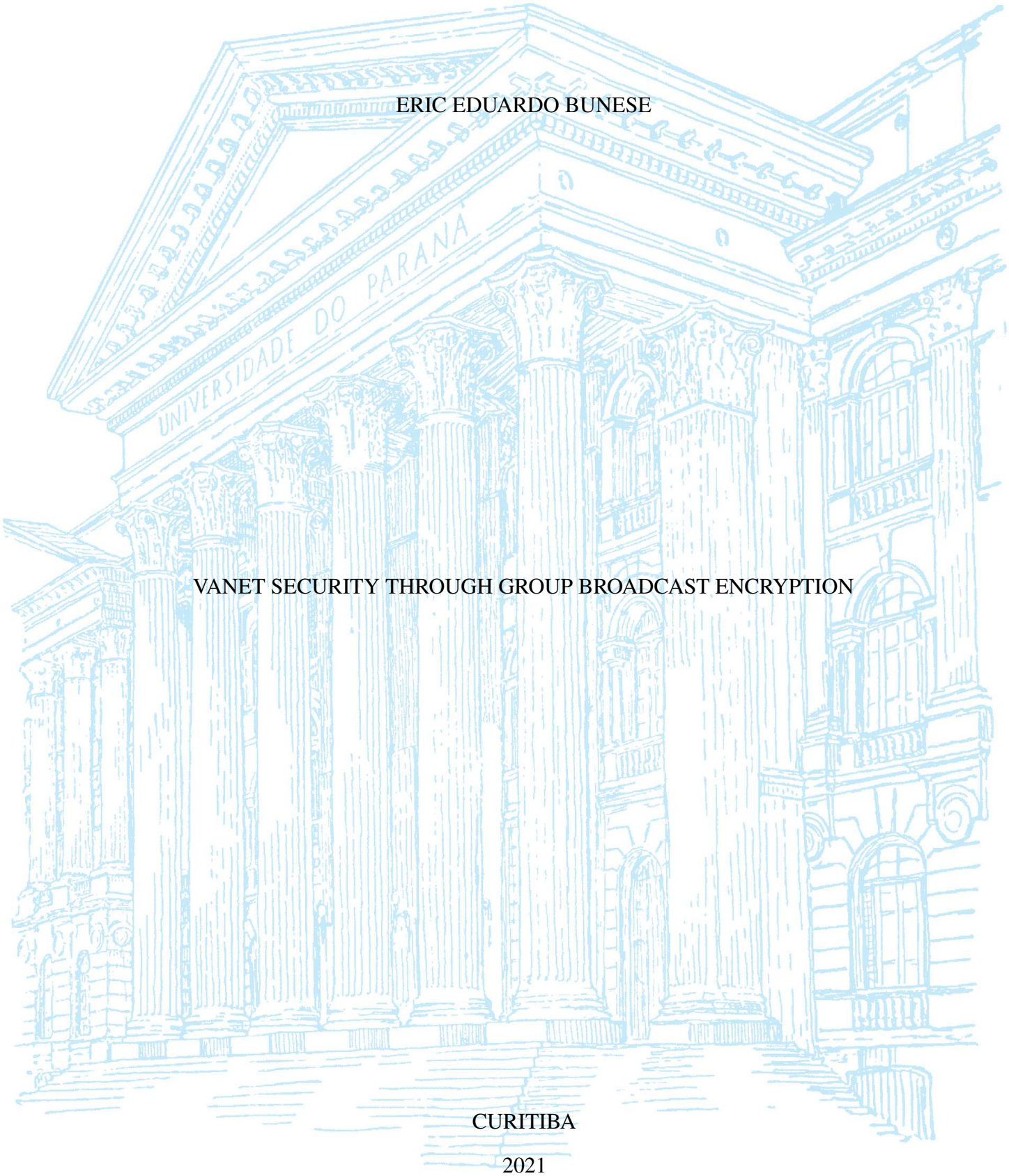
UNIVERSIDADE FEDERAL DO PARANÁ

ERIC EDUARDO BUNESE



VANET SECURITY THROUGH GROUP BROADCAST ENCRYPTION

CURITIBA

2021

ERIC EDUARDO BUNESE

VANET SECURITY THROUGH GROUP BROADCAST ENCRYPTION

Dissertação apresentada como requisito parcial à obtenção do grau de Mestre em Informática no Programa de Pós-Graduação em Informática, Setor de Ciências Exatas, da Universidade Federal do Paraná.

Área de concentração: *Ciência da Computação*.

Orientador: Luiz Carlos Pessoa Albini.

Coorientador: Eduardo Todt.

CURITIBA

2021

CATALOGAÇÃO NA FONTE – SIBI/UFPR

B942v Bunese, Eric Eduardo

Vanet security through group broadcast encryption [recurso eletrônico]/ Eric Eduardo Bunese – Curitiba, 2021.

Dissertação (Mestrado) – Programa de Pós-Graduação em em Informática, Setor de Ciências Exatas da Universidade Federal do Paraná

Orientador: Prof. Dr. Luiz Carlos Pessoa Albini

1. Criptografia de dados. 2. Tecnologia da informação. 3. Sistema de telefonia celular. I. Albini, Luiz Carlos Pessoa. II. Título. III. Universidade Federal do Paraná.

CDD 004.6

Bibliotecária: Vilma Machado CRB9/1563

TERMO DE APROVAÇÃO

Os membros da Banca Examinadora designada pelo Colegiado do Programa de Pós-Graduação em INFORMÁTICA da Universidade Federal do Paraná foram convocados para realizar a arguição da Dissertação de Mestrado de **ERIC EDUARDO BUNESE** intitulada: **VANET SECURITY THROUGH GROUP BROADCAST ENCRYPTION**, sob orientação do Prof. Dr. LUIZ CARLOS PESSOA ALBINI, que após terem inquirido o aluno e realizada a avaliação do trabalho, são de parecer pela sua APROVAÇÃO no rito de defesa.

A outorga do título de mestre está sujeita à homologação pelo colegiado, ao atendimento de todas as indicações e correções solicitadas pela banca e ao pleno atendimento das demandas regimentais do Programa de Pós-Graduação.

CURITIBA, 23 de Fevereiro de 2021.

Assinatura Eletrônica
24/02/2021 10:53:33.0
LUIZ CARLOS PESSOA ALBINI
Presidente da Banca Examinadora (UNIVERSIDADE FEDERAL DO PARANÁ)

Assinatura Eletrônica
24/02/2021 12:43:41.0
PAULO ROBERTO DE LIRA GONDIM
Avaliador Externo (UNIVERSIDADE DE BRASÍLIA)

Assinatura Eletrônica
24/02/2021 08:37:00.0
CARLOS ALBERTO MAZIERO
Avaliador Interno (UNIVERSIDADE FEDERAL DO PARANÁ)

To Porthos and Judy.

ACKNOWLEDGEMENTS

This work was partially supported by CAPES. Our thanks and acknowledgements go to this institution.

RESUMO

Redes vehiculares Ad hoc (VANETs) são uma especialização de redes móveis Ad hoc (MANETs), aplicadas a veículos como carros, trens e ônibus. Estas redes são implementadas sobre uma camada de comunicação sem fio, como por exemplo Bluetooth, Wi-Fi, 4G ou até mesmo 5G. É possível desfrutar de diversas aplicações a partir da comunicação entre veículos, desde melhoria na segurança de transeuntes até a inclusão de funcionalidades de conveniência social ao tráfego diário. Porém, funções de segurança precisam de uma rede de alta velocidade, que dificulta muito o cenário sem fio. Entregar tantas mensagens em tão pouco tempo já é um desafio, porém, adicionar uma camada de segurança da informação e verificação pode lesar a rede significativamente. Neste trabalho, um estudo foi conduzido para avaliar a possibilidade de utilizar uma solução híbrida entre criptografias simétricas e assimétricas para comunicação veicular, apresentando a proposta de utilização de Group Broadcast Encryption (Criptografia de grupo) como solução de segurança para VANETs, trazendo um desempenho mais próximo da utilização de criptografia simétrica, diminuindo o total de mensagens necessárias para trafegar em rede, e assim, o tempo de resposta. Simulações foram preparadas e executadas utilizando o The ONE, e foram feitas comparações do uso de três algoritmos de criptografia para VANETs. Os dados resultantes apresentam a viabilidade do uso de criptografia em grupo (*Group Broadcast Encryption*) para simplificar a fase de segurança da informação, reduzindo a capacidade de armazenamento e diminuindo de forma significativa o número de mensagens na rede.

Palavras-chave: Vehicular network, cellular, cryptography, security

ABSTRACT

Vehicular Ad-Hoc networks (VANETs) are a specialized type of MANET, applied to vehicles such as cars, trains and buses, implemented on top of wireless communication protocols such as Bluetooth, Wi-Fi, 4G or 5G. There are many different applications when connecting vehicles, ranging from improving the safety of commuters to adding social convenience to everyday traffic. However, safety functions require high speed networks, and add great weight to the wireless scenario. Delivering several messages in such a small amount of time is already a challenge, however, adding security and verification layer burdens the network into failing this task. This article demonstrates the feasibility of using a hybrid solution between symmetric and asymmetric cryptography to allow safe vehicular communications. In this work, we present the possibility of using Group Broadcast Encryption as a security solution for VANETs, thus, achieving a better performance of the same order as using symmetric cryptography, by decreasing the number of messages in the network, and in consequence, response times. Simulations were set-up and run using The ONE, comparing the usage of three different cryptography layers for VANETs. The resulting data promises that group broadcast encryption can be used to simplify the encrypting phase, reduce required storage and significantly decrease the number of messages in the network.

Keywords: Vehicular network, cellular, cryptography, security

LIST OF FIGURES

LIST OF TABLES

LIST OF ACRONYMS

MANET	Mobile Adhoc Network
VANET	Vehicular Adhoc Network
V2V	Vehicle-to-vehicle communication
V2I	Vehicle-to-infrastructure communication
RSU	Road-side Unit
OBU	Onboard Unit
GBE	Group Broadcast Encryption
The One	The Opportunistic Network Environment Simulator
WDMM	Working Day Movement Model

LIST OF SYMBOLS

V	Set of all nodes
n	Any given node of V
S	Sub-set of reachable trustworthy nodes of V by n, where $S \subset V$
$N_{\mathcal{F}}$	Founder nodes
m	Number of Founder nodes
IC	Cost to initialize group
N_i	Identification of node i
SK_i	Private key of node i
PK_i	Public key of node i
MSK	Master private key of system
$size(f_m^i(x))$	Size of each sub-share of the MSK generated by nodes
MPK	Master public key of system
MSK_i	Share of master private key hold by node i
Ω	Size of subset of Founder nodes contacted by a joining node
Δh	Average number of hops between nodes
NM	Cost of a new member joining the group
NR	Cost of key revocation

CONTENTS

1 INTRODUCTION

Mobile Ad hoc networks, MANETs, are a specific type of network which is not limited by infrastructure (Stojmenovic, 2002). This means that no sort of access points, routers or any device that is not considered a node is necessary to the network. All of the work load and exchanged messages in this network should be handled by participating nodes, which can be sensors, mobile phones or even vehicles. When vehicles are communicating wirelessly, the network can be defined as a Vehicular Ad hoc network, or, VANET. Since vehicles are moving around roads in a city or across the country, the network topology is highly dynamic and complex (Karagiannis et al., 2011; Engoulou et al., 2014; Mejri et al., 2014; Hasrouny et al., 2017).

Connecting vehicles with each other is the foundation for safety and convenience features, such as traffic prediction, accident prevention or emergency notifications (Hartenstein e Laberteaux, 2008; Karagiannis et al., 2011; Mejri et al., 2014; Bariah et al., 2015). However, VANET applications require constant position and direction updates from vehicles in the vicinity, meaning that there is a lot of sensitive user information being shared at all times. In order to implement VANET applications in the market, VANET security needs to be revised, updated and endorsed (Mishra et al., 2016; Deeksha et al., 2017; Hasrouny et al., 2017; Ali et al., 2019).

A VANET can be implemented by equipping vehicles with a network unit called the *OBU*, or On Board Unit. This computing component enables wireless communication with other nearby nodes, using a Wi-Fi, Bluetooth, 4G or 5G link. It's also very common for OBUs to be tamper-proof, in order to add another layer of hardware security to the network (Hartenstein e Laberteaux, 2008; Karagiannis et al., 2011; Mejri et al., 2014; Hasrouny et al., 2017). Vehicles are not necessarily the only nodes participating in VANETs, as similiar, static Road-side Units, *RSUs* can be implemented along the roads, acting as trusted management or routing links. Finally, VANETs can also be connected to the Internet, using longer, slower, and more distant forms of wireless communications already implemented in mobile devices. Internet integration also enable further security implementations, as trusted authentication servers can be contacted and queried (Hartenstein e Laberteaux, 2008; Sun et al., 2010; Hao et al., 2011; Karagiannis et al., 2011; Bariah et al., 2015; Tzeng et al., 2017).

