Security Framework for Vehicular Applications

Matthias Gerlach*, Horst Rechner*, Tim Leinmüller**,
*Fraunhofer Institute for Open Communication Systems (FOKUS), {Matthias.Gerlach | Horst.Rechner}@fokus.fraunhofer.de,
**DaimlerChrysler AG, Group Research and Advanced Engineering, Tim.Leinmueller@DaimlerChrysler.com

***Abstract*— Vehicular ad hoc networks (VANETs) will enable new applications that increase safety and convenience of the passengers in the car. Most applications for VANETs are applications in a distributed system. They use information provided by different cars, and road side units. Different, sometimes complementary means exist to establish a trustworthy and privacy preserving system; they include certification, reputation systems, plausibility checking, and frequently changing pseudonyms.**

Often, these measures are tailored to specific aspects of the system and cannot easily be combined in an overall security architecture, nor can they easily be used by application developers. In this work, we present a framework to integrate trust and privacy services for the use in vehicular environments. The main contributions of this paper are (1) a consistent architecture for securing vehicular communications that can easily be used by application developers, (2) the principle of security sensors including a model for trust establishment for the vehicular domain and (3) the context mix model for preserving location privacy of vehicles.

I. Introduction

Vehicular communication based on wireless short-range technology enables spontaneous information exchange among vehicles and with road-side stations. This in turn facilitates a plethora of new applications for safety, traffic efficiency, and infotainment using direct or multi-hop communication at low cost. For these applications, security is mandatory and should be an integral part of the whole system.

Security threats and the corresponding security requirements in vehicular environments have been described in detail in [1] and [2]. In a nutshell, the security measures shall prevent privacy violations, denial of service attacks against the system, and the insertion of forged data into the system. As denial of service attacks are in general hard to prevent, we focus on establishing trust between the vehicles and on privacy.

A. Related Work

Currently, there are several projects concerning vehicular networks, such as *Network on Wheels* [3], *Willwarn* [4], and *GST* [5]. The *C2C-CC* [6] and *IEEE WAVE* (the 1609 suite of standards and IEEE 802.11p) represent the standardization efforts in Europe and the U.S., respectively. Concerning security in vehicular communication, the *SEVECOM* project started recently [7]. The security architecture developed by the *Vehicle Safety Communications Consortium (VSCC)* and subsequently submitted to *IEEE P1609.2* can be seen as the only approach for a security architecture in vehicular networks that is under standardization so far [8]. It defines a public-key-infrastructure (PKI)-based approach for securing messages sent in a vehicle-to-vehicle and vehicle-to-infrastructure fashion. The standard, however, does not address privacy issues, multi-hop communication, and how the network can be protected against malicious certified nodes. Gerlach introduces general concepts for a vehicular security architecture in [9]; Gerlach et al. in [10] describe a security architecture that is developed further in this paper. Work by Hubaux and Raya addresses security issues in vehicular communication, mainly in a PKI setting. In [11], they discuss attacks on vehicular networks and security requirements, propose a PKI based solution and outline open issues. In [12], the authors propose different mechanisms for certificate revocation. They also discuss privacy issues in vehicular networks. In [13], assumptions, security requirements and principles, including architectural aspects, are discussed. As it is simple to manipulate sensor information, the plausibility of information should be assessed upon reception. Golle et al. provide a framework to detect and correct false information in [14]. Leinmüller et el. research plausibility of position information in [15] and [16]. In this paper, we present an implementation framework that is able to integrate these different existing solutions for the use in demonstration scenarios and – in the long run – in field tests.

B. Outline

This paper is organized as follows. In Section II we look at the functions that have to be provided by a security system and identify different functional layers. In Sections III and IV we introduce two novel core concepts for the implementation framework, namely using sensor fusion for integrating security sensors and the context mix concept for increasing privacy. Section V outlines the implementation framework that is currently under development. Finally, Section VI concludes this paper and outlines future work.

II. Functional Architecture for a Security System

The functional layers depicted in Fig. 1 describe a decomposition of the security system into groups of use cases for a specific functionality. While the lowest layer is concerned with vehicle and application registration and identification, the higher layers are concerned with proper system operation, appropriate security measures and user privacy protection. The decomposition describes a complete view of a security solution under rather general security and application requirements.

