A Public Key Infrastructure for Securing Vehicle-to-Everything Communication

Leonardo Gonçalves
leonardogoncalves@tecnico.ulisboa.pt

Instituto Superior Técnico
Advisors Prof. Nuno Santos & Eng. Carlos Cardoso

Abstract. *Vehicle-to-everything* (V2X) communication has increasingly become the target of research and standardization efforts in Europe, America and Asia. This term refers to the exchanging of information between a vehicle and any entity that may affect the vehicle; such entities can be, for example, other vehicles, pedestrians or roadside units i.e. semaphores, road barriers, signs, etc. V2X communication is an essential feature of autonomous vehicles in the future. Such vehicles are envisioned to increase road safety, driver comfort, and fuel economization through traffic efficiency. This work aims to survey the state-of-the-art of the V2X communication from a cyber-security point of view. We analyze different existing solutions regarding the *Public Key Infrastructure* (PKI) and the standardization efforts needed to support a V2X driving environment.

Keywords: privacy · communication · vehicles · vehicular had-hoc network · public key infrastructure · digital signatures · digital certificates.

Table of Contents

1 Introduction

According to the European commission statistics [6] 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) [16] 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. 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) [19] [14]. 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 [7] and ETSI TS 103 097 standards [5] 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. How-

ever, 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.

1.1 Goals

This work addresses the problem of designing and implementing a vehicular PKI solution that allows for V2X message authentication while preserving the privacy of its users. This report will specify the system to produce, an easy-to-evaluate vehicular PKI that supports the enrollment of new ITS-S, provisioning of valid certificates to its users and the removal of compromised stations or PKI entities. The goal of this work is to design a PKI solution based on the most recent European standards and follows 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.
  - Unauthorized parties must not be able to link a V2X message to that vehicle's previously sent messages.
  - It should not be possible to deduce a given vehicle's location by analyzing previous communications to and from the vehicle.
- Confidentiality
  - Information transmitted to or from a given ITS station must not be disclosed to unauthorized parties.
- Integrity
  - Information transmitted to or from a given ITS station must be protected against unauthorized modifications or tampering during transmission.
- Authenticity
  - It should not be possible for users to spoof another legitimate ITS station to communicate with other stations.
  - It should not be possible for an ITS station to receive management and configuration information from unauthorized entities. For example spoofing of a *Certification Authority* (CA).
- Availability
  - Access to ITS services and applications should not be prevented to legitimate users by malicious activity.

1.2 Expected Contributions

The proposed solution consists of a vehicular PKI and a simulator to evaluate its correctness. Our solution will extend mPKI, a currently operating traditional PKI which is the product of Multicert. In order to extend mPKI, we will start by developing a Java package that implements the new certificate formats. The next step involves integrating such package in mPKI to allow it to issue the certificates for the end-entities (ITS-S) and CAs. The final step is to develop a simulator to test the correctness of the PKI. In this phase we will develop a Java simulator that will simulate the end-entities and the interaction between such end-entities (V2X) and the vehicular PKI.

The remainder of this report is organized as follows: Section 2 surveys the state-of-the-art and existing solutions, Sections 3 - 4 specify our solution and how to evaluate its results, Section 5 schedules future work, finally Section 6 concludes the report.

2 Related Work

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 [12] [5] [17] (Section 2.1). The second is named *Security Credential Management System* (SCMS) and is the American counterpart proposed in [25] (Section 2.2). For each solution, we analyze the corresponding PKI architecture and its most relevant operational aspects such as enrollment of vehicles, revocation of certificates, and others. In Section 2.3, 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. Lastly in Section 2.4, we introduce the existing V2X communication simulators and study how they can be used to evaluate our vehicular PKI solution.

2.1 Overview of the European Vehicular PKI Solution

The European vehicular PKI solution is a generic concept so it allows some flexibility in its implementation. Our proposed PKI will primarily be based on the European solution. Specifically we aim to readjust such solution so it can be integrated in Multicert's PKI.

2.1.1 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 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.

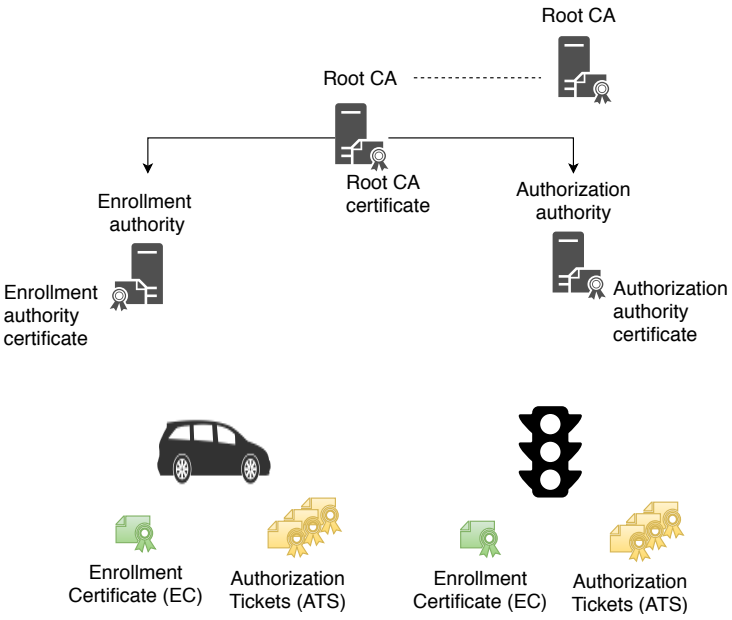

Fig. 1. European vehicular PKI architecture. [17]

