SECURITY ARCHITECURE FOR INFRASTRUTURE BASED VEHICULAR COMMUNICATIONS

Corneliu Mihail ALEXANDRESCU[1], Ştefan Gabriel ŞORIGA[2]

În rețelele VANET sunt necesare mecanisme de protejare a mesajelor şi identificare a vehiculelor sau utilizatorilor valizi. În acest sens, în această lucrare propunem o arhitectură de securitate care oferă o soluție completă şi practică, o soluție care poate fi rapid adoptată şi implementată. Obiectivul nostru este să proiectăm o arhitectură de referință care, pe de o parte, asigură un nivel suficient de protecție al utilizatorilor, şi, pe de altă parte, este practică. Soluția noastră se bazează pe primitive criptografice binecunoscute, deja testate şi studiate. În acelaşi timp, soluția noastră permite dezvoltarea şi îmbunătățirea ulterioară a sistemelor.

Vehicular Ad Hoc Networks (VANETs) require some mechanism to help authenticate messages and identify valid vehicles and users. We address these problems having developed a security architecture that provides a comprehensive and practical solution, a solution that can be quickly adopted and deployed. Our objective is to design a baseline architecture, which, on the one hand, provides a sufficient level of protection for users, and, on the other hand, is practical and deployable. Our solution relies on well-established and understood cryptographic primitives, which are already broadly implemented and scrutinized. At the same time, our solution allows deployed systems to be tuned or augmented, in order to meet more stringent future requirements.

Keywords: VANET security, IEEE 1609.2

1. Introduction

Vehicular ad hoc networks aim at enhancing safety and efficiency in transportation systems. In VANETs, vehicles and road-side infrastructure units (RSUs) are equipped with sensors and wireless communication devices, allowing vehicles to sense traffic and road conditions, and warn other nearby vehicles about potential emergency situations and traffic jams. Besides road safety applications, the advent of vehicular communications also opens up opportunities for travelling comfort applications, such as internet access from a car. In conclusion, vehicular communications offer a rich set of tools to drivers and road administrators. But, at the same time, they make possible a large set of abuses and attacks. Especially the

[1] Professor, Depart. of Telematics and Electronics for Transports, University POLITEHNICA of Bucharest, Romania, e-mail: cma@upb.ro

[2] Assist., Depart. of Telematics and Electronics for Transports, University POLITEHNICA of Bucharest, Romania, e-mail: stefan.soriga@upb.ro

wireless communication technology increases the possibilities for attackers, since the communication medium cannot provide any physical security. This leads to many different passive and active attack possibilities [1]. Consider, for example, nodes that inject false information or track the location and transactions of vehicles and infer sensitive information about their drivers. These kinds of exploits indicate that security and privacy-enhancing mechanisms are a prerequisite for deployment. Ideally, these security mechanisms should provide the following desirable properties [2]:

Authenticity. A vehicle needs to authenticate other legitimate vehicles, and messages sent out by other legitimate vehicles. In addition, the recipient, vehicle or RSU, should be able to verify that the message has not been tampered with in transit.

Privacy. RSUs and casual observers should not be able to track down a driver's trajectory in the long term. The privacy requirement is seemingly contradictory to the authenticity requirement: suppose a vehicle presents the certificate to an RSU in one location, and later presents the same certificate to another RSU in a different location. Then if these two RSUs compare the information that they have collected, they can easily learn that the owner of the certificate has traveled from one location to another.

Short-term Linkability. For privacy, an eavesdropper should not be able to link messages in the long-term. However, some VANET applications require that in the short-term a recipient be able to link two messages sent out by the same vehicle. Short-term linkability does not violate drivers' privacy, because vehicles mobility pattern is constrained. If a vehicle is detected at some location X at time t, then at $t + \tau$ (where τ represents a small time increment), the vehicle must be in the vicinity of location X. Therefore, being able to track a vehicle in the short-term does not impact users' privacy.

Traceability and Revocation. An authority should be able to trace a vehicle that abuses the VANET. In addition, once a misbehaving vehicle has been traced, the authority should be able to revoke it in a timely manner. This prevents any further damage that the misbehaving vehicle might cause to the VANET.

Efficiency. To make VANETs economically viable, the on-board processing and communications units (OBUs) have resource-limited processors. Therefore, the cryptography used in VANET should not incur heavy computational overhead.

