonymous credentials of all the vehicles in its administrative region and to search a very large database. In addition, as safety messages are signed by sender, corresponding certificate should be included in every message so that receiving nodes can verify the signature. This mechanism increases the message length as well as the computational overhead because sender's certificate needs to be validated along with the signature validation. Certificate revocation is a major problem for pseudonym-based solutions because certificate revocation lists (CRLs) grows rapidly as it includes all pseudonyms issued to nodes whose certificate is revoked.

Using group signature schemes is another proposed solution for privacy protection. In [13], and [14], nodes form groups, administrated by a group manager, to sign anonymously the safety messages on behalf of the group. While members can validate the authenticity of the messages without disclosure of the sender's identity, only group manager can determine the sender's identity. Group manager is also responsible for revoking the certificates. Different from this approach, [12] proposes to use group signature scheme to sign pseudonyms in order to ease the authority's workload. While nodes use pseudonyms to sign messages, they produce their own pseudonyms using group signature technique. Offering great potential for privacy protection for VANET, group signature schemes introduce scalability problem. Forming groups containing large number of vehicles and allowing mobility between regions administrated by different group managers are the problems to be addressed.

Another approach, different from pseudonym usage and group signature schemes, is proposed in [15]. While offering to utilize standard Public Key Infrastructure (PKI) when an RSU is not available, this approach uses Hash-based Message Authentication Code (HMAC) and one way hash function to provide anonymity and message

authentication. A node, near an RSU, shares a secret symmetric key with RSU and signs the messages with the symmetric HMAC code. This scheme is highly dependent to RSU's existence and computational power as RSU is the only entity to validate the messages in its region. Nodes accept a message after it is verified by RSU which introduces a certain message delay in the system.

The most prominent industrial effort in this domain in Europe is carried out by the Car 2 Car Communication Consortium [16] and several projects such as Secure Vehicular Communication (SEVECOM) [17], while in the USA it is addressed by the Dedicated Short Range Communications (DSRC) [18] consortium, especially the IEEE P1609.2 Working Group [19].

Some commercial products already make use of vehicular communication without taking the security aspect into account. For example, insurance companies install black boxes in cars to collect their usage data and to calculate insurance costs accordingly. Another related application is GPS car tracking.

2.4 Security of Mix-Zones against Traffic Flow Analysis

To improve privacy protection, mix-zones usage, where all nodes within a region changes their pseudonyms at the same time, is proposed for schemes using pseudonymous authentication mechanism. Nodes at mix-zones have to change not only their pseudonyms but also all identity related parameters including IP and MAC addresses.

We believe that in any scheme it is not possible to change all identity related parameter automatically taking into consideration that the internet usage characteristics are also identity related parameter. An attacker, monitoring a node's internet connections before and after mix-zones, may use this information to link to pseudonyms successively used. Wide-variety of services offered by the infotainment framework may allow a node to be tracked. Internet provides large number of sites offering different services like Internet radio, TV, instant messaging or e-mailing. Connections

to these services, given by a large number of service providers, can be used to identify a network node among the others. For example, a vehicle, having two internet connections, one for a specific internet radio and other for a corporate e-mail server, before entering a mix-zone will probably keep these connections after passing that mix-zone. Analyzing the connections before and after that mix-zone, an attacker can link the two pseudonyms used by the vehicle between the mix-zone.

The aim of this section is, by conducting a simulation, to question if mix-zone based solutions have vulnerability against traffic flow analysis on infotainment services.

VANET is modeled in JAVA as a 10x10 Manhattan network with d = 4, where d is the number of road segments that meet at each intersection. Each road is bi-directional and consists of six lanes (three for each direction). There are 100 intersections on the map and at each intersection a mix-zone is created. Vehicles randomly enter to the map from one of the 40 entry points. Each vehicle entering the map, randomly picks an exit point excluded the entry point. Having destination, vehicles on the map, uses shortest path to reach the exit point. When a vehicle reaches its target and moves out of the map, a new vehicle enters the map at a randomly selected entry point allowing the density of vehicles in simulation remains constant.

The number of vehicles, M, (or density of vehicles per mix-zone) is taken as a parameter taking into consideration that mix-zone performance is highly dependent on the density of vehicles which is denoted as ρ and calculated as $\rho=M/(10x10)$.

The number of infotainment sites available to vehicles for connection is another parameter defined in simulation. Vehicles randomly make connections to available infotainment sites. The number of these sites plays an important role in the simulation as the probability of a two different nodes connecting to the same site, which is an important attribute for identifying nodes, depends on the number of the available sites. Infotainment sites are divided into three categories; type1, type2 and type3. Each of these categories has different characteristics. The first category represents the sites like corporate e-mail servers, organizational web sites, etc. This category can be described

as a special purpose services and will have wide variety of sites. Second category represents broadcasting services like internet radio, TV or movie sites. In this category, it is obvious that the connections are very likely to remain open and the connection times are longer than other sites. The last category represents popular sites like online newspapers, social networks, and portal sites and is expected to be connected by a big portion of VANET users. Considering these categories, it is believed that the number of sites available to users should be different on each category and type1 must have more sites than the others. Number of sites in type2 and type3 are takes as 10% of the number of sites in type1. For example, if type1 consists of 50 sites then type2 and type3 will have 5 sites in each.

When a vehicle is initialized in the simulation it randomly creates connections on randomly selected types. This randomization includes zero connection which means a vehicle may start in simulation without having any connection.

Table 2.1 - Connection probabilities by types

		Number Of Connections			
Connection Types		1	2	3	4
	type 1	1/2	1/10	1/100	0
	type 2	7/10	1/10	1/100	0
	type 3	6/10	1/10	1/100	0

In Table 2.1 it can be seen the predefined parameters for initializing a new connection of a given type in the vehicle's connection update procedure. It can be read as follows: if a node wants to initialize a connection of type1 and it does not have any connection of type1, then the probability of making a connection of that type is 1/2. But if it has already one connection of type1 then this probability will be 1/10. As it can be seen from the table, a vehicle cannot have 4 connections of the same type.

Each vehicle updates its connections every second and randomly decides whether to close the connection or not according to predefined probabilities in Table 2.2. Probabilities are given for 20 seconds for the ease of reading.

It is assumed that the attacker is able to listen all wireless communications in the map except the communications in mix-zones. Attacker randomly chooses a vehicle as a target and tries to track it from its entry point to exit. After every mix-zone, try to link its old pseudonym to its new pseudonym. When its target enters a mix-zone, attacker assumes the vehicles exiting the mix-zone as candidates and after a predefined period the attacker makes its decision among the candidates using expert systems.

