An Ecosystem for Securing Vehicle-to-Everything Communication

Leonardo José Abreu Gonçalves
Departamento de Engenharia Informática
Instituto Superior Técnico

Advisors: Prof. Nuno Miguel Carvalho dos Santos and Eng. Carlos Manuel Pereira Cardoso

I. Introduction

According to the European commission statistics [1] in the year 2016 over 25000 people died in road accidents in Europe, furthermore it is estimated that for every death on Europe's roads there is 4 permanently disabling injuries such as damage to the brain or spinal cord. Intelligent transportation that is capable of assisting the driver and connect vehicles can reduce accidents significantly. *Intelligent Transportation Systems* (ITS) [2] are applications that allow vehicles to connect and coordinate their actions. This cooperation of vehicles is expected to increase road safety and traffic efficiency by assisting the drivers to make better decisions and advising new routes based on the traffic conditions.

One fundamental aspect of ITS is the V2X communication. Vehicles equipped with this technology are able to share data in real time with other vehicles, road infrastructure (roadside units) and pedestrians utilizing short-ranged wireless signals. Such data may be related to sender's presence on the road, or related to road events so that other vehicles affected by that specific occurrence (e.g. road obstacle) are notified. While vehicles transmit these types of data, roadside units transmit regional data such as speed limits, timing of semaphore lights or information about traffic deviation. Vehicles communicating with other vehicles, pedestrians and infrastructure on the road create a decentralized network known as *Vehicular Ad Hoc Network* (VANET) [3] [4]. This type of communication allows the developing of ITS applications that can signal various kinds of events, for example, cover forward collision warnings, emergency vehicle approaching, lane change warning/blind spot coverage, road works warning, and many more. Thus, V2X enhances the vehicle's perception of environment much beyond the driver's visual horizon and vehicle sensing capabilities.

Security becomes fundamental in VANETs, which are threatened by a range of potential attacks, such as distribution of forged messages, tracking of user vehicles and denial of service. The consequences of such threats can be extremely serious, and may range from disruption of the transportation to serious damage to public safety on the road. Our work focuses on a PKI mechanism that aims to address some of previous cyberattacks. The IEEE 1609.2 [5] and ETSI TS 103 097 standards [6] specify protocols for V2X communication security and recommend the usage of digital certificates to sign the messages, thus making the public key infrastructure essential. The basic idea is that all *ITS Stations* (ITS-S) i.e. vehicles and *Roadside Units* (RSU), which are equipped with a V2X communication unit have to be registered with the PKI. Only with valid certificates these stations are able to send authenticated messages that will be trusted by the receiving stations. The certificates provided by the V2X PKI have to be stored in the hardware security module known as *On-Board Unit* (OBU) or *On-Board Equipment* (OBE).

Although this basic approach allows for message authentication, care must be taken in the design of the PKI as so to avoid privacy violations. Certificates used for V2X communications must not contain any information that links them to a particular vehicle or owner, e.g. a license plate number; such information would allow vehicle tracking by simply listening to the communications. However, removing all identifying information from certificates i.e. using pseudonym certificates is not sufficient. If a vehicle uses a single pseudonym during its lifetime, then this certificate can again be used to track the vehicle. To defeat this scheme, an attacker would only need to observe a vehicle using the same certificate at different locations to be able to link that certificate to the victim vehicle. The most common approach to assure privacy at this level is to store a pool of short-lived pseudonym certificates (also known as authorization tickets) in each vehicle's OBU. Vehicles periodically change pseudonym to authenticate V2X messages in order to avoid long-term tracking. This mechanism implies that vehicles need to communicate with the PKI to request new pseudonym certificates whenever their locally stored list is expiring. In addition to pseudonym certificates, stations also need a long-term enrollment certificate tied to their identity to authenticate within the PKI. The result is a vehicular PKI that is architecturally different from a traditional PKI.

This work addresses the problem of designing and implementing a V2X system that allows the authentication of vehicular communications while preserving the privacy of its users. This report will specify the system to produce, a V2X ecosystem comprising a vehicular PKI and user vehicles which are able to enroll in the PKI and use valid certificates to authenticate V2X messages. The goal of this work is to implement such system based on the most recent European standards and according to the following requirements.

- Privacy
  - The drivers must remain anonymous on the road, meaning that unauthorized parties are not able to associate a V2X message to the vehicle/driver who sent it.
  - The messages transmitted during V2X communications must remain unlinkable to the vehicles which previously sent them.
  - The privacy of the vehicle's location should be protected by the usage of pseudonym certificates.

  - Deducing the vehicle's location or tracking vehicles should not be aided by analysing the vehicle's previous V2X communications.
- Confidentiality
  - Information transmitted to or from a given vehicle to the PKI, such as certificate requests and responses, should be protected against unauthorized access.
- Integrity
  - Information transmitted to or from a given vehicle must be protected against unauthorized modifications or tampering during transmission.
- Authenticity
  - When a vehicle receives a V2X message it should be able to trust that such message is relevant and was created by a legitimate ITS-station.
  - When a vehicle receives a message from the PKI, such as a certificate response, it should be able to trust such message was created by the intended CA as a response to the initial request.
  - Configuration information originated from the PKI should arrive to the vehicles in a state that allows them to confirm the origin and integrity of the message.
- Availability
  - Access to the PKI services should not be prevented to legitimate vehicles by malicious activity.

The remainder of this thesis is organized as follows: Section II provides an overview of the state-of-the-art regarding the technologies and mechanisms used to manage the identity of vehicles during V2X communications. Section III refers to the architecture and implementation of the solution to achieve the goals previously described. Then, Chapter IV provides an experimental evaluation of the implemented system, where we display the performance tests done to the solution and describe the security properties achieved. Finally, Chapter V concludes this document by summarizing the work, describing the achieved goals and suggesting future work to improve the solution.

II. State of the Art

In this Section we analyze the existing work related to V2X communication. In order to have a high level understanding of how V2X communications work, we start by specifying two vehicular PKI solutions. The first is named *A Generic Public Key Infrastructure for Securing Car-to-X Communication* and has been proposed by the corresponding stakeholders in Europe [7] [6] [8] (Section II-A). The second is named *Security Credential Management System* (SCMS) and is the American counterpart proposed in [9] (Section II-B). In Section II-C, we provide an overview the existing standards behind the European solution, which will be the basis of our work, and present a lower level and more detailed notion of the V2X communication functioning.