3. Related work

In the recent years, the security problems has been well understood: at least three concerted efforts, the IEEE 1609.2 working group [1], the NoW project [3], and the SeVeCom project [4], [5], [6], were developed VANET security

architectures. Their common basic elements include the use of certification authorities and public key cryptography to protect V2V and V2I messages. Message authentication, integrity, and nonrepudiation, as well as protection of private user information are identified as primary requirements. Besides, also [7] and [2] have discussed security challenges in vehicular network and give potential solutions.

4. IEEE 1609.2 Security

The IEEE 1609 WAVE communication standards enhance 802.11 to support wireless communications among vehicles and the roadside infrastructure. The IEEE 1609.2 standard proposes an infrastructure based on public key cryptography. In vehicular public key infrastructure (PKI) a certificate authority (CA) certifies the public/private key pair of the vehicle or RSU. A public key certificate links the public key to its owner's identity, which is certified and issued by a CA. Because a single, centralized certificate authority doesn't scale, the CAs must be organized in a hierarchical manner for effective management.

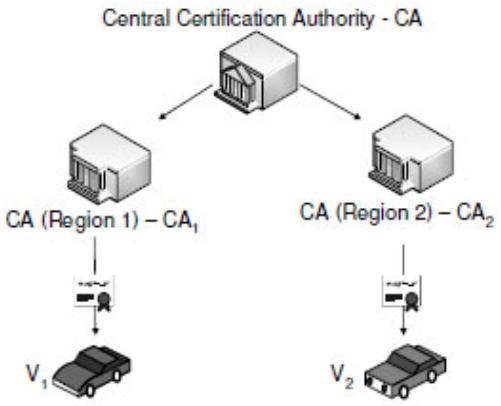

Fig. 1. A certificate authority hierarchy

Each node is registered with only one CA, and has a unique long-term identity (a pair of private and public cryptographic keys), and it is equipped with a long-term certificate. A list of node attributes and a lifetime are included in the certificate. To enable interactions between nodes from different regions, CAs may provide certificates for other CAs (cross-certification). These regional CAs are linked with each other via a top level CA through secure wire-line communication. Each node is registered with only one CA, and has a unique long-term identity (a pair of private and public cryptographic keys), and it is equipped

with a long-term certificate. A list of node attributes and a lifetime are included in the certificate. We use the following notations:

Certificate Authority	
	ID_{CA} – certificate authority ID
	$Cert_{CA}$ – CA's own certificate
	K_{CA}- CA's public key
	k_{CA}- CA's private key
Node (vehicle or RSU)	
	V – node's unique long-term identity
	k_V - node's private key
	$Cert_V$ – vehicle's certificate

Each node, vehicle or RSU, is equipped with a certificate

$$Cert_V\{ID_{CA}, S_{CA}, K_V, A_V, \tau\}, \quad (1)$$

where,

S_{CA} – CA's signature,
K_V - vehicle's public key,
A_V – node attributes,
τ – certificate lifetime.

To be able to validate the certificate signed by CA, the public key of certificate authority, K_{CA}, should be available to each entity in the system. To ensure the integrity and authenticity of a message, the sender signs the message with his private key. After receiving the message, the recipient verifies the signature using the public key of the transmitter. Formally, we can represent the mechanism as follows:

V_1: Signs the message, $S_V = \sigma_{k_V}(m)$,
with $\sigma_{k_V}()$ denoting V's signature and m the clear text message.
Forms the message, $M = \{m, S_V\}$.

$V_1 \rightarrow V_2$: M

V_2: Verifies the signature, $m = \gamma_{K_V}(S_V)$, using V_1's public key.

The CA manages long-term identities, credentials, and cryptographic keys for vehicles. The interaction of nodes with the CA does not need to be continuous. This is a major advantage for VANETs that use as gateways the road-side infrastructure or other infrastructure-based networks (e.g., mobile networks). Nonetheless, signed messages can be trivially linked to the certificate of the signing node. To make communications anonymous, it is necessary to remove from certificates all information identifying the node; rather than utilizing the same long-term public and private key for securing communications, each vehicle utilizes multiple short-term private-public key pairs and certificates. These short-term public keys are known as *pseudonyms*, that is, public keys without any

information identifying V. A pseudonym has a lifetime and an identifier of the corresponding pseudonym provider, which is in general an entity distinct from the CA. They are preloaded and renewed periodically. Normally, to prevent tracking, each vehicle contains a set of pseudonyms, $P = \{K_V^1, K_V^2, K_V^3, \ldots, K_V^n\}$. The vehicle uses each of them for a short period of time, τ, and then switches to another, not previously used pseudonym. This way, messages signed under different pseudonyms cannot be linked. For the i-th pseudonym K_V^i for node V, the CA provides a certificate $Cert_{CA}(K_V^i)$, which is simply a CA signature on the public key K_V^i. The private key k_V^i corresponding to the pseudonym K_V^i is used by the node to digitally sign messages. To enable validation, the pseudonym and certificate of the signer are attached in each message. With $\sigma_{k_V^i}()$ denoting V's signature under its i-th pseudonym, the message format is:

$$m, \sigma_{k_V^i}(m), K_V^i, Cert_{CA}(K_V^i), \qquad (2)$$

The CA maintains a map from the long-term identity of V to the $\{K_V^i\}$ set of pseudonyms provided to a node. The rate of pseudonym changes depends on the degree of protection the vehicle seeks and determines the size of the pseudonym set, n, the node should obtain.

The components of the IEEE 1609.2 security infrastructure (Fig. 2) are based on industry standards for public key cryptography, including support for elliptic curve cryptography (ECC), WAVE certificate formats, and hybrid encryption methods, in order to provide secure services for WAVE communications.

The security infrastructure is also responsible for the administrative functions necessary to support core security functions such as safe keeping the pseudonyms and certificate revocation. Note that certificate revocation is essential to any security system based on the public key infrastructure, which has not been addressed in the current IEEE 1609.2. In addition, IEEE 1609.2 does not define driver identification and privacy protection, and has left a lot of issues open. The most common way to revoke certificates is the distribution of certificate revocation lists (CRLs) that contain the most recently revoked certificates. CRLs are provided when infrastructure is available. The main disadvantage of this scheme is that the CRL may grow quickly such that it takes a long time to check through the whole CRL to see if a given certificate is valid or not [9].

Regarding physical security of information, both vehicles and RSUs have to be equipped with a tamper proof device (TPD), whose purpose is to store and physically protect sensitive information and provide a secure time base.

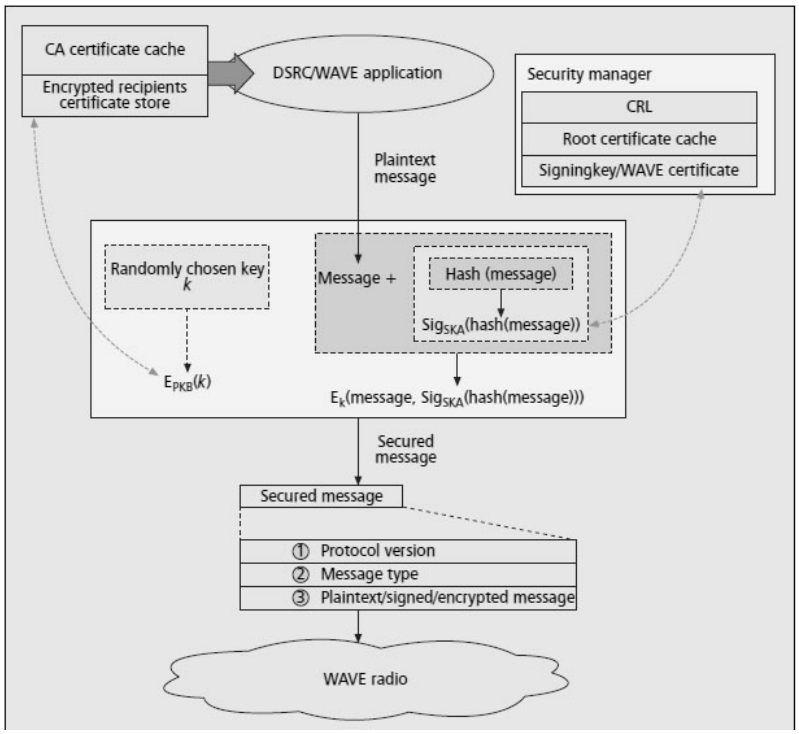

Fig. 2. The IEEE Std 1609.2 security services framework

If TPDs were to be tampered with to extract private keys, the physical protection of the unit would ensure that the sensitive information would be erased. Essentially, the TPD is the basis of trust.

5. Distributed certificate and application architecture

The basic idea is that if a user wants to participate in a VANET, he purchases a payment-processing device. Each device will have an ID and an associated certificate. A certificate will essentially be represented by its public and private key pair (K_U, k_U) and be valid inside a region R. During initialization the device will be registered with the user's account; user's information will be maintained with the provider and will not be stored in the device. The basic procedure is illustrated in Fig. 3.