Table 2.2 - Connection closing probabilities

		Probability of closing a connection
Connection Types	type 1	40 % per 20 seconds
	type 2	7 % per 20 seconds
	type 3	15 % per 20 seconds

The probability of successfully tracking a vehicle from entry point to its target by the attacker is represented by Table 2.3. Number of sites is represented as type1, type2 and type3. For example (50, 5, 5) represents that type1 has 50 available sites and type2 and type3 has 5. Results show that the attacker can only track its target until it reaches its destination with the probability at most 30% and the probability drops to 5% in dense traffic.

Table 2.4 represents the probability of successfully tracking a vehicle after a mix-zone by the attacker. Results show that even with the higher density, attacker can successfully link the old and new pseudonyms over 50%. Success rate increases up to 90% with the decreasing number of the vehicle density as expected.

Table 2.3 - Probabilities of successfully tracking a vehicle from entry point to target

		Number Of Sites														
Density Of Vehicles		50	5	5	100	10	10	200	20	20	400	40	40	800	80	80
	5	% 34			% 28			% 32			% 30			% 32		
	10	% 13			% 23			% 12			% 13			% 18		
	20	% 15			% 13			% 12			% 12			% 8		
	40	% 4			% 5			% 7			% 7			% 8		
	80	% 6			% 5			% 8			% 8			% 9		

Table 2.4 - Probability of successfully tracking a vehicle after a mix-zone

		Number Of Sites														
Density Of Vehicles		50	5	5	100	10	10	200	20	20	400	40	40	800	80	80
	5	% 87			% 81			% 89			% 87			% 87		
	10	% 76			% 81			% 78			% 76			% 81		
	20	% 73			% 70			% 72			% 75			% 73		
	40	% 66			% 67			% 67			% 67			%64		
	80	% 61			% 57			% 61			% 61			% 61		

It is also expected that number of sites should have influence on probability. The fewer number of sites should make attacker harder to distinguish vehicles after a mix-zone. Results show that the number of sites does not affect probability as expected. One explanation can be as follows: even the least number of sites (in this simulation 60) is available; it still provides quite big space for distinguishing the vehicles taking into consideration of different parameters like number of connections of the target vehicle before and after mix-zone and connection types. It is reasonable to say that in real world very large numbers of sites would be available and this situation will not harden attacker's job.

3 System Model

In this section, the main elements of the VANET architecture, which addresses identity, credential, and key management and secure communication, is outlined.

3.1 System Authorities

As PKI is best suited for identifying the VANET nodes, a number of certification authorities (CAs) should be established for managing identities and credentials. A local CA is responsible for a region (country, state, etc.) and all VANET nodes in this region are registered with it. In order to provide mobility of VANET nodes between regions, all local CAs are registered with a root CA.

3.2 Identification

Each node is registered with only one local CA. In registration process, local CA issues a certificate, containing unique identity and validity period information of the node, and a public-private key pair. The local CA is also responsible for the eviction of nodes or the withdrawal of compromised cryptographic keys via the revocation of the corresponding certificates.

3.3 Tamper Proof Device (TPD)

We assume that all vehicles in VANET contain several sensors and on board unit (OBU) to produce safety related messages. OBU collects information from sensors and generates safety messages. Before transmitting the message, some cryptographic operations have to be performed.

In any cryptographic system, the confidentiality and the integrity of the private keys are essential whilst vehicles by their nature are highly vulnerable for tampering. Security and reliability of VANET relies on the tamper resistant hardware security modules (HSM) in which the cryptographic keys are stored and cryptographic operations, such as digital signatures and encryption, are performed.

To reduce the risk of its compromise by attackers, the device should have its own battery, which can be recharged from the vehicle, and clock, which can be securely resynchronized, when passing by a trusted roadside base station.

HSM never gives away the sensitive information outside and is hardly mounted on the vehicle. HSM is designed to erase all the sensitive information it contains if it is physically tampered or unmounted from the vehicle. In addition to being tamper proof, HSM also needs to be resistant against side-channel attacks to prevent the reveal of the sensitive information.

The access to this device should be restricted to authorized people. For example, cryptographic keys can be renewed at the periodic technical checkup of the vehicle. As its name implies, the TPD contains a set of sensors that can detect hardware tampering and erase all the stored keys to prevent them from being compromised. The availability of this feature makes the TPD on one hand too sensitive for VANET conditions (for example, the device can be subject to light shocks because of road imperfections; TPDs also cannot tolerate extreme temperatures that may not be unusual for vehicles) and on the other hand too expensive for non-business consumers. In fact, current commercial products contain cryptographic coprocessors, are oriented towards financial applications and cost several thousands of dollars.

3.4 Secure Communication

Digital signatures are the basic tools to secure communications and are used for all messages. To satisfy both the security and anonymity requirements, we adopt shared and periodically changed public-private key sets mechanism. Rather than utilizing the

same long-term public and private key for securing communications, each vehicle utilizes the shared private-public key pairs for message authentication. Non-repudiation requirement is fulfilled by using long-term identities which are protected for anonymity purposes by the local CA's public key. Mapping between the message and its sender can only be done by local CA.

4 Secure and Privacy Protecting Protocol (S3P)

In order to provide secure communications in VANET we use digital signatures as an underlying mechanism. Local CAs are responsible for managing the credentials of VANET nodes. Certificate revocation is also undertaken by local CAs. All nodes (including vehicles and RSUs) are equipped with an HSM and all cryptographic operations are performed in this tamper proof device. Notations used for defining the protocol are given in

Table 4.1 – Notations and descriptions

Notation	Description
ID_N	Identity of the VANET node N.
Prv_N	Private key of node N.
Pub_N	Public key and certificate of node N.
Prv_{Ai}, Pub_{Ai}	Key pair in anonymity key set at index *i*.
Prv_{Ei}, Pub_{Ei}	Key pair in emergency key set at index *i*.
a||*b*	Concatenating *b* to *a*.
Sign(m, K)	Signing m with the key *K*.
Verify(m, K)	Validating the signature *m* with the key *K*.
Enc(m, K)	Encrypting *m* with the key *K*.
Dec(m, K)	Decrypting *m* with the key *K*.

4.1 Certificate Management

Each node N has a unique identity IDN given by its local CA in a globally adopted format. Local CA issues public-private key pair (PrvN, PubN) and certificate

containing necessary information, such as node type (private vehicle, public vehicle, RSU, etc.), validity period, etc. about node N.

