Chapter 6
Security in Vehicular Ad Hoc Networks (VANETs)

Weidong Yang[1]

6.1 Introduction

6.1.1 Overview

As an important component of the intelligent transportation system (ITS) and a novel form of mobile ad hoc network, Vehicular Ad Hoc Networks (VANETs) have attracted much attention from government, academic institutions and industry. In the U.S., the Federal Communications Commission (FCC) has allocated 75 MHz (5.85-5.925 GHz) in the 5.9 GHz band as a new Dedicated Short Range Communications (DSRC) spectrum for vehicular communication. In Europe, the European Telecommunications Standards Institute (ETSI) has also allocated a radio spectrum of 30 MHz (5.875-5.905 GHz) at 5.9 GHz. Similar bands exist in Japan. IEEE has also formed the new IEEE 802.11p task group[1], which focuses on DSRC PHY and MAC layer standard for Wireless Access for the Vehicular Environment (WAVE). Based on the IEEE 802.11p, a higher layer standard IEEE 1609 has been released for trial use[2]. Besides such efforts, many national and international projects devoted to VANETs, such as, the Research and Innovative Technology Administration (RITA) in the United States, the Car-to-Car Communication Consortium (C2C-CC) in European, and the Advanced Safety Vehicle Program (ASV) in Japan.

As is shown in Fig. 6.1, A VANET is a distributed, self-organizing communication network built up by moving vehicles, which contain both Inter-Vehicle (V2V) communications between vehicles and Vehicle-to-Roadside (V2R) communications between vehicles and roadside units (RSUs)[3]. The applications of VANETs can be divided into two major categories: safety

[1] Henan University of Technology, China.

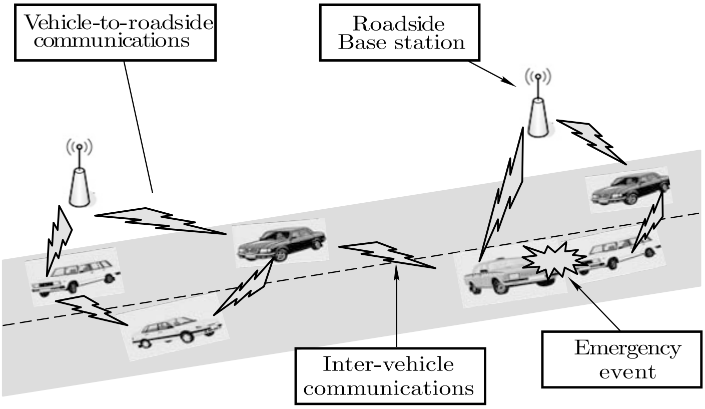

Vehicle-to-roadside communications
Roadside Base station
Inter-vehicle communications
Emergency event

Fig. 6.1 An illustration of VANETs.

and non-safety. Safety applications include collision and other safety warnings, which can be further categorized as safety-critical and safety-related applications. Non-safety applications include real-time traffic congestion and routing information, high-speed tolling, mobile infotainment, and many others.

Despite the fact that vehicles are organized mostly in an ad hoc manner, VANETs have significantly different characteristics compared to traditional mobile ad hoc networks (MANETs). The characteristics of VANETs are given as follows[4,5].

- *Applications.* While most MANET articles do not address specific applications, the common assumption in MANET literature is that MANET applications are identical (or similar) to those enabled by the Internet. In contrast, as we showed above, VANETs have completely different applications.
- *Energy Efficiency.* While in MANETs a significant body of literature is concerned with power-efficient protocols, VANETs enjoy a practically unlimited power supply.
- *Addressing.* Faithful to the Internet model, MANET applications require point-to-point (unicast) with fixed addressing; that is, the recipient of a message is another node in the network specified by its IP address. However, VANET applications often require dissemination of the messages to many nodes (multicast) that satisfy some geo-graphical constraints and possibly other criteria (e.g., directions of movement).
- *Mobility Model.* In MANETs, the random waypoint (RWP)[6] is (by far) the most commonly employed mobility model. However, most existing literature recognized that RWP would be a very poor approximation of real vehicular mobility. When designing a simulation environment, proper vehicular mobility models must be defined in order to produce realistic mobility patterns.

- *Frequent link disconnections.* Unlike nodes in MANETs, vehicles generally travel at much higher speeds, especially on highways. Ascribed to high mobility of vehicles, the topology of a VANET changes rapidly from time to time, causing intermittent communication links.
- *Availability of location information.* Satellite navigation systems are becoming more prevalent in vehicular transportation these days. Making good use of location information by GPS in communication service provision not only can reduce delivery latency of message dissemination (i.e., for road safety services) but also can increase system throughput (i.e., for infotainment services).

The special behavior and characteristics of VANETs create some challenges for vehicular communication, which can greatly impact the future deployment of these networks. A number of technical challenges need to be resolved in order to deploy vehicular networks and to provide useful applications, especially in the aspects of security and privacy[7]. A VANET inherits all the known and unknown security weaknesses associated with MANETs, and could be subject to many security and privacy threats. It is obvious that any malicious behavior of users, such as a modification and replay attack with respect to the disseminated messages, could be fatal to the other users. In addition, the issues in VANET security become more challenging due to the unique features of networks, such as the high mobility of the nodes and the large scale of the network. Furthermore, privacy protection must be achieved in the sense that the user related privacy information, including the driver's name, license plate, speed, position, and traveling routes along with their relationships, has to be protected; while the authorities should be able to reveal the identities of message senders in case of dispute such as a crime/car accident scene investigation, which can be used to look for witnesses. Therefore, it is critical to develop a suite of elaborate and carefully designed security mechanisms for achieving security and privacy preservation in a VANET.

An overview of VANET security can be found in reference [8]. Various consortia are presently addressing VANET security and privacy issues, including the Crash Avoidance Metrics Partnership (CAMP) Vehicle Safety Communications-Applications project, the Vehicle Infrastructure Integration (VII) project, the SeVeCom project, the Embedded Security for Cars (ESCAR) Conference and others. The trial-use standard IEEE 1609.2 (previously named P1556) also addresses security services for VANETs.

6.1.2 VANET Security and Privacy Requirements

The security requirements are derived from primary security goals like confidentiality, integrity and availability. From a review of existing literature[7,14], the general security requirements of a VANET can be derived as authentication, integrity and consistency, confidentiality, availability, access control, non-repudiation and privacy. A security system for safety messaging in a VANET should satisfy the following requirements.

1. *Authentication*

Authentication is a major requirement in VANET as it ensures that the messages are sent by the actual nodes and hence attacks done by the greedy drivers or the other adversaries can be reduced to a greater extent. Authentication in the VANET can be divided into two categories: ID authentication and entity authentication. ID authentication ensures that a message is trustable by correctly identifying the sender of the message. With ID authentication, the receiver is able to verify a unique ID of the sender. The ID could be the license plate or chassis number of the vehicle. Vehicle reactions to events should be based on legitimate messages (i.e., generated by legitimate senders). Therefore, we need to authenticate the senders of these messages. Entity authentication ensures that the recently received message is fresh and live. It ascertains that a message is sent and received in a reasonably small time frame.

2. *Integrity and Consistency*

Integrity requirements demand that the information from the sender to the receiver must not be altered or dropped. The legitimacy of messages also encompasses their consistency with similar ones (those generated in close space and time), because the sender can be legitimate while the messages contains false data.

3. *Confidentiality*

Confidentiality requires that the information flowing from sender to receiver should not be eavesdropped. Only the sender and the receiver should have access to the contents of the message, e.g. instant messaging between vehicles.

4. *Availability*

In safety applications like post-crash warning, the wireless channel has to be available so that approaching vehicles can still receive the warning messages. If the radio channel goes out (e.g. jamming by an attacker), then the warning cannot be broadcasted and the application itself becomes useless. Hence availability should be also supported by alternative means.

5. *Access Control*

Access control is necessary for an application that distinguishes between different accessing levels of a node or infrastructure component. This is established through specific system-wide policies, which specifies what each node is allowed to do in the network. For instance, an authorized garage may be allowed to fully access wireless diagnostics, whereas other parties may only be granted limited accesses. Another form of access control can be the exclusion of misbehaving nodes (e.g. by an intrusion detection system using a trust management scheme) from the VANET by certificate revocation or other means.

6. *Non-repudiation*

Drivers causing accidents should be reliably identified. A sender should not be able to deny the transmission of a message (it may be crucial for investigation to determine the correct sequence and content of messages exchanged before the accident).

7. *Privacy*

Privacy is an important factor for the public acceptance and successful deployment of VANETs. With vehicular networks deployed, the collection of vehicle-specific information from overheard vehicular communications will become particularly easy. Then inferences on the drivers' personal data could be made, and thus violate her or his privacy. The vulnerability lies in the periodic and frequent vehicular network traffic messages which will include, by default, information (e.g., time, location, vehicle identifier, technical description, trip details) that could precisely identify the originating node (vehicle) as well as the drivers' actions and preferences. Hence, the privacy of drivers against unauthorized observers should be guaranteed.

6.1.3 Security Threats in Vehicular Ad Hoc Networks

A VANET can be compromised by an attacker from manipulating either vehicular system or the security protocols. Hence two kinds of attacks can be visualized against vehicular systems: attacks against messages and attacks against vehicles. Next, we explore the most significant vulnerabilities of vehicular communications.

1. *In the case of an accident*

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

2. *In-transit traffic tampering*

Any node acting as a relay can disrupt communications of other nodes: it can drop or corrupt messages, or meaningfully modify messages, so that the reception of valuable or even critical traffic notifications or safety messages can be manipulated. Moreover, attackers can replay messages (e.g., to illegitimately obtain services such as traversing a toll check point). In fact, tampering with in-transit messages may be simpler and more powerful than forgery attacks.