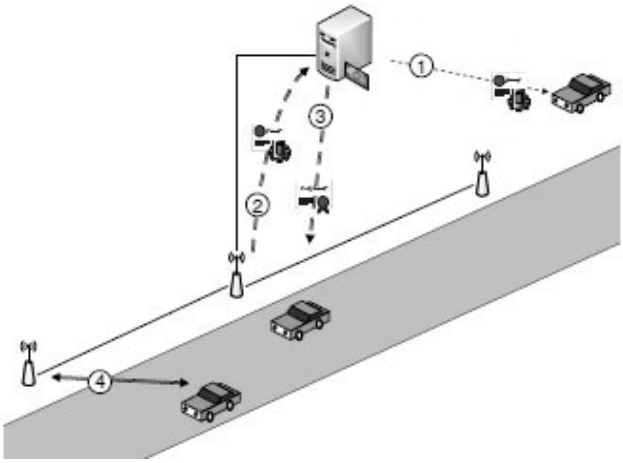

Fig. 3. Distributed certificate architecture

When a user enters a service area and wants to use the service, he makes the payment for the service using onboard payment device. The payment-authorization/service-request message will be encrypted using provider's public key, thus hiding the device ID/certificate and services requested from eavesdroppers. The user is issued a pseudonym and other IDs necessary for the service by the provider. The concerned server is also informed of the service purchased and temporary credentials. The temporary credentials can also be used to provide desired security attributes for VANET applications including vehicle to vehicle communications. As a baseline service, the user can obtain just the temporary credentials, in this case the temporary credentials will not be sent to servers. Certificate, IP address, MAC address can all be issued on temporary basis and refreshed several times during a service period. They are encrypted to ensure security and privacy. Initially, they can be encrypted using a random session key sent along the request. Later, they can be encrypted using current public key. The certificate of CA is hard coded in the device. This enables other users to check validity of a certificate. So, if a user U having a public key pair (K_U, k_U) (for initial request these are the permanent keys associated with the payment device) wants to acquire temporary credentials for the time duration defined by (t_s, t_f) and within the region R from a service provider S with a public key pair (K_S, k_S), there are following transactions:

1. U:

Generate $M = \{t_s, t_f, R, U\}$, where (t_s, t_f) are the start and finish times between which a particular pseudonym (P) will be valid.

Compute $M_R = E_{K_S}(M)$, where $E_{K_S}(M)$ defines an encryption function on message M using the public key K_S.

Compute $N_R = S_{k_U}(M)$, where $S_{k_U}(M)$ defines a signature function on message M using a private key k_U.

2. $U \rightarrow S$: M_R, N_R
3. S:

Extract M

$V_{K_U}(M, N_R)$, a signature verification function. It verifies the signature by computing it with received signature N_R, after decrypting it with the corresponding public key K_U. The service provider records device's public key during this process.

Verify ID U and associated account

Generate $P = \{t_s, t_f, R, K_P\}, k_P$, where P is the temporary certificate/pseudonym with associated private key k_P.

Compute $M_P = E_{K_U}(P, k_P)$

Compute $N_P = S_{k_S}(P)$

Compute $N_K = S_{k_S}(k_P)$

4. $S \rightarrow U$: M_P, N_P, $N_{R|}$
5. U:

Extract P and k_P

$V_{K_S}(P, N_P)$, $V_{K_S}(K_P, N_K)$

Temporary credential $\{t_s, t_f, R, k_P, K_P\}$ is valid for time duration defined by (t_s, t_f) and within region R. User uses (P, N_P) as a temporary certificate.

This mechanism can be easily extended for additional services. If other applications (web access, email, etc..) are available then these can be offered as extended services. In this case a user indicates the service which he desires to use in a service request authorization message. The payment processing provider issues the temporary credentials to the user and also forwards these credentials along with the details of service to the concerned server. The user can then initiate request to the service provider using this temporary credentials.