A. European Vehicular PKI Architecture

The European PKI uses long-term certificates named enrollment certificates and short-term certificates known as pseudonym certificates or authorization tickets. Enrollment

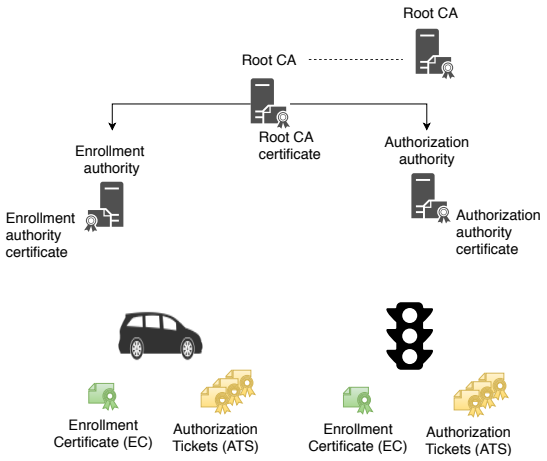


Figure 1: European vehicular PKI architecture.

certificates are tied to the vehicle's identity to authenticate the vehicle within the PKI back-end. Authorization tickets have the identifying information removed and are used in V2X communications for privacy reasons. The European PKI considers an hierarchical structure as we can see in Figure 1. Such architecture is composed of a *Root Certification Authority* (RCA), an *Enrollment Authority* (EA), and an *Authorization Authority* (AA). For a given trust domain the RCA certificate is the root of trust for all certificates in that hierarchy, this means that a vehicle will only trust an incoming message if the certification chain starting on the received authorization ticket to the root CA certificate is valid. The RCA is responsible for issuing certificates for enrollment authorities and authorization authorities. If there are multiple RCAs, trust between them can be established by using cross certification. No other cross certification between CAs is allowed. The EA has the responsibility of validating that a vehicle can be trusted and only if so, issuing an enrollment certificate for that vehicle as a proof of its identity. Finally, the AA exists to allow vehicles to apply for specific services and permissions on the road. These privileges are denoted by means of authorization tickets (pseudonyms), which are issued by the AA for the applying ITS-S.

1) Enrollment Process: Before an ITS-S is able to participate in the V2X communication it must be registered within the PKI. The enrollment request message should be sent from the ITS-S to the Enrollment Authority, to protect the users privacy the request must be encrypted. After the ITS-S enrollment request the target EA must reply with a successful or failed response message, to protect user privacy the response should also be encrypted. The successful ITS-S enrollment response should contain the enrollment certificate and the chain of certificates back to the originating enrollment CA. In the failed ITS-S enrollment, the response contains the error code i.e. the reason for the unsuccessful enrollment response.

2) Authorization Ticket Provisioning: Pseudonym certificates are short-lived certificates which express the permissions that a specific enrolled vehicle has on the road while hiding its identity. Consequently they are refereed as *Authorization Tickets* (ATs) by ETSI. To avoid long-term tracking, a vehicle rotates authentication tickets from its local pool

to authenticate V2X messages. However, it needs to request new ATs from the authorization authority once there are few valid ATs stored locally. The update can be done over-the-air or at an authorized dealership (during vehicle maintenance), i.e. roadside-units and workshops can act as a proxy for certificate requests. Because our solution will also use ATs, important decisions must be done regarding the frequency that ATs are provisioned to the enrolled vehicles and how such vehicles rotate certificates from their pool.

3) Authorization Ticket Request Process: In this section we study the sequence of messages used by vehicles to request valid ATs in Europe. The solution that we propose will assume such protocol for the simulated vehicles' requests for ATs. At a high level, a vehicle uses its enrollment certificate to prove its enrollment to the AA, only then the AA can issue the ATs. The ETSI TS 102 941 [10] standard specifies in detail the message format for the AT request and response. In regards to the process [7], the vehicle sends a request to a predefined AA. The request includes the vehicle's enrollment certificate, the certificate of the corresponding Enrollment Authority, and the to be certified public key. To protect user's privacy the enrollment certificate may be encrypted with the public key of the corresponding EA. In this case the AA is not able to create a link between the authorization tickets and the enrollment certificate of a specific vehicle. Consequently, when an AA receives such requests it cannot verify the enrollment of the requesting vehicle. In order to do so, the AA sends a request with the (encrypted) vehicle's enrollment certificate to the correct EA (identified by the EA certificate present in the original request). Only if the vehicle's enrollment certificate is valid the AA will get a positive response from the EA validating the enrollment of the vehicle. Upon receiving such response, the AA has the responsibility of issuing the ATs for the vehicle.

4) Message Signing and Verification: In this section we analyze how the secured V2X messages are signed and verified by the vehicles. Such information will allow us to correctly test the communications between vehicles in the proposed simulator. In regards to sending messages, the sender of V2X messages signs all outgoing messages with the private key of a valid AT. Afterwards, the message with the appended signature and pseudonym certificate is broadcast. When a station receives a message, the senders authenticity and message integrity is verified by decrypting the signature with the public key from the appended AT. The sender's authenticity is only accepted if verification of the received AT up to a root CA is possible. Vehicles are preloaded with the known and trusted authorization authority certificates at manufacture. However if the Authorization Authority certificate that corresponds to the received AT is not locally stored, the message receiver cannot validate the sender's authenticity. In this case, the message receiver must create a new message requesting the missing Authorization Authority certificate and send it to the original message sender. Then, the receiver of this request must respond with the Authorization Authority certificate.

5) Certificate Revocation: In the European solution detecting and preventing misbehavior by means of a misbehavior entity is not yet supported. Revocation is done in respect to the long-term enrollment certificate of ITS-S and CA certificates. The ITS stations are eventually removed from the system by rejecting new requests for ATs. In this concept the EA links the revocation information of the vehicle to its long-term enrollment certificate.