2.1.2 ITS-S Security Life Cycle

The ITS-S security life cycle is relevant for our solution because through its analysis we are able to understand which stages every vehicle undergoes during

its life time. Each stage involves a change in the vehicle's state against the PKI. The analysis of such stages allows us to understand when and what information needs to be initialized in the vehicles and transfered between vehicles and the PKI. Specifically, which variables need to be initialized in the vehicles' OBU at bootstrap, what needs to be done in regards to the vehicle enrollment and authorization, and how can we update PKI management information throughout the life cycle of the vehicles. ETSI TS 102 941 [4] standards specify the ITS-S' security life cycle comprising four stages: the manufacture, enrollment, authorization and maintenance. The first stage of an ITS-S' life cycle is the manufacture, it is at this stage that all the information needed for the enrollment is initialized within the station itself (OBU) and within the EA. The next stage is the enrollment where the ITS-S requests its enrollment certificate from the EA. Having received the enrollment certificate, the ITS-S can now request authorization tickets from the AA. The request for authorization tickets represents the authorization stage. In the case that a EA or AA is removed or added from the group of trusted authorities, the enrolled ITS stations must be notified. The update can be done by distributing the *Certificate Revocation List* (CRL), or during ITS-S maintenance stage in a controlled environment.

When a vehicle is manufactured the OBU and EA need to be initialized. Within the vehicle's OBU, it is necessary to provide information regarding the vehicle's identity and information to allow the vehicle to interact with the PKI. In regards to identifying information, an unique identifier and a public/private key pair to be used for cryptography are created. Optionally, a canonical certificate can be installed which associates the canonical identifier with the public key of the vehicle. In this case, the certificate chain back the root authority needs also to be installed. To allow a vehicle to connect to the PKI, it is necessary to install the network address and public key certificate of the EA and AA that will issue certificates for that vehicle. In addition, the set of known and trusted EA certificates is installed to allow the vehicle to initiate the enrollment process. To grant that such vehicle is able to verify the authenticity of incoming V2X messages, the set of known and trusted AA certificates must also be provided. In order to support vehicle enrollment, within the EA it is necessary to provide information that identifies the manufactured vehicle: a unique vehicle identifier, the location profile information for the vehicle, and the public key that belongs to the vehicle's key pair. In the next sections we study how the European Vehicular PKI operates regarding the provisioning of certificates, their revocation ,and how the messages are signed and verified by the user vehicles. Our vehicular PKI must support such operational aspects.

2.1.3 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 shall be sent from the ITS-S to the Enrollment Authority, to protect the users privacy the request must be encrypted. According to ETSI TS 102 941 [4] this message shall contain the following fields:

- Message signer information, i.e. the canonical certificate or the public key provided to the ITS-S at bootstrap to globally identify it.
- certificate request, i.e. the information to be presented in the enrollment certificate. For example, the ITS-S' public key, start time, end time and other certificate specific data.
- The digital signature of the message sender (requesting ITS-S) calculated over all of the message fields.

After the ITS-S enrollment request the target EA must reply with a successful or failed response message, to protect user privacy the response shall also be encrypted. The successful ITS-S enrollment response shall contain the enrollment certificate and the chain of certificates back to the originating enrollment CA. In the failed ITS-S enrollment, the response shall contain the error code i.e. the reason for the unsuccessful enrollment response.

2.1.4 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. There is the need to adopt a model that specifies the authorization ticket provisioning. The model used in Europe is defined in CAR 2 CAR Communication Consortium (C2C-CC) [12]. In this model the required frequency of updates, the delivered level of privacy and security can be expressed by three determining parameters:

- **Certificates valid simultaneously:** It can be defined that the requester can use several ATs with the same start and expiry date.
- **Authorization ticket validity time period:** Is defined by the time between start and expiry timestamps of the ATs.
- **Overall covered time-span:** The time that is covered by the batch of authorization tickets.

A vehicle receives a "super-batch" that contains a set of ATs, the duration of the "super-batch" is the *overall covered time-span* and is in the order of years. The "super-batch" is composed of "sub-batches" which contain a sub-set of certificates (e.g., 20) valid for the same time period (e.g., a week). During that week the vehicle uses and reuses the 20 valid certificates. The certificate usage pattern can vary from device to device, e.g. a device could use a certificate for 5 minutes after start-up, then switch to another certificate, and use that either for 5 minutes, or until the end of the journey. Figure 2 illustrates this method.

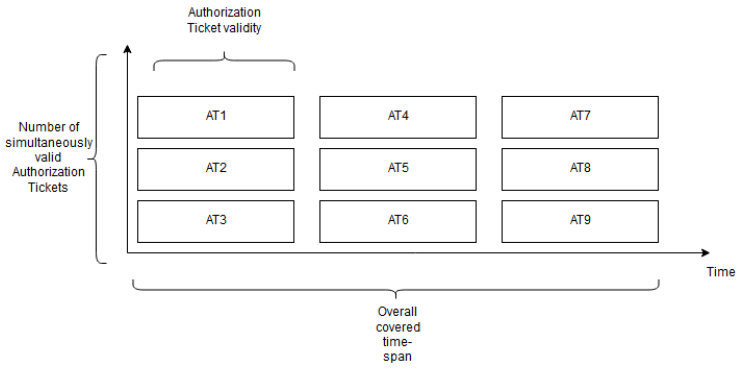

Fig. 2. Authorization Ticket provisioning model.

