Efficient and Robust Pseudonymous Authentication in VANET

Giorgio Calandriello†, Panos Papadimitratos‡, Jean-Pierre Hubaux‡
Antonio Lioy†

†Dipartimento di Automatica e Informatica
Politecnico di Torino, Italy
{giorgio.calandriello,lioy@polito.it}

‡Laboratory for Computer Communications and Applications
EPFL, Switzerland
{panos.papadimitratos, jean-pierre.hubaux@epfl.ch}

ABSTRACT

Effective and robust operations, as well as security and privacy are critical for the deployment of vehicular ad hoc networks (VANETs). Efficient and easy-to-manage security and privacy-enhancing mechanisms are essential for the wide-spread adoption of the VANET technology. In this paper, we are concerned with this problem; and in particular, how to achieve efficient and robust *pseudonym*-based authentication. We design mechanisms that reduce the security overhead for safety beaconing, and retain robustness for transportation safety, even in adverse network settings. Moreover, we show how to enhance the availability and usability of privacy-enhancing VANET mechanisms: Our proposal enables vehicle on-board units to generate their own pseudonyms, without affecting the system security.

Categories and Subject Descriptors

C.2.0 [**General**]: Security and protection; C.2.1 [**Network Architecture and Design**]: Wireless communication

General Terms

Performance, Reliability, Security

Keywords

Vehicular networks, Security, Performance, Reliability

1. INTRODUCTION

Vehicular ad hoc networks (VANETs) aim at enhancing safety and efficiency in transportation systems. They comprise network nodes, that is, vehicles and road-side infrastructure units (RSUs), equipped with on-board sensory, processing, and wireless communication modules. Vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) communication can enable a range of applications. Among these, primarily safety will be enabled, as numerous research and development initiatives indicate, by vehicles frequently *beaconing* their position, along with warnings on their condition or environment. Nonetheless, VANETs can be vulnerable to attacks and jeopardize users' privacy. For example, an attacker could inject beacons with false information, or collect vehicles' messages, track their locations, and infer sensitive user data. To thwart such attacks, security and privacy-enhancing mechanisms are necessary or, in fact, a prerequisite for deployment.

This has been recently well understood; currently, at least three concerted efforts, the IEEE 1609.2 working group [20], the NoW project [18], and the SeVeCom project [26], are developing VANET security architectures. Their common basic elements include the use of *Certification Authorities (CAs)* and public key cryptography to protect V2V and V2I messages. Message authentication, integrity, and non-repudiation, as well as protection of private user information are identified as primary requirements.

Solutions for these seemingly contradictory goals are influenced by proposals targeting the wire-line Internet. *Pseudonymity* or *pseudonymous authentication* requires that each node is equipped with multiple credentials, e.g., certified public keys that do not reveal the node identity, e.g., as those in the early [15], termed *pseudonyms*. This way, messages signed under different pseudonyms cannot be linked [29, 17, 30, 6]. Meanwhile, [12] mentions VANET as an application for *group signatures* [16, 7, 31, 13], that is, cryptographic primitives for *anonymous authentication*. This is a stronger property than pseudonymous authentication, as *any* two group signatures generated by a node *cannot* be linked. Pseudonymous authentication has already gained wide acceptance in the VANET community [20, 18, 26], while anonymous authentication incurs additional overhead, as it will become clear in the rest of the paper. This is why our investigation focuses on pseudonym-based systems.

Many technical issues and interesting questions remain to be answered in order to deploy pseudonym-based secure vehicular communications. Given that the security overhead will be significant, is it possible to reduce it without weakening security? Do vehicular communications, and in particular safety applications, remain robust? What is the effect of the security and the pseudonym-based mechanisms on safety applications? Can the cost of providing vehicles with large numbers of pseudonyms be waived and still meet security and privacy requirements?

In this paper, we provide answers to these questions. To the best of our knowledge, this is the first investigation on these aspects of the pseudonym-based approach. In Sec. 2,

VANET'07, September 10, 2007, Montréal, Québec, Canada.

we define more precisely the problem at hand and outline our approach to addressing it. The scheme components are presented in Sec. 4. We evaluate the proposed mechanisms and compare to alternatives, analyzing overhead, efficiency and robustness in Sections 5 and 6.

2. SYSTEM MODEL

V2V and V2I Communication over the wireless medium employs the *Dedicated Short Range Communications (DSRC)* data link technology [2]. Vehicles transmit periodic status updates (*beacons*) on a common channel dedicated to emergency situations, among the available seven DSRC channels. In this work, we are interested in the type of messaging that enables safety applications. Beaconing rates ρ range from 3 beacons to 10 *beacons/sec*, with the latter currently envisioned as mandatory for safety applications. We assume $\rho = 10$ in the paper. We do not dwell on the beacon content; information, for example, on the vehicle condition, position, speed, or various warnings, as well as accurate time can be obtained by on-board sensors and other hardware (e.g. GPS) [20].

We focus on vehicle-initiated transmissions, as there is no need to safeguard the privacy of infrastructure nodes. We assume that each node (vehicle) has a long-term, unique identity, that combines a number of attributes [27]. Nodes also have cryptographic keys associated with their long-term identities, managed by an authority we refer to, for simplicity, as *Certification Authority (CA)*. All legitimate nodes are registered with the CA, which can evict a node if necessary, i.e., revoke its credentials. The CA is also vested with the legal power to disclose node identities when necessary, if presented with one or more messages, for liability attribution. This is especially needed when privacy-enhancing technologies are used.

3. PROBLEM AND APPROACH OVERVIEW

We are interested in a secure vehicular communications system that provides message authentication, integrity, and non-repudiation. At the same time, cryptographically protected messages should not allow for their sender to be identified, and two or more messages generated by the same node should be difficult to link to each other. More precisely, messages produced by a node over a protocol-selectable period of time τ can be linked, but messages m_1, m_2 generated at times t_1, t_2 respectively, such that $t_2 > t_1 + \tau$, cannot. The shorter τ is, the fewer the linkable messages are, and the harder tracking a node becomes.