Considering the types of nodes that can be connected to a VANET, there are two types of communications that can be executed in the network. *Vehicle-to-vehicle communication* (*V2V*) is used for position and direction updates, traffic and accident prediction and emergency notifications between pairs or groups of vehicles. Meanwhile, *Vehicle-to-infrastructure* (*V2I*) is necessary when a vehicle is in communication with a static node, or a RSU. Such messages are usually used for authenticity checking, signature validation, group connection and other VANET management information (Hartenstein e Laberteaux, 2008; Karagiannis et al., 2011; Engoulou et al., 2014; Mejri et al., 2014; Bariah et al., 2015).

When taking part in a VANET environment, vehicles will be constantly sharing and receiving their positions, direction, and possibly other private or personal information which need to be processed, verified, and used as input for traffic and accident prediction algorithms. Ensuring that the content of these messages is made private, pristine, and legitimate is essential for the network. User privacy must be guaranteed so that consumers will want VANET solutions to be implemented in their purchases, and that any private or personal information is secure from exploiters or attackers. Any content that is shared in the network must be authentic and verified, so that deviant users do not attempt to spread misinformation for their personal gain. This leads

to the conclusion that cryptography and security algorithms are a requirement for such a product (Bariah et al., 2015; Mishra et al., 2016; Deeksha et al., 2017).

Vehicular networks require a high speed, low latency network foundation in order to enable position sharing and traffic safety applications, as vehicles need to send, on average, 10 messages per second, with a critical delay of 100ms before the information becomes pointless (Qu et al., 2015; Mishra et al., 2016). Routing and ensuring the delivery of these messages in such a scenario is already challenging, and adding a security layer that will encrypt, sign and process every message will burden the network to fail its mission.

In order to understand the security implementations required to a VANET, it's important to comprehend a few of the different types of attacks and attackers in wireless and vehicular networks (Engoulou et al., 2014; Bariah et al., 2015; Qu et al., 2015; Al Hasan et al., 2016; Mishra et al., 2016; Deeksha et al., 2017). Table 1.1 presents the four different types of attackers in VANETs, characterized in two orthogonal classes: level of activity and participation in the network.

Table 1.1: Types of attackers

	Active	**Passive**
Internal	Participate in the network by sending and receiving data from other nodes.	Participate in the network, but only read and extract data from within.
External	Create input for nodes from outside the network, and are able to intercept messages to obtain data.	Intercept data in the network, usually known as a man-in-the-middle.

Internal attackers are actual signed-in users in a VANET, connected to the network in an existing, or spoofed vehicle. They can fully participate in the network, by sending and receiving messages. In general, internal attackers can use the network to their advantage, and will try to spread havoc by sending false information.

External attackers are not part of the VANET. They will attempt to break the security layers and gain information from the outside, or, will create false sensorial input to active VANET nodes, that can misinterpret the information and spread false information without even knowing.

Active attackers are always trying to create and spread faselhood in the VANET. In order to reroute vehicles, create traffic jams, or even abduct vehicles in traffic.

Passive attackers will try to gain information from the network, that can be exploited through blackmail or simply breaking user privacy.

Table 1.2: Primitives of security

Primitive	Description	Available Mechanisms
Privacy	Connected nodes only need to share their pseudonym with each other, while the identity is only exchanged between the soliciting node and the trusted certificate authority. No other personal or private information can be derived from the pseudonym.	Asymmetric cryptography can ensure user privacy.
Non Repudiation	It is not possible to repudiate or deny any previous behavior on the network, as only the owner of the private key can use it to sign and authenticate messages.	Signing messages and asymmetric cryptography can ensure non repudiation.
Availability	The network must be available and all the dependencies must be accessible at any time.	Network infrastructure and trusted authorities help protect availability.
Integrity	Exchanged messages must be received in pristine condition, as they cannot be modified.	Signing messages can guarantee Integrity.
Authenticity	Exchanged messages can never be modified. Every message must be verified along with its origin.	Signing messages can guarantee Authenticity.

Table 1.2 presents five security properties that help prevent and detect these attackers, usually known as the Primitives of Security (Mejri et al., 2014; Bariah et al., 2015; Mishra et al., 2016; Deeksha et al., 2017; Hasrouny et al., 2017). As presented in Table 1.2, cryptography is a building block that ensures the five properties, and may enable proper VANET security. However, the dynamic network topology and latency sensitive information transmitted in the VANET contribute to a more vulnerable environment, as it is not viable to apply market standard security implementations (Mejri et al., 2014; Bariah et al., 2015; Mishra et al., 2016; Deeksha et al., 2017; Hasrouny et al., 2017).

There are two main types of cryptography that can be applied to vehicular networks in order to provide security; Symmetric and Asymmetric. The main difference between the two concepts is the cardinality of keys, and how nodes are supposed to cipher and decipher messages using them.

Symmetric cryptography solutions require that two or more nodes share a common key to cipher and decipher messages. Once a pair of nodes has agreed upon a key, they apply the algorithm and key to the message generating a new string to be sent in the network. In VANETs, it is not practical to store and apply a different key for every neighboring node in the network,

as it might be necessary to cipher a single message dozens of times. When using symmetric cryptography for many different nodes, storing and identifying which key should be used can also become a challenge (Mejri et al., 2014; Qu et al., 2015; Deeksha et al., 2017).

It's important to note that sharing the symmetric encryption key between more than two nodes is a major security fault, as a single maleficent node can simply leak the protected key to other entities, usually through a different communication medium. While avoiding or protecting this security fault is not part of this research's scope, this solution is considered for efficiency comparisons in Chapter 4.

Current market algorithms for asymmetric cryptography such as RSA, ECDMA can be applied to the VANET context. Vehicles can share their public key, which will be used to cipher information to be sent in the network. Whenever a destination node receives a message, it can attempt to decipher the information using its private key. Managing the VANET on an asymmetric environment might be a challenge, though. Having a key that must be used in order to cipher information to every destination node means using a lot of storage and processing power spent on securing the same information. Doing so dozens of times for every outgoing message will become a burden and generate delays in the network. However, when deciphering messages, nodes only need to store and utilize their own private keys, meaning that the cardinality of received messages is always one. Several solutions created for securing VANETs rely on using asymmetric cryptography (Sun et al., 2010; Jin e Papadimitratos, 2015; Qu et al., 2015; Tzeng et al., 2017; Deeksha et al., 2017).

Another candidate cryptography solution for vehicular networks might be *Group Broadcast Encryption*. This framework merges the benefits of both symmetric and asymmetric algorithms without further drawbacks. It's possible to create an asymmetric-like environment, so that every node will have its own private and public keys, with a performance that is closer to that of the symmetric algorithms, as the key cardinality is not one to one. Nodes build their private keys using other nodes's public keys. Whenever a message is encrypted, every node that had its public key used in the cipher will be able to decipher and read the message. Group Broadcast Encryption, (da Silva e Albini, 2013) is very similar to existing Group Based solutions such as (Hasrouny et al., 2015; Ullah et al., 2017; Lim et al., 2017; Zhang et al., 2018, 2019), and could provide a more decentralized, distributed environment for VANET security.

The goal of the conducted work is to adapt, evaluate and test the feasibility of implementing a Group Broadcast Encryption framework, as presented by (da Silva e Albini, 2013) in the VANET environment. The main hypothesis of the conducted work is that it should be possible to improve performance in cryptographic computations for vehicular networks by cutting the usage of superfluous messages in the wireless environment. Group Broadcast Encryption should be very efficient in this scenario, as it provides a symmetric-like interface with asymmetric-like security. Network simulations were run using The ONE, comparing three different cryptography algorithms on top of a VANET. Simulation results demonstrate that group broadcast encryption presents asymmetric-like security, with a symmetric-like performance, decreasing encryption and decryption times as well as the number of messages in the network.

The remaining of this article include a VANET security review and related work (Chapter 2), the proposed solution using Group Broadcast Encryption and its cost analysis (Chapter 3), simulation setup, algorithm comparison, discussion and simulation results (Chapter 4), followed by the conclusions (Chapter 5).

2 VANET SECURITY

Vehicular Adhoc Networks compose one piece of Intelligent Transportation Systems *ITS*, in addition to the growing field of sustainable, self driving vehicles and the Internet of Things. Since VANETs extend general Mobile Adhoc Networks, they inherit existing problems and vulnerabilities regarding the security of information and the privacy of its users. Despite enabling many different traffic safety and convenient solutions for commuters and general transportation, the popularization of this emerging field relies on its security and privacy (Hartenstein e Laberteaux, 2008; Karagiannis et al., 2011; Engoulou et al., 2014; Bariah et al., 2015; Qu et al., 2015; Al Hasan et al., 2016; Mishra et al., 2016; Deeksha et al., 2017; Ali et al., 2019).

When constructing security solutions, the level of security will be inversely proportional to the implementation performance. Given the highly loaded, dynamic and fast moving characteristics of VANETs, achieving the ideal balance between security and performance is imperative. This chapter presents a literature review on VANET security implementations, and how they balance performance and security.

A full system VANET security implementation is proposed by (Whyte et al., 2013). In their article, the authors divide VANET communication into four fundamental use cases, presented in Table 2.1. Despite being incomplete, the work presented by (Whyte et al., 2013) lays out standard procedures for defining a VANET protocol, that supports the solution to be presented in Chapter 3.

Table 2.1: Fundamental VANET use cases

Use Case	Description
1. Bootstraping	Nodes and other devices must be authenticated through a centralized Certificate Authority.
2. Pseudonym Distribution	Every node will be assigned a unique, untraceable pseudonym, in order to ensure a secure privacy layer.
3. Misbehavior Reporting	However incomplete at the time of this study, this use case should provide an interface for reporting misbehaviors in the network to the centralized authority servers.
4. Audit	Received misbehavior reports are verified and analyzed by the centralized authority, that is enabled to take preemptive or investigative action. This use case was also incomplete at the publication of this study.

The first two use cases defined by (Whyte et al., 2013) create an entry layer for nodes, that need to be authenticated in order to participate in the secure communications channel. Bootstrapping and pseudonym distribution ensure that all nodes agree on communication protocols, channels, encryption and hashing functions, and each will have an untraceable public pseudonym.

In order to understand how the technology can be made safer, it is important to acknowledge current vulnerabilities and exploitations that can be used against VANETs. Table

1.1 presents some types of passive attacks that can be pulled on wireless and vehicular networks. This types of attacks are characterized by a "read-only" policy, in which attackers will gain or acquire access to the confidential information that is sent in the network, but will not actively participate in the communication protocols, acting as a sort of spy or ghost towards the other nodes (Engoulou et al., 2014; Mejri et al., 2014; Bariah et al., 2015; Qu et al., 2015; Al Hasan et al., 2016).

Table 2.2: Passive attacks on VANETs

Attack Name	**Description**
Snooping	Reading other node's data by gaining unauthorized access into their medium of communication.
Traffic analysis	Passively analysing the network traffic, requests, responses and updates.
Position tracking	Gathering another vehicle's position updates and tracking it into a physical car, breaking privacy.

Since passive attacks usually require network access, they can be prevented by using cryptography. When a ciphered message is intercepted in the medium, an attacker will have to bruteforce or execute cryptanalysis on the collected information in order to discover its content. Both methods require heavy computation and mathematical thinking, and usually take much longer than the validity of said information to the attacker. However, the safe encrypted layer should only be set-up over a secure and validated access medium that implements proper authentication for incoming nodes. In other words, attackers should not be able to exploit the connection phase of the communications protocol to gain a valid encryption key (Bariah et al., 2015). The authentication layer should also rely on existing network security solutions, to prevent attackers from reverse engineering the connection phase, and gaininig illegitimate access to the encrypted layer. Table 2.3 presents the three layers of network security.

Table 2.3: Network Security Layers

Layer	**Description**
Encrypted Layer	Communication should be done on top of encrypted channel.
Authentication Layer	Required step on the communications protocol for accessing the Encrypted Layer.
Wireless Medium	General, unsafe wireless channel used for communications.

In order to implement cryptography and prevent the usual types of passive attacks, such as presented in Table 1.1, the communications protocol must implement an authentication process, so that new nodes can be verified and protected. Having a global, centralized authority allows for a controller pseudonym distribution that enables better user privacy and security audits, providing non repudiation and investigative tools for the competent entities.