2.1.5 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 [4] standard specifies in detail the message format for the AT request and response. In regards to the process [12], 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 list of public keys. 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 and the calculated AT overall covered time-span (e.g. 1 year) to the correct EA (identified by the EA certificate present in the original request). The EA maintains a database that stores a timestamp marking the deadline which the vehicle will still have valid ATs (calculated using the overall covered time-span of the request). Only if the vehicle's enrollment certificate is valid and no ATs are issued for the time which the vehicle still has valid ATs 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. This procedure prevents a vehicle from requesting ATs for the same time interval from different or the same AAs.

2.1.6 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 (more details present in Section 2.3).

2.1.7 Certificate Revocation

Sometimes it may be necessary to remove bad actors from the system. This requirement influences the architecture of the PKI, namely there must be an entity responsible for detecting misbehaving actors. Once detected, a bad actor must be removed from the communication.

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. If the vehicle requests new ATs then the AA forwards the request to the respective EA which checks the revocation information of the requester. In respect to the revocation of any CA certificate a distributed CRL is used. In this scheme, the CA certificates that are compromised are revoked manually by the PKI administration; the certificate identifier is posted in the CRL and signed by the root CA; finally, the CRL is distributed inside the PKI backbone and connected ITS stations. The CRL for EA and AA certificates is defined in ETSI TS 102 941 [4] standard

2.2 Overview of the American Vehicular PKI Solution

In this section we provide an overview of the Security Credential Management System as a matter of reference. Although this theses follows the European PKI solution, it is relevant to reference the American vehicular PKI in order to understand the main differences and similarities between solutions. In comparison to the architecture of the European vehicular PKI the American counterpart is noticeably more complex. However, it shares some similarities with the European vehicular PKI. The main differences include an increased focus on privacy against attacks from SCMS insiders, the handling of certificate revocation, and the method for provisioning certificates based on the butterflykey expansion algorithm.

2.2.1 American Vehicular PKI Architecture

SCMS considers an hierarchical structure as we can see in Figure 3. 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:

- **Device Configuration Manager (DCM):** Provides authenticated information about changes in the configuration of SCMS's components, for example an authority changing its certificate. It is also used to inform an enrollment CA that a device is eligible to receive enrollment certificates.
- **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.
- **Linkage Authority (LA):** The main goal of this component is to improve certificate revocation. LAs generate linkage values, which are used in the certificates and support efficient revocation (more on this later). There are two LAs in the SCMS, referred to as LA1 and LA2. The splitting prevents the operator of an LA from linking certificates belonging to a particular device.
- **Location Obscurer Proxy (LOP):** The main goal of this component is to improve the security against SCMS insiders. The LOP hides the location of the requesting device by changing source addresses. Additionally, when forwarding information to the Misbehavior Authority (MA), the LOP shuffles the reports to prevent the MA from determining the reporters' routes.
- **Misbehavior Authority (MA):** Processes misbehavior reports to identify potential misbehavior by devices, and if necessary revokes and adds devices to the CRL. It also initiates the process of linking a pseudonym certificate to the corresponding enrollment certificates, and adding the enrollment certificate to an internal blacklist.
- **Request Coordination (RC):** Ensures that a device does not request more than one set of pseudonyms for a given time period. It coordinates activities between different RAs.

An early version of SCMS has already been implemented, operated and tested in the safety pilot project [3]. 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 [2]

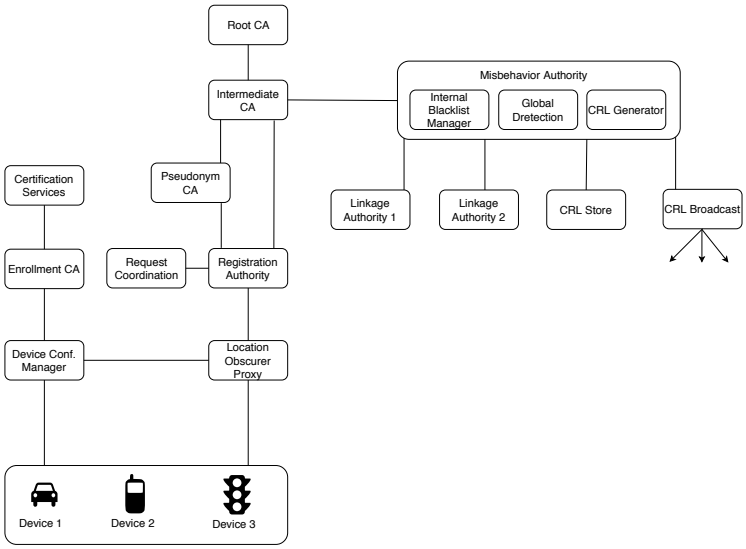

Fig. 3. SCMS overall system architecture [25].