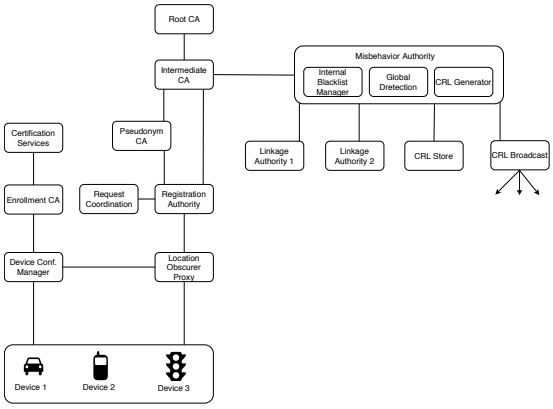


Figure 2: SCMS overall system architecture.

B. American Vehicular PKI Architecture

SCMS considers an hierarchical structure as we can see in Figure 2. Comparing with the European vehicular PKI architecture, there are some components with a similar function and others that introduce new functionality. In regards to the similar components, SCMS assumes a *Root CA*, an *Enrollment CA*, and a *Pseudonym CA* (PCA) which corresponds to the authorization authority in the European architecture. As to the remaining components, their functionality is as follows:

- **Registration Authority (RA):** Validates, processes, and forwards requests for pseudonym certificates to a pseudonym CA.
- **CRL Store (CRLS):** Stores and distributes CRLs. CRLs are signed by the CRL Generator.
- **CRL Broadcast (CRLB):** Broadcasts the current CRL, may be done through road side units or satellite radio system, etc.
- **Misbehavior Authority (MA):** Processes misbehavior reports to identify potential misbehavior by devices, and if necessary revokes and adds devices to the CRL.

An early version of SCMS has already been implemented, operated and tested in the safety pilot project [11]. Safety pilot is a scaled-down evaluation of V2X that uses real vehicles and roadside infrastructure in order to understand the safety benefits of connecting vehicles. SCMS documentation can be found in [12]

1) Pseudonym Certificate Request Process: The request for pseudonym certificates in itself is different from the European solution. For this the butterfly key expansion algorithm [9] is used. Butterfly keys allow a device to request an arbitrary number of certificates, each encrypted with a different encryption key and each containing a different signing key. The request contains only one seed for the verification public key, one seed for the encryption public key, and two expansion functions. Without butterfly keys, vehicles would have to send a signing key and a unique encryption key for each requested certificate. Butterfly keys reduce upload size, allowing requests to be made even in suboptimal connectivity conditions, and also reduce the computation to be done by

the vehicle to calculate the keys. More information about the request process can be found in [9].

2) Misbehavior Reporting: In contrast to the European PKI, the American PKI supports misbehavior reporting by user vehicles. This feature aims to improve the security against SCMS outsiders by reporting their malicious messages. Devices will send misbehavior reports to the MA

C. Secured Message and Certificate Formats Standard

One of the main concerns of V2X communication is the ITS interoperability. Standardization of the communication protocol becomes fundamental with so many vehicles from different manufacturers using the road and sharing information. To achieve this goal there are dedicated work groups within standardization organizations that address security and privacy concerns. While ETSI Automotive Intelligent Transport Systems represents the main standardization stakeholders in Europe [13], IEEE 1609.2 represents the main standardization stakeholders in the U.S [5]. Such standardization efforts are the basis of the security and privacy of the European and American vehicular PKI solutions respectively. A survey about recent standardization activities in Europe (ETSI) has been done by IEEE in [8].

In regards to the secured message and certificates formats. IEEE 1609.2 [5] standard defines the formats for secured V2X messages and public key certificates to be used in SCMS. ETSI TS 103 097 standard [6] presents security profiles for the messages and certificates also based on the IEEE 1609.2 standard. For example, ETSI TS 103 097 uses the definition of possible fields that a certificate may contain (the format) present in IEEE and, based on these options, builds the specific profiles (the necessary fields) for the root CA certificates, authorization tickets, enrollment certificates, and other certificates to be used in the European solution. The same process applies with the profiles for the secured V2X messages.

The V2X ecosystem that we propose will be primarily based on the European PKI. Consequently, it is relevant that we understand the contents of the secured V2X messages and the certificates used by it. In order to do so, we provide an overview of the ETSI TS 103 097 standard.

1) Security Profile for Cooperative Awareness Messages: Cooperative awareness messages (CAM) are messages that are exchanged between ITS-S. As the name implies, these messages are used to achieve cooperative awareness on the road. This means that road users such as vehicles (cars, trucks, trains, etc.), road-side units (traffic lights, gates, barriers, etc.) and people are aware of each other's positions, speed and other dynamic variables. To achieve this goal, it is essential that this type of messages is periodically broadcast by each road user to all its neighbors. CAMs are used to support traffic management and safety services. In the normal cases CAMs are sent multiple times per second with the component signer identifier containing the reference of the signing authorization ticket (8 byte certificate digest). However, in oder to distribute the currently used AT, every second a CAM is sent with the signer identifier containing the full certificate. If a vehicle receives a CAM signed by a previously unknown AT, it should include the currently used AT immediately in its next CAM, instead of including just the digest. In this case, the timer for the next inclusion of the full certificate should be restarted to one second.

Besides distributing the currently used AT a vehicle also needs to request the unknown certificate present on the revived CAM for message verification purposes. Specifically, if a vehicle receives a CAM with the signer identifier containing an unknown certificate digest, then it will include that digest in its next CAM to broadcast the request for the full certificate. It is also possible for a vehicle to receive a CAM containing the full signing authorization ticket but this certificate is signed by an unknown authorization authority certificate. In this case the vehicle should include in its next CAM the digest of the unknown authorization authority certificate which is present on the AT itself.

If a vehicle receives a CAM containing a request for an unknown certificate, then it searches the list of certificate digests existing in that message. If the digest of the currently used authorization ticket is found, then the vehicle includes the full certificate in its next CAM instead of the digest to distribute the certificate.

2) Security Profile for Decentralized Environmental Notification Messages: Decentralized environmental notification messages (DENM) are messages designed to provide asynchronous warning notifications to vehicles. DENMs are event triggered and are broadcast to notify the users of a hazardous event. For example, an emergency vehicle approaching or an accident on the road. These messages have to be broadcast to all users affected by the event, sometimes multiple hops are needed to achieve this.

In order to reduce the verification delay at the receiver side CAMs are always sent with the full signing authorization ticket in the signer identifier. Because it is important for vehicles to know were the event occurred, these messages will always include in the header the generation location.