The second type of attack in wireless networks and VANETs is the active attack. In this scenario, the malign entities gain or acquire an entry in the encrypted layer of the network, and will use it to disrupt or spread misleading information to other nodes. Table 2.4 shows a few

different types of active attacks on VANETS (Engoulou et al., 2014; Mejri et al., 2014; Bariah et al., 2015; Qu et al., 2015; Al Hasan et al., 2016).

Table 2.4: Active attacks on VANETs

Attack	Description
Replay	Repeating previous verified messages from the network to recreate past scenarios. Replaying accident notification messages, for example.
Sybil	Emulating several different nodes in a single vehicle, virtually flooding the VANET and consuming other vehicles' resources.
Jamming	Disrupting a healthy signal using strong interference.
Spamming	Sending useless information to other nodes on a constant pace.
Denial of service	Using one or several nodes in the network to constantly consume bandwidth and resources by sending an abnormal amount of useless messages.
Timing	Deliberately slowing VANET response times by failing to forward messages to neighboring nodes.
Illusion	Creating false information regarding road safety information, such as virtually spawning a road hazard to slow down other vehicles.
Network control	Gaining control of the majority of the network, ruling out usually benign distributed decisions.

Unlike passive attacks, it is not possible to prevent or stop active attacks from happening by implementing cryptography, as such incursions can still occur in the encrypted layer, presented in Table 2.3. In order to detect and prevent active attacks, it is possible to improve the authentication layer, or even implement reactive protocols that can detect and revoke misbehaving nodes. Trust management is also a possibility for such a reactive protocol, and is better detailed in 3.2.

Table 2.5: Attacking the VANET's security primitives

Primitives	Description
Availability and Non-repudiation	Attacking the availability generally means the network is made unsustainable to one or many nodes. Generally, the availability is connected to the network's traceability, and attacking it can impact non-repudiation and the network confidence. Some common availability attacks are Denial of Service, Jamming, Spamming and Black Hole Attacks.
Authenticity and Privacy	Attacking authenticity and privacy of a network implies that the content shared within it is untrustworthy, and that other connected nodes are possibly unverified and malicious. Some common atacks are Sybil, Replay, Spoofing, Position faking, Key replication, and message tampering.
Confidentiality	Even in an encrypted connection, every node is capable of breaking confidentiality by sharing exchanged information to a different medium. Some other common confidentiality attacks are Eavesdropping, location tracking and Traffic analysis.

Table 2.5, derived from (Mejri et al., 2014) presents how different types of attacks interact with the Primitives of network security 1.2.

Authentication is generally the initial step towards joining a network. However, using simple credentials to pass the authentication layer cannot be labelled as secure. Despite providing the foundations for using cryptography, these credentials can be traced back by attackers to a single user, breaking their privacy (Bariah et al., 2015; Ali et al., 2019). Generally, every vehicle or RSU in the authenticated layer is protected by a private identity or certificate. In order to join in the secure communications, a node presents its certificate to others in the unsafe layer, that can then grant access by verifying the origin's certificate with its signature. These documents can be generated by a Certificate Authority (CA) or self signed by the vehicles themselves. Certificates and pseudonyms can also be refreshed periodically, further protecting the identity and privacy of the original document.

Identity based solutions attempt to prevent credentials falsification and tracing, by adding an intermediate, private, untraceable identity between a certificate and the physical node in the network, and is widely considered a standard for VANET security authentication (Sun et al., 2010; Bariah et al., 2015; Jin e Papadimitratos, 2015; Tzeng et al., 2017; Ali et al., 2019).

The intermediate identity between a certificate and its owner can be called a Pseudonym or Token, that is randomly generated in the first phase of the communications protocol. A node should contact a trusted, public Certificate Authority to be assigned a temporary token and certificate. The authority is responsible for issuing and keeping track of existing valid pseudonyms and their respective owners. In VANETs, authorities can either be Road-side units or trusted remote cloud servers. Instead of sharing their own personal identities with other nodes in the VANET medium, vehicles should always use their pseudonym as their identification, and recycle that token constantly. Along with the token, authorities can also assist issuers to generate a

unique private/public key pair, which should be used for both signing and encryption of messages. Figure 2.1 presents the transaction between the vehicle and the Road-side unit/remote certificate authority for issuing a pseudonym/token.

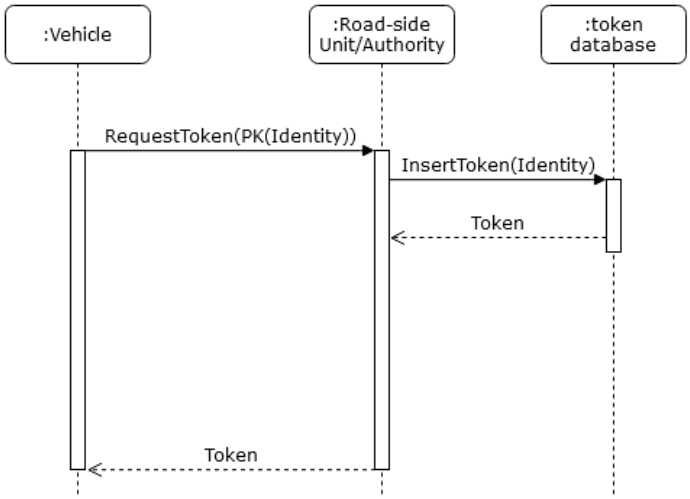


Figure 2.1: Token issuing sequence diagram

When a node receives a message in the encrypted layer that is protected by a pseudonym and authorization certificates, it is important to ensure those are valid. This means that every incoming message in the VANET must be decrypted and verified, by checking the origin certificate with the Certificate Authority public key, which leads to a second execution of cryptography algorithms is necessary. It is also possible to implement the verification directly with the authority, by handling this request through the RSU, or via the Internet (Sun et al., 2010; Bariah et al., 2015; Jin e Papadimitratos, 2015; Tzeng et al., 2017), using a slower but more secure channel.

(Jin e Papadimitratos, 2015) presents a similar solution for improving message verification. Every vehicle appends previously verified messages to its outgoing posts in the network. Receiving vehicles then check if the verified messages match the work done, similar to a cipher block chain scheme. The presented solution improves response times and decreases verification delays as the number of neighboring vehicles increases, proving to be a viable adjunct solution for VANET security. On the downside, the network must withstand longer and more complicated messages in the wireless medium, which could generate interference. The security metadata should also be recycled frequently, as repeating content could simplify cryptanalysis and help attackers discover private/public key pairs.

Message verification represents roughly fifty percent of the security workload, the other half being encrypting and decrypting messages. Developing new and faster verification algorithms will directly impact the VANET security and response times. In the work presented by (Tzeng et al., 2017), secure, tamper-proof road-side units can be used for batch message verification in three steps. In the network setup phase, hashing algorithms and private/public key pairs are agreed upon, between vehicles and RSUs. After the network has been setup, RSUs generate anonymous pseudonyms for every available vehicle in its communications range. The vehicles will then use this pseudonym for signing messages they send in the VANET. During communications, RSUs assist vehicles in message verification, by executing several instances in paralell. While this solution relies upon existing infrastructure and security protocols, it can be impracticable to implement, as RSUs have a limited range, and many would have to be deployed

to cover existing roads. As messages are verified asynchronously by the road-side units, it is impossible for a node to label a received message as suspicious or malign immediatly.

(Ullah et al., 2017) presents a spatial solution for improving user privacy and VANET security. Every node in the network is reponsible for keeping track of a set of neighbors, defined as that node's VANET group. With the evolution of network topology as vehicles change speeds and directions, nodes should recycle their pseudonyms and reconstruct their groups, protecting their privacy and the strength of the encryption algorithm. Renewing the tokens presents a twenty percent increase in the privacy strength score, as presented by the authors.

Given the dynamic topology of VANETs, and how vehicles with similar destinations travel in packs, group-based VANET security solutions were explored. Such solutions take advantage of node cooperation and coordination for ensuring the principles of network security, defined in Table 1.2. So that secure communications can hapen in the groups whereabouts, nodes need to agree upon one or many encryption keys, be that a single shared key for the whole group, as presented in Figure 2.2, or an unique key for every pair of vehicles, as presented in Figure 2.3.

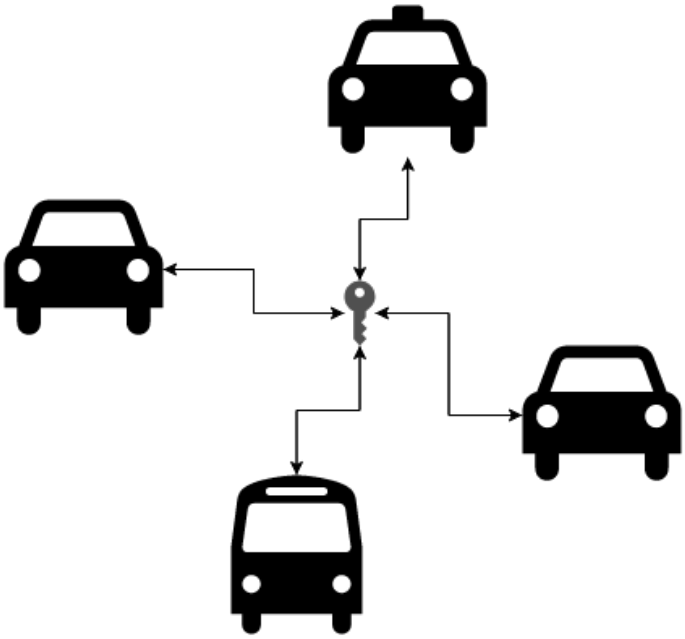


Figure 2.2: Shared group key

Sharing a single encryption key for a whole group, however, is a security vulnerability, as a single untrustworthy node can leak the secret to outsiders, breaking the group privacy. While using a unique key for every pair of vehicles will have an impact on both storage and performance. As nodes need to keep track of every neighbor's keys. Key agreement algorithms, such as (Naresh e Murthy, 2015; Mejri et al., 2014), are important for vehicles to agree on encryption keys to be used for the group. However, such solutions are built on top of expensive cryptography algorithms, as explained in Chapter 1, or require a trustworthy controller node or Road-Side Unit, responsible for initializing the key parameters.

Group based solutions can also improve user privacy. (Deng et al., 2020) presents a Vehicular Social Network (VSN) solution that takes advantage of groups to change the cryptography algorithm when a vehicle needs to change its pseudonym. Local vehicles will rotate their encryption systems and keys, and, by changing the pseudonym, direction, and location, a vehicle can issue a new pseudonym in a secure manner, that cannot be tied to its previously used pseudonym. Using a similar solution will prevent position tracking, and improve user privacy.

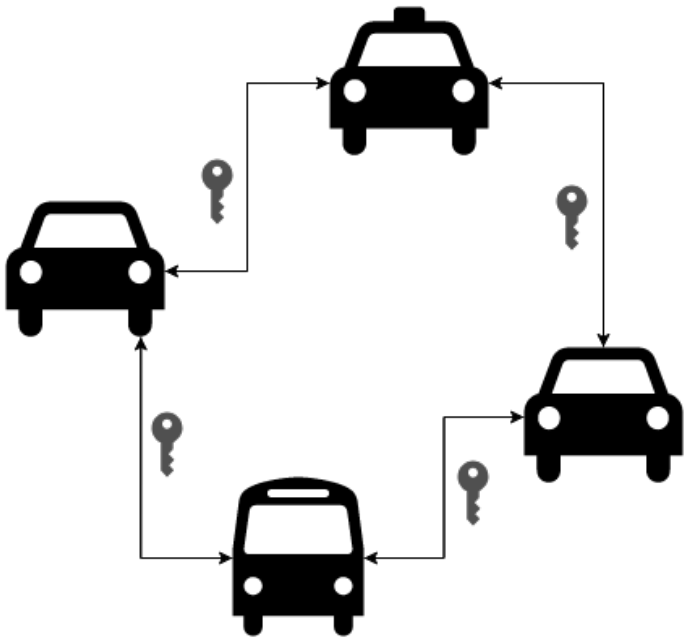


Figure 2.3: Peer to peer group keys

Vehicles that take part in a VANET will control their own group views. Which means that any particular node in communication with its neighbors will consider those neighbors as part of its group. Meanwhile, each neighbor can also have a different set of nearby vehicles that form that vehicle's group. In the presented solution, no vehicle or Road-Side Unit is considered a trusted moderator.

Regarding Group Cryptography, (Lim et al., 2017) present a VANET security solution the short group signature protocol. RSUs form an authorization domain that broadcasts beacon messages for certified vehicles to connect to. Once a vehicle detects a beacon, it communicates with the closest RSU to do a mutual authentication step and exchange a symmetric encryption key. Once their communication is secured, the vehicle is then routed to a Leader RSU (L-RSU) that will provide the group cryptography credentials. In their results, the authors present that their protocol significantly decreases the number of exchanged messages for bootstrapping a new node in the group, while also being tolerant to man-in-the-middle attacks, by using a private/public key pair for the first communication between the vehicle and a RSU. In the provided results, encryption and verification times are also smaller than using a standard asymmetric scheme.