2.2.2 Pseudonym Certificate Provisioning Model

Regarding the model used for provisioning pseudonym certificates, the SCMS assumes the same model as in CAR 2 CAR Communication Consortium [12] (European solution) illustrated in Figure 2. The proposed parameters for this model are:

- **Certificate validity time period:** 1 week.
- **Certificates valid simultaneously (batch size)**: 20 to 40 certificates.
- **Overall covered time-span (super-batch size):** 1 to 3 years.

2.2.3 Pseudonym Certificate Request Process

The request for pseudonym certificates in itself is different from the European solution. For this the butterfly key expansion algorithm [25] 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 present in [25]:

2.2.4 Misbehavior Reporting

In contrast to the European PKI, the American PKI supports misbehavior re-

porting 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 via the LOP which will obscure the source and shuffle the reports from multiple reporters, this is done to prevent the MA from reconstructing the reporter's path based on the reports. The format of a misbehavior report is not fully defined yet, but a report will potentially include reported messages in addiction to the reporter's signature and certificate, and will be encrypted by the reporter for the MA.

2.2.5 Global Misbehavior Detecting and Revocation

The algorithms necessary for global misbehavior detection have not been developed at the time of this writing. However, the interface which allows SCMS components to retrieve linkage information is already specified. Revocation is tightly bound to the linkage information which basically allows the MA to find whether multiple reported messages point to the same device. The revocation process is described step-by-step in [25]. In this section we learned about the European and American vehicular PKIs and about their most relevant operational aspects. In the next section we present the standards which shape the formats of the certificates and messages used in the European vehicular PKI.

2.3 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 [8], IEEE 1609.2 represents the main standardization stakeholders in the U.S [7]. 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 [17].

In regards to the secured message and certificates formats. IEEE 1609.2 [7] standard defines the formats for secured V2X messages and public key certificates to be used in SCMS. In this standard the V2X message authenticity and integrity are based on the *Elliptic Curve Digital Signature Algorithm* (ECDSA). The message confidentiality protection is based on AES symmetric encryption (AES-CCM mode). For the transport of symmetric keys the *Elliptic Curve Integrated Encryption Scheme* (ECIES) is used. ETSI TS 103 097 standard [5] assumes the same cryptosystem as IEEE 1609.2 and presents security profiles for the messages and certificates also based on the IEEE 1609.2 standard. This means that ETSI's profiles are specific types of messages and certificates which are based on particular options available on the definitions of the base 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 PKI solution 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.

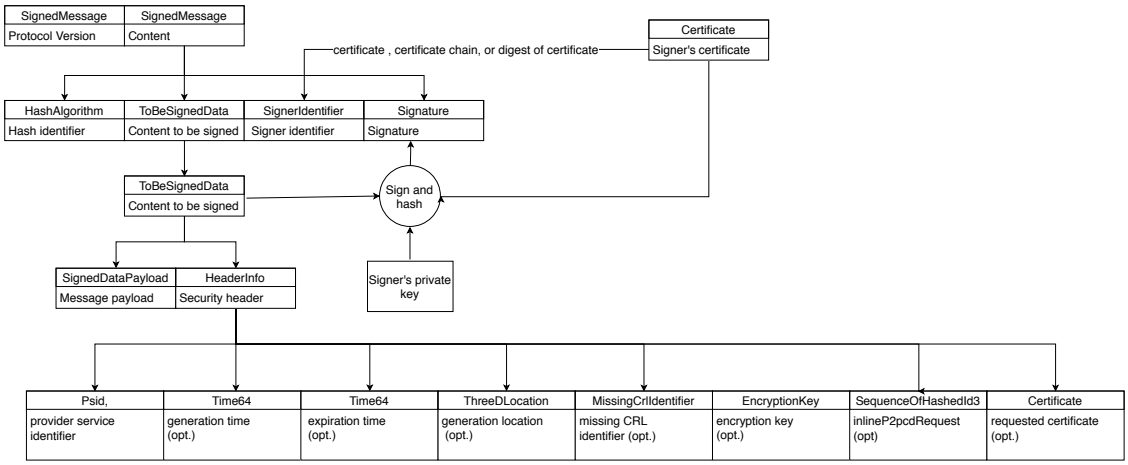

Fig. 4. IEEE 1609.2 signed message format used by ETSI TS 103 097.

2.3.1 Secured Messages Formats

Figure 4 depicts the format of a secured message. Generally a message can be transmitted as unsecured data, signed, encrypted, or signed and encrypted data. In the case of road security messages, the message should be signed and thereby include the hash algorithm, the content to be signed, the signer identifier, and the signature itself. Such security messages need to carry the signing certificate (authorization ticket) to reduce the processing delay at the receiver side. However, in order to reduce the network bandwidth consumption it is possible to include in the message a reference to the signing authorization ticket. For this purpose, the message signer identifier contains the certificate identifier as the 8 bytes certificate digest instead of the full certificate.

The content to be signed includes all of the message components that will be protected by the signature. Such components are the security header and the message payload. The security header includes components that are relevant for the security layer, such as the provider service identifier and some optional message validation data (generation time, expiration time, generation location,

missing CRL identifier, encryption key, *inlineP2pcdRequest,* and requested certificate).

A message is signed by an authorization ticket, which is turn is signed by an authorization authority certificate. Consecutively the authorization authority certificate is signed by a higher authorization authority certificate. The chain ends at a root CA certificate which issued itself. At a high level, at least one certificate in this chain must be known and trusted by the receiving station in order for it to be able to trust an incoming message.