3. *Masquerading*

The attacker actively pretends (impersonates) to be another vehicle by using false identities and can be motivated by malicious or rational objectives. Message fabrication, alteration, and replay can also be used towards masquerading. A masquerader can be a threat: consider, for example, an attacker masquerading as an emergency vehicle to mislead other vehicles to slow down and yield.

4. *Privacy violation*

With vehicular networks deployed, the collection of vehicle specific information from overheard vehicular communications will become particularly easy. Then inferences on the drivers' personal data could be made, and thus violate her or his privacy. The vulnerability lies in the periodic and frequent vehicular network traffic. In all such occasions, messages will include, by default, information (e.g., time, location, vehicle identifier, technical description, trip details) that could precisely identify the originating node (vehicle) as well as the drivers' actions and preferences.

5. *Denial of Service (DoS)*

The attacker may want to bring down the VANET or even cause an accident. There are many ways to perform this attack, either by sending messages that would lead to improper results or by jamming the wireless channel (this is called a Denial of Service, or DoS attack) so that vehicles cannot exchange safety messages.

6. *Hidden vehicle*

In this scenario, a vehicle broadcasting warnings will listen for feedback from its neighbors and stop its broadcasts if it realizes that at least one of these neighbors is better positioned for warning other vehicles. This reduces congestion on the wireless channel. A hidden vehicle attack consists in deceiving vehicle A into believing that the attacker is better placed for forwarding the warning message, thus leading to silencing A and making it hidden (has stopped broadcasting).

7. *Tunnel*

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

8. *Sinkhole attack*

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

9. *Wormhole attack*

The attacker connects two distant parts of the ad hoc network using an extra communication channel as a tunnel. As a result, two distant nodes assume they are neighbors and send data using the tunnel. The attacker has the possibility of conducting a traffic analysis or selective forwarding attack. This also extends the range of the attacker.

10. *Sybil attack*

Large-scale peer-to-peer systems face security threats from faulty or hostile remote computing elements. To resist these threats, many such systems employ redundancy. However, if a single faulty entity can present multiple identities, it can control a substantial fraction of the system, thereby undermining this redundancy. The Sybil attack especially aims distributed system environments. The attacker tries to act as several different identities/nodes rather than one. This allows him to forge the result of a voting used for threshold security methods.

11. *On-board tampering*

Other than communication protocols, an attacker may select to tinker with data (e.g., velocity, location, status of vehicle parts) at their source, tampering with the on-board sensing and other hardware. In fact, it may be simpler to replace or by-pass the real-time clock or the wiring of a sensor, rather than modifying the binary code implementation of the data collection and communication protocols.

6.2 Security Architecture Framework for Vehicular Ad Hoc Networks

6.2.1 Overview

Currently, many of the researches in VANET are paying more attention on the development of a proper MAC layer (the definition of the MAC and physical layer protocols has greatly progressed, for instance IEEE 802.11p, a specially designed version of IEEE 802.11) rather than security architecture and protocols for VANET. The most prominent industrial effort in this domain is carried out by Car 2 Car Communication Consortium, the IEEE 1609.2 working group, the NoW project and the SeVeCom project with all of them developing VANET Security architecture. All of them take the use of Certification Authority (CA) and public key cryptography to protect V2V and V2I messages as their basic elements. It has now become an established consensus that public key cryptography is the way to go about for VANETs. This is mainly due to the fact that the messages are broadcasted and one-to-one communication is not the norm. Due to this fact, symmetric key cryptography will incur huge costs in frequent key establishment procedures and they are also difficult to implement as the nodes are constantly on the move. For all the perspective security protocols, message authentication, integrity and non-repudiation, as well as protection of private user information are identified as primary requirements.

6.2.2 PKI for Vehicular Ad Hoc Networks

Vehicles are registered in different states and they're huge in numbers. They will travel long distances so that they can be well beyond their registration areas. All of these are requiring a robust and flexible key management scheme. The involvement of authorities in vehicle registration implies the need for a certain level of centralization. Vehicles not only have to be identified by base stations, but also have to be identified by each other (without invoking any server), so that communications by base station (as in cellular networks) is not enough for VC, and this creates a problem of scalability. In addition, symmetric cryptography does not provide the non-repudiation property that allows the accountability of drivers' actions (e.g., in the case of accident reconstruction or finding the originator of forgery attacks). Hence, the use of public key cryptography is a more suitable option for deploying vehicular communications security.

This implies the need for a public key infrastructure (PKI). As stated above, VANET is usually a hybrid network with the possibility to access a stationary network at least temporarily, so that using a centralized PKI approach with a TTP which issues certificates and revokes them is an appropriate idea. Therefore, a PKI with the certification authority CA (the trust center) is used to introduce trust within the network. Fig. 6.2 shows the basic setup of the PKI. In order to communicate, a node (CA) has to be registered at the trust center. By fulfilling the registration process, the vehicles can get a certificate signed with the key of the CA. The CA is responsible for checking if the right vehicle get the right key and if the vehicle is worthy of trusting before issuing the signed certificate. Every subscriber within the network knows the public key of the CA and can check the validity of any public key certificate issued by the CA. Therefore, any two vehicles can exchange and validate their public keys without having access to any other node or gateway. If the certificates are valid, the vehicles can trust each other and establish a secure connection.

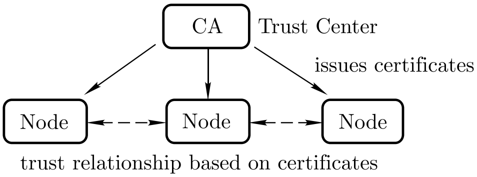


Fig. 6.2 Public key infrastructure.

6.2.3 Trusted Architecture for Vehicular Ad Hoc Networks

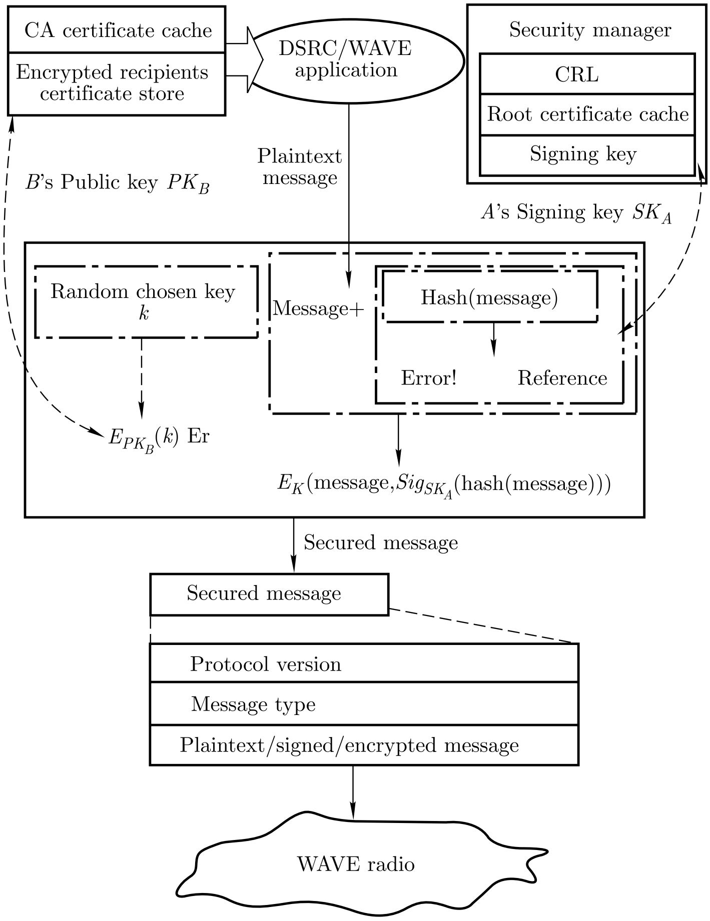
In this subsection, we first discuss IEEE 1609.2 standard which specifies methods of securing Wireless Access in Vehicular Environment (WAVE) messages against various attacks. We then describe security architecture for VANETs based on the PKI and security hardware[9]. A secure VANET communication scheme based on TPMs[10] is also given.

6.2.3.1 IEEE 1609.2 Security Framework

The IEEE 1609 communication standards, also known as Dedicated Short Range Communications (DSRC) protocols, have emerged recently to enhance 802.11 to support wireless communications among vehicles for the roadside infrastructure. The IEEE 1609.2 standard addresses the issues of securing WAVE messages, in order to fight against eavesdropping, spoofing, and other attacks. The components of the IEEE 1609.2 security infrastructure which are based on industry standards for public key cryptography, includes support for elliptic curve cryptography (ECC), WAVE certificate formats, and hybrid encryption methods, in order to provide secure services for WAVE communications that are shown in Fig. 6.3. To support core security functions such as certificate revocation, the security infrastructure is also needed to be responsible for the administrative functional necessities. Note that certificate revocation is essential to any security system based on the public key infrastructure, which has not been addressed in the current IEEE 1609.2 by considering the unique features of vehicular networks. In addition, IEEE 1609.2 does not define driver identification and privacy protection, and has left a lot of issues open.

6.2.3.2 Security architecture based on security hardware and the PKI

Here security hardware means two hardware modules among the vehicle onboard equipment for security, namely the event data recorder (EDR) and the tamper-proof device (TPD). Whereas the EDR only provides tamper-proof storage, the TPD also possesses cryptographic processing capabilities. The EDR has the function of recording the vehicle's critical data, such as position, speed, time, etc., during emergency events, similar to an airplane's black box.



*Assume that A is the sender and B is the receiver

Fig. 6.3 The IEEE 1609.2 security services framework for creating and exchanging WAVE messages between WAVE devices.

These data are useful in accident reconstruction and the attribution of liability. EDRs have already been installed in a lot of road vehicles, especially trucks. These can also record the safety messages received if critical events happen.