(Zhang et al., 2018) present a group-based security solution for Vehicular Clouds, which are, in their primitive form, location-based groups used for remote computation. In their scheme, nearby vehicles mutually authenticate and form a symmetric-based group. Whenever a vehicle needs to send a message to another vehicle, wheter that be in their cloud or not, the destination's ID, current session and private group key are used to cipher and route that message. While the presented solution provides interfaces for one-to-one, one-to-many, and many-to-many message directions, their greatest fault is not providing a mechanism for storing and retrieving neighbor IDs, laying the foundation for a memory-bound solution. In their paper, the authors also introduce a Cloud Manager, whose purpose is to act as a group leader for solving member revokation and key recycling.

Further developing message verification times, (Zhang et al., 2019) present a group-based, temporal, and spatial solution. Geographic positions, such as cities are under the jurisdiction of a centralized authority, responsible for keeping every vehicles details, managing

the connection via Road-Side Units. Groups are also controlled by the RSU, that enables an asymmetric communication between vehicles, by setting up private and public keys of newcomers. In this implementation, received messages can be verified individually or in batches, using previous public pre-checks provided by other nodes.

(Tan e Chung, 2019) presents a blockchain architecture for secure authentication and key management. In their article, a cloud-based topology is created for the Trusted Authority, responsible for vehicle verification and details storage. Road-Side Units form an edge layer between the cloud and users, enabling long distance connections between vehicles and the authority. The Consortium Blockchain is then used for updating key agreements, as new connections are formed and revoked, distributedly maintaining the network security.

In (Al-Shareeda et al., 2020), the authors present a lightweight security scheme that does not rely on batch verification of messages. In their system, after the initial setup, vehicles communicate with RSUs to gain access to the secure medium. RSUs are responsible for mutually authenticating incoming vehicles and sharing the private group key with every node connected to the group. Every node is protected by a pseudo-ID, generated during the mutual authentication phase, that should be used during any communication, and are also used as vehicle signatures. The verification process is defined by an exclusive-or and general hash functions between the origin's pseudo-ID, the private group key and the message signature. Whenever a message is rejected or an identity is failed to be verified, the system uses BAN logic to revoke any further communication from that origin. In traffic dense areas, the authors present that their solution is sufficient to provide mutual authentication between nodes, along with preserved vehicle Identity and Traceability.

Table 2.6: VANET Security Strengths and Weaknesses

Mechanism Description	References
Symmetric Cryptography Securing exchanged information with a single common encryption key for the whole group, or, for every pair of nodes. Algorithm example: AES. **Strengths**: Low memory usage, faster encryption/decryption. **Weaknesses**: Vulnerable to key leaks, does not provide Non-repudiation, and Traceability, difficult to manage when handling multiple keys.	(Mejri et al., 2014; Qu et al., 2015; Ali et al., 2019).
Asymmetric Cryptography Public-key infrastructure, Message signature and Authentication. Can be implemented using algorithms such as RSA or Elliptic Curve Cryptography. **Strengths**: Robust encryption, protects user privacy and enables Authentication, Non-repudiation and Traceability. **Weaknesses**: Demands more storage as several public keys need to be stored, key agreement algorithms are more complicated and require more messages.	(Sun et al., 2010; Mejri et al., 2014; Qu et al., 2015; Jin e Papadimitratos, 2015; Tzeng et al., 2017; Ali et al., 2019).

Identity-based / Pseudonym privacy protection Using pseudonyms as a privacy mechanism for preventing the identification of a virtual node to an existing vehicle. **Strengths**: Provides an extra layer of privacy protection to the network, while ensuring Non-repudiation and Traceability. **Weaknesses**: Adds a first authentication step, during which nodes must issue their certificates in order to create their pseudonyms.	(Sun et al., 2010; Bariah et al., 2015; Jin e Papadimitratos, 2015; Tzeng et al., 2017; Ali et al., 2019; Al-Shareeda et al., 2020).
Group-based Cryptography Virtually joining nearby nodes in self sutained groups, responsible for their data protectiong and spread. **Strengths**: Takes advantage of the dynamic VANET topology and spatial distribution. Nodes have a smaller conectivity, providing a leaner environment. **Weaknesses**: Vulnerable to spoofing, tracking and key leaking attacks. Generally, key agreement protocols are very demanding and require a Group Leader role.	(Hasrouny et al., 2015; Ullah et al., 2017; Lim et al., 2017; Zhang et al., 2018, 2019; Ali et al., 2019).
Group Broadcast Encryption Every node is responsible for handling their group view, organizing which nodes are trustworthy and can decode their messages, by constructing a private-group key. **Strengths**: Decentralized solution that empowers every node to protect their information. Provides a security threshold that's similar to using Asymmetric Cryptography while demanding fewer messages, memory usage and processing times. **Weaknesses**: Key agreement protocol is complex.	(da Silva e Albini, 2013).
Trust Management Nodes should be able to handle which connections are trustworthy, in order to validate incoming messages and detect malicious nodes. **Strengths**: Enables real time security reactions towards untrustworthy nodes. **Weaknesses**: High level trust management requires dedicated memory and processing power for identifying and reacting to direct trust.	(Greca, 2018).

Table 2.6 presents how different VANET security techniques and cryptography types interact with known attacks and vulnerabilities, along with their scalability limitations. In the scope of the conducted research, the main variable used to measure computational efficiency and response times is the amount of exchanged messages in the network, as decreasing the number of communications will directly impact CPU time and general availability. The contents of 2.6 are derived from (Ali et al., 2019).

3 VANET SECURITY THROUGH GROUP BROADCAST ENCRYPTION

In this chapter, the proposed solution is presented. The primary goal of the conducted research was to improve cryptography performance in VANETs by decreasing the amount of messages sent in the wireless channel, while keeping an acceptable level of security and privacy protection, using an implementation of Group Broadcast Encryption as the principal cryptography and key agreement solution. Vehicles will be free to manage their own groups, taking advantage of a trust management solution, and verifying other vehicles offline, or online, directly with the global authority. A MANET Groud Broadcast Encryption system was presented by (da Silva e Albini, 2013). This framework is used as a building block for the present solution for VANETs, with the goal to determine its feasibility, performance and security.

3.1 ARCHITECTURE

A Group Broadcast Encryption solution relies on some key concepts presented in this section and Table 3.1.

Table 3.1: Group Broadcast Encryption Concepts

Concept	Description
Group View	The set of neighboring trusted nodes that a source node identifies as part of its group.
Private Group Key	A private key for a specific node in a group, that is obtained by combining different destination nodes' public keys. Used for broadcast encryption.
Public Group Key	A public group key for incoming communication from outside of a Broadcast Group.

A Group View is how a single node in a VANET understands its participation in the network, and is composed of a set of trusted neighboring destination nodes. In other words, each node is responsible for curating their group views by taking care of their Private Group Key, which leads to a decentralized, simpler network management. In this VANET solution, since there are no group leaders and every node has a different understanding of the group members, using a Public Group Key is out of scope for this dissertation. However, the lack of such a key does not impact on the solution's performance or security.

A Private Group Key is created by combining different destination's public keys, or subshares, into a single encryption device. Since every node has a unique Private Group Key, every node will have a different Group View, that should be constantly changing as the network topology evolves with incoming and outgoing vehicles.

The lifecycle of a Group is extremely difficult to illustrate, as every participant will have a different understanding of the existing members, and the dynamic network topology of a VANET contributes to an ever changing Group View for each counterpart. Which is why, in order to implement Group Broadcast Encryption in a VANET, every vehicle will have its own concept

of a dynamic group. Through position data and implemented timeouts, nodes can disconnect outgoing members in order to save memory and prevent an endless group. Generally speaking, at any given time, a set of trusted neighbor nodes can be considered as a group. Consider a set of all nodes V, a single node n where $n \in V$, and a sub-set S of V, where $S \subset V$. Every member of S is a trustworthy neighbor of n. In this concept, S can be considered the group-view of n.

The required hardware architecture for this solution does not differ from existing and accepted schemes, as presented in (Whyte et al., 2013; Bariah et al., 2015; Mishra et al., 2016; Ali et al., 2019). Vehicles are augmented by a dedicated OBU, capable of wireless communications such as Wi-Fi, 4G, 5G or LTE, used for networking with other vehicles and Road-side Units. RSUs can be implemented alongside most roads, act as access points for issuing certificates, participate in groups and work as an issue detection station. The proposed solution can also be implemented along other existing mechanisms, such as trust management and attack detection.

3.1.1 Initialization

Following the footsteps of (Whyte et al., 2013), the bootstraping phase of the solution requires vehicles to request access to the VANET to a trusted Certificate Authority, in order to generate a certificate and pseudonym. Ideally, a centralized, universal CA is available on the Internet, and can be accessed by using 4G, 5G, LTE wireless links, or by cable through the Road-Side Units. This interaction is presented in Algorihtm 1, Figure 3.1 and Figure 3.2. For the remainder of this section, every algorithm's point of view is that of a single node, that has a set of neighbors S in its broadcast range. For this source node, every untrustworthy node in broadcast range is considered revoked from S.

Algorithm 1 Getting a verified certificate

1: Given a node n and the Certificate Authority CA
2: n sends its identity and a new certificate to CA, using HTTPS on top of a wireless access-point.
3: CA will sign n's certificate with its private key.

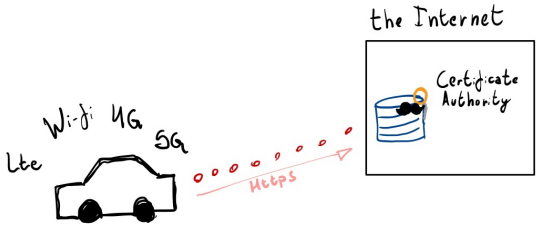


Figure 3.1: Requesting Signed Certificate

Vehicles will then broadcast their pseudonyms along with group joining requests and await an invitation from other nodes, as presented in Algorithm 2, Figure 3.3, Figure 3.4, Figure 3.5, Figure 3.6 and Figure 3.7. Each node can privately decide to grant access to a newcomer, and can reply their pseudonyms to the joining vehicle, Algorithm 3 and Figure 3.8. Existing groups can also share authentication information regarding the new member, in order to distributedly decide if it should be accepted into the group. This step is very similar to what was presented in Figure 2.1 and the process defined by (Whyte et al., 2013).

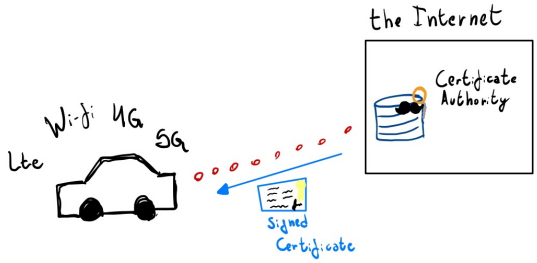


Figure 3.2: Obtaining Signed Certificate

Algorithm 2 Group joining process

1: Given a node n, a time threshold T.
2: n broadcasts its signed certificate to V, and awaits to be invited to a VANET group.
3: If the time T passes and no invitations were received, n will create a new group, alone.
4: If an invitation was received within the time threshold T, n will add the incoming public keys to its private group key, thus, joining the group S.