Intuitively, it suffices that a node switches to a new signing key, and the corresponding public key, every τ seconds. Then, only messages signed (verified) with the same secret (public) key can be linked to each other. This is essentially the idea behind pseudonym schemes, explained further in Sec. 4.1, with τ corresponding to the period one pseudonym is used. We emphasize that our goal here is not a formalization of such notions, nor to prove the protection pseudonym schemes provide against various types of adversaries.

Our objective is to design a *usable* system based on pseudonyms, which satisfies the above-stated security requirements and allows liability attribution. Equally important, we aim at a *modular design* that in the future will allow system users to upgrade mechanisms (e.g., cryptographic primitives) or adjust system parameters (e.g., security levels), so that more stringent requirements can be met.

First, we propose that each vehicle generate its own pseudonyms, in order to eliminate the need of pre-loading, storing and refilling pseudonyms and the corresponding private keys. This way, the burden of key and pseudonym (and essentially identity) management is greatly reduced; and so is the cost of obtaining the pseudonyms over an "off-line" channel (e.g., localized, wired or wireless, connection to specialized infrastructure, or 3G cellular link downloads). Moreover, the usability and efficiency of the system is enhanced: (i) vehicles do not need to be side-lined or to compromise their user's privacy if previously unused pseudonyms are no longer available, (ii) no "over-provisioning" in the supply of pseudonyms is necessary.

However, self-generation of pseudonyms comes, at first glance, at a higher transmission and processing cost. We propose a number of optimizations to reduce overhead, which is significant as pseudonym-based approaches mandate that in general pseudonyms are attached to the transmitted messages [29, 30, 6]. Without compromising security (i.e., all messages are cryptographically validated), we integrate in our system methods for pseudonym dissemination, validation, and generation, which significantly reduce overhead. Moreover, we identify a new limitation due to pseudonym changes: their impact on safety applications. We propose a scheme to mitigate it and, more generally, investigate robustness in adverse wireless networking settings, considering emergency braking as a case study.

4. SCHEME OUTLINE

In this section, we describe our proposed scheme that relies on the concept of pseudonymous authentication, which we term *Baseline Pseudonym (BP)* (Sec. 4.1). Our scheme allows the on-the-fly generation of own pseudonyms using *Group Signatures (GS)* (Sec. 4.2), in combination with the BP approach; this is what we term as *Hybrid* scheme (Sec. 4.3). Optimizations for efficiency and robustness are presented in Sec. 4.4.

4.1 Baseline Pseudonyms

Each node (vehicle) V is equipped with a set of *pseudonyms*, that is, *public keys* certified by the CA without any information identifying V. 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 message 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 and m the signed message payload, the message format is:

$$m, \sigma_{k_V^i}(m), K_V^i, Cert_{CA}(K_V^i) \quad (1)$$

The CA maintains a map from the long-term identity of V to the $\{K_V^i\}$ set of pseudonyms provided to a node. If presented with a Msg.(1), the CA can perform the inverse mapping and identify the signer.

Each pseudonym is used at most for a period τ and then discarded. We abstract away a number of implementation details one could consider, such as the dynamic adaptation of the period of pseudonym usage, other policies for pseudonym change, factors rendering a pseudonym change unnecessary

(e.g., a TCP connection to an access point), and interactions of pseudonym changes with the network stack [26]. They are important yet largely orthogonal to this investigation. Nevertheless, the extent of a pre-load or the frequency of refills with a set of K_V^i and the corresponding k_V^i, $Cert_{CA}(K_V^i)$, can affect the usability of the system.

Upon receipt of Msg.1, a node, with the public key of the CA assumed available, validates $Cert_{CA}(K_V^i)$. It makes use of a *Certificate Revocation List* (CRL), also assumed to be distributed to vehicles via the infrastructure [30]. If successful, i.e., K_V^i is not included in the CRL and CA signature on K_V^i is valid, the node validates $\sigma_{k_V^i}(m)$.

4.2 Group Signatures

Each node V is equipped with a secret *group signing key* gsk_V, with the *group* comprising as members all vehicles registered with the CA. A *group public key* gpk_{CA} allows for the validation (by any node) of any *group signature* $\Sigma_{CA,V}$ generated by a group member. Intuitively, a group signature scheme allows any node V to sign a message on behalf of the group *without* V's identity being revealed to the signature verifier. Moreover, it is impossible to link any two signatures of a legitimate group member. Note that no public key or other credentials need to be attached to a message anonymously authenticated; the format is:

$$m, \Sigma_{CA,V}(m) \tag{2}$$

The concept of group signatures, introduced by Chaum [16], is revisited in numerous works, e.g., [7, 31, 13, 11], with formal definitions in [8, 9]. For the rest of the discussion, we assume and utilize the group signature scheme proposed in [12]. If the identification of a signer is necessary, the CA can perform an *Open* operation [8, 9] and reveal the signer's identity. A *Revocation List (RL)* can also be constructed, so that a signature by a revoked group member can be identified by any verifier. Again, we assume the revocation method proposed in [12].

4.3 Hybrid Scheme

The combination of the pseudonym scheme with the GS scheme is the basic element of our proposed scheme. Each node V is equipped with a group signing key gsk_V and the group public key gpk_{CA}. Rather than generating group signatures to protect messages, a node generates an own set of pseudonyms $\{K_V^i\}$ (and corresponding private k_V^i keys), and uses gsk_V to generate a group signature $\Sigma_{CA,V}()$ on each pseudonym K_V^i.