An owner or a mechanist can easily accesse the vehicle electronics, especially the data bus system. Therefore, the cryptographic keys of a vehicle need proper hardware protection, namely a TPD. The TPD will take care of storing all the cryptographic material and performing cryptographic operations, especially signing and verifying safety messages. After connecting a set of cryptographic keys to a given vehicle, the TDP guarantees the accountability property as long as it remains inside the vehicle. The TPD needs to be independent from its external environment. It should own its own clock and have a battery that is periodically recharged from the vehicle's electric circuits. The general secure architecture based on security hardware and PKI is given by Fig. 6.4.

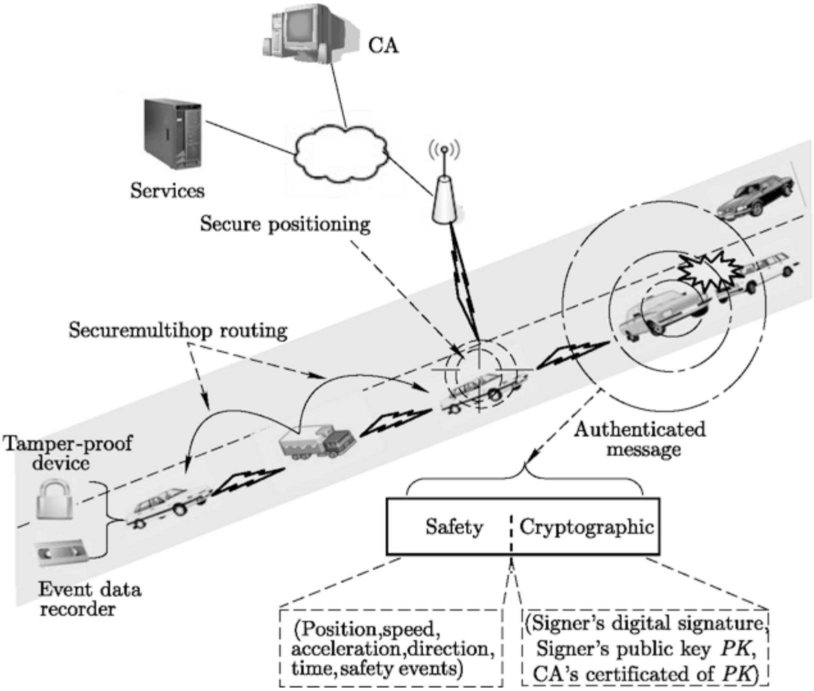

CA
Services
Secure positioning
Securemultihop routing
Tamper-proof device
Event data recorder
Authenticated message
Safety
Cryptographic
(Position,speed, acceleration,direction, time,safety events)
(Signer's digital signature, Signer's public key *PK*, CA's certificated of *PK*)

Fig. 6.4 Security architecture based on security hardware and the PKI.

6.2.3.3 Secure VANET communication scheme based on TPMs

The Trusted Platform Module (TPM)[11] can be integrated into the vehicle onboard equipment for implementing the security requirements. It is a general purpose of hardware chip designed for secure computing. A TPM is a piece of hardware, requiring a software infrastructure, which is able to protect and store data in shielded locations. A TPM has also cryptographic capabilities such as a SHA-1 engine, an RSA engine, and a random number generator. Fig. 6.5 illustrates the main components of a TPM.

Here are the two levels where the security model are using TPMs to secure VANET works. The first level permits a trusted channel to be established between any two vehicles. This means that the two vehicles are satisfied that each is running an untampered version of the security software, and that no intentional data attack or Sybil attack is being attempted. The second level aims at information verification. It builds on trusted channels, and is to ensure that a vehicle's configuration does not contain erroneous readings.

Implementing trusted channels relies directly on the TPM's attestation

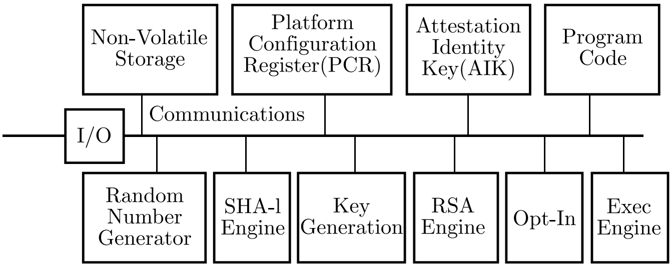


Fig. 6.5 Architecture of a TPM.

mechanism. A vehicle can trust another if the latter can demonstrate that its software has not been tampered and the source of the software can be verified. The issue in deploying a TPM on VANET nodes is to assign roles to the actors in the TPM protocols. In reference [10] the following are assumed:

(1) Car manufacturers sign the platform credentials for their vehicles. To assume that a manufacturer takes responsibility for all embedded devices on their vehicles is rational. Further, manufacturers are relatively few in numbers and are well-known in the sense that certificates signed by these principals should be recognizable to all vehicles and automobile authorities.

(2) Automobile authorities are responsible for organizing technical reviews. In most countries, car owners are obliged to submit their cars to a technical review every 2 to 3 years. If a car fails the technical review, it cannot be driven on the road. Automobile authorities are thus well-known principals that can act as privacy CAs that can sign AiK credentials.

The TPM gives us a means to securely attribute a vehicle identifier. This can be signed by an automobile authority. When vehicles exchange messages, we can use the attestation protocol and then ensure the integrity and authenticity of these messages.

The second level of security, which is information verification, is based on three simple procedures.

(1) Auto-measuring. A vehicle's software maintains data on the vehicle's acceleration and deceleration capabilities, as well as related data such as tire denseness (which embedded devices are now able to measure). These values evolve so the vehicle continuously updates them. These values are obviously important for the platoon scenario where neighboring vehicles need to agree on minimal distances.

(2) Challenge-response protocol. This procedure is needed to find out unintentional errors in information transmitted by a vehicle that are due to permanent errors in the sensor of the vehicle. Vehicles that are close together should possess the same readings for many information types, for example, such as temperature, time, and location. It aims to permit a vehicle to challenge another with respect to any of these readings.

(3) Technical review. The automobile authorities organize technical reviews. The vehicles with VANET functionality must include reviews of the

correct functions of all sensor devices. Further, it is expected that any changes that need to be made to the application software are made at this moment. Since the TPM can only be used to help verify that the software on a platform has not been tampered with, it is very important to know that the absence of security flaws or bugs in the software itself does not guarantee. The three procedures conduce to detect and isolate permanent errors in readings.

Figure 6.6 shows the different components of the embedded architecture and the data flow. As it shows, for instance, in auto-measuring, sensors embedded in vehicle give results of their measures to the application. Then the application asks the TPM to sign the data. The TPM checks the PCR value associated with this application and signs data provided by the application. Then it will store this data in a dedicated repository.

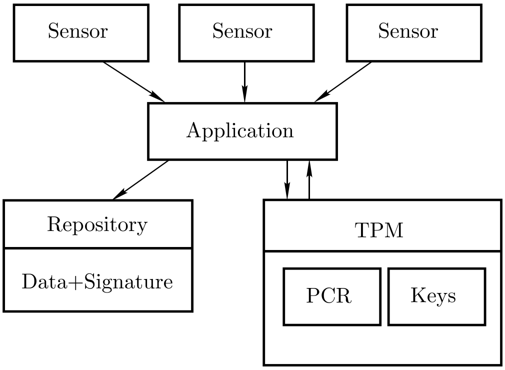


Fig. 6.6 The embedded architecture.

In order to detect unintentional errors, the details for challenging another vehicle are given in Fig. 6.7. The challenger sends a query about data it

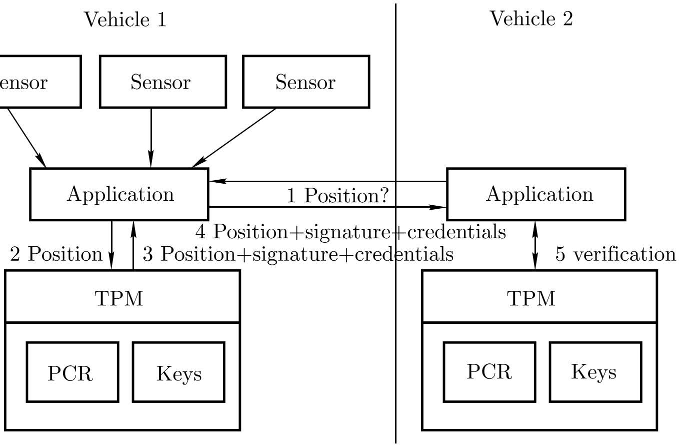


Fig. 6.7 The challenge-response protocol.

can verify, the current position in the example. Then the challenged vehicle collects the appropriate data, and gives this data to its TPM. The TPM checks the PCR values associated with this application and signs data. The application sends to the challenger the signed data and associated credential. The challenger verifies the signature and compares the given position to its own current position to detect misconfiguration of the positioning unit of the challenged vehicle.

6.2.4 Key Management and Authentication Scheme

In this section we present the scheme of key management and authentication under the security architecture based on the PKI and security hardware.

6.2.4.1 Key Management

We will address below the issues of cryptographic key distribution, certification, and revocation.

1. *Cryptographic Information Types and Key Distribution*

To be part of a VANET, each vehicle has to store the following cryptographic information[12]:

(1) an electronic identity called an electronic license plate (ELP) issued by a government, alternatively an electronic chassis number (ECN) issued by the vehicle manufacturer. These identities (further referred to simply by ELP) should be unique and cryptographically verifiable (this can be achieved by attaching a certificate issued by the CA to the identity) in order to identify vehicles to the police in case this is required (identities are hidden from the police). It seems to the physical license plates, the ELP should be changed when the owner changes or moves, e.g., to a different region or country.