It is assumed that anonymity of RSUs is not necessary and RSUs use their private key to sign messages. If the anonymity of public vehicles (public buses, ambulances, police cars, etc) is considered to be pointless, same strategy used with RSUs can be adopted. In order to validate the signature of messages generated by RSUs, receiving nodes should be able to check the validity of the RSU's certificate. Local CA manages CRL to provide the list of untrusted nodes. When an RSU is compromised or reported as misbehaving, local CA updates the CRL adding its identity and distributes to all nodes.

To provide anonymity, local CA generates two sets of public-private key pairs and corresponding certificates. Each set contains number of n key pairs and certificates. The first set, called anonymity key set, is used by vehicles for signing the safety messages while the second set, called emergency key set, is used for signing purposes in emergency situations such as certificate revocation caused by a misbehaving vehicle. Every key pair in key sets is valid for a period decided by the local CA. There are two key pairs valid in a given period. One is in anonymity key set and the other is in emergency key set. Depending on the situation, normal or emergency, one pair is actively used by the nodes for signing the safety messages. At the end of the period, nodes start using the next key pair in the active key set. If the active key pair is the last key pair in the key set, nodes connect to local CA, over a secure communication, to get next key sets in order to use at the end of the lifetime of the current key pair.

For example, if the local CA chooses n as four and validity period of a key pair as a week, every node will have eight key pairs and corresponding certificates, four ((PrvA1, PubA1), (PrvA2, PubA2), (PrvA3, PubA3), (PrvA4, PubA4)) in anonymity key set and four ((PrvE1, PubE1), (PrvE2, PubE2), (PrvE3, PubE3), (PrvE4, PubE4)) in emergency key set, to use in four weeks. First week, nodes use (PrvA1, PubA1) key pair if no emergency situation is reported by local CA. At the end of the first week, nodes change the active key pair and start using (PrvA2, PubA2). If an emergency

situation is reported in the second week, nodes immediately switch to emergency key set and (PrvE2, PubE2) is utilized. From the beginning of the fourth week, nodes start to connect to the local CA and obtain the next key sets which will be in use for the four next weeks.

The emergency situation, reported by the local CA, indicates that there is a misbehaving node in the network and key sets need to be updated. Local CA generates new key sets and sends notification message to all nodes. After receiving the emergency situation message, every node switch to emergency key set and updates its key sets in the validity period of the corresponding emergency key pair in use.

To provide mobility in VANET, node N entering different region, connects, over a secure communication, to that region's local CA to get key sets in use. Local CA checks the N's certificate online and if it is valid sends the active key sets to the node N.

4.2 Secure and Private Communication

In order to provide message authentication and integrity protection, VANET nodes, except RSUs, sign safety messages with the active private key from the anonymity key set (in case of emergency, emergency key set is used). As all keys used in VANET, reside in tamperproof hardware security module, nodes in VANET assume that a node using these keys is a trusted entity of the system and any message signed with the active private key is considered to be authentic.

By using the shared active private key to sign safety messages, VANET nodes preserve their anonymity against global adversary and provide authentic information to the system.

Message non-repudiation is another requirement to be fulfilled by our protocol. Using tamperproof HSM does not prevent an adversary to alter the sensory data and mislead the other nodes by false information. In such case we need to identify misbehaving node in order to exclude it from the system. To prevent a node to deny having sent a message, all nodes sign safety messages with their long term private key. Providing the

non-repudiation, the latter operation reveals the identity of the node and violates the anonymity requirement. Only authorized entity, in this case the local CA, should be able to identify the signer of the message. To keep the identity of the node anonym, the signature created with the node's private key is encrypted with local CA's public key.
An adversary may record some safety messages generated by trusted nodes and resend to other nodes in validity period of the key pairs, which are used to sign these messages, in order to mislead target vehicles. To avoid such replay attacks, safety messages should contain time information and receiving nodes must check the validity of the safety messages by controlling the time information. To provide time stamps in these messages, HSM should have internal clock and keep it up to date by getting regularly time information from local CAs or RSUs. HSM is responsible for adding time stamps to safety messages and checking the validity of time information in received messages.

Figure 4.1 shows the protocol steps for creating and verifying the safety messages.

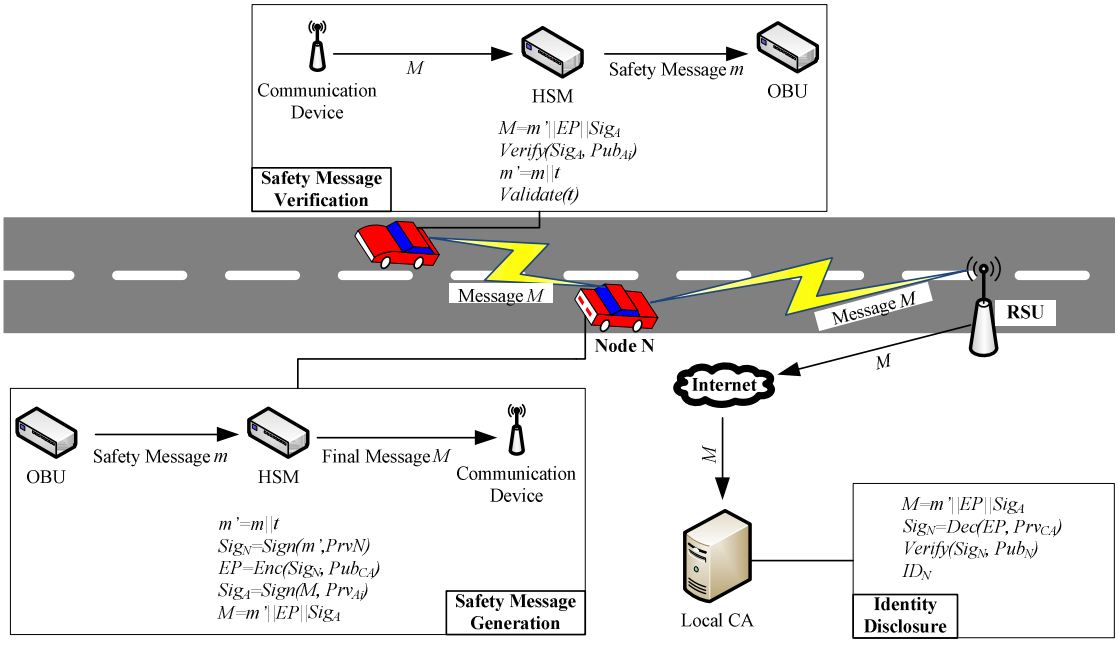

Figure 4.1 – Safety message generation, verification and identity disclosure.