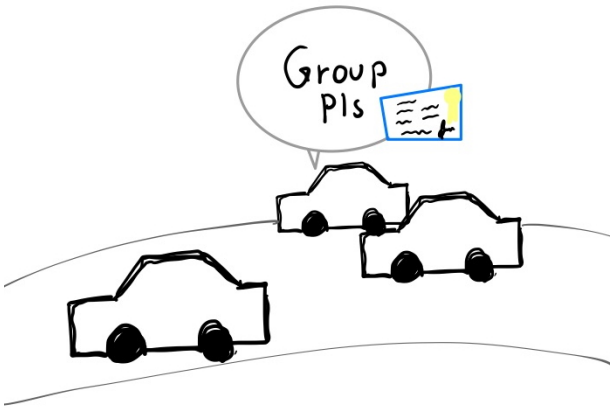


Figure 3.3: Broadcasting request to join a group

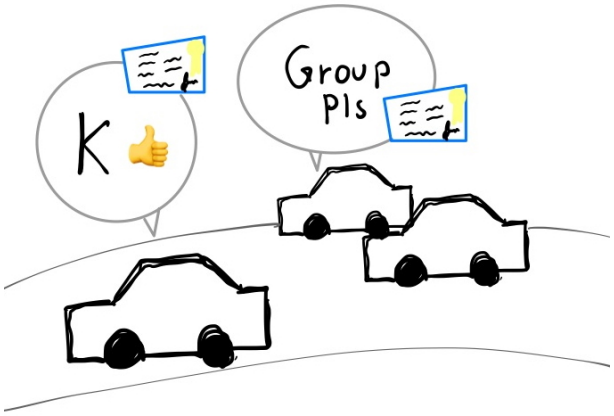


Figure 3.4: Getting a response to join a group

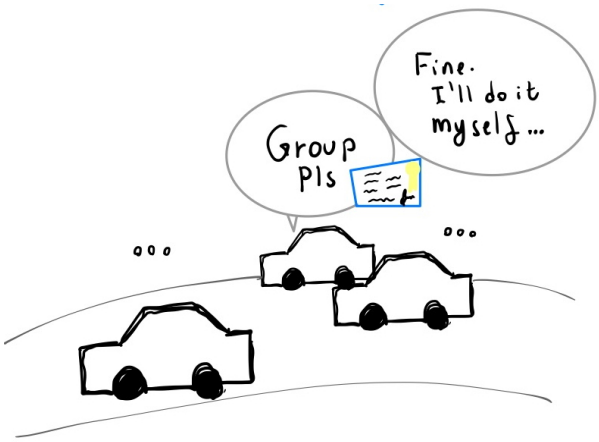


Figure 3.5: Creating a new group after time threshold

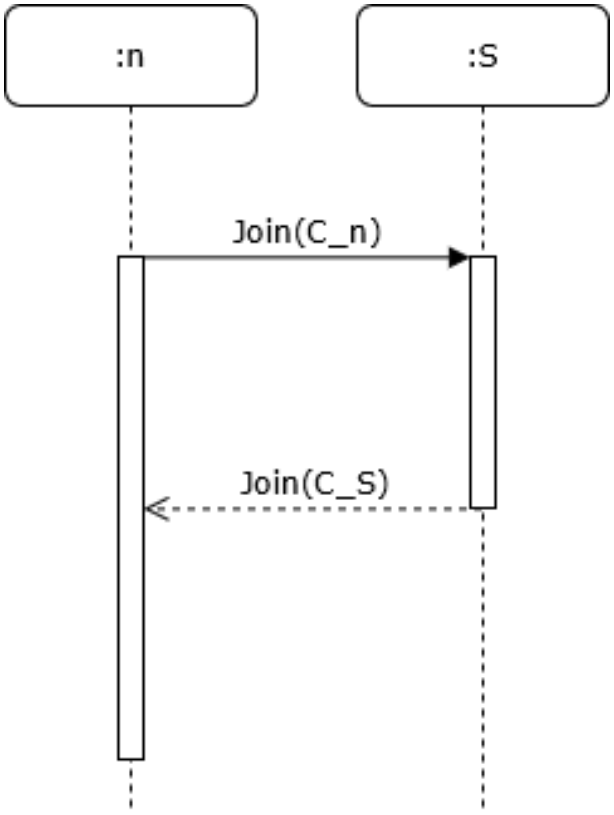


Figure 3.6: Broadcasting request to join a group - Sequence Diagram

3.1.2 Exchanging Secure Messages

Once vehicles control their group views, message sharing is possible in the VANET environment. There are four possible message types, as defined below in Table 3.2.

Table 3.2: Message types in VANET groups

Type	**Description**	**Defined in**

Plain	Simply uses the wireless link to send open information. This is the fastest way to transfer data between nodes, but provides no security, privacy or authentication.	Algorithm 4.
Group messages	A source node *S* sends the message *M* to the group, using its group view private key, protecting the message from any receiving node whose public key was not used to compose *S's* private group key.	Algorithm 5.
Secure group message	A source node *S* sends the message *M* to the group. The message is signed using *S's* private key, then ciphered using the private group key. This extra layer of authentication is recommended for control messages within the group, such as when *S* attempts to deny access to a newcomer or kick and revoke an untrustworthy member.	Algorithm 6.
Member revocation	Revoking a group member is a very simple process. The source node *S* removes the malicious node's M public key from its private group key. After this, whenever *S* sends a message, *M* cannot read it. *S* can also notify its other neighbors of this process in order to build a trust management system between the nodes.	Algorithm 8.
Member Disconnection	Disconnecting a member for the group is pretty similar to member revocation. However, it's a "ban-free" mechanism for simply removing destination nodes that can no longer be reached, as they gained distante from the source nodes. Through position updates and message timeouts, it's possible to detect a node has left the coverage area of the group, and can be disconnected.	Algorithm 9.

While plain messages provide no security or privacy, communication between vehicles outside of a group should be possible in the VANET wireless link. This type of message can be used for sharing public keys, requesting to join groups or even receiving invitations to join an existing group. In general, plain or groupless messages should be used for setup purposes only, and should never contain sensitive information. Algorithm 4, Figure 3.9 and Figure 3.10 present the groupless broadcast interface for wireless nodes.

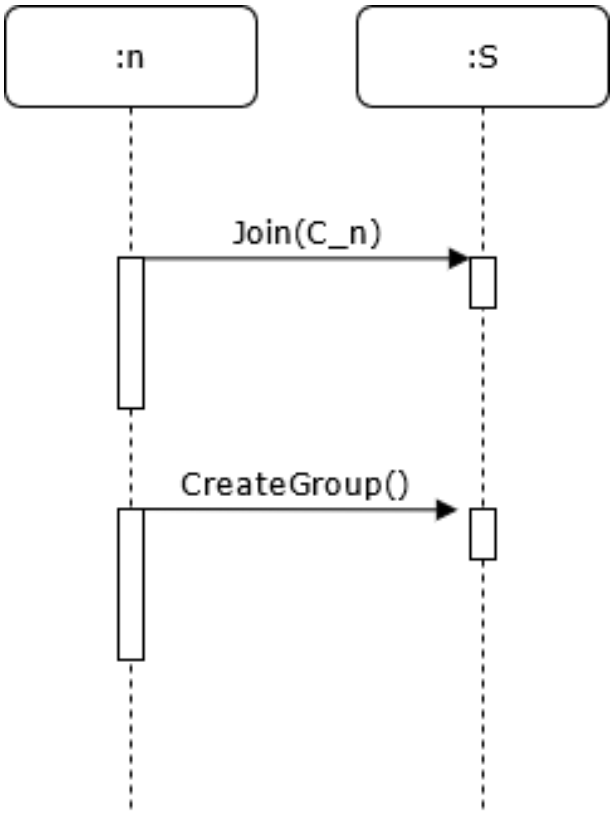


Figure 3.7: Creating a new group after time threshold - Sequence Diagram

Algorithm 3 Replying to a group join request

1: Given a node n and an incoming signed certificate C_x from node x, where $x \ni S$.
2: n attempts to verify the certificate using known Certificate Authorities' public keys.
3: If the certificate is valid, n adds C_x to its private group key, and sends its public key to Node x, thus adding x to S.
4: If the certificate cannot be verified, n ignores the request, and warns its group that it could not verify C_x.

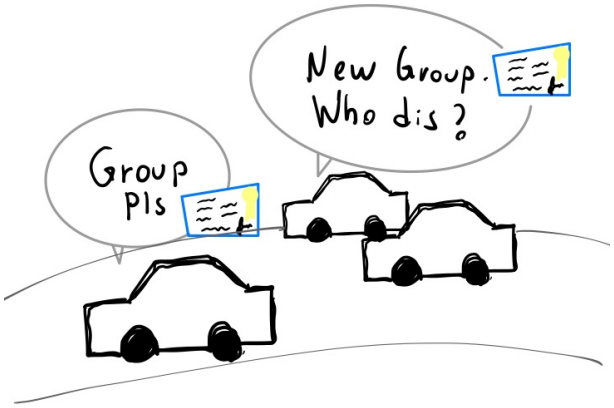


Figure 3.8: Replying to a group joining request

Algorithm 4 Sending a plain message in the network

1: Given two nodes n and a node $x \in S$;
2: n broadcasts the message msg to S;
3: x receives and interprets the message msg from n.

Figure 3.9: Sending a plain message in the network.

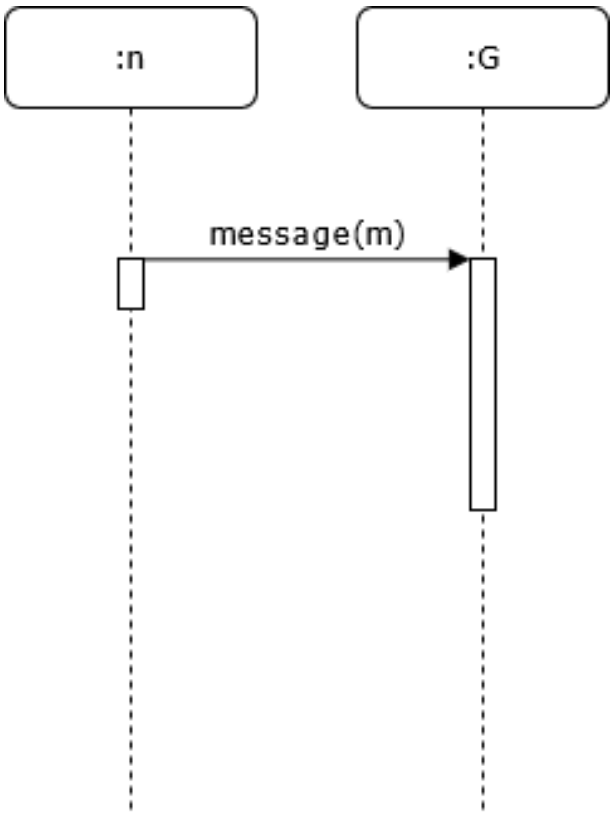


Figure 3.10: Sending a plain message in the network - Sequence Diagram

Once a vehicle is part of a group, and has its own group view and secure neighbors, it can send a protected message to the group, as defined in Algorithm 5, Figure 3.11, Figure 3.12 and Figure 3.13. While group messages are secure and can only be understood by vehicles that are part of the source node's group, they do not necessarily need to be signed. Unsigned group messages can only be sent by authenticated, previously verified vehicles, and are protected by the private key used to encrypt it. The lack of signature ensures a faster response time for processing the message. However, unsigned messages should only be applied for position and constant updates, and should never be used for more important operations, such as group control and emergency notifications.

Algorithm 5 Sending a group message in the network

1: Given a node n and its group S.
2: n ciphers the message msg using its private group key and broadcasts it to S.
3: $M1 \leftarrow GBE(msg, PGK_n)$
4: Nodes of S receive and decipher the message using their private keys.

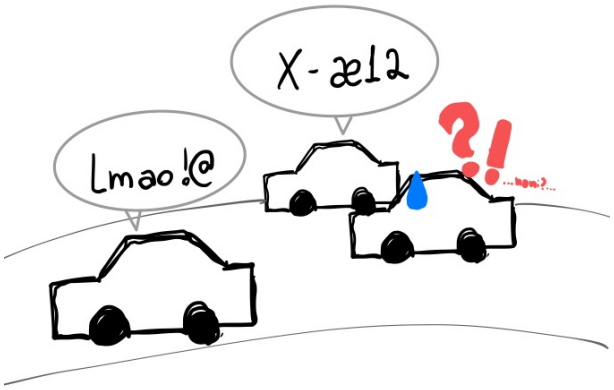


Figure 3.11: Sending group message in the network.