(2) Anonymous key pairs that are used to preserve privacy. An anonymous key pair is a public/private key pair that is authenticated by the CA. However it contains neither information about nor public relationship with (i.e., this relationship cannot be discovered by an observer without a special authorization) the actual identity of the vehicle (i.e., its ELP). Usually, a vehicle will possess a set of anonymous keys to prevent tracking.

Now the ELP is the electronic equivalent of the physical license plate, it should be installed in the TPD of the vehicle onboard equipment using a similar procedure. It means that the governmental transportation authority will preload the ELP at the time of vehicle registration (in the case of the ECN, the manufacturer is responsible for its installation at production time).

The transportation authority or the manufacturer preloads anonymous keys. Besides, while ELPs are fixed and should accompany with the vehicle for a long duration, anonymous key sets have to be periodically renewed after all the keys have been used or their lifetimes have expired. During the periodic vehicle checkup (typically yearly) or by similar procedures this renewal can be done.

Over and above, the ELP and anonymous keys, each vehicle should be preloaded with the CA's public key.

2. *Key certification*

CA will be responsible for issuing key certificates to vehicles. Here are two solutions.

(1) Governmental transportation authorities: The corresponding transportation authorities (which are usually regional) will register vehicles in different countries. The advantage of this option is that the certification procedure will be under the direct control of the concerned authority. Although the ELP and keys of each vehicle are certified by a regional authority in a given country, vehicles from different regions or countries should be able to authenticate each other. This problem is usually solved by including the certificate chain leading to a common authority, but in the case of VANET, it would tremendously increase the message overhead. This certificate chain can be replaced by a single certificate by making the CA of the traveling vehicle's transit. Also can destination region recertify the ELP and the anonymous keys of the vehicle after verifying them with the public key of the CA that registered the vehicle? This requires the installation of base stations at the region borders.

(2) Vehicle manufacturers: Considering the limited number and the trust already endowed in them, certificates can also be issued by vehicle manufacturers. The advantage of this approach is to reduce overhead. In fact, in order to be able to verify any other vehicle it encounters, which is not the case if the CA is a local authority, each vehicle will need to store a small number of manufacturer public keys. However, this approach could lead to non-governmental institutions being involved in law enforcement mechanisms.

3. *Key revocation*

The owner's identity, certified and issued by a CA, is connected with the public key by a public key certificate. Various attacks including man-in-the-middle attacks and impersonation attacks can be effectively prevented with the help of a public key certificate. However, a user's certificate could be repealed due to some unexpected reasons. For instance, to maintain system security, the certificate should be repealed once the private key corresponding to the public key specified in the certificate is identified as compromised.

The traditional PKI architecture use certificate revocation scheme most through the certificate revocation list (CRL), a list of revoked certificates stored in central repositories prepared in CAs. Based on such centralized architecture, alternative solutions to CRL could be a certificate revocation system (CRS), certificate revocation tree (CRT), the Online Certificate Status Protocol (OCSP)[13], and other methods. Usually, these schemes are required to be highly available of the centralized CAs, where frequent data transmission with vehicles to obtain timely revocation information may cause significant overhead. Therefore, the centralized CRL architecture may only exist in fantasy with the high-speed mobility and large quantity of network entities in VANETs.

In order to solve the problem, Lin et al.[14] came up with a novel RSU-aided certificate revocation (RCR) mechanism for performing certificate revocation. As illustrated in Fig. 6.8, there are three types of network entities: the authority (denoted as CA), RSUs, and vehicles. The relationship between these three is explained as follows. The CA manages the RSUs, and both of them are assumed to be trustworthy. The RSUs are connected to the Internet through either wired Ethernet or WiMAX, or any other networking technology. Furthermore, the CA provides each RSU a secret key, while the corresponding public key is an identity string containing the name of the RSU, the physical location, and the authorized message type. By this approach, the messages can be signed by an RSU with the help of an identity-based signature.

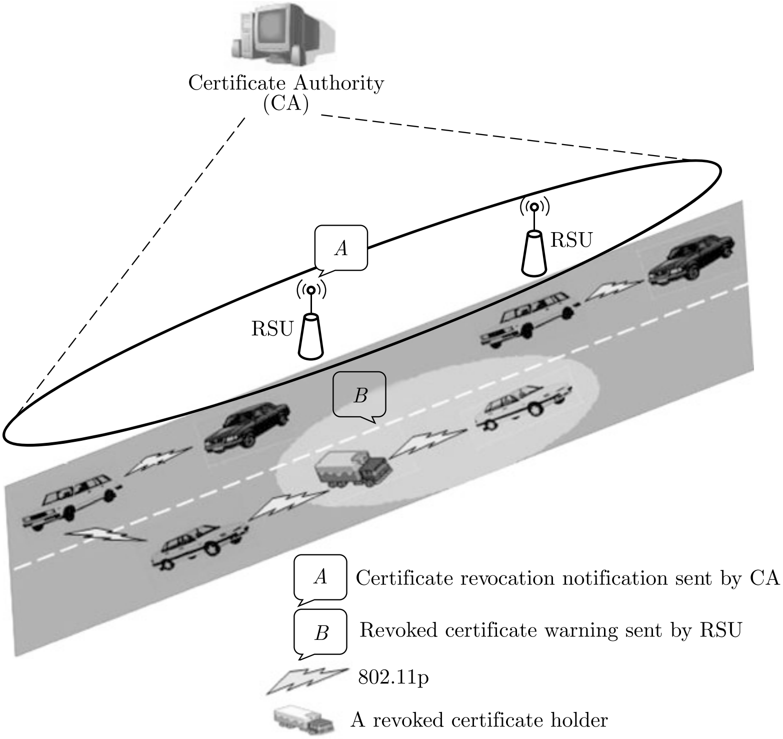

Certificate Authority
(CA)
A
RSU
RSU
B
A
Certificate revocation notification sent by CA
B
Revoked certificate warning sent by RSU
802.11p
A revoked certificate holder

Fig. 6.8 The RSU-aided key revocation scheme.

The CA will inform all the RSUs about a certificate revocation once a certificate is revoked. Each RSU then checks the status of the certificates contained in all the messages broadcasted by the passing vehicles. If a certificate has been confirmed as revoked, the RSU will broadcast a warning message such that all other approaching vehicles can update their CRLs and avoid communicating with the compromised vehicle. Since vehicle movement can be predicted based on its driving conditions (e.g., direction, speed, position), the RSU can further notify all neighboring RSUs of where the compromised vehicle may go. In addition, due to RSUs' normally sparse location, a rather limited number of vehicles will be notified even if all the RSUs broadcast the corresponding message. Therefore, in order to make the warning message disseminate more effective, the warning message among vehicles can be forwarded through inter-vehicle communications, that is to say, disseminated by each vehicle, hop by hop, throughout its predefined lifetime.

However, in order to avoid being detected, while passing through an RSU, a compromised vehicle may intentionally disable message broadcasting. This is also referred as a silent attack, which can easily be handled by granting every RSU the privilege of signing the certificate of each vehicle. In this case, whenever a vehicle passes through an RSU, the vehicle asks the RSU to sign its certificate, where the signature serves as evidence that can demonstrate its authenticity and legitimacy to other vehicles. The corresponding messages will be ignored if a vehicle is using a certificate that has not been verified by an RSU for a certain period of time and is discovered by a neighbor vehicle. Therefore, according to resisting compromised vehicles, the VANET can gain the security and safety with the least amount of effort.

6.2.4.2 Authentication Scheme

The safety messages in VANETs can be classified into three classes[12], based on their properties related to privacy and real-time constraints, as shown in Table 6.1. Traffic information messages are used to disseminate traffic conditions in a given region and thus affect public safety only indirectly (by preventing potential accidents due to congestion); hence they are not time-critical. General safety-related messages are used by public safety applications such as cooperative driving and collision avoidance and hence should satisfy stringent constraints such as an upper bound on the delivery delay. Liability-related messages are distinguished from the previous class because they are exchanged in liability-related situations such as accidents. Therefore, the liability of the message originator should be determined by revealing his identity to the law enforcement authorities. A common property of all the message classes is that they are mainly standalone and there is no content dependency among them. Apart from data specific to traffic events, position, speed, direction, and acceleration of the vehicle are also concluded within a typical safety message. In case the sender is trapped in an abnormal situation, for instance, an accident, these data would help receivers compute their positions concerning the sender and examine if they are in danger.

All the message classes share another common property, in which they don't contain any sensitive information, and confidentiality is not required. As a result, the exchange of safety messages in a VANET needs authentication but not encryption. For message authentication, there is a simplest and most efficient way, which is, assigning to each vehicle a set of public/private key pairs, that will allow the vehicle to digitally sign messages and thus authenticate itself to receivers.

Table 6.1 Message classes and properties

Class/Property	Legitimacy	Privacy Protection		Real-time
		Against Others	Against Police	Constraints
Traffic Information	yes	yes	yes	
General Safety Messages	yes	yes	yes	yes
Liability-Related Messages	yes	yes		yes

A practical authentication scheme is shown as follows. Before a vehicle sends a safety message, it signs it with its private key and includes the CA's certificate as follows:

$$V \rightarrow * : M, sigSK_V[M|T], Cert_V$$

where V designates the sending vehicle, $*$represents all the message receivers, M is the message which is actually hashed before being signed, SK_V is V 's private key, $|$ is the concatenation operator, and T is the timestamp to ensure message freshness (it can be obtained from the security device TPD). It should be noted that, because of the burden of the inherent preliminary handshake where the communicating parties exchange the nonces, using nonces instead of timestamps is not desirable. Using sequence numbers also incurs overhead as they need to be maintained. $Cert_V$ is the public key certificate of V later.