Essentially, the nodes generate and "self-certify" K_V^i on-the-fly, producing $Cert_{CA}^H(K_V^i)$. The H superscript denotes the hybrid scheme and differentiates this certificate from that of the BP approach. The CA subscript denotes that the certificate was generated by a legitimate node registered with CA. Similarly to Msg.(1), V attaches the $Cert_{CA}^H(K_V^i)$ to each message, and signs with the corresponding k_V^i:

$$m, \sigma_{k_V^i}(m), K_V^i, Cert_{CA}^H(K_V^i) \tag{3}$$

Upon receipt of a Msg.(3), the group signature $\Sigma_{CA,V}(K_V^i)$ is validated, using the gpk_{CA} and the RL. If successful, the receiver infers that a legitimate system (group) member generated pseudonym K_V^i. We emphasize that, as per the properties of group signatures, the receiver/verifier of the certificate *cannot* identify V and *cannot* link this certificate and pseudonym to any prior pseudonym used by V. Once the legitimacy of the pseudonym is established, the validation of $\sigma_{k_V^i}(m)$ is identical to that for Msg.(1). To identify the signer of message, an *Open* on the $Cert_{CA}^H(K_V^i)$ group signature is necessary; the message m is bound to K_V^i via $\sigma_{k_V^i}(m)$, and K_V^i is bound to V via $\Sigma_{CA,V}(K_V^i)$.

4.4 Optimizations

We identify optimizations to reduce overhead (Optimizations 1 and 2) and enhance robustness (Optimization 3). All three are applicable for both BP and the Hybrid scheme.

Optimization 1. At the sender side, the $Cert_{CA}^H(K_V^i)$ is computed only once per K_V^i, because $Cert_{CA}^H(K_V^i)$ remains unchanged throughout the pseudonym lifetime τ. For the same reason, at the verifier's side the $Cert_{CA}^H(K_V^i)$ is validated upon the first reception and stored, even though the sender appends it to multiple (all) messages. For all subsequent receptions, if the $Cert_{CA}^H(K_V^i)$ has already been seen, the verifier skips its validation. This optimization is useful because $\tau >> \rho^{-1}$.

Optimization 2. The sender appends $\sigma_{k_V^i}(m), K_V^i, Cert_{CA}^H(K_V^i)$ once every α messages (beacons); it appends only $\sigma_{k_V^i}(m)$ on the remaining $\alpha-1$ ones. We call α the *Certificate period*. Msg.(1) and (3) remain unchanged, and all messages will carry a 4-byte *keyID* field, that is, a random number indicating which K_V^i must be used to validate the $\sigma_{k_V^i}(m)$. The *keyID* does not affect privacy as there is a 1:1 correspondence with K_V^i. When a pseudonym change occurs, the new triplet $\sigma_{k_V^{i+1}}(m), K_V^{i+1}, Cert_{CA}^H(K_V^{i+1})$ must be computed and transmitted. V will sign messages with the new k_V^{i+1} corresponding to K_V^{i+1} from then on.

Optimization 2 can harm protocol robustness, if the message carrying $K_V^{i+1}, Cert_{CA}^H(K_V^{i+1})$ is not received. Then, nodes in range of V must wait for α messages for the next pseudonym/certifcate transmission, while being unable to validate *any* message from V. The danger arises especially if the vehicles are close to each other and/or move at high relative speeds. Thus, we propose the following scheme to mitigate this problem:

Optimization 3. The transmission of $K_V^{i+1}, Cert_{CA}^H(K_V^{i+1})$ is repeated for p consecutive messages when the K_V^{i+1} is issued. We denote p at the *Push period*.

5. CRYPTOGRAPHIC OVERHEAD

We choose to employ ECDSA as the basic signature algorithm [4], the group signature (GS) proposed by Boneh and Shacham [12], and security level of t=80 bits for message signatures and t=128 bits for CA certificates in BP and for GS. The rationale is that high security is not necessary for the short-lived K_V^i, although it is for the long-term GS keys. Table 1 shows the costs for signature, verification and overhead for the chosen algorithms.

As implementations of group signatures and the chosen GS were not available to us, we calculated the number of 32-bit word multiplications required for GS signing and verifying, extracting the relevant data from [14] and [22]. To obtain a processing delay estimate, we ran benchmarks for the

Algorithm	Security level (bits)	Sign (s)	Verify (s)	Signature (bytes)	Public key (bytes)	Private key (bytes)
ECDSA-192	80	5e-4	3e-3	48	25	24
ECDSA-256	128	8e-4	4.2e-3	64	33	32
GS	80	1.78e-2	1.56e-2	151	278	43
GS	128	5.37e-2	4.93e-2	225	800	64

Table 1: Computation costs and communication overhead for different signing algorithms.

multiplications. For ECDSA we also measured the OpenSSL [3] delays. All measurements were made on a Centrino machine with the clock speed set at 1.5 GHz; this is a close approximation for the CVIS vehicle PC, a platform adopted for the future development of VANET applications [1].

5.1 Computation

For cryptographic primitives, the individual delays are shown for each operation in Table 1. For individual message cryptographic processing delays, it suffices to add the corresponding cryptographic primitives delays as explained below. We recall that the security levels are $t = 80$ for $\sigma_{k_V^i}(m)$ and $t = 128$ for $Cert_{CA}(K_V^i)$ and $\Sigma_{CA,V}(m)$.

For BP, a sender computes a $\sigma_{k_V^i}(m)$ for each message, and each receiver will validate one $\sigma_{k_V^i}(m)$ per message and one $Cert_{CA}(K_V^i)$ for each pseudonym. Thus, the costs per message will be: 5e-4 s/msg for signing, and 7.2e-3 ms/msg for verification.

For GS, each vehicle will need to generate and verify one $\Sigma_{CA,V}(m)$ per message, either transmitted or received. This costs around 5.37e-2 s/msg for signing and 4.93e-2 s/msg for verification.

For Hybrid, the cost is one $\sigma_{k_V^i}(m)$ generation and verification per message and one $Cert_{CA}^H(K_V^i)$ generation and verification per pseudonym. The costs are 5.42e-2 s/msg for signing, and 5.23e-2 s/msg for verification.

The costs so far are do not include any optimization. If we consider Optimization 1, we normalize the processing delays over one pseudonym lifetime τ; the cost of certification is amortized over τ and the cost per message becomes the one shown in Table 2. We elect, here without loss of generality, an indicative value $\tau = 60$ sec, and we recall the message rate of $\rho = 10$ messages per second. When Optimization 2 is in place, the costs of the shorter messages simply sum up to one ECDSA signature.