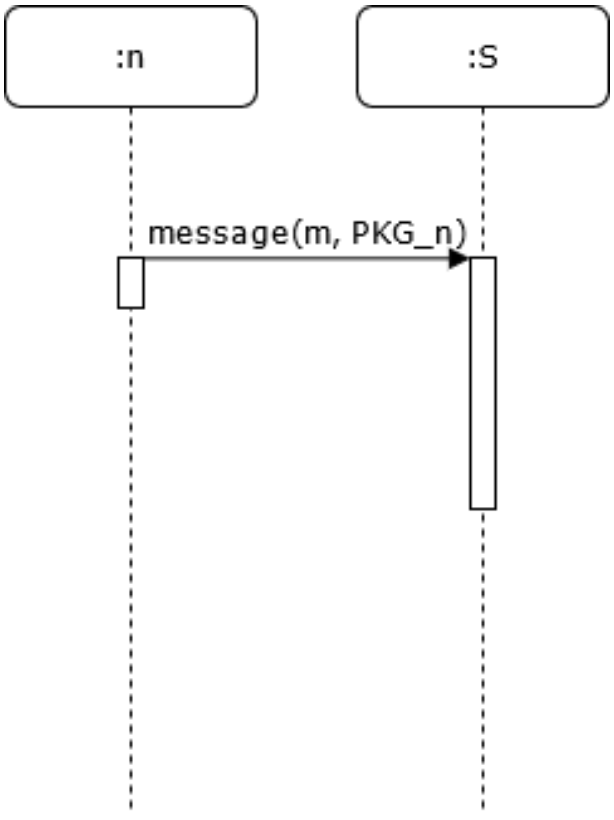


Figure 3.12: Sending group message in the network - Sequence Diagram

Signed messages, on the other hand, require that the source vehicle adds its signature to very important messages sent in the VANET medium. The message content is hashed and encrypted by the source node's private key, and appended to the initial value. The complete

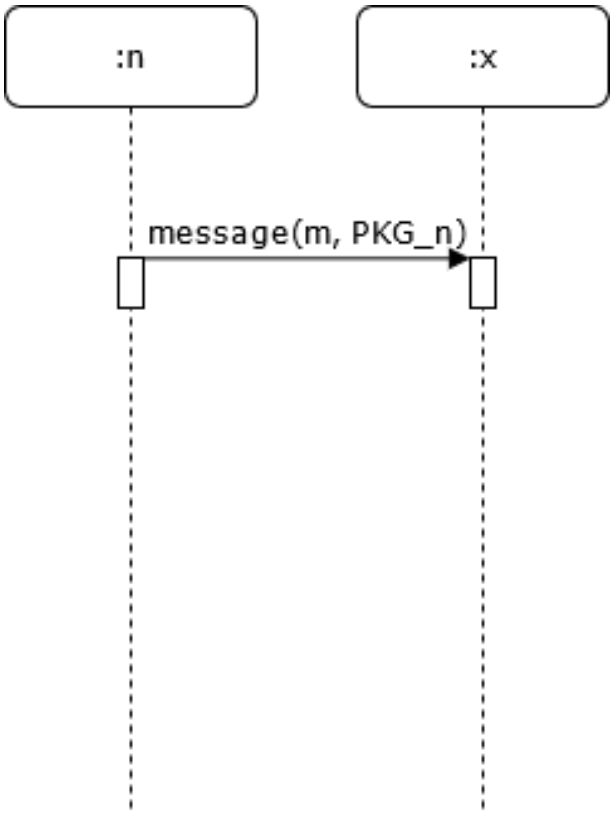


Figure 3.13: Sending group message in the network - Sequence Diagram

payload is then encrypted using the private group key before broadcasting the message in the wireless network. The additional signature step is essential for verifying important messages that are not as timely critical as simple group updates, and must be used for control messages and emergency notifications. Algorithm 6, Figure 3.14 and Figure 3.15 present the verified message concept.

Algorithm 6 Sending a verified group message in the network

1: Given a node n and its group S.
2: n ciphers the message msg using its private group key, signs it with it's private key, and broadcasts it to the network.
3: $M_1 \leftarrow GBE(msg, msg + PGK_n(msg))$
4: Other nodes of S receive, decipher and verify the message using their private keys.

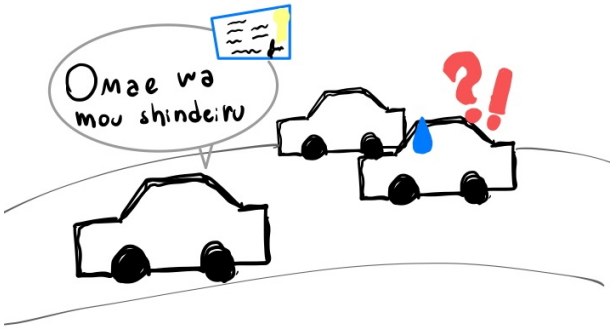


Figure 3.14: Sending a verified message in the network.

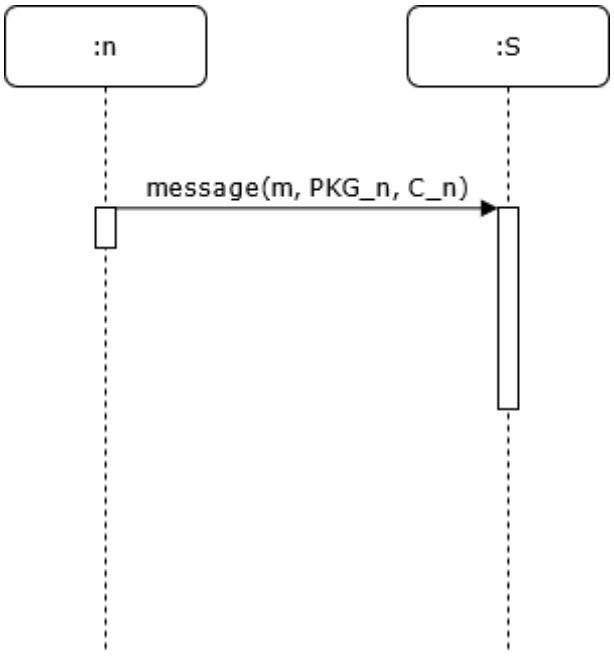


Figure 3.15: Sending a verified message in the network - Sequence Diagram

Another particular use case of messages is for breaking the gap between groups. Since nodes have different sets of destination vehicles, they can forward received messages to other groups. Algorithm 7 presents this concept. Group bridging is a process that allows for complex routing and the appropriate expansion of the recipients of a given message, in simple terms, forwarding the information.

Algorithm 7 Group bridge cost analysis

1: Given two nodes n and x where $x \in S_n$.
2: n broadcasts a message to S.
3: x broadcasts the same message to S_x

3.1.3 Member Revocation

In the event of detecting a malign node, it is possible to revoke its privileges in the network. The process is very simple: the source vehicle removes the other's public key from its private group key. Doing so, every message that is shared by the source will not be readable by the destination node, effectively taking it out of the group view. Ideally, this process should be done by the majority of the vehicles in the vicinity. Finally, the member revocation messages are a particular use case of signed group messages. As every vehicle is responsible for controlling their group members, they can simply remove a particular vehicle of their destination group. When that happens, a secure notification can be sent to other members, as a hint towards unstrusted, misbehaving nodes. Algorithm 8 and Figure 3.16 present this use case. It's then up to the receiving nodes to decide if they should also revoke that node from their group views.

Algorithm 8 Member Revocation

1: Given a source node n and a malicious node x where $x \in S_n$.
2: n determines that x is malicious, and warns the vehicles other vehiles of S that x is being revoked.

It is also possible to remove a destination node from the group without revoking or banning their presence. This is necessary when that node is no longer reachable through short

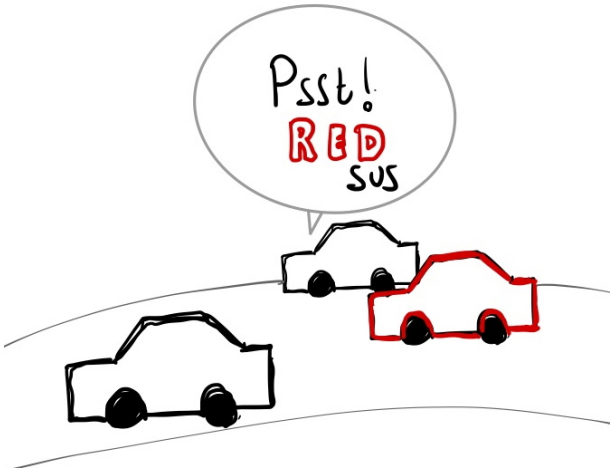


Figure 3.16: Member Revocation Broadcast.

distance communication, as it effectively moved away from the group. In this case, disconnection detection is used to determine when this use case happens, and the group members simply remove that node from their Group View in order to save resources.

Algorithm 9 Member Disconnection

1: Given a source node n, its group S and a node x where $x \in S$.
2: n determines that x is unreachable, as no position updates have com from it until a timeout rings and the final position updates received indicated it moving away from S.
3: n removes x from its private group key.

3.2 TRUST MANAGEMENT

While Trust Management is not in the scope of this work, it is important to note its effect when applied to VANETs and Group Broadcast communications (Greca, 2018). When vehicles are responsible for tracking other vehicles actions in the network, a score can be used to represent how trustworthy neighboring nodes are to a source node. Through testing or behavior analysis, a source node can detect and distrust a malign neighbor in the network, and take action in order to protect itself and other trustworthy neighbors. The trust score threshold that defines how trustworthy a node is differs between presented solutions, so its values are considered out of scope for this dissertation. If enough secure nodes distrust a potentially malign neighbor, it can be effectively removed from participating in the network, by having the source nodes revoke its part in their private keys. Further actions can be taken in order to investigate the malign intent in the network, by reporting the suspect to a competent organ, which will be able to trace the origin of the pseudonym certificate.

In this particular set-up, every node within a local group can have their own "group view". A group view is a subset of the actual local network group, and the source vehicle manages the trustworthy connections within it.

Nodes are responsible for their own group views, deciding which vehicles should they keep as destinations and which should be revoked. This decision process happens based on the input of a trust management system. (Greca, 2018) defines a solution for trust management in VANETs, where every node will evaluate how much it can trust its neighbors. If a trust weight is past a threshold value for a single node, that node can be revoked from the group view. Keeping the trust value for neighboring nodes is also important for additional judgment on top of received

messages. Whenever a node receives a message from another node, the content can be ignored or processed, depending on how much the source node is trusted.

3.3 IMPLEMENTATION COST ANALYSIS

In this section, a mathematical cost analysis is presented concerning the amount of messages required to maintain the system working. The work presented here is derived from (da Silva e Albini, 2013). Table 3.3 presents a set of symbols used, it is based on (da Silva e Albini, 2013).

Table 3.3: Notation for Key Management

Item	Description
V	Set of all nodes
n	Any given node of V
S	Sub-set of reachable trustworthy nodes of V by n, where $S \subset V$
$N_{\mathcal{F}}$	Founder nodes
m	Number of Founder nodes
IC	Cost to initialize group
N_i	Identification of node i
SK_i	Private key of node i
PK_i	Public key of node i
MSK	Master private key of system
$size(f_m^i(x))$	Size of each sub-share of the MSK generated by nodes
MPK	Master public key of system
MSK_i	Share of master private key hold by node i
Ω	Size of subset of Founder nodes contacted by a joining node
Δh	Average number of hops between nodes
NM	Cost of a new member joining the group
NR	Cost of key revocation

Decreasing the average amount of messages is an interesting approach towards VANET security efficiency. Fewer messages on the network directly contribute to reducing communications delay and interference, while also keeping processing time and power usage low on the CPU, as it is not required to compute many different encryption keys to the same message. Finding the ideal balance between security and efficiency should enable VANETs to operate in a low latency, fast response time environment as it's expected.

3.3.1 Group setup

Setting up the group requires joining vehicles to broadcast their verified public keys to nearby vehicles, these vehicles are defined as the Founding Nodes $N_{\mathcal{F}}$ of the Group. Whenever an existing group captures this message, some sort of a vote, as described in algorithm 10, takes place to determine if the joining vehicle should be a part of the group. As explained in the algorithm, the number of messages required should be linear to the number of nodes in the existing VANET group. Considering a set of founding nodes with m members, the cost to initialize the key management, denoted by IC in Equation 3.1:

$$IC = m \cdot (m-1) \cdot size(f_m^i(x)) \tag{3.1}$$

in which $size(f_m^i(x))$ is the size of each sub-share of the MSK generated by nodes. A sub-share, being a fraction of the master key for every fouding member from $N_{\mathcal{F}}$. As nodes must be close during the initialization phase, hop count is not considered (da Silva e Albini, 2013).

When a new node joins an existing group, the communication overhead is defined as follows: Considering that a new node contact Ω members of the set of founding nodes in order to request authorization to act as a group member, the cost for a new member to join the group, denoted by NM, is defined in Equation 3.2.

$$NM = (\Omega \cdot sizeof(ReqMsg) + \Omega \cdot size(f_{new}^i(x))) \cdot \Delta h \tag{3.2}$$

in which $ReqMsg$ is the message sent by n_{new} to the nodes of the group, $f_{new}^i(x)$ is each sub-share of MSK sent to n_{new} and Δh is the average of hops between nodes (da Silva e Albini, 2013). A sub-share of MSK is a node's public key part in another's private group view key, created by combining the neighbors' keys.

Algorithm 10 Group setup cost analysis

1: Given a candidate node C, and N vehicles in a group $G(V_1 ... V_N)$.
2: C begins to broadcast its signed certificate.
3: K vehicles ($1 \leq K \leq N$) receive C's certificate and copy the message to the group. $GK[PRVi(m)]xK$ messages are sent to the group.
4: J vehicles ($0 \leq J \leq N$), who could not verify C's certificate, should notify others. $GK[PRvi(NOT(C))]xJ$ messages are sent to the group.
5: Vehicles who could verify C or accept the neighbor's view on the matter, should transmit their public keys to C, and add C's public key to their private group key. $PUc[m]$.
6: In total, up to 2N + 1 messages are sent in the network to setup a new vehicle to the VANET group.