Protocol steps for creating safety messages are listed as follows:

1) HSM receives safety message *m* generated by OBU according to the received sensory information.

2) HSM produce *m'* adding time stamp *t* to the *m*. $m'=m||t$

3) HSM calculates signature Sig_N of the *m'* by using the node N's private key. $Sig_N=Sign(m', Prv_N)$

4) HSM encrypts Sig_N using the local CA's public encryption key Pub_{CA} and create encrypted packet *EP*. $EP=Enc(Sig_N, Pub_{CA})$

5) HSM generates packet *M* by concatenating *m'* and *EP*. $M=m'||EP$

6) HSM produce signature Sig_A, over *M*, using the active private key Prv_{Ai} of the anonymity key set (emergency key set in case of emergency). $Sig_A=Sign(M, Prv_{Ai})$

7) HSM passes over *M* and Sig_A communication device to broadcast the safety message.

As anonymity for RSUs is not required, RSUs sign safety messages with their own private signature key without generating encrypted packets (*EP*s).

When a vehicle receives a safety message ($M||Sig_A$), it performs the following steps:

1) Safety message is passed to HSM by communication device.

2) HSM validates the signature Sig_A using the active public key Pub_{Ai} of the anonymity key set (emergency key set in case of emergency). $Verify(Sig_A, Pub_{Ai})$

3) If the signature is valid, HSM extracts *m'* from *M* simply removing the *EP*. If HSM fails to validate the signature, the message is discarded.

4) HSM obtains the timestamp *t* and message *m* from *m'* and checks if the message is created in a valid period of time. If this check fails, HSM discards the message.

5) HSM passes *m* to the on board unit for processing.

If Local CA receives a report indicating that a node is misbehaving, following actions are performed:

1) Local CA obtains the safety message ($M||Sig_A$) generated by the suspicious node.

2) Local CA extracts the *EP* from *M*.

3) Local CA calculates the Sig_N from *EP* by decrypting it with its private key Prv_{CA}. $Sig_N=Dec(EP, Prv_{CA})$

4) From Sig_N, local CA extracts the node's certificate thus its identity.

5) Local CA validates the signature Sig_N using the node's public key Pub_N. $Verify(Sig_N, Pub_N)$

If the signature is valid, local CA successfully identifies the node reported as misbehaving.

4.3 Certificate Revocation

Any scheme using public key infrastructure (PKI) should employ an efficient certificate revocation protocol. A network entity's private key may be compromised or a node may become untrusted. Although compromise of the private keys, which are stored in tamperproof hardware security modules, is not considered as a threat, a mechanism should be offered to prevent a VANET node, identified as misbehaving entity, from injecting fake messages in to the network.

In Figure 4.2 certificate revocation procedure is given for vehicle *N*. Local CA identifies the misbehaving node from its safety messages as described in the previous

section. Local CA generates emergency message and broadcasts it to the network. Emergency message, signed by local CA, contains time stamp and the misbehaving nodes identity that is encrypted with the misbehaving node's public key. Local CA generates new key sets to be used after the expiration of the emergency key pair.

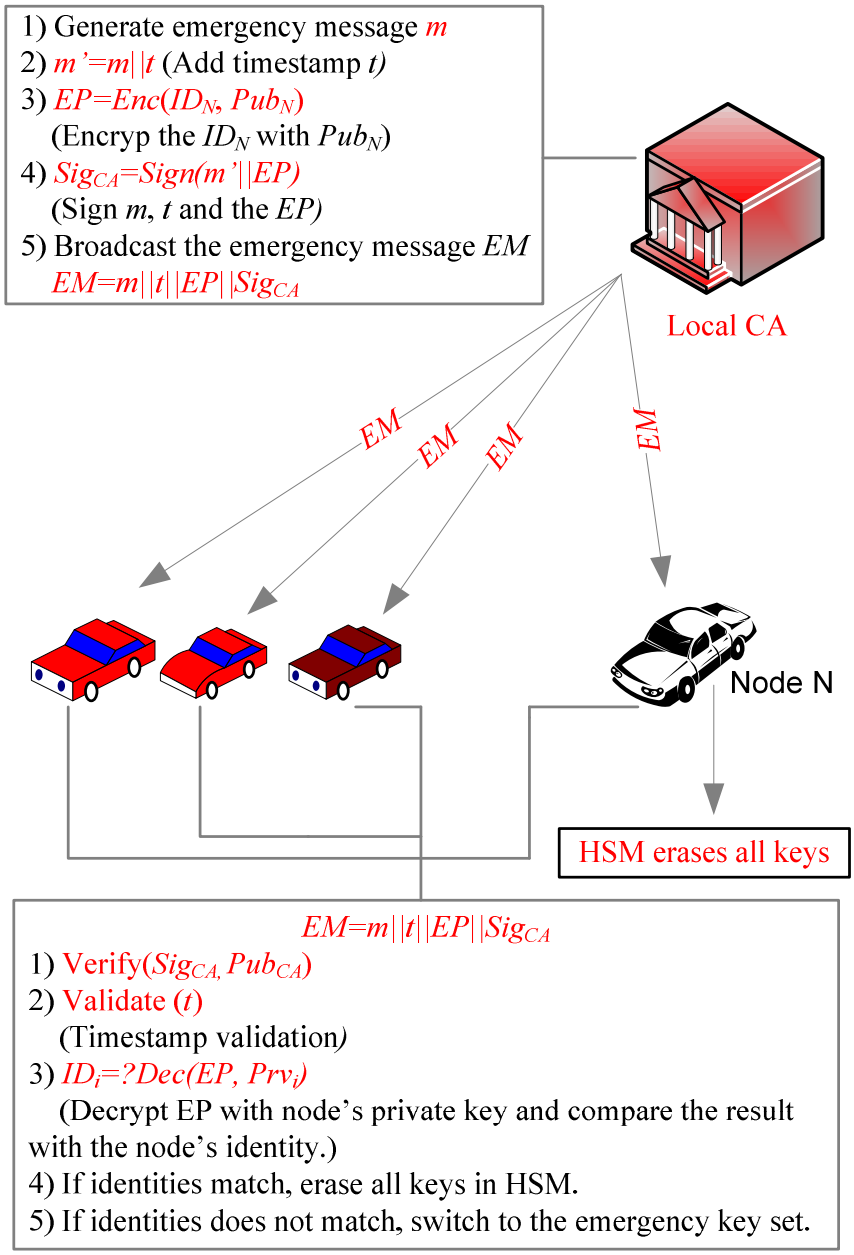

Figure 4.2 - Certificate revocation for node N

HSM in each node, after receiving the emergency message, first validates the signature and the time stamp in the message. If this is an authenticated message, HSM decrypts the part, containing the misbehaving node's identity, with its private key and compares the result with its own identity. If HSM finds its own identity in the emergency message, it erases all the keys it contains and becomes nonfunctional. Other nodes switch immediately to emergency key set and start to use corresponding key pair in this set for signing and signature validation. This key pair will be in use in its validity period and nodes have to update their key sets before the expiration of the emergency key pair.