5.2 Overhead

The K_V^i and the $Cert_{CA}(K_V^i)$ are 89 bytes for BP, and with $\sigma_{k_V^i}(m)$ the overhead is 137 bytes per message. For GS, the overhead is $\Sigma_{CA,V}(m)$, thus 225 bytes per message. For Hybrid, the overhead is $\sigma_{k_V^i}(m), K_V^i, Cert_{CA}^H(K_V^i)$, in total 298 bytes per message.

These figures do not consider Optimization 2: its effect on the overhead will be presented in Sec. 6.

For the $\alpha - 1$ messages transmitted without attaching the pseudonym and the certificate, the overhead is $\sigma_{k_V^i}(m)$ plus a randomly generated by the signer *keyID*, to indicate which pseudonym this message is generated under and facilitates verification; the overhead is 52 bytes.

Scheme	Sign (s)	Verify (s)	Overhead (bytes)
BP	5e-4	3e-3	137
GS	5.37e-2	4.93e-2	225
Hybrid	6e-4	3.1e-3	298

Table 2: Processing costs, in seconds, for different signing schemes, over $\tau = 60$ sec, for Optimization 1, and packet overhead, in bytes, without optimization.

5.3 Storage and Generation costs

For BP, all the needed K_V^i have to be loaded to the OBU at the refilling session. We must over-provision the vehicles in order to account even for the heaviest car usage until the next refilling session e.g. 8-10 hours of car use per day, every day. This approximates to $i = 200.000$ assuming a refilling period of one year. Each k_V^i, K_V^i pair is 113 bytes, given by one ECDSA-192 key pair and one CA certificate (e.g. one ECDSA-256 signature). These figures account for a storage of 22 MBytes.

For GS, each node V needs only to store its gsk_V and the gpk_{CA}. Similarly, for the Hybrid Scheme, pseudonyms will be generated when needed on the fly. A generation is more or less equivalent to a signature for ECDSA.

Finally, the approaches based on GS require a periodic refreshing and storage of gsk_V and gpk_{CA} (but not constrained by the possibility of running out of keys).

5.4 Revocation

To compare the costs of these different approaches, we normalize the cost for checking a given pseudonym or group signature against one entry of the (certificate) revocation list. We note that for BP, the size of CRL is much higher than the size of RL for GS and Hybrid as it includes revoked pseudonyms, not nodes. We assume the numbers for the system configuration as in Sec. 5.3.

For BP, each K_V^i must be checked against a CRL through a number of string comparisons linear to the CRL size. For GS, the revocation check, performed for each $\Sigma_{CA,V}(m)$, involves two pairing calculations followed by a string comparison. The RL is linear in the number of revoked group members. For Hybrid, revocation requires two pairing calculations, per K_V^i (similar to BP); the RL is again linear in the number of revoked group members.

We assume that a string comparison takes 1e-6 sec and a pairing calculation 5e-3 sec (as per the benchmarks above). If *RLCheck* is the cost of comparing one entry in the RL for GS, the cost for the same operation for BP is 1e-4* *RLCheck*.

Scheme	Sign K_V^i	Verify K_V^i	Sign m	Verify m	Overhead (bytes)
Hybrid	38752	33994	793	3394	224
Zeng [37]	10536	20056	4758	2379	171

Table 3: Costs associated to different signing schemes, in number of multiplications, for $t = 80$ bits, without Optimization 2.

Then, V's revocation for BP implies including in the CRL all the not yet used pseudonyms, thus the cost is in the order of $1e5$ per node. This cost depends on the data structure and search algorithm used to handle the revocation data in memory. If we assume the use of a hash map, the cost is essentially an O(1) search time (in this case, a prior hash map construction is needed with a cost of O(n)). The cost would then be 1e-4 * RLCheck.

For GS, this will take$\tau * \rho * RLCheck = 600 * RLCheck$, while for Hybrid it will be simply RLCheck. This implies that BP is the least costly solution in this aspect, not considering memory requirements to handle a (possibly) large RL.

Revoked vehicles could mount a DoS attack by forcing other parties to verify not valid signatures. We observe that this is not worse than jamming the whole spectrum. While this attack is surely possible it does not open new vulnerabilities.

5.5 An alternate approach

It is worthwhile to compare our approach with the one proposed in [37] and [6]. These data are summarized in Table 3. Zeng's approach is faster and bears less overhead, however we do not choose it in our system because it does not satisfy the modularity requirement. Also, choosing $t = 128$ bits implies choosing the same high security level message signing.

5.6 Pseudonym lifetime

As τ gets smaller, a vehicle becomes less traceable, yet at a performance premium, in particular due to signature verifications. We vary τ from 60 down to 3 seconds, and recalculate the signing and verification costs for Hybrid: they range from the above given values up to 4.6e-3 and 2.3e-3 s/msg respectively. These costs are still low, yet in a dense neighborhood the cost per node can be high. In fact, time spent on signatures validation would then be significant with respect to τ. For example, for 100 vehicles within range, each beaconing at 10 beacons per second, the total verification time for a given V would be $2.3e$-1 sec, i.e., 7.7% of the total time. This is more clearly a problem, as we consider here only the safety messaging and not other application traffic. Finally, recall that τ also impacts the size of (C)RLs and revocation performance.

6. OPTIMIZATION EVALUATION

6.1 Simulation Setup

We design a simulator in C and consider the study in [33], which shows that given a realistic radio propagation model [25] and a situation of heavy traffic the reception probability is at best 60%. This corresponds to a setup of nodes transmitting 500-byte long (including security overhead) beacons at $\rho = 10$ msg/sec, with a nominal transmission range $R = 200$ meters [35, 29, 33, 2]. We model the channel as a linearization of the packet reception probability function in [33]:

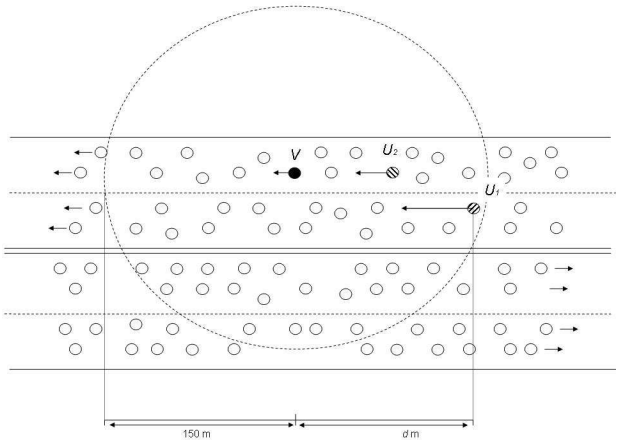

Figure 1: Simulation setup.

$$P(d) = -0.004d + 0.6 \quad (4)$$

where d is the distance between the receiver and the transmitter, and P the reception probability.