6. Proposed security architecture

In this section, we propose an efficient VANET key management system which meets all of the requirements presented in section 1. We assume that OBUs have inexpensive hardware while RSUs have greater computational power, and communication coverage exists to allow OBU certificate update and revocation distribution. In our system, roadways are divided into geographic regions with regional authorities (RAs) acting as certificate authorities for their region. Within a region, a RA certifies vehicles generating temporary keys which are used to authenticate vehicles. As traffic enters a region, each vehicle anonymously requests a pseudonym from the RA. If the requesting vehicle has not been

revoked, the RA responds with a pseudonym. A higher authority issues certificates to RAs, and certifies them as valid intermediary authorities. Also on the network there is an authorizing authority (AA) that deals with storing information concerning the identity and properties of nodes (users, infrastructure equipment, buses, police cars, etc..). Physically, AA is a database. In this database we differentiate the user from the car and even the owner from the user. In this respect, we make the following proposals:

- the *owner* is a vehicle attribute. In the database, besides VIN (Vehicle Identification Number) and ELP (Electronic License Plate), there is a record named "owner";
- the *user* is the person who drives the car;
- a regional authority may issue distinct CRLs for both entities, car and user (e.g., the vehicle is not complying with current technical requirements, or a revoked driving license, user inappropriate behavior on network), etc..

An anonymous key issued by a RA does not reveal any information about the identity of the vehicle/user. In our case, the identity of nodes is not certified by long-term keys generated based on their individual characteristics. All information regarding the identification of a vehicle or person is in a database. Therefore, all this information is recorded by the AA. Regional authorities only issue pseudonyms to nodes. The vehicle sends ID and validity proof encrypted using RA's public key to RA. RA sends ID and validity proof to AA. AA confirms to RA vehicle is legitimate. RA generates a pseudonym for the vehicle. The AA knows which vehicle and user was certified, but not what certs were used, RA knows what certs were issued, but not what vehicle they were issued to. Even though an attacker gains control of the RSU in a region, such an attacker is unable to track vehicles, generate certificates for other regions, etc.. The user always proves his credentials to AA before it can access pseudonyms. Each vehicle generates its own set of short-lived public/private key pair to sign and encrypt messages it sends. In this way we don't need to store or "refill" a set of pseudonyms and we eliminate the computational and transmission overhead caused by necessary key management mechanisms. The message format is:

$$M = \{K^+, V, U, P, S, t_s, t_f, L, \tau\}, \quad (3)$$

K^+ - the public key generated by OBU at initialization or by the person who uses a laptop, smartphone, tablet, etc..

V – vehicle identifier. This identifier is sent automatically by OBU. An owner/user cannot prevent the processes. If a user connects to the network with a laptop, this record is left free.

U, P – the username and password for the user. This information, along with access rights are in the AA database.

S – the service identifier. It can be a list of applications identifiers $S = \{Ap_1, Ap_2, \dots, Ap_n\}$.

t_s, t_f – are the start and finish time between which a particular pseudonym will be valid.

L – location information. Position verification in general aims at preventing malicious or defective nodes to pretend to be at arbitrary positions and triggering wrong safety messages or justifying to have ”the right” to send a valid warning message for a certain region [10].

τ - timestamp. A first step in processing procedure consists in comparing the message reception time to the message creation time stamp. This can give estimations if the message creation time is plausible or not.

If a user/vehicle wants to acquire temporary credentials, he is following the procedure:

V:

1. It generates the public and private key, K^+ and K^-
2. Type the login information to properly form the message: $M = \{K^+, V, U, P, S, t_s, t_f, L, \tau\}$.
3. Encrypt the message. $M_E = E_{K_{RA}^+}(M)$, where $E_{K_{RA}^+}$ defines an encryption function on message M using the public key K_{RA}^+.
4. Sign the message: $\sigma_{K^-}(M_E)$

$V \rightarrow RA$: $\{M_E, \sigma_{K^-}(M_E), K^+\}$.

RA:

1. $\gamma_{K^+}(\sigma_{K^-}(M_E))$. RA verifies the signature and records device’s public key. So far, the identity of the transmitter does not matter.
2. Decrypt the message with his own private key $D_{K_{RA}^-}(M_E)$ and obtain the message $M = \{K^+, V, U, P, S, t_s, t_f, L, \tau\}$.
3. Verifies L, τ and sends V, U, P to AA.
4. AA confirms the access rights.
5. Generates the pseudonym, which is simply the RA’s signature on node’s public key: $Cert_{RA}(K^+)$. It is useless to issue another key pair.

Thus, RA certificate is $Cert_{RA} = \{t_s, t_f, RA, \sigma_{K_{RA}^-}, K^+\}$, where

RA – RA's identifier,

$\sigma_{K_{RA}^-}()$ - certificate's signature

K^+ - pseudonym.