Considering report of another misbehaving node before the use of the new anonymity key set, length of the emergency period should be minimal. On the other hand, in such situation VANET nodes need to download new key sets from local CA in this period.

If nodes try to connect separately to the local CA to download new key sets in a small amount of time, performance issues may arise. To resolve this issue, local CA may encrypt the key sets with a shared key and broadcast the encrypted key set packet. In this case, nodes update their key sets without the need of a connection to the local CA. Shared key should be updated regularly and nodes, over a secure channel, download the shared key individually from local CA.

The mobility of nodes between regions managed by different local CAs is an important requirement in VANET. In this case, visiting node needs to download the active key sets in use from visited region's local CA. Local CAs are connected to each other by wired line and visiting node's identities are validated online. Visited region's local CA connects to the local CA that the visitor is registered to, and validates visiting node's identity

4.4 Communication Overhead

S3P protocol uses public-key cryptography to provide secure and privacy protecting communications in VANET. This introduces some problems concerning communication overhead. If RSA is chosen for producing encrypted packet and signature generation, a security message packet would grow rapidly.

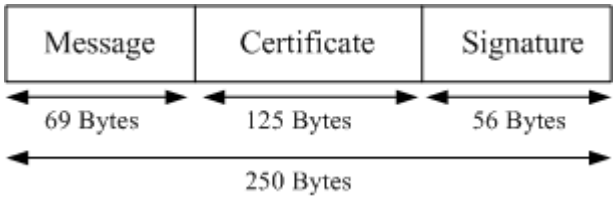

Figure 4.3 - Reference PKI based packet structure

Figure 4.3 shows the format of a signed message according to 39[21]. To be a better implementation, a proposed protocol should not at least exceed 250 bytes per packet. If RSA-2048 is chosen for encrypted packet and signature generation, the resulting packet would be as in Figure 4.4.

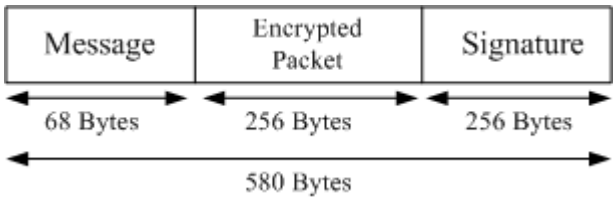

Figure 4.4 - S3P packet format with RSA-2048

To reduce the size of the packet we would adopt elliptic curve based signature schemes like ECDSA while continuing to generate encrypted packet with RSA-2048. We obtain the packet format in Figure 4.5. By using ECDSA for signature generation we reduced the size of the packet but it is still bigger than the size of reference packet format.

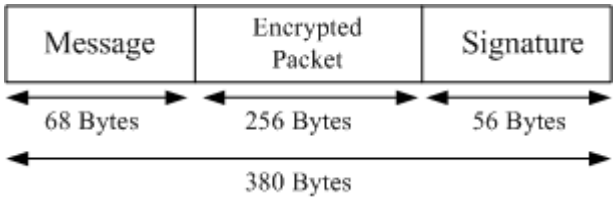

Figure 4.5 - S3P packet format with ECDSA

We may stop using RSA to produce encrypted packet and adopt using symmetric encryption scheme like AES. The resulting packet format would be like in Figure 4.6.

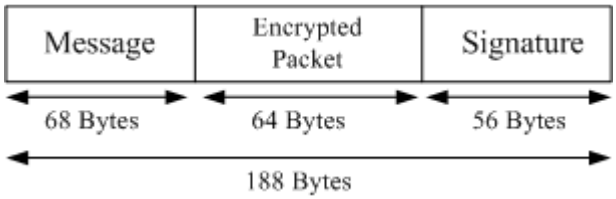

Figure 4.6 - S3P-Symmetric packet format

Using symmetric encryption for generating encrypted packet reduces the packet size but introduce another overhead. In this case every node registered to a localCA should share a symmetric key with that localCA. Key distribution does not introduce any complexity while the key management does. If the number of vehicles registered to a localCA increases, the number of symmetric keys that the localCA should maintain also linearly increases. When localCA needs to identify a vehicle from its safety message, it has to try every symmetric key until it finds the correct key. Assuming that the localCA would rarely be in need to identify a vehicle from its message this overhead may be ignored.

5 Protocol Evaluation

5.1 Evaluation against Security Requirements

In this section, our protocol is evaluated against the security requirements given in section 2.2.

Message authentication and integrity should be provided to assure the nodes that the message comes from valid source and has not been altered along the way. In the proposed protocol all messages are signed by the sender, thus receivers are able to check the validity of the sender. Signature operation also protects the message from being modified during the transmission.

In VANET, nodes should not be able to deny sending a message in order to provide liability. To provide *message non-repudiation*, every node, in this protocol, sending a message also signs the message with its own private key. This signature is used by the local CA to determine the node's identity. As the signature of vehicles is encrypted with the local CA's public key, only local CA is able to learn the node's identity.

Nodes receiving the safety messages, first validate the signature generated using the anonymity key (emergency key in case of emergency). Nodes also check the time information included in the message in order to ensure that the sender is alive when the message is sent. A node having the anonymity and the emergency key sets are considered by the system as an authenticated entity thus *entity authentication* is also provided by the protocol.

Tracking vehicles and collecting private information of a node should be prevented to provide *privacy protection*. Messages generated by a vehicle should not disclose the

sender's identity in order to keep vehicle anonymous. Our protocol utilizes the anonymity and the emergency key sets shared by all nodes in a region managed by a local CA for signing the safety messages. Two messages generated by a node also cannot be linked for tracking purposes as the key used to sign these messages is system wide. This mechanism allows vehicles to be anonymous as well as an authenticated system entity.