In the simulations, a node V is chosen as transmitter and a node U as receiver (U_1 in Fig. 1), positioned at an initial distance of d meters (U always behind V). U moves with a constant relative speed v with respect to V. The simulation ends when U is ahead out of range ($P() = 0$) of V. Results for each simulation scenario are averaged over 5000 randomly seeded iterations.

To understand the system dependency from the channel quality, we consider two additional probability reception functions:

$$P(d) = -0.006d + 1 \quad (5)$$

and

$$P(d) = -0.004d + \frac{1}{7} * sin(\frac{\pi}{125}d) + 1 \quad (6)$$

Eq.5 is a linearization of the one in [32], corresponding to a static scenario with $\rho = 4$ msg/sec and $R = 200$m. Eq.6 is artificially generated to represent favorable channel conditions, following the trends in [33, 32], taking $P(0) = 1$ and $P < 0.1$ for d=200m. All three equations are plotted in Fig. 2.

The first metric we are interested in is the (mean) distance at which U receives $K_V^i, Cert_{CA}^H(K_V^i)$, after V switched to the i-th pseudonym K_V^i. This is important: the higher the distance from V at which $\sigma_{k_V^i}(m)$ can be validated, the more time the driver of U has at its disposal to react to any detrimental situation. The second metric of interest is the security overhead with as a function of α.

6.2 Overhead

We calculate overhead as the load offered due to security, summing the overhead for all messages transmitted per node over one certificate period α, divided by α. Thus, the overhead of Msg.1 and 3 is amortized over the entire period α.

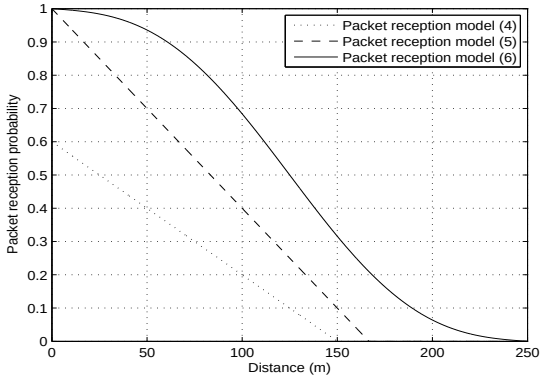

Figure 2: Probability reception rates.

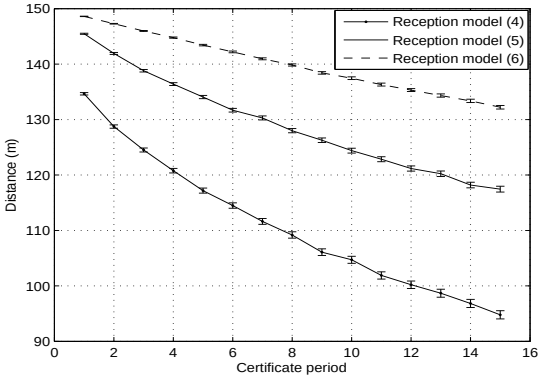

Figure 4: Reception of first certificate for $d = 150$ m, $v = 20$ Km/h.

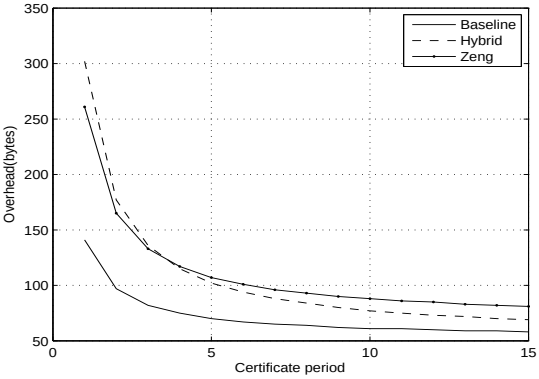

Figure 3: Overhead vs. Certificate period.

Fig. 3 shows the resultant overhead when Optimization 2 is applied. Clearly, without the optimization the overhead is high, but then decreases rapidly as α increases.

The minimum overhead is achieved for the maximum $\alpha = 15$ used here, resulting in 58, 69, and 81 bytes for the BP, Hybrid, and Zeng approaches respectively. We observe that Zeng's approach, by using high security even for the short-lived pseudonyms, performs worse than Hybrid, thus incurring 17% higher overhead at $\alpha = 15$ messages.

We also observe that for all approaches the overhead can be reduced down to a point of negligible marginal utility. If we consider this to be the point at which any subsequent increase in α will reduce overhead only by one extra byte, this point will be for BP at $\alpha = 9$ (with overhead of 62 bytes), for Hybrid at $\alpha = 13$ (overhead 72 bytes) and for Zeng at $\alpha = 12$ (overhead 85 bytes).

6.3 Basic Results

Next, we evaluate how α affects the distance at which U receives the first $K_V^i, Cert_{CA}^H(K_V^i)$, with $v = 20$ Km/h and $d = 150$ meters. We repeat the simulation for all the three probability reception functions, with results in Fig. 4.