The message receiver needs to be able to construct a chain from the message signing certificate to a known root. However, vehicles are constantly rotating authorization tickets to sign safety messages, normally exchanging only a reference to that certificate (authorization ticket hash). In addition, in many cases vehicles share the road with previously unknown vehicles for the first time. There needs a peer-to-peer (p2p) mechanism to distribute certificates on the road. This mechanism in embedded into the secured messages, specifically the receiver can use the *inlineP2pcdRequest* component of a message to request unknown certificates from other senders.This functionality allows us test the correctness, performance and overhead of the distribution of unknown authorization tickets on the road.

2.3.2 Secure Messages Profiles

In the previous section we learned about the possible message components and their meaning. Here, we analyze the already standardized safety message profiles for the *cooperative awareness messages* and *decentralized environmental notification messages*. For each type of message, we analyze their goal, how they are sent, and how they are received by the vehicles. The proposed simulator will implement the V2X communications between simulated vehicles. As such, it assumes these two specific types of messages secured by the ATs provided by the proposed vehicular PKI.

2.3.2.1 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 shall 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 the component *inlineP2pcdRequest* of 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 the *inlineP2pcdRequest* of its next CAM the digest of the unknown authorization authority certificate which is present on the AT itself (see more in Section Certificate formats).

If a vehicle receives a CAM containing a request for an unknown certificate i.e. with the component *inlineP2pcdRequest* on the security header, then the vehicle searches the list of certificate of digests existing in that component. If the digest of the currently used authorization ticket is found in that list, then it includes the full certificate in the component signer identifier of its next CAM instead of the digest. In the case that a vehicle finds a digest referencing a valid authorization authority certificate in that list, it should include such certificate in the component requested certificate of its next CAM to broadcast the response. It is possible that multiple neighbor vehicles have stored the requested AA certificate, in order to prevent unnecessary broadcast responses, a vehicle only includes the AA certificate in its next CAM if before the generation of this message no other CAM was revived containing the AA certificate in the component requested certificate.

2.3.2.2 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 reviver 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.

2.3.3 Certificate Formats

In the previous sections we have seen the secured message formats and profiles, which are relevant for V2X communications. In this section we introduce the existing certificate formats, which are relevant to secure such messages. Our goal is to encode such formats in a Java package and then integrate it into mPKI.

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. For example, authorization tickets carry the digest of their corresponding authorization authority certificate. The certificate identifier is a unique name that identifies the certificate's issuer (i.e. name of a certification authority). The application permissions contain one or more pairs of provider service identifier (PSID) and service specific permissions (SSP). While PSID indicates a specific service, message or application the SSP indicates the permissions within that service. For example, there may be an SSP value associated with PSID of a CAM that indicates that the vehicle is privileged to send such message for a specific vehicle role (e.g emergency vehicle) or for a specific roadside unit (e.g traffic lights).

2.3.4 Signed Message Validity Checks

Before a vehicle is able to trust an incoming message it must check the message's validity. This is done by verifying that no certificate in the certification chain is revoked and the signing certificate chains to an already trusted CA certificate. Also the signature present in the message can be verified using the public key expressed in the certificate. The message payload must be consistent with the permissions expressed in the certificate (by the PSID/SSP pairs). Finally the message must not be not expired, i.e. the message validity is within the certificate's validity period and the message was generated within the permitted geographic location of the signing certificate. Figure 5 depicts the consistency between a signed message and the signing authorization ticket.

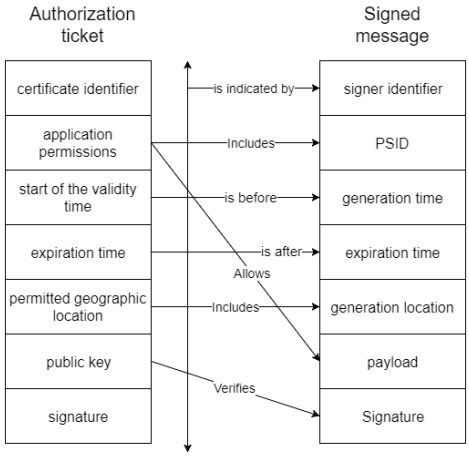

Fig. 5. Relation between a signed message and the signing certificate.

2.4 Overview of the ITS Simulators

Implementing V2X communication is expensive and may prove to be dangerous to test using real vehicles. Furthermore, to properly measure the benefits of V2X communication we need to evaluate it at a large scale, for example hundreds of ITS-S. Before we can conduct a field test, a simulation framework which is able to test the communications between vehicles and infrastructure of whole cities is needed. This implies simulation at three different domains: traffic simulation to generate the road networks and traffic demand; network simulation to allow vehicle connectivity by wireless technology (e.g. IEEE 802.11p DSRC, and IEEE 1609.4 WAVE) [11] [21]; and ITS application simulation to trigger the communication. In this section we list some of the most known ITS simulators. A survey on the most known simulation tools and techniques for vehicular communications and applications can be found in [23].