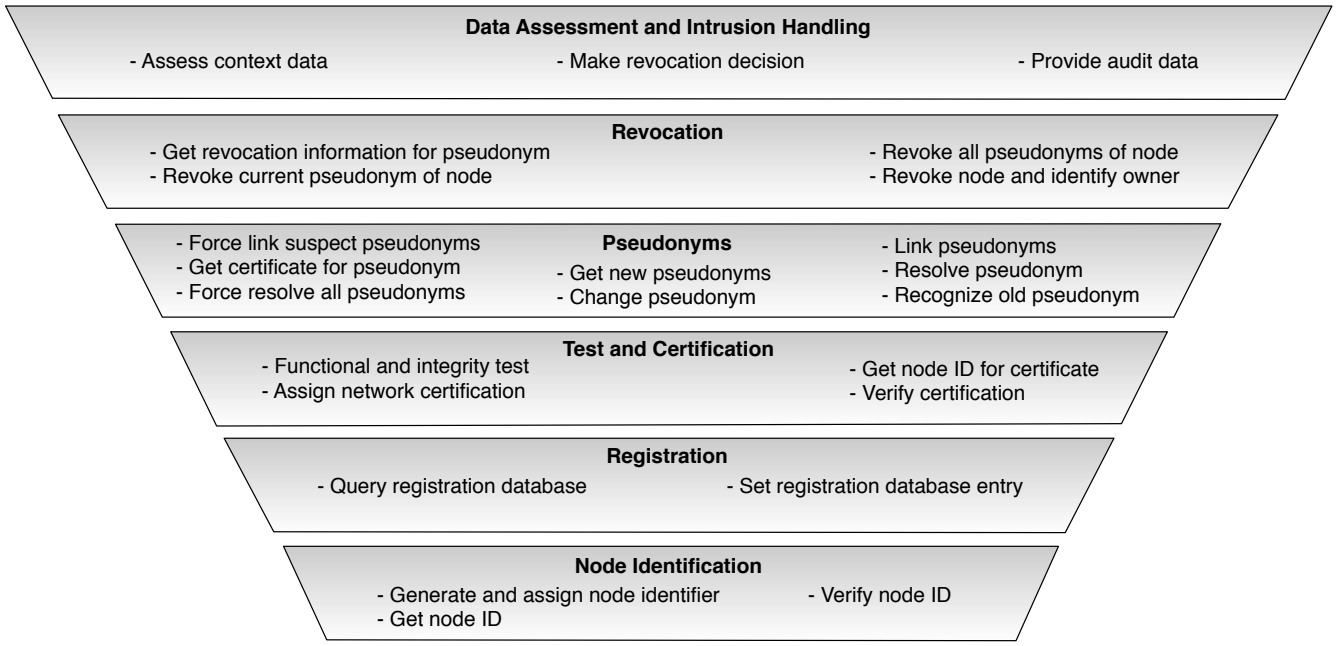

Fig. 1. Functional layers of the security architecture including use case names for the main functions in each layer

Hence, a concrete security solution may not need all components or even all layers or to be present.

The lowest layer is concerned with the *identification* of nodes, i.e., OBUs and RSUs[1]. A node is given a *node identifier* that makes it uniquely identifiable within the security system. An identifier is defined as "an object that can act as a reference to something that has an identity, " as defined by Stoneburner in [17]. An identity makes a unit globally unique. The main purpose of the identifier (and the identification system as a whole) is to be able to link a node to its owner and the vehicle it is installed in within a data base. In addition, service personnel may employ the identifier to recognize a specific type of node (e.g., those manufactured in a certain week and with a specific software version installed) or a specific node. The identifier shall not be used for communication to enable the user to remain anonymous. As outlined previously, anonymity is a key requisite in vehicular communication since a lot of privacy relevant data (e.g., Position, Speed) is distributed without protection. The identification step is similar to assigning a MAC ID to a network interface card in WiFi networks or the *international mobile equipment entity (IMEI)* number to mobile phones.

The *registration layer* is responsible for storing and retrieving data for that particular unit. This includes the owner of the unit, the vehicle it is installed in, and more. The registration process is similar to the step of getting a license plate. Registration data can be distributed among different databases. With respect to the security system, the registration of the owner with the unit is the most relevant step. This may be done in a dedicated database or – if the node is a fixed part of a vehicle – be based on the vehicle registration database backed with the information which unit is part of which vehicle (which may be registered in a separate database owned by, e.g., the vehicle manufacturer). While the owner is the legal entity that is responsible for the vehicle, it may not necessarily be the person that is driving the car. This is current practice in registration of cars with insurance companies and authorities.

The *test and certification layer* is responsible for assessing the correctness of operation of a node. This process is meant to ensure that only nodes with verified properties may actively participate in the communication. One or several digital certificates issued by the testing authority vouch for the correct operation of the node. In addition, different roles may be assigned to a node. Certificates in the certification layer shall not be used for the communication. The test and certification process is meant for the detection of defect systems and to prevent unauthorized insertion of data into the network. It is a means to control the fulfillment of requirements with respect to the performance, behavior and reliability of a system. However, manipulation is still possible after a system has been checked.

The *pseudonym layer* provides a basic level of anonymity by introducing the possibility to use changing pseudonyms that cannot be linked by unauthorized parties (a) to the vehicle, (b) to the acquirer and (c) among each other. Pseudonyms shall express the same roles as the certificate issued for the node such as being a police car. They are used for the communication system and are equivalent to a certified MAC/IP address that is bound to a cryptographic key. Changing pseudonyms provides a fair amount of privacy against an outside attacker while allowing legitimate users to link, resolve and recognize

[1]OBU – On Board Unit, RSU – Road Side Unit

pseudonyms (see Fig. 1). Privacy provision of the system can be important even to meet the regulatory requirements of certain countries. The requirement for escrow depends on the impact of failing security on the system users. Clearly, if life or the functionality of the whole transportation system are at stake, quick node revocation is more important than if failing security only results in a couple of false messages.