As expected, U receives the first $K_V^i, Cert_{CA}^H(K_V^i)$ closer to V when α is higher; the confidence intervals also widen. The reason is that if a $K_V^i, Cert_{CA}^H(K_V^i)$ is missed, for high α will have approached V at the next $K_V^i, Cert_{CA}^H(K_V^i)$ transmission. This remains true for all simulated scenarios. For channel Eq.(4), at $\alpha = 15$ the $K_V^i, Cert_{CA}^H(K_V^i)$ is received on the average at 95 meters, and for more favorable channel conditions improves to 117 or 132 meters. Regarding the standard deviation of the reception distance: it increases with α. It ranges from 1.07 to 3.58 to 7.54 for $\alpha = 1$, and from 11.41 to 18.89 to 26.73 for $\alpha = 15$, depending on the channel model. Clearly, for favorable channels standard deviation is lower; e.g., for Eq.(6) and $\alpha = 15$ results are comparable to the ones for Eq. 4 and $\alpha = 1$.

6.4 Impact of pseudonym change on safety

A change of K_V^i could be dangerous for vehicles close to V. We investigate this next, taking into account (low) initial distances (at the time of pseudonym change) and (high) relative speeds between the V and other vehicles.

Impact of initial vehicle distance.

We vary d between V and U (e.g., U_2 in Fig. 1), and experiment with d set to 10, 30 and 50 meters. The results are shown in Fig. 5. he three curves have similar trends, with a slight difference at closer distances, due to the fact that P is higher for lower d; note that curves are not parallel.

Impact of relative speed.

The certificate dissemination also varies depending on the relative speed between V and U. Thus we fix $d = 30$ meters and set v =10, 20 and 50 Km/h. The results are in Fig. 6. Again, the higher v the worse the results.

By comparing Figures 5 and 6 we observe that increasing v from 20 to 50 Km/h is equivalent to having $d = 10$ meters rather than 30 for $\alpha = 15$ messages. More important, we see that for both figures we have negative values. This means that the pseudonym and certificate are received *after* U passed ahead of V, or collided with it, if they were in the same lane. This make clear the issue that careful design of optimizations is need. Otherwise, efficiency can compromise robustness.

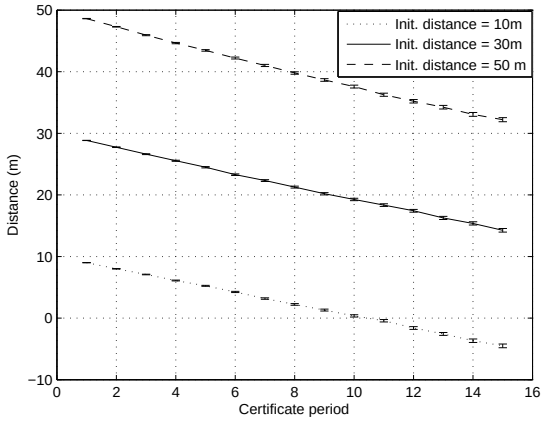

Figure 5: Reception of first certificate for reception function 4, $v = 20$ Km/h.

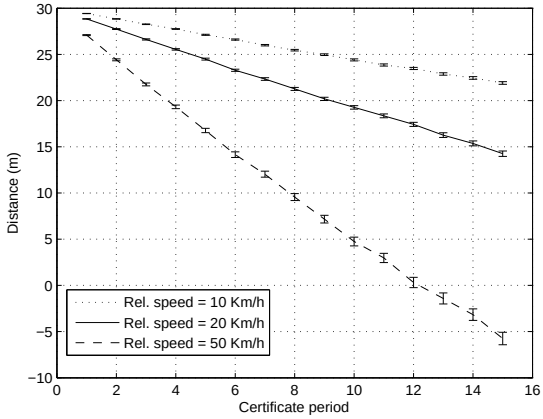

Figure 6: Reception of first certificate for reception function 4, $d = 30$ m.

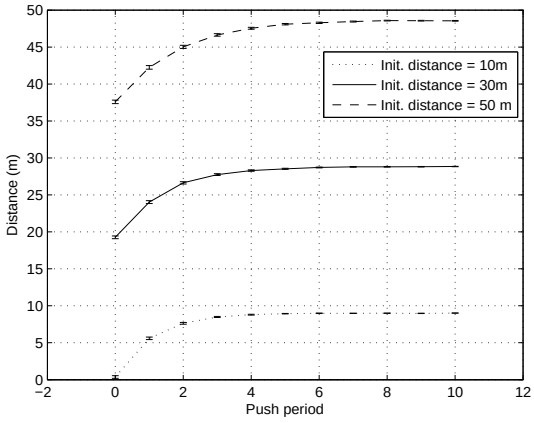

Figure 7: Reception of first certificate for reception function 4, $v = 20$ Km/h, $\alpha = 10$ messages.

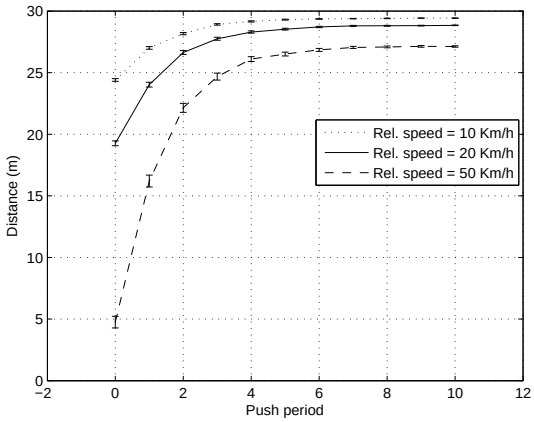

Figure 8: Reception of first certificate for reception function 4, $d = 30$ m, $\alpha = 10$ messages.

6.5 Enhancing robustness

We show here how Optimization 3 can address this problem, with small distances or high relative speeds between V and U when switching to a new K_V^{i+1} We fix $\alpha = 10$ messages (this is a value closer to the point of negligible marginal utility with respect to the overhead). We vary the *push period* p from 0 (no pushing) to 10 messages. The results are in Fig. 7 and Fig. 8.

Optimization 3 is effective: From the fourth pushed message it enables reception within 5 meters after the pseudonym change, regardless of speed and initial distance. Clearly, the actual reception distance depends on those parameters. We also observe that increasing p, thus redundancy, does not improve robustness further, e.g., for p beyond 4 to 6; this is a guideline to balance robustness with efficiency.