Using the certificate, the receivers of the message have to extract and verify the public key of V, and then verify V 's signature using its certified public key. In order to do this, the receiver should have the public key of the CA, which can be preloaded as described above. If the message is sent in an emergency context, which means that it belongs to the liability-related class, this message should be stored (including the signature and the certificate) in the EDR for further potential investigations in the emergency.

This authentication scheme has failed in taking the scalability issue and resulted communication overhead into consideration. Furthermore attaching a digital signature and a certificate to each safety message for the sake of security inevitably creates overhead that can be larger than the message itself. Therefore Zhang et al.[15] proposes an RSU-aided message authentication scheme, called RAISE, which explores the unique features of VANETs by employing RSUs to assist vehicles in authenticating messages. With RAISE, when an RSU is detected nearby, vehicles start to associate with the RSU. Then, the RSU assigns a unique shared symmetric secret key and a pseudo ID that is shared with other vehicles. With the symmetric key, each vehicle generates a symmetric keyed-hash message authentication (HMAC) code, and then broadcasts a message by signing the message with the symmetric HMAC code instead of a PKI-based message signature. Other vehicles receiving the messages signed with the HMAC code are able to verify the message by using the notice about the authenticity of the message disseminated by the RSU.

The detailed implementation of RAISE is presented in the following. The notations are listed in Table 6.2 for ease of presentation.

Table 6.2 Notations

Notations	Descriptions
R_i:	the i-th RSU
V_i:	the i-th vehicle
M_i:	the message sent by V_i
K_i:	the key shared between V_i and R_i
ID_i:	a pseudo identity of V_i assigned by R
U:	an entity, which could be an RSU R or a vehicle V_i
T:	the current time
PK_U:	the public key of U
SK_U:	the private key of U
C_U:	U's certificate
$\{m\}_{SK_U}$:	U's digital signature on m
$H(\cdot)$:	a one-way hash function such that SHA-1
$HMAC(\cdot)$:	a keyed-hash message authentication code
$\|\|$:	message concatenation operation

1. *Symmetric key establishment*

Once a vehicle V_i detects that there is an RSU R_i nearby, V_i initiates a mutual authentication process and establishes a shared secret key with R_i. This can be achieved by adopting the Diffie-Hellman key agreement protocol secured with public key based signature scheme. The mutual authentication and key agreement processes are shown as follows:

$$V_i \rightarrow R : g^a, \{g^a\}_{SK_{V_i}}, C_{V_i}$$
$$R \rightarrow V_i : ID_i \| g^b, \{ID_i \| g^a \| g^b\}_{SK_R}, C_R$$
$$V_i \rightarrow R : \{g^b\}_{SK_{V_i}}$$

where g^a and g^b are elements of the Diffie-Hellman key agreement protocol, and the shared key between R_i and V_i is $K_i \leftarrow g^{ab}$. When receiving the first message from V_i, R_i can verify V_i's public key PK_{V_i}, and then use PK_{V_i} to verify the signature $\{g^a\}_{SK_{V_i}}$ on g^a. In a similar manner, V_i authenticates R_i. If the above three flows succeed, the mutual authentication process is done. At the same time, in the second flow, R_i assigns a pseudo identity ID_i to the vehicle V_i. The pseudo ID is uniquely linked with K_i. With ID_i, R_i can know which vehicle sends the message, and can further verify the authenticity of the message with their shared symmetric key. Therefore, R_i maintains an ID-Key table in its local database.

2. *Hash aggregation*

Once the vehicle V_i obtains the symmetric key K_i from the RSU R_i, V_i uses K_i to compute the message authentication code $HMAC(ID_i||M_i)$ on $ID_i||M_i$, where ID_i is V_i's pseudo identity assigned by R_i and M_i is the message to be sent. Then, V_i one-hop broadcasts $ID_i||M_i||HMAC(ID_i||M_i)$. Since K_i is only known by R_i in addition to V_i itself, only R_i can verify M_i. Thus, to make other vehicles be able to verify the authenticity of M_i, and at the same time to reduce communication overhead, the RSU R_i is responsible to aggregate multiple authenticated messages in a single packet and to send it out. The detailed process is shown as follows:

(1) R_i checks whether the time interval between the current time and the time when R_i sent the last message authenticity notification packet is less than a predefined threshold. If so, go to Step 2. Otherwise, go to Step 4.

(2) When R_i receives a message, $ID_i||M_i||HMAC(ID_i||M_i)$, sent by the vehicle V_i, R_i first checks whether ID_i is in R_i's ID-Key table. If yes, go to Step 3. Otherwise, go to Step 4.

(3) R_i uses ID_i's K_i to verify $HMAC(ID_i||M_i)$. If it is valid, R_i computes $H(ID_i||M_i)$ and then go to Step 1. Otherwise, drop the packet.

(4) R_i aggregates all hashes generated at Step 3, i.e., $HAggt = H(ID_1||M_1) ||H(ID_2||M_2)||\ldots||H(ID_n||M_n)$, and signs it with its private key SK_{R_i}. Then, R_i one-hop broadcasts $HAggt||\{HAggt\}_{SK_{R_i}}$to vehicles within its communication range.

3. *Verification*

When the other vehicles sent messages to a vehicle, received vehicles only buffers the received messages in its local database without verifying them immediately. The buffered record has the following format: M_i, ID_i, $H(ID_i||M_i)$. Once vehicles obtain the signed packet $HAggt||\{HAggt\}_{SK_{R_i}}$ from the RSU, they are able to verify the buffered messages one by one. First, vehicles use the RSU's public key PK_{R_i} to verify the signature $\{HAggt\}_{SK_{R_i}}$. If it is valid, vehicles will check the validity of the previously received messages buffered in the record in the local database. This is done by comparing whether there is a match between the buffered record with the de-aggregate message.

A vehicle generates a HMAC for each launched message with RAISE. The HMAC can only be generated by the vehicle that has the key assigned by the RSU. When the adversary tempers a message, the RSU cannot find a responding validation key that can compute a matching HMAC for the message, and therefore the tempered message will be ignored. On the side, for each vehicle, there is a unique key stored in the ID-Key table in the RSU side. When an RSU finds out a key that can verify the HMAC, the RSU knows the identity of the message sender, and as a result the source is authenticated.

6.3 Secure Communication protocols for Vehicular Ad Hoc Network

6.3.1 Overview

There are many communication patterns in the VANET. Different communication patterns require different secure mechanisms to thwart security and privacy infringements. Therefore, we have to identify first which communication protocols will finally be used. In reference [16], the SeVeCom project extrapolates three basic communication patterns:

(1) Beaconing (Periodic, single-hop broadcasts, containing e.g. a vehicle's location, heading etc.).

(2) Restricted Flooding/Geocast (Multi-hop broadcast over a certain number of hops restricted by TTL or by specified geographic destination region).

(3) Geographic uni-cast routing (Multi-hop, hop-by-hop forwarding of packets, either for uni-cast end-to-end connections for any cast requests or for subsequent flooding/geocast in a remote destination region).

Basic questions about secure communication regard to which and how security mechanisms can be used to secure communication protocols, and how these security mechanisms can be integrated with the actual functional components, like the routing or medium access. Therefore, the usage of communication patterns instead of concrete protocols has the advantages that we stay independent of the implementation details and security mechanisms can easily be adapted to similar communication protocols.

6.3.2 Secure Beaconing

In VANETs, beaconing denotes a mechanism which broadcasts information periodically over a single hop, which means that they are not relayed by receiving nodes. Besides some identifiers, the information typically includes the vehicle's own position and additional information like speed or heading direction. Beacons are usually not forwarded, i.e. are consumed after one hop. This kind of communication is useful for instance for all cooperative awareness applications.

As a basic goal, a receiver needs to be able to verify authenticity and integrity of beacons. This means that a vehicle must be able to trust in the content of a beacon message in a way that

- the sender is actually a valid participant of the network (e.g., a vehicle, RSU, traffic sign, etc.),
- the identified sender has sent the message, not another one,
- the data is up-to-date,

- the data has not been altered.

During the setup of the system, the secure beaconing component is hooked into the data delivery path. When we assume network layer beaconing, the secure beaconing component is attached between the network and a link layer. Using this hook, the secure beaconing component will process the beacon data both upon sending and upon reception of a beacon.

When a beacon message is lined up to be sent, the hook redirects the message to the secure beaconing component. To be able to scan the content of the beacon, the message format must be known to the secure beaconing component, at least to some extent.

As mentioned earlier, typical beacons will include at least

- the current vehicle identifier (pseudonym) X,
- the current vehicle location *locX.*

In addition, the secure beaconing also requires a current time stamp (t_c) to be included in the beacon message in order to be able to ensure freshness of beacons.

For both efficiency and security reasons, these fields should not be duplicated in a beacon message. Hence, the implementation has to reuse existing fields. In case that the required fields are not included already, they have to be appended by the secure beaconing. Moreover, even if the required fields are already included, secure beaconing has to ensure that they comply with the security requirements. For instance, if the application has already added the field for the vehicle position, but this position information is not accurate enough for security reasons, another, appropriate location has to be appended by the secure beaconing.

Finally, the PAYLOAD should contain:

$$\text{PAYLOAD} = X|locX|\cdots$$

After these preprocessing steps, the component uses signing capabilities of the identification and trust management module. Moreover, the current time t_c is returned together with the signature, as the hardware security module provides a function to sign with timestamp.

After that, the beacon message will comprise payload, timestamp, signature, and certificate:

$$\text{BEACON} = \text{PAYLOAD}|t_c|sigSK_x(\text{PAYLOAD}|t_c)|certPK_x$$

The signed BEACON will then be returned into the data delivery path.