The *revocation layer* is concerned with excluding nodes from the system. It contains a database of revoked pseudonyms and distributes this data to all nodes in the system if necessary, depending on the scale of the revocation decision. The scale can range from only node-local to system-wide revocation. A reaction to detected attacks carried out by a node is to exclude this node from the system. Other reasons not directly owed to system operation, such as a stolen unit or prevention of criminal activity may also require a revocation service.

The *data assessment and intrusion handling layer* is responsible for assessing data, auditing them and detecting and handling misbehavior. Misbehavior and faulty nodes can sometimes not be distinguished, we use the word misbehavior to also include faulty nodes. The decision to ignore data or to initiate the revocation process is taken in this layer. If revocation of nodes is desired, an authority and appropriate mechanisms must exist to decide if a node must be revoked. In large networks, where automatic detection and reaction is necessary, this layer is particularly important. Besides system-wide detection of malicious and false data, node-local detection and reaction is necessary to minimize the impact of malicious or malfunctioning nodes. This functionality is crucial, as it can be assumed to be the common case and happen more often than crossing a node that has been revoked.

A. Discussion

As mentioned above, not all layers of the stack are mandatory. Under some circumstances some layers may even need to be explicitly excluded. The necessity for the test and certification layer, for the pseudonym layer and for the data assessment and intrusion handling layer is undisputed. These layers provide both trust services (by certification and data assessment, e.g. plausibility) and privacy, as is necessary and feasible in vehicular environments. Concerning identification and registration of vehicles, the implementation of such a service will enable the stakeholders of this service to infringe the users privacy. The benefit of registration may well be included in other systems than the vehicular communication system, such that users have a choice to use the system. In addition, setting up such a service is cost intensive and may better be combined with a business model in a different domain. Hence it may be prudent to explicitly state that the registration service has no link to the vehicular communication system. This would require anonymous testing of vehicles. Typically, any PKI based solution needs a revocation service. The feasibility of revocation in vehicular environments depends on the correct identification of malicious nodes, the scalability of the revocation system (for a more detailed discussion, see for example [12] and [18]), and willingness to take the financial burden of maintaining the infrastructure required for this. The presented framework allows for including or leaving out the discussed layers by being flexible in the support of different trust establishment algorithms that require or do not require a revocation layer.

III. Trust and Confidence – Security Sensors

Building on the sociological trust model described in [19], in our setting, the truster is an application and the trustee would be any entities providing context information. As entities could be nearby vehicles, on-board sensors, road side units and the like, we distinguish four major classes of informations that could form a trust relation in vehicular networks:

1) Raw sensor information, which could either be provided by on-board sensors or by means of a communication service from another nearby vehicles.
2) Higher level sensor information, i.e., information that are aggregates of several pieces of raw sensor information.
3) Services, such as a communication service.
4) Attributes, such as being a vehicle, being a police car, treating information confidential, and the like. Attributes are often modelled using certificates. Note that we do not make a difference between attribute certificates and identity certificates here.

Despite the common restriction of trust to services (c.f. [20], [21], [22]), we apply the term also to attributes and assertions. We argue that both services and assertions can be seen as attributes in a trust relationship even though the algorithms to establish trust may differ[2].

On the local system, trust must be established on incoming sensor data. The next sections describe how we represent trust internally, and how the different trust values can be included in the application logic using sensor fusion.

A. Trust and Confidence Values

As stated in [23], there are different possibilities to express trust. Examples can be found in PGP, where a public key can be tagged with the discrete values for *unknown*, *untrusted*, *marginally trusted*, or *completely trusted* [24]. Marsh proposes to use values in the interval $[-1, 1)$ where -1 expresses complete distrust and 1 represents "blind trust". Marsh argues that blind trust should not be used, therefore he excludes the use of the value 1 from the trust valuation. Even though he argues that representing distrust is a valuable feature of his formalism, he points out that it limits the use of operators on the values, and exhibits problems at extreme values and zero. Gambetta in [25] proposes to describe trust as a value in the interval $[0, 1]$, where 0 represents complete distrust and 1 represents complete trust. Similarly, Golle et al. define "validity" as a value in this interval [14]. In our opinion, validity represents the same information as a trust value. Finally, Mui et al. formalize trust as the conditional expectation of the reputation of the trustee given his prior encounters with the trustee. As reputation is

[2]They may include reputation systems, plausibility checks and certificate verification, for example

modelled as a value in the interval $[0, 1]$, the expected value, and hence trust, will be in the same range.

For our system, based on the discussion above, we propose to model trust or confidence as a value in the interval $[0, 1]$. These values represent the high-level security sensor information discussed in the following section.

B. Security Sensor Fusion

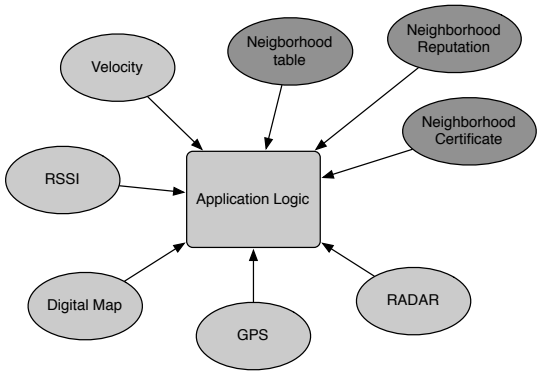

Fig. 2. Interpreting the communication system as additional sensors

Applications that use environmental data to react, such as active safety applications in vehicular environments, deal with a plethora of different sensors. They use them to create a world model that is employed to take certain actions, e.g., informing the driver about an imminent collision. Adding a communication interface yields more environmental data sources, which can also be modeled as sensors (depicted dark grey in Figure 2). The communication interface can transmit any sensor readings from a remote source to the local node.