6.6 Case study: emergency braking

To understand the impact on safety applications, we consider V making an emergency brake. U has to brake as well in order to avoid the collision, but it can do so after it receives $K_V^i, Cert_{CA}^H(K_V^i)$ and its driver becomes aware of the alert (some time *after* receiving it). As this case study serves as a proof of concept only, we consider here a simple scenario without other vehicles in between U and V.

We simulate in a custom simulator the scenario with V moving at 50, 65 and 80 Km/h depending on the channel model employed, 4, 5, 6 respectively . We chose different speeds as the channel models are an abstraction of channel saturation, which depends on the vehicle density, and vehicles move slower when their density is higher. U's speed is set 20 Km/h higher than V's, d to 150 meters, the reaction time of U is chosen randomly between 0.75 and 1.5 s, both vehicles' length is 4 m, deceleration is -6 m/s^2. α ranges from 1 to 10, to show the impact of Optimization 2 on the system, and there are no pseudonym changes during the simulation. We repeated the simulation 10000 times per scenario and calculated the percentage of crashes. U becomes warned after it successfully receives $K_V^i, Cert_{CA}^H(K_V^i)$ and the emergency message.

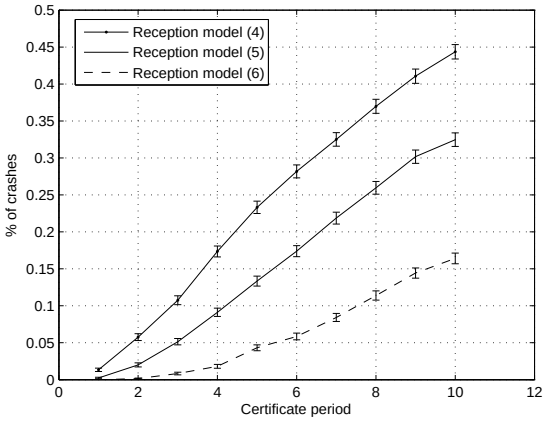

Figure 9: Percentage of crashes for the emergency braking scenario.

Reception function	Push	no. of crashes
4	yes	57
4	no	61
5	yes	47
5	no	57
6	yes	40
6	no	75

Table 4: Impact of pushing certificates in the emergency braking scenario, $d = 75$ m.

The results, reported in Fig. 9 show that the certificate period affects the number of crashes. This ranges from almost no crashes for all reception models to a significant value when Optimization 2 is in use. The crash occurrence percentages are not negligible. However, we make the following observations: (i) we consider only vehicles coming to a full stop but in many cases a simple slowdown would be sufficient to avoid the danger, (ii) we do not consider the actual road conditions that correspond to the given, adverse channel model; when the road is congested, drivers tend to adjust their speed to their neighbors, thus avoiding or significantly reducing the exagerated here danger, (iii) the application model we employ could be improved, for example, by buffering emergency messages originating from not yet validated vehicles, in order to speed up the provision of the alert to the driver once the message is validated. Note that switching to a lower α could in general add to the channel saturation, thus not always leading to an improvement [34]. Optimization 3 is of little use at the experiments with $d = 150$ meters, because at high distances the reception probability is close to zero and thus certificate retransmissions are not effective. Optimization 3 becomes effective when d=75 meters: in this case, the number of crashes is reduced, as shown in Table 4, where the pushing is fixed to p=10 messages.

7. RELATED WORK

The idea of pseudonym self-generation for ubiquitous computing is proposed, independent of our work in [37], and more recently [6] mentioned that the cryptosystem in [37] can be applied to VANET. These works do not consider all the system-level issues we consider in this work, such as certificate distribution and application robustness. Our findings and mechanisms apply to their work, as our results show, essentially complementing and extending it.

The use of pseudonyms was first envisioned by [15], and their use is explored by more recent works, e.g., [29, 17, 18, 26]. A number of recent works are concerned with different aspects of security and privacy of vehicular networks, either outlining challenges [36], [28], describing particular attacks [21], [10] or more general attack overviews [5], offering general suggestions towards solutions [17], [30], proposing mechanisms [19], [29], [24], [20], or trying to offer a general system overview [27]. The work of [23] combines public and symmetric key cryptography to authenticate messages, and it is complementary to our work. [33] and [32] provide extensive analysis of VANET channel conditions and modeling, and [34] provides a distributed algorithm to fairly share the available bandwidth. Due to space limitations, we refrain from discussing related work in further detail and refer our readers to references within the cited works.

8. CONCLUSIONS AND FUTURE WORK

In this paper, we propose a new approach to managing credentials in VANETs. With the BP approach as reference, we propose on-board, vehicle, on-the-fly pseudonym generation and self-certification. This alleviates one of the most significant limitations of the pseudonym-based approach: the need for complex management. To achieve this, we propose the use of group signatures, in order to ensure that legitimate nodes can anonymously and in a liable manner generate their pseudonyms.

We analyze the proposed scheme in terms of computational and overhead costs, and we show that it maintains its cost closely comparable to that of BP. This is due to the optimizations we propose, applicable to our scheme as well as to others. We also investigate robustness for safety messaging. We identify a new critical weakness of the pseudonym-based approach: the impact of pseudonym changes on the safety applications; and we show how to address it.

Overall, we contribute a detailed investigation of a range of system issues, and we show how pseudonym-based authentication can be applied in practice in VANETs, establishing it as a viable approach.

For future work, we intend to study how to dynamically adapt the certificate period α to the system conditions. Simulations capturing actual traffic and vehicle movement models can validate findings in that direction, as well as further investigations on robustness. Finally, we will investigate aspects of revocation that are beyond the scope of this paper, such as system deployment, CRL distribution, and existence of multiple CAs.

Acknowledgements

We would like to thank Marco Aime and Emanuele Cesena (Politecnico di Torino) for the discussions in the initial phase of the work.

9. REFERENCES

[1] The CVIS project, http://www.cvisproject.org/.

[2] DSRC: Dedicated short range communications. http://grouper.ieee.org/groups/scc32/dsrc/index.html.

[3] Openssl. http://www.openssl.org.