3) Certificate Formats: The certificate formats include profiles for the root CA, enrollment authority, authorization authority and end-entities certificates (authorization tickets and enrollment certificates). Generally a certificate is composed of the issuer identifier, certificate identifier, application permissions, permitted geographic location, start of the validity time, expiration time, public key and the signature. In order to construct the certification chain, each non-root certificate carries the issuer identifier which is a reference (8 byte digest) that points to the certificate that belongs to the issuer CA.

D. Discussion

In this section we provide a brief overview of the vehicular PKI solutions presented and analyze their advantages and disadvantages. We have seen that in Europe exists *A Generic Public Key Infrastructure for Securing Car-to-X Communication* [7] and in America exists the *Security Credential Management System* [9]. In regards to the European PKI, the first disadvantage comes in the vehicle's request for authorization tickets. This solution assumes that every vehicle has to calculate a list of keypairs containing one signing and verification key for each of the requested authorization tickets. Since vehicles typically request a bundle of certificates to be used in a timespan of years, the generation of keys results in an increased computing overhead within the OBU whenever a vehicle needs to request new Authorization

tickets. In addition, this process also increases the size of the request, which has to contain all of the verification keys. Although this system has these disadvantages it provides a simple architecture that is compatible with mPKI and is based on the most accessible standards. These advantages provide us with a good starting point for the implementation of the proposed V2X system. In regards to the American solution, the main disadvantages are that the underlying standard is payed to obtain and most importantly, the complexity of its architecture and protocol makes it much less compatible with mPKI.

Having in mind the advantages and disadvantages of the existing vehicular PKI solutions, we decided to base our PKI solution on the European vehicular PKI. However, as we have discussed before, the European vehicular PKI is a generic concept. For this reason it cannot be immediately implemented in mPKI. For example, one of the main aspects that is not specified in this solution is the interface between vehicles and CAs. Next, we present the changes to the European vehicular PKI that we assumed in order to define our V2X environment.

E. Summary

In this section we have studied the state-of-the-art PKIs used in the identity management of vehicles during V2X communications in Europe and America. We started by overviewing the architecture of such solutions, then approached some of the operational aspects needed to support a V2X environment. We Finalize this section by providing detailed overview of the standards behind the European PKI. In the next section, we present our V2X environment which is based on the European PKI and standards. We start by overviewing its architecture, then we detail its implementation.

III. Proposed Solution

A. Overview of the System Architecture

Our solution is primary based on the European vehicular PKI and ETSI's standards. As such, the architecture of the PKI is similar to that of A Generic Public Key Infrastructure for Securing Car to-X Communication [7] that uses the certificates and message formats standardized by ETSI [6] to secure V2X communications, as specified in Section II-A. In regards to the software needed to support such PKI, Figure 3 provides an overview of its main components.

1) System Components:

V2X Library: The V2X Library is designed as a software library package. Its main goal is to create all the data structures specified in the latest version of the ETSI TS 103 097 standard. Such structures combine to form the certificates and messages that are used by the vehicles and PKI. In addition to this, the library also allows is users (PKI Manager and Vehicle Manager) to perform cryptographic operations related with the generation of certificates, signature of messages as well as their verification. The usage of this library allows the PKI Manager and Vehicle Manager to perform such operations in conformance with the approved standards and algorithms.

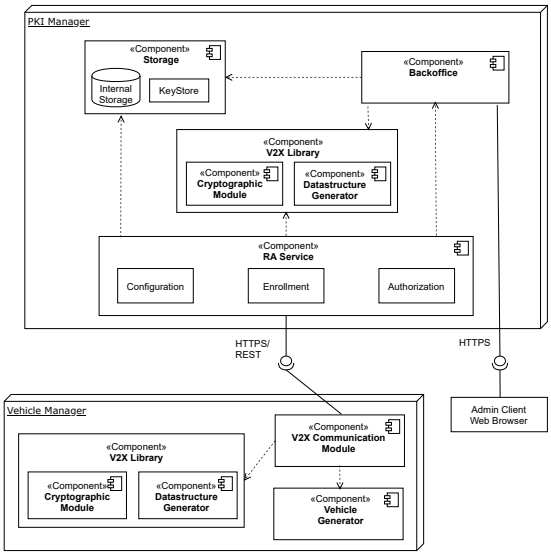


Figure 3: Overall system architecture.

PKI Manager: The PKI Manager is a web application that functions as a backoffice to the PKI. Such application aims to provide administrative services related to the creation and storage of the vehicular PKI. For example, it enables an admin user to log in and create new CAs, their cryptographic keys and certificates. The PKI Manager allows the creation and configuration of a new PKI or even change the structure of an already existent PKI. This application is connected to a database and keystore to provide persistence regarding the PKI information.

RA Service: The RA Service is designed as an API (Application Programming Interface) of the PKI Manager, its main goal is to act as proxy between the vehicles of the Vehicle Manager and the CAs existing on the PKI Manager. The RA has two responsibilities: verifying the vehicle's identity and supporting their requests for enrollment certificates and authorization tickets. The former task requires the RA to perform an initial vehicle configuration, much like authentication, the goal is to "remember" and securely identify each connection to an already configured vehicle. The later task involves sending such requests to the correct CAs for certificate issuing and responding to the vehicles with the requested certificates. The RA Service uses encryption and digital signatures to ensure that the exchange of certificates is secure and the privacy of the end-entities is protected.

Vehicle Manager: As the name suggests, the Vehicle Manager aims to manage the vehicles that will participate in V2X communication. This application runs on its own process and can be configured to create a given number of vehicles each as a client of the RA Service. Once created, the vehicles can contact the RA Service in order to request the end-entity certificates. Only with this initial configuration done, the vehicles are able to start communication with each other in respect to the Vehicle Manager configuration.