3.3.2 Revocation

Whenever a node is detected as malicious, or cannot be trusted anymore by a source node, the source node warns the rest of the group that it is revoking the malicious node from its private group view. While this should pose no effect to other nodes, it can be taken as input for trust management solutions. Algorithm 8 presents this step.

The cost to revoke the private key of a given node N_b depends on the number of nodes which have considered N_b compromised. Each node which detects the misbehavior of N_b sends a accusation message to all nodes of the group. Thus, considering γ accusers, the key revocation cost, defined as NR, is defined in Equation 3.3:

$$\begin{aligned} NR = (\gamma \times t) \cdot sizeof(AcMsg) + (t)^2 \cdot \\ sizeof(revMsg) + BcastMsg \end{aligned} \tag{3.3}$$

in which $AcMsg$ is the accusation message sent by accusers to the whole group, $revMsg$ is the revocation message and $BcastMsg$ is the broadcast encryption message sent to all nodes (da Silva e Albini, 2013).

3.4 SUMMARY

To summarize the proposed solution framework and general goals of this project, Table 3.4 is presented below.

Table 3.4: Summary and Project Goals

Goal	Description
Group Broadcast Encryption	Implement, simulate and test using Group Broadcast Encryption as a VANET security framework.
Decrease Message Count	It should be possible to improve VANET security performance by requiring fewer messages to be sent in the network.

The conducted research led to the belief that VANET security can have its performance improved by utilizing Group-Based solutions for decreasing the number of exchanged messages in the network, while taking advantage of the network's topology and dynamic environment. In the proposed solution, vehicles are independent and free to control their own views of neighboring groups, but must be verified by a globally trusted authority to be able to issue communications credentials. After the verification step, every connecting node (Vehicle or RSU) must mutually verify their peers. Group Broadcast Encryption cryptography is used to secure any communication between different nodes in the VANET medium. With the framework described above, VANET security should be solid, attack resistant and lean enough for the fast changing VANET environment.

4 SECURITY AND PERFORMANCE BENCHMARK

In this chapter, the environment preparation, simulations and collected results are presented. Three security algorithms are executed in the VANETs, one using symmetric cryptography, another using asymmetric cryptography, and the implementation of what was defined in Chapter 3, and analysed in Chapter 3.3.

4.1 SIMULATION SET-UP

In order to evaluate the solution, an urban simulation environment was used to generate the movement model and connectivity tables, according to a VANET context. In the simulator, vehicles drive freely around the roads of the city, generating information about their connectivity to other nearby vehicles. After generating the connectivity tables, the output data is processed for each algorithm, in order to obtain use statistics, such as number of neighbors, number of messages used to connect to a new node, number of messages broadcast in the network, and number of messages used to disconnect a node. The Opportunistic Network Environment (Keranen, 2008) was the chosen application to execute these simulations, as it is a widely regarded simulation tool for VANETs, easily extendable and simple to set-up. Our simulations have been run on the Helsinki city map, the standard map used in the simulator, also using the Working Day Movement Model (Ekman et al., 2008), which will route vehicles in the map from their homes, to their work location and back. Some vehicles also run errands during the day and after the work hours, going by the city with less traffic.

Simulations were run thirty times for twelve hour days with an eight hour work shift, and provide the connectivity tables for every vehicle in each timestamp. The connectivity tables were then processed in the three algorithms, providing the results.

4.2 SIMULATING SECURITY IMPLEMENTATIONS

After generating the movement models and connectivity tables, three network security protocols were executed on top of every simulated day in Helsinki: One utilizing Symmetric cryptography, one running Asymmetric cryptography, and a final one using Group Broadcast encryption.

The implementations are defined as follows:

- **Symmetric Cryptography Peer-to-Peer**: Every pair of nodes agrees on a symmetric key to communicate with. Exchanged messages must be ciphered once for every destination node, and each message should be augumented with the source node's identifier.
- **Symmetric Cryptography**: Every reachable node is added to a growing group that shares a common key. Whenever a vehicle is added or removed from the group, a new key is generated and shared between all members.
- **Asymmetric Cryptography**: Every reachable node is managed by a source node, keeping all the neighbor's public key. Every message sent is ciphered once for every connected neighbor.

- **Group Broadcast**: Every reachable node is added to the source node's group view. Messages sent from this particular node are readable by every node whose public key was used in the creation of the group view key.

During this work's practical part, simulations are run considering the use of a single symmetric key for a forever merging and growing group, in order to decrease the key cardinality to a single key, used to both cipher and decipher. While this will create many different security vulnerabilities and further management complications, this experiment is interesting further on, in order to prove the Group Broadcast efficiency. The vulnerabilities created by sharing a single key for several vehicles are ignored and not a part of this project's scope. On the other hand, using a Peer-to-Peer symmetric key was discarded due to the high number of keys and identifiers that need to be stored and used in cryptography, effectively increasing the number of exchanged messages in the wireless environment.

The implementation used in the symmetric cryptography is easily identifiable as the least secure, since every node gets the common key. This was used in order to properly demonstrate the performance gain of using Group Broadcast Encryption, as it's only necessary to encrypt a message once, and it will be readable by the whole group, the same amount of work required for using a single symmetric key. Better security could be achieved using symmetric cryptography, if every pair of nodes had a single key. This latter solution would be greatly outperformed by using group broadcast encryption, as fewer messages would be necessary to share the same information in the network.

Table 4.1: Number of messages per algorithm

	Symmetric	**Asymmetric**	**Group Broadcast**
Connection	Three-way handshake between two nodes, plus message for sharing pseudonyms	Diffie-Helman between each pair of nodes.	Three-way handshake between two nodes, plus message for sharing pseudonyms
Messages	Single encrypted message for all neighbors	One encrypted message for each neighbor	Single encrypted message for all neighbors.
Revocation	Single encrypted message for all neighbors	One encrypted message for each neighbor	Single encrypted message for all neighbors.

The main variable observed in these simulations is the number of messages. Counters are used in order to identify how many messages are required to add a vehicle to a group, how many to remove a vehicle from a group, and, mainly, how many messages are required in order to send a secure message within this group. Table 4.1 describes how the three types of algorithms chosen interact with the number of messages.

Table 4.2: Number of stored keys per algorithm

Symmetric P2P	**Symmetric**	**Asymmetric**	**Group Broadcast**
1 key for every pair of nodes (N^2)	1 spatially local group key	1 key for every node (N)	1 assymetric key for every node

The presented solution is also very memory efficient, as every vehicle only needs to store one private group key. In comparison to the other algorithms, it is the same cardinality of the Symmetric solution. Table 4.2 presents the number of stored keys per algorithm.

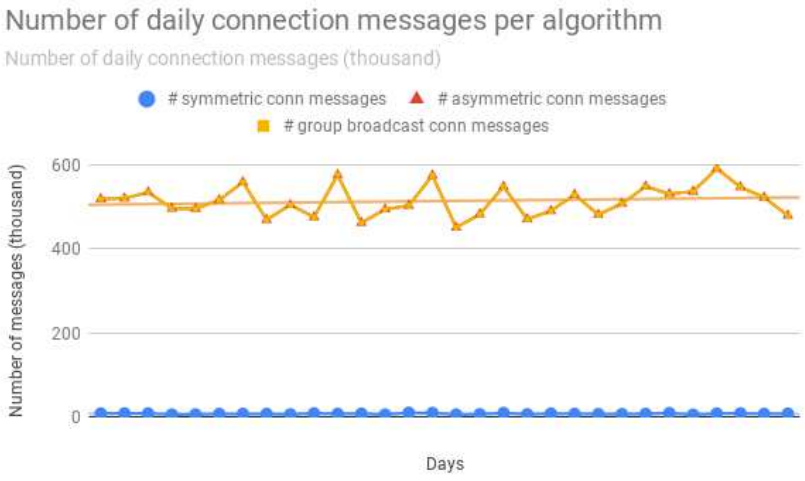


Figure 4.1: Number of connection messages per algorithm per day

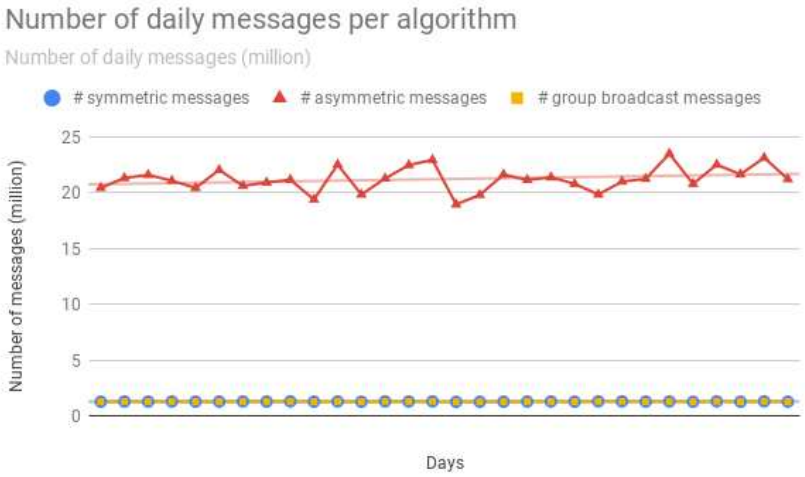


Figure 4.2: Number of secure messages per algorithm per day

As seen in table 4.1, the group broadcast solution should be bound to have the same cardinality and number of messages as one of the other two algorithms, in each of the three types of messages that are required. For connection messages, the amount is equal to the number of messages required to set-up the asymmetric encryption. This happens because, by definition, group broadcast encryption is an asymmetric algorithm, and sharing the public keys is required. The symmetric solution is by far the simplest solution to set-up, because every vehicle in the simulation is sharing a common key. While this is not exactly useful for comparing connection messages, it will prove very important for the comparison of secure group messages. For sending secure messages in the group, the group broadcast encryption solution works just like the symmetric solution, by sending a single message to the whole group view, while the asymmetric solution is burdened to send one message for every neighbor. Finally, when revoking a vehicle from the group, the source node only notifies its neighbors that it is doing so. In conclusion, the symmetric and group broadcast solution only send a single message, while the asymmetric solution is burdened again to send one message for every other node.

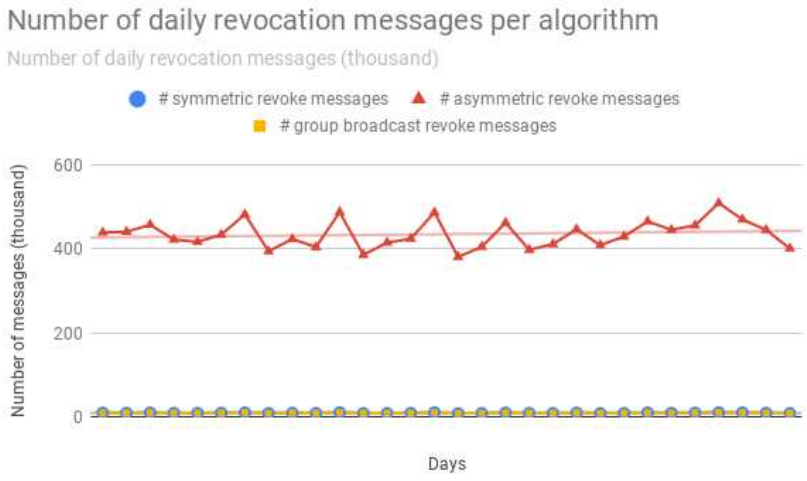


Figure 4.3: Number of revocation messages per algorithm per day

Figure 4.1 presents the number of messages used by each algorithm in order to set-up the encryption. In this image, the number of messages used to set-up the asymmetric encryption and the group broadcast encryption were exactly the same, as discussed on table 4.1.

Figure 4.2 displays the amount of exchanged secure messages for each algorithm in every simulation. In this image, symmetric encryption and group broadcast encryption share the exact same number of messages (one for each neighbor every update), and are about twenty times lower than the number of messages required by the asymmetric encryption.

In the implemented simulations, nodes notify their groups whenever they detect a disconnection or untrustworthy node, effectively revoking their keys from the group view. Figure 4.3 shows the number of revocation messages sent. For the simulations, the only type of disconnection message sent was for out-of-range disconnections. Once again, the number of messages sent in the network was equal between symmetric and group broadcast encryption, with the number of messages required for the asymmetric encryption being about four hundred times bigger.