When a beacon arrives at a vehicle, it is passed over to the secure beaconing component via the hooking interface. The component will first check the attached signature by using the verify method of the identification management module. If the signature can be verified, further post-processing is applied, like the freshness check to prevent replay of old messages. If the signature is invalid, the message is either discarded immediately or marked as invalid by the component. The choice depends on whether applications also want to process invalid packets and should be configurable.

After the signature check, which includes a certificate check, it can be guaranteed that

- The message was sent once by the given sender X,
- The message has not been altered,
- The sender is a valid network participant.

Moreover, as the messages must not be replayed from vehicles passing by earlier, the freshness check needs to validate that the message's timestamp is recent. This requires determining the current time, which is provided by the hardware security module.

Noted that the freshness check should explicitly tolerate propagation delay, an allowed deviation of several seconds seems reasonable to prevent large-scale replay. This treatment also has the advantage that clocks do not need to be tightly synchronized. Nevertheless, if an older message is received, it is discarded.

Due to their high frequency, a number of challenges on the application of crypto mechanisms arise:

(1) Because of their frequency, beacons can cause a substantial part of the overall channel load. This situation is aggravated if every packet has to carry a complete set of security data like signature and certificate. Therefore, it would be desirable to reduce the channel load by more sophisticated security solutions. At the same time, each packet should be self-contained, i.e. authentication and integrity checks should be achievable without the context of other packets to allow for fast evaluation of time-critical packets.

(2) A similar problem due to high frequency of beacons originates from the computational requirements of asymmetric crypto operations. It is well known that creation and verification of asymmetric signatures can consume considerable amount of time. For example, if we assume beacons to be sent with frequency f and the current vehicle density is d, then f signature operations and $f \times d$ signature verifications have to be performed per second. Moreover, as some applications need time-critical communication to some extent, the sum of both the time for creation and verification plays a role.

6.3.3 Secure Restricted Flooding/Geocast

Flooding is an approach that is used for a number of applications in VANETs to distribute information very quickly among the immediate surroundings of a vehicle. The basic principle involves multi-hop broadcast forwarding, which means that every node rebroadcasts the message once. As this cannot be done network-wide, the rebroadcast is usually restricted by either a time-to-live (TTL) counter value or a geographic destination area (GDA).

The purpose of this security component is to ensure integrity, authenticity and reliability of this mechanism. As a primary goal, the component is intended to prevent malicious vehicles being able to disturb the mechanism by means of rerouting, tampering and dropping. As a secondary goal, the module should be able to cope with attacks that intend to exploit the flooding mechanism to disturb the whole network operativeness. This is particularly important since flooding is a relatively costly mechanism that consumes a lot of bandwidth especially when node density is high.

Different actions need to be taken depending on whether a packet is incoming or outgoing. And in this case if it is created by the current node or forwarded only. Moreover, the applied security mechanisms partly depend on the mechanism used, i.e. whether the flooding restriction is TTL-based or GDA-based. An outgoing message may either originate from the current node or is to be forwarded by the current node. The required security processing differs notably.

For all messages created by one of the applications of node X, a signature has to be computed and a timestamp t_c has to be added if not already included. If the forwarding is TTL-restricted, then also a hash chain mechanism has to be applied, because malicious forwarders could decrease the TTL and thus increase the multi-hop propagation area. Such an increase leads, of course, to waste network bandwidth. If the restriction is given by a fixed geographic destination region, the hash chain is not necessary.

Hence, the first step is to include a timestamp t_c or to ensure that an accurate timestamp is already included. This is done together with the signature. The second step is to compute the hash chain in case of TTL-restricted forwarding. Therefore, the component has to generate a random base value v, apply a hash function TTLMAX times on it and append the result h_v as well as v to the message. As third step, the signature has to be created and the certificate for the used key (long term ID or pseudonym) has to be attached. For this step, it is important to distinguish between mutable and immutable fields (F_{m} and F_{im}). Fields like the TTL value or the hash chain base value v change during the forwarding, whereas other, immutable fields such as the payload, the source address or the end of the hash chain h_v does not change.

The signature should only be computed for these immutable fields, and not include mutable ones. For GDA-restricted forwarding, the message looks like this:

$$F_{\mathrm{im}} = \mathrm{PAYLOAD}|X|\mathrm{GDA}|t_c$$
$$\mathrm{PACKETGDA} = F_{\mathrm{im}}|sigSK_x(F_{\mathrm{im}})|certPK_x$$

For TTL-restricted forwarding, the message includes the following:

$$F_{\mathrm{im}} = \mathrm{PAYLOAD}|X|h_v|t_c$$
$$F_{\mathrm{m}} = v|\mathrm{TTL}$$
$$\mathrm{PACKETTTL} = F_{\mathrm{im}}|F_{\mathrm{m}}|sigSK_x(F_{\mathrm{im}})|certPK_x$$

Packets forwarded by the local node need to be processed after the routing procedure. In particular, the hash chain base value v has to be replaced by

$h(v)$, i.e. the hash chain has to be shortened by one element, because the routing has decreased the TTL value.

Other fields, especially the signature and the immutable fields are not modified by forwarding nodes, but play a role to check incoming messages.

The primary purpose for the inspection of all incoming packets is checking security policies. One of these policies is the verification of the attached signature as well as the certificate. If the signature or the certificate cannot be verified, the message should be dropped. Moreover, more checks are necessary to ensure security. In summary, an incoming message should pass all the following checks before continuing processing (e.g. routing).

After receiving these messages, the receiver will execute certificate check, signature check, timestamp check, GDA size check, and hash chain check. If any of these checks fails, the message must not be forwarded. Regarding local reception, it is either discarded immediately or marked as invalid by the component. The choice depends on whether applications also want to process invalid packets and should be configurable.

Though these basic measures already can help against attackers, there are still some problems to be addressed:

(1) As soon as the notion of node location plays a role, there is always an attack opportunity against the positioning system that provides nodes with the current position. Thus, secure positioning could help for all position-related packet types in the network. If not all vehicles in a certain area are tricked in parallel (e.g. by a fake GPS satellite), also a heuristic approach to position verification can be helpful.

(2) Simple flooding and geocast mechanisms typically use broadcasts to send packets to all neighbors at once. Therefore, packets are not acknowledged by the receivers, which allow an attacker to selectively destroy packets on the data link layer. For a receiving node, the attack would just look like a collision which happens regularly in wireless ad hoc networks. Because both flooding and geocast generate a lot of redundancy if every intermediate node rebroadcasts a packet, this is not a problem in a large area where multiple paths exist and where an attacker only has a local impact. But, on highways, the radius of the transmission range is often enough to block all packets of one message from further forwarding. As there is no retransmission, these packets will get lost and the flooding/geocast ends there.

6.3.4 Secure Geographic Routing

With geographic routing, we denote multi-hop single-path forwarding method according to the principle of greedy geographic routing. The message's destination is a geographic coordinate rather than a node address. The basic concept of geographic forwarding is to pass messages always to a neighbor node, which is geographically closer to the destination than the current node.

To be able to select such a next hop for a packet, every node needs to know its one-hop neighbors and their current positions. The greedy geographic routing requires a periodic beaconing service to get the described neighbor information. More advanced mechanisms can work without beaconing. However, these mechanisms also have drawbacks, and as we need beaconing in VANETs anyway, we refer to the original form here.

The reason why this type of routing was favored over topological routing protocols for ad hoc networks like AODV or DSR is that has significant advantages in ad hoc networks with very high dynamics like it is the case in inter-vehicle networks.

To secure geographic routing, there are several aspects to be considered. Like in the previously described patterns, packets must be integrity protected and it is helpful to guarantee that packets can only be generated by legitimate participants of the network, such as registered vehicles or RSUs. This can be achieved by signing packets.

The more difficult aspects concern one of the building blocks of geographic routing, the beaconing. Apart from the general security considerations of beaconing, there are more problems to be solved with geographic routing.

6.4 Privacy Enhancing and Secure Positioning

6.4.1 Overview

Privacy preservation is an important design requirement for VANETs, where the source privacy of safety messages is envisioned to emerge as a key security issue because some privacy-sensitive information, such as the driver's name, license plate, vehicle model, position, and driving route, could be intentionally deprivatized so that the personal privacy of the driver is jeopardized. Thus, the safety message's authentication with source privacy preservation is critical for a VANET that is considered for practical implementation and commercialization. In particular, the privacy preservation in VANETs should be conditional, where senders are anonymous to receivers while senders should be traceable to the CA. The CA with the traceability can reveal the source identity of a message once a dispute occurs to the safety message.

In VANETs, position is one of the most important data for vehicles. Each vehicle needs to know not only its own position but also those of other vehicles in its neighborhood. The Global Positioning System (GPS) is the most widespread outdoor positioning system for mobile devices today. The system is based on a set of satellites that provide a three-dimensional positioning with an accuracy of around 3 m. However, GPS signals are weak, can be spoofed, and are prone to be jammed[17]. Moreover, vehicles can intentionally lie about their positions. Hence the need for a secure positioning system that will also support the accountability and authorization properties, frequently related to a vehicle's position.

6.4.2 Privacy Protection Enhancing Scheme

Some approaches have been proposed that claim to effectively provide privacy protection in Vehicle Communications (VCs). However, privacy requirements are often only implicitly stated. The explicit set of privacy requirements identified in section 1 allows us to assess the actual level of privacy protection achieved by an approach. VCs privacy approaches can be coarsely divided into five general categories; they are basic pseudonym approaches, extended pseudonym approaches, symmetric key approaches, group signature approaches, and IBC approaches. In reference [18], representative approaches from these categories are selected and how they fulfill the requirements are discussed.