2) Communication: As we can see from Figure 3 we assume a client-server architecture, where the server is the PKI Manager and the Vehicle Manager acts as the client. On the server-side, the client requests will enter through the RA Service API, which then communicates with the back end and

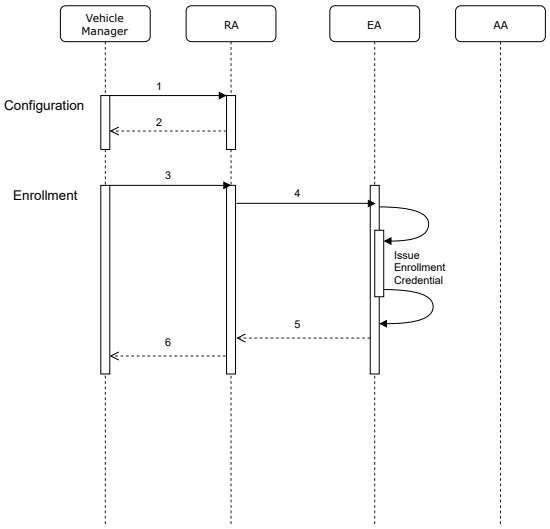


Figure 4: Vehicle configuration and enrollment process flow.

V2X Library to access the PKI services. As the RA Service is part of the PKI Manager project, the communication with the back end is achieved through simple service calls. Regarding the V2X Library, because this component exists on a different project we used it as a dependency of the server, which gives the server access to all the public classes and methods that the V2X Library has to offer.

3) Protocol: The interaction between the Vehicle Manager generated vehicles and the PKI is divided into several phases. Essentially, vehicles use the information installed at manufacture time to request the enrollment credential, later using this credential in order to request authorization tickets to start V2X communications. As seen in Figure 3, the only way that a vehicle can reach the CAs on the PKI Manager is through the RA Service. Specifically through the RA's services of **configuration**, **enrollment** or **authorization**, each of these services represents a phase in the vehicle to PKI interaction. The SSL protocol is used to secure the integrity, privacy and authenticity of the sensitive data that is transmitted during by the Vehicle Manager and RA Service communications.

Vehicle Configuration: As we can see from message 1 to 2 represented on Figure 4, the first phase is the configuration. This phase aims to support the vehicle and RA configuration at vehicle manufacture time. Specifically, when the Vehicle Manager is generating new vehicles it uses this service in order to register them and to provide such vehicles with the information regarding the PKI. The secure communication channel between the Vehicle Manager and RA Service allows the both RA to keep track of the trustworthy vehicles, and the vehicles to trust the PKI information which is responded by the RA. Such configuration will be the base of the trust that the RA has on the client vehicles for future interactions.

Vehicle Enrollment: The second phase is the enrollment. Vehicles which have completed the initial configuration can use this service to request the enrollment credential. To secure this type of communication we provide security at two levels: at the channel level through the SSL protocol and at the application level through the usage of digital certificates, signatures and encryption. Messages 3 to 6 represented on Figure 4 depict the communication steps of this service.

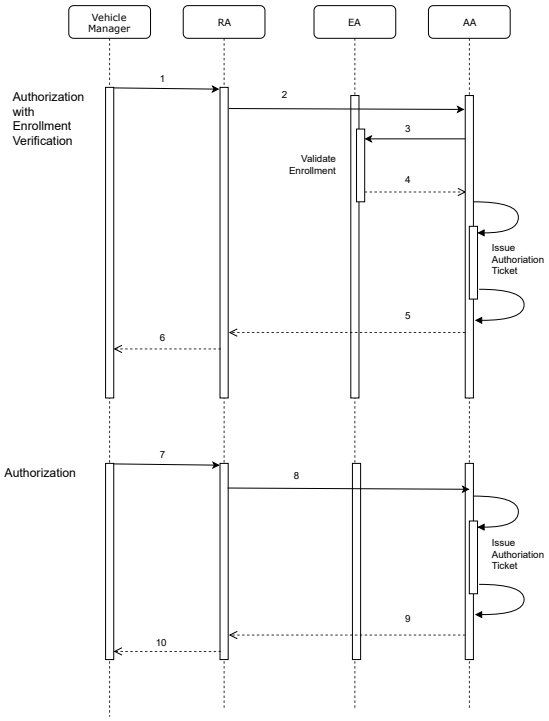


Figure 5: Vehicle authorization process flow.

The first step is made by the vehicle, in order to request the enrollment credential it first builds an enrollment request structure which is signed and encrypted for the attributed EA. This request is the base of the application provided security. Because the request itself is encrypted to the EA, the RA upon receiving it is not able to associate the vehicle's identifier with its future enrollment credential. Even so, the RA is able to perform the first vehicle identity verification. This is done by validating if the vehicle is already configured and is stored on its database. If this is the case, then the RA will send message 4 to the target EA, allowing the EA to perform the second vehicle authentication verification. Unlike the first validation done by the RA which depends mostly on the security at the channel level, the second validation depends essentially on the security provided by the application. Only if the vehicle's signature is valid, the EA will issue the enrollment credential and include it in the response message 5. Such response is signed by the EA, encrypted to the vehicle and then sent to the RA which has the responsibility of returning it to the original vehicle using message 6.

Vehicle Authorization: The last phase is the authorization which can be started by the enrolled vehicles. Each time a vehicle requests one authorization ticket to an AA its enrollment must be verified. However, if such validation is performed by the AA the privacy of the vehicle would be compromised as the AA would have access to both the vehicle enrollment information (identity) and the pseudonym authorization ticket. To protect the vehicle's privacy, the enrollment validation is done by the EA which issued the vehicle's enrollment credential. Since a vehicle needs more than one authorization ticket, the consequence of this added privacy is the need to send more messages decreasing the

performance of the vehicle authorization process as a whole.

As we can see from Figure 5, we have two flows from which a vehicle can request an authorization ticket. This enables us to leverage the RA in order to speed up the authorization process for each vehicle. The authorization process for a vehicle consists of multiple requests, if the vehicle is requesting the first authorization ticket the enrollment will be validated by the EA, otherwise the RA informs the AA that the vehicle's enrollment was previously validated. For the first authorization request, the vehicle starts by building the authorization request structure, such request contains all the information that the EA needs to validate the vehicle's enrollment status and that the AA needs in order to issue an authorization ticket. To ensure security and privacy in this type of communication, the vehicle's enrollment information such as its enrollment signature is encrypted to the EA and the authorization information encrypted to the AA. The vehicle sends message 1 containing the authorization request to the RA which knows how many requests the vehicle will perform during its authorization process. Message 2 shows the request being sent to the AA which decrypts it and sends the enrollment information to the EA in message 4. The EA decrypts its part of the request, validates the vehicle's enrollment signature, using the vehicle's enrollment credential (referenced by the request), and validates if such certificate is valid. If the verifications are in order, the EA sends message 4 which notifies the AA that the vehicle is authentic. The AA then issues the Authorization ticket, builds an authorization response structure which is signed, encrypted to the vehicle and sent to the RA within message 5 as a positive authorization.