While group broadcast encryption does not necessarily decrease CPU load for encrypting messages, using it in a VANET environment can significantly decrease the amount of work required to send secure messages in the network, because the source node works just as if it had a single key for every other neighboring node. This ensures that the scaling of the number of messages is linear to the number of neighboring nodes instead of exponential, guaranteeing that the network is not flooded with repeated messages that were encrypted with a different key.

Table 4.3: Group Broadcast Encryption comparison towards VANET Security

Mechanism Description	References
Symmetric Cryptography Group Broadcast Encryption provides a Symmetric-like network environment, significantly decreasing network load and message count, while also ensuring that an Asymmetric-like security protocol is in place. **Strengths**: Similar network load, faster response times. **Weaknesses**: Vulnerable to key leaks, does not provide Non-repudiation and Traceability, difficult to manage when handling multiple keys.	(Mejri et al., 2014; Qu et al., 2015; Ali et al., 2019).

Asymmetric Cryptography Public-key infrastructure, Message signature and Authentication. All of these mechanisms can also be implemented in a Group Broadcast environment. **Strengths**: Similar security protocols, mechanisms, but heavier protection. **Weaknesses**: Much higher network load than Group Broadcast Encryption, slower processing times for encrypting, decrypting and verifying messages.	(Sun et al., 2010; Mejri et al., 2014; Qu et al., 2015; Jin e Papadimitratos, 2015; Tzeng et al., 2017; Ali et al., 2019).
Identity-based / Pseudonym privacy protection Similar mechanisms can, and should be implemented in Group Broadcast Encryption. **Strengths**: Provides an extra layer of privacy protection to the network, while ensuring Non-repudiation and Traceability. **Weaknesses**: Adds a first authentication step, during which nodes must issue their certificates in order to create their pseudonyms.	(Sun et al., 2010; Bariah et al., 2015; Jin e Papadimitratos, 2015; Tzeng et al., 2017; Ali et al., 2019; Al-Shareeda et al., 2020).
Group-based Cryptography Virtually joining nearby nodes in self sutained groups, responsible for their data protectiong and spread. **Strengths**: Takes advantage of the dynamic VANET topology and spatial distribution. Nodes have a smaller connectivity, providing a leaner environment. **Weaknesses**: Vulnerable to spoofing, tracking and key leaking attacks. Generally, key agreement protocols are very demanding and require a Group Leader role.	(Hasrouny et al., 2015; Ullah et al., 2017; Lim et al., 2017; Zhang et al., 2018, 2019; Ali et al., 2019).
Group Broadcast Encryption Every node is responsible for handling their group view, organizing which nodes are trustworthy and can decode their messages, by constructing a private-group key. As presented in this chapter, this is a viable solution for VANET security, that provides fewer control messages in the network, which affects response times. **Strengths**: Decentralized solution that empowers every node to protect their information. Provides a security threshold that's similar to using Asymmetric Cryptography while demanding fewer messages, memory usage and processing times. **Weaknesses**: Key agreement protocol is complex.	(da Silva e Albini, 2013).

Trust Management Trust management is widely recommended for VANET Security, and can be implemented alongside Group Broadcast Encryption. **Strengths**: Enables real time security reactions towards untrustworthy nodes. **Weaknesses**: High level trust management requires dedicated memory and processing power for identifying and reacting to direct trust.	(Greca, 2018).

Table 4.3 extends Table 2.6, first presented in Chapter 2. Here, the cryptography algorithms and mechanisms are directly compared to Group Broadcast Encryption, further proving its feasibility towards a VANET Security protocol. As mentioned in the table, such a solution can directly decrease network load by requiring fewer control messages for key agreement, group construction and revokation, while also ensuring that the network remains Ad-Hoc, as no further infrastructure or leaders need to be implemented. Table 4.3 also provide some input on how to integrate the presented solution to existing and proposed protocols for VANETs.

It's possible to integrate Group Broadcast Encryption alongside many existing security mechanisms and protocols, such as Public Key Infrastructure (PKI), (Sun et al., 2010; Mejri et al., 2014; Qu et al., 2015; Jin e Papadimitratos, 2015; Tzeng et al., 2017; Ali et al., 2019), Identity based and message verification (Sun et al., 2010; Bariah et al., 2015; Jin e Papadimitratos, 2015; Tzeng et al., 2017; Ali et al., 2019; Al-Shareeda et al., 2020) and Trust Management (Greca, 2018). Table 4.4 presents a revised layered organization of VANET security mechanisms, similar to what was presented on Table 2.3.

Table 4.4: Network Security Layers

Layer	Description
Application Layer	VANET Applications are connected to the network using this interface.
Trust Layer	Trust Management solutions run in this layer, detecting and reacting to malign input on the network.
Encrypted Layer (Group Broadcast Encryption)	Communication should be done on top of encrypted channel. Group Broadcast Encryption can be implemented here, as long as the underlying layer is setup properly.
Authentication Layer	Required step on the communications protocol for accessing the Encrypted Layer. In order to implemented Group Broadcast Encryption, the initialization steps presented in Chapter 3 must be implemented.
Wireless Medium	General, unsecure wireless channel used for communications.

5 CONCLUSION

In this dissertation, the developed research on VANET Security supports the proposition that Group-Based solutions provide a leaner framework, decreasing message count, memory usage and computation times. Group Broadcast Encryption proves to be a solid, lean foundation to enable independent nodes to form neighboring clusters for fast wireless communication, on top of a mutually verifiable joining process. Vehicle identities can be presever by using renewable pseudonyms, while non-repudiation is made possible by the group signature, which can be deconstructed by global authorities for audit purposes. The full proposition was presented in Chapter 3, along with a performance benchmark, Chapter 4, which indicates that Group Broadcast algorithms are effective in decreasing the amount of exchanged messages between vehicles, thus improving the performance and response times. This is important to create a simpler and faster network, which requires fewer mechanisms for controlling the general state of the network, while also decreasing the resources consumed by each node, such as memory and CPU-time. While the solution is not as lean as simply using symmetric cryptography, it is far more economical than using a fully asymmetric system, keeping the security principles of asymmetric cryptography. As the main focus of the conducted research was to evaluate Group broadcast encryption as a VANET security application in a simulated environment, future developments are able to build and test similar solutions in a physical environment. VANETs can benefit from using Group Broadcast Encryption systems, as they enable a democratic environment for their nodes, that are free to control their trusted connections.

In conclusion, the present research supports the usage of Group Broadcast Encryption as a feasible scheme for VANET security, along with existing mechanisms such as Trust Management, Pseudonyms and Mutual Authentication models. Further research can be conducted in order to improve routing and message relaying between group borders, as well as implement and verify Group Broadcast Encryption for VANETs in a physicall environment. Further integrations with edge computing technologies are also an interesting possibility for future work. Table 4.4 elaborates these possibilities.

The implemented benchmark presented in Chapter 4 is available on Github, under an MIT license. An article was also published here under an Open Access license during the course of this work.

REFERENCES

Al Hasan, A. S., Hossain, M. S. e Atiquzzaman, M. (2016). Security threats in vehicular ad hoc networks. Em *2016 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, páginas 404–411. IEEE.

Al-Shareeda, M. A., Anbar, M., Alazzawi, M. A., Manickam, S. e Al-Hiti, A. S. (2020). Lswbvm: A lightweight security without using batch verification method scheme for a vehicle ad hoc network. *IEEE Access*, 8:170507–170518.

Ali, I., Hassan, A. e Li, F. (2019). Authentication and privacy schemes for vehicular ad hoc networks (vanets): A survey. *Vehicular Communications*, 16:45–61.

Bariah, L., Shehada, D. e Yeun, C. Y. (2015). Recent advances in vanet security: A survey. Em *IEEE 82nd Vehicular Technology Conference (VTC2015-Fall)*. IEEE.

da Silva, E. e Albini, L. C. P. (2013). Towards a fully self-organized identity-based keymanagement system for manets. Em *International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*. IEEE.

Deeksha, Kumar, A. e Bansal, M. (2017). A review on vanet security attacks and their countermeasure. Em *4TH IEEE International Conference on Signal Processing, Computing and Control (ISPCC 2k17)*. IEEE.

Deng, X., Xin, X. e Gao, T. (2020). A location privacy protection scheme based on random encryption period for vsns. *Journal of Ambient Intelligence and Humanized Computing*, 11:1351 – 1359.

Ekman, F., Keränen, A., Karvo, J. e Ott, J. (2008). Working day movement model. *Mobility models '08: Proceedings of the 1st ACM SIGMOBILE workshop on mobility models*, 1(1):33 – 40.

Engoulou, R. G., Bellaïche, M., Pierre, S. e Quintero, A. (2014). Vanet security surveys. *Computer Communications*, 44:1–13.

Greca, R. D. M. (2018). Truman: Trust management for vehicular networks. Dissertação de Mestrado, Pós-Graduação em Informática - Universidade Federal do Paraná.

Hao, Y., Cheng, Y. e Zhou, C. (2011). A distributed key management framework with cooperative message authentication in vanets. *IEEE Journal on Selected Areas in Communications*, 29(3):616–629.

Hartenstein, H. e Laberteaux, K. (2008). A tutorial survey on vehicular ad hoc networks. *IEEE Communications magazine*, 46(6):164–171.

Hasrouny, H., Bassil, C., Samhat, A. E. e Laouiti, A. (2015). Group-based authentication in v2v communications. Em *Fifth International Conference on Digital Information and Communication Technology and its Applications (DICTAP)*. IEEE.

Hasrouny, H., Samhat, A. E., Bassil, C. e Laouiti, A. (2017). Vanet security challenges and solutions: A survey. *Vehicular Communications*, 7:7–20.

Jin, H. e Papadimitratos, P. (2015). Scaling vanet security through cooperative message verification. Em *IEEE Vehicular Networking Conference (VNC)*. IEEE.

Karagiannis, G., Altintas, O. e Ekici, E. (2011). Vehicular networking: A survey and tutorial on requirements, architectures, challenges, standards and solutions. *IEEE Communications Surveys & Tutorials*, 13(4):584–616.

Keranen, A. (2008). *Opportunistic Network Environment simulator*. Tese de doutorado, Helsinki University of Technology, Helsinki - Finland.

Lim, K., Tuladhar, K. M., Wang, X. e Liu, W. (2017). A scalable and secure key distribution scheme for group signature based authentication in vanet. Em *2017 IEEE 8th Annual Ubiquitous Computing, Electronics and Mobile Communication Conference (UEMCON)*, páginas 478–483. IEEE.

Mejri, M. N., Ben-Othman, J. e Hamdi, M. (2014). Survey on vanet security challenges and possible cryptographic solutions. *Vehicular Communications*, 1(2):53–66.

Mishra, R., Singh, A. e Kumar, R. (2016). Vanet security: Issues, challenges and solutions. Em *International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT)*. IEEE.

Naresh, V. S. e Murthy, N. V. (2015). Elliptic curve based dynamic contributory group key agreement protocol for securegroup communication over ad-hoc networks. *International Journal of Network Security,*, 17(5):588–596.

Qu, F., Wu, Z., Wang, F.-Y. e Cho, W. (2015). A security and privacy review of vanets. *IEEE Transactions on Intelligent Transportation Systems*, 16(6):2985–2996.

Stojmenovic, I. (2002). *Handbook of Wireless Networks and Mobile Computing*. JOHN WILEY & SONS, INC.

Sun, J., Zhang, Y. e Fang, Y. (2010). An identity-based security system for user privacy in vehicular ad hoc networks. *IEEE Transactions on Parallel and Distributed Systems*, 21(9):1227–1239.

Tan, H. e Chung, I. (2019). Secure authentication and key management with blockchain in vanets. *IEEE Access*, 8:2482–2498.

Tzeng, S.-F., Horng, S.-J., Li, T., Wang, X., Huang, P.-H. e Khan, M. K. (2017). Enhancing security and privacy for identity-based batch verification scheme in vanets. *IEEE Transactions on Vehicular Technology*, 66(4):3235–3248.

Ullah, I., Wahid, A., Shah, M. A. e Waheed, A. (2017). Vbpc: Velocity based pseudonym changing strategy to protect location privacy of vehicles in vanet. Em *International Conference on Communication Technologies (ComTech)*. IEEE.

Whyte, W., Weimerskirch, A., Kumar, V. e Hehn, T. (2013). A security credential management system for v2v communications. Em *2013 IEEE Vehicular Networking Conference*, Boston, MA, USA.

Zhang, C., Xue, X., Feng, L., Zeng, X. e Ma, J. (2019). Group-signature and group session key combined safety message authentication protocol for vanets. *IEEE Access*, 7:178310–178320.

Zhang, L., Meng, X., Choo, K.-K. R., Zhang, Y. e Dai, F. (2018). Privacy-preserving cloud establishment and data dissemination scheme for vehicular cloud. *IEEE Transactions on Dependable and Secure Computing*, 17(3):634–647.