[4] IEEE 1363a 2004. IEEE standard specifications for public-key cryptography- amendment 1: Additional techniques, 2004.

[5] A. Aijaz, B. Bochow, F. Dötzer, A. Festag, Matthias Gerlach, Rainer Kroh, and Tim Leinmüller. Attacks on intervehicle communication systems - an analysis. In *3nd International Workshop on Intelligent Transportation*.

[6] F. Armknecht, A. Festag, D. Westhoff, and K. Zeng. Cross-layer privacy enhancement and non-repudiation in vehicular communication. In *WMAN 2007, Bern*, March 2007.

[7] G. Ateniese and G. Tsudik. Group signatures à la carte. In *SODA '99: Proceedings of the tenth annual ACM-SIAM symposium on Discrete algorithms*, pages 848–849, Philadelphia, PA, USA, 1999. Society for Industrial and Applied Mathematics.

[8] M. Bellare, D. Micciancio, and B. Warinschi. Foundations of group signatures: formal definition, simplified requirements and a construction based on trapdoor permutations. In Eli Biham, editor, *Advances in cryptology - EUROCRYPT 2003*, volume 2656 of *Lecture Notes in Computer Science*, pages 614–629, Warsaw, Poland, May 2003. Springer-Verlag.

[9] M. Bellare, H. Shi, and C. Zhang. Foundations of group signatures: The case of dynamic groups. In *CT-RSA'05, Lecture Notes in Computer Science, Springer-Verlag*, 2004.

[10] J. Blum and A. Eskandarian. The threat of intelligent collisions. *IT Professional*, Vol.6:24– 29, 2004.

[11] D. Boneh, X. Boyen, and H. Shacham. Short group signatures, 2004.

[12] D. Boneh and H. Shacham. Group signatures with verifier-local revocation. In *CCS '04*, pages 168–177, New York, NY, USA, 2004. ACM Press.

[13] E. Brickell, J. Camenisch, and L. Chen. Direct anonymous attestation. In *CCS '04*, pages 132–145, New York, NY, USA, 2004. ACM Press.

[14] M. Brown, D. Hankerson, J. Lopez, and A. Menezes. Software implementation of the nist elliptic curves over prime fields. In *CT-RSA 2001*, pages 250–265, London, UK. Springer-Verlag.

[15] D. Chaum. Security without identification: Transaction systems to make big brother obsolete. *Commun. ACM*, 28(10):1030–1044, 1985.

[16] D. Chaum and E. van Heyst. Group signatures. In *EUROCRYPT*, pages 257–265, 1991.

[17] M. Gerlach. VaneSe - an approach to vanet security. In *V2VCOM 2005, San Diego, California, USA*, 2005.

[18] M. Gerlach, A. Festag, T. Leinmüller, G. Goldacker, and C. Harsch. Security architecture for vehicular communication. In *WIT 2005*, Hamburg, Germany.

[19] P. Golle, D. Greene, and J. Staddon. Detecting and correcting malicious data in vanets. In *VANET '04*, pages 29–37, New York, NY, USA, 2004. ACM Press.

[20] IEEE1609.2. IEEE trial-use standard for wireless access in vehicular environments - security services for applications and management messages, July 2006.

[21] M. Jakobsson, X. Wang, and S. Wetzel. Stealth attacks in vehicular technologies. In *VTC-Fall 2004)*.

[22] N. Koblitz and A. Menezes. Pairing-based cryptography at high security levels. Cryptology ePrint Archive, Report 2005/076, 2005.

[23] K. Laberteaux and Y.-C.Hu. Strong vanet security on a budget. In *ESCAR 2006*.

[24] M. Li, R. Poovendran, K. Sampigethaya, and L. Huang. Caravan: Providing location privacy for vanet. In *ESCAR 2005*.

[25] M. Nakagami. The m-distribution, a general formula of intensity distribution of the rapid fading. In W. G. Hoffman, editor, *Statistical methods in radio wave propagation*, pages 3–36, Oxford, 1960. Pergamon.

[26] 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.

[27] P. Papadimitratos, V. Gligor, and J.-P. Hubaux. Securing Vehicular Communications - Assumptions, Requirements, and Principles. In *ESCAR 2006*.

[28] B. Parno and A.Perrig. Challenges in securing vehicular networks. In *HotNets-IV*, 2005.

[29] M. Raya and J.-P. Hubaux. The security of vehicular ad hoc networks. In *SASN '05*, pages 11–21, New York, NY, USA, 2005. ACM Press.

[30] M. Raya, P. Papadimitratos, and J.-P. Hubaux. Securing vehicular communications. IEEE Wireless Communications magazine, Volume 13, Issue 5, October 2006.

[31] P. F. Syverson and S. G. Stubblebine. Group principals and the formalization of anonymity. In *World Congress on Formal Methods*, pages 814–833, 1999.

[32] M. Torrent-Moreno, S. Corroy, F. Schmidt-Eisenlohr, and H. Hartenstein. IEEE 802.11-based one-hop broadcast communications: understanding transmission success and failure under different radio propagation environments. In *MSWiM '06*, pages 68–77, New York, NY, USA, 2006. ACM Press.

[33] M. Torrent-Moreno, D. Jiang, and H. Hartenstein. Broadcast reception rates and effects of priority access in 802.11-based vehicular ad-hoc networks. In *VANET '04*, pages 10–18, New York, NY, USA, 2004. ACM Press.

[34] M. Torrent-Moreno, P. Santi, and H. Hartenstein. Distributed fair transmit power adjustment for vehicular ad hoc networks. In *SECON '06*, pages 479–488, 2006.

[35] Q. Xu, T. Mak, J. Ko, and R. Sengupta. Vehicle-to-vehicle safety messaging in DSRC. In *VANET '04*, pages 19–28, New York, NY, USA, 2004. ACM Press.

[36] M. El Zarki, S. Mehrotra, G. Tsudik, and N. Venkatasubramanian. Security issues in a future vehicular network.

[37] K. Zeng. Pseudonymous pki for ubiquitous computing. In *EuroPKI*, pages 207–222, Turin, Italy, June 2006.