At this point the RA knows that the vehicle is enrolled and that the authorization was successful, and is able to return the response to the vehicle in message 6. For the remaining requests the vehicle builds the authorization request and sends it to the RA as usual, as seen in message 7. The RA knows that the vehicle's enrollment has been validated for this authorization process and sends the request to the AA in message 8, the AA decrypts the requests, validates the authorization information, issues the authorization ticket, and returns the response to the RA in message 9. In the last step, message 10 shows the RA returning such request to the vehicle which stores the requested certificate. The same flow is repeated until the vehicle has all the authorization tickets for the time specified by the RA.

B. V2X Library

The architecture of the V2X Library can be described as a layered structure, where at the highest level we can find the classes which allow the generation of the more specific data structures such as the EA, AA and Root CA certificates; vehicular authorization and enrollment certificates; and the V2X messages such as CAMs. As we go down a level the can find the more general substructures which the higher level structures depend. Finally, at the lower level we can find the classes which enable the transmission of such data structures in a cross-platform way using the Abstract Syntax Notation 1 (ASN.1) Canonical Octet Encoding Rules (COER).

C. PKI Manager

Certificate issuing is an essential part of this work, however due to time constraints we were not able to extend mPKI. Our solution to this problem was to create the PKI Manager, a web application that functions as a demo PKI. The PKI Manager contains a backoffice which was implemented with the sole purpose of demonstrating the creation of vehicular certificates.

The PKI Manager is a web application implemented using the Spring Boot framework. This framework produces a stand-alone application and aims to reduce the time spent configuring it. The application can accessible through the browser only by system administrators. This backoffice application uses the V2X Library and through its services provides a GUI where administrators can visually create and configure a demo vehicular PKI.

D. RA Serivce

The RA service is a module of the PKI Manager, this means that it is implemented within the same Java project and thus shares the same resources, such as the database and the internal service layer. However, to give the illusion of distance between the RA and the PKI, we decided to separate the database into the part used by the PKI Manager and the part which is used by the RA. In addition, we also separated the internal services which the PKI provides to the RA from the services used by the PKI Manager to manage the PKI. At a high level, the RA Service is a RESTful API from which the PKI Manager is able to communicate over the internet with other software programs, providing the services of **configuration**, **enrollment** and **authorization**.

E. Summary

In this chapter we presented our solution to the problem of implementing a PKI solution that allows a V2X environment while preserving the privacy of the vehicles. We started by overviewing the architecture of the solution, learning about its main components: PKI Manager, V2X Library and Vehicle Manager. We studied how such components are organized in a client-server architecture, assuming the PKI Manager as the server and the Vehicle Manager as the client. In addition, we seen that these two components communicate through the API of the server, the RA Service and utilize the V2X Library in order to generate the necessary messages. In the next chapter we perform an experimental evaluation to our V2X system, where we test the performance of the server, the resource usage of the client and finalize by discussing the security and privacy properties achieved.

IV. Evaluation

This Section describes the evaluation of our V2X system, which comprises the PKI Manager and Vehicle Manager. The three main non-functional requirements of our V2X system are performance, security, and privacy. With this in mind, we intend to answer the following questions:

1) Since interoperability is a constant concern, does the system provide acceptable performance?
2) Does the system deliver the necessary conditions for vehicle privacy, authentication and overall security?
3) Does the implemented V2X environment provide vehicle privacy at an acceptable cost?

The following sections address these three questions, staring with the performance and resource usage tests, moving to the privacy and security concerns.

A. Performance

Performance is a fundamental concern when testing systems that address a sensitive subject such as road safety. The most common metrics of performance are latency and throughput: latency is the time necessary to complete certain operations and throughput in our case is defined by the number of requests that the PKI Manager processes in a determined time frame.

To perform this evaluation several tests were done regarding the backoffice application, and the interaction between the Vehicle Manager and PKI Manager. The environment in which they execute is the following:

- A single installation of the PKI Manager application as the server.
- A basic but functional vehicular PKI including one Root CA, one Enrollment CA, and one Authorization CA.
- A single installation of the Vehicle Manager application as the client.

Our goal with this setup is to provide a simple but complete V2X environment, where we can use the installation of the PKI Manager to test the performance of the backoffice and RA Service, and use the Vehicle Manager to execute tests focused the resource usage of V2X communications.

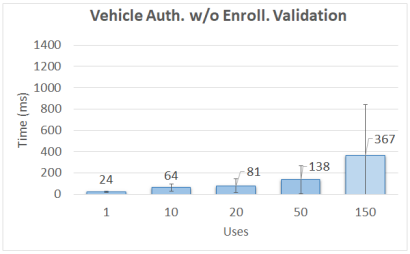
1) Backoffice Application: The operations performed by the administrator application are simple but essential in order to have a functional PKI to serve as the backbone of trust within V2X communications. The processing is mostly located at the V2X Library, creating a waiting period in the backoffice application. On the administrator side, it makes sense to only test the latency of the operations performed by one user only, as this application is not meant to be used concurrently by a large number of users. Next, we take a look over the operations evaluated on the backoffice: the generation of keys and certificates.

In Table I we can observe the time necessary to complete the operations involving our backoffice application, which preclude user interaction. Overall from the perspective of an administrator the results are reasonable and will not negatively affect the usability of the application. Regarding the operation to add a certificate, the operations for adding a single key or certificate are fast, taking on average 71.6 and 73.7 milliseconds. There is a linear relation between the time that the system takes to add a certificate and the number of keys that exist on storage. This happens due to the nature of the operation, in which the server has to query the database and the keystore in order to find the cryptographic keys of the subject and issuer before calling the V2X Library to issue the certificate, and finally storing it on the database. This increase in delay is not critical, as considering a realistic number of keys (i.e., in the order of a few dozen keys) it does not affect the user experience to a point that it is noticeable.