2.4.1 Vehicle Mobility and Networking Simulators

Vehicle mobility simulators are specialized in generating the road networks and traffic demand. At this level of simulation it is important to support: a realistic representation of traffic flow that may range from a single road junction to a whole city; the support for adding new functionality and integration with other simulation tools (e.g. an interface that allows to retrieve traffic simulation data and control the simulation using external functions). In this category of simulators there are two promising candidates: SUMO [10] and VISSIM [18]. Network simulators have the responsibility of representing the network protocols that transmit ITS information through the VANET, to a back-end or Internet service. In this category of simulators there are three promising candidates: ns-3 [15], OMNeT++ [24] and JiST/SWANS [9].

2.4.2 Integrated ITS Simulators

Integrated ITS Simulators are frameworks that couple different domain simulators in order to create a functional V2X environment. At this level of simulators it is important to support a bidirectionally-coupled simulation [22] of road traffic and network traffic (the mobility of vehicles affects communication and vice-versa). In this category of simulators there are three promising candidates: Veins [22], iTETRIS [1], and VSimRTI [20].

2.5 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* [12] and in America exists the *Security Credential Management System* [25]. 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. The second disadvantage comes in the revocation of certificates. The European PKI does not consider distribution of CRLs containing authorization tickets within the vehicular network. As a result, this solution allows a window of vulnerability where malicious vehicles have their enrollment certificate revoked but still have a pool of valid authorization tickets, which allows them to send authenticated message for the duration of that pool. 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 solution. 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 vehicular PKI.

3 Proposed Solution

Our solution implies the creation of a vehicular PKI and a simulator to test its correctness. To create this new PKI we decided to extend Multicert's product (mPKI) with new functionalities. Such functionalities will allow mPKI to support the new certificate formats; the new requests for ATs and enrollment certificates; and the creation of ATs in bundle to respond to the vehicles. mPKI is built using JEE 7 (Java enterprise Edition) and already supports the formats for the X509, CVC-Passport and CVCEis certificates (used in the Portuguese citizen card). In order to extend mPKI a Java package will be developed containing all data structures and auxiliary classes necessary for the creation of the certification authorities and certificates defined in Section 3.1. In regards to the simulator its goal will be to test the vehicle's enrollment, the requests for authorization tickets, revocation of certificates, and finally, the sending and receiving of correct and incorrect CAM and DENM messages. In this section we propose a vehicular PKI solution that issues certificates for V2X communication while preserving the privacy of its users. We start by providing an overview of the system architecture and then we provide a detailed description of the protocol.

3.1 Overview of the Architecture

Our solution is primary based on the European vehicular PKI and ETSI's standards. As such, the architecture of our PKI is similar to that of *A Generic Public Key Infrastructure for Securing Car-to-X Communication* [12] that uses the certificate and message formats standardized by ETSI [5] to secure V2X communications.

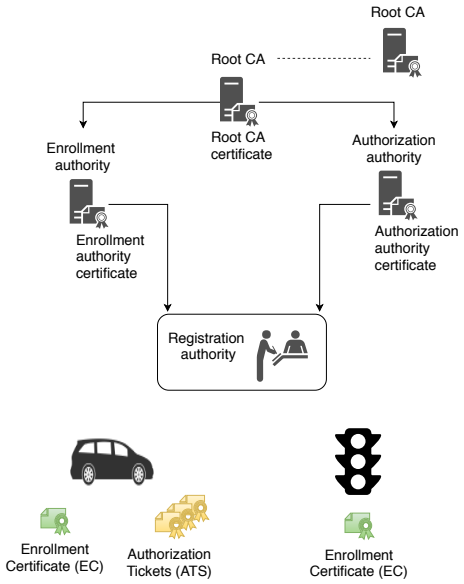

Fig. 6. Architecture of the solution and the digital certificates used by its components.

As seen in Figure 6, the components are:

- The root certification authority.
- The enrollment authority.
- The authorization authority.
- The Registration Authority (RA).

The **root CA** has a self signed root CA certificate which corresponds to the top-most certificate of the hierarchy, the private key of this certificate is used to sign the certificates of the CAs bellow in the hierarchy. The **enrollment authority** holds an enrollment authority certificate issued by the root ca. This authority issues the long-term enrollment certificate for the ITS stations. The **authorization authority** holds an authorization authority certificate issued by the root ca. This authority has the responsibility to issue short-term pseudonym authorization tickets used by the enrolled vehicles.

Finally as a new component, we added a **registration authority** to act as proxy between the end-entities and the CAs. The RA has two responsibilities: verifying the vehicle's identity and supporting their requests for enrollment certificates and authorization tickets. The later involves sending such requests to the correct CAs for certificate issuing and responding to the vehicles with the requested certificates. All of the certificates specified will be based on the formats standardized by ETSI. In order to protect the sensitive data that is exchanged between CAs and ITS stations (e.g. vehicle identifier), the connection between the different components will be done using the SSL/TLS protocol [13] to provide message freshness and privacy during communication.