6.4.2.1 Basic pseudonym approaches

In the context of VCs, pseudonyms commonly refer to pseudonymous public key certificates. These certificates are generated in a predefined way. They do not contain any identifiable information and cannot be used to link to a particular user or to another pseudonymous certificate. Vehicles are equipped with pseudonyms and their corresponding secret keys. When sending a message, a vehicle signs it with its secret key and attaches the signature and the pseudonym certificate to the message so that receivers can verify the signature. Vehicles also have to change pseudonyms often to make it hard for an attacker to link different messages from the same sender.

In reference [19], the SeVeCom project is proposed, which defines baseline security architecture for VC systems. Based on a set of design principles, SeVeCom defines an architecture that comprises different modules, each addressing certain security and privacy aspects. In privacy aspects, the SeVeCom approach employs a hierarchical CA structure, in which CAs manage and issue long-term identities to vehicles. Pseudonyms are issued by pseudonym providers and are only valid for a short period of time. When issuing pseudonyms, a pseudonym provider authenticates a vehicle by its long-term identity and keeps the pseudonyms-to-identity mapping in case of liability investigation. The secret keys of the pseudonyms are stored and managed by a Hardware Security Module (HSM), which is tamper-resistant to restrict the parallel usage of pseudonyms. Provided with a pseudonym, pseudonym resolution authorities can resolve an identity by accessing the pseudonyms-to-identity mappings at a pseudonym provider. Owing to the short lifetime of pseudonyms, the need for credential revocation is minimized. It is basic that only a vehicle's long-term identity is revoked to prevent it from acquiring new pseudonyms from a pseudonym provider. Consequently, CAs only need to distribute CRLs to pseudonym providers, which are part of the infrastructure network.

6.4.2.2 Extended pseudonym approaches

Approaches in this category aim to either improve or enhance specific aspects of the basic pseudonym approaches.

The PKI+ approach[20] is based on bilinear mappings on elliptic curves. It retains the concept of the well-known PKI approach, but provides the additional benefit. In the approach users are autonomous in deriving public keys, certificates and pseudonyms which minimizes the communication to the certificate authority. A user obtains a master key and certificate from a CA after it proves its identity and knowledge of a user secret x to the CA. The user can then self-generate pseudonyms by computing a public key from the master certificate, the secret x, and a random value. A certificate is computed as a signature of knowledge proof s over the public key and the master public key. The certificate also includes the version number *Ver* of the CA public key for revocation purposes. The user signs a message m by computing the signature of knowledge proof m_s on m. A receiver of m can verify the message with the public key in the pseudonym. When revoking a user, the CA publishes a new version information Ver', which has to be used by all users to update their keys. Ver' is chosen so that it is incompatible with the master key and master certificate of the revoked user. The advantage of the PKI+ approach is that vehicles do not need to contact a CA or pseudonym provider to obtain new pseudonyms. The disadvantages of the approach are that Sybil attacks based on unlikable pseudonyms are hard to detect and that the CA has no means to control the amount of self-generated pseudonyms.

The blind signature approach[21] applies blind signatures and secret sharing in the pseudonym issuance protocol to enforce distributed pseudonym resolution. In the approach, a user blinds the public key to be signed and presents shares of it to a number of CAs in the pseudonym issuance process. Each CA is possessed of a partial secret of the secret key shared by all CAs in a secret sharing scheme. Each CA signs the presented blinded key part with its partial secret key, returns it to the user, and stores a corresponding partial resolution tag in its database. The user can unblind and combine the received results, yielding a certificate which can be verified with a public key commonly to all CAs. The certificate is only valid if k of n CAs participated in the issuance process. Otherwise the threshold of the secret sharing scheme is not reached, thus resulting in an incomplete signature. To resolve a pseudonym, more than t CAs have to cooperate in a second secret sharing scheme to compute a joint resolution tag for the presented pseudonym and compare it to all tags in the database. The advantage of the scheme is that it effectively prevents misuse of resolution authority. The disadvantage of the scheme is that it incurs considerable overhead by requiring a number of authorities to take part in the certification of a single pseudonym. In addition, pseudonym resolution requires comparisons with all tags stored in the revocation database, and consequently, does not scale well with the number of users.

6.4.2.3 Symmetric key approaches

Symmetric cryptography schemes are more efficient for time-critical applications for the reason that symmetric cryptography schemes require less computational effort than asymmetric operations. However, symmetric encryption has to somehow emulate asymmetric properties in order to achieve authentication.

The TESLA approach[22] is based on the TESLA lightweight broadcast authentication mechanism[23]. TESLA uses time as the creator of asymmetric knowledge to create asymmetric properties similar to public key cryptography, assuming that network nodes are loosely synchronized. Time synchronization requirements for VANET nodes are to be feasibly given by current technology. In the scheme, a user computes a key chain and releases keys subsequently in fixed time intervals. Each message is authenticated with a key that has not yet been released according to the key schedule, and receivers have to buffer messages until the corresponding key is released and the message can be verified. The authenticity of a message can be verified with any key higher up in the chain. The advantage is that TESLA keys are much shorter in length than public keys and are thus more efficient. To enhance trust, each vehicle also has a set of pseudonyms signed by a CA. Pseudonyms are only used to sign anchors of the key chains. When two vehicles enter each other's reception range, they first exchange certificates to obtain each other's TESLA anchors. Subsequently, they only use symmetric TESLA keys to authenticate messages. Keys belonging to the same key chain as the presented anchor can be traced back to it and thus verified. The proposed scheme significantly reduces the security overhead comparing to the current DSRC draft standard on security (IEEE P1609.2). It provides efficient authentication while reducing certificate exchanges to a minimum. However, for time-critical safety applications the delay in authentication may create problems. Otherwise, in the scheme the keys expire too quickly and actual receivers might not receive disclosed keys. Therefore, TESLA keys are not suitable for multi-hop forwarding.

6.4.2.4 Group signature approaches

A Group signature scheme is a method for allowing a member of a group to anonymously sign a message on behalf of the group. It provides conditional anonymity to members of a group. Each group member can create signatures which can be verified with a common group public key. Essential to a group signature scheme is a group manager, who is in charge of adding group members and has the ability to reveal the original signer in the event of disputes. Only the group manager is able to determine the identity of a signer.

The hybrid approach[24] uses group signatures to reduce the overhead of key and pseudonym management. Vehicles are members of a group and equipped with a secret group signing key and the group public key. Each vehicle generates random public/secret key pairs to be used for pseudonymous communications. The public keys are signed with the group secret key, yielding a pseudonym certificate that can be verified with the group public key. When communicating, vehicles sign the outgoing messages with the secret key of the pseudonym and attach the pseudonym to the message. Upon receipt of such a message, a receiver can verify that the pseudonym was created by a legitimate group member with the group public key. When necessary the group manager is able to open group signatures and retrieve the signer's identity. The scheme enables vehicle on-board units to generate their own pseudonyms, without affecting the system security. One advantage of the scheme is that it obviates the need to acquire new pseudonyms periodically. However, in the scheme revocation of group membership is a scalability issue nevertheless.

The GSIS approach[25] is based on short group signatures and identity-based signature techniques. In the approach, a CA acts as the group manager and has the ability to reveal the original signer. The CA computes a group public key and group secret keys for each vehicle in the group from their unique identifiers. With the identifier and a part of the secret key, a CA can determine the identity of a group member. Therefore accountability can be achieved while at the same time impersonation attacks are prevented. Similar to the hybrid approach, a vehicle signs messages with its own secret key and receivers can verify them with the group public key. Revocation is achieved by distributing revocation lists. One difference to other schemes is that revocation lists are only allowed to grow to a threshold t to avoid increasing verification times. When t vehicles have been revoked, the group key and individual secret keys are updated. The disadvantage of this scheme is that the CRL may grow quickly, which may not only have a large CRL size, but also take a long time to look through the whole CRL to see if a certificate is still valid or not.

6.4.2.5 IBC approaches

Identity-based cryptography (IBC) is a type of public-key cryptography in which a publicly known string representing an individual or organization is used as a public key. The public string is the individual's (or organization's) identity that could include an email address, domain name, or a physical IP address. Presented with a signature, a verifier can check its validity merely by knowing the sender's identity.

The efficient conditional privacy preservation (ECPP) approach[26] utilizes both IBC and group signatures. In the scheme, a trusted authority TA sets up an IBC scheme and publishes its system parameters. Each vehicle has a unique identity, which is used to authenticate with the TA to obtain a pseudonym. When a vehicle submits its identity, TA generates a pseudo identifier by encrypting the vehicle identifier with its public key and extracting a corresponding private key from it. The vehicle can use the resulting key pair as a pseudonym in anonymously authentication processes with RSUs under control of TA. When a vehicle enters the vicinity of a RSU, it requests a short-time anonymous key certificate to take part in a local group signature scheme. For this reason, the group identifier is also used as the group public key. The RSU checks that the presented pseudonym is not listed on a CRL, and issues a group membership certificate, which is valid only for a short period of time. The RSU also retains a mapping between group membership certificate and pseudonym. Where after, the vehicle can perform group signatures on messages by proving possession of a membership certificate, and therefore communicate anonymously with other vehicles. By opening the group signature of a message and retrieving the identifier of the RSU that issued the group membership certificate, the TA is able to realize identity resolution. The RSU can then be contacted and returns the pseudonym corresponding to the presented membership certificate. In the last step, the TA decrypts the pseudonym with the symmetric key and yields the real vehicle identifier. The advantage of the scheme is that it can provide fast anonymous authentication and privacy tracking while minimizing the required storage for short-time anonymous keys.

6.4.3 Secure Positioning Scheme

Secure positioning will play a significant role in many vehicular applications, making it critical to determine that a message did indeed originate at a given location. For example, secure positioning would prevent an attacker sitting on the side of the road from claiming to be a vehicle traveling on the highway. It would also prevent an adversary from using another communication medium to replay a message heard in one location as though it had originated in a different location.