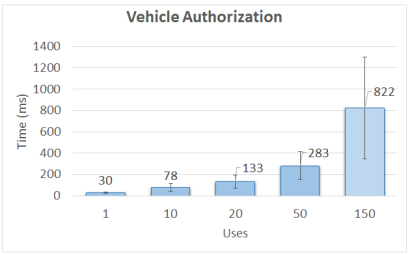
2) RA Service: The RA Service API is the gateway from which the vehicles are able to interact with the PKI to request certificates. Comparing with the backoffice application, the operations done by the RA Service are more complex and subject to more load due to the possibility of concurrent

Operation	*Time*	*Keys*	*StandardDeviation*
Add key	71.6ms	1	14.0ms
Add certificate	73.7ms	4	10.0ms
Add certificate	81.0ms	20	13.5ms
Add certificate	103.0ms	30	14.0ms
Add certificate	128.6ms	40	16.0ms
Add certificate	178.7ms	60	19.5ms

Table I: Time needed to add a certificate and a key, calculated over twenty measurements.



(a) Vehicle Authorization w/o enrollment validation



(b) Vehicle Authorization

Figure 6: Latency measurements for the Authorize operation

requests from the client vehicles. Therefore, the tests done to this component are designed to evaluate the behaviour of the API at different levels of server load Figure 6 shows the tests done to the RA Service: latency resulting from authorization of vehicles.

The most interesting comparison allowed by these tests is the comparison of the latency introduced by the two different vehicle authorization methods. As we can see from plots (c) and (d), the authorization without vehicle configuration introduced by our RA Service is on average faster than the standard full authorization flow. For example, considering the lowest server load of 1 user the new authorization flow represents an improvement of 16.6% of the full authorization. As the number of concurrent users increases so does the improvement reaching its maximum value on the 150 user threshold with an improvement of 55.7%. Since this version of the authorization is the most frequently used throughout the authorization process of the vehicles, the performance gains mean that the efforts presented in Section III-A3 to optimize the authorization process as a whole were successful from a performance point of view.

3) Vehicle Manager: At the Vehicle Manager level, the focus of the latency/throughput metrics tests shifts from performance to resource usage.

In order to have a general notion of the possible quantities and the necessary storage capacity, Table II lists the type

Certificate	$Size$	$Quantity$	$TotalSize$
Root CA	272 bytes	20	5.3KB
Enrollment CA	288 bytes	100	28.1KB
Authorization CA	288 bytes	100	28.1KB
Enrollment Credential	208 bytes	1	0.2 KB
Authorization Ticket	200 bytes	1825	356.4KB
AT private key	32 bytes	1825	57.0KB
			475.1KB

Table II: Estimated vehicle certificate storage demand

of certificates, their size, and possible quantity. Regarding the possible quantity of ATs stored within the vehicles, we assumed that a vehicle uses on average five certificates per day with a refill time of one year. In terms of numbers, the expected AT number can be calculated as (5 certificates) * (365 days) which equals to 1825 ATs. For the number of CA certificates we assumed a decently sized truststore installed on the vehicle. As we can see from Table II, the authorization tickets and their corresponding private keys are the elements which demand more storage space (413.4KB), corresponding to approximately 87% of total. It is also important to note that the number of the certificates can change from vehicle to vehicle depending on the usage pattern and the desired level of privacy, so the overall certificate storage demand may not be the same for different types of vehicles. This is further discussed in the following section.

B. Security

In this section we analyse how our system achieves the security goals previously defined in Chapter I. In particular, we discuss the following security properties: confidentiality, data authenticity, and authorization.

1) Confidentiality: To ensure confidentiality, all the information transmitted from and to the client is sent through an SSL secure channel. In this communication channel the Vehicle Manager is authenticated, where the server requires its certificate in order to accept communications.

In addition, we also provide application level encryption where every certificate request is encrypted by the vehicle so that only the intended recipient CA is able to decrypt it. This feature combined with the encryption of the response to the original vehicle allows the confidentiality of the vehicle's identity, preventing its disclosure during the certificate requests.

2) Data Authenticity: Data authenticity enables the receiver of a message to confirm the origin and integrity of the data. In our system, this property applies to both the client-server communication and V2X messages.

Regarding the Vehicle to PKI communication, the authenticity of the messages is achieved at two levels: first at the channel and then at the application levels. The sender of messages is the Vehicle Manager and the recipient is the PKI Manager. When the vehicles are manufactured, the RA trusts the new connections because of the SSL connection with the Vehicle Manager, which can be seen as the vehicle manufacturer. However, after the configuration of such vehicles the authenticity of their certificate requests is achieved at the application level, through the use of digital signatures.

The authenticity of the V2X messages is a similar process where a vehicle being the recipient of a V2X message is able to trust the authenticity of the message by verifying the signature of the sender vehicle against a trusted certificate chain.

3) Authorization and Authentication: Authorization over services requires user authentication. In our system this is applied in two aspects: the authentication of the administrator to manage the PKI and the client vehicles to request certificates.

The authentication of the administrator is accomplished through the use of a username/password mechanism implemented by the *Spring Security* framework.

Regarding the authorization of the vehicles, this is achieved through the RA service of vehicle configuration. Only vehicles which completed this service are recognized by our server and are able to request certificates.

C. Privacy

In our system preserving the privacy of users is one of the most important concerns. Losing privacy might mean that an attacker is able to track vehicles using the V2X communications, which is undesired. The main system variables that might affect the privacy of its users are the number of pseudonym certificates, and the usage pattern. The more pseudonym certificates a vehicle has available, the better is the privacy because the vehicle will have more choices when changing pseudonym. However, as described in the previous sections the increase of privacy comes with the overhead of the increase in certificate storage demand, number of individual certificate requests, and consequently processing at the vehicle.

A general rule of thumb that is important to consider when using pseudonyms is that a pseudonym is only effective in a crowd. In vehicular communications this means that a pseudonym certificate is only effective if there are more vehicles around using them. If a vehicle is isolated on a road sending V2X messages signed with a pseudonym certificate then that certificate can be immediately mapped to the sender, even considering a high pseudonym change frequency.