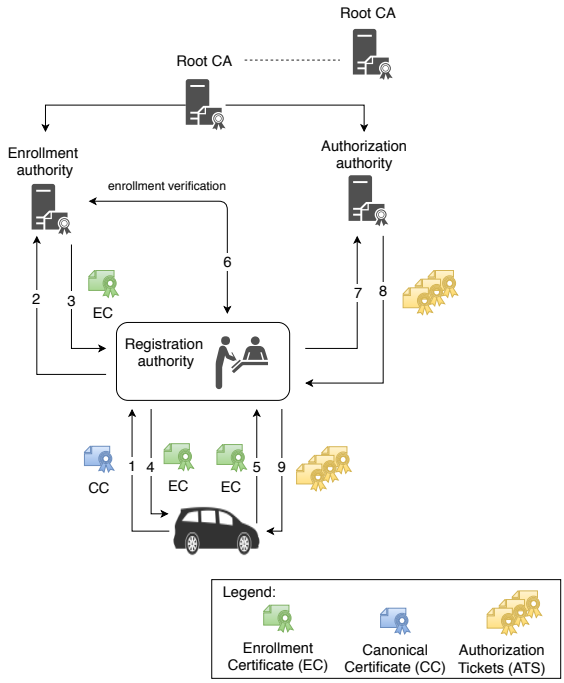

Fig. 7. Diagram of the protocol that is going to be developed.

3.2 Protocol

The interaction between an ITS station and the PKI is divided into several phases. We assume the stations' life cycle as specified in Section 2. Essentially, at manufacture vehicles are bootstrapped with a public/private key pair, an unique identifier, a canonical certificate that associates the vehicle's public key with the identifier, and the information of the trusted CAs.

With this information the vehicle can begin the **enrollment phase** to obtain the enrollment certificate. This phase comprises messages 1 to 4 represented in Figure 7. First, the vehicle sends the enrollment request to the RA. Such request is signed by the vehicle's private key and is composed of the vehicle's canonical certificate with the certification chain back to a root CA, the validity period, and other certificate related information. Upon receiving such request the RA has the responsibility of verifying the vehicle's identity by cheeking the signature and validating the canonical certificate. The first is done by decrypting the signature using the vehicle's public key (stored in the RA at vehicle manufacture). The later is done by validating the certification chain starting on the received canonical certificate up to a trusted root CA (i.e. decrypting the signature of each certificate with the public key of the signing CA). When the identity of the vehicle is confirmed the RA sends an enrollment certificate issue request to the EA. Such request is represented by message 2 in the diagram and contains the certificate related information and vehicle's public key. Upon receiving such request the EA issues the vehicle's enrollment certificate and sends it back to the RA which forwards it to the vehicle, using messages 3 and 4 respectively.

Having the enrollment certificate the vehicle can now start the **authorization phase** to obtain valid ATs. This phase comprises messages 5 to 9 from the diagram. While message 6 represents the verification of the vehicle's identity, messages 8 and 9 represent the actual certificate exchange. In message 5 the vehicle sends to the RA its enrollment certificate, the identifier of the EA that issued the enrollment certificate, and the list of public keys to be certified by the ATs. The vehicle is able to find the identifier of the EA in its enrollment certificate as standardized by ETSI 103 097 (certificate formats) [5]. To prevent the RA from linking which ATs are issued for that specific vehicle, the enrollment certificate will be encrypted with the public key of the corresponding EA. Upon receiving this request the RA forwards the encrypted enrollment certificate to the correct EA, as shown in the message 6. The EA decrypts the vehicle's enrollment certificate and verifies its validity, if it's valid then the EA sends message 6 to the RA which is a positive response. Upon receiving such response, the RA is able to request valid ATs from the AA by creating the request message 7 which contains the vehicle's public keys. The AA has the responsibility of creating ATs and sending them to the RA (message 8). Finally, the RA responds to the vehicle with the requested ATs in message 9.

Implementing this protocol in mPKI involves various steps and the development of auxiliary software. The first step is to create a Java library that contains the necessary data structures and mechanisms for creating the certificates standardized by ETSI. At this point we will be able to test if the certificates created are in conformance with the standard. Such package needs to be integrated with mPKI to enable the creation of certificates for the CAs and end-entities. The next step involves the development of the RA in a way that it serves as a proxy between the vehicles and CAs. To achieve this, we will create a web service named RAService witch exposes two services: the enrollment and the authorization of vehicles. Each service has the responsibility of accepting the vehicle's

request, forward it to the correct CA, and deliver the certificates to the original vehicle. The final step involves creating a Java simulator to test the V2X communications supported by the PKI and its certificates.

4 Evaluation

In order to test the system's behavior, a simulator will be built. This simulator will provide a controlled environment where users can test the security of the protocol and the V2X communications. Due to time constraints, our simulator will not support realistic traffic and network simulation. Instead, first we focus on delivering something simpler. The main goal of the proposed simulator is to create end-entities and test their connectivity with the proposed PKI and V2X communication.

In regards to the connectivity with the PKI, we aim to test the communication between vehicles and the RAService in the form of enrollment and authorization requests. This involves that the vehicles are able to generate keys, compose the request messages, and finally store the certificates that came in the response. This feature will allows us to verity that the format of the messages is correct, the correctness of the protocol, and test the performance and scalability of the PKI in responding to such requests with newly issued certificates.

In regards to V2X communication, we aim to verify that the messages being transmitted between vehicles and RSUs are correctly sent by the source and validated by the destination. This involves sending CAM and DENM messages which are authenticated by the ATs issued by the proposed PKI. This feature will allows us to verify that the format of the secured messages is correct, malicious messages are discarded by the vehicles, and finally to evaluate the performance and scalability overhead introduced by the message authentication and broadcasting.

To test the system's implementation, software tests will be performed at the end of the development of each of the system's components. Testing will allow us to validate that the system is being developed in conformance with the requirements previously defined in Section 1.1.