GPS-based positioning has a lot of disadvantages. It cannot be used for indoor positioning or for positioning in dense urban regions: in those cases, because of the interferences and obstacles, satellite signals cannot reach the GPS devices. Furthermore, civilian GPS was never designed for secure positioning. Civilian GPS devices can be spoofed by GPS satellite simulators, which produce fake satellite radio signals that are stronger than the real signals coming from satellites. Until now, there is little work done on secure positioning without GPS. One possible approach would be to extend the protocols that have been proposed for secure localization in sensor networks to this new setting. Unfortunately, these protocols such as reference [27 – 29] focus on allowing a sensor to securely determine its own position (rather than the positions of its neighbors) or rely on the presence of multiple base stations. In reference [30], Parno et al. propose to leverage the properties of the vehicular environment to provide a new method of secure relative localization. In their scheme, a vehicle's relative location is defined by its entanglement with other vehicles. Each vehicle will regularly broad cast its identity (a public key) along with its signature of a current timestamp. When a vehicle receives such a broadcast, it signs the other vehicle's ID and rebroadcasts it. In other words, when vehicle A receives public key K_B from vehicle B, it adds a signature $\{K_B\}_{SK_A}$with its private key SK_A to its regular broadcast. When vehicles pass each other traveling in opposite directions, this will allow both streams of traffic to perform relative localization (see Fig. 6.9). If vehicle B hears vehicle C rebroadcast A's identity before it rebroadcasts B's identity, then B can conclude that A is ahead of him/her. Vehicle B can aggregate multiple indicators (i.e., from vehicles D and E) to provide further assurance of A's position. Furthermore, vehicle B can evaluate the entanglement data for those vehicles as well to determine how much weight to give their reports.

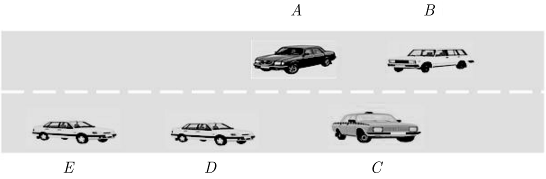


Fig. 6.9 Secure relative localization (vehicle B can use broadcasts from vehicles C, D and E to determine A's location).

This scheme helps to perform relative localization. But this approach incurs overhead and does not provide absolute positions. The final solution will probably be a hybrid system that will use a combination of GPS, radars, wheel rotation sensors, digital maps, and roadside beacons, depending on the availability and reliability of each of these techniques.

6.5 Conclusion

The area of vehicular ad hoc networks has been developed significantly during the past decade. Several new applications are enabled by this new kind of communication network. However, as those applications have impact in road traffic safety, strong security requirements must be achieved. New mechanisms have to be developed to deal with the inherent features of these networks (extreme node's speed, decentralized infrastructure, etc.). In this chapter, we present an overview of the current security issues over VANETs. We have identified the security and privacy requirements and security threats in VANETs. We have also described security architecture for VANETs based on the PKI and security hardware, and introduced a secure VANET communication scheme based on TPMs. Furthermore, we have presented the scheme of key management and authentication under the security architecture based on the PKI and security hardware. Security routing solutions for V2V, V2I, and group communications have also been analyzed representatively. Finally, we have discussed privacy enhancing and secure positioning schemes.

Secure communications for vehicular ad hoc networks have become an important research issue these years. Several future research lines can be pointed out in VANET security area. Although several mechanisms have been proposed, some issues still have to be addressed (e.g. privacy problems due to radio frequency fingerprinting). Simulation results are often offered to evaluate current proposals. However, a common scenario to evaluate alternatives does not exist. Finally, hardware implementation of efficient cryptographic primitives is required in vehicles. In this way, achieving computation availability would be eased[31].

References

[1] IEEE Draft Std P802.11p/D2.0 (2006) Wireless Access in Vehicular Environments (WAVE).

[2] IEEE Std 1609.2-2006 (2006) IEEE Trial-Use Standard for Wireless Access in Vehicular Environments-Security Services for Applications and Management Messages. IEEE, New York.

[3] Luo J, Hubaux J P (2004) A Survey of Inter-Vehicle Communication, EPFL Technical Report IC/2004/24. http://infoscience.epfl.ch/record/52616/files/IC_TECH_REPORT_200424.pdf. Accessed 10 October, 2011.

[4] Sichitiu M L, Kihl M (2008) Inter-vehicle communication systems: a survey. IEEE Communication Surveys and Tutorials, 10(2): 88 – 105.

[5] Hartenstein H, Laberteaux K P (2008) A tutorial survey on vehicular ad hoc networks. IEEE Communications Magazine, 46(6): 164 – 171.

[6] Camp T, Boleng J, Davies V (2002) A Survey of Mobility Models for Ad Hoc Network Research. Wireless Commun. & Mobile Comp., special issue on Mobile Ad Hoc Networking: Research, Trends and Applications, 2(5): 483 – 502.

[7] Hubaux J P, Capkun S. Jun L (2004) The security and privacy of smart vehicles. IEEE Security and Privacy magazine, 2 (3), 49 – 55.

[8] Raya M, Hubaux J P (2007) Securing Vehicular Ad Hoc Networks. Journal of Computer Security, Special Issue on Security, Ad Hoc and Sensor Networks, 15(1): 39 – 68.

[9] Raya M, Papadimitrators P, Hubaux J P (2006) Securing vehicular communications. Wireless Communications, 13(5): 8 – 15.

[10] Guette G, Bryce C (2008) Using TPMs to secure ad hoc networks. In: Proceedings of the 2nd IFIP WG 11.2 international conference on information security theory and practices: smart.devices, convergence and next generation networks, pp. 106 – 116.

[11] Trusted Computing Group (2007) TPM main specification. Main Specification Version 1.2 rev. 103, Trusted Computing Group.

[12] Raya M, Hubaux J P (2007) Securing vehicular ad hoc networks. Journal of Computer Security, 15: 39 – 68.

[13] Wohlmacher P (2000) Digital certificates: a survey of revocation methods. In: Proceedings of ACM Wksp. Multimedia, pp. 111 – 114.

[14] Lin X D, Lu R X, Zhang C X, Zhu H J, Ho P H (2008) Security in vehicular ad hoc networks. IEEE Communications Magazine, 46(4): 88 – 95.

[15] Zhang C, Lin X D, Lu R X, Ho P H (2008) An efficient RSU-aided message authentication scheme in vehicular communication networks. In: Proceedings of the IEEE Conference on Communications, pp. 1451 – 1457.

[16] Antonio Kung (2008) Security Architecture and Mechanisms for V2V / V2I, D2.1 v3.0. http://www.sevecom.org/Pages/ProjectDocuments.html. Accessed 10 October, 2011.

[17] Capkun S, Hubaux J P (2006) Secure positioning in wireless networks. IEEE Journal on Selected Areas in Communications, 24(2): 221 – 232.

[18] Schaub F, Ma X D, Kargl F (2009) Privacy requirements in vehicular communication systems. In: Proceedings of the International Conference on Computational and Engineering, pp. 139 – 145.

[19] Papadimitratos P, Buttyan L, Holczer T (2008) Schoch E, Freudiger J, Raya M, Ma Z, Kargl F, Kung A, Hubaux J P, Secure vehicular communications: Design and architecture. IEEE Communications Magazine, 46(11): 100 – 109.

[20] Armknecht F, Festag A, Westhoff D, Zeng K (2007) Cross-layer privacy enhancement and non-repudiation in vehicular communication. In: Proceeding of the 4th Workshop on Mobile Ad Hoc Networks, pp. 1 – 12.

[21] Fischer L, Aiijaz A, Eckert C, Vogt D (2006) Secure revocable anonymous authenticated inter-vehicle communication. In: Proceeding of the 4th Workshop on Embedded Security in Cars (ESCAR).

[22] Hu Y C, Laberteaux K P (2006) Strong VANET security on a budget. In: Proceeding of the 4th Workshop on Embedded Security in Cars (ESCAR).

[23] Perrig A, Canetti R, Tygar J D, Song D (2002) The TESLA broadcast authentication protocol. RSA Cryptobytes, 5(2): 2 – 13.

[24] Calandriello G, Papadimitratos P, Hubaux J P, Lioy A (2007) Efficient and robust pseudonymous authentication in VANET. In: Proceeding of the 4th ACM Intl workshop on Vehicular ad hoc networks, pp. 19 – 28.

[25] Lin X, Sun X, Ho P H, Shen X (2007) Gsis: A secure and privacy-preserving protocol for vehicular communications. IEEE Trans. Vehicular Technology, 56(6): 3442 – 3456.

[26] Lu R, Lin X, Zhu H, Ho P H, Shen X (2008) Ecpp: Efficient conditional privacy preservation protocol for secure vehicular communications. In: Proceeding of the 27th Conference on Computer Communications, pp. 1229 – 1237.

[27] Sastry N, Shankar U, Wagner D (2003) Secure verification of location claims. In: Proceedings of WiSe, pp. 1 – 10.

[28] Ji X, Zha H Y (2004) Sensor positioning in wireless ad hoc sensor networks using multidimensional scaling. In: Proceeding of the 23rd conference of the IEEE Computer and Communications Societies, pp. 2652 – 2661.

[29] Bras L, Oliveira M, Carvalho N B, Pinho P (2010) Low power location protocol based on ZigBee wireless sensor networks. In: Proceeding of the international conference on Indoor Positioning and Indoor Navigation, pp. 15 – 17.

[30] Parno B, Perrig A (2005) Challenges in securing vehicular networks. In: Proceedings of the Workshop on Hot Topics in Networks.

[31] de Fuentes J M, Gonzalez-Tablas A I, Ribagorda A (2010) Overview of security issues in vehicular ad hoc networks. Handbook of Research on Mobility and Computing. IGI Global, Hershey.