Sensor fusion is a common approach to combine different sensor readings in order to yield a better view of a node's environment (see e.g., [26]). Sensor fusion is commonly used in robotics, military applications and transportation to improve the perception of nodes about their environment. It refers to combining the signals from different sensors in a new – fused – signal. The ultimate goal of sensor fusion techniques is to generate a complete model of the world inside the node that can be used for a decision making process (be it avoiding obstacles or finding the perfect route to a destination).

Borrowing from the concepts of sensor fusion, sensor reasoning done in an application as depicted in Figure 2 can tightly be integrated with a security solution. Interpreting security mechanisms as "security sensors" yields a new class of sensors that – similar to other sensors – can contribute their view on the world. It is then up to the application developer to take security sensors into account as needed.

Sensor fusion can be applied on different levels. While security sensors assessment of data is a rather high-level contribution (it already has a semantic), low-level sensor fusion techniques can be used to increase the security of data as well. E.g., comparing the given coordinates of neighbors with the features extracted using RADAR may yield an assertion of the correctness of the sent data. Low-level sensor fusion is no new concept and therefore not discussed further in this paper.

The high-level semantics of security sensors is modeled as a probability. This probability can be interpreted as the level of trust, the system assigns to a certain value. This semantics is similar for different types of security sensors. For example, in networks of trust, the trust in the authenticity of a certificate can be expressed as a probability derived from the product of the trustworthiness of the nodes in the certification chain. Similarly, many reputation systems output the reputation of a node in the interval between 0 and 1. In addition, probabilistic (Bayesian) reasoning is a well-understood and often employed technique. Note that, in order to fully utilize this approach, it is necessary to create a sensor model for each security sensor that can be used to estimate the effectiveness and significance of a security sensor reading.

The advantage of interpreting the security system as consisting of "security sensors" is obvious: first, reasoning with security sensors is not different from using other sensors, and hence intuitive. Second, the modular approach implied leaves room for upgradeability of the sensors and underlying sensor models. Last but not least, this approach leaves each application developer the choice to include or not security sensors in the decisions of his application.

C. Discussion

The main idea described in this section is to enhance context information, i.e., sensor readings, with trust values that are modelled as additional sensors. Our approach is unique in that it combines different trust measures and gives application developers the means to combine them.

Representing trust as probabilities yields some interesting properties: first it gives us a means to compare and combine different trust values. Second, it provides an easy to understand interface to a security system for vehicular ad hoc networks for application developers. Third, calculating with probability yields interesting assertions about the system. What is important to decide is to set the thresholds for the application to act on the indicated values. This decision must also be based on the security sensor model that needs to be provided with the different mechanisms integrated into the system.

IV. Context Mixes for Increased Privacy

Besides establishing trust in incoming sensor data and representing it to applications for further processing, privacy must not be neglected. Changing pseudonyms is the standard technique for protecting the location privacy of the users in mobile environments. It anonymizes the users by obfuscating her name (using the pseudonym) and the location by changing pseudonyms, making the vehicle untraceable.

However, simply changing a pseudonym is not sufficient. This fact can be confirmed based on the simulations and analyzes carried out by Sampigethaya et al. in [27]: the authors state, that under "correlation tracking", an attack that uses physical parameters and constraints, changing the pseudonyms at arbitrary intervals yields an anonymity set below 2, and

hence no anonymity at all [27, Figure 6]. In addition, it is important that the pseudonym change must be carried out at the same time by all identifiable entities on the local system, including all addresses (i.e., IP, MAC address).

We propose to include the use of context information (such as the number of neighbors, their direction and speed) for initiating a pseudonym change. Like this, nodes cooperatively identify good opportunities to blend in a number of vehicles and hence increase their anonymity. Following the terms *mix-zones* (Beresford) and *mix-nets* (Chaum) we call these situations *mix-contexts*.

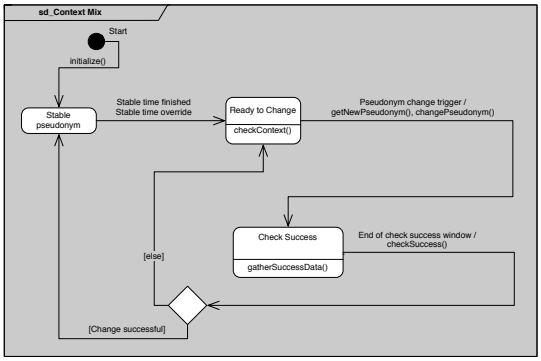

Fig. 3. General algorithm for pseudonym change using context mixes

Fig. 3 depicts a general state diagram of a pseudonym change algorithm. The minimal stable time may be configured to account for the application requirement of a stable communication session. After the stable time finishes, the node waits for the trigger to change its pseudonym, checks if the change has been successful and then enters the next period of stable pseudonym to run through the process again.