Pseudonym certificates are particularly vulnerable to reuse attacks where the attacker is able to map a specific certificate to a vehicle. In this case the pseudonym becomes compromised because the attacker will be able to link that certificate to the vehicle when it reuses it. The more certificates are compromised the bigger the chance of the attacker to infer a vehicle's route by observing the compromised certificates being reused at different locations. However, assuming that vehicles carry thousands of pseudonyms, a few compromised certificates might not be conclusive for the attacker. It all depends on the power of the attacker to compromise certificates by controlling areas and/or following the victim vehicle, the number of certificates reused by the vehicle, and the randomness in which they are selected.

Vehicles configures with a low pseudonym change frequency are particularly vulnerable to short-term tracking attacks where the attacker is following the vehicle and is able to: (a) see the vehicle using the certificate while isolated, or (b) infer it by the captured messages which are signed constantly by the same certificate. We concluded that the best way to preserve privacy is to increase the number

of pseudonyms, which enables the possibility of a higher pseudonym change frequency and lowers the need to reuse certificates.

More concretely, the price in terms of storage in order to have acceptable privacy depends on the travel distance of the vehicles. In the previous section, we estimated that if a vehicle uses on average 5 pseudonyms in a day with a refill time of one year it would need approximately 486.6KB of certificate storage space. Depending on the distance that a vehicle travels each day, it may need to use more that 5 certificates resulting in the reusage of certificates, and privacy loss in most cases. To assure that this does not happen a vehicle that spends more time on the road needs more certificates. For example, assuming a vehicle changes certificate each 10 minutes and an average travel time of 2 hours each day. The vehicle would need approximately 12 certificates a day in a total of 4380 for a year. Assuming the same number of PKI certificates as in the previous section IV-A3, this number of pseudonyms would increase the storage demand to approximately 1MB.

D. Summary

In this chapter we presented the evaluation results of the developed V2X system. We started by evaluating the performance of the PKI Manager, then looked at the resource usage of the Vehicle Manager and finished by discussing the security properties and privacy of the system.

The performance of the PKI Manager was measured at two different variants: the backoffice application and the RA Service. In the evaluation of the backoffice we measured the latency of the most complex PKI operations, the generation of keys and CA certificates. We concluded from the results that the measurements were reasonable, considering the complexity of the operations, and would not negatively affect the usability of the application by an administrator. From the results it was possible to conclude that the most expensive operations in terms of latency are the full authorization and enrollment of vehicles. The new authorization of vehicles introduced by our RA Service was faster than the standard full process.

Lastly we presented how our system provides the security properties of confidentiality, data authenticity and authorization in both vehicle to PKI and V2X Communications. We concluded this chapter by analyzing how the privacy of the vehicles is maintained in V2X communications, discussing the different levels of privacy achieved by different vehicle types and pseudonym usage patterns.

V. Conclusions

In the work, we developed a V2X ecosystem which is compatible with the most recent standardization efforts done by ETSI. Starting with the PKI Manager containing the vehicular PKI and the RA Service, which vehicles to interact with the PKI in conformance with the standardized certificate request formats. Going to the Vehicle Manager representing the role of a vehicle manufacturer, generating client vehicles an allowing V2X communication within its functionalities.

The developed system achieves a first approach to identity management of the vehicles on the road while preserving their privacy. The PKI Manager provides a way of analyzing the necessary certificates and messages, allowing the measurement of the size of such data structures and the impact that they might have on the vehicles who store them. The Vehicle Manager provides a method of seeing the V2X communications in action, allowing us to analize the privacy of the communication by looking at the received and sent messages for each participant vehicle.

Our contribution with the RA Service manages to improve the Vehicle to PKI interaction by serving as a proxy capable of configuring new vehicles and authorizing them for future communications. According to the results, the RA Service also manages to improve the performance of the vehicle authorization process as a whole.

References

[1] "European commission mobility and transport," https://ec.europa.eu/transport/road_safety/specialist/statistics_en, 2016.

[2] Y. Lin, P. Wang, and M. Ma, "Intelligent transportation system (its): Concept, challenge and opportunity," in *Big Data Security on Cloud (BigDataSecurity), IEEE International Conference on High Performance and Smart Computing (HPSC), and IEEE International Conference on Intelligent Data and Security (IDS), 2017 IEEE 3rd International Conference on*. IEEE, 2017, pp. 167–172.

[3] S. Rehman, M. A. Khan, T. Zia, and L. Zheng, "Vehicular ad-hoc networks (vanets)—an overview and challenges," vol. 3, pp. 29–38, 01 2013.

[4] E. C. Eze, S. Zhang, and E. Liu, "Vehicular ad hoc networks (vanets): Current state, challenges, potentials and way forward," in *Automation and Computing (ICAC), 2014 20th International Conference on*. IEEE, 2014, pp. 176–181.

[5] "Ieee standard for wireless access in vehicular environments; security services for applications and management messages," https://standards.ieee.org/findstds/standard/1609.2-2016.html, 2016.

[6] "Intelligent transport systems (its); security; security header and certificate formats," http://www.etsi.org/deliver/etsi_ts/103000_103099/103097/01.02.01_60/ts_103097v010201p.pdf, 2015.

[7] N. Bißmeyer, H. Stübing, E. Schoch, S. Götz, J. P. Stotz, and B. Lonc, "A generic public key infrastructure for securing car-to-x communication," in *18th ITS World Congress, Orlando, USA*, vol. 14, 2011.

[8] B. Lonc and P. Cincilla, "Cooperative its security framework: Standards and implementations progress in europe," in *World of Wireless, Mobile and Multimedia Networks (WoWMoM), 2016 IEEE 17th International Symposium on A*. IEEE, 2016, pp. 1–6.

[9] W. Whyte, A. Weimerskirch, V. Kumar, and T. Hehn, "A security credential management system for v2v communications," in *Vehicular Networking Conference (VNC), 2013 IEEE*. IEEE, 2013, pp. 1–8.

[10] "Intelligent transport systems (its); security; trust and privacy management," http://www.etsi.org/deliver/etsi_ts/102900_102999/102941/01.01.01_60/ts_102941v010101p.pdf, 2012.

[11] "U.s. department of transportation. safety pilot model deployment," http://safetypilot.umtri.umich.edu/.

[12] "Scms cv pilots documentation," https://wiki.campllc.org/display/SCP/SCMS+CV+Pilots+Documentation.

[13] "Etsi automotive intelligent transport systems," http://www.etsi.org/technologies-clusters/technologies/automotive-intelligent-transport, 2017.