We see that the authentication procedure, which ends with the issuance of a pseudonym, requires a valid username and password in the AA database. Also, the database records the permissions required for each user to access the services available in the system. AA maps the pseudonym to user and network credentials (IP, MAC). This mapping ensures tracebility and revocation requirements. On the other hand, our mechanism will change the IP/MAC addresses at each pseudonym change. Note that we ask for a time interval. This is the validity period desired for that pseudonym. Although it may have a default value, the on board unit should enable the user to modify this information. We know that a frequent pseudonym change ensures user privacy, but if this change is happening very often the probability of tracking and detection is decreased. Moreover, some applications require a stateful communication session, meaning that the communicating parts needs to save information about the session history in order to be able to communicate. Pseudonym change every 10 ms can increase the privacy, but certainly will make the communication imposible, since no node will be able to send a packet in response to a previous one. In conclusion, there must be a balance between intimacy and operational requirement such as the performance and stability of sessions.

7. Conclusions

In this work, we presented an efficient way to fulfill the security and privacy properties necessary for key management in VANETs. Our solution addresses the scalability issues found in implementations based on IEEE 1609.2 standard and is also complete, meaning that, unlike distributed solutions, it can provide the necessary security requirements for applications common to intelligent transport systems. We have introduced a mechanism that rely on the availability of infrastructure to handle identity and credential management, and to secure communication while enhancing privacy. Although this sounds like a drawback, and when somebody needs a fully ad-hoc VANET, it is a real drawback, this method will make the network less overhead, and the communication faster, and provides an easier way to recognize the adversary vehicles. Our design approach seeks to produce a baseline architecture and solution that is both comprehensive and practical. On the one hand, our solution combines well accepted cryptographic primitives and concepts (e.g., pseudonyms)

adopted and standardized. The use of well-established security mechanisms facilitates deployment. On the other hand, our solution is adaptable, so that mechanisms or other changes aiming at higher protection levels can be introduced transparently in the future.

On-board units use short-lived keys to sign messages used for VANET communication. These short-lived keys are certified by regional authorities based on user credentials located in a database. During key updates, RAs verify that the requesting user/vehicle it has not been revoked; however, the RAs do not learn the user or vehicle's identity. This allows a valid node to acquire a certificate for a temporary key and preserve the user's privacy. Users are issued with temporary certificates which can only be used within a specific geographic area and within a particular time period. This property also simplifies the certificate revocation procedure. If a message is identified to abuse the VANET, authorities can trace the certificate request back to the signer. The authorities can further revoke the misbehaving nodes so that it is no longer able to participate in the VANET.

In our future work we would like to make simulation for this proposed method.

REFERENCES

[1] *J.T. Isaac, S. Zeadally, and J.S. Camara*,"Security attacks and solutions for vehicular ad hoc networks", in IET Communications, vol. 4, issue 7, April 2010, pp. 894-903.

[2] *A. Studer, E. Shi, F. Bai, and A. Perrig,*"TACKing Together Efficient Authentication, Revocation, and Privacy in VANETs", in 6th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks, SECON '09, June 2009.

[3] *IEEE 1609.2*,"IEEE trial-use standard for wireless access in vehicular environments - security services for applications and management messages", July 2006.

[4] *M. Gerlach., A. Festag, T. Leinmuller, G. Goldacker, and C. Harsch,*"Security architecture for vehicular communication", in WIT 2005, Hamburg, Germany.

[5] *P. Papadimitratos, L. Buttyan, J-P. Hubaux, F. Kargl, A. Kung, and M. Raya*,"Architecture for secure and private vehicular communications", in ITST '07, Sophia, Antipolis, France.

[6] *P. Papadimitratos, L. Buttyan, T. Holczer, E. Schoch, J. Freudiger, M. Raya, Z. Ma, F. Kargl, A. Kung, and J-P. Hubaux*,"Secure vehicular communication systems: design and architecture", in IEEE Wireless Communication Magazine, Nov. 2008.

[7] *P. Papadimitratos, G. Mezzour, and J-P. Hubaux*,"Certificate revocation list distribution in vehicular communication systems", in Proceedings of the fifth ACM international workshop on Vehicular Inter-Networking VANET'08, Sept. 2008.

[8] *B. Aslam, and C. Zou,*"Distributed certificate and application architecture for VANETs", in Military Communications Conference, MILCOM 2009, Oct. 2009.

[9] *X. Lin, R. Lu, C. Zhang, H. Zhu, P. Ho and X. Shen,*"Security in vehicular ad hoc networks", in IEEE Communications Magazine, Vol. 46, No. 4, 2008, pp. 88-95.

[10] *G. Yan, S. Olariu, and M. Weigle,*"Providing VANET security through active position detection", in Computer Communications, Vol. 31, No. 12, 2008, pp. 2883-2897.