After initialization, the system enters the pseudonym cycle and waits for expiry of the stable time interval. Under certain circumstances, a pseudonym change may be sensible before the stable time is over; in this case the stable time is overridden. The system is then ready to change its pseudonym, and in this state permanently assesses its context (i.e., neighborhood information) in search for a mix context that suffices the target level of anonymity. If this mix context is eventually found, a new pseudonym is retrieved and set. Simply put, the target level of anonymity can be a certain number of nodes with similar direction within a certain range. After changing the pseudonym, the system assesses whether the change was successful (i.e., if enough similar nodes changed their pseudonym at the same time) or not in order to start the whole process again, or try to change the pseudonym again, respectively.

A. Pseudonym Change Triggers - Mix Contexts

Dey defines context as "(...) any information that can be used to characterize the situation of an entity (...)" [28]. Using this definition, a *mix context* is defined as any situation that provides sufficient anonymity with respect to an attacker to change a pseudonym. Depending on the desired level of protection, this may simply be the number of nodes in the neighborhood irrespective of their properties, or the nodes with similar properties, such that they would be indistinguishable for an attacker. A pseudonym change algorithm using mix contexts is a *context mix*. A context mix provides unlinkability between pseudonyms after a change.

A mix context shall provide sufficient anonymity to a node changing its pseudonym. This requires that the neighborhood of the node and the general situation must be such that the entropy of the situation *after the change* is sufficiently high. Hence, a node must permanently assess its context according to the expected entropy if it changes its pseudonym. The expected entropy also depends on the attacker; this implies that every node may need to implement a reference attacker to estimate its level of privacy. Currently, we define the availability of more than N nodes in a defined area as *mix context*. In addition to simply changing the pseudonym in the right context, we define a *minimal stable time* where the node is supposed not to change its pseudonyms. This is important in order to prevent frequently terminated connections, and it bounds the number pseudonyms used per node.

B. Discussion

Simulations about the effectiveness of mix contexts show that they improve location privacy in vehicular environments [29]. On the other hand, a couple of issues have to be taken in mind: first, increasing the minimum stable time decreases the probability to meet a node changing its pseudonym. Therefore we introduced a *change ready flag* that is broadcast by a node where the minimal stable time expired. Thus, when two nodes with this flag set meet, the probability that they will change their pseudonym at the same time increases. Second, if different nodes take different context information into account, they will change their pseudonyms in different situations. In addition, the more context information is considered, the fewer situations will occur where a node changes its pseudonym. Thus, it may be important that pseudonym change algorithms are the same for all nodes in the network. Third, the parameters for the algorithms need to be refined in order to optimize the privacy provisions. In particular, *minimum stable time* will need to be adjusted to realistic values and its impact examined. Finally, the applicability of the algorithm in real life scenarios still has to be proved. This includes estimating a sensible minimal stable time, including data about when the vehicle is started, and the like.

V. Implementation Framework

A. Overview

Figure 4 depicts the components of a security system. The interactions between the components are indicated by straight lines with arrows leaving from the initiating components. Each interaction point is assigned a letter for easy reference[3]. The figure contains three classes of components:

[3]The interface letters will not be used in this paper

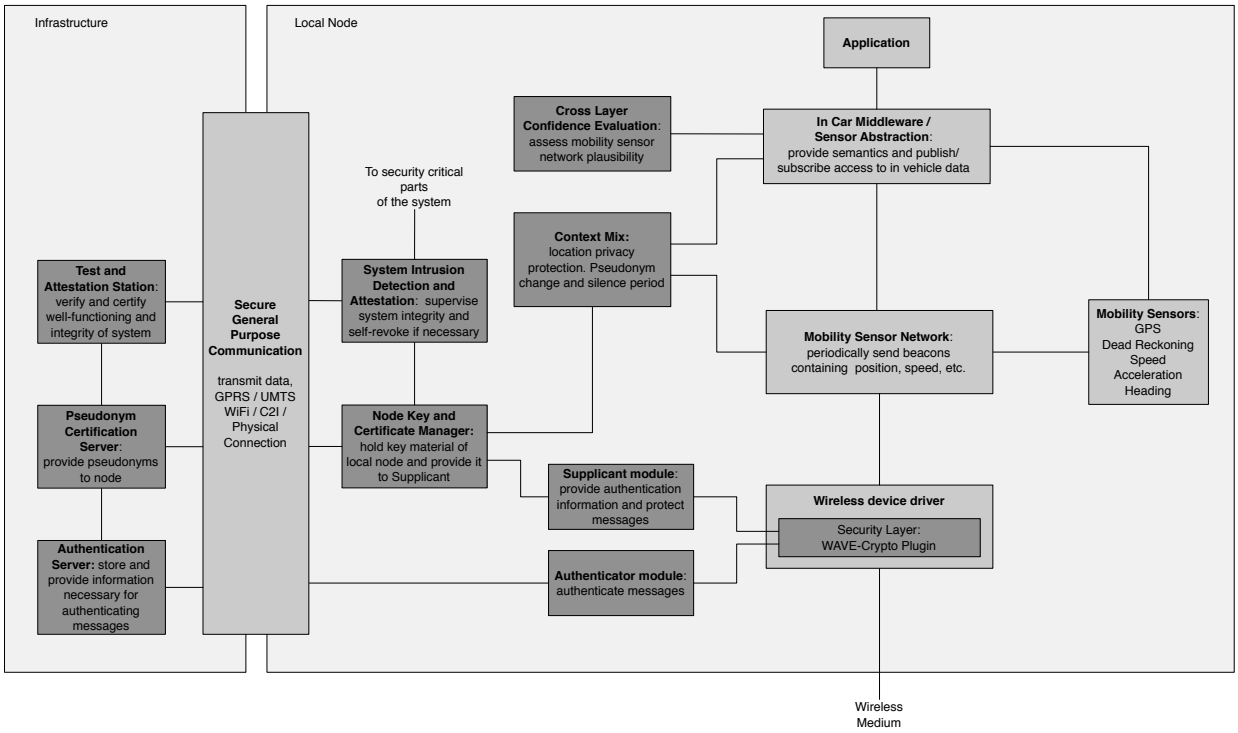

Fig. 4. Components of the security system. Security relevant parts in dark grey

- Operating platform components, i.e., components that are not security related but represent the major parts of the execution environment for vehicular applications,
- Node local security components, i.e., the components that, as part of the local system control the security parts, and
- Infrastructure security components, that represent the security infrastructure necessary for the security system.

We will briefly describe those components in the following sections.

B. Operating Platform Components

The components of the operating platform are:

- Wireless device driver
- Mobility sensor network
- Mobility sensors
- In car middleware
- Application

The *wireless device driver* is responsible for receiving and sending messages over the wireless channel. Currently, ordinary WiFi technology is used. In the future, 802.11p [30] may be used, as it provides more robust access in vehicular environments. Within the 802.11 stack, a security layer is situated, that is responsible for adding and stripping of security payload from the packets. A *mobility sensor network* is a network of mobile nodes that exchange information about their kinematic state. The component in the stack is responsible for sending out and receiving position information from other vehicles. It is the basis for creating an up-to-date neighborhood database. *Mobility sensors* are those sensors that indicate the kinematic state of the vehicle. These are position, speed, heading, and acceleration. Actual sensors can be GPS, dead reckoning, accelerometers, and the like. The mobility sensors have different accuracy. The *in car middleware* provides an abstraction to the different sensor data available in the vehicle. This is important, since data sources provide different data in vehicles of different brands or series. Within the application environment, the middleware typically provides publish/subscribe access to the sensor data available on the vehicle for the use in applications. Finally, the *application* uses the in car middleware to take decisions based on the received data and sensor readings. With respect to security, the different security mechanisms within the vehicle can be included just as another sensor indicating the confidence in a specific sensor value.

C. Node-local Security Components

Mostly on the left hand side of Figure 4, the security relevant components can be found highlighted in dark grey. These are:

- Authenticator module
- Supplicant module
- Node key and certificate manager
- System intrusion detection and attestation
- Context mix
- Cross layer confidence evaluation
- Secure general purpose communication

The *authenticator* and supplicant modules have been named according to the convention in 802.1X [31]. The authenticator is responsible for authenticating messages and stripping off security headers and trailers, i.e., *decapsulating* incoming messages. This may include the verification of certificate chains. The authenticator should report a confidence value and its algorithm identifier back to the security layer, such that it can be passed upwards with the message. As described in Section III, the confidence value indicates the trust represented by a certain certificate or – more general – algorithm executed in the authenticator. The *supplicant* is responsible for *encapsulating* outgoing messages, i.e., adding security information and carrying out the appropriate algorithms to secure a message. For the pseudonym system, this corresponds to attaching the pseudonym, i.e., an anonymous certificate, to the given message and sign it with the appropriate key. The certificate in the pseudonym authorizes the message to be sent within the vehicular environment. The *node key and certificate manager* is responsible for holding and protecting the key material for the node. It holds the private key corresponding to the pseudonyms that this node is allowed to use. The node key and certificate manager interacts with the infrastructure (the pseudonym distribution server) to obtain new pseudonyms (certificates). Pseudonyms are represented by WAVE certificates [8] that are currently specified in a trial use standard. *System intrusion detection and attestation* is able to detect tampering on the hardware, sensors or software modules of the vehicle. If the local system detects jumps in the input, unauthorized changes in the software or hardware configuration, the pseudonyms are disabled and cannot be used by the supplicant anymore. The functionality is similar to the attestation feature of the trusted platform module (see [32]). The *context mix* module is responsible for managing pseudonym use and changing the pseudonym of the vehicle intelligently, according to the algorithm outlined in Section IV. Based on the information about neighboring nodes, the context mix triggers pseudonym change and – if necessary – a silence period. In line with the discussion in Section III, *cross layer confidence evaluation* adds confidence evaluation as additional "security sensor" readings to the information provided by the mobility sensor network. This information is typically provided by indicating a confidence value together with the algorithm type. As an example, take a vehicle equipped with radar sensors for scanning the vicinity of the car and a GPS receiver. When receiving the position of a new car via a network beacon, this value is published in the middleware. Additional information such as a certified and valid signature with the message received from that car would be attached to this information as a confidence value with the algorithm type $CONF\_ALGO\_CERT\_CA$. Additional local sensor information, such as the local radar image indicating that the broadcasted position is indeed matching a car would be attached as another confidence value with type $CONF\_ALGO\_RADAR$. With both values – the car's position and the corresponding confidence values along with the algorithm – applications can assess the data according to the confidence values relevant for them. A module that has its place in both the infrastructure and the local node is the *general purpose communication module*. It provides a channel that can be used to obtain new pseudonyms, do a remote attestation of a node, and retrieve up to date authentication information such as revocation lists and root certificates. Appropriate protocols will have to be developed or selected for these purposes.