5.2 Formal Protocol Verification with ProVerif

ProVerif [20] is an automatic cryptographic protocol verifier which is based on a representation of the protocol by Horn clauses, in the formal model that is called Dolev-Yao model. By being able to use many different cryptographic primitives, including shared- and public-key cryptography (encryption and signatures), hash functions, and Diffie-Hellman key agreements, specified either as rewrite rules or as equations, ProVerif can handle an unbounded number of sessions of the protocol, even in parallel, and an unbounded message space.

By using ProVerif, it is possible to prove secrecy, authentication, strong secrecy and equivalences between processes that differ only by terms, mechanisms of a protocol [20].

The proposed protocol is modeled in spi-calculus and verified with ProVerif version 1.84 against secrecy of signature and encryption private keys, authentication, and message non repudiation requirements. Formal model of the protocol in spi-calculus is the following:

```
1. free net.
2. free timestamp.
3. fun host/1.
4. reduc getkey(host(x)) = x.
5. fun hash/1.
6. fun pk/1.
7. fun pencrypt/2.
8. reduc pdecrypt(pencrypt(x, pk(y)), y) = x.
```

```
9. fun sign/2.
10. reduc verify(sign(x,y), pk(y)) = x.
11. not skAnon. not skA. not skCA.
12. query evinj: endMessageAuth(x,y) ==> evinj: beginMessageAuth(x,y).
13. query ev: endIdDisclose(x,y) ==> ev:beginIdDisclose(x,y).
14. let processA = new m; new timestamp;
15. let ts = timestamp in
16. let h = hash((m, ts, hostA)) in
17. let ep = pencrypt((hostA, sign(h, skA)), pkCA) in
18. let hh = hash((m, ts, ep)) in
19. out(net, (m, timestamp, ep, sign(hh,skAnon)));
20. event beginMessageAuth(m, ts);
21. event beginIdDisclose(m, ts).
22. let processB =
23. in(net, (me, t, e, s));
24. let ts = timestamp in
25. if ts = t then
26. if hash((me, t, e)) = verify(s, pkAnon)then
27. event endMessageAuth(me, ts).
28. let processCA=
29. in(net, (me, t, e, s));
30. let ts = timestamp in
31. if ts = t then
32. if hash((me, t, e)) = verify(s, pkAnon)then
33. let (id, ss) = pdecrypt(e, skCA) in
34. if hash((me, t, id)) = verify(ss,getkey(id)) then
35. event endIdDisclose(me, t).
36. process
37. new skAnon; let pkAnon = pk(skAnon) in
38. out(net, pkAnon);
39. new skA; let pkA = pk(skA) in
40. out(net, pkA);
41. new skCA; let pkCA = pk(skCA) in
42. out(net, pkCA);
43. let hostA = host(pkA) in
44. out(net, hostA);
45. ((!processA)|(!processB)|(!processCA))
```

In lines 1 and 2, two public parameters, known by everyone including the attackers, are defined: *net* (communication) and *timestamp* (message generation time). Between lines 3 and 10, there are 5 functions are defined. The key word *fun* indicates the name of the function and parameter that it accepts while the key word *reduc* is used to define the inverse of that function. The following functions are defined in the mentioned lines:

1. *host*: This function is utilized for assigning a key to a host while its reduction *getkey* serves to obtain a key which is assigned to a specific host.

2. *hash*: Provides a hash value of a message given in as a parameter.

3. *pk*: Calculates the public key of the given private key.

4. *pencrypt*: With this function defines a public key encryption and encrypts the message given as a parameter with the given public key. Its inverse, *pdecrypt* is serves as a public key decryption utility.

5. *sign*: This function is used to produce signature over a message while its reduction, *verify*, provides the verification of the signature calculated over the given message.

In lines 11-13, the objectives of the verification process are given. Line 11 indicates that three private keys (skAnon – anonymity private key for signing the safety messages, skA – node A's private key, and skCA – local CA's private key) are needed to be kept secret and any attacker (insider or outsider) should not be able to obtain these keys. With this line ProVerif checks that the secrecy of the private keys are provided by the protocol and no attacker is able to obtain them. Line 12 means that for any safety message received there must be only one message sent by an authenticated network node. By receiving this line, ProVerif checks that the receivers in VANET are assumed any authenticated message is coming from a legitimate network node and all messages are fresh meaning that no replay attack is possible. In line 13, similar to line 12, ProVerif is told to check that for every message sent by a legitimate network node if it is possible to identify the sender. This check allows if the protocol fulfills the *non-repudiation* criterion.

Lines 14, 22, 28, and 36 define processes for ProVerif which may be considered as functions in traditional procedural programming languages. In line 14, *processA*, which represent a safety message sending legitimate VANET node A, is defined while in line 22 *processB*, a legitimate message receiving node B is declared. While the creation of processA, a new message *m* and new time information *timestamp* at which *m* is created, are instantiated by the definition of the processA. Similarly in line 28, *processCA*, a local CA trying to identify a network node from its valid safety message is defined. Finally, like main function of traditional procedural programming languages as C, unnamed process, for ProVerif to start examination, is introduced in line 36.

In processA, at first, timestamp, that is a global variable, like communication channel net, and reachable by any network entity including attackers, is retrieved and then first hash value *h*, over message *m*, timestamp *ts*, and identity of sender *hostA*, is calculated in lines 15 and 16 respectively. In line 17, identity of sender and the signature, calculated over first hash value *h* using A's private key, are encrypted using public-key encryption scheme with the local CA's public key. In 18, second hash value *hh* is computed over message *m*, timestamp and the encrypted packet *ep* which is the result of the previous line. In 19, A sends out the message, timestamp, ep and signature computed over second hash value *hh* using the anonymity signing key *skAnon* over the channel net. In lines 20 and 21, processA notifies the ProVerif that message authentication and non-repudiation verification checks are marked.

In processB, after receiving the message *me*, timestamp *t*, encrypted packet *e*, and signature *s* from channel net in line 23, firstly, timestamp is checked against replay attacks in lines 24 and 25, then signature *s* is verified using anonymity public key *pkAnon* in 26. If these checks pass, process notifies ProVerif that message authentication is successful by line 27.

In processCA, as in process, the message *me*, timestamp *t*, encrypted packet *e*, and signature *s* are received from channel net in 29. After checks on message freshness in 30 and 31, *s* is verified using anonymity public key *pkAnon* in 32. In 33, by decrypting encrypted packet *e* utilizing localCA's private key, sender's id and signature *ss* are obtained. By using sender's id to find its public key, the signature *ss* is verified in 34 and if this check passes, in 35, processCA notifies ProVerif that the message non-repudiation mechanism works as expected.