5 Scheduling of Future Work

The Future work is scheduled as follows:

- Sep 19 - October 31: Developing of the Java package that generates the ETSI certificates.
- Nov 1 - Nov 30: Integration of the Java package into mPKI.
- Dec 1 - Dec 31: Development of the RAservice.
- Jan 1 - Jan 31: Development of the simulator.
- Fev 1 - Fev 28: Practical evaluation of the protocol.
- March 1 - March 31 : Improvements to the solution from the evaluated results and writing of the dissertation.
- April 1 - May 9: writing of the dissertation.
- May 10: Deliver of the dissertation.

6 Conclusions

Vehicle-to-everything communication has the capability of improving the safety of our roads by connecting vehicles, road infrastructure and pedestrians. As a system that deals with critical information it has strong security requirements, especially privacy and message authentication. As seen in this report, we analyzed two existing PKI solutions that address such security requirements.

Having these PKI solutions and the standardization efforts in mid, we decided to implement a PKI similar to the European vehicular PKI by extending mPKI with the new CAs. The main challenges in this process will be making sure that the AA is able to issue a bundle of ATs with minimum performance overhead and to minimize the network bandwidth consumption of the response that contains such certificates.

Finally, care must be taken in implementation of the certificate formats and protocol as so to avoid security and privacy flaws. It is also important to consider modifiability in the developed software to support changes in the standards or even the possibility of adding new functionality to the PKI.

References

1. itetris homepage. http://www.ict-itetris.eu/
2. Scms cv pilots documentation. https://wiki.campllc.org/display/SCP/SCMS+CV+Pilots+Documentation
3. U.s. department of transportation. safety pilot model deployment. http://safetypilot.umtri.umich.edu/
4. 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)
5. 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)
6. European commission mobility and transport. https://ec.europa.eu/transport/road_safety/specialist/statistics_en (2016)
7. 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)
8. Etsi automotive intelligent transport systems. http://www.etsi.org/technologies-clusters/technologies/automotive-intelligent-transport (2017)
9. Barr, R., Haas, Z.J., van Renesse, R.: Jist: An efficient approach to simulation using virtual machines. Software: Practice and Experience **35**(6), 539–576 (2005)
10. Behrisch, M., Bieker, L., Erdmann, J., Krajzewicz, D.: Sumo–simulation of urban mobility: an overview. In: Proceedings of SIMUL 2011, The Third International Conference on Advances in System Simulation. ThinkMind (2011)
11. Bhoi, S.K., Khilar, P.M.: Vehicular communication: a survey. IET Networks **3**(3), 204–217 (2013)

12. Bißmeyer, N., Stübing, H., Schoch, E., Götz, S., Stotz, J.P., Lonc, B.: A generic public key infrastructure for securing car-to-x communication. In: 18th ITS World Congress, Orlando, USA. vol. 14 (2011)
13. Dierks, T.: The transport layer security (tls) protocol version 1.2 (2008)
14. Eze, E.C., Zhang, S., Liu, E.: Vehicular ad hoc networks (vanets): Current state, challenges, potentials and way forward. In: Automation and Computing (ICAC), 2014 20th International Conference on. pp. 176–181. IEEE (2014)
15. Font, J.L., Iñigo, P., Domínguez, M., Sevillano, J.L., Amaya, C.: Architecture, design and source code comparison of ns-2 and ns-3 network simulators. In: Proceedings of the 2010 Spring Simulation Multiconference. p. 109. Society for Computer Simulation International (2010)
16. Lin, Y., Wang, P., Ma, M.: 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. pp. 167–172. IEEE (2017)
17. Lonc, B., Cincilla, P.: 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. pp. 1–6. IEEE (2016)
18. Lownes, N.E., Machemehl, R.B.: Vissim: a multi-parameter sensitivity analysis. In: Proceedings of the 38th conference on Winter simulation. pp. 1406–1413. Winter Simulation Conference (2006)
19. Rehman, S., Khan, M.A., Zia, T., Zheng, L.: Vehicular ad-hoc networks (vanets)—an overview and challenges **3**, 29–38 (01 2013)
20. Schünemann, B.: V2x simulation runtime infrastructure vsimrti: An assessment tool to design smart traffic management systems. Computer Networks **55**(14), 3189–3198 (2011)
21. Siegel, J.E., Erb, D.C., Sarma, S.E.: A survey of the connected vehicle landscape–architectures, enabling technologies, applications, and development areas. IEEE Transactions on Intelligent Transportation Systems (2017)
22. Sommer, C., German, R., Dressler, F.: Bidirectionally coupled network and road traffic simulation for improved ivc analysis. IEEE Transactions on Mobile Computing **10**(1), 3–15 (2011)
23. Sommer, C., Härri, J., Hrizi, F., Schünemann, B., Dressler, F.: Simulation tools and techniques for vehicular communications and applications. In: Vehicular ad hoc Networks, pp. 365–392. Springer (2015)
24. Varga, A., Hornig, R.: An overview of the omnet++ simulation environment. In: Proceedings of the 1st international conference on Simulation tools and techniques for communications, networks and systems & workshops. p. 60. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering) (2008)
25. Whyte, W., Weimerskirch, A., Kumar, V., Hehn, T.: A security credential management system for v2v communications. In: Vehicular Networking Conference (VNC), 2013 IEEE. pp. 1–8. IEEE (2013)