D. Security Infrastructure Components

The security infrastructure consist of:

- Authentication Server
- Pseudonym distribution system
- Trust and attestation station
- Secure general purpose communication

The *authentication server*, has similar duties as those defined in the IEEE 802.1X standard [31]. It is responsible for managing and distributing authentication information within the vehicular network. In particular, this includes distribution of root certificates and certificate revocation lists. The service of the authentication service should be implemented both as push and pull service, to allow for the dynamic adaption to the networking environment. The *Pseudonym distribution system* certifies pseudonyms of nodes or provides those to the nodes themselves. Different possibilities to implement such a pseudonym distribution system exist: these range from remotely providing fresh pseudonyms upon turning the ignition key to a pre-installed set of pseudonyms for a longer period of time. The *trust and attestation station* is responsible for testing a node using the appropriate procedures. This shall ensure that only well-behaving nodes are allowed to be part of the network. For admitting a node to the network, its on board system and sensors as well as the installed software has to be tested. This may require certified software, a secure execution environment that support restricting the software that can be run on the system and appropriate system intrusion detection mechanisms that have to be present on the local system. Finally, as described above, *secure general purpose communication* on the infrastructure side connects the security infrastructure with the node local system.

E. Discussion

We argue that the modules described in this section cover the major components necessary for implementing a security framework for vehicular environments. The focus on the mobility sensor network does not restrict the overall architecture in terms of supported additional sensors but for a first implementation covers the main feature of networking in the vehicular environment – the position and speeds of surrounding vehicles.

VI. Conclusion and Future Work

In this paper, we present a consistent implementation framework for security in vehicular environments. The main contributions were first the description of confidence valuation as the major concept for assessing the security of incoming data;

confidence assessments can be integrated into the application logic using sensor fusion techniques. Second the presentation of a concept for increasing the location privacy in vehicular communication called context mix, where the pseudonym of a vehicle is changed only if sufficient anonymity can be expected. This type of pseudonym change results in better location privacy for the users, and will contribute to the better acceptance of vehicular communication. The third contribution is the presentation of the components of an implementation framework that integrates the above solutions. This framework is currently being specified and will enable the integration and test of different trust establishment mechanisms; in addition, the implementation will demonstrate a feasible setup for a security solution in field-test scenarios. Future work is to refine the component model and implement selected parts for a complete security demonstration platform.

ACKNOWLEDGEMENTS

This work has been carried out in the *NOW – Network on Wheels* [3] project supported by the German Ministry for Education and Research under contract No. 01AK064.

The authors would like to thank Bernd Bochow, Gabriele Goldacker, and Felix Güttler (FhI FOKUS), Björn Schünemann (Hasso-Plattner-Institut, Potsdam) and Andreas Festag (NEC Deutschland GmbH) for their comments and valuable contributions.

REFERENCES

[1] A. Aijaz, B. Bochow, F. Dötzer, A. Festag, M. Gerlach, R. Kroh, and T. Leinmüller, "Attacks on Inter Vehicle Communication Systems - an Analysis," The Network on Wheels Project, Tech. Rep., 2005. [Online]. Available: http://www.network-on-wheels.de/documents.html

[2] M. Gerlach, "Assessing and Improving Privacy in VANETs," in *Proceedings of Fourth Workshop on Embedded Security in Cars (ESCAR)*, November 2006.

[3] The Network on Wheels (NOW) Project, "NOW website," 2004. [Online]. Available: http://www.network-on-wheels.de

[4] The Willwarn Project, "The Willwarn Project Website," 2005. [Online]. Available: http://www.prevent-ip.org/willwarn

[5] The Global Systems of Telematics (GST) Project, "GST website," 2005. [Online]. Available: http://www.gstproject.org

[6] The Car-to-Car Communication Consortium (C2C-CC), "C2C-CC website," 2006. [Online]. Available: http://www.car-to-car.org

[7] The Secure Vehicle Communication (SeVeCOM) Project, "SeVeCOM home," 2006. [Online]. Available: http://www.sevecom.org

[8] "IEEE Trial-Use Standard for Wireless Access in Vehicular Environments - Security Services for Applications and Management Messages," 2006. [Online]. Available: http://ieeexplore.ieee.org/servlet/opac?punumber=11000

[9] M. Gerlach, "VaneSe - An approach to VANET security," in *Proceedings of First International Workshop on Vehicle-to-Vehicle Communications (V2VCOM)*, O. Altintas and W. Chen, Eds., San Diego, California, USA, July 2005.