Finally, in the main process, in lines 37, 39, and 41, new private keys are created and their corresponding public keys are derived using *pk* function defined in line 6. To announce the public keys, in lines 38, 40, and 42 these public keys are sent out to the public channel net. Next, host A's public key is associated with its identity and announced in lines 43 and 44. The most important part of the process lies in line 45 which means that processA, processB and processCA can be run in parallel for unlimited number. Each of these processes should run at least once but can run multiple times until ProVerif decides that no further analysis is necessary.

After running ProVerif version 1.84 with this model, the proposed protocol is found secure in terms of private key secrecy, message and entity authentication, and message non-repudiation.

5.3 Performance Evaluation of S3P

In this section, we use the ns-3 simulator to evaluate the performance of S3P in terms of the message loss ratio and the message end-to-end delay compared with the group signature scheme in [13] and the standard PKI-based signature scheme in [21]. During simulation safety messages are sent, every 300 ms by each vehicle, on IEEE 802.11a which is used as the medium access control layer transmission protocol. The bandwidth of the channel is taken as 6 Mbps while communication range of IVC is

200m. Equations for calculating average message loss ratio and average message delay are adopted from [15].

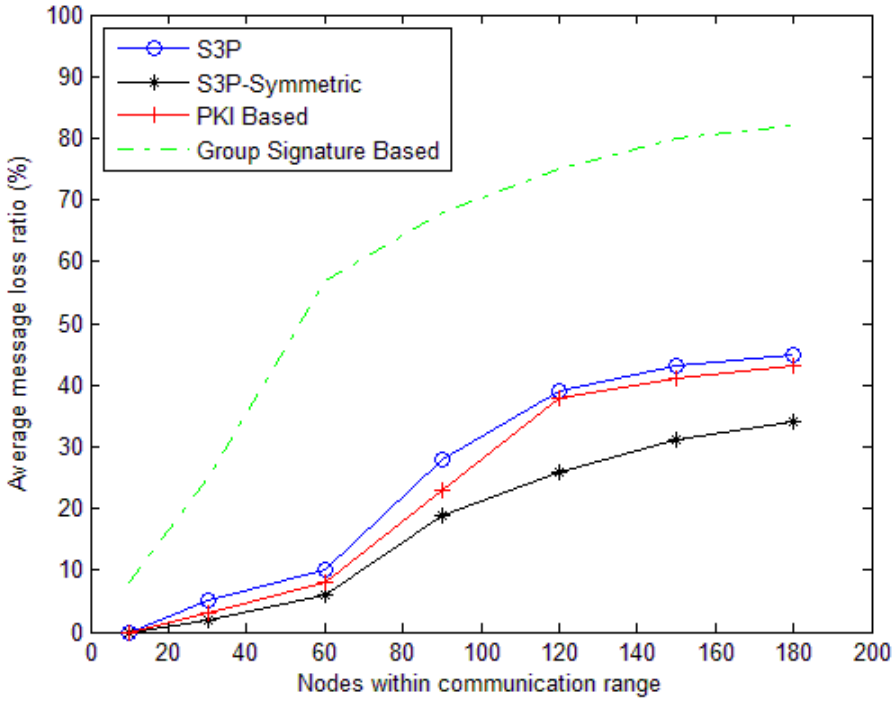

Figure 5.1 - Average message loss ratio

5.3.1 Message Loss Ratio

Average message loss ratio (ML) is defined in equation (1), where *N* represents total number of vehicles in the simulation. Total number of messages received in the medium access control layer by the vehicle *i* is represented by M_{mac}^{i} while the total number of messages consumed by the vehicle *i* in the application layer is denoted by M_{app}^{i}.

$$ML = \frac{1}{N}\sum_{i=1}^{N}\left(M_{app}^{i}/M_{mac}^{i}\right) \quad (1)$$

Figure 5.1 shows the relationship between the message loss ratio and the traffic load. Here traffic load is represented by the number of nodes within communication range. The group signature scheme has the highest loss ratio due to need for high computation resources in the application layer. Other schemes has similar loss ratios as they possess nearly same characteristics and only small differences like payload size and certificate control mechanisms play role in average message loss ratio. The better scheme for loss ratio is the S3P-Symmetric as it provides symmetric encryption to reduce payload size.

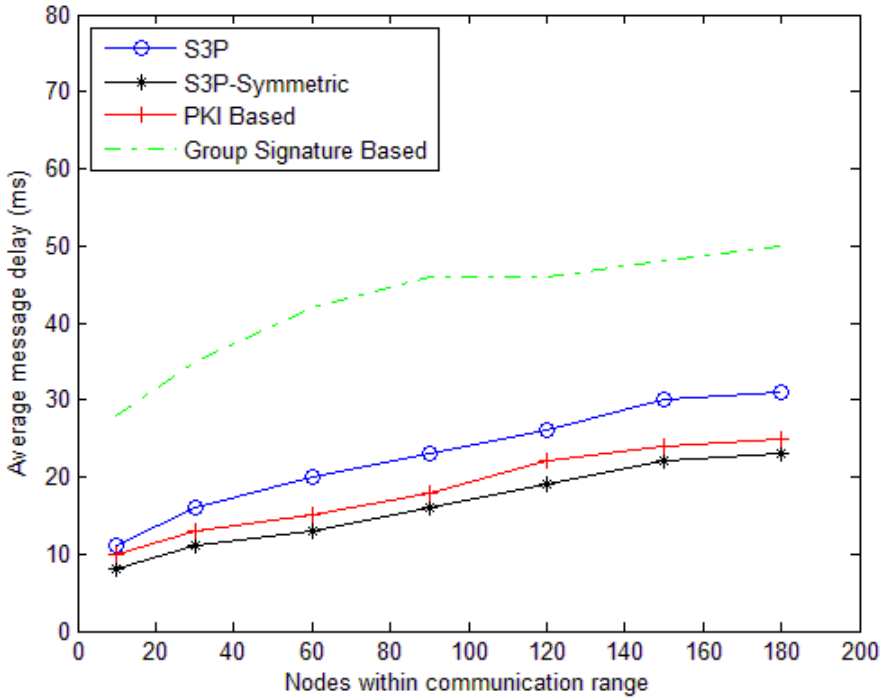

Figure 5.2 - Average message delay

5.3.2 Message Delay

Average message delay (ML) is defined in equation in (2), where *N* represents the total number of vehicles in the simulation. *M* is the number of messages sent by the vehicle *i* and *K* is the number of adjacent vehicles within the communication range of vehicle *i*. $T_{recv}^{i,k,m}$ represents the moment that the vehicle *k* in the application layer, receives the *m*th

message from the vehicle *i*. $T_{recv}^{i,k,m}$ represents the moment that the vehicle *i* in the application layer sends the *m*th message to the vehicle *k*.

$$MD = \frac{1}{N}\sum_{i=1}^{N}\frac{1}{MK}\sum_{m=1}^{M}\sum_{k=1}^{K}(T_{recv}^{i,k,m} - T_{send}^{i,k,m}) \quad (2)$$

In Figure 5.2 it can be seen that the group signature scheme has highest message delays and can be considered the worst of these four schemes in terms of message delay times. The reason behind the highest message delays of group signature scheme lies in complex and time consuming signature verification mechanism. The rest three schemes have again similar results and S3P-Symmetric leads from message delay point of view.