[10] M. Gerlach, A. Festag, T. Leinmüller, G. Goldacker, and C. Harsch, "Security Architecture for Vehicular Communication," in *Proceedings of Fourth Workshop on Intelligent Transportation Systems (WIT)*, Hamburg, Germany, March 2007.

[11] M. Raya, P. Papadimitratos, and J.-P. Hubaux, "Securing Vehicular Communications," *IEEE Wireless Communications Magazine, Special Issue on Inter-Vehicular Communications*, October 2006.

[12] M. Raya, D. Jungels, P. Papadimitratos, I. Aad, and J.-P. Hubaux, "Certificate revocation in vehicular networks," Laboratory for Computer Communications and Applications (LCA), EPFL, Tech. Rep. LCA-REPORT-2006-006, 2006.

[13] P. Papadimitratos, V. Gligor, and J.-P. Hubaux, "Securing Vehicular Communications - Assumptions, Requirements, and Principles," in *Proceedings of Fourth Workshop on Embedded Security in Cars (ESCAR)*, 2006.

[14] P. Golle, D. Greene, and J. Staddon, "Detecting and Correcting malicious data in VANETs," in *Proceedings of the first ACM workshop on Vehicular ad hoc networks (VANET)*, October 2004, pp. 29–37.

[15] T. Leinmüller, C. Maihöfer, E. Schoch, and F. Kargl, "Improved Security in Geographic Ad Hoc Routing through Autonomous Position Verification," in *Proceedings of VANET 2006*, September 2006.

[16] T. Leinmüller, E. Schoch, and F. Kargl, "Position Verification Approaches for Vehicular Ad Hoc Networks," *IEEE Wireless Communications Magazine*, October 2006.

[17] G. Stoneburner, "Underlying Technical Models for Information Technology Security," National Institute of Standards and Technology (NIST), Tech. Rep. NIST Special Publication 800-33, December 2001. [Online]. Available: http://csrc.nist.gov/publications/nistpubs/800-33/sp800-33.pdf

[18] T. Leinmüller, E. Schoch, and C. Maihöfer, "Security Issues and Solution Concepts in Vehicular Ad Hoc Networks," in *Proceedings of the Fourth Annual Conference on Wireless On demand Network Systems and Services (WONS 2007)*, Obergurgl, Austria, January 2007.

[19] D. H. McKnight and N. L. Chervany, "The meanings of trust," MISRC Working Paper, University of Minnesota - Management Information Systems Research Center, 1996, http://misrc.umn.edu/wpaper/wp96-04.htm. [Online]. Available: http://misrc.umn.edu/wpaper/wp96-04.htm

[20] S. Marsh, "Formalising trust as a computational concept," Ph.D. dissertation, University of Stirling, Department of Computing Science and Mathematics, 1994.

[21] L. Mui, M. Mohtashemi, and A. Halberstadt, "A computational model of trust and reputation," in *Proceedings of the 35th Hawaii International Conference on System Sciences*, 2002.

[22] P. Michiardi and R. Molva, "Core: a collaborative reputation mechanism to enforce node cooperation in mobile ad hoc networks," in *Proceedings of the IFIP TC6/TC11 Sixth Joint Working Conference on Communications and Multimedia Security*. Deventer, The Netherlands, The Netherlands: Kluwer, B.V., 2002, pp. 107–121.

[23] R. Kohlas and U. Maurer, "Confidence valuation in a public-key infrastructure based on uncertain evidence," in *Proceedings of Public Key Cryptography*, ser. LNCS, H, Ed., 2000, pp. 93–112.

[24] P. Zimmermann, "Pgp user's guide part 1: Essential topics," Phil's Pretty Good Software, Tech. Rep., 1994.

[25] D. Gambetta, *Can we Trust Trust?* Department of Sociology, University of Oxford, 2000, ch. 13, pp. 213–237.

[26] R. C. Luo, C.-C. Yih, and K. L. Su, "Multisensor Fusion and Integration: Approaches, Applications, and Future Research Directions," *IEEE Sensors Journal*, vol. 2, no. 2, pp. 107–119, April 2002.

[27] L. Huang, K. Sampigethaya, , K. Matsuura, R. Poovendran, K. Sezaki, and M. L, "Caravan: Providing location privacy for VANET," in *Proceedings of Escar 2005*, 2005.

[28] A. K. Dey and G. D. Abowd, "Towards a better understanding of context and context-awareness," in *Proceedings of 1st International Symposium on Handheld and Ubiquitous Computing*, 1999, pp. 304–307.

[29] M. Gerlach and F. Güttler, "Privacy in VANETs using Changing Pseudonyms - Ideal and Real (Poster Presentation)," in *Proceedings of 65th Vehicular Technology Conference VTC2007-Spring*, Dublin, Ireland, April 2007.

[30] IEEE, "IEEE standard 802.11p. draft amendment: Wireless access in vehicular environments (WAVE)," 2004, draft 1.0.

[31] IEEE, "IEEE standard 802.1X," 2001.

[32] Trusted Computing Platform Alliance (TCPA), "TCPA website," 2005. [Online]. Available: http://www.trustedcomputing.org