6 Conclusion

In this work, a novel privacy protecting and secure protocol for vehicular ad hoc networks has been proposed along with the examination of existing solutions. By using shared key sets and PKI techniques, this protocol allows nodes to preserve their anonymity and achieves the message non-repudiation. Certificate management and revocation does not introduce any computational and communication overhead. Mobility of VANET nodes between regions is also provided without any complexity and security vulnerability. Proposed scheme is analyzed against the security requirements and, as it fulfills the all criteria, evaluated as successfully implementable.

References

[1] M. Raya, and J.P. Hubaux, "The security of vehicular ad hoc networks" in *Proc. 3rd ACM Workshop on Security of ad hoc and Sensor Networks*, Alexandria, pp. 11-21, (2005).

[2] M. Raya, P. Papadimitratos, and J.P. Hubaux, "Securing Vehicular Communications", *IEEE J. Wireless Communication*, vol. 13, pp. 8-15, (2006).

[3] M. Raya, and J.P. Hubaux, "Securing vehicular ad hoc networks", *J. Computer Security*, Special Issue on Security, Ad Hoc and Sensor Networks, vol. 15, no. 1, pp. 39–68, (2007).

[4] X. Lin, R. Lu, C. Zhang, H. Zhu, P.H. Ho, and X. Shen, "Security in vehicular ad hoc networks", *IEEE Communications Magazine*, vol. 46, pp. 88-95, (2008).

[5] F. Dötzer, "Privacy Issues in Vehicular Ad Hoc Networks" *Privacy Enhancing Technologies*, Lecture Notes in Computer Science, Springer, Berlin/Heidelberg, vol. 3856, pp.197-209, (2006).

[6] L. Huang, K. Matsuura, H. Yamane, and K. Sezaki, "Enhancing wireless location privacy using silent period," in *IEEE Wireless Communications and Networking Conference (WCNC)*, pp. 1187–1192, (2005).

[7] K. Sampigethaya, L. Huang, M. Li, R. Poovendran, K. Matsuura, and K. Sezaki, "CARAVAN: providing location privacy for VANET" in Workshop on Embedded Security in Cars (ESCAR), (2005).

[8] P. Papadimitratos, L. Buttyan, J.P. Hubaux, F. Kargl, A. Kung, M. Raya, "Architecture for Secure and Private Vehicular Communications" in *7th Int. Conf. on ITS*, ITST'07, Sophia Antipolis, France, pp. 1-6, (2007).

[9] G. J. Freudiger, M. Raya, and M. Feleghhazi, "Mix zones for location privacy in vehicular networks" in *Proc. First Int. Workshop on Wireless Networking for Intelligent Transportation Systems (WiN-ITS'07),* Vancouver, Canada, (2007).

[10] L. Buttyán, T. Holczer and I. Vajda, “On the Effectiveness of Changing Pseudonyms to Provide Location Privacy in VANETs”, *Security and Privacy in Ad-hoc and Sensor Networks*, Lecture Notes in Computer Science, Springer, Berlin / Heidelberg, vol. 4572, pp.129-141, (2007).

[11] P. Papadimitratos, L. Buttyan, T. Holczer, E. Schoch, J. Freudiger, M. Raya, M. Zhendong, F. Kargl, A. Kung, J.P. Hubaux, “Secure vehicular communication systems: design and architecture”, *IEEE J. Communications Magazine,* vol. 46, issue 11, pp. 100-109, (2008).

[12] G. Calandriello, P. Papadimitratos, J.P. Hubaux, A. Lioy, “Efficient and Robust Pseudonymous Authentication in VANET” in *Proc. 4th ACM int. Workshop on Vehicular Ad Hoc Networks*, Montreal, Quebec, Canada, pp. 19-28, (2007).

[13] X. Lin, X. Sun, P.-H. Ho, and X. Shen, “GSIS: a secure and privacypreserving protocol for vehicular communications” *IEEE Transaction on Vehicular Technology*, Vol. 56, No. 6, pp. 3442-3456, (2007).

[14] J. Guo, J.P. Baugh and S. Wang, “A group signature based secure and privacy-preserving vehicular communication framework”, in *Mobile Networking for Vehicular Environments*, pp. 103–108, (2007).

[15] C. Zhang, X. Lin, R. Lu and P.-H. Ho, “RAISE: an efficient rsu-aided message authentication scheme in vehicular communication networks”, in Proc. IEEE ICC 2008, Beijing, China, May 19-23, (2008).

[16] Car 2 Car Communication Consortium, http://www.car-2-car.org/ (2010).

[17] Secure Vehicular Communication, http://www.sevecom.org/ (2010).

[18] Dedicated Short Range Communications, 5.9 GHz DSRC. http://grouper.ieee.org/groups/scc32/dsrc/ (2010).

[19] "IEEE 1609 - Family of Standards for Wireless Access in Vehicular Environments (WAVE)", (2010).

[20] http://www.proverif.ens.fr/, (2010).

[21] “IEEE Standard 1609.2 - IEEE Trial-Use Standard for Wireless Access in Vehicular Environments - Security Services for Applications and Management Messages”, (2010).

Biographical Sketch

Ali Osman Bayrak was born in İzmir in 27 May 1979. He is graduated from İzmir Buca Şirinyer High School in 1997. Between 1998 and 2004 he was a student at the Computer Engineering Department of Galatasaray University where he had his B.Sc. He is a student of Master of Computer Enginering at the University of Galatasaray. In 2004 he joined the Galatasaray University as research assistant at the Faculty of Engineering and Technology, where he worked till 2005. After the military service, in 2006 he started to work at the Cryptographic Research Center of National Research Institute of Electronics and Cryptology (TÜBİTAK – UEKAE).